

Recibido 9 Jul 2015
Aceptado 28 Sep 2015

ReCIBE, Año 4 No. 3, Noviembre 2015

Una Revisión Sistemática en los Marcos de Trabajo de Desarrollo Software en las MiPyMEs Productoras de Software

Antonio A. Aguilera

Facultad de Matemáticas, Universidad Autónoma de Yucatán, México

aaguilet@correo.uady.mx

Grisel B. Ancona

Facultad de Matemáticas, Universidad Autónoma de Yucatán, México

griz.ancona@gmail.com

Edwin J. León

Facultad de Matemáticas, Universidad Autónoma de Yucatán, México

eleon@correo.uady.mx

Juan P. Uacán

Facultad de Matemáticas, Universidad Autónoma de Yucatán, México

juan.ucan@uady.mx

Resumen: Una revisión sistemática es considerada como un método o protocolo que permite sintetizar investigaciones científicas de diversas áreas, la Ingeniería de Software no es la excepción. En este artículo se presenta este protocolo aplicado en la búsqueda de publicaciones relacionadas con las adaptaciones a los marcos de trabajo de desarrollo de software llevadas a cabo en MiPyMEs desarrolladoras de software, en el período comprendido de 1995 al 2013, centrándose en tendencias, países y sectores que publican; así como en los modelos, metodologías, estándares, procesos técnicos, procesos de implementación y procesos de reúso del software. Los resultados obtenidos muestran que los países con mayor número de publicaciones sobre el tópico seleccionado son Estados Unidos y Reino Unido, que en su conjunto suman 21%; y que otros países suman el 79% restante. Se observa que el sector con mayor número de publicaciones es el sector educativo con 76%, seguido por los centros de investigación con 18% y el sector privado con el 6%. El modelo de procesos y la metodología más utilizados son CMMi con 33% e IDEAL con 22%, respectivamente. Los estándares más utilizados, con el 18% cada uno, son el ISO/IEC 15504 y CORBA; y los procesos del ciclo de vida del software mayormente abordados son: análisis de requisitos del sistema con un 33% y construcción con un 29%.

Palabras Clave: MiPyME, revisión sistemática, modelos, metodologías, estándares de desarrollo de software, procesos técnicos, procesos de implementación, procesos de reúso del software.

A Systematic Review in Software Development Frameworks for MSMEs of Software

Abstract: A systematic review is considered a protocol applied in several scientific areas, and the application in Software Engineering is not the exception. This article presents the applied and adapted protocol in searches related to software development frameworks for MSMEs of software; the data cover the period from 1995 to 2013, focusing on trends, countries and publishing sectors;

as well as models, methodologies, standards, technical processes, implementation processes and software reuse processes. Results for countries publishing on the selected topic are as follows: United States and United Kingdom show the highest number of publications together, 21%, while other countries keep the rest 79%. The greatest number of publications is presented in the education sector with 76%, followed by research centers 18%, and the private sector with 6%. The most used process model and methodology are the CMMI and the IDEAL with 33% and 22%, respectively. The most widely used standards, with 18% each, are the ISO/IEC 15504 and CORBA; and the most addressed software life cycle process are the analysis of system requirements with 33% and the construction with 29% respectively.

Keywords: MSMEs, systematic review, models, methodologies, software development standards, technical processes, implementation processes, software reuse processes.

1. Introducción

En la actualidad se han desarrollado modelos, estándares y metodologías como CMMi (CMMI Product Team, 2010), IDEAL (McFeeley, 1996), ISO/IEC 15504 (International Standard Organization, 2004) y CORBA (Mowbray y Zahavi, 1995) que ayudan a las grandes organizaciones a mejorar sus procesos de desarrollo y productos de software. El establecimiento de estos modelos y metodologías, dado los problemas que presentan las MiPyMEs, debiera realizarse de un modo particular y visiblemente diferente a como se hace en organizaciones de mayor tamaño; debido a que no es tan sencillo como el hecho de considerar los programas de mejora como versiones a escala de las grandes compañías (Richardson, 2001), por lo que las MiPyMEs desarrolladoras de software requieren prácticas eficientes de Ingeniería de Software adaptadas a su tamaño y tipo de negocio (Fayad, Laitinen, y Ward, 2000).

En este artículo se presenta una revisión sistemática de la literatura acerca del tópico: adaptaciones en los marcos de trabajo de desarrollo de software en las MiPyMEs, en el período comprendido de 1995 al 2013, no obstante el análisis de los datos se realizó durante el 2014. El propósito de este trabajo es presentar información actualizada sobre las tendencias de este tópico como son: países y sectores que abordan el tema, así como modelos, metodologías, estándares, procesos técnicos, procesos de implementación y procesos de reúso del software, reportados en este tipo de empresas; debido a la diversidad de modelos, metodologías y estándares, se realizó una estrategia adicional para identificar a sus autores, los cuales son reportados en éste artículo. Esta revisión sistemática se centra en aportaciones reportadas sobre procesos técnicos, procesos de implementación y procesos de reúso, del ciclo de vida del software. Otras revisiones similares, como las realizadas por: Gómez, Aguilera, Ancona, y Gómez (2014) se enfoca en aportaciones reportadas sobre procesos de soporte del área de calidad del ciclo de vida del software; Pino, García, y Piattini (2006) que se enfoca en las aportaciones que se han realizado en los procesos del ciclo de vida del software; Dybå y Dingsøyr (2008) que se enfoca en los beneficios y limitaciones de las metodologías ágiles; Wangenheim, Rossa, y Salviano (2010) orientada a la revisión de modelos de capacidad y madurez del proceso de software; o Duron, Muñoz, y Mejía (2013) enfocada en conocer cómo se realiza la implementación de la mejora de procesos software.

