

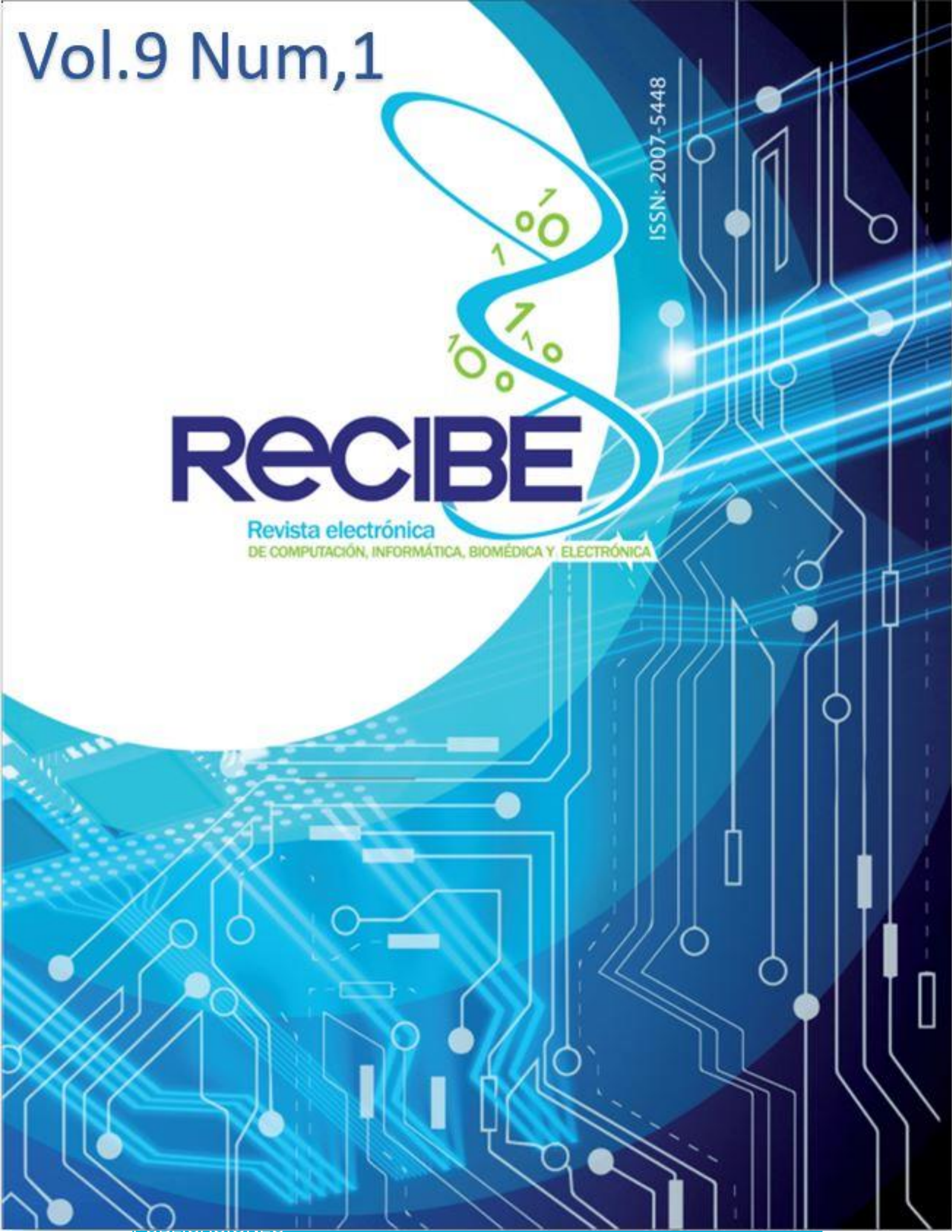
Vol.9 Num,1

ISSN: 2007-5448

RECIBE

Revista electrónica

DE COMPUTACIÓN, INFORMÁTICA, BIOMÉDICA Y ELECTRÓNICA



Índice

Computación e Informática

- Reconocimiento de la denominación de billetes a través de una aplicación móvil con reconocimiento de imagen.** C-1
M.T. I. Francisco Gutierrez Vera, Alejandro Cota Roriguez, Miguel Aangel Sierra Lozada, Claudia Cristina Ortega González
- Experiencias en el Desarrollo de Aplicaciones Móviles con Interfaces basadas en la Interacción Física.** C-2
Pablo Martín Vera, Rocío Andrea Rodríguez, Mariano Kaimakamian Carrau
- Explorando la influencia de los roles de belbin en la calidad del diseño lógico de una base de datos generado por estudiantes de ingeniería de software.** C-3
Raúl Antonio Aguilar Vera, Adriana Peña Pérez Negrón, Julio Cesar Díaz Mendoza, Juan Pablo Ucán Pech
- Metuiga “metodología para el diseño de sistemas basados en interfaces tangibles de usuario y técnicas de gamificación” en la enseñanza de porciones matemáticas para niños ciegos.** C-4
Luis Roberto Ramos Aguiar, Francisco Javier Álvarez Rodríguez
- Hacia un sistema de software basado en ihc para el apoyo de niños con capacidades auditivas diferentes.** C-5
Gerardo Cruz González, Carlos alberto Fernández y Fernández, Jorge Aguilar Cisneros
- Esencialización de la práctica gestión de requisitos de rup.** C-6
Jhonny Edison Chapal Vallejo, John Snyder Guerrero Riascos, Alexander Barón Salazar

Electrónica

- Industrial wireless channel measurements in a 2.4 GHz ISM radio band using a low-cost SDR-based channel sounder.** E-1
Grzegorz Korzeniewski, Roberto Carrasco Álvarez

Biomédica

- Desarrollo y tendencias de la ingeniería biomédica en México.** B-1
José de Jesús Agustín Flores Cuautle, Citlalli Jessica Trujillo Romero, Rigoberto Martínez Méndez, Blanca Estela González Sánchez, Ingrid Lizette Sánchez Carmona

Título: Reconocimiento de la denominación de billetes a través de una aplicación móvil con reconocimiento de imagen

M.T. I. FRANCISCO GUTIERREZ VERA¹

francisco.gutierrez@itcelaya.edu.mx

ALEJANDRO COTA RORIGUEZ¹

janocota@hotmail.com

MIGUEL ANGEL SIERRA LOZADA¹

14030667@itcelaya.edu.mx

Claudia Cristina Ortega González¹

claudia.ortega@itcelaya.edu.mx

¹Tecnológico Nacional de México en Celaya

Resumen.

Es sabido que el reconocimiento de objetos en imágenes está a la orden del día. No es raro observar que Facebook recomienda a sus usuarios etiquetar personas en fotografías, por mencionar algún ejemplo. En varios países, especialmente Estados Unidos y miembros de la Unión Europea, ya se cuenta con herramientas para detectar billetes y sus denominaciones a través de aplicaciones móvil.

En esta investigación se busca el desarrollo de una herramienta capaz de reconocer y clasificar los billetes mexicanos actualmente en circulación, a través de una aplicación móvil capaz de tomar fotografías o seleccionar imágenes existentes para la detección, se creó una red neuronal entrenada para lograr nuestro objetivo: una detección y clasificación de billetes que sea eficiente en al menos el 90% de los casos.

1. Introducción

Zaforas (2017) Dice que el enorme desarrollo que está viviendo la tecnología asociada a la Inteligencia Artificial (IA) está dando lugar en los últimos tiempos a nuevas herramientas y aplicaciones espectaculares. Una de las áreas donde los avances han sido más notables es el reconocimiento de imágenes, en parte gracias al desarrollo de nuevas técnicas de Deep Learning o aprendizaje profundo. Hoy en día tenemos a nuestro alcance sistemas más precisos que los propios humanos, en las tareas de clasificación y detección en imágenes.

Algunas de las aplicaciones que tiene el reconocimiento de imágenes son:

- Etiquetado de imágenes
- Verificación de usuarios basada en rostros
- Análisis de opinión
- Análisis de clientes
- Diagnóstico de enfermedades
- Realidad aumentada
- Detección de matriculas

También comenta que una de las tareas más complejas a la hora de abordar proyectos que integren técnicas avanzadas de IA es el desarrollo de los modelos y la puesta en producción de los mismos. En el caso de las técnicas de reconocimiento de imágenes, existen modelos pre-entrenados que las principales plataformas en la nube nos ofrecen. Algunos ejemplos son AWS Rekognition, Google Cloud Visión API y Azure Computer Visión API. También existen plataformas que permiten el entrenamiento de un modelo propio, como es el caso de TensorFlow.

En diversos países ya se ha llevado a cabo el reconocimiento de imagen para la detección de su respectivo papel moneda. En 2015 el Instituto Nacional de Tecnología Industrial de Córdoba, Argentina, publicó un artículo que habla del proyecto "Software libre de Reconocimiento de billetes para personas en situación de discapacidad visual", que busca ayudar a los débiles visuales en el uso diario de los billetes argentinos, de manera que no tengan que depender de un mediador que les diga la denominación de los billetes. Este desarrollo se sustenta principalmente en la utilización de una librería denominada OpenCV para el procesamiento de las imágenes. Este proyecto fue desarrollado por Moretti, Amado, Puntillo, Caniglia (2015)

Gayon (2017) presentó el proyecto de titulación "DESARROLLO DE UNA APLICACIÓN PARA RECONOCIMIENTO DE BILLETES POR MEDIO DE PROCESAMIENTO DE IMÁGENES PARA PERSONAS CON DIVERSIDAD VISUAL BASADA EN TECNOLOGÍA ANDROID" desde la Universidad Libre Sede Bosque Popular en Bogotá Colombia. El cual también busca ayudar a las personas con diversidad visual a poder identificar con facilidad los diferentes billetes con los que se cuenta en Colombia. En este proyecto también se usó como núcleo la librería OpenCV. Este asunto de las personas con diversidad visual no es un tema menor, así lo indican Singh, Choudhury, Vishal y Jawahar (2014) en donde establecen que para 2013 había 285 millones de personas con esta situación.

A pesar de los resultados exitosos que se han obtenido en dichos países, aun no se cuenta con una aplicación móvil que sea capaz de detectar los billetes mexicanos en sus diferentes denominaciones. Para buscar estas aplicaciones se realizó una indagación a través de la Play Store de Google con los criterios de búsqueda "Billetes mexicanos", "Detector de billetes mexicanos" y "Reconocedor de billetes mexicanos", de estas búsquedas se revisaron los primeros 100 resultados, entre los cuales no se encontró alguno que sea una aplicación capaz de reconocer la denominación de los billetes mexicanos.

La presente investigación busca desarrollar una aplicación móvil que sea capaz de reconocer y clasificar por su denominación el papel moneda mexicano (veinte, cincuenta, cien, doscientos y quinientos), como medio coadyuvante en la detección de billetes. Se busca que la aplicación sea capaz de identificar correctamente dichos billetes desde fotografías tomadas en diferentes posiciones, diferente iluminación y con diferentes contrastes, esto con una precisión de al menos 90%.

2. Metodología

La metodología consta de siete etapas (ver figura 1), las cuales se explican a detalle en la presente sección.

Figura 1

Metodología en cascada de las etapas del proyecto (Imagen Propia).



2.1 Análisis de herramientas factibles

A lo largo de esta etapa se investigó acerca de las diferentes herramientas o *frameworks* disponibles en la actualidad que se ajustan al propósito de la presente investigación. Inicialmente, las herramientas consideradas fueron las siguientes:

- Tensorflow y Tensorflow Lite.
- OpenCV.
- Azure Computer Vision API

Se consideró que la herramienta a usar debía cumplir con ciertos requisitos, los cuales se establecieron de la manera siguiente:

- Procesamiento de imágenes. La herramienta o *framework* debe ser capaz de recibir una imagen como entrada de datos para realizar la inferencia necesaria y proporcionar resultados correctos.
- Detección de objetos. La herramienta o *framework* debe ser capaz de reconocer objetos de ciertas clases bien definidas (billetes mexicanos) y proporcionar como resultado las coordenadas de una región rectangular que enmarque cualquier detección.
- Aprendizaje automático. La herramienta o *framework* debe ser capaz de recibir un conjunto de datos etiquetados para generar su propio conocimiento acerca del problema y otorgar resultados relevantes.
- Rapidez en el procesamiento. La herramienta o *framework* debe ser capaz de realizar la inferencia de forma rápida en los límites de usabilidad aceptables, para que la experiencia del usuario final sea favorable.
- Compatibilidad con el sistema Android. Dado a que se propone construir una aplicación para el sistema Android, es estrictamente necesario el cumplimiento de este requisito.
- Procesamiento local. La herramienta o *framework* debe ser capaz de trabajar sin conexión a internet.
- Facilidad del desarrollo. Opcional, el uso de la herramienta o *framework* debe estar bien documentado. Es deseable que la herramienta se ajuste a ciclos de desarrollo por prototipos.

Al haber comparado las características ofrecidas por cada herramienta contra las establecidas se descartó Azure Computer Vision API, debido a que se trata de una plataforma en la nube que requiere forzosamente una conexión a internet. Además, no ofrece la detección de objetos como tal, sino la clasificación de una imagen completa (sin enmarcar las detecciones).

Por otro lado, OpenCV y Tensorflow ofrecen características muy similares. Sin embargo, Tensorflow ofrece una librería compatible optimizada para Android (Tensorflow Lite), lo cual implica un mejor desempeño al momento de realizar la inferencia. Consideramos de gran importancia que la herramienta sea rápida, así que al final se optó por Tensorflow y Tensorflow Lite en conjunto. Adicionalmente, Tensorflow cuenta con modelos preestablecidos para el trabajo de ciertas situaciones, siendo una de ellas el reconocimiento de objetos. Esto nos permite agilizar el desarrollo al construir sobre un modelo existente.

2.2 Establecimiento del marco de trabajo

En esta etapa, se procedió al establecimiento de las características del modelo. Cabe destacar que Tensorflow trabaja con redes neuronales para el reconocimiento de objetos, por lo que hay ciertos aspectos y parámetros que se deben tener en cuenta. En este tipo de desarrollos lo ideal es tomar modelos de redes existentes y adaptarlos a la medida. Ahora, si se toma en cuenta que una red neuronal promedio suele ser enorme para un dispositivo móvil, se debe seleccionar el modelo con cuidado. De acuerdo con Howard et al. (2017) existe un modelo de red neuronal eficiente para

aplicaciones móviles y embebidas llamado *MobileNet*, que soluciona la cuestión previamente mencionada. En conjunto con la red neuronal es necesario establecer un algoritmo de detección. Uno de ellos optimizado para realizar detecciones, generar propuestas de detección, y arrojar las cajas de detección en un solo pase es el algoritmo SSD (*Single-Shot Detector*), propuesto por Liu et al. (2015). De igual manera, se elige este algoritmo por su eficiencia para su aplicación en un entorno móvil. Como algoritmo de optimización para el entrenamiento se eligió el *optimizador ADAM*, ya que según Kingma y Ba (2014) resulta bastante eficiente para casos de aprendizaje con datos que tienden a variar o incorporar "ruido" (siendo el caso de las condiciones de iluminación, rotación, distorsión y truncamiento de los billetes).

Otro aspecto importante del desarrollo de este tipo de modelos es que el hecho de construir una red neuronal desde cero y entrenarla hasta la convergencia podría tomar meses. Por esto, es recomendable tomar una red neuronal previamente entrenada con otro conjunto de datos y construir sobre la misma. Para este caso, en el repositorio de Github de Tensorflow existen algunos modelos ya entrenados. Este proyecto toma como base el modelo entrenado con el conjunto de datos COCO (*Common Objects in Context*), cuyo propósito descrito por Lin et al. (2014) es detectar más de 90 objetos comunes que serían reconocidos por un niño de 4 años.

La red neuronal recibe como entrada la imagen redimensionada a 300 píxeles de ancho por 300 píxeles de alto. Como salida, la red neuronal proporciona cuatro tensores que contienen las ubicaciones de las cajas de las detecciones, las clases de las detecciones, el grado de confianza de las detecciones, y el número de detecciones (ver figura 2, grafo de inferencia y red neural). A partir de estos resultados, escalar las dimensiones de las cajas y traducir las clases a los tipos de billetes detectados para dibujar sobre la imagen original es una tarea trivial. Una vez definido el modelo y sus parámetros, se estableció el flujo de trabajo de acuerdo al diagrama de la figura 3. El desarrollo siguió un patrón iterativo basado en el entrenamiento del modelo.

Figura 3

Flujo de trabajo para el entrenamiento y liberación de la app (imagen Propia).

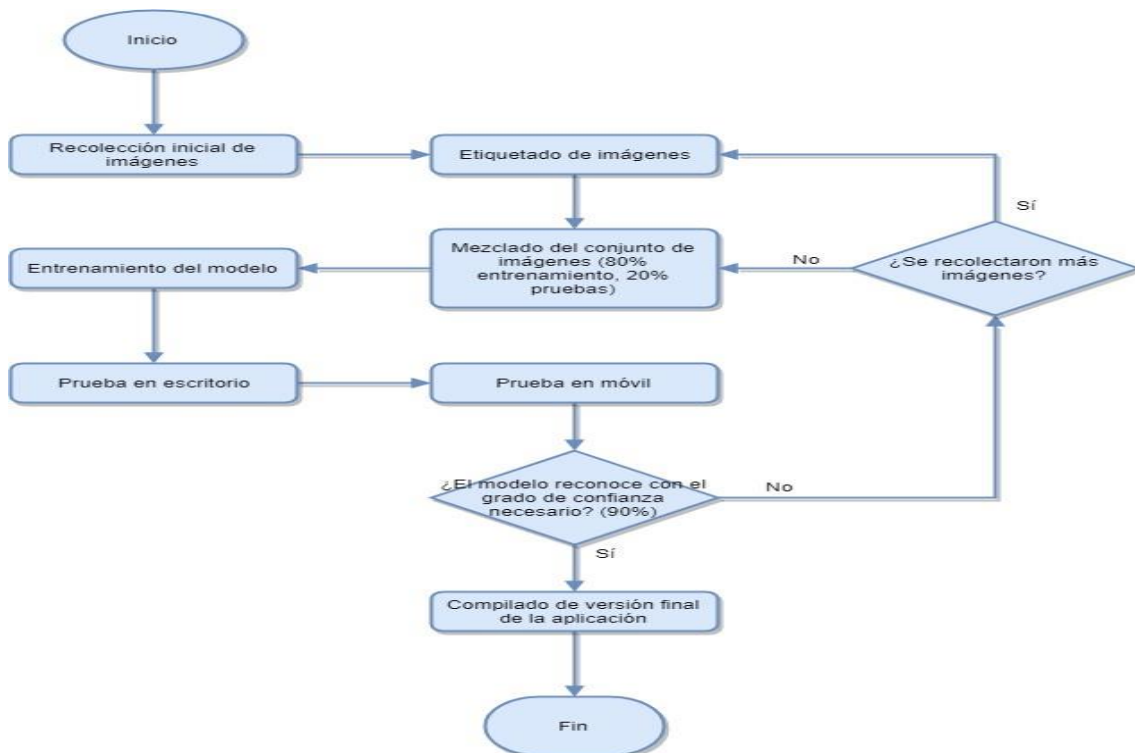
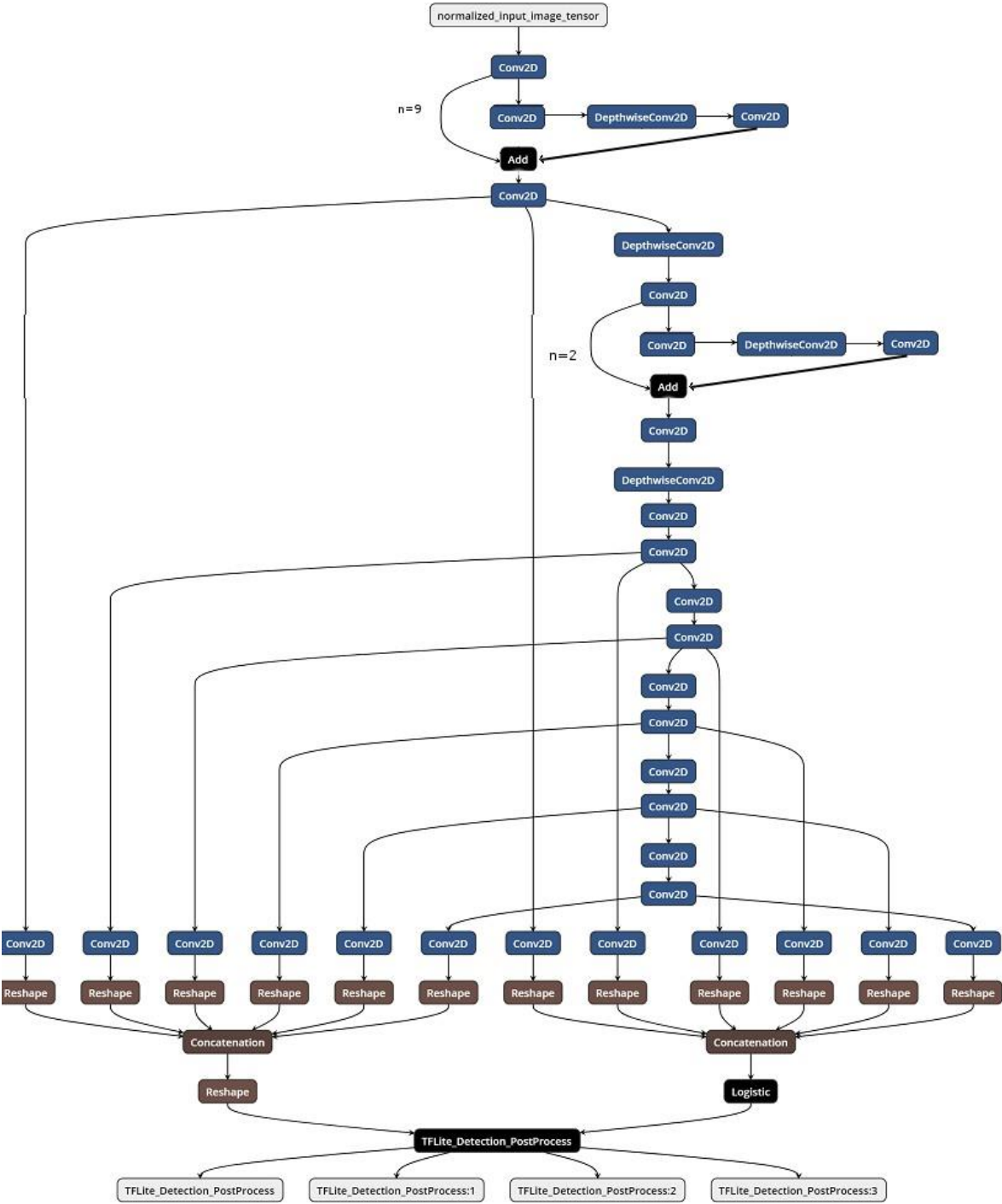


Figura 2

Grafo de inferencia, se aplican varias etapas de conversión a 2D y redimensionamiento para aumentar el grado de eficiencia (Imagen Propia).



2.3 Recolección y etiquetado de imágenes

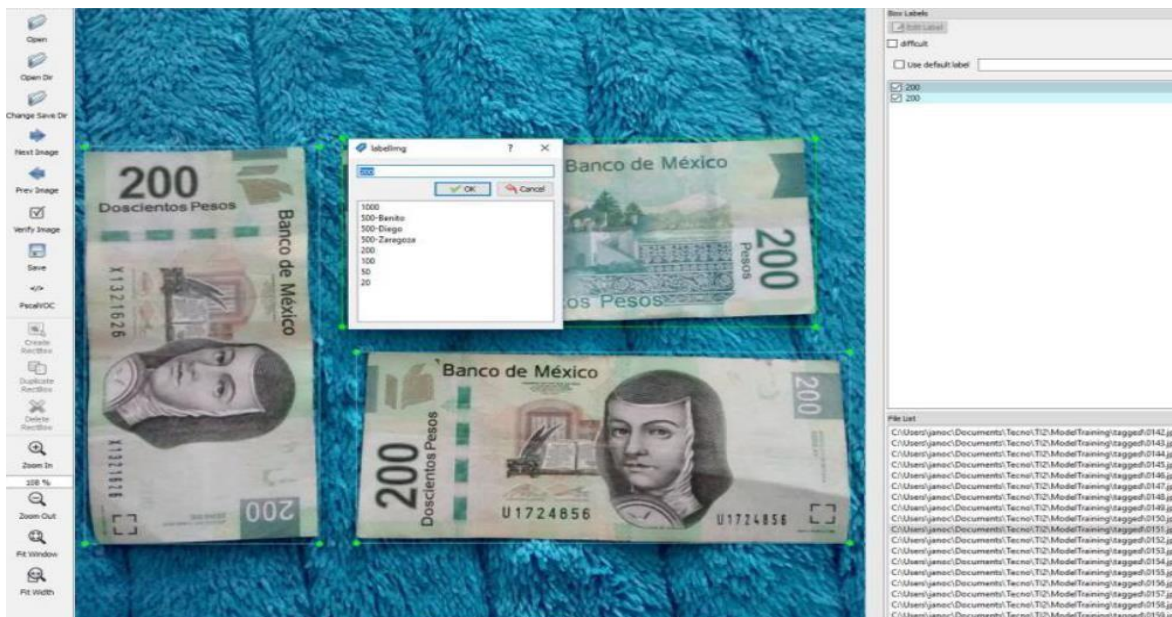
Antes de proceder al entrenamiento es necesario construir un *dataset* o conjunto de datos. Este debe contener una cantidad suficiente de imágenes para cada clase de objeto a reconocer. Es decir, para esta investigación debe contener imágenes de cada billete mexicano propuesto.

Como regla general, se recomienda contar con al menos 100 imágenes por clase para prototipos de detección de objetos. Para esta investigación, a la fecha del último ciclo de desarrollo, se recolectaron 431 imágenes en total incluyendo billetes solos, en conjunto, billetes en diferentes posiciones, truncados, deformados y a diferentes distancias, además de incluir imágenes sin billetes para mejorar la precisión del modelo.

Después de recolectar imágenes, la siguiente tarea es el etiquetado de las mismas. Con ayuda de un software de etiquetado como *Labellmg* se dibujan rectángulos sobre cada billete presente en una imagen y se les asigna una etiqueta correspondiente al nombre de la clase (ver figura 4, etiquetado de las imágenes). Una vez que todas las imágenes se han etiquetado procede con la siguiente etapa.

Figura 4

Configuración de la utilería Labellmg, la cual permite definir una etiqueta descriptiva de acuerdo a la imagen reconocida (imagen Propia).



2.4 Entrenamiento del modelo

Durante esta etapa se alimenta a Tensorflow con el *dataset* etiquetado para que el modelo genere su propio conocimiento del problema a través de un proceso de aprendizaje automático. Comúnmente, dicho proceso es conocido como entrenamiento.

Antes de iniciar el entrenamiento se deben cumplir dos prerequisites. Primero, se debe dividir el *dataset* en dos subconjuntos: de entrenamiento y de pruebas. El subconjunto de entrenamiento es utilizado por Tensorflow para el proceso de aprendizaje automático y es recomendable que contenga el 75% del total de las imágenes. El 25% de imágenes restantes forma el subconjunto de pruebas, que es utilizado para reforzar el aprendizaje mediante la ejecución de la detección de objetos y comparación de los resultados con la salida esperada (etiquetas provistas por el usuario). El *dataset* de la presente investigación se dividió en subconjuntos de 323 y 108 imágenes respectivamente.

El segundo prerequisite es convertir el *dataset* a un formato aceptado por Tensorflow. Previo a la popularización de este tipo de proyectos era necesario programar diferentes herramientas y *scripts* para dicha conversión. Sin embargo, algunos desarrolladores han liberado utilidades de código abierto para realizar la conversión, las cuales se han tomado y modificado ligeramente de acuerdo a las necesidades del presente proyecto.

Una vez cumplidos estos prerequisites, se procede a ejecutar el entrenamiento. De acuerdo a un número de iteraciones y un tamaño de lote establecido, Tensorflow toma un lote de imágenes y ejecuta los procesos necesarios para el aprendizaje automático hasta llegar al límite de iteraciones. Cabe destacar que el tamaño de lote y el número de iteraciones define la cantidad de veces que se ejecuta un pase sobre el *dataset* en su totalidad (un *epoch*) durante el proceso de aprendizaje. Para la presente investigación se estableció un tamaño de lote de 8 imágenes y un límite de 100,000 iteraciones, para un total de 2,476.78 *epochs*.

Es necesario considerar que entrenar un modelo de red neuronal requiere ciertos recursos, y no se trata de una tarea fácil, motivo por el cual Tensorflow ofrece versiones para trabajar con *CPU* (procesador) y *GPU* (unidad de procesamiento gráfico, por sus siglas en inglés). Generalmente, este tipo de tareas se ven beneficiadas del uso de una o múltiples *GPUs* debido a la naturaleza del problema (operaciones simples con un alto grado de paralelismo). Dicha consideración se tomó en cuenta para la presente investigación, y se seleccionó una versión de Tensorflow compatible con *GPUs*. La selección de hardware y software del equipo de cómputo usado para el entrenamiento fue la siguiente:

- CPU: AMD Ryzen 5 1600 (6 núcleos, 12 hilos) @ 3.825 GHz (*overclock*)
- GPU: NVIDIA GeForce GTX 1050 Ti (768 núcleos CUDA, 4 GB GDDR5, driver 416.34) @ 1,949 MHz / 4,503 MHz (*overclock*)
- Memoria: 2x8 GB DDR4 (2,400 MHz) en configuración *dual-channel*
- Sistema operativo: Windows 10 1803 (64 bits)
- Python v3.6.6
- Tensorflow: tensorflow-gpu v1.11.0 (CUDA Toolkit v9.0.17, cuDNN SDK 9.0 v7.3.1.20)

Parámetros como el tamaño de lote y las dimensiones de entrada del modelo tienen un impacto directo en recursos de memoria y en la rapidez del proceso, por lo que fueron ajustados de acuerdo al hardware sobre el que se realizó el entrenamiento.

Una ventaja de haber seleccionado Tensorflow para el desarrollo de la presente investigación es que durante el entrenamiento se generan *checkpoints* o puntos de control cada cierto número de iteraciones. Estos *checkpoints* pueden ser usados como si se tratara del modelo final, lo que propicia el desarrollo por prototipos y fue de utilidad para el desarrollo en paralelo de las siguientes etapas.

Terminado el entrenamiento o para realizar pruebas con un checkpoint es necesaria una conversión del modelo, puesto que Tensorflow y Tensorflow Lite trabajan con formatos diferentes. El modelo de Tensorflow contiene operaciones para el entrenamiento y otras etapas de pre y postprocesamiento de los datos, las cuales no son necesarias en un modelo de Tensorflow Lite y son removidas. Idealmente, el modelo no debe verse afectado por la conversión.

2.5 Desarrollo de la aplicación

La aplicación fue desarrollada para el sistema operativo Android. Como requisito mínimo se estableció la versión 4.4 (Kitkat) del mismo para cubrir el 95.3% de los dispositivos activos al mes de noviembre de 2018 (ver figura 5, distribución acumulada de versiones de Android) de acuerdo con la información proporcionada por Android Studio en su versión 3.2.1, entorno de desarrollo utilizado para la presente investigación. La única dependencia de la aplicación es Tensorflow Lite en su versión 1.10.0.

Figura 5

Distribución acumulada de las versiones de Android de acuerdo a Android Studio (imagen Propia).

ANDROID PLATFORM VERSION	API LEVEL	CUMULATIVE DISTRIBUTION
4.0 Ice Cream Sandwich	15	
4.1 Jelly Bean	16	99.6%
4.2 Jelly Bean	17	98.1%
4.3 Jelly Bean	18	95.9%
4.4 KitKat	19	95.3%
5.0 Lollipop	21	85.0%
5.1 Lollipop	22	80.2%
6.0 Marshmallow	23	62.6%
7.0 Nougat	24	37.1%
7.1 Nougat	25	14.2%
8.0 Oreo	26	6.0%
8.1 Oreo	27	1.1%

La aplicación es muy básica, con 2 vistas o actividades diferentes. La primera permite seleccionar entre tomar una fotografía con una aplicación de cámara ya existente en el dispositivo móvil o seleccionar una imagen de la galería, además de permitir ajustar el grado de confianza mínimo aceptable para las detecciones (ver figura 6, vista principal de la aplicación).

La segunda vista presenta los resultados de la inferencia con sus detecciones correspondientes en forma de rectángulos coloreados además de información relevante sobre la detección actual en forma de texto (ver figura 7, vista de resultados de la aplicación).

2.6 Pruebas y depuración

Para verificar el modelo se realizaron pruebas tanto en computadora de escritorio como en dispositivo móvil. Para las pruebas se usó el mismo equipo de cómputo en el que se realizó el entrenamiento. Los dispositivos móviles usados fueron: un Motorola XT1032 (Android 7.1.1) y una tableta genérica (Android 4.4). Las imágenes usadas para las pruebas fueron las mismas existentes en el *dataset*.

Una vez liberada la versión final se procedió a realizar el experimento en la aplicación, el cual consistió en probar con cinco imágenes usadas durante el entrenamiento del modelo, así como con cinco fotos tomadas en el momento a través de la aplicación para obtener los valores de certeza.

Figura 6

Vista Principal de la aplicación, con tres áreas, dos para la imagen y la final para el resultado (imagen propia).



Figura 7

Diferentes billetes con deficiente reconocimiento, durante el entrenamiento (imagen propia).



Figura 8

Billetes de 500 identificados como de 200, durante el entrenamiento (imagen propia).



2.7 Análisis de resultados y conclusiones

Después de haber realizado las pruebas correspondientes, los resultados son guardados como evidencia para su posterior análisis. Se recolectaron principalmente los resultados de la detección en computadora de escritorio, se probaron diferentes *checkpoints* del modelo hasta terminar el entrenamiento. Una vez terminado el entrenamiento, se recolectaron los resultados de la aplicación móvil. Posterior a la recolección descrita, se calcularon índices de detección para los modelos de Tensorflow (escritorio) y Tensorflow Lite (móvil) se aplicó t-student para los resultados de la app y del emulador, con la finalidad de establecer si los resultados son homogéneos y con ello validar la hipótesis planteada, la formula utilizado fue la mostrada en la figura 9, aunque se obtuvieron 2 columnas con resultados no son dependientes entre ellas ya que es la misma aplicación ejecutándose en 2 dispositivos diferentes. Gorgas, Cardiel y Zamorano (2011) establecen que la distribución t de Student es sumamente importante para la estimación y el contraste de hipótesis sobre la media de una población.

Figura 9

Fórmula de t de student para muestras pequeñas menores de 60 (Gorgas, Cardiel y Zamorano 2011).

$$\text{Prueba T para Muestra Única} = \frac{\bar{X} - \mu_0}{\frac{\sigma}{\sqrt{n}}}$$

Nota. La fórmula involucra la media de la muestra menos μ (valor buscado), dividido entre la proporción de la desviación estándar y la raíz del tamaño de la muestra.

3. Resultados

Las imágenes de muestra, que se pueden observar en las figuras 7, 8, 9 y 10, denotan varios elementos visuales que nos permiten interpretar parte de los resultados. Es importante resaltar que las figuras 7 y 8 pertenecen a pruebas realizadas con la aplicación móvil, mientras que las figuras 9 y 10 fueron tomadas de las pruebas realizadas con la versión de escritorio del modelo una vez terminado el entrenamiento del mismo, por lo que varía lo que se puede apreciar en cada una. Para marcar el área donde se detecta la posibilidad de encontrar un billete se dibuja un rectángulo cuyo color varía según la denominación del billete encontrado. En la versión móvil se encuentra una sección de estadísticas donde se puede apreciar el grado de confianza mínimo con el que se detectaran los billetes; carga, detección y dibujo son los respectivos tiempos que le tomo a la aplicación realizar cada uno de esos procesos. Finalmente en la sección de detecciones, se indica por cada uno de los billetes detectados, con el mismo color usado para dibujar el rectángulo con el que se marcó la imagen, cual fue la denominación que se detectó y con qué grado de confianza. Para la versión de escritorio, las estadísticas son mostradas en una ventana independiente a la imagen y tanto la denominación como el grado de confianza son puestos dentro del mismo marco dibujado en la imagen.

Como se puede observar en las figuras 7 y 8, el modelo detecta un billete de doscientos pesos del tamaño de toda la imagen en lugar del resultado esperado (ver figuras 9 y 10), situación que se presentó a lo largo de las 10 pruebas individuales realizadas. En algunos casos, como se puede observar en la figura 7, es detectado alguno de los billetes de manera correcta, con un grado de confianza bastante bajo, como consecuencia, si éste es aumentado a un 90% (configuración óptima) la aplicación no sería capaz de detecciones aceptables.

La aplicación móvil alimentó las imágenes exitosamente al modelo de Tensorflow Lite. Los resultados de las 10 pruebas se muestran en la tabla 1, comparados con los obtenidos de las mismas imágenes en la versión de escritorio. En la tabla se pueden observar tres cosas importantes. Primeramente, el tiempo de procesamiento es en promedio dieciséis veces mayor para la aplicación móvil. La segunda situación observable es que para la aplicación de escritorio la confiabilidad promedio es de 95%, la versión móvil apenas alcanza un 32% lo cual es considerablemente bajo y nada satisfactorio. Por último se denota que la cantidad de billetes detectados no pasa de dos en la versión móvil, cuando se deberían de detectar hasta cuatro en algunas situaciones, como es el caso de las pruebas 9 y 10. Cabe destacar que la desviación estándar nos indica que los datos estadísticos de la versión móvil no son estables.

Con los resultados obtenidos se confirma que la hipótesis es errónea, puesto que no se logró detectar la denominación de los billetes mexicanos a través de una aplicación móvil con una precisión del 90%, y se declara el experimento como fallido.

Figura 9

Resultado obtenido desde el emulador de Android de la PC, con 2 billetes al mismo tiempo (imagen propia).



Nota. Se puede observar que se colocaron los dos billetes en diferente alineación, y que se etiquetaron adecuadamente con el nombre del personaje y su nombre.

Figura 10

Resultado obtenido desde el emulador de Android de la PC, con 4 billetes al mismo tiempo.



Nota. Se puede observar que se alinearon los billetes un tanto cuanto aleatorio, la aplicación reconoció bien 3 de los 4 billetes, en el caso del billete de 50 y 100 no se definió un nombre por eso solo aparece el número (imagen propia).

Tabla 1

Comparativa de Resultados del experimento, entre el dispositivo móvil y la emulación de Android en una PC

Prueba	No. detecciones		Tiempo lectura (ms)		Tiempo Detección (ms)		Total del Proceso (ms)		Grado de Confianza	
	Móvil	PC	Móvil	PC	Móvil	PC	Móvil	PC	Móvil	PC
1	1	2	464	68	1553	17	2295	132	39%	99%
2	1	1	777	51	892	17	2496	103	37%	99%
3	1	1	275	67	1321	16	1749	119	38%	99%
4	1	1	167	60	951	17	1051	111	32%	99%
5	1	1	634	30	1004	18	1832	81	35%	99%
6	2	2	815	47	988	18	2485	110	15%	91%
7	1	3	819	47	1045	17	2544	122	37%	92%
8	1	2	818	35	798	18	2289	99	34%	99%
9	2	4	721	76	1559	17	2600	165	18%	87%
10	1	4	145	82	870	19	1057	173	35%	95%
Promedio	1.20	2.10	563.50	56.30	1098.10	17.40	2039.80	121.50	32.00%	95.90%
Desv. Estándar	0.42	1.20	277.72	17.19	278.70	0.84	593.76	28.66	8.45%	4.43%

Nota. Se realizaron 10 muestreos, incorporando los tiempos que conllevaba cada prueba, tiempo de lectura, detección y proceso, en el renglón promedio se puede observar que siempre el móvil tardaba más o era menos eficiente.

4. Discusión

Haciendo uso de Tensorflow Lite en un entorno móvil, la detección no presentó resultados satisfactorios. Sin embargo, según las pruebas hechas en la computadora de escritorio durante el entrenamiento del modelo, con la versión completa de Tensorflow, el modelo fue más que exitoso. De una muestra de cuarenta imágenes tomadas aleatoriamente del *dataset* y alimentadas al modelo sólo un billete no fue reconocido, mientras que los demás fueron detectados exitosamente con un grado de confianza mayor al 90%.

En la tabla se observa que el porcentaje de certeza está en 32%, la ineficacia de la aplicación puede deberse a incompatibilidades no documentadas entre Tensorflow y Tensorflow Lite ya que, como se mencionó en el presente artículo, es necesaria una conversión de formato entre modelos. Cabe destacar que la herramienta proporcionada para una etapa de la conversión (Tensorflow Lite Converter, o TOCO) es incompatible con Windows, y fue necesario utilizar una distribución de Linux (Ubuntu 16.04.5 LTS en Windows, con WSL). La enorme diferencia en capacidad de procesamiento entre un dispositivo móvil y una PC puede haber influido negativamente en los resultados obtenidos. Otra causa probable es la diferencia en el número de versiones, siendo 1.11.0 para Tensorflow y 1.10.0 para Tensorflow Lite.

Para validar los datos se aplicó la fórmula de la figura 9. Obteniendo los valores de $t = -0.00217161112836$ para los datos de la app y de $t = 6.19836460565E-5$ para el emulador de android en la computadora, para ambos casos el valor de la tabla de student es de 1.8331. En el caso de la app se establece que $t < 1.8331$ por lo tanto la hipótesis se rechaza, mientras que para el emulador en la PC $t > 1.8331$ la hipótesis podría ser aceptada, sin embargo la investigación estaba orientada hacia un dispositivo móvil.

Una posible alternativa al problema presentado es la implementación de Tensorflow Mobile, el predecesor de Tensorflow Lite, puesto que no se requiere conversión alguna en el proceso de desarrollo del modelo. Sin embargo, se trata de una herramienta cuyo soporte está en desarrollo, además de no contar con el mismo grado de optimización que Tensorflow Lite.

Dicho esto, es muy probable que en versiones futuras de Tensorflow Lite el modelo pueda otorgar resultados satisfactorios. Mientras tanto, se logró un excelente progreso en el modelo de Tensorflow, lo que puede servir para otro tipo de aplicaciones en entornos no móviles que requieran la detección confiable de billetes mexicanos y sus correspondientes denominaciones.

Otra posible explicación del por qué la app no dio buenos resultados con respecto del emulador en la Pc puede deberse a las capacidades de los microprocesadores, en donde la capacidad de una PC supera a un móvil, y esto daría pie de entrada a otro tipo de pruebas para probar esta nueva hipótesis.

5. Referencias

Althafiri, E.& Sarfraz, M. & Alfarras, M. (2012) Bahraini paper currency recognition. Journal of Advanced Computer Science and Technology Research, 2012.

Debnath, K. K.& Ahdikary, J. K. & Shahjahan, M. (2009). *A currency recognition system using negatively correlated neural network ensemble*. International Conference on Computers and Information Technology.

Gorgas J, & Cardiel N, & Zamorano J. (2011). *Estadística Básica para estudiantes de ciencias*, Universidad Complutense, España

Gayon, C. (2017). *Desarrollo de una aplicación para reconocimiento de billetes por medio de procesamiento de imágenes para personas con diversidad visual basada en tecnología android*. Septiembre 27, 2018, de Universidad libre sede bosque popular facultad de ingeniería de sistemas Bogotá. Sitio web: <https://goo.gl/8SA1rQ>

Howard, A. G. & Zhu, M. & Chen, B. & Kalenichenko, D. & Wang, W. & Weyand, T. & Andreetto, M. & Adam, H. (2017). *MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications*. Noviembre de 2018, de arXiv Sitio web: <https://arxiv.org/abs/1704.04861>

Kingma, D. P. & Ba, J. (2014). *Adam: A Method for Stochastic Optimization*. Noviembre de 2018, de arXiv Sitio web: <https://arxiv.org/abs/1412.6980>

Lin, T. & Maire, M. & Belongie, S. & Bourdev, L. & Girshick, R. & Hays, J. & Perona, P. & Ramanan, D. & Zitnick, C. L. & Dollár, P. (2014). Microsoft COCO: Common Objects in Context. Noviembre de 2018, de arXiv Sitio web: <https://arxiv.org/abs/1405.0312>

Liu, W. & Anguelov, D. & Erhan, D. & Szegedy, C. & Reed, S. & Fu, C. & Berg, A. C. (2015). SSD: Single Shot MultiBox Detector. Noviembre de 2018, de arXiv Sitio web: <https://arxiv.org/abs/1512.02325>

Moretti, I, & Amado, J, & Puntillo, D, & Caniglia, C. (2015). *Software libre de Reconocimiento de billetes para personas en situación de discapacidad visual*. Septiembre 27, 2018, de Instituto Nacional de Tecnología Industrial, Centro regional Córdoba, Córdoba, Argentina Sitio web: <https://goo.gl/8NsH4S>

Singh, S. & Choudhury, S. & Vishal, K.& Jawahar, C.V. (2014). *Currency Recognition on Mobile Phones*, 22nd International Conference on Pattern Recognition,

Zaforas, M. (2017). *Inteligencia Artificial como servicio: reconocimiento de imágenes*. Septiembre 27, 2018, de Paradigma Sitio web: <https://goo.gl/UT5FJH>



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

Experiencias en el Desarrollo de Aplicaciones Móviles con Interfaces basadas en la Interacción Física

Experiences in the Development of Mobile Applications with Interfaces based on Physical Interaction

Pablo Martín Vera.¹

Rocío Andrea Rodríguez.¹

Mariano Kaimakamian Carrau .²

¹Centro de Altos Estudios en Tecnología Informática (CAETI)

²Universidad Abierta Interamericana (UAI)

Av. Montes de Oca 745, Ciudad Autónoma de Buenos Aires, Argentina

Resumen

El hardware de los dispositivos permite desarrollar aplicaciones más complejas que aprovechen los sensores de los mismos en beneficio de sus usuarios. Este artículo presenta diversas aplicaciones desarrolladas basadas en el uso de uno o más sensores disponibles, abriendo un panorama amplio a nuevas maneras de interactuar con los dispositivos. Las interfaces físicas permiten interactuar de forma más natural e intuitiva, moviendo, girando o simplemente apoyando el dispositivo sobre una superficie. Se podrá evidenciar como los desarrollos realizados proponen nuevas formas de interacción.

