

ISSN 2007-5448



# RECIBE

Revista electrónica  
DE COMPUTACIÓN, INFORMÁTICA, BIOMÉDICA Y ELECTRÓNICA

# Índice

## Computación e Informática

<b>Pruebas de software utilizando la técnica de revisión de código: un estudio exploratorio</b> Juan Pablo Ucán Pech, Raúl Antonio Aguilar Vera, Julio César Díaz Mendoza, Antonio Armando Aguilera Güemez	<b>C1</b>
<b>Uso de métricas de interactividad para la interpretación de publicaciones de una fan page institucional en Facebook</b> Paola E. Velazquez-Solis, Jorge Eduardo Ibarra-Esquer, M. Angélica Astorga-Vargas, Brenda Leticia Flores Rios, Mónica Carrillo Beltrán, Jesús Caro-Gutiérrez, Raúl Antonio Aguilar Vera	<b>C2</b>
<b>Inteligencia Artificial en Sistemas de recomendación Médica: Una revisión de la Literatura</b> Javier A. Calderón Blas, María Angélica Cerdán, Angel Juan Sánchez García, Saúl Domínguez Isidro	<b>C3</b>
<b>Detección automática de brócoli en imágenes RGB-D</b> Luis Talavera Ramirez, Luis Antonio Quiroz Mercado , Héctor Alejandro Montes Venegas, Rosa María Valdovinos Rosas, José Raymundo Marcial Romero	<b>C4</b>
<b>Combinación de Realidad Aumentada con Control Gestual de Manos para Apoyar el Aprendizaje de las Letras en Niños de Preescolar</b> Maria Antonieta Abud Figueroa, Raúl de Jesús Sánchez Martínez, Ulises Juárez Martínez, Lisbeth Rodríguez Mazahua, Hilarión Muñoz Contreras	<b>C5</b>
<b>EL Estudio comparativo de algoritmos para el reconocimiento de dígitos manuscritos. Caso: MNIST</b> Yasiel Conde Bernal, Saul Lazcano Salas, Maricela Quintana López, Saturnino Job Morales Escobar, Asdrúbal López Chau	<b>C6</b>
<b>Implementaciones actuales del modelo confianza cero para entornos en la nube: una revisión sistemática</b> Johann Castillo Oliva, Bruno Hiroshi Espinosa Luna, Alberto Carlos Mendoza de los Santos	<b>C7</b>
<b>Diagnóstico de TDAH con Machine Learning y Sensores: Un Mapeo Sistemático</b> Iván de Jesús Cetina Ucán, Antonio Aguilera Güemez, Raúl Antonio Aguilar Vera, Juan Pablo Ucán Pech	<b>C8</b>
<b>Plug and play (upnp): métodos de ataque y medidas de protección</b> Leticia Chuquiana Casicana, Galo López Sevilla	<b>C9</b>
<b>Métricas de productividad y gestión del trabajo basadas en psp para estudiantes de ingeniería de sistemas: caso universidad eafit</b> Paola Vallejo, Andrés Echeverri Jaramillo, Liliana González Palacio, Rafael David Rincón	<b>C10</b>
<b>La gestión de procesos en las empresas de software de San Francisco de Campeche</b> Diana Concepción Mex Alvarez, Luz María Hernández Cruz, Charlotte Monserrat de Jesús Llanes Chiquini, Andres Castro Villagrán, Susana Friné Moguel Marín, Carolina Chi Arceo	<b>C11</b>
<b>Revisión de literatura de elementos de minería de procesos para la minería de procesos de software</b> Silvia Jaqueline Urrea Contreras, Brenda Leticia Flores Rios, Angélica Astorga Vargas, Jorge E Ibarra Esquer, Félix F. González Navarro, Iván A García Pacheco	<b>C12</b>
<b>Los Algoritmos, metricas y validacion en la estimación del esfuerzo y su impacto en la gestion de proyectos DevOps un mapeo sistematico de la literatura</b> Iliana Alvarado, Noe Alejandro Castro Sanchez, Blanca Dina Valenzuela Robles, Rene Santaolaya Salgado, Juan Gabriel González Serna	<b>C13</b>
<b>Deudas técnicas en el sistema transaccional</b> Javier Fabricio Cucalón-Gaibor, Jacobo Sandoval Gutierrez	<b>C14</b>
<b>Enseñanza de programación para niños y jóvenes con discapacidad visual: mapeo sistemático de la literatura</b> Raúl Arturo Peredo-Estudillo, Juan Carlos Pérez Arriaga, Gerardo Contreras-Vega, Ángel J. Sánchez-García	<b>C15</b>
<b>Electrónica</b>	
<b>Efecto de la oscilación térmica diaria sobre celdas de carga de galgas extensiométricas genéricas</b> Rubén Esparza Orozco, Abel Quevedo Nolasco, Carlos Ramírez Ayala, Roberto Ascencio Hernández , Bertha Patricia Zamora Morales	<b>E1</b>

*Recibido 31 Oct. 2023*

*ReCIBE, Año 12 No. 2, Nov. 2023*

*Aceptado 01 Nov. 2023*

## **Software testing using the code review technique: an exploratory study**

### **Pruebas de software utilizando la técnica de revisión de código: un estudio exploratorio**

Juan Pablo Ucán Pech<sup>1</sup>

Raúl Antonio Aguilar Vera <sup>1\*</sup>

*avera@correo.uady.mx*

Julio César Díaz Mendoza<sup>1</sup>

Antonio Armando Aguilera Güemez<sup>1</sup>

<sup>1</sup> *Facultad de Matemáticas, Universidad Autónoma de Yucatán, Mérida, Yucatán, México.*

\*Corresponding Author

## **ABSTRACT**

This paper presents an exploratory study with Belbin roles, in particular role types, in an individual activity consisting of code review. The objective is to identify if, in addition to any of the three types of roles, the position of the fault and the gender of the subject influence the activity of detecting faults in the code. To create an experimental context for the review process, during a work session the subjects, who were software engineering students, used a code with injected faults for their review. With respect to the types of roles, the results of the experiment do not show significant differences either in the efficiency index obtained by the subjects in the testing process, or with the confusion index of failures. On the other hand, regarding the position of the fault, the results show significant differences between the faults detected in the first half of the code with respect to the remaining second half. Regarding the gender of the subject, the experiment does not show a significant difference between the detected faults.

At the end of this report, researchers should perform a short-term analysis of the faults introduced in the code to obtain a better version of the experimental object, allowing them to perform a second controlled experiment under less restrictive conditions.

## **KEYWORDS**

Code Review, Experimentation, Faults in the code, Software Testing.

## **RESUMEN**

Este artículo presenta un estudio exploratorio con los roles de Belbin, en particular los tipos de roles en una actividad individual que consiste en la revisión de código. El objetivo es identificar si, además de alguno de los tres tipos de roles, la posición de la falta y el género del sujeto influyen en la actividad de detección de faltas en el código. Para crear un contexto experimental para el proceso de revisión, durante una sesión de trabajo los sujetos, que eran estudiantes de ingeniería de software, utilizaron un código con faltas inyectadas para su revisión. Respecto a los tipos de roles, los resultados del experimento no muestran diferencias significativas en el índice de eficiencia obtenido por los sujetos en el proceso de prueba, así como tampoco en el índice de confusión de fracasos. Por otro lado, en cuanto a la posición de la falta, los resultados muestran diferencias significativas entre las faltas detectadas en la primera mitad del código respecto al resto de la segunda mitad del código. En cuanto al género del sujeto, el experimento no muestra una diferencia significativa entre las faltas detectadas.

Al final de este reporte, los investigadores deberán realizar un análisis a corto plazo de las faltas introducidas en el código para obtener una mejor versión del objeto experimental, lo que les permitirá realizar un segundo experimento controlado en condiciones menos restrictivas.

## **PALABRAS CLAVE**

Revisión de Código, Experimentación, Faltas en el código, Pruebas de Software.

## **1. INTRODUCTION**

The body of knowledge developed and accumulated over half a century after the so-called Software Crisis (Bourque & Fairley, 2014) has maintained a dynamic of constant improvement in terms of quality, both of its processes and of the artifacts that are generated by the first. The studies reported on software development and management processes have been analyzed based on various factors; however, the intrinsic social aspect of the discipline (Juristo & Moreno, 2001) has led to consider the human factor as an aspect of unique importance to your research. (Morales & Vega, 2018) proposes a catalog of human factors that are critical to the success of proposals to improve the software process, and among these factors is the role played by a Software Engineer within the work team.

In studies on the roles played by team members, a distinction is made between roles that focus on individual activities, on the one hand, and roles that are described in terms of the tasks they can perform as a team. Among studies on roles, Belbin's propositions (Belbin, 1981, 1993) are among the most popular in academic and professional contexts.

The purpose of this research is to expand the study related to the use of Belbin roles, in particular the types of roles, in an individual activity—in the context of software development—that corresponds to code review. The aim is to identify whether, in addition to any of the three types of roles, the position of the fault and the gender of the subject have an influence on the fault detection activity.

The following section presents the theoretical framework that supports Belbin's role theory, as well as the fault detection. Section three, four and five presents three analysis a controlled experiment carried out with teams of students – as experimental subjects – in the software testing task, particularly static code review. The three analysis was by role type, fault position and by gender respectively. Finally, section six presents the conclusions of the empirical study, as well as future work identified by the researchers.

## **2. BACKGROUND**

### **2.1. Belbin Roles**

Belbin (1981, 1993) maintains that a team role refers to the way of behaving, contributing and relating to other people at work and although some of the roles are natural, other roles could be adopted by the individual himself and some may even be discovered after being adopted. The nine roles proposed by Belbin can be grouped around the type of conduct in three different categories as described in Table 1.

**Table 1.** Belbin Roles Categories

Type	RoI	Characteristics
Action	Sharper (S) Implementer (I) Completer-Finisher (CF)	They are those roles that initiate, develop and finish tasks.
Mental	Plant (P) Monitor-Evaluator (ME) Specialist (SP)	They are those roles that have the knowledge and skills required for the task, as well as a critical vision for its realization.
Social	Chairman (CH) Resource Investigator (RI) Teamworker (TW)	They are those that promote communication and cohesion both among the members of the team and with the people with whom the team interacts.

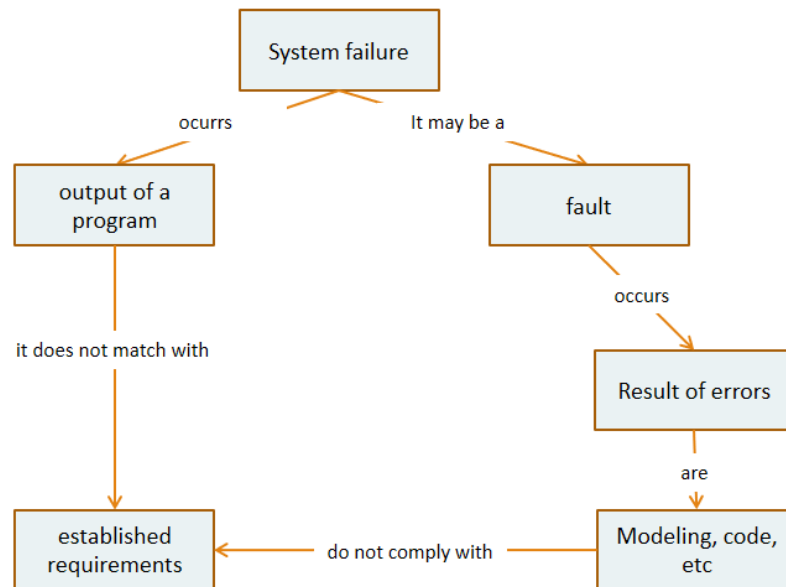
Estrada & Peña (2013) presents the use of Belbin's role theory in individual tasks in the context of Software Engineering. This reported a controlled experiment with students performing activities related to the stages of requirements, design and coding; the authors conclude that some roles have greater input in certain activities, particularly pointing to the Implementer role in the coding task.

Another study is reported in (Aguilar et al., 2022), in which the possible differences between the nine roles in tasks related to the Logical Design of a Database are explored; the authors report that the Monitor-Evaluator role presents significant differences, having obtained a better quality grade than the other six roles participating in the experiment.

## 2.2. Fault Detection In The Code

The way that Software Engineering has to evaluate a software artifact is known as Empirical Verification, this way of acting does not provide a definitive solution.

When we evaluate code, on many occasions the words error and fault are often confused or used without distinction as if they were the same. In a similar way it occurs with two other words such as failure and defect, however, they are not the same. For example in the Figure 1 we can see that one system failure occurs when an output of a program it does not match with the established requirements, but a system failure it may be a fault that occurs as results of errors. On the other hand, the results of errors are when modeling, code or other artifacts do not comply with established requirements.



**Figure 1.** Failure, fault, error and their relations.

According to Juristo et al. (2006), there are four concepts that are usually used indifferently, and however they represent different constructs:

- Error: human action that produces a fault.
- Fault: something is wrong with a product (model, code, document, etc.).
- Failure: manifestation of a fault.
- Defect: error, fault or failure.

Classifying coding faults is not an easy task, in the literature; it is possible to find a variety of classifications about types and causes of faults that can be made, mainly by novice programmers. These classifications have been generated as a result of empirical studies on patterns found in programming activity (Ucán et al., 2022). Basili & Selbi (1987) exposes a typology indicating that faults can be classified in the following six classes of faults: Initialization, Computation, Control, Interface, Data and Cosmetic.

### 3. EXPERIMENTAL ANALYSIS I: BY ROLE TYPE

The purpose of this controlled experiment is to explore whether any of the types of roles proposed by Belbin (Action, Mental or Social) present a performance in the code review process that is statistically different from that of the other two. Two performance indicators have been particularly considered: Effectiveness and Confusion.

*Effectiveness* Index: The quotient of the number of faults correctly detected by the experimental subject, between the total number of faults injected.

*Confusion* Index: The quotient of the total number of incorrect errors detected between the total number of errors reported by the experimental subject.

The controlled experiment was carried out with students who completed their first programming course. According to the classification of programming experience created by Dreyfus & Dreyfus (1986), the experimental subjects can be classified as beginning students.

**3.1. Factor and alternatives**

According to Table 1, Belbin classified the nine team roles into three types of roles: Action Roles (AR), Mental Roles (MR), and Social Roles (SR); therefore, in this first experimental analysis, the type of role is considered as a factor, and the three types as alternatives: AR, MR and SR.

**3.2. Hypotesis and variables**

The first pair of statistical hypotheses uses as a dependent variable, a metric linked to the quality of the individual process, the effectiveness (effectiveness index) in identifying faults in the code.

- H<sub>01</sub>: The means of the effectiveness index in the detection of faults in the code, by the three Types of Belbin Roles, do not present differences.
- H<sub>11</sub>: The means of the effectiveness index in the detection of faults in the code, by the Belbin Role Types, differ in at least a couple of these.

The second variable is linked to the error made by the evaluator in the fault identification task, when incorrectly identifying a type of fault; human confusion (confusion index) will be used as a metric in the identification of faults in the code. The statistical hypotheses derived from said variable are the following:

- H<sub>02</sub>: The means of the confusion index in the detection of faults in the code, by the three Types of Belbin Roles, do not present differences.
- H<sub>12</sub>: The means of the confusion index for the detection of faults in the code, by the Belbin Role Types, differ in at least a couple of these.

**3.3. Experimental unit**

The experimental unit, also known as the experimental object, is the piece or sample that is used to generate a value that is representative of the result of the experiment. In our study, since the activity is the detection of faults in the code, the experimental object is the code reviewed by the subjects during the experiment.

Date: \_\_\_\_\_  
 Name: \_\_\_\_\_  
 Begin time: \_\_\_\_\_  
 End time: \_\_\_\_\_  
 Number of faults found: \_\_\_\_\_

Faults descriptions

No.	Type	Line or position	Fault description

**Figure 2 – Data collection instrument.**



For the purposes of our study, twelve faults were injected into the code, these faults used the classification proposed in (Basili & Perricone, 1984) as a reference and were distributed throughout the 214 lines of code (LOC). For the activity to be carried out with the code, an instrument was designed in which the registration of the information required for the identification of the subject is requested, as well as each of the faults detected (see Figure 2).

### 3.4. Experimental design

The most appropriate experimental design for our study is the factorial design with a source of variation and three alternatives (see Table 2). The dependent variables are numerical metrics that are obtained from the analysis of the information obtained with the instrument illustrated in Figure 2, which will be used by the experimental subjects to record the information described in the previous section.

**Table 2** –Factorial design with one source of variation and three alternatives

Factor	Alternatives	Dependent variables
Role type	Action Roles (Type 1)	Effectiveness, Confusion
	Social Roles (Type 2)	
	Mental Roles (Type 3)	

### 3.5. Experimental subjects

The convenience sample used in the experiment consisted of 24 of the students enrolled in the aforementioned course, these participants already had the essential knowledge of the C programming language as well as programming logic under the structured paradigm. To identify the type of role of each subject, at the end of one of the class sessions during the course, the Belbin self-perception test was applied to all students enrolled in the course. With this, the authors identified the role assumed — and therefore the type of role—for each of the possible subjects. Finally, the list was refined with the students who voluntarily wanted to collaborate with the study and participated in the experimental session. Genero et al., (2014) argued that a student-based sample allows the researcher to obtain preliminary evidence to confirm or refute hypotheses that can then be tested in an industrial context.

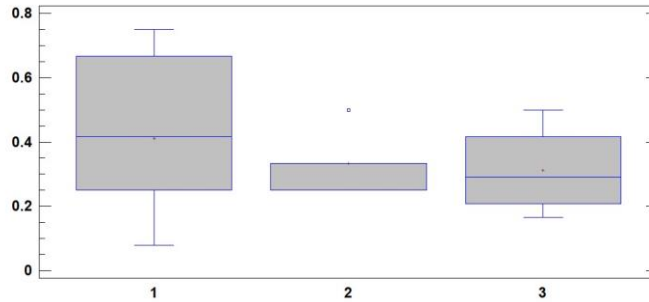
### 3.6. Descriptive Analysis

Table 3 presents some of the most important measures of central tendency and variability for the Effectiveness variable. We can identify that the mean and median are higher with type roles 1, although the type of role with the least variability is 2 also; it is also possible to observe that the unbalanced sample contains fewer subjects with types 2 and 3.

**Table 3** –Statistical summary for the Efficiency Index

Type	#	Mean	Median	SD
1	15	0.410889	0.416667	0.23279
2	5	0.333333	0.333333	0.102062
3	4	0.3125	0.291667	0.142319
Total	24	0.378333	0.333333	0.198304

To compare the three alternatives, we generate a box-and-whisker plot. Such a plot can allow us to observe the dispersion and symmetry of the three data sets; in Figure 3 we can see less dispersion in types 2 and 3, however, there is no gap between the three data sets, which suggests that there is no significant difference between the three types of roles.



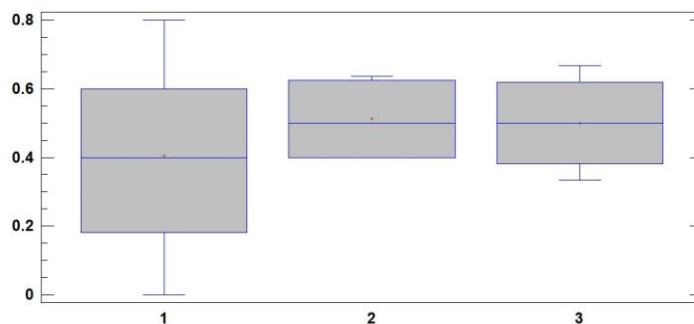
**Figure 3 – Box and Whisker Plot for the Effectiveness.**

Regarding the Confusion variable, Table 4 contains some of the most important measures. We can observe that type 1 presents a lower value in the mean and median, although there is a greater dispersion among the three types of roles.

**Table 4 – Statistical summary for the Confusion Index**

Type	#	Mean	Median	SD
1	15	0.405091	0.4	0.249227
2	5	0.512273	0.5	0.115615
3	4	0.5	0.5	0.148054
Total	23	0.443239	0.422619	0.213396

In Figure 4, we can observe and compare the mean, mode and dispersion of the three data sets and it is noteworthy that the roles of type 2 and 3 in 75% of the cases they coincide with 50% of those of type 1. That is, there is a certain gap, which will have to wait for the inferential analysis to determine if these differences are significant.



**Figure 4 – Box and Whisker Plot for the Variable Confusion.**

### 3.7. Inferencial analisis

With the purpose of statistically evaluating the differences between the alternatives of the Effectiveness and Confusion ratio variables, the one-way Analysis of Variance was applied (Gutierrez & De la Vara, 2012); the result of evaluating with ANOVA is illustrated in Table 5.

**Table 5 –ANOVA for the Variables Effectiveness and Confusion**

Metric	Ratio F	p-value
Effectiveness	0.53	0.5970
Confusion	0.62	0.5466

In both variables, a p-value much greater than 0.05 is obtained, which indicates that the null hypothesis is accepted; however, it is necessary to evaluate the model in both cases to be able to consider said hypotheses as true.

### 3.8. Validation of Model Assumptions

To validate the first assumption, Shapiro Wilk test allows us to evaluate whether a variable has a normal distribution or not (see Table 6).

**Table 6 – Shapiro Wilk Test for the Variables Effectiveness and Confusion**

Metric	Test	p-value
Effectiveness	0.945844	0.2198
Confusion	0.982342	0.9348

The p value for the Shapiro Wilk Test in both variables is greater than 0.05, therefore, it is possible to assume that both samples have a normal distribution. Regarding the second assumption, that of homoscedasticity, Levene's test allows the evaluation of significant differences between the variances of the two data sets (see Table 7).

**Table 7 – Levene's Test for the Variables Effectiveness and Confusion**

Metric	Test	p-value
Effectiveness	3.42508	0.0516
Confusion	1.91691	0.1719

The p value for the Levene test in both variables is greater than 0.05, which indicates that there is no statistically significant difference between the standard deviations of the alternatives, with a 95.0% confidence level; the above, both for the Effectiveness variable and for the Confusion variable.

Finally, for the assumption of data independence, although the subjects present independence as independent individuals, we chose to run the Durbin-Watson test, in order to identify if there is no relationship in at least the temporal sequence of the data. In the Table 8 the p value for the Durbin-Watson Test in both variables is greater than 0.05, which confirms our suspicion of independence.

**Table 8 – Durbin-Watson Test for the Variables Effectiveness and Confusion**

Metric	Test	p-value
Effectiveness	1.57585	0.1544
Confusion	2.06244	0.5553

**4. EXPERIMENTAL ANALYSIS II: BY FAULT POSITION**

The purpose of this second analysis is to explore whether the position of the fault has any influence on its detection. The correct number of faults detected in both halves of the code has been considered as a metric. However, because the faults are of different types, we have considered the metric on an ordinal scale.

**4.1. Factor and alternatives**

The position of the fault in the code is considered as a factor, with alternatives being the number of faults in the first half of the code (FHC) and the number in the second half (SHC).

**4.2. Hypotesis and variables**

The statistical hypotheses for this second analysis use the visibility of the faults as the dependent variable, and for this a metric linked to the result of the individual fault detection process is used, the number of faults correctly detected in both sections of the code (FHC and SHC).

- $H_{03}$ : The median of the errors detected in the FHC by the software engineering students is equal to the median of the errors detected in the SHC by said students.
- $H_{13}$ : The median of the faults detected in the FHC by the software engineering students differs from the median of the faults detected in the SHC by said students.

**4.3. Experimental design**

The most appropriate experimental design for our study is the factorial design with a source of variation under a paired sample scheme (see Table 9), that is, two measurements are obtained from each of the 24 subjects, the number of failures correctly detected. in the FHC and the number of faults correctly detected in the SHC.

**Table 9 –A source of variation with paired samples**

Experimental subject	Measurement 1	Measurement 2
1	# Faults in the FHC	# Faults in the SHC
...	...	...
24	# Faults in the FHC	# Faults in the SHC

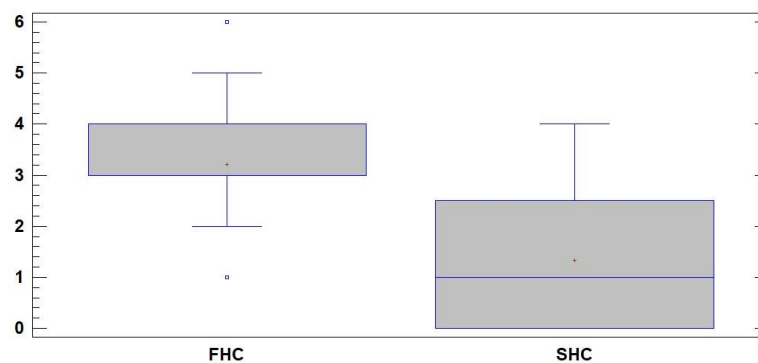
#### 4.4. Descriptive Analysis

Table 10 presents some of the most important measures of central tendency and variability for the visibility of the faults variable. We can identify that the mean and median are higher with type roles 1, although the type of role with the least variability is 2; also, it is also possible to observe that the unbalanced sample contains fewer subjects with types 2 and 3.

**Table 10** –Statistical summary for the visibility of the faults variable

Type	#	Mean	Median	SD
FHC	24	3.20833	3.0	1.28466
SHC	24	1.33333	1.0	1.46456

In Figure 5, we can observe that the FHC data sample has a bias to the left while the SHC data sample has greater symmetry. On the other hand, comparing both samples, quartile 1 of the FHC is above quartile 3 of the SHC, which leads us to think that there is a difference in the number of offenses detected in the FHC compared to the SHC. In this case, more faults are detected in the FHC.



**Figure 5** – Box and Whisker Plot for the visibility of the faults variable.

Figure 6 shows the results of the correct type faults by position, which were found by the subjects during the experiment.



**Figure 6** – Faults by position.

In Figure 6, you can observed that most of the correct faults found are in the first half of the code, and one fault that was not found was the initialization fault.

#### 4.5. Inferencial analisys

Since two paired samples were uncomparated using an ordinal mean for the dependent variable, the Wilcoxon signed rank test was selected, which evaluates the null hypothesis that the median of PM-SM is equal to 0.0 in contrast to the alternative that the PM-SM median is not equal to 0.0

**Table 11** –Wilcoxon signed rank test for the Variable Fault position

Metric	Ratio F	p-value
# of Faults	3.84157	0.000122292

Given that the p-value is less than 0.05, the null hypothesis is rejected and we can consider that there is a significant difference between the medians of the faults detected in the FHC with respect to the faults detected in the SHC (see Table 11).

### 5. EXPERIMENTAL ANALYSIS III: BY GENDER OF REVIEWER

The objective of this third analysis is to explore whether the gender of the reviewer has any influence on the detection of faults in the code. The correct number of faults detected in the code has been considered as a metric. As in the second analysis, it has been considered that because the failures are of different types, the metric is considered on an ordinal scale.

#### 5.1. Factor and alternatives

The biological gender of the reviewer is considered as the source of variation, the alternatives being Male gender (MG) and female gender (FG).

#### 5.2. Hypotesis and variables

The statistical hypotheses for this second analysis use the visibility of the faults as a dependent variable, and for this a metric linked to the result of the individual fault detection process is used, the number of faults correctly detected by the subjects based on their biological gender. (MG and FG),

- $H_{04}$ : The median of the errors detected by the MG is equal to the median of the errors detected by the FG.
- $H_{14}$ : The median of the faults detected by the MG differs from the median of the faults detected by the FG.

#### 5.3. Experimental design

The most appropriate experimental design for our study is the factorial design with one source of variation and two alternatives (see Table 12), where the alternatives are: the male gender and the female gender.

**Table 12** –Factorial design with one source of variation and two alternatives

Experimental Unit	1	2	3	...	24
Gender	MG	FG	FG	...	MG

**5.4. Descriptive Analysis**

Table 13 presents some of the most important measures of central tendency and variability for fault detection by gender. We can identify that the mean and median are higher for the female gender, and that the standard deviation is lower for this alternative.

**Table 13** –Statistical summary for the faults detected by gender

Type	#	Mean	Median	SD
MG	14	4.28571	4.0	2.70124
FG	10	4.9000	4.5	1.91195

In the Figure 7 we can see that the female sample has less variability, but said range of values is included in the range of the male sample, so from a visual perspective, it seems that there are no significant differences between both alternatives.



**Figure 7** –Box and whisker plot for the faults detected variable.

**5.5. Inferencial analysis**

Since the purpose of the experimental design is to compare two alternatives from independent samples, the Mann Withney U test was selected.

**Table 14** –W de Mann-Whitney test for faults detected variable

Metric	W	p-value
# of Faults	82.5	0.477348

Given that the p-value is greater than the value of the test's significance level (0.05), the null hypothesis is accepted, so it is possible to conclude that there is no significant difference between the faults detected by the male testers and those of the feminine gender (see Table 14).

## 6. CONCLUSIONS

In this paper, a first controlled experiment was presented with software engineering students with the objective of investigating whether one of the types of Belbin roles is compatible with the testing process using code review techniques, likewise, a deeper analysis investigating whether the position of the faults and the gender of the subjects show significant differences between the detected faults.

As a result, it was observed in the experiment that:

- do not show significant differences between the three types of role,
- show significant differences between the faults detected in the first half of the code with respect to the remaining second half,
- do not show a significant difference between the detected errors by gender.

It is worth mentioning that some subjects identified the correct fault in the code; however, they confused the type of fault. On the other hand, the majority of the correct faults found are found in the first positions of the code, this may be because the subjects did not have time to review the last lines of code.

Finally, some faults were not detected possibly because they were not designed as well as others (e.g. as mentioned in the previous paragraph, they caused confusion, forget, etc.)

The reflection on the part of the researchers forces them to carry out a short-term analysis of the faults injected into the code in order to obtain a better version of the experimental object, with which a second controlled experiment can be carried out with under less restrictive conditions and a larger sample.



## REFERENCES

- Aguilar, R., Peña, A., Díaz, J., & Ucán, J. (2022) Influence of Belbin's Role Theory on Database Design: Experimenting with Software Engineering Students. *Programming and Computer Software*, 48(8), 489-498.
- Basili, V., & Perricone, B. (1984) Software errors and complexity: an empirical investigation. *Communicatons of the ACM*, 27(1), pp. 42-52.
- Basili, V., & Selby, R. (1987). Comparing the effectiveness of software testing strategies, *IEEE transactions on software engineering*, (12), 1278-1296.
- Belbin, M. (1981), *Management teams. Why they succeed or fail*. John Wiley & Sons.
- Belbin, M. (1993). *Team roles at Work*. Elsevier Butterworth Heinemann.
- Bourque, P., & Fairley, R. (2014). *Guide to the Software Engineering Body of Knowledge (SWEBOK V3.0)*. IEEE Computer Society.
- Dreyfus, H. & Dreyfus, S. (1986) *Mind over Machine. The Power of Human Intuition and Expertise in the Era of the Computer*. Basil Blackwell.
- Estrada, E., & Peña, A. (2013). Influencia de los roles de equipo en las actividades del desarrollador de software. *Revista Electrónica de Computación, Informática, Biomédica y Electrónica*, 2(1), 1-19.
- Genero, M., Cruz-Lemus, J., & Piattini, M. (2014). *Métodos de investigación en ingeniería de software*. Ra-Ma.
- Gutierrez, H., & De la Vara, R. (2012). *Análisis y Diseño de Experimentos*. McGraw Hill.
- Juristo, N., & Moreno, A. (2001). *Basics of Software Engineering Experimentation*. Kluwer Academic Publishers.
- Juristo, N. Moreno, A., & Vegas, S. (2006). *Técnicas de evaluación de software*. Universidad Politecnica de Madrid.
- Morales, N., & Vega, V. (2018). Factores Humanos y la Mejora de Procesos de Software. Propuesta inicial de un catálogo que guíe su gestión. *Revista Ibérica de Sistemas y Tecnologías de Información*, (29), pp. 30-42.
- Ucán, J., Aguilar, R., & Díaz, J. *Entornos Virtuales Inteligentes basados en Colaboración para Asistir el Aprendizaje de la Programación*. CONAIC.
- Weigers, K. (1996). *Creating a Software Engineering Culture*. (pp. 28-34). Dorset House Publishing.

## BIOGRAPHICAL NOTES



**Juan Pablo Ucán Pech** has a degree in Computer Science by Faculty of Mathematics of the Autonomous University of Yucatan, a Master in Computer Systems by Technological Institute of Merida, and Ph. D. in Computer Systems by the Postgraduate and Research Directorate of the University of the South, Mexico. He is currently a professor at the Faculty of Mathematics of the Autonomous University of Yucatan. He is a member of the Academic Body Research of Software Engineering for Education. He has the recognition of the National System of Researchers as a National Candidate Researcher by CONAHCYT and has the recognition of the Program for the Professional Development of Teachers (PRODEP) since 2014. His research work focuses on the areas of Software Engineering and Computer Science Educational.



**Raúl Antonio Aguilar Vera** obtained the degree of Ph. D. from the Polytechnic University of Madrid, Spain (European Doctor Mention) and the Master's degree in Software Engineering from the same Institution; he also has the degree of Master in Higher Education from the Autonomous University of Yucatan. He is currently a professor at the Faculty of Mathematics of the Autonomous University of Yucatan. He is responsible for the Academic Body Research of Software Engineering for Education. He has been a member since 2014 of the CONAHCYT National Researchers System (Level 1). His research work includes the areas of Software Engineering and Educational Computing.



**Julio César Díaz Mendoza** is an Industrial Engineer by Technological Institute of Merida, and has a Master's degree in Information Technology by Inter-American Development University. He is a member of the Academic Body Research of Software Engineering for Education. He is currently a professor at the Faculty of Mathematics of the Autonomous University of Yucatan and an Institutional Representative before the National Association of Computer Education Institutions (ANIEI). He has recognition from the Program for Teacher Professional Development (PRODEP). His research work includes the areas of Software Engineering and Educational Computing.



**Antonio Armando Aguilera Güemez** has a degree in Computer Science by Faculty of Mathematics of the Autonomous University of Yucatan, Ph. D. and Master in Computer Science by Institute of Technology and Higher Studies of Monterrey. He is currently a professor at the Faculty of Mathematics of the Autonomous University of Yucatan. He is a collaborator of the Academic Body Research of Software Engineering for Education. He has the recognition of the National System of Researchers as a National Candidate Researcher by CONAHCYT and has the recognition of the Program for the Professional Development of Teachers (PRODEP) since 2022. His research work includes the areas of Software Engineering and Educational Computing.



## Using Interactivity Metrics for interpretation of posts from an Institutional fan page on Facebook

## Uso de métricas de interactividad para la interpretación de publicaciones de una fan page institucional en Facebook

Paola E. Velazquez-Solis <sup>1</sup>  
Paola.velazquez@uabc.edu.mx

Jorge E. Ibarra-Esquer <sup>2</sup>  
jorge.ibarra@uabc.edu.mx

M. Angélica Astorga-Vargas <sup>2</sup>  
angelicaastorga@uabc.edu.mx

Brenda L. Flores-Rios <sup>1</sup>  
brenda.flores@uabc.edu.mx

Mónica Carrillo Beltrán <sup>1</sup>  
monica@uabc.edu.mx

Jesus Caro-Gutierrez <sup>1</sup>  
Jesus.caro@uabc.edu.mx

Aguilar Vera Raúl Antonio <sup>3</sup>  
avera@correo.uady.mx

<sup>1</sup>Instituto de Ingeniería, Universidad Autónoma de Baja California, Mexicali, Baja California. México.

<sup>2</sup>Facultad de Ingeniería, Universidad Autónoma de Baja California, Mexicali, Baja California. México.

<sup>3</sup> Facultad de Matemáticas, Universidad Autónoma de Yucatán, Mérida, Yucatán, México.

**Abstract—** Nowadays digital landscape, enhancing user engagement on social media platforms is crucial for elevating the visibility of universities within their respective communities. Through online presence, universities can transform their research activities and outcomes into valuable assets that extend well beyond the academia. As a result, the relationship with their community strengthens, opening new opportunities for collaboration. Interaction data from Social networks allows to identify the elements of User Engagement and, with this data, is possible to comprehend user behavior in order to create an online university community. A quantitative analysis of user engagement, viewed through university's commitment to its community, was conducted. The research findings offer a strategic tool to enhance the dissemination of scientific content geared toward user engagement. This study was carried out using data from a university's Facebook fan page, which served as a platform for sharing scientific content, news, and event updates. The research followed a methodology for analyzing social network data, which enabled the identification of key elements contributing to user engagement. The investigation revealed that variations in engagement for individual posts could be explained by a regression model, using the most correlated variables extracted from the fan page's interaction report. Further exploration, employing data mining techniques, underscored that self-generated content played the most pivotal role in driving user engagement.

**Keywords—** Social media, Interactivity, User Engagement, Facebook fan page, Spearman's Correlation Coefficient, Clustering

**Resumen—** Hoy en día, en el panorama digital el mejorar la participación de los usuarios en las plataformas de redes sociales es crucial para elevar la visibilidad de las universidades dentro de sus respectivas comunidades. A través de la presencia en línea, las universidades pueden transformar sus actividades y resultados de investigación en activos valiosos que se extienden mucho más allá del ámbito académico. Como resultado, la relación con su comunidad se fortalece, abriendo nuevas oportunidades de colaboración. Los datos de interacción de las redes sociales permiten identificar los elementos de User Engagement y, con estos datos, es posible comprender el comportamiento de los usuarios para crear una comunidad universitaria en línea. Se llevó a cabo un análisis cuantitativo de la participación de los usuarios, visto a través del compromiso de la universidad con su comunidad. Los resultados de la investigación ofrecen una herramienta estratégica para mejorar la difusión de contenido científico orientado a la participación de los usuarios. Este estudio se llevó a cabo utilizando datos de la página de fans de Facebook de una universidad, que sirvió como plataforma para compartir contenido científico, noticias y actualizaciones de eventos. La investigación siguió una metodología para analizar datos de redes sociales, que permitió identificar elementos clave que contribuyen a la participación de los usuarios. La investigación reveló que las variaciones en la participación en publicaciones individuales podrían explicarse mediante un modelo de regresión, utilizando las variables más correlacionadas extraídas del informe de interacción de la página de fans. Una mayor exploración, empleando técnicas de minería de datos, subrayó que el contenido autogenerado desempeñaba el papel más fundamental a la hora de impulsar la participación de los usuarios.

**Palabras clave—** Social media, interactividad, participación del usuario, página de fans de Facebook, coeficiente de correlación de Spearman, agrupamiento.

## 1. Introduction

Social networks are used by over 2.95 billion people (Liyanage & Premarathne, 2021), and have changed the way we create content, share information, communicate, and relate to each other (Froment et al., 2022). In the educational context, the appearance of Social networks makes the teaching-learning processes have a more social and customizable character in students, incorporating Social networks into academic activities (Shang et al., 2011). ResearchGate is a specialized academic Social network aimed at scientists and researchers to share publications, seek collaborators, communicate work in progress, and build a scholarly reputation (Sababi et al., 2017). It is a site that closely and realistically reflects the level of research activity of institutions, forming virtual Social networks centered on academic institutions (Yan & Zhang, 2018). But, the dynamics of scientific communication have changed, accelerating the knowledge feedback cycle (Campos-Freire & Rúas-Araújo, 2016). Facebook as an example of a generalist Social network, is used by most higher education institutional taking advantage of its benefits for the institutional dissemination of knowledge. In trying to increase the reach of their posts, higher educational institutions frequently vary the content of their posts, hoping to increase the engagement of their followers or users. In order to determine if there is an ideal content mix, it is important first to understand the current landscape (Peruta & Shields, 2017).

From a technological standpoint, interactivity is a representation of user preferences and the way they perceive information, feeding algorithms with data for future recommendations. It is a measure of how many users take part in the modification of the format and content of an environment in real time. Interpreting interactivity as a measurement strategy provides the potential to take advantage of data created from monitoring and analyzing posts on Social networks, benefiting from the expert knowledge (Steinmetz et al., 2020).

User activities and conversation metrics like reactions, which are described as the total number of "like", "love," "haha," "wow," "sad," and "angry" in the post; the number of times it is shared, and the total number of comments in a post are the most common benchmarks on Facebook, that describe interactivity and form the basis for its analysis and discussion (Furtado Avanza & Moreira Pinheiro, 2018; Sutanto, 2016). However, there are other metrics such as the number of people who viewed the post at least once. The number of people who viewed the content in News Feed, no need to interact with the post. And the number of clicks on posts that led to destinations or experiences on or off Facebook. From these metrics, it is assumed that a high level of interactivity generates various types of attitudes about the contents of fan pages or groups which exist on Facebook. These attitudes can be positive or negative (Kaye, 2021). All these are limited by the comparison between the results of interactivity in different posts of the same nature (Ballesteros Herencia, 2018; Sutanto, 2016).

Analyzing structured and quantifiable data like the user ratings, number of clicks and links, and network connections, among others (Chen et al., 2013; Savelev et al., 2021) provides a numerical value to the data. User Engagement on Social media platforms is one of the most discussed metric in the scientific community (Molina et al., 2021) whose level can be estimated by analyzing interactivity (Kim & Yang, 2017; Shahbaznezhad et al., 2021) that considers the qualitative meaning of an individual's cognitive state and the active relationship they have with the content (Furtado Avanza & Moreira Pinheiro, 2018). User Engagement is perceived as a significant factor for overseeing the use of Social networks, monitoring the acceptance of their contents, and understanding the level of interaction and interest that each publication generates (Furtado Avanza & Moreira Pinheiro, 2018).

Different approaches have recently been proposed for User Engagement calculation. This calculation requires formulas in which the total number of likes, comments and shares are added. In Facebook, the User Engagement is calculated with total of all interactions in the post (Figure 1). Due to the nature of the Facebook data, qualitative research methods require additional adaptation to capture visual, virtual, and textual interactions in Social networks accurately (Chen et al., 2013).

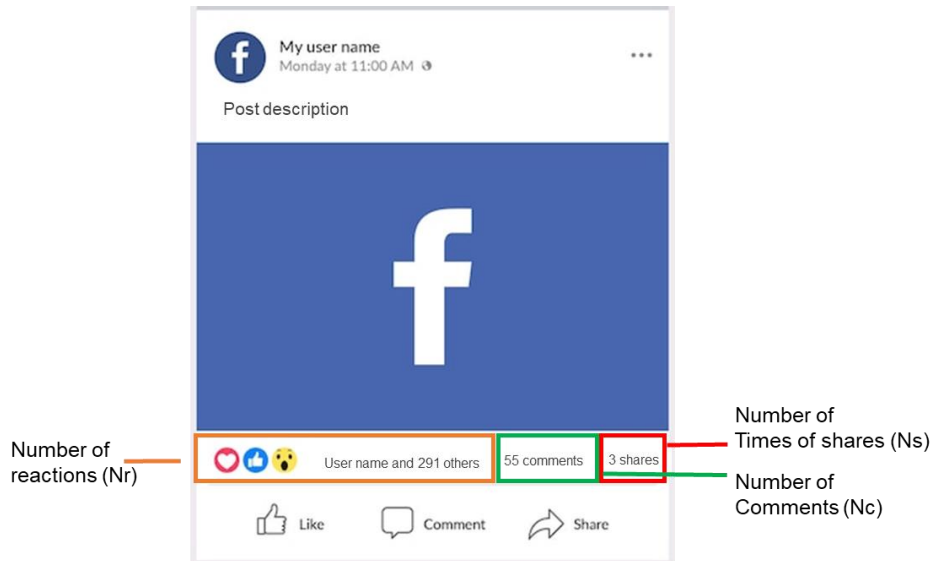


Figure 1. Formula for User Engagement in Facebook.

The scientific community considers the analysis of social network data as a multidisciplinary research area that enables the creation, extension, and adaptation of the data analysis models (Savelev et al., 2021). These models empirically provide quantitative and qualitative results using statistical methods based on evidence and objective data, which allow achieving improvements (Velazquez-Solis et al., 2022), performing a search exploration the information on Facebook for differentiating behavioral profiles of students and teachers (Karademir Coşkun et al., 2020). Various methods and techniques are implemented in the analysis of the Social networks data, like variable association (Eriksson et al., 2019; Phuntusil & Limpiyakorn, 2017), regression models (Purba et al., 2020), predictive models with process mining (Li & De Carvalho, 2019), and data mining techniques such as naive Bayes (Aviles & Esquivel, 2019), principal component analysis (Bharati et al., 2015), and K-means (Liyanage & Premarathne, 2021). Also, the unstructured textual information requires an analysis to understand the emotions, feelings, thoughts, opinions, and life experiences.

The objectives of this research are: 1) Use Interactivity metrics for interpretation how different types of content affect engagement in fan pages on Facebook. An Institutional fan page used for dissemination of scientific content, news, and events, created and maintained at a higher education institution, serves as the case of study 2) Create a regression model that predicts and explains variations in engagement for posts on the fan page based on the interaction reports from a two-year period.

The paper is organized as follows. In section II, related work about the qualitative analysis in Social networks and Facebook interactivity are described. Then, in section III we present the methodology; later in section IV data analysis and findings of the results are explained. Moreover, in section V conclusions and future work are presented.

## 2. Related work

### 2.1 Quantitative analysis on Social networks

Purba et al., (2020), a new metric for people who are not among the followers but liked a post from a user was proposed. For that, a regression model was developed using metadata, sentiment, hashtag to predict the outsider's percentage. On another hand, in the study of Eriksson et al., 2019, they take an explorative consumer engagement approach to analyze three interior design brands' postings on Facebook and Instagram. In order to, investigate differences in consumer brand post engagement regarding these postings.

The investigate of Aviles & Esquivel (2019) was centered on mining social networks data specifically on users' post from Facebook fan pages among selected Universities. The main objective of this study was mining social media data to identify their stakeholders' social networks behaviors and make knowledge-driven decisions using Naïve Bayes Classifier. The authors Liyanage & Premarathne (2021) introduce an approach to detect Facebook clone profiles by using a clustering technique on its profile attributes and network connections. With the Agglomerative hierarchical clustering algorithm and Jaccard similarity measurement, they list the possible clones with their percentages.

### 2.2 Facebook Interactivity

Measurement of User Engagement is calculated by three different methods: surveys and interviews, implicit measures, and web analytics (Chan-Olmsted et al., 2017). In web analytics, User Engagement is obtained from the behavior shown by users through their activity on digital platforms. Therefore, a series of metrics are required to quantify interaction-related variables, such as the total number of reactions on the post ( $N_r$ ), the number of user comments on the post ( $N_c$ ), and the number of times users shared the post ( $N_s$ ) (Bonsón & Ratkai, 2013; Hu & Chen, 2016; Jayasingh, 2019; Pilař et al., 2021).

There exist diverse formulas and models for calculating User Engagement based on data obtained from Facebook fan pages or groups. Table 1 presents the evolution of metrics used to calculate User Engagement identified over 2013 - 2020. When reviewing each work, eight of them had no identifiable rationale for variables or mathematical verification about the origin of the formula they were implementing. Having information from the variables used by different formulas and models helps to identify an integrated measure of User Engagement, underscoring the importance of the parameters normalized (Eriksson et al., 2019).

All the formulas shown in Table I are based on an addition of the different types of interactions, expanding from the formula used by Facebook (equation 1).

$$\text{User Engagement on Facebook} = N_r + N_c + N_s \quad (1)$$

A difference is established in the formula proposed by (Jayasingh, 2019), where coefficients, or weights, are used for each type of interaction as a representation of the effort needed. In addition, engagement rate is calculated over a fraction of the total followers of the page, represented by the 0.8 exponent.

Year	Metrics	Proposed Formula	Authors
2013	Likes Comments Sharing Followers Messages	$\frac{N_L + N_C + N_S}{\frac{N_{Messages}}{N_F}}$	Bonsón & Ratkai Hoffmann
	Likes Comments Sharing Followers	$\frac{N_L + N_C + N_S}{N_F}$	Niciporuc
2014	Likes Comments Sharing Followers Scope	$\frac{N_L + N_C + N_S}{\mu_{Scope}}$	Oviedo-García et al.
	Reactions Comments Sharing Likes on fan page	$\frac{N_R + N_C + N_S}{N_{Likes\ fan\ page}}$	Vadivu & Neelamalar
2017	Reactions Comments Sharing Followers	$\frac{N_R + N_C + N_S}{N_F}$	Herrera-Torres et al.
	Reactions Comments Sharing Likes on fan page	$\frac{N_R + N_C + N_S}{N_{Likes\ fan\ page}}$	Peruta & Shields
	Reactions Comments Sharing Likes on fan page	$\frac{N_R + N_C + N_S}{N_{Likes\ fanpage}^{0.8}}$	Phuntusil & Limpiyakorn
2018	Reactions Comments Sharing Scope	$\frac{N_R + N_C + N_S}{Scope\ post}$	Ballesteros-Herencia
2019	Reactions Comments Sharing	$N_R + 5N_C + 10N_S$	Ge & Grezel Eriksson et al.
	Reactions Comments Sharing Likes on fan page	$\frac{N_R + 5N_C + 10N_S}{N_{Likes\ fanpage}^{0.8}}$	Jayasingh Sudarsan
2020	Reactions Comments Sharing Posts Followers	$\frac{N_R + N_C + N_S}{N_{Post} \times N_{Followers}}$	Martínez-Sala & Segarra-Saavedra

Table 1. Formulas for User Engagement.

### 2.3 Qualitative analysis on Social networks

There are two directions in which qualitative analysis is recommended. The first, commonly known as qualitative research, is focused on the fields of sociology, medicine, psychology, and linguistics.

The second direction, called intellectual, is used in the field of Artificial Intelligence for qualitative analysis and is characterized by reasoning, simulation, and data analysis (Tsvetkov, 2021). A qualitative approach to social network data analysis supports the creation of meaning by placing technology in specific social contexts, places, and times. On the other hand, there is a need to implement inductive methods that allow us to delve deeper into the culture of digital spaces, considering users' perspectives as complex and heterogeneous (Voorveld et al., 2018).

A review of related work identified the application of various types of methods, which established data empirically using accepted statistical methods (He et al., 2017; Karademir Coşkun et al., 2020; Phuntusil & Limpiyakorn, 2017),



such as nonparametric tests (Bharati et al., 2015; Molina et al., 2021), analysis of variance (ANOVA) and coefficient correlation analysis for metrics analysis (Gil-quintana et al., 2022; Velazquez-Solis et al., 2022), statistical analysis models (Voorveld et al., 2018), regression models for interpretate and predicate metrics (Wheatley & Vatnoey, 2020), association models, predictive or data mining models like Naive Bayes (He et al., 2017), Principal Component Analysis (Gil-quintana et al., 2022), and K-means (Liyange & Premarathne, 2021). The Pearson's correlation coefficients, ANOVA test, chi-square test, and linear discriminant were used for a dataset analysis (Agusriandi et al., 2020).

The models employed are based on the extraction of knowledge from the data sets as by the ANOVA and text mining, to analyze the data later with the goal of understanding virtual social behavior, and also the feelings and emotions of interacting persons (Tsvetkov, 2021; Wheatley & Vatnoey, 2020). There are methodological proposals that present applications of algorithms for content analysis of text data (Chan-Olmsted et al., 2017), PageRank algorithm and HITS algorithm, Facebook API (Tsinovoi, 2020), JavaScript and REST API (Dabbagh & Kitsantas, 2012). In some of them, data mining and cleansing activities in the process of social network data analysis are similar (Eriksson et al., 2019; Phuntusil & Limpiyakorn, 2017; Pilař et al., 2021; Stieglitz et al., 2018; Surya Gunawan et al., 2020).

In the work of (Franz et al., 2019), qualitative research was conducted on Facebook users and their activity on the social network. They sought options to address health issues through the analysis of social network generated texts, concluding that qualitative and quantitative analysis on Facebook involves participation of the researcher in the study of observed patterns of information to identify relational themes, grammatical elements and the valence of sentiments contained in Facebook posts and associated comments. In trying to increase the reach of posts, organizations often vary the content, hoping to increase the User Engagement of their followers. The identified methods only rely on users' Social networks or their behavior, while the role of content contributed by fan page owners on Social networks is ignored (Al-Dheleai & Tasir, 2017).

### 3. Methodology

An adaptation of the methodologies for statistical and qualitative analysis of social network data proposed by (Agusriandi et al., 2020; Peruta & Shields, 2017; Pilař et al., 2021; Stieglitz et al., 2018; Surya Gunawan et al., 2020) was performed establishing four phases: 1) Sample selection and data collection, 2) Data preprocessing, 3) Application of methods and techniques, and 4) Visualization and Analysis of results. Fig 2 displays the adapted methodology. Each of the phases is explained in further detail in the following sections.

A research question was posed as a guide for the study: What is the content type that helps to increase interactivity of users interest in scientific dissemination and scientific work?

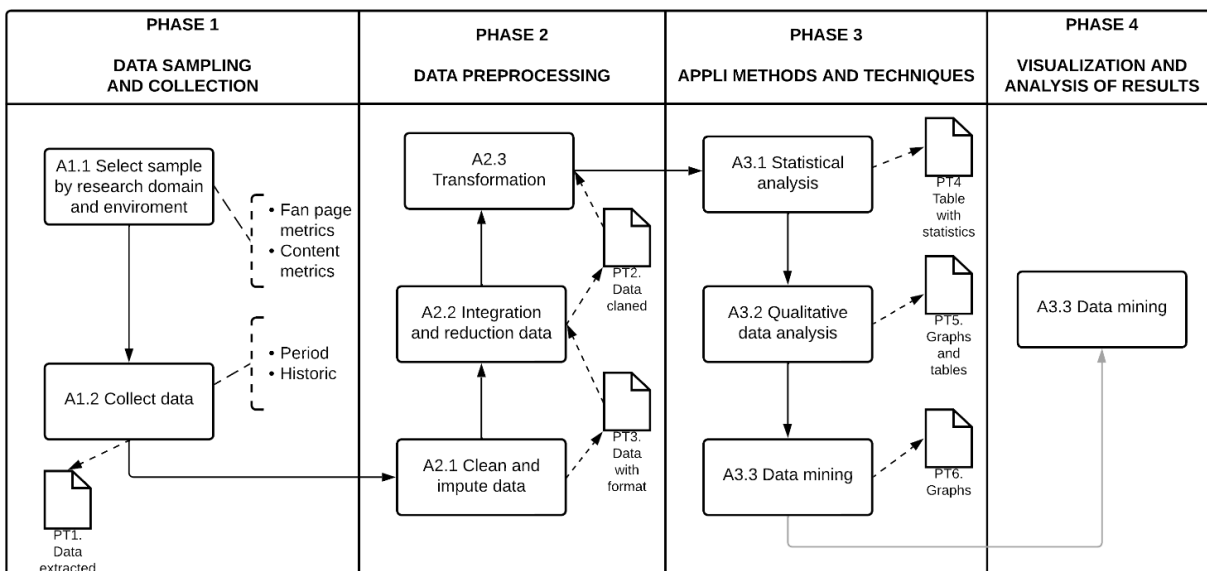


Fig 2. Methodology adapted from Peruta & Shields (2017); Stieglitz et al. (2018); Agusriandi et al. (2020); Surya Gunawan et al. (2020); Pilař et al. (2021).

### 3.1 Phase 1 - data sampling and data collection

The case study approach was adopted for Mujeres en la Investigación - UABC Facebook's fan page ([www.facebook.com/mujeresinvestigacion](http://www.facebook.com/mujeresinvestigacion)) managed by the community-university program of a Higher Education Institution. Since 2013, this fan page shares institutional information and results of scientific research activities carried out with the participation of 11 female researchers from the areas of Engineering Physics, Chemistry and Environment. To date, the fan page has a total of 1,570 followers and its publications have accumulated 1,393 likes.

The strategy defined for a two years period was that the selection of the extracted data would be a series of scientific demonstration and dissemination events in an online format, organized and published through this fan page. The data were generated between January 1, 2021 and December 31, 2022, corresponding to 345 posts. Taking into account that the number of likes, comments, and shares continuously changes in the publications, an update of the data sample was performed as of May 30 to have the most recent number of reactions, comments, clicks, and shares in the same 345 publications. Authors Agusriandi et al., (2020) and Niciporuc (2014) recommend allowing a period for users to reflect on the content after it has been posted. It was decided that one month after the last publication should be sufficient, since most reactions are often received within the first 48 hours. The result of this phase generated an XLSX or CSV document.

### 3.2 Phase 2 - data preprocessing

The calculation of User Engagement (E) of each publication was performed using the formula proposed by Jayasingh (Bonsón & Ratkai, 2013; Oviedo-García et al., 2014).

$$E = \frac{N_R + 5N_C + 10N_S}{N_{Likes\ fanpage}^{0.8}} \quad (2)$$

User Engagement results were placed as a new variable to analyze the User Engagement index of the followers towards the fan page. Subsequently, the processes of cleaning and imputation of missing data were performed. When applying statistical techniques, a preliminary analysis was performed to verify the normality distributions and homogeneity of variance in the sample with a Kolmogorov-Smirnov test. SPSS and RStudio tools were used to support the results processing and graphic visualization.

### 3.3 Phase 3 - apply methods and techniques

The main analysis included descriptive statistics of publications in general and type of content, and a nonparametric chi-square analysis. A chi-square analysis was chosen because there is a set of variables, where both the dependent and independent variables are categorical. An analysis of the variables was carried out by means of a Spearman's Rank correlation coefficient, with the intention of creating a proposed multiple regression model to explain the dependent variable (User Engagement). Section IV presents the results of phase 4.

## 4. Results and discussion

Sample initial exploration included characterizing the set of followers of the fan page. Although the content generated on that page is from women Figure 3 presents the 1,532 followers, 90% belong to locations in Mexico, where 83% correspond to women in the age range between 25 - 50 years and 17% of male followers distributed between 25 - 45 years mostly. The followers distribution is understandable due to the fan page objective which is to position the diffusion of scientific content created by women researchers.

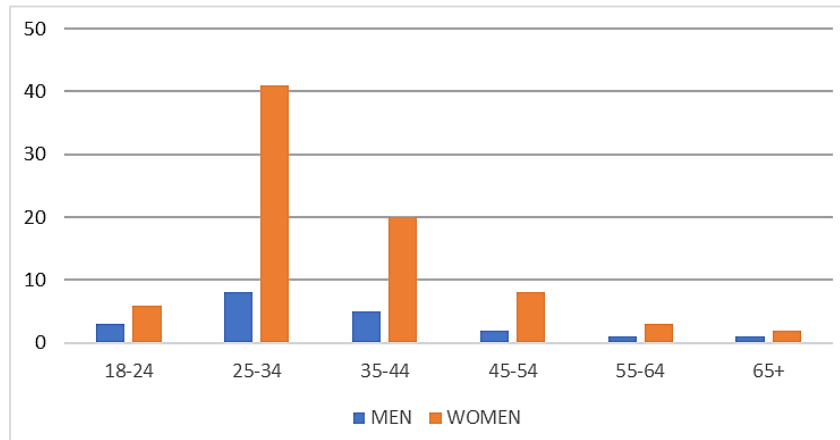


Fig 3. Age distribution of fan page followers.

Over the 24 months period from January 2021 to December 2022, the sample recorded a total of 345 posts. Table 2 shows the distribution of three quantitative metrics selected and extracted from the dataset: number of comments, number of reactions, and number of times a post was shared, separated by months of each year. Focusing on all-User Engagement metrics, there were an increase. This means that reactions increased by 951.5%, comments by 1528.9% and shares by 425.8%. The 3<sup>rd</sup> month (March 2022) shown the greater interactivity. This was due to the realization of an event in live streaming to dissemination science with 7289 interactions in posts, unlike March 2021 with 93 interactions. Among the most interesting findings is that in 2021, 7th month was the one that had the most interactions with a single publication made. Month 4 in both years maintained similar interactivity, since there are activities carried out by the fan page every year. In most months increase is observed, but there are particular cases where a decrease was observed, such as months 9, 10 and 11 of the year 2022.

2021				2022			
Month	Metrics			Month	Metrics		
	Reactions	Comments	Share		Reactions	Comments	Share
1	29	0	11	1	771	165	59
2	61	2	17	2	472	56	57
3	79	2	12	3	5996	994	299
4	254	25	52	4	393	507	71
5	48	1	2	5	122	181	11
6	7	0	1	6	373	415	4
7	252	103	23	7	30	1	2
8	0	0	0	8	43	1	0
9	40	1	0	9	22	1	5
10	7	1	0	10	11	2	0
11	83	16	2	11	7	0	2
12	8	1	0	12	19	1	1
TOTAL	868	152	120	TOTAL	8259	2324	511

Table 2. Distribution of reactions, comments and shares extracted from the dataset.

Five different categories of posts were identified: 1) Video, 2) Photo, 3) Link, 4) Live video, and 5) Text. Table 3 describes the distribution of posts according to the categories created. Looking more in depth at the different categories, significant findings emerge: In 2021, the month with the highest number of interactions (Table 2) was July, but it only had one Photo-type publication. In the case of the year 2022, the concentration of more interactions is in March, where about 85% of the posts were elements of the photo category. In Fig 4 it was identified that the number of Text-type posts decreased considerably. Video-type posts also decreased, but links to other external elements increased. Posts with photos remained constant. It is separated per month with the purpose of visualizing the behavior of the sets of categories in an ordered way. Table III compares the results of both years separated by month. The difference between the values can be observed, standing out the first six months of the year 2022. It was calculated with both formulas in the two periods and they were compared between them. In general, if there is an increase from one year to the next, regardless of the formula. In month eleven there is a decrease only in engagement on Facebook. As there are more interactions, more User Engagement is generated. It can be seen that in 2021 the

Engagement with the formula, as calculated by Facebook, is higher than that obtained with the 2022 posts. In month 8 of 2021, the publications had no interactions, impacting User Engagement. While month 7 obtained the highest value in this period.

2021						2022					
Month	Categories					Month	Categories				
	1	2	3	4	5		1	2	3	4	5
1	6.66	7.76	17.65	4.17	0	1	0	14.73	15.79	0	0
2	13.33	24.14	29.41	29.17	16.67	2	0	11.63	15.79	25.00	0
3	6.66	12.93	23.53	8.33	16.67	3	40	48.84	15.79	12.50	25
4	46.66	17.24	11.76	29.17	0	4	0	7.75	10.53	25.00	25
5	13.33	12.07	5.88	16.67	0	5	20	5.43	5.26	37.50	25
6	6.66	1.72	5.88	0	0	6	20	3.88	5.26	0	0
7	0	0.86	0	0	0	7	0	0.78	10.53	0	0
8	0	0	0	0	16.67	8	0	1.55	0	0	0
9	0	8.62	5.88	4.17	0	9	0	1.55	5.26	0	0
10	6.66	4.31	0	4.17	0	10	0	0.78	10.53	0	25
11	0	7.76	0	4.17	33.33	11	0	1.55	0	0	0
12	0	2.59	0	0	16.67	12	20	1.55	5.26	0	0

Table 3 Percentage of post with the categories.

Month	Engagement 2021		Engagement 2022	
	Facebook	Jayasigh	Facebook	Jayasigh
1	2.8571	2.9460	45.2272	29.4831
2	2.3488	1.8079	29.25	19.6131
3	3.2608	2.4640	102.662	58.3242
4	9.1944	7.4097	64.7333	71.9644
5	2.4285	1.0314	24.1538	25.9515
6	2	1.2610	113.1429	105.4626
7	378	295.8294	11	5.4398
8	0	0	22	7.1212
9	3.4166	1.1126	9.3333	7.6158
10	1.1428	0.5086	3.25	1.5577
11	6.3	3.1155	4.5	4.0057
12	2.25	0.9643	5.25	2.5221

Table 4. Comparison of the two years between the formulas by month

Table 5 allowed us to observe the correlations between the different categories extracted from the data set. This helped to verify the relationship between content type and period of time as related variables, where the incidence is visible in the elements of type photo. Also, in Table 5 we can see that in the calculation of the User Engagement with formula proposed by Jayasigh (2019) and Facebook. The Text-type contents would be the ones that have the least influence; while Video content and Links generate a medium impact on the relationship.

User Engagement Formulas	Categories				
	1) Video	2) Photo	3) Link	4) Live video	5) Text
Facebook	0.41	0.78	0.42	0.47	0.16
Jayasigh	0.33	0.77	0.32	0.48	0.21

Table 5. Comparison of correlations by categories

When making a cluster with a total of posts that were made distributed by the various calculations of the implemented equations. Fig 5 distributes the posts from lowest to highest User Engagement with both formulas. By the elbow method it will be reduced that the appropriate number of groups is three. The first group (blue) corresponds to publications that are low due to interactivity and for consequence have poor User Engagement. Group two (green) has the publications with a commitment between 100 and 300. Finally, group three (red) represents the publications with a commitment greater than 300 due have more interactivity. It can be seen in the graph that there is a greater concentration of publications in group one, denoting that most of the commitment in the publications does not have the expected response. It is important to mention that group two contains the publications corresponding to a scientific dissemination event. In a more detailed analysis of the publications, Fig 6 provides that the clusters identified are the same as those obtained in the previous analysis; i.e., group 1 of the graph in Fig 5 is the same group 1 in Fig 6.

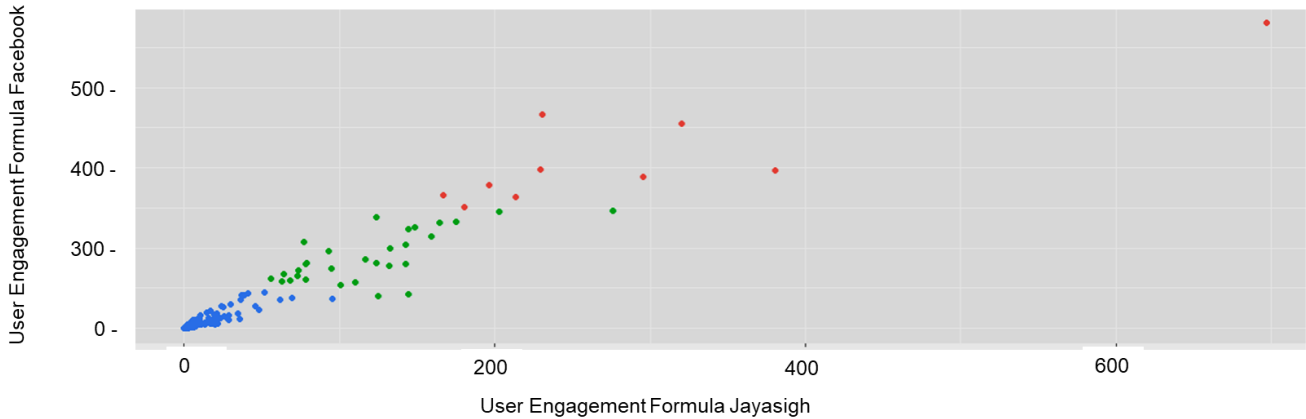


Fig 5. User Engagement with both formulas

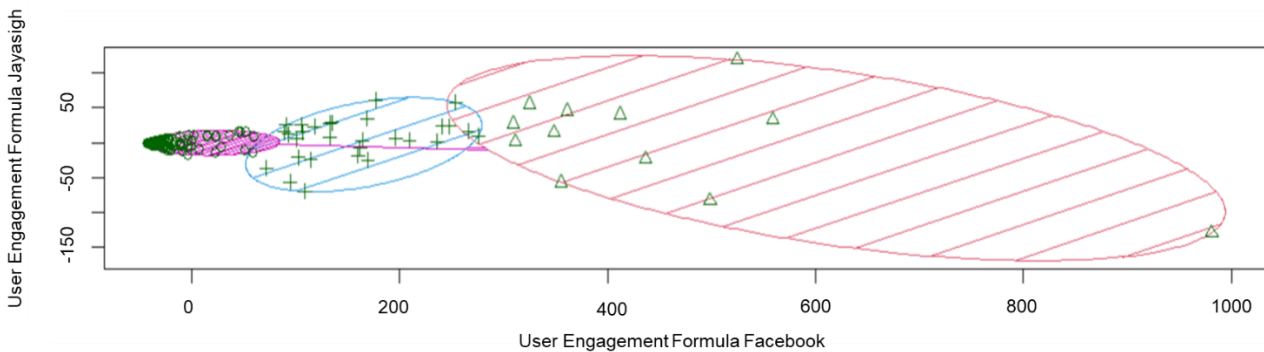


Fig 6. User Engagement clustering with both formulas

A visualization of the groups can be made over the space of the first two principal components. The first cluster explains the elements that Jayasigh (2019) formula and the one provided by Facebook have low interaction and therefore little User Engagement. In the second, we have a User Engagement of 100 to 300 from Facebook, but with the Jayasigh proposal at 50. From the data set obtained for the fan page behaviour, the variables that were used to be classified per type and number of interactions in the calculation of User Engagement were selected. Therefore, it is inferred that the greater interactions and depends content type, the value User Engagement is affected.

Applied qualitative analysis in unstructured data, relationships and groups are extracted that allowed us to take advantage not only of the descriptions obtained but also of the relationships and rules. In this way, qualitative analysis supports the identification of various types of relationships within data interactivity in the Facebook social network. A regression model analysis was performed taking into account interactions and content type as independent variables, resulting in a multiple linear regression model to predict User Engagement, and is defined as follows:

$$\text{Regression model of User Engagement} = 0.56_{Interactions} - 0.29_{Photo} - 1.58_{Text} + 172.18_{Video} + 22.23_{Live} \quad (3)$$

The regression model defined (equation 3) obtained a coefficient of multiple determination ( $R^2$ ) of 0.59 of influence of the independent variables ( $p$ -value of  $2.2e-16$ ). The equation shows that User Engagement is highly influenced by the video type content, no matter if it's live or not. It also indicates that User Engagement decreases for each photo or text content type tends. This is considered highly representative and allows interpreting User Engagement in future publications based on content type and number of interactions. In calculating the value of User Engagement, a fan page with consistency in ranges of User Engagement in its posts is considered to have engaged users (Ballesteros Herencia, 2018). The regression model allowed an interpretation of the impact of the categories by type of content. This is considered highly representative and allows predicting User Engagement in future publications based on content type

## 5. Conclusions and future work

It was possible to create scientific outreach strategies for increasing the visibility of scientific content and improve interest of the educational community, taking advantage of Facebook's versatility. In this work, it is mostly highlighted the User Engagement formula for obtaining metrics of great value because it includes the visualization of content and the interactivity that users have on the Facebook fan page. It is also relevant that there is a positive correlation between the User Engagement a publication receives and the content type, especially in scientific videos created by the case study fan page.

In the analysis and visualization of the variables of each post, clustering was also applied as a data mining technique, demonstrating that self-produced content provides a higher Engagement. And finally, a regression model was establish to User Engagement percentage of a sample of posts about scientific dissemination on a Facebook fan page. The creation of new models for the interpretation of Facebook variables resulted in a proprietary multiple linear regression model that allows to identify User Engagement based on the different categories in each post.

Consequently, it is advisable to use qualitative analysis of Facebook data, methodologies for social network data analysis, and a linear regression model as strategic tools to improve the practices of creation and dissemination of content focused on Science and Engineering topics. In order to highlight the importance of this research an adequate treatment and analysis for social network data, was determined by the type of followers and consumption of the posts created.

As future work, these results will be used to conduct more comparative studies under different periods, where better practices are applied in the posts of fan pages and groups of different undergraduate programs contents and, through these tests, the different user behaviors will be analyzed providing data that generalizes the possibilities of Facebook in Higher Education.

## Acknowledgment

This research was supported by Consejo Nacional de Ciencia y Tecnología (CONACyT) under grant 668484 for the first author.

## References

- Agusriandi, Sukaesih Sitanggang, I., & Hartono Wijaya, S. (2020). *Student Performance Based on Activity Log on Social Network and e-Learning*. 10(6).
- Al-Dheleai, Y. M., & Tasir, Z. (2017). Using facebook for the purpose of students' interaction and its correlation with students' academic performance. *Turkish Online Journal of Educational Technology*, 16(4), 170–178.
- Aviles, J., & Esquivel, R. (2019). Mining Social Media Data of Philippine Higher Education Institutions Using Naïve Bayes Classifier Algorithm. *SSRN*. [https://doi.org/Aviles, Joey and Esquivel, Rosanna, Mining Social Media Data of Philippine Higher Education Institutions Using Naive Bayes Classifier Algorithm \(April 27, 2019\). Available at SSRN: https://ssrn.com/abstract=3379025 or http://dx.doi.org/10.2139/ssrn.3379025](https://doi.org/Aviles, Joey and Esquivel, Rosanna, Mining Social Media Data of Philippine Higher Education Institutions Using Naive Bayes Classifier Algorithm (April 27, 2019). Available at SSRN: https://ssrn.com/abstract=3379025 or http://dx.doi.org/10.2139/ssrn.3379025)
- Ballesteros Herencia, C. A. (2018). El índice de engagement en redes sociales, una medición emergente en la Comunicación académica y organizacional. *RAZÓN Y PALABRA Primera Revista Electrónica En Iberoamérica Especializada En Comunicación*, 22(3\_102), 96–124.
- Bharati, P., Zhang, W., & Chaudhury, A. (2015). Better knowledge with social media? Exploring the roles of social capital and organizational knowledge management. *Journal of Knowledge Management*, 19(3), 456–475. <https://doi.org/10.1108/JKM-11-2014-0467>
- Bonsón, E., & Ratkai, M. (2013). A set of metrics to assess stakeholder engagement and social legitimacy on a corporate Facebook page. *Online Information Review*, 37(5), 787–803. <https://doi.org/10.1108/OIR-03-2012-0054>
- Campos-Freire, F., & Rúas-Araújo, R. (2016). Uso de las redes sociales digitales profesionales y científicas: el caso de las 3 universidades gallegas. *El Profesional de La Información*, 25(3), 431. <https://doi.org/10.3145/epi.2016.may.13>
- Chan-Olmsted, S., Wolter, L. C. , & Wang, R. (2017). Toward a multidimensional framework of media engagement: Conceptualizing consumer experience and connection with media content in a digital environment. *Emma Conf 2017. European Media Management Association.*, 585.
- Chen, X., Madhavan, K., & Vorvoreanu, M. (2013). A web-based tool for collaborative social media data analysis. *Proceedings - 2013 IEEE 3rd International Conference on Cloud and Green Computing, CGC 2013 and 2013 IEEE 3rd International Conference on Social Computing and Its Applications, SCA 2013*, 383–388. <https://doi.org/10.1109/CGC.2013.67>
- Dabbagh, N., & Kitsantas, A. (2012). Personal Learning Environments, social media, and self-regulated learning: A natural formula for connecting formal and informal learning. *Internet and Higher Education*, 15(1), 3–8. <https://doi.org/10.1016/j.iheduc.2011.06.002>
- Eriksson, N., Sjöberg, A., Rosenbröijer, C.-J., & Fagerström, A. (2019). Consumer brand post engagement on Facebook and Instagram-A study of three interior design brands. *International Conference on Electronic Business (ICEB) at AIS Electronic Library (AISeL)*, 12–20. <https://aisel.aisnet.org/iceb2019>
- Franz, D., Marsh, H. E., Chen, J. I., & Teo, A. R. (2019). Using facebook for qualitative research: A brief primer. *Journal of Medical Internet Research*, 21(8), 1–12. <https://doi.org/10.2196/13544>
- Froment, F., García-González, A. J., & Cabero, J. (2022). The relationship of Twitter with teacher credibility and motivation in university students. *Comunicar*, 30(71), 1–12. <https://doi.org/10.3916/C71-2022-10>
- Furtado Avanza, M., & Moreira Pinheiro, W. (2018). Audience Measurement in Press Through Social Interactivity on Facebook in Brazil. *Mediaciones*, 14(20), 48–69. <https://doi.org/10.26620/uniminuto.mediaciones.14.20.2018.48-69>
- Gil-quintana, J., de León, E. V., Osuna-acedo, S., & Marta-lazo, C. (2022). Nano-Influencers Edutubers: Perspective of Centennial Generation Families in Spain. *Media and Communication*, 10(1), 247–258. <https://doi.org/10.17645/mac.v10i1.4760>

- He, Y., Wang, C., & Jiang, C. (2017). Mining Coherent Topics With Pre-Learned Interest Knowledge in Twitter. *IEEE Access*, 5, 10515–10525. <https://doi.org/10.1109/ACCESS.2017.2696558>
- Hu, Y. H., & Chen, K. (2016). Predicting hotel review helpfulness: The impact of review visibility, and interaction between hotel stars and review ratings. *International Journal of Information Management*, 36(6), 929–944. <https://doi.org/10.1016/j.ijinfomgt.2016.06.003>
- Jayasingh, S. (2019). Consumer brand engagement in social networking sites and its effect on brand loyalty. *Cogent Business and Management*, 6(1). <https://doi.org/10.1080/23311975.2019.1698793>
- Karademir Coşkun, T., Erdoğan, F., & Kokoç, M. (2020). Students' Profiles of Information Seeking Behaviors on Facebook: Relations to Personality, Time Management, Gender, and Facebook Use. *International Journal of Technology in Education and Science (IJTES)*, 4(2), 144–159. [www.ijtes.net](http://www.ijtes.net)
- Kaye, L. K. (2021). Exploring the “socialness” of social media. *Computers in Human Behavior Reports*, 3, 100083. <https://doi.org/10.1016/j.chbr.2021.100083>
- Kim, C., & Yang, S. U. (2017). Like, comment, and share on Facebook: How each behavior differs from the other. *Public Relations Review*, 43(2), 441–449. <https://doi.org/10.1016/j.pubrev.2017.02.006>
- Li, G., & De Carvalho, R. M. (2019). Process Mining in Social Media: Applying Object-Centric Behavioral Constraint Models. *IEEE Access*, 7, 84360–84373. <https://doi.org/10.1109/ACCESS.2019.2925105>
- Liyanage, C. R., & Premarathne, S. C. (2021). Clustered Approach for Clone Detection in Social Media. *International Journal on Advanced Science, Engineering and Information Technology*, 11(1), 99–104. <https://doi.org/10.18517/ijaseit.11.1.9272>
- Molina, O. E., Reynaldo, D., & Cancell, F. (2021). Engagement and desertion in MOOCs: Systematic review Engagement and desertion in MOOCs: Systematic review El engagement y la deserción en los MOOCs: Revisión sistemática. *Comunicar, October*, 111–124.
- Oviedo-García, M. Á., Muñoz-Expósito, M., Castellanos-Verdugo, M., & Sancho-Mejías, M. (2014). Metric proposal for customer engagement in Facebook. *Journal of Research in Interactive Marketing*, 8(4), 327–344. <https://doi.org/10.1108/JRIM-05-2014-0028>
- Peruta, A., & Shields, A. B. (2017). Social media in higher education: understanding how colleges and universities use Facebook. *Journal of Marketing for Higher Education*, 27(1), 131–143. <https://doi.org/10.1080/08841241.2016.1212451>
- Phuntusil, N., & Limpiyakorn, Y. (2017). Predicting engaging content for increasing organic reach on facebook. *Lecture Notes in Electrical Engineering*, 424, 637–644. [https://doi.org/10.1007/978-981-10-4154-9\\_73](https://doi.org/10.1007/978-981-10-4154-9_73)
- Pilař, L., Kvasničková Stanislavská, L., Kvasnička, R., Bouda, P., & Pitrová, J. (2021). Framework for social media analysis based on hashtag research. *Applied Sciences (Switzerland)*, 11(8). <https://doi.org/10.3390/app11083697>
- Purba, K. R., Asirvatham, D., & Murugesan, R. K. (2020). An analysis and prediction model of outsiders percentage as a new popularity metric on Instagram. *ICT Express*, 6(3), 243–248. <https://doi.org/10.1016/j.icte.2020.07.001>
- Sababi, M., Marashi, S. A., Pourmajidian, M., Pourtabatabaei, S. S., Darki, F., & Sadrzadeh, M. R. (2017). View of How accessibility influences citation counts\_ The case of citations to the full text articles available from ResearchGate.pdf. *Journal on Research Policy & Evaluation*, 1.
- Savelev, A. O., Karpova, A. Y., Chaykovskiy, D. V., Vilnin, A. D., Kaida, A. Y., Kuznetsov, S. A., Igumnov, L. O., & Maksimova, N. G. (2021). The high-level overview of social media content search engine. *IOP Conference Series: Materials Science and Engineering*, 1019(1). <https://doi.org/10.1088/1757-899X/1019/1/012097>



- Shahbaznezhad, H., Dolan, R., & Rashidirad, M. (2021). The Role of Social Media Content Format and Platform in Users' Engagement Behavior. *Journal of Interactive Marketing*, 53, 47–65. <https://doi.org/10.1016/j.intmar.2020.05.001>
- Shang, S. S. C., Li, E. Y., Wu, Y. L., & Hou, O. C. L. (2011). Understanding Web 2.0 service models: A knowledge-creating perspective. *Information and Management*, 48(4–5), 178–184. <https://doi.org/10.1016/j.im.2011.01.005>
- Steinmetz, C., Rahmat, H., Marshall, N., Bishop, K., Thompson, S., Park, M., Corkery, L., & Tietz, C. (2020). Liking, Tweeting and Posting: An Analysis of Community Engagement through Social Media Platforms. *Urban Policy and Research*, 1–21. <https://doi.org/10.1080/08111146.2020.1792283>
- Stieglitz, S., Mirbabaie, M., Ross, B., & Neuberger, C. (2018). Social media analytics – Challenges in topic discovery, data collection, and data preparation. *International Journal of Information Management*, 39(October 2017), 156–168. <https://doi.org/10.1016/j.ijinfomgt.2017.12.002>
- Surya Gunawan, T., Aleah Jehan Abdullah, N., Kartiwi, M., & Ihsanto, E. (2020). Social Network Analysis using Python Data Mining. *2020 8th International Conference on Cyber and IT Service Management, CITSM 2020*. <https://doi.org/10.1109/CITSM50537.2020.9268866>
- Sutanto, S. M. (2016). *The Use of Facebook Fanpage as Promotional Media for the Novel Book fans by using the Model of SMCR Berlo*.
- Tsinovoi, A. (2020). The management of visibility in digital diplomacy: Infrastructures and techniques. *First Monday*, May. <https://doi.org/10.5210/fm.v25i5.10116>
- Tsvetkov, V. Y. (2021). Qualitative analysis relations. *IOP Conference Series: Materials Science and Engineering*, 1047(1). <https://doi.org/10.1088/1757-899X/1047/1/012127>
- Velazquez-Solis, P. E., Flores-Rios, B. L., Astorga-Vargas, M. A., Ibarra-Esquer, J. E., González-Navarro, F. F., & Aguilar Vera, R. A. (2022). Analysis of scientific dissemination posts on Facebook from a social media approach. In In press (Ed.), *CISTI'2022 - 17ª Conferencia Ibérica de Sistemas y Tecnologías de Información*.
- Voorveld, H. A. M., van Noort, G., Muntinga, D. G., & Bronner, F. (2018). Engagement with Social Media and Social Media Advertising: The Differentiating Role of Platform Type. *Journal of Advertising*, 47(1), 38–54. <https://doi.org/10.1080/00913367.2017.1405754>
- Wheatley, D., & Vatnoey, E. (2020). 'It's Twitter, a bear pit, not a debating society': A qualitative analysis of contrasting attitudes towards social media blocklists. *New Media and Society*, 22(1), 5–25. <https://doi.org/10.1177/1461444819858278>
- Yan, W., & Zhang, Y. (2018). Research universities on the ResearchGate social networking site: An examination of institutional differences, research activity level, and social networks formed. *Journal of Informetrics*, 12(1), 385–400. <https://doi.org/10.1016/j.joi.2017.08.002>

## Authors



**P. Velazquez-Solis** She is a student of Ph.D. at the Universidad Autónoma de Baja California in an Engineering Institute. Her research interests include social media, data mining, data Analysis and she is also consulting in private and public sector companies and dedicated to teaching, giving classes in local universities.



