

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
ESCUELA DE POSGRADO



**“REVISIÓN SISTEMÁTICA DE COMPARACIÓN DE MODELOS DE
PROCESOS SOFTWARE”**

Tesis para optar el grado de Magister en Informática con mención en Ingeniería
de Software que presenta

CHRISTIAN ENRIQUE CANO SALAZAR

Dirigido por

MG. ABRAHAM DÁVILA

San Miguel, 2015

AGRADECIMIENTOS

Quiero agradecer a todas aquellas personas que me han apoyado a la realización de mi tesis de maestría:

A mi profesor del curso de Calidad de Software, el Prof. Abraham Dávila, gracias a él me interesé en el tema de Calidad de Software, aprendí en cada una de sus clases, que por cierto eran muy interesantes, es un gran maestro y un gran asesor. Le agradezco por la confianza depositada en mí, por estar siempre dispuesto a ayudarme, por absolver mis dudas con prontitud, por sus observaciones, por sus consejos, gracias estimado profesor. Es un gusto poder trabajar con usted.

Así también, un agradecimiento especial al Dr. Marcelo Pessoa y al Dr. Andrés Melgar por su cordial apoyo en la revisión de mi trabajo de tesis y por su cordial tiempo para poder orientarme en todo momento, además de sus buenos consejos.

A mis amigos, por alentarme a terminar mi tesis, por siempre desearme éxitos, por todos los lindos momentos compartidos que siempre quedarán en mi memoria.

Y finalmente, a cada uno de los integrantes de mi familia: mi madre, mi padre, mi abuela y mi hermano que siempre me han ayudado a alcanzar cada una de las metas que me he propuesto en la vida.

PREMIOS y/o RECONOCIMIENTOS

El presente trabajo de tesis fue aceptado para ser presentado y publicado en la **XLI Conferencia Latinoamericana de Informática CLEI 2015**, con el siguiente título: ***“Comparison of Software Process Models. A Systematic Literature Review”***.

Este trabajo ha sido realizado dentro del proyecto ProCal-ProSer financiado por Innóvate Perú, bajo el Contrato 210-FINCYT-IA-2013 y parcialmente soportado por el Departamento de Ingeniería y el Grupo de Investigación y Desarrollo de Ingeniería de Software (GIDIS) de la Pontificia Universidad Católica del Perú.

RESUMEN

La crisis del software a nivel mundial caracterizada por la baja calidad de los productos y el incumplimiento de los proyectos ha propiciado a que se tomen iniciativas para mejorar esta situación. En ese contexto los modelos de capacidad y madurez organizacional aparecieron y proliferaron de acuerdo a las necesidades de los colectivos empresariales o la percepción de los que propusieron esos modelos. Lamentablemente, un mercado con una gran diversidad de modelos como CMMI, ISO/IEC 12207, MoProSoft, MPS.Brasil, Agil SPI, ISO/IEC 29110 entre otros; no contribuye a que la industria mejore siendo necesario saber el grado de cobertura que tienen estos modelos para que las empresas que adoptaron o adopten uno, puedan migrar o adoptar otro pues les resulta más conveniente a sus intereses. Los modelos siguen apareciendo en el contexto de las tecnologías de información por lo que tener un esquema sistemático para determinar la cobertura es una necesidad relevante. En esta tesis de maestría se propone evaluar de manera comparada las metodologías o técnicas utilizadas para la comparación de modelos. Para realizar el estudio se utilizó una revisión sistemática de la literatura en bases de datos relevantes. A partir de la revisión se identificó algunas referencias de trabajos de comparación basados en implementaciones, otros de comparación siguiendo un esquema sistemático y manual, y un esquema basado en una representación gráfica denominada Composition Trees (CT). A partir de los modelos analizados se pudo determinar que los modelos pueden ser mejorados y en algunos casos combinados para obtener uno de mejores prestaciones para hacer un análisis comparativo de modelos de manera sistemática con posibilidades de extender a tres o más modelos en el análisis, a trabajar de manera bidireccional en la comparación y calcular de una manera sencilla la cobertura de ambos modelos.

LISTA DE FIGURAS

Figura 01. Modelos de calidad más comunes que las organizaciones toman en cuenta para implantarlos.

Figura 02. Elementos del modelo de proceso y sus relaciones

Figura 03. Publicaciones por tipo de recurso

LISTA DE TABLAS

Tabla 01. Diferencias entre revisión Tradicional y Revisión sistemática [43]

Tabla 02. Fases de la revisión sistemática [40]

Tabla 03. Palabras clave utilizadas en el estudio

Tabla 04. Symposiums y eventos

Tabla 05. Procedimiento de búsqueda

Tabla 06. Tendencia de las publicaciones

Tabla 07. Métodos de investigación encontrados

Tabla 08. Principales características de los métodos encontrados

Tabla 09. Experiencias de comparación por tipo de método o técnica de comparación

LISTA DE SIGLAS

RSL : Revisión Sistemática de la Literatura

CCT : Comparison Composition Trees

SAM : Software Analysis Method

RUP : Rational Unified Process

CMM: Capability Maturity Model

ISO/IEC: International Organization for Standardization and the International Electrotechnical Commission

LISTA DE ANEXOS

Anexo 01: Cadenas de búsqueda y resultados encontrados.

ÍNDICE GENERAL

AGRADECIMIENTOS.....	2
PREMIOS y/o RECONOCIMIENTOS	3
RESUMEN	4
LISTA DE FIGURAS	5
LISTA DE TABLAS.....	6
LISTA DE SIGLAS.....	7
LISTA DE ANEXOS.....	7
1. INTRODUCCIÓN	10
1.1. PROBLEMÁTICA	10
1.3. OBJETIVOS ESPECIFICOS.....	11
1.4. MÉTODOS Y PROCEDIMIENTOS.....	12
1.5. JUSTIFICACIÓN	12
1.6. DELIMITACIÓN	14
2. MARCO CONCEPTUAL	15
2.1. MODELOS DE PROCESOS.....	15
2.2. MODELOS DE PROCESOS DE SOFTWARE	15
2.2.1. MODELOS DE CALIDAD DE SOFTWARE.....	17
2.2.2. MODELOS DE MADUREZ DE SOFTWARE.....	18
2.3. ELEMENTOS DE LOS MODELOS DE PROCESOS SOFTWARE	18
2.3.1. ELEMENTOS DE ESTRUCTURACIÓN DEL SISTEMA	20
2.3.3. ACTIVIDAD	20
2.3.4. CICLO DE VIDA	21
2.3.5. MÉTODO	22
3. REVISIÓN SISTEMÁTICA DE LA LITERATURA	24
3.1. CONCEPTOS GENERALES DE LA REVISIÓN SISTEMÁTICA DE LA LITERATURA	24
3.1.1. DEFINICIÓN	24
3.1.2. RAZONES PARA OPTAR POR UNA REVISIÓN SISTEMÁTICA DE LA LITERATURA (SRL).....	25
3.1.3. IMPORTANTES CARACTERÍSTICAS DE UNA REVISIÓN SISTEMÁTICA DE LA LITERATURA (RSL) 25	
3.1.4. DIFERENCIAS ENTRE UNA REVISIÓN SISTEMÁTICA Y REVISIÓN CONVENCIONAL.....	26
3.1.5. FASES DE UNA REVISIÓN SISTEMÁTICA	27
3.1.5.1. PLANIFICACIÓN DE LA REVISIÓN	28
3.1.5.2. REALIZACIÓN DE LA REVISIÓN.....	29
3.1.5.2.1. IDENTIFICACIÓN DE LA INVESTIGACIÓN	30
3.1.5.2.2. SELECCIÓN DE LOS ESTUDIOS PRIMARIOS	30
3.1.5.2.3. ESTUDIO DE EVALUACIÓN DE LA CALIDAD	30
3.1.5.2.4. SEGUIMIENTO Y EXTRACCIÓN DE DATOS	31