Palabras clave. Interfaces Físicas, Dispositivos Móviles, Sensores, Sensor de Proximidad, NFC, GPS, Brújula, Barómetro.

Abstract

The hardware of the devices allows the development of more complex mobile applications that take advantage of their sensors for the benefit of their users. This article presents different applications developed that are based on the use of one or more available sensors, opening a broad panorama to new ways of interacting with devices. The physical interfaces allow for interacting in a more natural and intuitive way, moving, rotating or simply resting the device on a surface. It will be able to demonstrate how the developments made propose new forms of interaction.

Keywords: Physical Interfaces, Mobile Devices, Sensors, Proximity Sensor, NFC, GPS, Compass, Barometer.

1. Introducción

Los dispositivos móviles han tenido una alta inserción en el mercado, sobre todo el teléfono celular. Llama la atención ver que hay más cantidad de líneas activas que de habitantes. “Según CICOMRA (Cámara de Informática y Comunicaciones de la República Argentina) en la actualidad en Argentina se encuentran unos 57.850.200 teléfonos celulares en servicio (con líneas activas), (CICOMRA, 2014). Esa cantidad de equipos supera ampliamente la cantidad de habitantes del país. Tomando como basamento Octubre del 2010: según el INDEC la cantidad de habitantes era de 40.117.096 (INDEC, 2010) y la cantidad de celulares con líneas activas era de 56.624.200 (CICOMRA, 2014). Por lo que la penetración del servicio de telefonía móvil es aproximadamente 141,15% (cantidad de líneas activas/cantidad de habitantes x 100)” (Giulianelli, Pons, Vera, et.al., 2012).

Los teléfonos de gama media y principalmente los de alta (smartphones) poseen una gran cantidad de componentes y sensores (ver figura 1).

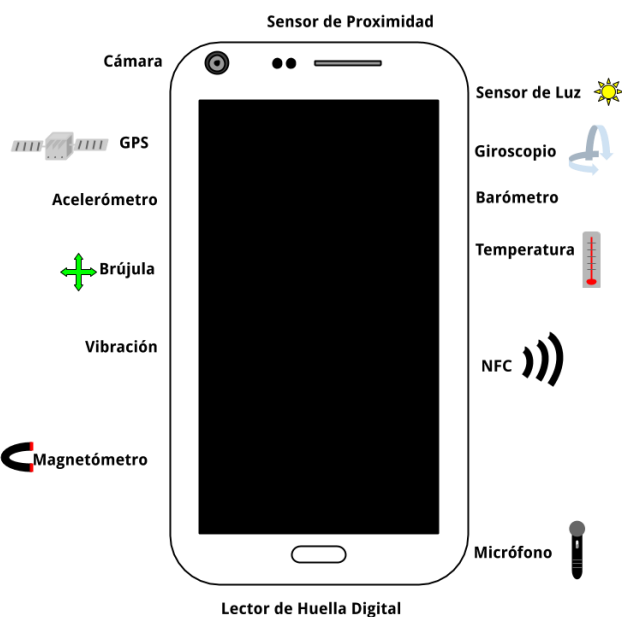


Figura 1. Componentes y Sensores de un Smartphone

El hardware de estos dispositivos permite realizar distintos tipos de aplicaciones las cuales pueden aprovechar los sensores de los dispositivos. Actualmente con HTML 5 (W3Schools, 2019) y las APIs del W3C (Consortio Web a nivel internacional) (W3C, 2017) es posible realizar desarrollos que accedan al hardware lo cual previamente era únicamente posible desde soluciones nativas (desarrolladas para un sistema operativo en particular) (Rodríguez, Vera, Martínez, et.al., 2014). Estos componentes y sensores podrán ser de gran utilidad al momento de querer realizar nuevas interfaces que sean más naturales e intuitivas.

1.1 Nuevas Interfaces

No sólo las pantallas táctiles permiten interactuar de una forma distinta con las aplicaciones, sino que también el conjunto de sensores y componentes que poseen permitirán generar nuevas formas de interacción, las cuales son más naturales y simplifican las acciones del usuario.

La Real Academia Española (RAE) define a la interfaz de dos formas distintas:

- “Conexión o frontera común entre dos aparatos o sistemas independientes” (RAE, 2019).
- “Conexión, física o lógica, entre una computadora y el usuario, un dispositivo periférico o un enlace de comunicaciones” (RAE, 2019).

La segunda definición es la que la RAE propone para el ámbito informático que agrega al usuario como parte de la posible conexión planteada. Si bien las definiciones son generales, es necesario considerar que se ha avanzado poco a poco a un cambio conceptual en las interfaces y deja de ser tan sólo si la interfaz es gráfica (GUI – Interfaz gráfica de Usuario) o de línea de comando, apareciendo múltiples opciones entre ellas: interfaces por voz (lo cual es muy común sobre todo en teléfonos celulares), gestos (muy usado por ejemplo en celulares y consolas de juegos)... la elección de la interfaz debe ser seleccionada según las características de la aplicación, las del dispositivo sobre el cual correrá e incluso las características de sus usuarios (si es que no está dirigida a un público en particular). Si bien en su mayoría las interfaces han mejorado la forma de interacción, algunas muy vinculadas a un hardware específico, continúan presentando dificultades en su uso. Un claro ejemplo es el manejo de un Smart-tv desde el control remoto, si bien en algunos casos se ha incorporado el manejo por voz, en la mayoría de los controles los usuarios deben desplazarse por medio de botones sin un mouse, ni pantalla táctil haciendo muy difícil por ejemplo escribir un texto para realizar una búsqueda. Del mismo modo en un Smartwatch al ser la pantalla reducida son limitadas las acciones que el usuario puede hacer desde dispositivo y también la información a mostrar ya que poseen un área de visualización

muy reducida. En estos casos puede recurrirse a tener una aplicación (app) que le permita al usuario desde otro dispositivo (ej: celular, tablet, etc...) visualizar la información de configuración o bien descargar información generada (ej: estadísticas de ritmo cardiaco, cantidad de pasos en el día...). Del mismo modo un celular podría convertirse en un control remoto mediante una app para facilitar el uso de un Smart-tv. En la figura 2 se representa el primer enfoque en donde el usuario tiene un conjunto de acciones a realizar y una primera interfaz que puede ser desde un dispositivo intermedio por ejemplo su teléfono celular y desde allí interactuar con el dispositivo final (por ejemplo Smart-tv, Smartwatch), lo cual no quita que en forma menos amena y hasta en algunos casos de acciones reducidas podría tener una interfaz directa con el dispositivo 2.

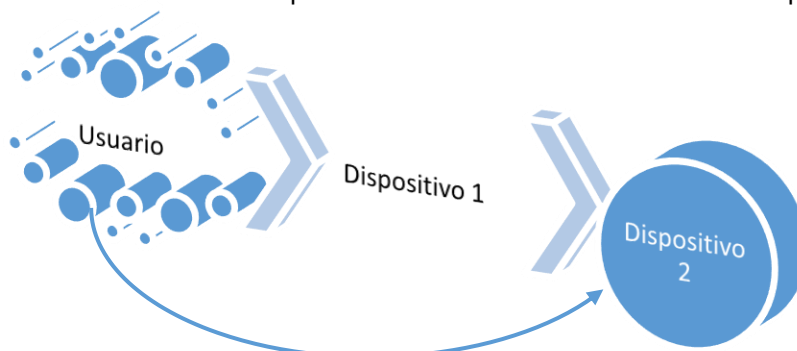


Figura 2. Niveles de interfaces para el acceso al Dispositivo 2, en forma directa o por medio de otro dispositivo

Mucho se menciona sobre las interfaces en cuanto a su diseño, interacción... pero también se cuenta con interfaces que deben estar "ausentes", lo esperado es que el usuario no deba interactuar con la aplicación. A modo de ejemplo, puede plantearse el caso de una alarma conectada en una casa, si la alarma funciona correctamente salvo en el caso de detectar una intrusión la alarma pasará inadvertida. En las redes de área personal BAN (Body Area Network) (Seyedi, Kibret, Lai, et.al., 2013), algo parecido debe suceder, el usuario podría tener un conjunto de dispositivos vestibles (wereable) entre ellos el smartwatch, así como un gran número de dispositivos los cuales pueden estar incorporados en prendas diarias. Si los mismos no sensan ninguna anomalía o condición que deba ser informada, el usuario de dichas prendas debería no notar la presencia de estos sensores. "Actualmente se dispone de prendas inteligentes capaces de analizar la frecuencia cardiaca, las alteraciones en el ritmo y la morfología del ECG (Electrocardiograma)" (Pérez-Villacastín, Gaeta, 2015). "La ropa del futuro llevará incorporada sensores capaces de controlar nuestro estado de salud. De forma inmediata, los datos se transformarán en señales eléctricas que podrán ser enviadas a un ordenador o a cualquier dispositivo móvil, como un teléfono, para que las interprete un médico o el propio usuario" (Guerrero, 2012). En el 2016, una empresa privada en conjunto con la Universidad Ramón Llull (Barcelona, España) diseñaron una remera que según la temperatura corporal, por WIFI se comunica con el aire acondicionado y lo regula (ESeficiencia, 2016). Esto podría en gran escala ser aplicado a batas para pacientes hospitalizados, a adultos mayores que viven solos y tienen dificultad o están distraídos y olvidan regular la calefacción del ambiente, etc. Los puntos negativos serán la disponibilidad de dichas prendas masivamente, así como los costos que puedan tener las mismas. Es importante considerar que en este caso puede no existir una interfaz ni necesidad de conexión directa con el usuario, sino que en cambio pueden comunicarse con otra persona que lo supervise por ejemplo con un médico. En este caso se presenta otro enfoque en el que el usuario tiene el dispositivo vestible del cual puede desconocer como da los datos, que datos aporta o como se interpretan y es otro usuario quién accede a una interfaz en la cual obtiene dichos datos o recibe notificaciones y alertas (ver figura 3).

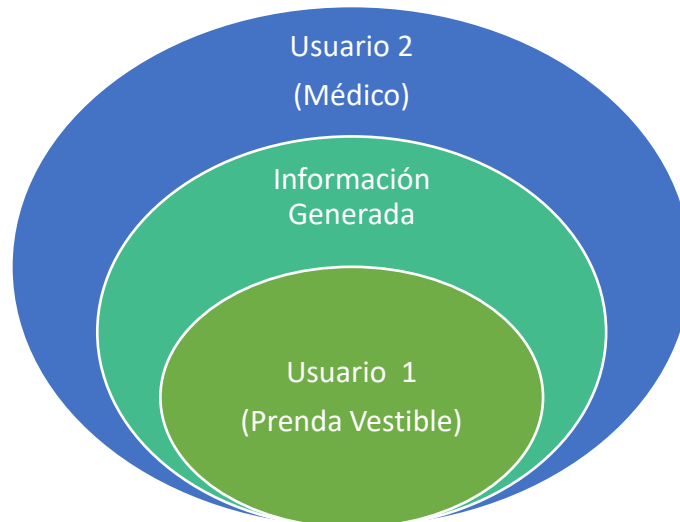


Figura 3. Dos usuarios del sistema uno provee datos sin necesitar conocer el sistema ni interactuar con él y el otro usuario en el nivel final es quién actúa con una aplicación en otro dispositivo (distinto al que genera la información, en este caso sensores incorporados en dispositivos vestibles).

En este enfoque en el que por ejemplo el usuario cuenta con un smartwatch, la tecnología estaría consigo en todo momento y lugar, como omnipresente. Y se la suele denominar “computación ubicua” (Caserotto, 2012), en donde los sensores están empotrados en la ropa del usuario siendo completamente transparente y no modificando los hábitos del mismo.

1.2 Interfaces con Interacción Física

Las interfaces con interacción física son aquellas en las que los movimientos de un usuario desencadenan una acción particular, por ejemplo, girar el celular en un sentido u otro para aumentar o disminuir la cantidad de minutos en un temporizador (ver figura 4), Esta aplicación debe poder controlarse de dos formas:

- De forma tradicional utilizando la pantalla táctil arrastrando la aguja al tiempo deseado.
- Sin necesidad de tocar la pantalla (por ejemplo, para cuando la persona está cocinando y tiene las manos sucias). Este segundo modo solo se habilitará cuando el dispositivo esté apoyado sobre una superficie plana y se configurará girándolo. Una vez que se detecte la finalización del movimiento se le dará un tiempo al usuario (por ejemplo 5 segundos) y se comenzará la cuenta atrás automáticamente.



Figura 4. Ejemplo de interfaz para un temporizador con interacción física (controlada mediante sensores de movimiento)

No sólo utilizar los sensores del dispositivo constituye el campo de las interfaces con interacción física, también deben considerarse los gestos. Motorola en algunos de sus teléfonos celulares incorpora gestos para activar la cámara de fotos, la linterna y controlar otras características; de este modo los usuarios eligen si activar o no dichos gestos pudiendo de forma rápida realizar las funcionalidades. En este ejemplo esta interfaz con interacción física requiere que el usuario active los gestos y los recuerde, pero es una alternativa ágil para acceder a dichas tareas. Se puede decir que existen dos interfaces aquella que se basa en buscar la aplicación puntual dentro de las pantallas y aquella que se genera como atajo por medio de gestos. Un claro ejemplo de interfaces con interacción física está dado por el sensor Kinect de Microsoft incorporado inicialmente en su consola de videojuegos XBOX y capaz de reconocer los movimientos del cuerpo de una persona. Este sensor ha dado origen a diversos proyectos académicos por ejemplo aplicados a niños con TEA (Trastorno del Espectro Autista), en donde se puede trabajar con diversos aspectos cognitivos e incluso de socialización, con pequeñas aplicaciones construidas para este fin (Rodríguez Rodríguez, 2017), (Contreras, 2019).

2. Contexto

La Universidad Abierta Interamericana (UAI) realiza actividades de vinculación para acercar a todos los niveles educativos a la tecnología y también a la investigación académica. Esto se genera a través de diversos eventos, como por ejemplo: la Semana de la Ciencia y la Tecnología (al que se unen diversas instituciones a lo largo de Argentina) (UAI, 2018), el CIITI (Congreso Internacional de Innovación Tecnológica) (UAI, 2019) ... también por medio de visitas abiertas que se realizan a los laboratorios del CAETI (Centro de Altos Estudios en Tecnología Informática), ver figura 5. Acercar la tecnología a la comunidad académica desde el nivel primario hasta la universidad, incluyendo también público en general en diversas actividades es un desafío que la Universidad se propone todos los años. Desde el laboratorio de Ingeniería de Software en lo que respecta a los proyectos vinculados con dispositivos móviles, surge una primera aplicación que será presentada más adelante en este artículo que ha sido utilizada en uno de estos encuentros en la que se hace uso del sensor de proximidad acercando el conocimiento a los niños más pequeños mediante un truco de magia, esto ha sido el puntapié inicial para luego generar una aplicación que permita navegar la web a través de gestos realizados en forma aérea detectados por el sensor de proximidad (lo cual se describirá en sección 2.1 del presente artículo). Luego se ha trabajado con la tecnología NFC (Near Field Communication) (NFC.org, 2017), que será abordada en la sección 2.2, para realizar eventos de proximidad lo que ha dado por resultado construir un tablero de llamadas rápidas.

También se cuenta con un desarrollo, el cual se está mejorando actualmente, de un prototipo para el monitoreo de personas mayores con necesidades especiales por medio de una aplicación que dispara alertas ante situaciones de peligro. Esta aplicación denominada VIGIA permite utilizar diversos sensores del dispositivo de forma no invasiva y preventiva (se presenta en forma resumida en la sección 2.3). Es por eso que este artículo presenta un compendio de las aplicaciones desarrolladas, que permitirá hacer énfasis en el uso de sensores generando interfaces físicas más intuitivas e incluso en algunos casos como lo es en la aplicación VIGIA más inmersivas en donde el usuario puede no detenerse a pensar en la aplicación en sí ya que la misma debe ser transparente sin interrumpir con su vida cotidiana. Es decir, las aplicaciones pueden tener objetivos tan diversos como entretener a niños por medio de la magia, que los lleve a indagar cómo se logra el resultado y descubrir la existencia de sensores, como acompañar a personas en su vida cotidiana sin invadirlos e inferir en sus tareas. Las interfaces deben favorecer la comunicación de los usuarios con la aplicación, por lo cual deben diseñarse las aplicaciones móviles tomando en cuenta los perfiles y necesidades particulares de los usuarios, así como las pantallas reducidas y otras limitaciones que pudieran tener los dispositivos en los cuales serán utilizadas.



Figura 5. A la izquierda visita a los laboratorios del CAETI, a la derecha curso de capacitación para docentes y alumnos en Sede Oeste (Castelar) - Argentina.

3. Trabajos Relacionados

Existen diversos trabajos que aprovechan el hardware de los dispositivos móviles, entre ellos volcados a la enseñanza para la simulación de laboratorios sobre todo utilizado en el campo de física, que permiten tomar mediciones de magnitudes por ejemplo la velocidad angular. Algunos autores presentan incluso la creación de un laboratorio móvil basado en los sensores del celular (Gonzalez, 2014), (Castro, 2013). El uso de sensores y otros componentes de los dispositivos móviles se ha extendido a otras áreas como, por ejemplo, en el campo de la salud para el monitoreo constante de los parámetros de un paciente a través de un dispositivo móvil, como ser un teléfono, entre los trabajos en esa área con desarrollo en Android se encuentran (Kong, 2016), (Robledo, 2015). También se proponen soluciones con el mismo enfoque utilizando dispositivos vestibles como puede ser un reloj (smartwatch), entre dichos trabajos se encuentran los de (Kulkarni,2016), (Jiang,2011).

Distintos autores abordan la temática de las interfaces físicas, también conocidas como interfaces tangibles (por sus siglas en inglés TUI). Entre los artículos que exploran las consideraciones para este tipo de interfaces se encuentra (Quin, 2009) que plantea métodos interactivos de diseño de productos que intentan romper la barrera establecida por la interfaz gráfica de usuario tradicional, vinculando la acción de copiar y pegar elementos con un instrumento físico. Para manejar una computadora o un determinado equipamiento aparecen este tipo de interfaces, como así también para el campo de la enseñanza “el uso de las Interfaces de usuario tangibles (TUI) como tecnología educativa ha ganado un interés sostenido a lo largo de los años con un acuerdo común sobre su capacidad innata para involucrar e intrigar a los estudiantes en pedagogías de aprendizaje activo” (Attard, 2019). En esta área mucho de lo que se aborda es con una mesa de trabajo física sobre la cual apoyan objetos físicos que son reconocidos y digitalizados (Bonillo, 2019), (Reyes, 2019), (Attard, 2019).

Más allá de las aplicaciones concretas el manejo de la interfaz por medio de gestos es un área de gran importancia, entre otros trabajos (Gupta, 2016), (Murao, 2011), (Trong, 2019) abordan la temática sobre el sistema operativo Android.

Todos los trabajos mencionados tienen puntos de contacto con el presente artículo. A continuación, se presentarán las soluciones desarrolladas por nuestro grupo de investigación: 2 vinculadas con el sensor de proximidad, 1 basada principalmente con el uso de NFC y 1 que integra el uso de diversos sensores.

4. Soluciones desarrolladas

Cada una de las soluciones desarrolladas a partir de la base de incorporar uno o más sensores para el uso de las mismas, es por ello que cuentan con interfaces con interacción física. Algunas de las interfaces tan sólo requieren realizar un gesto aéreo identificado detectado por el sensor de proximidad o apoyar el celular sobre una base con etiquetas NFC, otras en cambio, consideran en su desarrollo un conjunto de sensores disponibles en los smartphones. En esta sección se presentan los sensores utilizados para las distintas aplicaciones creadas.

4.1. Sensor de Proximidad

Se describe a continuación dos aplicaciones desarrolladas haciendo uso del sensor de proximidad:

1. **Mago Colorido:** Se desarrolló una aplicación web a la que se denominó Mago Colorido, en la que según el tiempo de detección de obstrucción del sensor de proximidad muestre un color en la pantalla. ¡Elijan un color! A partir de allí los niños pequeños pudieron evidenciar esta especie de truco de magia. Los únicos colores seteados eran: Rojo, Verde, Azul, Amarillo y Rosa. La persona que tiene el teléfono celular (el mago) acerca su mano a la pantalla del teléfono y hace aparecer el color en la pantalla. Para esto la aplicación da un feedback mediante vibración, de forma tal que una vibración identifica un color, dos vibraciones otro y así sucesivamente, tan solo debe acordarse el orden de los colores. No se setearon más colores dado que el tiempo total para esperar al último color aumentaría significativa, así como la posible falla del mago por no recordar el orden de los colores, fue por eso que se hizo una encuesta entre los conocidos del grupo de investigación para probar los colores que se repiten con más frecuencia. Llamativamente no sólo los niños sino las personas adultas dicen inicialmente uno de los 4 colores primeramente seteados apenas se les consulta, a lo cual luego se agregó el rosa. Ver el asombro en los rostros de los niños e intentar decir otro color para ver si también funcionaba es una forma de medir el éxito de esta pequeña aplicación. Se ha probado también con personas más grandes que ya al segundo intento ofrecen otras ocurrencias de colores: Bordo, fucsia, Naranja, etc. Para esto se configuró una página que decía no tengo ese color. La motivación es que el público se pregunte ¿cómo lo logran?, de hecho, con público adulto en general no asociado a la tecnología, nos ha ocurrido que presuponen que el truco se efectúa por medio de la voz, hasta nos han escrito el color en un papel para que el dispositivo “no los escuche”. Esta

pequeña aplicación que permite descubrir un sensor del dispositivo que puede explicarse claramente a los niños y al público en general, ya que es el mismo que se utiliza convencionalmente cuando se apaga la pantalla al momento de llevar el teléfono celular a la oreja para atender un llamado, despierta la curiosidad en los muchos usos que se le puede dar a los sensores del dispositivo. Por otra parte, el desarrollo realizado para lograr esto permitió investigar en mayor medida sobre el funcionamiento del sensor y trasladar el conocimiento a otro tipo de aplicaciones.

2. Gestos de Proximidad: Con la base de experiencia previa en el uso del sensor de proximidad, se analiza más a fondo la tecnología para construir una nueva solución. Lo primero a saber es que el sensor de proximidad sólo mide si hay presencia de un objeto cercano ó no (por ejemplo: la oreja al realizarse una llamada), pero no mide distancia de dicho objeto con respecto al sensor. Los valores arrojados por el sensor es 0 cuando detecta un objeto ó superior cuando no lo detecta (ver figura 6).

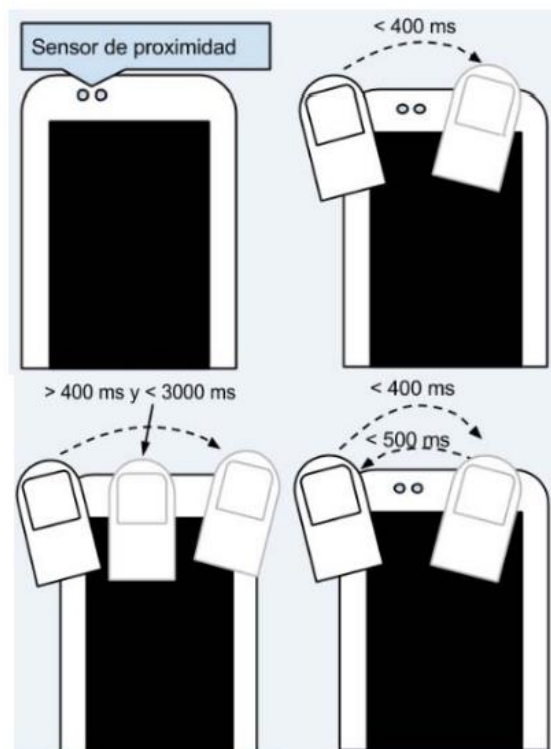


Figura 6. Tiempos configurados para la detección de pases sobre el sensor de proximidad

Se desarrolló una función en JavaScript que permite capturar eventos según si se obstruye el sensor o se libera, lo que permite pasar la mano o un dedo sobre el sensor y que este lo note y pueda reaccionar al respecto. Se creó una API que permite detectar 3 tipos de gestos: pase corto, pase largo ó pase doble (ida y vuelta), se realizaron diversas mediciones de tiempos que dependieron de dos factores: (1) reacción y velocidad de los usuarios (en lo que diferencia a un pase corto de uno largo), (2) exactitud del sensor, dado que entre los diferentes equipos hay sensores con detección más precisa que otros. Esto origino como resultado una publicación académica en el IEEE (Vera, Rodríguez, 2016) con los resultados de la prueba y la configuración de los tiempos que permitieron que la aplicación funcione correctamente más allá del usuario y de la calidad del sensor de proximidad del equipo utilizado.

Teniendo disponible este sensor y la función para detectar los eventos se proponen entonces implementar las siguientes ayudas a la navegación:

- Utilizar la doble pasada: por ejemplo, para mostrar en un menú desplegable las opciones de navegación, esto evitará tener que hacer scroll para moverse hasta donde se encuentre el menú o el ícono que abre el mismo. El menú desplegable podrá a su vez cerrarse mediante una pasada corta. Otra opción sino se desea utilizar en menú

desplegable es que la doble pasada lleve automáticamente a la parte superior de la pantalla donde se encuentran las opciones.

- Utilizar la pasada larga: para volver al “home” del sitio web. Creando un acceso directo desde cualquier lugar del sitio para ir a la página principal. Estos ejemplos de uso dejan en evidencia las alternativas adicionales que se le ofrecen a los usuarios finales, mediante el uso del sensor de proximidad.

Como puede observarse estas facilidades adicionales a la interfaz de la aplicación constituyen lo que se denomina una interfaz física, el usuario realiza gestos en el aire (cerca del sensor de proximidad) lo cual denominamos en forma general gestos de proximidad y mediante dichos gestos desencadena acciones de forma rápida que le evitarán por ejemplo hacer scroll en una página para regresar a la parte superior.

4.2. NFC

NFC (Near Field Communication) es una tecnología inalámbrica que permite transferir datos entre dispositivos que estén próximos entre sí (aproximadamente a 10 centímetros de distancia). La frecuencia de transmisión es de 13,56 MHz (no requiere de licencia para su uso) (NFC FORUM, 2013). La comunicación se realiza entonces cuando se cuenta con dos dispositivos con NFC (Rodríguez, Vera, Martínez, et. al., 2013):

- Activo: El dispositivo genera un campo electromagnético y este se usa para establecer la información leyendo datos o transmitiendo datos al otro dispositivo
- Pasivo: Sin posibilitar el campo electromagnético, tienen información que un dispositivo activo podrá leer o modificar.

Existen distintas soluciones implementadas que utilizan NFC, en Argentina la tarjeta SUBE en la cual se carga crédito que permite pagar los pasajes del transporte público posee un tag NFC. Es una tecnología simple de implementar y los tag de NFC tienen bajo costo. La tarjeta actúa como dispositivo pasivo. Son dispositivos activos las máquinas que tiene cada transporte público que leerán los datos de la tarjeta restarán el importe del viaje al crédito existente, etc... así también lo son los teléfonos con NFC, muchos usuarios tienen descargada la aplicación que permite ver el saldo de sus tarjetas, cargarles crédito, etc...

Se desarrolló una solución particular utilizando esta tecnología:

1. Tablero NFC: planificado para poder realizar llamadas rápidas, como si fuera una agenda de contactos en donde con sólo apoyar el teléfono en la fotografía de una persona se dispara el llamado. Esto podría ser utilizado por cualquier persona que quiera realizar una llamada en forma rápida, es sumamente útil para adultos mayores no familiarizados con la tecnología, así como para personas no videntes que requieren de un programa lector que las asista para realizar un llamado con el teléfono celular, podrían realizar una llamada rápida apoyando el celular en un tablero señalizado en braille. Pero no implica que la solución sea realizada exclusivamente para estos casos, imaginemos un empleado corporativo que tiene el teléfono celular, otorgado por la empresa para cuestiones laborales y que tiene junto a su estación de trabajo un tablero NFC con los contactos a los que debe llamar frecuentemente. Ofrecer una alternativa rápida de interactuar con un dispositivo es algo que pueden aprovechar todos los usuarios.

4.3. Utilización de diversos sensores

Se desarrolla una aplicación que integra distintos sensores que posibilitan monitorear a un usuario, inicialmente fue desarrollada para teléfonos celulares con Android (los prototipos iniciales con sus pantallas traducidas a inglés fueron presentados en un congreso del IEEE (Vera, Kaimakamian Carrau, Rodríguez, 2018) y luego migrada a Wear OS para ser utilizada desde un smartwatch. La conveniencia de su migración es que resulta más transparente, al usuario además las tecnologías vestibles como lo son el smartwatch aseguran que siempre tendrán consigo el dispositivo en este caso el reloj, por el contrario un teléfono celular podría ser apoyado sobre una mesa mientras la persona se desplaza. “La utilización de tecnologías del tipo “vestible” ha aumentado notablemente en los últimos años, permitiendo medir distintas variables fisiológicas, ambientales y de movimiento, a través de dispositivos anexados a la muñeca y/o la ropa del usuario” (Ponce, Lopez, Piccini, et. al., 2016).

La aplicación fue programada en Android, se utilizó SQL Lite y se desarrolló mediante una arquitectura de capas, que permiten separar la funcionalidad de la interfaz de usuario para luego poder adaptarla y migrarla a un dispositivo vestible basado en Android.

Esta aplicación utiliza diversos sensores entre ellos: GPS, Brújula, Acelerómetro, Barómetro, NFC, a continuación, se describe cómo surge y brevemente el funcionamiento de la misma.

1. Vigia: Cuando se crea esta aplicación se pone el foco en personas que requieran un monitoreo particular y no vivan acompañadas, de forma tal que no pueden ser monitoreadas. Es ideal para personas que requieran asistencia (Ej: Personas con discapacidad, Adultos mayores, etc.), es importante comprender que cada persona tiene necesidades particulares que deben ser consideradas. Las dificultades que puedan presentarse pueden estar dentro de sus viviendas, así como cuando salen a hacer compras o realizan un paseo, pudiendo perder el sentido de la orientación. Las tecnologías móviles pueden proporcionar soluciones para estos problemas, generando un sistema de control no invasivo que puede ser monitoreado por familiares y amigos. Sucede en muchos casos que las personas se encuentran restringidas a no salir de un radio sin un acompañante por temor a que algo pueda ocurrirles, este sistema les daría cierta libertad pudiendo informar en tiempo real de lo que se detecte que pueda causar dificultad en la persona monitoreada. Vigia es una aplicación integral en la que es posible configurar que tipo de alarmas se generarán en base a las necesidades de monitoreo particulares, así también el o los destinatarios de los avisos emitidos por esas alarmas. La aplicación utiliza los sensores en dispositivos como:
 - GPS, brújula, acelerómetros: Estos sensores en conjunto permiten saber por ejemplo si la persona se aleja de un radio predefinido, en qué dirección lo hace, con qué velocidad lo que permite conocer si está viajando en algún transporte.
 - Barómetro: Permite conocer la altitud a la que se encuentra, por ejemplo, una persona que tiene problemas en las rodillas y no debe subir ni bajar escaleras podría alertar si la persona está en un sótano o en la terraza de la vivienda.
 - NFC: Que permite tener un registro de la medicación que la persona va tomando. Si se utiliza un simple sistema de alarmas para avisar que se debe tomar un medicamento, la persona puede posponer la alarma o apagarla y omitir luego tomar la medicación. La incorporación del NFC genera el hábito de que cada vez que toma una medicación cargar a través de un tag NFC al sistema que ya tomó la medicación. Este tag se podría pegar en la caja del medicamento para obligar a la persona a moverse donde está el medicamento para apagar la alarma reduciendo la posibilidad de que no la ingiera.

El buen uso de esta aplicación puede ayudar a mejorar la calidad de vida y así también poder guardar registro de lo ocurrido para poder establecer estadísticas y parámetros que puedan contribuir a que el personal médico conozca mejor al paciente.

5. Limitaciones

Todas las aplicaciones mencionadas hacen uso del hardware presente en una gran variedad de dispositivos móviles actuales, sin embargo, en el mercado aún existen equipos que no disponen de los sensores necesarios. Un claro ejemplo es la disponibilidad de NFC que en ciertos mercados como el europeo se utiliza en gran escala, pero en mercados latinoamericanos, al no utilizarse tan frecuentemente, los fabricantes no los incluyen en las versiones regionales para abaratar costos. A modo de ejemplo en la figura 7 se muestra una captura de pantalla tomada de la página gsmarena.com. Esta figura muestra una parte de las especificaciones del celular Motorola Moto G8 plus donde puede verse que el sensor NFC está disponible dependiendo del mercado al cual esté enfocado.

COMMS	WLAN	Wi-Fi 802.11 a/b/g/n/ac, dual-band, Wi-Fi Direct, hotspot
	Bluetooth	5.0, A2DP, LE, aptX
	GPS	Yes, with A-GPS, GLONASS, BDS, GALILEO
	NFC	Yes (market dependent)
	Radio	FM radio
	USB	2.0, Type-C 1.0 reversible connector, USB On-The-Go

Figura 7. Características técnicas del Motorola Moto G8 plus

Otro de los problemas y limitaciones que se pueden encontrar es el versionado del sistema operativo. Frecuentemente equipos que tiene el hardware suficiente y los sensores por cuestiones comerciales no reciben actualizaciones en la versión del sistema operativo. Esto puede ser un inconveniente a la hora de desarrollar ya que ciertas APIs y funcionalidades que facilitan el uso de los sensores se van incorporando en los SDKs de las nuevas versiones de los sistemas operativos móviles haciendo que no puedan utilizarse en versiones anteriores o requieran de un desarrollo mucho más laborioso y tedioso.

Como último punto podemos nombrar que para las aplicaciones de monitoreo lo ideal es utilizar tecnologías vestibles como por ejemplo un smartwatch. Estos equipos aún no están lo suficientemente masificados y son costosos. Además, existen diversas marcas y algunas con sistemas operativos propietarios y de poca difusión. Es por eso que se eligió Android para el desarrollo ya que se puede utilizar en celulares y luego portar la solución a equipos con WearOS adaptando la interfaz.

6. Conclusiones

Existe una gran variedad de aplicaciones que pueden verse favorecidas con el uso del hardware actualmente disponible en los dispositivos móviles, conocer el funcionamiento de los distintos sensores permitirá desarrollar aplicaciones más especializadas. En este artículo se presentaron diversas aplicaciones con objetivos muy distintos que plantean diferentes formas de interacción desde los gestos aéreos al pasar la mano cerca del sensor de proximidad, hasta apoyar simplemente el dispositivo sobre una superficie para lanzar un evento como lo es una llamada por medio del tablero NFC o bien una aplicación más inmersiva la cual no requiere comunicación alguna con el usuario al que directamente protege, sino que lanza alertas a personas designadas en el círculo íntimo del usuario en caso de detectar riesgos. Este último desarrollo inició como un prototipo desde el teléfono celular, para ser migrado luego a un smartwach para que sea más inmersivo. La tecnología puesta al servicio de las personas ha sido el eje de este artículo enfocado en nuevas interfaces las cuales mediante interacción física permiten establecer otros esquemas de interacción.

REFERENCIAS

- Attard, G., De Raffaele, C., & Smith, S. (2019). TangiBoard: a toolkit to reduce the implementation burden of tangible user interfaces in education.
- Bonillo, C., Marco, J., Baldassarri, S., & Cerezo, E. (2019). KitVision toolkit: supporting the creation of cognitive activities for tangible tabletop devices. *Universal Access in the Information Society*, 1-29.
- Caserotto, G. A. (2012). *Computación ubicua, sensibilidad al contexto y mashups* (Tesis Especialista en Ingeniería de Software, Facultad de Informática, Universidad Nacional de La Plata).
- CICOMRA - Cámara de Informática y Comunicaciones de la República Argentina (2011). *Estadísticas – Telefonía Móvil, Argentina*
<http://www.cicomra.org.ar>
- Contreras, V. H. (2019). *Interfaces Gestuales: Herramienta innovadora para complementar el aprendizaje en niños con TEA* (Tesis Especialista en Ingeniería de Software, Universidad Nacional de La Plata).
- Eseficiencia: Portal de Eficiencia y Servicios Energéticos. (2016). *Desarrollan ropa inteligente conectada a la climatización del hogar*.
<https://www.eseficiencia.es/2016/09/16/desarrollan-ropa-inteligente-conectada-a-la-climatizacion-del-hogar>
- Giulianelli, D. A., Pons, C., Vera, P. M., Rodríguez, R. A., Trigueros, A., Fernández, V., & Marko, I. (2012). Técnicas para la validación de un sitio web móvil. In XIV Workshop de Investigadores en Ciencias de la Computación.
- Guerrero Teresa, (2012). *Ropa 'inteligente' para vigilar el estado de salud*.
<https://www.elmundo.es/elmundo/2012/05/03/nanotecnologia/1336065302.html>
- González, M. Á., González, M. Á., Llamas, C., Martín, M. E., Vegas, J., Martínez, Ó., ... & Herguedas, M. (2014, October). Mobile phones for teaching physics: using applications and sensors. In Proceedings of the Second International Conference on Technological Ecosystems for Enhancing Multiculturality (pp. 349-355).
- Gupta, H. P., Chudgar, H. S., Mukherjee, S., Dutta, T., & Sharma, K. (2016). A continuous hand gestures recognition technique for human-machine interaction using accelerometer and gyroscope sensors. *IEEE Sensors Journal*, 16(16), 6425-6432.
- INDEC (Instituto Nacional de Estadísticas y Censos), *Datos Definitivos-Censo 2010*, Argentina, (2010)
<https://www.indec.gob.ar/indec/web/Nivel3-Tema-2-41>
- Kong, X., Fan, B., Nie, W., & Ding, Y. (2016, October). Design on mobile health service system based on Android platform. In *Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC)*, 2016 IEEE (pp. 1683-1687). IEEE.
- Kulkarni, C., Karhade, H., Gupta, S., Bhende, P., & Bhandare, S. (2016, January). Health companion device using IoT and wearable computing. In *Internet of Things and Applications (IOTA)*, International Conference on (pp. 152-156). IEEE.
- Murao, K., Terada, T., Yano, A., & Matsukura, R. (2011, June). Evaluating gesture recognition by multiple-sensor-containing mobile devices. In *2011 15th Annual International Symposium on Wearable Computers* (pp. 55-58). IEEE.
- NFC Forum (2013). *About the technology*
<https://nfc-forum.org/what-is-nfc/about-the-technology/>

NFC.org,(2017). *Near field communication*
<http://nearfieldcommunication.org/>

Pérez-Villacastín, J., & Gaeta, E. (2015). ¿Prendas inteligentes para cuidar a las personas o personas inteligentes que utilizan prendas para cuidarse? *Revista Española de Cardiología*, 68(7), 559-561.

Ponce, S., Lopez, N., Piccinini, D., Roberti, M., Avetta, S., Andino, N., ... & Garcia, C. (2016). Desarrollo de un sistema vestible de medición de variables fisiológicas. In *Proc VII Congreso de Ingeniería Biomédica*.

Qin, J., Guan, Y., & Ji, H. (2009, November). TUI interactive product design. In *2009 IEEE 10th International Conference on Computer-Aided Industrial Design & Conceptual Design* (pp. 1455-1458). IEEE.

RAE - Real Academia Española (2019). *Interfaz*.
<https://dle.rae.es/?id=Lu1HQaO>

Rebolledo-Nandi, Z., Chavez-Olivera, A., Cuevas-Valencia, R. E., Alarcon-Paredes, A., & Alonso, G. A. (2015, March). Design of a versatile low cost mobile health care monitoring system using an android application. In *Health Care Exchanges (PAHCE), 2015 Pan American* (pp. 1-4). IEEE.

Reyes-Flores, A., Mezura-Godoy, C., & Benítez-Guerrero, E. (2019, October). Requirements Analysis for Tangible User Interfaces for Collaborative Use. In *2019 IEEE International Conference on Engineering Veracruz (ICEV)* (Vol. 1, pp. 1-7). IEEE.

Rodríguez Rodríguez, A. (2017). Recursos digitales en Educación Especial. Propuesta educativa para centrar la atención de niños con TEA a través del sensor Kinect.

Rodríguez, R. A., Vera, P. M., Martínez, M. R., Pons, C., Valles, F. E., & de La Cruz, L. V. (2014). Reducing the Gap between Native and Web Applications.

Rodríguez, R. A., Vera, P. M., Martínez, M. R., Giulianelli, D. A., & Valles, F. E. (2013). Context Aware Applications on Mobile Environments-Engaged by the use of NFC. In *International Conference on Multimedia, Scientific Information and Visualization for Information Systems and Metrics* (pp. 29-31).

Seyedi, M., Kibret, B., Lai, D. T., & Faulkner, M. (2013). A survey on intrabody communications for body area network applications. *IEEE Transactions on Biomedical Engineering*, 60(8), 2067-2079.

Trong, K. N., Bui, H., & Pham, C. (2019, October). Recognizing hand gestures for controlling home appliances with mobile sensors. In *2019 11th International Conference on Knowledge and Systems Engineering (KSE)* (pp. 1-7). IEEE.

UAI (2018), *Semana Nacional de la Ciencia y la Tecnología*.
<https://noticias.uai.edu.ar/facultades/tecnolog%C3%ADa-inform%C3%A1tica/xvi-semana-nacional-de-la-ciencia-y-la-tecnolog%C3%ADa/>

UAI (2019), *Congreso Internacional en Innovación Tecnológica Informática*.
<https://www.uai.edu.ar/ciiti/>

Vera, P. Kaimakamian Carrau, M., Rodríguez R. (2018). Mobile follow-up system for elderly and disabled people, *Congreso Argentino de Ciencias de la Informática y Desarrollos de Investigación (CACIDI)*, Buenos Aires, 2018, pp. 1-5.
doi: 10.1109/CACIDI.2018.8584361

Vera, P. M., & Rodriguez, R. A. (2016). Creating and Using Proximity Events on Mobile Websites. *IEEE Latin America Transactions*, 14(11), 4579-4584.

W3C (2017), *W3C Mission*
<https://www.w3.org/Consortium/mission>

W3schools (2019), *HTML 5 Introduction*.
https://www.w3schools.com/html/html5_intro.asp

NOTAS BIOGRÁFICAS



Pablo Martín Vera. Doctor en Ciencias Informáticas (UNLP - Universidad Nacional de La Plata). Ingeniero en Informática (UNLaM – Universidad Nacional de La Matanza). Docente de grado y posgrado en UNLaM, UTN (Universidad Tecnológica Nacional) y UAI (Universidad Abierta Interamericana). Es además director de proyectos de investigación en el CAETI (Centro de Altos Estudios en Tecnología Informática) de la UAI, destacándose entre sus principales áreas de interés la gamificación, las tecnologías móviles, la usabilidad y el desarrollo dirigido por modelos (MDD). En cuanto a la formación de recursos humanos es director de becarios y director de tesis de maestría y doctorado. Es revisor de distintas publicaciones académicas y revistas indexadas.



Rocío Andrea Rodríguez. Doctora en Ciencias de la Computación (UNLP–Universidad Nacional de La Plata), Ingeniera en Informática (UNLaM–Universidad Nacional de La Matanza). Profesora de Grado en UNLaM y UTN (Universidad Tecnológica Nacional); profesora de posgrado en UAI (Universidad Abierta Interamericana) y UNLaM. Directora de proyectos de investigación en el CAETI (Centro de Altos Estudios en Tecnología Informática) de la UAI UAI. Además es directora de becarios, pasantes y tesistas. Ha participado como jurado de tesis y es revisora de: artículos, proyectos de extensión universitaria y programas cofinanciados. Es autora de libros, capítulos de libros y artículos académicos.