**J. E. Ibarra-Esquer** He is a professor at the Universidad Autónoma de Baja California in the Faculty of Engineering. His research interests include the Internet of Things and Data Mining and has taken part in research projects related to improving the learning processes for students in the Computer Engineering major.



**M. Astorga-Vargas** She is a professor of software process improvement at the Universidad Autónoma de Baja California/ Faculty of Engineering. Her research interests are Software Engineering with a focus in Software Process Improvement in small companies, and the effectiveness in software development teams.



**B. L. Flores-Rios** She is a researcher of knowledge management and software process improvement at the Universidad Autónoma de Baja California/Engineering Institute. Her research interests are Software Process Improvement and quality in small enterprises and the impact of Knowledge Engineering on Software Engineering.



**M. Carrillo-Beltrán** She is a researcher of Food Biotechnology at the Universidad Autónoma de Baja California/ Engineering Institute. Her research interests include innovation ecosystem where emerging technologies are applied for the development, sensory evaluation and marketing of healthier and safer foods and beverages with real benefits and recognizable added values, focusing on human nutrition, sustainability and planetary health.



**Jesús Caro Gutiérrez** He is a researcher attached to the UABC Engineering Institute. He has been recognized as a Candidate for the National System of Researchers and Recognition of the Performance of Academic Staff (PREDEPA-UABC). He is a member of the Mexican Society of Computer Science and his main lines of work are computer vision and data science, applied to the characterization of nanomaterials.



**Raúl Antonio Aguilar Vera** obtained the degree of Doctor from the Polytechnic University of Madrid, Spain and the Master's degree in Software Engineering from the same Institution, he also has the degree of Master in Higher Education from the Autonomous University of Yucatan. He is currently a professor at the Faculty of Mathematics of the Universidad Autónoma de Yucatán. He has been a member since 2014 of the CONAHCYT National Researchers System (Level 1). His research work includes the areas of Software Engineering and Educational Computing.



This work is under a Creative Commons Attribution-NonCommercial-ShareAlike 2.5 Mexico license .

## **Inteligencia Artificial en Sistemas de recomendación Médica: Una revisión de la Literatura.**

### **Artificial Intelligence in Medical Recommendation Systems: A Literature Review.**

Javier A. Calderón-Blas<sup>1</sup>  
javierblas01721@gmail.com  
María Angélica Cerdán<sup>1,2</sup>  
acerdan@uv.mx  
Ángel J. Sánchez-García<sup>1</sup>  
angesanchez@uv.mx  
Saúl Domínguez-Isidro<sup>1</sup>  
sauldominguez@uv.mx

<sup>1</sup>Facultad de Estadística e Informática, Universidad Veracruzana, Xalapa, México

<sup>2</sup>Instituto Tecnológico Superior de Xalapa, Xalapa, México

**Resumen:** En México, aproximadamente una de cada tres personas busca información sobre atención médica a través de Internet, considerando factores como la especialidad, horarios, ubicación, costo, eficacia del tratamiento y calidad del trato médico. Para abordar estas búsquedas, se han desarrollado sistemas de recomendación que ofrecen sugerencias personalizadas basadas en información explícita e implícita del usuario. Estos sistemas utilizan diversas fuentes de datos, incluyendo el Internet de las cosas, para equilibrar la precisión, valor agregado, dispersión y estabilidad en las recomendaciones. A pesar de la amplia aplicación en eSalud, la investigación actual carece de un enfoque específico en los sistemas de recomendación de médicos. Este estudio tiene como objetivo presentar los trabajos más recientes en esta área, analizando enfoques, métodos, técnicas y tendencias a través de una revisión sistemática de la literatura siguiendo el proceso de la ingeniería de software basada en evidencia (EBSE). Los resultados muestran una mayor cantidad de técnicas de recomendación basadas en aprendizaje automático, lógica heurística y combinación de ambos enfoques (híbridos). Finalmente, se identifican áreas de investigación abiertas y tendencias en este campo, que serán de interés para futuras aplicaciones en el ámbito médico.

**Palabras Clave:** eSalud, sistemas de recomendación, Inteligencia Artificial, Software, revisión sistemática de la literatura.

**Abstract:** In Mexico, approximately one in three people search for information about medical care through the Internet, considering factors such as specialty, hours, location, cost, treatment effectiveness, and quality of medical treatment. In order to address these searches, recommender systems have been developed that offer personalized suggestions based on explicit and implicit user information. These systems use various data sources, including the Internet of Things, to balance accuracy, added value, dispersion, and stability in recommendations. Despite the wide application in eHealth, current research needs a specific focus on physician recommendation systems. This study aims to present the most recent works in this area, analyzing approaches, methods, techniques, and trends through a systematic literature review following the evidence-based software engineering (EBSE) process. The results show more recommendation techniques based on machine learning, heuristic logic, and a combination of both approaches (hybrids). Finally, open research areas and trends in this field are identified, which will be of interest for future applications in the medical field.

**Keywords:** eHealth, recommender system, Artificial Intelligence, Software, systematic literature review.

## 1. Introducción

Las Tecnologías de la Información y Comunicaciones, han tenido un gran impacto en todas las áreas del quehacer humano. Su uso para la atención de las necesidades en temas de salud se ha definido como eSalud (OMS, s.f.). Uno de los temas clave dentro de esta área, es el manejo de datos al servicio del consumidor de servicios sanitarios (Priego Álvarez, 2005).

La adquisición de servicios de salud como cualquier proceso de consumo, se ha visto influenciado por el uso masivo de Internet, incluyendo: la evaluación de alternativas, la decisión y el comportamiento post-consumo. En este sentido, más del 70% de las personas, recurren a la web en búsqueda de información sobre la salud y un aproximado del 30%, toma decisiones con base a la información encontrada (Bravo, 2021).

Esta tendencia se reforzó derivado de la pandemia de COVID-19, ya que, ante el confinamiento, las personas vieron en los medios digitales una alternativa para la búsqueda de medicamentos, insumos, servicios de asistencia y servicios médicos (Iglesias Ososres y Saavedra Camacho, 2020). Según (Villafranco, 2019), en México, una de cada tres personas, buscan información sobre atención médica, por Internet o por teléfono.

En la búsqueda de atención médica, no sólo se consideran factores como la especialidad, horarios de atención, ubicación y costo, sino que, a los pacientes o sus cuidadores, también les interesan otros aspectos, como los relativos a la eficacia de su atención y el trato personal que brinda el profesional, lo que ha derivado en la creación de los sitios web para calificación de médicos (Deng et al., 2019).

Para resolver este tipo de búsquedas, se han desarrollado los sistemas de recomendación, mecanismos inteligentes que brindan al usuario sugerencias personalizadas, a través del análisis de la información que se proporciona de manera explícita y de la derivada de la interacción que el usuario tiene con la plataforma de manera implícita, con la finalidad de ofrecer una mejor sugerencia (Ramos, 2019).

Estos sistemas utilizan diferentes fuentes de información, usando datos demográficos, contenido propio de la oferta, obtenido de la colaboración, de las redes sociales y más recientemente, de la adquisición de datos derivada del Internet de las cosas, para realizar las predicción y recomendación, equilibrando factores de precisión, valor agregado, dispersión y estabilidad en las recomendaciones, buscando cumplir las expectativas y requisitos de los usuarios. Generalmente, la mayor parte de los recomendadores aplican enfoques híbridos (Bobadilla et al., 2013).

Dada la amplia aplicación de los sistemas de recomendación en la eSalud en procesos de diagnóstico, pronóstico y tratamiento médico, se han desarrollado análisis de los temas más relevantes en este campo (Calero et al., 2016), sin embargo, no se particulariza en el proceso de recomendación de médicos o se incluye de manera parcial (Trang et al., 2020), sin profundizar en las más recientes aportaciones en particular. Por lo que el presente estudio, tiene por objetivo informar sobre los trabajos más recientes sobre los sistemas de recomendación de médicos, para analizar los enfoques con los que son abordados, identificando los métodos, técnicas, herramientas utilizadas y la evaluación de resultados, para identificar temas abiertos y tendencias, que pueden ser de útiles para los interesados en nuevas o futuras aplicaciones.

Este artículo está estructurado de la siguiente manera. La Sección 2 presenta trabajos relacionados a esta investigación. La Sección 3 describe el método de búsqueda para la selección de estudios primarios. La Sección 4 muestra los resultados y análisis de cada pregunta de investigación. La Sección 5 describe algunas amenazas a la validez de este trabajo. Finalmente, la sección 6 extrae las conclusiones del trabajo y presenta el trabajo futuro.

## 2. Trabajo Relacionado

Se realizaron búsquedas manuales en las que se identificaron artículos que proponen sistemas de recomendación en diversos campos, incluyendo recomendación laboral, elementos de comercio electrónico, rutas de aprendizaje, entre otros. De los estudios centrados en el análisis de sistemas de recomendación encontrados, únicamente dos son revisiones sistemáticas de la literatura (RSL).

En el estudio de Katarya (2017), se llevó a cabo una revisión sistemática de sistemas de recomendación de grupos (GRS por sus siglas en inglés), que ofrecen recomendaciones personalizadas a grupos de usuarios, considerando sus intereses, pasatiempos, entretenimiento, experiencias pasadas y planificación futura. Los autores clasificaron las técnicas de GRS en once categorías de filtrado, incluyendo enfoques basados en contenido, colaborativos, basados en contexto, semántica, comunitarios, etiquetado, análisis de sentimiento, híbridos, demografía, modelos de grupo y filtrado de grupos.

Por otro lado, en el artículo de Patel et al. (2017), se realizó una clasificación de diversas técnicas de recomendación en tres categorías: filtrado basado en contenido, filtrado colaborativo y filtrado híbrido. Se analizaron varios algoritmos en cada una de estas categorías, concluyendo que las técnicas de filtrado colaborativo presentan ventajas significativas en comparación con las otras técnicas. Aunque los enfoques híbridos buscan mejorar la calidad del sistema de recomendación, su implementación se considera compleja y costosa.

En el ámbito médico, se encontró el trabajo de Pincay et al. (2019), que realizó una revisión sobre los Sistemas de Recomendación de Salud (HRS por sus siglas en inglés). Estos sistemas están diseñados para respaldar a los profesionales médicos en la detección de enfermedades y brindar orientación a los pacientes en la promoción de su salud y el mantenimiento de su bienestar. El análisis incluyó artículos publicados entre 2006 y 2018 en cinco bases de datos científicas diferentes. Una conclusión interesante del estudio es que un "porcentaje significativo de las iniciativas empleó técnicas de inteligencia artificial y lógica difusa para lograr una mayor precisión y rendimiento de los sistemas". Por tanto, nuestro trabajo contribuirá a la clasificación de las técnicas de inteligencia artificial utilizadas en este tipo de sistemas.

En Thongchotchat et al. (2021), se llevó a cabo una RSL centrada en el estilo de aprendizaje, que es la preferencia de aprendizaje de un estudiante. El objetivo era identificar técnicas para desarrollar sistemas de aprendizaje con soporte informático que puedan crear un camino de aprendizaje personalizado para cada alumno. Este estudio involucró fuentes de datos como IEEE Xplore y ScienceDirect y encontró que la teoría de Felder & Silverman fue la más utilizada, con 72.5%, y que la aplicación adecuada es el algoritmo de recomendación más utilizado, con 42.5% del total.

Finalmente, en 2022, se publicó la revisión de Gupta y Bindal (2022), que evaluó los sistemas de recomendación multimodales. Los autores concluyeron que se han desarrollado diversas estrategias de recomendación para satisfacer las necesidades en diversas circunstancias, incluyendo sistemas basados en contenido, colaborativos, demográficos y enfoques híbridos. Este artículo también exploró los diferentes desafíos que pueden surgir en estos sistemas, como el arranque en frío, la escasez, la escalabilidad y la diversidad."

Sin embargo, no se encontró ninguna revisión sistemática en el área médica o afines. Este hecho es de gran relevancia, dado que existen diferencias entre recomendar productos y recomendar a un médico, ya que la veracidad de la sugerencia de un buen médico tiene un impacto directo en un diagnóstico preciso y, en consecuencia, en un tratamiento efectivo. Por lo tanto, el objetivo de esta revisión es identificar los componentes clave relacionados con la recomendación en este ámbito, centrándose específicamente en las técnicas de Inteligencia Artificial utilizadas para realizar sugerencias de calidad y cómo se evalúan estos sistemas.

### 3. Método de Investigación

El método utilizado para esta investigación es el propuesto por Kitchenham, et al. (2015), el cual se divide en las etapas de planeación y la conducción. A continuación, se describen los elementos para llevar a cabo esta revisión.

#### 3.1 Etapa de Planeación

En esta etapa, se plantean las preguntas de investigación que guían esta investigación. En la Tabla 1 se presentan las preguntas de investigación junto con su motivación asociada.

<b>Pregunta de investigación</b>	<b>Motivación</b>
PI1. ¿Qué estrategias de Inteligencia Artificial utilizan los sistemas de recomendación médica?	Presentar un panorama general de las técnicas de Inteligencia Artificial que se han utilizado para hacer una recomendación precisa, y que pueden ser utilizadas o modificadas por los desarrolladores de software.
PI2. ¿Cómo se evalúan los sistemas de recomendación médica?	Conocer la manera en que evalúan los sistemas de recomendación, especialmente en el área médica, con el fin de generar propuestas que puedan compararse con los sistemas encontrados en la literatura.
PI3. ¿Cuáles fueron los resultados de la evaluación de los sistemas de recomendación médica?	Conocer los resultados de las evaluaciones de este tipo de sistemas permitirá generar propuestas que puedan mejorar los resultados encontrados en la literatura.
PI4. ¿Qué trabajo futuro se identificó en los estudios seleccionados?	Identificar las propuestas de mejora a los sistemas encontrados en los estudios primarios, con el fin de mejorar los resultados de la literatura.

Tabla 1. Preguntas de Investigación

De las fuentes sugeridas por las guías Kitchenham et al. (2015) y Petersen et al. (2015) se escogieron IEEEExplore, ACM Digital Library, Science Direct y Springer Link, debido a que en estas fuentes se publican artículos sometidos a rigurosos procesos de revisión por pares, lo que garantiza la calidad y validez de los resultados presentados. Además, considerando que el dominio de aplicación es relativa al área médica, fuentes como Science Direct y Springer Link brindan artículos de áreas multidisciplinarias. Las fuentes seleccionadas se pueden visualizar en la Tabla 2.

<b>Fuente</b>	<b>Sitio web</b>
IEEEExplorer	<a href="https://ieeexplore.ieee.org/Xplore/home.jsp">https://ieeexplore.ieee.org/Xplore/home.jsp</a>
ACM Digital Library	<a href="https://dl.acm.org/">https://dl.acm.org/</a>
Science Direct	<a href="https://www.sciencedirect.com/">https://www.sciencedirect.com/</a>
Springer Link	<a href="https://link.springer.com/">https://link.springer.com/</a>

Tabla 2. Fuentes de información seleccionadas



Para la generación de la cadena de búsqueda, se identificaron los términos claves para la búsqueda de artículos. Los términos definidos para esta RSL se muestran en la Tabla 3.

Término clave	Término en inglés	Término relacionado
Sistema	System	-
Doctor	Doctor	-
Recomendador médico	Medical recommender	Medical assistance Medical services Healthcare
Paciente	Patient	-

Tabla 3. Términos clave de búsqueda

La cadena de búsqueda representa la entrada para recuperar estudios relevantes de las fuentes de datos mostradas en la Tabla 2. Se generaron diversas cadenas de búsqueda a partir de las palabras de la Tabla 3 enlazados con operadores booleanos (OR y AND), y se seleccionó aquella que obtuvo un mejor desempeño, en términos de *precision* y *recall*.

La métrica *precision* es la proporción de documentos recuperados que son relevantes para la consulta, véase la ecuación (1); por su parte, la métrica *recall* es la proporción de documentos relevantes que se recuperan con éxito, véase la ecuación (2).

$$Precision = \frac{\text{Documentos relevantes recuperados}}{\text{Total de documentos recuperados}} \quad (1)$$

$$Recall = \frac{\text{Documentos relevantes recuperados}}{\text{Total de documentos relevantes}} \quad (2)$$

Se seleccionó una cadena de búsqueda diferente para cada fuente de datos. La Tabla 4 muestra la cadena de búsqueda aplicada para cada fuente, con su respectiva medida de *recall* y *precision*. Como se puede observar en la Tabla 4, en IEEEExplore y Science Direct se alcanzó un *recall* de 0.8 (sugerido por Kitchenham et al. (2015)), mientras que para Springer y ACM Digital Library los valores más altos obtenidos fueron inferiores a 0.6, pero se tomó la cadena de búsqueda con mayor valor de *recall*, y como segundo criterio el valor de precisión más alto.

Fuente	Cadena de búsqueda	Valor de <i>recall</i>	Valor de <i>precision</i>
IEEEExplore	"Recommender system" AND (Doctor OR Patient OR Medical Assistance)	0.83	0.26
ACM Digital Library	(Doctor OR Medical Assistance) AND "Recommender system"	0.59	0.14
Science Direct	"Recommendation system" AND (Doctor OR "Medical services" OR "Medical Assistance" OR Healthcare)	0.8	0.22
Springer Link	"Recommender system" AND (Doctor OR Medical services OR Medical Assistance)	0.56	0.1

Tabla 4. Cadenas de búsqueda aplicadas por fuente de datos

### 3.2 Etapa de Conducción

En este apartado se describen los criterios de selección de los artículos encontrados con las cadenas de búsqueda y se explica el proceso de selección de los estudios primarios que serán analizados.

#### 3.2.1 Criterios de selección

Se propusieron cinco criterios de inclusión (CI) y dos criterios de exclusión CE, para seleccionar los estudios primarios de los estudios recuperados en las fuentes seleccionadas, véase Tablas 5 y 6, respectivamente.

Clave	Descripción
CI1	Artículos publicados entre 2016 y 2023.
CI2	Publicaciones escritas en idioma inglés.
CI3	Artículos con acceso completo.
CI4	Al leer el título y el resumen, el artículo da indicios de estar relacionado con los sistemas de recomendación médica.
CI5	Al leer el contenido de la publicación, da indicios de responder al menos una pregunta de investigación.

Tabla 5. Criterios de inclusión

Clave	Descripción
CE1	El artículo es un resumen, póster o capítulo de libro.
CE2	Investigaciones duplicadas.

Tabla 6. Criterios de exclusión

#### 3.2.2 Proceso de selección

Se definieron cuatro etapas para el proceso de selección de estudios. La Figura 1 muestra los criterios aplicados en cada etapa.

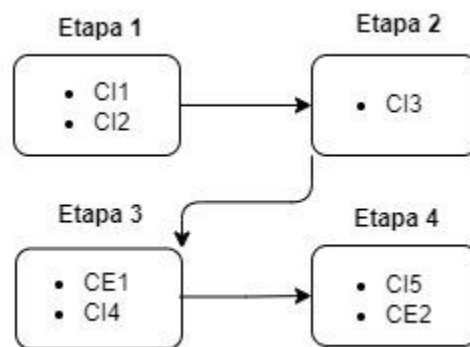


Figura 1. Proceso de selección de estudios primarios.

La Tabla 7 muestra el número de estudios primarios seleccionados por etapa, donde se puede observar que IEEEExplore seleccionó la mayor cantidad de estudios. Asimismo, la etapa 2 fue en la que se descartó la mayor cantidad de artículos. Finalmente, no se seleccionó ningún estudio de la biblioteca digital ACM Digital Library ni de Springer Link.

Fuente	Resultados	Etapa 1	Etapa 2	Etapa 3	Etapa 4
IEEEExplore	90	61	61	9	7
ACM Digital Library	2212	43	3	0	0
Science Direct	1558	882	32	2	2
Springer Link	1957	218	81	4	0
<b>Total</b>	<b>5817</b>	<b>1204</b>	<b>177</b>	<b>15</b>	<b>9</b>

Tabla 7. Número de estudios primarios seleccionados por etapa.

### 3.2.3 Proceso de Snowballing

Debido al escaso número de estudios seleccionados, se decidió aplicar una iteración del proceso de bola de nieve (snowballing) hacia adelante y una iteración hacia atrás como se muestra en la Figura 2.

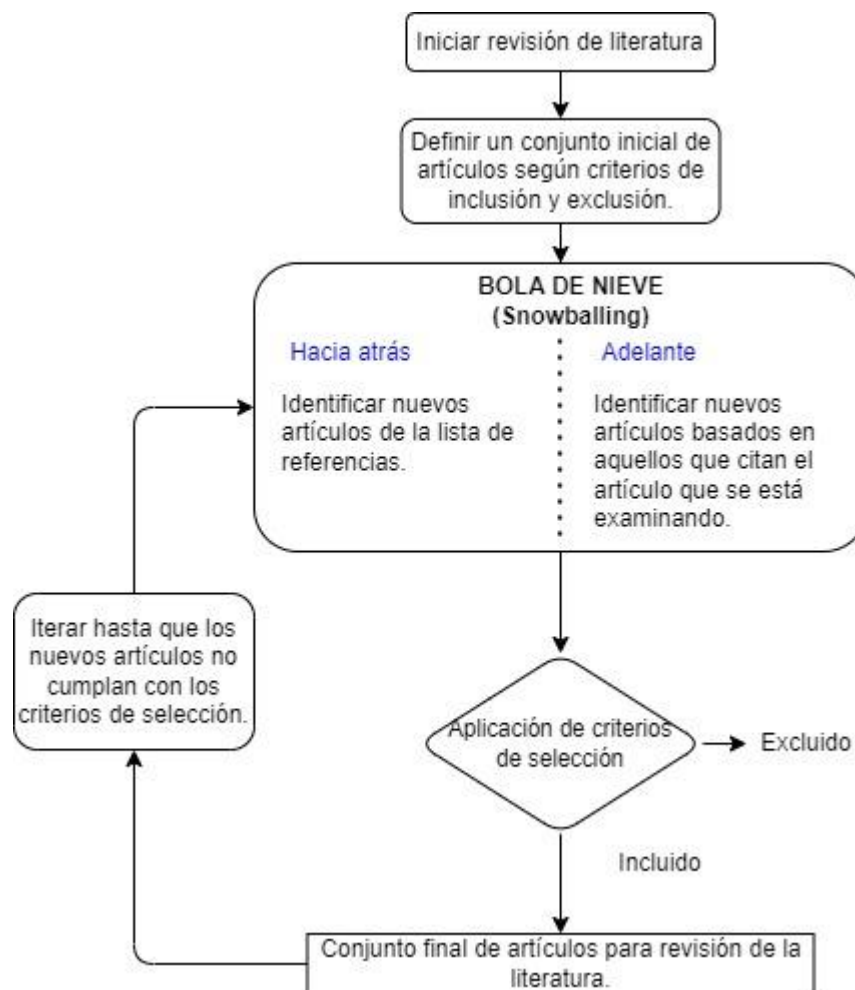


Figura 2. Proceso de bola de nieve (Snowballing).

Después de aplicar el proceso de bola de nieve, se agregaron tres estudios primarios más de dos fuentes diferentes que cumplieron con los criterios de selección, como se muestra en la Tabla 8.

Fuente	Sitio web	Número de estudios seleccionados
IOPscience	<a href="https://iopscience.iop.org/">https://iopscience.iop.org/</a>	2
JMIR Publications	<a href="https://www.jmir.org/">https://www.jmir.org/</a>	1

Tabla 8. Fuentes de datos agregadas después del proceso de snowballing.

#### 4. Resultados

La distribución final de los estudios seleccionados por fuente se muestra en la Figura 3, donde IEEEExplore, con siete publicaciones de artículos, es la fuente seleccionada con la mayor cantidad.

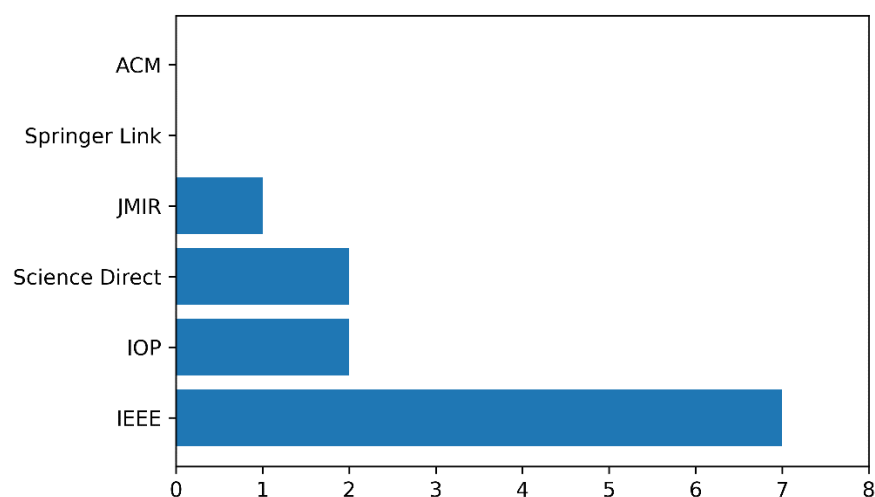


Figura 3. Número final de estudios primarios seleccionados por fuente.

La proporción de los estudios seleccionados por tipo de publicación se muestra en la Figura 4, siendo el mismo número de artículos de revistas y congresos. Por otra parte, en la Figura 5 se muestra la distribución de los estudios por año entre 2016 y 2023, se puede observar que 2018 y 2020 fueron los años con más artículos publicados en relación con los sistemas de recomendación médica. Los estudios primarios seleccionados se muestran en la Tabla 9 agrupados por fuente de información.

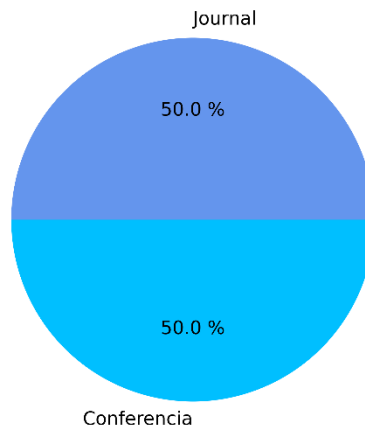


Figura 4. Proporción de estudios por tipo de publicación.

Fuente	Referencia
IEEEExplore	(Han et al.,2018), (Han et al.,2018b), (Bjansali & Nagwani, 2021) (Venkatesh et al., 2022), (Singh et al., 2016), (Swarnalatha et al., 2019), (Meng et al., 2019)
IOP	(Mawardi et al., 2020), (Kannan et al., 2020)
Science Direct	(Mondal et al., 2020), (Zhang et al., 2017)
JMIR	(Guo et al., 2016)

Tabla 9. Estudios primarios seleccionados

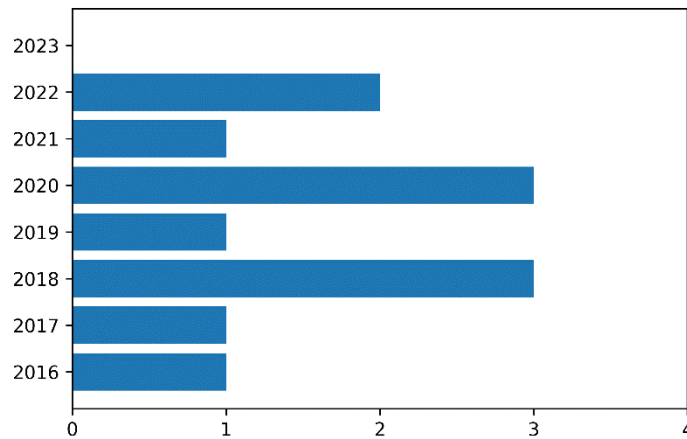


Figura 5. Distribución de año de publicación en el periodo 2016-2023.

Para el proceso de síntesis de la información se optó por la estrategia de síntesis narrativa. Este tipo de síntesis descrita por Popay et al. (2006) implica integrar los hallazgos de múltiples estudios en una narrativa coherente que resuma la evidencia disponible sobre nuestro tema de interés. A continuación, se presentan las respuestas a cada pregunta de investigación.

#### 4.1 PI1. ¿Qué estrategias de Inteligencia Artificial utilizan los sistemas de recomendación médica?

Un total de 19 técnicas o algoritmos fueron identificados en los 12 estudios primarios. La Figura 6 presenta las diferentes técnicas encontradas para la mejora de la precisión en las recomendaciones de los sistemas de recomendación médica (SRM), siendo el Filtrado colaborativo la más recurrente, seguido de los algoritmos de clasificación C4.5 y Naive Bayes y un modelo híbrido. Cabe mencionar que algunos estudios primarios, al proponer experimentos para una comparación de entre técnicas para recomendación, abordaron más de una técnica.

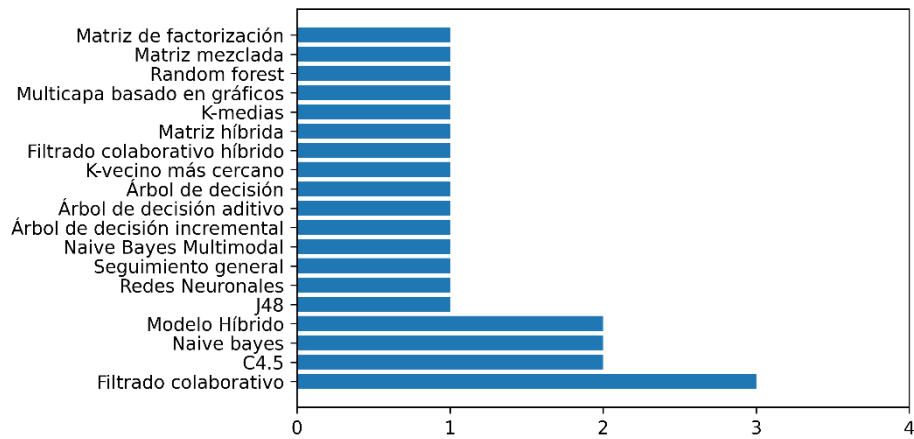


Figura 6. Distribución de técnicas encontradas para la recomendación médica.

Se definieron y agruparon las técnicas reportadas para hacer recomendaciones precisas mediante sistemas de recomendación médica en seis estrategias o enfoques. La Tabla 10 muestra la agrupación de las técnicas de la Figura 6 en seis enfoques.

<b>Enfoque</b>	<b>Técnicas</b>
Aprendizaje máquina	Aprendizaje supervisado: <ul style="list-style-type: none"> <li>• Árboles de decisión (C4.5, incremental, aditivo, J48)</li> <li>• Modelos Bayesianos (Naive Bayes)</li> <li>• K-Vecino más cercano (K-Nearest Neighbor)</li> <li>• Clasificadores ensamblados (Random Forest)</li> <li>• Redes Neuronales</li> </ul>
Lógica Heurística	Aprendizaje No supervisado <ul style="list-style-type: none"> <li>• K-medias (K-means)</li> </ul>
Híbrido	<ul style="list-style-type: none"> <li>• Filtrado colaborativo</li> <li>• Filtrado colaborativo híbrido</li> <li>• Filtrado colaborativo basado en reglas</li> </ul>
Matricial	<ul style="list-style-type: none"> <li>• Basado en contenido y filtrado colaborativo</li> <li>• Filtrado colaborativo e ICD-9</li> <li>• Filtrado colaborativo, K-Vecino más cercano y árboles de decisión</li> </ul>
Basado en reglas	<ul style="list-style-type: none"> <li>• Matriz de factorización (Factorization matrix)</li> <li>• Matriz mezclada (Mixed matrix)</li> <li>• Matriz Híbrida (Hybrid matrix)</li> </ul>
Procesamiento de Lenguaje Natural	<ul style="list-style-type: none"> <li>• Basado en reglas</li> <li>• Revisiones de usuarios</li> </ul>

Tabla 10. Técnicas agrupadas por enfoque.

El aprendizaje automático, una rama de la Inteligencia Artificial, permite a las máquinas adquirir conocimiento sin necesidad de ser programadas explícitamente para ello. Esta disciplina se subdivide en tres categorías principales: aprendizaje supervisado, aprendizaje no supervisado y semi-supervisado. Sin embargo, en los estudios revisados, se han abordado únicamente las dos primeras categorías.

En el aprendizaje supervisado, los algoritmos aprenden a partir de un conjunto de datos de entrenamiento, lo que requiere la intervención humana para proporcionar un resultado esperado. Por otro lado, en el aprendizaje no supervisado, los algoritmos aprenden de los datos buscando patrones o relaciones entre ellos. En este tipo de enfoque, los algoritmos no dependen de la intervención humana y tienen la capacidad de agrupar registros con características similares. En la Tabla 10 los algoritmos de aprendizaje máquina fueron agrupados por tipo (supervisado y no supervisado).

La lógica heurística (Burgos et al., 2007) es un procedimiento para el cual existe un alto grado de confianza que se clasifica como método heurístico, encontrando soluciones de acuerdo con la información disponible. En el razonamiento basado en reglas, la base de conocimiento inicial se basa en variables y un conjunto de reglas compuestas por una premisa y una conclusión, definidas para el problema y cómo se puede resolver (Moya-Rodríguez et al., 2012). El procesamiento del lenguaje natural mencionado por (Russell & Norving., 2004), es otra disciplina de la Inteligencia Artificial que se ocupa de la formulación e investigación de mecanismos para la comunicación humano-máquina mediante el uso del lenguaje natural, ya sea escrito, hablado o interactuado físicamente.

Por otro lado, los enfoques matriciales se representan a través de gráficos, representación basada en una indexación bidimensional de los nodos (Ponz et al., 2011). Finalmente, los modelos híbridos están formados por la integración de varios enfoques, donde cada uno tiene una representación y sigue un proceso diferente, pero cuando el modelado finaliza, éste tiende a generar una solución casi perfecta para brindar sugerencias al usuario, abonando con las ventajas de cada enfoque que forma el modelo híbrido.

La Figura 7 presenta la distribución de estudios primarios que tratan distintos enfoques definidos. Es importante señalar que algunos de estos estudios primarios, al diseñar experimentos para comparar técnicas de recomendación, abordaron más de un enfoque en sus investigaciones.

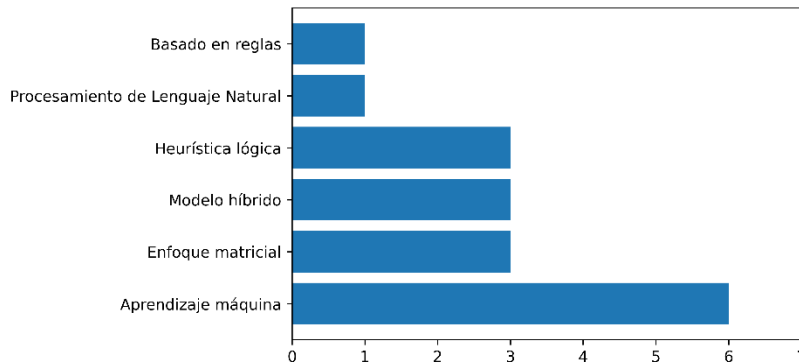


Figura 7. Distribución de estudios primarios que abordan los diferentes enfoques en la recomendación médica.

Como se observa en la Figura 7, el enfoque más frecuentemente mencionado en los estudios primarios es el uso del aprendizaje automático. Esto se debe a que se considera que no es muy complejo de implementar y que el entrenamiento proporcionado puede lograr una precisión adecuada en las recomendaciones, como se destacó en el estudio de Zhang et al. (2017). La implementación de técnicas de aprendizaje automático se ha centrado en la optimización del proceso de recomendación, particularmente en la reducción del consumo de recursos.

El segundo enfoque más utilizado es la lógica heurística. A pesar de no ser tan riguroso como otros enfoques, ha demostrado ofrecer un buen rendimiento en las sugerencias proporcionadas.

A continuación, se encuentran los modelos híbridos, que se distinguen por la integración de dos o más algoritmos que trabajan en conjunto para procesar la información del paciente y brindar recomendaciones altamente personalizadas.

Finalmente, se mencionan otros enfoques, como los enfoques matriciales, basados en reglas y de procesamiento del lenguaje natural. Estos enfoques son viables según la literatura, ya que ofrecen recomendaciones altamente personalizadas para los pacientes. Sin embargo, se ha observado que pueden tener un impacto negativo en el rendimiento del sistema y requerir un consumo significativo de recursos para su operación.



#### 4.2 PI2. ¿Cómo se evalúan los sistemas de recomendación médica?

Tal como se estableció en la planificación de esta investigación, el objetivo de esta segunda pregunta es identificar la forma de evaluar el desempeño de los sistemas de recomendación médica. Para este tipo de sistemas, se han reportado diversas métricas de evaluación, siendo la métrica de precisión (*precision*) la más destacada. La precisión mide la proporción de verdaderos positivos en relación con el número total de valores positivos predichos, permitiendo así evaluar la calidad del modelo en tareas de clasificación.

Otra métrica importante es la recuperación (*recall*). Esta métrica proporciona información sobre la capacidad del modelo para identificar correctamente la cantidad de registros de interés. En otras palabras, evalúa el porcentaje de usuarios que están interesados en un servicio específico y cuántas recomendaciones acertadas se logran en este sentido.

Otra métrica de rendimiento destacada es la Raíz del Error Cuadrático Medio (*RMSE* por sus siglas en inglés), que calcula el valor medio de la diferencia. Se utiliza principalmente cuando se trata de aprendizaje automático supervisado. Cuanto menor sea el *RMSE*, más precisas serán las predicciones. Un *RMSE* igual a 1 es el valor óptimo en el que los pronósticos tienen una precisión excepcional. El *RMSE* tiene ventajas sobre la métrica de error absoluto medio, ya que destaca los errores grandes, entre los pequeños, destacando los valores atípicos que dan importancia a los datos de salida.

La Tabla 11 detalla cada métrica de evaluación encontrada en los estudios primarios seleccionados.

Métrica	Referencias
Precisión (Precision)	(Han et al.,2018)(Bjansali & Nagwani, 2021)(Swarnalatha et al., 2019)
Recuperación (Recall)	(Swarnalatha et al., 2019)(Meng et al., 2019)
Root Mean Square Error (RMSE)	(Meng et al., 2019)(Zhang et al., 2017)
F-Measure	(Bjansali & Nagwani, 2021)(Swarnalatha et al., 2019)
Métrica propia	(Mawardi et al., 2020)
Área bajo la curva (AUC)	(Meng et al., 2019)
Mean Absolute Error (MAE)	(Meng et al., 2019)
Proporción de Falsos positivos (FPR)	(Swarnalatha et al., 2019)
Accuracy	(Swarnalatha et al., 2019)
Hit Ratio@n	(Han et al.,2018)

Tabla 11. Métricas de evaluación reportadas por los estudios primarios seleccionados.

#### 4.3 PI3. ¿Cuáles fueron los resultados de la evaluación de los sistemas de recomendación médica?

Con las técnicas de Aprendizaje Máquina encontradas y las métricas de evaluación reportadas, es importante conocer los resultados de sus evaluaciones, con el fin de facilitar la toma de decisiones en futuras implementaciones de sistemas de recomendación. Para ello aplica la síntesis narrativa para describir los diferentes resultados y conclusiones de los estudios.

Entre los hallazgos más importantes, podemos mencionar que el filtrado colaborativo y un modelo de confianza basado en el historial de consultas que tiene un paciente con el médico, obtienen márgenes considerablemente buenos a través de un modelo heurístico, en la medición de confianza y recomendando un 3% más de pacientes con médicos relevantes para ellos (Han et al.,2018).

El modelo de matriz de factorización mixta combinado con un proceso de selección basado en reglas funcionó mejor que simplemente usar el filtrado colaborativo. La aplicación de la codificación ICD-9 ayudó al rendimiento del modelo (Han et al., 2018b).

El clasificador Random Forest supera a los clasificadores C4.5 y Naïve Bayes en cuanto a su precisión. Los algoritmos de clasificación C4.5 Naïve Bayes superaron al algoritmo J48 (Bjansali & Nagwani, 2021). El factor de confianza radica en qué tan familiarizado está un paciente con un médico (Mondal et al., 2020).

El *RMSE* obtenido de las matrices de factorización híbridas es significativamente bajo, lo que demuestra que proporciona una precisión adecuada en los recomendadores médicos (Zhang et al., 2017). El mejor método entre C4.5 y K-Vecino más cercano es el clasificador K-Vecino más cercano, que reporta una precisión del 85% (Mawardi et al., 2020). Un enfoque bayesiano tiende a ser el clasificador más adecuado para un sistema de recomendación médico, ya que proporciona una precisión casi perfecta. Sin embargo, este método requiere que el usuario ingrese suficiente información para que el clasificador pueda ser entrenado y tener los al menos un caso para cada valor de clase (Kannan et al., 2020). Finalmente, las valoraciones de los pacientes pueden ser extremas, es decir, satisfechos o insatisfechos según la escala de valoración. Este tipo de evaluaciones afectan las recomendaciones que se brindan (Guo et al., 2016).

#### 4.4 PI4. ¿Qué trabajo futuro se identificó en los estudios seleccionados?

En esta cuarta y última pregunta de investigación, el objetivo fue identificar las oportunidades reportadas por los estudios encontrados sobre Sistemas de recomendación médica. A continuación, se describen los principales trabajos o mejoras futuras que se proponen.

En el estudio de Han et al. (2018a), se propone la expansión del modelo de filtrado colaborativo mediante la incorporación de información auxiliar, como metadatos médicos y de pacientes, transformándolo en un enfoque híbrido. Además, se plantea la implementación del sistema de recomendación en un entorno de producción y prueba, con el objetivo de obtener comentarios explícitos de los pacientes como retroalimentación.

En Han et al. (2018b), se sugiere la implementación de una función de emparejamiento dentro del sistema de recomendación, diseñada según las necesidades y preferencias de los pacientes, considerándola una característica clave en los servicios de salud digitales. También se propone evaluar la utilidad de las recomendaciones mediante encuestas controladas aleatoriamente y su integración en el sistema para obtener retroalimentación implícita, lo que contribuiría a la personalización de las sugerencias dentro del modelo. En Bhansali et al. (2021), se propone comparar y evaluar varios algoritmos de clasificación.

En Venkatesh et al. (2022), se destaca la importancia de evaluar el uso de la tecnología big data para generar recomendaciones de alimentos y medicamentos para los usuarios. Asimismo, se considera mejorar la eficiencia de los registros médicos personales y la adición de más algoritmos de recomendación en este estudio.

El trabajo de Singh et al. (2016) propone la ampliación de la base de datos de médicos en su sistema, permitiendo a los usuarios programar citas con los médicos, no limitándose a proporcionar recomendaciones, sino también facilitando el contacto con los profesionales. Además, se plantea la introducción de una función que permita a los usuarios compartir sus experiencias con los médicos y proporcionar comentarios sobre la eficacia de sus consultas.

En Zhang et al. (2017), se describe una línea de trabajo abierta que permite que las preferencias de los usuarios cambien con el tiempo y que importe datos de las redes sociales, lo que podría mejorar aún más la precisión de las recomendaciones.

Swarnalatha et al. (2019) propone la integración de redes heterogéneas y redes neuronales para mejorar el rendimiento de las recomendaciones. En Kannan et al. (2020), se plantea el desarrollo de un conjunto de datos que incluya información sobre enfermedades para relacionar a los médicos especialistas con esas enfermedades.

Además, en el mismo trabajo de Kannan et al. (2020), los autores se enfocan en implementar un modelo de operación por voz para permitir que el sistema se adapte al usuario. También experimentan con la adaptación de sugerencias personalizadas según la ubicación del paciente. Por último, en Guo et al. (2016), se propone incorporar un enfoque de agregación de clasificación no supervisada.

#### 5. Amenazas a la validez

Es importante mencionar que existen algunas amenazas a la validez de este trabajo. En primer lugar, es relevante mencionar que los autores no son expertos en el campo médico. No obstante, la investigación se enfocó en el análisis de técnicas de Inteligencia Artificial que permiten una recomendación más precisa en el ámbito médico, especialmente en lo que respecta a la selección de médicos, al diseñar e implementar un software con estas características.

En segundo lugar, es importante tener en cuenta el número de trabajos obtenidos con los criterios de selección inicialmente propuestos. No obstante, se realizó un esfuerzo para abordar esta limitación mediante un proceso de búsqueda de "bola de nieve" hacia adelante y hacia atrás, lo que permitió la inclusión de otros estudios primarios incluso desde fuentes de datos adicionales, con el fin de ampliar la base de evidencia y reducir esta preocupación.

#### 6. Conclusiones y trabajo futuro

Este artículo presentó una Revisión Sistemática de la Literatura, basada en la guía propuesta por por Kitchenham, et al. (2015), con el fin de identificar técnicas de Inteligencia Artificial y su desempeño, que apoyan a los ingenieros de software en la toma de decisiones a implementar en sistemas de recomendación, específicamente por recomendación de los médicos. En este trabajo, se seleccionaron 12 estudios primarios de seis fuentes de datos.

Como resultado de esta investigación se identificaron técnicas, que se clasificaron en: aprendizaje automático (aprendizaje supervisado y aprendizaje no supervisado), lógica heurística, basado en reglas, procesamiento del lenguaje natural, enfoque matricial y modelos híbridos.

Se identificaron diferentes métricas de evaluación en MRS, que fueron: Precisión, Recuperación, Error cuadrático medio (RMSE), Índice de aciertos, Medida f1, Precisión, Puntuación F, Tasa de falsos positivos, Error absoluto medio (MAE), Área bajo la curva (AUC) y cálculos manuales (Síntomas).

Además, se han identificado varias áreas para futuras investigaciones, siendo una de las más destacadas la inclusión de un mayor volumen de datos en el sistema. Esto podría ampliar la base de información disponible y enriquecer las recomendaciones. También se plantea la adición de funcionalidades adicionales para mejorar la experiencia del usuario y la eficacia del sistema.

Asimismo, se sugiere explorar la integración de más algoritmos y observar su comportamiento en el sistema, con el propósito de determinar su factibilidad y evaluar si pueden brindar resultados más sólidos y precisos.

Finalmente, se propone dar continuidad a esta investigación, explorando las oportunidades que ofrece la literatura para mejorar las recomendaciones. Esto podría incluir el uso de algoritmos ensamblados, una estrategia que rara vez se ha abordado en la literatura, o la implementación de enfoques híbridos que aprovechen las ventajas de múltiples técnicas para lograr recomendaciones más efectivas.

## Referencias

- [OMS] Organización Mundial de la Salud (s.f.). Observatorio Global para la eSalud. <https://www.who.int/observatories/global-observatory-for-ehealth>
- Bhansali, A., & Nagwani, N. K. (2021, May). A Prototype of Doctor Recommendation System Using Classification Algorithms. In 2021 Emerging Trends in Industry 4.0 (ETI 4.0) (pp. 1-4). IEEE.
- Bobadilla, J., Ortega, F. Hernando, A. y Gutiérrez, A. (2013). Recommender systems survey. *Knowledge-Based Systems*, 46. Pp. 109-132.
- Bravo, J. (2021, 7 mayo). Doctor Google y la salud digital. *El Economista*. <https://www.economista.com.mx/opinion/Doctor-Google-y-la-salud-digital20210507-0041.html>
- Burgos, D., Herder, E., & Olmedilla, D. (2007). Ten Competence: Building the European Network for the continuous development of competences. *Inteligencia Artif.*, 11(33), 79-84.
- Calero Valdez, A., Ziefle, M., Verbert, K., Felfernig, A. y Holzinger, A. (2016). Recommender Systems for Health Informatics: State-of-the-Art and Future Perspectives. *ML for Health Informatics, LNAI 9605*, pp. 391-414.
- Deng, Z., Hong, Z., Zhang, W., Evans, R. y Chen, Y. (2019). The Effect of Online Effort and Reputation of Physicians on Patients' Choice: 3-Wave Data Analysis of China's Good Doctor Website. *Journal of medical internet research*, 21(3), p. 1.
- Guo, L., Jin, B., Yao, C., Yang, H., Huang, D., & Wang, F. (2016). Which doctor to trust: a recommender system for identifying the right doctors. *Journal of medical Internet research*, 18(7), e6015.
- Gupta, S., & Bindal, A. K. (2022, November). Multi-Modality Recommender Systems: A Review. In *2022 Seventh International Conference on Parallel, Distributed and Grid Computing (PDGC)* (pp. 88-93). IEEE.
- Han, Q., de Troya, I. M. D. R., Ji, M., Gaur, M., & Zejnilovic, L. (2018, June). A collaborative filtering recommender system in primary care: Towards a trusting patient-doctor relationship. In 2018 IEEE International Conference on Healthcare Informatics (ICHI) (pp. 377-379). IEEE.
- Han, Q., Ji, M., Martinez De Rituerto De Troya, I., Gaur, M., & Zejnilovic, L. (2018). A hybrid recommender system for patient-doctor matchmaking in primary care. *Proceedings-2018 IEEE 5th International Conference on Data Science and Advanced Analytics, DSAA 2018*, 481-490.
- Iglesias Osos, S. y Saavedra Camacho, J. L. (2020). Tendencias de búsquedas en internet por la pandemia COVID-19 en Perú. *Revista Cubana de Higiene y Epidemiología*, 57.
- Kannan, R. J., Tamakuwala, H., Kale, S., & Bhowmick, H. R. (2020, September). Doctor Finder: Find doctors on the Go. In *IOP Conference Series: Materials Science and Engineering* (Vol. 925, No. 1, p. 012038). IOP Publishing.
- Katarya, R. (2017, December). A systematic review of group recommender systems techniques. In *2017 International conference on intelligent sustainable systems (ICISS)* (pp. 425-428). IEEE.
- Kitchenham, B. A., Budgen, D., & Brereton, P. (2015). *Evidence-based software engineering and systematic reviews* (Vol. 4). CRC press.
- Mawardi, V. C., & Naga, D. S. (2020, December). Application of Recommendation Medical Specialty Doctors Based on User Symptoms Using the C4. 5 Method and K-Nearest Neighbor. In *IOP Conference Series: Materials Science and Engineering* (Vol. 1007, No. 1, p. 012152). IOP Publishing

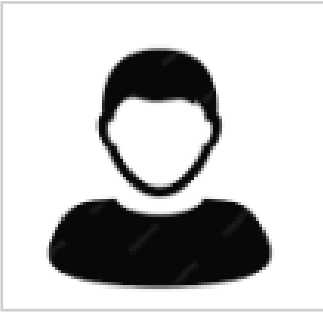
- Meng, S., Fan, S., Li, Q., Wang, X., Zhang, J., Xu, X., ... & Bhuiyan, M. Z. A. (2022). Privacy-aware factorization-based hybrid recommendation method for healthcare services. *IEEE Transactions on Industrial Informatics*, 18(8), 5637-5647.
- Mondal, S., Basu, A., & Mukherjee, N. (2020). Building a trust-based doctor recommendation system on top of multilayer graph database. *Journal of Biomedical Informatics*, 110, 103549.
- Moya-Rodríguez, J. L., Becerra-Ferreiro, A. M., & Chagoyén-Méndez, C. A. (2012). Utilización de Sistemas Basados en Reglas y en Casos para diseñar transmisiones por tornillo sinfín. *Ingeniería Mecánica*, 15(1), 01-09.
- Patel, B., Desai, P., & Panchal, U. (2017, March). Methods of recommender system: A review. In *2017 international conference on innovations in information, embedded and communication systems (ICIIECS)* (pp. 1-4). IEEE.
- Pincay, J., Terán, L., & Portmann, E. (2019, April). Health recommender systems: a state-of-the-art review. In *2019 Sixth International Conference on eDemocracy & eGovernment (ICEDEG)* (pp. 47-55). IEEE.
- Petersen, K., Vakkalanka, S., & Kuzniarz, L. (2015). Guidelines for conducting systematic mapping studies in software engineering: An update. *Information and software technology*, 64, 1-18.
- Popay, J., Roberts, H., Sowden, A., Petticrew, M., Arai, L., Rodgers, M., ... & Duffy, S. (2006). Guidance on the conduct of narrative synthesis in systematic reviews. A product from the ESRC methods programme Version, 1(1), b92.
- Ponz Tienda, J. L., Benloch Marco, J., Andrés Romano, C., & SENABRE, D. (2011). Un algoritmo matricial "RUPSP/GRUPSP" sin interrupción" para la planificación de la producción bajo metodología Lean Construction basado en procesos productivos. *Revista de la Construcción*, 10(2), 90-103.
- Priego Álvarez, H. R. (2005). Consumo en salud. Análisis mercadológico del comportamiento del consumidor sanitario. [Tesis Doctoral]. Universidad Autónoma de Barcelona.
- Ramos, O. (2019, 2 diciembre). Sistemas de recomendación | Qué son, tipos y ejemplos. GraphEverywhere. <https://www.grapheverywhere.com/sistemas-de-recomendacion-que-son-tipos-yejemplos/>
- Russell, S. J., & Norvig, P. (2004). *Inteligencia Artificial: un enfoque moderno* (No. 04; Q335, R8y 2004.).
- Singh, A., Kaur, I. G. P., & Dabas, C. (2018, August). Get-a-Doc: A Doctor Recommender System. In *2018 7th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO)* (pp. 219-223). IEEE.
- Swarnalatha, S., Kesavarthini, I., Poornima, S., & Sripriya, N. (2019, February). Med-recommender system for predictive analysis of hospitals and doctors. In *2019 International Conference on Computational Intelligence in Data Science (ICCIDS)* (pp. 1-5). IEEE.
- Thongchotchat, V., Sato, K., & Suto, H. (2021, May). Recommender System Utilizing Learning Style: Systematic Literature Review. In *2021 6th International Conference on Business and Industrial Research (ICBIR)* (pp. 184-187). IEEE.
- Trang Tran, T. N., Felfernig, A., Trattner, Ch. y Holzinger, A. (2020). Recommender systems in the healthcare domain: state-of-the-art and research issues. *Journal of Intelligent Information Systems*. (2021)57, pp. 171-201.

Venkatesh, B. H., Sai, A. P., Reddy, M. R., & Fathimabi, S. K. (2022, June). Cloud Based Personal Health Record Management System and Medical Recommender System. In 2022 7th International Conference on Communication and Electronics Systems (ICCES) (pp. 1744-1749). IEEE.

Villafranco, G. (2019, 19 abril). Esta plataforma te ayuda a encontrar al médico especialista que necesitas. <https://www.forbes.com.mx/esta-plataforma-te-ayuda-a-encontrar-al-medico-especialista-que-necesitas/>

Zhang, Y., Chen, M., Huang, D., Wu, D., & Li, Y. (2017). iDoctor: Personalized and professionalized medical recommendations based on hybrid matrix factorization. *Future Generation Computer Systems*, 66, 30-35.

## Notas Biográficas



Javier Alberto Calderón-Blas es estudiante de la Licenciatura en Ingeniería de Software de la Facultad de Estadística e Informática de la Universidad Veracruzana. Sus intereses se centran en la Inteligencia Artificial, Ciencia de Datos y UI/UX.



María Angélica Cerdán, Maestra en Ciencias de la Computación, docente de la Universidad Veracruzana Facultad de Estadística e Informática, Profesor con Perfil Deseable. Jefa de la División de la Ingeniería en Sistemas Computacionales, del Instituto Tecnológico Superior de Xalapa.



Ángel J. Sánchez-García es doctor en Inteligencia Artificial por la Universidad Veracruzana en 2018. Desde 2017 es profesor de tiempo completo de la Facultad de Estadística e Informática, Universidad Veracruzana. Sus intereses incluyen el aprendizaje automático y la Ingeniería de Software.



Saúl Domínguez-Isidro es doctor en Inteligencia Artificial, adscrito a la Facultad de Estadística e Informática de la Universidad Veracruzana. Sus intereses se centran en la solución de problemas mediante técnicas de inteligencia computacional y el desarrollo de software.



This work is under a Creative Commons  
Attribution-NonCommercial-ShareAlike 2.5 Mexico license .

# DetECCIÓN AUTOMÁTICA DE BRÓCOLI EN IMÁGENES RGB-D

## Automatic broccoli detection in RGB-D images

### Resumen

El brócoli es una hortaliza considerada de alto valor a nivel mundial. Sin embargo, en varios países su producción continuamente se ve impactada por una alta inestabilidad en la mano de obra causada por una amplia gama de fluctuaciones económicas, políticas y de fenómenos migratorios. Por este motivo, se han buscado alternativas de recolección autónomas y semiautónomas que faciliten este trabajo, incrementen la producción y reduzcan costos. En este artículo, se revisan algunas estrategias propuestas para la detección automática en el proceso de recolección de brócoli basadas en la segmentación y el filtrado de imágenes 3D, además de mostrar la eficacia de cada estrategia comparando los resultados obtenidos entre ellas.

Palabras clave

**Segmentación, filtrado de imágenes, Circulo de ajuste**

### Abstract

Broccoli is a vegetable considered to be of high value worldwide. However, in several countries its production is continually impacted by high labor instability caused by a wide range of economic and political fluctuations and migratory phenomena. For this reason, autonomous and semi-autonomous harvesting alternatives have been sought to facilitate this work, increase production and reduce costs. In this article, we review some proposed strategies for automatic detection in the broccoli harvesting process based on segmentation and 3D image filtering and show the effectiveness of each strategy by comparing the results obtained between them.

Keywords

**Segmentation, Image filtering, Circle fit**



## 1. Introducción

La recolección de hortalizas de alto valor como el brócoli es una tarea que consume tiempo a los agricultores por ser un trabajo aún se realiza enteramente de forma manual en todo el mundo.

Para cosechar el brócoli, los jornaleros deben identificar la cabeza del producto, determinar visualmente si ya tiene el tamaño adecuado y, finalmente, realizar un corte de forma inclinada en el tallo de hasta unos 12 cm. Un ejemplo de este proceso se puede ver en la Figura 1.

Este trabajo de cosecha manual es físicamente exigente y requiere que los trabajadores, en cada corte, se agachen, se inclinen y realicen una serie repetitiva de movimientos que, en el mediano y largo plazo, pueden resultar en lesiones de diversa índole. Aunado a esto, en varios países su producción constantemente se ha visto impactada por una alta inestabilidad en la mano de obra, causada por diversos factores que van desde condiciones económicas, dinámicas sociales, situaciones políticas y todas las complejidades involucradas en los fenómenos migratorios [1].



Figura 1. Proceso de recolección manual de brócoli realizado por cuadrillas de trabajadores.

## 2. Trabajos previos

Se han realizado varias investigaciones encaminadas a apoyar a los agricultores en la tarea de identificación y recolección del brócoli. En este sentido, el desarrollo de sistemas dotados de diversos grados de autonomía que permitan cosechar de forma continua ha recibido una atención considerable en las últimas décadas.

Ramírez [2] desarrolló uno de los primeros intentos de localizar las cabezas de brócoli utilizando un pequeño conjunto de 13 imágenes RGB. Este método detectó el brócoli basándose en el contraste y el análisis estadístico de la textura de la imagen. Mientras que, Block *et al.* [3] describieron un sistema que también usaba imágenes RGB para detectar brócoli de dos tipos de variedades. El método incluía un filtro de textura y un análisis de color de la apariencia de la cabeza.

Kusumam *et al.* [4] detectó cabezas de brócoli en datos tomados con un sensor RGB-D. Éste método combina la segmentación basada en las distancias entre puntos, la estimación de un vector de características compuesto por los valores de las normales de los puntos de cada segmento, un clasificador *Support Vector Machines* (SVM) y un filtro temporal para detectar y rastrear las cabezas de brócoli de dos variedades diferentes en una secuencia de tomas recolectadas a diferentes números de cuadros por segundo. Posteriormente, un enfoque similar aplicado a un conjunto mayor de datos RGB-D fue extendido a cuatro diferentes métodos de segmentación de mayor precisión que procesaban un número elevado de cuadros por segundo [5] y rastreaban de manera única cada cabeza detectada utilizando Filtros de Partículas [6].

De manera similar, por varios años las técnicas de aprendizaje profundo o *deep learning* (DL) se han aplicado con éxito en varios campos incluyendo la agricultura [7]. Las técnicas de DL han logrado un alto grado de clasificación y una ejecución muy eficiente. Sin embargo, la aplicación de DL requiere una gran cantidad de datos de entrenamiento debido a la complejidad de las tareas y un trabajo considerable de etiquetado no siempre disponibles debido al costo y tiempo de adquisición necesarios.

Por otro lado, Bender *et al.* [8] recolectaron un conjunto de datos de brócoli y coliflor en diferentes etapas de crecimiento y demostraron su detección usando una *Red Neuronal Convolutiva* (CNN, por sus siglas en inglés). Estos resultados, sin embargo, fueron obtenidos para toda la planta de brócoli visible en las imágenes y no únicamente para la cabeza de la hortaliza.

Trabajos más recientes propuestos por Block *et al.* [9] emplearon varias técnicas de DL en imágenes RGB para la detección de brócoli con altos grados de precisión, aún en condiciones de oclusión que ocurrían de manera natural o inducidas de manera artificial por los mismos investigadores [10]. Louedec *et al.* [11], utilizaron un modelo de CNN adaptado a los datos RGB-D representados por los valores de las normales de los puntos de cada cuadro adquirido por el sensor. Este enfoque arrojó una mayor generalización entre los diferentes conjuntos de datos utilizados en su experimentación.

De los estudios antes mencionados, el reconocimiento visual en 2D ha sido, por lo general, un paso previo para el funcionamiento de los sistemas autónomos reportados. Sin embargo, el procesamiento de información 3D es una fuente útil de información en diversas etapas del desarrollo de un producto agrícola para determinar características como su forma, tamaño y ubicación en el espacio.

Adicionalmente, se han creado herramientas de software que facilitan el procesamiento de datos 3D, tales como *Point Cloud Library*<sup>[2]</sup> y *Open3D*<sup>[3]</sup> que incluyen una amplia gama de funcionalidades específicas para procesar datos de nubes de puntos.

Un problema por considerar al procesar datos RGB-D es la existencia de objetos no deseados, tales como hojas o partes del mismo suelo donde se ha plantado la hortaliza (ver Figura 2). Esto dificulta el procesamiento de cada cuadro y demanda la utilización de algoritmos de filtrado y limpieza debido a que, al momento de segmentar los puntos, los elementos que no pertenecen a una cabeza de brócoli frecuentemente se agregan al mismo segmento.

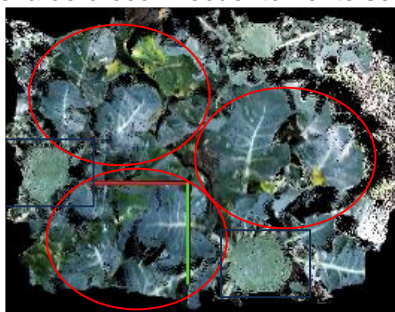


Figura 2. Elementos no deseados en la planta de brócoli tal como se encuentran en los campos de cultivo

En este artículo, se presenta un estudio sobre métodos de procesamiento de segmentos extraídos de imágenes RGB-D con el fin de depurar y filtrar el conjunto de clústers producidos en cada cuadro y, así, mejorar la detección y localización de las cabezas de brócoli. El desarrollo de pruebas se realiza utilizando datos obtenidos con un sensor Kinect en cultivos de Reino Unido y de España.

### 3. Herramientas y métodos

Para validar los métodos de filtrado, se utilizó un conjunto de datos de imágenes de brócoli adquiridas en campos de cultivo bajo diferentes condiciones meteorológicas con el sensor Kinect 2, el cual tiene una resolución de 1920x1080 RGB y 512x424 de resolución de profundidad [8].

El sensor se fijó dentro de una caja cerrada montada en un tractor agrícola con el fin de tener una iluminación uniforme para protegerlo de las condiciones externas.

El conjunto de datos se obtuvo de dos campos de cultivo de brócoli. El primero de ellos, obtenido de campos en Reino Unido, consta de 600 *cuadros* o imágenes RGB-D de una variante denominada *Ironman*. En tanto que las imágenes del segundo conjunto son procedentes de un campo de cultivo en España e incluye 300 *frames* o imágenes RGB-D de una variante de brócoli denominada *Titanium*. Una muestra de estos datos se puede ver en la Figura 3.

Las imágenes obtenidas por el sensor se encuentran en formato PCD (*point cloud data*), por lo que para su manipulación se hace uso de la biblioteca PCL (*point cloud library*) implementada en C++.

### 4. Segmentación de Nubes de Puntos

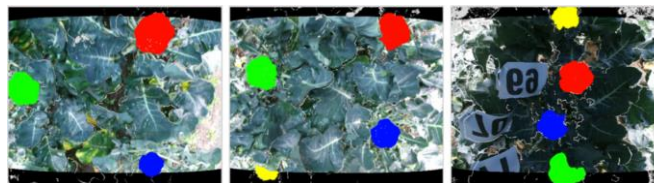
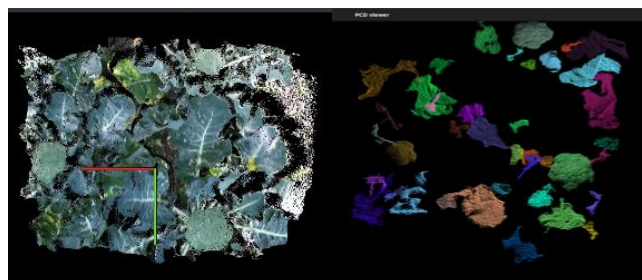


Figura 3. Imágenes RGB-D de cultivos de brócoli tomadas de campos del Reino Unido -dos primeras imágenes- y de España -tercera imagen. Las partes en color representan los puntos que han sido etiquetados como pertenecientes a cabezas de brócoli.

El primer paso del proceso de filtrado es la extracción de *clústers* o segmentación de los puntos en 3D. Para los experimentos presentados en este trabajo, se ha decidido utilizar un método de segmentación o *clusterización* basado en la distancia euclidiana entre los puntos de cada una de las imágenes RGB-D tomadas por el sensor [6].

La segmentación euclidiana comprueba la distancia entre dos puntos, si esta es inferior a un umbral, se considera que ambos pertenecen al mismo clúster. En este algoritmo un punto inicial cualquiera se incluye al *clúster* actual y luego se incluyen también todos los demás puntos que estén lo suficientemente cerca y, a partir de éstos, aún más puntos se agregan hasta que no se puede añadir ningún punto nuevo. Luego, se inicializa un nuevo *clúster* y el procedimiento se repite con el resto de los puntos que todavía no han sido procesados. El resultado de este proceso puede verse en las imágenes de la Figura 4.



a) Imagen Original

b) Segmentado

Figura 4. Resultado obtenido al aplicar el algoritmo de segmentación a un sólo cuadro tomado del conjunto de datos del Reino Unido. Los segmentos o clústers se visualizan cada uno en un color diferente.

## 5. Segmentación de Nubes de Puntos

Una vez obtenidos los *clústers* en cada imagen, en algunos de los segmentos que pueden ser identificados como brócoli, frecuentemente se ven acompañados de porciones de la imagen que realmente no son parte de una cabeza de la hortaliza. Un ejemplo de esto se puede observar en la Figura 5.

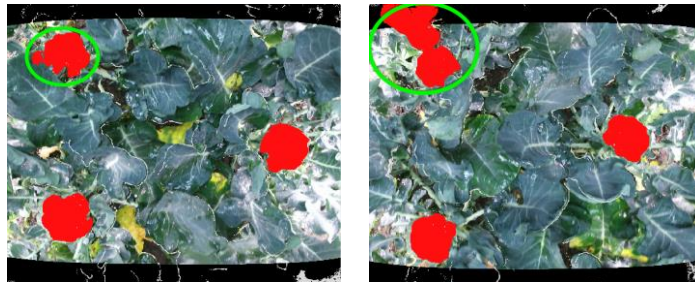
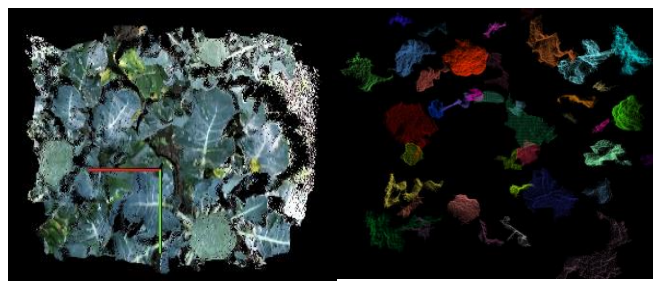


Figura 5. Algunos clústers, marcados con un círculo verde, se han segmentado con puntos que no pertenecen a una cabeza de brócoli.

Debido a este problema de *sobre-segmentado*, una alternativa es eliminar o filtrar los puntos adicionales con el fin de mejorar la identificación de aquellos que representan sólo las cabezas de brócoli. Para lograrlo, en este trabajo, se ha implementado un algoritmo denominado *Ajuste de Círculos* o *Circle Fit*. Este método encuentra el círculo que más se aproxima a los puntos de datos de un clúster.

Matemáticamente, un algoritmo de este tipo es un cálculo de optimización que maximiza o minimiza un parámetro determinado dentro de ciertos grados de libertad y que está sujeto a restricciones. Para el caso de ajuste de círculos, el tamaño y el punto central son las variables *flotantes* cuyos valores se ajustan y las desviaciones de los puntos [12]. Una muestra de este proceso de filtrado se muestra en la Figura. 6.



a) Imagen RGB-D Original

b) Clústers después de haber sido filtrados.

Figura. 6. Una muestra del proceso de filtración sobre los clusters segmentados de una imagen RGB-D.

El funcionamiento del algoritmo de filtrado es el siguiente: el primer paso es la integración y lectura de los *clústers* segmentados. Para cada *clúster*, se realiza lo siguiente:

- a) Ajuste del círculo al *clúster*. Primero, se aplica un algoritmo de ajuste de círculo a los puntos del *clúster*. Dado que PCL no proporciona un método directo de ajuste de círculo, se pueden utilizar otras bibliotecas o algoritmos para este propósito, como el método de ajuste de mínimos cuadrados (*least squares fitting*).
- b) Determinar los parámetros del círculo. Se calculan los parámetros del círculo ajustado, como las coordenadas del centro y el radio. Estos parámetros definen la forma y posición del círculo.
- c) Evaluar la calidad del ajuste del círculo. Se determina la calidad o bondad del ajuste del círculo al *clúster*. Esta evaluación puede basarse en varios criterios, como los errores residuales entre el círculo y los puntos o medidas estadísticas elementales.
- d) Filtrar el *clúster* basado en la calidad del ajuste del círculo. Para esto, se debe definir un umbral o criterio para determinar si la calidad del ajuste del círculo cumple con los criterios deseados. Si la calidad del ajuste del círculo está por debajo del umbral, se considera el *clúster* como un valor atípico y se filtra. En caso contrario se conserva el *clúster*.
- e) Guardar los *clústers* filtrados. Se guardan los *clústers* filtrados que pasaron el filtro de ajuste de círculo para su posterior procesamiento o análisis.

El esquema completo del proceso de filtrado por *circle fit* se muestra en la Figura 7.

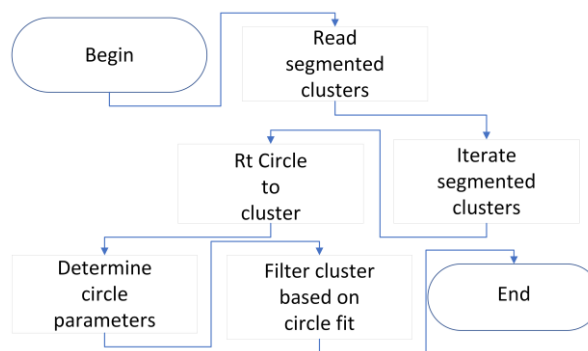


Figura 7: Esquema del proceso completo de filtrado de clústers.

## 6. Resultados

Después del proceso de filtración, se realizó una comparación de los clústeres resultantes con respecto a los datos identificados y etiquetados como cabezas de brócoli (*ground truth*).

En la Tabla 1 se muestra la precisión obtenida tanto en el proceso de segmentación como el de segmentación + filtrado.

Frames	Segmentación	Filtración	Porcentaje de mejora	Desviación Estandar
UK f1	0.95	0.97	0.02	0.01
UK f2	0.95	0.98	0.03	0.015
SP f1	0.88	0.94	0.06	0.03
SP f2	0.82	0.93	0.11	0.055

Tabla 1: Comparativo de porcentaje de precisión del proceso de filtrado respecto al proceso de segmentación aplicado en todos los conjuntos de datos.

En la tabla se puede observar que el proceso de filtrado mejora entre 2-3% la precisión de la segmentación para el conjunto de datos del Reino Unido, mientras que, para el conjunto de datos de España, se incrementa en más del 5%. Esto resulta en una identificación más homogénea en los conjuntos de datos utilizados en nuestros experimentos, aún y cuando las plantas de la hortaliza pertenecen a diferentes variedades y fueron tomadas en diferentes países y condiciones de cultivo.

## 7. Conclusiones

En este artículo, se presenta un estudio sobre métodos de procesamiento de segmentos extraídos de imágenes RGB-D con el fin de depurar y filtrar el conjunto de clústers producidos en cada cuadro con el objetivo de mejorar la detección de cabezas de brócoli en cultivos de dos países diferentes.

En nuestros experimentos se empleó una estrategia que utilizó el filtrado como postproceso de la segmentación y fue posible identificar de forma automática las cabezas de brócoli con una mayor precisión con respecto a los clústers segmentados. Se observó que el proceso de filtrado ayuda a mejorar la detección de los brócolis y hace más homogénea su identificación.

Este algoritmo puede ser usado para la generación de un modelo de reconocimiento y recolección de brócoli, puesto que es adaptativo según a la variedad de brócoli a reconocer en un cultivo.

## 8. Referencias

1. Duckett, T., Pearson, S., Blackmore, S., Grieve, B., Wen-Hua, C., Cielniak, G., Cleaversmith, J., Dai, J., Davis, S., Fox, C., From, P. J., Georgilas, I., Gill, H. S., Gould, I., Hanheide, M., Hunter, A. K., Iida, F., Mihalyova, L., Nefti-Meziani, S., . . . Yang, G. (2018). Agricultural Robotics: The Future of Robotic agriculture. arXiv (Cornell University). <https://doi.org/10.48550/arxiv.1806.06762>
2. Ramirez, R. A. (2006). Computer vision based analysis of broccoli for application in a selective autonomous harvester. ResearchGate. [https://www.researchgate.net/publication/279691048\\_Computer\\_Vision\\_Based\\_Analysis\\_of\\_Broccoli\\_for\\_Application\\_in\\_a\\_Selective\\_Autonomous\\_Harvester](https://www.researchgate.net/publication/279691048_Computer_Vision_Based_Analysis_of_Broccoli_for_Application_in_a_Selective_Autonomous_Harvester)
3. Blok, P. M., Barth, R., & Van Den Berg, W. (2016). Machine vision for a selective broccoli harvesting robot. IFAC-PapersOnLine, 49(16), 66-71. <https://doi.org/10.1016/j.ifacol.2016.10.013>.
4. Kusumam, K., Krajník, T., Pearson, S., Duckett, T., & Cielniak, G. (2017). 3D-Vision based detection, localization, and sizing of broccoli heads in the field. Journal of Field Robotics, 34(8), 1505-1518. <https://doi.org/10.1002/rob.21726>.
5. Montes, H. A., Louède, J. L., Cielniak, G., & Duckett, T. (2020) Montes, H. A., Le Louedec, J., Cielniak, G., and Duckett, T. Real-time detection of broccoli crops in 3D point clouds for autonomous robotic harvesting. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pages 10483–10488. <https://doi.org/10.1109/iros45743.2020.9341381>.
6. Montes, H. A. and Cielniak, G. (2022). Multiple broccoli head detection and tracking in 3D point clouds for autonomous harvesting. In Proceedings of the 36th AAAI Conference on Artificial Intelligence, workshop on AI for Agriculture and Food Systems, 2022.
7. Kamilaris, A., & Prenafeta-Boldú, F. X. (2018). Deep learning in Agriculture: A survey. Computers and Electronics in Agriculture, 147, 70-90. <https://doi.org/10.1016/j.compag.2018.02.016>
8. Bender, A., Whelan, B., & Sukkarieh, S. (2019b). A high-resolution, multimodal data set for agricultural robotics: a Ladybird 's-eye view of Brassica. Journal of Field Robotics, 37(1), 73-96. <https://doi.org/10.1002/rob.21877>
9. Blok, P. M., van Evert, F. K., Tielen, A. P. M., van Henten, E. J., & Kootstra, G. (2021). The effect of data augmentation and network simplification on the image-based detection of broccoli heads with Mask R-CNN. Journal of Field Robotics, 38(1), 85-104. Advance online publication. <https://doi.org/10.1002/rob.21975>
10. Blok, P. M., Van Henten, E., Van Evert, F., & Kootstra, G. (2021b). Image-based size estimation of broccoli heads under varying degrees of occlusion. Biosystems Engineering, 208, 213-233. <https://doi.org/10.1016/j.biosystemseng.2021.06.001>
11. Louedec, J. L., Montes, H. A. (2020). *et al.*, Segmentation and detection from organised 3D point clouds: A case study in broccoli head detection. In IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 285–293, <https://doi.org/10.1109/cvprw50498.2020.00040>.
12. Janeshewski, E. (s. f.). Calculations for circle fitting in CMM software. <http://www.mbccmm.com/cmmq/index.php/training/general/236-calculations-for-circle-fitting-in-cmm-software#:~:text=Calculations%20for%20Circle%20Fitting%20in%20CMM%20Software%201,is%20minimized%20C%20with%20no%20other%20constraints.%20M%C3%A1s%20elementos>

## Sobre los autores



Luis Fernando Talavera Ramírez estudiante de Ingeniería en computación participante en proyectos de investigación relacionados a tratamiento de imágenes digitales, ciencia y analítica de datos aplicada.



Luis Antonio Quiroz Mercado estudiante de Ingeniería en computación participante en proyectos de investigación relacionados a tratamiento de imágenes digitales, ciencia y analítica de datos aplicada.