Este trabajo está organizado de la siguiente manera: La sección 2 presenta la metodología. En la sección 3 se describe la aplicación del protocolo de revisiones sistemáticas, al dominio seleccionado. En la sección 4 se describe la extracción de la información. La sección 5 presenta el resumen de resultados. Finalmente la sección 6 muestra las conclusiones de este trabajo

2. Metodología

El término “Revisión Sistemática” (RS) es usado para referirse a una metodología específica de investigación, desarrollada para obtener y evaluar la evidencia disponible perteneciente a un tema específico (Kitchenham et al., 2001). RS es el proceso de resumir información existente sobre un fenómeno de forma minuciosa y empírica. Al final, la RS obtiene como resultado una conclusión general sobre los estudios individuales del fenómeno en cuestión. En este apartado se describe de manera general, en 3 pasos, el proceso de la revisión sistemática para el área de Ingeniería de Software, propuesto por Biolchini, Gomes, Cruz, y Orta (2005).

Paso 1. Desarrollo del protocolo (fase rigurosa e iterativa): cubre el plan general para la revisión sistemática de la literatura.

Identificación y selección de fuentes: sección cuyo objetivo es identificar y seleccionar las fuentes donde se llevará a cabo la búsqueda de los estudios primarios. Se compone de los siguientes apartados: definición de los criterios de la selección de fuentes, idioma de los estudios, identificación de fuentes, selección de fuentes después de la evaluación, verificación de las fuentes.

Selección de estudios: sección que tiene por objetivo identificar los estudios primarios y secundarios. Los estudios se seleccionan después de la aplicación de los criterios de inclusión y exclusión.

Paso 2. Extracción de información: fase en la que se ejecuta la búsqueda de estudios en las fuentes definidas; los estudios encontrados se evalúan con los criterios definidos.

Definición de los criterios de inclusión de información (CInf) y exclusión de información (CEInf): que tiene por objetivo definir los criterios con que será evaluada la información.

Plantillas para la extracción de información: sección que tiene por objetivo registrar los estudios primarios derivados del proceso de selección.

Ejecución de la extracción: sección en la que se realiza la evaluación de los estudios mediante los criterios definidos.

Paso 3. Resumen de resultados: fase en la que se muestran los resultados obtenidos de la aplicación de la fase de desarrollo del protocolo de RS y la fase de extracción de información.

Tendencia de los estudios: sección que muestra cómo han evolucionado los estudios primarios a lo largo del tiempo.

Clasificación: sección que muestra los resultados encontrados en los estudios primarios al aplicar los criterios de inclusión y exclusión de la información.

1. Desarrollo del protocolo

En esta sección se describen los pasos realizados para el desarrollo del protocolo para revisiones sistemáticas.

Formulación de la pregunta:

Enfoque de la pregunta: esta RS se lleva a cabo para identificar iniciativas o informes de experiencias en modelos, metodologías, estándares, procesos técnicos, procesos de implementación y procesos de reúso del software en las MiPyMEs, así como para dar a conocer las tendencias, países, y sectores con mayor número de publicaciones en este ámbito.

Amplitud y calidad de la Pregunta: esta sección ayuda a definir la sintaxis de la pregunta de investigación, el contexto en el cual se aplica la revisión y la pregunta de investigación a ser contestada, así como sus especificaciones semánticas (o rango de la pregunta).

Problema: un marco de trabajo establece la base para un proceso de software completo al identificar un número pequeño de actividades del marco de trabajo aplicables a todos los proyectos de software, sin importar su tamaño o complejidad (Pressman, 2006). Los procesos técnicos son utilizados para definir los requisitos de un sistema, transformar los requisitos en un producto efectivo, permitir una reproducción consistente del producto donde sea necesario, usar el producto, proveer el servicio requerido, sostener la prestación de los servicios y disponer del producto cuando es retirado del servicio. Los procesos de implementación del software son utilizados para producir un elemento (unidades de software) del sistema implementado en el software. Esos procesos transforman el comportamiento específico, las interfaces y las restricciones de implementación en acciones de implementación, dando como resultado un elemento del sistema que satisface los requisitos derivados de los requisitos del sistema. Los procesos de reuso del software soportan la habilidad de una organización de reusar elementos de software a través de los límites del proyecto. Esos procesos son únicos porque, por naturaleza, operan fuera de los límites de cualquier proyecto particular (ISO/IEC/IEEE, 2008). En la actualidad no existen estudios recientes que muestren las tendencias de las iniciativas del tópico seleccionado, por lo que creemos que es de interés dar a conocer qué países y en qué sectores se llevan a cabo los estudios, así como los modelos, metodologías, estándares, procesos técnicos, procesos de implementación y procesos de reuso del software, utilizados en las MiPyMEs. A continuación se describe la pregunta de la RS, la intervención, el efecto, los resultados de las mediciones, la población y la aplicación.

Pregunta: ¿Qué modelos, métodos, estándares, procesos técnicos, procesos de implementación y procesos de reuso del software se abordan en las adaptaciones de los marcos de trabajo de desarrollo de software para las MiPyMEs desarrolladoras de software?.

Intervención: marcos de trabajo para desarrollo de software en MiPyMEs desarrolladoras de software.

Efecto: iniciativas que tratan modelos, métodos, estándares, procesos técnicos, procesos de implementación y procesos de reúso del software para MiPyMEs desarrolladoras de software.

Resultados de las mediciones: número de iniciativas identificadas.

Población: publicaciones relacionadas con modelos, métodos, estándares, procesos técnicos, procesos de implementación y procesos de reúso del software en el contexto de las MiPyMEs.

Aplicación: organizaciones que emplean algún marco de trabajo de desarrollo de software en MiPyMEs desarrolladoras de software.

Identificación y selección de fuentes:

El Idioma de los estudios seleccionado es inglés, al ser el idioma universal.

Las fuentes incluyen las bases de datos digitales ACM, CiteeSeerX, IEEE Xplore, Sciencie Direct, Wiley y SEI, las cuales son comunmente utilizadas para investigaciones en este ámbito de la computación.

Cadenas de búsqueda: palabras reservadas extraídas del conjunto de palabras definidas en la pregunta. Combinando estas palabras clave con los operadores lógicos “AND” y “OR”, se obtuvo la cadena de búsqueda (que se muestra en la Tabla 1). Esta cadena ha sido adaptada a cada buscador de las fuentes de búsqueda.