Mariano Kaimakamian Carrau. Técnico Diseñador Multimedial (Escuela Da Vinci), Ingeniero en sistemas informáticos (Universidad Abierta Interamericana). Docente de grado en la Universidad Tecnológica Nacional. Investigador en el CAETI (Centro de Altos Estudios en Tecnología Informática) de la UAI. A nivel laboral, en los últimos diez años se ha desenvuelto en distintas áreas que incluyen el análisis y desarrollo de soluciones informáticas, hasta el liderazgo de equipo y gestión de proyecto; actualmente es arquitecto de software y DevOp, responsable de migraciones de sistemas stand alone a entornos web. Entre sus principales logros se encuentra haber participado en el desarrollo de la plataforma SECF dentro del marco del proyecto espacial nacional de la República Argentina.



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

EXPLORANDO LA INFLUENCIA DE LOS ROLES DE BELBIN EN LA CALIDAD DEL DISEÑO LÓGICO DE UNA BASE DE DATOS GENERADO POR ESTUDIANTES DE INGENIERÍA DE SOFTWARE

Raúl A. Aguilar Vera ¹
avera@correo.uady.mx

Adriana Peña Pérez Negrón ²
adriana.pena@cucei.udg.mx

Julio C. Díaz ¹
julio.diaz@correo.uady.mx

Juan P. Ucán Pech ^{1*}
juan.ucan@correo.uady.mx

¹ Facultad de Matemáticas, Universidad Autónoma de Yucatán, Mérida, Yucatán, México.

² Centro Universitario de Ciencias Exactas e Ingenierías, Universidad de Guadalajara Guadalajara, Jalisco, México

*Autor para Correspondencia

RESUMEN

En este artículo se presenta un estudio exploratorio en el que se analiza la influencia de los Roles de Belbin en actividades de desarrollo de software, particularmente, el Diseño Lógico de una Base de Datos; dicha actividad es considerada como un proceso de abstracción que consiste en transformar la representación de la Base de Datos vinculada a un problema, de un Modelo de Entidad Relación (Modelo Conceptual) a un Modelo Relacional (Modelo lógico). Las variables consideradas en el estudio fueron la Calidad del diseño y el Esfuerzo en la tarea. En virtud de ser un primer estudio (exploratorio), se utilizó una muestra basada en estudiantes avanzados de Ingeniería de Software. En el caso de la Calidad, los resultados permiten identificar diferencias significativas con el rol Monitor-Evaluador respecto de los otros seis roles, obteniendo éste un mejor grado de Calidad; en el caso de la variable Esfuerzo, no se identificaron diferencias entre tratamientos.

PALABRAS CLAVE

Experimentación, Diseño de Bases de Datos, Ingeniería de Software.

EXPLORING THE INFLUENCE OF BELBIN ROLES ON THE QUALITY OF THE LOGICAL DESIGN OF A DATABASE GENERATED BY SOFTWARE ENGINEERING STUDENTS

ABSTRACT

This paper presents an exploratory study that analyzes the influence of Belbin Roles in software development activities, particularly, the Logical Design of a Database; this activity is considered as an abstraction process that consists of transforming the representation of the Database linked to a problem, from a Relationship Entity Model (Conceptual Model) to a Relational Model (Logical Model). The variables considered in the study were the Quality of the design and the Effort in the task. By virtue of being a first study (exploratory), a sample based on advanced students of Software Engineering was used. In the case of Quality, the results allow to identify significant differences with the Monitor-Evaluator role with respect to the other six roles, obtaining a better grade of Quality; In the case of the Effort variable, no differences were identified between treatments.

KEYWORDS

Database Design, Experimentation, Software Engineering.

1. INTRODUCCIÓN

A medio siglo de la denominada “Crisis del Software”, la Ingeniería de Software (IS) dispone en la actualidad de un cuerpo de conocimientos aceptado en lo general por profesionistas e investigadores de la aún joven disciplina (Bourque & Fairley, 2014), no obstante, la dinámica de mejora en cuanto a calidad, tanto en sus procesos, como en los artefactos generados a través de los mismos, sigue siendo fuente de investigación para dicha disciplina ingenieril (Miramontes et al, 2016; Norambuena y Vega, 2017).

El estudio de los enfoques vinculados con los procesos de desarrollo, operación, mantenimiento y gestión del software, ha sido analizado con base en la relación de diversas variables, no obstante, el aspecto social intrínseco de la disciplina (Juristo & Moreno, 2001), ha llevado a considerar al factor humano, como un aspecto de singular interés para su investigación. En (Morales y Vega, 2018) se propone un catálogo de factores humanos críticos para el éxito de propuestas de mejora al proceso software, entre dichos factores se encuentra el rol que desempeña un Ingeniero de Software al interior del equipo de trabajo. En torno a dicho factor, Acuña, Juristo & Moreno (2006) caracterizan —mediante rasgos de personalidad— a los individuos para su asignación a los roles identificados en los equipos de desarrollo software, y utilizan la información del recurso humano de la organización en su conjunto, para identificar necesidades de entrenamiento.

El presente estudio, tiene como propósito explorar si los roles propuestos por Belbin (Belbin, 1993) tienen algún tipo de incidencia en tareas vinculadas con procesos software, en particular, en el diseño lógico generado para una base de datos, así como en el esfuerzo en la tarea.

2. ANTECEDENTES

Entre las investigaciones sobre los roles que desempeñan los integrantes de un equipo, existe hoy día diferenciación entre roles orientados a la actividad individual, es decir, que se encuentran descritos en función de la tarea que pueden desempeñar y los roles orientados al desempeño del equipo, es decir, se encuentran definidos en función de su comportamiento con sus pares al interior del equipo; este segundo tipo de rol, es conocido como rol de equipo y al parecer, su ausencia o presencia, tiene una influencia significativa en los logros del equipo (Aritzeta, Swailes & Senior, 2005).

De entre los estudios sobre roles de equipo, la propuesta de roles de Belbin (Bebin, 1981; Bebin, 1993) es sin lugar a dudas la más conocida entre consultores e investigadores, sobre todo, porque ofrece mecanismos para la identificación del rol primario que un individuo puede asumir en un equipo de trabajo, en función de su comportamiento, así como una propuesta sobre equilibrio y compatibilidad entre roles, al interior de un equipo de trabajo. En el ámbito de la IS, existen diversos estudios que se sustentan en la teoría de roles de Belbin para explorar mejoras tanto en los procesos de desarrollo, como en los productos generados con dichos procesos (Schoenhoff, 2001); la mayoría de los estudios utilizan como factor la integración de los equipos de desarrollo, considerando como alternativa —proceso controlado— equipos con roles compatibles (Pollock, 2009; Abdulrahman et al, 2017; Aguilar et al, 2020). Un segundo conjunto de trabajos, con menor evidencia al respecto, gira en torno al desempeño de los roles en tareas individuales (Henry & Stevens, 1999; Estrada y Peña, 2013).

3. DISEÑO LÓGICO DE BASES DE DATOS

El proceso de Diseño de una Base de Datos, como parte del desarrollo de un sistema de información, generalmente es concebido como un proceso de abstracción con diferentes niveles de representación, de acuerdo con (De Miguel, Piattini y Marcos, 1999) el proceso más consensuado es concebido en tres etapas y por tanto, tres niveles de representación: Diseño Conceptual, Diseño Lógico y Diseño Físico.

El diseño de una base de datos tiene su origen en las necesidades de datos especificadas en la fase de requisitos, lo cuales son transformados en un primer nivel de abstracción, en un Modelo Conceptual —Diseño Conceptual— en el que se representen los recursos de información de la organización con independencia de usuarios y/o aplicaciones en particular; el modelo de datos más recurrido para dicha abstracción, es el Modelo de Entidad/Relación, dicho modelo está basado en una percepción del mundo real consistente en objetos básicos llamados entidades y de relaciones entre estos objetos (Silberschatz, Korth & Sudarshan, 2020).

El segundo nivel de abstracción, tiene como objetivo transformar el Modelo Conceptual en un segundo nivel de abstracción, adaptándolo al modelo de datos en el que se apoye el Sistema de Gestión de Base de Datos (SGBD) que se vaya a utilizar para implementación —Diseño Lógico. Durante el siglo XX los sistemas relacionales dominaron el mercado y por ello el Modelo Relacional ha sido utilizado como modelo por excelencia para esta segunda etapa del proceso; dicho modelo es sin lugar a dudas el fundamento de la tecnología moderna de base de datos (Date, 2004), se ocupa de tres aspectos principales de la información: la estructura de datos, la manipulación de datos y la integridad de los datos; se basa en la teoría matemática de las relaciones, y los datos se estructuran lógicamente en forma de relaciones, comúnmente denominadas tablas. Debido a que el Modelo de Entidad/Relación y el Modelo Relacional comparten los mismos principios básicos de diseño, es posible aplicar un conjunto de reglas de derivación que permiten la transformación del Modelo Conceptual al Modelo Lógico Relacional (Silberschatz, Korth & Sudarshan, 2020).

Finalmente, el Diseño Físico tiene como propósito la implementación, lo más eficiente posible, del modelo lógico; para dicho proceso, se utiliza el sub-lenguaje de definición de datos del SGBD.

4. PLANIFICACIÓN DEL EXPERIMENTO CONTROLADO

Una de las metodologías empíricas más recurridas en el ámbito de la IS, es la experimentación, particularmente, la experimentación en entornos controlados (Basili, Selby & Hutchens, 1996); dicha metodología nos ayuda a identificar y en su caso, comprender las posibles relaciones entre factores y variables dependientes, ambos parámetros inmersos en quehacer del proceso software. Entre los elementos característicos de los estudios controlados encontrados en la literatura, se destaca el uso de grupos de estudiantes como sujetos experimentales; en (Genero, Cruz-Lemus y Piattini, 2014) los autores indican que dicha muestra académica permite al investigador obtener evidencia preliminar para confirmar o refutar hipótesis que pueden ser contrastadas posteriormente en contextos industriales.

El presente estudio tiene como propósito explorar, mediante la ejecución de un experimento controlado, la influencia de los roles de Belbin en la Calidad del Diseño Lógico de una Base de Datos, generado por estudiantes de Ingeniería de Software —previamente caracterizados con los roles de Belbin; así como en el Esfuerzo en la tarea.

4.1 Factor y Tratamientos

El factor considerado en el estudio es definido como el rol desempeñado por el sujeto en la tarea de Diseño Lógico de una Base de Datos, particularmente, se identifican nueve tratamientos, los cuales se corresponden con los nueve roles propuestos por Belbin (ver tabla 1).

Tabla 1. Caracterización de los Roles de Belbin

Tipo	Rol	Características
Acción	<i>Impulsor (Sharper: S)</i>	Retador, dinámico, trabaja bien bajo presión. Tiene iniciativa y coraje para superar obstáculos.
	<i>Implementador (Implementer: I)</i>	Disciplinado, leal, conservador y eficiente. Transforma las ideas en acciones.
	<i>Finalizador (Completer-Finisher: CF)</i>	Esmerado, concienzudo, ansioso. Busca los errores y las omisiones. Realiza las tareas en el plazo establecido.
Mental	<i>Cerebro (Planner: P)</i>	Creativo, imaginativo, poco ortodoxo. Resuelve problemas difíciles.
	<i>Monitor Evaluador (Monitor-Evaluator: ME)</i>	Serio, perspicaz y estratega. Percibe todas las opciones. Juzga con exactitud.
	<i>Especialista (Specialist: SP)</i>	Solo le interesa una cosa a un tiempo, cumplidor del deber. Aporta cualidades y conocimientos específicos.
Social	<i>Coordinador (Chairman: CH)</i>	Maduro, seguro de si mismo. Aclara las metas. Tiene iniciativa y coraje para superar obstáculos.
	<i>Investigador de Recursos (Resource Investigator: RI)</i>	Extrovertido, entusiasta, comunicativo. Busca nuevas oportunidades. Desarrolla contactos.
	<i>Cohesionador (Teamworker: TW)</i>	Cooperador, apacible, perceptivo y diplomático. Escucha y evita enfrentamientos.

4.2 Hipótesis y Variables

El primer par de hipótesis, utiliza como variable dependiente, una métrica vinculada con la Calidad del Diseño Lógico de la Base de Datos (DL-BD) generado por los estudiantes de IS.

- H₀₁: Las medias respecto de la calidad del DL-BD generadas por cada uno de los roles, no presentan diferencias.
- H₁₁: Las medias respecto de la calidad del DL-BD generadas por cada uno de los roles, difiere en al menos un par de éstos.

El segundo par de hipótesis, utiliza como variable dependiente, el Esfuerzo utilizado por los estudiantes de IS, para concluir la tarea de generar el Diseño Lógico de la Base de Datos (DL-BD).

- H₀₂: El esfuerzo registrado para generar el DL-BD por cada uno de los roles, no presenta diferencias significativas.
- H₁₂: El esfuerzo registrado para generar el DL-BD por cada uno de los roles, presenta diferencias significativas en al menos un par de éstos.

4.3 Métricas consideradas en el estudio

Para evaluar la calidad del DL-BD, seleccionamos como factor la Correctitud (McCall et al, 1977), entendido ésta como:

- Correctitud: El DL-BD se corresponde con las especificaciones del artefacto que le precede, en este caso, el Diseño Conceptual de la Base de Datos.

En virtud de que dicho factor es difícil de identificar y ante la falta de terminología estándar, hemos seleccionado como métrica, la Completitud, entendida ésta como:

- Completitud: Grado en el que el DL-BD generado por el sujeto experimental, es completo en comparación con los elemento del Diseño Lógico Correcto (DLC).

Operacionalmente, la Completitud (CO) será medida en función de un promedio entre los indicadores de: Relaciones o Tablas (IRE), Claves primarias (ICP) y Vínculos entre Relaciones (IVR), tal y como se especifica en las fórmulas I, II, III y IV, que se especifican a continuación:

$$CO = (IRE + ICP + IVR) / 3 \quad (I)$$

$$IRE = \frac{\sum_{i=1}^n \frac{(\sum_{j=1}^m X_{ij})}{m}}{n} \quad (II)$$

Donde X_{ij} es un valor dicotómico que representa la posesión o no del atributo j de la tabla i , de acuerdo con las n tablas del DLC

$$ICP = \frac{\sum_{i=1}^p Y_i}{p} \quad (III)$$

Donde Y_i es un valor dicotómico que representa la posesión o no de la clave primaria de la tabla i , de acuerdo con las p claves del DLC

$$IVR = \frac{\sum_{i=1}^q Z_i}{q} \quad (IV)$$

Donde Z_i es un valor dicotómico que representa la posesión o no del vínculo i entre dos tablas, de acuerdo con los q vínculos del DLC

En cuanto a la segunda variable dependiente, el Esfuerzo, operacionalmente se considerará el Tiempo Medio —en minutos— utilizado por un estudiante para concluir la tarea, en este caso, se obtendrá mediante la diferencia entre la hora de finalización y la hora de inicio.

4.4 Diseño Experimental

El diseño experimental que resulta más apropiado para nuestro estudio, es el diseño factorial con una fuente de variación, y nueve tratamientos para las dos variables dependientes (ver tabla 2).

Tabla 2. Diseño Experimental

Factor	Tratamiento	Variable Dependiente
Rol	Impulsor, Implementador, Finalizador, Cerebro, Monitor-Evaluador, Especialista, Coordinador, Investigador de Recursos y Cohesionador	Calidad Esfuerzo

4.5 Propuesta de Análisis Estadístico

Para el análisis de los datos, se generará una primera sección de estadística descriptiva con tablas de resumen para los valores obtenidos, así como gráficos de caja y bigotes para analizar visualmente el comportamiento de los mismos.

En el caso del análisis inferencial para las variables dependientes Calidad y Esfuerzo, elegimos el análisis de varianza de una vía, debido a que nos permite realizar pruebas de hipótesis para determinar si existe o no diferencias significativas entre las medias de los valores recogidos en la variable dependiente, para los diferentes tratamientos. El análisis de la varianza (Gutiérrez y De la Vara, 2012) es una técnica que permite construir con los datos, un modelo estadístico que describe el impacto de un solo factor categórico sobre una variable dependiente, el modelo lineal asociado es el siguiente:

$$Y_{ij} = \mu + \beta_i + \varepsilon_{ij} \quad (V)$$

donde Y_{ij} es la ij -ésima medición, μ es un parámetro común para todos los tratamientos, denominado media global, β_i es un parámetro que mide el efecto del i -ésimo tratamiento, y ε_{ij} es el error sistemático atribuible a la medición Y_{ij} .

5. EJECUCIÓN DEL EXPERIMENTO

Para el análisis de los datos, se generará una primera sección de estadística descriptiva con tablas de resumen para los valores obtenidos, así como gráficos de caja y bigotes para analizar visualmente el comportamiento de los mismos.

5.1. Participantes / Sujetos Experimentales

La muestra por conveniencia estuvo conformada por 33 estudiantes que cursaban la asignatura Experimentación en Ingeniería de Software, ubicada en el séptimo semestre del Plan de Estudios vigente de la carrera antes citada. A los 33 participantes se les administró el instrumento de auto-percepción propuesto por Belbin(1981), con el fin de identificar el rol primario que cada estudiante podría desempeñar; cabe destacar que dicho instrumento no incluye al rol especialista, y en el caso de la muestra, ningún estudiante estuvo identificado con el rol de investigador de recursos.

La tabla 3 presenta la muestra total de los 33 sujetos experimentales distribuidos en los siete tratamientos resultantes; para el análisis de la variable Calidad; tres productos fueron descartados en virtud de que los archivos digitales no resultaban claros para su análisis, así mismo, para la variable Esfuerzo, dos sujetos no registraron el tiempo de finalización de la tarea.

Tabla 3. Muestra Experimental

Rol	Muestra (#)		
	Total	Calidad	Esfuerzo
Cerebro	2	2	2
Cohesionador	4	3	4
Coordinador	4	4	2
Finalizador	8	7	8
Implementador	5	4	5
Impulsor	5	5	5
Monitor Evaluador	5	5	5
Total	33	30	31

5.2. Sesión Experimental

Para el experimento, el profesor —autor principal— realizó en una sesión previa a la sesión experimental, un repaso del tema vinculado al Diseño Lógico de Bases de Datos, tema analizado en una asignatura del sexto semestre denominada Diseño de Bases de Datos.

Durante la sesión experimental, el profesor entregó el Diseño Conceptual de una Base de Datos (figura 1), aclaró dudas del Modelo, y solicitó que registren la hora de inicio de la tarea, también proporcionó instrucciones para que los sujetos registren el tiempo de finalización, y posteriormente digitalicen y suban a la plataforma el Modelo Lógico generado.

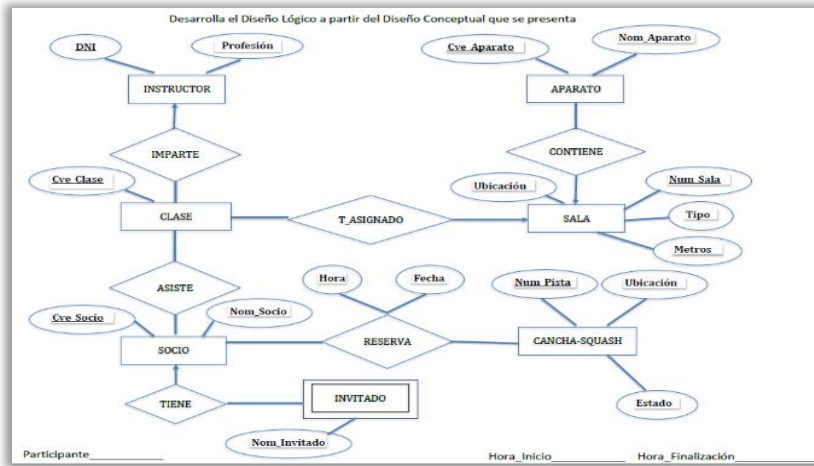


Figura 1. Diseño Conceptual de la BD proporcionado

Al final de la sesión, los sujetos entregaron el modelo lógico indicando los datos de identificación (ver figura 2), siendo dichos documentos, los objetos experimentales utilizados en la fase del análisis experimental.

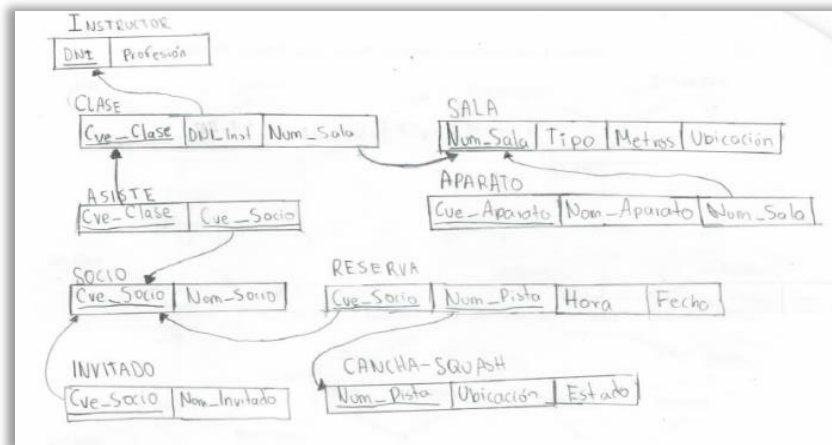


Figura 2. Ejemplo de Objeto experimental (Diseño lógico de la BD)

6. ANÁLISIS EXPERIMENTAL

En esta sección se presenta el análisis estadístico descriptivo e inferencial, para las dos variables dependientes, con base en los datos obtenidos en el experimento controlado.

6.1 Análisis Descriptivo

La Tabla 4 presenta algunas de las medidas más importantes de tendencia central y variabilidad para la variable Calidad; podemos observar que el rol Monitor-Evaluador presenta la media más elevada, y después del rol Cerebro, es el segundo tratamiento con menor variabilidad y el tercero con menor rango; cabe mencionar que ambos roles son clasificados como de tipo mental.

Tabla 4. Resumen estadístico de la variable Calidad

Rol	#	μ	σ	Min	Max	Rango
Cerebro	2	0.6489	0.0272	0.6296	0.6682	0.03858
Cohesionador	3	0.6857	0.1406	0.5555	0.8348	0.2793
Coordinador	4	0.7295	0.1030	0.6296	0.8379	0.2083
Finalizador	7	0.6920	0.1235	0.6172	0.9629	0.3456
Implementador	4	0.6172	0.0400	0.5740	0.6667	0.0925
Impulsor	5	0.7574	0.1483	0.5925	0.9583	0.3657
Monitor Evaluador	5	0.9484	0.0369	0.9027	1.0	0.0972

Para analizar visualmente el comportamiento de los datos, generamos un diagrama de caja y bigotes; dicho gráfico nos permite observar la dispersión y la simetría de los conjuntos de datos para cada uno de los tratamientos, así como la comparación entre los mismos; en la figura 3 podemos observar que el comportamiento desfasado de los datos del tratamiento Monitor-Evaluador, esto nos induce pensar que posiblemente exista una diferencia con los otros seis tratamientos.

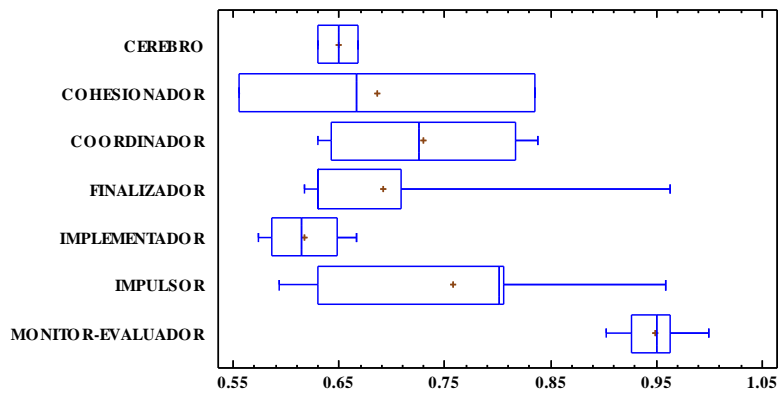


Figura 3. Diagrama de Caja y Bigotes para la variable Calidad

En la Tabla 5 se listan algunas de las medidas de tendencia central y variabilidad más importantes para la variable Esfuerzo; podemos observar que los roles de tipo social —Cohesionador y Coordinador— son los que presentaron mayor esfuerzo en tarea, siendo el Coordinador el rol con menor variabilidad de entre los siete tratamientos. Observamos también que el mayor tiempo destinado para la tarea ocupó apenas una tercera parte del tiempo planificado para la tarea —30 minutos.

Tabla 5. Resumen estadístico de la variable Esfuerzo

Rol	#	μ	σ	Min	Max	Rango
Cerebro	2	14.0	5.6568	10.0	18.0	8.0
Cohesionador	4	22.75	5.4390	17.0	30.0	13.0
Coordinador	2	20.0	1.4142	19.0	21.0	2.0
Finalizador	8	19.375	4.8384	12.0	28.0	16.0
Implementador	5	15.6	3.5071	11.0	20.0	9.0
Impulsor	5	15.4	4.8270	11.0	22.0	11.0
Monitor Evaluador	5	19.4	3.0495	15.0	23.0	8.0

Al igual que con la variable Calidad, generamos un diagrama de caja y bigotes para la variable Esfuerzo (ver figura 4); no observamos comportamientos que nos induzcan a pensar en diferencias entre tratamientos.

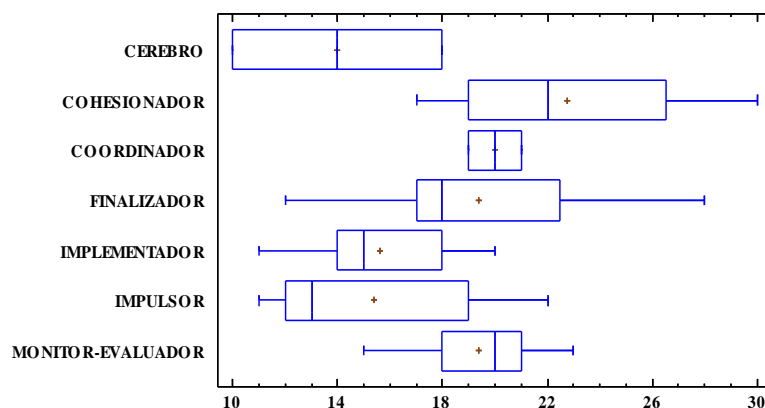


Figura 4. Diagrama de Caja y Bigotes para la variable Esfuerzo

6.2 Análisis Inferencial

Con el propósito de evaluar estadísticamente las diferencias entre los tratamientos de las variables Calidad y Esfuerzo, se aplicó el Análisis de Varianza de una vía; el resultado de evaluar con el ANOVA se ilustra en la tabla 6.

Tabla 6. ANOVA para las variables Calidad y Esfuerzo

Variable	Razón F	Valor de p
Calidad	4.69	0.0030*
Esfuerzo	1.85	0.1314

En el caso de la variable Calidad, el valor p de la prueba F es menor que 0.05 por lo cual, las hipótesis de nulidad H_0 puede ser rechazada, es decir, podríamos afirmar que existe al menos un par de tratamientos que presentan diferencias estadísticamente significativas entre sus medias, con un 5% de significancia.

Para identificar cuáles de los tratamientos son diferentes, generamos el gráfico de medias con intervalos de confianza de acuerdo con la prueba LSD, la cual se ilustra en la figura 5; dicho gráfico nos permite hacer una comparación visual y estadística de las medias de los tratamientos. Como podemos observar, al parecer el único tratamiento que presenta un desfase respecto de los otros seis tratamientos, es el correspondiente al rol Monitor-Evaluador.

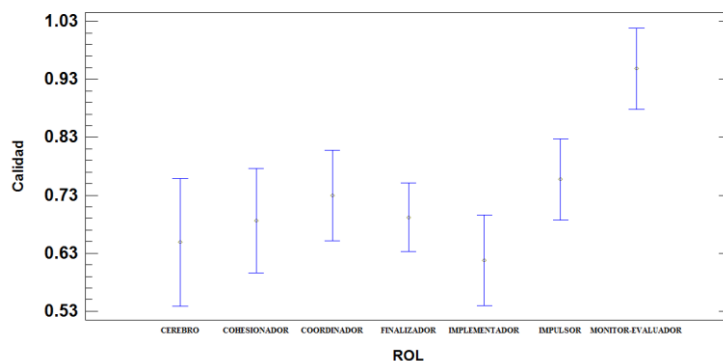


Figura 5. Gráfico de Medias (Método LSD) para la variable Calidad

La prueba de múltiples rangos para la variable calidad, mediante el método LSD, nos arroja diferencias entre 6 parejas de tratamientos tal y como se ilustra en la tabla 7; dicho análisis coincide con lo considerado con el análisis visual de la figura 5.

Tabla 7. Prueba de múltiples rangos para la variable Calidad

Contraste	Sig.	Diferencia	+/-Límites
Cerebro - Monitor-Evaluador	*	-0.299537	0.184751
Cohesionador - Monitor-Evaluador	*	-0.262757	0.161264
Coordinador - Monitor-Evaluador	*	-0.218904	0.14813
Finalizador - Monitor-Evaluador	*	-0.256437	0.129299
Implementador - Monitor-Evaluador	*	-0.331173	0.14813
Impulsor - Monitor-Evaluador	*	-0.191049	0.139658

En el caso de la variable Esfuerzo, el valor p de la prueba F es mayor que 0.05 por lo cual, deberíamos rechazar la hipótesis de nulidad H_0 , es decir, podemos afirmar que no existe evidencia, al menos con un 5% de significancia, de diferencias entre las medias de los tratamientos.

6.3 Validación de Modelos

Es importante mencionar que el modelo ANOVA tiene asociado tres supuestos que es necesario validar antes de utilizar la información que nos ofrece: (1) Los errores experimentales de sus datos se distribuyen normalmente, (2) No existe diferencia entre la varianza de los tratamientos y (3) existe independencia entre las muestras. Para validar el primer supuesto, usaremos el gráfico de probabilidad normal de residuos; dicha técnica permite evaluar visualmente su comportamiento en función de la distribución normal.

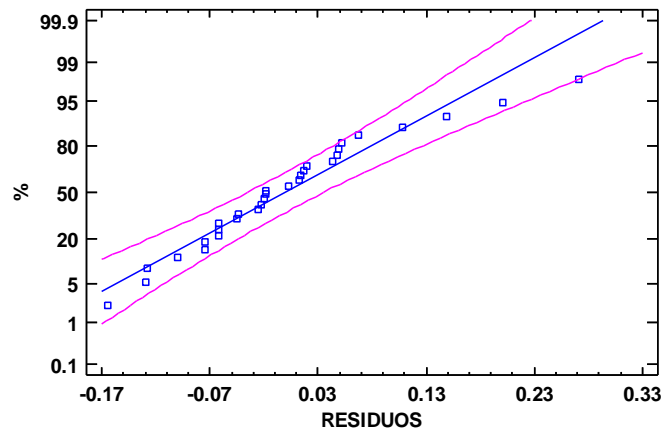


Figura 6. Gráfico de probabilidad normal para la variable Calidad

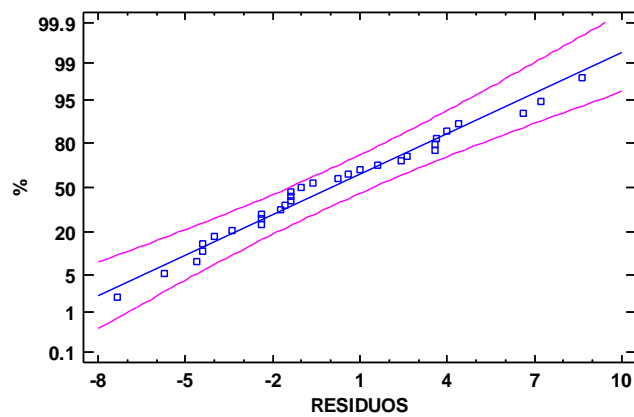


Figura 7. Gráfico de probabilidad normal para la variable Esfuerzo

Como se puede ver en las gráficas que se ilustran en las figuras 6 y 7, en ambas variables, Calidad y Esfuerzo, respectivamente, los puntos no muestran desviaciones de la diagonal, por lo que es posible asumir que los residuos tienen una distribución normal.

Por su parte, la prueba de Levene permite evaluar diferencias significativas entre las varianzas de los dos conjuntos de datos (homocedasticidad); en la tabla 8 podemos observar que en ambas variables (Calidad y Esfuerzo) el p valor es mayor que 0.05, lo cual nos indica que no existe una diferencia estadísticamente significativa entre las desviaciones estándar de los tratamientos, con un nivel del 95.0% de confianza.

Tabla 7. Prueba de Levene (Calidad y Esfuerzo)

Variable	Prueba	Valor de p
Calidad	1.78903	0.1459
Esfuerzo	0.784262	0.5907

Para validar el supuesto de independencia de los datos, generamos un gráfico de residuales versus secuencia, para cada variable (ver figuras 8 y 9).

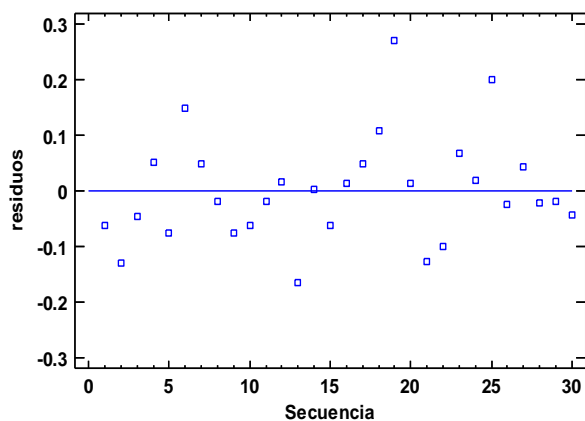


Figura 8. Gráfico de residuos vs. secuencia para la variable Calidad

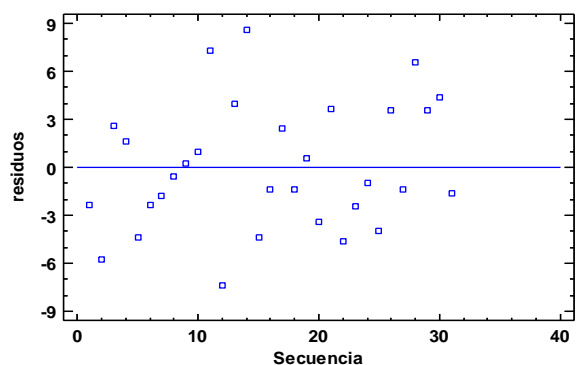


Figura 9. Gráfico de residuos vs. secuencia para la variable Esfuerzo

Podemos observar en ambos gráficos —figuras 8 y 9— que no existe una tendencia en los datos, por lo cual, es posible suponer, que los datos provienen de poblaciones independientes.

Superados los tres supuestos del modelo ANOVA para ambas variables, podemos afirmar que los comentarios derivados de la tabla 6, son válidos.

7. CONCLUSIONES

En este estudio presentamos un primer experimento controlado con el propósito de explorar si el uso de la teoría de roles de Belbin puede ser benéfico para identificar a los Ingenieros de Software con mejor habilidad en la tarea de Diseño Lógico de una Base de Datos.

Con el experimento controlado, encontramos que con la variable vinculada al producto —Calidad del Diseño Lógico de la BD— el rol Monitor-Evaluador presenta diferencias significativas, habiendo obtenido un grado de calidad mejor que el de los otros seis roles participantes en el experimento; el rol Monitor-Evaluador es uno de los dos roles de tipo mental identificados en el experimento; dicho resultado coincide parcialmente con los resultados reportados en (Henry & Stevens, 1999) en el sentido de que es uno de los tres roles identificados —Especialista, Monitor-Evaluador y Finalizador— con una buena aportación en la tarea de diseño; también coincide con lo comentado en (Schoenhoff, 2001), en torno a que es uno de los dos roles —Cerebro y Monitor-Evaluador— que presentan atracción por la fase de diseño.

Por su parte, para la variable esfuerzo en la tarea, los roles no mostraron diferencias significativas, posiblemente la tarea diseñada no fue lo suficientemente compleja para requerir mayores tiempos de dedicación, en la que se pudiesen observar diferencias entre los roles participantes.

Con lo obtenido en este estudio, los autores proponen continuar con el desarrollo de experimentos controlados, en tareas vinculadas con el proceso software, a fin de generar conocimiento en torno a posibles relaciones entre tipos de tareas y roles con un mejor desempeño en las mismas.

AGRADECIMIENTOS

Agradecemos el apoyo brindado por la Secretaría de Educación Pública (México) a través del proyecto P/PROFEXE-2020-31MSU0098J-13, así como a los alumnos de la Licenciatura en Ingeniería de Software de la UADY que participaron en el experimento.

REFERENCIAS

Abdulrahman, A., Omar, M., Ahmad, M. & Ahmed, V. (2017). An Analysis of Belbin Team Roles in Software Engineering Team. *Journal of engineering and Applied Sciences*, 12(Special Issue 5), pp. 6878-6883.

Acuña, S., Juristo, N. & Moreno, A. (2006). Emphasizing Human Capabilities in Software Development. *IEEE Software*, 23(2), pp.94–101.

Aguilar, R., Muñoz, M., Díaz, J. y Ucán, J. (2020). Explorando la influencia de los roles de Belbin en la especificación de requisitos de software. *Revista Ibérica de Sistemas y Tecnologías de la Información*. (36), pp. 34-49.

Aritzeta, A., Swailes, S. & Senior, B. (2005). *Team roles: Psychometric evidence, construct validity and team building*. Hull, UK: University Hull.

Basili, V., Selby, R. & Hutchens, D. (1996). Experimentation in Software Engineering. *IEEE Transactions on Software Engineering*, 12(7), pp. 733–743.

Belbin, M. (1993). *Team roles at Work*. Oxford, USA: Elsevier Butterworth Heinemann.

Belbin, M. (1981). *Management teams: Why they succeed or fail*. New York, USA: John Wiley & Sons.

Bourque, P & Fairley, R. (Eds.) (2014). *Guide to the Software Engineering Body of Knowledge (SWEBOK V3.0)*. IEEE Computer Society.

Date, C.J. (2004). *An introduction to database systems*. USA: Pearson Education

De Miguel, A., Piattini, M. y Marcos, E. (1999). *Diseño de Bases de Datos Relacionales*. España: Ra-Ma.

Estrada, E. y Peña, A. (2013). Influencia de los roles de equipo en las actividades del desarrollador de software. *Revista Electrónica de Computación, Informática, Biomédica y Electrónica*. (1), pp.1-19.

Genero, M., Cruz-Lemus, J. y Piattini, M. (2014). *Métodos de investigación en ingeniería de software*. España: Ra-Ma.

Gutiérrez, H. y De la Vara, R. (2012). *Análisis y Diseño de Experimentos (3ª ed.)*. Ciudad de México, México: McGraw Hill.

Henry, S. & Stevens, K. (1999). Using Belbin's leadership role to improve team effectiveness: An empirical investigation. *The Journal of Systems and Software*, 44, pp. 241-250.

Juristo, N. & Moreno, A. (2001). *Basics of Software Engineering Experimentation*. Boston, USA: Kluwer Academic Publishers.

McCall, J. A., Richards, P. K. & Walters, G. F. (1977). *Factors in Software Quality, Volumes I, II, and III*. US Rome Air Development Center Reports US Department of Commerce USA.

- Miramontes, J., Muñoz, M. Calvo-Manzano, J. y Corona, B. (2017). Establecimiento del estado del arte sobre el aligeramiento de procesos de software. *Revista Ibérica de Sistemas y Tecnologías de Información*, (17), pp. 16-25.
- Morales, N. y 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.
- Norambuena, B. y Vega, V. (2017). Minería de procesos de software: una revisión de experiencias de aplicación. *Revista Ibérica de Sistemas y Tecnologías de Información*, (21), pp. 51-66.
- Pollock, M. (2009). Investigating the relationship between team role diversity and team performance in information systems teams. *Journal of Information Technology Management*, vol. 20(1), pp .42-55.
- Schoenhoff, P.K. (2001). *Belbin's Company Worker, The Self-Perception Inventory and Their Application to Software Engineering Teams* (Master Of Science Thesis). USA: Virginia Polytechnic Institute and SU.
- Silberschatz, A, Korth, H. &Sudarshan, S. (2020). *Database Systems Concepts. 7th. ed.* USA: McGrawHill.

NOTAS BIOGRÁFICAS



Raúl Antonio Aguilar Vera obtuvo el grado de Doctor por la Universidad Politécnica de Madrid, España (Mención de Doctor Europeo) y el de Master 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 coordinador de la Sección Académica de Ingeniería de Software de la Academia Mexicana de Computación (AMEXCOMP). Su trabajo de investigación incluye las áreas de Ingeniería de Software e Informática Educativa.



Adriana Peña Pérez Negrón es doctora en Ingeniería de la Informática de la Universidad Politécnica de Madrid, España. Realizó una estancia investigación en la University of Salford, Reino Unido para la mención de “Doctor Europeo” Actualmente es profesor investigador en el Departamento de Ciencias Computacionales del CUCEI de la Universidad de Guadalajara, México. Es responsable del Cuerpo Académico Informática Aplicada del Programa para el Desarrollo Profesional Docente. Sus principales líneas de investigación son la Realidad Virtual y el estudio del comportamiento en medios digitales



Julio Cesar Díaz Mendoza es Ingeniero Industrial en Producción por el Instituto Tecnológico de Mérida (ITM). Maestro en Tecnologías de Información, por la Universidad Interamericana para el Desarrollo (UNID). Especialista en Docencia de la Universidad Autónoma de Yucatán. Actualmente es profesor en la Facultad de Matemáticas de la Universidad Autónoma de Yucatán, e imparte asignaturas en las carreras de Ingeniería de Software y Ciencias de la Computación. Su interés se enfoca al área de Ingeniería de Software Educativo.



Juan Pablo Ucán Pech es Doctor en Sistemas Computacionales por la Universidad del Sur, campus Mérida, México. Maestro en Sistemas Computacionales con especialidad en Ingeniería de Software por el Instituto Tecnológico de Mérida, México. Actualmente es profesor en la Facultad de Matemáticas de la Universidad Autónoma de Yucatán. Es miembro de la Academia Mexicana de Computación (AMEXCOMP). Su trabajo de investigación se centra en temas relacionados con la Ingeniería de Software, Ingeniería Web e Informática Educativa.



Esta obra está bajo una [licencia de Creative Commons Reconocimiento-NoComercial-CompartirIgual 2.5 México](https://creativecommons.org/licenses/by-nc-sa/2.5/mx/).

METUIGA “METODOLOGÍA PARA EL DISEÑO DE SISTEMAS BASADOS EN INTERFACES TANGIBLES DE USUARIO Y TÉCNICAS DE GAMIFICACIÓN” EN LA ENSEÑANZA DE PORCIONES MATEMÁTICAS PARA NIÑOS CIEGOS

Luis R. Ramos-Aguiar
roberto.ramos.nay@gmail.com

Francisco J. Álvarez-Rodríguez
fjalvar@correo.uaa.mx

Centro de Ciencias Básicas, Universidad Autónoma de Aguascalientes, Aguascalientes, Aguascalientes, México.

RESUMEN

A lo largo de su vida las personas con problemas visuales tienen retos significativos durante su proceso de aprendizaje, especialmente en las etapas más tempranas de este proceso. Se han desarrollado diferentes herramientas tecnológicas en las últimas décadas para apoyar a personas con problemas visuales cuando interactúan con computadoras. Sin embargo, aún existe una gran brecha tecnológica para este tipo de usuarios, específicamente en el diseño de estos sistemas los cuales en su mayoría no explotan la principal herramienta de las personas con problemas visuales para percibir el mundo “sus manos”. En esta investigación nos enfocamos en generar herramientas digitales siguiendo la metodología METUIGA utilizando interfaces tangibles de usuario y técnicas de gamificación; Se construyó una aplicación para la enseñanza de porciones matemáticas que incorpora la metodología planteada, utilizando los nuevos métodos para la implementación de interfaces tangibles de usuario y técnicas de gamificación. En esta investigación se realizaron pruebas de usabilidad obteniendo resultados satisfactorios.

PALABRAS CLAVE

Ingeniería, Interfaz Tangible, Modelo, Software, Desarrollo, Gamificación, Niños ciegos.