3.1.5.2.5. SÍNTESIS DE LOS DATOS	31
3.1.5.3. INFORMACIÓN Y DIFUSIÓN DE LA REVISIÓN	32
3.2. METODOLOGÍA DE LA INVESTIGACIÓN	32
3.3. REVISIÓN SISTEMÁTICA	34
3.3.1. INTRODUCCIÓN	34
3.3.2. DESCRIPCIÓN DE LAS ESTRATEGIAS DE BÚSQUEDA.....	35
3.3.3. CRITERIOS DE SELECCIÓN DE ESTUDIOS	38
3.3.4. RESULTADOS DEL ESTUDIO Y CLASIFICACIÓN DE LOS ESTUDIOS	39
3.3.5. METODOLOGÍAS Y ENFOQUES TÉCNICOS IDENTIFICADOS PARA LA COMPARACIÓN DE PROCESOS	41
3.3.6. CASOS O EXPERIENCIAS DE COMPARACIÓN DE MODELOS DE PROCESOS SOFTWARE.....	47
4. CONCLUSIONES Y TRABAJOS FUTUROS	49
5. REFERENCIAS BIBLIOGRÁFICAS	51

1. INTRODUCCIÓN

1.1. PROBLEMÁTICA

Los modelos de proceso software han tenido un gran desarrollo durante la última década, desde propuestas propias (en cada organización) hasta esfuerzos internacionales como los estándares ISO/IEC 12207 u otros que han logrado visibilidad internacional como CMMI, MoProSoft, MPS-Br o la reciente ISO/IEC 29110; entre otros. Las oportunidades existentes para lograr un reconocimiento público (certificación), la presión del mercado a las empresas para que desarrollen software de calidad y el alcance geográfico de los modelos configura una situación particular para que las empresas -como ha ocurrido en algunos países en Latinoamérica- tengan que decidir qué modelo usar como base para sus planes de desarrollo empresarial[1]. Un claro ejemplo de esa situación se configuró con el Proyecto RELAIS que buscó introducir dos modelos (MoProSoft y MPS-BR) orientados a pequeñas empresas que desarrollan software en México, Brasil, Perú y Colombia [1]. Además se debe considerar que en estos países: (i) ya existe la influencia de ISO 9001 y CMMI [1]; (ii) han participado del Proyecto COMPETISOFT [2]; y (iii) participan del desarrollo del nuevo estándar ISO/IEC 29110 [3]; por lo que se tiene alrededor de cuatro modelos de procesos en sus industrias de software.

De otro lado, en la revisión de la literatura realizada en [4], los autores han encontrado más de 315 estándares, guías y otros documentos prescriptivos que son mantenidos por 46 organizaciones diferentes. Otros autores [5][6] mencionan que existe un número considerable de modelos de procesos. Asimismo, todos están enfocados en mejorar la calidad del software pero con distintos matices, según las organizaciones que los desarrollaron y el dominio de aplicación al que está orientado [5][7][8].

La comparación de modelos es una actividad en la ingeniería de software que se ha realizado en algunos casos utilizando el juicio experto [4] y en otros mediante alguna técnica que ha permitido cierto nivel de descomposición de los elementos de procesos [4] y recientemente usando una técnica como Composition Tree que ayuda de una manera gráfica a realizar el análisis sobre dos modelos [9]. Estas comparaciones se han desarrollado en modelos como RUP y PMBOK [10], RUP y MoProSoft [11] o CMM e ISO 9001 [12] para saber cuánto cubre un modelo a otro modelo (cobertura), para definir un plan de acción para migrar de un modelo a otro, para desarrollar guías de adopción de un modelo luego de haber adoptado otro; o incluso para evaluar la evolución técnica de un modelo.

Esta primera etapa de la tesis está enfocada en estudiar y analizar cada uno de los métodos existentes de comparación y determinación de cobertura de modelos de procesos software, así como las experiencias de comparación de modelos de procesos software. A partir de esto se analizarán las ventajas y desventajas de cada metodología para tomarlo como referencia en el diseño de la nueva metodología. En base a lo anterior, esta tesis busca servir de base para un trabajo futuro que pueda englobar el desarrollo de una nueva metodología de comparación de dos o más modelos de proceso software con un esquema al menos semi-automático.

1.2. OBJETIVO GENERAL

La tesis es de naturaleza exploratoria y el objetivo de la revisión es investigar cuáles son las técnicas, metodologías y/o métodos para la comparación de modelos de procesos software y cómo han sido empleados.

1.3. OBJETIVOS ESPECIFICOS

Los objetivos específicos son:

OE01. Comprender los conceptos de modelos de procesos software e Investigar el estado del arte de los modelos de calidad de software y los modelos de madurez de software existentes.

OE02. Recopilar las técnicas, metodologías y/o métodos de comparación de modelos de procesos software.

OE03. Analizar las características de las metodologías, técnicas y/o métodos de comparación existentes, para determinar beneficios y limitaciones, así como posibilidad de integrarla para el desarrollo de una nueva metodología de comparación.

1.4. MÉTODOS Y PROCEDIMIENTOS

1.4.1. Investigación bibliográfica a detalle (Revisión sistemática) de cada uno de los métodos, técnicas o metodologías existentes sobre determinación de similitudes y diferencias entre modelos de procesos software.

1.4.2. Investigación bibliográfica a detalle (Revisión sistemática) de cada uno de los métodos, técnicas o metodologías existentes sobre determinación de cobertura entre modelos de procesos software.

1.4.3. Analizar y comparar los métodos, técnicas y/o métodos encontrados, resaltando las ventajas y desventajas de cada uno de ellos.

1.5. JUSTIFICACIÓN

Una de las necesidades más apremiantes de las empresas de desarrollo de software es la de obtener una certificación de calidad para poder competir en los distintos sectores del mercado. Las organizaciones se enfrentan al problema de realizar la implantación de un modelo apropiado, de tal manera que logre los beneficios esperados por la organización, y al mismo tiempo, que se reduzcan los costos y el esfuerzo de implantación.

Las organizaciones que desarrollan software están empezando a considerar entre sus estrategias de crecimiento, la adopción de un modelo de procesos software. Sin embargo, al

contar con una gran variedad de los mismos existe la necesidad de expertos que analicen sus características para entender sus similitudes y diferencias y tener la capacidad de recomendar el más apropiado para una empresa en particular.

La cantidad de modelos existentes [4], complica la toma de decisiones respecto del modelo de proceso software más conveniente, y una mala decisión puede resultar en altos costos y esfuerzo. En el caso de los modelos de madurez, el esfuerzo de implantación puede llevar años para pasar de un nivel de madurez a otro [4]. Además cada modelo tiene un vocabulario, semántica, estructura y nivel de detalle distinto y con requisitos de evaluación diferentes. Esto refleja la necesidad de la industria en contar con herramientas o metodologías, que permitan comparar, de manera más rápida y fiable, los modelos de procesos software para tomar mejores decisiones.

La generalidad de los modelos de procesos o la ambigüedad de los mismos hace necesario desarrollar modelos, métodos o herramientas que contribuyan a clarificar las características de cada modelo (de manera comparativa), identificar ventajas y desventajas de cada modelo para la realidad de cada ámbito de influencia y determinar elementos entre ambos modelos para así desarrollar estrategias de implementación adecuadas. Con la creciente diversidad y desarrollo de diversas propuestas de software para distintos contextos [13], los métodos sistemáticos de comparación de procesos de software son necesarios porque permiten realizar un análisis, entendimiento y evolución de los procesos [14]. Sin embargo, es difícil comparar modelos de procesos software de manera consistente, sistemática y automática, debido a que están descritos en distintas arquitecturas, nivel de detalle y en lenguaje natural.