Palabras clave con operadores lógicos
(“software development framework”) AND (small or “medium” or “small and medium” or “SME’s”) AND (settings or organizations or companies or company or “sized software company” or enterprises or “software enterprises”)

Tabla 1. Cadena de Búsqueda

Selección de fuentes después de la evaluación de criterios: Se verificó, si las fuentes se ajustan a los criterios previamente definidos, y se estableció la lista de fuentes, mostrada en la Tabla 2.

Número	Fuentes
1	ACM
2	CiteeSeerX
3	IEEE Xplore
4	Science Direct
5	Wiley
6	SEI

Tabla 2. Fuentes empleadas

Verificación de las fuentes: Tres investigadores del Grupo de Investigación en Tecnologías para la Formación en Ingeniería de Software, de la Facultad de Matemáticas en la Universidad Autónoma de Yucatán evaluaron el listado de fuentes, de acuerdo con los criterios de selección de fuentes, obtenida en la sección previa, donde de forma consensuada aprobaron el listado.

Selección de estudios:

Este apartado describe el proceso y criterios de la selección y evaluación de los estudios.

Definición de estudios: En la definición de criterios de inclusión (CI) de estudios y de exclusión (CE) de estudios, se identificaron 4 y 2, respectivamente, los cuales se muestran en la Tabla 3.

Criterio	Descripción
CI1	Incluye publicaciones cuyos títulos están relacionados con los marcos de trabajo de desarrollo de software en las MiPyMEs.
CI2	Incluye publicaciones que contengan palabras reservadas que coincidan con las definidas en la cadena de búsqueda.
CI3	Incluye publicaciones cuyo resumen esté relacionado con el tópico seleccionado.
CI4	Incluye publicaciones que han sido leídas parcial o totalmente.
CE1	Excluye publicaciones que no coinciden con los criterios de inclusión previos.
CE2	Excluye todas las publicaciones duplicadas.

Tabla 3. Definición de criterios de inclusión y exclusión de estudios.

Definición de tipos de estudios: se analizaron los estudios relacionados con los marcos de trabajo de desarrollo de software en las MiPyMEs.

Con respecto al procedimiento para seleccionar los estudios, se tomaron como criterios: el título de la publicación, el resumen de cada uno y en su mayoría se requirió revisar el contenido completo.

Ejecución de la selección: En la selección inicial de estudios se realizó la ejecución de las búsquedas, adaptando las cadenas a los motores de cada buscador. La Tabla 4 muestra los resultados obtenidos.

Fuente	Fecha Búsqueda	Resultados	Estudios Primarios
ACM	01/07/2013	141	13
CiteeSeerX	02/07/2013	36	6
IEEE Xplore	01/07/2013	405	5
Science Direct	16/01/2014	151	4
Wiley	03/07/2013	313	3
SEI	05/07/2013	49	2
Total		1095	33

Tabla 4. Número de estudios y resultados de las búsquedas.

Evaluación de la calidad de los estudios: para determinar la calidad de los estudios, se aplicaron los criterios de inclusión y de exclusión, donde se seleccionaron 33 estudios.

2. Extracción de información

En esta sección se describen los pasos realizados para la extracción de información, en los estudios primarios encontrados. Primero se definieron los criterios de inclusión de información (CInf) y exclusión de información (CEInf). En segundo lugar se elaboraron plantillas para el registro de los datos extraídos, por último, se ejecutó la extracción de la información aplicando los CInf, y los CEInf.

Definición de los criterios de inclusión de información (CInf) y exclusión de información (CEInf)

Respecto a este punto, se identificaron cuatro criterios de inclusión de información (CInf) y uno de exclusión (CEInf). La Tabla 5 muestra una descripción de estos criterios.

Criterio	Descripción
CI1inf	Recoger información sobre las tendencias en investigación e implementación respecto a los marcos de trabajo de desarrollo de software en las MiPyMEs.
CI2inf	Recoger información sobre países y sectores que investigan o implementan algún marco de trabajo de desarrollo de software en las MiPyMEs.
CI3inf	Identificar qué modelos, metodologías y estándares abordan los estudios de marcos de trabajo de desarrollo de software en las MiPyMEs.
CI4inf	Identificar qué procesos técnicos, procesos de implementación y procesos de reuso del ciclo de vida del software tratan los estudios.
CE1inf	Excluir la información que no esté relacionada con los criterios de inclusión definidos anteriormente.

Tabla 5. Definición de los criterios de inclusión y exclusión de información.

Plantillas para la Extracción de Información

Para el registro de los datos de identificación del estudio primario, se definieron plantillas que contienen los siguientes campos: identificador (id), año, y título del estudio. Para el registro de los datos a extraer mediante los criterios CIinf y los CEinf, se definieron plantillas con los siguientes campos: resumen, país, institución u organización, metodología, modelo o estándar y proceso del ciclo de vida del software que se aborda. Los datos registrados en estas plantillas se usaron para el apartado de resumen de resultados.

Ejecución de la Extracción

Con una evaluación imparcial de la información de los estudios primarios, los registros obtenidos se colocaron en las plantillas para la extracción de la información. La evaluación se realizó mediante observaciones de las ideas

principales, utilizando los criterios de inclusión y exclusión de la información (véase Tabla 5) y se organizó de acuerdo a los criterios de inclusión.

3. Resumen de resultados

En esta sección se presentan los resultados obtenidos de la aplicación del protocolo de la RS.

Tendencia de los estudios

En el análisis de tendencia de los estudios, se consideró el año del estudio. Como se puede observar en la Figura 1, los estudios muestran una tendencia incremental. Los años donde se tuvieron más estudios sobre el tópico seleccionado fueron 2010 y 2013. Estos resultados sugieren interés del tópico en la comunidad, en aportar al cuerpo de conocimiento de Ingeniería de Software, aspectos relacionados a los marcos de trabajo de desarrollo de software para las MiPyME's. Estos resultados corroboran los resultados publicados por Pino et al. (2006).