METUIGA "METHODOLOGY FOR THE DESIGN OF SYSTEMS BASED ON TANGIBLE USER INTERFACES AND GAMIFICATION TECHNIQUES" IN THE TEACHING OF MATHEMATICAL PORTIONS FOR BLIND CHILDREN

ABSTRACT

Throughout their lives, people with visual impairment present significant challenges during their learning process, especially in the early stages of this process. Different technological tools have been developed in the last decades to support visually impaired people when interacting with computers. However, there is still a big technological gap for this type of users, specifically in the design of these systems which mostly do not exploit the main tool of visually impaired people to perceive the world "their hands". In this research we focused on generating digital tools following the METUIGA methodology using tangible user interfaces and gamification techniques. An application was built for teaching mathematical portions that incorporates the proposed methodology, using the new methods for the implementation of tangible user interfaces and gamification techniques. In this research, usability tests were carried out with satisfactory results.

KEYWORDS

Blind Children, Development, Engineering, Gamification, Tangible Interface, Model, Software.

1. INTRODUCCIÓN

Durante muchos años, la investigación sobre la Interacción Humano-Computadora (IHC) se ha centrado en las Interfaces Gráficas de Usuario (IGU), que permiten la interacción con objetos digitales a través de la pantalla, el teclado y el ratón de la computadora. Esto ha hecho que la mayoría de las modalidades de interacción utilizadas para los juegos educativos exploten en gran medida el canal visual para presentar y mostrar los contenidos educativos con el fin de permitir a los estudiantes aprenderlos por imitación. Desafortunadamente, este enfoque no es adecuado para niños con discapacidad visual. Por lo tanto, se deben proponer soluciones alternativas pero equivalentes más accesibles y adecuadas para aquellos que no pueden ver.

Recientemente, un nuevo campo de investigación que involucra interfaces tangibles ha evolucionado fluidamente, inspirando a los desarrolladores e investigadores de IHC a reemplazar los elementos de la interfaz gráfica con objetos intuitivos y cotidianos. Convirtiéndose en una gran herramienta para la enseñanza de temas educativos a personas con problemas visuales, debido a que a lo largo de su vida se les enseña a construir, manipular y navegar por el mundo físico utilizando el sentido del tacto a través de sus manos.

En este artículo presentamos una aplicación para la enseñanza de porciones matemáticas en niños con problemas visuales entre 6 y 10 años. Para la construcción de la aplicación se utilizó la metodología METUIGA la cual tiene como objetivo la construcción de sistemas interactivos utilizando interfaces tangibles de usuario y técnicas de gamificación.

El resto del documento está organizado de la siguiente manera: La Sección 2 presenta información relacionada con las interfaces tangibles de usuario. La sección 3 describe el concepto de gamificación. La sección 4 presenta la metodología propuesta. La sección 5 presenta los prototipos desarrollados. La sección 6 presenta los resultados obtenidos y finalmente, la sección 7 presenta las conclusiones y trabajos futuros.

2. ANTECEDENTES

Debido a la inexistencia de una metodología formal para el diseño de sistemas interactivos utilizando interfaces tangibles y técnicas de gamificación y mucho menos relacionadas con niños con problemas visuales, donde se analicen aspectos del usuario en un contexto como la educación. Realizamos una investigación de diferentes metodologías para el diseño de sistemas interactivos con distintos enfoques (Centrados en el usuario, Juegos educativos, Juegos serios, software colaborativo, objetos de aprendizaje) con el propósito de identificar aquellas características que podrían ayudar en nuestra propuesta metodológica.

La primera metodología analizada fue MPIu+a (Saltiveri, 2004). Esta metodología está orientada hacia el diseño de sistemas interactivos centrados en el usuario. El modelo propuesto tiene las siguientes fases: análisis de requisitos, diseño, implementación, lanzamiento, prototipado y evaluación. Uno de los aspectos importantes de la propuesta es integrar la ingeniería de software con los principios de ingeniería de usabilidad y accesibilidad, proporcionando una metodología capaz de guiar a los equipos de desarrollo durante el proceso de implementación de un determinado sistema interactivo.

La segunda metodología analizada lleva por nombre LEGADEE (LEarning, GAmE, DEsign Environment) (Marfisi-Schottman, 2012). Es una herramienta útil para ayudar al diseño de juegos educativos. El objetivo de la metodología es facilitar la colaboración entre los diferentes actores quienes deben intervenir durante el diseño de un juego. La metodología está compuesta de varios bloques que corresponden a la secuencia de las fases que representan el proceso general de la creación del juego.

Nuestra tercera metodología analizada fue DODDLE (Document-Oriented Design and Development of Experimental Learning) (McMahon, 2009). Es una metodología de creación de juegos serios centrada en documentos. Las 4 etapas que involucra para la creación de juegos serios son: Análisis, propuesta de diseño, documentación de diseño y documentación de producción. Para validar el modelo, los autores propusieron a los estudiantes nuevos utilizarla para la concepción de un juego serio. Después las observaciones obtenidas, han notado que el vocabulario que provee es preciso para poder comunicarse entre los diferentes roles. Además, DODDLE ha obtenido un efecto positivo sobre la optimización del tiempo de producción, la calidad educativa y lúdica del juego.

Nuestra cuarta metodología analizada fue MECONESIS (Metodología para el Diseño de Juegos Serios para Niños con Implante Coclear) (Cano Mazuera, 2016). Es una metodología que toma en cuenta habilidades cognitivas del participante como un aspecto que podría servir de ayuda para la adecuación de una interfaz adaptable a las necesidades del niño. Además, es una metodología que sigue la filosofía de Diseño centrado en el usuario, tomando al usuario final desde las diferentes etapas de la metodología.

Nuestra quinta metodología analizada fue MPOBA (Modelo de proceso para el desarrollo de objetos de aprendizaje) (Massa et al., 2011). En esta metodología se integraron técnicas y métodos propios del Diseño Centrado en el usuario con la naturaleza y metas educativas de estos objetos digitales. Los resultados obtenidos de la aplicación de este modelo determinaron cambios sustanciales que fueron valorados positivamente por los estudiantes. Esta metodología tiene por objetivo mejorar la calidad de los objetos de aprendizaje, enfatizando la evaluación de los atributos de usabilidad.

Por último, se analizó la metodología MICEE (Metodología para software colaborativo educativo para la inclusión de personas con discapacidad considerando la UX) (Serna et al., 2019). Es una metodología que consta de cuatro fases de desarrollo (Conocer, Formar, Prototipar, Medir). En la fase conocer se especifican necesidades del sistema por parte de los usuarios, para esto se identifica el contexto físico, ambiental, preferencial, emocional y cognitivo de cada tipo de usuario. La etapa Formar busca comprender los requisitos del sistema para identificar los que se implementaran en el sistema para esto se genera un prototipo que contendrá las características de contenido educativo, colaboración y UX. La fase Construir se desarrolla por medio de código el sistema con las características de contenido educativo, inclusión, colaboración y UX. En la fase medir se verifica y valida el sistema. Por último, la característica explorar corresponde a la etapa de mantenimiento del ciclo de vida llamado cascada.

La mayoría de las metodologías analizadas (MPlu+a, LEGADEE, DODDLE, MPOBA) muestran que han sido aplicadas en diferentes contextos de uso educativo con usuarios promedios. Lo que indica que no han involucrado desde la etapa inicial a personas con algún tipo de discapacidad a excepción de las metodologías MICEE y MECONESIS. Otro punto importante es que no contemplan el uso de interfaces tangibles apoyadas por técnicas de gamificación.

3. METODOLOGÍA METUIGA

METUIGA (Metodología para el diseño de sistemas basados en interfaces tangibles de usuario y técnicas de gamificación) pretende ser una metodología que siga la filosofía de diseño centrado en el usuario y su objetivo principal sea el diseño de sistemas interactivos para niños con problemas visuales.

En el diseño de sus productos se deben tomar en cuenta dos objetivos complementarios, un objetivo táctil con la finalidad de ofrecer herramientas que puedan ser manipulables para el niño y otorgar retroalimentación conforme va progresando a través de audio, por último se deberán implementar escenarios de progresión que utilicen técnicas de gamificación, con el interés de ofrecer un entorno favorable de aprendizaje, donde se tenga en cuenta diferentes aspectos, como: retos, puntuación, recompensas, reglas, entre otros.

Para entender cómo es que se obtuvo la metodología METUIGA, vamos a definir primeramente el significado de método y metodología. Según (Pérez & Gardey, 2008) definen al método como un medio utilizado para llegar a un fin; por otro lado, (Hurtado & Toro, 2007) define metodología como el estudio de los modos o maneras de llevar a cabo algo, es decir, el estudio de los métodos.

La metodología METUIGA es el resultado de una serie de análisis e investigaciones realizadas con el objetivo de generar una metodología enfocada en el desarrollo de sistemas interactivos para usuarios ciegos siguiendo los principios de diseño centrado en el usuario y el ciclo de vida en cascada iterativo, además contará con herramientas para utilizar técnicas de gamificación e interfaces tangibles en el proyecto.

Algunos de los aspectos importantes que se consideraron al momento de diseñar la metodología fueron:

1. Definir una herramienta para lograr la interacción tangible la cual forme parte de la metodología para facilitar a los futuros desarrolladores la búsqueda de alguna herramienta para este propósito.
2. Definir un proceso de gamificación capaz de producir sistemas que enganchen a los usuarios objetivos.
3. Definir evaluaciones que se realicen al finalizar cada etapa de la metodología para medir la usabilidad del sistema y el cumplimiento de los objetivos planteados.
4. METUIGA está organizada en etapas que determinan la etapa de desarrollo en la cual nos encontramos. Ver figura 1

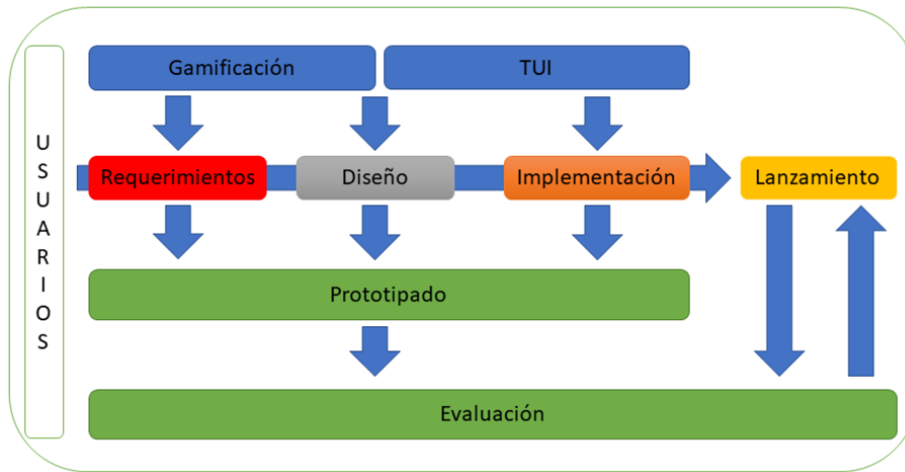


Fig. 1. Metodología para el diseño de sistemas basados en interfaces de tangibles de usuario y técnicas de gamificación para personas con problemas visuales (METUIGA). Fuente: Creación propia.

A continuación, se explica en que consiste cada etapa y las actividades que se implementan:

1. Usuarios

Un proceso de Diseño Centrado en el Usuario debe dejar claro de que es así sólo con mirar el esquema la primera vez. Esto es lo que queda reflejado al disponer a los usuarios en la parte izquierda abarcando el resto de las etapas de todo el proceso, en este caso, al ser el usuario alguien con problemas visuales es de gran importancia que tenga contemplado desde etapas tempranas en el proceso de desarrollo.

2. Gamificación

La etapa de gamificación trabaja en paralelo con las etapas de requerimientos y diseño con el objetivo de guiar a los equipos de desarrollo en la correcta selección de las mecánicas y dinámicas a implementar en base a las necesidades presentadas por el usuario en el proyecto.

3. Interfaces tangibles de usuario

La etapa de interfaces tangibles de usuario provee herramientas necesarias para que el equipo de desarrollo sea capaz de implementar este tipo de tecnología. Facilitando un esquema de construcción para la interfaz tangible de usuario y aconsejando herramientas de software para facilitar la detección de objetos y desarrollo del sistema interactivo.

4. Prototipado

Esta etapa se involucra desde la etapa inicial de metodología, dado que desde el comienzo del desarrollo de un sistema se necesita probar partes el mismo con multitud de objetivos para: Verificar funcionalidades, averiguar aspectos relacionados con la interfaz del sistema, validar la navegación, probar nuevas posibilidades de técnicas, entre otros.

5. Evaluación

Esta etapa se involucra desde la etapa inicial de la metodología, su objetivo consiste en probar algo. Tanto para saber si funciona correctamente o no, si cumple con las expectativas o no, o simplemente para conocer cómo funciona una determinada herramienta. La evaluación es un punto clave para la obtención de sistemas interactivos usables y accesibles. En esta etapa se aplican técnicas necesarias para recibir la retroalimentación por parte de los usuarios. También hace relación a las métricas de usabilidad y métodos de evaluación.

6. Requerimientos

La comunicación con los usuarios es un aspecto prioritario para las empresas que desarrollan sistemas de software, además las necesidades y las experiencias de los usuarios pueden cambiar. Durante esta etapa es importante dejar en claro el alcance y clasificación de gamificación que tendrá el sistema en base a las necesidades del usuario.

Tabla 1. Etapa Requerimientos metodología METUIGA en modo tabla basado en las mejores prácticas (Saltiveri, 2004); (Serna et al., 2019); (Agut, 2000);(Sommerville, 1997); (Pressman, 2005);

Etapa	Actividad
1. Variables de trabajo	Entrevistar a los usuarios finales de la aplicación para conocer las necesidades del sistema
2. Requerimientos	Identificar los requerimientos y restricciones principales del sistema a desarrollar
3. Características	Identificar las características de gamificación del proyecto
4. Casos de uso y actores involucrados	Generación del caso de uso general y actores involucrados en el sistema.
5. Requisitos de usuario	Identificar los requerimientos funcionales y no funcionales Identificar los requisitos de las interfaces de usuario

7. Diseño

En esta etapa se diseña la mejor solución posible, considerando que el problema fue claramente definido en la etapa de requerimientos. Sus etapas cubren diferentes funcionalidades, el diseño de la actividad, diseño de la información, y diseño de la interfaz tangible, así como las principales actividades que conforman el proceso global de los sistemas interactivos, interfaces tangibles de usuario y técnicas de gamificación.

Tabla 2. Etapa Diseño metodología METUIGA en modo tabla basado en las mejores prácticas (Zichermann & Cunningham, 2011); (Lozano et al., 2018);(Chorney, 2012);(Serna et al., 2019);(Saltiveri, 2004);

Etapa	Actividad
1. Desafíos y misiones	Identificar los desafíos y misiones a implementar en el sistema gamificado
2. Clasificación de misiones	Clasificar las misiones comunes, individuales y colectivas
3. Estructura de gamificación	Diseñar la Estructura de gamificación del sistema
4. Avatar del sistema	Diseñar la interfaz de un avatar para el sistema
5. Diagramas de clase	Diseñar los diagramas de clase del sistema
6. Diagramas de flujo	Diseñar los diagramas dinámicos del flujo del sistema
7. Diseños estáticos de interfaces	Construir los diseños estáticos de las interfaces involucrando a los usuarios finales y considerando las mecánicas y dinámicas a implementar
8. Diseños dinámicos de interfaces	Construir los diseños dinámicos de las interfaces involucrando a los usuarios finales
9. Construcción de la base de datos	Realizar el modelado y construcción del diseño de base de datos
10. Diseño de seguridad de base de datos	Realizar el diseño de seguridad de la base de datos
11. Clasificación de objetos tangibles	Diseñar y clasificar los objetos tangibles considerando la información de las interfaces diseñadas
12. Construcción de interfaz tangible	Diseñar y construir la interfaz tangible en base a las recomendaciones de la metodología

8. Implementación

También conocida como la etapa de codificación, ya que es donde se debe escribir el código software necesario que hará posible que el sistema finalmente implementado cumpla con las especificaciones establecidas en la etapa de requerimientos y responda al diseño del sistema.

Tabla 3. Etapa Implementación metodología METUIGA en modo tabla basada en las mejores prácticas (Serna et al., 2019); (Saltiveri, 2004); (Grau & Segura, 2008);

Etapa	Actividad
1. Estándar de codificación	Identificar un estándar de codificación para la programación del sistema
2. Aplicación del estándar	Aplicar el estándar de codificación seleccionado
3. Diagrama de organización	Diseñar un diagrama de organización de los componentes y módulos en el sistema
4. Diagrama de estructura	Diseñar un diagrama de estructura de directorios y archivos finales del sistema
5. Contenido de carpetas	Construir una tabla de Información de contenido de carpetas del sistema

9. Lanzamiento

En esta etapa se comprueba que se ha conseguido la aceptabilidad del sistema, mediante una correcta combinación de aceptabilidad social y práctica. En esta etapa es importante tener una retroalimentación del usuario a través de pruebas. Una vez concluida la etapa de evaluación, el software podrá ser lanzado al usuario.

Tabla 4. Etapa Lanzamiento metodología METUIGA en modo tabla basada en las mejores prácticas (Brooke, 1996); (Pino et al., 2006); (Nielsen, 1993);

Etapa	Actividad
1. Evaluar usabilidad según los usuarios	Identificar la usabilidad del sistema según los usuarios
2. Evaluar objetivos del sistema	Identificar la evaluación de los objetivos por parte de expertos en el área
3. Evaluar usabilidad heurística	Identificar la usabilidad de forma heurística del sistema según los usuarios

4. APLICACIÓN DE LA METODOLOGIA METUIGA

Se ha desarrollado una aplicación que tienen como objetivo apoyar en el aprendizaje de niños ciegos en el tema de porciones matemáticas. Se eligió este tema debido a que para un alumno de Primaria con deficiencias visuales, el concepto de fracción es algo bastante complejo ya que no se puede ayudar de su representación gráfica visual (Núñez, 2017).

La aplicación está compuesta de tres componentes: un sistema de seguimiento tangible, Objetos tangibles, y una aplicación. Los detalles de estos componentes son los siguientes:

4.1 SISTEMA DE SEGUIMIENTO TANGIBLE

Para detectar y seguir los objetos nosotros utilizamos ReactIVision (Reactivision, 2020) el cual es un marco de visión de computadora multiplataforma de código abierto para el seguimiento rápido y robusto de marcadores fiduciales adheridos a objetos físicos, así como para el seguimiento de dedos multitáctil (Ver Fig. 2).

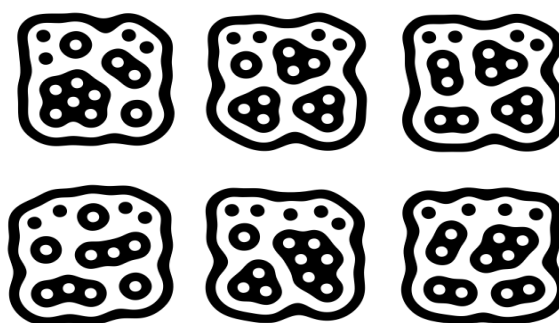


Fig. 2. Marcadores Fiducia Fuente: (Reactivision, 2020).

Para la configuración del hardware para el sistema de seguimiento adoptamos la configuración que nos sugirió la METUIGA (Fig. 4) que consta de un acrílico de 40 cm x 30 cm, una lámpara de 5v y para detectar los objetos montamos una cámara de alta definición marca Microsoft. Ver. Fig.3



Fig. 3. Hardware de nuestro sistema tangible. Fuente: Creación propia.

4.2 OBJETOS TANGIBLES

Utilizamos figuras elaboradas de cartón las cuales representan diferentes porciones matemáticas ($1/2$, $1/3$, $1/4$, etc.). Estos objetos son de gran importancia debido a que, para los estudiantes ciegos, los objetos transmiten diferentes porciones matemáticas con los cuales pueden crear imágenes mentales de las mismas. Por último, se asoció un marcador fiducial único en cada cara de cada objeto para identificarlo en nuestro sistema computacional.

4.3 PORCIONES TANGIBLES

La aplicación desarrollada siguiendo la metodología METUIGA lleva por nombre “Porciones tangibles” ha sido desarrollada con el objetivo de permitir al niño aprender sobre fracciones matemáticas. Las cuales están disponibles en la aplicación y son identificadas por medio de su marcador correspondiente.

La interacción con el sistema es bastante simple ya que el niño solo tiene que tomar literalmente el objeto y ponerlo sobre la mesa dependiendo de la actividad, posteriormente recibirá retroalimentación vía audio relacionada con su desempeño.

La actividad principal de la aplicación consta de una serie de preguntas sobre diferentes porciones matemáticas las cuales van aumentando de dificultad conforme va acertando, una vez que el niño identifica la respuesta y pone el objeto que él considera correcto sobre la interfaz tangible el sistema evalúa su respuesta y le comunica su resultado.

Las siguientes preguntas son las que el niño debe responder cuando está utilizando esta actividad.

1. Con que figura representas $1/2$ de un círculo
2. Con que figura representas $1/3$ de un círculo
3. Con que figura representas $1/4$ de un círculo
4. Con que figura representas $2/3$ de un círculo
5. Con que figura representas $2/4$ de un círculo
6. Con que figura representas $3/4$ de un círculo
7. Con que figura representas un entero de un círculo
8. Con que figura representas un entero y un tercio de un círculo

A continuación, se muestran los avances conseguidos hasta el momento en la construcción del software y la interfaz tangible siguiendo la metodología METUIGA.



Fig. 4. Pantalla principal de la aplicación. Fuente: Creación propia.



Fig. 5. Implementación de técnicas de gamificación (colección de medallas) Fuente: Creación propia.

5. EVALUACION PRELIMINAR DEL SISTEMA

Realizamos una prueba preliminar en la aplicación para medir la satisfacción de los usuarios al momento de utilizarlos.

5.1 Participantes y contexto

Para las primeras pruebas de evaluación solo se pudo reclutar a cuatro participantes normo visuales entre 6 y 10 años. La evaluación fue realizada en un entorno informal. Ellos utilizaron una la aplicación en supervisión de sus padres, posteriormente realizaron el cuestionario para conocer sus experiencias en cuanto a la satisfacción obtenida.

5.2 Métricas de usabilidad.

Para la medición de la satisfacción del sistema, se utilizó el instrumento System Usability Scale (SUS) (Brooke, 1996). Este cuestionario se compone de diez declaraciones relacionadas al uso del sistema. En este caso, nosotros adaptamos el cuestionario en relación con los diferentes sistemas y tipos de usuario. Los usuarios tenían que indicar en una escala de 0 a 5 el nivel de acuerdo o de desacuerdo que tenían con las declaraciones presentadas.

Según el instrumento SUS (Brooke, 1996) la obtención de resultados superiores a 68 se consideran por encima del promedio, indicando que tiene un nivel de satisfacción aceptable. En caso de que ese resultado se encuentre por debajo del promedio el sistema no cuenta con una satisfacción deseada.

La tabla 5 muestra los resultados obtenidos utilizando la SUS por participante. El valor final está entre 0 y 100, siendo 100 el más alto grado de satisfacción en el usuario. Con una media de 95 en el SUS, se puede concluir que el nivel de satisfacción fue alto.

Tabla 5. Resultados de satisfacción por participante. Fuente: Creación propia.

Participante	SUS Score
1	97.5
2	100
3	90
4	92.5
Media	95

6. CONCLUSIONES

En este trabajo de investigación se mostró la metodología METUIGA, la cual todavía está en proceso de desarrollo y tiene por objetivo apoyar en la construcción de sistemas interactivos utilizando interfaces tangibles de usuario y técnicas de gamificación.

Se empleó la metodología METUIGA en la construcción de una aplicación funcional, mediante la cual se demuestra que siguiendo la metodología se logró obtener un sistema que cumplan con características tangibles y técnicas de gamificación.

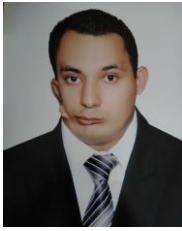
Como trabajo futuro, se pretende seguir con la experimentación de la metodología propuesta creando nuevas aplicaciones orientadas a personas con problemas visuales, así como extender las funcionalidades de la aplicación presentada con el objetivo de agregar más actividades que ayuden a mejorar las habilidades de porciones matemáticas en usuarios ciegos o con baja visión para mejorar su conocimiento sobre este tema el cual les genera mucha dificultad comprender.

Después de revisar los resultados de la prueba de usabilidad, el siguiente paso es realizar la misma evaluación con usuarios con problemas visuales del estado de Aguascalientes, México para comparar sus resultados con personas normo visuales y realizar los cambios correspondientes al modelo y la aplicación aquí presentada.

REFERENCIAS

- Brooke, J. (1996). SUS-A quick and dirty usability scale. In P.W.Jordan, B. Thomas, B.A. Weerdmeester, and I.L. McClelland (Eds.), *Usability Evaluation in Industry* (pp. 189-194). London, UK: Taylor and Francis
- Cano Mazuera, S. P. (2016). *Propuesta Metodológica para el Diseño de Juegos Serios para Niños con Implante Coclear* (Tesis Doctoral). Universidad del Cauca, Colombia.
- Chorney, A. I. (2012). Taking The Game Out Of Gamification. *Dalhousie Journal of Interdisciplinary Management*. <https://doi.org/10.5931/djim.v8i1.242>
- Ferré, X., & Segura, M. (2008). *Desarrollo orientado a objetos con UML*. Facultad de Informática, Universidad Politécnica de Madrid. Recuperado de <https://www.uv.mx/personal/maymendez/files/2011/05/umltotal.pdf>
- Hurtado, I., & Toro, J. (2007). Paradigmas y métodos de investigación en tiempos de cambio. Caracas: CEC.
- Saltiveri, T. G. (2004). *MPIu+a. Una metodología que integra la Ingeniería del Software, la Interacción Persona-Ordenador y la Accesibilidad en el contexto de equipos de desarrollo multidisciplinares* (Tesis Doctoral). Universidad de Lleida, España.
- Lozano, M. D., Penichet, V. M. R., Leporini, B., & Fernando, A. (2018). *Tangible User Interfaces to Ease the Learning Process of Visually-Impaired Children*. <https://doi.org/10.14236/ewic/HCI2018.87>
- Marfisi-Schottman, I. (2012). *Méthodologie, modèles et outils pour la conception de Learning Games* (Doctoral dissertation). INSA de Lyon, Français.
- Massa, S., De Giusti, A., & Pesado, P. (2011). MPOBA: un Modelo de Proceso para el desarrollo de Objetos de Aprendizaje. *XVII Congreso Argentino de Ciencias de La Computación*.
- McMahon, M. (2009). The DODDEL model: A flexible document-oriented model for the design of serious games. In *Games-Based Learning Advancements for Multi-Sensory Human Computer Interfaces: Techniques and Effective Practices*. <https://doi.org/10.4018/978-1-60566-360-9.ch007>
- Monferrer, R. (2000). Especificación de Requisitos Software según el estándar de IEEE 830. Universidad Jaime I. Departamento de Informática.
- Nielsen, J. (1993). Usability Heuristics. In *Usability Engineering*. <https://doi.org/10.1016/b978-0-08-052029-2.50008-5>
- Núñez, M. (2017). *Matemáticas para alumnos con problemas visuales* (Tesis Maestría). Universidad de la rioja, España.
- Pérez, J., & Gardey, A. (2008). *Definición de método*. <https://definicion.de/metodo/>
- Pino, F. J., Garcia, F., Ruiz, F., & Piattini, M. (2006). Adaptación de las Normas ISO/IEC 12207:2002 e ISO/IEC 15504:2003 para la Evaluación de la Madurez de Procesos Software en Países en Desarrollo. *IEEE Latin America Transactions*. <https://doi.org/10.1109/TLA.2006.1642455>
- Pressman, R. (2005). Ingeniería del Software - Un enfoque práctico - Sexta Edición - Ed. México: McGrawHill. In *Journal of Chemical Information and Modeling*. <https://doi.org/10.1017/CBO9781107415324.004>
- Reactivision. (2020). <http://reactivision.sourceforge.net/>
- Serna, V. V., Rodriguez, F. J. A., Arteaga, J. M., Gallegos, J. C. P., & Robles, T. A. (2019). MICEE Methodology. *Proceedings - 2019 International Conference on Inclusive Technologies and Education, CONTIE 2019*. <https://doi.org/10.1109/CONTIE49246.2019.00024>
- Sommerville, I. (2011). *Software engineering 9th Edition*. ISBN-10, 137035152, 18.
- Zichermann, G., & Cunningham, C. (2011). *Gamification by design: Implementing game mechanics in web and mobile apps*. O'Reilly Media, Inc.

NOTAS BIOGRÁFICAS



Luis Roberto Ramos Aguiar Licenciado en Sistemas computacionales por la Universidad Autónoma de Nayarit y actualmente, estudiante de Maestría en Ciencias de la Computación en la Universidad Autónoma de Aguascalientes. Su área de investigación es Ingeniería de Software.



Francisco Javier Álvarez Rodríguez, Profesor de Ingeniería de Software adscrito al Departamento de Ciencias de la Computación, Universidad Autónoma de Aguascalientes (U.A.A.). Doctor en Metodología de la Enseñanza por el IMEP (México). Doctor en Ingeniería por la UNAM (México). Ha sido Decano del Centro de Ciencias Básicas en la U.A.A., así como Jefe de Departamento de Sistemas Electrónicos. Miembro de núcleos académicos de diversos posgrados de la U.A.A. Doctorado en Ciencias de la Computación, Doctorado Interinstitucional en Ciencias, Maestría en Ciencias con opción a Matemática y Computación. Autor de libros y artículos sobre la línea Objetos de Aprendizaje y Procesos de Desarrollo de Software. Actualmente es presidente del Consejo Nacional de Acreditación de programas de Informática y Computación, A.C.



Esta obra está bajo una [licencia de Creative Commons Reconocimiento-NoComercial-CompartirIgual 2.5 México](https://creativecommons.org/licenses/by-nc-sa/2.5/mx/).

HACIA UN SISTEMA DE SOFTWARE BASADO EN IHC PARA EL APOYO DE NIÑOS CON CAPACIDADES AUDITIVAS DIFERENTES

Gerardo Cruz González ¹
gercruz@mixteco.utm.mx

Carlos Alberto Fernández y Fernández ¹
caff@mixteco.utm.mx

Jorge Aguilar Cisneros ²
jorge.aguilar@upaep.mx

¹ Instituto de Computación, Universidad Tecnológica de la Mixteca, Huajuapán de León, Oax. México

² Departamento de Ingeniería, Universidad Popular Autónoma del Estado de Puebla, Puebla, México

RESUMEN

Este artículo presenta una propuesta para que las personas con discapacidad auditiva, especialmente los niños, interactúen con un sistema de software que les permita practicar actividades primordiales de comunicación como: decir nombres personales, indicar estados de ánimo y manifestar malestares físicos por medio de la Lengua de Señas Mexicana. La propuesta consiste en desarrollar una interfaz interactiva de computadora utilizando la metodología de diseño centrada en el usuario, aplicando heurísticas de usabilidad para usuarios con discapacidad auditiva. El objetivo del proyecto es que los niños aprendan a expresarse en su lengua materna; que se integren con mayor facilidad al núcleo social al que pertenecen y que sean menos dependientes de los oyentes con los que conviven. Para entender al usuario se realizará una investigación de campo en la escuela de educación especial de la localidad y la medición de la efectividad, eficiencia y satisfacción del usuario se efectuará en un laboratorio de usabilidad. En el proyecto colaborarán especialistas en problemas de lenguaje, maestros de lengua de señas y expertos en interacción hombre-computadora.

PALABRAS CLAVE

IHC, Inclusión, Interfaces, LSM, Sordos, Usabilidad

TOWARDS AN HCI-BASED SOFTWARE SYSTEM FOR THE SUPPORT OF CHILDREN WITH DIFFERENT HEARING CAPABILITIES

ABSTRACT

This article presents a proposal for hearing impaired people, especially children, to interact with a software system that allows them to practice basic communication activities such as: saying personal names, indicating moods and manifesting physical discomfort through the Mexican Sign Language. The objective is to develop an interactive computer interface using the user-centered design methodology, applying usability heuristics for hearing impaired users. The aim of the project is for children to learn to express themselves in their mother tongue; that they integrate more easily into the social nucleus to which they belong and that they are less dependent on the listeners with which they live. To understand the user, field research will be carried out at the local special education school and the measurement of the effectiveness, efficiency and user satisfaction will be carried out in a usability laboratory. Specialists in language problems, sign language teachers and experts in human-computer interaction will collaborate in the project.

KEYWORDS

HCI, Deaf, Inclusion, Interfaces, LSM, Usability

1. INTRODUCCIÓN

La Organización Mundial de la Salud (OMS) (World Health Organization (WHO), 2020b), estima que al menos el 5% de la población mundial, que equivale a 466 millones de personas, tiene pérdida auditiva profunda, hecho que las convierte en discapacitadas auditivas. De acuerdo con la OMS, 432 millones son adultos y 34 millones son niños. La organización también estima que para 2050, el 10% de la población mundial tendrá pérdida auditiva discapacitante. En México, el Censo de Población y Vivienda del año 2010 (Instituto Nacional de Estadística y Geografía (INEGI), 2013), contabilizó 5,739,270 personas con algún tipo de discapacidad, de las cuales 694,451 (12.1%) tienen algún nivel de sordera. El censo determinó que el 21% de las personas con sordera de entre tres y veintinueve años va a la escuela, pero que solamente el 48% alcanza el cuarto año de primaria. Esto provoca que las personas con discapacidad auditiva (PCDA) formen un grupo social marginado en el ámbito educativo, social y económico. Las estadísticas muestran que el estado de Oaxaca tiene 81.5 sordos por cada diez mil habitantes, lo que ubica al Estado en el tercer lugar a nivel nacional (INEGI, 2013).

Por otro lado, la integración a la sociedad no debe ser un derecho exclusivo para las personas sin discapacidades. Debe ser un derecho para todos. En el caso de la información, ésta debe estar disponible para todo aquel ciudadano que la necesite, en el formato y por el medio que las personas tengan a su disposición. En el caso de las personas sordas o con deficiencia auditiva, ese medio es la Lengua de Señas Mexicana (LSM), la cual se puede expresar a través de medios impresos como libros y revistas, por medios electrónicos como la televisión o el internet, o por contacto directo con personas signantes. Así mismo, los humanos somos seres sociales y debemos tener la capacidad de establecer comunicación con los diferentes entes que conforman la sociedad a la que pertenecemos (familia, vecinos, escuela, seguridad, gobierno). Esta capacidad de comunicación es indispensable para desarrollar relaciones personales, pero también para ofrecer y recibir auxilio en casos de emergencia que pongan en peligro la integridad física de las personas y de sus bienes.

A raíz del COVID-19 (World Health Organization (WHO), 2020a), este derecho ha sido reconocido por el Poder Judicial de la Ciudad de México (D'Artiguez, 2020), al conceder un amparo a la comunidad de sordos de la ciudad para recibir información en LSM por parte de las autoridades. El amparo obliga al presidente de la República, a los titulares de Salud, al Consejo de Salubridad General, al Jefe de Gobierno de la Ciudad de México, así como al titular de la Dirección General de análisis de medios y contenidos visuales del Instituto Federal de Telecomunicaciones (IFETEL) a que de manera inmediata utilicen la LSM en cada comunicación oficial que emitan. Este amparo protege el derecho a recibir información en el formato y por el canal adecuado, pero no es muy útil si el ciudadano con discapacidad auditiva no entiende la LSM.

Aunque en principio pareciera que solamente la comunidad sorda es la que debe aprender LSM, un análisis más a fondo pone de manifiesto que esta lengua también debe ser aprendida por las personas oyentes que conviven o trabajan con miembros de la comunidad sorda, como lo pueden ser familiares, maestros, empleadores, compañeros de trabajo, etc. Actualmente y en aras de la inclusión, existen diversas instituciones públicas y privadas (Yañez, 2019) que enseñan LSM, sin embargo, en la comunidad en donde se lleva a cabo está investigación, la única institución que enseña lengua de signos es el Centro de Atención Múltiple No. 4 (CAM-4).

Por lo general y hasta donde se sabe, los profesores se concentran en escuelas de educación especial que a su vez se instalan en ciudades de medianas a grandes. Esto implica que muchas personas que viven alejadas de las escuelas de educación especial queden al margen de este beneficio. Una manera de colaborar en la disminución de esta marginación es mediante el desarrollo de un sistema de software que ayude a los interesados a practicar la LSM sin la intervención de profesores.

Buscando aplicaciones de software desarrolladas con este propósito, se encontraron cuatro (Tabla 1), en la tienda del sistema Android. Éstas están diseñadas para que el usuario practique vocabulario y dejan de lado otros aspectos como la gramática y la sintaxis de la LSM.

Tabla 1. Aplicaciones para aprender LSM

No.	Interfaces	Dactilológico	Campos semánticos	Frases	Dimensionalidad
1	Dilo en señas (García Gutiérrez & Monsiváis González, 2015)	No	Sí	No	No
2	Aprende señas: LSM (Moreno Robles, 2018)	No	sí	No	No
3	Dimelo Pro (Franco, 2013)	Sí	Sí	No	No
4	Diccionario LSM (Chnt co., 2020)	No	Sí	No	No

Como se observa en la Tabla 1, todas las aplicaciones enseñan vocabulario (campos semánticos), una enseña la seña para cada letra del alfabeto español, ninguna enseña frases y ninguna enseña a diferenciar el múltiple significado que una palabra puede tener (dimensionalidad de la palabra).

Por esta razón se propone el desarrollo de un sistema de software que le permita al usuario la práctica de la LSM para decir pronombres personales, estados de ánimo y a manifestar malestares físicos. Además, el sistema permitirá ejercitarse en la ejecución de sustantivos que estarán divididos en campos semánticos.

El desarrollo del sistema se realizará de acuerdo con la metodología de diseño centrado en el usuario (UCD, por sus siglas en inglés) documentado en (International Organization for Standardization (ISO), 1999b) y en (ISO, 2010a). Esta metodología es un medio para garantizar que la interacción entre el hombre y la computadora (IHC) sea satisfactoria, es decir, que el sistema resuelva problemas específicos de usuarios específicos en un contexto específico de uso. La aplicación de UCD se explica en la sección 8.

2. REVISIÓN DE LITERATURA Y OBRAS RELACIONADAS

Existen diversos trabajos académicos que se han abocado al estudio de la problemática que rodea a las personas con discapacidad auditiva. En esta sección se presentan aquellos que se consideran relevantes para este proyecto.

El trabajo de investigación (Cruz-González et al., 2018), presenta un estudio contextual de niños con discapacidad auditiva (NCDA) que asisten al CAM-4. El estudio muestra las características de los niños, de sus maestras y del entorno en el que llevan a cabo su proceso de enseñanza-aprendizaje.

El proyecto (Martins et al., 2018), tiene como objetivo crear un sistema que proporcione acceso remoto a clases. El sistema incluye la interpretación en línea de la lengua de señas portuguesa (LSP). El sistema también lo pueden utilizar estudiantes presenciales sordos que requieran clases en LSP, lo que pone en igualdad de condiciones a los estudiantes con discapacidad auditiva respecto de los estudiantes oyentes. El trabajo documentado en (Teófilo et al., 2018), es un traductor en tiempo real del idioma portugués hablado al portugués escrito. Utiliza tecnología de realidad virtual y de inteligencia artificial para poder hacer efectiva la traducción de lengua oral a lengua escrita. En este sistema el texto se envía al dispositivo móvil del espectador con la finalidad de que los sordos puedan entender lo que se dice en una película u obra de teatro.

La investigación presentada en (Takala y Sume, 2018), muestra la realización de un estudio en Finlandia. En ese país desde la década de 1990, los niños con discapacidad auditiva reciben un implante coclear antes de cumplir los tres años, por lo cual ellos pueden asistir a escuelas regulares sin restricciones. El estudio analiza los resultados de esta inclusión en donde se aprecia que los niños con discapacidad auditiva mejoran su aprendizaje, pero los niños oyentes se retrasaron.

El documento (Meliones y Duta, 2019) presenta una aplicación para el sistema Android que permite que una persona hablante y una con discapacidad auditiva se puedan comunicar. El hablante se coloca un micrófono para capturar la voz, la voz se convierte a texto el cual se transmite al dispositivo móvil del no oyente y este lee el texto. Por otro lado (Jeyalakshmi y Revathi, 2018), propone un sistema que reconoce frases habladas por PCDA que han sido oralizadas. Las frases se convierten en audio fácilmente entendibles por personas oyentes. Este trabajo se justifica porque las PCDA oralizados estructuran y pronuncian las frases de manera diferente a como lo hacen los oyentes, lo que dificulta entender lo que dicen. Por otra parte (Kularia et al., 2020), propone un prototipo para ayudar a los sordos a detectar sonidos o ruidos de alarma que ocurren en el hogar y que no sean voces. Los ruidos con los que trabaja el sistema son los que emiten las alarmas de incendios, de presencia de intrusos, de movimiento y de apertura y cierre de puertas. El objetivo de la aplicación es disminuir la dependencia que tienen los discapacitados auditivos de sus familiares oyentes en un entorno doméstico. En esta misma línea está la investigación (Berger y Maly, 2018), que propone una solución que extrae las voces presentes en el sonido ambiental para su conversión a texto, y posterior presentación en el lente Google Glass, de un adulto mayor que está perdiendo la habilidad de escuchar. En esta misma línea de investigación está (Razalli et al., 2018) donde se muestra que el texto con imágenes ayuda a las personas a entender la idea fundamental de un documento escrito, aun si el lector no entienda o conozca el significado de cada una de las palabras escritas.

El trabajo de investigación documentado en (Syukri y Humaera, 2019), propone la enseñanza del inglés utilizando tarjetas flash, libros plegables y pósteres. La propuesta está enfocada en estudiantes adolescentes de Indonesia que padecen sordera profunda y que están aprendiendo inglés como segundo idioma. El proyecto solamente considera la enseñanza de vocabulario, pero no va más allá. En este mismo sentido se realizó el proyecto de (Jabar y Ahmad, 2018). Este tiene el propósito de enseñar a los niños malasio de entre ocho y diez años, a leer el idioma bahasa melayu. Para conseguirlo utilizan elementos multimedia, es decir, textos, gráficos, ilustraciones y videos, integrados mediante herramientas de autoría e-learning.

El trabajo presentado en (Nathan et al., 2018), analiza la importancia de que las aplicaciones para sordos, que se ejecutan en teléfonos celulares, cumplan con cinco criterios de usabilidad: memorabilidad, accesibilidad, satisfacción, eficacia y eficiencia, ya que desde su punto de vista no lo hacen y que son indispensables para que sean útiles a los usuarios.

La investigación (Yeratziotis y Zaphiris, 2018) presenta doce heurísticas para ayudar a los expertos de IHC y desarrolladores web en la evaluación y el diseño de sitios web para usuarios sordos. Esto mediante la medición de problemas de accesibilidad y usabilidad que pueden influir en la experiencia de usuario en la Web. Otro trabajo que también propone heurísticas de desarrollo de aplicaciones para sordos es (Cruz-González et al., 2019) el cuál utilizó UCD para desarrollar interfaz para niños sordos. Durante el desarrollo, la aplicación se sometió a pruebas de usabilidad para garantizar la satisfacción de los usuarios. También (Risald et al., 2018) muestra una aplicación desarrollada conforme a UCD para asistir a los sordos en casos de emergencia médica. La aplicación solamente utiliza íconos para representar el tipo de emergencia (accidente, embarazo, ataque cardíaco y golpe o caída). El usuario selecciona el icono adecuado y la aplicación determina el hospital más cercano. De esa manera el usuario recibe la atención médica necesaria.

Finalmente (Murtaza et al., 2019) presenta el desarrollo de un traductor de lengua de señas a texto. Tiene como objetivo desarrollar un sistema IHC que convierta el lenguaje de señas en texto para personas con discapacidad auditiva. El sistema aparte de considerar las señas también toma en cuenta los gestos y el lenguaje corporal para decodificar el mensaje emitido por la persona con discapacidad auditiva.

3. CONTEXTO DE LOS NIÑOS CON DEFICIENCIA AUDITIVA

En la zona de la mixteca se tienen municipios y agencias municipales con un alto porcentaje de personas con discapacidad auditiva, al grado que los presidentes municipales, los padres de familia y los maestros regulares, han promovido que maestros de educación especial, acudan a las primarias, una o dos veces por semana, a enseñar LSM a los NCDA que lo requieran. Es decir, como son muchos los niños, es más fácil traer a los profesores que llevar a los niños a una escuela de educación especial. Por otro lado, en entrevista con profesoras del CAM-4, ellas explicaron que muchos de los niños que requieren educación especial, por limitaciones económicas, en lugar de asistir a la escuela cinco días a la semana, solamente acuden de una a tres veces a clase, lo que limita la oportunidad de que aprendan la LSM. En la Figura 1, se observa a una maestra de LSM con su grupo de estudiantes.



Figura 1. Niños del CAM-4 en clase de LSM.