La industria del software busca encontrar un modelo de procesos que se adapte fácilmente a su modelo de negocio, por lo que se han realizado investigaciones para poder encontrar un método de comparación que les muestre un alto grado de fiabilidad. Es una necesidad conocer

las diferencias y similitudes entre modelos para poder tomar una decisión acertada y así evitar ciertos riesgos que puedan estar implicados con los objetivos del negocio.

Es por ello que a través de este estudio se pretende establecer una base teórica para que sirva como base para un desarrollo futuro de un nuevo método de comparación y determinación de cobertura entre modelos de procesos software, rescatando características importantes de los métodos y/o técnicas existentes, y adicionando otras características que puedan generar mayor valor al proceso o al resultado de la comparación.

1.6. DELIMITACIÓN

En los últimos años la gran cantidad de modelos de procesos software se han ido clasificando en dos grupos grandes en función al contexto para el cual fueron creados. Por un lado, existen modelos de procesos de software que fueron creados basados en el contexto de empresas de desarrollo de software de gran envergadura. Por otro lado, existen modelos de procesos de software que fueron creados para el contexto de pequeñas organizaciones de desarrollo de software que tienen un mayor nivel de detalle en su representación. Son dos contextos distintos que engloban un conjunto de necesidades y objetivos distintos. Por tanto, el presente trabajo se delimitará a desarrollar un método de comparación de modelos de procesos software en el contexto de las pequeñas organizaciones, dentro del cual se encuentran modelos como MoProSoft, Competisoft, Agile.SPI, MPS.Br, Mantema, ISO/IEC 29110 Basic Profile, entre otros.

2. MARCO CONCEPTUAL

2.1. MODELOS DE PROCESOS

La gestión eficaz de los proyectos requiere que se tomen decisiones en relación con el proceso de desarrollo, la asignación de recursos, y el horario. Todas estas decisiones en última instancia, requerirán alguna estimación de las consecuencias de las decisiones tomadas. Los administradores de proyectos expertos pueden confiar en la experiencia para estimar estos aspectos cuantitativos, pero los cambios en el personal, la tecnología y requisitos pueden invalidar la experiencia previa [15].

Los modelos de procesos apoyan la mejora de procesos, proporcionando orientación operativa con respecto a la secuencia crítica de los pasos del proceso, los flujos de información, y responsabilidades organizativas. Además, estos modelos tienen la capacidad para comprobar la integridad del proceso de trabajo [16][17][18]. Estos modelos son distintos porque ahondan en los detalles del proceso utilizado para desarrollar software. Muchos trabajos, hasta la fecha, han sido en el desarrollo de herramientas y métodos para el modelado y la automatización de procesos de software.

2.2. MODELOS DE PROCESOS DE SOFTWARE

Los modelos de procesos software se han utilizado durante más de veinte años para estimar la mano de obra y el calendario necesario para los proyectos de software. Las primeras aproximaciones al desarrollo de proyectos de modelado atacando el problema, se han dado a lo largo de dos dimensiones. Estas aproximaciones fueron para estimar el esfuerzo y la duración mediante la identificación de factores clave y determinación su contribución, siendo los modelos de Walston y Felix [20] y Boehm [21] ejemplos de esta aproximación. Estos modelos requieren nuevos coeficientes para el modelo en cada nuevo entorno. Boehm

expandió modelos empíricos para utilizar una variedad de factores basados en la fase de proyecto [21].

Hoy en día, las organizaciones que desarrollan software deberían considerar en sus estrategias de crecimiento las necesidades del mercado para ser competitivas. Las necesidades de los clientes y de los usuarios pueden incidir en mejorar la calidad del software, liberar la versión del software en menor tiempo, o que el software se elabore dentro de los límites del presupuesto acordado. Una aproximación a la satisfacción de dichas necesidades se lograría a través de la evaluación del proceso empleado para desarrollar el software, y tras los resultados del diagnóstico, planear una iniciativa de mejora.

Desde los años noventa del siglo XX, han surgido diversos modelos de calidad en los cuales se recopilan prácticas que los expertos consideran esenciales y que pueden contribuir a incrementar la calidad del software [19]. Dichos modelos se fundamentan en los principios de la gestión de la calidad total (TQM, por sus siglas en inglés) en donde el enfoque hacia la calidad cubre todas las áreas de la organización, desde los niveles directivos hasta los ingenieros individuales y en donde todas las actividades que se realizan en el ciclo de vida del software tienen impacto en la calidad final obtenida [19]. Estos modelos de calidad se basan en la convicción de que el proceso usado para desarrollar el producto de software influye en la calidad del producto y que un buen indicador de la calidad del producto se puede obtener si se conocen las prácticas o actividades que la organización lleva a cabo [19].

De hecho, la tendencia vigente es que las organizaciones, para acceder a distintos mercados, implanten simultáneamente distintos modelos de calidad (Fig. 01). Esto hace que surja el problema de la implantación de múltiples modelos [19]. En este estado, aunque se pretenden lograr los beneficios esperados al implantar cada modelo, la organización debería ser consciente de las limitaciones de recursos, tanto humanos como financieros, para emprender una iniciativa de esta naturaleza y de los potenciales conflictos en la coordinación entre

múltiples modelos de calidad. Entonces surgen un conjunto de interrogantes, ¿Qué se puede hacer para lograr los beneficios que señalan los modelos y al mismo tiempo, reducir los costos de implantación y evaluación?, ¿Cómo afrontar los cambios que se dan en los estándares como resultado de su evolución inherente?, ¿Cómo mantener la alineación con los procesos implantados en la organización?, y ¿Cómo se pueden reutilizar los productos generados y la experiencia obtenida de la implantación de un modelo para reducir costos en la implantación de uno nuevo?



Figura 01. Modelos de calidad más comunes que las organizaciones toman en cuenta para implantarlos [19].

2.2.1. MODELOS DE CALIDAD DE SOFTWARE

Los modelos de calidad pueden presentarse en diferentes estructuras y con distintos requisitos de conformidad. Los modelos de calidad usados en procesos de evaluación o certificación son prescriptivos, es decir, establecen las prácticas o requisitos del proceso que tienen que ser adoptados en una organización o proyecto concreto. Algunos ejemplos de éstos son los estándares y los modelos de madurez (CMMI, ISO15504) [19]. Los estándares contienen las

prácticas recomendadas, enunciadas como requisitos, que deben ser implantadas en el proceso de desarrollo de software. Estos requisitos pudieran servir de base para establecer contratos entre un cliente y un proveedor. Modelos de calidad representativos de este grupo son el ISO 9000 o el ISO/IEC 12207 [19].

2.2.2. MODELOS DE MADUREZ DE SOFTWARE

Los modelos de madurez, por su parte, presentan un mapa de ruta con el cual una organización puede determinar el grado en el cual su proceso de desarrollo es efectivo para lograr las metas del negocio, introducir innovaciones tecnológicas y usar los recursos limitados de la empresa eficientemente [19]. Generalmente tienen de 5 a 6 niveles de capacidad, y a partir de una evaluación, se determina el nivel que le corresponde a la organización evaluada. Capacidad se refiere al grado en el cual el proceso de desarrollo de software se puede categorizar en distintos niveles, según se gestiona, define, administra cuantitativamente y optimiza. Modelos como CMMI e ISO15504 pertenecen a esta categoría [19].