Figura 1. Publicaciones del tópico por año.

Clasificación

Los estudios cubren:

Países que han publicado sobre el tópico seleccionado:

En el análisis de los países que publican, se identificó el país del autor principal del estudio. Como se muestra en la Figura 2, el 21 % corresponde a 2 países, que son: Estados Unidos con el 12%, y Reino Unido con el 9%. El 79 % restante corresponde a otros países: África, Alemania, Argentina, Australia, Austria, Canadá, China, Colombia, Corea, España, Finlandia, Grecia, Irlanda, Italia, Kenya, Malasia, México, Portugal, Suecia, Suiza y Tailandia. Estos resultados sugieren que el tópico seleccionado es de interés a nivel mundial, principalmente en países desarrollados y emergentes. Éste interés muy probablemente se deba a la dependencia que hay entre las economías mundiales y las MiPyMEs de este sector, que representan el 99% (Fayad et al., 2000); así como a la existencia de programas de apoyo gubernamentales tales como SEI (Software Engineering Institute en Estados Unidos), SPIRE (Software Process Improvement in Regions of Europe) (European Commission ESPRIT/ESSI Programme, 1993), MR-MPS Modelo de referencia de procesos (Weber et al., 2004), SIMEP-SW (Sistema Integral para la Mejora de los Procesos Software en Colombia) (Hurtado y Bastarrica, 2005), COMPETISOFT (Mejora de Procesos para Fomentar la Competitividad de la Pequeña y Mediana Industria del Software de Iberoamérica) (COMPETISOFT, 2006), MoProSoft (Modelo de Procesos para la Industria de Software en México) (Oktaba et al., 2003), entre otros.

Países con publicaciones sobre los marcos de trabajo de desarrollo software en la MiPyME

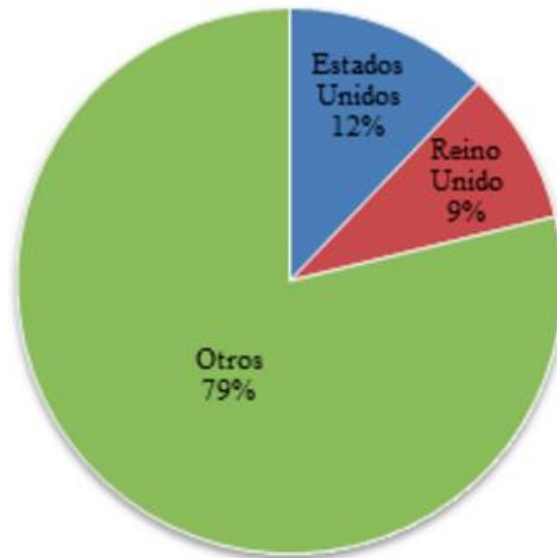


Figura 2. Países con publicaciones sobre el tópico seleccionado.

Resultados por tipos de sectores:

En el análisis del tipo de sectores que publican, se consideró como sector educativo a las organizaciones que se dedican a la educación superior (universidades, academias, escuelas, institutos y facultades). Como centros de investigación, aquellas organizaciones que se dedican a la generación de conocimiento y su aplicación; y finalmente se consideró como sector privado a aquellas organizaciones que tienen como fin un beneficio económico y que no pertenecen al sector educativo, ni a un centro de investigación. En el caso de los centros de investigación, adscritos a una organización educativa, los estudios se catalogaron como producidos por el centro de investigación. Como se aprecia en la Figura 3, el sector que más publica es el sector educativo con un 76%, seguido del centro de investigación con 18% y finalmente el sector privado con 6%. Estos resultados sugieren que las aportaciones se mantienen en el ámbito conceptual y que es escasa la vinculación con el sector privado, lo que se corrobora de acuerdo con los resultados publicados por Vega, Fernández de Lucio, y Huanca (2007).

Sectores que publican sobre sobre los marcos de trabajo de desarrollo software en la MiPyME

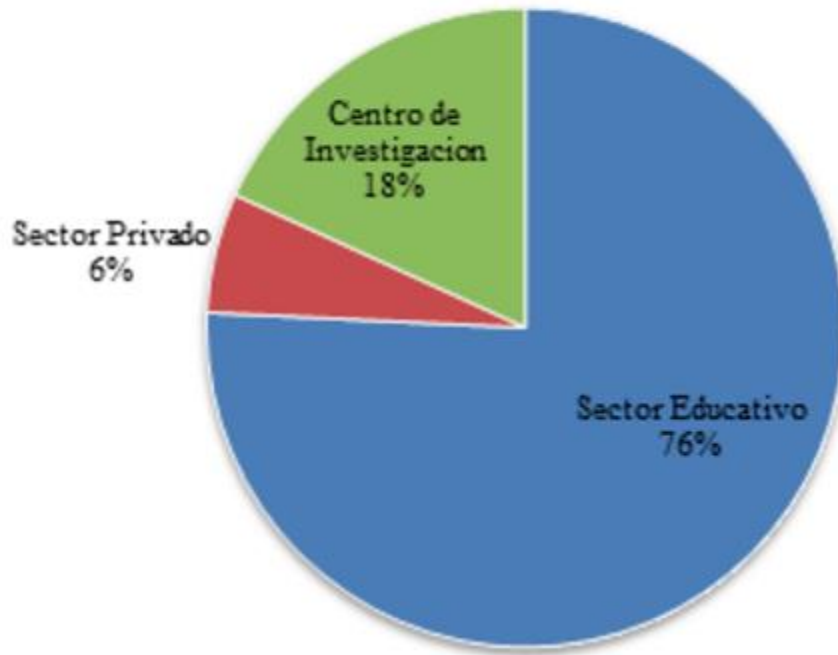


Figura 3. Sectores que publican en el t3pico seleccionado.