Héctor Alejandro Montes Venegas es Doctor en Ciencias Computacionales por la Universidad de Lincoln en el Reino Unido. Cuenta con siete años de experiencia en diferentes empresas del sector público y privado donde trabajó principalmente como asesor externo dando servicios en desarrollo de software, capacitación de personal y asesoría general en sistemas de computo. Actualmente, es profesor de tiempo completo en la División de Computo de la Facultad de Ingeniería en la Universidad Autónoma del Estado de México. Sus intereses de investigación incluyen varias ramas de la Inteligencia Artificial, en particular algoritmos de Computo Evolutivo aplicados a problemas de optimización combinatoria y Percepción Robótica que involucra la adquisición de información mediante sensores (principalmente cámaras) y los consecuentes procesamiento, análisis e interpretación de los datos que producen (ya sean imágenes en 2D y 3D) para diversas tareas de robótica móvil y aprendizaje máquina. Ha dirigido y codirigido varias tesis de Licenciatura y de Maestría relacionadas con estas líneas de investigación.



Rosa María Valdovinos Rosas. Dra. en ciencias computacionales con doble grado, miembro del SNI nivel II. Ha participado en más de 15 proyectos de investigación con impacto social y científico, además cuenta con más de 50 artículos en revistas indizadas, más de 20 capítulos de libro, más de 20 publicaciones en congresos, 3 libros, más de 100 conferencias dictadas a nivel nacional e internacional y 1 licencia de patente otorgada. Ha formado más de 40 estudiantes de nivel licenciatura, Maestría y Doctorado, de estos últimos todos con reconocimiento por el SNI. Colabora en la organización de eventos de alcance nacional e internacional para la divulgación de la ciencia, así como acciones orientadas a niñas en la computación y vocaciones tempranas a nivel básico, secundaria y superior. La producción científica reporta más de 1000 citas, con un índice H14 y de acuerdo con el *International AD Scientific index 2023* es una las 10,000 científicas más influyentes a nivel Nacional.



José Raymundo Marcial Romero. Doctor en Ciencias de la Computación por The University of Birmingham, UK, en The School of Computer Science. Profesor investigador de tiempo completo en la Facultad de Ingeniería de la Universidad Autónoma del Estado de México. Miembro del Sistema Nacional de Investigadores del CONACYT nivel 1. Miembro de la academia Mexicana de Ciencias de la Computación. Sus aportaciones más relevantes se han dado en las áreas de aplicación de teoría de grafos, semántica de lenguajes de programación y la lógica computacional. De acuerdo con Google scholar ha publicado más de 90 artículos de investigación para revista, capítulos de libro. Ha impulsado proyectos estratégicos para la Universidad en la que labora desde la Secretaría de Planeación 2017-2022 y la Secretaría de Docencia 2022 a la fecha.



This work is under a Creative Commons

Attribution-NonCommercial-ShareAlike 2.5 Mexico license .



# **Combinación de Realidad Aumentada con Control Gestual de Manos para Apoyar el Aprendizaje de las Letras en Niños de Preescolar**

**Combining Augmented Reality with Hand Gesture Control to Support Letter Learning in Preschool Children.**

**Maria Antonieta Abud Figueroa<sup>1</sup>**

**Raúl de Jesus Sánchez Martínez<sup>1</sup>**

**Ulises Juárez Martínez<sup>1</sup>**

**Lisbeth Rodríguez Mazahua<sup>1</sup>**

**Hilarián Muñoz Contreras<sup>1</sup>**

<sup>1</sup>Tecnológico Nacional de México / Instituto Tecnológico de Orizaba

**Resumen**—El conocimiento de las letras es el primer paso para lograr la capacidad para leer y escribir adecuadamente, por lo que es de suma importancia en el proceso enseñanza-aprendizaje desarrollar actividades que capten la atención y el interés del niño por aprender. Actualmente el uso de las tecnologías de la información (TICs) ofrece la posibilidad de ampliar las estrategias didácticas que se pueden utilizar para el mejoramiento del quehacer educativo. El uso de la Realidad Aumentada es una de las tecnologías de mayor aplicación en el área educativa, por lo que en este trabajo se presenta el desarrollo de una herramienta que combina el uso de Realidad Aumentada con un control de interfaz a través del movimiento de las manos para introducir a los niños en el conocimiento de las letras. Esta aplicación despliega modelos tridimensionales de las letras y objetos que las utilizan y permite a los usuarios interactuar con ellos a través de movimientos de las manos. Su aplicación con niños de preescolar demostró ser una herramienta entretenida que ayudó de forma positivo al aprendizaje de las letras.

**Palabras clave:** Aprendizaje de letras; Interacción Humano-Máquina; Realidad Aumentada; Control Gestual de Manos;

**Abstract**—Knowledge of letters is the first step towards acquiring the ability to read and write properly, making it of utmost importance in the teaching-learning process to develop activities that capture the child's attention and interest in learning. Currently, the use of Information Technologies (IT) offers the possibility to expand didactic strategies that can be used for the improvement of educational practices. Augmented Reality is one of the most widely applied technologies in the educational field. This paper presents the development of a tool that combines the use of Augmented Reality with hand motion interface control to introduce children to letter knowledge. This application displays three-dimensional models of letters and objects that use them, allowing users to interact with them through hand movements. Its application with preschool children proved to be an entertaining tool that positively contributed to letter learning.

Keywords: Letter learning; Human-Machine Interaction; Augmented Reality; Hand Gesture Control.

## 1. INTRODUCCIÓN

El aprendizaje de las letras es básico en los niños ya que además de mejorar su capacidad de comunicación influye positivamente en su desarrollo cerebral. Al aprender los sonidos de las letras se estimulan conexiones neuronales en distintas áreas del cerebro que intervienen en el desarrollo del lenguaje. De igual forma se estimula la representación mental de conceptos abstractos, la concentración y la memoria. Lo cual es indispensable para el aprendizaje de la lectoescritura.

Tradicionalmente para que un estudiante aprenda las letras se le ponen ejercicios repetitivos de trazo y reconocimiento de la letra (a veces se incluye la mayúscula y la minúscula simultáneamente), lo que supone que el estudiante después de realizar estos ejercicios será capaz de identificar dicha letra sin embargo, esto puede conducir a dificultad para reconocer la letra ya que su representación puede variar en múltiples formas cambiando su tamaño, color, rotación, volumen y forma, entre otros. Para evitar estas complicaciones el uso de las tecnologías de información ofrece nuevas formas atractivas a los niños para realizar este aprendizaje. Dentro de las nuevas tecnologías útiles para este objetivo se encuentra la Realidad Aumentada (RA) ya que permite superponer imágenes virtuales sobre la realidad física lo que ofrece grandes oportunidades en el ámbito del aprendizaje recreando escenarios del mundo real en un entorno virtual, lo que influye en su rápida adopción en sectores de industria y educación. Otra de las tecnologías que hoy en día impacta en el uso de la computadora como medio didáctico es la inclusión de cámaras infrarrojas capaces de detectar gestos del usuario, *LeapMotion Controller* es una cámara infrarroja que permite identificar los movimientos de las manos a gran detalle con lo que se logra una manipulación de elementos de la computadora realizando gestos libres con las manos.

También, el desarrollo de nuevas formas de interfaces de usuario permite una interacción directa con los escenarios virtuales, lo que facilita su análisis, posibilitando una efectiva comunicación entre el entorno virtual y el usuario.

En este artículo se presenta el desarrollo un software que incorpora el uso de Realidad Aumentada con una interfaz humano-máquina que permite la interpretación de los movimientos de la mano del usuario por medio de la cámara *LeapMotion* y un traductor integrado que permite la manipulación de las letras con gestos manuales. En la sección 2 Metodología se presenta primeramente una revisión de trabajos relacionados con el proyecto, seguido del diseño y desarrollo del sistema propuesto. Posteriormente describen los resultados obtenidos, las conclusiones y trabajo futuro.

## 2. METODOLOGÍA

En esta sección se presenta la metodología empleada en el desarrollo del Proyecto, iniciando con la revisión de los principales trabajos relacionados con el tema, para continuar con la propuesta de arquitectura, desarrollo del mismo.

### 2.1 Trabajos Relacionados

Antes del inicio del proyecto se realizó una revisión de trabajos relacionados con el mismo. En esta sección se presentan aquellos que tienen mayor relación con el proyecto.

El aprendizaje de geometría es importante dentro del área de matemáticas, sin embargo, muchos estudiantes presentan problemas al trabajar este tema. Con el objetivo de brindar una herramienta de apoyo al estudio de la geometría (H. Q. Le and J. I. Kim, 2017) propusieron un marco de trabajo para su aprendizaje basado en RA y una interacción humano-máquina a través de reconocimiento de gestos manuales. En su investigación, para la detección de gestos emplearon el dispositivo *Leap Motion* utilizando marcadores de posición y la biblioteca ARToolkit para la renderización de los modelos tridimensionales que se obtenían con una cámara.

(S. Schutera et al., 2021) presentaron el desarrollo de una aplicación para dispositivos Android orientada a la enseñanza de geometría vectorial a la cual denominaron *cleARmaths*. Para su desarrollo incluyeron tecnología de realidad aumentada utilizando el SDK *Vuforia* para el seguimiento de marcadores de imagen y *Unity* para la captura y renderización de imágenes en tiempo real.

En el estudio de (Karambakhsh et al., 2019), presentaron el desarrollo de una aplicación para el aprendizaje de anatomía basada en una red neuronal convolucional capaz de reconocer expresiones humanas y utilizarlas como instrucciones. Se utilizó la cámara de profundidad *RealSense* para detectar la forma de los objetos escaneados generando un modelo 3D mediante su proyector láser infrarrojo. Utilizaron además la biblioteca *TensorFlow* de *Python* para visualizar los modelos generados por la red neuronal. Para el estudio utilizaron el dispositivo *HoloLens*, que permite rastrear los movimientos de las manos utilizando una cámara integrada y una pantalla interactiva que permite superponer a la imagen del fondo la red neuronal.

(Y. et al., 2019) desarrollaron la herramienta *MagicHand* la cual permite el control de dispositivos del Internet de las Cosas (IoT) usando RA y gestos manuales. En este trabajo utilizaron la interacción que permite la RA con el objetivo de mejorar la manipulación visual de los objetos interactivos en los dispositivos IoT. Se incluye un programa que permite la interacción con los sistemas de IoT a través de sonido e iluminación. Ofrece además el reconocimiento de códigos QR con lo que se pueden crear objetos virtuales e identificar figuras geométricas.

Otra aplicación importante de la RA es en el área de la telemedicina. En el estudio de (P. Bifulco et al, 2014) emplearon la realidad aumentada para desarrollar una aplicación de telemedicina con el objetivo de entrenar a usuarios sin experiencia en la realización de electrocardiogramas. Este sistema utilizó un maniquí para simular al paciente, sobre este maniquí se colocaron puntos de control los cuales fueron conectados al dispositivo, con lo que utilizando una cámara se logró reconocer los objetos. Se incluyeron, además audios y texto para dar instrucciones al usuario.

En el estudio del idioma inglés es muy importante el aprendizaje de la fonética ya que el no comprender los sonidos lingüísticos genera errores en su pronunciación que, a su vez, afectan la comunicación oral y escrita. Con la finalidad de facilitar el aprendizaje de la fonética en inglés (I. Nugraha et al, 2019) presentaron un modelo de enseñanza basado en realidad aumentada. En este proyecto utilizaron la herramienta *Aurasma*, actualmente conocida como *HP Reveal*, la cual permite combinar objetos virtuales y videoclips en un entorno simultáneamente real y virtual. Para el manejo de la RA se utilizaron tarjetas con imágenes que sirvieron de marcadores. Al enfocar estas imágenes con la cámara del dispositivo móvil, aparece un objeto 3D correspondiente a la imagen que contiene la tarjeta y se escucha una grabación con la pronunciación correcta mostrando las palabras.

En la misma temática del aprendizaje del inglés como segundo idioma, (V. Lin, et al, 2020) exploraron el aprendizaje referente a las habilidades de escritura. Se desarrolló una aplicación de RA orientada a la práctica de escritura ubicua consciente del contexto, buscando incrementar la motivación y reforzando la memoria a largo plazo. Los autores compararon trabajos escritos producidos por los estudiantes mientras utilizaban la aplicación frente a los mismos escritos obtenidos mediante el método tradicional de enseñanza, encontrando en la percepción de los estudiantes diferencias significativa, con lo que se demostró que la aplicación desarrollada ayudó a incrementar el recuerdo a largo plazo, ya que se presentó un incremento en la motivación mejorando la percepción del proceso de escritura. En el desarrollo de la aplicación se utilizó el software *Unity*.

En el estudio de (N. SyafiqahSafiee and A. Wanis Ismail, 2018) presentaron la creación de un entorno virtual dentro del cual es posible manipular los objetos a través de gestos manuales en RA. Para lograrlo utilizaron el dispositivo *LeapMotion* para la captura de los gestos manuales del usuario que posibilitaron manipular los objetos en la aplicación de RA desarrollada. Para la implementación de la RA, se utilizó el SDK *Vuforia*.

Después del análisis de estos trabajos se concluye que en la mayoría de los mismos utilizaron RA a través de la plataforma de desarrollo *Unity*, utilizando el dispositivo *LeapMotion* como herramienta para la interacción con los objetos virtuales debido a su menor costo comparado con otros dispositivos como *Google Lens* y *HoloLens*. También, al momento de esta revisión, no se encontró reportado algún trabajo aplicado al aprendizaje de las

letras por lo que se consideró un área de oportunidad trabajar en esta área. Además, se consideró desarrollar la aplicación como un videojuego debido a que estará orientado a niños de preescolar.

## 2.2 Arquitectura Propuesta

En esta sección se presenta la arquitectura propuesta para el desarrollo de la aplicación de RA con control gestual de manos que apoye el aprendizaje de las letras.

Como primer paso se realizó un análisis de diferentes tecnologías disponibles para el desarrollo del proyecto, considerando las siguientes necesidades:

- Un kit de desarrollo de software que soporte RA.
- Un dispositivo de control capaz de identificar el movimiento de las manos.
- Un motor de videojuegos con las capacidades de integrar la RA con el control gestual.

Después de realizar un análisis de diferentes opciones se eligió *Vuforia Engine* como kit de desarrollo para RA, *LeapMotion* para la identificación del movimiento de las manos y *Unity3D* como motor de videojuegos. A continuación, se describe cada uno.

### 2.2.1 Software Development Kit (SDK) de Realidad Aumentada (RA)

Un SDK es un conjunto de herramientas de desarrollo específicas para una plataforma y sistema operativo que a menudo incluyen bibliotecas de funciones para un lenguaje de programación específico.

Para el caso del desarrollo de aplicaciones con RA, la plataforma *Vuforia Engine* es ampliamente utilizada debido a que es compatible con gran parte de los dispositivos móviles disponibles en el mercado. Permite integrar de forma fácil funcionalidades para la captura de imágenes de la cámara para desplegar los objetos de RA. Además, ofrece una gran variedad de imágenes, objetos y entornos de trabajo para el uso de marcadores. Ofrece también una biblioteca de imágenes que pueden utilizarse en las aplicaciones. Cuenta también con un *plugin* para su integración con los motores de videojuegos *Unity3D* y *Unreal Engine*. Ofrece una versión gratuita, aunque con limitaciones en cuanto al uso de la lectura de marcadores y áreas (Shamsee et al., 2015).

### 2.2.2 Dispositivo de control

Un dispositivo de control permite la interacción humano-computadora a través de la manipulación de objetos virtuales. Estos dispositivos incluyen controladores físicos, cámaras para seguir movimiento, elemento para el reconocimiento de voz y gestos. El uso de estas interfaces tiene como objetivo que el usuario interactúe de forma intuitiva con el entorno virtual.

Para el desarrollo de este proyecto se eligió el dispositivo *LeapMotion Controller* el cual ofrece un rastreo avanzado de manos y dedos, lo cual permite una interacción natural con los objetos digitales. Este dispositivo se basa en un modelo del esqueleto humano, ofreciendo una representación detallada de las manos. Ofrece compatibilidad con motores de desarrollo como *Unity3D* y *Unreal Engine*. Soporta lenguajes de programación como *C#*, *C++*, *Java*, *JavaScript*, *Python* y *Objective-C*. La empresa que distribuye el dispositivo es *Ultra Leap* y para el uso del dispositivo ofrece una licencia personal, que es gratuita para uso no comercial, lo que lo hace una buena opción para el desarrollo de proyectos de realidad aumentada y virtual. (UltraLeap, 2022).

### 2.2.3 Motor de videojuegos

Los motores de videojuegos proveen un marco de trabajo para desarrollar aplicaciones interactivas y videojuegos, que incluyen las siguientes funcionalidades que permiten al desarrollador centrarse en la creación del juego sin programar todo desde cero (J. Atherton and G. Wang, 2018):

1. **Renderizado:** componente que dibuja y renderiza los gráficos 2D y 3D en la pantalla, manejando los elementos visuales como son la iluminación, sombras, y otros efectos visuales.
2. **Motor de física:** elemento encargado de simular las leyes de la física en el juego, como son gravedad, colisiones, fricción entre otros elementos.
3. **Animación:** componente encargado de dar vida a los objetos y personajes que son parte del juego, de manera que su forma de actuar sea realista.

4. Inteligencia Artificial (IA): este componente es el encargado de que los elementos que intervienen en el juego presenten un comportamiento creíble respondiendo de forma inteligente a las acciones que realiza el jugador.
5. Escenarios gráficos: herramienta que permite diseñar y crear los escenarios en los cuales se desarrolla el juego.
6. Codificación: lenguaje de programación que acepta el motor de videojuego; puede ser un lenguaje de programación propio o un lenguaje de programación popular.

Para el presente proyecto se eligió el motor de videojuegos Unity3D debido a su popularidad que a que proporciona una gran cantidad de funcionalidades para crear experiencias interactivas y multimedia [14]. Uno de sus puntos fuertes es su capacidad para exportar los juegos desarrollados a varias plataformas como *Windows*, *Mac*, *iOS*, *Android*, consolas de videojuegos y dispositivos de realidad virtual entre otros. Originalmente admitía como lenguajes de programación *C* y *C++*, actualmente sólo utiliza *C#* (J. Haas,2014).

#### 2.2.4 Módulos de la aplicación

El software desarrollado cuenta con tres módulos: el Traductor de Gestos Manuales, El motor de Juego y la Biblioteca SDK distribuidos en una arquitectura en capas como se muestra en la Figura 1.

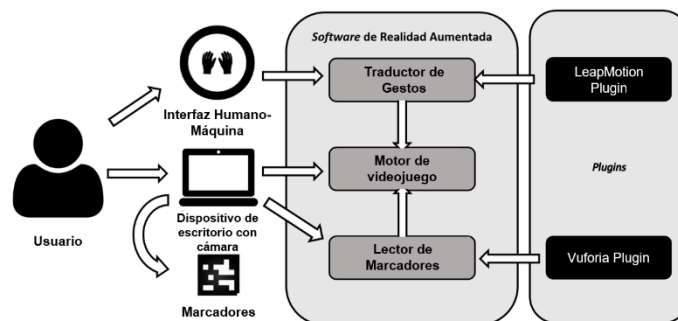


Figura 1. Arquitectura de la aplicación

El módulo Traductor de Gestos recibe desde la interfaz y en tiempo real la información de los movimientos de las manos del usuario y los envía al Motor de Juego, el cual es el encargado de obtener los modelos tridimensionales que se van a desplegar y traduce en forma inmediata los movimientos de las manos del usuario enviándolos al Motor de Juego desde donde se obtienen y manipulan los objetos obtenidos de la biblioteca SKD, en la cual se encuentra la información de los marcadores y modelos a desplegar.

Para el uso de la RA se desarrollaron los marcadores para lo cual fue necesario modelar diferentes figuras tridimensionales. Ejemplo de un marcador se observa en la Figura 2.



Figura 2. Marcador de la letra O

Para lograr el funcionamiento de la aplicación se requiere una computadora, que puede ser de escritorio o portátil, una cámara Web instalada en un trípode, los marcadores de las letras y la cámara infrarroja *LeapMotion*, distribuidas físicamente como se muestra en la Figura 3.

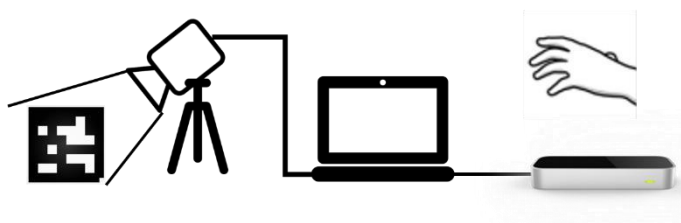


Figura 3. Disposición física del hardware

### 2.3 Resultados

La aplicación desarrollada tiene como objetivo ser una herramienta en apoyo al aprendizaje de las letras. Para lograr el objetivo se diseñaron varias escenas con las siguientes opciones: menú principal, menú de niveles, video de ayuda y sección de créditos como se observa en la Figura 4. Debido a que es una aplicación principalmente dirigida a niños de preescolar se sugiere que su uso sea bajo la supervisión de un adulto, preferentemente del profesor o profesora del niño.



Figura 4. Menú de niveles

Al ingresar a cada uno de los niveles del juego se escucha un audio de bienvenida con las instrucciones. El usuario deberá colocar frente a la cámara el marcador de la letra que se va a estudiar, al hacerlo aparecerá encima del marcador el modelo tridimensional del objeto relacionado con el marcador y se reproduce un audio indicando que se debe tomar el objeto, sostenerlo y repetir lo que escucha. Por ejemplo, si está repasando la letra 'o', aparece un oso escuchándose el audio "o de oso". Una vez que el niño repite la frase, debe soltar el objeto sobre el marcador, y el sistema se comporta de acuerdo al nivel de acuerdo a lo siguiente:

- Nivel uno: cuando el niño suelta el objeto sobre el marcador, aparecerán otros tres objetos con nombres que inician con la misma letra del objeto del marcador, y aparece en la esquina inferior derecha un número que corresponde a la cantidad de intentos de resolver el juego. Se reproduce al audio con las instrucciones indicando que se deben sujetar cada uno de los objetos y soltarlo sobre el marcador. Cada vez que se lleva el objeto al marcador se incrementa el contador de puntos y se reproduce el audio con el nombre respectiva del objeto. Cuando el usuario termina de colocar los tres objetos aparece un mensaje de felicitación y regresa al menú de niveles. La Figura 5 muestra la pantalla nivel 1 para la letra o.



Figura 5. Funcionamiento del nivel uno

- Nivel dos: Tiene un funcionamiento similar al del nivel uno, con la diferencia que aparecen cinco objetos en lugar de tres. Los dos objetos nuevos tienen un nombre que no inicia con la letra que se está estudiando. De modo que cuando uno de estos es sujetado y arrastrado por el usuario se le indica al usuario que es erróneo y todo el conjunto de objetos se reacomoda aleatoriamente. Esto se repite hasta que se eligen sólo aquellos objetos que inician con la letra correcta. La Figura 6 muestra el marcador de la letra E en este nivel.



Figura 6. Funcionamiento del nivel 2

- Nivel tres: Este nivel funciona de forma similar que el nivel dos, sólo que aquí en lugar de que los objetos sean figuras ahora representan la palabra del nombre del objeto. En la Figura 7 se muestra la pantalla de este nivel.





Figura 7. Funcionamiento del nivel 3

- Nivel cuatro: En este caso aparece una palabra con la diferencia de que incluye una o varias veces la letra que se está repasando en diferentes posiciones. El usuario deberá tomar cada letra y arrastrarla sobre el marcador. En caso de que la letra seleccionada sea errónea, ésta reaparece colocada en donde se tomó. Una vez que dicha palabra se ha limpiado de la letra correcta, las letras restantes desaparecen. La figura 8 muestra el funcionamiento de este nivel para la letra A en el nivel 4.



Figura 8. Diseño y funcionamiento del nivel 4

## 2 Resultados Obtenidos

El sistema desarrollado se probó en un Jardín de Niños privado de la ciudad de Orizaba, Veracruz a cargo de la directora de la institución y la maestra de preescolar. Para el estudio se contó con la participación de diez niños:

- Dos niños de Primer grado de preescolar.
- Cuatro niños Segundo grado de preescolar.
- Cuatro niños de Tercer grado de preescolar.

Los niños se eligieron considerando que la mitad fueran alumnos con rezago en su aprendizaje de las letras y la otra mitad niños con un nivel avanzado.

En el estudio se realizaron cuatro rondas de pruebas, realizando a los participantes unas preguntas de control antes de la prueba y después de la misma.

Previo a la prueba se realizaron las siguientes acciones:

1. Se mostró a cada niño el objeto del marcador para que identificara de qué objeto se trataba preguntándole ¿Cómo se llama esto?
2. Se le pidió que mencionara con qué letra empieza.
3. Finalmente, se le mostró la letra para que la identificara.

Al terminar cada prueba se cuestionó al niño sobre

1. Si le gustó el juego.
2. Si consideraba difícil
3. Se le pidió que mencionara que más le gustó o qué le disgustó
4. Finalmente, se le pidió que indicara si le gustaría volver a jugarlo.

Estas preguntas de control permitieron evaluar el grado en que el programa ayuda al estudiante a aprender las letras. En la Figura 9 se visualiza a un niño jugando con la aplicación.



Figura 9. Estudiante durante aplicación de prueba

Tomando en cuenta las respuestas a estas preguntas y las observaciones realizadas durante las pruebas se concluyen los siguientes resultados:

- De los diez estudiantes que participaron, nueve demostraron amplio interés por utilizar el software.
- En tres de los niños que presentan problemas de déficit de atención diagnosticada se mostró que la herramienta fue útil en el aprendizaje de las letras.
- Los niños presentaron gran interés en el uso de la RA, mostrándose emocionados al ver que los objetos que aparecían en pantalla.
- Otro aspecto que agradó a los niños fue la capacidad del sistema de poder agarrar los objetos con sus manos.
- Se demostró que, con el uso del programa, los niños con rezago académico en el aprendizaje de las letras pudieron identificar letras y palabras.
- El programa mostró ser útil también en el aprendizaje de nuevas palabras.
- Lo que más gustó a los niños fue el poder tomar los objetos con sus manos.
- Se identificó que el dispositivo LeapMotion presenta algunos problemas para detectar manos muy pequeñas, por lo que se recomienda explorar otras opciones de detección de manos.

### 3 Conclusiones

El software desarrollado ayuda al aprendizaje de las letras a través de la identificación del sonido con el símbolo respectivo, y su asociación con un objeto específico que contiene dicha letra. Por ser un software orientado a niños de preescolar, el niño debe estar acompañado de un adulto que lo oriente en su uso, evalúe los resultados y promueva la interacción entre los niños después de su aplicación. En el caso de estudio se trabajó con niños de preescolar supervisados por su profesora y se comprobó que el uso de RA y control gestual de manos a través de la herramienta *LeapMotion* fue exitoso puesto que se logró captar la atención de los niños ya que mostraron gran interés en la capacidad de tocar los objetos, palabras y letras con sus manos. Para lograr la evaluación del impacto en el proceso cognitivo de los niños se propone realizar pruebas con un grupo mayor de niños.

Para el uso del sistema se requiere una inversión de aproximadamente 5 mil pesos por los costos del dispositivo *LeapMotion*, la cámara web y el tripié, por lo que se sugiere como trabajo futuro explorar el uso de técnicas de visión artificial para el reconocimiento de las manos y evitar la necesidad de utilizar dispositivos adicionales a los ya incluidos en las computadoras personales. Se sugiere también incluir niveles orientados al aprendizaje de las sílabas, y considerar las letras r, c, s, q, p, g, j y ñ, que debido a que se pueden combinar requieren enseñarse de forma especial.

#### Agradecimientos

Se agradece al Tecnológico Nacional de México por el financiamiento para la realización de este trabajo. También se agradece al Consejo Nacional de Humanidades, Ciencias y Tecnologías (CONAHCYT) por el apoyo otorgado.

#### Referencias

- [1] Atherton, J., & Wang, G. (2018). Chunity: Integrated Audiovisual Programming in Unity. New Interfaces for Musical Expression, Virginia, US. [Online], from [http://www.nime.org/proceedings/2018/nime2018\\_paper0024.pdf](http://www.nime.org/proceedings/2018/nime2018_paper0024.pdf)
- [2] Bifulco, P., Narducci, F., Vertucci, R., Ambruosi, P., Cesarelli, M., & Romano, M. (2014). Telemedicine supported by Augmented Reality: An interactive guide for untrained people in performing an ECG test. *Biomed Eng Online*, 13(1), 1–16. doi: 10.1186/1475-925X-13-153.
- [3] Haas, J. (2014). *A History of the Unity Game Engine*, 44. [Online] from <https://api.semanticscholar.org/CorpusID:86824974>
- [4] Karambakhsh, A., Kamel, A., Sheng, B., Li, P., Yang, P., & Feng, D. D. (2019). Deep gesture interaction for augmented anatomy learning. *International Journal of Information Management*, 45, 328–336, doi: [10.1016/j.ijinfomgt.2018.03.004](https://doi.org/10.1016/j.ijinfomgt.2018.03.004)
- [5] Le, H. Q., & Kim, J. I. (2017). An augmented reality application with hand gestures for learning 3D geometry. *2017 IEEE International Conference on Big Data and Smart Computing, BigComp 2017*, 34–41. doi: 10.1109/BIGCOMP.2017.7881712.
- [6] Lin, V., Liu, G. Z., & Chen, N. S. (2020). The effects of an augmented-reality ubiquitous writing application: a comparative pilot project for enhancing EFL writing instruction. *Comput Assist Lang Learn*, 1–42. doi: 10.1080/09588221.2020.1770291.
- [7] Microsoft. (2022). Visual Studio Code. from <https://visualstudio.microsoft.com/es/>
- [8] Nugraha, I., Suminar, A. R., Octaviana, D. W., Hidayat, M. T., & Ismail, A. (2019). The application of augmented reality in learning English phonetics. *Journal of Physics: Conference Series*, 1402(7). doi: 10.1088/1742-6596/1402/7/077024.
- [9] Salinas, D., De Moraes, C., & Schwabe, M. (2018). Programa Para La Evaluación Internacional De Alumnos (Pisa) Pisa 2018 - Resultados - Nota País México. Oede, I–III, 1–12. [Online]. from [https://www.oecd.org/pisa/publications/PISA2018\\_CN\\_MEX\\_Spanish.pdf](https://www.oecd.org/pisa/publications/PISA2018_CN_MEX_Spanish.pdf)
- [10] Schutera, S., et al. (2021). On the potential of augmented reality for mathematics teaching with the application cleARmaths. *Educ Sci (Basel)*, 11(8). doi: 10.3390/educsci11080368.
- [11] Shamsee, H., Klebenov, N., & Fayed, D. (2015). CCNA Data Center DCICT 640-916: Official Cert Guide. Cisco.
- [12] Sun, Y., et al. (2019). Manipulating IoT Devices in Augmented Reality Environment MagicHand : A Deep Learning Approach towards Manipulating IoT Devices in Augmented Reality Environment, *Institute of Electrical and Electronics Engineers (IEEE)* 0–9 doi: 0.1109/vr.2019.8798053.
- [13] SyafiqahSafiee, N., & Ismail, A. W. (2018). AR Home Deco: Virtual Object Manipulation Technique Using Hand Gesture in Augmented Reality. *Innovations in Computing Technology and Applications*, 1–6, 3.
- [14] UltraLeap (sf). Leap Motion Controller. From <https://www.ultraLeap.com/product/leap-motioncontroller/#overview>.



## **Estudio comparativo de algoritmos para el reconocimiento de dígitos manuscritos. Caso: MNIST**

### **Comparative Study of Algorithms for Handwritten Digit Recognition. Case: MNIST**

**Yasiel Conde Bernal<sup>1</sup>**

**Dr. Saul Lazcano Salas<sup>1</sup>**

**Maricela Quintana López<sup>1</sup>**

**Saturnino Job Morales Escobar<sup>1</sup>**

**Asdrúbal López Chau<sup>1</sup>**

<sup>1</sup>Maestría en Ciencias de la Computación, Universidad Autónoma del Estado de México

## RESUMEN

El aprendizaje de máquinas dispone de varios algoritmos que ofrecen buenos resultados de exactitud en labores de clasificación de texto; no obstante, existen otros factores como el consumo de recursos que pueden influir o determinar en la elección de un algoritmo sobre otro. Por lo anterior, el presente trabajo ofrece una comparación entre varios algoritmos de clasificación como son k vecinos más cercanos, regresión logística, Naive Bayes, máquina de soporte de vectores, árbol de decisión, bosque aleatorio, perceptrón multicapa y red neuronal convolucional, que toma en cuenta varios aspectos que permiten determinar cuál algoritmo ofrece el mejor balance entre el consumo de tiempo y almacenamiento garantizando a su vez altos valores de la métrica de exactitud. Para el estudio se empleó el conjunto de datos MNIST correspondientes a los dígitos manuscritos del cero al nueve y se emplearon técnicas de búsqueda exhaustiva de hiperparámetros, cotejando los resultados de las métricas de clasificación exactitud, precisión, exhaustividad y F1-score, así como el consumo de recursos de tiempo y memoria.

Palabras claves: MNIST, redes neuronales, reconocimiento de patrones, métricas de clasificación, algoritmos de clasificación

## ABSTRACT

Machine learning has several algorithms that offer good accuracy results in text classification tasks; however, there are other factors such as resource consumption that can influence or determine the choice of one algorithm over another. Therefore, the present work offers a comparison between several classification algorithms such as k nearest neighbors, logistic regression, Naive Bayes, support vector machine, decision tree, random forest, multilayer perceptron and convolutional neural network, which takes into account several aspects that allow us to determine which algorithm offers the best balance between time and storage consumption, while guaranteeing high values of the accuracy metric. For the study, the MNIST data set corresponding to the handwritten digits from zero to nine was used and exhaustive hyperparameter search techniques were used, comparing the results of the classification metrics accuracy, precision, recall and F1-score, as well as the consumption of time and memory resources.

Keywords: MNIST, neural networks, pattern recognition, classification metrics, classification algorithms

## 1. INTRODUCCIÓN

El reconocimiento de patrones (RP) es un fenómeno que se da de forma natural en seres vivos, especialmente en humanos, que desde que nacen ya comienzan a reconocer y a apreciar el entorno alrededor, a través de los sentidos: el tacto, el olfato, el oído, el gusto y la vista. De forma general, se puede decir que el reconocimiento de patrones es el mecanismo con el que se cuenta para distinguir unas cosas de otras, relacionar cosas semejantes, formar grupos de cosas, tomar y explicar decisiones, describir objetos, entre otras.

En un sentido más formal, se puede definir al reconocimiento de patrones como “la ciencia que se ocupa de los procesos sobre ingeniería, computación y matemáticas relacionados con objetos físicos y/o abstractos, con el propósito de extraer información que permita establecer propiedades de o entre conjuntos de dichos objetos” (Ruiz-Shulcloper et al., 1999). Durante las últimas décadas, se han reportado interesantes resultados en el reconocimiento de patrones, entre los métodos tradicionales más exitosos se encuentran los de enfoque estadístico, como las reglas de decisión de Bayes paramétricas y no paramétricas, las máquinas de vectores de soporte, los algoritmos de refuerzo, entre otros (Zhang et al., 2020).

Las aplicaciones de aprendizaje automático en campos como la visión por computadora, la robótica, el reconocimiento de voz y el procesamiento del lenguaje natural, generalmente tienen como objetivo emular y acercarse lo más posible al desempeño humano (Braga-Neto, 2020). Por su parte las redes neuronales artificiales (RNA) también han jugado un papel importante en el desarrollo de la inteligencia computacional. Las RNA son cada vez más atractivas, efectivas, eficientes y exitosas para lograr el reconocimiento de patrones (RP) en muchos problemas.

El presente trabajo tiene como objetivo, realizar un estudio comparativo entre diferentes algoritmos empleados para el reconocimiento de patrones en imágenes con caracteres manuscritos, así como el ajuste de los parámetros de cada uno para obtener las mejores métricas de clasificación con base en los hiperparámetros seleccionados, empleando para ello la base de datos MNIST. Se manejarán las métricas de exactitud, precisión, exhaustividad y f1-score, además del lenguaje de programación Python y las librerías keras, tensorflow, sklearn, numpy, pandas y matplotlib para la implementación de los algoritmos.

Los algoritmos que se analizarán en el estudio comparativo serán K-vecinos más cercanos (KNN por sus siglas en inglés), Naive Bayes (NB), regresión logística (LR), árboles de decisión (DT), bosque aleatorio (RF), perceptrón multicapa (MLP) y red neuronal convolucional (CNN). Para la identificación de los parámetros se utilizará la técnica de grid search que los encuentra buscando exhaustivamente a través de un subconjunto especificado manualmente del espacio de hiperparámetros del algoritmo de destino.

Los resultados sentarán las bases para el desarrollo de una aplicación móvil que potencie las habilidades comunicativas de las personas sordociegas mediante el reconocimiento de símbolos manuscritos específicos de la comunidad, permitiendo a estos individuos comunicar necesidades básicas. Cabe resaltar que en esta población se dificulta la comunicación (aunque no se imposibilitan), puesto que, siendo personas ciegas o débiles visuales, son también sordas o débiles auditivas, requiriendo de apoyo para realizar actividades que parecieran sencillas a un individuo sin la discapacidad como por ejemplo buscar información, dialogar, interactuar con el entorno o desplazarse justo como el resto de las personas y este tipo de apoyo suele variar de un sordociego a otro. Por lo que el presente estudio aportará una guía para la elección de un algoritmo de reconocimiento de patrones y parámetros optimizados que permita obtener un nivel de exactitud en la clasificación superior al 97% y una alta precisión por símbolo, manteniendo de manera simultánea el menor consumo de memoria y tiempo de ejecución posible, con el fin de poder ser utilizada en dispositivos móviles.

## 2. DESARROLLO

La clasificación de textos es una tarea de fundamental importancia en diversas actividades y ha ido ganando terreno gracias a los desarrollos recientes en los campos de la minería de textos y el procesamiento del lenguaje natural. La velocidad a la que se crea actualmente la información textual ha superado por mucho las soluciones manuales para estas tareas, lo que significa que esos métodos no solo son útiles, sino también estrictamente necesarios (Gasparetto et al., 2022).

En el presente trabajo, se emplea el conjunto de datos modificado del Instituto Nacional de Estándares y Tecnología (MNIST), propuesto en 1994 como una combinación de dos de las bases de datos del NIST: la especial 1 y la especial 3, constituidas estas 2 por dígitos escritos por estudiantes de secundaria y empleados de la Oficina del Censo de los Estados Unidos, respectivamente (Yann, 1998). Se tomó la decisión de emplear este conjunto de datos debido a que es muy empleado en el reconocimiento de caracteres (Baldominos Gómez et al., 2019), llegando a ser llamado el “Hello World” del Machine Learning (Merenda et al., 2020). Otro punto a favor de la elección de este conjunto de datos es que es de acceso público, aspecto que beneficia su amplio reconocimiento dentro de la comunidad científica y que lo posiciona como un punto de referencia ideal a la hora de analizar algoritmos de clasificación de imágenes y texto. Las características de las imágenes de MNIST son de igual forma similares a las que se emplearán en la aplicación móvil para la comunidad sordociega, ya que ambas poseen igual dimensión, se encuentran en escala de grises y se corresponden con caracteres manuscritos.

La base de datos MNIST está compuesta por un total de 70 000 imágenes en formato de 28x28 píxeles en escala de grises y presenta una versión dividida en 60 000 datos para la etapa de entrenamiento y 10 000 para la de pruebas, subconjuntos estos que serán empleados en el presente trabajo, la descripción resumen se muestra en la tabla 1. La base de datos cuenta con un total de 10 clases que se corresponden con los dígitos del 0 al 9, en la figura 1 se muestra la distribución por clases en el conjunto de entrenamiento.

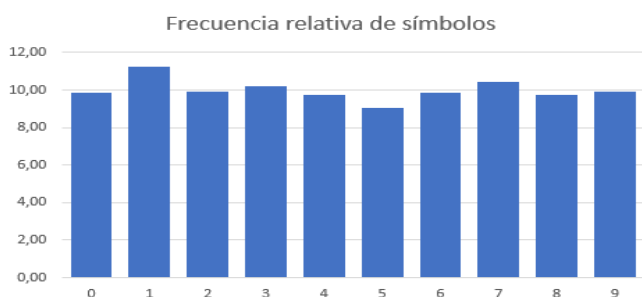


Figura 1. Frecuencia relativa de dígitos del conjunto de entrenamiento

Las clases se encuentran aceptablemente balanceadas, sobrepasando la media los símbolos 1, 3 y 7. En el caso del conjunto de prueba, la figura 2 muestra la frecuencia relativa por cada dígito.



Figura 2. Frecuencia relativa de los datos del conjunto de prueba

La proporción de clases en el conjunto de pruebas se encuentra balanceada en lo posible; no obstante, superan la media los dígitos 1, 2, 3, 7 y 9. El conjunto de entrenamiento está compuesto por aproximadamente el 85,71% de los registros y el de prueba por el 14,29%.

Conjunto de datos	Objetos	Tamaño (píxeles)
x_train	60 000	28*28 = 784
x_test	10 000	28*28 = 784
Total	70 000	

Tabla 1. Estructura de los conjuntos de entrenamiento y prueba

## 2.1 ALGORITMOS DE CLASIFICACIÓN

Para el aprendizaje automático se han desarrollado varios tipos de algoritmos entre los cuales se encuentran: MLP (Multilayer Perceptron), K-NN (K-Nearest Neighbor), NB (Naive Bayes), SVM (Support Vector Machines), DT (Decision Tree), entre otros (Llumiquinga Almeida, 2022). En la presente investigación se opta por el uso de los antes mencionados por ubicarse entre los más usados para la clasificación de textos (Gasparetto et al., 2022) y por presentar elevado desempeño en tareas de reconocimiento de patrones (Sen et al., 2020). Se incluye además en el comparativo la red neuronal convolucional (CNN por sus siglas en inglés) ya que durante la última época se ha destacado notablemente su empleo debido al elevado rendimiento que demuestran en tareas de clasificación de imágenes (Li et al., 2021). A continuación, se analizarán brevemente cada uno de los algoritmos citados.

### 2.1.1 K-Vecinos más cercanos (KNN)

Es un algoritmo de clasificación supervisado no paramétrico, que mediante el cálculo de distancias permite identificar a los vecinos más cercanos de un punto de consulta dado, de modo que se pueda asignar una etiqueta de clase a ese punto (Lévano-Rodríguez & Cerdán-León, 2022). KNN integra al conjunto de algoritmos de “aprendizaje perezoso”, esto significa que almacena solamente el conjunto de datos de entrenamiento sin pasar realmente por una fase de entrenamiento (Ehsani & Drabløs, 2020). Esto deja un gran trabajo a la etapa de predicción o clasificación, al realizar el cálculo durante esta etapa. El algoritmo emplea para su funcionamiento un conjunto de métricas de distancia como las que se listan a continuación:

- Distancia euclidiana
- Distancia de Minkowski
- Distancia de Manhattan
- Distancia de Chebyshev
- Similitud de coseno
- Coeficiente de Jaccard
- Distancia de Hamming

KNN es fácil de implementar y posee pocos hiperparámetros lo que constituye una ventaja, sin embargo, es costoso computacionalmente y lento, además sufre de la maldición de la dimensionalidad, uno de los principales problemas a tener en cuenta en la fase de clasificación, debido a que demanda una mayor cantidad de datos para entrenar el modelo en relación con la dimensión del vector de características. (Naranjo, 2022) (no está clara la idea)

### 2.1.2 Naïve Bayes

Es otro método de aprendizaje automático empleado para la resolución de problemas de clasificación. Este algoritmo es de tipo probabilístico y sienta sus bases en el teorema de Bayes. Asume que los predictores en un modelo Naïve Bayes son condicionalmente independientes o no están relacionados con ninguna otra característica del modelo. También supone que todas las características contribuyen por igual al resultado. El algoritmo funciona bien con tipos de datos heterogéneos y también con valores perdidos, debido al tratamiento independiente de cada variable predictiva para la construcción del modelo (Griffis et al., 2016).

Este método utiliza una variable score para apoyar el proceso de clasificación, en lugar de utilizar sólo el método de la probabilidad (Gutiérrez Esparza et al., 2017). Es un algoritmo sencillo de implementar y con parámetros fáciles de estimar, pero se afecta a frecuencia cero: la frecuencia cero ocurre cuando una variable categórica no existe dentro del conjunto de entrenamiento, y al estar basado en probabilidades, sufre cuando aparecen casos raros.

### 2.1.3 Regresión logística (LR)

La regresión logística es un instrumento estadístico empleado para el análisis de datos. Su origen se remonta a la década de los sesenta (Confield, Gordon y Smith 1961). LR se utiliza para describir datos y explicar la relación entre una variable binaria dependiente y una o más variables independientes nominales, ordinales, de intervalo o de relación. Los modelos LR generalmente incluyen solo las variables que se consideran "importantes" para predecir un resultado, normalmente, la predicción ofrece un número finito de resultados (Kirişci, 2019).



#### **2.1.4 Máquina de soporte vectorial (SVM)**

La máquina de soporte vectorial es un método popular de aprendizaje automático para clasificación, regresión y otras tareas de aprendizaje (Luu et al., 2020). Es una técnica de clasificación destacada, ampliamente utilizada desde sus inicios. Fue introducida por primera vez por Cortés y Vapnik en 1995 para problemas de clasificación binaria. SVM busca encontrar hiperplanos que determinen el límite de decisión que permita clasificar los puntos de datos en dos clases (Tanveer et al., 2022).

SVM, en su forma básica, es un clasificador binario lineal, que identifica un único límite entre dos clases. La SVM lineal asume que los datos multidimensionales son linealmente separables en el espacio de entrada. En particular, las SVM determinan un hiperplano óptimo (una línea en el caso más simple) para separar el conjunto de datos en un número discreto de clases predefinidas usando los datos de entrenamiento (Sheykhmousa et al., 2020).

#### **2.1.5 Árboles de Decisión (DT)**

DT es un enfoque de aprendizaje automático supervisado para resolver problemas de clasificación y regresión mediante la división continua de datos en función de un determinado parámetro. Las decisiones están en las hojas y los datos se dividen en los nodos. En el árbol de clasificación, la variable de decisión es categórica (resultado en forma de Sí/No) y en el Árbol de regresión, la variable de decisión es continua. DT tiene las siguientes ventajas: es adecuado para problemas de regresión y clasificación, es de fácil interpretación, presenta facilidad de manejo de valores categóricos y cuantitativos, es capaz de rellenar los valores faltantes en los atributos con el valor más probable y posee alto rendimiento debido a la eficiencia del árbol de algoritmo transversal (Ray, 2019).

#### **2.1.6 Bosque aleatorio (RF)**

Modelo propuesto por Kam Ho (1995) de Laboratorios Bell, y posteriormente desarrollado por Breiman (2001) quien presentó el modelo totalmente desarrollado. RF es una combinación de árboles predictores tal que cada árbol depende de los valores de un vector aleatorio probado independientemente y con la misma distribución para cada uno de estos, empleando varios modelos para resolver un único problema y combinando sus resultados para producir una solución precisa (Alarcón Flores, 2017).

Sorprendentemente, la combinación de muchos árboles de decisión basados en datos muestreados aleatoriamente funciona bien en la práctica porque la aleatoriedad ayuda a que el modelo evite el ajuste excesivo de los datos; sin embargo, su velocidad de predicción puede ser lenta si contiene una gran cantidad de árboles, afectando el rendimiento del modelo, ya que este depende del parámetro que representa la profundidad de los árboles aleatorios en el mismo (Cho et al., 2019).

#### **2.1.7 Perceptrón multicapa (MLP)**

Los perceptrones multicapa forman un tipo de red neuronal, compuestos por un sistema de neuronas o nodos simples interconectados, o sea, un modelo que representa un mapeo no lineal entre un vector de entrada y un vector de salida. Los nodos están conectados por pesos y señales de salida que son una función de la suma de las entradas al nodo modificadas por una simple función no lineal de transferencia o activación (Gardner & Dorling, 1998).

#### **2.1.8 Red neuronal convolucional (CNN)**

Las redes neuronales convolucionales (CNN) fueron introducidas por primera vez por Fukushima en 1998. Están formadas por neuronas, donde cada neurona tiene un peso y un sesgo que se pueden aprender. Contiene una capa de entrada, una capa de salida y múltiples capas ocultas, donde la capa oculta consta de una capa convolucional, una capa de agrupación, una capa totalmente conectada (FC) y varias capas de normalización.

La capa convolucional aplica una operación de convolución para fusionar dos conjuntos de información. Imita la retroalimentación de una neurona individual a los estímulos visuales. La capa de agrupación se utiliza para reducir la dimensionalidad, asociando la salida del grupo de neuronas en una capa con la neurona individual. La capa FC conecta cada neurona de una capa con cada neurona de otra capa. Su propósito principal es clasificar las imágenes de entrada en varias clases, según los conjuntos de datos de entrenamiento (Dhillon & Verma, 2020).

CNN ha estado alcanzado grandes logros y se ha convertido en una de las redes neuronales más representativas en el campo del aprendizaje profundo. La visión artificial basada en CNN ha permitido a las personas lograr lo que se consideraba imposible en los últimos siglos, como el reconocimiento facial, los vehículos autónomos, los supermercados de autoservicio y los tratamientos médicos inteligentes (Li et al., 2021).

## 2.2 Métricas de clasificación

Las métricas de clasificación son útiles para evaluar y comparar diferentes modelos de clasificación o técnicas de aprendizaje automático, permiten comparar su rendimiento y analizar su comportamiento ajustando diferentes parámetros. Algunas métricas se basan en los resultados de la matriz de confusión (Grandini et al., 2020) ya que estos permiten medir el rendimiento de un algoritmo de aprendizaje automático (Martínez-Toro et al., 2019). Las métricas de evaluación se emplean además para medir y resumir la calidad del clasificador entrenado cuando se prueba con los datos no vistos. En el presente trabajo se utilizan las métricas listadas a continuación para comparar el rendimiento de cada uno de los modelos propuestos.

- Exactitud (Accuracy): es la suma de las predicciones verdaderas, tanto positivas como negativas, dividida por el número total de predicciones hechas. Permite saber qué tan exacto es el resultado y ofrece información sobre los posibles errores que se pueden encontrar en la clasificación (Hernández et al., 2022).
- Precisión (Precision): es la fracción de elementos positivos verdaderos dividida por el número total de unidades predichas como positivas. Expresa la proporción de unidades que el modelo predice que son positivas y en realidad lo son, o sea, esta métrica dice cuánto se puede confiar en el modelo cuando predice que un individuo es positivo. En otras palabras, dentro de todas las predicciones positivas del clasificador, qué porcentaje de estos en realidad son positivos (Llumiñana Almeida, 2022).
- Exhaustividad (Recall): se emplea para medir la fracción de elementos positivos clasificados correctamente, mide la capacidad del modelo para encontrar todas las unidades positivas en el conjunto de datos; en otras palabras, mide la proporción de verdaderos positivos frente a falsos negativos.
- F1-score: representa la media armónica de precisión y exhaustividad, que proporciona una medida equilibrada del rendimiento general de un clasificador. Por ejemplo, si la precisión es del 80% y la recuperación es del 70%, la puntuación F1 es del 75% (Kalmegh & Padar, 2023). Esta es una métrica muy utilizada en problemas en los que el conjunto de datos a analizar está desbalanceado.

Se seleccionaron las métricas listadas porque permiten medir de manera imparcial y rigurosa el desempeño de los algoritmos de clasificación. Las métricas seleccionadas en el trabajo resultan útiles para probar la capacidad de un clasificador de clases múltiples, permitiendo comparar el rendimiento de modelos diferentes o analizar el comportamiento del mismo modelo ajustando diferentes parámetros (Grandini et al., 2020), además son frecuentemente utilizadas por investigadores, especialmente en conjuntos de datos con clases balanceadas (Borja-Robalino et al., 2020).

## 2.3 Hiperparámetros de algoritmos propuestos

El ajuste de hiperparámetros de un algoritmo de aprendizaje automático (ML) es una característica del proceso de aprendizaje del modelo. Los hiperparámetros suelen ser de varios tipos, discretos, categóricos y continuos, por lo que el mecanismo de ajuste es diferente entre los algoritmos de ML. Grid search es un estándar de optimización de parámetros existente en el aprendizaje automático, se basa en una búsqueda exhaustiva a través de un subconjunto, definido manualmente, del espacio de hiperparámetros del algoritmo de aprendizaje.

Existen otras propuestas, como la búsqueda aleatoria, la optimización bayesiana y la optimización de gradientes. Sin embargo, debido a su facilidad de ejecución y paralelización, su durabilidad en espacios de baja dimensión, grid search prevalece como el estado del arte para la optimización de hiperparámetros (Belete & Huchaiah, 2022). En la tabla 2 se muestra la configuración de hiperparámetros definida para cada algoritmo.

Algoritmo	Hiperparámetros
KNN	<b>n_neighbors:</b> [1, 3, 5, 7, 9, 11] <b>metric:</b> [euclidean, manhattan, minkowski, chebyshev, cosine, jaccard, hamming] <b>weights:</b> [uniform, distance] <b>leaf_size:</b> [5, 10, 20, 30, 35, 40, 45, 50] <b>algorithm:</b> [brute, kd_tree, ball_tree]
NB	<b>var_smoothing:</b> np.logspace(0, -9, num=1000)
LR	<b>penalty:</b> [l1, l2, elasticnet ] <b>C:</b> np.logspace(-3,3,60) <b>class_weight:</b> [balanced, None] <b>solver:</b> [ lbfgs, liblinear, newton-cg, newton-cholesky, sag, saga]
SVM	<b>kernel:</b> [ rbf, poly, sigmoid ] <b>gamma:</b> [ 1e-1, 1e-2, 1e-3, 1e-4, 1e-5, 1e-6, scale, auto] <b>C:</b> [1, 10, 100, 1000] <b>decision_function_shape:</b> [ovo, ovr]
DT	<b>criterion:</b> [gini, entropy, log_loss] <b>splitter:</b> [best, random] <b>min_samples_split:</b> [2,3,4,5,6,7,8,16] <b>class_weight:</b> [balanced, None] <b>min_weight_fraction_leaf:</b> np.linspace( 0, 0.5, 100 )
RF	<b>n_estimators:</b> [10,20,30,40,50,60,70,80,100,200,500] <b>max_features:</b> [None, sqrt, log2] <b>max_depth:</b> [4,5,6,7,8] <b>criterion:</b> [gini, entropy] <b>min_samples_split:</b> [2,4,8,16,32] <b>class_weight:</b> [balanced, None]}
MLP	<b>Activation:</b> [ relu, tanh, sigmoid, hard_sigmoid, linear] <b>optimizer:</b> [SGD, RMSprop, Adagrad, Adadelta, Adam, Adamax, Nadam] <b>epochs:</b> [1, 10, 20, 30, 40, 50] <b>learning_rate:</b> [0.1, 0.0001, 0.01] <b>batch_size:</b> [10, 20, 50, 80, 100] <b>initializer:</b> [lecun_uniform, normal, he_normal, he_uniform] <b>dropout_rate:</b> [0.3, 0.2, 0.8] <b>num_unit:</b> [5, 10, 20]

Tabla 2. Configuración de hiperparámetros por algoritmo

Luego de ejecutado el algoritmo de grid search para encontrar los parámetros óptimos, según los hiperparámetros propuestos para cada algoritmo y la base de datos analizada se obtuvieron los resultados mostrados en la tabla 3.

Algoritmo	Hiperparámetros
KNN	n_neighbors: 1 metric: cosine weights: uniform leaf_size: 5 algorithm: brute
NB	var_smoothing: 0.009203731996618
LR	penalty: l1 C: 0.2758531617629184 class_weight: balanced solver: liblinear
SVM	kernel: rbf gamma: scale C: 100 decision_function_shape: ovo
DT	criterion: log_loss splitter: best min_samples_split: 3 class_weight: None min_weight_fraction_leaf: 0
RF	class_weight: None criterion: entropy max_depth: 8 max_features: log2 min_samples_split: 2 n_estimators: 500
MLP	Activation: relu Optimizers: SGD Epochs: 1 Learning Rate: 0.1 Batch Size: 20 Initializers: lecun_uniform Dropout Rate: 0.3 Number of Units: 10

Tabla 3. Mejores parámetros por algoritmo según hiperparámetros especificados

Posteriormente se procedió con la implementación de cada uno para comparar sus resultados con lo obtenidos con una red neuronal convolucional que será diseñada e implementada de igual forma en la presente investigación y con bases en la propuesta de arquitectura de red neuronal convolucional de (Wang et al., 2020)

## 2.4 Entrenamiento y prueba de algoritmos

Para el desarrollo de la aplicación se empleó una laptop con 8 gigabytes de RAM, un procesador de 64 bits Intel Core I7 2620M de segunda generación y un disco de estado sólido de 256 gigabytes. El sistema operativo empleado fue Microsoft Windows 11 de 64 bits, el entorno de programación seleccionado fue Spyder y la versión de Python fue 3.10.9.

### 2.4.1 K-Vecino más cercano

La exactitud del algoritmo KNN para el conjunto de entrenamiento fue de 1.00 con un tiempo de 0.61 segundos. Por su parte en la predicción del conjunto de pruebas el resultado de la exactitud fue de 0.97 (valor este un poco inferior al del entrenamiento) y la operación tardó un tiempo de 42.27 segundos. La matriz de confusión, mostrada en la figura 3 ofrece resultados de clasificación por símbolos que se emplearon para calcular las métricas de rendimiento.

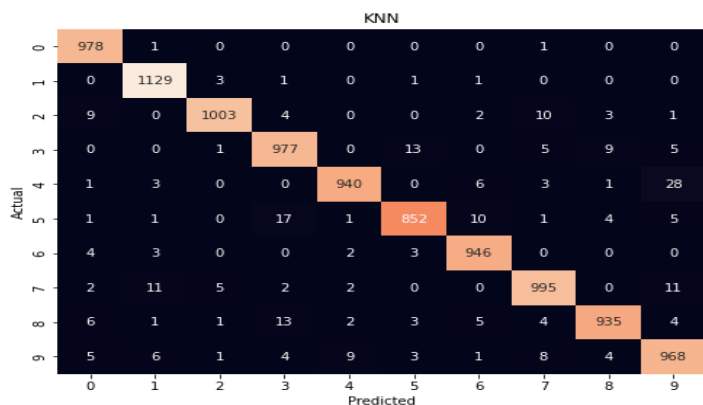


Figura 3. Matriz de confusión obtenida del conjunto de prueba mediante el algoritmo KNN

Como se puede apreciar en la figura el símbolo que más veces se clasificó incorrectamente fue el 4, que se clasificó como 9 en 28 oportunidades, seguido del 5 que se clasificó como 3 en 17 ocasiones. Los símbolos con mejores resultados de clasificación fueron el 0 y el 1. El dígito 2 obtuvo la mayor precisión, lo que indica que de los elementos clasificados como 2, el 99% realmente lo era; sin embargo, su métrica recall obtuvo valor inferior, esto sugiere que un 3% de los dígitos 2 fueron mal clasificados y un 97% bien clasificados. La tabla 4 ofrece un resumen de las métricas de clasificación calculadas a partir de la matriz de confusión, los valores fueron redondeados por exceso dejando dos lugares decimales.

Símbolo	Precisión	Recall	F1-Score
0	0.97	1.00	0.98
1	0.98	0.99	0.99
2	0.99	0.97	0.98
3	0.96	0.97	0.96
4	0.98	0.96	0.97
5	0.97	0.96	0.96
6	0.97	0.99	0.98
7	0.97	0.97	0.97
8	0.98	0.96	0.97
9	0.95	0.96	0.95

Tabla 4. Resumen de métricas para algoritmo KNN

### 2.4.2 Naïve Bayes (NB)

Luego de implementar el algoritmo NB y entrenarlo con ajuste óptimo obtenido según hiperparámetros definidos, el algoritmo ofreció una exactitud del 0.78, en un tiempo de 0.9754 segundos. La predicción de todo el conjunto de datos ofreció una exactitud de 0.79, ligeramente superior al entrenamiento, requiriendo para ello de un tiempo de 0.8727 segundos. En la figura 4 se puede apreciar la matriz de confusión.

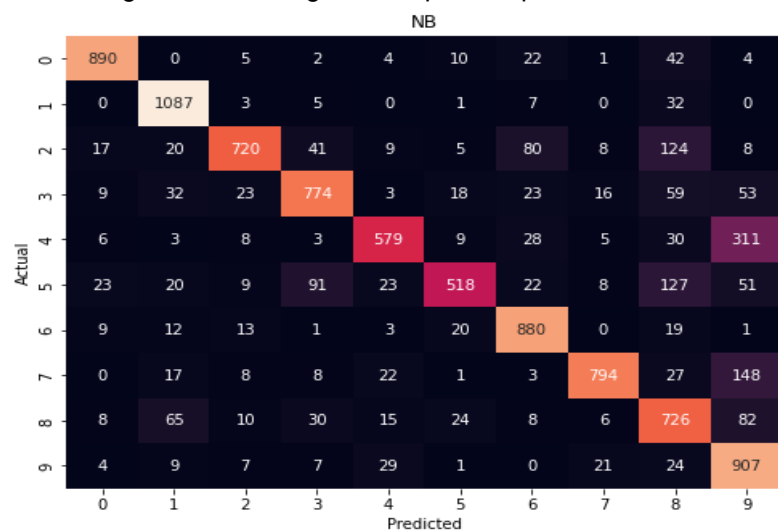


Figura 4. Matriz de confusión obtenida del conjunto de prueba mediante el algoritmo NB

El símbolo con mayor problema resultó ser el 5, que fue mal clasificado en 374 ocasiones, 127 veces de estas fue clasificado incorrectamente como 8. El símbolo con mejores resultados fueron el 0 y el 1, cabe resaltar que este para tampoco fue confundido entre sí en ningún momento. La tabla 5 ofrece el resultado detallado de métricas por símbolo.

Símbolo	Precisión	Recall	F1-Score
0	0.92	0.91	0.91
1	0.86	0.96	0.91
2	0.89	0.70	0.78
3	0.80	0.77	0.78
4	0.84	0.59	0.69
5	0.85	0.58	0.69
6	0.82	0.92	0.87
7	0.92	0.77	0.84
8	0.60	0.75	0.66
9	0.58	0.90	0.70

Tabla 5. Resumen de métricas para algoritmo NB

### 1.4.3 Regresión Logística

El algoritmo ofreció una exactitud de 0.92 durante el entrenamiento, consumiendo un tiempo de 39.5372 segundos. La predicción de todo el conjunto de pruebas le tomó un tiempo de 0.0420 segundos y ofreció una exactitud de 0.92 exactamente igual al entrenamiento. La matriz de confusión se muestra en la figura 5.

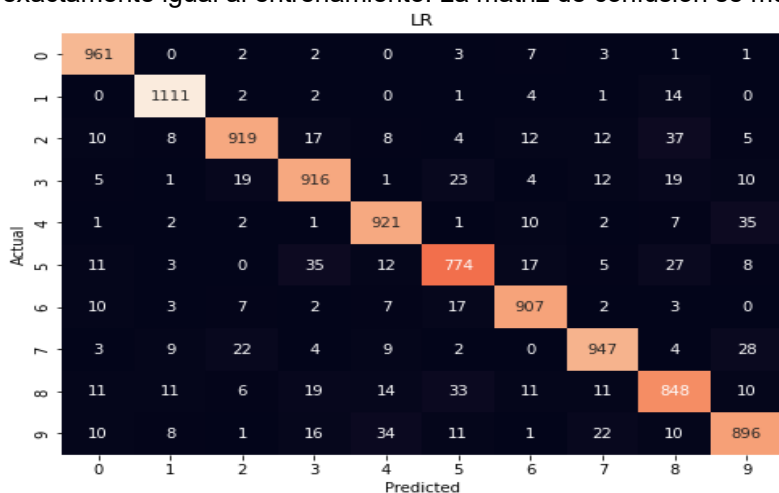


Figura 5. Matriz de confusión obtenida del conjunto de prueba mediante el algoritmo LR

El algoritmo ofreció una tasa aceptable de reconocimiento tanto durante el entrenamiento como durante la prueba. Los símbolos 5, 8, 9 y 2 ofrecieron la tasa más baja de reconocimiento correcto. La tabla 6 muestra un resumen de las métricas obtenidas de la matriz de confusión. Los símbolos mejores resultados de clasificación resultaron ser el 0 y el 1.

Símbolo	Precisión	Recall	F1-Score
0	0.94	0.98	0.96
1	0.96	0.98	0.97
2	0.94	0.89	0.91
3	0.90	0.91	0.91
4	0.92	0.94	0.93
5	0.89	0.87	0.88
6	0.93	0.95	0.94
7	0.93	0.92	0.93
8	0.87	0.87	0.87
9	0.90	0.89	0.90

Tabla 6. Resumen de métricas para algoritmo LR

### 1.4.4 Máquina de soporte vectorial

Este algoritmo ofreció una exactitud al conjunto de entrenamiento de 1.00 y la ejecución de toda la instrucción de entrenamiento al conjunto se tomó un tiempo de 217.6239 segundos. La predicción del conjunto de prueba tardó 96.3754 segundos y arrojó una exactitud de 0.98. La figura 6 muestra la matriz de confusión obtenida de la predicción de todo el conjunto de prueba. Los resultados de este algoritmo resultaron excelentes tanto para el conjunto de entrenamiento como para el de prueba, sin embargo, el tiempo requerido para realizar ambas labores fue muy alto. Las precisiones más bajas las obtuvieron los dígitos 5 y 8, con valores inferiores a 0,9. Los símbolos mejor clasificados fueron el 0 y el 1.

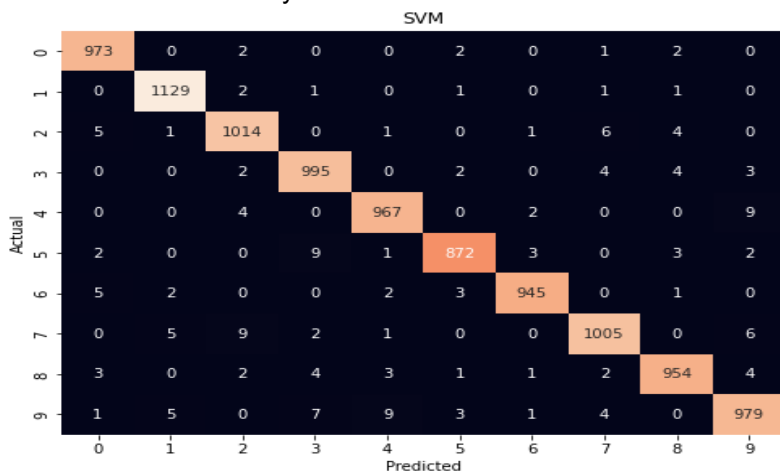


Figura 6. Matriz de confusión obtenida del conjunto de prueba mediante el algoritmo SVM

La tabla 7 muestra el resumen de las métricas de rendimiento de clasificación calculados a partir de la matriz de confusión.

Símbolo	Precisión	Recall	F1-Score
0	0.94	0.98	0.96
1	0.96	0.98	0.97
2	0.94	0.89	0.91
3	0.90	0.91	0.91
4	0.92	0.94	0.93
5	0.89	0.87	0.88
6	0.93	0.95	0.94
7	0.93	0.92	0.93
8	0.87	0.87	0.87
9	0.90	0.89	0.90

Tabla 7. Resumen de métricas para algoritmo SVM

### 1.4.5 Árbol de clasificación (DT)

El algoritmo ofreció una exactitud de 1.00 para el conjunto de entrenamiento, tardando en acometer la tarea, un tiempo de 21.97288 segundos. Para la predicción del conjunto de prueba, la exactitud descendió notablemente hasta 0.88 en un tiempo de 0.0270 segundos. La figura 7 muestra la matriz de confusión obtenida de la predicción del conjunto de prueba.

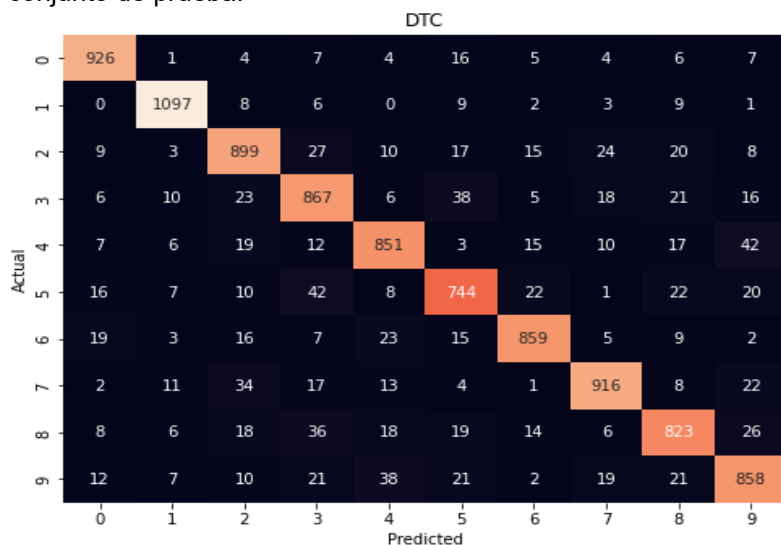


Figura 7. Matriz de confusión obtenida del conjunto de prueba mediante el algoritmo SVM

Nuevamente los símbolos mejor clasificados resultaron ser el 0 y el 1 y los que más problemas causaron durante la clasificación fueron el 5 y el 8. La tabla 8 ofrece los resultados de las métricas por cada símbolo.

Símbolo	Precisión	Recall	F1-Score
0	0.94	0.98	0.96
1	0.96	0.98	0.97
2	0.94	0.89	0.91
3	0.90	0.91	0.91
4	0.92	0.94	0.93
5	0.89	0.87	0.88
6	0.93	0.95	0.94
7	0.93	0.92	0.93
8	0.87	0.87	0.87
9	0.90	0.89	0.90

Tabla 8. Resumen de métricas para algoritmo DT

### 1.4.6 Bosque aleatorio (RF)

La exactitud para este algoritmo fue de 0.93 para el conjunto de entrenamiento, requiriendo para este proceso 14.8594 segundos. La predicción del conjunto de pruebas obtuvo una exactitud de 0.92 en un lapso de 0.2822 segundos. La figura 8 muestra la matriz de confusión ofrecida para la predicción del conjunto de pruebas por el algoritmo.



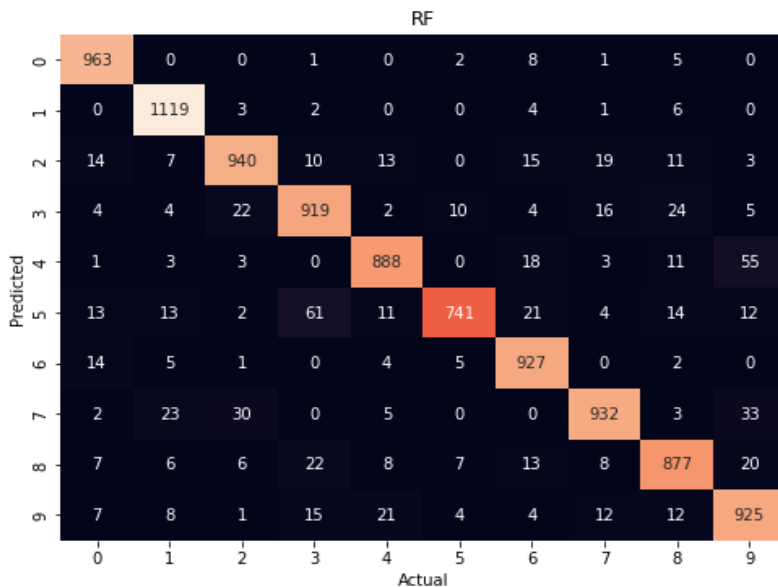


Figura 8. Matriz de confusión obtenida del conjunto de prueba mediante el algoritmo RF

Los símbolos que mayor confusión generaron durante el proceso de clasificación del conjunto de pruebas fueron el 3, 8, 9 y 5; por el contrario, los de mejores resultados fueron el 0 y el 1. Los tres símbolos más confundidos durante el proceso de clasificación fueron el 5 que clasificó como 3 en 61 ocasiones, el 4 que fue clasificado como 9 en 55 ocasiones y el 7 que fue clasificado como 9 en 33 ocasiones.

Símbolo	Precisión	Recall	F1-Score
0	0.95	0.98	0.96
1	0.93	0.99	0.96
2	0.93	0.91	0.92
3	0.89	0.92	0.90
4	0.93	0.91	0.92
5	0.96	0.87	0.91
6	0.93	0.96	0.94
7	0.93	0.90	0.92
8	0.91	0.88	0.90
9	0.88	0.91	0.90

Tabla 8. Resumen de métricas para algoritmo RF

### 1.4.7 Perceptrón Multicapa (MLP)

La estructura de la red neuronal multicapa es la mostrada en la figura 9, que utiliza un modelo secuencial. La capa de entrada cuenta con 512 neuronas y función de activación rectificación lineal. La segunda capa consta de 768 neuronas y función de activación rectificación lineal al igual que la capa precedente. La capa de salida consta de 10 neuronas y emplea función de activación softmax. La variable de las categorías `y_train` fue codificada mediante one hot encoding. La red neuronal propuesta cuenta con 803594 parámetros, todos estos entrenables. Se empleó el optimizador SGD, un total de 20 épocas y el tamaño del lote de validación seleccionado fue de 20.

Model: "sequential\_1"

Layer (type)	Output Shape	Param #
dense_1 (Dense)	(None, 512)	401920
dense_2 (Dense)	(None, 768)	393984
dense_3 (Dense)	(None, 10)	7690

=====  
Total params: 803,594  
Trainable params: 803,594  
Non-trainable params: 0

Figura 9. Resumen del modelo de perceptrón multicapa

El algoritmo propuesto obtuvo una exactitud de 0.9946 durante el entrenamiento, el tiempo consumido durante este proceso fue de 434.8619 segundos, la validación por lote ofreció una exactitud del 0.9817. El proceso de prueba duró un tiempo de 1.9344 segundos y devolvió una exactitud de 0.98. La precisión del algoritmo fue muy buena para todos los dígitos obteniendo para cada uno un valor entre 0.97 y 0.99. El desempeño del modelo durante las 20 épocas del entrenamiento fue adecuado, la pérdida fue disminuyendo por etapas tanto para el conjunto de entrenamiento como para el de validación de lote, así como también la exactitud del algoritmo aumentó de manera progresiva en cada etapa. Para más información ver figura 10.

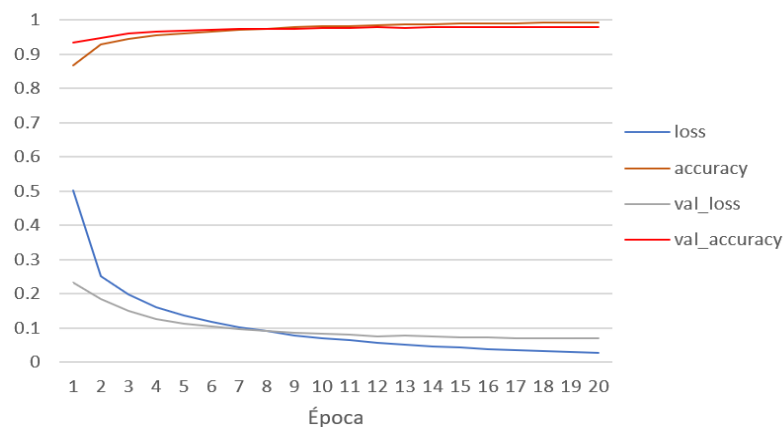


Figura 10. Gráfica de pérdida y exactitud por épocas

La figura 11 muestra la matriz de confusión obtenida para el conjunto de prueba. Por el eje vertical se pueden observar los valores reales y por el eje horizontal se encuentran los valores predichos por el algoritmo.

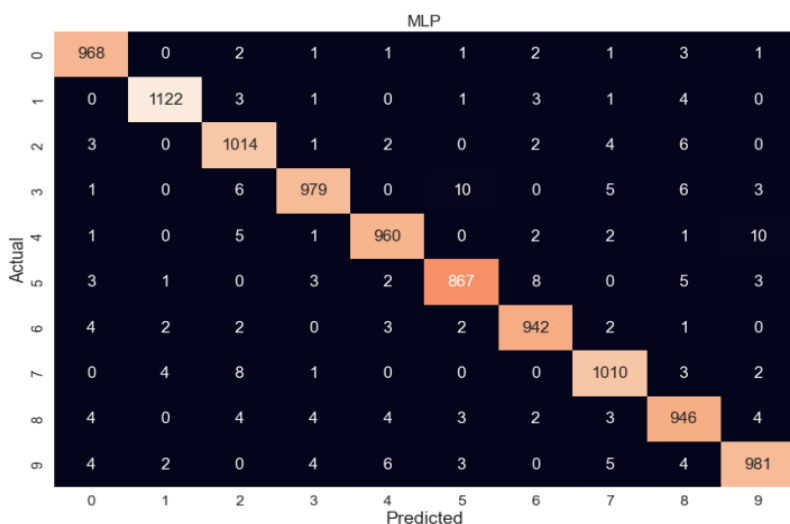


Figura 11. Matriz de confusión obtenida del conjunto de prueba mediante el algoritmo MLP

El símbolo con mejor tasa de reconocimiento fue el 1 y el 0. Los símbolos más mal clasificados fueron el 3, el 8 y el 9. La tabla 9 muestra los valores de las métricas de precisión obtenidas a partir de la matriz de confusión.

Símbolo	Precisión	Recall	F1-Score
0	0.98	0.99	0.98
1	0.99	0.99	0.99
2	0.97	0.98	0.98
3	0.98	0.97	0.98
4	0.98	0.98	0.98
5	0.98	0.97	0.97
6	0.98	0.98	0.98
7	0.98	0.98	0.98
8	0.97	0.97	0.97
9	0.98	0.97	0.97

Tabla 9. Resumen de métricas para algoritmo MLP

### 2.4.8 Red neuronal convolucional (CNN)

La red propuesta posee una estructura como la mostrada en la figura 11. Esta compuesta por una primera capa convolucional con máscara de 3x3, con 64 neuronas, una segunda capa convolucional con iguales características, una capa oculta de 512 neuronas, otra capa oculta con 256 neuronas y todas estas con función de activación rectificación lineal. La capa de salida está compuesta por 10 neuronas y emplea la función de activación softmax.

Model: "sequential\_29"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 26, 26, 64)	640
conv2d_1 (Conv2D)	(None, 24, 24, 64)	36928
max_pooling2d (MaxPooling2D)	(None, 12, 12, 64)	0
flatten (Flatten)	(None, 9216)	0
dense_87 (Dense)	(None, 512)	4719104
dense_88 (Dense)	(None, 256)	131328
dense_89 (Dense)	(None, 10)	2570

=====  
 Total params: 4,890,570  
 Trainable params: 4,890,570  
 Non-trainable params: 0

Figura 11. Resumen del modelo de red convolucional propuesta

El algoritmo ofreció exactitud de 0.9998 durante el entrenamiento en un tiempo prudencial de 13191 segundos. Luego de clasificado el conjunto de prueba la exactitud del algoritmo descendió hasta 98.30, esta operación se completó en un tiempo de 15.90 segundos. En la figura 12 se muestra la matriz de confusión resultante del análisis del conjunto de entrenamiento.

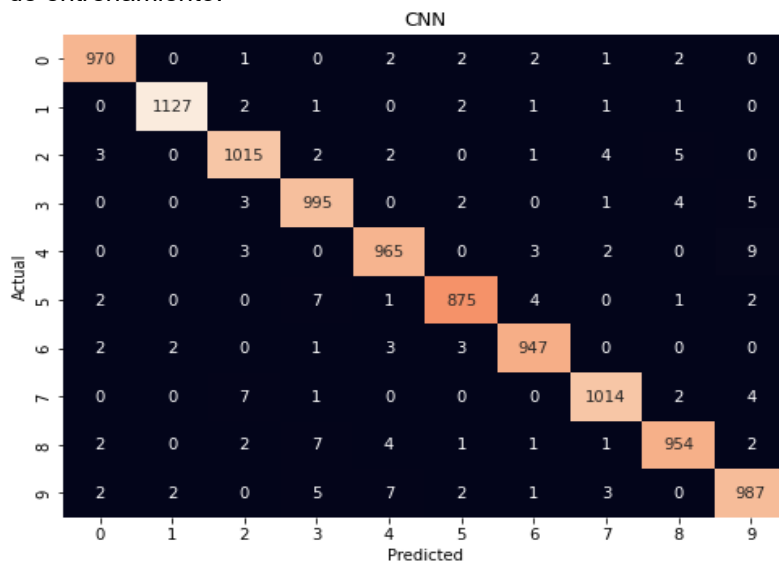


Figura 12. Matriz de confusión obtenida del conjunto de prueba mediante el algoritmo CNN

Los símbolos con mejor tasa de clasificación según los resultados de las métricas mostrados en la tabla 10 fueron el 1 y el 0. Los resultados fueron muy buenos para todos los símbolos, obteniendo valores de métricas entre 0.98 y 1.00 para cada uno.