2.3. ELEMENTOS DE LOS MODELOS DE PROCESOS SOFTWARE

Los modelos de proceso son los marcos de apoyo a la gestión de todo el ciclo de vida de SW, que van desde el análisis de requisitos hasta la implementación, la transición y la modificación del sistema [22]. En los últimos años, la descripción de los modelos de proceso se ha desarrollado en base al siguiente conjunto de elementos básicos del modelo [22]:

- Los elementos de estructuración del sistema proporcionan la terminología de cómo un sistema puede estar estructurado en partes.
- Los productos de trabajo definen el tipo de resultados producidos durante el proceso de desarrollo.

- Las actividades describen lo que se debe hacer en el curso de un proyecto y por quien esto tiene que ser hecho.
- Los ciclos de vida definen el orden en que las actividades deben ser realizadas.
- El método describe los conceptos de modelado, las notaciones y técnicas que se utilizan dentro de los productos de trabajo.

La estructuración de los modelos de procesos sobre la base de los elementos del modelo dados sigue el principio de separación de intereses, por ejemplo, los conceptos de modelado están separados de los productos del trabajo, produciendo una serie de ventajas [22]. El modelo de proceso puede adaptarse más fácilmente a las necesidades particulares de una empresa o de un proyecto. Por otra parte, determinados elementos del modelo, como notaciones y el ciclo de vida se pueden ampliar o modificar sin desestabilizar todo el modelo de proceso [22]. Este último aspecto es fundamental debido a la larga vida útil de los modelos de procesos en las empresas. En nuestro enfoque, los elementos del modelo son una base fundamental para la comparación de dos modelos de procesos [22].

Como un aspecto adicional de la clasificación, muchos modelos de procesos distinguen entre los flujos de trabajo básicos de desarrollo de sistemas y flujos de trabajo de apoyo a todo el proceso [22]. Flujos de trabajo principales se refieren a la especificación y la implementación del sistema en un sentido más estricto, mientras que los flujos de trabajo de soporte están típicamente relacionados con la gestión de proyectos, gestión de la configuración y la gestión de la calidad [22]. Se puede afirmar que la conformidad de los modelos de procesos en los niveles más bajos se ocupa principalmente de los flujos de trabajo básicos y sus productos, mientras que la conformidad de los flujos de trabajo de apoyo indican un alto grado de conformidad. En el resto de esta sección los elementos de proceso se caracterizan con más detalle.

2.3.1. ELEMENTOS DE ESTRUCTURACIÓN DEL SISTEMA

Los elementos de estructuración del sistema definen los conceptos para descomponer sistemas grandes en partes más pequeñas. En el “Modelo en V”, por ejemplo, se estructuran sobre la base de los conceptos de los segmentos, unidades, componentes y módulos [22].

2.3.2. PRODUCTO DE TRABAJO

Se define un producto de trabajo como una unidad de información que abarca datos sobre el proyecto y / o sistema de interés. Cada producto de trabajo se caracteriza por el propósito metódico para el que sirve. Ejemplos de diferentes tipos de productos de trabajo son la arquitectura de software y documentos de aplicación como un archivo de origen y el plan del proyecto [22].

Considerando, por ejemplo, una arquitectura de sistema tiene el propósito metódico para delinear la estructura y el comportamiento general de un sistema proporcionando una visión general de las propiedades relevantes del sistema sin cubrir todos los detalles técnicos, un documento de aplicación cubre una parte del código de la aplicación de la arquitectura de un sistema, y por lo tanto propiedades del sistema en todos los detalles.

Los productos de trabajo no siempre se refieren al sistema como un todo, pueden estar asociados con un componente o una clase, como la especificación de interfaz de un componente. En general, un modelo de proceso tiene que definir qué producto de trabajo se tiene que producir para que tipo de elemento estructurante del sistema [22].

2.3.3. ACTIVIDAD

Mientras que los productos de trabajo representan aspectos estáticos de un modelo de proceso, las actividades representan los aspectos dinámicos. Se consideran las actividades como los bloques de construcción básicos para la descripción de lo que se debe hacer en el

curso de un proyecto [22]. De esta manera, se caracteriza una actividad por los contextos en los que se puede tener o llevar a cabo una actividad, y para qué contextos estos se llevan. Se puede determinar un contexto determinado por la existencia de determinados productos de trabajo y los estados en que se encuentran. La realización de una actividad puede significar la generación, manipulación, o el análisis de los productos de trabajo. En un modelo de proceso se define, por ejemplo, una actividad “diseño del sistema” produce una arquitectura de sistema a partir de un contexto en el que existe una especificación de requisitos validado [22].

2.3.4. CICLO DE VIDA

Cuando se define el orden en el que las actividades se van a realizar se describe un ciclo de vida de los modelos de procesos. Por lo tanto, un ciclo de vida indica cuándo hacer qué. Ejemplos de ciclos de vida conocidos son el cascada [23], el espiral [24], así como el desarrollo incremental e iterativo [25].

Los diferentes ciclos de vida pueden tener algunas actividades en común, pero también se pueden introducir actividades específicas del ciclo de vida [22]. Por ejemplo, el análisis de riesgos es una actividad clave del modelo de espiral, pero no un elemento del modelo de cascada.

Las definiciones de ciclo de vida en muchos casos tienen un fuerte impacto en los productos de trabajo. Si una actividad se lleva a cabo varias veces durante el ciclo de vida (por ejemplo, por cada incremento), bien los productos de trabajo producidos en esta actividad se actualizan de forma consecutiva y modificada, o de cada ejecución de esta actividad produce nuevas instancias de los productos de salida (por ejemplo, una arquitectura del sistema para cada incremento) [22].

2.3.5. MÉTODO

El método se define como: “forma de hacer algo” [26]. En más detalle, se define como un método algo que consiste en [26]:

- Conceptos de modelado
- Notaciones
- Directrices metódicas

Por conceptos de modelado se refiere a los términos que se utilizan para cubrir la información de los productos de trabajo. Por ejemplo, se puede especificar una arquitectura basada en conceptos de modelado orientado a objetos, tales como clases, objetos, el paso de mensajes, el enlace en tiempo, y sub-tipificación.

Mediante el uso de notaciones se representa conceptos de modelado. Los diagramas de clases son un ejemplo adecuado de notación para ilustrar aspectos estáticos de arquitecturas de sistemas basados en objetos. Para la representación de los productos de trabajo generalmente se puede usar diferentes conceptos de modelado y, por tanto, diferentes notaciones, y viceversa. Por ejemplo, se puede usar los diagramas de clases, no sólo en el caso de la especificación de la arquitectura del sistema, sino también en el caso de la especificación de requisitos.

Las directrices y técnicas metodológicas abarcan los conocimientos acerca de cómo aplicar y hacer frente a los conceptos de modelado y notaciones, y cómo promulgar actividades. Ejemplos de ello son los cálculos para diagramas de transición de estado [27], y el caso normal de la estrategia para la obtención y definición de escenarios alternativos en relación con una determinada tarea o un caso de uso [27].

Aunque hay una estrecha relación entre los conceptos de elemento estructurante del sistema y método, se deben tratar como conceptos separados. Los elementos de estructuración del sistema definen las unidades para descomponer un sistema, que tiene impacto en los productos de trabajo y actividades, mientras que el método ofrece estas unidades de estructuración con el apoyo metódico y de notación que se necesita para el desarrollo práctico. En la Fig. 02 se muestra los elementos del proceso dado y sus relaciones con la sintaxis de los diagramas de clases UML y la clase de conceptos de modelado, de asociación y agregación [28].