Modelos de proceso usados:

En el an3lisis de los modelos usados, se tomaron en consideraci3n aquellos estudios que se basaron, cuando menos, en alguno de los siguientes modelos: CMMi, IDEAL, Cascada (Pressman, 2014), y UCD (Venturi y Troost, 2004). En los casos donde los estudios se apoyaron en m3s de un modelo, 3stos 3ltimos se separaron y se clasificaron en la categor3a correspondiente. Como se observa en la Figura 4, el modelo m3s usado es CMMi con un 33%, seguido por IDEAL con 22%. As3 mismo, se observa que la categor3a Otros (donde se incluyen los modelos CMM, Cascada y UCD) en conjunto suma el 45%. Estos resultados muestran que los modelos tratados en los estudios revisados, aun cuando son adaptaciones que se basan principalmente en modelos para las grandes empresas, el uso de los modelo para la MiPyME, es indudable. Esto sugiere

cierto grado de madurez en el conocimiento, la aceptación y la experimentación de dichos modelos en las MiPyMEs, aunque falta camino por recorrer.

Modelos de procesos usados

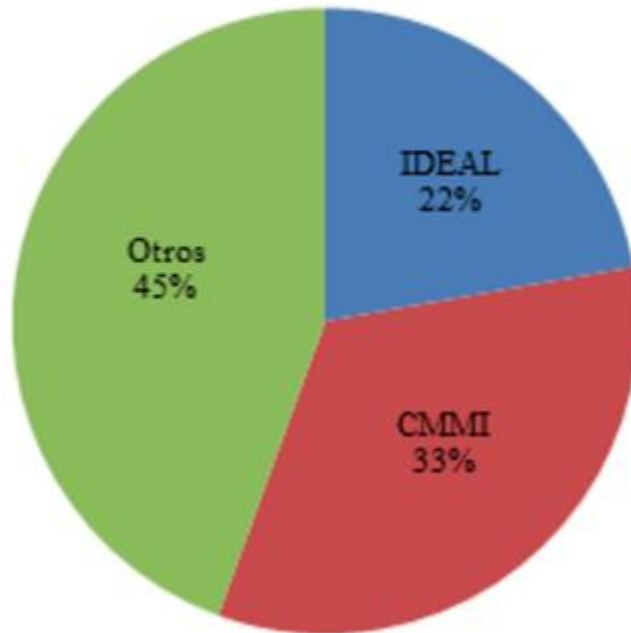


Figura 4. Modelos de procesos usados.

Metodologías consideradas:

En el análisis de las metodologías utilizadas, se consideraron los estudios que trataron, al menos, alguna de las siguientes metodologías: XP (Beck, 2000), COMPETISOFT (COMPETISOFT, 2006), SCRUM (Schwaber, 1995), RUP (Krunchten, 2004), Crystal Orange (Cadavid, Fernández, y Morales, 2013), RE (Manganelli y MKlein, 1995), MESOPYME (Calvo-Manzano, Cuevas, San Feliu Gilabert, Serrano, y García, 2006), V-Modell XT (Weit, 2005), MDA (Kleppe, Warmer, y Bast, 2003), OMT (Lorensen, Rumbaugh, y Blaha, 1991), Automotive OMG (Automotive SIG, 2007), SPICE (Automotive SIG, 2007), SQUARE (Mead y Stehney, 2005), y UCP (Karner, 1993). Si alguna publicación consideró más

de una, éstas fueron separadas y se clasificaron en la categoría correspondiente. Como se muestra en la Figura 5, el 12% de las metodologías se concentran en Xtreme Programing (XP). El 88% restante (representado por la categoría Otras) concentra a las otras metodologías: COMPETISOFT, SCRUM, RUP, Crystal Orange, RE, MESOPYME, V-Modell XT, MDA, OMT, Automotive OMG, SPICE, SQUARE, UCP. Estos resultados sugieren que XP es una las metodologías más usadas, lo que se corrobora de acuerdo con los resultados publicados por Dybå y Dingsøy (2008).



Figura 5. Metodologías consideradas.

Estándares utilizados:

En el análisis de los estándares empleados, se consideraron los estudios que trataron, al menos, algún estándar. Éstos se clasificaron en su categoría correspondiente: ISO/IEC 15504, CORBA, y Otros (ISO 9000 (International Standard Organization, 2010), ISO WD 26262 (Automotive Standards Committee of the German Institute for Standardization, 2005), ISO/IEC 61131-3

(International Electrotechnical Commission, 1993), OMG (OMG, 2015), UX (ISO/IEC 9241 – 2102010) (International Organization for Standardization, 2010), IEC 61508 (International Electrotechnical Commission, 1999), ISO 9241 – 11 (ISO/IEC, 1998)). Como se muestra en la Figura 6, los estándares de mayor uso son ISO/IEC y CORBA, cada uno, con un 18%. Así mismo, se observa que existen otros modelos que en su conjunto suman el 64%: ISO 9000, ISO WD 26262, ISO/IEC 61131-3, OMG, UX (ISO/IEC 9241 – 2102010), IEC 61508, ISO 9241 – 11. Estos resultados sugieren que la comunidad en Ingeniería de Software está apostando mayormente a estándares ISO, motivado probablemente, porque sean conocidos entre la comunidad y no tan complicados en su implementación, además de ofrecer la posibilidad de obtener una certificación internacional (Cater-Steel, Toleman, y Rout, 2005; Mas, Fluxà, y Amengual, 2012; Taramaa, Makarainen, y Ketola, 1995; Valdés, Visconti, y Astudillo, 2011).