La caracterización de los niños del CAM-4 es la siguiente (Cruz-González et al., 2018): proceden de familias con recursos económicos limitados, no hablan, no escriben español, no están oralizados, asisten a escuelas regulares y no conocen la LSM. Para comunicarse utilizan señas informales que solamente son entendibles en el entorno familiar, pero que no son útiles en otros lugares como la escuela. Todo esto origina que sean niños aislados, que experimenten frustración, marginación y presenten un nivel de conocimientos escolares considerablemente menor que sus compañeros oyentes.

Por otro lado, los NCDA con acceso a internet han descubierto que una manera fácil de comunicarse entre ellos es por medio de la grabación de videos que luego comparten con sus contactos de redes sociales. Esto se debe a que su lenguaje materno es la lengua de señas que requiere expresión corporal, pero no necesita del habla o de la escritura para llevarse a cabo.

4. CONTEXTO DE LA SOLUCIÓN PROPUESTA

Para coadyuvar en la solución de la problemática descrita en la sección anterior, en el año 2019 se liberó un sistema de software con una interfaz interactiva (Cruz-González, 2019) y (Cruz-González et al., 2019), para el tipo de usuario descrito anteriormente. Las características principales de esa interfaz son: dispone de un conjunto de 59 palabras agrupadas en grupos semánticos. El papel de profesor o de instructor lo tiene un avatar animado en 3D que muestra a los niños cómo mover el cuerpo para realizar señas de la LSM. La velocidad de la animación, ángulo de visión y la distancia relativa entre el usuario y el avatar son parámetros que se definen en tiempo de ejecución de acuerdo con las necesidades del usuario.

La ventaja de esta aplicación es que es interactiva y configurable de acuerdo con las necesidades o gustos del usuario, sin embargo, al considerar únicamente sustantivos, no es posible formar frases y practicar los diálogos en LSM. En la Figura 2 y Figura 3, se puede ver al avatar haciendo la seña de fruta, pero con la distancia y ángulo modificados. Por lo anterior, se propone incrementar la funcionalidad del sistema de software con el fin de que integre las señas para decir pronombres personales, estados de ánimo y a manifestar malestares físicos.



Figura 2. Seña de fruta



Figura 3. Fruta, Ángulo y distancia diferente

5. VENTAJAS DE LA SOLUCIÓN PROPUESTA

Las ventajas de la solución propuesta son:

1. La interfaz del sistema de software es configurable de acuerdo con los gustos o necesidades del usuario.
2. No es indispensable la presencia de un maestro de LSM.
3. En el diseño de la interfaz se aplicarán heurísticas de diseño para PCDA y NCDA.
4. El usuario podrá practicar pronombres personales, sentimientos, estados de ánimo y manifestar malestares físicos. Esto le permitirá al usuario decir frases como “*me duele el estómago*”, “*me siento mal*”, “*estoy lastimado*”, “*ella está triste*”, etc.
5. La copia de la aplicación es económica y rápida.
6. El usuario tendrá la oportunidad de practicar el vocabulario incluido en la versión actual.

Se considera que estas características son un diferenciador importante respecto de las aplicaciones que solamente enseñan dactilología y/o vocabulario de palabras aisladas.

6. ARQUITECTURA DE LA SOLUCIÓN PROPUESTA

Internamente, la aplicación se compone de tres módulos (actividades) interrelacionados mostrados en la Figura 4, la cual muestra la navegabilidad propuesta para la aplicación. El primer módulo corresponde a la actividad Elegir Personaje [EP]. Estando en ese módulo, el usuario puede elegir uno de ocho personajes disponibles y continuar para elegir el grupo de palabras o de frases que desea practicar. También el usuario puede salir de la aplicación.

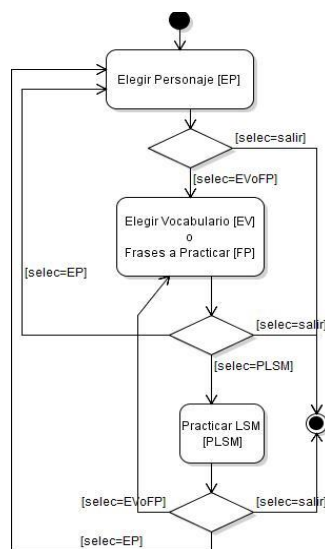


Figura 4. Actividades de la interfaz propuesta

El segundo módulo corresponde a la actividad Elegir vocabulario o frases a practicar [EV o FP]. En él, el usuario podrá seleccionar el grupo semántico o el grupo de frases en el que desea ejercitarse y avanzar al módulo Practicar LSM [PLSM]. También puede retroceder para cambiar de personaje o salir del programa.

El tercer módulo corresponde a la actividad Practicar LSM [PLSM]. Estando en él, el usuario podrá observar cómo es que el personaje ejecuta las señas. Las señas estarán acompañadas de una imagen o de un video que muestre el objeto o la idea de la seña ejecutada. Estando en este módulo, el usuario tendrá la facilidad de cambiar la velocidad de animación, rotar el personaje 45 grados a la izquierda o derecha, y acercar o alejar el personaje para apreciar los movimientos a distancias diferentes. Si el usuario lo desea, podrá regresar al primer o segundo módulo o dar por terminada la ejecución de la interfaz.

7. ETAPAS DEL PROYECTO DE INVESTIGACIÓN

Este proyecto consta de las etapas que se describen a continuación:

1. Estudio contextual: si bien en 2017 se aplicó un estudio contextual a los NCDA que asisten al CAM-4 de la localidad, se considera que es prudente realizar otro: algunos niños ya han concluido su estancia en la escuela, han ingresado nuevos niños y los maestros también han cambiado. Además, es necesario saber si la legislación, las reglas y los métodos de enseñanza permanecen o han sido modificados.
2. Desarrollo y pruebas de usabilidad: Para el desarrollo de la aplicación se utilizará UCD, lo que implica que las propuestas y avances deberán ser validadas por los usuarios. Las validaciones serán por medio de pruebas de usabilidad realizadas por los niños y maestras del CAM-4. Las pruebas de las maestras se considerarán como evaluaciones expertas ya que ellas conocen a los niños, son especialistas en problemas de lenguaje, enseñan LSM y utilizan diferentes aplicaciones de software para impartir sus clases.
3. Captura de movimiento y programación de la animación: Para la captura de movimiento se debe considerar un dispositivo capaz de detectar el movimiento de los brazos, codos, muñeca, palma de la mano y dedos ya que los movimientos combinados de estos elementos son esenciales para la LSM. También es importante que el sistema sensor detecte la expresión de la cara porque los gestos enfatizan o invierten (ironía) el significado de lo que se está diciendo. Con los datos del sensor es posible generar la animación e incorporarla a los personajes que formarán parte de la interfaz.

8. FASES DE DESARROLLO DEL SISTEMA DE SOFTWARE

El desarrollo del sistema estará guiado por la metodología UCD (Figura 5), definida tanto en (International Organization for Standardization, 1999b) como en (International Organization for Standardization, 2010a). UCD es un enfoque para el diseño y desarrollo de sistemas que tiene como objetivo hacer sistemas interactivos más útiles y usables. UCD se centra en las necesidades y requerimientos de los usuarios aplicando factores humanos y ergonómicos, así como principios de usabilidad. De acuerdo con Hassan y Montero (Montero y Santamaría, 2009), el proceso UCD se divide en entender y especificar el contexto de uso; especificar las necesidades o requisitos de los usuarios; producir soluciones de diseño que satisfagan las necesidades de esos usuarios y evaluar los diseños en función de los requerimientos o necesidades de los usuarios.

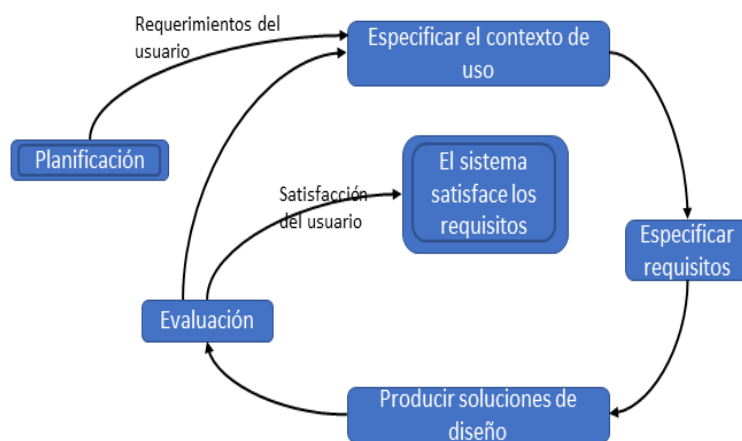


Figura 5. Metodología UCD

Para entender y especificar el contexto de uso se hará un estudio contextual en el CAM-4 de la localidad, ya que a esa escuela asisten NCDA que son atendidos por maestras de LSM que tienen posgrados en problemas de lenguaje. Del análisis de resultados del estudio contextual se definirán las necesidades y los requisitos de los usuarios y se estará en condiciones de proponer y desarrollar una solución que satisfaga las necesidades de los NCDA. El desarrollo del sistema será incremental y sometido por lo menos a tres pruebas de usabilidad con objeto de garantizar que el sistema cumpla con los requisitos de usuario.

9. CONCLUSIONES

Se deben desarrollar sistemas de software con interfaces que apoyen a las PCDA a entender la LSM ya que, aunque la información se difunde traducida a LSM, si la persona con discapacidad auditiva no la entiende la información que se le proporciona, ésta no será de utilidad. Para el desarrollo de este tipo de sistemas es importante conocer y aplicar las heurísticas de diseño enfocadas en las PCDA. Además, con la masificación de los dispositivos móviles es posible desarrollar aplicaciones que les permitan a los alumnos continuar su adquisición de conocimientos sin salir de casa. Se debe promover que la comunidad sorda se pueda comunicar mediante la LSM, ya que esta lengua forma parte de la lengua nacional mexicana y por lo tanto es un medio estandarizado por las autoridades de nuestro país.

REFERENCIAS

- Berger, A., y Maly, F. (2018). Prototype of a Smart Google Glass Solution for Deaf (and Hearing Impaired) People. En M. Younas, I. Awan, G. Ghinea y M. Catalan Cid (Eds.), *Mobile Web and Intelligent Information Systems* (Vol. 10995, pp. 38-47). Springer International Publishing. https://doi.org/10.1007/978-3-319-97163-6_4
- Chnt Co. (2020). *Diccionario LSM. Android*. Chnt Co. Descargado de https://play.google.com/store/apps/details?id=com.lsm.dicc.chnt&hl=es_419
- Cruz-González, G. (2019). *Desarrollo de una interfaz basada en la metodología de diseño centrada en el usuario para la práctica de la lengua de señas mexicana* (Tesis de maestría). Universidad Tecnológica de la Mixteca.
- Cruz-González, G., Fernández-Fernández, C. A., y Trujillo-Romero, F. (2018). Estudio contextual y propuesta de interfaz para la práctica de la lengua de señas mexicana en la mixteca oaxaqueña. *Abstraction & application*, 20, 12-23. 2007-2635.
- Cruz-González, G., Fernández-Fernández, C. A. y Trujillo-Romero, F. (2019). Hacia una propuesta de heurísticas de usabilidad para pruebas de HCI y de UX para niños con discapacidad auditiva: Caso de estudio. ReCIBE, Revista electrónica de Computación, *Informática, Biomédica y Electrónica*, 8(1), 3. <http://recibe.cucei.udg.mx/ojs/index.php/ReCIBE/article/view/127>
- D'Artigues, K. (marzo 17, 2020). Comunidad sorda logra orden para tener datos de COVID-19. *Animal Político*. Recuperado de <https://www.animalpolitico.com/2020/03/comunidad-sorda-gana-amparo-gobierno-covid-19/>
- Franco, D. (2013). *Dímelo Pro for Android*. Android, *APKPure.com*. Descargado de <https://apkpure.com/es/dimelo-pro/com.lionteamsoft.dimelo>
- García Gutiérrez, R. y Monsiváis González, J. G. (2015). *Dilo en señas*. Android. Descargado de https://play.google.com/store/apps/details?id=com.jaguarlabs.lsm&hl=es_419
- Instituto Nacional de Estadística y Geografía (INEGI), (2013). *Las personas con discapacidad en México: Una visión al 2010*. Instituto Nacional de Estadística y Geografía (México). Recuperado de https://www.ipomex.org.mx/recursos/ipo/files_ipo3/2018/44257/4/b202c98e9a2106f4c0f427b64f542c93.pdf
- International Organization for Standardization (ISO). (1999b). *ISO 13407-1999, Human-centred design processes for interactive systems*. Recuperado de <https://www.iso.org/obp/ui/#iso:std:iso:13407:ed-1:v1:en>
- International Organization for Standardization (ISO). (2010a). *ISO 9241-210:2010(en), Ergonomics of human-system interaction—Part 210: Human-centred design for interactive systems*. Recuperado de <https://www.iso.org/obp/ui/#iso:std:iso:9241:-210:ed-1:v1:en>
- Jabar, S. A., y Ahmad, A. C. (2018). The Design of Multimedia Interactive Courseware for Teaching Reading to Hearing Impaired Students. *International Journal of Academic Research in Progressive Education and Development*, 7(4), 223–230.
- Jeyalakshmi, C. y Revathi, A. (2018). Efficient speech recognition system for hearing impaired children in classical Tamil language. *International Journal of Biomedical Engineering and Technology*, 26, 84. <https://doi.org/10.1504/IJBET.2018.089261>
- Kularia, P., Bhutkar, G., Jadhav, S. y Jadhav, D. (2020). Prototype Design of Alert Device for Hearing Impaired Users. 133-138 en: Loizides, F. et al(eds.), *Human Computer Interaction and Emerging Technologies*. Cardiff: Cardiff. <https://doi.org/10.18573/book3.p>
- Martins M., Borges J., Justino E., Rocha T., Barroso J., Reis A. (2018). A Proposal for a Remote Interactive Class System with Sign Language Interpretation. In: Antona M., Stephanidis C. (eds) *Universal Access in Human-Computer Interaction. Virtual, Augmented, and Intelligent Environments*. UAHCI 2018. Lecture Notes en Computer Science, vol 10908. Springer, Cham. https://doi.org/10.1007/978-3-319-92052-8_42
- Meliones, A. y Duta, C. (2019). SeeSpeech: An Android application for the hearing impaired. En *PETRA '19: Proceedings of the 12th ACM International Conference on Pervasive Technologies Related to Assistive Environments*, 509-516. <https://doi.org/10.1145/3316782.3324013>

- Montero, Y. H. y Santamaría, S. O. (2009). *Informe APEI sobre usabilidad. Informes APEI*, 3, 1. ISBN 978-84-692-3782-3. <https://dialnet.unirioja.es/servlet/articulo?codigo=3101458>
- Moreno Robles, M. (2018). *Aprende señas: Lengua de Señas Mexicana*. Android. Descargado de <https://play.google.com/store/apps/details?id=rodolfo.com.systemsolutionanddevelopment.aprendiendolism&hl=es> 419
- Murtaza, Z., Akmal, H., Afzal, W., Gelani, H., Abdin, Z. y Gulzar, M. (2019). Human Computer Interaction Based on Gestural Cues Recognition/Sign Language to Text Conversion. 2019 *International Conference on Engineering and Emerging Technologies (ICEET)*, Lahore, Pakistan, 2019, pp. 1-6, doi: <https://10.1109/CEET1.2019.8711835>.
- Nathan, S., Hussain, A. y Hashim, N. L. (2018). Usability evaluation of DEAF mobile application interface: A systematic review. *Journal of Engineering and Applied Sciences*, 13: 291-297. ISSN: 1816-949x. <http://repo.uum.edu.my/25662/>
- Razalli, A., Thomas, R., Mamat, N. y Yusuf, N. (2018). Using Text with Pictures in Primary School to Improve Reading Comprehension for Hearing Impaired Students. *Journal of ICSAR*, 2, 19-27. jan. 2018. ISSN 25488600. <https://doi.org/10.17977/um005v2i12018p019>
- Risald, R., Suyoto, S. y Santoso, A. (2018). Mobile Application Design Emergency Medical Call for the Deaf using UCD Method Mobile Application Design Emergency Medical Call for the Deaf using UCD Method. *International Journal of Interactive Mobile Technologies (IJIM)*, Vol. 12. N. 3, p. pp 168-177, jul. 2018. ISSN 1865-7923. <https://doi.org/10.3991/ijim.v12i3.8754>
- Syukri, S. y Humaera, I. (2019). Gaining Motivation on English Learning for Special Need Students Using Flashcards, Foldable Books and Posters in EFL Context. Langkawi: *Journal of The Association for Arabic and English*, 5(2), 91-101. <https://doi.org/10.31332/lkw.v5i2.1303>
- Takala, M. y Sume, H. (2018). Hearing-impaired pupils in mainstream education in Finland: Teachers' experiences of inclusion and support. *European Journal of Special Needs Education*, 33(1), 134-147. <https://doi.org/10.1080/08856257.2017.1306965>
- Teófilo, M., Lourenço, A., Postal, J. y Lucena Jr, V. (2018). Exploring Virtual Reality to Enable Deaf or Hard of Hearing Accessibility in Live Theaters: A Case Study. En: Antona M., Stephanidis C. (eds) *Universal Access in Human-Computer Interaction. Virtual, Augmented, and Intelligent Environments*. UAHCI 2018. Lecture Notes in Computer Science, vol 10908. Springer, Cham. pp. 132-148. https://doi.org/10.1007/978-3-319-92052-8_11
- World Health Organization (WHO). (2020a). *Coronavirus. Coronavirus Disease (COVID-19) Pandemic*. Recuperado de <https://www.who.int/emergencies/diseases/novel-coronavirus-2019>
- World Health Organization (WHO). (2020b). *Deafness and hearing loss*. Recuperado de <https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss>
- Yañez, L. (23 de septiembre, 2019). Aprender Lengua de Señas Mexicana, un paso más a la inclusión. *Milenio*. Recuperado de <https://www.milenio.com/aula/lengua-de-senas-mexicana-el-idioma-que-se-debe-aprender>
- Yeratziotis, A. y Zaphiris, P. (2018). A Heuristic Evaluation for Deaf Web User Experience (HE4DWUX). *International Journal of Human-Computer Interaction*, 34(3), 195-217. <https://doi.org/10.1080/10447318.2017.1339940>

NOTAS BIOGRÁFICAS



Gerardo Cruz González. Es egresado de la escuela de Físico - Matemáticas de la Benemérita Universidad Autónoma de Puebla (BUAP). Tiene una maestría en ciencias con especialidad en Ingeniería en Sistemas Computacionales otorgado por la Universidad de las Américas-Puebla y una maestría en Medios Iterativos otorgado por la Universidad Tecnológica de la Mixteca. Ha laborado como profesor en la Facultad de Computación de la BUAP y en la Universidad Tecnológica de la Mixteca (UTM), en donde actualmente tiene el nombramiento de Profesor-Investigador de tiempo completo. En la UTM también ha sido jefe de la carrera de Ingeniería en computación y secretario de la División de Estudios de Posgrado. Su línea de investigación es la Interacción Humano-Computadora en el área de discapacidad auditiva.



Carlos Alberto Fernández y Fernández Egresado de la Facultad de Informática de la Universidad Veracruzana, con una Maestría en Ciencias de la Computación en la Fundación Arturo Rosenblueth. Recibió el grado de Doctor en Ciencias de la Computación en la Universidad de Sheffield, Inglaterra. Se encuentra adscrito al Instituto de Computación de la Universidad Tecnológica de la Mixteca. Ha sido coordinador de la Universidad Virtual y de la Maestría en Computación con especialidad en Sistemas Distribuidos. Trabaja dentro del área de Ingeniería de Software, particularmente en las líneas de modelado visual, métodos de desarrollo y especificación formal de software. Ha sido responsable del Cuerpo Académico de Ingeniería de Software en la UTM y miembro del Verification and Testing Research Group en la Universidad de Sheffield.



Jorge R. Aguilar Cisneros es doctor en ingeniería de software por parte de UPAEP. Obtuvo el grado de Licenciatura en Computación en la Facultad de Ciencias Físico-Matemáticas de la Benemérita Universidad Autónoma de Puebla (BUAP). Posteriormente estudió la Maestría en Ciencias en la Universidad de las Américas Puebla (UDLAP). El grado de Doctor en Ingeniería de Software lo obtuvo en la Universidad Popular Autónoma del Estado de Puebla (UPAEP). El Dr. Aguilar es miembro de AMEXCOMP, IEEE y RedMIS. Actualmente es profesor investigador del Decanato de Ingenierías en UPAEP. Imparte las materias relacionadas con Ingeniería de Software. Su línea de interés es la ingeniería de software aplicada en el sector automotriz.



Esta obra está bajo una [licencia de Creative Commons Reconocimiento-NoComercial-CompartirIgual 2.5 México](https://creativecommons.org/licenses/by-nc-sa/2.5/mx/).

ESENCIALIZACIÓN DE LA PRÁCTICA GESTIÓN DE REQUISITOS DE RUP

ESSENTIALIZATION OF THE RUP REQUIREMENTS MANAGEMENT PRACTICE

Jhonny E. Chapal Vallejo
ihonnychapal@gmail.com

John S. Guerrero Riascos
snydeerjs10@gmail.com

Alexander A. Barón Salazar
abaron_98@udenar.edu.co

Facultad de Ingeniería, Universidad de Nariño, Pasto, Nariño, Colombia.

RESUMEN

La construcción y aplicación de buenas prácticas en la gestión de requisitos de software es importante para obtener productos de alta calidad. En la industria de software se aplican procesos de mejoramiento continuo de prácticas para educación, análisis, especificación y validación de requisitos. Se evidencia que, en el contexto industrial, se hace compleja la aplicación de prácticas que proponen métodos de software por la dificultad para identificar los componentes de la práctica. Es el caso de la práctica Gestión de Requisitos de RUP. La aplicación de esta práctica es compleja porque no se identifican claramente los componentes de la práctica y los mecanismos de aplicación, seguimiento y evaluación. En este artículo se aplica el Modelo para la Definición de Prácticas en Ingeniería de Software a la práctica Gestión de Requisitos de RUP. Este proceso que busca identificar y definir los elementos esenciales para facilitar la aplicación, seguimiento y evaluación de prácticas se denomina esencialización. De esta manera a partir de una práctica existente se obtiene una práctica bien formada y nombrada que facilita su aplicación, seguimiento y evaluación en contextos reales. Una práctica esencializada facilita a los practicantes entender, aplicar y evaluar la práctica ya que define una estructura adecuada con elementos bien definidos. También aporta en el seguimiento a la salud y el progreso del esfuerzo de Ingeniería de Software mediante el uso del núcleo del estándar Essence.

PALABRAS CLAVE

Gestión de Requisitos, Esencialización de prácticas, Núcleo de Essence.

ABSTRACT

In the software requirements management, the construction and application of good practices are important to obtain high quality products. In the software industry, continuous improvement processes of practices are applied for the preparation, analysis, specification and validation of requirements. In the industrial context, it is evident that the application of practices that propose software methods becomes complex due to the difficulty in identifying the components of the practice. This is the case of the RUP Requirements Management practice. The application of this practice is complex because the components of the practice and the mechanisms of application, the monitoring and evaluation are not clearly identified. In this article, the Model for the Definition of Practices in Software Engineering is applied to the RUP Requirements Management practice. This process called essentialization seeks to identify and define the essential elements to facilitate the application, monitoring and evaluation of practices. In this way, from an existing practice, a well-formed and named practice is obtained that facilitates its application, monitoring and evaluation in real contexts. An essentialized practice makes it easier for practitioners to understand, apply and evaluate the practice as it defines an adequate structure with well-defined elements. It also contributes to monitoring the health and progress of the Software Engineering effort by using the kernel of the Essence standard.

KEYWORDS

Essence kernel, Essentialization of practices, Requirements Management.

1. INTRODUCCIÓN

La construcción y aplicación de buenas prácticas en la gestión de requisitos de software es importante para obtener productos de alta calidad (Londoño & Tabares, 2008). Otros autores llegan a una conclusión similar al afirmar que, en la Ingeniería de Requisitos es determinante lograr productos de software correctos, fiables y mantenibles. Por lo tanto, es necesario tener buenas prácticas para separar y especificar correctamente los requisitos, controlar su evolución y soportar los cambios (Tabares, et al., 2007).

En el contexto industrial de la Ingeniería de Software, se hace compleja la aplicación de prácticas debido a que, es difícil identificar y definir los componentes de la práctica en las formas de representación que diferentes autores proponen. En particular, en las propuestas que describen la práctica Gestión de Requisitos de Rational Unified Process (PGR-RUP), no se identifican claramente los componentes y los mecanismos de aplicación, seguimiento y evaluación. Por ejemplo, en trabajos como (Rational Software Company, 1998), (Metzner & Niño, 2016), (Vera, et al., 2019) y (Jacobson, et al., 2000), se describe la práctica de diversas maneras. En estas propuestas es difícil identificar y definir componentes de la práctica como: nombre de la práctica, criterios de entrada, criterios de finalización, actividades, tareas y productos de trabajo.

En (Barón, 2019) se propone un modelo para definir prácticas bien formadas y nombradas. Una práctica es bien nombrada si su nombre incluye: un verbo nominalizado, un adjetivo y un sustantivo. Así mismo, una práctica es bien formada si el conjunto de actividades cumple con reglas de coherencia, consistencia y suficiencia. El proceso para identificar y definir los elementos esenciales de la práctica se denomina esencialización.

En este artículo se presenta el proceso de esencialización de la PGR-RUP aplicando el modelo de Barón propuesto en (Barón, 2019). La PGR-RUP esencializada, facilita a los practicantes entender, aplicar y evaluar la práctica ya que se define una estructura adecuada con los componentes bien definidos. La PGR-RUP esencializada, facilita también, el seguimiento a la salud y el progreso del esfuerzo de Ingeniería de Software mediante el uso del núcleo del estándar Essence.

Este artículo está estructurado de la siguiente manera: en el capítulo II se presenta el contexto de esencialización de PGR-RUP. Más adelante, en el capítulo III, se describe el estudio sobre las formas de representación y descripción de la PGR-RUP. Posteriormente, en el capítulo IV, se presenta la conversión de la PGR-RUP en una práctica bien formada y nombrada. A continuación, en el capítulo V, se presentan las conclusiones y el trabajo futuro. Finalmente, se relacionan las referencias de la investigación.

2. CONTEXTO DE ESENCIALIZACIÓN DE PGR-RUP

A continuación, se presentan los fundamentos teóricos que son necesarios para el desarrollo de esta investigación.

2.1 ESSENCE, EL MARCO DE TRABAJO PARA DEFINIR PRÁCTICAS (OMG, 2018)

La comunidad Software Engineering Method and Theory (SEMAT, por su sigla en inglés), consolida un marco de pensamiento denominado Essence que involucra reestructurar la Ingeniería de Software y mejorar sus métodos.

Essence incluye un núcleo que proporciona un conjunto de elementos que prevalecen en cada esfuerzo de Ingeniería de Software. El núcleo de Essence incluye: (i) las áreas de interés, que permiten organizar los elementos del núcleo en cliente, solución y esfuerzo (ver Figura 1); (ii) los alfas, son las cosas con las que siempre se trabaja, permiten dimensionar el esfuerzo de Ingeniería de Software que el equipo realiza (ver Figura 2); (iii) los espacios de actividad, son las cosas que siempre se hacen, permiten definir el componente dinámico del núcleo (ver Figura 3); (iv) las competencias, son las habilidades necesarias que el equipo debe desarrollar para abordar el esfuerzo de Ingeniería de Software (ver Figura 4).

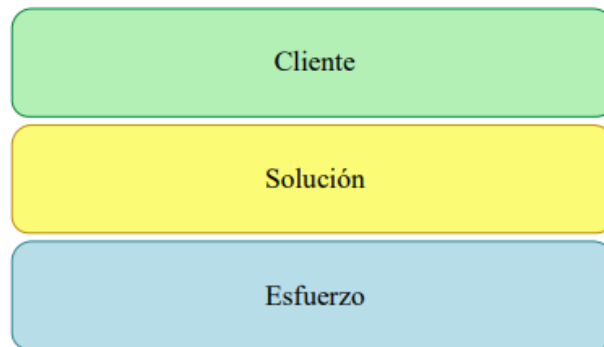


Figura 1. Áreas de interés del núcleo (OMG, 2018)

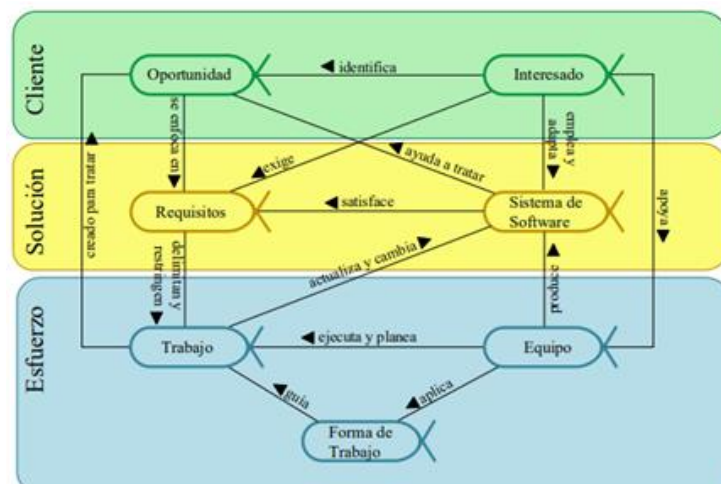


Figura 2. Alfas del núcleo (OMG, 2018)

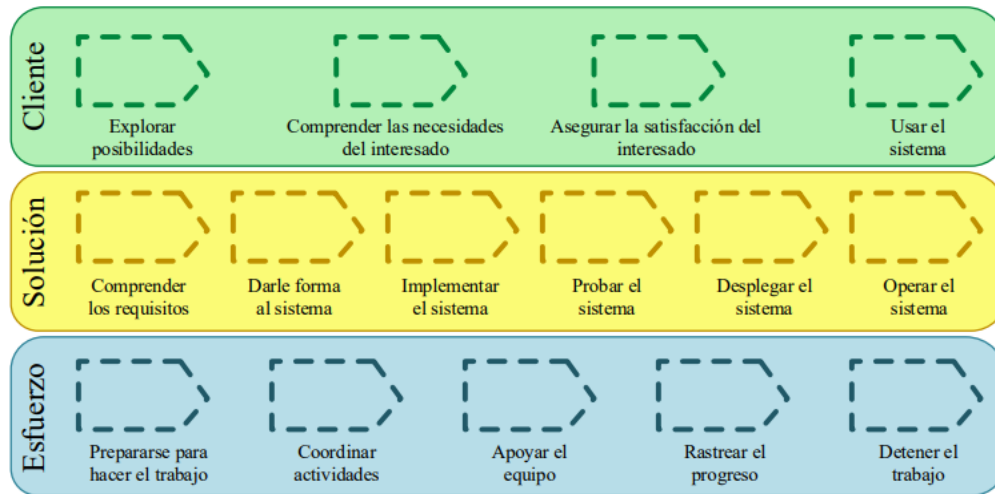


Figura 3. Espacios de actividad del núcleo (OMG, 2018)



Figura 4. Competencias del núcleo (OMG, 2018)

2.2 RUP Y LA GESTIÓN DE REQUISITOS, EL OBJETO DE ESTUDIO

El Proceso Unificado de Rational (RUP, por su sigla en inglés), es un proceso de desarrollo de sistemas que se fundamenta en principios de Ingeniería de Software. En RUP se define un proceso iterativo e incremental, centrado en la arquitectura y orientado por casos de uso. Además, RUP proporciona un enfoque disciplinado para asignar y administrar tareas y responsabilidades en una organización de desarrollo de software (Somerville, 2011). RUP se muestra de manera gráfica en la Figura 5.

2.2.1 Ciclo de vida

El ciclo de vida de RUP está estructurado en dos dimensiones, dinámica y estática: (i) estructura dinámica (horizontal), representa la dimensión del tiempo del proceso. Muestra el proceso expresado en ciclos, fases, iteraciones e hitos; (ii) estructura estática (vertical); describe cómo los elementos del proceso (actividades, disciplinas, artefactos y roles) se agrupan lógicamente.

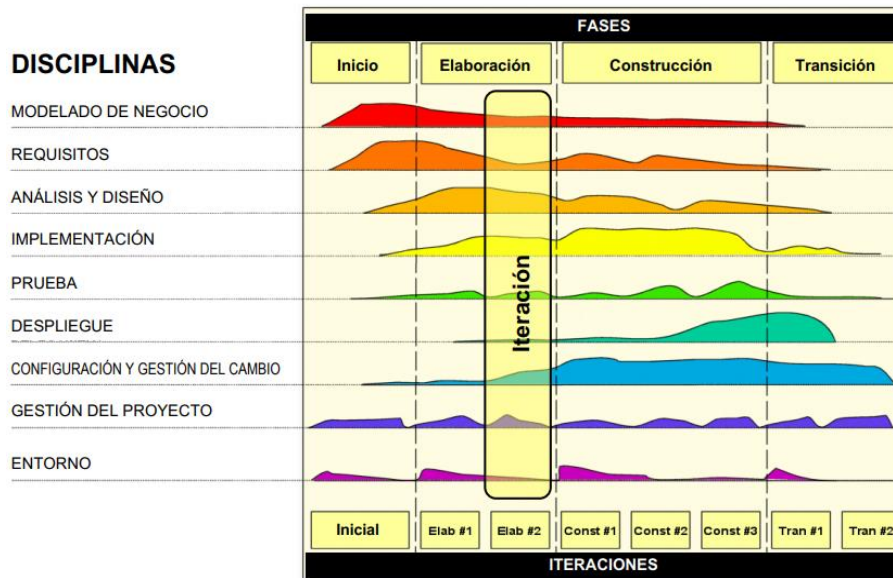


Figura 5. Ciclo de vida de RUP (Rational Software Company, 1998)

2.2.2 Mejores Prácticas de RUP (Rational Software Company, 1998)

- **Desarrollo Iterativo de Software:** Permite definir el problema, diseñar la solución, construir el software y probar el producto al final en iteraciones que liberan incrementos.
- **Arquitectura Basada en Componentes:** RUP se enfoca en el desarrollo temprano y la referencia de una arquitectura ejecutable robusta, antes de comprometer recursos para el desarrollo a gran escala.
- **Modelado Visual de Software:** RUP muestra cómo modelar visualmente el software para capturar la estructura y el comportamiento de arquitecturas y componentes.
- **Verificación de la Calidad del Software:** RUP incorpora actividades para planificar, diseñar, implementar, ejecutar y evaluar pruebas orientadas al aseguramiento de la calidad del proceso y del producto.
- **Gestión de Control de Cambios:** RUP define métodos para controlar, rastrear y monitorear los cambios para permitir un desarrollo iterativo exitoso.
- **Gestión de Requisitos de Software:** RUP describe cómo obtener, organizar y documentar la funcionalidad y restricciones requeridas; rastrear y documentar las compensaciones y decisiones; y capturar y comunicar fácilmente los requisitos comerciales. Las nociones de casos de uso y escenarios proscritos en el proceso han demostrado ser una excelente manera de capturar los requisitos funcionales y garantizar que éstos impulsen el diseño, la implementación y las pruebas de software, lo que hace más probable que el sistema final satisfaga las necesidades del usuario final.

2.3 MODELO PARA LA DEFINICIÓN DE PRÁCTICAS EN INGENIERÍA DE SOFTWARE (BARÓN, 2019)

Este es un modelo para la definición unificada y carente de ambigüedad de la práctica como constructo teórico en Ingeniería de Software. Los componentes se articulan de manera sistémica en el modelo y cumplen funciones específicas que se orientan a definir prácticas bien formadas y nombradas. Los componentes del modelo se muestran en la Figura 6 y se describen más adelante.

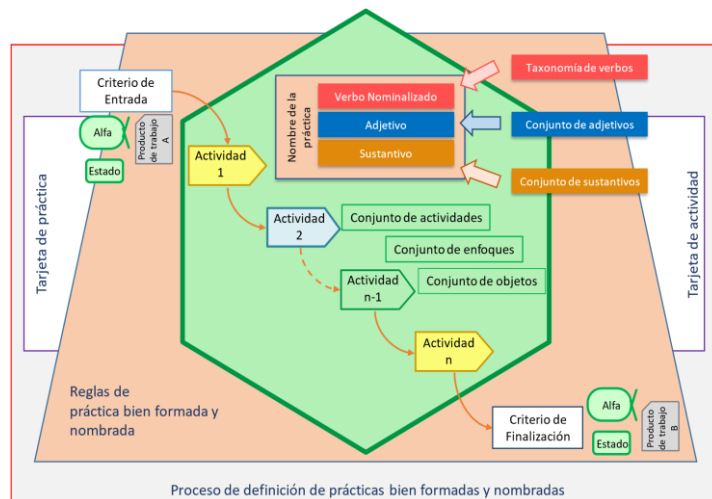


Figura 6. Componentes del modelo (Barón, 2019)

2.3.1 Criterio de entrada de la práctica

Permite determinar las condiciones que se deben cumplir para iniciar la práctica. El criterio de entrada se expresa en términos de sustantivos que pueden ser alfas y sub-alfas y un estado parcial o completo. El producto de trabajo es la evidencia de que un alfa se encuentra total o parcialmente en un estado.

2.3.2 Criterio de finalización de la práctica

Permite determinar las condiciones que se deben cumplir para terminar exitosamente la práctica. El criterio de finalización se expresa en términos de un sustantivo que puede ser alfa o sub-alfa y un estado parcial o completo. El producto de trabajo es la evidencia de que un alfa se encuentra total o parcialmente en un estado.

2.3.3 Conjunto de actividades

Permite definir el proceso de aplicación de la práctica. Entre actividades existen relaciones de secuencias y de transferencia de recursos. Cada actividad del conjunto de actividades de la práctica se conforma con acciones menores que se denominan tarea. La actividad tiene criterio de entrada y criterio de finalización que determinan las condiciones de inicio y finalización exitosa de la actividad.

2.3.4 Flujo de actividades

Permite definir el orden de ejecución de las actividades y la transferencia de recursos entre ellas.

2.3.5 Reglas de práctica bien formada

Una práctica se define como bien formada si el conjunto de actividades cumple con las reglas de coherencia, consistencia y suficiencia.

- **Regla de coherencia:** un conjunto de actividades de una práctica es coherente si el cumplimiento del criterio de finalización de cada actividad aporta en el progreso del sustantivo hacia el criterio de finalización de la práctica.
- **Reglas de consistencia:** un conjunto de actividades de una práctica es consistente si: (i) R1, existe al menos una actividad cuyo criterio de entrada es igual al criterio de entrada de la práctica; (ii) R2, existe al menos una actividad cuyo criterio de finalización es igual al criterio de finalización de la práctica; (iii) R3, para cada actividad del conjunto de actividades de la práctica, su criterio de entrada es igual al criterio de finalización de al menos otra actividad o al criterio de entrada de la práctica; (iv) R4, para cada actividad del conjunto de actividades de la práctica, su criterio de finalización es igual al criterio de entrada de al menos otra actividad o al criterio de finalización de la práctica. Las reglas R3 y R4 son válidas excluyendo las actividades que cumplen las reglas R1 y R2.
- **Regla de suficiencia:** un conjunto de actividades de una práctica es suficiente si el cumplimiento del criterio de finalización de cada actividad permite el progreso sustantivo hasta lograr el cumplimiento del criterio de finalización de la práctica.

2.3.6 Reglas de práctica bien nombrada

Una práctica es bien nombrada si su nombre incluye un verbo nominalizado, que indica lo que se hace con la práctica, un adjetivo que indica cómo se hace y un sustantivo que indica el objeto sobre el cual se aplica la práctica. Para definir los elementos que conforman el nombre de la práctica, en el modelo se propone una taxonomía de verbos nominalizados (ver tabla 1), un conjunto de adjetivos (ver tabla 2) y un conjunto de sustantivos (ver tabla 3).

Tabla 1. Taxonomía de verbos nominalizados propia de requisitos (Barón 2019)

Alfa	Requisitos		
Estado	Concebido Acotado Coherente	Aceptable	Tratado Cumplido
Taxonomía de verbos	Análisis	Verificación	Verificación
	Recolección	Aprobación	Aprobación
	Especificación	Aceptación	Aceptación
	Educción	Educción	Educción
	Negociación	Gestión	Gestión
	Priorización		
	Elicitación		

Tabla 2. Conjunto consolidado de adjetivos (Barón, 2019)

Conjunto consolidado de adjetivos			
Ágil	Colaborativo	Evolutivo	Probado
Automatizado	Compartido	Formal	Progresivo
Auto-organizado	Concurrente	Funcional	Prospectivo
Basado en actividades	Conjunto	Global	Repetible
Basado en casos de uso	Continuo	Incremental	Reutilizable
Basado en componentes	Cualitativo	Integrado	Sistemático
Basado en escenarios	Cuantitativo	Interdisciplinario	Sistémico
Basado en riesgos	Dirigido por pruebas	Iterativo	Temprano
Basado en valor	Disciplinado	Metódico	Valor compartido
Bien organizado	Empírico	Orientado al producto	Visual
Causal	Escalonado	Orientado por objetos	
Centrado en la arquitectura	Estructurado	Personalizado	

Tabla 3. Conjunto de sustantivos (Barón, 2019)

Alfas	Sub-alfas
Interesados	Representante de los interesados
Oportunidad	Necesidad
Requisitos	Ítem de Requisito
Sistema de software	Elemento del Sistema de Software
Equipo	Miembro del Equipo

Las reglas que rigen el proceso de nombramiento de una práctica son: (i) R1, el verbo nominalizado que se utiliza para nombrar la práctica es el verbo nominalizado que integra el conjunto de actividades que permite conducir el progreso del sustantivo hasta alcanzar el estado que se indica en el criterio de finalización de la práctica; (ii) R2, el adjetivo que se utiliza para nombrar la práctica es el adjetivo que integra los enfoques que se utilizan para realizar cada una de las actividades de la práctica; (iii) R3, el sustantivo que se utiliza para nombrar la práctica es el sustantivo que se indica en el criterio de finalización de la práctica.

2.3.7 Tarjetas de práctica y actividad

La tarjeta de práctica permite visualizar los elementos esenciales que describen la práctica de software. De la misma manera, la tarjeta de actividad permite visualizar los elementos esenciales que describen la actividad. Se utilizan para facilitar el seguimiento a la creación o conversión de prácticas bien formadas y nombradas. En la Figura 7 y Figura 8 se muestran las tarjetas de práctica y actividad respectivamente.

El diagrama muestra una tarjeta de práctica con un encabezado que contiene un hexágono y el texto: [Verbo nominalizado] [Adjetivo][Sustantivo].

Debajo del encabezado, hay un campo para [Breve descripción de la práctica].

El cuerpo de la tarjeta está dividido en dos secciones:

- Criterios de entrada:** Incluye un campo de selección (radio button) etiquetado como [Parcialmente en:] [(Sustantivo : Estado)] y un cuadro de texto para [productos de trabajo que son evidencias del cumplimiento del criterio de finalización].
- Criterios de finalización:** Incluye un campo de selección (radio button) etiquetado como [Contribuye con:] [(Sustantivo : Estado)] y un cuadro de texto para [productos de trabajo que son evidencias del cumplimiento del criterio de finalización].

En la parte inferior de la tarjeta, se muestra el número 1/1.

Figura 7. Tarjeta de práctica.

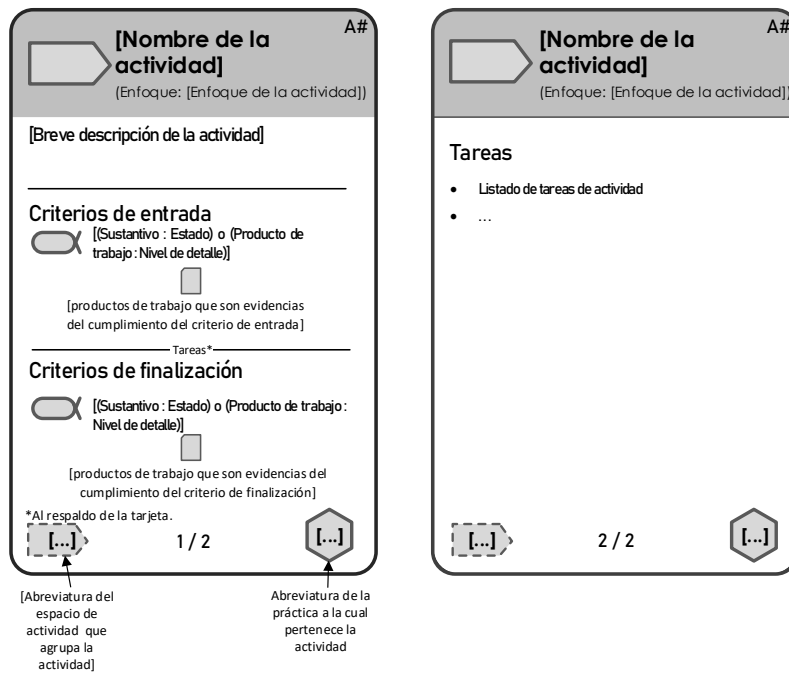


Figura 8. Tarjeta de actividad

2.3.8 Proceso de definición de prácticas bien formadas y nombradas

Los componentes del modelo se integran de manera sistémica para definir un proceso que permite guiar la definición de prácticas bien formadas y nombradas. Este proceso se puede aplicar en la creación de nuevas prácticas de Ingeniería de Software bien formadas y nombradas y en la conversión de prácticas existentes en prácticas bien formadas y nombradas. En la Figura 9 se muestra el proceso de definición de prácticas bien formadas y nombradas.