Símbolo	Precisión	Recall	F1-Score
0	0.99	0.99	0.99
1	1.00	0.99	0.99
2	0.98	0.98	0.98
3	0.98	0.99	0.98
4	0.98	0.98	0.98
5	0.99	0.98	0.98
6	0.99	0.99	0.99
7	0.99	0.99	0.99
8	0.98	0.98	0.98
9	0.98	0.98	0.98

Tabla 10. Resumen de métricas para algoritmo CNN

### 3. Resultados

Luego de analizar los desempeños de los algoritmos se puede corroborar que varios de estos ofrecieron resultados de exactitud superior al 95% durante el entrenamiento, la figura 13 muestra un gráfico con los resultados de exactitud por cada modelo en las etapas de entrenamiento y prueba.

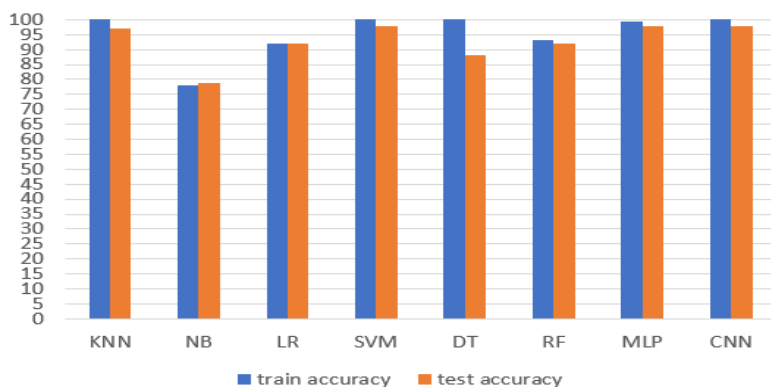


Figura 13. Muestra los resultados de exactitud por cada algoritmo durante las etapas de entrenamiento y prueba

Los algoritmos que superaron tanto para el conjunto de entrenamiento como para el de prueba el 95% de exactitud fueron KNN, SVM, MLP y CNN. Un caso particular fue el de DT que clasificó correctamente el 100% de los objetos durante el entrenamiento; sin embargo, su resultado descendió notablemente para el conjunto de prueba, lo que puede intuir en un modelo no ajustado completamente a los datos o sobreajuste, ya que el algoritmo memorizó los datos del entrenamiento no pudiendo clasificar correctamente nuevas muestras no analizadas durante el entrenamiento. Si se hace un análisis particularizado de la precisión que cada modelo estimó al conjunto de prueba por cada uno de sus símbolos, se puede corroborar que los modelos que mejores resultados tuvieron en esa métrica fueron de igual manera KNN, SVM, MLP y CNN. Los símbolos 0 y 1 obtuvieron el mayor valor de precisión en todos los algoritmos analizados. Ver figura 14 para mayor comprensión.

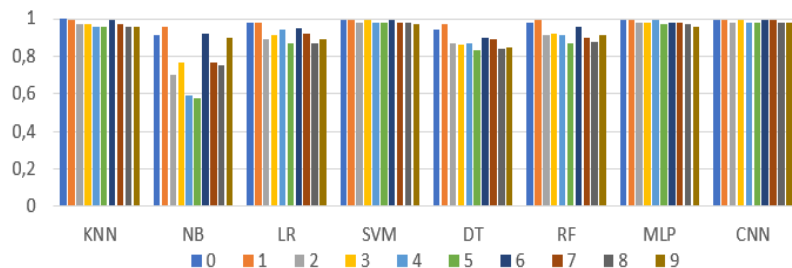


Figura 14. Precisión modelo/símbolos

Si se considera el consumo de tiempo por cada algoritmo, durante el entrenamiento, en orden ascendente se tiene la siguiente lista KNN, NB, RF, DT, LR, SVM, MLP y CNN, la figura 15 muestra un resumen del consumo de tiempo en escala logarítmica de los modelos.

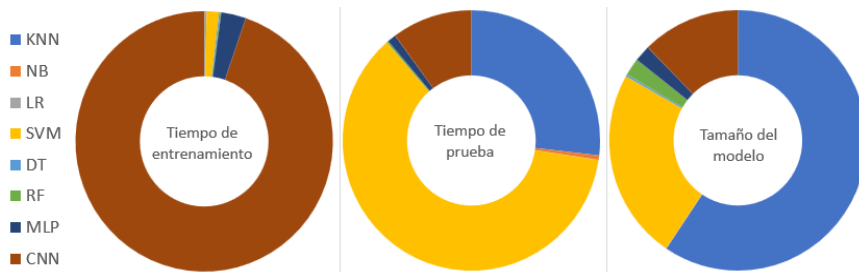


Figura 15. Consumo de tiempo y almacenamiento por modelo

En cuanto al tiempo consumido por los algoritmos para procesar los datos de prueba, los resultados por modelos en orden ascendente son los siguientes: DT, LR, RF, NB, MLP, CNN, KNN y SVM. Si se analiza el tamaño en kilobytes de los modelos en orden ascendente, se tiene la siguiente sucesión: LR, NB, DT, MLP, RF, CNN, SVM y KNN. Ver tabla 11 para más información.

Algoritmo	Tiempo de entrenamiento (segundos)	Tiempo de prueba (segundos)	Tamaño del modelo (kilobytes)
KNN	0.6127	42.2709	183986
NB	0.9755	0.8727	124
LR	39.5372	0.042	63
SVM	217.6239	96.3754	73582
DT	21.9729	0.027	810
RF	14.8569	0.3141	6693
MLP	434.8619	1.7201	6305
CNN	13190.9868	15.904	38249

Tabla 11. Consumo de recursos de los algoritmos analizados

#### 4. Conclusiones

El presente estudio permitió analizar el desempeño de 8 algoritmos muy empleados en las ciencias de la computación para tareas de inteligencia artificial como clasificación y reconocimiento de patrones. Los resultados ofrecidos están sujetos a los parámetros establecidos a cada algoritmo, y estos a su vez están relacionados con la búsqueda exhaustiva mediante grid search sobre los hiperparámetros definidos al inicio de la investigación.

Se pudieron obtener las métricas de clasificación para cada algoritmo obteniendo como resultado que 4 algoritmos superaron el 97% de exactitud tanto durante el entrenamiento como durante la prueba, estos fueron KNN, SVM, MLP y CNN, ubicándolos como los de mejor desempeño para el conjunto de datos analizado MNIST con ligeras diferencias entre estos. La métrica de precisión de los símbolos durante el proceso de prueba del conjunto de datos mostró valores elevados para los 4 algoritmos anteriores resultados, lo mismo sucedió con la métrica de exhaustividad y F1-Score. Los tiempos de entrenamiento arrojaron una diferencia bien marcada entre los algoritmos como se aprecia en la figura 16, consumiendo la mayor cantidad de tiempo el algoritmo CNN, seguido por MLP, SVM y KNN en ese orden. En el caso del consumo de tiempo en la etapa de prueba, los algoritmos que más tiempo emplearon fueron SVM, KNN, CNN y MLP en ese orden. El tamaño del almacenamiento ocupado por el modelo entrenado fue otro elemento tomado en consideración, el modelo que más espacio consumió fue KNN, seguido por SVM, CNN y finalmente MLP.

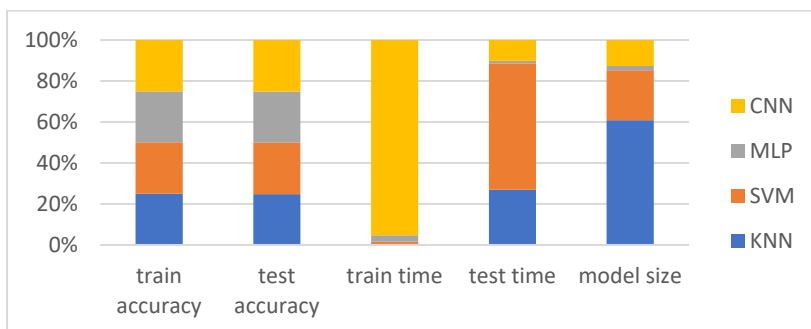


Figura 16. Resultados de exactitud, tiempo y memoria por algoritmo

El algoritmo que mejores resultados ofreció de manera general para las métricas, los tiempos de entrenamiento y prueba, así como de consumo de espacio en almacenamiento fue el de MLP, ofreciendo un balance entre todos los elementos medidos. Este algoritmo brindó una exactitud de 0.9946 y 0.98 para el entrenamiento y prueba respectivamente, consumiendo de manera moderada los recursos de tiempo y memoria, lo que lo convierte en el principal candidato a considerar para la implementación de una aplicación móvil que garantice el equilibrio entre el consumo de tiempo y almacenamiento manteniendo valores altos de exactitud.

## REFERENCIAS BIBLIOGRÁFICAS

- Alarcón Flores, J. B. (2017). Modelos de minería de datos: Random forest y adaboost, para identificar los factores asociados al uso de las TIC (internet, telefonía Fija y televisión de paga) en los hogares del Perú. 2014. <https://cybertesis.unmsm.edu.pe/handle/20.500.12672/7404>
- Baldominos Gómez, A., Sáez Achaerandio, Y., & Isasi Viñuela, P. (2019). A survey of handwritten character recognition with MNIST and EMNIST. <https://e-archivo.uc3m.es/handle/10016/38329>
- Belete, D. M., & Huchaiah, M. D. (2022). Grid search in hyperparameter optimization of machine learning models for prediction of HIV/AIDS test results. *International Journal of Computers and Applications*, 44(9), 875-886.
- Borja-Robalino, R., Monleon-Getino, A., & Rodellar, J. (2020). Estandarización de métricas de rendimiento para clasificadores Machine y Deep Learning. *Revista Ibérica de Sistemas e Tecnologías de Informação*, E30, 184-196.
- Braga-Neto, U. (2020). *Fundamentals of pattern recognition and machine learning*. Springer.
- Cho, G., Yim, J., Choi, Y., Ko, J., & Lee, S.-H. (2019). Review of machine learning algorithms for diagnosing mental illness. *Psychiatry investigation*, 16(4), 262.
- Dhillon, A., & Verma, G. K. (2020). Convolutional neural network: A review of models, methodologies and applications to object detection. *Progress in Artificial Intelligence*, 9(2), 85-112.
- Ehsani, R., & Drabløs, F. (2020). Robust Distance Measures for kNN Classification of Cancer Data. *Cancer Informatics*, 19, 1176935120965542. <https://doi.org/10.1177/1176935120965542>
- Gardner, M. W., & Dorling, S. R. (1998). Artificial neural networks (the multilayer perceptron)—A review of applications in the atmospheric sciences. *Atmospheric Environment*, 32(14), 2627-2636. [https://doi.org/10.1016/S1352-2310\(97\)00447-0](https://doi.org/10.1016/S1352-2310(97)00447-0)
- Gasparetto, A., Marcuzzo, M., Zangari, A., & Albarelli, A. (2022). A survey on text classification algorithms: From text to predictions. *Information*, 13(2), 83.
- Grandini, M., Bagli, E., & Visani, G. (2020). Metrics for multi-class classification: An overview. arXiv preprint [arXiv:2008.05756](https://arxiv.org/abs/2008.05756).
- Griffis, J. C., Allendorfer, J. B., & Szaflarski, J. P. (2016). Voxel-based Gaussian naïve Bayes classification of ischemic stroke lesions in individual T1-weighted MRI scans. *Journal of neuroscience methods*, 257, 97-108.
- Gutiérrez Esparza, G. O., Margain Fuentes, M. de L., Ramírez del Real, T. A., & Canul Reich, J. (2017). Un modelo basado en el Clasificador Naïve Bayes para la evaluación del desempeño docente. RIED. *Revista iberoamericana de educación a distancia*.
- Hernández, A. Z., Rosales, G. A. G., Santiago, H. J. J., & Lee, M. M. (2022). Métricas de rendimiento para evaluar el aprendizaje automático en la clasificación de imágenes petroleras utilizando redes neuronales convolucionales. *Ciencia Latina Revista Científica Multidisciplinar*, 6(5), 4624-4637.
- Kalmegh, S. R., & Padar, B. R. (2023). Empirical Study on Evaluation Metrics for Classification Algorithms. *International Journal of Advanced Research in Science, Communication and Technology*.
- Kirişci, M. (2019). Comparison of artificial neural network and logistic regression model for factors affecting birth weight. *SN Applied Sciences*, 1(4), 378.
- Lévano-Rodríguez, D., & Cerdán-León, F. E. (2022). Discriminación de masas mamográficas mediante K-Nearest Neighbor y atributos BIRADS. *Revista Científica de Sistemas e Informática*, 2(1), Article 1. <https://doi.org/10.51252/rcsi.v2i1.225>
- Li, Z., Liu, F., Yang, W., Peng, S., & Zhou, J. (2021). A survey of convolutional neural networks: Analysis, applications, and prospects. *IEEE transactions on neural networks and learning systems*.
- Llumiquinga Almeida, E. P. (2022). Estudio comparativo de los algoritmos de clasificación supervisada empleando datos artificiales. [B.S. thesis]. Quito, 2022.

- Luu, S. T., Nguyen, H. P., Van Nguyen, K., & Nguyen, N. L.-T. (2020). Comparison between traditional machine learning models and neural network models for vietnamese hate speech detection. 2020 RIVF International Conference on Computing and Communication Technologies (RIVF), 1-6.
- Martínez-Toro, G. M., Rico-Bautista, D., & Romero-Riaño, E. (2019). Análisis comparativo de predicción dentro de bases de datos de cáncer: Una aplicación de aprendizaje automático. *Revista Ibérica de Sistemas e Tecnologías de Informação*, E17, 113-122.
- Merenda, M., Porcaro, C., & Iero, D. (2020). Edge machine learning for ai-enabled iot devices: A review. *Sensors*, 20(9), 2533.
- Naranjo, L. T. L. (2022). Diseño de una interfaz cerebro computador (BCI) para la interacción con un sistema de rehabilitación de miembro superior.
- Ray, S. (2019). A quick review of machine learning algorithms. 2019 International conference on machine learning, big data, cloud and parallel computing (COMITCon), 35-39.
- Ruiz-Shulcloper, J., Guzmán Arenas, A., & Martínez-Trinidad, J. F. (1999). Enfoque lógico combinatorio al reconocimiento de patrones. I. Selección de Variables y Clasificación Supervisada, IPN, México.
- Sen, P. C., Hajra, M., & Ghosh, M. (2020). Supervised Classification Algorithms in Machine Learning: A Survey and Review. En J. K. Mandal & D. Bhattacharya (Eds.), *Emerging Technology in Modelling and Graphics* (Vol. 937, pp. 99-111). Springer Singapore. [https://doi.org/10.1007/978-981-13-7403-6\\_11](https://doi.org/10.1007/978-981-13-7403-6_11)
- Sheykhmousa, M., Mahdianpari, M., Ghanbari, H., Mohammadimanesh, F., Ghamisi, P., & Homayouni, S. (2020). Support vector machine versus random forest for remote sensing image classification: A meta-analysis and systematic review. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13, 6308-6325.
- Tanveer, M., Rajani, T., Rastogi, R., Shao, Y.-H., & Ganaie, M. A. (2022). Comprehensive review on twin support vector machines. *Annals of Operations Research*, 1-46.
- Wang, Y., Li, F., Sun, H., Li, W., Zhong, C., Wu, X., Wang, H., & Wang, P. (2020). Improvement of MNIST Image Recognition Based on CNN. *IOP Conference Series: Earth and Environmental Science*, 428(1), 012097. <https://doi.org/10.1088/1755-1315/428/1/012097>
- Yann, L. (1998). The mnist database of handwritten digits. R.
- Zhang, X.-Y., Liu, C.-L., & Suen, C. Y. (2020). Towards robust pattern recognition: A review. *Proceedings of the IEEE*, 108(6), 894-922.



Esta obra está bajo una licencia de Creative Commons Reconocimiento-NoComercial-CompartirIgual 2.5 México.

## **Implementaciones actuales del modelo confianza cero para entornos en la nube: una revisión sistemática**

**Current implementations of the zero trust model for cloud environments:  
a systematic review**

**Johann Castillo Oliva<sup>1</sup>**

**Bruno Hiroshi Espinosa Luna<sup>1</sup>**

**Alberto Carlos Mendoza de los Santos<sup>1</sup>**

<sup>1</sup>Universidad Nacional de Trujillo



## Resumen

La expansión de la computación en la nube en las organizaciones modernas enfrenta nuevos desafíos de seguridad de la información, en consecuencia, surge el paradigma de Confianza Cero como un método de reforzamiento para entornos en la nube. En relación a ello, esta revisión sistemática busca abordar dos preguntas de investigación: ¿Cuáles son las herramientas y procedimientos empleados en los últimos 5 años en las implementaciones de Confianza Cero para entornos de la nube? y ¿Cuáles son los métodos de evaluación utilizados en el modelo de Confianza Cero para entornos de la nube? Bajo la metodología PRISMA 2020, se analizaron 13 estudios de las bases de datos bibliográficas Scopus y Dimensions donde se destacan herramientas como blockchain, algoritmos criptográficos y modelos integrales de confianza, además de métodos de validación como pruebas de funcionalidad y análisis de seguridad. Por último, se da a conocer la falta de un método de evaluación uniforme para las aplicaciones del modelo Confianza Cero en la computación en la nube.

**Palabras clave:** control de accesos, seguridad de la información, seguridad de la nube, políticas de control, innovaciones

## Abstract

The expansion of cloud computing in modern organizations faces new information security challenges; consequently, the Zero Trust paradigm emerges as a reinforcement method for cloud environments. In this regard, this systematic review aims to address two research questions: What are the tools and procedures employed in the last 5 years in Zero Trust implementations for cloud environments? And what evaluation methods are used in the Zero Trust model for cloud environments? Under the PRISMA 2020 methodology, 13 studies from the Scopus and Dimensions bibliographic databases were analyzed, highlighting tools such as blockchain, cryptographic algorithms, and comprehensive trust models, as well as validation methods such as functionality tests and security analysis. Finally, it is highlighted that there is a lack of a uniform evaluation method for Zero Trust model applications in cloud computing.

**Keywords:** Access control, information security, cloud security, control policies, innovations

## 1. Introducción

La computación en la nube (CN en adelante) ha sido ampliamente adoptada por organizaciones modernas en los últimos años, abarcando desde pequeñas empresas emergentes hasta las más grandes (Guo et al., 2023; Lawan et al., 2021). Esta tendencia radica en los beneficios tangibles e intangibles que ofrece, como la reducción de costos operativos y de mantenimiento, ahorro de espacio, optimización de recursos, mayor escalabilidad y flexibilidad, aumento de satisfacción de empleados, entre otros. Además, factores recientes, como el incremento del trabajo remoto impulsan la adopción de un entorno en la nube (Bajdor, 2022; Mandal et al., 2021; Shetty & Rajesh, 2021; Zheng et al., 2023).

Sin embargo, a medida que la CN se populariza en las organizaciones modernas, surgen nuevas vulnerabilidades y amenazas para la seguridad de la información. Por lo tanto, resulta necesario fortalecer ciertos aspectos, incluyendo controles de acceso y la autenticación de usuarios (Akbar et al., 2023; Cheng et al., 2022; Gill et al., 2022). Adicionalmente, un factor importante para asegurar la integridad de los datos en la nube es el seguimiento de buenas prácticas por parte de los clientes (George & Sagayarajan, 2023).

En este contexto, surge el modelo de Confianza Cero (CC) como una propuesta para mejorar la seguridad de los servicios en la nube, dejando como complemento a los mecanismos pasivos de seguridad (Chen et al., 2021; S. Liu et al., 2022). El paradigma CC se basa en una postura de desconfianza, incluso de aquellos usuarios o dispositivos que pertenecen a la red de una organización estableciendo un nivel de seguridad a nivel de aplicación (Ali et al., 2022; Z. Liu et al., 2022; Rose et al., 2020), lo que conlleva a que sea considerado uno de los marcos más rigurosos para el control de acceso, especialmente cuando se complementa con tecnologías como *blockchain* y *machine learning* (Feng et al., 2023; Li et al., 2023; Rajasoundaran et al., 2021).

Expuestas estas ideas, con el objetivo de dar a conocer nuevas estrategias para fortalecer la seguridad de la información en un contexto de crecimiento en la adopción de la CN, se plantea la siguiente pregunta de investigación: ¿Cuáles son las herramientas y procedimientos que se usaron en los últimos 5 años en las implementaciones del modelo confianza cero para entornos de la nube?

Además, se destaca la importancia de validar estas herramientas y procedimientos a través de un proceso riguroso para garantizar la validez de sus resultados. Lo que conduce a la segunda pregunta de investigación: ¿Cuáles son los métodos de evaluación de las herramientas y procedimientos utilizados en el modelo confianza cero para entornos de la nube? Con esta pregunta, se busca determinar cómo se puede garantizar la efectividad y utilidad de estas herramientas y métodos analizados.

## 2. Metodología

### 2.1. Fundamentación de la metodología

La realización de esta revisión sistemática se basó en la declaración de la metodología PRISMA 2020 (Page et al., 2021), que establece un conjunto de pautas para la elaboración de estudios rigurosos con el fin de recopilar evidencia empírica y desarrollarla de forma estructurada para mejorar la calidad, transparencia y el valor de la literatura científica (Sohrabi et al., 2021).

La estructura de PRISMA consta de 27 ítems, los cuales orientaron la organización del estudio en las secciones: título, resumen, introducción, métodos, resultados, discusión y conclusiones. Siguiendo esta estructura, la redacción de la revisión empezó con el título, donde se identificó claramente el estudio como una revisión sistemática. Luego, en la introducción se dio a conocer la justificación del estudio y se presentaron las preguntas de investigación a ser abordadas.

En la sección de métodos, se describieron los criterios de inclusión y exclusión donde se realizó la eliminación de estudios duplicados y de poca relevancia, así como la selección de estudios que cumplieron el rango de antigüedad de 5 años para la revisión. También se detallaron las fuentes de información utilizadas, la estrategia de búsqueda aplicada en cada una de ellas, así como el procedimiento de selección y síntesis.

Todo esto con el objetivo de recolectar la información más relevante para dar respuesta a las preguntas de investigación planteadas. Además, esta información constituyó una base para la redacción de los resultados, discusión y conclusiones, secciones donde se llevó a cabo el resumen e interpretación general de los hallazgos. Finalmente, se presentaron las recomendaciones para futuras investigaciones.

## 2.2. Criterios de elegibilidad

La selección de estudios se realizó siguiendo criterios específicos con el propósito de identificar artículos relevantes para la temática de investigación y que fueran útiles para abordar las preguntas de investigación. Los criterios de inclusión se basaron en: (i) estudios que son de tipo artículo o artículos de conferencia, (ii) escritos en idioma inglés o español y (iii) publicados en el rango de tiempo entre 2019 y 2023. A su vez, se definieron como motivos de exclusión: (i) publicaciones no sometidas a revisión por pares de doble ciego, (ii) que no se centran en herramientas o implementaciones destinadas a entornos en la nube y (iii) estudios cuya temática se enfoca únicamente en modelos de evaluación para la arquitectura de CC en la nube.

## 2.3. Fuentes de información y estrategia de búsqueda

Las bases de datos seleccionadas como fuentes de información fueron *Scopus* y *Dimensions* consultadas por última vez el 7 de septiembre de 2023. La búsqueda se realizó en grupos de términos aplicados en los títulos, resúmenes y palabras clave relacionados con el tema central de la investigación, que fueron "Zero Trust" y "Cloud Computing". Además, con el fin de obtener resultados con mayor enfoque en la temática de investigación, se utilizaron los operadores booleanos "OR" y "AND", lo que resultó en las búsquedas específicas para cada base de datos que se detallan en la Tabla 1.

Base de datos	Términos de búsqueda
Scopus	TITLE-ABS-KEY ("Zero Trust" OR "Zero Trust security") AND (cloud AND (computing OR security OR environment OR infrastructure))
Dimensions	TITLE-ABS ("Zero Trust" OR "Zero Trust security" OR "Confianza Cero") AND ("cloud computing" OR "cloud security" OR "cloud environment" OR "cloud infrastructure" OR "computación en la nube" OR "entorno en la nube")

**Tabla 1.** Términos de búsqueda empleados

## 2.4. Proceso de selección

La selección de estudios se llevó a cabo de forma independiente por los autores, utilizando hojas de cálculo en el software Excel como herramienta de automatización. Esto permitió la eliminación de registros duplicados y el seguimiento de la exclusión progresiva de los artículos de acuerdo con los criterios de elegibilidad. Para ello, se crearon diferentes hojas de cálculo para registrar la exclusión de los artículos en cada una de las tres fases del diagrama de flujo de PRISMA: identificación, cribado e inclusión.

## 2.5. Extracción de datos y síntesis

La información se extrajo siguiendo un proceso sistemático, utilizando como herramienta de automatización el software Excel donde se emplearon tablas dinámicas para tabular los resultados de los estudios y su síntesis. En esta tabla se incluyeron los siguientes campos: título, año, enlace, tipo de documento, base de datos fuente, país, herramientas y procedimientos, y métodos de evaluación.

Adicionalmente, se utilizó la herramienta Zotero como gestor bibliográfico y biblioteca compartida donde fueron almacenados los artículos revisados. El software fue seleccionado con el fin aprovechar las funcionalidades para almacenar síntesis y otras anotaciones para cada artículo en la biblioteca de grupo, de tal forma que estas puedan ser revisadas de forma conjunta entre los autores.

## 3. Resultados

Se identificaron un total de 227 registros en las bases de datos bibliográficas consultadas, de los cuales se recuperaron 33 registros para realizar una revisión más detallada del contenido. Inicialmente, se consideraron estudios de potencial interés para la revisión; sin embargo, se descartaron aquellos que eran estudios enfocados en analizar otras propuestas de implementación del modelo CC como artículos de revisión (Divya & Sherin, 2022; Justice & Sample, 2022; Teodoro, 2022), ya que se buscaron estudios originales con un enfoque específico en herramientas y métodos de implementación. Además, se descartó un estudio cuya propuesta de implementación no estaba realmente dirigida a un entorno en la nube (Paul & Rao, 2023). Finalmente, se obtuvo una muestra de 13 artículos para la revisión como se muestra en la Figura 1.

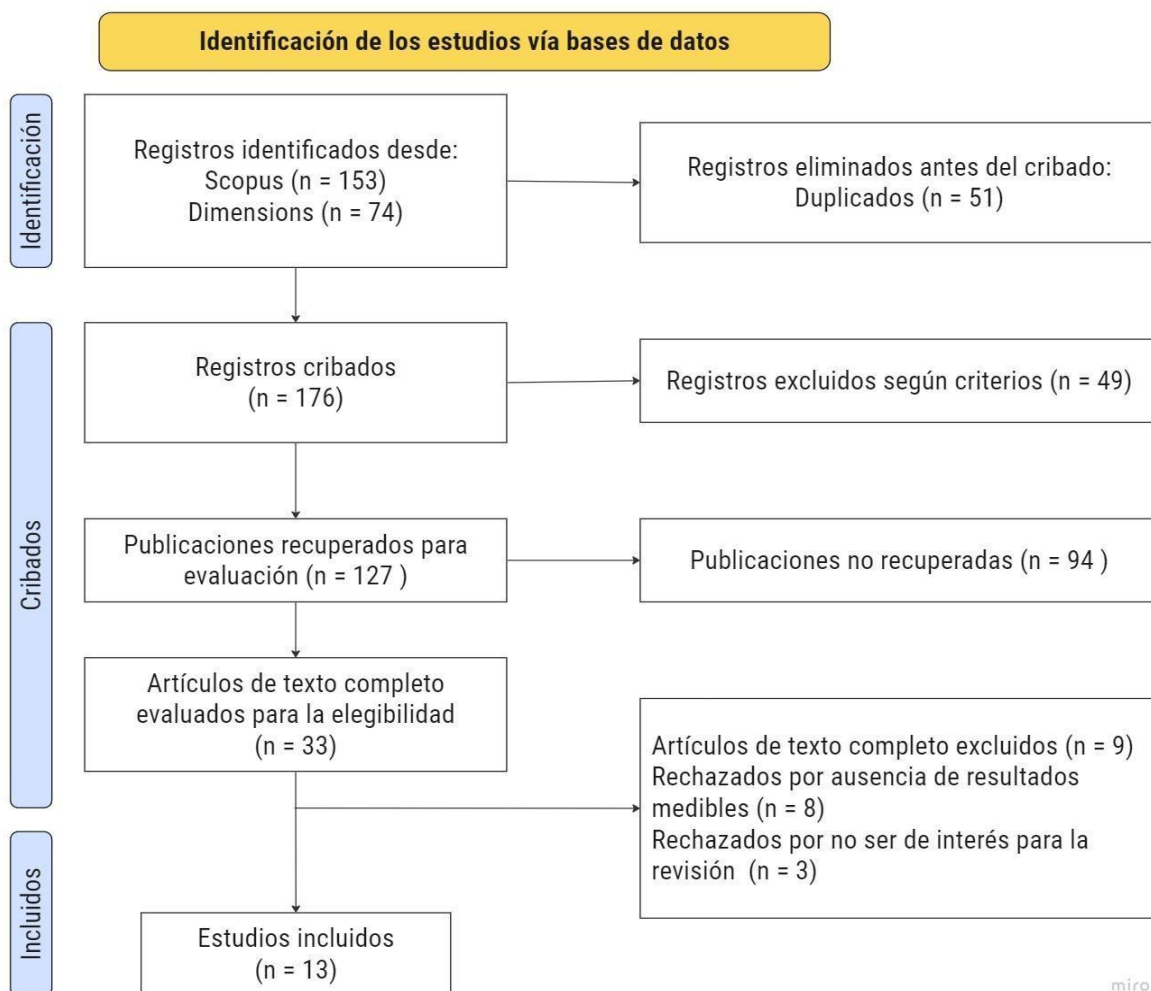


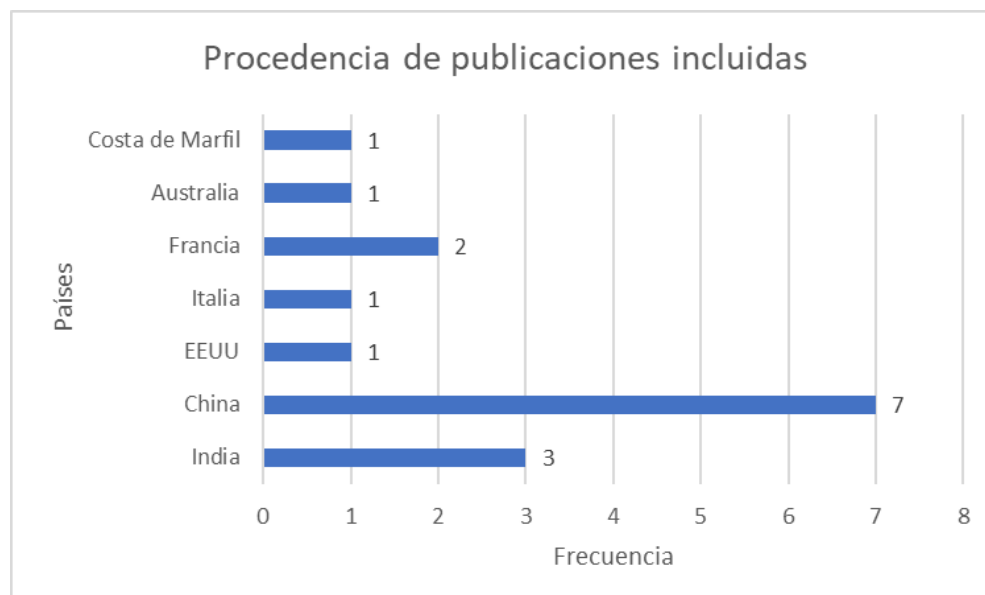
Figura 1. Diagrama de flujo de selección de estudios

Del total de artículos incluidos, 12 provienen de la base de datos Scopus y 1 de Dimensions. Los artículos se caracterizan por ser publicados desde el año 2021, siendo ese mismo el que presenta más publicaciones sobre el tema (38.46%). Mientras que en los años 2022 y 2023 se encontraron 4 artículos por año, representando cada uno el 30.77% del total, como se muestra en la Figura 2.



**Figura 2.** Publicaciones incluidas por año

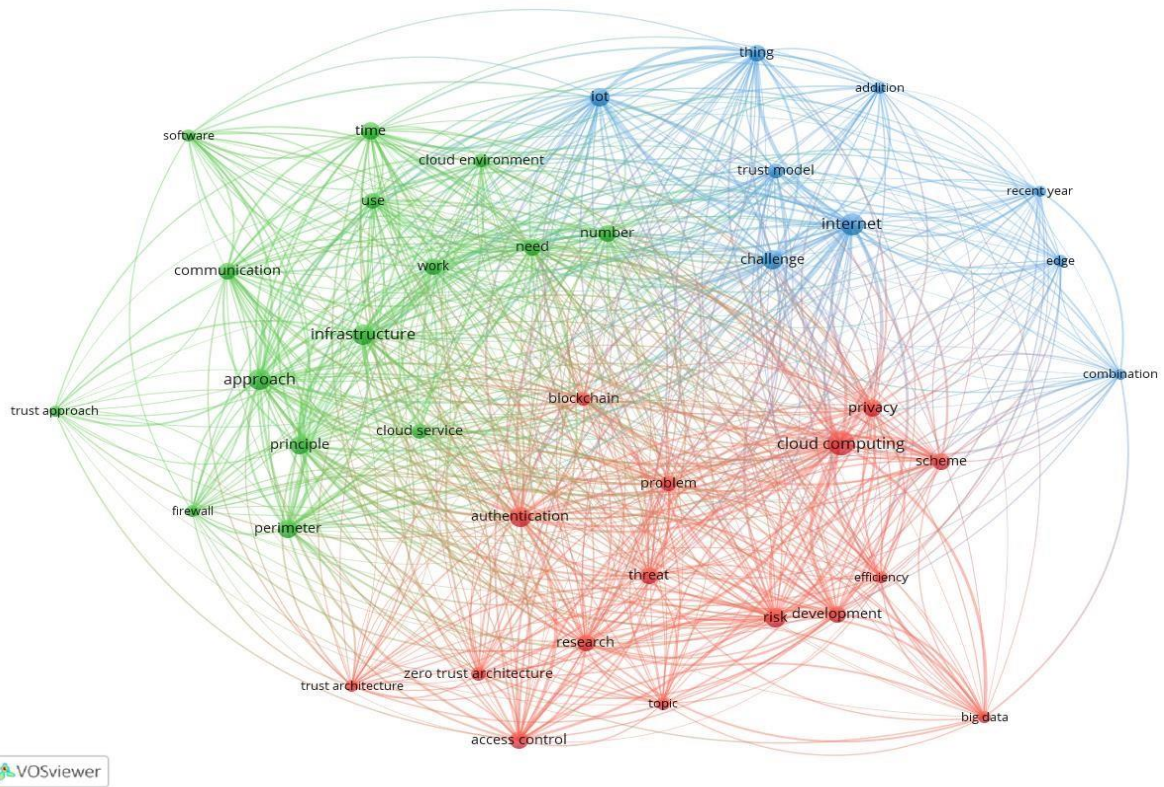
Respecto al origen de las publicaciones como se detalla en la Figura 3, China es el país con mayor número de investigaciones, seguido por India. Además, se identificaron dos artículos de colaboración internacional: en el artículo de Rajasoundaran et al. (2021) participaron investigadores de India, China y Estados Unidos, mientras que en el artículo de N'goran et al. (2022) colaboraron investigadores de Costa de Marfil y Francia.



**Figura 3.** Procedencia de publicaciones incluidas

Sobre los artículos identificados, se elaboró el mapa de co-ocurrencia utilizando el software VOSViewer (Figura 4), el cual permite visualizar los términos más relevantes y sus relaciones entre publicaciones. En este mapa, se destacan tres grupos resaltados en rojo, verde y azul, en los que los términos con mayor ocurrencia son "cloud computing" (72), "approach" (52) e "internet" (54), respectivamente.

Sin embargo, el enfoque principal del estudio es la "zero trust architecture", la cual está vinculada a 33 términos de un total de 40 ítems mostrados en el mapa. En relación a este, los términos con mayor fuerza de enlace son "risk" y "perimeter", ambos con una fuerza de enlace de 8 puntos, seguidos de "challenge" (7) y "cloud computing" (5).



**Figura 4.** Mapa de co-ocurrencia de términos

Adicionalmente, en la Figura 5 se presenta una nube de palabras clave utilizadas en las publicaciones incluidas, donde prevalece el término "zero trust" con 9 apariciones, seguido por "network" con 4 apariciones y "network security" con 3 apariciones



**Figura 5.** Palabras claves de publicaciones incluidas

De las 13 publicaciones incluidas, la Tabla 2 expone los siguientes hallazgos sobre las herramientas, procedimientos y métodos de evaluación.

Autores	Título	Herramientas o procedimientos	Métodos de evaluación
Ali et al. (2022)	A Maturity Framework for Zero-Trust Security in Multiaccess Edge Computing	Una aplicación de autenticación continúa basada en el algoritmo criptográfico PRESENT para equipos registrados colocados en servidores multiacceso del borde entre la red CC y la nube. Mediante ello, se mejora la tasa de éxito de autenticación, así como reducción del tiempo de autenticación a comparación del CC tradicional.	Formulación de marco de trabajo para analizar la madurez de la red actual de CC, en la que se presentan medidas de protección para una seguridad mínima viable basado en los pilares: dispositivos, usuarios, sesiones, aplicaciones, datos e infraestructura.
Chen et al. (2021)	A Security Awareness and Protection System for 5G Smart Healthcare Based on Zero-Trust Architecture	El modelo se enfoca en los sujetos, objetos, entornos y comportamientos implicados en un sistema de salud 5G inteligente, para proteger nube, borde y nodos. Se utilizan módulos de análisis del comportamiento de acceso de usuario, de evaluación de riesgo de equipos y de evaluación de la confianza, se reforzó la seguridad de las regiones de datos, computadoras, datos en la nube y dispositivos de Internet de las cosas en una red CC.	Uso de casos de prueba que verifica si el sistema responde con código de estado o de error para los módulos de evaluación de riesgo de equipos, evaluación de confianza y decisión de riesgo
Feng et al. (2023)	Blockchain enabled zero trust based authentication scheme for railway communication networks	La aplicación de <i>blockchain</i> permite dentro de una red CC un sistema descentralizado de acceso para almacenar credenciales mediante el árbol <i>Marke</i> y una autenticación bilateral en la que se registran usuarios, dispositivos y servicios en la nube implicados en una red de ferrocarril que converge redes públicas, privadas y de la nube. Asimismo, se usa un mecanismo de evaluaciones en tiempo real e histórico para mayor estabilidad del sistema.	Medición de eficiencia de la autenticación, de la actualización de información en el árbol y de la certificación en servicios. Así mismo, se evaluó la estabilidad cuando existe una caída repentina de rendimiento en la arquitectura actual.
Ferretti et al. (2021)	Survivable zero trust for cloud computing environments	El planteamiento de un modelo CC en la nube más agresivo que incluye para su protección a las bases de datos de políticas y accesos por separado y replicados, inicio de sesión único, certificados, control de accesos y proxy de accesos. Se realiza para proteger de cambios anómalos en las políticas de seguridad o toma de control de algún componente de seguridad por un atacante.	Análisis de seguridad a toda la arquitectura, donde se mide su supervivencia a ser comprometido mediante una tabla de costos de ataques, que presenta secuencias de ataque, componentes atacados y el costo de ataque.
Li et al. (2023)	A zero trust and blockchain-based defense model for smart electric	El esquema de seguridad para plataformas en la nube de las estaciones de carga para vehículos eléctricos hace uso de blockchain, CC y algoritmos criptográficos para una protección integral de los sistemas.	Uso del código Guobiao (estándares nacionales de China equivalentes a ISO) para el análisis de la seguridad de la información del proyecto

	vehicle chargers	Esta protección engloba la gestión de acceso, evaluaciones dinámicas de confianza y encriptación de las comunicaciones para asegurar la confidencialidad, integridad y disponibilidad del sistema en una estación de carga en China.	en relación a Blockchain y los algoritmos criptográficos (para servidores financieros y módulos de seguridad).
S. Liu et al. (2022)	Exploiting LSB Self-quantization for Plaintext-related Image Encryption in the Zero-trust Cloud	El algoritmo de autoaprendizaje basado en la incrustación de los dos bits menos significativos es una protección enfocada al ciclo de vida de datos en formato de imagen para almacenamiento de datos <i>cloud</i> . Se resuelven problemas relacionados a errores de desencriptación por ataques de ruido y sobrecargas de claves y llaves secretas que no se adaptan a las redes CC por ser ineficientes.	Análisis de ataque criptográficos, ruido y recorte de imágenes para conocer la efectividad del algoritmo empleado.
Z. Liu et al. (2022)	Data-Driven Zero Trust Key Algorithm	El esquema de seguridad CC basado en el análisis de datos de los comportamientos de los usuarios de la nube realiza una mejor detección de comportamientos anómalos y fue más preciso en determinar la confianza de los usuarios cuando acceden a información ubicada en la nube. Esta precisión reduce falsos positivos y sirve como un recurso adicional a las medidas de CC.	Análisis de valores de confianza en la nube y de agregación de eventos de seguridad permite conocer la precisión en la detección de comportamientos maliciosos dentro de una red CC en la nube.
Mandal et al. (2021)	Cloud-Based Zero Trust Access Control Policy: An Approach to Support Work-From-Home Driven by COVID-19 Pandemic	Una política de control de acceso basado en la predicción y detección de tráfico falsificado mediante MAC para proteger los recursos en la nube de una empresa, accedidos desde inicialmente desde la red propia de un usuario en su hogar. Se aplica una red CC para los recursos en la nube, un algoritmo basado en puntuaciones y umbrales para detectar comportamientos de los usuarios, analizando su IP, MAC y puerto. Este algoritmo es preciso y presenta una menor tasa de falsos positivos de accesos anómalos.	Uso de la curva característica de operación del receptor para comparar la tasa de identificación de verdaderos positivos (suplantación) y la tasa de falsos positivos. Este método de evaluación es usado en otros algoritmos con la misma finalidad de detectar tráfico malicioso.
Miller et al. (2021)	Securing Workflows Using Microservices and Metagraphs	El despliegue de flujos de trabajo que comparten información se puede asegurar mediante la verificación de políticas de seguridad de datos y una arquitectura de microservicios bajo principios de CC en la nube. Esto permite encontrar y evitar fugas de información en entornos en la nube. Finalmente, el uso de metagrafos es importante para manejar una gran cantidad de políticas de seguridad, y, por lo tanto, flujos de trabajo eficientes.	Evaluación de tiempo de ejecución y carga de CPU cuando se crean flujos de trabajo asegurados por las políticas de seguridad.



N'goran et al. (2022)	Trust Assessment Model Based on a Zero Trust Strategy in a Community Cloud Environment	El modelo de gestión de confianza de nubes comunitarias está compuesto de un gestor de recursos, de transacciones y de actualizaciones de confianza y de reputación tanto para los repositorios y los usuarios que conforman cada organización de la nube. Mediante el uso de valores de confianza recomendados por otros usuarios u organizaciones y el algoritmo <i>SeComTrust</i> planteado, se reconoce proveedores maliciosos según su comportamiento.	Análisis de funcionalidad y escalabilidad de detección de proveedores maliciosos del modelo de gestión planteado a comparación de otros modelos. El modelo detecta correctamente los proveedores maliciosos incluso cuando el número de participantes incrementa.
Rajasoundaran et al. (2021)	Machine learning based deep job exploration and secure transactions in virtual private cloud systems	Los servicios de trabajo en la nube seguros basados en aprendizaje automático son una herramienta para asegurar la confidencialidad y encriptación de los trabajos de usuario y sistema para nubes virtuales privadas basados en CC haciendo uso de modelo de cola para múltiples servidores y el modelo de análisis <i>VGAN</i> .	Evaluación mediante análisis de seguridad multinivel, donde se observa la tasa de detección de ataques DDoS, fuerza bruta, falsificación de IP, hombre en el medio y phishing.
Saleem et al. (2023)	Secure information processing for multimedia forensics using zero-trust security model for large scale data analytics in SaaS cloud computing environment	Utilización del algoritmo <i>Spatial Rich Model</i> en sus diversas presentaciones bajo criterios de CC, junto al análisis de comportamiento del usuario y servicio, fueron útiles para garantizar la confianza de respuestas brindadas por servicios en la nube de almacenamiento o procesamiento de imágenes. Estas herramientas sirven como recurso forense en caso de que el servicio en la nube haya sido comprometido o se hayan subido imágenes infectadas.	Análisis de dimensionalidad de características y tiempo de extracción de características para cada presentación del modelo <i>Spatial Rich Model</i> en la red CC.
Wang et al. (2023)	Research on Medical Security System Based on Zero Trust	Planteamiento del modelo ABEAC que analiza los comportamientos del usuario para acceso de controles dinámicos en los entornos en la nube aplicados en un modelo CC para entornos médicos, garantiza mayor protección de la información de los pacientes y los equipos médicos al considerar valores de riesgo por usuario y de forma global al tener en cuenta el número de comportamientos anómalos en total de la red.	Comparación con el modelo TMBRE, que es usado actualmente en sistemas médicos, en el cálculo del valor de riesgo y confianza del comportamiento de un usuario y múltiples usuarios en la red.

**Tabla 2.** Resultados de publicaciones incluidas

En la Tabla 3, se presentan las herramientas y procedimientos utilizados por los autores, los cuales están diseñados para fortalecer el control de accesos o mejorar el motor de políticas de las redes CC. Además, se detallan los métodos de evaluación aplicados a estas herramientas y procedimientos que, en términos generales, se pueden resumir de acuerdo a lo indicado en la tabla.

En general, se observa que el blockchain y los algoritmos estegoanalíticos son las herramientas más recurrentes, mientras que los modelos integrales de confianza predominan en los procedimientos. Por último, el método de evaluación más mencionado es el de pruebas funcionales seguido de los análisis de seguridad.

<b>Materia de estudio</b>	<b>Temas predominantes</b>	<b>Autores</b>	<b>Frecuencia</b>
Herramientas	Blockchain	(Feng et al., 2023; Li et al., 2023)	2
	Algoritmos estegoanalíticos	(S. Liu et al., 2022; Saleem et al., 2023)	2
	Algoritmos criptográficos	(Ali et al., 2022)	1
	Microservicios	(Miller et al., 2021)	1
	Machine Learning	(Rajasoundaran et al., 2021)	1
Procedimientos	Modelos integrales de confianza	(Chen et al., 2021; Ferretti et al., 2021; N'goran et al., 2022)	3
	Análisis de comportamientos	(Z. Liu et al., 2022; Wang et al., 2023)	2
	Políticas de control predictivas	(Mandal et al., 2021)	1
Métodos de evaluación	Pruebas funcionales	(Chen et al., 2021; Z. Liu et al., 2022; Mandal et al., 2021; Saleem et al., 2023)	4
	Análisis de seguridad	(Ferretti et al., 2021; S. Liu et al., 2022; Rajasoundaran et al., 2021)	3
	Comparación directa de funcionalidad con modelos existentes	(N'goran et al., 2022; Wang et al., 2023)	2
	Pruebas de rendimiento	(Feng et al., 2023; Miller et al., 2021)	2
	Análisis de madurez	(Ali et al., 2022)	1
	Estándares nacionales	(Li et al., 2023)	1

**Tabla 3.** Herramientas, procedimientos y métodos de evaluación encontrados

A modo de resumen, en el siguiente gráfico de proyección solar de dos niveles (Figura 6), se presentan las herramientas, procedimientos y métodos de evaluación utilizados para la implementación del modelo CC. Se destaca la frecuencia de aparición en los artículos mediante diferentes intensidades de color, de manera que los tonos más oscuros representan una mayor frecuencia.



**Figura 6.** Gráfico resumen de hallazgos

#### 4. Discusión

Las herramientas y procedimientos propuestos ofrecen ventajas en fortalecer los componentes del modelo CC. El uso de blockchain radica en crear centros descentralizados de identidades con menor carga para la red, de tal forma que el acceso es tolerante a fallos, y, mediante los hashes de los blockchain, evitar ataques relacionados al robo de credenciales y la suplantación de identidad (Feng et al., 2023; Li et al., 2023); mientras tanto los algoritmos estegoanalíticos cubren la necesidad de evitar filtración de datos por medio de archivos multimedia (S. Liu et al., 2022) y gestionar la confianza de en las interacciones entre usuarios y proveedores de almacenamiento de multimedia en la nube dentro de la red CC. (Saleem et al., 2023).

Entre los procedimientos, destacan los modelos integrales de confianza representan la implementación clásica del modelo CC, que abarca las políticas de acceso, manejo de identidades y análisis tanto de comportamientos como de valores de confianza y riesgo dentro de la red CC (Chen et al., 2021; Ferretti et al., 2021; N'goran et al., 2022).

Respecto a los métodos de evaluación de las herramientas y procedimientos, estos varían según el alcance de la propuesta, así como la finalidad del artículo. Li et al. (2023) hace uso de los estándares nacionales de China debido a que su proyecto está operativo para su uso cotidiano por la población, por ello tuvo que pasar por una serie de normativas que aseguran la seguridad de la información. Adicionalmente, se ha realizado comparaciones con modelos ya existentes en el mercado o sector, como en el caso del modelo *SeComTrust*, en el cuál es más efectivo en la detección de proveedores maliciosos a comparación de otros modelos como *Intertrust* o *TNASL* (N'goran et al., 2022), o un cálculo más efectivo de los valores de confianza y riesgo de usuarios del modelo *ABEAC* planteado por Wang et al. (2023) comparado con el modelo *TMBRE* usado actualmente en el sector salud.

Las pruebas de funcionalidad varía en cada publicación pero destacan el uso de casos de prueba para cada módulo planteado (Chen et al., 2021), dimensionalidad de características extraídas de recursos multimedia (Saleem et al., 2023), medición de tasa de verdaderos positivos y falsos positivos de detección de suplantación (Mandal et al., 2021) y el análisis de valores de confianza en la nube en tiempo real (Z. Liu et al., 2022). Asimismo, se hace uso de análisis de seguridad, en donde se simula distintos tipos de ataques según lo que debería proteger las herramientas o procedimientos; entre ellos, el algoritmo de S. Liu et al. (2022) que fue expuesto a ataques criptográficos y ruido, el modelo de Rajasoundaran et al. (2021) que fue valorada ante ataques de fuerza bruta o falsificación de IP y la arquitectura de Ferretti et al. (2021), en la cual se prueba la supervivencia del sistema ante distintos vectores de ataques para cada componente.

Finalmente, Ali et al. (2022) propone un marco de trabajo de madurez de la seguridad de una red CC implementada en la nube, en donde se determina cuáles son los requerimientos para una seguridad mínima viable. Este marco de trabajo puede ser usado por otros autores para analizar la condición actual de la red.

## **5. Conclusiones**

Las implementaciones del modelo de seguridad CC para entornos en la nube se ha visto fortalecida por la adopción de nuevas herramientas y procedimientos, las cuales destacan el uso de blockchain, algoritmos criptográficos o creación de nuevos modelos integrales de confianza, debido a que ofrece mejoras de rendimiento y nuevas capacidades como predicciones, análisis en tiempo real de confianza o tolerancia a fallos. Asimismo, estas tecnologías se adaptan a necesidades específicas según la forma de uso de la red en la nube en donde es utilizado, por lo que puede ser replicado por otros usuarios u organizaciones. Respecto a los métodos de evaluación de las herramientas y procedimientos, se realizan, por lo general, pruebas de funcionalidad, análisis de seguridad y comparaciones directas con modelos existentes. Sin embargo, los métodos de evaluación para cada implementación varían y no se conoce realmente qué mejora en la seguridad de información de toda la red CC se produce.

Por ello, se plantea como línea de investigación futura, que se debe crear un marco de evaluación estandarizado del modelo de seguridad CC para entornos en la nube, que permita conocer qué medidas mínimas o exigentes se deben adoptar según el nivel de criticidad de la red, de tal forma que sea posible conocer el estado actual de seguridad y el impacto de las nuevas adopciones tecnológicas en la seguridad de la red CC en la nube.

## 6. Referencias bibliográficas

- Akbar, undefined H., Zubair, undefined M., & Malik, undefined M. S. (2023). The Security Issues and challenges in Cloud Computing. *International Journal for Electronic Crime Investigation*, 7(1), Article 1. <https://doi.org/10.54692/ijeci.2023.0701125>
- Ali, B., Hijjawi, S., Campbell, L. H., Gregory, M. A., & Li, S. (2022). A Maturity Framework for Zero-Trust Security in Multiaccess Edge Computing. *Security and Communication Networks*, 2022, 1-14. <https://doi.org/10.1155/2022/3178760>
- Bajdor, P. (2022). Perception and evaluation of selected cloud computing factors in the light of conducted research among small and medium-sized enterprises. *Procedia Computer Science*, 207, 3788-3797. <https://doi.org/10.1016/j.procs.2022.09.440>
- Chen, B., Qiao, S., Zhao, J., Liu, D., Shi, X., Lyu, M., Chen, H., Lu, H., & Zhai, Y. (2021). A Security Awareness and Protection System for 5G Smart Healthcare Based on ZeroTrust Architecture. *IEEE Internet of Things Journal*, 8(13), 10248-10263. <https://doi.org/10.1109/JIOT.2020.3041042>
- Cheng, M., Qu, Y., Jiang, C., & Zhao, C. (2022). Is cloud computing the digital solution to the future of banking? *Journal of Financial Stability*, 63, 101073. <https://doi.org/10.1016/j.jfs.2022.101073>
- Divya, P., & Sherin, A. (2022). A Zero Trust Framework Security to Prevent Data Breaches and Mitigate the Cloud Network Attacks. *International Journal for Research in Applied Science and Engineering Technology*, 10, 3530-3538. <https://doi.org/10.22214/ijraset.2022.42976>
- Feng, Y., Zhong, Z., Sun, X., Wang, L., Lu, Y., & Zhu, Y. (2023). Blockchain enabled zero trust based authentication scheme for railway communication networks. *Journal of Cloud Computing*, 12(1), 62. <https://doi.org/10.1186/s13677-023-00411-z>
- Ferretti, L., Magnanini, F., Andreolini, M., & Colajanni, M. (2021). Survivable zero trust for cloud computing environments. *Computers & Security*, 110, 102419. <https://doi.org/10.1016/j.cose.2021.102419>
- George, A. S., & Sagayarajan, S. (2023). Securing Cloud Application Infrastructure: Understanding the Penetration Testing Challenges of IaaS, PaaS, and SaaS Environments. *Partners Universal International Research Journal*, 2(1), Article 1. <https://doi.org/10.5281/zenodo.7723187>
- Gill, S. H., Razzaq, M. A., Ahmad, M., Almansour, F. M., Haq, I. U., Jhanjhi, N., Alam, M. Z., & Masud, M. (2022). Security and privacy aspects of cloud computing: A smart campus case study. *Intelligent Automation and Soft Computing*, 31(1), Article 1. <https://doi.org/10.32604/IASC.2022.016597>
- Guo, R., Tafti, A., & Subramanyam, R. (2023). Internal IT modularity, firm size, and adoption of cloud computing. *Electronic Commerce Research*. <https://doi.org/10.1007/s10660023-09691-8>
- Justice, C., & Sample, C. (2022). Future Needs of the Cybersecurity Workforce. *International Conference on Cyber Warfare and Security*, 17(1), Article 1.

<https://doi.org/10.34190/iccws.17.1.33>

- Lawan, M. M., Oduoza, C., & Buckley, K. (2021). A Systematic Review of Cloud Computing Adoption by Organisations. *International Journal of Industrial and Manufacturing Systems Engineering*, 6(3), Article 3. <https://doi.org/10.11648/j.ijimse.20210603.11>
- Li, P., Ou, W., Liang, H., Han, W., Zhang, Q., & Zeng, G. (2023). A zero trust and blockchainbased defense model for smart electric vehicle chargers. *Journal of Network and Computer Applications*, 213, 103599. <https://doi.org/10.1016/j.jnca.2023.103599>
- Liu, S., Zhuang, Y., Huang, L., & Zhou, X. (2022). Exploiting LSB Self-quantization for Plaintext-related Image Encryption in the Zero-trust Cloud. *Journal of Information Security and Applications*, 66, 103138. <https://doi.org/10.1016/j.jisa.2022.103138>
- Liu, Z., Li, X., & Mu, D. (2022). Data-Driven Zero Trust Key Algorithm. *Wireless Communications and Mobile Computing*, 2022, 1-9. <https://doi.org/10.1155/2022/8659428>
- Mandal, S., Khan, D. A., & Jain, S. (2021). Cloud-Based Zero Trust Access Control Policy: An Approach to Support Work-From-Home Driven by COVID-19 Pandemic. *New Generation Computing*, 39(3-4), 599-622. <https://doi.org/10.1007/s00354-021-001306>
- Miller, L., Mérindol, P., Gallais, A., & Pelsser, C. (2021). Securing Workflows Using Microservices and Metagraphs. *Electronics*, 10(24), 3087. <https://doi.org/10.3390/electronics10243087>
- N'goran, R., Tetchueng, J.-L., Pandry, G., Kermarrec, Y., & Asseu, O. (2022). Trust Assessment Model Based on a Zero Trust Strategy in a Community Cloud Environment. *Engineering*, 14(11), 479-496. <https://doi.org/10.4236/eng.2022.1411036>
- Page, M. J., McKenzie, J. E., Bossuyt, P. M., Boutron, I., Hoffmann, T. C., Mulrow, C. D., Shamseer, L., Tetzlaff, J. M., Akl, E. A., Brennan, S. E., Chou, R., Glanville, J., Grimshaw, J. M., Hróbjartsson, A., Lalu, M. M., Li, T., Loder, E. W., Mayo-Wilson, E., McDonald, S., ... Moher, D. (2021). The PRISMA 2020 statement: An updated guideline for reporting systematic reviews. *The BMJ*, 372, undefined-undefined. <https://doi.org/10.1136/bmj.n71>
- Paul, B., & Rao, M. (2023). Zero-Trust Model for Smart Manufacturing Industry. *Applied Sciences (Switzerland)*, 13(1). Scopus. <https://doi.org/10.3390/app13010221>
- Rajasoundaran, S., Prabu, A. V., Routray, S., Kumar, S. V. N. S., Malla, P. P., Maloji, S., Mukherjee, A., & Ghosh, U. (2021). Machine learning based deep job exploration and secure transactions in virtual private cloud systems. *Computers & Security*, 109, 102379. <https://doi.org/10.1016/j.cose.2021.102379>
- Rose, S. W., Borchert, O., Mitchell, S., & Connelly, S. (2020). Zero Trust Architecture. *NIST*. <https://www.nist.gov/publications/zero-trust-architecture>
- Saleem, M., Warsi, M. R., & Islam, S. (2023). Secure information processing for multimedia forensics using zero-trust security model for large scale data analytics in SaaS cloud computing environment. *Journal of Information Security and Applications*, 72, 103389. <https://doi.org/10.1016/j.jisa.2022.103389>

- Shetty, J. P., & Rajesh, P. (2021). An overview of cloud computing in SMEs. *Journal of Global Entrepreneurship Research*, 11. <https://doi.org/10.1007/s40497-021-00273-2>
- Sohrabi, C., Franchi, T., Mathew, G., Kerwan, A., Nicola, M., Griffin, M., Agha, M., & Agha, R. (2021). PRISMA 2020 statement: What's new and the importance of reporting guidelines. *International Journal of Surgery*, 88, 105918. <https://doi.org/10.1016/j.ijssu.2021.105918>
- Teodoro, D. D. R. (2022). Cloud infrastructure architecture and the zero trust model as a cybersecurity strategy. *Revista Científica Multidisciplinar Núcleo Do Conhecimento*, 13(11), 204-232. <https://doi.org/10.32749/nucleodoconhecimento.com.br/technologyen/zero-trust-model>
- Wang, Z., Yu, X., Xue, P., Qu, Y., & Ju, L. (2023). Research on Medical Security System Based on Zero Trust. *Sensors*, 23(7), 3774. <https://doi.org/10.3390/s23073774>
- Zheng, M., Huang, R., Wang, X., & Li, X. (2023). Do firms adopting cloud computing technology exhibit higher future performance? A textual analysis approach. *International Review of Financial Analysis*, 90, 102866. <https://doi.org/10.1016/j.irfa.2023.102866>



Esta obra está bajo una licencia de Creative Commons Reconocimiento-NoComercial-CompartirIgual 2.5 México.

*Recibido 08 FEB. 2024*

*ReCIBE, Año 12 No.2, NOV. 2023*

*Aceptado 08 FEB. 2024*

## **Diagnóstico de TDAH con Machine Learning y Sensores: Un Mapeo Sistemático**

### **ADHD Diagnosis with Machine Learning and Sensors: A Mapping Study**

**Iván de Jesús Cetina Ucán**<sup>1</sup>  
**Antonio Armando Aguilera Güemez**<sup>1</sup>  
aaguilet@correo.uady.mx  
**Raúl Antonio Aguilar Vera**<sup>1</sup>  
**Juan Pablo Ucán Pech**<sup>1</sup>

<sup>1</sup> Facultad de Matemáticas, Universidad Autónoma de Yucatán, Mérida, Yucatán, México.



## **Resumen**

El trastorno de déficit de atención con hiperactividad (TDAH) es un trastorno del neurodesarrollo que tiene como características principales la hiperactividad y la falta de concentración en actividades cotidianas. Es un trastorno conocido por darse a tempranas edades y afectar a el desempeño escolar. El diagnóstico del TDAH ha sido un procedimiento complejo, al necesitar una gran recolección y análisis de datos de manera manual. Por esto mismo, se ha propuesto el uso de las herramientas de machine learning (ML) para mejorar la precisión y el tiempo del diagnóstico. Actualmente, existen una proliferación de las metodologías que reconocen diferentes tipos de datos de pacientes para el diagnóstico del TDAH. El objetivo de este estudio es revisar dentro del estado del arte las investigaciones previas sobre esta área y responder los cuestionamientos planteados. En el artículo se discutirán temas sobre los modelos de machine learning y los diferentes datos recolectados. Entre los resultados se ve como modelos de redes neuronales y Support Vector Machine (SVM) son los de mayor frecuencia. Además, que los datos de movimiento y aceleración tienen un mayor desempeño a comparación de las implementaciones por procesos neuronales a contra posición de lo esperado. Podemos concluir que los estudios relacionados a la adquisición de datos de movimiento tienen una gran promesa en su implementación y desarrollo en tiempos futuros para una mayor precisión en los resultados clínicos.

## **Palabras Clave**

TDAH; Aprendizaje Máquina; Diagnóstico; Trastorno por déficit de atención con hiperactividad; Sensores.

## **Abstract**

Attention deficit hyperactivity disorder (ADHD) is a neurodevelopmental disorder whose main characteristics are hyperactivity and lack of concentration in daily activities. It is a disorder known to occur early and affects school performance. The diagnosis of ADHD has been a complex procedure, requiring extensive manual data collection and analysis. For this reason, machine learning (ML) tools have been proposed to improve the accuracy and time of diagnosis. Currently, there is a proliferation of methodologies that recognize different types of patient data for diagnosing ADHD. This study aims to review previous research on this area within the state of the art and answer the questions raised. The article will discuss topics about machine learning models and the different data collected. The results show that neural network models and Support Vector Machine (SVM) are the most frequently used. Furthermore, movement and acceleration data have better performance compared to implementations by neural processes, contrary to what was expected. We can conclude that studies related to the acquisition of motion data have great promise in their implementation and development in future times for greater precision in clinical results.

## **Keywords**

ADHD; Machine Learning; Diagnosis; Attention deficit hyperactivity disorder; Sensors.

## 1. Introducción

El trastorno por déficit de atención con hiperactividad (TDAH) es un trastorno del neurodesarrollo (Johnson, 2005) el cual es caracterizado por patrones de falta de atención, hiperactividad e impulsividad que interfieren con el funcionamiento y el desarrollo diario (NIMH, 2021). Algunos de sus síntomas son: malas habilidades de organización, olvidos, distracción y son más propensos a recibir estímulos fuertes.

Aunque el TDAH afecta a personas de todas las edades, los síntomas suelen aparecer desde edades muy tempranas y pueden persistir hasta la adultez. Se estima que afecta alrededor del 5 al 10 por ciento de los niños (Polanczyk et al., 2014). Estos niños deben de llevar un diagnóstico junto con un especialista en el área (normalmente un neurólogo y/o psicólogo) (AAP, 2019).

El diagnóstico de TDAH implica una evaluación integral que incluye un historial médico y psiquiátrico completo, observación constante de los síntomas presentados, testimonio de padres y la aplicación de actividades recreativas con su evaluación (NIMH, 2021).

El área que rodea el diagnóstico del TDAH tiene dificultades propias las cuales obstaculizan la realización exitosa de la misma, por lo cual ocasiona batallas en la ejecución, como por ejemplo: La falta de información sobre el paciente para que el especialista pueda realizar el diagnóstico (Rader et., 2009); y la falta de herramientas prácticas funcionales cuantitativas con gran precisión para usar en el análisis del proceso de diagnóstico del TDAH para facilitar los resultados y reducir el tiempo (Keper et al., 2008).

Para la obtención de los datos del paciente se han implementado sensores, los cuales traen muchas facilidades en el reconocimiento del TDAH. Estos sensores proporcionan mediciones objetivas y continuas de varios parámetros fisiológicos y conductuales, permitiendo una evaluación más precisa de los síntomas del TDAH (Muñoz et al., 2018).

A la par del manejo de sensores se han desarrollado modelos Aprendizaje maquina (ML, por sus siglas en inglés, machine learning) los cuales pueden predecir si una persona tiene o no TDAH. Estos han demostrado una precisión y velocidad prometedora, aunque se requiere una validación y refinamiento adicionales antes de su implementación clínica generalizada (Slobodin, 2020). Con la gran cantidad de experimentos realizados se han hecho varios mapeos al estado del arte para conocer el ambiente actual del fenómeno. Aun así, algunos de estos mapeos se centran en conocer la distribución de las configuraciones de los modelos de únicamente métodos de resonancia magnética por ML (Periyasamy et al., 2021), sin comparar la precisión de varios métodos. Por lo mismo los estudios referentes a otros métodos de diagnósticos son escasos, por ende, los estudios que comparan los métodos son nulos.

En este estudio, se tiene como objetivo explorar el estado del arte actual sobre el diagnóstico del TDAH con ML y sensores, para conocer la tendencia de las metodologías llevadas a cabo en los modelos ML y sus sensores, tomando en cuenta los tipos de diagnóstico. En la sección dos, se hablará más a detalle sobre la metodología de la investigación.

## 2. Método de investigación

Para este estudio se ha decidido utilizar el método de estudio de mapeo sistemático. El cual tiene como objetivo proporcionar una visión general y clasificación de la literatura existente sobre un tema específico mediante el uso de un proceso sistemático y transparente de búsqueda, selección y análisis (Kitchenham, 2004). Para el proceso de mapeo, se decidió utilizar un enfoque utilizado en otro estudio por Aguilera & Gómez (2016), propuesto por Kitchenham (2004). Este proceso se divide en ciertas actividades, las cuales son:

- Planificación: Se definen los objetivos, las preguntas de investigación y la cadena de búsqueda. Se establecen los medios de investigación y los criterios de inclusión/exclusión.
- Ejecución: Se lleva a cabo el plan previamente definido, se ejecuta la cadena de búsqueda y se recopilan los datos obtenidos.
- Resultados: Presentamos los resultados obtenidos con representación gráfica y generamos discusiones.

## 3. Planeación

### 3.1. Preguntas de investigación (PI)

- PI1: ¿Cuál es la distribución de las contribuciones relacionadas con los tipos de datos recolectados para el diagnóstico del TDAH con ML?
- PI2: ¿Cuál es la distribución de las contribuciones relacionado con los tipos de clasificadores usados en el diagnóstico del TDAH con ML?
- PI3: ¿Cuál es el desempeño promedio de los modelos ML para el diagnóstico de TDAH para cada tipo de dato?

### 3.2. Cadena de búsqueda

Se definió una lista de palabras y términos que se utilizaron para la búsqueda de información, estos fueron extraídos de las preguntas de investigación. Las palabras son las siguientes: Trastorno por déficit de atención con hiperactividad, TDAH, Sensores, Fusión de sensores, Aprendizaje automático, Hiperactividad.

Para la búsqueda de información se ha elegido la base de datos Scopus, debido a que en ella se almacenan numerosas publicaciones referentes al área de machine learning y engloba muchas respuestas de otras bases de datos. Con esto, se define la siguiente cadena de búsqueda, en la que se utilizaron las palabras clave que se definieron previamente junto con los operadores lógicos AND y OR, que nos permitirán obtener las publicaciones de cada base de datos.

*TITLE-ABS-KEY((ADHD OR "Attention deficit hyperactivity disorder") AND ("Sensors" OR "Sensorfusion" OR "feature fusion" OR "Multiview") AND ("learning" OR "Machine learning") AND "Hyperactivity")*

### 3.3. Criterios de inclusión y Exclusión

Los resultados obtenidos al realizar la búsqueda se dividirán según los criterios de inclusión (IC) y exclusión (EX) definidos en la Tabla 1, esto nos permitirá identificar qué publicaciones se considerarán relevantes en el área de diagnósticos del TDAH.

**Tabla 1.** Criterios de inclusión y exclusión

<b>Criterio</b>	<b>Descripción</b>
IC1	Incluye aquellas publicaciones que fueron publicadas antes de junio de 2023.
IC2	Incluye aquellas publicaciones escritas en idioma inglés.
IC3	Incluye aquellas publicaciones que están relacionadas con las palabras clave seleccionadas.
EX1	Excluye todas las publicaciones que no tengan estas características.

### 4. Ejecución

Los siguientes son los resultados de la búsqueda realizada en la base de datos de Scopus. La Tabla 2 muestra el número de resultados obtenidos además de las publicaciones que pasaron los criterios de inclusión. Se puede ver que hay 20 colaboraciones destacables, teniendo un poco menos del 50 por ciento de los resultados arrojados al principio.

**Tabla 2.** Numero de diferentes publicaciones encontradas en cada fuente

<b>Fuentes</b>	<b>No. de resultados</b>	<b>No. de publicaciones después de criterios</b>
Scopus	42	20

## 5. Resultados

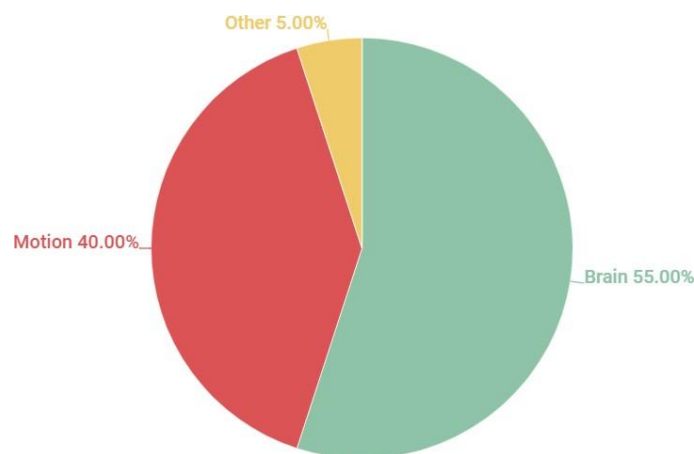
### 5.1. P11: ¿Cuál es la distribución de las contribuciones relacionadas con los tipos de datos recolectados para el diagnóstico del TDAH con ML?

La motivación viene por los tipos de diagnósticos que se pueden realizar a los pacientes, ya que en la actualidad existen muchas variaciones y cada uno tiene diferentes datos a analizar. Para poder categorizar de mejor manera los datos tomaremos como foco principal los 2 diagnósticos más comunes que recolectan datos por medio de sensores. Por la misma razón, los tipos de datos se han clasificado en las siguientes categorías: Cerebral y de Movimiento

**Cerebral:** Estos pueden ser imágenes cerebrales o resonancia magnética, principalmente con datos RGB o datos EEG y MEG. El diagnóstico de TDAH puede complementarse con enfoques neurocientíficos, como técnicas de neuro-imagen que brindan información sobre la estructura y función del cerebro (Sáenz et al., 2018).

**Movimiento:** Son datos corporales con marcas de tiempo que miden comportamiento. El diagnóstico de TDAH generalmente se basa en los síntomas conductuales observados, que incluyen falta de atención, hiperactividad e impulsividad (AAP, 2019).

Con el siguiente gráfico de la Figura 1 podemos ver que el número de contribuciones que recolectan datos cerebrales son muy superiores al resto de categorías, teniendo un porcentaje superior a 50 por ciento (con 11 contribuciones) (Hamedi et al., 2022) (Kim et al., 2021) (Sanchez et al., 2021) (Khadem et al., 2021) (Song et al., 2019) (Chen et al., 2019) (Gu et al., 2022) (Dongwei, et al., 2021) (Shengbing, et al., 2023) (Zhou, et al., 2021) (Hailong et al., 2019). Por otro lado, el segundo lugar es para los datos que se centran en el movimiento y el comportamiento (8 contribuciones) (Lee et al., 2023) (Lindhiem et al., 2022) (Muñoz et al., 2018) (Jaiswal et al., 2017) (O'Mahony et al., 2014) (Deok et al., 2023) (Xin et al., 2022). Tan solo hay una contribución que no habla de ninguna de las 3 planteadas previamente (Dolón et al., 2020).

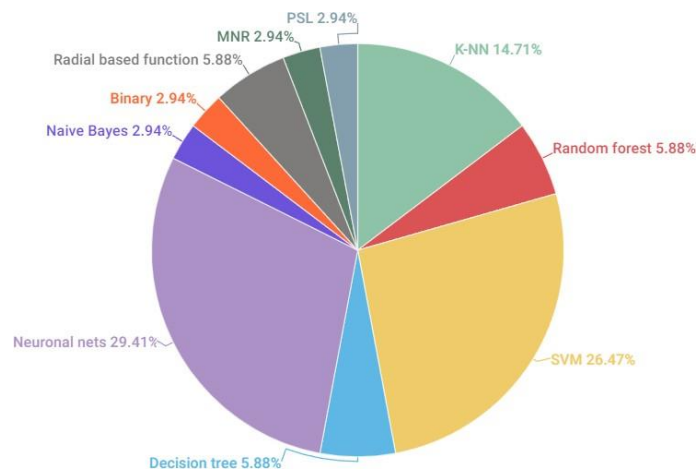


**Figura 1.** Distribución de tipos de datos detectados en cada publicación

### 5.2. PI2: ¿Cuál es la distribución de las contribuciones relacionado con los tipos de clasificadores usados en el diagnóstico del TDAH con ML?

El objetivo es tener una visión completa de los clasificadores más recurrentes en el fenómeno dentro del estado del arte. Los datos se representan en la Figura 2. Para la elaboración, se contabilizó cada clasificador usado en cada artículo, así que el número de clasificadores es mayor al de artículos.

Podemos observar que existe una tendencia a utilizar el SVM (usado por 9) (Kim et al., 2021) (Chen et al., 2019) (Song et al., 2019) (Jaiswal et al., 2017) (Faedda, et al., 2016) (O'Mahony et al., 2014) (Gu et al., 2022) (Dongwei, et al., 2021) (Zhou, et al., 2021) como algoritmo de clasificación. Otra frecuencia muy alta la tienen las redes neuronales, algunas de las destacadas son CNN (Muñoz et al., 2018) (Shengbing, et al., 2023), LSTM (Lee et al., 2023), GRU (Lee et al., 2023) y MLP (Song et al., 2019) (Deok et al., 2023) (GRU y LSTM con una sola contribución, MLP y CNN con 2), el resto no específica (Sanchez et al., 2021) (Faedda, et al., 2016) (Xin et al., 2022) (Hailong et al., 2019). Aun así, el K-NN (5 contribuciones) (Lee et al., 2023) (Hamedí et al., 2022) (Khadem et al., 2021) y Random Forest (2 contribuciones) (Lindhiem et al., 2022) (Faedda, et al., 2016) tienen frecuencias muy altas. En las lecturas, muchos clasificadores se mencionan solo una vez, por lo que es difícil comprender su relevancia para los datos, siendo



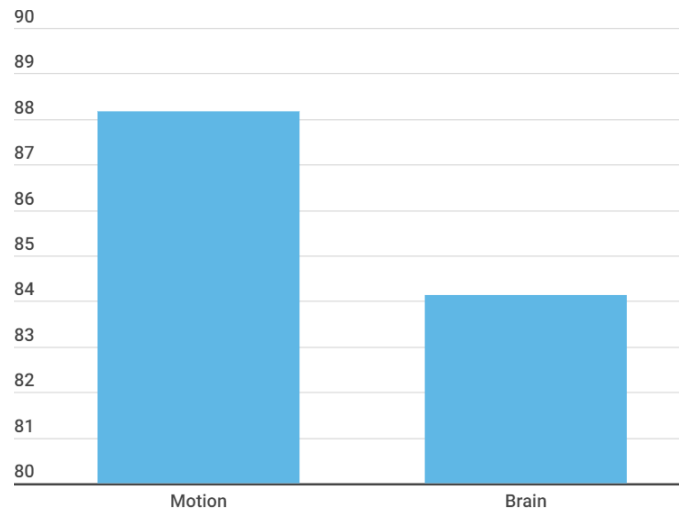
**Figura 2.** Distribución de los clasificadores usados en los modelos ML de las publicaciones

MNR (Faedda, et al., 2016), PSL (Faedda, et al., 2016), Native Bayes (Khadem et al., 2021) y un clasificador binario (Dolón et al., 2020) los mencionados. También se tienen a SVM con RBF (Khadem et al., 2021) (Hamedí et al., 2022) y Decision tree (Hamedí et al., 2022) (Khadem et al., 2021) con 2 aportaciones cada uno.

### 5.3. PI3: ¿Cuál es el desempeño promedio de los modelos ML para el diagnóstico de TDAH para cada tipo de dato?

Tomando en cuenta el tipo de dato recolectado de cada contribución se desea conocer el desempeño promedio en precisión de cada modelo y así tener un hecho concreto de cuál es el tipo de diagnóstico de TDAH usando ML más efectivo en la literatura actual.

En el caso del desempeño entre los clasificadores que usan datos de imágenes cerebrales y los que usan datos de movimiento, se representó en el gráfico de la Figura 3. En él se puede ver que la diferencia del desempeño entre ambas es bastante significativa, estando a favor de los diagnósticos por movimiento. Para la elaboración de la gráfica se promedió cada desempeño dado por cada clasificador usados en cada artículo.



**Figura 3.** Distribución del desempeño en promedio dividido por el tipo de dato que reconocen de las publicaciones

## 6. Discusión

Entre los dos tipos de diagnósticos de TDAH que se investigaron (Mentales y Movimiento), se observa en los resultados de la PI1 que los diagnósticos por datos cerebrales son más frecuentes. Este tipo de prueba es conocido por reflejar datos neuronales por el cual se podría reflejar patrones de TDAH (Siendo este un trastorno del neurodesarrollo). Mientras, el movimiento tiene una mayor exclusividad en comportamientos hiperactivos. Esto toma una mayor relevancia con la existencia del trastorno por déficit de atención (TDA) (Henderson et al. 2010) sin hiperactividad, del cual los diagnósticos de movimiento les sería difícil detectar. Dando como enfoque central en las investigaciones al TDAH como un impacto dentro de un ámbito neuronal.

Se puede apreciar en los resultados de la PI2, el SVM es el clasificador más usado dentro de los algoritmos tradicionales. Pero las redes neuronales son los clasificadores más utilizados, lo que demuestra un interés mayor por desarrollar estos modelos. Algunas de estas redes más utilizadas son: GRU, LSTM, CNN y MLP.

Asimismo, se muestra en los resultados de la PI3 la comparación del desempeño de cada tipo de diagnóstico de todos los experimentos. Los resultados sorprenden a favor de los datos por movimiento, ya que los resultados presentados en la PI1 demuestran que los datos cerebrales son más frecuentes. También se puede destacar que el promedio del desempeño de los modelos de datos de movimiento no llega al 90 por ciento de efectividad en promedio, señalando que existe una gran área de oportunidad para mejorar el desempeño. Por lo tanto, el uso de sensores de movimiento puede generar un mayor beneficio a las investigaciones del fenómeno, dando un mayor aporte a comparación de los sensores cerebrales (como EEG), ya que presentan un desempeño más alto, además de la capacidad de poder recopilar datos de manera no invasiva. En contra posición, la frecuencia de estos tipos es menor, lo cual sugiere muy poco conocimiento y confianza en su implementación en este fenómeno.

Otra cosa que destacar es que, hasta donde sabemos, no se ha producido alguna aportación del cual se realice un diagnóstico de TDAH combinando los dos tipos de recolección de datos. Lo que encadena a un área de oportunidad del cual se pueda realizar esta combinación y mejore el desempeño presentado individualmente (mayor a 90 por ciento).