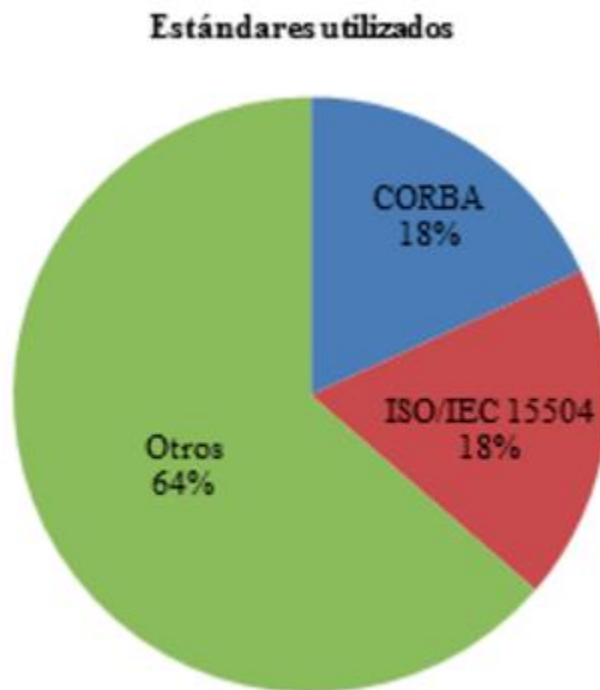


Figura 6. Estándares utilizados.

Procesos técnicos, proceso de implementación y procesos de reuso del software:

En el análisis de los procesos técnicos, procesos de implementación y procesos de reuso, del software, se seleccionaron los estudios que aportaron alguna mejora, al menos, en uno de los procesos mencionados. En los casos donde las publicaciones aportaron alguna mejora en más de un proceso técnico, procesos de implementación y procesos de reuso, del software, éstos últimos se separaron y se clasificaron en las categorías correspondientes (análisis de requisitos del sistema, análisis de los requisitos software, diseño de la arquitectura del sistema, diseño de la arquitectura del software, construcción, gestión del programa de reutilización). Como se observa en la Figura 7, los procesos técnicos, procesos de implementación y procesos de reuso, del software, que reciben contribuciones de los estudios seleccionados son: análisis de requisitos del sistema con un 33%, construcción con un 29%, análisis de los requisitos software con 15%, diseño de la arquitectura del software con 13%, diseño de la arquitectura del sistema con un 8% y gestión del programa de reutilización con un 2%. Algunas razones que quizá está potenciando la investigación en dichos procesos son: especificaciones incompletas, cambios continuos en requerimientos (Quispe, Marques, Silvestre, Ochoa, y Robbes, 2010), falta de formalidad en procesos y metodologías (Alexandre, Renault, y Habra, 2006), en particular informalidad en la descripción de componentes arquitectónicos y en el modelado, así como la carencia de control en los procesos (Hofer, 2002; Pino, Pedreira, Garcia, Rodriguez, y Piattini, 2010), problemas comunes en la MiPyMEs desarrolladoras de software.

Procesos técnicos, procesos de implementación y procesos de reuso del software.

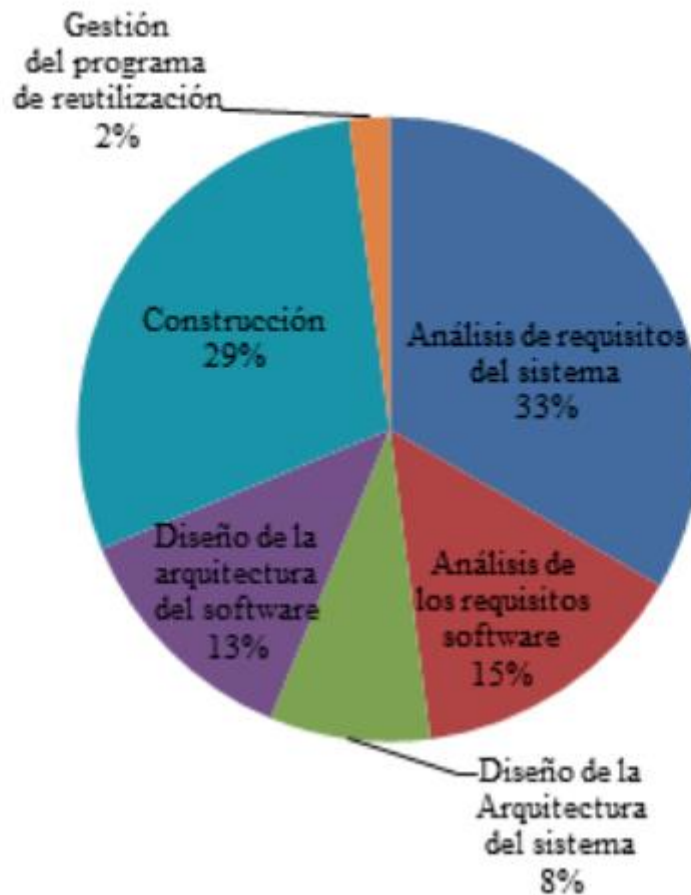


Figura 7. Procesos técnicos, procesos de implementación y procesos de reuso del software.

4. Conclusiones

Las revisiones sistemáticas permiten sintetizar la información existente sobre un fenómeno de forma minuciosa y empírica. Al final, la revisión sistemática obtiene como resultado una conclusión general sobre los estudios individuales del fenómeno en cuestión.

En este trabajo se han presentado los resultados de una revisión sistemática de la literatura sobre las contribuciones a los marcos de trabajo de desarrollo de

software, en el contexto de las MiPyME's desarrolladoras de software, en el periodo comprendido de 1995 a diciembre de 2013.

El interés de la comunidad de Ingeniería de Software, en relación al tópico seleccionado, muestra una tendencia incremental que sugiere continuará. Se encontró que a nivel mundial, principalmente en los países desarrollados y emergentes, la mayoría de estudios se han realizado en el sector educativo y en los centros de investigación, quedando relegado el sector privado. Estos resultados probablemente indiquen que la mayoría de las contribuciones del tema sean a nivel conceptual, quedando aún camino por recorrer referente a las aportaciones del sector privado.

En cuanto a modelos y metodologías desarrolladas para la MiPyME, los modelos preferidos para su adaptación son: CMMi e IDEAL. Por otra parte, la metodología más usada es Extreme Programming. Respecto a los estándares preferidos para su adaptación a la MiPyME, se encuentran: ISO/IEC 15504 y CORBA.