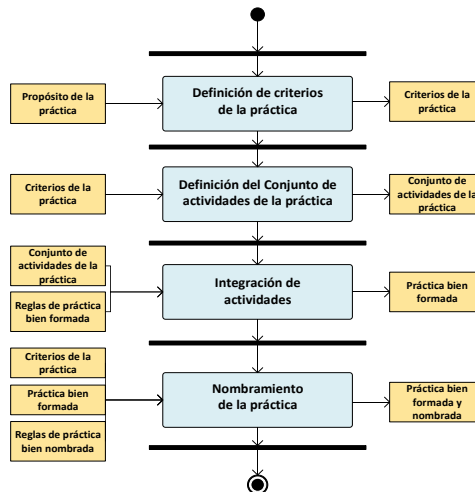


Figura 9. Proceso de definición de prácticas bien formadas y nombradas (Barón, 2019)

3. ESTUDIO SOBRE LAS FORMAS DE REPRESENTACIÓN Y DESCRIPCIÓN DE LA PGR-RUP

Este estudio permite establecer la manera como diferentes autores representan y describen la práctica Gestión de Requisitos de RUP. Este estudio consiste en una Revisión Sistemática de Literatura (RSL) que busca identificar y analizar estudios relevantes para la investigación. Como resultado, se obtiene un reporte de análisis de los estudios relevantes. Este estudio se compone de las actividades planeación, realización y reporte.

3.1. PLANEACIÓN DE LA RSL

En esta actividad se definen los elementos que orientan la RSL, se omite el uso de ventana temporal para criterios de inclusión y exclusión, ya que existen en la literatura de la Ingeniería de Software, artículos y textos clásicos que aún se utilizan para la enseñanza y aprendizaje de métodos de Ingeniería de Software como RUP. Las fuentes digitales de estudios utilizadas fueron consideradas pertinentes y suficientes para la investigación, dado que, son fuentes reconocidas en el contexto académico e industrial de la Ingeniería de Software. Los elementos que orientan la RSL se muestran en la tabla 4.

Tabla 4. RSL sobre la PGR-RUP

Propósito	Pregunta	Cadena de búsqueda
Establecer la manera como diferentes autores representan y describen la práctica Gestión de Requisitos de RUP.	¿Cómo diferentes autores representan y describen la práctica Gestión de Requisitos de RUP?	("Gestión de Requisitos" OR "Requirement Managment") AND ("RUP" OR "Rational Unified Process")
Fuentes de estudios		
<ul style="list-style-type: none"> - EBSCO - <i>Science Direct</i> - <i>Scopus</i> - Google académico - IEEE 		
Criterios de inclusión		
<ul style="list-style-type: none"> - Prácticas de Gestión de Requisitos - Prácticas de ingeniería de software - Práctica de RUP - Representación de la práctica 		
Criterios de exclusión		
<ul style="list-style-type: none"> - Estudios que orientan a RUP como una metodología ágil. - Estudios sobre metodologías ágiles. 		

3.2. REALIZACIÓN DE LA RSL

En esta actividad, se ejecutan las siguientes tareas.

3.2.1. Aplicar la cadena de búsqueda

La cadena de búsqueda que se define en la actividad de planeación, se aplica a los buscadores de cada una de las fuentes digitales de estudios seleccionadas y se obtiene un universo de estudios. Un extracto del universo de estudios se relaciona en la tabla 5.

Tabla 5. Universo de estudios

Autor	Título
(Arias <i>et al.</i> , 2018)	A Framework Managing Requirements of Software Product Line
(Kroll & Kruchten, 2003)	The Rational Unified Process Made Easy - A Practitioner's Guide to the RUP
(Baruah, 2015)	Requirement Management in Agile Software Environment
(Galic <i>et al.</i> , 2006)	Using a Single Business Pattern with the Rational Unified Process (RUP)
(Leyva & González, 2006)	Una Adaptación del Proceso Unificado de Desarrollo para la Creación de Portales Basados en Joomla
(Jones, 2010)	Software Engineering Best Practices
(Rational Software Company, 1998)	Rational Unified Process
(Passing, 2015)	Requirements Engineering in the Rational Unified Process
(Jacobson, 2000)	El Proceso Unificado de Desarrollo de Software
(Metzner & Niño, 2016)	El Proceso de Desarrollo RUP-GDIS
(Rueda, 2006)	Aplicación de la Metodología RUP para el Desarrollo Rápido de Aplicaciones Basado en el Estándar J2EE
(Vera, 2019)	Análisis de la Metodología RUP en el Desarrollo de Software Académico Mediante la Herramienta Django
(Londoño, 2008)	Análisis de la Ingeniería de Requisitos Orientada por Aspectos Según la Industria del Software
(Ambler, 2005)	A Manager's Introduction to the Rational Unified Process (RUP)
(Arias, 2007)	La ingeniería de Requerimientos y su importancia en el Desarrollo de Proyectos de Software
(Carrizo & Ortiz, 2015)	Modelos del Proceso de Educación de Requisitos: un Mapeo Sistemático
(Tabares <i>et al.</i> , 2007)	Un Método para la Trazabilidad de Requisitos en el Proceso Unificado de Desarrollo
(Jamal <i>et al.</i> , 2018)	STORE: Security Threat Oriented Requirements Engineering
(Bonfante, 2017)	Reference Model for the Integration of Business Modeling to Requirements Engineering: A Proposal from the Software Industry
(Wellsandt, 2014)	Qualitative Comparison of requirements Elicitation Techniques that are Used to Collect Feedback Information About Product Use
(García, 2018)	Experience of Using a Game of Improving Learning in Software Requirements Elicitation
(Alshazly, 2014)	Detecting Defects in Software Requirements Specification
(Maalem, 2016)	Challenge of Validation in Requirements Engineering
(Jiménez, 2016)	Modelo para la Definición Unificada de la Práctica como Constructo Teórico en Ingeniería de Software

3.2.2. Aplicar los criterios de inclusión y exclusión

Se realiza mediante 3 vueltas de análisis. Se aplica el concepto de semáforo de la siguiente manera: verde, el estudio se incluye en la investigación; rojo, el estudio se descarta para la investigación; amarillo, el estudio se somete a un análisis más riguroso en la siguiente vuelta.

En la primera vuelta se aplican los criterios de inclusión y de exclusión al título de cada uno de los estudios resultantes. En la segunda vuelta se aplican los criterios de inclusión y de exclusión al resumen de cada uno de los estudios que resultaron en amarillo en el análisis de la primera vuelta. En la tercera vuelta se aplican los criterios de inclusión y de exclusión a la introducción de cada uno de los estudios que resultaron en amarillo en el análisis de la segunda vuelta. Para esta vuelta, los estudios que resultaron en amarillo se mantienen en la investigación, para que en un análisis posterior se defina su inclusión o exclusión. Finalmente, se obtienen los estudios relevantes que pasan a la tarea de análisis. En la tabla 6 se muestra el resultado de esta tarea.

Tabla 6. Estudios relevantes para la investigación

Autor	Título
(Rational Software Company, 1998)	Rational Unified Process
(Jacobson, 2000)	El Proceso Unificado de Desarrollo de Software
(Kroll & Kruchten, 2003)	The Rational Unified Process Made Easy - A Practitioner's Guide to the RUP
(Ambler, 2005)	A Manager's Introduction to the Rational Unified Process (RUP)
(Metzner & Niño, 2016)	El Proceso de Desarrollo RUP-GDIS
(Vera, 2019)	Análisis de la Metodología RUP en el Desarrollo de Software Académico Mediante la Herramienta Django
(Jiménez, 2016)	Modelo para la Definición Unificada de la Práctica como Constructo Teórico en Ingeniería de Software

3.2.3. Análisis de los estudios relevantes que describen la PGR-RUP

Teniendo en cuenta los estudios relevantes que resultan de la tarea anterior, se realiza un análisis sobre los mecanismos de representación y descripción de la PGR-RUP. Este análisis permite identificar los componentes de la práctica a partir de la manera cómo diferentes autores la representan y describen. Los criterios que se utilizan en el análisis de los estudios relevantes son los que se identifican en (Barón, 2019) como componentes de una práctica. La nomenclatura utilizada para representar los resultados del análisis es:

N: No se encuentra el componente.

P: Se encuentra el componente parcialmente.

C: Se encuentra el componente completo.

Análisis orientado a la práctica bien formada:

Busca identificar los componentes que constituyen una práctica bien formada, los componentes son los propuestos en (Barón, 2019). El resultado de este análisis se muestra en la tabla 7 y se explica más adelante.

Tabla 7. Análisis de los mecanismos de representación y descripción de la PGR-RUP

Estudio relevante	Componentes de la práctica bien formada								
	Criterios de entrada		Criterios de finalización		Conjunto de actividades	Actividades			Flujo de actividades
	Alfa o sub-alfa	Estado	Alfa o sub-alfa	Estado		Criterio de entrada	Criterio de finalización	Enfoque	
(Rational Software Company, 1998)	P	P	P	P	N	N	P	N	P
(Metzner y Niño, 2016)	P	P	P	P	P	P	P	N	P
(Vera et al., 2019)	N	N	P	P	N	N	N	N	N
(Jacobson et al., 2000)	P	C	P	C	C	P	P	N	C
(Jiménez, 2016)	C	N	C	N	C	N	N	C	N

Como se aprecia en las filas de la tabla, ninguno de los estudios relevantes cumple con la totalidad de reglas de práctica bien formada. Por ejemplo, en (Metzner & Niño, 2016) se pueden encontrar implícitos los objetos que la práctica aborda; sin embargo, no se advierten estados. Además, el estudio contiene una serie de tablas generales de las cuales se pueden abstraer las actividades, los criterios de entrada, los criterios de finalización y el flujo de actividades. Sin embargo, el nivel de detalle no es suficiente para facilitar la aplicación de la práctica.

Análisis orientado a la práctica bien nombrada:

Busca identificar los componentes que constituyen una práctica bien nombrada. El resultado de este análisis se muestra en la tabla 8 y se explica más adelante.

Tabla 8. Análisis orientado a una práctica bien nombrada

Estudio relevante	Componentes de práctica bien nombrada		
	Nombre		
	Verbo Nominalizado	Adjetivo	Sustantivo
(Rational Software Company, 1998)	C	N	C
(Metzner y Niño, 2016)	N	N	C
(Vera et al., 2019)	C	N	C
(Jacobson et al., 2000)	C	N	C
(Jiménez, 2016)	C	N	C

Como se aprecia en las filas de la tabla, ninguno de los estudios relevantes cumple con la totalidad de reglas para que sea una práctica bien nombrada. Por ejemplo, en (Rational Software Company, 1998) se puede encontrar un verbo nominalizado, Gestión, un sustantivo, Requisitos; sin embargo, el nombre de esta práctica no tiene un adjetivo que indique la forma como se realiza la práctica.

Análisis orientado a los mecanismos de aplicación seguimiento y evaluación:

Busca identificar, en los estudios que son objeto de análisis, componentes que faciliten aplicar, seguir y evaluar la práctica. El resultado de este análisis se muestra en la tabla 9 y se explica más adelante.

Tabla 9. Análisis orientado a los mecanismos de administración de práctica

Estudios relevantes	Mecanismos de administración de práctica		
	Aplicación	Seguimiento	Evaluación
(Rational Software Company, 1998)	P	N	N
(Metzner y Niño, 2016)	P	P	N
(Vera et al., 2019)	P	N	N
(Jacobson et al., 2000)	P	P	N
(Jiménez, 2016)	P	P	N

Como se aprecia en las filas de la tabla, ninguno de los estudios analizados, integra componentes que permitan realizar de manera completa aplicar, seguir y evaluar la PGR-RUP. Por ejemplo, en (Jiménez, 2016) haciendo uso de las tablas y los gráficos se puede inferir la forma de aplicación y seguimiento de la práctica, pero no se define algún mecanismo de evaluación.

4. CONVERSIÓN DE LA PGR-RUP EN UNA PRÁCTICA BIEN FORMADA Y NOMBRADA

Aquí se presenta la propuesta de (Jiménez, 2016) en la que se describe la práctica PGR-RUP utilizando el núcleo de Essence. Esta descripción es el insumo para aplicar el proceso de definición de prácticas bien formadas y nombradas que propone (Barón, 2019).

4.1 La PGR-RUP según (Jiménez, 2016)

Esta descripción se realiza a través tablas y gráficos. En la tabla 10 se presenta un extracto de la manera como (Jiménez, 2016) realiza la descripción de la PGR-RUP.

Tabla 10. Extracto de la tabla de componentes de la PGR-RUP en la fase Inicio (Jiménez, 2016)

Espacio de actividad	Alfa	Producto de trabajo	Actividad
Explorar posibilidades	Oportunidad	Visión	Desarrollar visión
	Oportunidad	Lista de riesgos	Identificar y evaluar los riesgos
	Oportunidad	Caso de negocio	Desarrollar caso de negocio
	Oportunidad, Interesados	Glosario de negocio	Capturar un vocabulario común de negocio
	Oportunidad, Interesados	Reglas del negocio	Mantener reglas de negocio
	Oportunidad, Interesados	Visión del negocio	Establecer y ajustar objetivos
	Oportunidad, Interesados	Evaluación de la organización objetivo	Evaluar organización objetivo
	Oportunidad, Interesados	Modelo de objetos del negocio, Trabajadores del negocio, entidades del negocio, realización de casos de uso de negocio	Encontrar las entidades y los trabajadores del negocio

4.2 Conversión de la PGR-RUP

Para llevar a cabo la conversión se aplica el proceso de definición de prácticas bien formadas y nombradas.

4.2.1 Actividad 1. Definición de criterios de la práctica

El proceso para la definición de criterios de la práctica se muestra en la Figura 10 y se describe más adelante.

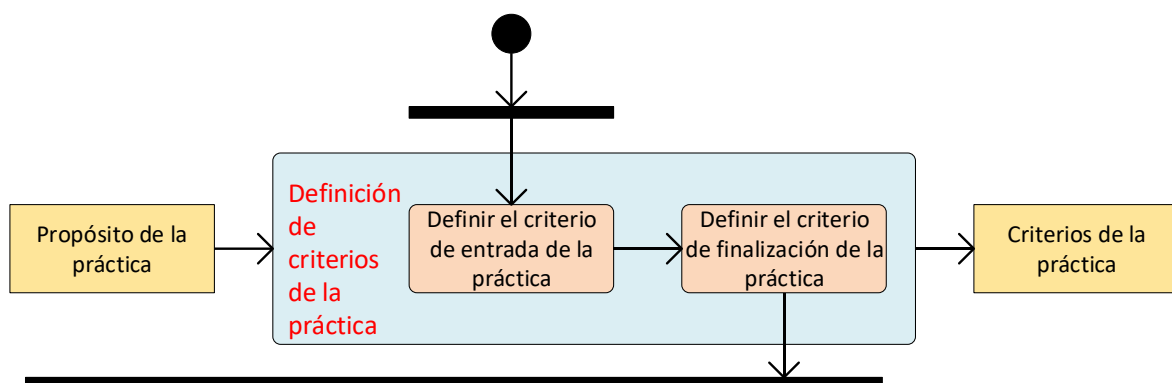


Figura 10. Definición de criterios de la práctica (Barón, 2019)

Definir criterio de entrada de la práctica

De acuerdo con el propósito de esta práctica, el criterio de entrada es (Oportunidad: Identificada), debido a que se ha identificado una oportunidad comercial, social o de negocio que podría abordarse con una solución basada en software.

Definir criterio de finalización de la práctica:

De acuerdo con el propósito de esta práctica, los criterios de finalización son (Requisitos: Dirigido) debido a que se han abordado suficientes requisitos para satisfacer la necesidad de un nuevo sistema de manera que sea aceptable para las partes interesadas y (Sistema software: Arquitectura seleccionada) debido a que se ha seleccionado una arquitectura que aborda el riesgo técnico clave y las restricciones organizativas aplicables.

4.2.2 Actividad 2. Definición del conjunto de actividades de la práctica

El proceso para la definición del conjunto de actividades de la práctica se muestra en la Figura 11 y se describe más adelante.

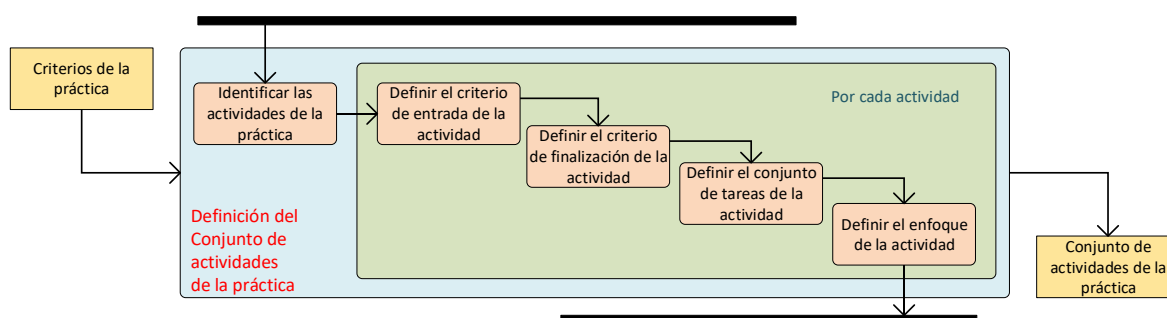


Figura 11. Definición del conjunto de actividades de la práctica (Barón, 2019)

Identificar las actividades de la práctica

Las actividades que conforman la práctica son: Crear visión, Identificar y proyectar el negocio, Formalizar el negocio, Determinar entidades y los trabajadores del negocio, Identificar actores y establecer la arquitectura de negocio, Evaluar viabilidad del negocio, Representar inicialmente los requisitos del negocio, Definir actores y detallar casos de uso, Especificar los requisitos del software, Estructurar el modelo de caso de uso, Priorizar casos de uso, Modelar prototipo de interfaz de usuario, Crear unidad de desarrollo, Diseñar prototipo de interfaz de usuario, Modelar clases, Modelar bases de datos, Diseñar clases, Diseñar bases de datos.

Definir el criterio de entrada de la actividad

Se define el criterio de entrada como condición para iniciar la actividad en términos de la dupla (Sustantivo: Estado) o (Producto de trabajo: Nivel de detalle). Las situaciones particulares que se contemplan para definir el criterio de entrada de la práctica se contemplan también para definir el criterio de entrada de la actividad. Los criterios de entrada se muestran en la tabla 11.

Definir el criterio de finalización de la actividad

Se define el criterio de finalización como condición que permite determinar la realización exitosa de la actividad en términos de la dupla (Sustantivo: Estado) o (Producto de trabajo: Nivel de detalle). La situación particular que se contempla para definir el criterio de finalización de la práctica, se contempla también para definir el criterio de finalización de la actividad. Los criterios de finalización se muestran en la tabla 11.

Definir el conjunto de tareas de la actividad:

Se relacionan las tareas que son parte de cada actividad. Las actividades con sus respectivas tareas se especifican en las tablas 12, 13 y 14 de acuerdo a la fase del ciclo de vida de RUP.

Tabla 11. Definición de criterios y enfoque de actividad

Actividad	Criterio de entrada		Criterio de Finalización		Enfoque
	Sustantivo	Estado	Sustantivo	Estado	
Crear visión	Oportunidad	Identificada	Oportunidad	Con solución requerida	Cualitativo
Identificar y proyectar el negocio	Oportunidad	Con solución requerida	Oportunidad	Con valor establecido	Incremental
Formalizar el negocio	Oportunidad	Con valor establecido	Oportunidad	Viable	Colaborativo
	Interesados	Reconocido	Interesados	Representado	Colaborativo
Determinar entidades y los trabajadores del negocio	Oportunidad	Con valor establecido	Oportunidad	Viable	Incremental
	Interesados	Reconocido	Interesados	Representado	Incremental
Identificar actores y establecer la arquitectura de negocio	Oportunidad	Con valor establecido	Oportunidad	Viable	Colaborativo
	Interesados	Representado	Interesados	Involucrado	Colaborativo
Evaluar viabilidad del negocio	Oportunidad	Con valor establecido	Oportunidad	Viable	Basado en riesgos
Representar inicialmente los requisitos del negocio	Requisitos	Concebido	Requisitos	Acotado	Incremental
Definir actores y detallar casos de uso	Requisitos	Concebido	Requisitos	Acotado	Incremental
	Interesados	Involucrado	Interesados	De acuerdo	Incremental
Especificar los requisitos del software	Requisitos	Acotado	Requisitos	Coherente	Incremental
Estructurar el modelo de caso de uso	Requisitos	Coherente	Requisitos	Aceptable	Estructurado
Priorizar casos de uso	Requisitos	Coherente	Requisitos	Aceptable	Colaborativo
Modelar prototipo de interfaz de usuario	Sistema software	Arquitectura seleccionada	Sistema software	Arquitectura seleccionada	Incremental
Crear unidad de desarrollo	Sistema software	Arquitectura seleccionada	Sistema software	Arquitectura seleccionada	Colaborativo
Diseñar prototipo de interfaz de usuario	Sistema software	Arquitectura seleccionada	Sistema software	Arquitectura seleccionada	Colaborativo
	Oportunidad	Viable	Oportunidad	Dirigido	Incremental
	Interesados	De acuerdo	Interesados	Satisfecho con la implementación	Colaborativo
	Requisitos	Aceptable	Requisitos	Dirigido	Incremental
Modelar clases	Sistema software	Arquitectura seleccionada	Sistema software	Arquitectura seleccionada	Incremental
Modelar bases de datos	Sistema software	Arquitectura seleccionada	Sistema software	Arquitectura seleccionada	Incremental
Diseñar clases	Sistema software	Arquitectura seleccionada	Sistema software	Arquitectura seleccionada	Colaborativo
	Requisitos	Aceptable	Requisitos	Dirigido	Incremental
Diseñar bases de datos	Sistema software	Arquitectura seleccionada	Sistema software	Arquitectura seleccionada	Colaborativo
	Requisitos	Aceptable	Requisitos	Dirigido	Incremental

Tabla 12. Definición de tareas de actividades en la fase de Inicio

Espacio de actividad	Actividad	Tareas
Explorar posibilidades	Crear visión	Identificar requisitos básicos del proyecto. Establecer características claves y limitaciones principales.
	Identificar y proyectar el negocio	Identificar y evaluar riesgos. Modelar caso de negocio.
	Formalizar el negocio	Redactar vocabulario común. Establecer o mantener reglas de negocio. Establecer y ajustar objetivos.
	Determinar entidades y los trabajadores del negocio	Crear modelo de objetos del negocio. Identificar trabajadores del negocio.
Comprender las necesidades de los interesados	Identificar actores y establecer la arquitectura de negocio	Identificar los actores del negocio. Estructurar el modelo de caso de uso. Definir la arquitectura de negocio.
	Evaluar viabilidad del negocio	Evaluar viabilidad con base en riesgos, caso de uso y visión del negocio.
Comprender los requisitos	Representar inicialmente los requisitos del negocio	Crear modelo de análisis y especificaciones.
	Definir actores y detallar casos de uso	Establecer los actores del negocio. Detallar caso de uso.
	Especificar los requisitos del software	Especificar los requisitos.
	Estructurar el modelo de los casos de uso	Estructurar los casos de uso utilizando la especificación de los requisitos y el modelo de casos uso.
Darle forma al sistema	Priorizar casos de uso	Priorizar casos de uso estructurados.
	Modelar prototipo de interfaz de usuario	Modelar prototipo utilizando lista priorizada de casos de uso.
Implementar el sistema	Crear unidad de desarrollo	Identificar unidad de desarrollo. Establecer unidad de desarrollo.
	Diseñar prototipo de interfaz de usuario	Diseñar prototipo utilizando el modelo del prototipo.

Tabla 13. Definición de las tareas de actividades en la fase de Elaboración

Espacio de actividad	Actividad	Tareas
Darle forma al sistema	Modelar clases	Modelar clases que intervienen en el software.
	Modelar bases de datos	Modelar bases de datos que intervienen en el software.

Tabla 14. Definición de las tareas de las actividades en la fase de Construcción

Espacio de actividad	Actividad	Tareas
Implementar el sistema	Diseñar clases	Diseñar clases que intervienen en el software.
	Diseñar bases de datos	Diseñar bases de datos que intervienen en el software.

Definir el enfoque de la actividad:

Se define la manera como se aborda la actividad en términos de un adjetivo; utilizando la información que aporta la fuente de la práctica o el conjunto de adjetivos que hacen parte del modelo. Los enfoques de cada actividad se muestran en la tabla 11.

4.2.3 Actividad 3. Integración de actividades

El proceso para el nombramiento de la práctica se muestra en la Figura 12 y se describe más adelante.

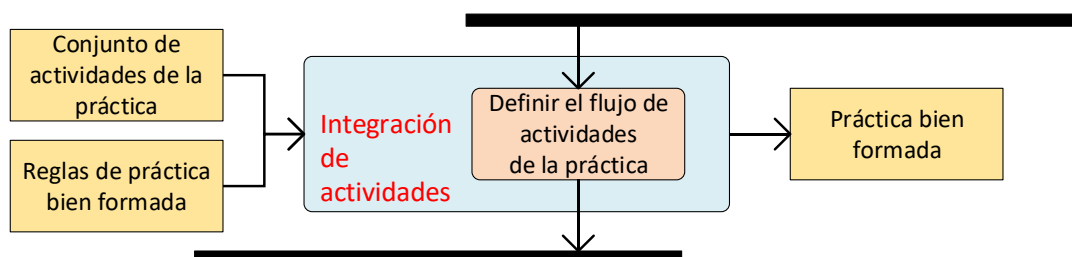


Figura 12. Integración de actividades (Barón, 2019)

Definir el flujo de actividades de la práctica

Se define el orden de ejecución de las actividades y la transferencia de recursos entre ellas aplicando las reglas de práctica bien formada, es decir, las reglas de coherencia, consistencia y suficiencia. En la Figura 13 se presenta el flujo de actividades de la práctica.

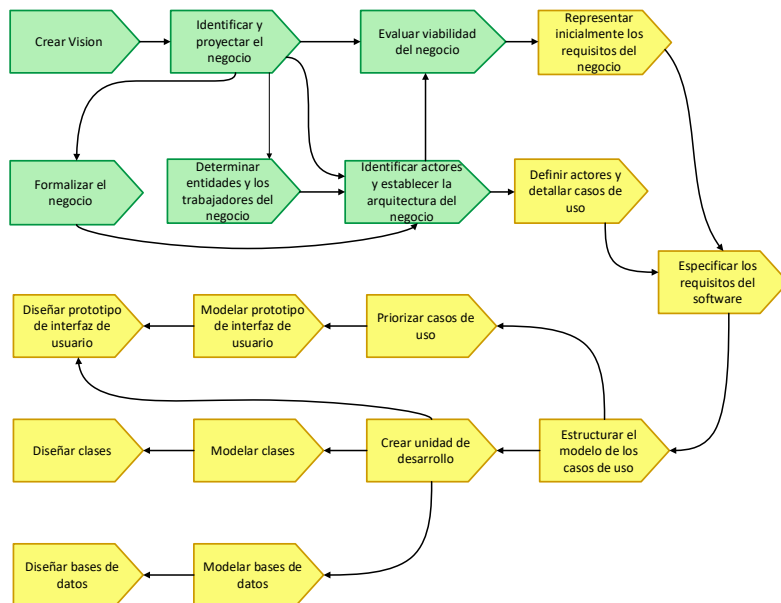


Figura 13. Flujo de actividades de la práctica

4.2.4 Actividad 4. Nombramiento de la práctica

El proceso para el nombramiento de la práctica se muestra en la Figura 14 y se describe más adelante.

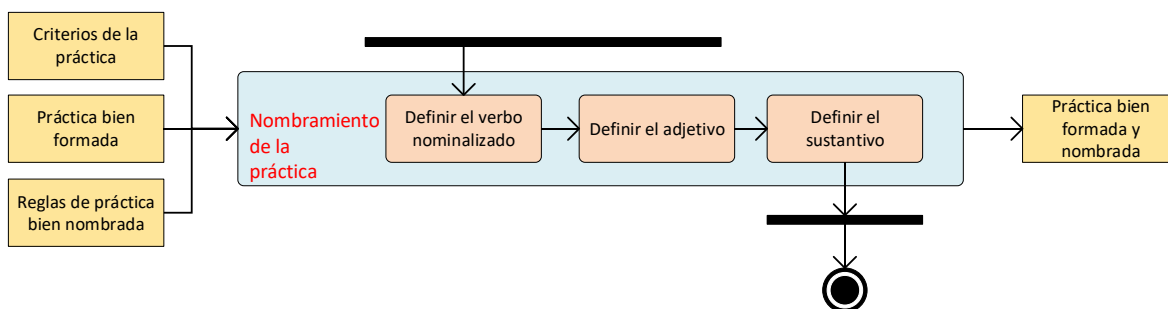


Figura 14. Nombramiento de la práctica (Barón, 2019)

Definir el verbo nominalizado

En el caso de la PGR-RUP, el verbo normalizado que resume el conjunto de actividades que conduce a la continua construcción y progreso del sustantivo requisitos hasta alcanzar el estado dirigido es Gestión. El verbo se puede identificar en la taxonomía de verbos nominalizados propia de requisitos (ver tabla 1).

Definir el adjetivo

En el caso de la PGR-RUP, los adjetivos que integran los enfoques usados para realizar las actividades de la práctica son Iterativo e Incremental. Iterativo debido a la naturaleza de RUP e Incremental debido a que la mayoría de las actividades desarrolladas tienen este enfoque. Los adjetivos se pueden identificar en la taxonomía de adjetivos (ver tabla 2).

Definir el sustantivo

Es el sustantivo que se indica en el criterio de finalización de la PGR-RUP. Requisitos es el sustantivo que, asociado con el estado dirigido, permite la realización exitosa de la práctica (ver tabla 3). De acuerdo con lo anterior, el nombre de la práctica esencializada es: Gestión Iterativa e Incremental de Requisitos de RUP (GIIR-RUP).

La práctica GIIR-RUP se compone de 18 actividades, cada una de ellas con su respectiva tarjeta. En la Figura 15 se presenta la tarjeta de práctica, en la Figura 16 se presenta la tarjeta de la actividad inicial y en la Figura 17 se presenta la tarjeta de la actividad final.

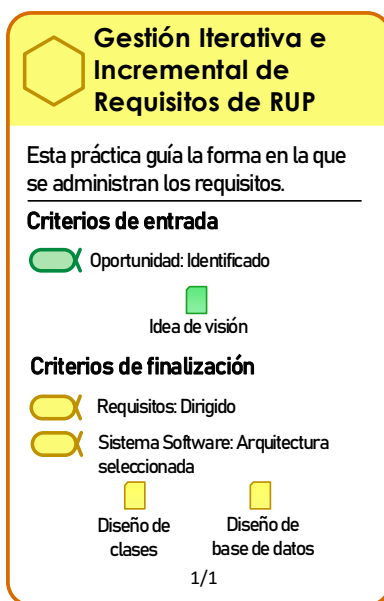


Figura 15. Tarjeta de práctica GIIR-RUP



Figura 16. Tarjeta de la actividad inicial de la práctica GIIR-RUP

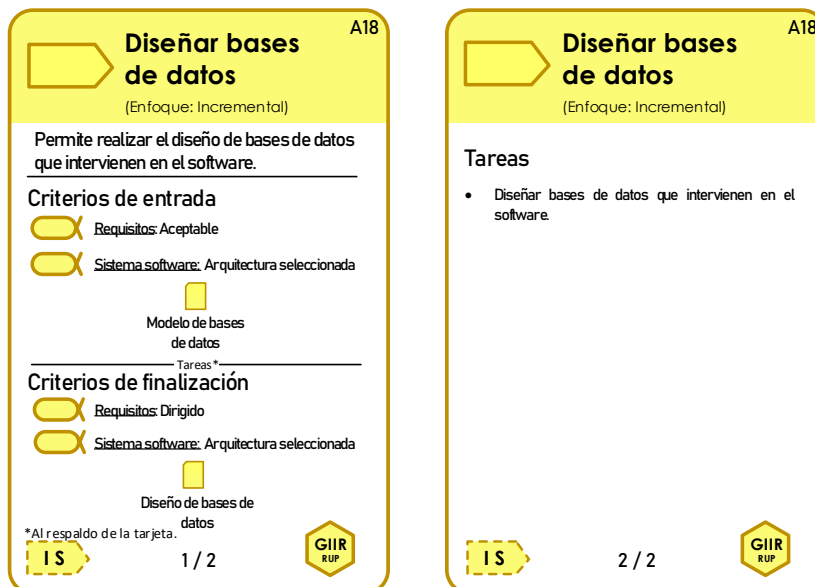


Figura 17. Tarjeta de la actividad final de la práctica GIIR-RUP

5. CONCLUSIONES Y TRABAJO FUTURO

En este artículo se presentó el proceso de esencialización de la PGR-RUP aplicando el modelo que propone (Barón, 2019). La PGR-RUP esencializada se denomina GIIR-RUP. En la práctica GIIR-RUP se identifican y definen claramente sus componentes. De esta manera, la práctica GIIR-RUP facilita a los practicantes aplicar, seguir y evaluar la práctica.

La RSL que orientó el estudio sobre las formas de representación y descripción de la PGR-RUP, se constituyó como una estrategia eficiente para, de manera estructurada e imparcial, identificar y analizar los estudios relevantes para la investigación y para reportar los resultados del estudio.

El Modelo para la Definición de Prácticas en Ingeniería de Software que propone (Barón, 2019), se constituyó en un mecanismo adecuado para definir la práctica GIIR-RUP como una práctica bien formada y nombrada. El modelo permitió identificar y definir los componentes de la práctica GIIR-RUP. Así, se facilita a los practicantes, aplicar, seguir y evaluar la práctica GIIR-RUP.

Essence se constituyó en el marco de trabajo adecuado, que integrado al modelo de (Barón, 2019), permite identificar y definir los componentes de la práctica GIIR-RUP empleando los elementos del núcleo.

Como trabajo futuro se proyecta la construcción de un software que soporte la aplicación, seguimiento y evaluación de la práctica GIIR-RUP. Se proyecta también, validar la práctica GIIR-RUP mediante su aplicación en contextos industriales y académicos como estrategia para promover su uso.

REFERENCIAS

- Barón, A. (2019). *Modelo para la Definición Unificada de Prácticas como Constructo Teórico en Ingeniería de Software* (Tesis de Doctorado). Universidad Nacional, Medellín, Colombia.
- Jacobson, I., Booch, G., & Rumbaugh, J. (2000). *El Proceso Unificado de Desarrollo de Software*. Madrid, España, Addison Wesley.
- Jiménez, L. (2016). *Representación en el Núcleo de Semat de Prácticas de Métodos de Desarrollo Basados en Planes* (Tesis de Maestría). Universidad de Nacional de Colombia, Medellín, Colombia.
- Londoño, L. F., Anaya, R., & Tabares, M. S. (2013). ANÁLISIS DE LA INGENIERÍA DE REQUISITOS ORIENTADA POR ASPECTOS SEGÚN LA INDUSTRIA DEL SOFTWARE. *Revista EIA*, 5(9), 43-52. Recuperado de <https://revistas.eia.edu.co/index.php/reveia/article/view/197>
- Metzner, C., & Niño, N. (2016). El Proceso de Desarrollo RUP-GDIS. *Revista Venezolana de Computación*, 3(1), 13-22. Recuperado de <http://svc.net.ve/ReVeCom/Vol03-No01/ReVeCom-vol03-no01-v1.1.pdf>
- Object Management Group. (2018). *About the essence specification version 1.2: Kernel and Language for Software Engineering*. Recuperado de: <https://www.omg.org/spec/Essence/1.2/PDF>
- Rational Software Company. (1998). *Rational Unified Process: Best practices for software development teams*. Recuperado de https://www.ibm.com/developerworks/rational/library/content/03July/1000/1251/1251_bestpractices_TP026B.pdf
- Somerville, I. (2011). *Software Engineering*. Boston, Massachusetts. Estados Unidos. Addison-Wesley
- Tabares, M. S., Barrera, A. F., Arroyave, J. D., & Pineda, J. D. (2013). UN MÉTODO PARA LA TRAZABILIDAD DE REQUISITOS EN EL PROCESO UNIFICADO DE DESARROLLO. *Revista EIA*, 4(8), 69-82. Recuperado de <https://revistas.eia.edu.co/index.php/reveia/article/view/186>
- Vera, D. A., Córdova, L. C., López Bermúdez, R. M., & Pacheco Mendoza, S. R. (2019). *Análisis de la metodología RUP en el desarrollo de software académico mediante la herramienta DJANGO*. *RECIMUNDO*, 3(2), 964-979. Recuperado de <https://www.recimundo.com/index.php/es/article/view/486/629>

NOTAS BIOGRÁFICAS



Jhonny Edison Chapal Vallejo es egresado del programa de Ingeniería de Sistemas de la Universidad de Nariño, es miembro del Grupo de Investigación Galeras.Net del Departamento de Sistemas de la Universidad de Nariño, actualmente se encuentra realizando el trabajo de grado bajo la modalidad de investigación en la línea de Software y Manejo de Información, la investigación está adscrita al Grupo de Investigación Galeras.Net.



John Snyder Guerrero Riascos es egresado del programa de Ingeniería de Sistemas de la Universidad de Nariño, es miembro del Grupo de Investigación Galeras.Net del Departamento de Sistemas de la Universidad de Nariño, actualmente se encuentra realizando el trabajo de grado bajo la modalidad de investigación en la línea de Software y Manejo de Información, la investigación está adscrita al Grupo de Investigación Galeras.Net.



Alexander Barón Salazar es Profesor Asociado del Departamento de Sistemas de la Universidad de Nariño y miembro del Grupo de Investigación Galeras.Net; su área de investigación es la Ingeniería de software. Su expediente académico se puede resumir así: Ingeniero de Sistemas; Especialista en Ingeniería de Software (Universidad Industrial de Santander); Especialista en Docencia Universitaria (Universidad de Nariño); Especialista en Desarrollo de Software (Universidad Eafit); Magister en Ingeniería Informática (Universidad Eafit) y Doctor en Ingeniería - Sistemas e Informática (Universidad Nacional de Colombia en la Sede Medellín). Como resultado de sus investigaciones ha desarrollado productos de software y ha sido ponente en eventos nacionales e internacionales. También, ha publicado libros y artículos en revistas científicas especializadas.



Esta obra está bajo una [licencia de Creative Commons Reconocimiento-NoComercial-CompartirIgual 2.5 México](https://creativecommons.org/licenses/by-nc-sa/2.5/mx/).

Industrial wireless channel measurements in a 2.4 GHz ISM radio band using a low-cost SDR-based channel sounder

Grzegorz Korzeniewski ¹
korzeniewski.gregor@gmail.com

Roberto Carrasco Álvarez¹
r.carrasco@academicos.udg.mx

¹CUCEI, Universidad de Guadalajara, Mexico

Abstract: Industrial wireless channel is a challenge for the design of communication systems, due to non-Line-of-Sight transmission, caused by the presence of many highly reflective obstacles, and machines in operation, which are a source of the increased noise level. The main effect, which must be analyzed, is multipath propagation. In this article, a low-cost sounding system is proposed, based on Software Defined Radio (SDR) equipment, with the intention of making sounding devices more accessible to a larger group of researchers. Likewise, the mathematical foundations and the software/hardware implementation of the wireless channel sounding system are presented, and the solutions to mitigate the synchronization issues and SDR limitations are also introduced. The performance of the proposed sounder is validated through a measurement campaign in an industrial workshop, considering the 2.4 GHz Industrial, Scientific, Medical (ISM) band. Channel sounding measurements corroborate the accuracy of the results, which converge with the channel mathematical models proposed for several industrial environments and reported in the state-of-the-art literature. In this sense, the proposed channel sounder can be used to investigate the wireless propagation environments.

Keywords: Channel sounding, Channel estimation, Industrial wireless channel, Multipath propagation, Software Defined Radio.

1 Introduction

Wireless communication is one of the greatest engineering achievements of the last several years. It has had an enormous impact on science, markets, and society, allowing us to communicate anytime, anywhere. Industry 4.0, the fourth industrial revolution, could not have come about without this technology. Its application can be found in various diagnostic and surveillance systems, sensor networks, and many others. One of the leading concepts for the modern industry is the Industrial Wireless Sensor Network (IWSN) technology, a network of sensors that can communicate with each other, increasing production efficiency throughout the industry. Another technology currently in high demand is the Internet of Things (IoT), which requires constant improvements to wireless communication systems.

Despite the abundance of improvements that wireless communication offers industry, several disadvantages must be considered. Different physical phenomena degrade the performance of a communication channel (i.e., multipath propagation, path loss, shadowing, channel interference, noise, and others), making the connection less reliable and more vulnerable. Consequently, it is important to investigate radio propagation, especially in industrial environments.

Channel sounding can be used to improve wireless communication systems. It is crucial to sound and analyze the transmission medium for network design upgrades. The knowledge obtained will increase link reliability and decrease maintenance expenses, in turn reducing production time and cost - a great benefit for industrial companies that decide to implement such innovations in their facilities.

There are various types of environments in industry, such as mines, fabrics, laboratories, etc., and each has different propagation characteristics. For example, there can be an environment which intensifies the multipath propagation effect due to many metallic objects in the area. Moreover, working electric motors in fabrics can produce shot-type noise.

To study the different propagation environments, several site-specific measurement campaigns are needed. To date, few investigations have attempted to characterize wireless channels in different industrial environments. In (Coll, 2014), channel parameters were measured in a metal workshop, a paper warehouse, a mine tunnel, and other sites. (Syed & Green, 2019) and (Düngen et al., 2019) present measurements for indoor industrial facilities with an application for IoT and industrial automation. In (Al-Samman et al., 2017), research was carried out on Ultrawideband wireless indoor channels in stationary and mobile scenarios. Similar measurements were carried out in (MacLeod et al., 2005), (Miaoudakis et al., 2005), (Luo et al., 2011). Likewise, a complete channel measurement campaign was conducted in (Cheffena, 2016) for the IWSN application. (Cheffena, 2014) presented a physical-statistical approach to channel characterization, while (Wassie et al., 2018) was focused on ultra-reliable communication techniques.

In general, the previous works were conducted using specialized and expensive channel sounding devices, such as Vector Network Analyzer (VNA), which complicates the study of new industrial propagation environments. An alternative to those systems is Software Defined Radio (SDR) based channel sounders, which are affordable and have an easily reconfigurable structure. However, there are still some serious drawbacks to this type of sounder, such as synchronization problems, sampling resolution, or hardware throughput limitations. Some projects have already implemented such SDR sounding systems (Wassie et al., 2019), (Boeglen et al., 2017), (Samayoa et al., 2018), (Hosseini & Matolak, 2018), (Maas et al., 2012) with different sounding techniques, such as OFDM modulation or the cross-correlation method. (Jamison & Frolik, 2018) has proved to be, in comparison with VNA, a highly-efficient channel sounder based on one of the cheapest SDR devices available on the market. (Boeglen et al., 2017) presented an SDR-based embedded, portable channel sounding system. (Fliedner et al., 2018) showed a design for an industrial environment channel sounder with the use of SDR equipment.

The purpose of this article is to enrich knowledge about wireless channel performance in industrial environments. A measurement campaign will be conducted in a mechanical engineering workshop, focusing on an analysis of the multipath propagation effect in the 2.4 GHz Industrial, Scientific, Medical (ISM) radio band. The collected data, together with the results from similar studies, will be helpful for network design improvements.

A further objective is to design a channel sounding system which would be accessible for a larger group of researchers, enabling the engineering society to gain a deeper comprehension of wireless propagation efficiency. The plan is to achieve this with low-cost SDR equipment. The proposed system concept and the mathematical model will be suitable for any application, environment, or frequency configuration, facilitating a possible reproduction of the presented measurement campaign in any other transmission band. Particular attention is paid for offering a solution to the synchronization problem. Additionally, system mobility will be enhanced with the use of Raspberry Pi 4 and GPS disciplined oscillators (GPSDO). Performance of the designed system will be tested by taking the aforementioned measurements in an industrial environment and validated using theoretical propagation models.