Finalmente, es importante resaltar que existen otros estudios secundarios que hacen uso de datos de movimiento y cerebrales. Uno de estos se enfoca en evaluar todos los modelos del diagnóstico de TDAH con machine learning que implementan CNN (Amado et al., 2023). El segundo es el más completo, ya que se enfoca en conocer la frecuencia de los sensores y clasificadores (clásicos y redes neuronales) (Zhou, 2021) de estudios que usan los datos mencionados (Wen et al., 2022). Análogamente a los anteriores, nosotros también estudiamos los algoritmos clásicos y redes neuronales, para conocer las frecuencias de los clasificadores y los sensores. Sin embargo, a diferencia de los demás, nosotros analizamos y comparamos el desempeño que han logrado, y no solo las frecuencias de ambos tipos de datos.

## **7. Conclusión**

Uno de los puntos fuertes de esta investigación es la comparativa de los diferentes tipos de datos reconocidos por los sensores con relación al tipo de diagnóstico realizado. La diferencia de los datos del cerebro con el movimiento radica en el trato que se les da en los experimentos. Los estudios que realizan experimentos con datos cerebrales tienen un mayor desarrollo, frecuencia e importancia dentro del fenómeno. En cambio, los estudios que experimentan con datos de movimiento presentan menor motivación en su implementación, pero tienen un mayor rendimiento en promedio (al menos en los casos analizados). Esto da como resultado en una posible área de oportunidad en los diagnósticos por datos de movimiento, teniendo un mejor desempeño promedio sin presentar tanto interés en los experimentos. Estos tipos de datos pueden ser implementados por modelos con clasificadores SMV o redes neuronales, al ser estos los de mayor frecuencia.

Ha habido muchos aportes significativos para el diagnóstico de TDAH con ML, aun así, se destaca el hecho de que ningún experimento haya trabajado con los datos cerebrales y movimiento de manera combinada para tener un sistema más completo. Por tanto, se desconoce cuál podría ser el aporte al desarrollar este tipo de experimentos y los resultados obtenidos. Este podría ser el comienzo de un posible futuro proyecto de investigación.

En un futuro se espera generar datos más detallados en un artículo más específico en esta área de investigación, demostrando el gran aporte que el Machine Learning nos puede dar en el análisis de datos y el tiempo de respuesta. Se planea exponer mayores preguntas y tener una vista más completa del estado del arte.



## Referencias

- Johnson, M. H. (2005). *Developmental cognitive neuroscience: An introduction*. Wiley-Blackwell.
- National Institute of Mental Health. (2021). Attention-Deficit/Hyperactivity Disorder (ADHD). Retrieved from <https://www.nimh.nih.gov/health/topics/attention-deficit-hyperactivity-disorder-adhd/index.shtml>
- Polanczyk G. V., Willcutt E. G., et. al. (2014). ADHD prevalence estimates across three decades: An updated systematic review and meta-regression analysis. *International Journal of Epidemiology*, 43(2), 434-442.
- American Academy of Pediatrics. (2019). Clinical practice guideline for the diagnosis, evaluation, and treatment of attention-deficit/hyperactivity disorder in children and adolescents. *Pediatrics*, 144(4), e20192528.
- Rader, R., McCauley, L., and Callen, E. C. (2009). Current strategies in the diagnosis and treatment of childhood attention-deficit/hyperactivity disorder. *American Family Physician*, 79(8), 657-665.
- Kemper AR, Maslow GR, et al. (2018). Attention Deficit Hyperactivity Disorder: Diagnosis and Treatment in Children and Adolescents. Agency for Healthcare Research and Quality (US). Comparative Effectiveness Reviews, No. 203
- Muñoz-Organero, M., Powell, L., Heller, B., Harpin, V., and Parker, J. (2018). Automatic Extraction and Detection of Characteristic Movement Patterns in Children with ADHD Based on a Convolutional Neural Network (CNN) and Acceleration Images. *Sensors (Basel)*. 18(11): 3924.
- Slobodin, O., Yahav, I., Berger, I. (2020). A Machine-Based Prediction Model of ADHD Using CPT Data. *Front Hum Neurosci*. 14:560021. eCollection.
- Periyasamy, R., Vibashan, V., Varghese, G., Aleem, M., (2021). Machine Learning Techniques for the Diagnosis of Attention-Deficit/Hyperactivity Disorder from Magnetic Resonance Imaging: A Concise Review. 69(6):1518-1523.
- Kitchenham, B. (2004). *Procedures for performing systematic reviews*. Keele University, Technical Report TR/SE-0401.
- Sáenz, A., Villemonteix, T., Massat, I., (2018). Structural and functional neuroimaging in attention-deficit/hyperactivity disorder. *Dev Med Child Neurol* 61(4):399-405. doi: 10.1111/dmcn.14050.
- Carr, L., Henderson, J., Nigg, J., (2010). Cognitive Control and Attentional Selection in Adolescents with ADHD Versus ADD. Pages 726-740.
- Lee W., Lee D., et. al. (2023). Deep-Learning-Based ADHD Classification Using Children's Skeleton Data Acquired through the ADHD Screening Game, 23(1), 246
- Lindhiem O., Goel M., et. al. (2022) Objective Measurement of Hyperactivity Using Mobile Sensing and Machine Learning: Pilot Study
- Hamedi N., Khadem A., et. al.(2022) Detecting ADHD Based on Brain Functional Connectivity Using Resting-State MEG Signals, Vol.9, No.2 110-118.
- Kim S., Baek J., et. al.(2021) Machine-learning-based diagnosis of drug-naive adult patients with attention-deficit hyperactivity disorder using mismatch negativity, 11:484 ;
- Sanchez A., Villanueva C., et. al. (2021) Classification of brain signals for RPAS control in the treatment of attention deficit hyperactivity disorder. vol.97 no.2.
- N. Hamedi, A. Khadem, et. al. (2021). An Effective Connectomics Approach for Diagnosing ADHD using Eyes-open Resting-state MEG, "11th International Conference on Computer Engineering and Knowledge (ICCKE), 110-114,
- Chen H., Song Y., Li X. (2019). A deep learning framework for identifying children with ADHD using an EEG-based brain network. *Neurocomputing*, 356, 83-96.

- Chen H., Chen W., et. al. (2019). EEG characteristics of children with attention-deficit/hyperactivity disorder. *Neuroscience*, 406, 444-456.
- Jaiswal S., Valstar M. F., et. al. (2017). Automatic Detection of ADHD and ASD from Expressive Behaviour in RGBD Data. 12th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2017), 762-769,
- Faedda G. L., Ohashi K., et. al. (2016). Actigraph measures discriminate pediatric bipolar disorder from attention-deficit/hyperactivity disorder and typically developing controls. *Journal of Child Psychology and Psychiatry*, 57(6), 706-716.
- O'Mahony N., Florentino-Liano B., et. al. (2014). Objective diagnosis of ADHD using IMUs. *Medical Engineering and Physics*, 36(7), 922-926.
- Dolón-Poza, M., Berrezueta-Guzman, J., Martín-Ruiz, M., (2020). Creation of an Intelligent System to Support the Therapy Process in Children with ADHD.: TICEC 2020, CCIS 1307, pp. 36–50, 2020.
- Gu, Y., Miao, S., et. al., (2022). ADHD Children Identification with Multiview Feature Fusion of fNIRS Signals. *IEEE Sensors Journal*, vol. 22, NO. 13,
- Deok-Won Lee, Sang-hyub Lee, et. al. (2023) Development of a Multiple RGB-D Sensor System for ADHD Screening and Improvement of Classification Performance Using Feature Selection Method. *Sci.* 2023, 13, 2798.
- Dongwei Li, Xiangsheng Luo, et. al. (2022). Yan Song Information-based multivariate decoding reveals imprecise neural encoding in children with attention deficit hyperactivity disorder during visual selective attention. 44:937–947
- Shengbing Pei, Chaoqun Wang, Shuai Cao, Member, IEEE, and Zhao Lv (2023). Data Augmentation for fMRI-Based Functional Connectivity and its Application to Cross-Site ADHD Classification. *IEEE transactions on instrumentation and measurement*, vol. 72
- Xin Qin, Jindong, Wang, Yiqiang Chen, et. al. (2022). Domain Generalization for Activity Recognition via Adaptive Feature Fusion. 2157-6904
- Xiaocheng Zhou, Qingmin Lin, et. al. (2021). Multimodal MR Images-Based Diagnosis of Early Adolescent Attention-Deficit/Hyperactivity Disorder Using Multiple Kernel Learning. *Front. Neurosci.* 15:710133.
- Ming Chen, BS, Hailong Li, et. al. (2019). A Multichannel Deep Neural Network Model Analyzing Multiscale Functional Brain Connectome Data for Attention Deficit Hyperactivity Disorder Detection. 2(1):e190012
- Amado-Caballero, P., Casaseca-de-la-Higuera, P., et. al. (2023). Insight into ADHD diagnosis with deep learning on Actimetry: Quantitative interpretation of occlusion maps in age and gender subgroups. *Artificial Intelligence In Medicine* 143 (2023) 102630.
- Wen Loh, H., Ping Ooi, C., et. al. (2022) Automated detection of ADHD: Current trends and future perspective. *Computers in Biology and Medicine* 146 (2022) 105525.
- Zhi-Hua Zhou (2021) *Machine Learning*. Springer Nature, 2021

## NOTAS BIOGRÁFICAS



**Iván de Jesús Cetina Ucán** es actualmente un estudiante matriculado en la Facultad de Matemáticas, llevando la licenciatura de ingeniería de software. Se encuentra realizando su tesis relacionada a Machine Learning y TDAH. Tiene como principales gustos, el aprendizaje máquina, el reconocimiento de la actividad humana y análisis de datos.



**Antonio Armando Aguilera Güemez** es Licenciado en Ciencias de la Computación por la Facultad de Matemáticas de la Universidad Autónoma de Yucatán, Ph. D. y Maestría en Ciencias de la Computación por el Instituto Tecnológico y de Estudios Superiores de Monterrey. Actualmente es profesor de la Facultad de Matemáticas de la Universidad Autónoma de Yucatán. Es colaborador del Cuerpo Académico de Investigación de Ingeniería de Software para la Educación. Cuenta con el reconocimiento del Sistema Nacional de Investigadores como Investigador Candidato Nacional por el CONAHCYT y cuenta con el reconocimiento del Programa de Desarrollo Profesional Docente (PRODEP) desde 2022. Su labor investigadora comprende las áreas de Ingeniería de Software y Computación Educativa.



**Raúl Antonio Aguilar Vera** obtuvo el grado de Doctor por la Universidad Politécnica de Madrid, España (Mención de Doctor Europeo) y el de Máster en Ingeniería de Software por la misma Institución, posee también el grado de Maestro en Educación Superior por la Universidad Autónoma de Yucatán. Actualmente es profesor en la Facultad de Matemáticas de la Universidad Autónoma de Yucatán. Es responsable del Cuerpo académico de Ingeniería de Software para la Educación. Es miembro desde 2014 del Sistema Nacional de Investigadores del CONAHCYT (Nivel 1). Su trabajo de investigación incluye las áreas de Ingeniería de Software e Informática Educativa.



**Juan Pablo Ucán Pech** es Licenciado en Ciencias de la Computación por la Facultad de Matemáticas de la Universidad Autónoma de Yucatán, Magíster en Sistemas Computacionales por el Instituto Tecnológico de Mérida y Doctor en Sistemas Computacionales por la Dirección de Posgrado e Investigación de la Universidad del Sur, México. Actualmente es profesor de la Facultad de Matemáticas de la Universidad Autónoma de Yucatán. Es miembro del Cuerpo Académico de Investigación de Ingeniería de Software para la Educación. Cuenta con el reconocimiento del Sistema Nacional de Investigadores como Investigador Candidato Nacional por el CONAHCYT y cuenta con el reconocimiento del Programa de Desarrollo Profesional Docente (PRODEP) desde 2014. Su labor investigadora se centra en las áreas de Ingeniería del Software y Ciencias de la Computación Educativas.



*Recibido 18 JUL. 2023*

*ReCIBE, Año 12 No.2, NOV. 2023*

*Aceptado 28 JUL. 2023*

## **Plug and play (upnp): métodos de ataque y medidas de protección**

### **Plug and play (upnp): attack methods and protection measures**

*Leticia Chuquiana Casicana*

lety.ch06@gmail.com

*Galo López Sevilla*

## **Resumen**

Universal Plug and Play (UPnP), facilita la conectividad entre dispositivos pertenecientes a una misma red local y la comunicación de programas con servidores de terceros, esta tecnología viene activa por defecto en algunos routers la cual evita la configuración manual para la apertura de puertos en el mismo, ya que se realiza de forma automática. Al ser una tecnología para la comunicación y conectividad ya sea entre dispositivo o programas en internet, puede llegar a ser inseguro, puesto que terceras personas mal intencionadas pueden acceder a la red aprovechando que este protocolo esta activo, poniendo en riesgo tanto la información personal como los dispositivos conectados a la misma red. Por ende, este estudio es importante ya que, ayuda a identificar los métodos de ataque que se realiza al protocolo UPnP y definir las medidas de protección que se pueden aplicar al usar esta tecnología. El objetivo de esta investigación es identificar los métodos de ataque y medidas de protección de UPnP, la cual tendrá una investigación bibliográfica con enfoque cualitativo, además se aplica la metodología RSL para la revisión sistemática de la literatura y desarrollo del artículo. Al finalizar este trabajo de investigación se espera contar con un documento de ayuda para mitigar los ataques informáticos a usuarios que tienen activo el protocolo UPnP.

**Palabras clave: UPnP, conexión, protocolos, RSL.**

## **Abstract**

Universal Plug and Play (UPnP), facilitates connectivity between devices belonging to the same local network and communication of programs with third-party servers, this technology is activated by default in some routers which avoids the manual configuration for opening ports in the router, as this is done automatically. As it is a technology for communication and connectivity either between devices or programs on the Internet, can become insecure, since illintentioned people can access the network taking advantage of the fact that this protocol is active, putting both personal information and devices connected to the same network at risk. Therefore, this study is important because, helps to identify the attack methods used to attack the UPnP protocol and define the protection measures that can be applied when using this technology. The objective of this research is to identify the methods of attacking and UPnP protection measures, which will have bibliographic research with a qualitative approach, in addition, the SLR methodology is applied for the systematic review of the literature and article development. At the end of this research work, it is expected to have a document that will be of help to mitigate cyber-attacks on users who have UPnP protocol active. **Keywords: UPnP, connection, protocols, RSL.**

## 1.- INTRODUCCIÓN

A nivel mundial los ataques cibernéticos han crecido a raíz de la pandemia y el teletrabajo, lo cual ha causado una ola de ataques a gran escala, ya sea a nivel empresarial como personal. Los ciberdelincuentes aprovechan cualquier vulnerabilidad para ingresar en los sistemas de las víctimas y realizar operaciones ilícitas. Existen varias formas de atacar un sistema o red, mediante el uso de herramientas, dispositivos y protocolos; como es el caso del protocolo Universal Plug and Play (UPnP).

UPnP es un conjunto de protocolos que permite la comunicación entre dispositivos de una misma red y servidores de terceros mediante la apertura automática de puertos, lo cual facilita la conexión (Maestre, 2015).

Sin duda es una tecnología muy buena ya que, no requiere de configuración manual para la apertura de puertos en los routers del hogar, por otro lado, también es un problema así lo mencionan Rapid7 (2013), en una investigación realizada en el 2012, en donde “se identificaron más de 80 millones de direcciones IP únicas que respondieron a las solicitudes de descubrimiento UPnP de Internet. En algún lugar entre 40 y 50 millones de direcciones IP son vulnerables”.

En este contexto, el problema radica en los inconvenientes que genera la conexión directa de los dispositivos a una red mediante el protocolo UPnP, ya que, este protocolo facilita la conexión y reconocimiento de dispositivos en una misma red local, de igual manera asigna una IP generada por el Protocolo de Configuración Dinámica de Host (DHCP) a cada dispositivo (Maestre, 2015), este proceso se lleva a cabo sin notificar al usuario. Frente a esta problemática, dentro de la investigación, se pretende dar respuesta a las siguientes preguntas: ¿Cuáles son los aspectos relacionados a la seguridad en UPnP?, ¿Cuáles son los métodos de ataque utilizados en UPnP?, ¿Cuáles son las medidas de protección recomendables para aplicar en UPnP?

De tal manera que, esta investigación es importante pues, se detalla los métodos de ataque a UPnP y sobre todo se especifica medidas de protección. Una vez definido la problemática e importancia del proyecto de investigación propuesto, el objetivo es **Identificar los métodos de ataque y medidas de protección de Plug and Play (UPnP)**, en donde se verifica lo siguiente: Determinación del estado del arte sobre las seguridades de UPnP, revisión de las metodologías para el análisis de las vulnerabilidades y métodos de ataque a UPnP, selección de una metodología para el análisis de las medidas de protección aplicables a UPnP. Para llevar a cabo este proyecto de investigación se utiliza una metodología, que consiste en procedimientos y técnicas para la indagación de un problema que requiere solución (Cohen & Gómez, 2019), es así que, se utiliza una investigación bibliográfica con enfoque cualitativo y se aplica la metodología RSL para la revisión sistemática de la literatura.

## 2.- ESTADO DEL ARTE

### 2.1. ¿Qué es un Ciberataque?

En la actualidad, los ciberataques han venido ganando terreno, gracias a los avances tecnológicos que no solo han beneficiado a la población, al contrario, también han desarrollado herramientas para realizar estos delitos informáticos (Izaguirre Olmedo & León Gavilánez, 2018), un ciberataque es un ataque malicioso en donde un ciberdelincuente aprovecha cualquier vulnerabilidad de un sistema informático para introducir virus con la intención de robar información o afectar el funcionamiento de las redes de comunicación y sistemas informáticos. Así mismo, Pons (2017) indica que, para un ciberdelincuente realizar estos delitos informáticos tiene ciertas ventajas, pues no se expone de manera física a la víctima, basta con equipos informáticos, conexión a internet y conocimientos técnicos en informática pueden ejecutar los ataques de manera anónima desde cualquier parte del mundo a cualquier persona u organización.

## 2.2. Ciberataques a Nivel Mundial

A nivel mundial los ciberataques aumentaron un 29% en el primer trimestre del año 2021, puesto que, los ciberdelincuentes aprovecharon el teletrabajo o trabajo remoto, que fueron restricciones y medidas de seguridad aplicadas por las empresas e instituciones educativas, debido a la pandemia (Check Point, 2021).

Por otra parte, Morales (2022) menciona que, el aumento de los ciberataques también se debe a los avances y desarrollo tecnológicos, que han abierto un camino hacia el mundo digital también conocido como ciberespacio, en donde un sin número de personas intercambian información entre sí por medio de la red, sin embargo, estos avances han sido utilizados por ciberdelincuentes que realizan delitos informáticos con el objetivo de sustraer información personal, de empresas u organizaciones. Los ataques más frecuentes en el año 2022 son: ciberdelincuencia, ciberacoso, ciber espionaje, ciber amenazas entre otros.

## 2.3. Ciberataques en Ecuador

Según, Alvarado (2020), en Ecuador el 43% de personas tienen acceso a internet, pero la mayoría desconoce los peligros y amenazas de su uso, siendo víctimas de ciberataques; Coello (2021) manifiesta que, en el año 2021 Ecuador se ubicó en el sexto puesto según el análisis de Ciberseguridad a nivel de Latinoamérica “el ministerio de Telecomunicaciones y de la Sociedad de Información (MINTEL), Ecuador tiene carencia de implantaciones políticas y estrategias nacionales en el ámbito de la ciberseguridad que podrían mejorarse y fortalecerse.”

## 2.4. ¿Qué es Plug and Play (UPnP)?

UPnP es un protocolo de comunicación entre dispositivos conectados en una misma red, se caracteriza por la apertura automática de diferentes puertos del router (Maestre, 2015); a cada dispositivo se asigna una IP de manera dinámica por DHCP, “UPnP permite que se abran y cierren puertos de forma automática por orden o solicitud de un dispositivo o programa” (Fundación Proydesa, 2022).

## 2.5. Peligros de mantener los puertos abiertos

Novoa & Salazar (2022), Miranda (2019), el router es el encargado de transmitir información entre los dispositivos y programas conectados a una red, por ende permitir la apertura automática resulta beneficiosa para evitar problemas o molestias al usuario, sin embargo, desde el punto de vista de la seguridad informática, es peligroso, ya que la mayoría de atacantes aprovechan esta vulnerabilidad para ingresar a los sistemas informáticos y cometer delitos.

Entre los peligros que conlleva mantener los puertos abiertos son: modificación de la configuración del Sistema de Nombres de Dominio (DNS), *man in the middle* (ataque de intermediario), ataque de denegación de servicio distribuido (DDoS), creación de redes falsas, entre otras (Miranda, 2019).

## 2.6. Métodos de ataque a (UPnP)

Los métodos de ataque más comunes en este tipo de protocolo son los ataques DoS y DDoS, ya que tienen el objetivo de secuestrar máquinas y usarlas como *bots* para atacar un servidor, enviando una serie de peticiones con el fin de colapsarlo (Microsoft, 2022).

Para ampliar con mayor detalle este tema, se especifica en la fase tres de la Metodología RSL.

### **3.- METODOLOGÍA DE LA INVESTIGACIÓN**

El enfoque de la investigación para este estudio fue el enfoque cualitativo, el cual se orienta a la recolección de datos con la finalidad de analizar y explicar mediante métodos y técnicas de investigación (Sánchez, 2019), así mismo, Iño (2018) define el enfoque cualitativo como: “llevar a cabo la generación de información, descripción, procesamiento, análisis e interpretación y redacción del trabajo”. Además, el tipo de investigación que se emplea en ese estudio es la investigación bibliográfica, para Lafuente & Martín (2017) corresponde a la búsqueda bibliográfica o documental de distintas fuentes de información, es la revisión de investigaciones ya existentes y comprobadas por medio de hipótesis, experimentos, entre otros. Esta investigación ayuda a este estudio a buscar información de distintos autores sobre UPnP. Por otra parte, para el desarrollo de esta investigación se utiliza la metodología RSL propuesta por Kitchenham, B. & Charters, S. (2007), consta de tres etapas: planeación, ejecución y reporte de resultados. Esta metodología sigue un proceso de identificación, evaluación y combinación de estudios primarios para realizar estudios secundarios (Carrizo & Moller, 2018).

#### **3.1. Planeación**

Es la primera etapa de la metodología RSL en donde se desarrolla la identificación de la necesidad, formulación de las preguntas de investigación, selección de la fuente, definición de la cadena de búsqueda, criterios de inclusión y exclusión.

##### ***Identificación de la necesidad***

El protocolo UPnP, es una tecnología que facilita la comunicación entre los dispositivos conectados a una misma red, de igual manera facilita la conexión con programas alojados en servidores de terceros, esto se da por la apertura automática de los puertos en el router; de esta manera el usuario no tiene la necesidad de activarlos manualmente para conectar un dispositivo o programa en la red (Maestre, 2015). Por otra parte, este protocolo no tiene la suficiente seguridad y tampoco identifica con exactitud que dispositivos o programas solicita la apertura de un puerto, pueden ser seguros o no, simplemente ante cualquier solicitud de apertura, el router permite la conexión mediante el protocolo UPnP. Es así que surge la necesidad de conocer los métodos de ataques utilizados en UPnP para plantear medidas de protección, mitigar posibles ataques y mantener una conexión segura en la red.

##### ***Formulación de las preguntas de investigación***

Para este estudio se definen las siguientes preguntas:

- P1. ¿Cuáles son los aspectos relacionados a la seguridad en UPnP?
- P2. ¿Cuáles son los métodos de ataque utilizados en UPnP?
- P3. ¿Cuáles son las medidas de protección recomendables para aplicar en UPnP?

##### ***Selección de la fuente***

Como fuente de búsqueda se seleccionó a IEEE, por ser una base de datos de investigación académica que proporciona acceso al texto completo de artículos y trabajos sobre Ciencias de la Computación, Ingeniería Eléctrica y Electrónica (IEEE Xplore, 2022). Además, es una de las bases de datos en donde se encontró una cantidad significativa de artículos relacionados al tema de estudio.

##### ***Definición de la cadena de búsqueda***

La cadena de búsqueda se definió de la siguiente manera:

TITLE-ABS-KEY (universal AND plug AND play) AND PUBYEAR > 2010

##### ***Criterios de inclusión y exclusión***

Los artículos investigados deben cumplir con los criterios de inclusión: Artículos relacionados con el tema Universal Plug and Play, deben estar escritos en inglés o español, documentos de revistas, conferencias o libros indexados en IEEE; se toma en cuenta artículos desde el año 2010 debido a que se encuentra información que aporta significativamente al tema de investigación. Por otra parte, se ignoran los estudios realizados antes del año 2010 y que no corresponda a fuentes de artículos, revistas o libros.



### 3.2. Ejecución

En esta etapa se realiza la selección de estudios primarios y extracción de datos encontrados:

#### Selección de estudios primarios

Se realiza la selección de estudios primarios de la siguiente manera: Se ejecuta la cadena de búsqueda definida en IEEE, en donde se obtuvo 120 resultados. Seguido se selecciona los estudios relacionados al tema, palabras clave y preguntas de investigación en donde se elige 52 estudios. Además, se aplica los criterios de inclusión y exclusión, en el cual se excluye 36 estudios ya que su contenido no era suficiente para responder las preguntas de investigación y los años de publicación eran antes del año 2010. Finalmente se selecciona 16 estudios para esta investigación (ver la Figura 1).

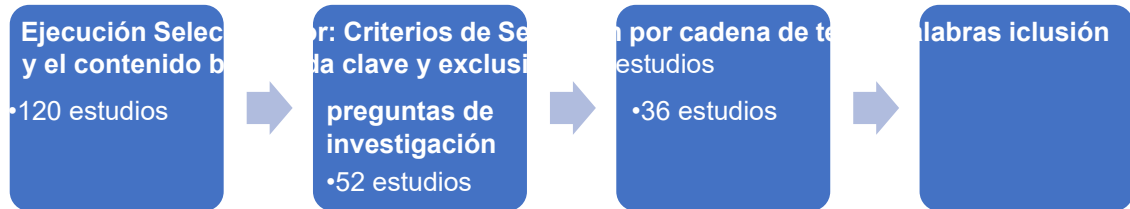


Figura 1. Selección de estudios primarios, elaboración propia.

#### Extracción de datos encontrados

Una vez seleccionado los estudios de mayor relevancia se procede a la extracción de la información, en donde se utiliza una matriz en Excel que contiene año, autor, título del artículo, URL y posible respuesta a las preguntas planteadas: P1, P2, P3, esto con el fin de llevar una mayor organización de los artículos de acuerdo al año e información obtenida (ver la Figura 2).

A	B	C	D	E
Año	Autor	Título del Artículo	URL	Posible respuestas
2022	Pravin Nair; Kunal N. Chaudhury	Regularización Plug-and-Play usando solucionadores lineales	<a href="https://ieeexplore.ieee.org/document/9913822">https://ieeexplore.ieee.org/document/9913822</a>	Convergencia objetiva y de punto fijo
2021	Aristotelis M Tsimitsios; Vassilis C Nikolaidis	PROTECCIÓN PLUG-AND-PLAY: UNA SOLUCIÓN ANTE LA COMPLEJIDAD DE DISEÑO DE ESQUEMAS DE PROTECCIÓN	<a href="https://ieeexplore.ieee.org/document/9692104">https://ieeexplore.ieee.org/document/9692104</a>	Para hacer frente a la complejidad de
2021	Xi Fang; Guoqi Sun; Wenjing Li; Liuwang Wang	Diseño de estrategia de identificación automática para módulos funcionales plug-and-play del controlador de	<a href="https://ieeexplore.ieee.org/document/9635173">https://ieeexplore.ieee.org/document/9635173</a>	La estrategia de identificación autom
2021	Eudes Rigoberto Apaza Estaño; Luis Enrique Baca Wiesse; Christian Augusto Romero Goyzueta	Diseño de firewall empresarial Plug and Play IPv6 basado en Iptables, Nettop y Linux	<a href="https://ieeexplore.ieee.org/document/9591064">https://ieeexplore.ieee.org/document/9591064</a>	

Figura 2: Extracción de datos encontrados, elaboración propia.

### 3.3. Reporte de resultados

En esta etapa se presenta los resultados obtenidos de la RSL.

#### P1. ¿Cómo funciona UPnP?

UPnP está diseñado para la conexión de dispositivos domésticos y de pequeñas empresas, se usa habitualmente para la conexión de Dispositivos del Internet de las cosas (IoT), Streaming de vídeo, juegos en línea, para la comunicación rápida entre dispositivos (Kayas, Hossain, Payton, & Islam, 2020).

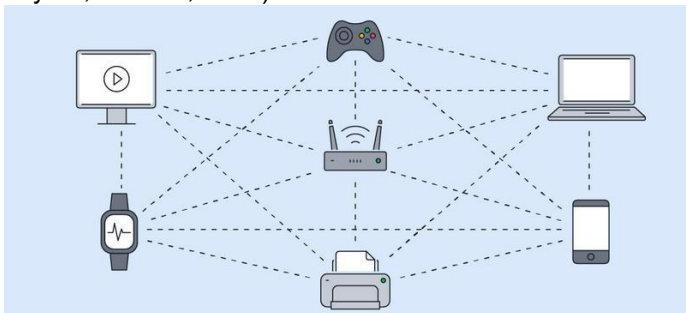
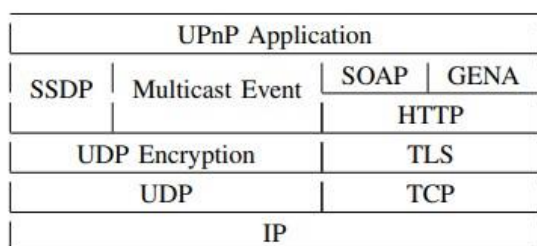


Figura 3. Conexión entre dispositivos UPnP, (Ghimiray, 2022)

En la Figura 3, muestra como es la conexión de los dispositivos UPnP conectados a una misma red, pueden estar conectados dos o más dispositivos entre sí. Además, los dispositivos UPnP se dividen en dos categorías: Dispositivos de Servicio (SD) y Puntos de Control (CP) en donde, SD es el servidor encargado de prestar el servicio; mientras que, CP es un cliente que utiliza los servicios proporcionados por SD (Kayas, Hossain, Payton, & Islam, 2020). Tanto CP como SD, pueden ser dispositivos o programas.

Por otra parte, el funcionamiento de UPnP pasa desapercibido para el usuario, ya que al conectar un dispositivo a la red este se une sin necesidad de alguna configuración manual, todo el proceso de conexión es interno entre los dispositivos, en donde intervienen protocolos como: Protocolo Simple de Descubrimiento de Servicios (SSDP), Protocolo de Transporte de Hipertexto (HTTP), Protocolo de Datagramas de Usuario (UDP), Protocolo de Control de Transmisión (TCP), Protocolo Simple de Acceso a Objetos (SOAP), Seguridad de la Capa de Transporte (TLS), Arquitectura General de Notificación de Eventos (GENA), Protocolo de Internet (IP) y Eventos *Multicast* (Pehkonen & Koivisto, 2010) (Mussi, Barreto, & Cugnasca, 2016).



**Figura 4.** Elementos usados en UPnP, (Pehkonen & Koivisto, 2010)

Como se muestra en la Figura 4, son varios los elementos que intervienen para establecer la conexión entre los dispositivos y una red, cada uno de los protocolos aportan funcionalidades que es aprovechada por UPnP, además, para su correcto funcionamiento UPnP utiliza los puertos 7676 y 1900 del protocolo TCP (Nava-Lopez & otros, 2019). Por otra parte, el proceso que realiza para conectar un dispositivo a la red consta de 6 fases (Grimmett & O'Neill, 2012), (Guo & Li, 2013); en la siguiente tabla se explica el proceso de conexión que utiliza UPnP, en donde hace uso de los elementos mencionados en la Figura 2:

<b>Dirección</b>	Un nuevo dispositivo se une a la red y se le asigna una dirección IP mediante DHCP.
<b>Descubrimiento</b>	El dispositivo anuncia su presencia enviando mensajes mediante protocolo UDP, para la búsqueda y publicidad de los dispositivos interviene el protocolo SSDP.
<b>Descripción</b>	El dispositivo muestra sus servicios a la red.
<b>Control</b>	Otros dispositivos de red comienzan a interactuar con el nuevo dispositivo por medio del protocolo SOAP.
<b>Eventos</b>	Por medio de los eventos, los dispositivos anuncian cambios en sus variables de estado al nuevo dispositivo e informan a este cuando necesitan sus servicios. Los mensajes de notificación de eventos se forman mediante GENA y se envían por HTTP sobre TCP.
<b>Presentación</b>	En algunos casos, el dispositivo contiene una interfaz de usuario a la cual se puede acceder mediante una URL y obtener información o usarla para cambiar la configuración del dispositivo.

**Tabla 1.** Etapas de conexión UPnP, basado en (Grimmett & O'Neill, 2012) y (Guo & Li, 2013)

Son buenos los beneficios y comodidad que brinda UPnP en la conexión y comunicación de dispositivos en la red; sin embargo, en el tema de seguridad según Zheng, Li & Chen (2011), UPnP no fue diseñado con suficiente seguridad, ya que según estadísticas el 20% de productos habilitados para UPnP están expuestos a amenazas externas e internas que aprovechan cualquier vulnerabilidad para atacar a los dispositivos. Nava-Lopez & otros (2019), también menciona que, un dispositivo se conecta dinámicamente a la red, se asigna una IP con la cual puede comunicarse con los demás dispositivos conectados en la misma red, pero al desconectarse no deja ningún rastro de la actividad realizada en la red, dejando agujeros de seguridad en donde personas mal intencionadas pueden aprovechar para ingresar a la red utilizando distintos métodos de ataque.

**P2. ¿Cuáles son los métodos de ataque utilizados en UPnP?**

El proceso de conexión de UPnP es automático, pero debe pasar por seis etapas en donde se pueden encontrar vulnerabilidades ya que no proporciona confiabilidad, integridad y autenticidad en los mensajes enviados por la búsqueda y publicidad de dispositivos conectados, además UPnP no detecta si las peticiones enviadas son desde servidores o dispositivos confiables, simplemente las acepta y pasa por el procesos de conexión sin ningún problema (Sales, Sales, Almeida, Perkusich, & Sales, 2013). Las etapas en donde corre el mayor riesgo son: descubrimiento, descripción, control y eventos; ya que, son las etapas en donde tienen mayor interacción los CP y SD (Kayas, Hossain, Payton, & Islam, 2020). En la Tabla 2, se describen los ataques más comunes encontrados en UPnP:

<b>Método de Ataque</b>	<b>Consecuencia</b>	<b>Autor</b>
Denegación de servicio	Un dispositivo malicioso sobrecarga una red UPnP enviando demasiadas solicitudes de descubrimiento, con el fin de que un dispositivo de servicio no se encuentre disponible para dispositivos legítimos. Como consecuencia de este ataque es el consumo excesivo de memoria, bajo rendimiento y bloqueo temporal del sistema.	(Kayas, Hossain, Payton, & Islam, 2020)
Desbordamiento de búfer	Esta vulnerabilidad sucede porque en la implementación de UPnP contiene un búfer no verificado en uno de sus componentes, el cual es el responsable de procesar los mensajes que anuncian la disponibilidad de dispositivos compatibles con UPnP en una red. Es posible desbordar el búfer mediante una inundación de mensajes desde un dispositivo malicioso, dando como resultado libre acceso al atacante para que obtenga el control del sistema, una vez, comprometido el sistema el atacante puede borrar toda evidencia después del ataque, siendo imposible rastrear al atacante.	(Hasib & Mottalib, 2010)
Consumo de energía	Los dispositivos UPnP funcionan enviando mensajes multicast periódicamente para notificar su presencia, este proceso consume energía a un nivel bajo. Si un atacante consigue obligar a los dispositivos enviar anuncios multicast con demasiada frecuencia, hará que el dispositivo consuma energía a un nivel más alto y posiblemente deje de funcionar.	(Hasib & Mottalib, 2010)

**Tabla 2.** Métodos de ataque UPnP

Los ataques descritos en la Tabla 2, son considerados como principales en UPnP, pero a raíz de estos se derivan los siguientes: suplantación de identidad de dispositivos de servicio, suplantación de puntos de control, agotamiento de recursos al implementar UPnP en redes IoT, falsificación de mensajes de publicidad, falsificación de eventos, inyección de malware (Carron, Wabersich, & Zeilinger, 2021), (Santos, et al., 2020); estos ataques son consecuencias de la poca seguridad que ofrece UPnP, ya que los atacantes se aprovechan de las vulnerabilidades como: ausencia de autenticación en anuncio y publicidad, falta de autenticación en la etapa de control, falta de verificación de integridad en la etapa de eventos, falta de verificación en la etapa de descubrimiento (Kayas, Hossain, Payton, & Islam, 2020).

**P3. ¿Cuáles son las medidas de protección recomendables para aplicar en UPnP?**

Las medidas de protección de los principales ataques a UPnP son las siguientes:

Ataque	Medida de Protección
Denegación de Servicio	Implementar un sistema de detección y prevención de intrusiones (IDS/IPS) que monitorizan las conexiones y alertan si se detecta intentos de acceso no autorizados o mal uso de protocolos. Deshabilitar los servicios UPnP en caso de presentar un fallo en el Sistema Operativo.
Desbordamiento de búfer	Este ataque se puede evitar utilizando la asignación de memoria dinámica en lugar de la asignación estática para la descripción del dispositivo, permitir la descarga de la descripción del dispositivo solo a través del puerto mayor 1024.
Consumo de energía	Instalando un software de medición de energía en el dispositivo UPnP, con el fin de monitorear la curva de consumo de energía; ya que el atacante debe ser alguien dentro de la red, es posible rastrearlo obteniendo la dirección IP de donde provienen los mensajes de búsqueda.

**Tabla 2.** Medidas de protección, basado en (Guo & Li, 2013) (Hasib & Mottalib, 2010), (Kayas, Hossain, Payton, & Islam, 2021)

Asimismo, como medidas generales de protección, se recomienda: actualizar el firmware del router, uso de contraseñas fuertes, revisar los dispositivos conectados en el router, actualizar los sistemas operativos de los dispositivos con las últimas versiones, reestablecer las configuraciones de fábrica del router, descargar e instalar software de fuentes confiables.

#### 4. CONCLUSIONES

- La funcionalidad de UPnP es compleja, ya que, no solo hace uso de otros protocolos, al nuevo dispositivo, este envía su IP asignada y otros datos a los demás dispositivos conectados para empezar a interactuar y al mismo tiempo enviar notificación de eventos cuando uno de ellos necesita los servicios de otro dispositivo.
- En cuanto a los ataques, se encontraron tres de mayor impacto, las cuales se filtran por medio de las vulnerabilidades de los protocolos con los que trabaja UPnP.
- Las medidas de protección para los principales ataques constan de la instalación de software específico para evitar dichos ataques que son comunes a nivel de seguridad informática, además se describen medidas generales de protección que dependen del factor humano; con la ausencia de dichas medidas preventivas, una red UPnP puede sufrir de amenazas críticas para la red y los dispositivos.

#### BIBLIOGRAFÍA

- Alvarado, J. E. (2020). ANÁLISIS DE ATAQUES CIBERNÉTICOS HACIA EL ECUADOR. *Revista Científica Aristas*. Obtenido de [https://revistacientificaistjba.edu.ec/images/home/documentos/Mayo\\_2020/2.pdf](https://revistacientificaistjba.edu.ec/images/home/documentos/Mayo_2020/2.pdf)
- Carrizo, D., & Moller, C. (2018). Estructuras metodológicas de revisiones sistemáticas de literatura en Ingeniería de Software: un estudio de mapeo sistemático. *Scielo*. Obtenido de Scielo: <http://dx.doi.org/10.4067/S0718-33052018000500045>
- Carron, A., Wabersich, K. P., & Zeilinger, M. N. (abril de 2021). Plug-and-Play Distributed Safety Verification for Linear Control Systems With Bounded Uncertainties. *IEEE Transactions on Control of Network Systems*. Obtenido de <https://doi.org/10.1109/TCNS.2021.3074218>
- Check Point. (2021). *Ransomware Exploits and Supply Chain Attacks Lead the Cyber Trends in the First Half of 2021*. Obtenido de <https://pages.checkpoint.com/cyber-attack-2021trends.html#:~:text=Global%20cyber%20attacks%20increased%20by,attack%20technique%20called%20Triple%20Extortion>.
- Coello, I. N. (2021). Análisis de ciberataques en organizaciones públicas del Ecuador y sus impactos administrativos. Obtenido de <https://dspace.ups.edu.ec/handle/123456789/20738>
- Cohen, N., & Gómez, G. (2019). *Metodología de la investigación, ¿para qué?: la producción de los datos y los diseños*. Buenos Aires: Editorial Teseo. Obtenido de [http://biblioteca.clacso.edu.ar/clacso/se/20190823024606/Metodologia\\_para\\_que.pdf](http://biblioteca.clacso.edu.ar/clacso/se/20190823024606/Metodologia_para_que.pdf)
- Fundación Proydesa. (2022). *¿Qué es UPnP y para qué sirve?* Obtenido de <https://www.proydesa.org/portal/index.php/noticias/1746-que-es-upnp-y-para-que-sirve>
- Ghimiray, D. (2022). *¿Qué es UPnP (Universal Plug and Play)?* Obtenido de <https://www.avg.com/es/signal/what-is-unpn>
- Grimmett, J., & O'Neill, E. (2012). UPnP: Breaking out of the LAN. *IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*. Obtenido de <https://doi.org/10.1109/WCNCW.2012.6215483>
- Guo, X., & Li, J. Z. (2013). Secure UPnP services based on group signature algorithm. *Proceedings of 2013 3rd International Conference on Computer Science and Network Technology*. Obtenido de <https://doi.org/10.1109/ICCSNT.2013.6967261>
- Hasib, A. A., & Mottalib, M. (2010). Vulnerability Analysis and Protection Schemes of Universal Plug and Play Protocol. *IEEE International Conference on Computational Science and Engineering*. Obtenido de <https://doi.org/10.1109/CSE.2010.37>
- IEEE Xplore. (2022). *IEEE Xplore Digital Library*. Obtenido de <https://biblioguias.uam.es/tutoriales/ieeexplore>

- Iño, W. G. (2018). Investigación educativa desde un enfoque cualitativo. *Dialnet*. Obtenido de <https://dialnet.unirioja.es/servlet/articulo?codigo=6521971>
- Izaguirre Olmedo, J., & León Gavilánez, F. (2018). Análisis de los ciberataques realizados en América Latina. *INNOVA Research Journal*, 172–181. Obtenido de <https://doi.org/10.33890/innova.v3.n9.2018.837>
- Kayas, G., Hossain, M., Payton, J., & Islam, S. M. (2020). VSDM: A Virtual Service Device Management Scheme for UPnP-Based IoT Networks. *IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference*. Obtenido de <https://doi.org/10.1109/UEMCON51285.2020.9298148>
- Kayas, G., Hossain, M., Payton, J., & Islam, S. M. (2021). SUPnP: Secure Access and Service Registration for UPnP-Enabled Internet of Things. *IEEE Internet of Things Journal*. Obtenido de <https://doi.org/10.1109/JIOT.2021.3058699>
- Kayas, G., Hossain, M., Payton, J., & Islam, S. R. (2020). An Overview of UPnP-based IoT Security: Threats, Vulnerabilities, and Prospective Solutions. *IEEE Annual Information Technology, Electronics and Mobile Communication Conference*. Obtenido de <https://doi.org/10.1109/IEMCON51383.2020.9284885>
- Kitchenham, B., & Charters, S. (2007). *Guidelines for Performing Systematic Literature Reviews in Software Engineering*.
- Maestre, J. (2015). *Domótica para Ingenieros*. Madrid: Editorial Paraninfo .
- Martín, S. G., & Lafuente, V. (2017). Referencias bibliográficas: indicadores para su evaluación en trabajos científicos. *Scielo*. Obtenido de [https://www.scielo.org.mx/scielo.php?script=sci\\_arttext&pid=S0187-358X2017000100151](https://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S0187-358X2017000100151)
- Microsoft. (2022). *Introducción a Plug and Play*. Obtenido de <https://learn.microsoft.com/eses/windows-hardware/drivers/kernel/introduction-to-plug-and-play>
- Miranda, C. K. (2019). Estudio de Riesgos y Vulnerabilidades de la Red LAN y Equipos del Infocentro San Juan. Obtenido de <http://dspace.utb.edu.ec/handle/49000/6901>
- Morales, J. G. (2022). Influencia del covid 19 en el incremento de los ciberataques a nivel mundial. Obtenido de <http://repository.unipiloto.edu.co/handle/20.500.12277/11574>
- Mussi, G., Barreto, L., & Cugnasca, C. E. (2016). An UPnP architecture for interoperability in Home Area Network. *IEEE International Symposium on Consumer Electronics (ISCE)*. Obtenido de <https://doi.org/10.1109/ISCE.2016.7797366>
- Nava-Lopez, I., Prudente-Tixteco, L., Olivares-Mercado, J., Sanchez-Perez, G., ToscanoMedina, K., & Castro-Madrid, L. C. (2019). Security Tool for UPnP protocol on Smart TV. *IEEE International Autumn Meeting on Power, Electronics and Computing (ROPEC)*. Obtenido de <https://doi.org/10.1109/ROPEC48299.2019.9057057>
- Pehkonen, V., & Koivisto, J. (2010). Secure Universal Plug and Play network. *Sixth International Conference on Information Assurance and Security*. Obtenido de <https://doi.org/10.1109/ISIAS.2010.5604189>
- Pons Gamon, V. (2017). Internet, la nueva era del delito: ciberdelito, ciberterrorismo, legislación y ciberseguridad. *URVIO - Revista Latinoamericana de Estudios de Seguridad*, (20), 80–93. Obtenido de <https://doi.org/10.17141/urvio.20.2017.2563>
- Rapid7. (2013). *Security Flaws in Universal Plug and Play: Unplug, Don't Play*. Obtenido de <https://www.rapid7.com/blog/post/2013/01/29/security-flaws-in-universal-plug-and-playunplug-dont-play/>
- Salazar, J. D., & Novoa, N. U. (2022). Análisis y recomendaciones del estado de seguridad de una empresa de compra de cartera y riesgos de negocios. Obtenido de <http://hdl.handle.net/20.500.12495/8124>

- Sales, T., Sales, L., Almeida, H., Perkusich, A., & Sales, M. (2013). Multilevel security in UPnP networks for pervasive environments. *IEEE Transactions on Consumer Electronics*. Obtenido de <https://doi.org/10.1109/TCE.2013.6490253>
- Sánchez, F. A. (2019). Fundamentos epistémicos de la investigación cualitativa y cuantitativa: Consensos y disensos. *Revista Digital De Investigación En Docencia Universitaria*, 13(1), 101–122. Obtenido de <https://doi.org/10.19083/ridu.2019.644>
- Santos, M., Cerdeira, F., Julio, E., Dembogurski, B., Silva, G., & Silva, E. (2020). An IEEE 1451 Standard-based Plug-and-Play Architecture to Empower the Internet of Things. *IEEE Latin America Transactions*. Obtenido de <https://doi.org/10.1109/TLA.2020.9400431>
- Zheng, H., Li, C., & Chen, Z. (2011). Petri Nets Based Modeling and Analysis of UPnP Security Ceremonies. *Third Pacific-Asia Conference on Circuits, Communications and System (PACCS)*. Obtenido de <https://doi.org/10.1109/PACCS.2011.5990235>



*Recibido 9 FEB. 2024*

*ReCIBE, Año 12 No.2, NOV. 2023*

*Aceptado 12 FEB. 2024*

## **Métricas de productividad y gestión del trabajo basadas en psp para estudiantes de ingeniería de sistemas: caso universidad eafit**

**Productivity and work management metrics based on psp for systems engineering students: case study at eafit university**

**Paola Vallejo<sup>1</sup>**

pvallej3@eafit.edu.co

**Andrés Echeverri Jaramillo<sup>1</sup>**

**Liliana González Palacio<sup>1</sup>**

**Rafael David Rincón<sup>1</sup>**

Universidad EAFIT, Medellín, Colombia



## **RESUMEN**

El Proceso Personal de Software (PSP) se utiliza en la industria del software, Ingeniería de Sistemas y programas de formación relacionados. Estudios previos muestran que su inclusión en el currículo mejora la capacidad de los estudiantes para estimar tiempo y recursos, así como para gestionar y controlar eficientemente el proceso de desarrollo de software. Este estudio busca evaluar el impacto de la implementación del PSP en estudiantes de primer semestre de Ingeniería de Sistemas de la Universidad EAFIT, a través de métricas de productividad y gestión del trabajo, con el propósito de mejorar su disciplina, desempeño y la calidad del desarrollo de software. Se presenta un caso de implementación de PSP en el curso Seminario de Ingeniería de Sistemas, a partir de un conjunto de preguntas orientadoras. Cada estudiante registra métricas como cantidad de líneas de código por hora, tiempo de desarrollo por programa y número de errores. Los hallazgos indican que la aplicación del PSP y las métricas de productividad pueden mejorar significativamente la disciplina y el desempeño de los estudiantes, permitiendo un seguimiento más preciso del progreso y una mejor gestión del trabajo. Este artículo ofrece una perspectiva valiosa para los educadores y estudiantes de Ingeniería de Sistemas que deseen implementar buenas prácticas de programación y gestión del trabajo, así como mejorar la calidad del producto desarrollado.

## **PALABRAS CLAVE**

Proceso Personal de Software (PSP), PSP en el aula, Medición de desempeño, Desarrollo de software.

## **ABSTRACT**

The Personal Software Process (PSP) is utilized in the software industry, Systems Engineering, and related training programs. Previous studies demonstrate that its inclusion in the curriculum enhances students' ability to estimate time and resources and efficiently manage and control the software development process. This study aims to evaluate the impact of implementing PSP on first-semester Systems Engineering students at EAFIT University, using productivity and work management metrics to improve their discipline, performance, and the quality of software development. A case of PSP implementation in the Systems Engineering Seminar course is presented, guided by a set of questions. Each student records metrics such as lines of code per hour, development time per program, and the number of errors. Findings indicate that applying PSP and productivity metrics can significantly improve students' discipline and performance, enabling more precise progress monitoring and better work management. This article provides valuable insights for educators and Systems Engineering students looking to implement good programming and work management practices and enhance the quality of the developed product.

## **KEYWORDS**

Personal Software Process (PSP), PSP in the classroom, Performance measurement, Software development.

## **1. INTRODUCCIÓN**

El Proceso Personal de Software (PSP) es ampliamente reconocido y adoptado en la industria del software, en la Ingeniería de Sistemas y programas de formación relacionados [Shen, Hsueh, & Lee, 2011; Razzaq et al., 2020]. Investigaciones anteriores han mostrado los beneficios de incorporar el PSP en el plan de estudios, mejorando la capacidad de los estudiantes para estimar el tiempo y los recursos necesarios para la realización de tareas, así como su competencia en la gestión y el control efectivo del proceso de desarrollo de software [Shen et al., 2011]. Este estudio investiga la aplicación práctica de métricas de productividad y técnicas de gestión del trabajo PSP en estudiantes de primer semestre del programa de Ingeniería de Sistemas de la Universidad EAFIT de Colombia. Se presenta un caso de implementación de los principios de PSP en dos grupos (61 estudiantes) del curso Seminario de Ingeniería de Sistemas, utilizando un conjunto de preguntas orientadoras.

Durante el estudio, cada estudiante registra métricas esenciales, como la cantidad de líneas de código por hora, el tiempo de desarrollo por programa y el número de errores. El estudio destaca los beneficios de utilizar PSP y métricas de productividad para mejorar la disciplina y el rendimiento de los estudiantes. Además, muestra cómo el PSP puede facilitar un seguimiento más preciso del progreso y una mejora en las capacidades de gestión del trabajo. Los resultados resaltan las ventajas de incorporar el PSP y las métricas de productividad a la enseñanza de Ingeniería de Sistemas.

En resumen, este artículo analiza la aplicación del PSP y las métricas de productividad en estudiantes de primer semestre de Ingeniería de Sistemas. Estas ideas son valiosas para educadores y estudiantes interesados en adoptar las mejores prácticas de gestión del trabajo, mejorando la calidad general de los productos de software que desarrollan y contribuyendo al objetivo general de promover la excelencia en los procesos de desarrollo de software y fortalecer el crecimiento profesional de los estudiantes.

Este artículo se organiza como se indica a continuación: la siguiente sección presenta un marco de referencia que relaciona conceptos y estudios sobre el PSP. La sección 3 presenta un paso a paso general para aplicar el PSP en el curso de Seminario de Ingeniería de Sistemas. La sección 4 presenta los resultados obtenidos. Finalmente, la sección 5 enuncia conclusiones y ofrece algunas perspectivas.

## **2. MARCO DE REFERENCIA**

### **2.1 Conceptos**

PSP es una metodología que brinda técnicas y herramientas para la gestión individual del trabajo en el desarrollo de software, permitiendo a los profesionales medir y mejorar su productividad y calidad en el proceso [Razzaq et al., 2020]. En la educación universitaria, el PSP ha sido objeto de estudios y experiencias que han mostrado diversos beneficios y aplicaciones. Estos incluyen mejorar la calidad del trabajo, establecer prácticas disciplinadas, desarrollar habilidades de gestión del tiempo, aprender a estimar tiempos y esfuerzos, fomentar buenas prácticas de programación, practicar la autoevaluación y el autoaprendizaje, y lograr una mejora continua en su proceso de desarrollo de software [Soto & Rafael, 2020; Ramingwong & Ramingwong, 2012].

## 2.2 Trabajo relacionado

En esta sección se presentan ejemplos de uso del Proceso Personal de Software (PSP) en cursos de Ingeniería de Sistemas a nivel universitario.

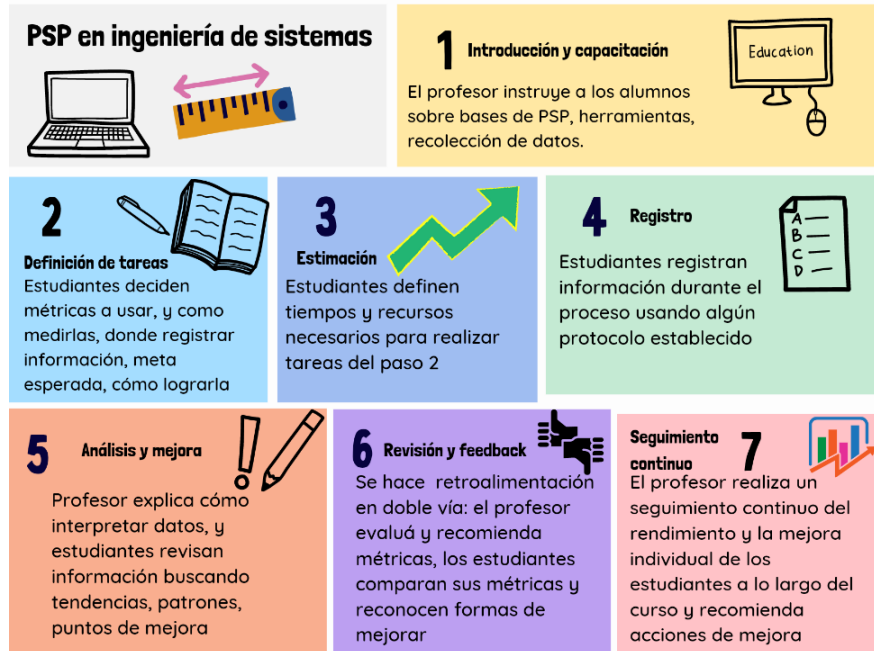
Kusakabe y otros analizaron cómo el uso de técnicas de estimación basadas en el PSP en un curso de ingeniería de software resultó en estimaciones más precisas y realistas, lo que mejoró la planificación y gestión de los proyectos [Kusakabe et al., 2020]. Denwattana encontró que los estudiantes, al aplicar PSP, mejoran significativamente la calidad de su trabajo, su habilidad para estimar tareas y administrar el tiempo de manera efectiva. Además, los estudiantes adquirieron una mentalidad más disciplinada y sistemática en su enfoque hacia el desarrollo de software [Ramingwong & Ramingwong, 2012].

Contreras y Soto, por su parte, realizaron dos estudios sobre el impacto de usar PSP para el rendimiento de los estudiantes. Los resultados mostraron mejoras significativas en la calidad del código, la estimación precisa de tareas y la reducción de defectos en el trabajo de los estudiantes. Además, se observó una mayor conciencia sobre el manejo de métricas [Soto & Rafael, 2020; Contreras-Vas et al., 2021]. Otros estudios sobre PSP en la academia están más orientados a conocer la percepción final de los estudiantes al pedirles que usen este modelo para medir su desempeño [Contreras et al., 2020]. El resultado final permitió a los autores mejorar las estrategias de enseñanza de los conceptos asociados a PSP y orientar de otras formas el curso.

Por último, se encontró una revisión de literatura sobre diversos experimentos donde se aplica PSP en entornos universitarios [Arduino & Bollati, 2020]. Se evidencia que, en general, los autores reconocen mejoras en el proceso de codificación de los estudiantes, mediante métricas como cantidad de líneas de código, disminución de errores y mayor productividad. Al analizar las referencias consultadas, es posible identificar un patrón de mejora en el desempeño de los estudiantes, igual se ratifica en nuestro trabajo, y se proponen opciones para darle continuidad e integrar con otros cursos del programa Ingeniería de Sistemas de la Universidad EAFIT.

### 3. MÉTODO

El siguiente es un paso a paso general para aplicar el Proceso Personal de Software (PSP) en el curso de Seminario de Ingeniería de Sistemas (ver figura 1).



**Figura 1.** Aplicación del PSP en Seminario de Ingeniería de Sistemas [Borstler et al., 2002; Venkatasubramanian et al., 2001; Maletic et al., 2001]

Las preguntas orientadoras del ejercicio indicado en la figura 1 fueron: ¿Cuáles son las métricas que se utilizarán en el desarrollo de software? ¿Cuál es mi desempeño actual? ¿Cuál es la meta esperada o deseada? ¿Cómo espera alcanzar la meta? ¿Qué se concluye del ejercicio? [Borstler et al., 2002; Venkatasubramanian et al., 2001; Maletic et al., 2001].

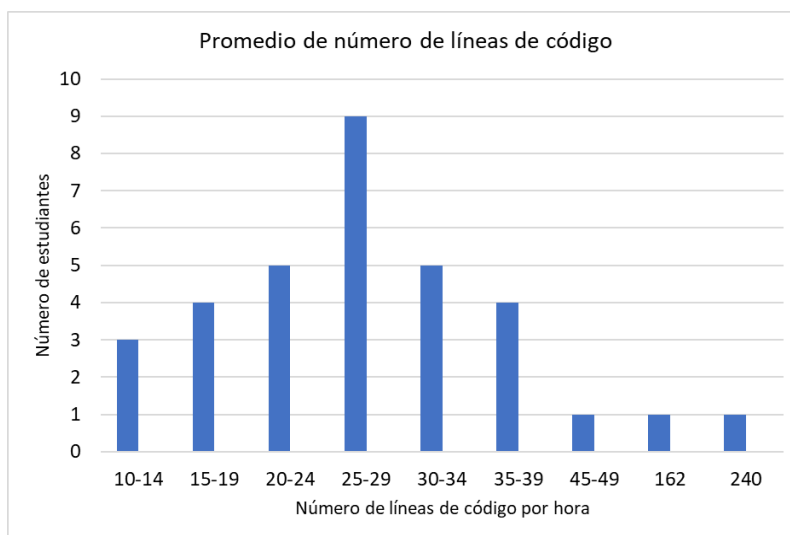
Los cursos involucrados en el ejercicio fueron: a) Fundamentos de Programación, donde los estudiantes aplican la teoría de PSP para evaluar su desempeño en las prácticas continuas de programación; b) Seminario de Ingeniería de Sistemas, planteando un reto donde los estudiantes usaban los resultados obtenidos en sus prácticas de programación para hacer seguimiento y recopilar información asociada a las preguntas orientadoras especificadas anteriormente (la cantidad de líneas de código por hora, errores, etc). La población estuvo compuesta por estudiantes de primer semestre de Ingeniería de Sistemas de la universidad EAFIT, cursando las dos asignaturas ya enunciadas.

Si otros docentes desean aplicar un ejercicio como el planteado en este artículo, es importante adaptar y personalizar los pasos mostrados en la figura 1 según las necesidades y características del curso a intervenir.

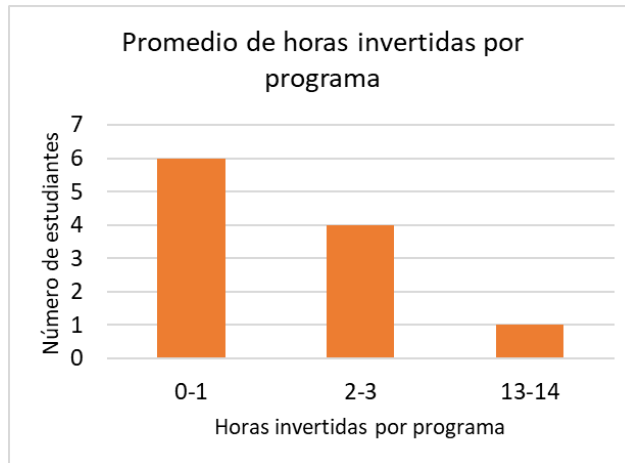
#### 4. RESULTADOS

Mediante este ejercicio, los estudiantes identificaron formas para medir su desempeño. La mayoría de ellos utilizaron métricas como: número de líneas de código por hora, cantidad de errores y tiempo de desarrollo.

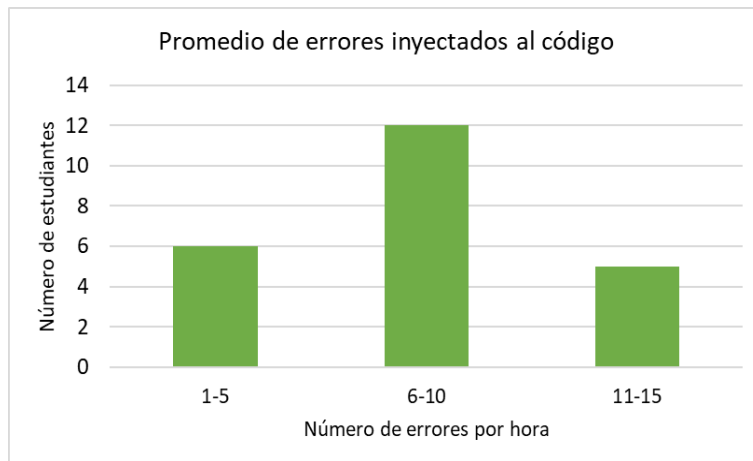
La figura 2 ilustra el promedio de número de líneas de código por hora. Se observa que la mayoría de los estudiantes dedicaron entre 25 y 29 horas al desarrollo de sus actividades, lo que sugiere un grupo significativo de individuos comprometidos y dispuestos a invertir un tiempo considerable en ellas. Además, se observan grupos más pequeños en otros rangos de tiempo, como 20 a 24 horas y 30 a 34 horas, que también demuestran un nivel de compromiso en la actividad. En contraste, algunos rangos de tiempo, como 45 a 49 horas, tienen una representación mínima, lo que indica que pocos participantes han invertido un tiempo tan extenso. La figura 3 presenta la cantidad de horas invertidas por cada programa, es notable que la mayoría de los estudiantes dedicaron entre 0 y 1 hora al desarrollo de sus programas, lo que indica una preferencia por programas que requieren una inversión de tiempo mínima o la simplicidad de los mismos. Por otro lado, un grupo significativo de estudiantes invirtió entre 2 y 3 horas, lo que sugiere un compromiso moderado con programas más extensos. Además, un estudiante dedicó un tiempo sustancial, entre 13 y 14 horas, a un programa en particular, lo que podría indicar un alto nivel de interés o la necesidad de una capacitación más exhaustiva. La figura 4 presenta el promedio de errores inyectados al código, se destaca que la mayoría de los estudiantes tienen un promedio de errores que oscila entre 6 y 10, lo que sugiere un nivel moderado de precisión en su trabajo. Además, 6 estudiantes presentan un promedio de errores más bajo, entre 1 y 5, indicando un mejor rendimiento. Por otro lado, 5 estudiantes tienen un promedio de errores en el rango de 11 a 15, lo que podría requerir una atención adicional para mejorar la calidad de su trabajo.



**Figura 2.** Promedio de número de líneas de código



**Figura 3.** Número de horas invertidas



**Figura 4.** Cantidad de errores inyectados

Es importante mencionar que se observaron datos inexactos. Por ejemplo, en la figura 2, se registraron valores de 162 y 240 líneas de código por hora, los cuales no parecen realistas. Al investigar más a fondo, se descubrió que los estudiantes que obtuvieron estos resultados inicialmente tomaron la medida en minutos y concluyeron que podían escribir 2.6 o 4 líneas de código por minuto. Esta extrapolación a horas resultó en valores poco precisos. Se especula que, durante la medición, no tuvieron en cuenta el tiempo dedicado a analizar el problema, diseñar la estructura del código, entre otros aspectos. Esto llevó a que finalizaran rápidamente cada programa, que les tomaba entre 10 y 20 minutos, lo cual no refleja una medida precisa de su productividad.

Además, surgieron otras medidas que no fueron tan usadas por los estudiantes pero que sería valioso incorporar. Dos ejemplos que destacan son el tiempo invertido en el diseño y el tiempo invertido en hacer pseudocódigo. Los estudiantes que usaron estas métricas mostraron mayor productividad al completar el código y también mostraron una visión más realista y clara de cómo mejorar sus tiempos. Estas prácticas les han aportado mucho,

especialmente considerando que son ejercicios de primer semestre, que, aunque son simples, tendrán un impacto aún mayor en proyectos de mayor escala. Este trabajo les ayuda a comprender la importancia de este tipo de prácticas y cuánto les beneficiarán. Otra medida relevante es el tiempo de interrupción según lo reportan los estudiantes. Algunos estudiantes midieron no solo cuánto tiempo dedicaron a realizar las tareas, sino también cuánto tiempo se distraían utilizando el celular, viendo vídeos en YouTube o utilizando redes sociales.

La mayoría de los estudiantes solo lograban en principio expresar sus aspiraciones en cuanto a métricas como líneas de código esperadas. Al comenzar este ejercicio lograron vislumbrar otras formas de medir su desempeño, y con un poco más de documentación lograron proyectar mejor su meta. Es lógico esto porque si hay desconocimiento de herramientas estructuradas para gestionar el trabajo, es complejo comenzar a registrar información y saber cómo emplearla para la mejora continua.

Además de esto, varios estudiantes reconocieron áreas de mejora en su desempeño en programación, incluyendo la procrastinación, la falta de planificación y la necesidad de practicar más. Los estudiantes reconocieron la importancia de implementar el diseño antes de escribir el código y utilizar los recursos disponibles. El uso de PSP se consideró valioso para mejorar sus habilidades. La práctica continua y la resolución eficiente de problemas son cruciales, y todos buscan un progreso constante en su desarrollo como programadores.

Los resultados reflejan que la implementación del PSP produce un impacto positivo en la disciplina y el desempeño de los estudiantes. Este enfoque no solo conduce a una mejora sustancial en la calidad de trabajo de los estudiantes, sino que también facilita un seguimiento más preciso del progreso individual y fomenta una gestión más eficiente y efectiva del trabajo en el contexto del desarrollo de software. Estos hallazgos subrayan la importancia de la adopción de prácticas estructuradas y el uso de métricas en la educación en Ingeniería de Sistemas, brindando beneficios tangibles tanto para los educadores como para los estudiantes.

## **6. CONCLUSIONES Y TRABAJO FUTURO**

Este artículo presenta un estudio donde se usan las métricas de productividad y gestión del trabajo basadas de PSP con estudiantes de Ingeniería de Sistemas en la Universidad EAFIT-Colombia. Hay diversas conclusiones, que se presentarán por cada pregunta orientadora del ejercicio.

En relación con la pregunta ¿Cuáles son las métricas que se utilizarán en el desarrollo de software? Es posible concluir que la mayoría de los estudiantes usó el número de líneas de código por hora, la cantidad de errores y tiempo de desarrollo para medir su desempeño. Aparecieron otras alternativas que los estudiantes no emplearon con frecuencia, pero cuya inclusión resultaría beneficiosa. Dos ejemplos destacados incluyen el tiempo dedicado al proceso de diseño y el tiempo invertido en la elaboración de pseudocódigo. Quienes incorporaron estas métricas demostraron una mayor eficiencia en la finalización del código, además de adquirir una mejor comprensión de cómo optimizar sus tiempos.

Frente a la pregunta ¿Cuál es mi desempeño actual? La mayoría de los estudiantes dedicaron 25 a 29 horas a sus actividades, lo que muestra un compromiso considerable.

También hay grupos más pequeños en los rangos de 20 a 24 horas y 30 a 34 horas, que indican un nivel de compromiso. Por otro lado, hubo poca representación en rangos de tiempo como 45 a 49 horas. En cuanto al tiempo invertido en programas, la mayoría dedicó de 0 a 1 hora, mientras que un grupo significativo invirtió entre 2 y 3 horas. Un estudiante destacó al invertir 13 a 14 horas en un programa. En cuanto a errores, la mayoría tuvo un promedio de 6 a 10, algunos tuvieron menos errores (1 a 5), y otros más (11 a 15), lo que podría requerir atención adicional.

Frente a la pregunta ¿Cuál es la meta esperada o deseada? y ¿Cómo espera alcanzar la meta? La mayoría de los estudiantes inicialmente solo expresaban sus expectativas en términos de líneas de código. Sin embargo, al comenzar el ejercicio, descubrieron formas adicionales de medir su desempeño y con más documentación lograron definir mejor sus metas. Esto es comprensible, ya que la falta de familiaridad con herramientas estructuradas dificulta el registro y uso de información para la mejora continua. Además, varios estudiantes identificaron áreas de mejora en su programación, como la procrastinación, la falta de planificación y la necesidad de practicar más. También reconocieron la importancia de diseñar antes de codificar y aprovechar los recursos disponibles. El uso de PSP se consideró valioso para mejorar sus habilidades. Todos buscan un progreso constante en su desarrollo como programadores a través de la práctica continua y la resolución eficiente de problemas.

Por último, ante la pregunta ¿Qué se concluye del ejercicio? El primer hallazgo valioso es que la medición del tiempo dedicado a las tareas junto con el tiempo de distracción proporciona una visión más completa sobre cómo los estudiantes administran su tiempo y cómo esto afecta su productividad. Esta perspectiva holística indica que los estudiantes están reconociendo la importancia de considerar tanto el tiempo de trabajo como el tiempo de distracción para obtener una imagen precisa de su eficiencia.

Los resultados de este estudio pueden ser de gran utilidad para identificar patrones y tendencias en el comportamiento de los estudiantes con respecto a las distracciones digitales. También dan un panorama más exacto del tiempo que le dedican a sus trabajos académicos. Es probable que los denominados “distractores” incidan a favor, si se saben canalizar.

La información recopilada servirá como base para el desarrollo de estrategias destinadas a minimizar o encaminar las interrupciones y mejorar la gestión del tiempo de los estudiantes, lo cual a su vez puede tener un impacto positivo en su eficacia y rendimiento académico, y, por ende, el índice de deserción. Además, los hallazgos revelan que las distracciones digitales, como el uso de dispositivos móviles, ver videos en YouTube y utilizar redes sociales, representan una fuente significativa de interrupción para los estudiantes. También se puede ratificar lo expresado en estudios previos, en cuanto a mejoras significativas en la calidad del código, la estimación precisa de tareas y la reducción de defectos en el trabajo de los futuros ingenieros. Se observó también una mayor conciencia sobre el manejo de métricas [Soto & Rafael, 2020; Contreras-Vas et al., 2021].

Como trabajo futuro se espera realizar un estudio longitudinal para evaluar los efectos a mediano plazo de la aplicación de PSP en la educación de Ingeniería de Sistemas. Esto implica monitorear a una cohorte de estudiantes durante varios semestres (en cursos como Ingeniería de Software y Proyecto Integrador) para evaluar su mejora en la estimación del



tiempo de finalización de tareas, la gestión del proceso de desarrollo de software y la mejora del rendimiento general. También se puede dirigir hacia el trabajo de (proceso de software en equipo) TSP. Además, nos interesa colaborar con profesionales de la industria del software para evaluar la transferibilidad de las habilidades y prácticas de PSP desde entornos académicos hasta entornos de desarrollo de software del mundo real. Sería interesante explorar la alineación entre las habilidades adquiridas a través de PSP en el contexto académico y las expectativas y requisitos de los profesionales de la industria.

## REFERENCIAS

Shen, W.-H., Hsueh, N.-L., & Lee, W.-M. (2011). Assessing PSP effect in training disciplined software development: A Plan–Track–Review model. *Information and Software Technology*, 53(2), 137-148.

Razzaq, A., Ahmad, S., Khalil, A., & Ahmad, S. (2020). Systematic Mapping Study: Augmenting Personal Software Process Analysis For Extreme Programming Teams. *Authorea Preprints*.

Ramingwong, S., & Ramingwong, L. (2012). Implementing a personal software process (PSPSM) course: a case study. *Journal of Software Engineering and Applications*, 5 (8), 639-644.

Soto, B. P., & Rafael, G. R. (2020). Habilidades de Personal Software Process (PSP) para la industria del software en Latinoamérica. *Industrial data*, 23(1), 229-244.

Kusakabe, S., Araki, S., Katamine, K., & Umeda, M. (2020). Analyzing Motivation in Personal Software Process Education Course with a Qualitative Approach. In 2020 9th International Congress on Advanced Applied Informatics (IIAI-AAI) (pp. 298-303). IEEE.

Contreras-Vas, J. Á., Arias-Masa, J., Hidalgo-Izquierdo, V., & Martín-Espada, R. (2021). Personal Software Process: A Study from an Academic Perspective. In *Computer Supported Qualitative Research: New Trends in Qualitative Research (WCQR2021)* 5 (pp. 143-154). Springer.

Contreras, J. Á., Arias, J., Hidalgo, V., & Martín, R. (2020). Análisis cualitativo de las respuestas de los alumnos sobre aspectos del curso del Proceso de Software Personal. *New Trends in Qualitative Research*, 4, 304-316.

Arduino, G. A., & Bollati, V. A. (2020). Proceso Personal de Software aplicado a la formación universitaria: una revisión sistemática de literatura. In 2020 IEEE Congreso Bienal de Argentina (ARGENCON) (pp. 1-7). IEEE.

Borstler, J., Carrington, D., Hislop, G. W., Lisack, S., Olson, K., & Williams, L. (2002). Teaching PSP: Challenges and lessons learned. *Ieee Software*, 19(5), 42-48.

Venkatasubramanian, K., Roy, S. B. T., & Dasari, M. V. (2001). Teaching and using PSP in a software engineering course: an experience report. In *Software Engineering Education and Training Annual Conference*.

Maletic, J. I., Howald, A., & Marcus, A. (2001, February). Incorporating PSP into a traditional software engineering course: an experience report. In Proceedings 14th Conference on Software Engineering Education and Training ('In search of a software engineering profession') (pp. 89-97). IEEE.

## NOTAS BIOGRÁFICAS



**Andrés Echeverri Jaramillo** es Estudiante de ingeniería de sistemas en la universidad de EAFIT, actualmente está como practicante en CodeBulls S.A.S.



**Paola Vallejo** es Doctora en Ciencias de la Computación. Actualmente, es docente e investigadora en la Universidad EAFIT en Medellín, Colombia. Imparte asignaturas relacionadas con Ingeniería de Software. Su interés se enfoca en arquitecturas de software, diseño de software y código limpio.



**Liliana Gonzalez Palacio** es Doctora en ingeniería. Actualmente, es docente e investigadora en la Universidad EAFIT en Medellín, Colombia. Imparte asignaturas relacionadas con Ingeniería de Software. Su interés se enfoca en ingeniería de requisitos, educación gamificada en ingeniería, tecnología para población con discapacidad.



**Rafael David Rincón Bermúdez** es Magister en Sistemas de Calidad del Tec de Monterrey, México, Magister en Matemáticas Aplicadas de la U Eafit de Medellín, Colombia. Certificado como Data Scientist en Big Data por ARCITURA. Consultor en temas de Analytics, Definición, Mejora y Automatización de Procesos de Negocio, Automatización Robótica de Procesos con RPA, Análisis de Información, Implementación de Sistemas de Calidad, Modelos y Estándares Internacionales de Calidad de Software, como CMMI (Capability Maturity Model Integration), Competisoft, ITIL, entre otros. Imparte asignaturas relacionadas con Calidad de Software, Estadística y Transformación Digital. Su interés se enfoca en Automatización con RPA, Inteligencia Artificial, Analítica de Datos, EcoSistemas Digitales.



Esta obra está bajo una licencia de Creative Commons Reconocimiento-NoComercial-CompartirIgual 2.5 México.

*Recibido 09 DIC. 2023*

*ReCIBE, Año 12 No.2, NOV. 2023*

*Aceptado 12 FEB. 2024*

## **La gestión de procesos en las empresas de software de San Francisco de Campeche**

### **Process management in the software companies of San Francisco de Campeche**

**Diana Concepción Mex Alvarez<sup>1</sup>**  
**Luz María Hernández Cruz<sup>1</sup>**  
**Charlotte Monserrat de Jesús Llanes Chiquini<sup>1</sup>**  
**Andrés Castro Villagrán<sup>2</sup>**  
**Susana Friné Moguel Marín<sup>1</sup>**  
**Carolina Chi Arceo<sup>1</sup>**

<sup>1</sup>Universidad Autónoma de Campeche

<sup>2</sup>Instituto tecnológico superior de Champotón

## **RESUMEN**

Las micro, pequeñas y medianas empresas desarrolladoras de software, tanto en países industrializados como en los de menor grado de desarrollo, se enfrentan a diversos retos ante una competitividad mundial. Conocer las fortalezas y debilidades representa un factor determinante para mejorar su capacidad y direccionar la economía en el impulso de esta industria. En este trabajo presentamos los resultados de un estudio diagnóstico a las empresas de la industria de software que operan en la ciudad de San Francisco de Campeche, sobre el desarrollo de sus procesos y las métricas que emplean para medir sus productos y servicios. Para efectuar la medición se consideraron los indicadores obtenidos de una revisión sistemática previa para conocer las características de la gestión de sus procesos. Los resultados muestran las principales características en las que desarrollan sus procesos; y de esta manera examinar las áreas de oportunidad con el fin de elevar su productividad y crecimiento económico, lo que se traduce en beneficio para la sociedad campechana.

## **PALABRAS CLAVE**

Industria, Software, Procesos, Gestión.

## **ABSTRACT**

Micro, small and medium-sized software development companies, both in industrialized and less developed countries, face various challenges in the face of global competitiveness. Knowing their strengths and weaknesses is a determining factor for improving their capacity and directing the economy to promote this industry. In this paper we present the results of a diagnostic study of software industry companies operating in the city of San Francisco de Campeche, on the development of their processes and the metrics they use to measure their products and services. To carry out the measurement, the indicators obtained from a previous systematic review were considered to know the characteristics of the management of their processes. The results show the main characteristics in which they develop their processes; and in this way examine the areas of opportunity in order to increase their productivity and economic growth, which translates into benefits for the society of Campeche.

## **KEYWORDS**

Industry, Software, Processes, Management.

# 1. INTRODUCCIÓN

## 1.1. Empresas desarrolladoras de software

En el estudio de los mercados industriales, la industria de software a nivel mundial ha presentado un crecimiento notorio [1]. La industria del software es considerada una industria de conocimiento perteneciente a un sector estratégico de clase mundial que requiere de altos niveles de innovación y desarrollo [2].

Debido a los altos niveles de innovación, desarrollo y competitividad, el crecimiento de las empresas de software se limita [3] al no poseer los niveles de capacidad necesarios para responder a los retos de esta industria [1]. Las micro pequeñas medianas empresas (MiPymes) desarrolladoras de software, tienen como características principales: contar con recursos humanos y capital reducido, no contar con un estándar de desarrollo implementado, comprometerse a cualquier tipo de trabajo que se les presenta, notándose la falta de especialización en un nicho determinado y generado estimaciones en tiempo y recursos poco reales. [4]

Este comportamiento ha suscitado intereses a nivel gubernamental, académico y empresarial que posicionan a esta industria como centro de desarrollo de las naciones por basarse en la innovación, conocimiento y sector de servicios [1]. En la actualidad no existen estudios recientes que indiquen o detallen la condición en la que se encuentra la industria del software en San Francisco de Campeche, por ello se diseñó un instrumento para recabar los principales indicadores que permitan conocer las métricas empleadas para evaluar sus productos y servicios.[5]

## 1.2. Aspectos de optimización para la mejora continua de sus procesos

En este trabajo se indagó sobre la manera en la que las empresas desarrolladoras de software toman decisiones que repercutirán en la organización y despliegue de estas. El Análisis y Resolución Causal al igual que la Innovación y Despliegue Organizacional, establecen prácticas que permiten optimizar el proceso a nivel de proyecto o de la organización y requiere un entendimiento cuantitativo del proceso para poder ser efectivas.