También se encontró que los esfuerzos están centrados en el análisis de los requisitos, la construcción y el diseño de la arquitectura del software.

Se espera que estos resultados contribuyan al cuerpo de conocimientos del tema en cuestión y sirva de guía para profundizar la investigación en algunos de los marcos de trabajo encontrados.

Referencias

Al-Habashna, A., Dobre, O., Venkatesan, R., & Popescu, D., (2012) "Second-Order Cyclostationarity of Mobile WiMAX and LTE OFDM Signals and Application to Spectrum Awareness in Cognitive Radio Systems," IEEE JOURNAL OF SELECTED TOPICS IN SIGNAL PROCESSING, 6(1), 26-42.

Automotive Standards Committee of the German Institute for Standardization. (2005). ISO/WD 26262: Road Vehicles – Functional Safety. Preparatory Working Draft, Technical Report.

Beck, K. (2000). *Extreme programming explained: Embrace change*. Boston: addition-Wesley.

Biolchini, J., Gomes, P., Cruz, A., y Orta, G. (2005). *Systematic review in software engineering*, RT-ES679/05. Rio de Janeiro: Systems Engineering Computer Science Department.

Cadavid, A. N., Fernández, J. D., y Morales, J. (2013). *Revisión de metodologías ágiles para el desarrollo de software*. *Prospectiva*, 11 (2), 30-39.

Calvo-Manzano, J. A., Cuevas, G., San Feliu Gilabert, T., Serrano, R., y García, I. (2006). *A software process improvement solution for small and medium-size enterprises*. Pittsburgh, Carnegie Mellon University: Software Engineering Institute.

Cater-Steel, A., Toleman, M., y Rout, T. (2005). *Process improvement for small firms: An evaluation of the RAPID assessment-based method*. *Information and Software Technology*, 48(5), 323-334.

CMMI Product Team. (2010). *CMMI for Development, Version 1.3*. CMU/SEI-2010-TR-033. Pittsburgh Carnegie Mellon University: Software Engineering Institute.

COMPETISOFT. (2006). *COMPETISOFT - Mejora de Procesos para Fomentar la Competitividad de la Pequeña y Mediana Industria del Software de Iberoamérica*. Versión 0.2.

Desharnais, J., Zarour, M., y April, A. (2007). *Very Small Enterprises (VSE) Quality Process Assessment*. (Artículo de investigación). Department of Software and IT Engineering: École de Technologie Supérieure. Recuperado de: ResearchGate, <http://www.researchgate.net/publication/228844814>

Duron, B., Muñoz, M., y Mejia, J. (2013). *Actual state of implementing software process improvements in software organizations*. Iberian Conference on Information Systems and Technologies (CISTI).

Dybå, T., y Dingsøyr, T. (2008). *Empirical studies of agile software development: A systematic review*. *Journal Information and Software Technology*, 50, 833-859. doi: 10.1016/j.infsof.2008.01.006

Fayad, M. E., Laitinen, M., y Ward, R. P. (2000). *Software Engineering in the Small*. *Communications of the ACM*, 43, 115-118. doi:10.1145/330534.330555

Gómez, G., Aguilera, A., Ancona, G. B., y Gómez, O.S. (2014). *Avances en las Mejoras de Procesos Software en las MiPyMEs Desarrolladoras de Software: Una Revisión Sistemática*. *Revista Latinoamericana de Ingeniería de Software*, 2(4), 262-268.

Hofer, C. (2002). Software Development in Austria: Results of an Empirical Study among Small and Very Small Enterprises. 28 th Euromicro Conference, 361-366. doi: 10.1109/EURMIC.2002.1046219.

Hurtado, J., y Bastarrica, C. (2005). PROYECTO SIMEP-SW. Universidad del Cauca.

International Electrotechnical Commission. (1993). Programmable controllers Part 3—Programming languages. IEC 61131–3. Geneva.

International Electrotechnical Commission. (1999). IEC 61508 Functional safety of electrical/electronic/programmable electronic safety-related system. Parts 1-7. International Electro-technical Commission. International Electrotechnical Commission.

International Organization for Standardization. (2010). ISO 9241-210:2010 Ergonomics of human-system interaction -- Part 210: Human-centred design for interactive systems. International Organization for Standardization

International Standard Organization. (2004). ISO/IEC 15504-2:2003 Information technology— Process assessment — Part 2: Performing an assessment. Switzerland: International Standar Organization.

International Standard Organization. (2010). ISO 9000 - Quality management. Recuperado de www.iso.org

ISO/IEC. (1998). Ergonomic requirements for office work with visual display terminals (VDT)s - Part 14 Menu dialogues, ISO/IEC 9241-14.

ISO/IEC/IEEE. (2008). IEEE STD 12207-2008 Standard for Systems and Software Engineering - Software Life Cycle Processes. IEEE.

Karner, G. (1993). Resource estimation for Objectory projects. Objective Systems SF AB (copyright owned by Rational Software).

Kitchenham, B.A., Pfleeger, S.L., Pickard, L.M., Jones, P.W., Hoaglin, D.C., El-Emam, K., y Rosenberg, J. (2001). Preliminary Guidelines for Emprical Research in Software Engineering. National Research Council Canada: Institute for Information Technology.

Kleppe, A., Warmer, J., y Bast, W. (2003). MDA Explained: The Model Driven Architecture: Practice and Promise. Boston, Addison-Wesley Longman Publishing Co.

Krunchten, P. (2004). The racional process: An introduction. Addison-Wesley Professional.

Lorensen, W., Rumbaugh, J., y Blaha, M. (1991). Modelado y diseño orientado a objetos. Metodología OMT.

Manganelli, R. L., y MKlein, M. (1995). Cómo hacer reingeniería. Bogotá, Grupo Editorial Norma.

Mas, A., Fluxà, B., y Amengual, E. (2012). Lessons learned from an ISO/IEC 15504 SPI programme in a company. *Journal of Software: Evolution and Process*, 24(5), 493-500. doi: 10.1002/smr.501

McFeeley, B. (1996). IDEAL: A User's Guide for Software Process Improvement. Pittsburgh, Carnegie Mellon University: Software Engineering Institute.