This article is organized as follows: Chapter 2 introduces the basics of radio wave propagation, while Chapter 3 presents the mathematical model of the channel sounding process. Chapter 4 explains physical implementation of the system. Chapter 5 describes the measurement campaign. Chapter 6 offers a discussion of the sounding system, and Chapter 7 presents the project conclusions.

2 Radio wave propagation concepts

In wireless communication, information is propagated by means of an electromagnetic wave. There are many phenomena that interfere with wave propagation, the most important being reflection, diffraction, and scattering. Reflection occurs when the propagated electromagnetic wave strikes an object significantly larger than the wavelength, i.e., buildings, walls, or the ground. Diffraction takes place on the sharp edges of the objects, causing the wave to bend. Scattering happens when the wave is propagated through a medium containing many small objects per unit volume (e.g. tree leaves, street signs, or lamp posts) (Rappaport, 2002).

In the following sections, the basic concepts of the radio channel and the physical phenomena that affect its performance will be explained.

2.1 Channel model

The word "channel" can have many meanings, but in its general sense, it means everything between the signal source and its sink. Basically, a channel model can be considered a mathematical or algorithmic representation of the propagation medium's transfer characteristics (Jeruchim et al., 2002).

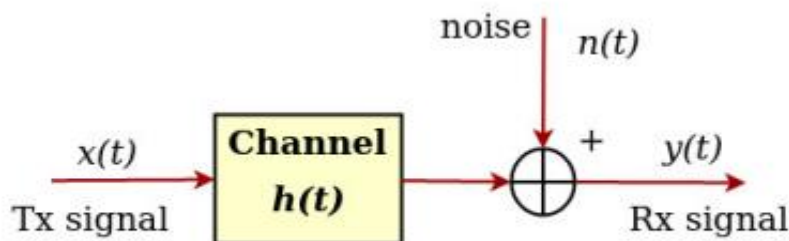


Figure 1: Basic channel model

In the figure 1, the channel model is mathematically represented as a linear, time-invariant filter with impulse response $h(t)$, whose input is the transmitted signal $x(t)$. If the signal is transmitted in the presence of white Gaussian noise $n(t)$, filtered signal and noise components are summed and the output signal $y(t)$ has the form of Equation (1) (Grzybkowski et al., 2006).

$$y(t) = \int_{-\infty}^{\infty} x(\tau)h(\tau)d\tau + n(t) \quad (1)$$

To characterize a radio channel, it is necessary to analyze the physical phenomena which assist with the propagation of electromagnetic waves and their interaction with the environment. Channel performance greatly depends on the operating frequency, the transmission environment, transceiver movement, and other factors. Typically, the channel parameter analysis is divided into two propagation modes. One of these is the Line-of-Sight (LoS) scenario, in which there are

no obstacles in the path between the transceivers (the devices "see" each other). In the other, non-Line-of-Sight (NLoS), the path is obstructed.

Unlike the stationary and predictable wired techniques, the wireless channel is random and difficult to analyze. It is often defined as a stochastic process and requires a statistical approach for its description. In practice, many channel models are based on fitting external (empirical) observations (Stuber, 2017).

2.2 Fading channel classification

The fading channel is a communication channel which experiences variation of the attenuation of a signal as a function of time, geographic position, and radiofrequency. Fading is often described as a random process. Classification of fading types is depicted in Figure 2 (Stuber, 2017).

The two main groups of fading type are large-scale fading and small-scale fading. Large scale fading is due to distance-dependend path loss of signal and shadowing provoked by large objects such as buildings and hills. Small-scale fading is characterized as rapid fluctuations of the signal over a small distance or a short period. One small-scale effect is multipath propagation, which can be further divided into flat and frequency selective fading. Small-scale fading also refers to the Doppler effect, which can be interpreted as fast or slow (Tse & Viswanath, 2004). Each of the abovementioned effects will be explained in the following chapters.

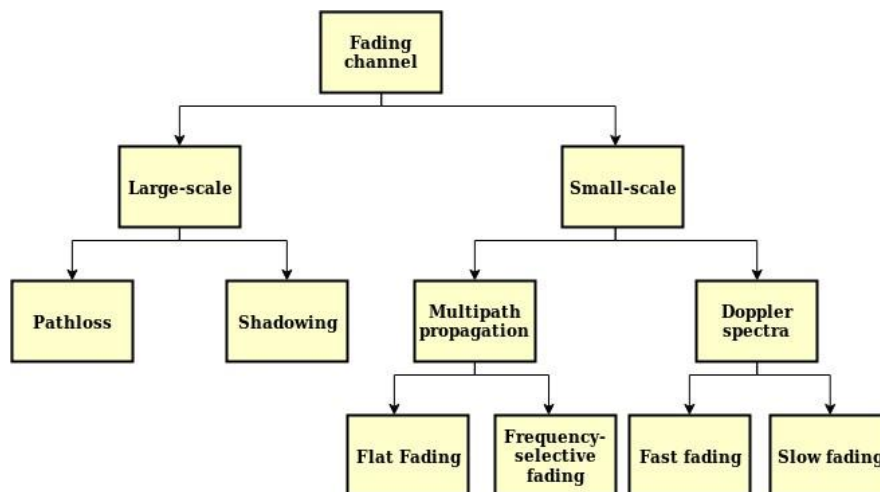


Figure 2: Fading channel classification.

2.3 Large-scale fading

Large-scale fading models signal strength over a large distance between the transmitter and the receiver. The signal received by a mobile radio from a transmitter can be represented as

$$S_r = S_t + G_{tx} + G_{rx} + L_p \quad [dBm], \quad (2)$$

where S_r is the power of the received signal in dBm, S_t is the power of transmitted signal in dBm, G_{tx} and G_{rx} are antenna gain (dB) of a transmitter and receiver, respectively, and L_p is the propagation loss in dB. The equation 2 is often called a "link budget". The antenna gain depends

on the antenna's physical design, position, and orientation. The L_p component is the most difficult to define and relies on two effects: path loss and shadowing (Jeruchim et al., 2002).

2.3.1 Pathloss

Most propagation loss models are based on the fact that the average received signal power decreases logarithmically with distance. Path loss is an exponential function and can be represented by Equation 3

$$L_p = \alpha + \beta \log_{10}(R) \quad [dB], \quad (3)$$

where α and β are parameters determined by the propagation medium, and R is the transmitter-receiver separation distance. The parameter values for each model can be defined by measurement campaigns in specific environments and scenarios (Jeruchim et al., 2002).

2.3.2 Shadowing

The equation 3 does not consider the shadowing effect. In the analysis, it is necessary to include the case in which the propagated wave is "shadowed" by buildings or other obstacles, leading to the immense difference between the measured signal and its predicted average strength. It can be mathematically represented by adding to the Equation 3 a random variable X_σ , as follows

$$L_p = \alpha + \beta \log_{10}(R) + X_\sigma \quad [dB]. \quad (4)$$

X_σ is a zero-mean Gaussian distributed random value in dB with standard variation σ . It symbolizes a random shadowing effect that occurs over a large number of measurements for the same transmitter-receiver separation but with different obstacles in the propagation path (Rappaport, 2002).

2.4 Small-scale fading

Small-scale fading describes rapid fluctuations to signal strength over a short-range or period, so the large-scale effect can be neglected for this analysis. This fading type is the consequence of the constructive and destructive interference of the multiple signal paths between the transmitter and the receiver. This means that the radio receives many copies of the emitted signal a phenomenon known as multipath propagation. Furthermore, the movement of transceivers or enclosing objects results in the Doppler frequency shift on multipath components (Rappaport, 2002).

2.4.1 Multipath propagation

The example of multipath propagation is presented in Figure 3. Emitted waves can be reflected in the environment by reflectors, for example surrounding buildings and objects, creating "echoes" of the signal. Echoes of the signal are called multipath components. These multipath components are weighted and delayed versions of the original propagated signal, which combine at the receiver antenna. Thus, the impulse response $h(t)$ of a multipath channel is defined mathematically as follows:

$$h(t) = \sum_{m=0}^{M-1} a_m \delta(t - \tau_m), \quad (5)$$

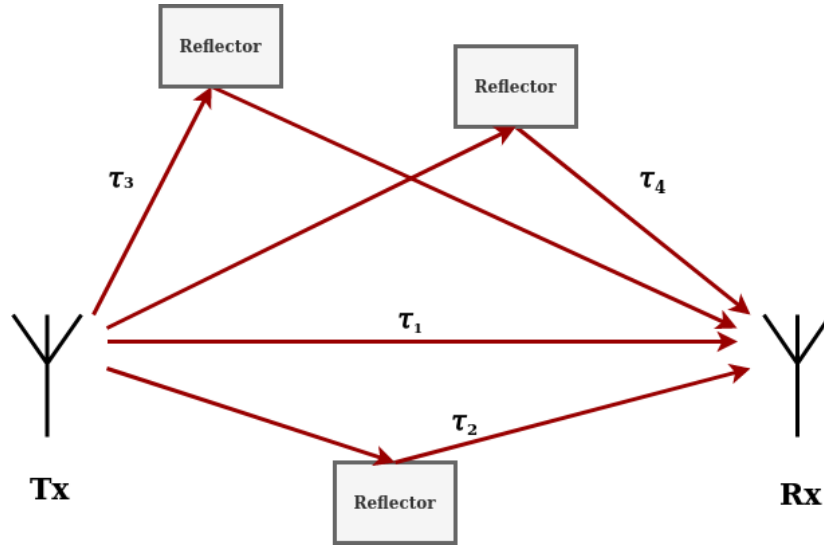


Figure 3: Example of multipath propagation.

where a_m and τ_m are the amplitude and arrival time of a m -th signal path, respectively, $\delta(t)$ is the Dirac delta function, and M is the maximum number of paths.

Each component has a random amplitude and phase, inducing small-scale signal strength fluctuations and provoking signal time dispersion and delays, leading to intersymbol interference (Rappaport, 2002). This has a significant impact on communication performance, making it crucial to analyze the following channel parameters related to this phenomenon:

- **Power Delay Profile (PDP)**, $P_h(t)$, describes the distribution of the received signal power in time from a transmitted impulse and is defined as a spatial average of squared channel impulse response $h(t)$, as follows:

$$P_h(t) = R_{hh}(0, \tau) = E[|h(t)|^2], \quad (6)$$

where $E[\cdot]$ is the expected value and R_{hh} is the autocorrelation function.

- **Mean delay spread**, τ_{mean} , is the averaged multipath delay and is defined as the first moment of the PDP (Eq.7), with the units in seconds.

$$\tau_{mean} = \frac{\sum_i \tau_i P_h(\tau_i)}{\sum_i P_h(\tau_i)} \quad [s] \quad (7)$$

- **RMS delay spread**, τ_{rms} , is a measurement of the channel time dispersiveness and determines the maximum symbol rate achievable by a communication system before intersymbol interference occurs. Therefore, it is often cited as the most important multipath parameter, determining the overall wireless channel performance. It is defined as the second central moment of the PDP (Eq.8) (Grzybowski et al., 2006).

$$\tau_{rms} = \sqrt{\frac{\sum_i (\tau_i - \tau_{mean})^2 P_h(\tau_i)}{\sum_i P_h(\tau_i)}} \quad [s] \quad (8)$$

- **Max delay spread**, τ_{max} , is the maximum time spread of the multipath components when a predefined threshold is exceeded and is defined as

$$\tau_{max} = \tau_{last} - \tau_{first} \quad [s], \quad (9)$$

where τ_{first} and τ_{last} are the arrival time of the first and the last multipath components, respectively.

- **Coherence bandwidth**, $B_{x\%}$, is a statistical parameter that helps to define whether the channel is frequency selective over a given frequency band or frequency non-selective (flat). It is a measure of a frequency range over which the attenuation is constant and linear in phase. The coherence bandwidth has been related by inverse proportionality to τ_{rms} , as defined in the following equation:

$$B_{x\%} = \frac{1}{\alpha \tau_{rms}} \quad [Hz], \quad (10)$$

where values of α span between 5 and 50, which corresponds to the respective correlation values of 50% and 90% (Coll, 2014).

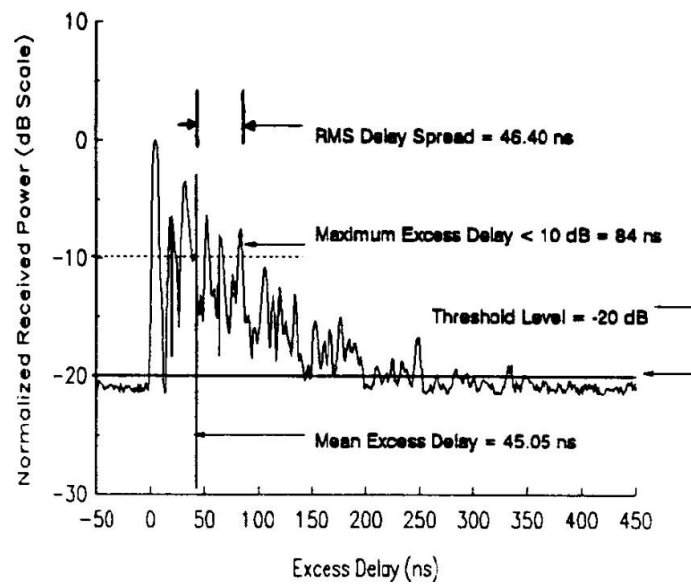


Figure 4: RMS delay spread, mean delay spread and max delay spread shown in an example of an indoor power delay profile (Rappaport, 2002).

In Figure 4, the aforementioned parameters are highlighted in the power delay profile of an indoor channel. In this example, the received power is normalized. The threshold for the maximum excess delay is set to -10 dB and the noise level to -20 dB. It can be seen that the delay values for the indoor environment are in the range of 1-200 ns.

2.4.2 Doppler Effect

The relative movement of the transmitter and the receiver cause multipath components to experience a shift in frequency. This effect is called the Doppler effect and is proportional to the velocity and motion direction of a transceiver with respect to the direction of arrival of the received multipath wave. Doppler shift f_d is given by the following equation:

$$f_d = -\frac{v}{\lambda} = -f_c \frac{v}{c} \quad [\text{Hz}], \quad (11)$$

where f_c is the carrier frequency (in Hz), v is the velocity of the source or sink of the signal and c is the speed of light (both in $\frac{m}{s}$). Doppler shift is negative when the transceivers move away from each other (Rappaport, 2002).

2.5 Industrial environment characteristics

All the aforementioned concepts occur in the real communication system, and their intensity depends greatly on the site and surroundings. In the industrial environment, more Non-Line-of-Sight type of propagation is expected due to the communication medium being filled with obstacles, leading to signal attenuation and shadowing. Moreover, metallic machinery can provoke an intensified multipath effect caused by many reflections, diffractions, and scattering, and any random movement by people or robots in the area may cause time-variant effects in the propagation. Channel performance can be highly degraded if these phenomena are not taken into consideration when designing an industrial wireless communication system (Coll, 2014).

3 Channel sounding mathematical model

In this chapter, the mathematical foundations of the channel sounding process will be presented, showing how the channel impulse response (which is a stochastic process) can be estimated by means of cross-correlation function and white Gaussian noise. Furthermore, the estimation of PDP, using the autocorrelation matrix method, will be presented.

3.1 White Gaussian noise

The first step in designing the mathematical model is to understand what white Gaussian noise is and what its properties are. White Gaussian noise is a sequence of uncorrelated random variables with a Gaussian distribution. Without a loss of generality, an assumption of zero mean and variance $E[n^2(t)] = \sigma^2$ is made for the Gaussian noise. An important property of white noise is its constant power spectral density, meaning uniform power of each frequency. The average noise power at a period (t_1, t_2) can be calculated as follows:

$$P = E \left[\frac{1}{t_2 - t_1} \int_{t_1}^{t_2} n^2(t) dt \right] = E[n^2(t)] = \sigma^2. \quad (12)$$

In the channel sounding process, it helps to analyze all frequencies of the bandwidth at the same time. An example of white Gaussian noise (samples in time domain and power spectral density) is shown in Figure 5.

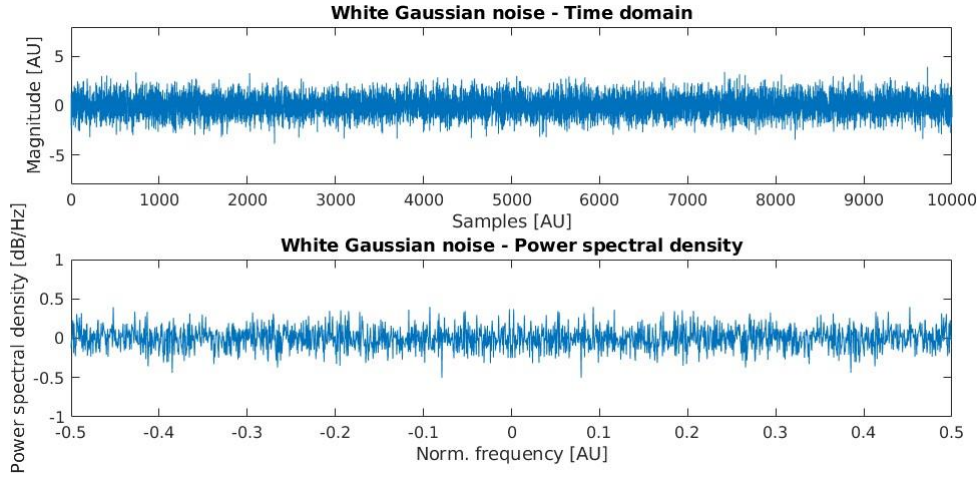


Figure 5: Example of white Gaussian noise. Time domain samples and power spectral density.

Another significant characteristic of white noise is its autocorrelation function, which is defined as:

$$R(\tau) = n(\tau) * n(-\tau) = E[n(t)n(t - \tau)] = \sigma^2\delta(\tau), \quad (13)$$

where $*$ denotes the convolution operation. This implies $R(\tau) = 0$ for any lag $\tau \neq 0$. The autocorrelation function of white noise will play a crucial role in the design of a channel sounding process (Hayes, 1996).

3.2 Channel impulse response estimation

Cross-correlation is a measure of similarity of two signals as a function of the displacement of one relative to the other. This function will be used to estimate the channel impulse response. By definition, the cross-correlation function between signal $y(t)$ and $x(t)$ can be represented as (Proakis & Manolakis, 2006)

$$R_{yx}(\tau) = y(\tau) * x(-\tau). \quad (14)$$

The channel can be thought of as a Linear, Time-Invariant system (LTI). The output signal $y(t)$ of the system can be represented as a convolution of the impulse response function $h(t)$ and the input signal $x(t)$ (Sharif & Sha'ameri, 2007)

$$y(t) = h(t) * x(t). \quad (15)$$

In this case, the input of the system $x(t)$ will be white Gaussian noise and $h(t)$ is the channel impulse response that will be estimated.

If the signal $y(t)$ in the Eq.14 is substituted by the signal definition shown in Eq.15, the cross-correlation can be represented as follows:

$$R_{yx}(\tau) = y(\tau) * x(-\tau) = h(\tau) * x(\tau) * x(-\tau). \quad (16)$$

It can be easily seen that $x(\tau) * x(-\tau)$ is the autocorrelation of white noise (eq. 13).

$$R_{yx}(\tau) = h(\tau) * x(\tau) * x(-\tau) = h(\tau) * R_{xx}(\tau) = h(\tau) * \sigma^2 \delta(\tau). \quad (17)$$

Using convolution properties for scalars and delta functions, the equation is developed in the following manner:

$$R_{yx}(\tau) = h(\tau) * \sigma^2 \delta(\tau) = \sigma^2 h(\tau), \quad (18)$$

where σ^2 is the average power of the input signal.

As proved by means of equations 14 - 18, the cross correlation function of an output signal of a system with white noise input is equal to the system's impulse response

As noted above, the channel impulse response function is a stochastic process. To improve estimation, it is necessary to analyze several realizations of the process. Cross-correlation should be repeated several times, and the results should be averaged to obtain a better channel impulse response estimation.

3.3 Estimation of power delay profile

The next part of the channel analysis is to estimate the power delay profile. For this, the autocorrelation function will be calculated using the previously estimated channel impulse response $h(t)$, as shown in the Eq.6.

The autocorrelation function is a second-order statistical characterization of a random process. Before this function can be computed, a conversion of the continuous impulse response $h(t)$ into discrete domain $h[n]$ is needed, which is carried out as follows:

$$h(t) * g(t)|_{t=nT_s} = h[n], \quad (19)$$

where T_s is the sampling period, and $g(t)$ is the function that delimits the signal bandwidth to a certain value.

As a result, the discrete impulse response $h[n]$ can be represented, based on the Equation 5, as follows:

$$h[n] = \sum_{m=0}^{M-1} a_m g(nT_s - T_m), \quad (20)$$

and can be stated in a vector form as:

$$h[n] = [h[0], h[1], h[2] \dots, h[p-1]], \quad (21)$$

where p is given by the maximum delay of a_m and the duration of $g(t)$ function.

The autocorrelation function R_{hh} of impulse response $h[n]$ is established as follows:

$$R_{hh}[n_1, n_2] = E \left[\left(\sum_{m=0}^{M-1} a_m g(n_1 T_s - T_m) \right) \left(\sum_{m'=0}^{M'-1} a_{m'} g(n_2 T_s - T_{m'}) \right) \right]. \quad (22)$$

Further, a_m is a random variable, and for PDP estimation, is assumed to be uncorrelated scattered (US). The US assumption is satisfied if the phase of any multipath component is uncorrelated to another component with a different delay (Molisch, 2011). Thus, the final form of the autocorrelation function is

$$R_{hh}[n_1, n_2] = \sum_{m=0}^{M-1} \sigma_m^2 g(n_1 T_s - T_m) g(n_2 T_s - T_m), \quad (23)$$

where σ_m^2 is the power of the m -th component.

The autocorrelation function $R_{hh}[n_1, n_2]$ can be expressed in a matrix form, as follows:

$$\mathbf{R}_{hh} = \begin{pmatrix} R_{hh}[0,0] & R_{hh}[0,1] & R_{hh}[0,2] & \cdots & R_{hh}[0,p-1] \\ R_{hh}[1,0] & R_{hh}[1,1] & R_{hh}[1,2] & \cdots & R_{hh}[1,p-1] \\ R_{hh}[2,0] & R_{hh}[2,1] & R_{hh}[2,2] & \cdots & R_{hh}[2,p-1] \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ R_{hh}[p-1,0] & R_{hh}[p-1,1] & R_{hh}[p-1,2] & \cdots & R_{hh}[p-1,p-1] \end{pmatrix}. \quad (24)$$

The elements of the main diagonal of the matrix \mathbf{R}_{hh} are the values of the autocorrelation function that define the estimation of Power Delay Profile

$$P_h[n] = [R_{hh}[0,0], R_{hh}[1,1], R_{hh}[2,2], \dots, R_{hh}[p-1,p-1]]. \quad (25)$$

4 Channel sounding implementation

In practice, the channel sounding process requires a device capable of sampling a radio signal at the desired frequency, with a suitable antenna gain and radiation pattern. The most important factor to consider is the synchronization between transceivers. The physical implementation of the channel sounding system will be presented in this chapter. Additionally, a sounding signal structure and a solution for the synchronization problem will be offered.

4.1 Equipment

The key equipment used for channel sounding is the bladeRF x40. This product from the Nuand company is an SDR, which means that the radio components (such as mixers, filters, modulators, etc.) can be implemented by software. The frequency range of 300 MHz to 3.8 GHz is suitable for measurements in the 2.4 GHz ISM band. The maximum bandwidth of the instrument is 28 MHz. It is capable of acquiring 12-bit IQ samples with a sample rate of up to 40 MS/s, which results in the maximum time resolution of 25 ns. To achieve such a high data rate between the bladeRF board and the controller unit, the producer implemented a USB 3.0 interface. BladeRF has an independent transmission and reception signal paths, so it can operate in a Full-Duplex mode. It also has an external clock port, crucial for solving the synchronization problem. Two bladeRF boards are used in the system: one for signal transmission and another for reception (Nuand, 2014).

The outgoing signal is amplified by an external power amplifier to reach 15 dBm of transmission power. A Crystek CRBAMP-100-6000 amplifier, with 18 dB gain and 1 dB compression point at 17 dBm, is used in this implementation. The received signal is amplified using bladeRF's onboard low noise amplifier. Both bladeRFs are connected, via an SMA connector, to omnidirectional antennas suitable for the 2.4 GHz frequency band. The gain of the antennas is 2 dBi and the length is 10.8 cm.

A pair of Leo Bodnar's Mini Precision GPSDOs provide a reference clock for the SDR devices. The instrument output is a 3.3 V CMOS signal. The output frequency can be programmed to any value between 400 Hz and 810 MHz. For the purposes of this project, the frequency is set to 38.4 MHz to meet the bladeRF specifications. The GPSDOs are connected to bladeRF's external clock port with an SMB cable. The device uses an active GPS antenna with a 3 m cable and is powered by 5V DC from a USB port (LeoBodnar, 2015).

In the proposed bladeRF system configuration, a controlling unit with a USB 3.0 interface and a minimum 2 GB RAM is required to both input the transmission signal to the SDR device and to receive data. The newest version of a single-board computer, Raspberry Pi 4 with 2 GB RAM, has been chosen to be the transmission side of the sounder due to its USB 3.0 ports, essential for correct sounder operation. For the reception part and system controlling unit, an ASUS N550J laptop with Ubuntu Linux 18.04 operating system, Intel Core i7 CPU, and 16 GB RAM was used.

4.2 Channel sounding system

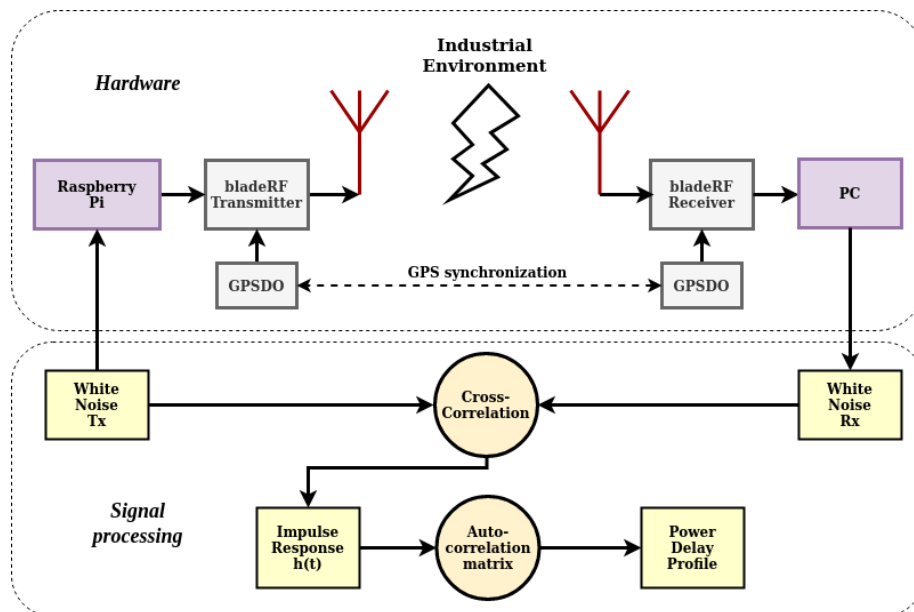


Figure 6: Sounding system diagram.

Figure 6 shows the channel sounding system, which is divided into 2 sections: hardware implementation and signal processing. The PC unit is the sounder controller unit. One of its tasks is to establish a connection (over a local WiFi network) with the remote transmission part of the system, which is controlled by Raspberry Pi, and send it a command to start the channel sounding process. The PC's other task is to handle the signal processing part of the system. Both tasks are performed by a Python program developed by the author, with the help of open-source Python libraries.

The first step of the sounding process is to generate a white noise sequence, which is produced by the PC unit and sent to Raspberry Pi. The same signal is stored for further processing in the signal processing part. The sounding signal is passed to bladeRF, amplified, and propagated through the omnidirectional antenna to the industrial environment. On the receiver side,

there is another bladeRF device connected to the antenna and controlled by the PC unit. Both bladeRF boards are controlled via USB 3.0 interface and bladeRF-cli (bladeRF Command Line Interface, a software tool provided by the board's producer) basic commands. The main problem with the sounding system is the synchronization between the transceivers, clarified in chapter 4.4. Received samples are stored and passed to the signal processing part.

The cross-correlation function of transmitted and received white noise is calculated in the signal processing section, obtaining the estimation of the channel impulse response used to compute the auto-correlation matrix in order to estimate the PDP.

The proposed system software is available in a public repository (link to the repository), which contains Python scripts and a Graphical User Interface to perform channel impulse response estimation with a user-friendly program.

4.3 Sounding signal structure

The following transmission signal structure is proposed for the sounding system: The duration of the signal is 1 second, and the signal is divided into 100 000 windows of a duration of 10 μ s. Each window is filled with white Gaussian noise in a time ratio of 1% (0.1 μ s), leaving the remaining part with a null value, as shown in Figure 7. This is to assure that each window will be an uncorrelated realization of the sounding process. Filling the window with only white Gaussian noise would influence the subsequent estimation, which does not coincide with the assumptions made in the mathematical model proposed in chapter 3. The estimation of the impulse response is concluded with an average of 100,000 realizations.

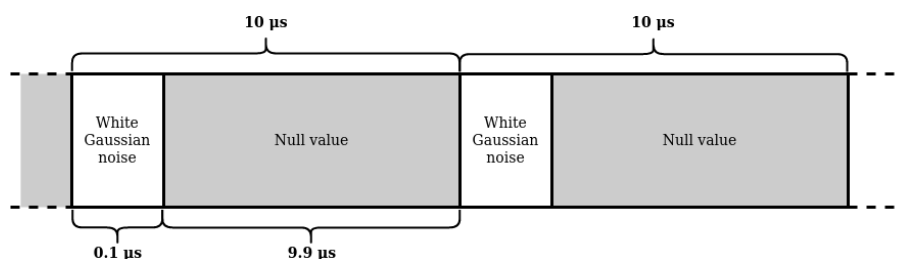


Figure 7: Sounding signal structure diagram.

4.4 Synchronization

Synchronization is an important part of the channel sounding process. Leaving this problem unsolved leads to several errors in signal modulation and processing. In SDR-based sounders, there are two issues to deal with.

The first problem to solve is radio oscillator synchronization. In the modulation process, it is crucial that the transceivers rely on the same reference clock in order to avoid synchronization errors, i.e., frequency shift. In this article, the proposed solution is to use GPSDO. This reference oscillator locks its output to pulses sent by GPS satellites. Hence, every SDR used will share the same reference clock for the modulation process.

Another synchronization process must be done in the data processing part. In order for the received white noise to correspond to the transmission sequence samples, it must first be aligned to its first sample. In practice, the sounding signal will not necessarily be at the beginning of received samples due to hardware (bladeRF and the controller unit) delays. The solution is to add

a synchronization sequence at the beginning of the transmission. In this case, a single tone of a known 20 kHz frequency is transmitted for 0.5 seconds, followed by a 0.1-second break, and then the sounding signal transmission is begun (the values were chosen empirically). A diagram of the synchronization sequence is shown in Figure 8. This enables determining precisely where the sounding process starts. It is a pure post-processing problem, meaning that this alignment is set after the transmission, not in the real-time operation.

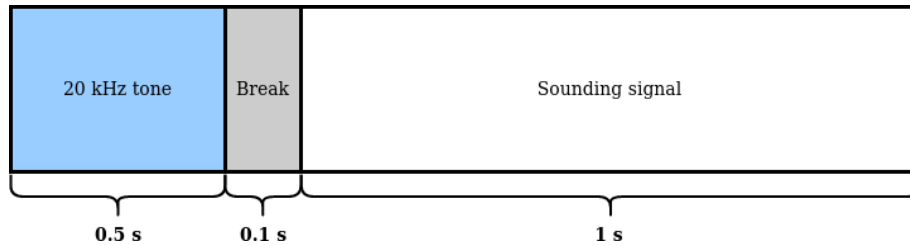


Figure 8: Transmission synchronization sequence diagram.

5 Measurement campaign results

This section describes a measurement campaign. First, the details of the measured environment will be presented. Furthermore, several data will be analyzed. The main focus of the experiment was on the multipath propagation effect. A spatially-averaged PDP was estimated and the results were compared with the Saleh-Valenzuela indoor channel model.

5.1 Environment description

The measurement campaign was performed in a workshop of the mechanic faculty at the University of Guadalajara, Mexico. The building has an open space hall with dimensions of 42 m × 18 m. It consists of concrete floor and walls, a 12-m high metallic ceiling, metallic stairs, balcony, and railings. The outside wall is made of glass. A view of the workshop is shown in Figure 9.



(a)



(b)

Figure 9: A view of the workshop, where the experiment was performed: a) General view
b) Glass wall

Due to the absence of machinery or other obstacles, each measurement was performed in a Line-of-Sight scenario. There was no movement during the experiment by the antennas, the people, or the objects.

Figure 10 presents the building dimensions along with measurements points. The transmitting antenna was situated at a fixed position Tx, and the receiver antenna at four different measurement points (Rx1-Rx4). It is worth noting that the Rx1 point was placed under the balcony. The height of the transmitter and receiver antenna was 0.45 m and 0.75 m, respectively.

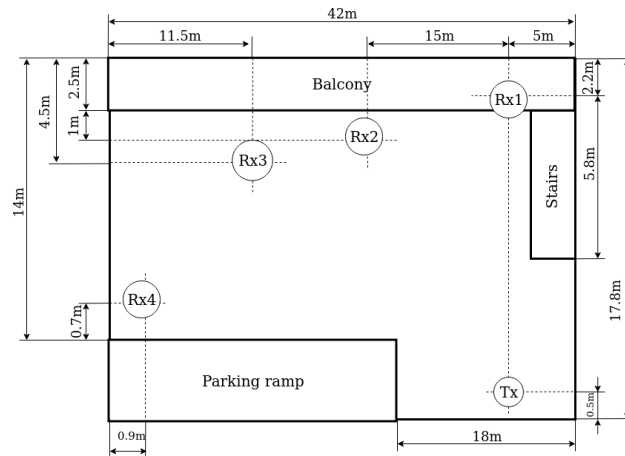


Figure 10: Building measurements and antenna positions. Tx: transmitting antenna. Rx1-Rx4: receiving antenna positions. Note: antenna Rx1 is placed under the balcony.

5.2 Multipath propagation analysis

During the measurement campaign, the multipath propagation effect was analyzed with equipment calibrated for the carrier frequency of 2.3 GHz, 28 MHz bandwidth, and 25 ns of sample time resolution. At each Rx position, the channel was sounded 5 times and an average PDP was estimated. Figures 11 a)-d) show the estimation of normalized PDP at each measuring points.

A dominant multipath component, corresponding to the Line-of-Sight signal path, is observed in each of the estimated PDPs. Moreover, several paths are noticeably delayed and attenuated with respect to the dominant component and can be related to signal reflections off the building's walls, floor, or ceiling.

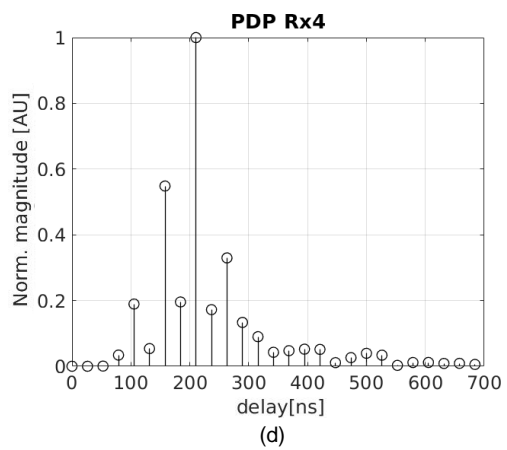
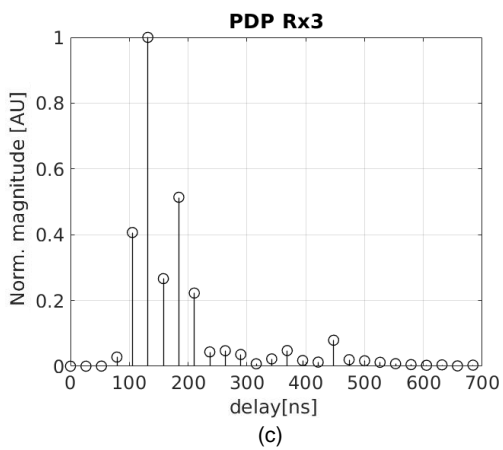
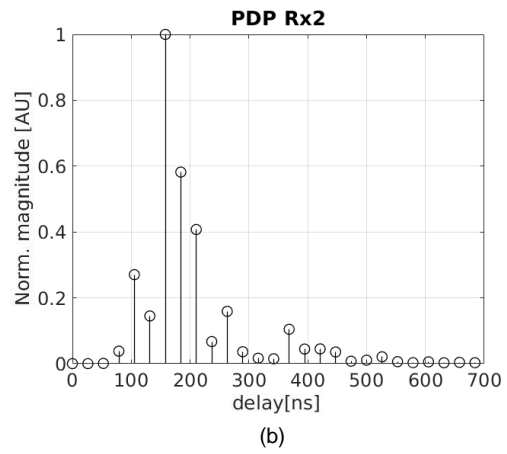
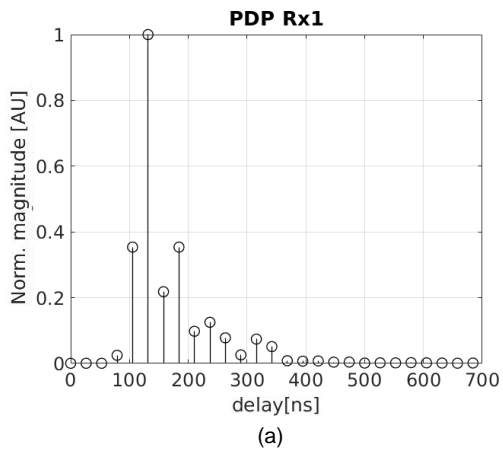


Figure 11: Estimation of normalized Power Delay Profile at measurement points: a) Rx1 b) Rx2 c) Rx3 d) Rx4.

The parameters t_{mean} , t_{rms} and t_{max} were computed for each of the estimated PDPs. Furthermore, coherence bandwidth $B_{50\%}$ was estimated for the correlation value of 50%. The results are presented in Table 1, along with the separation distance.

Measurement point	Tx-Rx distance [m]	t_{mean} [ns]	t_{rms} [ns]	t_{max} [ns]	$B_{50\%}$ [MHz]
Rx1	15	162.2	57.6	263.2	3 470
Rx2	20	195.9	80.3	447.4	2 491
Rx3	28	175.5	85.5	447.4	2 340
Rx4	36	227.8	90.3	526.3	2 216

Table 1: Multipath parameters estimation

As can be seen, values of delay parameters increase with distance, while the coherence bandwidth decreases simultaneously, deteriorating wireless channel performance. Special attention must be given to t_{rms} , as it is often a crucial parameter when defining overall network link efficiency. A similar increase of distance leading to an increase of t_{rms} has been seen in the literature (Düngen et al., 2019), (Karedal et al., 2007). For the purposes of this article, an estimation model of RMS delay spread propagation, proposed in (Greenstein et al., 1997), will be used, defined as:

$$t_{rms}(d) = T_1 d^p, \quad (26)$$

where d is the distance between antennas, T_1 is the value of t_{rms} at 1m of distance, and p is the exponent of the propagation.

A curve fitting method was used to estimate the parameters of Equation 26. The estimated values are $T_1 = 20.12$ ns and $p = 0.43$. The value of the exponent p is bigger than the reported $p = 0.1$ in (Karedal et al., 2007), where an industrial hall in an NLoS scenario was analyzed on a short Tx-Rx separation (2 m to 16 m). This greater p value might be due to a different type of analyzed environment, but it also shows that in a relatively large, open space, the propagation medium tends to experience a rapid distance-related increase of t_{rms} despite its Line-of-Sight nature. Figure 12 illustrates the RMS delay spread as a function of distance between transceivers and the model approximation in logarithmic scale.

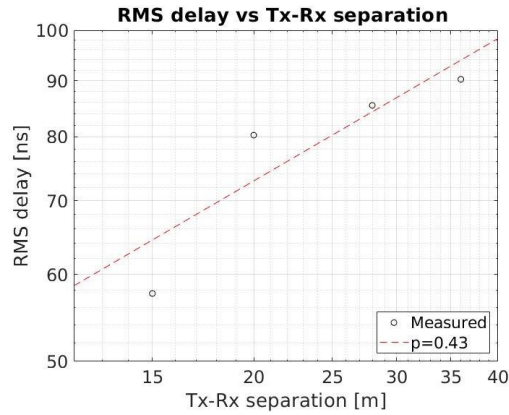


Figure 12: RMS delay spread as a function of Tx-Rx separation in logarithmic scale. Circles represent the measured data and the dashed line is the RMS delay spread propagation model with $p = 0.43$.

The next step in the analysis is to compute a spatially-averaged PDP (sPDP) using the locally estimated PDPs. sPDP is illustrated in Figure 13. Similar to the local profile outlines, a Line-of-Sight path is remarked as the dominant component, and some paths with significant delay differences are perceived as well.

The propagation parameters are estimated once again for sPDP and presented in Table 2. The measured RMS delay spread $t_{rms} = 77.2$ ns is not as high as the results for high-delay environments ($t_{rms} > 150$ ns), but greater than low-delay environment values (<30 ns), both reported in (Coll, 2014). Additionally, the estimated RMS delay is higher than a compared $t_{rms} = 34$ ns, measured for a similar environment in LoS scenario in (Syed & Green, 2019). The factors that may provoke an elevated RMS delay spread value are metallic ceiling and rails. Also, the building lacks materials that could absorb the signal energy, such as wood, paper, or others, provoking a possible RMS delay reduction. Nevertheless, the glass wall could be the element which decreases the final value of the parameter. In addition, the working area, free of metallic machinery or other obstacles in the Line-of-Sight, does not provoke an intensified signal reflection occurrence.

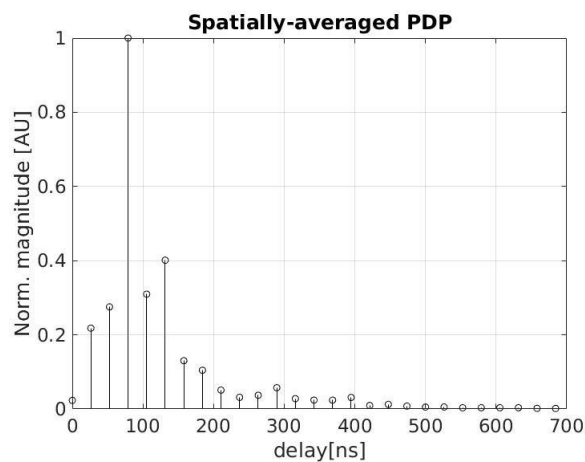


Figure 13: Spatially-averaged Power Delay Profile

Parameter	t_{mean} [ns]	t_{rms} [ns]	t_{max} [ns]	$B_{50\%}$ [MHz]
Value	113.3	77.2	447.4	2 589

Table 2: Spatially-averaged PDP multipath parameters estimation

In the following step, the measured sPDP is compared with the Saleh-Valenzuela indoor channel model (Saleh & Valenzuela, 1987) chosen because it is referenced in a wide range of studies. This simulation model separates multipath rays into groups called clusters. The power of arriving clusters decays exponentially, as do rays inside the cluster (Fig. 14).

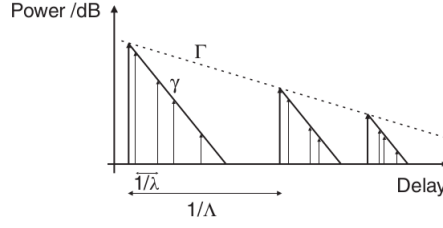


Figure 14: Saleh-Valenzuela Indoor Channel Model illustration. (Molisch, 2011)

The Saleh-Valenzuela model defines the impulse response $h(t)$ as follows:

$$h(t) = \sum_{l=0}^{L-1} \sum_{k=0}^{K-1} \beta_{kl} e^{j\theta_{kl}} \delta(t - T_l - \tau_{kl}), \quad (27)$$

where L and K are maximum number of clusters and multipath components inside clusters, respectively, β_{kl} is the amplitude of k -th ray in l -th cluster, and θ_{kl} is the phase of the kl -th ray. T_l refers to time arrival of the l -th cluster and τ_{kl} is the arrival delay of k -th ray with respect to first ray of the l -th cluster. The amplitude β_{kl} is computed with the following equation:

$$\overline{\beta_{kl}^2} = \overline{\beta^2(0, 0)} e^{-T_l/\Gamma} e^{-\tau_{kl}/\gamma}, \quad (28)$$

where $\beta^2(0, 0)$ is the average power of the first ray received, Γ and γ are exponential decay constants for cluster and rays, respectively. The ray decay constant is reported to increase as a function of time delay $\gamma(\tau)$, thus is modeled as follows:

$$\gamma(\tau) = \gamma_0 + a\tau, \quad (29)$$

where γ_0 is the ray power decay constant of the first cluster and a is the function slope. The arrival times of clusters T_l and rays τ_{kl} are Poisson-distributed, with different interval time constants. Λ parameter indicates an average cluster arrival ratio (average number of clusters per nanosecond), and λ represents the same ratio for rays inside the cluster (Karedal et al., 2007).