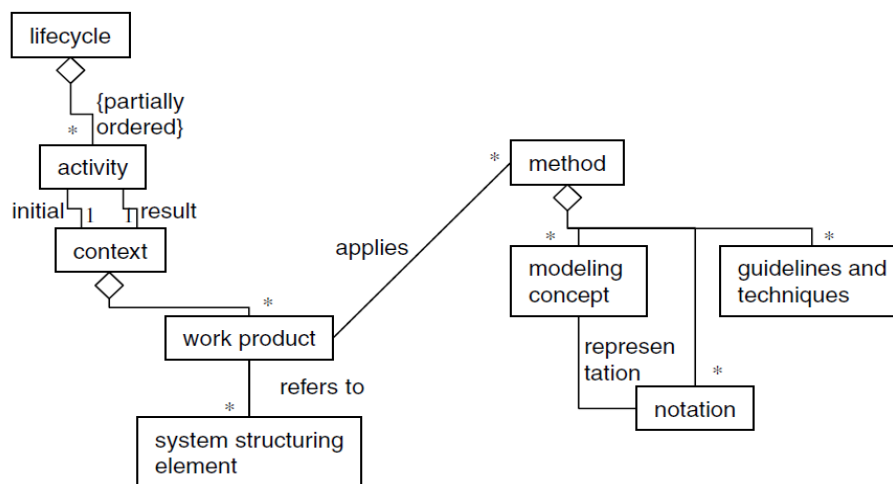


Figura 02. Elementos del modelo de proceso y sus relaciones [22].

3. REVISIÓN SISTEMÁTICA DE LA LITERATURA

3.1. CONCEPTOS GENERALES DE LA REVISIÓN SISTEMÁTICA DE LA LITERATURA

En esta sección se presenta la revisión sistemática de la literatura (RSL) a detalle; donde se describe la investigación metodológica, el diseño y la ejecución de la RSL.

Las siguientes subsecciones describen en detalle el proceso de una revisión sistemática de la literatura (RSL):

3.1.1. DEFINICIÓN

La RSL ha llegado a ser una metodología de investigación popular desde los años 1990 [39]. En esos años fue sumamente utilizado en la investigación médica y dentro de ese campo existen un número respetable de documentos muy bien documentados [39]. El número de investigadores en ingeniería de software que utilizan revisiones sistemáticas se ha ido incrementando continuamente desde el año 2004 [39]. Muchas revistas tienen secciones especiales para artículos basados en revisiones sistemáticas, así como, un número significativo de conferencias buscan contar con trabajos en esta categoría [39].

De acuerdo a Kitchenham, "A systematic literature review (often referred to as a systematic review) is a means of indentifying, evaluating and interpreting all available research relevant to a particular research question, or topic area, or phenomenon of interest" [40]

Los estudios fundamentales de una revisión sistemática son conocidos como estudios primarios, y la revisión sistemática de la literatura por sí misma es conocida como una forma de estudio secundario [39].

3.1.2. RAZONES PARA OPTAR POR UNA REVISIÓN SISTEMÁTICA DE LA LITERATURA (SRL)

Una revisión sistemática de la literatura debe llevarse a cabo de acuerdo con una estrategia de búsqueda predefinida. La estrategia de búsqueda debe contemplar la exhaustividad de la búsqueda.

Hay muchas razones para llevar a cabo una revisión sistemática de la literatura descrita por Kitchenham[40]. Las razones más comunes son:

- Para resumir las pruebas existentes en relación con un tratamiento o tecnología por ejemplo para resumir la evidencia empírica de los beneficios y limitaciones de un método ágil en específica
- Para identificar las lagunas en la investigación actual con el fin de sugerir áreas para la investigación posterior
- Proporcionar un marco de referencia o de fondo base para posicionar adecuadamente las nuevas actividades de investigación

3.1.3. IMPORTANTES CARACTERÍSTICAS DE UNA REVISIÓN SISTEMÁTICA DE LA LITERATURA (RSL)

La revisión sistemática de la literatura es distinta de una revisión de la literatura convencional. Algunas de las características que han contribuido a la identificación de dichas diferencias se indican a continuación [40]:

- Uno de los elementos importantes en la revisión sistemática de la literatura es el desarrollo del protocolo de la revisión. El protocolo de la revisión especifica las preguntas de investigación que se abordan y el método que se utiliza para la realización de una revisión en específica.

- Una estrategia de búsqueda definida se utiliza para llevar a cabo la revisión. El objetivo de la estrategia de búsqueda es identificar el mayor número posible de las literaturas relevantes.
- La estrategia de búsqueda y los resultados se han documentado para que en un futuro sirvan como referencia para el lector o investigador.
- Para evaluar el potencial estudio primario, la RSL requiere la especificación de los criterios de inclusión y exclusión para las selecciones de estudio.
- La revisión sistemática específica que la información necesaria para extraer de los estudios primarios se evalúan a través de criterios de calidad. Formularios de extracción de datos u otras herramientas de revisión se utilizan para documentar la información extraída.
- La revisión sistemática de la literatura se considera como pre-requisito para el meta-análisis cuantitativo que ofrece estudios de investigación integrados de diversas fuentes sobre el mismo tema.

3.1.4. DIFERENCIAS ENTRE UNA REVISIÓN SISTEMÁTICA Y REVISIÓN CONVENCIONAL

A continuación se presenta la tabla 01 con las principales diferencias entre la revisión de la literatura tradicional o convencional y la revisión sistemática, que presentan Dyba et al. [41] y que son una adaptación de un trabajo de Mulrow y Cook [42] [43].

Tabla 01. Diferencias entre Revisión Tradicional y Revisión Sistemática [43]

Característica	Revisión Convencional	Revisión sistemática
Preguntas	A menudo son amplias en alcance	A menudo se enfocan en preguntas de investigación
Cómo se identifican las investigaciones	Usualmente no se indica y son potencialmente parciales	Se indican las fuentes (bases de datos de búsqueda) y estrategias de búsquedas
Selección de investigaciones	Usualmente no se indica y es potencialmente parciales	Selección basada en criterios y son uniformemente aplicadas
Apreciación	Variable	Valoración crítica y rigurosa
Síntesis	A menudo son solo resúmenes cualitativos	Síntesis cuantitativa y/o cualitativa
Inferencias	A veces basadas en evidencias	Usualmente basadas en evidencias

Las revisiones sistemáticas permiten ser más exhaustivo en la investigación a realizar, por esa razón son más confiables y exactas a comparación de las revisiones convencionales.

3.1.5. FASES DE UNA REVISIÓN SISTEMÁTICA

El proceso propuesto de las revisiones sistemáticas es un proceso de tres fases: la planificación de la revisión, la realización de la revisión e información y difusión de la revisión (ver Tabla 02). Los pasos de cada fase se pueden adaptar, modificar, o dejar de lado de acuerdo al problema que se está investigando [40].

Tabla 02. Fases de la Revisión Sistemática [40]

Fase	Etapas
Planificación de la revisión	Identificación de la necesidad de una revisión
	Puesta en marcha de una revisión
	Especificación de las preguntas de investigación
	Desarrollo de un protocolo de revisión
	Evaluar el protocolo de revisión
Realización de la revisión	Identificación de la investigación
	Selección de los estudios primarios
	Estudio de evaluación de la calidad
	Seguimiento y extracción de datos
	Síntesis de los datos
Información y difusión de la revisión	Especificación de los mecanismos de difusión
	Formateo del informe principal
	Evaluación del informe

Las etapas mencionadas pueden aparecer de forma secuencial, sin embargo es importante reconocer que muchas de las etapas podrían implicar una ejecución de manera iterativa hasta encontrar o conseguir la información más adecuada.

3.1.5.1. PLANIFICACIÓN DE LA REVISIÓN

La planificación es la fase inicial de la SRL que comprende un plan de todos los pasos a seguir.