Mead, N., y Stehney, T. (2005). Security quality requirements engineering (SQUARE) methodology. *SESS '05 Proceedings of the 2005 workshop on Software engineering for secure systems—building trustworthy applications*, 30(4), 1-7. doi>10.1145/1083200.1083214

Mowbray, T. G., y Zahavi, R. (1995). The essential CORBA: systems integration using distributed objects. New York: Wiley.

Oktaba, H., Esquivel, C. A., Ramos, A. S., Martínez, A. M., Osorio, G. Q., López, M. R. et al. (2003). Modelo de Procesos para la Industria de Software MoProSoft, Versión1.1.

Ochoa, S.F., Quispe, A., Vergara, A., y Pino J. A. (2010). Improving Requirements Engineering Processes in Very Small Software Enterprises Through the Use of a Collaborative Application. 14th International Conference on Computer Supported Cooperative Work in Design.

OMG. (2015). Object Management Group. Recuperado de: www.omg.org

Pino, F., García, F., y Piattini, M. (2006). Revisión sistemática de mejora de procesos software en micro, pequeñas y medianas empresas. *Revista Española de Innovación, Calidad e Ingeniería del Software*, 2(1), 6-23.

Pino, F., Pedreira, O., Garcia, F., Rodriguez, M., y Piattini, M. (2010). Using Scrum to the Guide the Execution of Software Process Improvement in Small Organization. *The Journal of System and Software*, 83(10), 1662-1677

Pressman, R. S. (2006). Ingeniería del Software. Un enfoque práctico. (Sexta edición). McGraw Hill.

Pressman, R. S. (2014). Ingeniería del Software. Un enfoque práctico. (Tercera edición). McGraw Hill.

Quispe, A., Marques, M., Silvestre, L., Ochoa, S., y Robbes, R. (2010). Requirements Engineering Practices in Very Small Software Enterprises: A Diagnostic Study. XXIX International Conference of the Chilean Computer Science Society, 81-87. doi: 10.1109/SCCC.2010.35

Richardson, I. (2001). Software process matrix: a small company SPI model. *Software Process: Improvement and Practice*, 6(3), 157-165.

Weit e.V. (2005). The V-Modell XT. Recuperado de www.v-modell-xt.de

European Commission ESPRIT/ESSI Programme. (1993). Software Process Improvement in Regions of Europe (SPIRE). Recuperado de: <http://www.cse.dcu.ie/spire/>

Schwaber, K. (1995). The scrum development process. Proceedings of the Tenth Annual Conference on Object-Oriented Programming Systems, Languages, and Applications, Austin, Texas, USA.

Taramaa, J., Makarainen, M., y Ketola, T. (1995). Improving Application Management Process through Qualitative Framework. *International Conference on Software Maintenance*, 327-336. doi:10.1109/ICSM.1995.526554

Valdés, G., Visconti, M., y Astudillo, H. (2011). The Tutelkan Reference Process: A Reusable Process Model for Enabling SPI in Small Settings. 18th European Conference, EuroSPI 2011, 172, 179-190.

Vega, J. M., Fernández de Lucio, I., y Huanca, R. (2007) . La Relación Universidad-Empresa en América Latina: ¿Apropiación Incorrecta de Modelos Foráneos?. *Journal of Technology Management & Innovation*, 2(2), 97-109.

Venturi, G., y Troost, J. (2004). Survey on the UCD integration in the industry. Proceedings of the third Nordic conference on Human-computer interaction, 449-452. doi: 10.1145/1028014.1028092

Wangenheim, G. C., Rossa, J. C., Salviano, C. F., y Wangenheim, A. (2010). Systematic Literature Review of Software Process Capability/Maturity Models. Proceedings of International Conference on Software Process. Improvement And Capability determination (SPICE).

Weber, K., Araújo, E., Rocha, A., Machado, C., Scalet, D., y Salviano, C. (2004). Brazilian Software Process Reference Model and Assessment Method. *Computer and Information Sciences - ISCIS 2005*, 402-411.

Notas biográficas:

Antonio A. Aguilera Es Licenciado en Ciencias de la Computación, egresado de la Facultad de Matemáticas (FMat) de la Universidad Autónoma de Yucatán (UADY), Maestro en Ciencias Computacionales por el Instituto Tecnológico y de Estudios Superiores de Monterrey (ITESM), campus Monterrey. Actualmente es Profesor de Tiempo Completo e integrante del Grupo Académico de Tecnologías para la Formación en Ingeniería de Software en la FMat. El área de investigación de su interés gira en torno a la Ingeniería de Software e Informática Educativa.

Grisel B. Ancona Es Ingeniera de Software por la Universidad Autónoma de Yucatán (UADY). Actualmente es directora de la organización Women Who Code Mérida, e Ingeniera de Software en la empresa Grou Up. La línea de investigación de su interés es en torno a la mejora de procesos software.

Edwin J. León Es Licenciado en Ciencias de la Computación por la Universidad Autónoma de Yucatán (UADY), Maestro en Administración de Tecnologías de Información por el Instituto Tecnológico y de Estudios Superiores de Monterrey (ITESM), campus Monterrey. Actualmente es Profesor Asociado en la Facultad de Matemáticas e integrante del Grupo Académico de Ingeniería de Software de la UADY. La línea de investigación de su interés es en torno procesos de desarrollo de Software.

Juan P. Ucán Es Doctor en Sistemas Computacionales por la Dirección de Posgrado e Investigación de la Universidad del Sur, campus Mérida, Maestro en Sistemas Computacionales con especialidad en Ingeniería de Software por el Instituto Tecnológico de Mérida, Licenciado en Ciencias de la Computación por la Facultad de Matemáticas de la Universidad Autónoma de Yucatán (UADY). Es Profesor Titular en la Facultad de Matemáticas e integrante del Grupo Académico de Tecnologías para la Formación en Ingeniería de Software de la UADY. 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.