To compute the model parameters, the sPDP samples are divided into clusters by the visual decision of the author. Decay factors are estimated with the help of the curve-fitting method. The parameters of the S-V model used for the simulation are listed in Table 3.

Parameter	Λ [1/ns]	λ [1/ns]	Γ [ns]	γ_0 [ns]	a [A.U.]
Value	0.0095	0.0358	69.2	23.65	0.062

Table 3: Used parameters of Saleh-Valenzuela model for simulation process.

Finally, the RMS delay spread of the simulated PDP is calculated and compared to the measured value, which is presented in Table 4. Figure 15 shows the comparison between the measured sPDP of the building and the simulation. As can be noted, the measured and the simulated t_{rms} value is within the same range, and the outline of model approximation PDP matches the estimated value.

Data	t_{rms} [ns]
Measured	77.2
S-V model	59.2

Table 4: Comparison of the RMS delay spread of the measured and the simulated data.

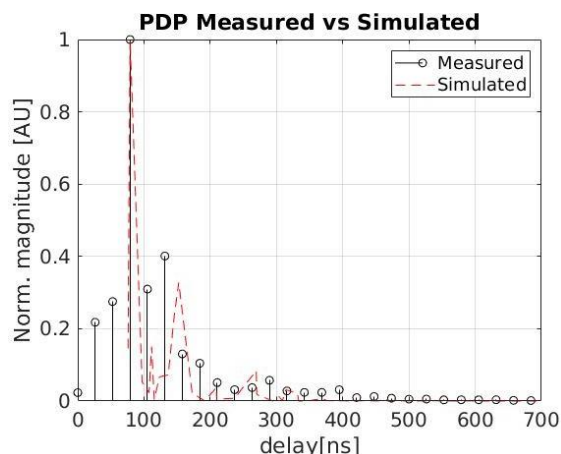


Figure 15: Saleh-Valenzuela model simulation and a comparison to the measured sPDP of the building.

6 System performance discussion

Table 5 shows a comparison of the designed system with other similar projects. The parameters compared are the software used in each project, the sampling resolution, the mobility, and the approximate system cost for a 2-node (Tx and Rx) sounder. The system cost approximation does not consider the cost of a PC unit.

Reference	Software	Samp. res.	Mobility	Approx. cost
(Maas et al., 2012)	GNU Radio	90 ns	medium	\$975 USD
(Boeglen et al., 2017)	GNU Radio	25 ns	high	\$12,000 USD
(Fliedner et al., 2018)	GNU Radio	10 ns	medium	\$25,000 USD
(Jamison & Frolik, 2018)	GNU Radio	50 ns	high	\$600 USD
(Samayoa et al., 2018)	not specified	50 ns	high	\$3,000 USD
(Hosseini & Matolak, 2018)	GNU Radio	20 ns	high	\$4,000 USD
(Wassie et al., 2019)	not specified	25 ns	low	\$50,000 USD
Ref. VNA (Coll, 2014)	-	2 ns	low	\$20,000 USD
The proposed system	Python & bladeRF-cli	25 ns	high	\$1,250 USD

Table 5: Comparison of SDR-based channel sounders with the proposed system, including a VNA sounder as a reference point.

As can be seen, the achieved sampling resolution of the proposed system - 25 ns - can be considered a satisfactory value, as it is better than the values reported in (Maas et al., 2012), (Jamison & Frolik, 2018), (Samayoa et al., 2018) and equal to the sampling resolution of a more expensive high-end system presented in (Wassie et al., 2019). However, (Fliedner et al., 2018) and (Hosseini & Matolak, 2018) show that there is still room for improvement; the sampling resolution is not as high as in VNAs (<2 ns), but it is sufficient for a good multipath effect analysis.

Several factors could contribute to such a good sampling resolution. First, the use of a USB 3.0 interface between the SDR device and the controller unit eliminates "bottlenecks" of other interfaces, i.e., the previous USB generation (USB 2.0), widely implemented in low-cost SDR devices. Second, the cross-correlation sounding method used does not require modulation techniques as other sounding routines do, which allows taking advantage of the full sampling rate provided by the SDR equipment without limiting controller unit performance. Hence, the system implementation uses only basic bladeRF-cli commands and Python libraries, which also speeds up the signal processing part. The GNU Radio, used in almost every similar project mentioned above, is a useful tool, but it consumes many controller unit processing resources and slows down the achievable system sampling rate.

Most of the aforementioned projects tend to have high sounder mobility. This is due to the nature of the small and integrated SDR boards. Additionally, mobility may be enhanced by an embedded system implementation, as presented in (Boeglen et al., 2017). The proposed sounder also follows this trend with reasonably high-mobility due to the use of Raspberry Pi 4 for the transmission part of the system. This single-board computer is a small, cheap, and powerful solution for the controlling unit.

Finally, a comparison of the approximate system costs is carried out. It can be seen that the use of SDR equipment does not necessarily mean a low-cost sounder. The budget for this project was approximately \$1,250 USD, which includes two bladeRF x40 boards, two GPS DOs, a signal amplifier, and a Raspberry Pi 4 - significantly cheaper than those presented in (Wassie et al., 2019), (Boeglen et al., 2017), (Fliedner et al., 2018). The proposed sounder is within the budget range defined by those projects.

The analysis of the proposed channel sounding system performance and the comparison with similar projects will define future work. First, a change of the frequency operation range is required to perform a propagation analysis in 5.8 GHz transmission band. This will be accomplished by implementing the newer version of the bladeRF board (BladeRF 2.0) or by changing the RF front end. Second, a possible new approach to the subject would be an increased system sampling rate, as this would permit a 5G wireless channel performance analysis. It is hoped to solve this using an array of multiple synchronized SDR devices performing a shifted sampling. Another solution would be the use of an additional FPGA to act as the controller unit, replacing the limited USB interface. The authors look forward to enlarging the number of possible sounder applications by implementing MIMO propagation techniques. Lastly, a future system reconfiguration is considered to perform a real-time sounding, which could be used in cognitive radio technology to continuously sound and analyze channel performance. However, this would require a powerful processing unit, such as an FPGA or a DSP.

7 Conclusions

In this article, two achievements were demonstrated. First, the design process and the complete implementation of a channel sounding system were presented. Second, the system was used to perform a measurement campaign, whose objective was to characterize an industrial environment wireless channel.

This article offers a detailed mathematical model, a hardware implementation and a solution to the synchronization problem. The developed sounder software is available in an open repository to facilitate access to a channel sounding tool to a larger group of researchers.

The proposed wireless channel sounder system was validated with a popular Saleh-Valenzuela indoor channel simulation model and both the estimated PDP and the RMS delay spread were within the same value range, demonstrating that an SDR equipment allows a reasonable multipath effect analysis and can successfully substitute VNA devices for the sounding purpose.

The analyzed industrial environment can be identified as an intermediate delay propagation medium. The measured RMS delay spread (77.2 ns) is higher than in a high-delay environment (approx. >150 ns) and lower than in low-delay environments (approx. <30 ns). However, the estimated value is surprisingly higher than that expected for a LoS scenario. Such an increased value is due to the intensified signal reflections caused by the open hall's dimensions and building materials.

The multipath propagation effect was measured at various positions and antenna separation lengths. The collected data analysis confirms that there is an increase in the RMS delay spread as a function of distance. The delay value follows Equation 26 with parameters $T_1 = 20.12$ ns and $p = 0.43$. A comparison of the p exponent with other studies shows that this parameter is closely related to the type of environment being analyzed. Its value is influenced by building dimensions, site equipment, and machinery, and LoS/NLoS type of propagation. The analyzed workshop experienced a fast RMS delay increase.

System performance was compared to other SDR-based projects, demonstrating that the proposed system has achieved a good sampling rate (reachable for such systems) and that the proposed sounder follows the high mobility trend. In addition, the goal for a low-cost system has also been accomplished, with an approximate cost of \$1,250 USD. The proposed mathematical model and solutions to SDR limitations facilitate system reconfiguration. The flexibility of this sounder encourages seeking more applications and performing more wireless channel measurements.

References

- Al-Samman, A., Rahman, T., Hadri, M., Khan, I., & Chua, T. (2017). Experimental uwb indoor channel characterization in stationary and mobility scheme. *Measurement*, 111, 333–339.
- Boeglen, H., Traore, A., Peinado, M. M., Lefort, R., & Vauzelle, R. (2017). An sdr based channel sounding technique for embedded systems. In *2017 11th european conference on antennas and propagation (eucaap)* (p. 3286-3290).
- Cheffena, M. (2014, September). Industrial indoor multipath propagation — a physical-statistical approach. In *IEEE 25th annual international symposium on personal, indoor, and mobile radio communication (pimrc)*. Washington, DC, USA.
- Cheffena, M. (2016). Propagation channel characteristics of industrial wireless sensor networks. *IEEE Antennas and Propagation Magazine*, 58(1), 66-73.
- Coll, J. (2014). *Channel characterization and wireless communication performance in industrial environments* (Doctoral thesis). Department of Information Technology and Communication, KTH Royal Institute of Technology, Stockholm, Sweden.

- Düngen, M., Hansen, T., Croonenbroeck, R., Kays, R., Holfeld, B., Wieruch, D., & Schulze, H. (2019). Channel measurement campaigns for wireless industrial automation. *Automatisierungstechnik*, 67(1), 7-28.
- Fliedner, N. H., Block, D., & Meier, U. (2018). A software-defined channel sounder for industrial environments with fast time variance. In *2018 15th international symposium on wireless communications systems (iswcs)* (p. 1-6).
- Greenstein, L., Erceg, V., Yeh, Y., & Clark, M. (1997). A new path-gain/delay-spread propagation model for digital cellular channels. *IEEE Transactions on Vehicular Technology*, 46(8), 477-485.
- Grzybowski, M. J., Staniec, K., & Erlebach, K. (2006). Propagation modelling. In M. Nawrocki, M. Dohler, & A. Aghvami (Eds.), *Understanding umts radio network modelling, planning and automated optimisation. theory and practice* (p. 67-115). London, England: John Wiley Sons Ltd.
- Hayes, M. H. (1996). *Statistical digital signal processing and modeling*. New York, USA: John Wiley Sons Ltd.
- Hosseini, N., & Matolak, D. W. (2018). Wide band channel characterization for low altitude unmanned aerial system communication using software defined radios. In *2018 integrated communications, navigation, surveillance conference (icns)* (p. 2C2-1-2C2-9).
- Jamison, J., & Frolik, J. (2018). A software-defined radio approach to multi-link channel characterization. In *2018 IEEE 19th wireless and microwave technology conference (wamicon)* (p. 1-4).
- Jeruchim, M. C., Balaban, P., & Shanmugan, K. S. (2002). *Simulation of communication systems. modeling, methodology, and techniques* (2nd ed.). New York, USA: Kluwer Academic Publishers.
- Karedal, J., Wyne, S., Almers, P., Tufvesson, F., & Molisch, A. (2007). A measurement-based statistical model for industrial ultra-wideband channels. *IEEE Transactions on Wireless Communications*, 6(8), 3028-3037.
- LeoBodnar. (2015). Mini precision gps reference clock [Computer software manual]. Northamptonshire, United Kingdom. Retrieved from <http://www.leobodnar.com/>
- Luo, S., Polu, N., Chen, Z., & Slipp, J. (2011). Rf channel modeling of a wsn testbed for industrial environment. In *2011 IEEE radio and wireless symposium* (p. 87-93). Phoenix, USA.
- Maas, D., Firooz, M. H., Zhang, J., Patwari, N., & Kasera, S. K. (2012). Channel sounding for the masses: Low complexity gnu 802.11b channel impulse response estimation. *IEEE Transactions on Wireless Communications*, 11(1), 1-8.
- MacLeod, H., Loadman, C., & Chen, Z. (2005). Experimental studies of the 2.4-ghz ism wireless indoor channel. In *3rd annual communication networks and services research conference (cnsr'05)*. Halifax, Canada.
- Miaoudakis, G. K. A., Lekkas, A., & Koubias, S. (2005). Radio channel characterization in industrial environments and spread spectrum modem performance. In *Proc. of IEEE ETFA* (p. 87-93). Catania, Italy.
- Molisch, A. F. (2011). *Wireless communications* (2nd ed.). London, England: John Wiley Sons Ltd.
- Nuand. (2014). Bladerf x40 product brief [Computer software manual]. New York, USA. Retrieved from <https://www.nuand.com/bladeRF-brief.pdf>
- Proakis, J. G., & Manolakis, D. G. (2006). *Digital signal processing* (4th ed.). Madrid, Spain: Pearson Prentice Hall.
- Rappaport, T. S. (2002). *Wireless communications. principles and practice* (2nd ed.). New York, USA: Prentice Hall.

- Saleh, A. A. M., & Valenzuela, R. (1987). A statistical model for indoor multipath propagation. *IEEE Transactions on Vehicular Technology*, 5(2), 128-137.
- Samayoa, Y., Kock, M., Blume, H., & Ostermann, J. (2018). Low-cost channel sounder design based on software-defined radio and ofdm. In *2018 IEEE 88th Vehicular Technology Conference (VTC-Fall)* (p. 1-5).
- Sharif, Z., & Sha'ameri, A. Z. (2007, December). The application of cross correlation technique for estimating impulse response and frequency response of wireless communication channel. In *The 5 student conference on research and development –scored 2007*.
- Stuber, G. L. (2017). *Principles of mobile communication* (4th ed.). Atlanta, USA: Springer International Publishing.
- Syed, N. A. A., & Green, P. J. (2019). Wideband communication channel sounding for wireless industrial internet-of-things applications. In *2019 IEEE VTS Asia Pacific Wireless Communications Symposium (APWCS)* (p. 1-5).
- Tse, D., & Viswanath, P. (2004). *Fundamentals of wireless communication*. Berkley, USA: Cambridge University Press.
- Wassie, D. A., Rodriguez, I., Berardinelli, G., Tavares, F. M. L., Sorensen, T. B., & Mogensen, P. (2018, September). Radio propagation analysis of industrial scenarios within the context of ultra-reliable communication. In *Proc. 87th veh. technol. conf. (vtc-spring)* (p. 1-6). Porto, Portugal.
- Wassie, D. A., Rodriguez, I., Berardinelli, G., Tavares, F. M. L., Sørensen, T. B., Hansen, T. L., & Mogensen, P. (2019). An agile multi-node multi-antenna wireless channel sounding system. *IEEE Access*, 7, 17503-17516.



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

Desarrollo y tendencias de la ingeniería biomédica en México

José de Jesús Agustín Flores Cuautle¹

Citlalli Jessica Trujillo Romero^{2*}

Rigoberto Martínez Méndez³

Blanca Estela González Sánchez⁴

Ingrid Lizette Sánchez Carmona⁴

¹ CONACYT-Tecnológico Nacional de México, I. T. Orizaba

²División de investigación en Ingeniería Médica, Instituto Nacional de Rehabilitación LGII., México

³Universidad Autónoma del Estado de México, Facultad de Ingeniería. Cerro de Coatepec s/n, Ciudad Universitaria

⁴ Tecnológico Nacional de México, I. T. Orizaba

*Corresponding author: yoda34@hotmail.com

“Desarrollo y tendencias de la ingeniería biomédica en México”

Resumen

La ingeniería biomédica una ciencia interdisciplinaria, a la cual se incorporan cada vez más estudiantes por medio de las diferentes instituciones que ofrecen programas de estudio relacionados con esta disciplina, con la finalidad de entender el rumbo que actualmente está tomando esta disciplina, es necesario conocer los principales programas de estudio que ofrecen las diversas instituciones, de la misma forma, finalmente se presenta una reflexión sobre los temas de actualidad relacionados con esta disciplina.

Ingeniería Biomédica

La Ingeniería Biomédica es un área del conocimiento multidisciplinaria, de acuerdo con el *Imperial College London* y a la *IEEE Engineering in Medicine & Biology Society* ("Aims and scope," 2020; "Definition of Biomedical Engineering," 2020), la ingeniería biomédica es una disciplina que se enfoca al estudio de la evolución del conocimiento de la ingeniería, la biología y la medicina para hacer del mundo un lugar más saludable. Dicho objetivo se logra mediante actividades interdisciplinarias entre las cuales destacan las dos siguientes:

- La adquisición de conocimiento nuevo y entendimiento de los sistemas vivos mediante el uso innovador de técnicas experimentales y analíticas basadas en las ciencias de la ingeniería.
- El desarrollo de nuevos dispositivos, algoritmos, procesos y sistemas que promuevan la biología y la medicina y que a su vez mejoren la práctica médica y la prestación de los servicios de salud.

Para lograr los objetivos antes descritos se debe incluir no sólo la aplicación de los avances tecnológicos de la ingeniería al área médica, sino también a las ciencias básicas de la vida. Los retos de un ingeniero biomédico son utilizar su experiencia y conocimientos en la ingeniería, la medicina, la biología, la física, las matemáticas y las comunicaciones para tratar de resolver los problemas de salud que se presentan en el mundo. La Figura 1 muestra la interacción de las diferentes disciplinas de las que se sirve un Ingeniero biomédico para realizar su labor. Un ingeniero biomédico trabaja en una intersección entre la ciencia, la medicina y las matemáticas con la finalidad de resolver problemas médicos y biológicos; así como diseñar, desarrollar y administrar la tecnología en las unidades de salud (Bravo Hernandez, 2020; Voigt & Magjarević, 2014).

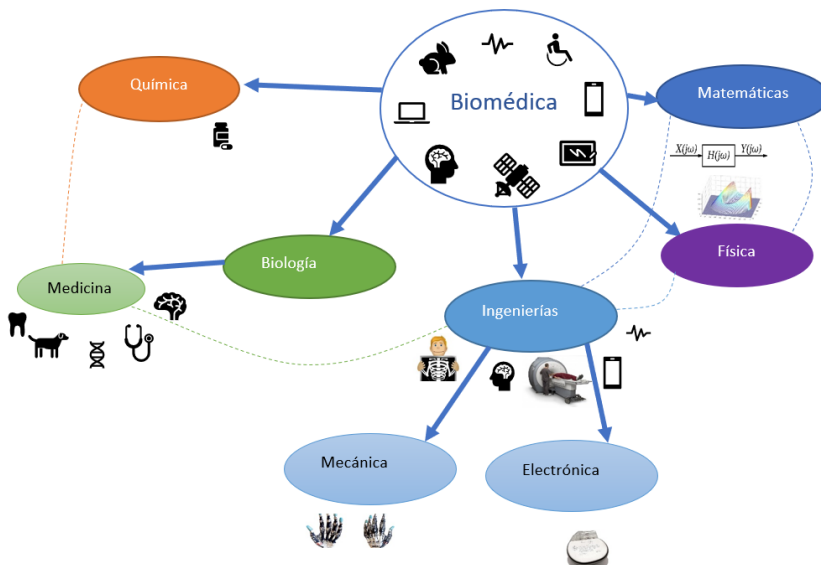


Fig. 1 La ingeniería biomédica y su interacción con otras disciplinas.

Es importante resaltar que debido a la gran variedad de disciplinas que intervienen en la ingeniería biomédica, los ingenieros biomédicos pueden y aplican conocimientos de electricidad, electrónica, mecánica, química y todos aquellos relacionados con la ingeniería. Por lo tanto, no sólo aquellos con un título de ingeniería biomédica participan de esta disciplina. De manera general, todo aquel que tenga un título de ingeniería que le haya permitido obtener la experiencia en una o más de las áreas de investigación y aplicación de la ingeniería biomédica puede participar activamente en esta disciplina (Enderle, Bronzino, & Blanchard, 2005).

La ingeniería biomédica en el mundo

Alrededor del mundo, la ingeniería biomédica es vista como una disciplina con un gran potencial y con una predicción de crecimiento importante. De acuerdo con el Wall Street Journal (Harris, 2017), en Estados Unidos se estima que el número de ingenieros biomédicos se incrementó en un 72% respecto al 2010. Por otro lado, estimaciones hechas por el departamento de empleo de los Estados Unidos prevén que la expectativa de crecimiento para esta disciplina es del 23%, con un salario medio de 86,000 dólares americanos al año (Statistics, 2017). No sólo en Estados Unidos se dan este tipo de expectativas, en general las profesiones relacionadas con la salud tienen proyecciones de crecimiento elevadas. Por esta razón, no es de sorprenderse que las diversas instituciones educativas alrededor del mundo ofrezcan entre sus programas de estudio la carrera de ingeniería biomédica, no sólo a nivel licenciatura sino también a nivel posgrado.

Es notable el hecho de que la carrera de ingeniería biomédica es ofrecida no sólo por universidades reconocidas por sus programas en Ingeniería, sino que escuelas tales como la Escuela de Medicina John Hopkins también ofrecen este tipo de ingeniería tanto a nivel licenciatura como a nivel posgrado ("Johns Hopkins Biomedical Engineering Graduate," 2020; "Johns Hopkins Biomedical Engineering Undergraduate," 2020). Diversos estudios dan cuenta del incremento de instituciones académicas que ofrecen en sus planes de estudio programas relacionados con la ingeniería Biomédica. Tan sólo en Estados Unidos se pasó de 102 universidades en el año 2002 a 119 en 2006, en tanto que alrededor del mundo se reportan 10,453 instituciones académicas ofreciendo este tipo de programas ("Biomedical engineers data by country," 2016).

De igual manera, alrededor del mundo existen sociedades civiles dedicadas a la promoción y desarrollo de la ingeniería biomédica, entre las principales podemos mencionar la *Biomedical Engineering Society* ("Biomedical Engineering society," 2020), el *American Institute for Medical and Biological Engineering* ("American Institute for medical and biological engineering," 2020), la *Engineering in Medicine & Biology Society* ("Engineering in medicine and biology society," 2020). Estas asociaciones agrupan cerca de 500,000 miembros alrededor del mundo; además, tienen presencia en más de 160 países. La Tabla 1 muestra algunas de las principales sociedades civiles dedicadas a la promoción de la ingeniería biomédica; así como el número de miembros con los que cuentan.

Tabla 1 Principales sociedades civiles enfocadas a la promoción de la ingeniería biomédica

Nombre de la Sociedad	Número de miembros
Biomedical Engineering Society (BMES) ("Biomedical Engineering society," 2020)	7,000
The American Institute for Medical and Biological Engineering (AIMBE) ("American Institute for medical and biological engineering," 2020)	50,000
Engineering in Medicine and Biology Society (EMBS), ("Engineering in medicine and biology society," 2020)	9,100
The Institute of Electrical and Electronic Engineers (IEEE).("Aims and scope," 2020)	425,000

Breve historia de la Ingeniería biomédica en México

Existen diversas definiciones de ingeniería biomédica, dentro de estas, la mayoría hace uso de palabras y frases tales como: "aplicación de ciencias exactas", "solución de problemas del sector salud", "biomedicina", "bioelectrónica", etc. ("Ingeniería Biomédica," 2016; "Ingeniería Biomédica," 2016; "Universidad de Barcelona," 2016). Con base en lo anterior, es difícil encontrar el momento exacto en el cual surge la ingeniería biomédica en México. Algunos autores hablan del nacimiento de la ingeniería biomédica basados en la creación del Instituto de Salubridad y Enfermedades Tropicales (ISTE) en 1939 (Mendez & Leehan, 2003; Perez-Tamayo, 2004). Sin embargo, la mayoría de quienes han escrito sobre esta disciplina, coinciden en dos momentos claves para la misma:

- 1) Creación del Instituto Nacional de Cardiología, lugar en el cual científicos de la talla de Norbert Wiener, Albert Grass y Arturo Rosenblueth, entre otros, comienzan labores de investigación relacionadas no sólo con la fisiología, sino con el desarrollo de utensilios y equipos que facilitaran sus labores de investigación (Glantz, 2015; Pineda Romero & González Guerrero, 2001; Voigt & Magjarević, 2014).
- 2) El surgimiento del primer programa académico de Maestría en Ciencias en Bioelectrónica en el recién creado CINVESTAV hacia 1974 por el Dr. Joaquín Remolina. Este posgrado nace como una escisión del Departamento de Farmacología y Toxicología (Cadena & Azpiroz, 2003; Muñoz Martínez, 2005)

Hasta ese momento sólo se contaba con centros de investigación y laboratorios y es hasta 1973 cuando la Universidad Iberoamericana crea el primer programa de licenciatura en Ingeniería Biomédica ("Sociedad Mexicana de Ingeniería Biomédica," 2016). Otras universidades como la Universidad Nacional Autónoma de México cuentan con su propia historia acerca de cómo surgen los programas de ingeniería biomédica en las mismas. El Instituto de Investigaciones Biomédicas (1969) es reconocido como el organismo que dirige los destinos de los planes y programas de estudio en ingeniería biomédica y que tiene como antecedente directo el Laboratorio de Estudios Médicos y Biológicos (1941) ("Instituto de Investigaciones Biomedicas," 2016). En 1973 se crea el proyecto académico de licenciatura, Maestría y Doctorado en investigación biomédica básica de la Universidad Autónoma de México (Kaethe, 2016).

Respecto a los esfuerzos fuera del ámbito netamente académico, podemos mencionar la fundación de la Sociedad Mexicana de Ingeniería Biomédica hacia 1978 ("Sociedad Mexicana de Ingeniería Biomédica," 2016), la cual se define como "una Asociación civil no lucrativa con intereses profesionales, académicos y científicos". Al año siguiente se publica La Revista Mexicana de Ingeniería Biomédica con la finalidad de dar difusión a los trabajos realizados en esta área ("Sociedad Mexicana de Ingeniería Biomédica," 2016).

La Ingeniería Biomédica en México

En lo que respecta a el ámbito formativo, la encuesta global de estudios sobre ingeniería biomédica 2016 reporta 13 instituciones en México que ofrecen programas de estudio relacionados con esta disciplina ("Biomedical engineers data by country," 2016). Por otro lado, la Sociedad Mexicana de Ingeniería Biomédica reporta 45 instituciones contando los diferentes campus localizados en 21 estados de la República Mexicana (Educacion, 2020). En la figura 2 se muestran las diferentes áreas en las que los investigadores nacionales obtuvieron su doctorado, separadas por área y por año. En dicha gráfica se aprecia un incremento gradual de investigadores, tanto en el área de la biomédica como en la bioingeniería, incrementándose en forma exponencial en el último año que abarca este estudio.

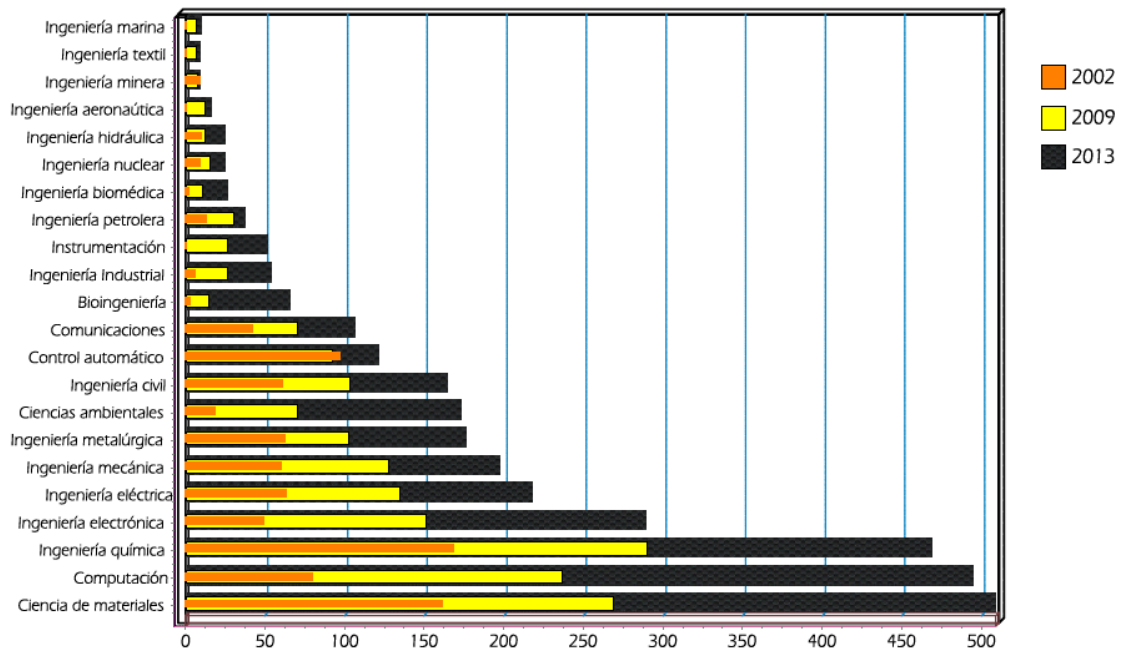


Figura 2. Distribución de las áreas donde los investigadores obtuvieron su doctorado (Perez Angon, 2014)

Programas de estudio de ingeniería biomédica por estados

En lo referente a los estudios de posgrado específicamente en el área de ingeniería biomédica, se deben mencionar instituciones tales como el Instituto Tecnológico de Tijuana, la Universidad Nacional Autónoma de México, la Universidad Autónoma Metropolitana, la Universidad Autónoma de Pachuca y la Universidad Iberoamericana. Sin embargo, si tomamos en cuenta carreras que se encuentran estrechamente relacionadas con la Ingeniería Biomédica, tales como Física Médica, Bioelectrónica, Instrumentación, etc. el número de posgrados ofrecidos en México se incrementa exponencialmente. De acuerdo con la Sociedad Mexicana de Ingeniería Biomédica, actualmente se encuentran registrados en el país más de 50 programas de ingeniería biomédica en México ("Sociedad Mexicana de Ingeniería Biomédica," 2016).

La Tabla 2 muestra algunos de los programas de licenciatura que se ofertan en el país; por otro lado, la Tabla 3 muestra los programas de posgrado. Así mismo, se muestran las áreas del Sistema Nacional de Investigadores (SNI) correspondientes a los diferentes programas de estudio.

Tabla 2 licenciaturas ofrecidas en México relacionadas con la Ingeniería Biomédica

Programa		Institución	Entidad	Área SNI
Licenciatura en ingeniería biotecnología	en	Universidad popular autónoma del estado de Puebla	Puebla	Biología
Licenciatura biotecnología	en	Benemérita universidad autónoma de Puebla	Puebla	Biología, ingeniería y tecnología
Licenciatura biotecnología	en	Universidad autónoma de Aguascalientes	Aguascalientes	Química Biología
Licenciatura ingeniería biomédica	en	Universidad La Salle de México	México	
Licenciatura ingeniería biomédica	en	Universidad de Guanajuato	Guanajuato	Medicina
Licenciatura ingeniería biomédica	en	Universidad Autónoma Metropolitana	CDMX	Medicina
Ingeniería biomédica		Universidad de Guadalajara	Guadalajara	Ciencias exactas e ingenierías
Ingeniería biomédica		Universidad Politécnica de Pachuca	Hidalgo	Ciencias exactas e ingenierías
Licenciatura en investigación biomédica	en	UNAM	México	
Ingeniería biomédica		Universidad iberoamericana	México	
Licenciatura en biomedicina	en	BUAP	Puebla	Ciencias naturales y de la salud
Ingeniería bioelectrónica		Universidad panamericana	Aguascalientes	
Ingeniería en biomédica		Instituto politécnico nacional	CDMX	Medicina
Ingeniería biomédica		Universidad Modelo	Yucatán	

Tabla 3 Programas de posgrado ofrecidos en México relacionadas con la Ingeniería Biomédica

Programa	Institución	Entidad	Área SNI
Maestría En Ciencias En Biomedicina	Universidad Autónoma De Querétaro	Querétaro	Medicina Y Ciencias De La Salud
Maestría En Ciencias Biomédicas	Universidad Autónoma De Sinaloa	Sinaloa	Medicina Y Ciencias De La Salud
Maestría En Ciencias Biomédicas Y De La Salud	Universidad Autónoma Del Estado De Hidalgo	Hidalgo	Medicina Y Ciencias De La Salud
Maestría En Ciencias En Biomedicina Molecular	Centro De Investigación Y De Estudios Avanzados Del Instituto Politécnico Nacional	Ciudad De México	Medicina Y Ciencias De La Salud
Maestría En Ingeniería Biomédica	Universidad Autónoma Metropolitana	Ciudad De México	Ingenierías
Maestría En Ciencias Biomédicas	Universidad Juárez Autónoma De Tabasco	Tabasco	Biología Y Química
Maestría En Ciencias En Biomedicina Y Biotecnología Molecular	Instituto Politécnico Nacional	Ciudad De México	Medicina Y Ciencias De La Salud
Maestría En Biomedicina Molecular	Instituto Politécnico Nacional	Ciudad De México	Medicina Y Ciencias De La Salud
Maestría En Ciencias Biomédicas	Instituto Tecnológico Y De Estudios Superiores De Monterrey	Nuevo León	Medicina Y Ciencias De La Salud
Maestría En Ciencias Biomédicas	Universidad Autónoma De Guerrero	Guerrero	Medicina Y Ciencias De La Salud
Maestría En Biotecnología	Universidad Autónoma De Sinaloa	Sinaloa	Biotecnología Y Ciencias Agropecuarias
Maestría En Biotecnología	Universidad Autónoma Del Estado De Morelos	Morelos	Biotecnología Y Ciencias Agropecuarias
Maestría En Biotecnología	Universidad Autónoma Metropolitana	Ciudad De México	Biotecnología Y Ciencias Agropecuarias
Maestría En Biotecnología	Universidad Del Papaloapan	Oaxaca	Biotecnología Y Ciencias Agropecuarias
Maestría En Ciencias En Biotecnología	Universidad Politécnica Del Estado De Morelos	Morelos	Biotecnología Y Ciencias Agropecuarias
Maestría En Biotecnología	Universidad Popular Autónoma Del Estado De Puebla, A.C.	Puebla	Biotecnología Y Ciencias Agropecuarias
Maestría En Biotecnología	Universidad Autónoma De Chihuahua	Chihuahua	Biotecnología Y Ciencias Agropecuarias
Maestría En Ciencias Con Especialidad En Biotecnología	Instituto Tecnológico Y De Estudios Superiores De Monterrey	Nuevo León	Biotecnología Y Ciencias Agropecuarias

Tendencias de los planes de estudio

Con la finalidad de realizar un mejor análisis respecto a las asignaturas más representativas relacionadas con los planes de estudio de la licenciatura de Ingeniería Biomédica, las asignaturas se agruparon en 4 grandes áreas: básica, formativa, investigación y terminación formal. Es importante mencionar que, de acuerdo con los planes de estudio, el nombre de las asignaturas puede diferir entre una y otra institución. Sin embargo, en esencia las materias mencionadas tratan los mismos tópicos.

Tabla 4 Asignaturas representativas de la carrera de Ingeniería Biomédica Elaboración propia Con datos de ("Sociedad Mexicana de Ingeniería Biomédica," 2016)

INSTITUCION	UNIV. POLI. DE CHIAPAS	IP N	U N	UNIV. AUTÓNOMA METROPOLITANA	UNIV. AUTO. EDO. MEX.	UNIV. AUTO. GUADALAJARA	UNIV. POLI. BICENTENARIO
Asignatura							
Matemáticas	X	X	X	X	X	X	X
Biología / Anatomía	X	X	X	X	X	X	X
Instrumentación Biomédica	X		X			X	X
Electrónica Analógica	X				X	X	X
Sensores Y Actuadores // Biomédicos	X						X
Señales Y Sistemas				X		X	
Procesamiento Digital de Señales						X	

Derivado de la Tabla 4 se aprecia que áreas tales como: Rehabilitación, biomateriales y física médica son prácticamente ignoradas en dichos planes de estudio a pesar de ser áreas claves para el desarrollo de la Ingeniería. La Tabla 4 muestra que los planes de estudio no siguen las tendencias mundiales en cuanto a nuevas áreas de investigación en Ingeniería Biomédica. Una búsqueda en una base de datos ("Elsevier Journals," 2020) arroja un promedio de 800,000 artículos en los últimos 5 años, cuando se coloca como palabra de búsqueda la frase "Ingeniería Biomédica", de esta búsqueda términos como biomateriales (30000), rehabilitación (10000), háptica (32000), Robótica (12000) son palabras comunes en esta búsqueda y como se ha mencionado ya, apenas han sido tomas en cuenta en el desarrollo de los planes y programas de estudio.

Una mirada más allá del mundo académico.

En México se creó el Centro Nacional de Excelencia Tecnológica en Salud (CENETEC) cuya misión es cubrir las necesidades de gestión y evaluación de tecnologías lo cual es uno de los principales propósitos de la Ingeniería Biomédica. El CENETEC incorpora dentro de su organigrama una Dirección de Ingeniería Biomédica cuya principal función es “atender los temas relacionados a la planeación y gestión de equipo médico, a través de la coordinación de los esfuerzos sectoriales e intersectoriales y promoviendo el desarrollo de las áreas de Ingeniería Biomédica” (“Centro Nacional de Excelencia Tecnológica en Salud”, 2020). En el año 2008 se propone la creación de los Centros estatales de Ingeniería Biomédica (CEDIB), cuya misión principal es apoyar a la toma de decisiones y el uso racional de los recursos dentro del Sistema Nacional de Salud en México. Sin embargo, a pesar de ser una iniciativa de gran utilidad en el área de la Ingeniería Biomédica este tema cayó en el olvido en tanto que en 2009 el CENETEC pasa a ser colaborador de la organización Panamericana de la Salud.

Retos

Es claro que el número de investigadores en el área de ingeniería biomédica se ha incrementado en los últimos años y esta tendencia continua al alza. Sin embargo, como en muchas otras áreas de investigación, la mayor parte de los profesores se caracteriza por tener una trayectoria netamente académica sin contar con experiencia alguna en el ámbito hospitalario, por ejemplo, que es uno de los sectores primordiales para esta área de estudio. Derivado de lo anterior, es claro que, como en muchas otras disciplinas, falta pasar del sector académico al sector productivo. Por otro lado, se debe tomar en cuenta también la falta de inversión en el sector salud, especialmente en el sector público, con todas las consecuencias que esto conlleva.

Por lo tanto, es necesario integrar a los planes y programas de estudio las líneas de investigación emergente. Esto con el fin de proporcionar una formación más amplia, integral y actual a los estudiantes de este tipo de áreas. De esta forma, el egresado no sólo tendrá los conocimientos básicos para cumplir a cabalidad su labor; sino al mismo tiempo tendrá las posibilidades de familiarizarse inmediatamente con las tecnologías que se están desarrollando en este momento y que serán de uso común en años venideros. Aunado a lo anterior, se hace presente la necesidad de atender los problemas crónico-degenerativos que se derivan del aumento de la edad en la población. Por lo tanto, es necesario atender los padecimientos propios de las personas en edad avanzada. A este respecto, tecnologías como *wearable sensors*, dispositivos móviles; es decir, sistemas embebidos que permiten no sólo el monitoreo continuo de pacientes, sino que al mismo tiempo facilitan el desarrollo de equipos de uso específico con poca instrumentación adicional es cada vez más frecuente. Sobre todo, en el monitoreo de padecimientos crónicos tal como la diabetes y la hipertensión, problemas que afectan de sobremanera a la población mexicana. La tendencia hacia el uso del *3D printing* no sólo como herramienta para construcción de prototipos, sino para la fabricación de férulas, prótesis, e incluso *scaffolds* es un área que debe de fomentarse desde la academia para aprovechar las ventajas de esta tecnología y que debe de ir de la mano con el estudio de materiales. La Ingeniería Biomédica ha presentado grandes avances durante los últimos años; sin embargo, aún es necesario seguir implementando cambios para potenciar la resolución de problemas reales que se tienen en el sector salud de nuestro país.

Bibliografía

- Aims and scope. (2020). Retrieved from <https://ieeexplore.ieee.org/xpl/aboutJournal.jsp?punumber=51>
- American Institute for medical and biological engineering. (2020). Retrieved from <https://aimbe.org/>
- Biomedical Engineering society. (2020). Retrieved from Biomedical Engineering society
- Biomedical engineers data by country. (2016). *Global health observatory data repository*. Retrieved from <http://apps.who.int/gho/data/view.main.BIOENGHW8v>
- Bravo Hernandez, H. (2020). Dispositivos medicos, el rostro visible de la tecnologia de la salud. Retrieved from <https://www.sectei.cdmx.gob.mx/comunicacion/nota/dispositivos-medicos-el-rostro-visible-de-la-tecnologia-de-la-salud?fbclid=IwAR2cdcvtEqGBxoe2hkKmM2kuEdHWmaVHsDMk3VDNRuMJuXh11D8pijtFMg>
- Cadena, M., & Azpiroz, J. (2003). *Overview of the Biomedical Engineering History in Mexico: A Personal Point of View* (Vol. 4).
- Centro Nacional de Excelencia Tecnológica en Salud (2020). Retrieved from Centro Nacional de Excelencia Tecnológica en Salud
- Definition of Biomedical Engineering. (2020). Retrieved from <http://www3.imperial.ac.uk/pls/portallive/docs/1/51182.PDF>
- Educacion, C. d. (2020). IES Totales en Mexico. Retrieved from <http://ies.somib.org.mx/>
- Enderle, J. D., Bronzino, J. D., & Blanchard, S. M. (2005). *Introduction to Biomedical Engineering*: Elsevier Academic Press.
- Engineering in medicine and biology society. (2020). Retrieved from <https://www.embs.org/about-embs/>
- Glantz, R. G. (2015). Entre la experimentación y los modelos abstractos. Breve historia de vida de Arturo Rosenblueth (1900-1970). *Antropología. Boletín Oficial del INAH*, 99(2015).
- Harris, R. (2017). Th breakdown in biomedical reseach. Retrieved from <https://www.wsj.com/articles/the-breakdown-in-biomedical-research-1491576749>
- Ingenieria Biomedica. (2016). Retrieved from <http://www.ingenieriabiomedica.org/#!queesingenieriabiomedica/ch6q>
- Ingeniería Biomédica. (2016). Retrieved from <http://www.ipn.mx/educacionsuperior/Paginas/Ing-Biomedica.aspx>
- Instituto de Investigaciones Biomedicas. (2016). Retrieved from <http://www.biomedicas.unam.mx/acerca-del-instituto/quienes-somos/>
- Johns Hopkins Biomedical Engineering Graduate. (2020). Retrieved from <https://www.bme.jhu.edu/graduate/overview/>
- Johns Hopkins Biomedical Engineering Undergraduate. (2020). Retrieved from <https://www.bme.jhu.edu/undergraduate/objectives-outcomes/>
- Kaethe, W. (2016). La formación de investigadores en el proyecto academico de licenciatura, maestria y doctorado en investigacion biomedica basica. In U. Instituto de Investigaciones Biomédicas (Ed.).
- Mendez, M. C., & Leehan, J. A. (2003, 17-21 Sept. 2003). *Overview of the biomedical engineering history in Mexico: a personal point of view*. Paper presented at the Engineering in Medicine and Biology Society, 2003. Proceedings of the 25th Annual International Conference of the IEEE.

- Muñoz Martínez, E. J. (2005). Biografías de personajes ilustres, Joaquin Remolina Lopez. In G. Viesca Lopez (Ed.), *Ciencia y tecnología en México en el siglo XXI (Vol. IV)*. Mexico: Academia Mexicana de Ciencias.
- Perez-Tamayo, R. (2004). La investigación biomédica en México. *Gaceta Médica de México*, 140(S1), 6.
- Perez Angon, M. A. (2014). *Atlas de la Ciencia Mexicana 2014* Vol. 1.
- Pineda Romero, M. M., & González Guerrero, K. (2001). REFLEXIONES SOBRE BIOÉTICA E INGENIERÍA BIOMÉDICA: INTRÍNSECA COMPLEMENTARIEDAD. *Revista Latinoamericana de Bioética*, 10(2).
- Sociedad Mexicana de Ingeniería Biomédica. (2016). Retrieved from <http://somib.org.mx/>
- Statistics, B. o. L. (2017). Occupational Outlook Handbook, 2016-17 Edition, Biomedical Engineers. Retrieved from <https://www.bls.gov/ooh/architecture-and-engineering/biomedical-engineers.htm>
- Universidad de Barcelona. (2016). Retrieved from http://www.ub.edu/web/ub/es/estudis/oferta_formativa/graus/fitxa/l/G1074/index.html
- Voigt, H., & Magjarević, R. (2014). Ingeniería Biomédica en México (Una Visión Sucinta). In H. Voigt & R. Magjarevic (Eds.), *Launching IFMBE into the 21st Century: 50 Years and Counting* (pp. 201-203). Berlin, Heidelberg: Springer Berlin Heidelberg.



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