El punto de partida de la SLR es la identificación de la necesidad seguido por la puesta en marcha de la revisión y la formación de las preguntas de investigación que deben ser respondidas por el RSL [40].

La pregunta de investigación es formulada y presentada en el protocolo de la revisión.

Después de analizar la necesidad de la RSL en un área en particular, las bases de datos y fuentes disponibles se buscan a fondo documentación que responden a las preguntas de investigación propuestas.

El objetivo de la búsqueda es encontrar si ya existe alguna RSL que responde a la o las preguntas propuestas. Un protocolo de revisión se desarrolla especificando los pasos involucrados en la RSL. El protocolo contiene todos los pasos establecidos para la RSL, como se establece en la referencia [8]:

- Antecedentes de la RSL
- Identidad de la pregunta de investigación del RSL
- Lista de las bases de datos de la que se desea buscar diversas fuentes de información.
- Detalles de los criterios de inclusión y exclusión
- Listas de verificación para la evaluación de la calidad de los estudios individuales.
- Técnicas empleadas para la extracción de datos son establecidas y una técnica de validación se prevé para la manipulación de los datos extraídos.
- Una tabla de tiempo se prepara para las fechas de inicio y plazos para las distintas fases de la RSL.
- Finalmente el protocolo construido es revisada por expertos.

3.1.5.2. REALIZACIÓN DE LA REVISIÓN

Esta fase se inicia después de la aceptación de la revisión del protocolo y se compone de varios pasos.

El primer paso es identificar las fuentes de las que se realiza el SLR. Este proceso de identificación se inicia mediante la búsqueda y consulta de todas las bases de datos disponibles para la literatura primaria.

3.1.5.2.1. IDENTIFICACIÓN DE LA INVESTIGACIÓN

Una estrategia de búsqueda se determina y se sigue para llevar a cabo la revisión. Esta estrategia de búsqueda se lleva a cabo con la colaboración de los bibliotecarios y las sugerencias de los expertos en el campo correspondiente. La búsqueda se realiza en las bases de datos electrónicas, así como otras posibles fuentes, tales como revistas, registros de investigación y las listas de referencias obtenidas de los estudios primarios. El sesgo de publicación se debe reducir tanto como sea posible, lo que significa que los resultados positivos, negativos y nulos deben publicarse. Todo el proceso de búsqueda se documenta, por lo que puede ser transparente, replicable y posible volver a analizar.

3.1.5.2.2. SELECCIÓN DE LOS ESTUDIOS PRIMARIOS

La selección de los estudios primarios se utiliza para identificar y seleccionar los materiales de estudio más adecuados y pertinentes de los documentos de búsqueda. Este proceso de identificación se realiza con criterios de selección de estudio que incluyen tanto los criterios de inclusión y exclusión [40]. Estos criterios se basan en la o las preguntas de investigación. La selección de los estudios es un proceso de varias etapas, criterios de selección deben ser interpretadas libremente por lo que un estudio identificado en la búsqueda electrónica y/o manual puede excluirse claramente. El siguiente paso es aplicar criterios de inclusión y exclusión en base a las cuestiones prácticas tales como el lenguaje, la revista, los autores, los ajustes, los participantes o sujetos, diseño de la investigación, el método de muestreo, fecha de publicación, etc. A veces, los investigadores toman una tercera etapa en la selección proceso basado en los criterios detallados de calidad [40].

3.1.5.2.3. ESTUDIO DE EVALUACIÓN DE LA CALIDAD

La calidad de los documentos primarios identificados se analiza, lo que a su vez ayuda a la modificación de los criterios de inclusión y exclusión. La calidad se basa en tres factores: la

polarización, la validez interna y validez externa [40]. Un instrumento de calidad que es una herramienta de medición de la calidad se ha desarrollado para evaluar la calidad. Esta herramienta de medición es básicamente una lista de 21 factores que deben ser evaluados para cada estudio primario. Los estudios primarios se evalúan con respecto a las demandas de la lista de control y el nivel de calidad de los estudios primarios se evalúan mediante el uso de este instrumento de calidad [40].

3.1.5.2.4. SEGUIMIENTO Y EXTRACCIÓN DE DATOS

El objetivo de esta etapa es diseñar formas de extracción de datos para registrar la información obtenida a partir del estudio primario con precisión [40]. Los datos de los estudios primarios se extraen y se almacenan en los formularios de extracción de datos definidos [40]. Las duplicaciones se deben evitar durante esta fase. La extracción de datos de obra inédita o continuada se debe informar y las dudas deben aclararse con el autor de los estudios primarios. La estrategia de la extracción de datos se compone de dos estados: un análisis preliminar y análisis secundario [40].

3.1.5.2.5. SÍNTESIS DE LOS DATOS

La síntesis de los datos consiste en resumir los resultados de los estudios primarios incluidos. En resumen, los datos extraídos se sintetizan a continuación, con el fin de informar de los resultados de los estudios primarios examinados [40]. Esta síntesis de los datos extraídos en realidad le da la respuesta a la pregunta de investigación propuesta. Las respuestas extraídas no pueden provenir de un único estudio. La respuesta final podría construirse a partir de indicios y pistas de varios trabajos de investigación y la extracción puede realizarse de diferentes formas de fuentes. Todas las fuentes de las que se deducen respuestas deben ser especificadas y registradas de manera que sean de gran valor para la referencia futura.

Una respuesta a las preguntas planteadas puede ser analizada, a través de un gráfico o una prueba teórica de los estados [40]. Síntesis descriptiva, la síntesis cuantitativa y meta análisis son algunos tipos de métodos de síntesis de datos. Los datos sintetizados se presentan entonces utilizando diversas técnicas, como el diagrama de bosque, etc. gráfico en embudo o parcela ayuda a identificar el grado de sesgo de publicación [40].

3.1.5.3. INFORMACIÓN Y DIFUSIÓN DE LA REVISIÓN

La fase final de una revisión sistemática de la literatura consiste en anotar los resultados de la revisión. La RSL se informa, ya sea como parte de la tesis o en un artículo. Este último tiene una restricción de tamaño. Es muy importante que la revisión sea reportada con precisión.

3.2. METODOLOGÍA DE LA INVESTIGACIÓN

La revisión sistemática se lleva a cabo siguiendo las directrices de Kitchenham y Charter para la realización de la revisión sistemática de la literatura (RSL). Revistas y artículos de diferentes fuentes se exploran mediante el uso de palabras clave de búsqueda. El propósito es obtener material de estudio pertinente para responder a las preguntas de búsqueda planteadas. En general, el proceso de revisión se realiza a través de los siguientes pasos que se adaptan a partir de [40]

- i. Protocolo de preparación que incluye la definición del proceso
 - El proceso
 - Las preguntas de investigación
 - Los criterios de inclusión y exclusión
 - El procedimiento de análisis

- ii. Conducta del estudio piloto
 - Definición de la estrategia de búsqueda
 - La elección de las bibliotecas digitales y otras fuentes de materiales
 - Búsqueda
 - La revisión de los resultados
 - Resumir y analizar los resultados
 - Refinar las consultas para la búsqueda real
- iii. Conducta de la búsqueda real
 - Selección de bases de datos y las consultas de búsqueda en base a los resultados del estudio piloto
 - Búsquedas
 - Eliminación de duplicados
 - Aplicación de los criterios de inclusión y exclusión
 - Clasificación sin los artículos excluidos
 - Resumen y análisis de los resultados
- iv. Extracción de datos
 - Revisión de los artículos
 - Recopilación de información de los artículos
 - Clasificación de los artículos
 - Identificación de los estudios primarios
- v. La evaluación de calidad de los estudios
- vi. Análisis de los resultados
- vii. Desarrollo de conclusiones
- viii. Reporte final