### 1.2.1. Análisis y Resolución Causal (CAR)

El análisis y resolución causal identifica las causas de los defectos y de otros problemas, tomando acciones para prevenir que ocurran en el futuro.

1. Las causas raíz de los defectos y de otros problemas son determinadas sistemáticamente.
2. Las causas raíz de los defectos y de otros problemas son tratadas sistemáticamente para prevenir su ocurrencia futura. [7]

### 1.2.2. Innovación y Despliegue Organizacional (OID)

La innovación y despliegue organizacional selecciona y despliega mejoras incrementales para mejorar la habilidad de la organización para alcanzar los objetivos de calidad y rendimiento del proceso.

1. Las mejoras de proceso y tecnología que contribuyen a cumplir los objetivos de calidad y desempeño del proceso son seleccionadas.
2. Las mejoras medibles a los procesos y a las tecnologías de la organización son desplegadas continua y sistemáticamente.
  - 2.1 Establecer y mantener los planes para desplegar las mejoras seleccionadas de proceso y de tecnología
  - 2.2 Gestionar el despliegue de las mejoras seleccionadas de proceso y de tecnología
  - 2.3 Medir los efectos de las mejoras desplegadas de proceso y de tecnología. [8]

## 1.3. Modelos de proceso

En esta sección, se exploró detalladamente los modelos de proceso adoptados por las empresas analizadas, reconociendo su importancia para el desarrollo efectivo y el crecimiento sostenible en la industria del software. Asimismo, se examinó las certificaciones relacionadas con las capacidades del personal, destacando su papel crucial en el impulso hacia la mejora continua de estas empresas.

A continuación, se presentan análisis específicos de los modelos de procesos más relevantes:

### **1.3.1 MoProsoft**

El Modelo MoProSoft está dirigido a las Micro y pequeñas empresas o áreas internas dedicadas al desarrollo o mantenimiento de software, este fue el motivo de elegir la norma mexicana como una referencia de certificación y conocer el porcentaje de las empresas de software de la ciudad de San Francisco de Campeche cuentan con una base que les permita estandarizar sus proyectos. MoProSoft cuenta con diferentes niveles para que las empresas estén certificadas. [9]

### **1.3.2 CMMI**

El Modelo de Madurez de Capacidades (CMMI) es un metamodelo de proceso exhaustivo que se impulsa en un conjunto de sistemas y capacidades de ingeniería del software que deben presentarse conforme las organizaciones alcanzan diferentes niveles de capacidad y madurez del proceso. [6]

### **1.3.3. Personal está certificado en PPS**

El proceso personal del software por sus siglas PPS pone el énfasis en la medición personal tanto del producto del trabajo que se genera como de su calidad, así como también responsabiliza al profesional acerca de la planeación del proyecto y delega en el practicante el poder de controlar la calidad de todos los productos del trabajo de software que se desarrollen. [10]

### **1.3.4. Scrum**

Scrum es una de las metodologías ágiles más populares para gestión de proyectos de desarrollo de software y la que se ha optado por implantar en la mejora de la norma. Actualmente en el mercado se ofrecen certificaciones de acuerdo con los diferentes roles. [11]

### **1.3.5. Kanban**

Kanban, basada en una metodología que realiza el trabajo mediante el uso de table-ros con notas para dar seguimiento del progreso y cuellos de botella, su principal función es reducir el tiempo de fabricación en el mundo del software [12]

Existen niveles de certificación Kanban para su implementación. [13].

## **2. METODOLOGÍA**

La guía de observación fue creada después de analizar detenidamente varios aspectos que ayudaron a elegir indicadores específicos. Estos aspectos abarcan, entre otros, las prácticas más efectivas en la industria, la revisión de la literatura académica y las necesidades específicas identificadas al principio de la investigación. Para conocer el desarrollo de los procesos de la industria manufacturera en la Ciudad de San Francisco de Campeche, se crearon indicadores, los cuales se basaron en su importancia para medir y evaluar partes importantes de cómo las empresas gestionan sus procesos. Para elegir los indicadores finales, se llevó a cabo un proceso que incluyó revisión y validación por expertos en la industria. Además, siempre que fue posible, se obtuvo comentarios directos de las propias empresas observadas.

### **2.1 Diseño de instrumento**

Los indicadores que permiten conocer las características de la gestión de los procesos de la industria de software se definieron con la ayuda de una revisión sistemática.

El instrumento se originó a partir de una metodología que se presenta en el artículo “Propuesta de indicadores para evaluar la industria de software de una región”, donde Mex Alvarez, Manzanilla Yeh, Hernández Cruz, Cab Chan y Ortiz Cuevas, realizaron una revisión sistemática de la literatura de estudios secundarios previos en fuentes oficiales internacionales y nacionales. [5]. Los resultados de la revisión sistemática fueron un total 63 indicadores para medir los “Procesos”.

En la tabla 1 observamos los indicadores generados para medir la categoría de procesos.

ID	Indicador	ID	Indicador	ID	Indicador	ID	Indicador	ID	Indicador
1	Conocimientos con los que cuenta las empresas sobre MoProsoft.	4.2	Porcentaje de empresas certificadas en el nivel 4	7.4	Porcentaje de empresas que utilizan la verificación	8.4	Porcentaje de empresas que utilizan la gestión acuerdo de proveedor	11.1	Porcentaje de empresas con personal certificado en Scrum Product Owner
1.1	Porcentaje de las empresas que tienen personal capacitado en MoProsoft	4.3	Porcentaje de empresas certificadas en el nivel 3	7.5	Porcentaje de empresas que utilizan la validación	8.5	Porcentaje de empresas que utilizan la medición y análisis	11.2	Porcentaje de empresas con personal certificado en Advanced Certified Scrum Product Owner
1.2	Porcentaje de las empresas certificadas en MoProsoft	4.4	Porcentaje de empresas certificadas en el nivel 2	7.6	Porcentaje de empresas que utilizan el enfoque del proceso organizacional	8.6	Porcentaje de empresas que utilizan el proceso y garantía de calidad del producto	11.3	Porcentaje de empresas con personal certificado en Advanced scrum master
1.3	Porcentaje de las empresas que no tienen personal capacitado ni esta certificada en MoProsoft	4.5	Porcentaje de empresas certificadas en el nivel 1	7.7	Porcentaje de empresas que utilizan la definición del proceso organizacional	8.7	Porcentaje de empresas que utilizan la gestión de la configuración	11.4	Porcentaje de empresas con personal certificado en Scrum Master
2	Empresas que se encuentran certificadas en los niveles de CMMI	5	Aspectos de optimización para la mejora continua de sus procesos	7.8	Porcentaje de empresas que utilizan la capacitación en materia de organización	8.8	Porcentaje de las empresas que no hacen uso de los aspectos básicos para la Gestión de proyectos	11.5	Porcentaje de empresas con personal certificado en otros tipos
2.1	Porcentaje de empresas certificadas en el nivel 5	5.1	Porcentaje de empresas que utilizan la innovación en organización y despliegue	7.9	Porcentaje de empresas que utilizan la gestión integrada de proyectos	9	Porcentaje de las empresas que tienen certificado su personal en PSP	12	Porcentaje de las empresas que tienen certificado su personal en Kanban
2.2	Porcentaje de empresas certificadas en el nivel 4	5.2	Porcentaje de empresas que utilizan el análisis causal y resolución	7.10	Porcentaje de empresas que utilizan la gestión de riesgos	9.1	Niveles de certificación del personal de las empresas en PSP	13	Tipos de roles de Kanban con los que se encuentra certificado el personal de las empresas
2.3	Porcentaje de empresas certificadas en el nivel 3	5.3	Porcentaje de las empresas que no hacen uso de los aspectos de optimización	7.11	Porcentaje de empresas que utilizan el análisis de decisión y resolución	9.1	Porcentaje de las empresas que tienen certificado su personal en el nivel PSP0	13.1	Porcentaje de las empresas con personal certificado en Kanban Management Professional 1 (KMP1)
2.4	Porcentaje de empresas certificadas en el nivel 2	6	Aspectos de medición y control para la Gestión cuantitativa de los procesos	7.12	Porcentaje de empresas que utilizan equipos integrados	9.2	Porcentaje de las empresas que tienen certificado su personal en el nivel PSP0.1	13.2	Porcentaje de las empresas con personal certificado en Team Kanban Practitioner (TKP)
2.5	Porcentaje de empresas certificadas en el nivel 1	6.1	Porcentaje de empresas que utilizan el rendimiento del proceso organizacional	7.13	Porcentaje de empresas que utilizan Org. Medio ambiente para la integración	9.3	Porcentaje de las empresas que tienen certificado su personal en el nivel PSP1	13.3	Porcentaje de las empresas con personal certificado en Kanban Management Professional 2 (KMP2)
3	Conocimientos con los que cuenta las empresas sobre MoProsoft.	6.2	Porcentaje de empresas que utilizan la gestión de Proyectos cuantitativos	7.14	Porcentaje de empresas que utilizan la gestión de proveedores integrada	9.4	Porcentaje de las empresas que tienen certificado su personal en el nivel PSP1.1	13.4	Porcentaje de las empresas con personal certificado en otro tipo
3.1	Porcentaje de las empresas que tienen personal capacitado en CMMI	6.3	Porcentaje de las empresas que no hacen uso de los aspectos de medición y control	7.15	Porcentaje de las empresas que no hacen uso de los aspectos para la estandarización de sus procesos	9.5	Porcentaje de las empresas que tienen certificado su personal en el nivel PSP2		
3.2	Porcentaje de las empresas certificadas en CMMI	7	Porcentaje de empresas que estandarizan sus procesos	8	Aspectos básicos para la Gestión de proyectos que utilizan las empresas	9.6	Porcentaje de las empresas que tienen certificado su personal en el nivel PSP2.1		
3.3	Porcentaje de las empresas que no tienen personal capacitado ni esta certificada en CMMI	7.1	Porcentaje de empresas que utilizan el desarrollo de requisitos	8.1	Porcentaje de empresas que utilizan la administración de requisitos	9.7	Porcentaje de las empresas que tienen certificado su personal en el nivel TSP		
4	Empresas que se encuentran certificadas en los niveles de CMMI	7.2	Porcentaje de empresas que utilizan la solución técnica	8.2	Porcentaje de empresas que utilizan la planificación del proyecto	10	Porcentaje de las empresas que tienen certificado su personal en Scrum		
4.1	Porcentaje de empresas certificadas en el nivel 5	7.3	Porcentaje de empresas que utilizan la integración de productos	8.3	Porcentaje de empresas que utilizan el programa de monitoreo y control	11	Tipos de roles de Scrum con los que se encuentra certificado el personal de las empresas		

Tabla 1. Indicadores en la categoría de procesos.

Fuente: Mex Alvarez, 2021



## 2.2 Aplicación del Instrumento

Se elaboró un cuestionario con 14 preguntas de opción múltiple, para recoger los 63 indicadores, el cual se aplicó a las empresas desarrolladoras de software con domicilio fiscal en la Ciudad de San Francisco de Campeche. De acuerdo con un estudio realizado en marzo de 2022 se conoce que las empresas desarrolladoras con sede en San Francisco de Campeche son un total de 6, para los fines de este trabajo se obtuvieron los datos de 5 de ellas. [14]

## 3. RESULTADOS

De acuerdo con los resultados obtenidos en el cuestionario, respecto a los modelos de proceso, ninguna empresa en este momento está certificada por algún modelo de procesos que respalde la estandarización de su operatividad, como se presenta en la tabla 2.

Indicadores	MoProSoft	CMMI	Scrum	Kanban
El personal está capacitado	0,00%	0,00%	0,00%	0,00%
La empresa está certificada	0,00%	0,00%	0,00%	0,00%

Tabla 2. Modelos de procesos.

Puede observarse en la tabla 3 que el 80% de las empresas, es decir cuatro de cinco, se apega a la (OID), y el restante a (CAR). La tabla 4 nos presenta que una empresa emplea un aspecto del listado de aspectos de medición y control para la Gestión cuantitativa de los procesos, que es el Rendimiento del proceso organizacional; y las otras no realizan ese tipo de prácticas de manera cuantitativa.

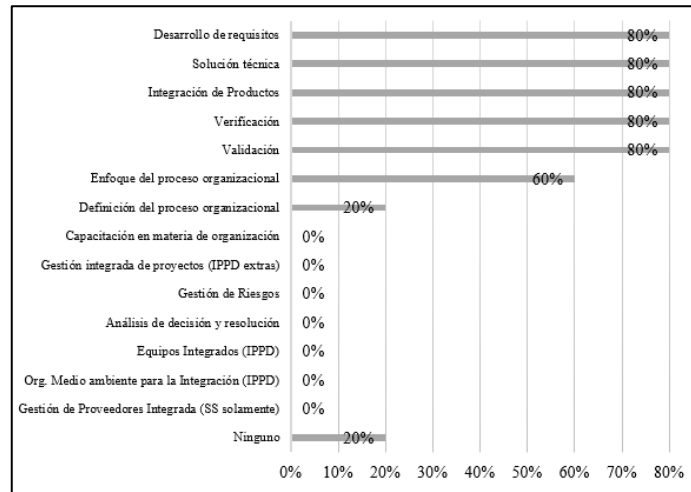
Indicadores	Porcentaje
Innovación en organización y despliegue (OID).	80,00%
Análisis Causal y Resolución (CAR).	20,00%
Ninguno	0,00%

Tabla 3. Aspectos de optimización para la mejora continua de sus procesos.

Indicadores	Porcentaje
Rendimiento del proceso organizacional	20,00%
Gestión de Proyectos cuantitativos	0,00%
Ninguna	80,00%

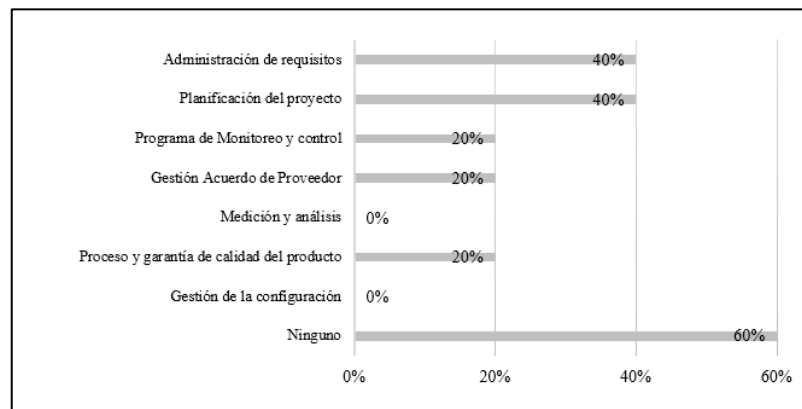
Tabla 4. Aspectos de medición y control para la Gestión cuantitativa de los procesos

En este mismo orden de ideas, en la gráfica 1, cuatro de cinco (80%) empresas desarrolladoras consideran cinco aspectos del listado de la estandarización de los procesos: Desarrollo de requisitos, Solución técnica, Integración de productos, Verificación y Validación. Tres de cinco (60%) empresas desarrolladoras consideran en su estandarización el Enfoque del Proceso Organizacional. Una de las cinco (20%) considera en la estandarización de sus procesos la Definición de procesos organizacionales. También es notorio que solo una empresa que representa el 20%, no emplea ningún tipo de aspecto de estandarización de procesos.



Gráfica 1. Aspectos para la estandarización de procesos

Con relación a los aspectos básicos para la gestión de proyectos, en la gráfica 2, podemos observar que tres de las cinco (60%) empresas no emplean ninguno de los aspectos de la lista, dos de las cinco (40%) emplea la Administración de requisitos y la Planificación del proyecto, y por lo menos una de las empresas encuestadas (20%) emplean el Programa de Monitoreo y Control, así como la Gestión Acuerdo de Proveedor y el Proceso y garantía de calidad del producto.



Gráfica 2. Aspectos básicos para la Gestión de proyectos

## 4. CONCLUSIONES

Las pequeñas y medianas empresas (pymes) objeto de estudio muestran un claro interés en la estandarización de sus procesos, destacando la valoración de prácticas como la validación, la verificación, la integración de productos, la solución técnica y el desarrollo de requisitos. Sin embargo, hasta la fecha de este estudio, ninguna de estas empresas ha obtenido una certificación o se encuentra en proceso de certificación según normativas reconocidas. Aunque una de las pymes afirmó haber logrado la certificación según la Norma Técnica NMX-059/01-NYCE-2005 del modelo de procesos MoProSoft en su nivel 2, gracias al programa gubernamental "México First". En estudios posteriores es conveniente investigar las razones del por qué las empresas no han apostado por las certificaciones de sus procesos y de su personal. A pesar de que ninguna empresa está certificada en Scrum, todas manifiestan fuera del cuestionario que es la metodología que emplean por la practicidad y las herramientas gratuitas disponibles.

Para mejorar su efectividad y optimizar el proceso a nivel de proyecto o de la organización, se obtuvo que cuatro de las empresas en Campeche optan por prácticas relacionadas con la Innovación y Despliegue Organizacional. La elección de Scrum se destaca, ya que les permite generar mejoras incrementales de manera natural, elevando así la capacidad de la organización. Es relevante mencionar que solo una de las empresas tiene planes y personal dedicado para llevar a cabo Análisis y Resolución Causal, identificando las causas de defectos y problemas.

Respecto a los aspectos de medición y control para la gestión cuantitativa de los procesos, de acuerdo con los resultados observamos que solo una de las empresas no tiene una evaluación cuantitativa. No obstante, se identificó que una empresa realiza un control de rendimiento del proceso organizacional, identificando brechas en el desempeño con respecto a los objetivos comerciales e implementando mejoras para cerrar estas brechas. El 80% de las empresas consideran en su estandarización 5 de 14 aspectos listados en la encuesta, y 7 de los 14 aspectos, que están más orientados al monitoreo y evaluación no están considerados. Esto nos obra un área de oportunidad para que las empresas desarrollen sus habilidades en la Gestión Integrada. Por otro lado, la gestión de proyectos es una actividad que la mitad de las empresas no ejecutan de manera formal, esto refuerza la importancia de incentivar el interés por elevar los estándares de los procesos internos, pero al mismo tiempo los resultados nos indican que es necesario reforzar las habilidades de la medición y análisis y gestión de la configuración.

## REFERENCIAS

- [1] J. J. Aguirre Ramírez, J. A. Castañeda Zamora, I. D. Hernández Umaña, A. L. Pérez Patiño y J. Robledo Velásquez. "Gestión de las capacidades de innovación tecnológica para la competitividad de las empresas antioqueñas de software" Medellín : Universidad Nacional de Colombia, 2011.
- [2] A. Pérez y L. González, "Estudio de la industria del software a nivel internacional, nacional y departamental" 2009 (accedido el 28/09/2022)
- [3] L. Merchan y A. Urrea, "Caracterización de las empresas pertenecientes a la industria emergente de software del sur occidente colombiano.", *Avances en sistemas e informática*, vol. 4, n.º 2, pp. 107–116, 2007.
- [4] C. F. Ortiz Paniagua y E. Arredondo Ortega, "Competitividad y factores de éxito en empresas desarrolladoras de software", *revista Venezolana de Información, Tecnología y Conocimiento*, vol. 11, n.º 3, 2014.
- [5] D. C. Mex Alvarez, E.J. Manzanilla Yeh, L.M. Hernández Cruz, J.R. Cab Chan & N.G. Ortiz Cuevas, 2021. "Propuesta de indicadores para evaluar la industria de software de una región | Revista Electrónica Sobre Cuerpos Académicos y Grupos de Investigación". <http://www.cagi.org.mx/index.php/CAGI/article/view/246> (accedido el 30/09/2022)
- [6] R. S. Pressman. *Ingeniería del software. Un enfoque práctico*. 7ma ed. México, D.F. McGRAW-HILL Interamericana Editores, 2010.
- [7] C. J. P. Escobar. (2010, Jul 05) Resumen de Análisis causal y resolución, [Internet] Available: <https://asprotech.blogspot.com/2010/07/resumen-de-analisis-causal-y-resolucion.html> (accedido el 30/09/2022).
- [8] C. J. P. Escobar. (2010, Jul 05) Resumen de Innovación y despliegue en la Organización, [Internet] <https://asprotech.blogspot.com/2010/07/resumen-de-innovacion-y-despliegue-en.html> (accedido el 30/09/2022).
- [9] M. A. Astorga Vargas, J. M. Olguin Espinoza y B. L. Flores Rios, "Caracterización de los factores de cambio para la implantación de MoProSoft en las PyMES", In Congreso Internacional de Ciencias Computacionales (CICOMP), 2006.
- [10] Fases/Niveles de PSP. PSP Proceso De Software Personal, [Internet] <https://ingenieriadestallerpsp.weebly.com/fasesniveles-de-psp.html> (accedido el 30/09/2022).
- [11] K. Schwaber y J. Sutherland. *La Guía de Scrum-La Guía Definitiva de Scrum: Las Reglas del Juego*, 2020.
- [12] D.J. Anderson, A. Carmichael. *Kanban Esencial Condensado*. 1ra ed. [Seattle, Washington. Mauvius Group Europe SL, 2017.
- [13] Método Kanban, un camino alternativo a la agilidad, [Internet] <https://scrum.mx/kanban> (accedido el 21/01/2024).
- [14] D. C. Mex Alvarez, Hernández Cruz, C.M. Llanes-Chiquini, Chan & C.A. Pérez Canul. "Procesos empresariales de la industria de software de la ciudad de San Francisco de Campeche". *Journal-Economic Development Technological Chance and Growth*", 2022.

## NOTAS BIOGRÁFICAS

	<p><b>Diana Concepción Mex Alvarez.</b> Ingeniera en Sistemas Computacionales, Maestra en Dirección de Ingeniería de Software y Maestra en Pedagogía, cuenta con Especialización en Entornos Virtuales de Aprendizaje y en Competencias Docentes para la Educación Media Superior. Doctorante en Proyectos en la línea de Investigación: Tecnologías de la Información y Comunicación (TIC). Ha impartido cátedra a Nivel Medio Superior y Superior en diversas instituciones públicas y privadas. Ostenta la distinción del perfil PRODEP otorgado por la secretaria de Educación Pública desde el año 2016 a la fecha. Es líder del Cuerpo Académico: UNACAM-CA-60 “Ciencias de la Computación”, reconocido por la Secretaría de Educación Pública, cultivando la línea de investigación: Formación de recursos humanos, desarrollo y aplicación de las Ciencias de la Computación en los sectores: primario, industrial, de servicios y educativo.</p>
	<p><b>Luz María Hernández Cruz.</b> Ingeniera en Sistemas Computacionales por la Universidad Autónoma de Campeche, grado de Máster en Gestión de Tecnologías de la Información por la Universidad del Mayab, Doctora en Proyectos con Línea de Investigación Tecnologías de la Información y las Comunicaciones en la Universidad Internacional Iberoamericana (UNINI). En el ámbito académico, ha impartido cátedra de Nivel Superior en la Universidad Interamericana del Norte, en el Instituto Tecnológico Superior de Hopelchén y en la Universidad Autónoma de Campeche. Ha impartido cátedra a Nivel Superior y Posgrado en diversas instituciones públicas y privadas. Actualmente es Profesor e Investigador de la Universidad Autónoma de Campeche. En el ámbito de la investigación científica posee el Reconocimiento de PRODEP y es miembro del Cuerpo Académico UNACAM-CA-60 Ciencias de la Computación. Líneas de generación y aplicación de conocimiento (LGAC).</p>
	<p><b>Charlotte Monserrat de Jesús Llanes Chiquini.</b> Es profesor e investigador Titular “B” de la Universidad Autónoma de Campeche, con una antigüedad de 32 años. Igualmente ha incursionado en el ámbito tanto público como privado. Doctorado en Ciencias de la Administración por la Universidad Nacional Autónoma de México (UNAM). Su Maestría en Psicología de la Educación es de la Universidad Autónoma de Campeche. Licenciatura en Informática (ITC). Seminario presencial en Beijing, China sobre China-Mexico Business Culture Exchanges en la Academia de Entrenamiento para los Funcionarios Económicos Internacionales (AIBO). Conferencista nacional e internacional. Experta en los temas de Comportamiento Organizacional, Inteligencia emocional, ODS para emprendedores. Ha publicado diversas ponencias para Congresos nacionales e internacionales, publicación de capítulos de libro y de artículos en revistas indexadas internacionales, así como, la publicación de su autoría del libro con ISBN “Maestro hoy, ¿Analfanauta digital mañana?” entre otras participaciones.</p>

	<p><b>Andrés Castro Villagrán.</b> Ingeniero industrial con maestría en ciencias de la educación. Es profesor e investigador en la División de Ingeniería en Gestión empresarial del Tecnológico Nacional de México/Instituto Tecnológico Superior de Champotón presidente de la Academia de Ingeniería en Gestión empresarial y Líder del Cuerpo Académico Innovación y tecnología para la educación y el desarrollo. Se especializa en temas de salud y seguridad en el trabajo, gestión empresarial y docencia. Miembro de la RED ICALC.</p>
	<p><b>Susana Friné Moguel Marín.</b> Cuenta con estudios de maestría y doctorado en el área de Educación. Ha participado en los procesos electorales federales del estado de Campeche como consejera electoral distrital y local. Es profesor investigador con más de 20 años de experiencia. Es co-autora del Libro Estrategias por Competencia en el Aula. Forma parte de diversas redes entre los que podemos mencionar: Red Nacional de Investigadores en Educación y Valores (REDUVAL), Red de Mujeres Líderes de las Américas. Actualmente es Integrante del Comité de Participación Ciudadana (CPC) del Sistema Anticorrupción del estado de Campeche y Miembro de la Red Nacional de CPC del Sistema Nacional Anticorrupción en las comisiones de Educación y Género en donde funge como presidenta.</p>
	<p><b>Carolina Chi Arceo.</b> Estudiante de la Licenciatura como Ingeniero en Sistemas Computacionales la Facultad de Ingeniería de la Universidad Autónoma de Campeche. Prestadora de servicio social en el Cuerpo Académico Ciencias de la Computación, UNACAM-CA-60 de la Facultad de Ingeniería de la Universidad Autónoma de Campeche. Colaboradora en el Proyecto " Aplicación de la Norma NMX-I-059-NYCE (MOPROSOFT)", con clave de registro 66-UAC-22</p>



Esta obra está bajo una licencia de Creative Commons Reconocimiento-NoComercial-CompartirIgual 2.5 México.

*Recibido 06 FEB. 2024*

*ReCIBE, Año 12 No.2, NOV. 2023*

*Aceptado 12 FEB. 2024*

## **Revisión de literatura de elementos de minería de procesos para la minería de procesos de software**

### **Literature review of process mining elements for software process mining**

Silvia Jaqueline Urrea Contreras<sup>1</sup>

Brenda L. Flores Rios<sup>1</sup>

brenda.flores@uabc.edu.mx

Angélica Astorga Vargas<sup>2</sup>

Jorge E. Ibarra-Esquer<sup>2</sup>

Félix F. González-Navarro<sup>1</sup>

Iván A. García Pacheco<sup>3</sup>

<sup>1</sup>Instituto de Ingeniería,

<sup>2</sup>Facultad de Ingeniería. Universidad Autónoma de Baja California, Baja California, México.

<sup>3</sup>División de Estudios de Posgrado, Universidad Tecnológica de la Mixteca, Oaxaca, México.

## **RESUMEN**

Para la aplicación de la Minería de Procesos, es fundamental la identificación de elementos clave del proceso, tales como eventos, casos, trazas. Estos elementos han sido formalmente definidos a través de notaciones matemáticas que incluyen las relaciones entre ellos y se requiere un conocimiento especializado para su aplicación en proyectos. Por medio de una semi-Revisión Sistemática de Literatura se analizaron las representaciones formales de los elementos y perspectivas de la Minería de Procesos. Como resultado se presenta un marco referencial que incluye definiciones, propiedades y aplicaciones, con el objetivo de brindar a los investigadores, profesionales y practicantes de la Ingeniería de Software conocimiento tópico de Minería de Procesos que sea de utilidad en proyectos de Minería de Procesos de Software.

## **PALABRAS CLAVE**

Minería de Procesos. Ingeniería de software. Semi-Revisión Sistemática de Literatura. Conocimiento tópico.

## **ABSTRACT**

For the application of Process Mining, the identification of key elements of the process, such as events, cases, traces, is essential. These elements have been formally defined through mathematical notations that include the relationships between them, requiring specialized knowledge for their application in projects. Through a semi-Systematic Literature Review, the formal representations of the elements and perspectives of Process Mining were analyzed. As a result, a reference framework is presented that includes definitions, properties, and applications, with the objective of providing researchers, professionals, and practitioners of Software Engineering with topical knowledge of Process Mining that is useful in Software Process Mining projects.

## **KEYWORDS**

Process Mining, Software Engineering, Semi-Systematic Literature Review, Topic Knowledge.



# 1. INTRODUCCIÓN

La Minería de Procesos establece un puente entre el conocimiento del dominio de las tecnologías de la información y las Ciencias Administrativas, contribuyendo significativamente tanto a la comunidad de Ciencias de la Computación como al Modelado de Procesos de Negocio (Marin-Castro & Tello-Leal, 2021). Por otro lado, la Minería de Procesos es una disciplina que aprovecha el volumen y la disponibilidad de los datos, así como la importancia de extraer conocimiento (Van der Aalst, 2016), por medio de diversos algoritmos (*Alpha*, heurístico, genético, difuso, entre otros) y técnicas (*Handover of work*, *Token replay*, Alineaciones), lo que resulta en mejoras y optimización de los procesos (Urrea-Contreras et al., 2021). Los algoritmos adoptan una formalización matemática en sus entradas (Leemans, van Zelst & Lu, 2023), lo que permite la definición de métricas y medidas de rendimiento para evaluar y comparar los procesos. Las técnicas pueden aplicarse a una variedad de procesos, incluyendo los procesos administrativos, médicos, de logística, productivos o incluso al proceso de desarrollo de software (Keith & Vega, 2016).

Actualmente, en las empresas de desarrollo de software se aplica la Minería de Procesos con el propósito de extraer el conocimiento almacenado en sus sistemas de gestión a partir de los datos generados durante la ejecución de las actividades. A esta aplicación se le denomina Minería de Procesos de Software (Rubin, Lomazova & Van der Aalst, 2014; Liu et al., 2016; Erdem, Demirörs & Rabhi, 2018). En este contexto, se están desarrollando y aplicando diversas técnicas, algoritmos y herramientas, tanto específicas como genéricas, para la extracción de registros de eventos (Dakic, 2020). El analizar los registros de eventos creados a partir de los datos del proceso de desarrollo de software permite a las empresas de desarrollo de software poseer una visión holística del ciclo de vida del proceso y brindarles información relacionada a las áreas de mejora en cada etapa para la optimización del flujo de trabajo completo.

Para la aplicación de la Minería de Procesos, es esencial identificar elementos clave del proceso, tales como eventos, casos, trazas, entre otros. Estos elementos han sido formalmente definidos a través de notaciones matemáticas que incluyen las relaciones entre ellos, como la relación entre eventos, casos específicos y actividades (Van der Aalst, 2016). Las notaciones establecen un lenguaje formal y estructurado, ofreciendo un conocimiento explícito para representar las actividades y sus flujos de trabajo facilitando la interpretación, el análisis y modelado de procesos.

Bajo una perspectiva de conocimiento, la principal aportación de este artículo es implementar la metodología Semi-Revisión Sistemática de Literatura (semi-SLR por sus siglas en inglés) para sintetizar la información y el conocimiento tópico existente sobre las definiciones formales de los elementos de la Minería de Procesos: evento, caso, traza, registros de eventos, red Petri y perspectivas. Por medio de esta síntesis, se desea establecer un marco referencial con el conocimiento semántico para su socialización e interiorización entre los roles que participan en la identificación de patrones, desviaciones y oportunidades de mejora, especialmente en los roles como analistas de negocio, responsable de mejora de procesos, gestor de proyectos pero que no cuentan con conocimientos de Ciencia de Datos, Ciencia de Procesos, Minería de Procesos, programación para el pre-procesamiento de datos, entre otro conocimiento especializado (Urrea-Contreras et al., 2022).

El documento se estructura de la siguiente manera: la sección 2 presenta un contexto de la Minería de Procesos de Software; la sección 3 especifica la metodología semi-SLR; en la sección 4 se interpretan y analizan las definiciones formales e importancia de los elementos de Minería de Procesos para contribuir en los procesos de software. Finalmente, la sección 5 muestra las discusiones y la sección 6 las conclusiones y trabajo futuro.

## 2. MINERÍA DE PROCESOS DE SOFTWARE

La rápida evolución de los sistemas de gestión está permitiendo controlar procesos empresariales, organizacionales, componentes, considerando recursos tecnológicos y humanos. En consecuencia, estos sistemas registran datos, por medio de mensajes, transacciones, registro de actividades, seguimiento de tareas, entre otras propiedades. Estos datos se extraen para construir registros de eventos y realizar tres tipos básicos de minería: el descubrimiento, la conformidad y la mejora de procesos (Dakic, 2020). De esta forma, los registros de eventos en la Minería de Procesos representan los datos de ejecución histórica de las actividades definidas por modelos de procesos almacenados como una tabla (Van der Aalst, 2016; Rivas & Bayona-Oré, 2019). No obstante, uno de los desafíos más significativos reside en la extracción e integración de los registros de eventos procedentes de múltiples fuentes de datos o repositorios.

En el trabajo relacionado se usa el término Minería de Procesos de Software (Rubin, Lomazova, & Van der Aalst, 2014; Liu et al., 2016; Erdem, Demirörs, & Rabhi, 2018) para referirse a la posibilidad de llevar a cabo diversas acciones. Por ejemplo: 1) Utilizar técnicas de descubrimiento de procesos para analizar el comportamiento real del proceso de desarrollo de software, 2) Aplicar técnicas de verificación de conformidad para detectar desviaciones en los procesos, 3) Identificar discrepancias en los procesos de software, obteniendo datos que pueden ser utilizados para planificar correcciones y mejoras futuras (Keith & Vega, 2016) y 4) Realizar predicciones en los problemas relacionados con el rendimiento o la conformidad a partir de la alineación de los modelos y el comportamiento real (Urrea-Contreras et al., 2022).

Se han documentado diversos ejemplos de la aplicación exitosa de la Minería de Procesos de Software, que proporcionan una visión más profunda de los desafíos y soluciones en este campo (Keith & Vega, 2016; Van der Aalst, 2015; Caldeira & e Abreu, 2016; Dong et al., 2017; Štolfa, 2018; Marques, da Silva, & Ferreira, 2018; Kneuper, 2018). Estos estudios muestran cómo las empresas de software enfrentan problemas concretos, como errores en el modelado de procesos y desviaciones en su aplicación, lo que conlleva costos significativos. Uno de los aspectos críticos resaltados en estos ejemplos es la disponibilidad de datos adecuados para el análisis, que con frecuencia se encuentran no estructurados y requieren un trabajo de pre-procesamiento.

Es relevante destacar que la Minería de Procesos de Software continúa siendo considerada como un campo de investigación en desarrollo (Rubin, Lomazova, & Van der Aalst, 2014; Erdem, Demirörs, & Rabhi, 2016) por lo que es importante seguir realizando investigaciones desde la representación formal, análisis y aplicación de técnicas, algoritmos y herramientas que apoyen a la innovación del descubrimiento de procesos de software, asociadas a las necesidades de conocimiento y la aplicación de estrategias.

## 3. METODOLOGÍA

Se implementó una Semi-Revisión Sistemática de Literatura (semi-SLR por sus siglas en inglés) propuesta por Snyder (Snyder, 2019) centrada en las notaciones matemáticas utilizadas en la Minería de Procesos. Una semi-SLR es un enfoque de investigación utilizado que permite contar con una visión general del estado actual del trabajo relacionado y examina cómo ha evolucionado el conocimiento dentro de uno varios campos o disciplinas a lo largo del tiempo (Snyder, 2019).

### 3.1 Planificación de la Revisión

Esta fase tiene como propósito recopilar, analizar, extraer y sintetizar la investigación sobre las notaciones formales y definiciones utilizadas en la Minería de Procesos para su integración en una guía técnica de MoProPEI-MP (Urrea-Contreras et al., 2018) y aplicabilidad en proyectos de Minería de procesos de software.

### *Pregunta de investigación*

La Pregunta de Investigación (PI) se centra en la importancia de la conceptualización (conocimiento tópico) de los elementos identificados por (Van der Aalst, 2016) para evitar ambigüedades y apoyar en su interpretación y comprensión. Se formuló de la siguiente manera:

PI: ¿Cuáles definiciones o representaciones formales existen sobre elementos de la Minería de procesos? donde el concepto de registros de eventos se basa en los siguientes supuestos (Kneuper, 2018):

- 1) Un registro de eventos contiene datos de varios procesos,
- 2) Un proceso (o su ejecución) consta de casos o instancias individuales y
- 3) Un caso consta de diferentes eventos, donde cada evento pertenece exactamente a un caso.

### *Selección de fuentes y estrategias de búsqueda*

Para abordar la PI, se seleccionaron tres fuentes de datos para la búsqueda de la literatura relevante: *ACM Digital Library*, *Scopus* y *Springer*. Estas fuentes se eligieron estratégicamente debido a su amplio alcance en el campo de estudio y su capacidad para proporcionar un acceso completo a una variedad de estudios académicos, revistas y conferencias. La combinación de estas fuentes permitió realizar una búsqueda de la literatura existente relacionada con la investigación.

### *Cadena de búsqueda*

Después de establecer las fuentes, se procedió a identificar el siguiente conjunto de términos clave para la búsqueda: *Process mining*, *Software process mining*, *Process mining representation*, *Process mining elements*, *Process mining formalisms*, *Process mining analysis*, *Process mining perspectives*, *Discovery*, *Conformance*, *Enhancement*. Los términos claves se ajustaron mediante el uso de operadores lógicos para adaptarlos a las particularidades de cada una de las fuentes.

### *Criterios de inclusión y exclusión*

Con el fin de distinguir entre los estudios relacionados con el tema y aquellos que aportan información pertinente, se han establecido criterios de inclusión y exclusión. Los criterios de inclusión se refieren a las características específicas que deben cumplir los estudios para ser considerados relevantes para la investigación. Por otro lado, los criterios de exclusión consisten en una serie de características que permiten descartar aquellos documentos que no contribuyan de manera significativa al objetivo de la investigación.

Los criterios definidos fueron:

Criterios de inclusión: 1) Estudios en idioma inglés y español; 2) Estudios a partir del año 2013; 3) Estudios que contengan información relacionada a las notaciones matemáticas de la Minería de Procesos; 4) Estudios publicados en revistas y memorias de congresos especializados.

Criterios de exclusión: 1) Estudios repetidos (duplicados); 2) Estudios a los que no se tenga acceso; 3) Estudios que no cumplan los criterios de inclusión.

## **3.2 Realización de Revisión**

Se realizó una búsqueda global de todos los artículos que contenían los términos antes mencionados en sus títulos y resumen. La semi-SLR reveló un conjunto de 1104 artículos, distribuidos de la siguiente manera: *ACM Digital Library* (930 artículos), *Scopus* (89 artículos) y *Springer* (85 artículos). Se procedió a la exclusión de aquellos que no estaban directamente relacionados con el objetivo del estudio actual, centrándose específicamente en la identificación de formalismos y el uso de notaciones matemáticas en la Minería de Procesos.

Posteriormente, a los artículos identificados se les aplicó la técnica de *Snowball Sampling*

(Leemans, van Zelst & Lu, 2023; Parker, Scott, & Geddes, 2019), lo que permitió verificar que las representaciones matemáticas que presentaban algunos artículos eran basados en referencias. Esta técnica se aplicó de manera iterativa, para consultar las referencias de los artículos y se eligieron aquellos que fueron pertinentes para este estudio. En total, se obtuvo una muestra de 13 artículos que exploran el tema de notaciones matemáticas en el contexto de la Minería de Procesos.

## 4. INTERPRETACIÓN DE RESULTADOS

Para conocer el conocimiento semántico del significado y representación de las relaciones entre los distintos elementos de la Minería de Procesos, se compararon sus definiciones y representaciones formales. De esta manera, el enfoque consistió en analizar cómo se conciben y se expresan matemáticamente estas relaciones dentro del ámbito de la Minería de Procesos.

### 4.1 Universos

Se establecen universos conceptuales con el propósito de proporcionar un contexto para las definiciones formales, permitiendo una comprensión estructurada de los elementos de la Minería de Procesos con el propósito de contextualizar las definiciones formales.

Definición 1 (Universos). Se utilizan las siguientes notaciones para los universos:  
 $\epsilon$  denota el universo de eventos únicos; es decir, el conjunto de todos los posibles identificadores que se refieren a eventos concretos,  
A denota el conjunto de todos los posibles nombres de atributos,  
V denota el conjunto de todos los posibles valores de atributos,  
C denota el universo de casos.

### 4.2 Evento

Un proceso describe la ejecución del conjunto de actividades. Cada actividad ejecutada da lugar a uno o más eventos que registran información que describe la instancia del proceso, la actividad ejecutada, el tiempo de ejecución y datos adicionales como los atributos y valores involucrados (Van Eck, 2022).

**Tabla 1.** Comparativa de definiciones y representaciones formales para evento

Cita	Definición y representación formal
Van Eck, 2022	<i>Un evento <math>e \in \varepsilon</math> es una tupla de <math>n</math> pares nombre-pares de valores. Es decir, <math>e = ((a_1, v_1), (a_2, v_2), \dots, (a_n, v_n))</math> con atributos <math>a_1, a_2, \dots, a_n \in Attr</math>, <math>\forall 1 \leq i &lt; j \leq n : a_i \neq a_j</math>, y valores <math>v_1, v_2, \dots, v_n \in Val</math>.</i>
De Leoni, Van der Aalst, & Dees, 2016	<i>Un evento <math>e</math> es una asignación de valores a características, es decir, <math>e \in C \rightarrow U</math>. En el resto <math>\varepsilon = C \rightarrow U</math> es el universo de eventos.</i>
Diamantini, Genga, & Potena, 2016	<i>Un evento <math>\sigma</math> es una instancia de una especificación de entidad, y se identifica por un identificador único. Denotaremos por <math>\odot</math> el universo de eventos, es decir, el conjunto de todos los posibles identificadores de eventos. Cada evento se describe mediante un conjunto de atributos. Sea <math>A</math> el conjunto de nombres de atributos; para cualquier evento <math>\sigma \in \odot</math> y atributo <math>\alpha \in A</math>, <math>\# \alpha(\sigma)</math> es el valor del atributo <math>\alpha</math> para el evento <math>\sigma</math>. Cada evento se describe al menos con un nombre de atributo de etiquetado, tal que <math>\#nombre(\sigma) = (\#recurso(\sigma), \#actividad(\sigma)) \in ES</math>, y por un atributo timestamp que identifica el momento en que se produjo el suceso.</i>
Senderovich, 2017	<i>Sea <math>\varepsilon</math> el conjunto de todos los eventos posibles, es decir, identificadores únicos de eventos. Sea <math>\varepsilon^*</math> el conjunto de todas las secuencias finitas sobre <math>\varepsilon</math> siendo <math>\epsilon \in \varepsilon^*</math> la traza vacía. Los eventos se asocian con atributos, por ejemplo, marcas de tiempo, actividades, ubicaciones y recursos. Denotamos por <math>A_\Sigma</math> el conjunto de todos los espacios de atributos de eventos: <math>A_\Sigma = \{A_i \mid i \in I\}</math>, siendo <math>A_i</math> el espacio de atributos <math>i_{th}</math> y <math>I</math> es el conjunto de índices de atributos de eventos.</i>

### 4.3 Caso

Los casos, como los eventos, pueden tener atributos y por cada caso se puede registrar una traza de eventos (Marin-Castro & Tello-Leal, 2021). Los casos al ser representaciones de instancias de la actividad están ligadas a poseer atributos y características únicas que pueden pasar por diferentes actividades durante la ejecución (Tabla 2).

**Tabla 2.** Comparativa de definiciones y representaciones formales para caso

Cita	Definición y representación formal
Senderovich, 2017	<i>El conjunto de casos es <math>C \subseteq \varepsilon^*</math>, es decir, un conjunto de secuencias finitas de eventos. Exigimos que cada evento aparezca como máximo una vez en algún caso. Un caso <math>c = \langle c_1, \dots, c_n \rangle \in C</math> es, por tanto, una secuencia finita de sucesos tal que <math>c_i \in \varepsilon</math>, <math>i = 1, \dots, n</math>, siendo <math>n</math> el número de sucesos por caso.</i>
Hompes et al., 2015	<i>Sea <math>C</math> el universo de casos, es decir, el conjunto de todos los posibles identificadores de casos. <math>n(c)</math> es el valor del atributo <math>n \in N</math> para el caso <math>c \in C</math> (<math>n(c) = \perp</math> si <math>c</math> no tiene ningún atributo llamado <math>n</math>).</i>
Seeliger, Nolle, & Mühlhäuser, 2018	<i>Sea <math>C</math> el conjunto de todos los posibles identificadores de casos. Los casos también pueden tener atributos, así que para cada caso <math>c \in C</math> y un atributo <math>a \in A</math>, definimos <math>\#_a(c)</math> como el valor de un atributo <math>a</math> para el caso <math>c</math>.</i>
Jokonowo et al., 2019	<i>Sea <math>C</math> el caso. Un caso son las actividades del ciclo de vida del proceso para cualquier evento <math>c \in C</math>.</i>

#### 4.4 Traza

Los eventos de un caso se representan en forma de traza, es decir, una secuencia de únicos eventos (Van der Aalst, 2016). Estas trazas constituyen la entrada tradicional de muchos enfoques de descubrimiento de procesos y otras técnicas de minería de procesos (Van Eck, 2022). Las trazas capturan el seguimiento de las actividades realizadas y las decisiones tomadas en el proceso (Tabla 3).

**Tabla 3.** Comparativa de definiciones y representaciones formales para traza

Cita	Definición y representación formal
Van Eck, 2022	<i>Una traza <math>\sigma = \langle e_1, e_2, \dots, e_n \rangle \in \varepsilon^*</math> es una secuencia finita de <math>n</math> eventos ordenados en el tiempo que han sido registrados para una instancia de proceso específica. Para todos los <math>e_i, e_j \in \sigma</math> con <math>1 \leq i &lt; j \leq n</math>, se cumple que (1) <math>\#_{time}(e_i) \leq \#_{time}(e_j)</math>, (2) <math>\#_{pi}(e_i) = \#_{pi}(e_j)</math> y que (3) <math>e_i \neq e_j</math>.</i>
De Leoni, Van der Aalst, & Dees, 2016	<i>Una traza <math>t \in \varepsilon^*</math> es una secuencia de eventos. Sea <math>T \in \varepsilon^*</math> el universo de las trazas.</i>
Diamantini, Genga, & Potena, 2016	<i>Una traza <math>l</math> es una secuencia finita de eventos, <math>l = \langle \sigma_1, \sigma_2, \dots, \sigma_n \rangle</math>, <math>\sigma_i \in \Sigma</math>, <math>\#_{timestamp}(\sigma_i) &lt; \#_{timestamp}(\sigma_j)</math>, <math>i &lt; j</math>.</i>
Hompes et al., 2015	<i>Una traza es una secuencia finita de eventos <math>\sigma \in \varepsilon^*</math>, tal que cada evento aparece sólo una vez, es decir, para <math>1 \leq i \leq j \leq  \sigma </math>: <math>\sigma(i) \neq \sigma(j)</math>.</i>

## 4.5 Registro de eventos

Un registro de eventos almacena datos sobre la ocurrencia de actividades que fueron registradas por los sistemas de información mientras soportan la ejecución de un proceso. Cada ejecución de una instancia de proceso da lugar a una secuencia de eventos (Van Eck, 2022; De Leoni, Van der Aalst & Dees, 2016). La mayoría de los registros de eventos almacenan información adicional (Van der Aalst, 2016) que dependen de las técnicas de Minería de Procesos y las preguntas en cuestión que se necesiten responder con su análisis, relacionadas con la conformidad y el rendimiento. La Tabla 4 presenta una comparativa de definiciones de este término.

**Tabla 4.** Comparativa de definiciones y representaciones formales de registro de eventos

Cita	Definición y representación formal
Van Eck, 2022	<i>Un registro de eventos <math>L = \{\sigma_1, \dots, \sigma_n\} \subset \varepsilon^*</math> es un conjunto finito de <math>n</math> trazas. Para todos los <math>\sigma, \sigma' \in L</math> con dos eventos cualesquiera <math>e \in \sigma, e' \in \sigma'</math> se cumple que <math>\#_{pi}(e) \neq \#_{pi}(e')</math>.</i>
De Leoni, Van der Aalst, & Dees, 2016	<i>Un registro de eventos <math>L</math> es un conjunto múltiple de trazas, es decir <math>L \in (T)</math>.</i>
Diamantini, Genga, & Potena, 2016	<i>Un registro de eventos <math>L</math> es el registro de una colección de trazas. Cada evento en una traza (registro de eventos) es único, es decir, <math>\forall \sigma_i, \sigma_j \in L, i \neq j \Rightarrow \sigma_i \neq \sigma_j</math>.</i>
Hompes et al., 2015	<i>Un registro de eventos es un conjunto de casos <math>L \subseteq C</math> tal que cada evento aparece como máximo una vez en todo el registro, es decir, para cualquier <math>c, c' \in L</math> tal que <math>c \neq c'</math>: <math>set(\hat{c}) \cap set(\hat{c}') = \emptyset</math>.</i>

## 4.6 Red Petri

Los modelos de procesos, como técnicas de descubrimiento de procesos, permiten representar el comportamiento de un proceso o sistema, por medio de varias notaciones de modelado o representaciones gráficas (Van der Aalst, 2016, Van Eck, 2022). De esta forma, los modelos de procesos apoyan la descripción del orden de ejecución de las actividades y las opciones que se pueden realizar, también conocido como flujo de control.

**Tabla 5.** Comparativa de definiciones y representaciones formales de registro de red Petri

Cita	Definición y representación formal
Li, 2019	<p>Una red Petri es una tupla <math>N = (P, T, F)</math> donde <math>P</math> es un conjunto finito de lugares, <math>T</math> es un conjunto finito de transiciones tal que <math>P \cap T = \emptyset</math>, y <math>F \subseteq (P \times T) \cup (T \times P)</math> es un conjunto de arcos dirigidos, llamados relaciones de flujo. Una red Petri marcada es un par <math>(N, M)</math>, donde <math>N = (P, T, F)</math> es una red de Petri y <math>M \in \mathbb{B}(P)</math> es un subconjunto sobre <math>P</math> que denota el marcado de la red.</p>
Leemans, 2018	<p>Sea <math>A</math> el alfabeto de actividades y sea <math>\tau</math> una etiqueta especial (silenciosa) tal que <math>\tau \notin A</math>. Una red Petri es una tupla <math>PN = (P, T, F, \ell)</math>, donde:</p> <ul style="list-style-type: none"> <li>- <math>P</math> es un conjunto finito de lugares</li> <li>- <math>T</math> es un conjunto finito de transiciones, tal que <math>P \cap T = \emptyset</math></li> <li>- <math>F</math> es un subconjunto finito de arcos dirigidos: <math>F \in \mathcal{B}((P \times T) \cup (T \times P))</math></li> </ul> <p><math>\ell</math> es una función de etiquetado de transiciones: <math>\ell: T \rightarrow A \cup \{\tau\}</math></p>
Kalenkova et al., 2016	<p>Una red Petri es una tupla <math>PN = (P, T, F)</math> con <math>P</math> el conjunto de lugares, <math>T</math> el conjunto de transiciones, <math>P \cap T = \emptyset</math>, y <math>F \subseteq (P \times T) \cup (T \times P)</math> la relación de flujo</p>
Van Zelst, 2019	<p>Sea <math>P</math> un conjunto de lugares y sea <math>T</math> un conjunto de transiciones s.t. <math>P \cap T = \emptyset</math>. Sea <math>F = (P \times T) \cup (T \times P)</math> la relación de flujo. Además, dejemos que <math>\Sigma</math> denote el universo de etiquetas, que <math>\tau \notin \Sigma</math> y que <math>\lambda: T \rightarrow \Sigma \cup \{\tau\}</math> denota la función de etiquetado de las transiciones. Una red de Petri <math>N</math>, es una tupla <math>N = (P, T, F, \lambda)</math>.</p>
Syamsiyah, 2020	<p>Una red de Petri es una tupla <math>\mathcal{N} = (P, T, F, \ell)</math> donde:</p> <ul style="list-style-type: none"> <li>- <math>P</math> es un conjunto de lugares</li> <li>- <math>T</math> es un conjunto de transiciones, <math>P \cap T = \emptyset</math></li> <li>- <math>F \subseteq (T \times P) \cup (P \times T)</math> es una relación de flujo</li> </ul> <p><math>\ell: T \rightarrow U\_A \cup \{\tau\}</math> es una función de etiquetado.</p>

## 4.7 Aplicabilidad de las definiciones formales

Representación en lenguaje formal: Los formalismos proporcionan un lenguaje formal y una notación matemática estandarizada que los autores muestran para una comunicación clara y precisa entre los expertos en Minería de procesos. Al tener un lenguaje común, los profesionales pueden compartir y discutir sus ideas, hallazgos y resultados de manera más efectiva, lo que facilita la colaboración y el intercambio de conocimientos en el campo. La notación matemática utilizada en los formalismos identificados muestra varias ventajas. En primer lugar, brinda un nivel de dominio y exactitud que no siempre se puede lograr con descripciones en lenguaje natural. La notación matemática permite una representación más rigurosa de los conceptos (por lo regular los artículos seleccionados contemplaban una sección de conceptos preliminares) y relaciones dentro de los procesos, lo que evita ambigüedades en la interpretación de los resultados. Otra ventaja de la notación matemática es su capacidad para simplificar y generalizar conceptos complejos. Al representar los procesos de manera abstracta y simbólica, se pueden identificar patrones y regularidades que pueden pasar desapercibidos en las descripciones verbales o visuales.



Esto permite una comprensión más profunda de los procesos y la capacidad de aplicar resultados y conocimientos a diferentes contextos y dominios.

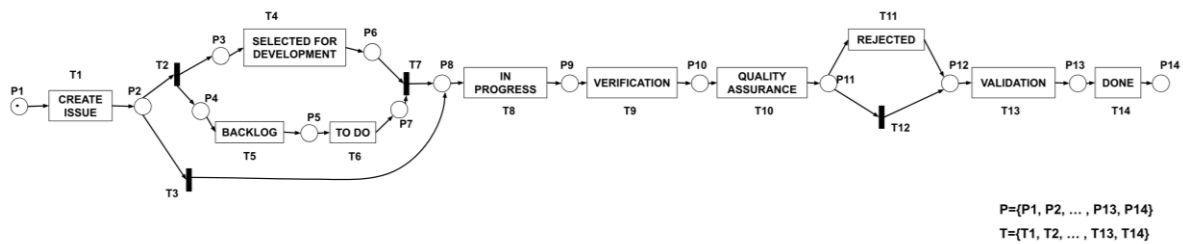
**Algoritmos y técnicas:** Además de permitir la representación de los procesos, los formalismos en la Minería de procesos desempeñan un papel crucial en la aplicación de algoritmos y técnicas para analizar los datos de los procesos y extraer conocimiento que permite la mejora y optimización de los procesos (Urrea-Contreras et al., 2021). Los formalismos proporcionan una estructura y un marco de referencia para aplicar estas técnicas de análisis de datos. Por ejemplo, mediante el uso de redes de Petri, se pueden aplicar algoritmos de descubrimiento de procesos que permiten identificar patrones recurrentes, desviaciones, cuellos de botella y oportunidades de mejora en los procesos. Además, los formalismos permiten la definición de métricas y medidas de rendimiento que se utilizan en la evaluación y comparación de los procesos. Estas métricas pueden ser utilizadas para identificar áreas problemáticas, evaluar la eficiencia y efectividad de los procesos, y tomar decisiones basadas en datos para la optimización y reingeniería de los mismos.

**Identificación de atributos:** Las representaciones matemáticas muestran los diferentes atributos relevantes en el análisis de los procesos. Al utilizar fórmulas matemáticas, es posible expresar de manera precisa las relaciones entre los atributos, lo que facilita la identificación de patrones y tendencias entre ellas. Las representaciones matemáticas proporcionan un marco para la identificación y el análisis de los atributos en la Minería de Procesos, lo que contribuye a una comprensión más profunda y precisa de los procesos analizados. Tal es el caso de la definición de perspectiva por (Hompes et al., 2015) (Tabla 6), la cual contempla los atributos y la proyección de múltiples perspectivas. Por otro lado, más en lo particular se encuentra la descripción de la perspectiva de flujo de control por (De Leoni, Van der Aalst, & Dees, 2016), donde la descripción de esta perspectiva se realiza basada en las notaciones de manipulación de traza, exponiendo definiciones relacionadas al número de ejecuciones de actividad, primera ocurrencia de actividad, actividad previa en la traza y actividad actual.

**Tabla 6.** Definición y representación formal de perspectiva

Cita	Definición y representación formal
Hompes et al., 2015	<p><i>Sea <math>P</math> una perspectiva. <math>\uparrow_P : C \rightarrow R^m</math> denota la función que asigna un caso a un vector real de longitud <math>m</math> según la perspectiva <math>P</math>. <math>m</math> es el número de atributos en <math>P</math>, por ejemplo, el número de recursos diferentes en el registro. <math>c/P</math> denota la proyección del caso <math>c \in L</math> a una perspectiva <math>P</math>. Además que <math>c/\{P_1, P_2, \dots, P_K\} = c[\uparrow_{P_1}]_1 \    \ (c[\uparrow_{P_2}] \   ) \dots \    \ \uparrow_{P_K}</math>, es decir, se concatenan los vectores de perfil resultantes de la proyección a múltiples perspectivas.</i></p>

Por otro lado, el modelar un proceso de software por medio de una red Petri permite conocer el flujo de trabajo, las actividades, transiciones y condiciones que pueden presentar. En la Fig. 1, se observa como en la red Petri las posiciones (P) se representan como círculos y se numeran de P1 al P14 para indicar su identificación única. Las transiciones (T) se representan como barras verticales y también se numeran del T1 al T14. Mediante flechas dirigidas, se indican las relaciones de incidencia entre las posiciones y las transiciones. Una flecha que va de una posición a una transición indica que la transición consume recursos de esa posición (por ejemplo de P2 a T2 o T3), mientras que una flecha que va de una transición a una posición indica que la transición genera o produce recursos en esa posición (por ejemplo de T10 a P11). Esta representación permite visualizar las interacciones entre las posiciones y las transiciones, así como sus conjuntos, proporcionando una comprensión clara de cómo se produce el flujo de recursos en el sistema.



**Figura 1.** Red Petri de los estados de transición de un proceso de software

Se observa como existen flujos paralelos entre las posiciones P3 y P4 que dependen de los eventos que describen el inicio de la actividad y su terminación entre P6 y P7 respectivamente. El analizar en conjunto y aplicar las definiciones de caso y traza se pueden institucionalizar mejores prácticas y/o enfoques para la gestión de proyectos de software, la asignación de recursos pero sobre todo, la toma de decisiones basadas en diagnósticos, la obtención de nuevos modelos o los datos existentes y generados por las empresas de desarrollo de software.

## 5. DISCUSIÓN DE RESULTADOS

Tomando las descripciones anteriores y las definiciones basadas en conjuntos de los elementos de la Minería de Procesos (de la Tabla 1 a la Tabla 5), se establecieron las siguientes definiciones:

**Definición 2 (Evento).** Un evento  $e$  es una asignación de valores, el cual se describe mediante un conjunto de atributos. Sea  $e \in \varepsilon$  para cualquier evento,  $a \in A$  para los atributos y  $v \in V$  para los valores, denotamos que,  $e = ((a_1, v_1), (a_2, v_2), \dots, (a_n, v_n))$ , en donde  $a \subseteq v$ .

**Definición 3 (Caso).** El conjunto de casos es  $C \subseteq \varepsilon^*$ , es decir, un conjunto de secuencias finitas de eventos. Cada evento aparece como máximo una vez en algún caso. Un caso  $c = \langle c_1, c_2, \dots, c_n \rangle \in C$ . Los casos también pueden tener atributos, así que para cada caso  $c$ , un atributo  $a \in A$ , definimos  $c = \langle (c_1, a_1), (c_2, a_2), \dots, (c_n, a_n) \rangle$ .

**Definición 4 (Traza).** Una traza puede ser vista como un caso, secuencia finita de eventos. Sea  $t$  una traza y  $\varepsilon^*$  el conjunto de todas las secuencias finitas sobre  $\varepsilon$ ,  $t \in \varepsilon^*$  por lo que  $t = \langle e_1, e_2, \dots, e_n \rangle$ , donde  $n$  es el número de eventos por traza.

**Definición 5 (Registro de eventos).** Un registro de eventos  $L$  es una secuencia finita de trazas donde  $L = (t_1, t_2, \dots, t_n)$ .

**Definición 6 (Petri net).** Una red de Petri es una tupla  $N = (P, T, F, \ell)$  donde:  
 $P$  es un conjunto de lugares (posiciones).  
 $T$  es un conjunto de transiciones,  $P \cap T = \emptyset$  (transiciones, conjuntos finitos).  
 $F \subseteq (T \times P) \cup (P \times T)$  es una relación de flujo (conjunto de arcos dirigidos).  
 $\ell: T \mapsto \mathbb{A} \cup \{\tau\}$  es una función de etiquetado.

De acuerdo a la definición de los elementos de Minería de Procesos y sus relaciones al extraer los datos de los repositorios se sugiere el siguiente orden: identificar los eventos (Definición 2) agrupándolos en sus correspondientes casos (Definición 3) para crear un registro de eventos (Definición 5). Una vez definido el registro de eventos, se genera un modelo basado en los datos extraídos que puede ser representado con una red Petri (Definición 6) para posteriormente verificar la conformidad de los eventos que es representado por medio de las trazas (Definición 4). El modelo podría extenderse con las perspectivas de Minería de Procesos con la identificación de los atributos correspondientes (Tabla 7). De esta manera, la Tabla 7 presenta la relación de atributos que se utilizan en el proceso de minado de las perspectivas de Minería de Procesos. Estos atributos son variables que se extraen y analizan para obtener información relevante sobre el flujo, el tiempo y la relación organizacional de los procesos. Asimismo, la Tabla 7 muestra una lista de los atributos relacionados a las perspectivas de flujo de control, organizacional, del caso y del tiempo, junto con ejemplificaciones de atributos como el orden de actividades, *timestamp* y caso. Estos atributos pueden incluir características como la duración de una actividad, el estado de un evento, el rol de un participante, la secuencia de actividades, entre otros.

La relación de atributos en la Tabla 7 proporcionaría una guía para los investigadores y profesionales de Ingeniería de Software que realizan minado de procesos, ayudándoles a identificar los atributos relevantes que deben considerar en su análisis. Asimismo, esta relación de atributos sirve como un recurso útil para comprender y aplicar las perspectivas de Minería de Procesos en diferentes contextos y proyectos.

Es importante destacar que los atributos específicos y su relevancia pueden variar dependiendo del dominio de aplicación y los objetivos del minado. Por lo tanto, esta tabla brindaría una visión general de los atributos comunes utilizados en el minado de perspectivas, pero se recomienda adaptarla y ajustarla según las necesidades y particularidades de cada proyecto.

**Tabla 7.** Atributos para el minado de perspectivas de Minería de Procesos de software

<b>Flujo de Control</b>	<b>Organizacional</b>	<b>Caso</b>	<b>Tiempo</b>
Actividad	ID Actividad	Caso***	<i>Timestamp*</i>
<i>Orden Actividades*</i>	Actividades (Tipo)		Horas invertidas
	<i>Timestamp*</i>		Horas efectivas
	Horas invertidas		
	<i>Recursos**</i>		
<i>Orden Actividades*</i> Timestamp (Inicio/Fin) <ul style="list-style-type: none"> <li>- Fecha DD/MM/AA</li> <li>- Hora HH:MM</li> </ul>			
	<i>Recursos**</i> Roles Estructuras organizacionales <ul style="list-style-type: none"> <li>- Departamentos</li> <li>- Equipos/proyectos</li> </ul>		
		Caso*** El atributo de caso dependerá del análisis	

## CONCLUSIONES

Actualmente, la comunidad de Ingeniería de Software desea desarrollar estrategias basadas en conocimiento y orientadas al análisis de datos, Inteligencia Artificial, la Minería de Procesos de Software, entre otras, aplicables a las diversas problemáticas del desarrollo de proyectos con el propósito de fortalecer o alcanzar altos niveles de capacidad de procesos. En este documento, se presentan los formalismos de los elementos de la Minería de Procesos junto a su representación matemática, proporcionando una representación semántica de los conceptos y relaciones dentro de los procesos, evitando posibles confusiones derivadas del lenguaje natural. Las representaciones matemáticas del marco referencial también permiten identificar patrones que pueden pasar desapercibidos en las descripciones verbales o visuales, para su comprensión e implementación en la mejora de procesos de software. Como trabajo futuro, se realizará una sesión de *Focus Group* para la socialización y validación del marco referencial obtenido.

## AGRADECIMIENTOS

El primer autor agradece al Consejo Nacional de Humanidades, Ciencias y Tecnologías (CONAHCyT) por el financiamiento brindado bajo la referencia CVU-841596.

## REFERENCIAS

Caldeira, J., & e Abreu, F. B. (2016). Software development process mining: Discovery, conformance checking and enhancement. In 2016 10th International Conference on the Quality of Information and Communications Technology (pp. 254-259). IEEE.

Dakic, D., Stefanovic, D., Lolic, T., Narandzic, D., & Simeunovic, N. (2020). Event log extraction for the purpose of process mining: a systematic literature review. In Innovation in Sustainable Management and Entrepreneurship: 2019 International Symposium in Management (SIM 2019) (pp. 299-312). Springer International Publishing.

De Leoni, M., Van der Aalst, W. M., & Dees, M. (2016). A general process mining framework for correlating, predicting and clustering dynamic behavior based on event logs. *Information Systems*, 56, 235-257.

Diamantini, C., Genga, L., & Potena, D. (2016). Behavioral process mining for unstructured processes. *Journal of Intelligent Information Systems*. 47 (1), 5-32.

Dong, L., Liu, B., Li, Z., Wu, O., Babar M. A., & Xue, B. (2017). A mapping study on mining software process. *Software Engineering Conference (APSEC)* (pp. 51-60). IEEE.

Erdem, S., Demirörs, O., & Rabhi, F. (2018). Systematic mapping study on process mining in agile software development. In *Software Process Improvement and Capability Determination: 18th International Conference (SPICE 2018)* (pp. 289-299). Springer International Publishing. [https://doi.org/10.1007/978-3-030-00623-5\\_20](https://doi.org/10.1007/978-3-030-00623-5_20)

Hompes, B., Buijs, J. C. A. M., Van der Aalst, W. M., P. Dixit, & Buurman, J. (2015). Discovering deviating cases and process variants using trace clustering. In *Proceedings of the 27th Benelux Conference on Artificial Intelligence (BNAIC)*, 5 (6).

Jokonowo, B., Sarno, R. Rochimah, S., & Priambodo, B. (2019). Process mining: Measuring key performance indicator container dwell time. *Indones. J. Electr. Eng. Comput. Sci*, 16 (1), 401-411.

- Kalenkova, A. A., Van der Aalst, W. M., Lomazova, I. A., & Rubin, V. A. (2016). Process mining using BPMN: relating event logs and process models. In Proceedings of the ACM/IEEE 19th International Conference on Model Driven Engineering Languages and Systems (pp. 123-123). IEEE.
- Keith B., & Vega, V. (2016). Process mining applications in software engineering, In Trends and Applications in Software Engineering: Proceedings of CIMPS 2016 (pp. 47-56). [https://doi: 10.1007/978-3-319-48523-2\\_5](https://doi.org/10.1007/978-3-319-48523-2_5)
- Kneuper, R. (2018). Selected Current Trends in Software Processes. In Software Processes and Life Cycle Models, (pp. 303-326). Springer.
- Leemans, M. (2018). Hierarchical Process Mining for Scalable Software Analysis. Ph.D. dissertation, Eindhoven University of Technology.
- Leemans, S. J., van Zelst, S. J., & Lu, X. (2023). Partial-order-based process mining: a survey and Outlook. Knowledge and Information Systems, 65 (1), 1-29.
- Li, G. (2019). Process mining based on object-centric behavioral constraint (OCBC) models. Ph.D. Thesis, Eindhoven University of Technology, Mathematics and Computer Science, Eindhoven, The Netherlands.
- Liu, C., Van Dongen, B., Assy N., & Van der Aalst, W. M. (2016). Component behavior discovery from software execution data. In 2016 IEEE Symposium Series on Computational Intelligence (SSCI) (pp. 1-8). IEEE. [https://doi: 10.1109/SSCI.2016.7849947](https://doi.org/10.1109/SSCI.2016.7849947)
- Marin-Castro, H. M., & Tello-Leal, E. (2021). Event log preprocessing for process mining: a review. Applied Sciences, 11 (22), 10556. <https://doi.org/10.3390/app112210556>
- Marques, R., da Silva, M. M., & Ferreira, D. R. (2018). Assessing agile software development processes with process mining: A case study. In 2018 IEEE 20th Conference on Business Informatics (CBI) (pp. 109-118). IEEE Press.
- Parker, C., Scott, S., & Geddes, A. (2019). Snowball sampling. SAGE research methods foundations.
- Rivas, M. H., & Bayona-Oré, S. (2019). Algoritmos de minería de proceso para el descubrimiento automático de procesos. Revista Ibérica de Sistemas e Tecnologías de Informação (RISTI), 31, 33-49.
- Rubin, V., Lomazova, I., & Van der Aalst, W. M. (2014). Agile development with software process mining, In Proceedings of the 2014 International conference on software and system process (pp. 70-74). <https://doi.org/10.1145/2600821.2600842>
- Seeliger, A., Nolle, T., & Mühlhäuser, M. (2018). Process explorer: an interactive visual recommendation system for process mining. In KDD Workshop on Interactive Data Exploration and Analytics.
- Senderovich, A. (2017). Queue Mining: Service Perspectives in Process Mining. In BPM (Demos).
- Snyder, H. (2019). Literature review as a research methodology: An overview and guidelines. Journal of business research, 104, 333-339.
- Štolfa, J. (2018). Application of Process Mining in Intelligent Process Support. Ph.D. Thesis.

- Syamsiyah, A. (2020). In-database preprocessing for process mining. Technische Universiteit Eindhoven.
- Urrea-Contreras, S. J., Flores-Rios, B. L., Astorga-Vargas, M. A., & Ibarra-Esquer, J. E. (2021). Process Mining Perspectives in Software Engineering: A Systematic Literature Review. In 2021 Mexican International Conference on Computer Science (ENC) (pp. 1-8). IEEE Press. [https://doi: 10.1109/ENC53357.2021.9534824](https://doi.org/10.1109/ENC53357.2021.9534824)
- Urrea-Contreras, S. J., Flores-Rios, B. L., Astorga-Vargas, M. A., Ibarra-Esquer, J. E., & González-Navarro, F. F. (2018). MoProPEI-MP an Adaptation in the Selection of Process Mining Tools for Software Processes. Jornadas Iberoamericanas de Ingeniería de Software e Ingeniería del Conocimiento 2018 (JIISIC) (pp. 151-162).
- Urrea-Contreras, S. J., Flores-Rios, B. L., Astorga-Vargas, M. A., Ibarra-Esquer, J. E., González-Navarro, F. F., García Pacheco, I. A., & Pacheco Agüero, C. L. (2022). Process Mining Model Integrated with Control Flow, Case, Organizational and Time Perspectives in a Software Development Project. International Conference in Software Engineering Research and Innovation (CONISOFT) (pp. 92-101). IEEE. [https://doi: 10.1109/CONISOFT55708.2022.00022](https://doi.org/10.1109/CONISOFT55708.2022.00022)
- Van der Aalst, W. M. (2015). Extracting event data from databases to unleash process mining, In BPM-Driving innovation in a digital world. (pp. 105-128). Springer.
- Van der Aalst, W. M. (2016). Data science in action. In Process mining, Springer, Berlin, Heidelberg.
- Van Eck, M. L. (2022). Process Mining for Smart Product Design. Eindhoven University of Technology.
- Van Zelst, S. J. (2019). Process mining with streaming data. Technische Universiteit Eindhoven.

## NOTAS BIOGRÁFICAS

**Silvia Jaqueline Urrea Contreras** Doctora en Ciencias en el área de la Computación por la Universidad Autónoma de Baja California. Sus áreas de investigación son Minería de procesos aplicada a la Ingeniería de software y Mejora de procesos de software. Cuenta con experiencia profesional como miembro de equipos de proyectos de software.

**Brenda Leticia Flores Rios** Doctora en Ciencias en Computación por UABC. Actualmente, es Coordinadora de Investigación y Posgrado del Instituto de Ingeniería UABC. Miembro del Cuerpo Académico de Cómputo Científico nivel consolidado, de la Academia Mexicana de Computación y de la Red Mexicana de Ingeniería de software. Cuenta con el reconocimiento PRODEP-SEP y SNI nivel I. Sus áreas de investigación son Ingeniería de software, Mejora de procesos de software e Ingeniería del conocimiento.

**María Angélica Astorga Vargas** Doctora en Ciencias en Computación por la Universidad Autónoma de Baja California. Profesora-investigadora del Programa Educativo de Licenciado en Sistemas Computacionales en la Facultad de Ingeniería UABC. Miembro del Cuerpo Académico de Cómputo Científico y de la Academia Mexicana de Computación. Cuenta con el reconocimiento al Perfil Deseable PRODEP-SEP y SNI nivel I. Sus áreas de investigación son la Ingeniería de software y la mejora de procesos de software con enfoque en la conformación efectiva de equipos de proyectos de software.

**Jorge Eduardo Ibarra Esquer** es profesor investigador adscrito al programa educativo Ingeniero en Computación y al programa de Maestría y Doctorado en Ciencias e Ingeniería en la Facultad de Ingeniería de la UABC. Cuenta con el reconocimiento como Investigador Nivel I y con el Reconocimiento a Perfil Deseable que otorga el Programa PRODEP-SEP. Entre sus áreas de interés se encuentran el Internet de las Cosas, la Minería de Datos, y la aplicación de esta última al descubrimiento de conocimiento.

**Félix Fernando González-Navarro** Doctor por la Universidad Politécnica de Cataluña, España. Cuenta con el reconocimiento como Investigador Nivel I y con el Reconocimiento a Perfil Deseable PRODEP-SEP. Es líder del Cuerpo Académico de Cómputo Científico nivel consolidado, miembro de la Academia Mexicana de Computación. Las líneas de investigación son Aprendizaje Máquina y Minería de Datos.

**Iván A. Gacía Pacheco** cursó los estudios de posgrado en la Universidad Politécnica de Madrid, España. Actualmente se desempeña como profesor-investigador en la División de Estudios de Posgrado de la Universidad Tecnológica de la Mixteca. Es miembro del SNI nivel 1. Los intereses son la Ingeniería de Software, específicamente los métodos de evaluación y mejora del proceso software aplicados a las pequeñas empresas, el desarrollo de herramientas educativas para mejorar la formación de los futuros ingenieros de software, y el desarrollo de herramientas computacionales para reducir el esfuerzo y maximizar el beneficio económico de los pequeños entornos de desarrollo.



*Recibido 27 OCT. 2023*

*ReCIBE, Año 12 No.2, NOV. 2023*

*Aceptado 29 NOV. 2023*

## **Algorithms and Metrics in Effort Estimation and Impact on DevOps Project Management: A Systematic Mapping of the Literature**

## **Algoritmos y métricas en la estimación del esfuerzo y su impacto en la gestión de proyectos DevOps: un mapeo sistemático de la literatura**

Iliana Lizbeth Alvarado Lara<sup>1</sup>  
d23ce054@cenidet.tecnm.mx  
Noé Alejandro Castro Sánchez<sup>1</sup>  
Blanca Dina Valenzuela Robles<sup>1</sup>  
René Santaolaya Salgado<sup>1</sup>  
Gabriel González Serna<sup>1</sup>

<sup>1</sup>*Ciencias Computacionales* Tecnológico Nacional de México/CENIDET Cuernavaca, Morelos, México



**Resumen** — En proyectos de desarrollo de software, la estimación del esfuerzo desempeña un papel crítico en el éxito o fracaso de éstos, ya que está intrínsecamente ligada al alcance, el costo y el tiempo. A pesar de numerosos esfuerzos por mejorar la precisión de las estimaciones, la mayoría de la literatura existente se centra en proyectos de desarrollo de software tradicionales, dejando un vacío en cuanto a la estimación del esfuerzo para la parte operativa en proyectos DevOps. Esta investigación se enfoca en identificar los algoritmos/métodos, métricas y procesos de validación que actualmente se han implementado para la estimación del esfuerzo. Dando como resultado diversas técnicas, incluyendo algoritmos basados en aprendizaje automático, estimación del tamaño de software, listas de verificación, técnicas de estimación ágil y marcos de trabajo. Estas técnicas han evaluado la precisión de las estimaciones utilizando una variedad de métricas, tales como la media de la magnitud del error relativo, mediana de la magnitud del error relativo, predicción en el nivel n, precisión estandarizada, error medio absoluto, error cuadrático, residuo medio absoluto, magnitud del error relativo, mediana del sesgo del error relativo equilibrado, error relativo, error absoluto, error cuadrático relativo raíz, error cuadrático medio raíz, precisión, error cuadrático medio, coeficiente de variación, media y desviación estándar. Estos estudios se han validado mediante encuestas, casos de estudio, historias de usuario, entrevistas, factores y proyectos extraídos de repositorios de datos como Cocomo81, Nasa 93, Maxwell, China y ISBSG. Los resultados de esta investigación tienen el potencial de aportar nuevos enfoques para mejorar el proceso de estimación en proyectos DevOps, llenando un vacío importante en la literatura actual sobre este tema.

**Palabras clave** — Estimación del esfuerzo, gestión de proyectos, DevOps, alcance, costo, tiempo.

**Abstract** — In software development projects, effort estimation is critical in their success or failure, as it is intrinsically linked to scope, cost, and time. Despite numerous efforts to improve estimation accuracy, most existing literature focuses on traditional software development projects, leaving a gap in effort estimation for the operational aspect of DevOps projects. This research aims to identify the algorithms/methods, metrics, and validation processes currently implemented for effort estimation, resulting in various techniques, including machine learning-based algorithms, software size estimation, checklists, agile estimation techniques, and frameworks. These techniques have assessed estimation accuracy using a variety of metrics such as mean relative magnitude error, median relative magnitude error, prediction at n level, standardized accuracy, mean absolute error, squared error, mean absolute residual, relative magnitude error, median balanced relative error bias, relative error, absolute error, square root relative squared error, square root mean squared error, accuracy, mean squared error, coefficient of variation, mean, and standard deviation. These studies have been validated through surveys, case studies, user stories, interviews, factors, and projects extracted from data repositories such as Cocomo81, Nasa 93, Maxwell, China, and ISBSG. The results of this research have the potential to provide new approaches to enhance the estimation process in DevOps projects, filling a significant gap in the current literature on this topic.

**Keywords** — Effort estimation, project management, DevOps, scope, cost, time.

# 1. INTRODUCCIÓN

La estimación, según McConnell, (2006), consiste en proyectar la duración y/o el costo de un proyecto. Esta definición adquiere importancia en el ámbito del desarrollo de software, donde la comunidad científica ha dedicado más de tres décadas a perfeccionar estas estimaciones; su relevancia es respaldada en el informe del "CHAOS Report 2020" (*The Standish Group - View News*, s. f.).

En Grimstad et al., (2006), Pincirolí, (2023) y en la Guía del PMBOK, (2017) se enfatiza que las estimaciones del esfuerzo son problemas graves en la industria del software. Cuando son demasiado pesimistas pueden perderse oportunidades de negocio, mientras que un exceso de optimismo puede ir seguido de pérdidas económicas significativas. Algunos autores argumentan que existen elementos que impactan y aumentan la complejidad de los proyectos, entre ellos se encuentran el alcance, costo y tiempo, también conocidos como la triple restricción y que dichos elementos forman parte de la estimación del esfuerzo para el desarrollo de software. Por lo tanto, es necesario definir al menos las referencias del proyecto en cuanto a los elementos mencionados, de modo que la ejecución del proyecto pueda ser medido y comparado con esas referencias y que se pueda gestionar el desempeño.

A través de los años se han propuesto una serie de cuestiones fundamentales que son cruciales para mejorar las estimaciones del esfuerzo de proyectos ágiles/DevOps. Es de suma importancia integrar cada uno de los diversos enfoques de solución, factores, métricas de precisión, algoritmos/métodos, valores de precisión y procesos de validación propuestos con el fin de comprender y mejorar el proceso de estimaciones del esfuerzo de proyectos ágiles/DevOps (Valenzuela et al., 2023).

Con el objetivo de abordar estas cuestiones fundamentales y proporcionar una comprensión integral de la estimación del esfuerzo en el contexto de proyectos ágiles/DevOps, el presente artículo sigue la siguiente estructura: en la segunda sección, se presenta un mapeo sistemático de la literatura sobre la estimación del esfuerzo en proyectos de software ágiles/DevOps, seguido por la tercera sección que presenta las respuestas a las preguntas de investigación enunciadas en la sección 2.2 de esta investigación y la cuarta sección presenta las conclusiones derivadas de ésta.

## 2. OBJETIVO Y MÉTODO DE LA INVESTIGACIÓN

### 2.1. OBJETIVO DE LA INVESTIGACIÓN

Esta investigación tiene como objetivo proporcionar una comprensión integral de la estimación del esfuerzo en el desarrollo de software ágil/DevOps y establecer una base sólida para proponer nuevos enfoques que mejoren este proceso crucial en el área de software.

### 2.2. MÉTODOLÓGIA DE LA INVESTIGACIÓN

En este trabajo de investigación se siguió el mapeo sistemático propuesto por Kitchenham, el cual está diseñado para dar una visión general de un área de investigación mediante la clasificación y recuento de contribuciones en relación con las categorías de dicha clasificación (Kitchenham & Charters, 2007). Es importante destacar que un mapeo y una revisión bibliográfica sistemática comparten algunos puntos en común por ejemplo en la selección de estudios (Petersen et al., 2015). La Figura 1 muestra las actividades que se realizaron durante el proceso de esta investigación siguiendo las pautas sugeridas por Kitchenham.

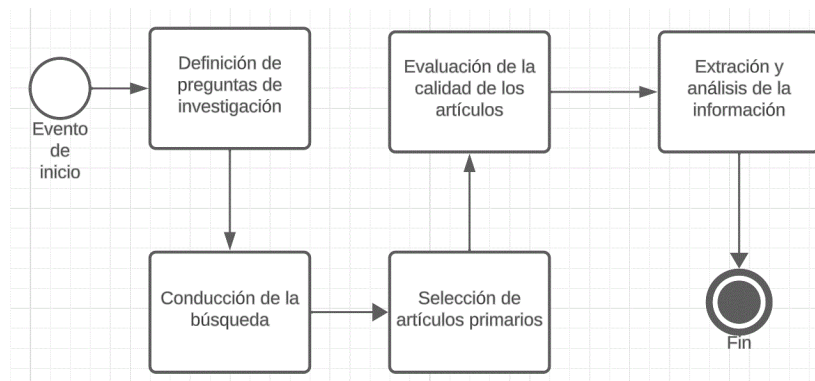


Figura 1. Proceso para el mapeo sistemático adaptado de [7].

### 2.2.1. Definición de preguntas de investigación

Se definieron tres preguntas de investigación para abordar el objetivo principal del estudio.

- PI1: ¿Qué algoritmos/métodos se han implementado para la estimación del esfuerzo en proyectos de software?
- PI2: ¿Qué métricas se han implementado para la estimación del esfuerzo en proyectos de software y qué precisión presentan los algoritmos implementados en la estimación del esfuerzo en proyectos de software?
- PI3: ¿Qué proceso de validación se ha implementado para cada algoritmo/método identificado?

### 2.2.2. Conducción de la búsqueda

El proceso de búsqueda de los estudios relevantes fue llevado a cabo mediante la siguiente cadena de búsqueda: (Devops OR Development OR Operation) AND ("triple constraint" OR "Effort estimation" OR "Predicting Effort" OR "Estimating Effort"). Esta cadena fue adaptada y ejecutada en Scopus y Google Académico. Para la realización de esta investigación, en la búsqueda se consideraron artículos dentro de una ventana de tiempo de seis años, es decir, de enero del 2018 a mayo del 2023. Se implementó el diagrama de PRISMA (Page et al., 2021), para representar el flujo del proceso de filtrado de artículos a través de las diferentes fases de una revisión sistemática y/o mapeo, la cual indica la cantidad de registros identificados, incluidos y excluidos, tal como se muestra en la Figura 2. La búsqueda fue realizada en las bases de datos de Google Académico y Scopus, lo que permitió encontrar las técnicas o métodos que se han utilizado para estimar el esfuerzo en el desarrollo de software con base en metodologías ágiles. Se obtuvieron 900 artículos para su revisión, de los cuales 810 son provenientes de Google Académico y 90 artículos de Scopus. En la primera fase se excluyeron 16 artículos que no cumplían con los criterios iniciales. En la siguiente fase se seleccionaron 848 artículos para la revisión de títulos y resúmenes de los cuales se excluyeron un total de 796 artículos. Del resto de artículos se hizo una revisión completa y de estos 52 artículos, 19 fueron excluidos por ser estudios secundarios. Finalmente, se incluyeron solamente 33 artículos que cumplieron con los criterios de inclusión y que responden a las preguntas de investigación.

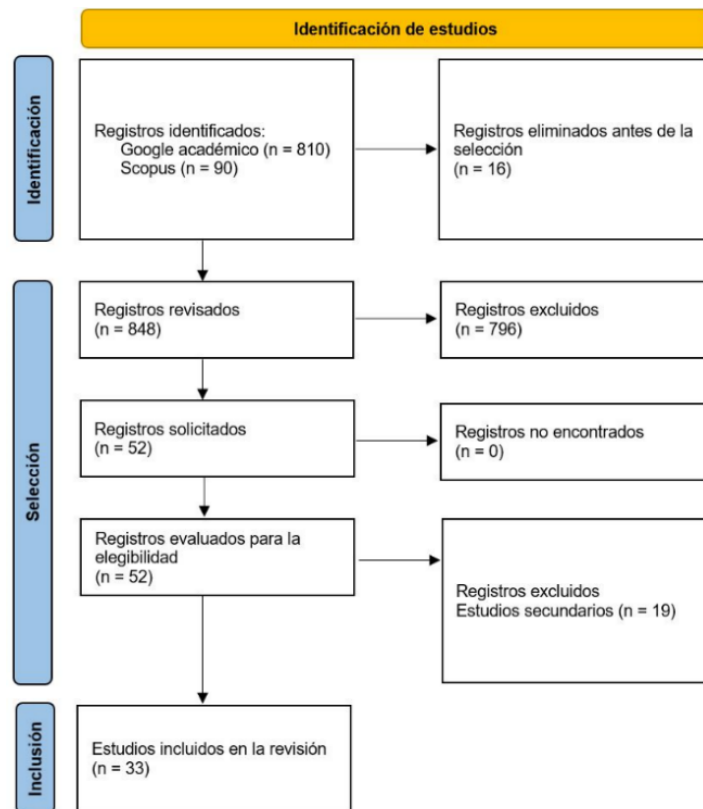


Figura 2. Flujograma del proceso de filtrado de artículos basado en PRISMA.

### 2.2.3. Selección de estudios primarios

Una vez obtenidos los estudios primarios potencialmente relevantes, se evalúa su pertinencia real. La **Figura 3. Criterios de inclusión y exclusión** muestra los criterios de inclusión y exclusión evaluados para recuperar la lista final.

*Figura 3. Criterios de inclusión y exclusión*

<b>Criterios de inclusión</b>	<b>Criterios de exclusión</b>
Publicaciones por año: 2018-2023	Tipo de trabajo: libros, capítulos de libros, tesis, material didáctico.
Idioma: inglés	Tipo de búsqueda: revisión de la literatura, reportes, encuestas, estudios comparativos.
Título o Resumen: Contiene la cadena de búsqueda	Opiniones o artículos de debate.
Tipo de literatura: Estudios primarios	Artículos disponibles sólo en presentaciones o en resúmenes. Artículos no relevantes para las preguntas de investigación.

### 2.2.4. Evaluación de la calidad de los artículos

La selección de artículos se basó en una revisión de cuatro niveles:

- CI1: Revisión del título
- CI2: Revisión del resumen, introducción y conclusiones
- CI3: El texto completo del artículo debe estar disponible.
- CI4: Revisión sobre el texto completo para determinar si el estudio cumple con los criterios de inclusión.

Así mismo, se listan los criterios de evaluación que permiten validar la calidad de los estudios seleccionados:

- a) El artículo se enfoca en investigar factores, técnicas, estándares, buenas prácticas, modelos predictivos, y/o métodos para estimar el esfuerzo para proyectos de software ágil y/o DevOps.
- b) El artículo ofrece una descripción clara del problema de investigación.
- c) El artículo sigue un proceso de investigación estructurado y fundamentado.
- d) El artículo expone de manera clara y detallada los resultados obtenidos.
- e) El artículo ha sido publicado en una revista, conferencia o congreso relevante.
- f) El artículo ha sido citado por otros autores.
- g) El artículo describe claramente trabajos futuros o alternativas de investigación

La Figura 4 muestra el listado de artículos que fueron seleccionados a partir de los criterios de inclusión, exclusión y calidad para su análisis en esta investigación. Dichos estudios abarcan desde técnicas de aprendizaje automático hasta enfoques basados en métodos tradicionales de estimación, lo que refleja la diversidad de enfoques utilizados en la comunidad de desarrollo de software.

*Figura 4. Artículos relacionados con la estimación de esfuerzo en el desarrollo de proyectos de Software*

<b>Artículo</b>	<b>Referencia</b>	<b>Artículo</b>
1	Abusaeed et al., (2023)	A fuzzy AHP-based approach for prioritization of cost overhead factors in agile software development
2	Adnan et al, (2019)	Ontology-oriented software effort estimation system for e-commerce applications based on extreme programming and scrum methodologies
3	Alshammari, (2022)	Cost estimate in scrum project with the decision-based effort estimation technique
4	Angara et al, (2020)	DevOps Project Management Tools for Sprint Planning, Estimation and Execution Maturity
5	Angara et al, 2018	Towards benchmarking user stories estimation with COSMIC function points: A case example of participant observation

6	Bhaskaran et al 2019	A hybrid effort estimation technique for agile software development (HEETAD)
7	Bilgaiyan et al, (2019)	Effort estimation in agile software development using experimental validation of neural network models
8	Butt et al, (2022)	A software-based cost estimation technique in scrum using a developer's expertise
9	Choetkietikul et al, 2019	A deep learning model for estimating story points
10	Dantas et al, (2019)	An effort estimation support tool for agile software development: an empirical evaluation
11	Dave, (2021)	An efficient framework for cost and effort estimation of scrum projects
12	Gandomani et al, 2019	Planning poker in cost estimation in agile methods: averaging vs. consensus
13	Hacaloglu & Demirors, 2019	Measureability of functional size in agile software projects: Multiple case studies with COSMIC FSM
14	Kaushik et al., (2020)	A comparative analysis on effort estimation for agile and non-agile software projects using DBN-ALO
15	Khuat & Le, 2018	A novel hybrid ABC-PSO algorithm for effort estimation of software projects using agile methodologies
16	Mallidi & Sharma, (2021)	Study on agile story point estimation techniques and challenges
17	Meedeniya & Thennakoon, (2021)	Impact factors and best practices to improve effort estimation strategies and practices in DevOps
18	Prasada Rao et al., 2018	An agile effort estimation based on story points using machine learning techniques
19	Premalatha & Srikrishna, (2019)	Effort estimation in agile software development using evolutionary cost- sensitive deep belief network
20	Prykhodko & Prykhodko 2019	A multiple non-linear regression model to estimate the agile testing efforts for small web projects
21	Radu, (2019)	Effort prediction in agile software development with bayesian networks
22	Raslan & Ramadan, (2018)	An enhanced framework for effort estimation of agile projects
23	Rola & Kuchta, (2019)	Application of fuzzy sets to the expert estimation of scrum-based projects
24	Scott & Pfahl, 2018	Using developers' features to estimate story points
25	Sharma & Chaudhary, (2020)	Linear regression model for agile software development effort estimation
26	Sharma & Chaudhary, (2021)	Analysis of software effort estimation based on story point and lines of code using machine learning
27	Tanveer et al, (2019)	An evaluation of effort estimation supported by change impact analysis in agile software development
28	Tripathi, (2021)	Evaluation of the feasibility of parametric estimation in DevOps continuous planning
29	Usman et al., (2018)	Developing and using checklists to improve software effort estimation: a multi-case study
30	Vera-Rivera et al 2020	Automatic generation of sprint planning in agile software development, assignment of user stories to developers using genetic algorithms
31	Vetrò et al., (2018)	Combining data analytics with team feedback to improve the estimation process in agile software development
32	Vyas & Hemrajani, (2021)	Predicting effort of agile software projects using linear regression, ridge regression and logistic regression
33	Zakrani et al, 2018	Support vector regression based on grid-search method for agile software effort prediction

## 2.2.5.Extracción y análisis de la información

En la Figura 5 se muestra de manera detallada la clasificación conforme a los métodos y/o algoritmos utilizados en cada estudio seleccionado en el marco de esta investigación, así como las métricas implementadas para medir la precisión del método y/o algoritmo. Así mismo, se observa en la columna cuatro el resultado de la aplicación de dichas métricas para medir la precisión de las estimaciones realizadas con los diferentes algoritmos o métodos utilizados; esto ofrece una visión general de cómo se desempeñan las diferentes estrategias para la estimación del esfuerzo en el desarrollo de proyectos de software. Finalmente, en la columna cinco se muestran los procesos de validación que se han utilizado para cada algoritmo o método identificado, destacando que las diversas estrategias utilizadas abarcan desde proyectos, encuestas, casos de estudio, entrevistas hasta conjuntos de datos.

Figura 5. Algoritmos, métricas, precisión y proceso de validación identificados.

<b>Estudio</b>	<b>Algoritmo/método</b>	<b>Métrica</b>	<b>Precisión</b>	<b>Proceso de validación</b>	
(Zakrani et al., 2018)	(SVR-RBF) Support Vector Regression with RBF kernel	MMRE	0.062	Conjunto de datos (21 proyectos)	
		MdMRE	0.0426		
		Pred	(0.25%) 100		
(SVM-RBF-GS) SVR-RBF optimized by Grid Search		MMRE	0.164		
		MdMRE	0.0893		
		Pred	(0.25%) 80.952		
(Angara et al., 2018)	COSMIC-FP	No reportado	No reportado	Encuestas y conjunto de datos (49 encuestas 11 proyectos)	
(Hacaloğlu & Demirörs, 2019)	COSMIC FSM	No reportado	No reportado	Conjunto de datos (4 proyectos)	
(Scott & Pfahl, 2018)	SVM (Support Vector Machines)	SA	46.334 - 93.152	(Conjunto de datos) 8 proyectos	
(Khuat & Le, 2018)	ABC (artificial bee colony)	MAE	0.366 - 3.453	(Conjunto de datos) 21 proyectos	
		R2	0.9734		
		MAR	3.12		
		MMRE	5.69		
		MdMRE	3.33		
		Pred	(0.08%) 66.67		
	PSO (particle swarm optimization)		R2	0.9732	
			MAR	3.15	
			MMRE	5.84	
			MdMRE	5.18	
	ABC-PSO (integración de algoritmos)		Pred	(0.08%) 61.9	
			R2	0.9626	
MAR			3.66		
MMRE			6.69		
(Prasada Rao et al., 2018)	Adaptive Neuro-Fuzzy Modeling	MdMRE	7.14	Conjunto de datos (2 proyectos)	
		Pred	(0.08%) 61.9		
		MMRE	3.9587		
		Costo	19.5922		
		Tiempo	5.8700		
		Costo	3.2042		
	Generalized Regression Neural Network		Costo	16.2868	
			Tiempo	6.6430	
			MMRE	3.9079	
			Costo	8.4277	
			Tiempo	4.8335	
			Costo	2.7864	
	Radial Basis Function Networks		Costo	9.9604	
			Tiempo	8.0909	
			MMRE	57.1429	
			Pred	76.1905	
			Costo	76.905	
			Tiempo	76.1905	
Planning Poker		Costo	76.1905	Conjunto de datos (29 proyectos)	
		RE	0.09		
		Media	0.03		
		Consenso	0.03		
		AE			

<b>Estudio</b>	<b>Algoritmo/método</b>	<b>Métrica</b>	<b>Precisión</b>	<b>Proceso de validación</b>
		Media	42.5	
		Consenso	16.5	
(Choetkietikul et al., 2019)	Deep learning: long short-term memory (LSTM) and recurrent highway network (RHN)	SA MAE	9.16 - 69.62 0.66 - 10.90	Conjunto de datos (16 proyectos)
(Vera-Rivera et al., 2020)	Algoritmos genéticos	No reportado	No reportado	2 casos de estudio
(Bhaskaran et al., 2019)	Puntos de función e historias de usuario	No reportado	No reportado	No reportado
(Prykhodko & Prykhodko, 2019)	Regresión lineal	R2 MMRE Pred	0.9847 0.0565 (0.25) 1.0	Conjunto de datos (40 proyectos)
	Regresión no lineal múltiple	R2 MMRE Pred	0.9818 0.0443 (0.25) 1.0	
(Dantas et al., 2019)	Aprendizaje automático: árboles de decisión	MRE	10.12 - 58.12	26 backlogs con 24 profesionales
(Tanveer et al., 2019)	Gradient Boosted Trees (GBT). Planning Poker	MAE MMRE Pred	0.88 1.03 (0.25) 0.5	345 historias de usuario
	Juicio de expertos	MAE MMRE Pred	0.72 1.55 (0.25) 0.5 0.22 0.5 (0.25) 75	
(Rola & Kuchta, 2019)	Números difusos	No reportado	No reportado	Conjunto de datos (2 proyectos)
(Bilgaiyan et al., 2018)	ANN-feedforward back-propagation neural	MSE MMRE Pred	0.052 0.1349 (0.25) 95.2301	Conjunto de datos (21 proyectos)
	Elman neural network	MSE MMRE Pred	0.056 0.1480 (0.25) 94.8659	
(Angara et al., 2020)	Machine learning: Naive Bayes Classifier	Coefficient e de variación Media Desviación estándar	14% - 49% 1.67 - 2.67 0.41 - 0.98	Conjunto de datos (21 proyectos)
(Sharma & Chaudhary, 2022)	Redes neuronales	MSE R2	6.220651 17.03561	Conjunto de datos (21 proyectos)
	Algoritmos genéticos	MSE R2	6.742859 21.46326	
(Kaushik & Tayal, 2019)	Deep belief network (DBN) con Antlion optimization (ALO)	MMRE MdMRE Pred	0.02 0.02 (0.25) 98	Conjunto de datos (4 conjuntos de datos ágiles (Cocomo 81, Nasa 93, Maxwell y China) 3 conjuntos de datos no ágiles (Zia, Company Dataset 1 y Company Dataset 2))
(Vyas & Hemrajani, 2021)	Regresión lineal	RMSE MMRE Pred	16.86 0.15 (0.25) 71.42	Conjunto de datos (21 proyectos)
	Regresión logística	RMSE MMRE Pred	14.06 0.19 (0.25) 71.42	

<b>Estudio</b>	<b>Algoritmo/método</b>	<b>Métrica</b>	<b>Precisión</b>	<b>Proceso de validación</b>
	Regresión Ridge	RMSE MMRE Pred	7.75 0.13 (0.25) 85.71	
(Dave, 2021)	Sistemas difusos Marco de trabajo	No reportado	No reportado	Conjunto de datos (ISBSG data sets Atlassian JIRA)
(Raslan et al., 2018)	Lógica difusa	MER MMER Pred	0.22 - 0.24 0.25 (0.25) 80	Conjunto de datos (10 proyectos)
(Tripathi, 2021)	Marco de trabajo	No reportado	No reportado	No reportado
(Alshammari, 2022)	Métodos difusos	MER	0.22	Encuestas
(Butt et al., 2022)	Estadística descriptiva	R2 Tiempo Costo	 0.092 0.016	30 cuestionarios
(Abusaeed et al., 2023)	Proceso de jerarquía analítica difusa Marco de trabajo	No reportado	No reportado	Conjunto de datos (154 proyectos)
(Vetrò et al., 2018)	Regresión lineal Coeficiente de correlación de Spearman	R2	0.00 - 0.02	Conjunto de datos (4 proyectos)
(Usman et al., 2018)	Listas de verificación	BRE	-0.05	No reportado
(Adnan et al., 2019)	Modelos ontológicos	Accuracy MMRE Pred	85% 0.091 (0.25) 0.85	Conjunto de datos (20 proyectos)
	Planning Pocker	Accuracy MMRE Pred	75% 0.131 (0.25) 0.75	
	Delphi	Accuracy MMRE Pred	55% 0.231 (0.25) 0.55	
(Mallidi & Sharma, 2021)	Planning Pocker Talla de camiseta Votación por puntos Sistema de cubos Grande / Incierto / Pequeño Método de Ordenación Dividir hasta el tamaño máximo o menos	No reportado	No reportado	No reportado
(Premalatha & Srikrishna, 2019)	Deep belief network (DBN)	RRSE RAE MSE RMSE Accuracy MAE MMRE Pred	17.21% 9.23% 0.0232 0.054 99.487% 0.023 6.1 (0.25) 100, (0.10) 100	Conjunto de datos (160 tareas (dataset 1: 40 tareas, dataset 2: 50 tareas, dataset 3: 70 tareas))
(Radu, 2019)	Redes bayesianas	No reportado	No reportado	No reportado
(Meedeniya & Thennakoon, 2021)	Modelo conceptual	R2	22%	25 entrevistas
(Sharma & Chaudhary, 2020)	Regresión lineal	R2 MMRE	0.9476 0.099	Conjunto de datos (21 proyectos)
	Árboles de decisión	MMRE	0.38	
	Potenciación del gradiente	MMRE	0.16	



<i>Estudio</i>	<i>Algoritmo/método</i>	<i>Métrica</i>	<i>Precisión</i>	<i>Proceso de validación</i>
	Bosque aleatorio	MMRE	0.25	

### 3. RESULTADOS

Como respuesta a las preguntas de investigación planteadas, a continuación, se muestran los resultados obtenidos del análisis realizado a los 33 trabajos seleccionados para el mapeo sistemático:

*PI1: ¿Qué algoritmos/métodos se han implementado para la estimación del esfuerzo en proyectos de software?*

A lo largo del tiempo, se han desarrollado taxonomías que facilitan la clasificación de los métodos de estimación del esfuerzo, lo que contribuye significativamente a una mejor comprensión del campo de investigación y permite deducir diversas implicaciones prácticas y valiosas. En este contexto destacan algunos autores ampliamente reconocidos por sus propuestas de taxonomías, que abordan diversas técnicas de estimación del esfuerzo, incluyendo modelos algorítmicos, juicio de expertos, analogías, contexto de la estimación y técnicas predictoras del esfuerzo (Yang et al., 2023), (Boehm, 1984), (Mendes, 1d. C.) y (Usman et al., 2017). De acuerdo con esta clasificación se agruparon los estudios de este análisis, en la que se listan los diversos algoritmos/métodos que se han propuesto por la comunidad científica para estimar el esfuerzo en el desarrollo de proyectos de software bajo metodologías ágiles (Zakrani et al., 2018), (Angara et al., 2018), (Hacaloğlu & Demirörs, 2019), (Scott & Pfahl, 2018), (Khuat & Le, 2018), (Prasada Rao et al., 2018), (Gandomani et al., 2019), (Choetkiertikul et al., 2019), (Vera-Rivera et al., 2020), (Bhaskaran et al., 2019), (Prykhodko & Prykhodko, 2019), (Dantas et al., 2019), (Tanveer et al., 2019), (Rola & Kuchta, 2019), (Bilgaiyan et al., 2018), (Angara et al., 2020), (Sharma & Chaudhary, 2022), (Kaushik & Tayal, 2019), (Vyas & Hemrajani, 2021), (Dave, 2021), (Raslan et al., 2018), (Tripathi, 2021), (Butt et al., 2022), (Abusaeed et al., 2023), (Vetrò et al., 2018), (Usman et al., 2018), (Adnan et al., 2019), (Mallidi & Sharma, 2021), (Premalatha & Srikrishna, 2019), (Radu, 2019), (Meedeniya & Thennakoon, 2021), (Sharma & Chaudhary, 2020).

*PI2: ¿Qué métricas se han implementado para la estimación del esfuerzo en proyectos de software y qué precisión presentan los algoritmos implementados en la estimación del esfuerzo en proyectos de software?*

En el marco de esta investigación sobre la estimación del esfuerzo en proyectos de desarrollo de software, es esencial evaluar y medir la precisión de los métodos utilizados. Para alcanzar una comprensión completa de la efectividad de estas técnicas, se han implementado diversas métricas diseñadas específicamente para cuantificar la precisión en la estimación del esfuerzo. Los resultados a esta pregunta se presentan en la Tabla III columna tres, los trabajos del [8]-[40] incluyen magnitud del error relativo, mediana de la magnitud del error relativo, predicción en el nivel n, precisión estandarizada, error medio absoluto, error cuadrático, residuo medio absoluto, magnitud del error relativo, mediana del sesgo del error relativo equilibrado, error relativo, error absoluto, error cuadrático relativo raíz, error cuadrático medio raíz, precisión, error cuadrático medio, coeficiente de variación, media y desviación estándar (Zakrani et al., 2018), (Scott & Pfahl, 2018), (Khuat & Le, 2018), (Prasada Rao et al., 2018), (Gandomani et al., 2019) y (Choetkiertikul et al., 2019), (Prykhodko & Prykhodko, 2019), (Dantas et al., 2019), (Tanveer et al., 2019) (Bilgaiyan et al., 2018), (Angara et al., 2020), (Sharma & Chaudhary, 2022), (Kaushik & Tayal, 2019), (Vyas & Hemrajani, 2021).

*PI3: ¿Qué proceso de validación se ha implementado para cada algoritmo/método identificado?*

Uno de los pilares fundamentales en cualquier investigación científica es la validación rigurosa de los métodos y resultados. En el contexto de esta investigación centrada en la estimación del esfuerzo en proyectos de desarrollo de software, se ha dado énfasis particular a la evaluación de los procesos de validación implementados en cada uno de los estudios considerados. Los procesos de validación aplicados en cada estudio se presentan en la Tabla III columna cinco, en donde se observa que los estudios fueron validados con conjuntos de datos, encuestas, casos de estudio y cuestionarios (Zakrani et al., 2018), (Angara et al., 2018), (Hacaloğlu & Demirörs, 2019), (Scott & Pfahl, 2018), (Khuat & Le, 2018), (Prasada Rao et al., 2018), (Gandomani et al., 2019), (Choetkiertikul et al., 2019), (Vera-Rivera et al., 2020), (Bhaskaran et al., 2019), (Prykhodko & Prykhodko, 2019), (Dantas et al., 2019), (Tanveer et al., 2019), (Rola & Kuchta, 2019), (Bilgaiyan et al., 2018), (Angara et al., 2020), (Sharma & Chaudhary, 2022), (Kaushik & Tayal, 2019), (Vyas & Hemrajani, 2021), (Dave, 2021), (Raslan et al., 2018), (Tripathi, 2021), (Butt et al., 2022), (Abusaeed et al., 2023), (Vetrò et al., 2018), (Usman et al., 2018), (Adnan et al., 2019), (Mallidi & Sharma, 2021), (Premalatha & Srikrishna, 2019), (Radu, 2019), (Meedeniya & Thennakoon, 2021), (Sharma & Chaudhary, 2020).

En la Figura 6 se destaca la frecuencia con la que se han aplicado las métricas para medir la precisión de las estimaciones, proporcionando una evaluación objetiva de la adecuación de las estimaciones a los resultados reales del proyecto.

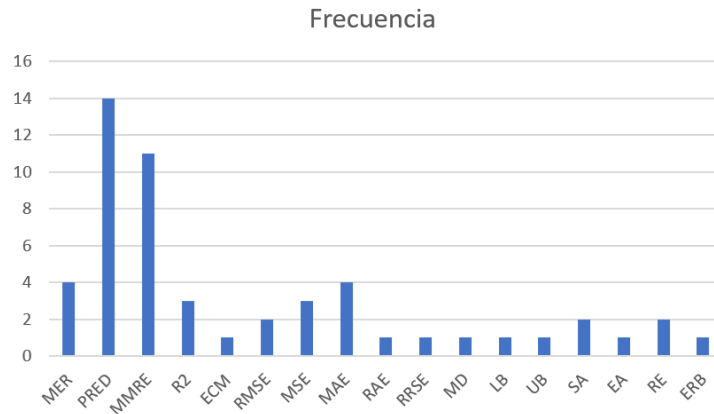


Figura 6.. Frecuencia de uso de las métricas implementadas en las investigaciones

#### 4. DISCUSIÓN

De acuerdo con el análisis realizado en este mapeo sistemático de la literatura los procesos de validación tienen diferente número de proyectos, por lo que es difícil generalizar la validez de estos procesos.

Es importante señalar que Bhaskaran et al., (2019), Tripathi, (2021), Usman et al., (2018), Mallidi & Sharma, (2021) y Radu, (2019) no han reportado explícitamente su método de validación, lo que podría limitar la capacidad de generalizar los hallazgos en este mapeo, o bien puede sugerir la posibilidad de sesgo en la implementación del método para medir la precisión de las estimaciones.

Solo Kaushik & Tayal, (2019) proporciona información sobre el conjunto de datos utilizado para validar la investigación, que incluye Cocomo81, Nasa 93, Maxwell, China y ISBS. La falta de información sobre los criterios de los proyectos en los demás estudios dificulta la reproducción de las investigaciones y complica la comparación de los valores de precisión de diversos métodos o algoritmos.

La precisión (PRED) y la magnitud media del error relativo (MMRE) son las métricas más utilizadas en la evaluación de la precisión de las estimaciones en relación con los resultados reales. Siendo Deep Belief Network (DBN) con Antlion Optimization (ALO) el algoritmo implementado con la mejor precisión, arrojando un valor de MMRE=0.02; por otro lado, el método más frecuentemente sugerido para llevar a cabo la estimación del esfuerzo en entornos ágiles es el Planning Poker. Llegar a una conclusión definitiva acerca de la idoneidad de estos algoritmos o métodos para la estimación en entornos DevOps se presenta como un desafío significativo. Esto se debe a la falta de revelación de información detallada sobre las características fundamentales, tales como complejidad, tamaño o naturaleza de los proyectos, en los estudios existentes

#### 5. CONCLUSIONES

En este estudio sobre la estimación del esfuerzo en proyectos de desarrollo de software ágiles/DevOps, se han abordado preguntas fundamentales que arrojan luz sobre la complejidad de este proceso esencial en la ingeniería de software. A través de un análisis detallado de los 33 estudios primarios seleccionados, de los cuales el 25% (8 artículos) son artículos de conferencias y el 75% (24 artículos) son revistas internacionales, esto indica la calidad de la investigación. Se obtuvieron valiosas respuestas que contribuyen al conocimiento y la mejora de la estimación del esfuerzo en este campo. Se identificó la diversidad de algoritmos que han sido implementados en la estimación del esfuerzo para el desarrollo de proyectos de software bajo metodologías ágiles, esto sugiere a los investigadores y profesionales utilizar diversas estrategias para abordar este desafío. También se identificaron las métricas que han sido implementadas para medir la precisión entre el valor estimado y el valor real del esfuerzo para el desarrollo de proyectos de software, además de las métricas que han sido utilizadas con mayor frecuencia por la comunidad científica. Al identificar los algoritmos que han sido implementados permitirá a la comunidad científica mejorar la precisión en la planificación, ayudando a establecer expectativas realistas sobre los tiempos estimados.

Este estudio aporta una comprensión profunda de la estimación del esfuerzo en el desarrollo de software ágil y proporciona una base sólida para mejorar las prácticas de estimación. La identificación de factores influyentes, la selección de métodos adecuados y la aplicación de métricas de precisión y procesos de validación son esenciales para avanzar hacia estimaciones más precisas y efectivas en el campo de la ingeniería de software. Se enfatiza que

la mayoría de la literatura existente se centra en proyectos de desarrollo de software tradicionales y ágiles dejando una gran brecha para el enfoque DevOps.

La precisión en la estimación del esfuerzo, respaldada por algoritmos confiables y métodos ágiles impacta directamente en la capacidad de la gestión de proyectos DevOps para cumplir con los objetivos de entrega rápida y eficiente. Este estudio da la pauta para realizar estimaciones continuas en un entorno ágil y dinámico como DevOps, adaptando o modificando los algoritmos o métodos basados en evidencias que arrojan una mejor o mayor precisión.

## 6. RECONOCIMIENTOS

Esta investigación fue apoyada por el Consejo Nacional de Humanidades, Ciencias y Tecnologías (CONAHCYT) con la beca No. CVU: 328123. Los autores reconocen al Tecnológico Nacional de México el apoyo prestado para la realización del presente trabajo.

## 7. REFERENCIAS

- Abusaeed, S., Khan, S. U. R., & Mashkoor, A. (2023). A Fuzzy AHP-based approach for prioritization of cost overhead factors in agile software development. *Applied Soft Computing*, 133, 109977. <https://doi.org/10.1016/j.asoc.2022.109977>
- Adnan, M., Afzal, M., & Asif, K. (2019). Ontology-Oriented Software Effort Estimation System for E-commerce Applications Based on Extreme Programming and Scrum Methodologies. *The Computer Journal*, 62, 1605-1624. <https://doi.org/10.1093/comjnl/bxy141>
- Alshammari, F. (2022). Cost estimate in scrum project with the decision-based effort estimation technique. *Soft Computing*, 26, 1-13. <https://doi.org/10.1007/s00500-022-07352-w>
- Angara, J., Prasad, S., & Sridevi, G. (2018). Towards Benchmarking User Stories Estimation with COSMIC Function Points-A Case Example of Participant Observation. *International Journal of Electrical and Computer Engineering (IJECE)*, 8(5), Article 5. <https://doi.org/10.11591/ijece.v8i5.pp3076-3083>
- Angara, J., Prasad, S., & Sridevi, G. (2020). DevOps Project Management Tools for Sprint Planning, Estimation and Execution Maturity. *Cybernetics and Information Technologies*, 20(2), 79-92. <https://doi.org/10.2478/cait-2020-0018>
- Bhaskaran, N. A., Jayaraj, Dr. V., & Professor, School of Computer Science Engineering & Applications, Bharathidasan University Tiruchirappalli, India,. (2019). A Hybrid Effort Estimation Technique for Agile Software Development (HEETAD). *International Journal of Engineering and Advanced Technology*, 9(1), 1078-1087. <https://doi.org/10.35940/ijeat.A9480.109119>
- Bilgaiyan, S., Mishra, S., & Das, M. N. (2018). Effort estimation in agile software development using experimental validation of neural network models. *International Journal of Information Technology*, 11. <https://doi.org/10.1007/s41870-018-0131-2>
- Boehm, B. W. (1984). Software Engineering Economics. *IEEE Transactions on Software Engineering*, SE-10(1), 4-21. <https://doi.org/10.1109/TSE.1984.5010193>
- Butt, S. A., Khalid, A., Ercan, T., Ariza-Colpas, P. P., Melisa, A.-C., Piñeres-Espitia, G., De-La-Hoz-Franco, E., Melo, M. A. P., & Ortega, R. M. (2022). A software-based cost estimation technique in scrum using a developer's expertise. *Advances in Engineering Software*, 171, 103159. <https://doi.org/10.1016/j.advengsoft.2022.103159>
- Choetkiertikul, M., Dam, H. K., Tran, T., Pham, T., Ghose, A., & Menzies, T. (2019). A Deep Learning Model for Estimating Story Points. *IEEE Transactions on Software Engineering*, 45(7), 637-656. <https://doi.org/10.1109/TSE.2018.2792473>
- Dantas, E., Costa, A., Vinicius, M., Perkusich, M., Almeida, H., & Perkusich, A. (2019). *An Effort Estimation Support Tool for Agile Software Development: An Empirical Evaluation*. 82-87. <https://doi.org/10.18293/SEKE2019-141>
- Dave, C. V. (2021). An Efficient Framework for Cost and Effort Estimation of Scrum Projects. *International Journal for Research in Applied Science and Engineering Technology*, 9(11), 1478-1487. <https://doi.org/10.22214/ijraset.2021.39030>
- Gandomani, T. J., Faraji, H., & Radnejad, M. (2019). Planning Poker in cost estimation in Agile methods: Averaging Vs. Consensus. *2019 5th Conference on Knowledge Based Engineering and Innovation (KBEI)*, 066-071. <https://doi.org/10.1109/KBEI.2019.8734960>
- Grimstad, S., Jørgensen, M., & Moløkken-Østfold, K. (2006). Software effort estimation terminology: The tower of Babel. *Information and Software Technology*, 48(4), 302-310. <https://doi.org/10.1016/j.infsof.2005.04.004>
- Guía del PMBOK. (2017). GUÍA DE LOS FUNDAMENTOS PARA LA DIRECCIÓN DE PROYECTOS (PMBOK). *Project Management Institute, Inc.*

- Hacaloğlu, T., & Demirörs, O. (2019). *Measurability of functional size in agile software projects: Multiple case studies with COSMIC FSM*. IEEE. <https://doi.org/10.1109/SEAA.2019.00041>
- Kaushik, A., & Tayal, D. (2019). A Comparative Analysis on Effort Estimation for Agile and Non-agile Software Projects Using DBN-ALO. *Arabian Journal for Science and Engineering*, 45. <https://doi.org/10.1007/s13369-019-04250-6>
- Khuat, T. T., & Le, M. H. (2018). A Novel Hybrid ABC-PSO Algorithm for Effort Estimation of Software Projects Using Agile Methodologies. *Journal of Intelligent Systems*, 27(3), 489-506. <https://doi.org/10.1515/jisys-2016-0294>
- Kitchenham, B., & Charters, S. (2007). *Guidelines for performing Systematic Literature Reviews in Software Engineering*. 2.
- Mallidi, R. K., & Sharma, M. (2021). Study on Agile Story Point Estimation Techniques and Challenges. *International Journal of Computer Applications*, 174(13), 9-14. <https://www.ijcaonline.org/archives/volume174/number13/31736-2021921014>
- McConnell, S. (2006). *Software Estimation: Demystifying the Black Art*.
- Meedeniya, D., & Thennakoon, H. (2021, agosto 12). *Impact Factors and Best Practices to Improve Effort Estimation Strategies and Practices in DevOps*. <https://doi.org/10.1145/3484399.3484401>
- Mendes, E. (1d. C.). Cost Estimation Techniques for Web Projects. En <https://services.igi-global.com/resolvedoi/resolve.aspx?doi=10.4018/978-1-59904-135-3>. IGI Global. <https://www.igi-global.com/book/cost-estimation-techniques-web-projects/www.igi-global.com/book/cost-estimation-techniques-web-projects/207>
- Page, M. J., McKenzie, J. E., Bossuyt, P. M., Boutron, I., Hoffmann, T. C., Mulrow, C. D., Shamseer, L., Tetzlaff, J. M., Akl, E. A., Brennan, S. E., Chou, R., Glanville, J., Grimshaw, J. M., Hróbjartsson, A., Lalu, M. M., Li, T., Loder, E. W., Mayo-Wilson, E., McDonald, S., ... Alonso-Fernández, S. (2021). Declaración PRISMA 2020: Una guía actualizada para la publicación de revisiones sistemáticas. *Revista Española de Cardiología*, 74(9), 790-799. <https://doi.org/10.1016/j.recesp.2021.06.016>
- Petersen, K., Vakkalanka, S., & Kuzniarz, L. (2015). Guidelines for conducting systematic mapping studies in software engineering: An update. *Information and Software Technology*, 64, 1-18. <https://doi.org/10.1016/j.infsof.2015.03.007>
- Pinciroli, F. (2023). *Strategies for agile software development based on technical and environmental complexity factors*. XXVIII Congreso Argentino de Ciencias de la Computación (CACIC) (La Rioja, 3 al 6 de octubre de 2022). <http://sedici.unlp.edu.ar/handle/10915/149423>
- Prasada Rao, Ch., Siva Kumar, P., Rama Sree, S., & Devi, J. (2018). An Agile Effort Estimation Based on Story Points Using Machine Learning Techniques. En V. Bhateja, J. M. R. S. Tavares, B. P. Rani, V. K. Prasad, & K. S. Raju (Eds.), *Proceedings of the Second International Conference on Computational Intelligence and Informatics* (pp. 209-219). Springer. [https://doi.org/10.1007/978-981-10-8228-3\\_20](https://doi.org/10.1007/978-981-10-8228-3_20)
- Premalatha, H., & Srikrishna, C. (2019). Effort Estimation in Agile Software Development using Evolutionary CostSensitive Deep Belief Network. *International Journal of Intelligent Engineering and Systems*, 12, 261-269. <https://doi.org/10.22266/ijies2019.0430.25>
- Prykhodko, N. V., & Prykhodko, S. B. (2019). A MULTIPLE NON-LINEAR REGRESSION MODEL TO ESTIMATE THE AGILE TESTING EFFORTS FOR SMALL WEB PROJECTS. *Radio Electronics, Computer Science, Control*, 2, Article 2. <https://doi.org/10.15588/1607-3274-2019-2-17>
- Radu, L.-D. (2019). Effort Prediction in Agile Software Development with Bayesian Networks: *Proceedings of the 14th International Conference on Software Technologies*, 238-245. <https://doi.org/10.5220/0007842802380245>
- Raslan, A., Darwish, N., & Cairo University. (2018). An Enhanced Framework for Effort Estimation of Agile Projects. *International Journal of Intelligent Engineering and Systems*, 11(3), 205-214. <https://doi.org/10.22266/ijies2018.0630.22>
- Rola & Kuchta. (2019). Application of Fuzzy Sets to the Expert Estimation of Scrum-Based Projects. *Symmetry*, 11, 1032. <https://doi.org/10.3390/sym11081032>
- Scott, E., & Pfahl, D. (2018). Using developers' features to estimate story points. *Proceedings of the 2018 International Conference on Software and System Process*, 106-110. <https://doi.org/10.1145/3202710.3203160>
- Sharma, A., & Chaudhary, N. (2020). Linear Regression Model for Agile Software Development Effort Estimation. *2020 5th IEEE International Conference on Recent Advances and Innovations in Engineering (ICRAIE)*, 1-4. <https://doi.org/10.1109/ICRAIE51050.2020.9358309>
- Sharma, A., & Chaudhary, N. (2022). Analysis of Software Effort Estimation Based on Story Point and Lines of Code using Machine Learning. *International Journal of Computing and Digital Systems*, 12(1), 131-140. <https://doi.org/10.12785/ijcds/1201012>

- Tanveer, B., Vollmer, A. M., Braun, S., & Ali, N. (2019). An evaluation of effort estimation supported by change impact analysis in agile software development. *Journal of Software: Evolution and Process*, 31. <https://doi.org/10.1002/smr.2165>
- The Standish Group—View News. (s. f.). Recuperado 15 de enero de 2024, de <https://www.standishgroup.com/news/45>
- Tripathi, D. D. R. (2021). Evaluation of the Feasibility of Parametric Estimation in Devops Continuous Planning. *International Journal for Research in Applied Science and Engineering Technology*, 9(9), 542. [https://www.academia.edu/52397035/Evaluation\\_of\\_the\\_Feasibility\\_of\\_Parametric\\_Estimation\\_in\\_Devops\\_Continuous\\_Planning](https://www.academia.edu/52397035/Evaluation_of_the_Feasibility_of_Parametric_Estimation_in_Devops_Continuous_Planning)
- Usman, M., Börstler, J., & Petersen, K. (2017). An Effort Estimation Taxonomy for Agile Software Development. *International Journal of Software Engineering and Knowledge Engineering*, 27, 641-674. <https://doi.org/10.1142/S0218194017500243>
- Usman, M., Petersen, K., Börstler, J., & Santos Neto, P. (2018). Developing and using checklists to improve software effort estimation: A multi-case study. *Journal of Systems and Software*, 146, 286-309. <https://doi.org/10.1016/j.jss.2018.09.054>
- Valenzuela, B., Alvarado, L., Santaolaya, R., & Reyes, H. (2023). Identification of methods, approaches, and factors in effort estimation for DevOps projects: A systematic literature mapping. *2023 Mexican International Conference on Computer Science (ENC)*.
- Vera-Rivera, F., Barbosa-Mora, J., & Gaona, M. (2020). Generación automática de la planificación de la entrega en desarrollo de software ágil, asignación de historias de usuario a los desarrolladores usando algoritmos genéticos. *Aibi revista de investigación, administración e ingeniería*, 8, 29-38. <https://doi.org/10.15649/2346030X.735>
- Vetrò, A., Dürre, R., Conoscenti, M., Fernández, D. M., & Jørgensen, M. (2018). Combining Data Analytics with Team Feedback to Improve the Estimation Process in Agile Software Development. *Foundations of Computing and Decision Sciences*, 43(4), 305-334. <https://doi.org/10.1515/fcds-2018-0016>
- Vyas, M., & Hemrajani, N. (2021). *PREDICTING EFFORT OF AGILE SOFTWARE PROJECTS USING LINEAR REGRESSION, RIDGE REGRESSION AND LOGISTIC REGRESSION*. 13(2).
- Yang, X., Liu, J., & Zhang, D. (2023). A Comprehensive Taxonomy for Prediction Models in Software Engineering. *Information*, 14(2), Article 2. <https://doi.org/10.3390/info14020111>
- Zakrani, A., Najm, A., & Marzak, A. (2018). Support Vector Regression Based on Grid-Search Method for Agile Software Effort Prediction. *2018 IEEE 5th International Congress on Information Science and Technology (CiSt)*, 1-6. <https://doi.org/10.1109/CIST.2018.8596370>

## BIOGRAFÍA DE LOS AUTORES



M.C. Iliana Lizbeth Alvarado Lara recibió el grado de Maestra en Ciencias en Ciencias de la Computación en el Tecnológico Nacional de México campus Centro Nacional de Investigación y Desarrollo Tecnológico (CENIDET) México. Actualmente es estudiante de doctorado en el CENIDET. Sus áreas de investigación son Gestión de proyectos, Estimación de proyectos de software, Ingeniería de software y Automatización de procesos de software.



Dra. Blanca Dina Valenzuela Robles, profesora del Tecnológico Nacional de México campus Centro Nacional de Investigación y Desarrollo Tecnológico CENIDET. Obtuvo el grado de Maestría y Doctorado en Ciencias de la Computación en el Tecnológico Nacional de México campus Centro Nacional de Investigación y Desarrollo Tecnológico (CENIDET) México. Área de interés es la Ingeniería de Software, específicamente, e-learning y servicios web, el análisis y diseño de metodologías de reuso de software, patrones de diseño, metodologías para el desarrollo de proyectos de software, gestión de proyectos de software, arquitecturas de software, DevOps.



Dr. René Santaolaya Salgado, es profesor del Tecnológico Nacional de México/Centro Nacional de Investigación y Desarrollo Tecnológico, ubicado en Cuernavaca, Morelos, México. Sus áreas de investigación incluyen “Modelos de Procesos de Software”, “Arquitectura de Software”, “Reuso de Software”, “Servicios Web”, “Microservicios” y “Computación en la Nube”. El profesor Santaolaya recibió su grado de Doctor en Ciencias de la Computación con especialidad en Ingeniería de Software del Instituto Politécnico Nacional/Centro de Investigación en Computación. El es miembro senior del IEEE y ha sido analista, desarrollador, consultor, y líder del Cuerpo Académico de Ingeniería de Software de su Centro de Trabajo.



Dr. Juan Gabriel González Serna, doctor en ciencias computacionales por el CIC-IPN, miembro del Sistema Nacional de Investigadores nivel II (SNI-II) del CONAHCyT, sus áreas de investigación son Interacción Humano Computadora, Evaluación UX, computación afectiva y análisis de emociones.



Dr. Noé Alejandro Castro Sánchez, realizó sus estudios de maestría y doctorado en el Centro de Investigación en Computación del Instituto Politécnico Nacional, especializándose en Procesamiento de Lenguaje Natural. Es miembro del Sistema Nacional de Investigadores, vocal de la Sociedad Mexicana de Inteligencia Artificial, miembro de la Asociación Latinoamericana de Tecnologías del Lenguaje y docente del Centro Nacional de Investigación y Desarrollo Tecnológico (CENIDET). Sus trabajos e intereses de investigación se insertan en el área de Procesamiento de Lenguaje Natural e incluyen la minería de opiniones y análisis de sentimientos, el humor computacional y el análisis de contenido, aplicados al sector salud, educación y seguridad.



Esta obra está bajo una licencia de Creative Commons Reconocimiento-NoComercial-CompartirIgual 2.5 México.

*Recibido 29 Jul. 2023*

*ReCIBE, Año 12 No.2, NOV. 2023*

*Aceptado 28 JUL. 2023*

## **Deudas técnicas en el sistema transaccional**

## **Technical debts in the transactional system**

**Javier Fabricio Cucalón Gaibor<sup>1</sup>**  
**Jacobo Sandoval Gutierrez<sup>1</sup>**

<sup>1</sup>Universidad Autónoma Metropolitana

## Resumen

En las empresas internacionales de servicios financieros transaccionales ocasionalmente se realizan actualizaciones de programa con errores no identificables por el compilador. Estos errores definidos como deuda técnica provocan fallas en el sistema transaccional al no cumplir el modelo de calidad. Las investigaciones consultadas no han reportado la existencia de nuevas métricas para mitigar las fallas por las deudas técnicas. Por lo anterior, el objetivo de la investigación realizada fue obtener un conjunto de nuevas métricas para incorporarlas al modelo de calidad. La metodología consistió en recopilar una muestra de programas, definir variables y métricas, y evaluar la causa-efecto para encontrar las deudas técnicas. El experimento consistió en obtener de dos empresas las muestras de los programas modificados en el software BASE24-eps, se definieron cinco métricas. El resultado mostró que los programas tienen una mayor probabilidad de encontrar al menos una deuda técnica relacionadas a bug y code smell. La investigación, permite concluir la importancia de incorporar métricas al proceso de calidad.

**Palabras clave:** Deuda técnica, Métrica, Modelo de Calidad.

## Abstract

In international transactional financial services companies, program updates are occasionally made with errors not identifiable by the compiler. These errors, defined as technical debt, cause failures in the transactional system by not complying with the quality model. The investigations consulted have not reported the existence of new metrics to mitigate failures due to technical debts. Therefore, the research objective was to obtain new metrics to incorporate into the quality model. The methodology consisted of compiling a sample of programs, defining variables and metrics, and evaluating the cause-effect to find the technical debts. The experiment obtained samples of modified programs in the BASE24-eps software from two companies, and five metrics were defined. The result showed that the programs have a higher probability of finding at least one technical debt related to bug and code smell. This research allows concluding the importance of incorporating metrics into the quality process..

**Keywords:** Metrics, Quality model, Technical debt.



## 1. Introducción

Las empresas internacionales de servicios financieros transaccionales realizan modificaciones sobre productos de software para poder implementar reglas que cumplan las especificaciones de sus negocios. El proceso de desarrollo de software inicia desde la definición del requerimiento hasta el despliegue del cambio en el ambiente productivo, sin embargo, se ha encontrado con frecuencia que dentro del proceso se cumple con el modelo de calidad y aún así, se siguen generando errores conocidos como deudas técnicas en los programas modificados (Gartner, 2023; SONAR, 2023).

Uno de los productos de software es B24-eps, sus ventajas radican en ser un software de clase mundial que esta operando en todos los continentes del mundo, en diversos tipos de empresas como instituciones financieras, bancos, redes de integración de servicios financieros, retails y otras empresas que necesitan integrar medios de pago para servicio de sus clientes y así estos puedan realizar transacciones financieras a través de los distintos canales de pago como ATMs, POSs, Tellers, botones de pago, webservices, entre otros (ACI-Worldwide, 2023). Las deudas técnicas se explican como los errores, deficiencias y carencias que tiene el código modificado por acciones deliberadas o involuntarias de los programadores, estos no son errores de compilación (Stopford, Wallace, y Allspaw, 2017; Kruchten, Nord, y Ozkaya, 2012). Hay 3 tipos de deudas técnicas, la primera son los bugs, los cuales generan de forma aleatoria las interrupciones en la ejecución de los programas provocando una ausencia temporal o permanente de la disponibilidad del producto de software y en consecuencia una pérdida de dinero para las empresas al no poder ofrecer los servicios a sus clientes. La segunda son los code smells, los cuales generan una complejidad en el mantenimiento futuro de los programas provocando altos costos para las empresas debido al mayor tiempo que los programadores necesitan para realizar las modificaciones. Y la tercera son las reglas de seguridad, las cuales provocan condiciones de vulnerabilidad ante los ataques informáticos en el producto de software y por ello, se aumentan los costos en las empresas por las pérdidas de información o la ausencia temporal o permanente de los servicios a sus clientes (Alves y cols., 2016).

La familia de normas ISO/IEC 25000, conocida como System and Software Quality Requirements and Evaluation (SQuaRE) es un conjunto de normas cuyo objetivo es la creación de un marco de trabajo común para realizar la evaluación de calidad del producto de software. Esta se conforma por las siguientes divisiones, ISO/IEC 2500n División de Gestión de la Calidad, ISO/IEC 2501n División de Modelo de Calidad, ISO/IEC 2502n División de Medición de Calidad, ISO/IEC 2503n División de Requisitos de Calidad y ISO/IEC 2504n División de Evaluación de Calidad.

Las normas ISO/IEC 25000 contienen la norma internacional ISO/IEC 25010 que provee un modelo de calidad para productos de software, categorizando sus propiedades en ocho características que son: adecuación funcional, eficiencia de rendimiento, compatibilidad, usabilidad, confiabilidad, seguridad, mantenibilidad y portabilidad. Cada característica esta compuesta por un conjunto de sub características relacionadas (ISO, 2014).

### **1.1. Proceso de modificación**

B24-eps es un software que implementa reglas de negocio del sistema de servicios financieros transaccionales basados en los estándares mundiales y así las empresas pueden operar su ecosistema de medios de pago con los cuales trabajan. Sin embargo, en cada región se pueden implementar reglas de negocio particulares, por ejemplo, por normativas del país correspondiente, por las políticas propias de la empresa que tienen el objetivo de aprovechar al máximo los canales transaccionales existentes o simplemente por la implementación de nuevos modelos transaccionales.

Por este motivo, es necesario realizar modificaciones a los programas del software B24-eps que permita implementar estas reglas de negocio particulares que no son parte del producto base. En el proceso de desarrollo de las reglas de negocio en los programas B24-eps se tienen tres fases de trabajo. Las dos primeras fases son: fase de modificación que logra implementar la regla de negocio solicitada en el requerimiento, luego de esto, viene la fase de compilación de los programas modificados, en donde se logran identificar aquellos errores que no cumplen con las reglas de programación del compilador C++. Lo anterior es un proceso iterativo que finaliza cuando se obtiene un programa modificado que cumple con la regla de negocio solicitada y con cero errores de compilación.

En la tercera fase se verifica el cumplimiento del modelo de calidad. El proceso consiste en comparar que ciertas líneas de código del programa modificado coincidan con las métricas de calidad. Las métricas de calidad son un conjunto de sintaxis con diferentes formatos como pueden ser letras, números, longitudes de dígitos, rangos, entre otros o reglas que deben ser escritas en los programas modificados para cumplimiento de la métrica (Atlassian, 2023; Pavlič, Hliš, Heričko, y Beranič, 2022; Mamun, Martini, Staron, Berger, y Hansson, 2019).

### **1.2. Solución a las deudas técnicas**

La norma internacional ISO/IEC 25010 no es aplicable completamente para todos los tipos de productos de software, algunas características o subcaracterísticas no son aplicables o faltan otras nuevas dentro de la norma. Por ejemplo, la subcaracterística de instalabilidad es usualmente no aplicable al tipo de software web services, porque estos son ejecutados remotamente en el lado del servidor. Y el caso de B24-eps sucede algo similar, esta norma no es totalmente aplicable para este producto de software.

En consecuencia, para cubrir la falencia de la norma internacional ISO/IEC 25010 algunos productos de software han desarrollado, a partir de este, sus propios modelos de calidad que le permitan identificar la deuda técnica de los programas modificados. Por consiguiente, B24-eps no tiene su propio modelo de calidad de software.

Una de las herramientas Open Source de análisis de código estático mas comúnmente usada tanto en la academia como en la industria es SonarQube (Saarimaki, Baldassarre, Lenarduzzi, y Romano, 2019). Esta herramienta se puede obtener como un servicio desde la plataforma sonarcloud.io o puede ser descargada e instalada en un servidor privado. SonarQube permite el análisis de los programas verificando que el código cumpla con la reglas establecidas en las métricas . En caso de una violación a cualquiera de las reglas se genera el reporte de una deuda técnica

(Li, Soliman, y Avgeriou, 2022). Al conjunto de métricas para un tipo de lenguaje de programación o producto de software se lo conoce como modelo de calidad. Actualmente SonarQube es usado por 7M+ de usuarios y 400k+ de empresas en los campos de la salud, IT, gobierno, educación, energía, telecomunicaciones, industriales, seguros, etc. alrededor del mundo (Alfayez, Winn, Alwehaibi, Venson, y Boehm, 2023; Katin, Lenarduzzi, Taibi, y Mandić, 2022).

## 2. Método

El método se concibió de la siguiente manera: La primera etapa consistió en identificar un conjunto de archivos que han sido actualizados o modificados, que no tengan errores de compilación y que cumplan con las métricas estándar del modelo de calidad actual. Seleccionar una muestra representativa aleatoria de la población identificada. Estudiar las líneas de código modificadas por el programador y comparar que cada modificación del programa se encuentre homologada. En caso, de no estar homologada se considera como una probable deuda técnica y por tanto, se agregará a una lista de métricas. Finalmente, el resto de los archivos serán auditados con las nuevas métricas para obtener una estadística.

Para el trabajo experimental, se eligió como base la norma ISO/IEC 25010 y se tuvo acceso a los archivos en producción de dos diferentes empresas, es decir, aquellos que superaron las pruebas de compilación y se encuentran en ejecución. En total 273 programas fueron recopilados, el 46% corresponde a la extensión \*.h y 54% a \*.cpp. De estos programas se propuso un intervalo de confianza del 98%, un error de 0.5 y una desviación estándar de 1.5 para conocer  $n = 50$  como el número de muestra aleatorias para validar el experimento. Del conjunto muestra se realizó el análisis de las líneas de código para poder desglosar los siguientes criterios.

- Líneas de código totales por tipo de programa. Se refiere a la cantidad de líneas que tiene el programa en texto plano, sin importar si el compilador las utiliza o no.
  - Líneas de código efectivas por programa. Son las líneas que el compilador interpreta para interactuar en el sistema.
  - Líneas de código comentadas por programa. Son todas las líneas de orientación para el programador que no son utilizadas por el compilador, pero sirven para conocer la evolución del programa.
- Cambios realizados en las líneas de código efectiva. Es la comparación del número de cambios realizados para cumplir una modificación y su repercusión en líneas de código efectivo.
- Cambios realizados en las líneas de código comentadas. Es la comparación del número de cambios realizados para cumplir una modificación y su repercusión en líneas de código comentada.
- Listado de las métricas. Son características no homogéneas que potencialmente son o serán una deuda técnica.

El procedimiento realizado a cada programa seleccionado de forma aleatoria se agrupo por análisis a las líneas de código, análisis de los cambios de código y el cambio no homogéneo, tal cual se muestra en la Figura1.

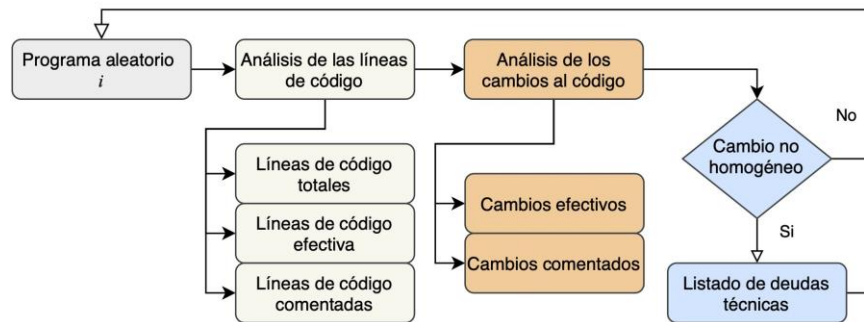


Figura 1: Análisis de los programas para encontrar las deudas técnicas

Los análisis de líneas de código son tareas cuantificables por la sintaxis del propio programa. Sin embargo, los análisis a los cambios al código son tareas que requieren la solicitud de una orden de trabajo y su comparación entre los cambios realizados y que tengan diferencias mínimas para el mismo objetivo o llamado de la función.

Adicionalmente, detectar un cambio no homogéneo requiere agrupar las órdenes de trabajo similares y después compararlas entre sí. Estos últimos dos tipos de análisis, requieren de programadores experimentados en el lenguaje técnico y operativo de los sistemas financieros.

Para determinar si las extensiones de los programas son diferentes entre sí, se realizó una prueba *t* con dos medias para comprobar la distribución de toda la población. Además, se realizaron tres pruebas de regresión para determinar la correlación entre los criterios analizados:

- Número de cambios vs líneas de código totales.
- Número de cambios vs líneas de código efectivo.
- Número de cambios vs líneas de código comentado.

Después de obtener la lista métricas de deuda técnica y las relaciones que puedan existir entre los programas. Se realizó un distribución de probabilidad para los escenarios más probables. Y con ello, determinar las futuras deudas que seguirán apareciendo.

### 3. Resultados

El número de líneas de código por cada tipo de documento estuvieron en un rango desde 65 hasta 12,017 con una media de 957 para la extensión \*.h y de 122 hasta 10,783 con una media de 1,211 para \*.cpp. El resultado de la prueba para toda la población considerando un intervalo de confianza 95%, con 271 grados de libertad cuando  $t_0 > t_1$  ( $0.05 > 1.65$ ), se determinó, que no hay diferencia sustancial entre los dos tipos de extensiones.

Para el análisis comparativo entre las líneas de código efectivo independientemente del tipo de extensión se obtuvo un promedio de 94.5% en comparación con el código comentado del 5.5%. El resultado de la prueba para toda la población considerando un intervalo de confianza 95%, con 271 grados de libertad cuando  $t_0 > t_1$  ( $0.05 > 1.65$ ), se determinó, que no hay diferencia entre los dos tipos de documentos.

En las pruebas de correlación para los casos de líneas de código totales y efectivas en comparación con el número de cambios se obtuvo un coeficiente de correlación de 0.19 y 0.17 respectivamente. Sin embargo, el coeficiente de correlación fue de 0.96 para el caso de líneas comentadas por el número de cambios. En la Figura 2 se muestran la relación del número de cambios en comparación con las líneas de código comentadas para el caso de la extensión CPP.

De los documentos revisados se tuvieron 5 métricas con inconsistencias en los documentos de la muestra. Las características que deberían estar homogéneas son las siguientes métricas:

- Métrica No. 1 - Token Registry Metric: Su existencia y valor a TRUE garantiza que las funciones que setean los valores desde los tokens a los TDEs o desde los TDEs a los tokens en formato binario o carácter y en procesamiento de solicitud o respuesta reemplacen a las funciones del producto en su versión original.
- Métrica No. 2 - Class Prefix Metric: El nombre de las clases customizadas deben empezar con la palabra `csm_`. Esto garantizará la diferencia con las clases del producto en su versión original.
- Métrica No. 3 - SSID Event Metric: Es obligatorio que el SSID de los eventos de una clase customizada este entre los valores 150-299 incluidos. Esto garantizará la convivencia de las clases customizadas y las del producto en su versión original.

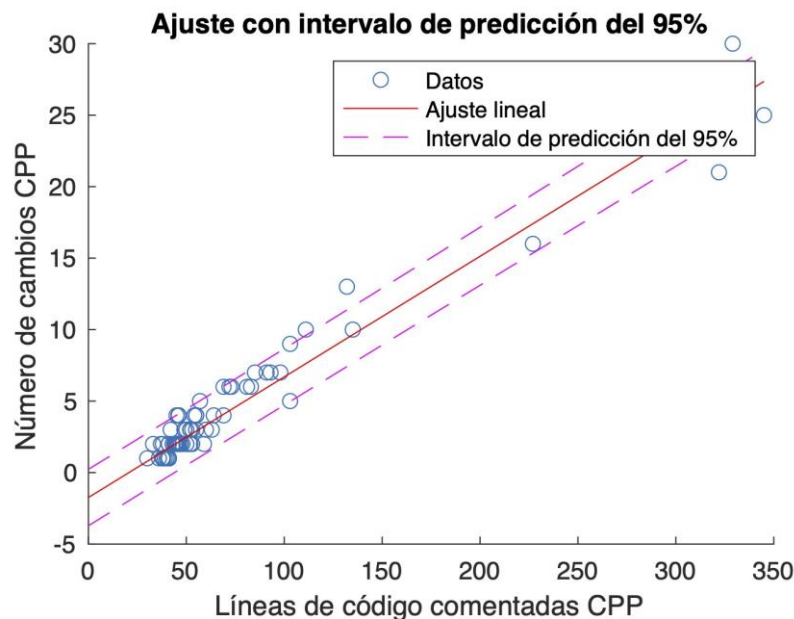


Figura 2: Correlación de los cambios y líneas comentadas

- Métrica No. 4 - TDE Object ID Metric: Es obligatorio que el object ID de un TDE de forma customizada este entre los valores 30,000-31,999 incluidos. Esto garantizará la convivencia de las clases customizadas y las del producto en su versión original.
- Métrica No. 5 – File Last Letter Metric: El nombre de los documentos debe terminar con la letra q y luego seguir la extensión .h o .cpp. Esto se considera un buena práctica para identificación temprana de las clases customizadas en comparación con las del producto en su versión original.

La auditoría aplicada al total de la población mostró que los archivos contienen al menos una de las cinco métricas propuestas en un 89%.

Para el caso de la métrica uno apareció en un 5%, para dos 27%, tres 24%, cuatro 13% y para cinco un 31%. Mientras el 54% presentó máximo una métrica y del restante 46% máximo dos métricas un 85%. Las frecuencias de aparición se muestran en la Figura 3

La distribución de probabilidad considera cinco escenarios. El primero sin deuda técnica, y del segundo hasta el cuarto con 1, 2, 3 y 4 deudas técnicas respectivamente. En la Figura 4 se puede apreciar, una mayor probabilidad de encontrar una, dos y tres deudas técnicas en la auditoría de un programa al azar. En el antepenúltimo escenario se lograría un programa sin deudas técnicas y en último lugar sería menos probable tener más de cuatro deudas técnicas. En tanto, la esperanza para una auditoría realizada al azar tendría un valor de 1.52 de deudas técnicas.

En la Tabla 1 se muestran las clasificaciones de las métricas encontradas en los programas. No se encontró evidencia de alguna deuda relacionada con la seguridad. Dos de las métricas se relacionan con las del tipo smell code. Las del tipo Bug fueron tres métricas. Aunque, los porcentaje de los tipos de deudas fueron equilibradas, con un 42% para Bug y el restante para smell code. Adicionalmente, no se encontraron más de dos tipos de deuda técnica en una métrica.

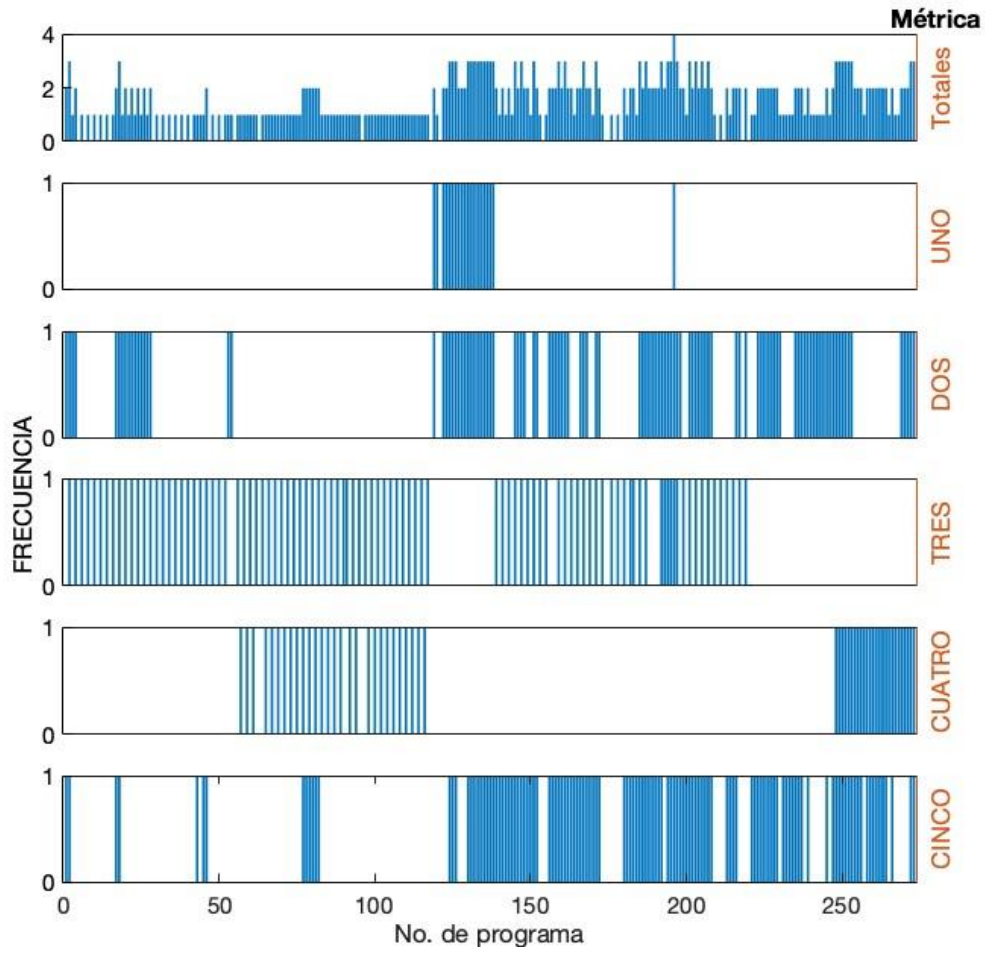


Figura 3: Auditoría para todos los archivos

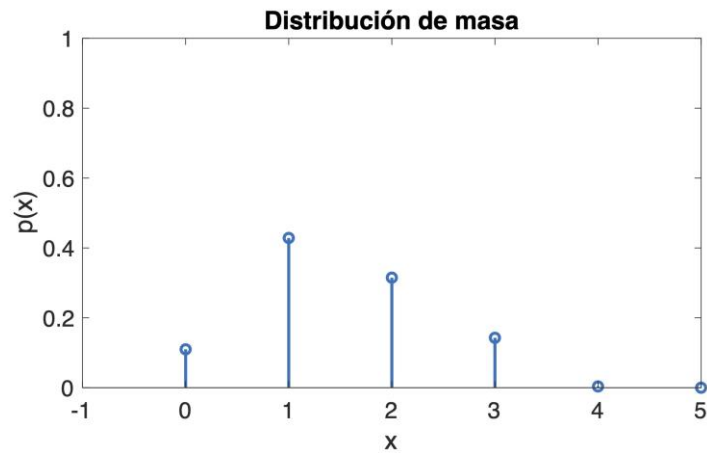


Figura 4: Distribución de probabilidad de las deudas técnicas Tabla 1:  
Tipo de deuda técnica para cada métrica

Métrica	Bug	Smell code	Seguridad
Token Registry Metric	Si	No	No
Class Prefix Metric	No	Si	No
SSID Event Metric	Si	No	No
TDE Object ID Metric	Si	No	No
File Last Letter Metric	No	Si	No

#### 4. Conclusiones

El desarrollo de la investigación permitió cumplir el objetivo de identificar algunas métricas que necesitan ser incorporadas a los modelos de calidad para evitar dos tipos de deudas técnicas conocidas como bugs y code smells. No se encontró alguna deuda técnica del tipo de seguridad en los programas analizados. La implicación de las métricas tipo bugs, como la No. 1 Token Registry Metric, la No. 3 SSID Event Metric y la No. 4 TDE Object ID Metric, evitarían interrupciones en la ejecución de los programas y la ausencia temporal o permanente de la disponibilidad del producto de software. De la misma manera las métricas tipo code smells, como la No. 2 Class Prefix Metric y la No. 5 File Last Letter Metric, simplificarían en el futuro los mantenimientos a los programas. El porcentaje de correlación con el número de cambios en contraste con las líneas comentadas tuvieron un 96%. Es decir, si se revisan e integran los comentarios en alguna etapa del proceso de compilación o producción del programa se podría minimizar las deudas técnicas.

El método fue válido para los programas de dos diferentes empresas de servicios financieros transaccionales que utilizan la herramienta B24-eps. Los pasos sugeridos en el método pueden ser replicados por cualquier empresa del tipo financiero independientemente de la herramienta utilizada. Con lo anterior, cada empresa podría sugerir alguna actualización de la norma o ratificar las métricas propuestas en esta investigación.

Al menos dos vertientes de trabajos futuros se han encontrado. La primera vinculada a la relación que pueda existir entre la deuda técnica y el tipo de programador. La segunda idea se puede centrar en la creación de un complemento automatizado en la detección de deudas técnica en una plataforma de análisis de código estático como Sonarqube.

#### 5 Referencias

- ACI-Worldwide. (2023). *Manage multi-channel transactions with base24-eps*. Descargado de <https://recibe.page.link/ACI23>
- Alfavez, R., Winn, R., Alwehaibi, W., Venson, E., y Boehm, B.(2023). How sonarqube-identified technical debt is prioritized: An exploratory case study. *Information and Software Technology*, 156, 107147. doi: <https://doi.org/10.1016/j.infsof.2023.107147>



- Alves, N. S., Mendes, T. S., Mendonça, M. G. D., Spinola, R. O., Shull, F., y Seaman, C. (2016). Identification and management of technical debt: A systematic mapping study. *Information and Software Technology*, 70, 100-121. doi: <https://doi.org/10.1016/j.infsof.2015.10.008>
- Atlassian. (2023). *Escaping the black hole of technical debt* | atlassian. Descargado de <https://recibe.page.link/Atlassian23>
- Gartner, I. (2023). *Definition of technical debt - gartner information technology glossary*. Descargado de <https://recibe.page.link/Gartner23>
- ISO. (2014). *Iso 25000 square series*. Descargado de <https://recibe.page.link/ISO14>
- Katin, A., Lenarduzzi, V., Taibi, D., y Mandić, V. (2022). On the technical debt prioritization and cost estimation with sonarqube tool. , 302-309. doi: [https://doi.org/10.1007/978-3-030-97947-8\\_40](https://doi.org/10.1007/978-3-030-97947-8_40)
- Kruchten, P., Nord, R. L., y Ozkaya, I. (2012). Technical debt: From metaphor to theory and practice. *IEEE Software*, 29, 18-21. doi: <https://doi.org/10.1109/MS.2012.167>
- Li, Y., Soliman, M., y Avgeriou, P. (2022). Automatic identification of self-admitted technical debt from four different sources. *Empirical Software Engineering* 2023 28:3, 28, 1-38. doi: <https://doi.org/10.1007/s10664-023-10297-9>
- Mamun, M. A. A., Martini, A., Staron, M., Berger, C., y Hansson, J. (2019). Evolution of technical debt : An exploratory study. , 2476, 87-102. Descargado de [https://recibe .page.link/Mamun19](https://recibe.page.link/Mamun19)
- Pavlič, L., Hliš, T., Heričko, M., y Beranič, T. (2022). The gap between the admitted and the measured technical debt: An empirical study. *Applied Sciences* 2022, Vol. 12, Page 7482, 12, 7482. doi: <https://doi.org/10.3390/app12157482>
- Saarimaki, N., Baldassarre, M. T., Lenarduzzi, V., y Romano, S. (2019). On the accuracy of sonarqube technical debt remediation time. *2019 45th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, 317-324. doi: <https://doi.org/10.1109/SEAA.2019.00055>
- SONAR. (2023). *Technical debt | definition guide* | sonar. Descargado de [https://recibe .page.link/SONAR23](https://recibe.page.link/SONAR23)
- Stopford, B., Wallace, K., y Allspaw, J. (2017). Technical debt: Challenges and perspectives. *IEEE Software*, 34, 79-81. doi: <https://doi.org/10.1109/MS.2017.99>



Esta obra está bajo una licencia de Creative Commons Reconocimiento-NoComercial-CompartirIgual 2.5 México.

*Recibido 15 MAR. 2024*

*ReCIBE, Año 12 No.2, NOV. 2024*

*Aceptado 18 MAR. 2024*

**ENSEÑANZA DE PROGRAMACIÓN PARA NIÑOS Y  
JÓVENES CON DISCAPACIDAD VISUAL: MAPEO  
SISTEMÁTICO DE LA LITERATURA**

**TEACHING PROGRAMMING FOR CHILDREN AND YOUNG  
PEOPLE WITH VISUAL DISABILITY: SYSTEMATIC MAPPING OF  
LITERATURE**

Raúl A. Peredo Estudillo<sup>1</sup>  
[raulperestu@icloud.com](mailto:raulperestu@icloud.com)  
Juan C. Pérez Arriaga<sup>1</sup>  
Gerardo Contreras Vega<sup>1</sup>  
Ángel J. Sánchez García<sup>1</sup>

<sup>1</sup>Facultad de Estadística e Informática, Universidad Veracruzana, Xalapa, México

**Resumen:**

En la actualidad la educación de los estudiantes se enfoca en las disciplinas Ciencia, Tecnología, Ingeniería y Matemáticas (modelo integral STEM por sus siglas en inglés). Una estrategia efectiva para incorporar la ingeniería y la tecnología en la educación es promover el pensamiento computacional, donde la programación desempeña un papel fundamental en el desarrollo de habilidades lógicas y del pensamiento lógico estructurado. Sin embargo, es esencial destacar que las personas con discapacidad visual enfrentan desafíos específicos para adquirir estas competencias. El propósito de esta investigación es llevar a cabo un mapeo sistemático de la literatura con el fin de analizar los métodos, técnicas y herramientas utilizados para enseñar programación a niños y jóvenes con discapacidad visual. En este artículo, se exponen los resultados del mapeo sistemático de la literatura en el que se seleccionaron un total de 23 estudios. Los principales hallazgos incluyeron cinco métodos: aprendizaje háptico, resolución de casos, trabajo en equipo, enseñanza activa y enfoque basado en problemas. En cuanto a las técnicas, se identificaron cinco: basadas en tecnología, interactivas, basadas en juegos, en parejas y colaborativas. Para concluir, se encontraron cuatro tipos de herramientas: asistidas, hápticas basadas en audio y lectores de pantalla.

**Palabras Clave:** Discapacidad visual, Personas ciegas, Programación, Enseñanza, Accesibilidad, Mapeo sistemático.

**Abstract:**

Today's student education focuses on the disciplines of Science, Technology, Engineering and Mathematics (STEM) (STEM integrated model). Science, Technology, Engineering and Mathematics (STEM). model). An effective strategy for incorporating engineering and engineering and technology in education is to promote computational thinking, where programming programming plays a fundamental role in the development of logic and thinking skills. development of logical skills and structured logical thinking. Without However, it is essential to note that the visually impaired face specific challenges in acquiring these skills. specific challenges in acquiring these skills. The purpose of this The purpose of this research is to carry out a systematic mapping of the literature in order to analyze the methods, techniques and tools used to teach to analyze the methods, techniques and tools used to teach programming to visually impaired children and youth. programming to children and youth with visual impairment. In this article, the results of the the results of the systematic literature mapping in which a total of 23 studies were selected. a total of 23 studies were selected. The main findings included five methods: haptic learning, case solving, teamwork, active teaching and problem-based approach, active teaching and problem-based approach. In terms of techniques, five were five were identified: technology-based, interactive, game-based, paired and collaborative. and collaborative. In conclusion, four types of tools were found tools: assisted, audio-based haptic and screen readers.

**Keywords:** Visual impairment, Blind people, Programming, Teaching, Accessibility, Systematic mapping.

## 1. Introducción

La discapacidad visual se refiere a la pérdida parcial o total de la capacidad visual, evaluada en términos de la capacidad de lectura a diferentes distancias, el campo y la agudeza visual (World Health Organization: WHO, 2023). La Clasificación Internacional de Enfermedades 11 divide el deterioro de la visión en dos grupos: a distancia y cercano (Clasificación estadística internacional de enfermedades y problemas de salud relacionados (11ª ed.), 2019).

En el grupo de deterioro a distancia, se establecen niveles según la gravedad, desde suave hasta ceguera, basados en la agudeza visual. Para el deterioro de visión cercana, se utiliza la medida N6 o M.08 a 40 cm como criterio (Clasificación estadística internacional de enfermedades y problemas de salud relacionados (11ª ed.), 2019). En la actualidad, el campo de STEM (Ciencia, Tecnología, Ingeniería y Matemáticas) representa el 20% de los puestos de trabajo, con una cifra de 26 millones de empleos solo en los Estados Unidos (Delgado, 2022). El pensamiento computacional (PC) se define como el proceso mediante el cual se plantean problemas de tal manera que sus soluciones pueden representarse mediante secuencias de instrucciones y algoritmos (Wing, 2006).