3.3. REVISIÓN SISTEMÁTICA

Para este estudio, se ha seguido el esquema formal establecido en Kitchenham [40] y las recomendaciones sobre cómo proponer preguntas en Santos [44], también se ha establecido una estrategia de búsqueda y análisis. En esta sección se presentan los siguientes pasos:

3.3.1. INTRODUCCIÓN

En este estudio se utilizó las recomendaciones de Kitchenham [40] sobre cómo definir y llevar a cabo una revisión sistemática. Para ello, se definió como objetivo del estudio identificar los métodos y técnicas para la determinación de la comparación y la cobertura de los modelos de procesos de software. También, revisar cada uno de los modelos identificados, destacando fortalezas y debilidades. Las preguntas de investigación se han desarrollado siguiendo las directrices presentadas por Santos [44], que incluye los siguientes elementos a seguir:

- **Población:** Este es el conjunto de elementos que serán objeto de revisión. Estos son los documentos donde se presentan y aplican métodos o enfoques técnicos de comparación y determinación de cobertura entre los modelos de procesos de software.
- **Intervención:** Esto es lo que se evalúa en el conjunto de elementos de la población bajo prueba. Estos son los métodos y enfoques técnicos para la comparación y la determinación de la cobertura.
- **Comparación:** Elementos que sirven de base para la comparación, teniendo en cuenta los objetivos de este trabajo.
- **Resultados:** Esta es la información de salida que se espera de la investigación. Estudio comparativo de los métodos y enfoques técnicos para la comparación y la determinación de la

cobertura entre los modelos de procesos de software y la identificación de las características de los métodos y enfoques técnicos encontrados.

En base a estas premisas, las siguientes preguntas de investigación se plantean:

- ¿Qué metodologías o técnicas (o enfoques técnicos) existen para la comparación y la determinación de la cobertura entre los modelos de procesos de software y qué características tienen?
- ¿Qué casos o experiencias de comparación de modelos de procesos de software se han hecho en los últimos años?

Los resultados finales esperados de la revisión sistemática son los métodos o enfoques técnicos para comparar modelos de procesos utilizados en los últimos años para determinar las similitudes y diferencias entre dos modelos de procesos de software analizando los distintos aspectos de cada uno.

3.3.2. DESCRIPCIÓN DE LAS ESTRATEGIAS DE BÚSQUEDA

La revisión sistemática de la literatura se realizó en dos etapas. En la primera etapa se realizó la búsqueda en bases de datos electrónicas utilizando las palabras clave que guían la investigación. Las cadenas de búsqueda (ver Tabla 03) fueron generadas a partir de la combinación de términos y sinónimos clave utilizando OR y AND. Estos estudios se obtuvieron de las siguientes bases de datos:

- IEEE Xplore (<http://ieeexplore.ieee.org/>)
- Scopus (<http://www.scopus.com/>)
- ACM Digital Library (<http://portal.acm.org>)
- ScienceDirect (www.sciencedirect.com)

En la segunda etapa, se llevó a cabo una búsqueda específica en el evento internacional SPICE (Conference on Process Improvement and Capability dEtermination in Software, Systems Engineering and Service Management), cuyos trabajos están relacionadas con la norma ISO / IEC 15504 y en el Simposios brasileños SBQS (Brazilian Symposium on Software Quality) y SBES (Brazilian Symposium on Software Engineering), donde sus publicaciones están relacionadas con la Ingeniería de Software que incluyen modelos de procesos de software. En estas conferencias se realizaron búsquedas en los artículos publicados entre 2007 y 2013. Estos eventos registran un buen número de trabajos relacionados al tema, que ha aumentado de manera significativa, gracias a la presencia de investigadores extranjeros. Esto se refleja en los últimos artículos que están indexados en el ámbito de las bibliotecas digitales [45]. En el caso específico de SBQS, la búsqueda incluyó artículos en inglés, portugués y español para obtener el mayor número de artículos relevantes para contestar las preguntas de investigación.

Tabla 03. Palabras clave utilizadas en el estudio

PALABRAS CLAVE UTILIZADAS EN EL ESTUDIO		JUSTIFICACIÓN DE ELECCIÓN DE PALABRAS CLAVE
Población	"ISO/IEC 12207" OR "ISO/IEC 29110" OR "Software Factory" OR "MoProSoft" OR "MPS.BR" OR "Software Process Model" OR "Software Company" OR "Software Engineering" OR "Software Process Improvement" OR "Taxonomies" OR "Taxonomy"	Debido a la gran proliferación de modelos de procesos orientados pequeñas empresas, como son MPS.BR, Moprosoft, Competisoft, ISO/IEC 29110, entre otros. Se ha presentado la necesidad de poder determinar qué modelo de proceso es mejor respecto de otro de acuerdo al contexto en que se desea utilizar. Es por ello, que en este criterio se consideraron palabras clave con los nombres de algunos modelos. No se tomaron en cuenta modelos como CMMI porque están orientados a empresas grandes que, por lo general, no tienen necesidad de realizar un estudio para elegir qué modelo adoptar, dado que eligen modelos, como CMMI, por temas de consolidación, madurez y marketing. Asimismo, como existen modelos en distintos ámbitos y estudios de

		comparación de los mismos, es que se ha colocado cadenas de búsqueda relacionados con ingeniería de software, modelos de procesos de software y taxonomías, este último se incluyó debido a una sugerencia de un revisor en un congreso al que fue enviado este trabajo.
Intervención	"Comparison of Software Process Model" OR "Compare Software Process" OR "Software Process Models Comparison" OR "Coverage determination"	Relacionado al tema de intervención, lo que se busca es englobar artículos de investigación relacionados con la comparación de modelos de procesos, específicamente de software. Es por ello, que en las cadenas de búsqueda se consideran palabras de búsqueda como "Comparison" y "Compare".
Resultados	"Methodology" OR "Technique" OR "Method"	Como el objetivo de esta revisión sistemática es sintetizar y analizar la información sobre trabajos relacionados a métodos y técnicas de comparación de modelos de procesos software, se incluyó en este criterio las palabras de búsqueda "Methodology", "Technique" y "Method"
Estrategia de búsqueda	Population AND Intervention AND Outcomes	

Tabla 04. Symposiums y eventos

TIPO	FUENTE	SIGLA
Symposiums	Brazilian Symposium of Quality Software	SBQS
	Brazilian Symposium of Software Engineering	SBES
Eventos	International SPICE Conference on Process Improvement and Capability determination in Software, Systems Engineering and Management Services	SPICE

3.3.3. CRITERIOS DE SELECCIÓN DE ESTUDIOS

Una búsqueda inicial de los artículos devuelve un gran número de estudios que no son relevantes [40]. Por lo tanto, se sigue una revisión iterativa e incremental para llevar a cabo la revisión sistemática. El término iterativo indica la repetición de una o más actividades; el término incremental indica que la solicitud proviene de un subconjunto inicial de las fuentes, hasta que se va a toda la revisión. En el presente trabajo se llevó a cabo un análisis iterativo, ya que la ejecución (búsqueda, extracción de información, y la visualización de los resultados) de la revisión sistemática se ejecuta por completo en una fuente de la búsqueda, y luego los demás. Además, es gradual en el sentido de que el artículo (el producto) de la revisión sistemática crece y cambia con cada iteración hasta que sea firme. Por lo tanto, los criterios de inclusión y exclusión deberían basarse en la investigación que se relaciona con el tema. Por lo tanto, los estudios totalmente irrelevantes se descartan al principio.