La programación se convierte en una valiosa herramienta para cultivar este tipo de pensamiento lógico y estructurado, ayudando a desempeñar un papel crucial en STEM al proporcionar una base sólida para abordar problemas, desarrollar soluciones eficientes y aprovechar la tecnología de manera efectiva.

La problemática por tratar se relaciona con la falta de información y recursos disponibles para enseñar programación a personas con discapacidad visual. En la mayoría de los casos, los educadores y personas interesadas en brindar esta enseñanza se enfrentan a un desafío significativo debido a la ausencia de métodos, herramientas y técnicas de enseñanza adecuadas para este público. La falta de acceso a información relevante dificulta la inclusión de personas con discapacidad visual en el mundo de la programación, limitando sus oportunidades de desarrollo.

Por otro lado, los principales aportes de este estudio son fundamentales para fomentar el desarrollo del pensamiento lógico estructurado desde una edad temprana, a través de la programación. Esto no solo beneficia a las personas con discapacidad visual al abrir nuevas oportunidades educativas y profesionales, sino que también enriquece el campo de la programación al promover la diversidad y el acceso a talento variado.

Este mapeo sistemático de la literatura (MSL) se centra en métodos y herramientas de enseñanza en programación, buscando ampliar el acceso y promover la inclusión educativa para todas las personas, incluyendo aquellas con limitaciones o barreras, en busca de igualdad de oportunidades. El presente MSL está estructurado como se describe a continuación: La sección 2 presenta los antecedentes y trabajos relacionados. La sección 3 presenta el método de investigación empleado para la ejecución de esta investigación. La sección 4 presenta los resultados de la investigación. La sección 5 presenta la discusión. La sección 6 las amenazas a la validez y la sección 7 conclusiones.

## 2. Trabajo Relacionado

Un estudio realizado por Al-Ratta & Al-Khalifa (2013), explora la literatura sobre la enseñanza de la programación a personas ciegas, calificándola mediante una revisión sistemática y abordando temas como: necesidades específicas, desafíos en cursos convencionales y la posible necesidad de lenguajes de programación adaptados. El artículo establece dos tipos de cursos para alumnos ciegos: uno que utiliza manuales en Braille y otro que presenta instrucciones programadas en forma de audiolibros en casetes. Se observa que los estudiantes ciegos tienden a depender más de la experiencia concreta previa que de la abstracción, lo que dificulta seguir la lógica de programación. Además, se destaca que las personas con discapacidad visual no son una entidad homogénea, ya que cada individuo puede tener diferentes necesidades y habilidades. Por lo tanto, se sugiere

enfocarse en el desarrollo de herramientas que se adapten a diversas categorías de usuarios, lo que abre un nuevo campo de investigación.

Ying Li (2016), propone un estudio con el objetivo de explorar y presentar un enfoque educativo que utilice el pensamiento computacional como base para enseñar programación. Destaca la importancia del pensamiento computacional en la ciencia de la computación y enfatiza que los estudiantes no deben centrarse en la memorización de conocimientos o en imitar a las computadoras para resolver problemas, sino en desarrollar su propio pensamiento y conciencia a través del aprendizaje. El estudio subraya cómo este enfoque impacta tanto en el proceso de enseñanza como en el aprendizaje de los estudiantes, destacando la relevancia del pensamiento computacional en la educación en programación.

Nuestra investigación se basa en estos enfoques y los orienta hacia el ámbito académico con el propósito de fomentar el pensamiento lógico estructurado desde una edad temprana en niños con discapacidad visual. Además, incentivar el desarrollo de habilidades que les sean beneficiosas durante su proceso de formación educativa.

### 3. Método de Investigación

Para llevar a cabo esta investigación se siguió el método propuesto por Kitchenham et al. (2015) el cual se describe a continuación:

#### 3.1 Etapa de Planeación

Para esta investigación se definieron tres preguntas de investigación y se muestran en la Tabla 1.

Pregunta de investigación	Motivación
PI1. ¿Cuáles son los métodos y técnicas utilizados para enseñar programación a niños y jóvenes con discapacidad visual?	A partir de la identificación de métodos y técnicas efectivas para enseñar programación para este grupo de estudiantes, se pueden desarrollar enfoques adecuados que les permitan desarrollar habilidades en programación y fomentar su participación en el campo de las áreas STEM.
PI2. ¿Cuáles son las herramientas disponibles para la enseñanza de programación dirigidas a niños y jóvenes con discapacidad visual?	Al conocerlas herramientas disponibles y su utilidad, se podrán tomar decisiones informadas sobre qué recursos utilizar en entornos educativos y cuál puede ser el impacto de uso en su aprendizaje.
PI3. ¿Cuál es la efectividad de las técnicas, métodos y herramientas utilizadas en la enseñanza de programación para niños y jóvenes con discapacidad visual?	Permitirá identificar las mejores prácticas y enfoques que maximicen el aprendizaje y la inclusión de estos estudiantes. Asimismo, identificar parámetros o formas de medición de efectividad en esta área de investigación para ser aplicada en futuras investigaciones.

Tabla 1. Preguntas de Investigación

De las fuentes sugeridas por la guía de Kitchenham et al. (2015) se escogieron IEEEExplore, ACM Digital Library, Science Direct y Springer Link y MDPI como base de datos complementaria. Las fuentes seleccionadas se pueden visualizar en la Tabla 2.

Fuente	Sitio web
IEEEExplorer	<a href="https://ieeexplore.ieee.org/Xplore/home.jsp">https://ieeexplore.ieee.org/Xplore/home.jsp</a>
ACM Digital Library	<a href="https://dl.acm.org/">https://dl.acm.org/</a>
Science Direct	<a href="https://www.sciencedirect.com/">https://www.sciencedirect.com/</a>
Springer Link	<a href="https://link.springer.com/">https://link.springer.com/</a>
MDPI	<a href="https://www.mdpi.com">https://www.mdpi.com</a>

Tabla 2. Fuentes de información seleccionadas

Para la generación de la cadena de búsqueda, se identificaron los términos claves para la búsqueda de artículos. Los términos definidos para esta RSL se muestran en la Tabla 3.

Término clave	Término en inglés	Término relacionado
Discapacidad visual	Visual Impairment	Blind people
Enseñanza	Teaching	Learning
Programación	Programming	Coding

Tabla 3. Términos clave de búsqueda

La cadena de búsqueda representa la entrada para recuperar estudios relevantes de las fuentes de datos mostradas en la Tabla 2. Se generaron diversas cadenas de búsqueda a partir de las palabras de la Tabla 3 enlazados con operadores booleanos (OR y AND), y se seleccionó aquella que obtuvo un mejor desempeño, en términos de *precision* y *recall*.

La métrica *precision* es la proporción de documentos recuperados que son relevantes para la consulta, véase la ecuación (1); por su parte, la métrica *recall* es la proporción de documentos relevantes que se recuperan con éxito, véase la ecuación (2).

$$Precision = \frac{\text{Documentos relevantes recuperados}}{\text{Total de documentos recuperados}} \quad (1)$$

$$Recall = \frac{\text{Documentos relevantes recuperados}}{\text{Total de documentos relevantes}} \quad (2)$$

La Tabla 4 muestra la cadena de búsqueda aplicada, con su respectiva medida de *recall* y *precision*.

Cadena de búsqueda	Valor de <i>recall</i>	Valor de <i>precision</i>
("Visual impairment" OR "Visual disability" OR "Blind Students" OR "Visually Impaired") AND ("Teaching" OR "Learning") AND ("Programming" OR "Coding")	0.1	0.02

Tabla 4. Cadenas de búsqueda aplicadas por fuente de datos

### 3.2 Etapa de Conducción

En este apartado se describen los criterios de selección de los artículos encontrados con las cadenas de búsqueda y se explica el proceso de selección de los estudios primarios que serán analizados.

#### 3.2.1 Criterios de selección

Se propusieron cinco criterios de inclusión (CI) y dos criterios de exclusión CE, para seleccionar los estudios primarios de los estudios recuperados en las fuentes seleccionadas, véase Tablas 5 y 6, respectivamente.

Clave	Descripción
CI1	La publicación debe estar escrita en el idioma inglés.
CI2	La fecha de publicación es de 2015 - 2023.
CI3	La publicación debe ser artículo de investigación sobre software (Artículo de Journal, conferencia, revista).
CI4	El título o abstract de la publicación debe dar indicios de responder al menos una pregunta de investigación.
CI5	Una lectura completa de la publicación debe responder al menos una pregunta de investigación.

Tabla 5. Criterios de inclusión

Clave	Descripción
CE1	Estudios que son carteles, presentaciones, resúmenes o tutoriales.
CE2	No se tiene acceso completo a la publicación.
CE3	Estudios duplicados.

Tabla 6. Criterios de exclusión

#### 3.2.2 Proceso de selección

Se definieron cuatro etapas para el proceso de selección de estudios. La Figura 1 muestra los criterios aplicados en cada etapa.

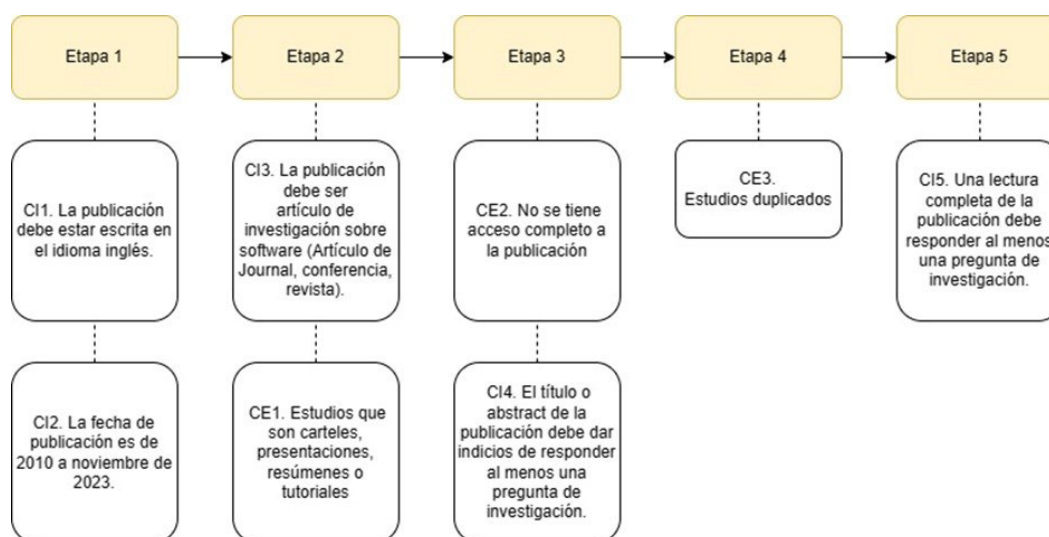


Figura 1. Proceso de selección de estudios primarios.

La Tabla 7 muestra el número de estudios primarios seleccionados por etapa, donde se puede observar que IEEEExplore seleccionó la mayor cantidad de estudios.

Fuente	Resultados	Etapa 1	Etapa 2	Etapa 3	Etapa 4	Etapa 5
IEEEExplore	361	268	263	32	32	19
ACM Digital Library	1605	1084	780	10	2	1
MDPI	175	143	132	11	3	2
Science Direct	175	3999	113	6	0	0
Springer Link	2284	1628	554	65	2	1
<b>Total</b>	<b>12188</b>	<b>7122</b>	<b>1842</b>	<b>124</b>	<b>39</b>	<b>23</b>

Tabla 7. Número de estudios primarios seleccionados por etapa.

### 3.3 Síntesis

La extracción de datos fue realizada con base en las preguntas de investigación, identificando 23 artículos que las contestan y categorizando estos hallazgos por métodos, técnicas, herramientas y efectividad. Para elaborar la síntesis narrativa se utilizó la metodología propuesta por Popay et al. (2006). Esta metodología está dirigida a las revisiones sistemáticas que se ocupan de la efectividad de alguna intervención o de los factores que incluyen la implementación de las intervenciones. El proceso de síntesis, así como las referencias de los estudios primarios analizados se encuentra en [https://uvmx-my.sharepoint.com/:x/g/personal/zs19014050\\_estudiantes\\_uv\\_mx/Ef0MGoxl9pxGqh8hSXDyUxMB5VffUplw-cGQ95sWm4JRA?rttime=Mevjm08d3Eg](https://uvmx-my.sharepoint.com/:x/g/personal/zs19014050_estudiantes_uv_mx/Ef0MGoxl9pxGqh8hSXDyUxMB5VffUplw-cGQ95sWm4JRA?rttime=Mevjm08d3Eg)

### 4. Resultados

El año con mayor número de artículos publicados fue 2022, con 5 artículos. Durante los años 2015 hasta el 2018 se obtuvo una media de 3 artículos por año y a partir del 2019 al 2021 se detectó una disminución en la cantidad de publicaciones de menos de 2 artículos por año. La Figura 2 muestra la distribución de los estudios primarios encontrados por fuente de publicación y la Figura 3 muestra la distribución de los estudios por año de publicación.

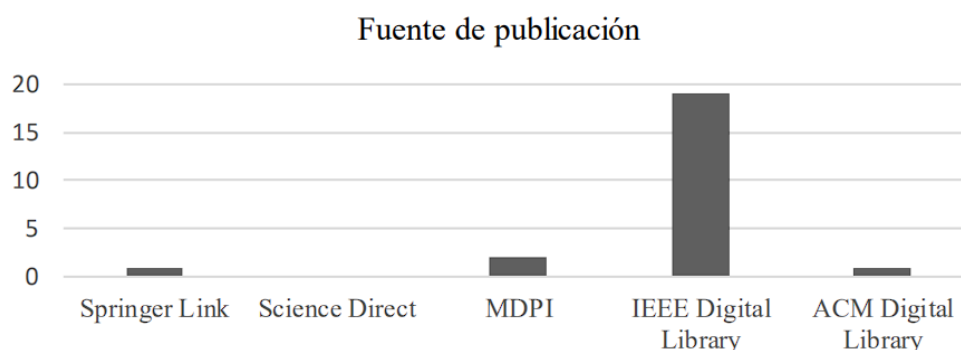


Figura 2. Distribución de estudios por fuente de publicación.



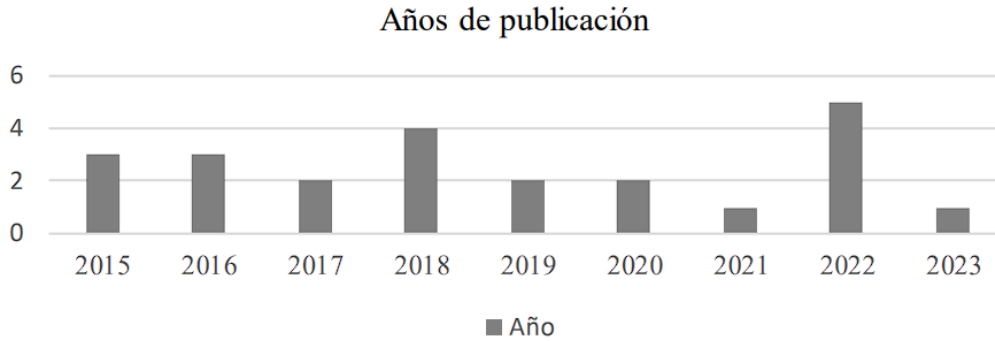


Figura 3. Distribución de estudios por año de publicación.

Para el proceso de síntesis de la información se optó por la estrategia de síntesis narrativa. Este tipo de síntesis descrita por Popay et al. (2006) implica integrar los hallazgos de múltiples estudios en una narrativa coherente que resuma la evidencia disponible sobre nuestro tema de interés. A continuación, se presentan las respuestas a cada pregunta de investigación.

#### 4.1 PI1. ¿Cuáles son las técnicas y métodos de enseñanza que sirven para enseñar programación a niños y jóvenes con discapacidad visual?

De los 23 estudios revisados, se identificaron 5 diferentes métodos de aprendizaje, el aprendizaje háptico, la resolución de casos, el trabajo en equipo, la enseñanza activa y el enfoque basado en problemas como se muestra en la Figura 4. Los métodos de resolución de casos, trabajo en equipo, enseñanza activa y enfoque basado en problemas tuvieron una menor frecuencia de aparición, todos encontrados en el estudio S23.

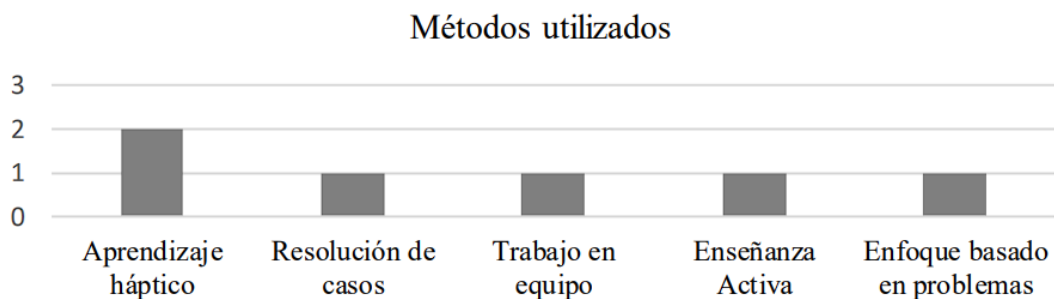


Figura 4. Frecuencia de métodos.

El aprendizaje háptico presenta una mayor frecuencia en los resultados, ya que se basa en el uso de materiales físicos o manipulativos para facilitar el aprendizaje y la comprensión de conceptos abstractos o complejos [10]. En el estudio S02, se utilizan bloques físicos 2D para enseñar programación de forma táctil. Una cámara captura y procesa la información, luego un compilador traduce el código, y los resultados se presentan mediante voz.

Por otro lado, los métodos de enseñanza con una mejor frecuencia fueron el enfoque de enseñanza activa y el enfoque basado en problemas. La enseñanza activa implica la participación de los estudiantes en el proceso de aprendizaje, fomentando su implicación, reflexión y construcción de conocimientos (Torres et al., 2014). Por su parte, el enfoque basado en problemas se basa en presentar a los estudiantes situaciones complejas o desafiantes que requieren su análisis crítico, investigación y búsqueda de soluciones (Labra et al., 2011).

Además, se identificaron cinco técnicas de enseñanza en cuatro artículos diferentes. Las técnicas de aprendizaje colaborativo y la técnica de aprendizaje interactivo y envolvente fueron las más mencionadas, como se muestra en la Figura 4. El aprendizaje háptico presenta una mayor frecuencia en los resultados, ya que se basa en el uso de materiales físicos o manipulativos para facilitar el aprendizaje y la comprensión de conceptos abstractos o complejos (Torres et al., 2014). En el estudio S02, se utilizan bloques físicos 2D para enseñar programación de forma táctil. Una cámara captura y procesa la información, luego un compilador traduce el código, y los resultados se presentan mediante voz.

Por otro lado, los métodos de enseñanza con una mejor frecuencia fueron el enfoque de enseñanza activa y el enfoque basado en problemas. La enseñanza activa implica la participación de los estudiantes en el proceso de aprendizaje, fomentando su implicación, reflexión y construcción de conocimientos (Del Carmen Bernal González & Dueñas, 2009). Por su parte, el enfoque basado en problemas se basa en presentar a los estudiantes situaciones complejas o desafiantes que requieren su análisis crítico, investigación y búsqueda de soluciones (Labra et al., 2011).

Además, se identificaron cinco técnicas de enseñanza en cuatro artículos diferentes. Las técnicas de aprendizaje colaborativo y la técnica de aprendizaje interactivo y envolvente fueron las más mencionadas, como se muestra en la Figura 5.

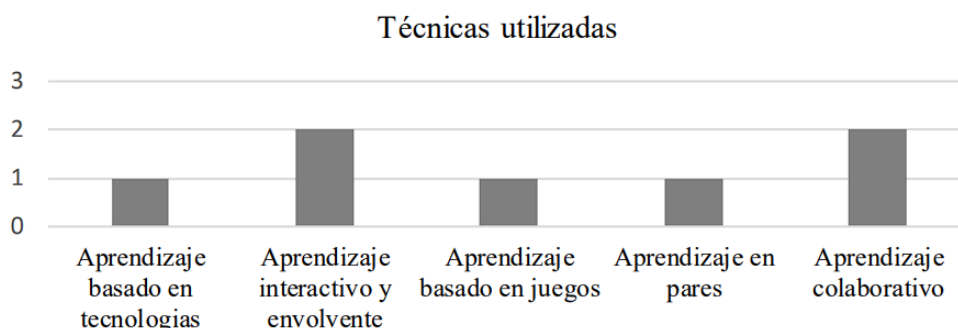


Figura 5. Frecuencia de técnicas.

El aprendizaje colaborativo promueve la construcción conjunta del conocimiento, alentando la interacción y cooperación en grupos de estudiantes. Se basa en la idea de que trabajar juntos, compartir ideas y resolver problemas facilita el aprendizaje (Chew et al., 2020). Un ejemplo concreto de esta técnica es la enseñanza en pares utilizada en el artículo S23 donde un estudiante universitario de informática enseña conceptos y prácticas de programación a un estudiante de secundaria con discapacidad visual utilizando lectores de pantalla y software de programación adaptados.

La segunda técnica de aprendizaje que más se reportó fue la técnica de aprendizaje interactivo y envolvente. Esta técnica se basa en la premisa de que los estudiantes aprenden mejor cuando están activamente comprometidos y participan de manera activa en su entorno de aprendizaje (Freeman et al., 2014). El artículo S11 destaca por su enfoque en promover la interacción con materiales educativos para estudiantes con discapacidad visual. En este estudio, los alumnos utilizaban bloques físicos para identificar variables y luego los unían de manera secuencial para representar declaraciones lógicas.

4.2 PI2. ¿Cuáles son las herramientas disponibles para la enseñanza de programación dirigidas a niños y jóvenes con discapacidad visual?

Una herramienta es un dispositivo que realiza o ayuda en el desempeño de tareas de procesos de usuario u organización que respaldan, directa o indirectamente, el logro de objetivos de producción («ISO/IEC/IEEE International Standard - Systems and software engineering--Vocabulary», 2017). Puede ser tanto física como digital.

La figura 6 muestra la frecuencia por tipo de herramienta donde el lector de pantalla fue la herramienta más mencionada durante la recopilación de información estando presente en los estudios S3, S4, S6, S10, S14, S15, S17, S18, S21, S22, S23 es una herramienta de software ampliamente utilizada que brinda asistencia a las personas con discapacidad visual al proporcionarles información auditiva sobre el contenido mostrado en la pantalla de un dispositivo electrónico (Hayward et al., 2004).

El segundo tipo de herramienta más reportada fue la háptica. Las herramientas hápticas juegan un papel fundamental al proporcionar una experiencia táctil y sensorial que compensa la falta de visión. Estas herramientas permiten a las personas con discapacidad visual explorar y percibir el entorno de manera más completa, brindándoles información táctil y sensorial que les ayuda a comprender y navegar por su entorno estando presente en los estudios S7, S9, S10, S12, S17, S18, S20, S21, S23.

En las figuras 7, 8, 9 y 10 se puede apreciar con mayor detalle qué herramientas se encontraron en cada categoría.

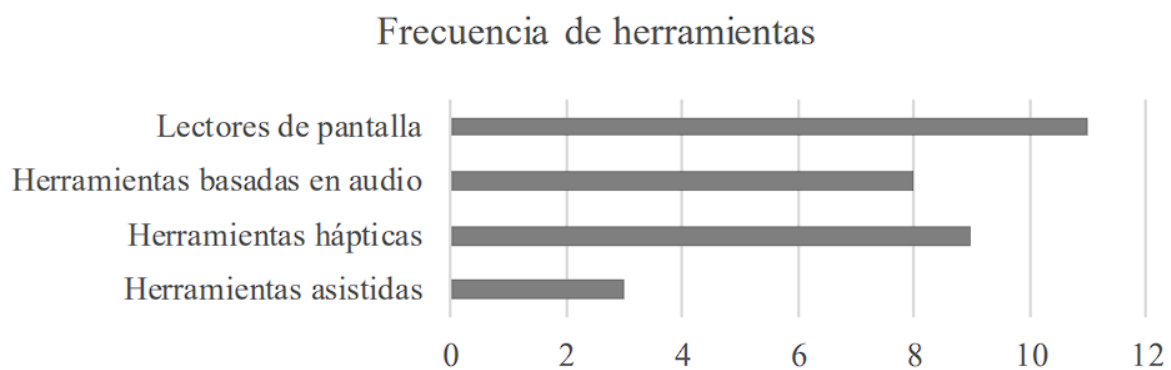


Figura 6. Frecuencia de herramientas.

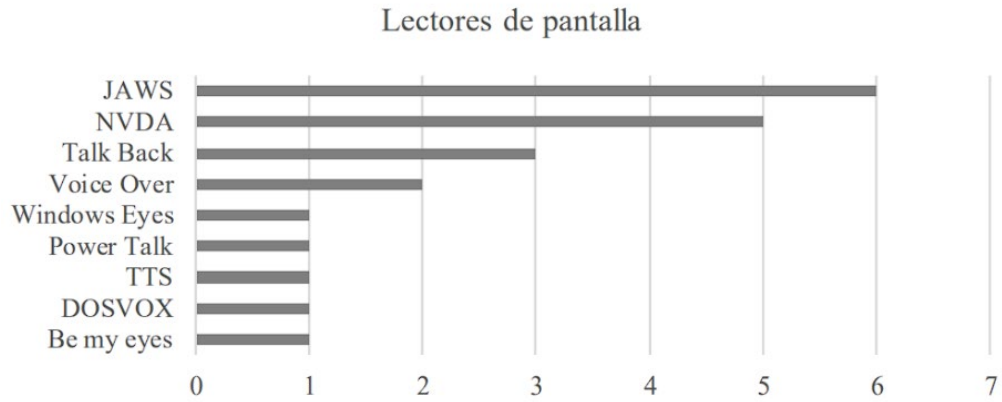


Figura 7. Frecuencia lectores de pantalla.

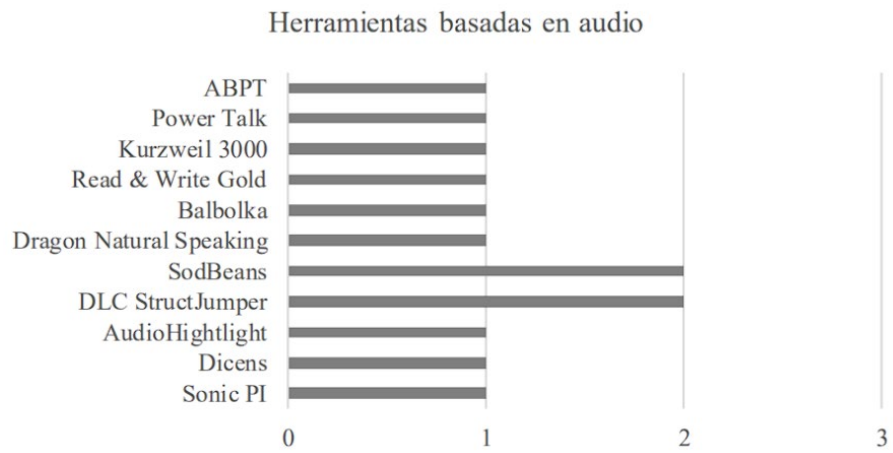


Figura 8. Frecuencia herramientas basadas en audio.

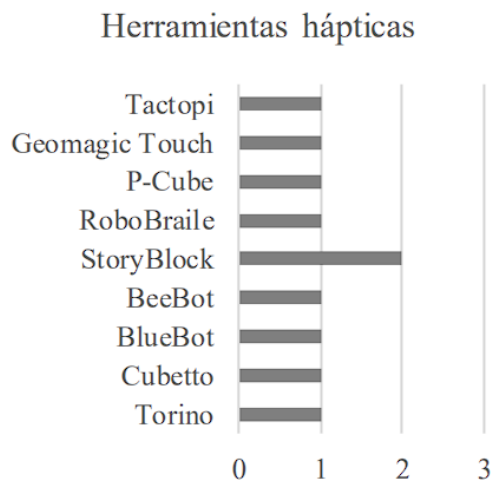


Figura 9. Frecuencia herramientas hápticas.

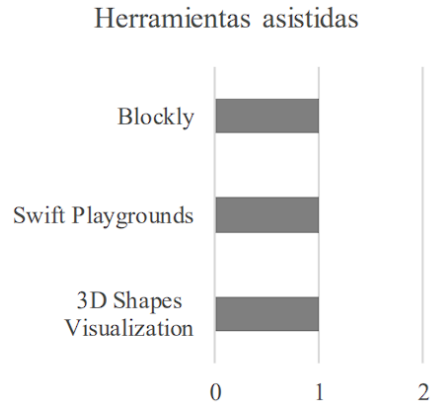


Figura 10. Frecuencia herramientas asistidas.

4.2 PI3. ¿Cuál es la efectividad de las técnicas, métodos y herramientas de enseñanza de programación en niños y jóvenes con discapacidad visual?

La efectividad se refiere a la capacidad de lograr resultados deseados u objetivos 7 establecidos. Es utilizada para evaluar y medir el grado de éxito o eficacia en la consecución de metas específicas («ISO/IEC/IEEE International Standard - Systems and software engineering--Vocabulary», 2017).

La evaluación de la efectividad se vio obstaculizada debido a la falta de pruebas realizadas por muchos autores con relación a los métodos, técnicas o herramientas que se analizaron. Esto dificulta el establecimiento de parámetros precisos para la medición de la efectividad. No obstante, en los estudios S9, S12 y S23 se empleó el tiempo de resolución de problemas como indicador de efectividad. Aunque este enfoque puede ser válido en ciertos casos, es esencial recordar que existen otros factores que pueden influir en la eficacia de un método, técnica o herramienta.

La comprensión se refiere a la habilidad de entender y procesar la información que se presenta a través de distintas formas de comunicación, como el lenguaje oral o escrito, imágenes, gestos y otros medios (Perkins, 1999). Cuando se evalúa la comprensión, es crucial adoptar enfoques personalizados que se adapten a las necesidades y capacidades individuales. Para medirla, existen diversas estrategias, como la evaluación basada en el rendimiento, que se utilizó en los estudios S2, S20 y S23.

En la Figura 11, se pueden observar la frecuencia de los parámetros mencionados en los artículos.

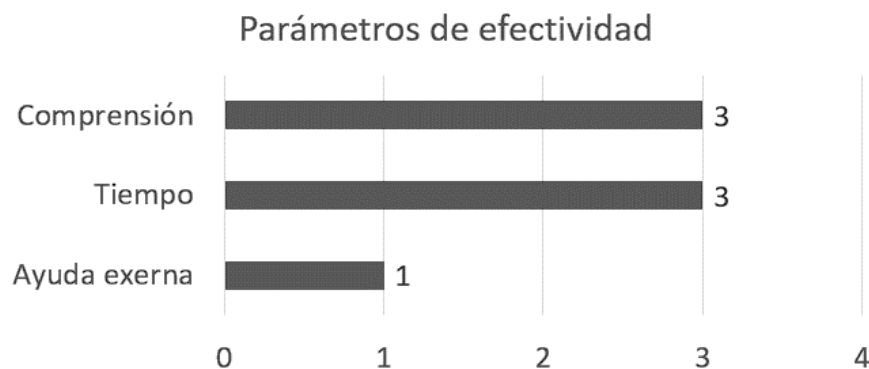


Figura 11. Frecuencia parámetros de efectividad.

## 5. Amenazas a la validez

En los resultados de esta investigación se identificaron algunas amenazas, como la falta de acceso a la mayoría de los documentos completos. La búsqueda en un número reducido de fuentes aumenta el riesgo de que la muestra de literatura revisada no sea representativa o completa. Otras posibles amenazas incluyen la posibilidad de que los autores que llevaron a cabo las prácticas hayan omitido información relevante, lo que podría conducir a la alteración de los resultados. Estas amenazas se mitigaron mediante el uso de formatos para la síntesis, validaciones de la cadena de búsqueda, y ampliando los años de investigación para capturar un mayor número de estudios primarios.

## 6. Conclusiones y trabajo futuro

En conclusión, la investigación destaca la importancia de ampliar los métodos de enseñanza para niños y jóvenes con discapacidad visual. Se han identificado enfoques prometedores, como el enfoque háptico, el aprendizaje colaborativo y el aprendizaje interactivo. Además, se ha observado que herramientas como los lectores de pantalla y las herramientas hápticas ofrecen beneficios significativos en el proceso de aprendizaje. Aunque la evaluación de la efectividad plantea desafíos, se sugiere utilizar el tiempo de resolución de problemas como un indicador inicial para evaluar los resultados. Es necesario seguir investigando y desarrollando estrategias educativas inclusivas que faciliten un aprendizaje efectivo y significativo en programación desde una edad temprana para todos los estudiantes con discapacidad visual. Como parte de trabajos futuros, se espera que esta investigación sirva como base para el desarrollo de conocimiento en este ámbito.

## Referencias

World Health Organization: WHO. (2023, 10 agosto). *Ceguera y discapacidad visual*. <https://www.who.int/es/news-room/fact-sheets/detail/blindness-and-visual-impairment>

*Clasificación estadística internacional de enfermedades y problemas de salud relacionados (11ª ed.)*. (2019). Organización Mundial de la Salud. <https://icd.who.int/en>

Delgado, P. (2022, 3 noviembre). *Educación STEM: ¿Qué es y cómo sacarle provecho?* Observatorio / Instituto para el Futuro de la Educación. <https://observatorio.tec.mx/edu-news/educacion-stem-que-es-y-como-sacarle-provecho/>

Wing, J. M. (2006). Computational thinking. *Communications of The ACM*, 49(3), 33-35. <https://doi.org/10.1145/1118178.1118215>

Al-Ratta, N. M., & Al-Khalifa, H. S. (2013). Teaching programming for blinds: A review. *Fourth International Conference on Information and Communication Technology and Accessibility*. <https://doi.org/10.1109/icta.2013.6815285>

Li, Y. (2016). Teaching programming based on Computational Thinking. *2016 IEEE Frontiers in Education Conference*. <https://doi.org/10.1109/fie.2016.7757408>

Kitchenham, B., Budgen, D., & Brereton, P. (2015). Evidence-Based software engineering and systematic reviews. En *Chapman and Hall/CRC eBooks*. <https://doi.org/10.1201/b19467>

Zhang, H., Babar, M. A., & Tell, P. (2011). Identifying relevant studies in software engineering. *Information & Software Technology*, 53(6), 625-637. <https://doi.org/10.1016/j.infsof.2010.12.010>

Popay, J., Roberts, H.M., Sowden, A.J., Petticrew, M., Arai, L., Rodgers, M., & Britten, N. (2006). Guidance on the conduct of narrative synthesis in systematic Reviews. A Product from the ESRC Methods Programme. Version 1.

Torres Cosío, Verónica; Vázquez Alejandre, Emilia Jazmín; Cossio Franco, Edgar (2014). Accesibilidad de los materiales educativos a través de la percepción háptica. En: No Solo Usabilidad, nº 13, 2014. <nosolousabilidad.com>. ISSN 1886-8592

Del Carmen Bernal González, M., & Dueñas, M. S. M. (2009). METODOLOGÍAS ACTIVAS PARA LA ENSEÑANZA y EL APRENDIZAJE. *Revista panamericana de pedagogía*. <https://doi.org/10.21555/rpp.v0i14.1790>

Labra, P., Kokaly, M. E., Herrera, C. I., Concha, A., Sasso, P., & Vergara, M. (2011). El enfoque ABP en la formación inicial docente de la Universidad de Atacama: el impacto en el quehacer docente. *Estudios pedagógicos*, 37(1), 167-185. <https://doi.org/10.4067/s0718-07052011000100009>

Chew, S. W., Huang, X., Hsu, F., & Chen, N. (2020). Enhancing Critical Thinking Skills of Elementary School Students through Collaborative Learning. *2020 IEEE 20th International Conference on Advanced Learning Technologies (ICALT)*. <https://doi.org/10.1109/icalt49669.2020.00082>

Freeman, S., Eddy, S. L., McDonough, M. J., Smith, M. K., Okoroafor, N., Jordt, H., & Wenderoth, M. P. (2014). Active learning increases student performance in science, engineering, and mathematics. *Proceedings of the National Academy of Sciences of the United States of America*, 111(23), 8410-8415. <https://doi.org/10.1073/pnas.1319030111>

"ISO/IEC/IEEE International Standard - Systems and software engineering--Vocabulary," in *ISO/IEC/IEEE 24765:2017(E)*, vol., no., pp.1-541, 28 Aug. 2017, doi: 10.1109/IEEESTD.2017.8016712.

Hayward, V., Astley, O. R., Cruz-Hernandez, M., Grant, D. F., & Robles-De-La-Torre, G. (2004). Haptic interfaces and devices. *Sensor Review*, 24(1), 16-29. <https://doi.org/10.1108/02602280410515770>

"ISO/IEC/IEEE International Standard - Systems and software engineering--Vocabulary," in *ISO/IEC/IEEE 24765:2017(E)*, vol., no., pp.1-541, 28 Aug. 2017, doi: 10.1109/IEEESTD.2017.8016712.

Perkins, D. (1999). ¿Qué es la comprensión? En W. Stone (Comp.), *La Enseñanza para la Comprensión: vinculación entre la investigación y la práctica*. Buenos Aires, Argentina: Paidós.

## Notas Biográficas

	<p><b>Raúl Arturo Peredo-Estudillo</b> es estudiante de la Licenciatura en Ingeniería de Software de la Facultad de Estadística e Informática de la Universidad Veracruzana. Sus intereses principales son el desarrollo móvil Android y IOS, diseño de interfaces UI y UX y por temas de investigación enfocados hacia la accesibilidad.</p>
	<p><b>Juan Carlos Pérez-Arriaga</b> es Maestro en Ciencias de la Computación por la Fundación Arturo Rosenblueth. Profesor de la Facultad de Estadística e Informática de la Universidad Veracruzana, desde el 2006. Sus intereses incluyen seguridad en el desarrollo de software y desarrollo de software accesible.</p>
	<p><b>Gerardo Contreras Vega</b> es Licenciado en Informática, cuenta con una maestría en Ciencias de la Computación y estudia el Doctorado en Tecnologías de la Información. Cuenta con experiencia de 27 años en la Universidad Veracruzana, donde ha laborado como Técnico Académico y Profesor de Tiempo Completo. Sus áreas de interés son la seguridad de la información, el manejo de servicios e infraestructura de red y la accesibilidad para personas con discapacidad. Disfruta de utilizar y promover el software libre.</p>
	<p><b>Ángel J. Sánchez-García</b> es doctor en Inteligencia Artificial por la Universidad Veracruzana en 2018. Desde 2017 es profesor de tiempo completo de la Facultad de Estadística e Informática, Universidad Veracruzana. Sus áreas de investigación de interés son Aprendizaje Máquina, Minería de datos, Estimación en Ingeniería de Software.</p>





*Recibido 18 OCT. 2023*

*ReCIBE, Año 12 No.2, NOV. 2023*

*Aceptado 18 ENE. 2024*

## **Efecto de la oscilación térmica diaria sobre celdas de carga de galgas extensiométricas genéricas**

**Effect of Daily Thermal Oscillation on Generic Strain Gauge Load Cells**

**Rubén Esparza Orozco<sup>1</sup>**

eoruben9006@gmail.com

**Abel Quevedo Nolasco<sup>1</sup>**

**Carlos Ramírez Ayala<sup>1</sup>**

**Roberto Ascencio Hernández<sup>1</sup>**

**Bertha Patricia Zamora Morales<sup>2</sup>**

<sup>1</sup>Colegio de Postgraduados

<sup>2</sup>Instituto Nacional de Investigaciones Forestales, Agrícolas y Pecuarias.

## RESUMEN

El objetivo de esta investigación fue evaluar el efecto de la oscilación térmica diaria en sobre tres celdas de carga de galgas extensiométricas (C1, C2 y C3) como elemento sensible de instrumentos de pesaje. Para ello se fabricaron tres básculas digitales con capacidad nominal de 5 kg, equipadas individualmente con un dispositivo electrónico que cuenta con un sistema de reloj, un termómetro y un sistema de almacenamiento de datos. Las básculas fueron dispuestas dentro de un invernadero del Colegio de Postgraduados, Campus Montecillos, del 18 de mayo al 25 de mayo durante 164 horas consecutivas. A cada báscula se aplicó una carga sólida de 5 kg de manera sostenida para registrar las series de tiempo de oscilación térmica y error observado en intervalos de una hora. Ambas series mostraron una relación directa lo que permitió aplicar modelos para compensar el error que registró cada sensor. Los modelos permitieron reducir los errores promedio de 9.79 g, 8,40 g y 11.91 g a 1.39 g, 0.4 g y 1.13 g, respectivamente para cada celda de carga.

**Palabras clave:** celdas de carga, oscilación térmica, error, modelos de compensación.

## ABSTRACT

The objective of this research was to assess the effect of daily thermal oscillation on three strain gauge load cells (C1, C2, and C3) as the sensitive element of weighing instruments. Three digital scales with a nominal capacity of 5 kg were manufactured, each equipped with an electronic device featuring a clock system, a thermometer, and a data storage system. The scales were placed inside a greenhouse at the Colegio de Postgraduados, Montecillos Campus, from May 18 to May 25, continuously for 164 hours. A sustained 5 kg load was applied to each scale to record the time series of thermal oscillation and observed error at one-hour intervals. Both series showed a direct relationship, allowing the application of models to compensate for the error recorded by each sensor. The models reduced the average errors of 9.79 g, 8.40 g, and 11.91 g to 1.39 g, 0.4 g, and 1.13 g, respectively, for each load cell.

**Keywords:** load cells, thermal oscillation, error, compensation models.

## 1. INTRODUCCIÓN

El Proyecto de Norma Oficial Mexicana PROY-NOM-010-SCFI-2014, presenta las definiciones técnicas, requisitos técnicos, procedimientos de clasificación y evaluación de instrumentos para pesar de funcionamiento no automático. De dicho documento se extraen y puntualizan los siguientes conceptos.

- “Instrumento para pesar: Instrumento de medición para determinar la masa de un cuerpo utilizando la acción de la gravedad sobre este cuerpo. Este instrumento también puede utilizarse para determinar otras magnitudes, cantidades, parámetros o características relacionadas con la masa”.
- “Instrumento para pesar de funcionamiento no automático: Instrumento que requiere la intervención de un operador durante el proceso de pesada para decidir si el resultado de pesada es aceptable”.
- “Error de indicación: indicación de un instrumento menos el valor convencionalmente verdadero de la masa correspondiente”.
- “División de escala ( $d$ ): diferencia entre dos valores consecutivos, para una indicación digital, expresado en unidades de masa”.
- “División de escala de verificación ( $e$ ): valor expresado en unidades de masa, utilizado para la clasificación y verificación de un instrumento para pesar.

Los instrumentos para pesar electrónicos comúnmente utilizan celdas de galgas extensiométricas como elemento sensible. Las galgas extensiométricas se disponen en un circuito eléctrico conocido como puente de Wheatstone, en el que actúan como resistencias variables cuyo valor es proporcional a la carga que se aplica sobre la celda de carga. Si la celda de carga esta en reposo, el voltaje del puente es 0 V; al aplicar una carga, la resistencia de la galga extensiométrica cambia y se genera un voltaje en el puente para reequilibrar el circuito. Los cambios de voltaje son enviados a un micro controlador para su transducción a unidades de peso (Dong & Hansen, 2023; McCauley & Nackley, 2022).

Adicionalmente, de acuerdo con el PROY-NOM-010-SCFI-2014 las pruebas de desempeño deben realizarse a una temperatura ambiente estable. La temperatura es considerada estable siempre que la velocidad de variación no sobrepase los 5°C por hora. También se especifica que las observaciones de error deben hacerse un periodo máximo de 30 minutos. Sin embargo, en algunas aplicaciones se requiere que un instrumento para pesar opere de forma continua bajo condiciones de oscilación térmica natural o controlada por un periodo indeterminado de tiempo (días, semanas, meses y hasta años). Tal es el caso de los lisímetros de pesada, en los que el cambio de masa permite determinar la evapotranspiración de cultivo). Varios investigadores han propuesto diferentes diseños de lisímetros de pesada empleando para ellos instrumentos para pesar de funcionamiento automático de principio electrónico (Beeson, 2011; McCauley et al., 2021).

Evidentemente, las celdas de carga son sensibles a la oscilación térmica, además de que dicha sensibilidad es variable entre ellas, aun siendo de las mismas características. La sensibilidad se manifiesta con una desviación positiva con el aumento de temperatura y una desviación negativa con el descenso de temperatura; es decir, respectivamente, una aparente adición o sustracción de masa del receptor de carga del instrumento (Beeson, 2011).

El objetivo de esta investigación fue observar y compensar los errores registrados en cada una de las celdas de carga derivados de la oscilación térmica diaria dentro de un invernadero del Colegio de Postgraduados, Campus Montecillos.

## **2. METODOLOGÍA**

Para observar la sensibilidad de las celdas de carga a la temperatura se fabricaron tres básculas digitales de funcionamiento automático con capacidad nominal de 5 kg equipadas individualmente con un dispositivo electrónico que cuenta con un sistema de reloj, un termómetro y un sistema de almacenamiento de datos.

Previamente al arranque del funcionamiento automático de las básculas se realizaron las pruebas de desempeño indicadas en el PROY-NOM-010-SCFI-2014 requeridas para instrumentos para pesar de funcionamiento no automático para verificar el correcto funcionamiento de las mismas. Para ello se seleccionó una división de escala ( $d$ ) y una división de escala de verificación ( $e$ ) igual a 1g. Se efectuaron las pruebas de desempeño: prueba de pesada, repetitividad y retorno a cero.

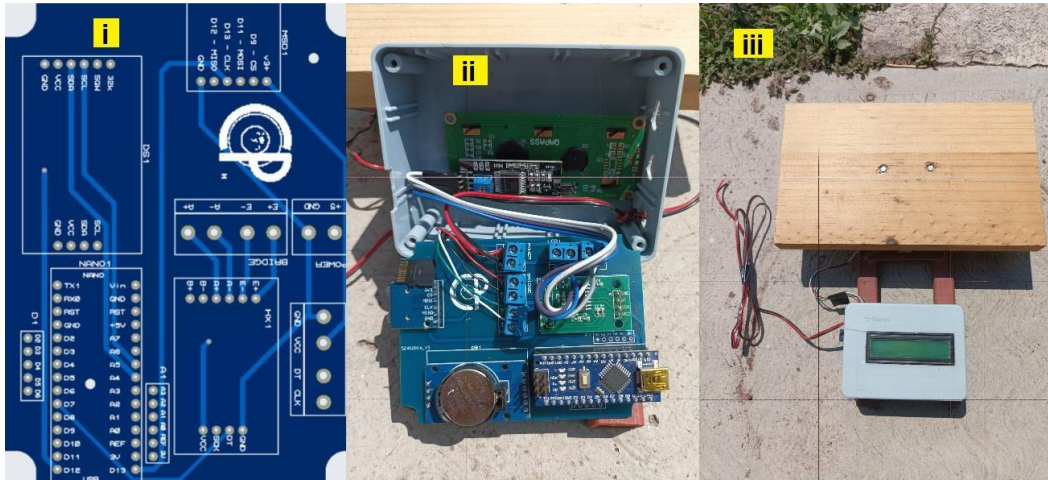
Posteriormente, las básculas fueron dispuestas dentro de un invernadero del Colegio de Postgraduados, Campus Montecillo, y a cada una se aplicó una carga sólida de 5 kg de manera sostenida para registrar las series de tiempo de oscilación térmica y error registrado en intervalos de una hora durante 6 días.

### **2.1. Fabricación de los instrumentos para pesar**

Cada báscula cuenta se equipó con un dispositivo electrónico que integra: un micro controlador Arduino y un amplificador HX711 para la transducción de la señal de la celda de carga; un módulo reloj, un módulo mSD con una tarjeta mSD de 16 GB para el almacenamiento de datos y display LCD 16x2 para el mostrar información al usuario. Los componentes fueron montados sobre una tarjeta PCB y fueron resguardados dentro de un gabinete plástico (Figura 1).

La calibración de las celdas de carga consistió en medir el voltaje que se crea en el puente Wheatstone al momento de aplicar una masa calibrada y conocida por el usuario. A esta relación se le conoce como factor de carga y es característico de cada celda de carga. Para ello se utilizó el programa "Calibration" de la librería HX711 sobre la IDE Arduino (Kallhovd, 2022).

El código de trabajo del sistema de pesaje se desarrolló con base en los ejemplos incluidos en la librería HX711 (Kallhovd, 2022). En él, se indicó el factor de celda que se obtuvo durante la calibración de cada celda de carga y se desarrollaron las líneas de código para habilitar los módulos: reloj, almacenamiento de datos y salida de datos por la pantalla LCD.



**Figura 1.** Instrumento para pesar de celda de carga. (i) la tarjeta PCB para el montaje y conexión de los componentes electrónicos, (ii) dispositivo electrónico, y (iii) la integración del instrumento para pesar.

## 2.2. Descripción de las pruebas de desempeño de funcionamiento no automático

- Pesada: Aplicar cargas calibradas a partir de cero de manera creciente hasta la capacidad Máx del instrumento; similarmente las cargas deben retirarse hasta regresar a cero. El error observado en cada carga no debe ser superior al error máximo permitido para esa carga.
- Prueba de retorno a cero: Una carga de por lo menos  $1/3$  la capacidad máxima del instrumento debe permanecer sobre la plataforma de carga por media hora. Al retirar la carga el error desde cero no debe ser superior a  $0.5e$ .
- Repetividad: Una misma carga debe aplicarse y retirarse en al menos cinco repeticiones. La diferencia en las cifras registradas no deber ser superior al error máximo permitido para esa carga.

### 2.3. Registro de series de tiempo error y temperatura en funcionamiento automático

El registro de las series de tiempo error y temperatura se inició a las 16:00 hrs del día 18 de mayo de 2023. Los registros se hicieron en intervalos de 1 hora y fueron registrados automáticamente las memorias mSD. Las tres básculas operaron de manera automática e ininterrumpida hasta las 8:00 horas del día 25 de mayo de 2023, 164 horas totales.

## 3. RESULTADOS Y DISCUSIÓN

### 3.1. Resultados de pruebas de desempeño de funcionamiento no automático

De los resultados de las pruebas de desempeño en funcionamiento no automático se determinó que los tres instrumentos que se construyeron cumplen con los requisitos metrológico de un instrumento para pesar de funcionamiento no automático, con exactitud Clase III, para una división de la escala de verificación  $e = 1$  g. Por simplificación, solo se presentan los resultados de la celda de carga C1.

#### □ Prueba de pesada

El Error Observado (EO) al aplicar cada carga fue menor al error máximo permitido (EMP) para esa carga, con división de escala  $e = 1$  g (Cuadro 1).

Carga calibrada (g)	Carga observada (g)		Error obser. (g)		Error máx permisible, EMP (g)	Pasa si EO < EMP
	Creciente	Decreciente	Creciente	Decreciente		
0	0.06	-0.01	0.06	0.01	0.5	OK
10	10.07	10.03	0.07	0.03	0.5	OK
500	500.24	500.17	0.24	0.17	0.5	OK
1000	1000.24	1000.25	0.24	0.25	1.0	OK
2000	2000.44	2000.45	0.44	0.45	1.0	OK
4229	4230.08	4230.2	1.08	1.2	1.5	OK
4729	4730.12	4730.09	1.12	1.09	1.5	OK
4999	5000.08	5000.17	1.08	1.08	1.5	OK
6499	6500.03	---	1.13	1.03	1.5	OK

**Cuadro 1.** Resultado de prueba de pesada de la celda de carga C1.

### □ Prueba de retorno a cero

La desviación de la indicación cero después de un periodo de carga de media hora de duración fue 0.25 e y 0.37 e, para las cargas Max y 1.3 Max, respectivamente (Cuadro 2). Los errores observados fueron menores a  $< 0.5$  e, error máximo permitido para esta prueba, por lo que el IBP pasa la prueba.

	Inicio	Final
Hora	13:40:00	14:10:00
Temperatura (°C)	27.25	27.25
Carga Max (g)	4999.72	0.25
Carga 1.3 Máx (g)	6500.22	0.37

**Cuadro 2.** Resultados de retorno a cero de la celda de carga C1.

### □ Prueba de repetividad

La diferencia entre los resultados de cinco pesadas de las cargas 0.5Max, Max y 1.3Max es menor al error máximo que se permite para esa carga (Cuadro 3).

0.5 Máx (g)	Máx (g)	1.3Max (g)
2500.33	4999.88	6500.12
2500.26	4999.67	6500.22
2500.31	4999.45	6500.11
2500.08	4999.34	6500.15
2500.13	4999.53	6500.23

**Cuadro 3.** Prueba de repetividad de la celda de carga C1.

### 3.2. Sensibilidad de las celdas de carga a la oscilación térmica

Las condiciones de inicio fueron: fecha 18 de mayo de 2023, 16:00 horas, temperatura 34.5 °C, error igual a 0 (cero) en todos los dispositivos.

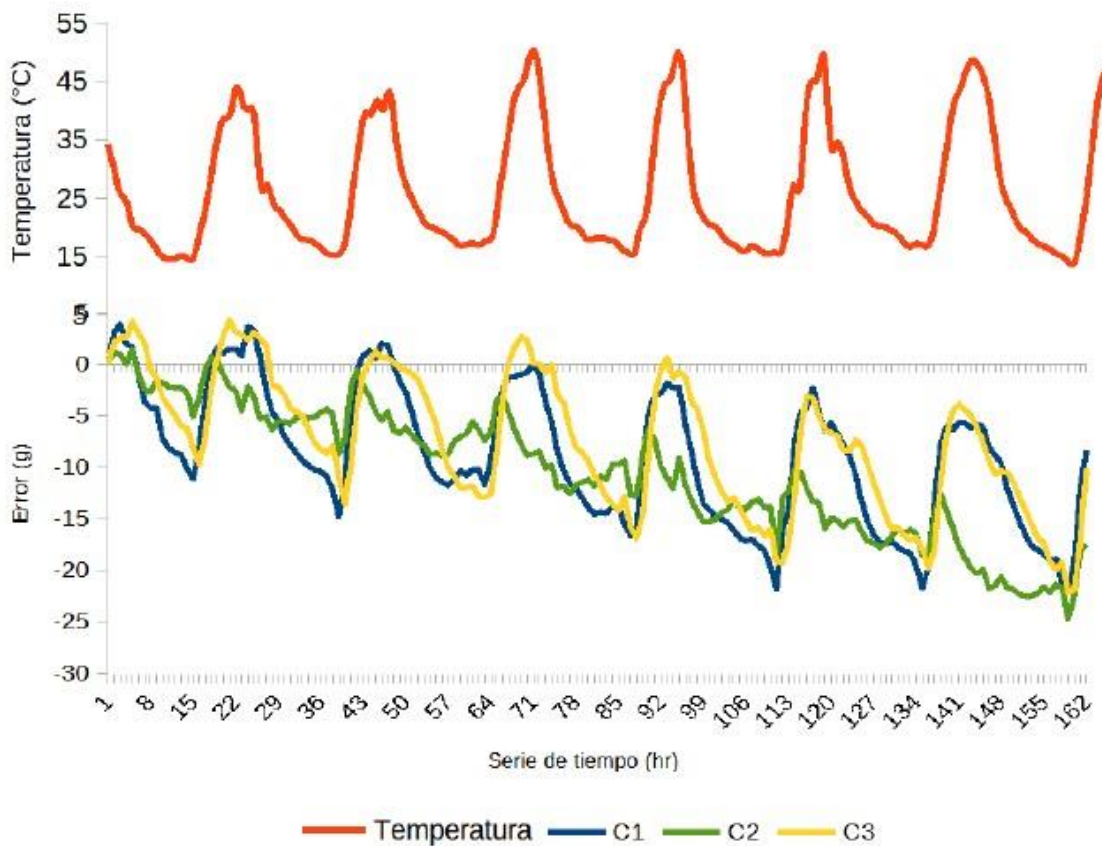
Las series de tiempo error y temperatura revelaron la sensibilidad de cada una de las celdas de carga (C1, C2 y C3) a la oscilación de temperatura durante las pruebas de operación (Figura 2).



**Figura 2.** Pruebas de funcionamiento automático en tres instrumentos para pesar (capacidad 5 kg).

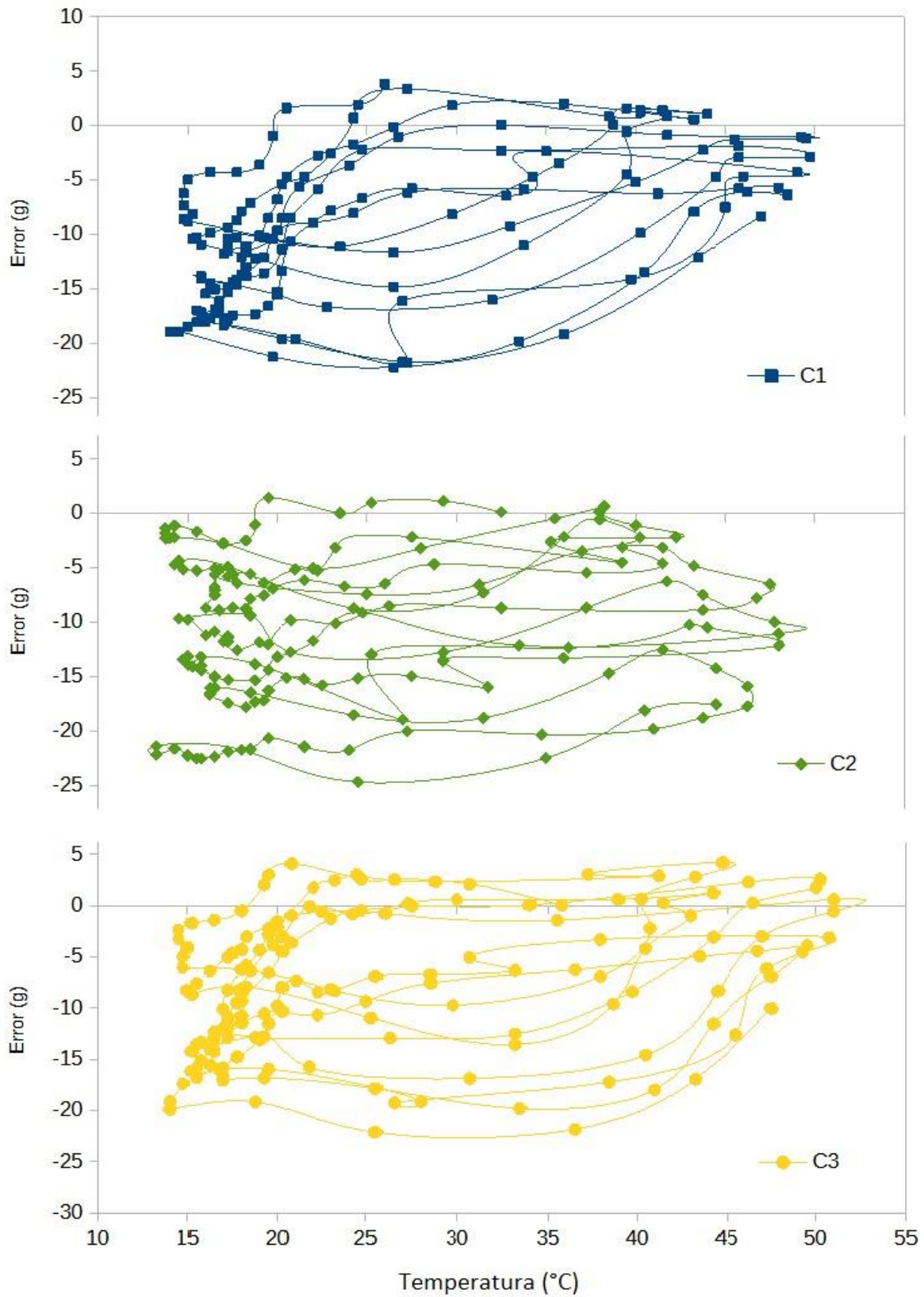
Como menciona Beeson, 2011, el ciclo diario de calentamiento-enfriamiento produce una aparente adición de masa con la subida de temperatura o sustracción de masa con el descenso de temperatura. Adicionalmente, esta aparente adición/sustracción de masa no es equilibrada, tiene una tendencia decreciente que se acumula con forme se cierran ciclos de calentamiento-enfriamiento, es decir el error aumenta por debajo del cero inicial (Figura 3).





**Figura 3.** Series de tiempo Temperatura-Error en cada una de las celdas de carga.

En la Figura 4, el par ordenado temperatura-error revela que la sensibilidad de las celdas de carga no es lineal, sino describen una trayectoria diferenciada dependiendo de si la temperatura instantánea se localiza en una fase de enfriamiento o calentamiento. Este tipo de bucles ha sido observado en materiales con memoria de forma en los que una fuerza motriz (temperatura, tensión, fuerza magnética) produce un cambio en alguna propiedad física del material y la misma puede invertirse o retirarse, y se conocen como fenómeno de histéresis. (Ivshin & Pence, 1994). Además, la tendencia decreciente del error genera curvas que recuerdan a un resorte estirado por debajo del cero.



**Figura 4.** Datos originales del par ordenado temperatura-error.

### 3.3. Compensación del error

A continuación, se describen la metodología que se siguió para intentar compensar el error observado debido a la sensibilidad de las celdas de carga a la temperatura:

#### i. Identificación de las temperaturas máxima y mínima.

Durante el periodo de observación, la temperatura mínima se presentó a las 6:00 horas de la mañana, y la temperatura máxima a las 14:00 horas. Con ello, cada día se fraccionó en las fases de enfriamiento y calentamiento, respectivamente con una duración promedio de 16 y 8 horas.

Es posible que la duración tan extensa de la fase de enfriamiento, de origen a la acumulación del error por debajo del cero. La rapidez con la que se presenta la fase de calentamiento parece ser insuficiente para reestablecer consistentemente las propiedades físicas de la celda de carga previas al último ciclo de enfriamiento.

#### ii. Ejecución del comando Tara

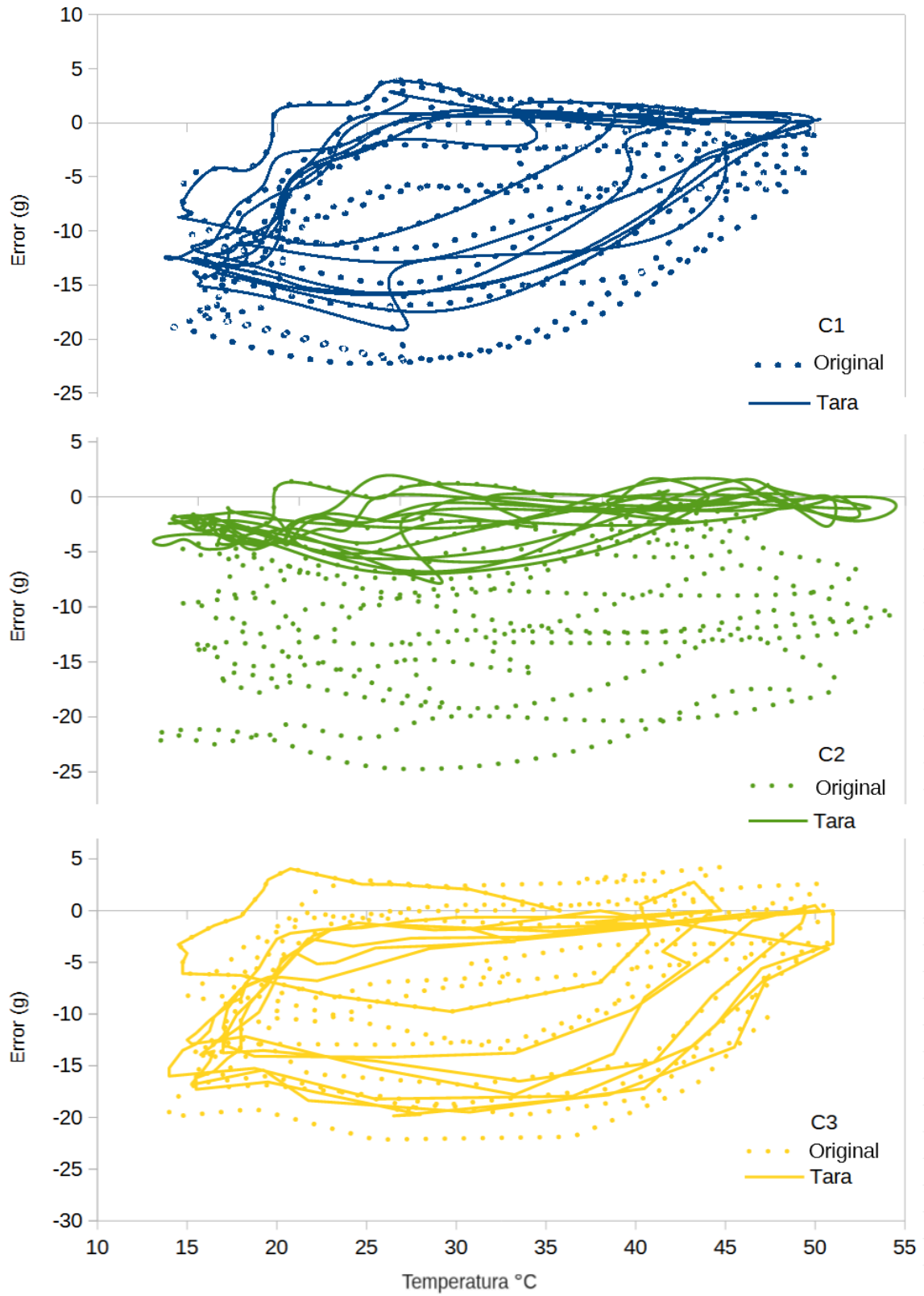
La tendencia decreciente en las series de tiempo del error se corrigió forzando un retorno del error a cero a las 14:00 horas de cada día, momento de ocurrencia de la temperatura máxima. De esta manera, se evitó la elongación de los bucles de histéresis por debajo del cero, efecto resorte, curvas continuas de la Figura 5. Esta compensación también hace más evidente que las celdas de carga son distintamente sensibles a la temperatura. La celda 2 es aparentemente menos sensible a la oscilación térmica diaria, mientras que las celdas 1 y 3 presentan un mayor error registrado y curvas de histéresis muy claras (Figura 5).

Se consideraron los periodos de enfriamiento y calentamiento para ajustar modelos polinómicos de tercer grado que permitieran compensar el error debido a la oscilación térmica diaria (Figura 6, Figura 7 y Figura 8). De las series de datos se han excluido los datos del arranque al primer registro de temperatura máxima, tiempo que se considera necesario para el acondicionamiento de la celda de carga a la atmósfera de operación.

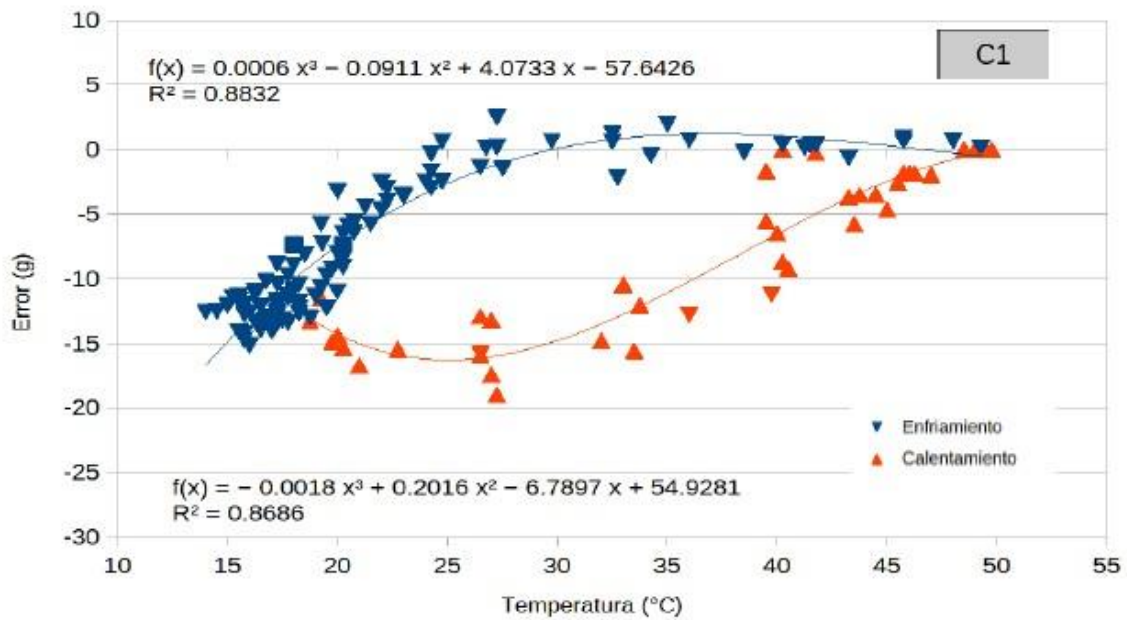
Con la aplicación de los modelos a los registros de peso se redujo el error que se genera por el efecto de la temperatura sobre la celda de carga (Cuadro 4 y Figura 9).

Celda de carga	C1	C2	C3
Originales (g)	9.79	11.91	8.40
Tara (g)	7.65	2.46	8.62
Compensados (g)	1.39	0.41	1.13

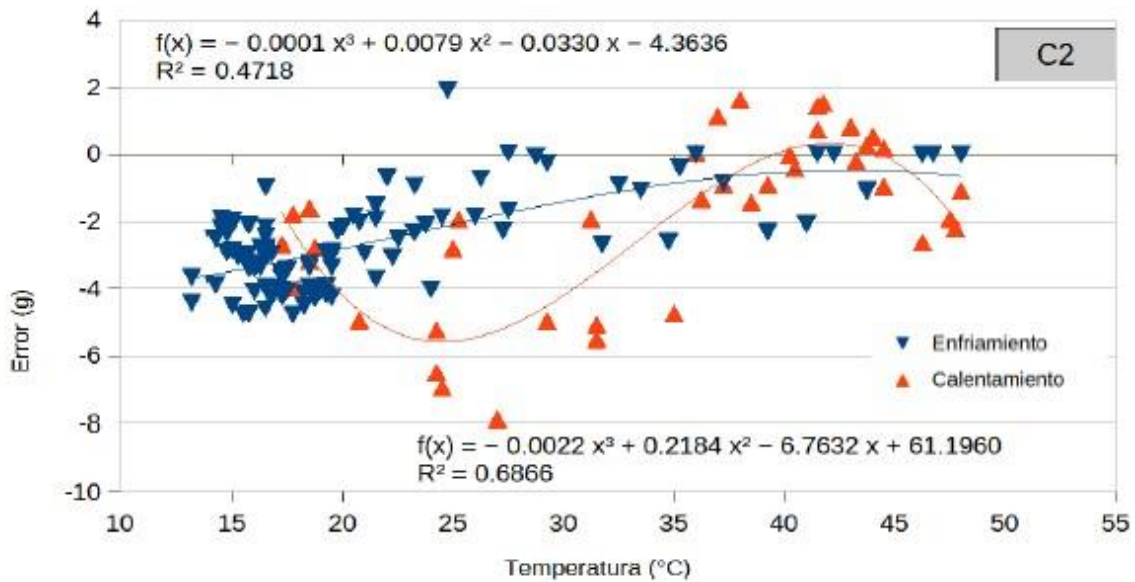
**Cuadro 4.** Resumen de error medio observado para los datos originales, ajuste por comando tara y modelos polinómicos ajustados a las fases de enfriamiento calentamiento, previo ajuste por comando tara.



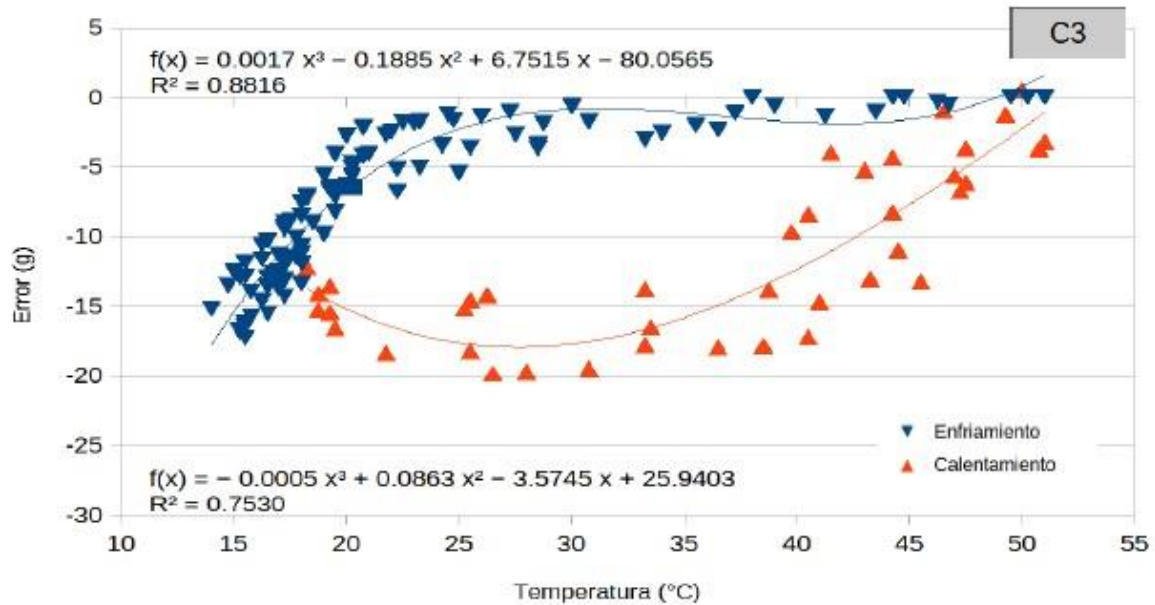
**Figura 5.** Curvas de par ordenado Temperatura- Error después de forzar el retorno a cero del error.



**Figura 6.** Curva de temperatura-error y modelos de compensación de error, celda de carga (C1).



**Figura 7.** Curva de temperatura-error y modelos de compensación de error, celda de carga (C2).

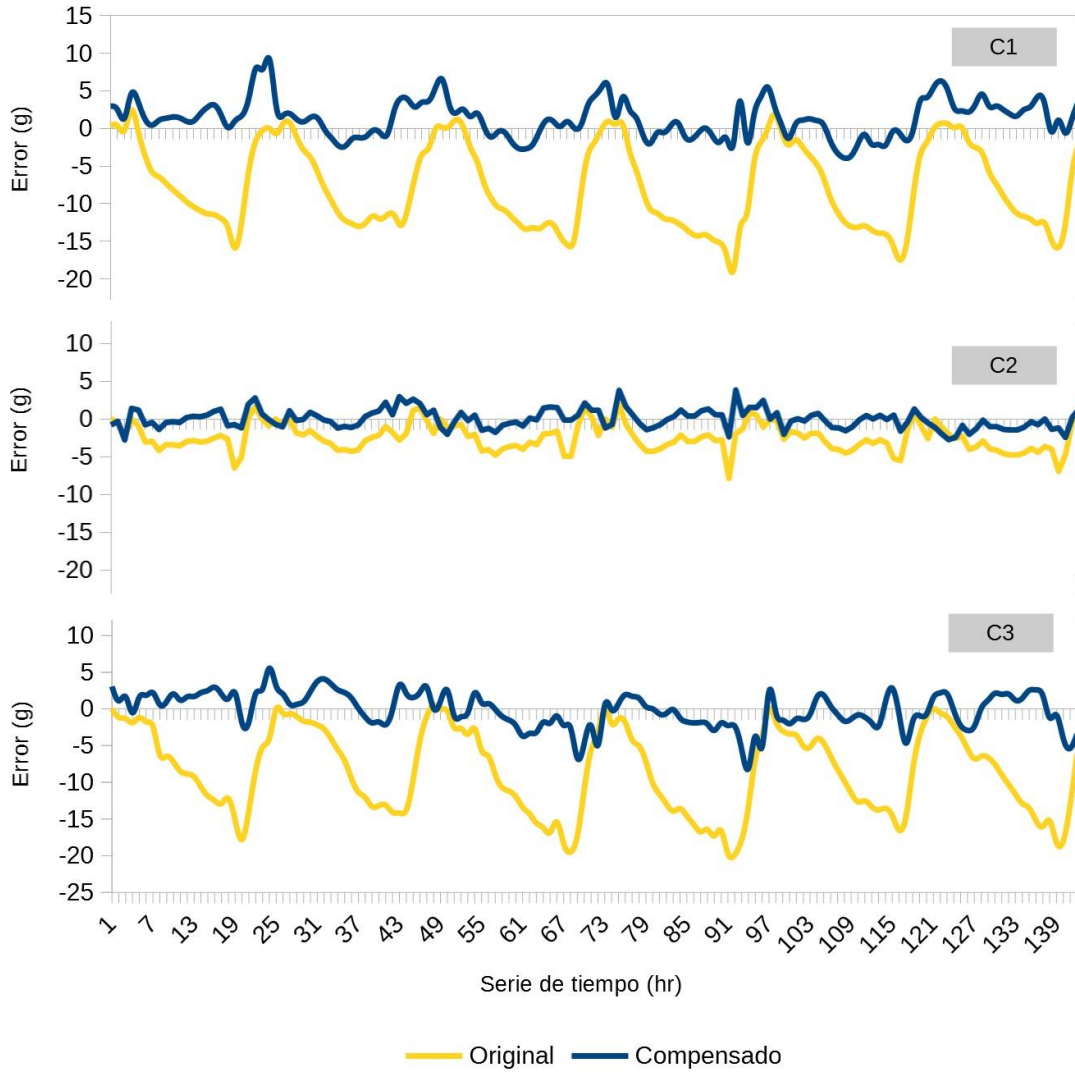


**Figura 8.** Curva de temperatura-error y modelos de compensación de error, celda de carga (C3).

Es necesario verificar que la metodología propuesta permita compensar el error en tiempo real. Esto debería ser posible siempre que los instrumentos para pesar operen bajo el mismo régimen de temperatura en el que los modelos de compensación fueron desarrollados. En este sentido, es importante resaltar el tiempo de observación fue de seis días consecutivos, en los que el rango de variación térmica, y la hora de ocurrencia de temperaturas máxima y mínima de cada día son relativamente constantes. Por lo que, se requiere de series de tiempo de una longitud anual para realizar observaciones del efecto de la oscilación térmica a lo largo del año y prever su compensación.

#### 4. CONCLUSIONES

Las series de tiempo de temperatura y error mostraron una relación directa, lo que permitió aplicar modelos para compensar el error que registró cada sensor en gabinete (post levantamiento de datos). Los modelos permitieron reducir los errores promedio de 9.79 g, 8,40 g y 11.91 g a 1.39 g, 0.4 g y 1.13 g, respectivamente para cada celda de carga. Sin embargo, la compensación del error observado se hizo post levantamiento y resulta insuficiente para aplicaciones que requieren de un instrumento para pesar cuyo funcionamiento sea automático y continuo.



**Figura 9.** Series de tiempo del error registrado originales (amarillo) y datos compensados por los modelos generados (azul).

## 5. REFERENCIAS

Dong Y., & Hansen H. (2023). Development and design of an affordable field scale weighing lysimeter using a microcontroller system. *Smart Agricultural Technology*, 4, 1–8. <https://doi.org/10.1016/j.atech.2022.100147>

Ivshin Y., & Pence T. (1994). A constitutive model for hysteretic phase transition behavior. *Int. J. Engng Sci.* Vol. 32, No 4, pp. 681 – 704, 1994. Elsevier Science.

Kollhovd O. (2022) Library for the HX711 24-bit ADC for weight scales. V1.2.12. [https://github.com/olkal/HX711\\_ADC](https://github.com/olkal/HX711_ADC)

McCauley, D., & Nackley, L. (2022). Development of mini-lysimeter system for use in irrigation automation of container-grown crops. *HardwareX*, 11(1), 1–18. <https://doi.org/10.1016/j.ohx.2022.e00298>

McCauley, D., Levin, A., & Nackley, L. (2021). Reviewing Mini-lysimeter Controlled Irrigation in Container Crop Systems. *HortTechnology*, 31(6), 634–641. <https://doi.org/10.21273/HORTTECH04826-21>

PROY-NOM-010-SCFI-2017. Instrumentos de medición – Instrumentos para pesar de funcionamiento no automático – Requisitos técnicos y metrológicos, métodos de prueba y de verificación.

[https://dof.gob.mx/nota\\_detalle.php?codigo=5533602&fecha=01/08/2018#gsc.tab=0](https://dof.gob.mx/nota_detalle.php?codigo=5533602&fecha=01/08/2018#gsc.tab=0).



Esta obra está bajo una licencia de Creative Commons Reconocimiento-NoComercial-CompartirIgual 2.5 México.