La inclusión de un documento está determinada por la relevancia en relación con las cuestiones de búsqueda que se realiza mediante el análisis de los título, resumen, palabras clave, conclusiones y, en algunos casos, una revisión de la introducción. Como se indicó anteriormente, se propusieron los siguientes criterios:

- **Criterios de inclusión 01:** se tendrán en cuenta las publicaciones relacionadas que tengan métodos y enfoques técnicos para comparar y determinar la cobertura entre los modelos de procesos de software. Además, se tendrán en cuenta los artículos que presentan experiencias de comparación de modelos de procesos de software.
- **Criterios de inclusión 02:** Para temas específicos tales como el análisis de comparación se revisarán artículos desde 2003, porque se busca tener como referencia publicaciones recientes relacionados a técnicas o experiencias de comparación.

- **Criterios de exclusión 03:** Se dará prioridad a los elementos, con no más de 5 años de antigüedad. Sin embargo, para referencias conceptos generales se podrán considerar artículos con mayor antigüedad.

3.3.4. RESULTADOS DEL ESTUDIO Y CLASIFICACIÓN DE LOS ESTUDIOS

Los resultados de la revisión sistemática realizada son presentados en esta sección.

Para la clasificación de estudios, el procedimiento de búsqueda realizado en las bases de datos se llevó a cabo utilizando la cadena completa propuesta, en el método PICOC, para los casos de SCOPUS, ACM y Science Direct, sin embargo para IEEE Explore se realizaron 3 modificaciones a la cadena total propuesta, para mayor detalle del proceso de búsqueda revisar el ANEXO 01. Al finalizar la búsqueda en las bases de datos se obtuvieron 263 estudios iniciales, a los cuales se realizó una revisión preliminar, logrando verificar que un total de 121 no se repiten lo que significa que dichos artículos no se encuentran en otras bases de datos. De este grupo se seleccionaron 31 para su relevancia, de los cuales 22 fueron seleccionados como estudios primarios relevantes (ver Tabla 05). A partir de la selección inicial de 31 artículos, los cuales se seleccionaron sobre la base de una lectura atenta de los títulos, resúmenes, palabras clave, conclusiones y trabajos futuros. Se realizó un análisis por porcentaje para determinar la tendencia de publicaciones por año (ver Tabla 06), así como un análisis de las publicaciones por tipo de fuente (ver Fig. 03).

En la Tabla 06, se realizó un análisis de 22 trabajos de investigación utilizados, que han sido seleccionados como los estudios primarios. Además, la Tabla 07 muestra la distribución por tipo de artículo con los 22 estudios primarios, en los cuales 40.9% son investigaciones sobre experiencias de comparación y 59% restante, son estudios de caso.

Tabla 05. Procedimiento de búsqueda

BASES DE DATOS	ESTUDIOS					PORCENTAJE
	Fecha de búsqueda	Descubiertos	No repetidos	relevantes	primario	(%)
Scopus	11/2014	19	17	5	3	13.63%
IEEE Explore	11/2014	24	11	5	3	13.63%
ACM	11/2014	131	41	4	2	9.09%
Science Direct	11/2014	73	36	6	3	13.63%
Conferencias	11/2014	16	16	11	9	40.90%
	Total	263	121	31	22	100.0%

Tabla 06. Tendencia de las publicaciones

AÑO	PORCENTAJE (%)	FRECUENCIA
1997	3.23%	1
1999	3.23%	1
2000	3.23%	1
2001	6.45%	2
2004	3.23%	1
2006	3.23%	1
2007	6.45%	2
2008	9.68%	3
2009	16.13%	5
2010	12.90%	4
2011	9.68%	3
2012	19.35%	6
2013	3.23%	1

Figura 03. Publicaciones por tipo de recurso

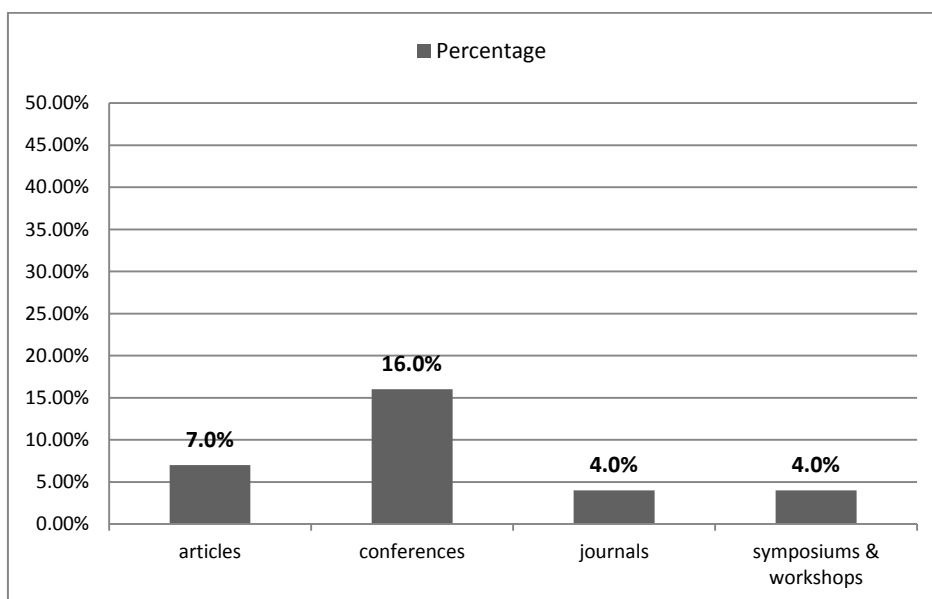


Tabla 07. Métodos de investigación encontrados

Método de investigación	Frecuencia	Porcentaje (%)
Experiencias de comparación	9	40.90%
Casos de estudio	13	59.09%

3.3.5. METODOLOGÍAS Y ENFOQUES TÉCNICOS IDENTIFICADOS PARA LA COMPARACIÓN DE PROCESOS

Con base en la evidencia para responder a la pregunta: ¿Qué metodologías o técnicas (o enfoques técnicos) existen para la comparación y la determinación de la cobertura entre los modelos de procesos de software y qué características tienen?

Se ha identificado cinco métodos para hacer la comparación de modelos de procesos de software: (i) "Método de comparación descriptivo" (ii) "Software de Análisis Método - SAM" (iii) "El análisis de correspondencias de los elementos del proceso", (iv) "Taxonomía" y (v) basado en la técnica de Composition Trees.

El "*Método de comparación descriptivo*" utilizado en [30] [31] [32] se basa en una observación sistemática y descripción detallada de un modelo de proceso de software. El proceso seguido es el siguiente. En primer lugar, describir en detalle el modelo destacando aspectos como principios básicos, la complejidad, los requisitos necesarios, el costo y los conceptos generales del modelo. En segundo lugar, describir en detalle las ventajas que ofrece el modelo, así como las desventajas o limitaciones que deben tenerse en cuenta a la hora de decidirse a utilizar este modelo, incluyendo un análisis de riesgo. En tercer lugar, identificar todos los aspectos o características comunes a todos los modelos para comparar, y luego enumerarlos incluyendo las ventajas y desventajas encontradas en cada uno. Por último, hacer un cuadro comparativo donde las filas son los aspectos comunes, ventajas y desventajas, y donde las columnas representen cada modelo a comparar buscando determinar la relación entre los elementos de los modelos de procesos de software en comparación y realizar una cuantificación de la relación entre los elementos de comparación.

El "*Método de Análisis de Software (SAM)*" utilizado en [46] [47] se introduce para promover una mejor comprensión de los procesos de software. Esto incluye sus similitudes, diferencias y relaciones. SAM consta de tres pasos principales: (1) La *Elaboración*, que se basa en la cobertura de todos los datos que se especifican en las referencias de base. Una referencia de base es idealmente la referencia estándar para la definición del proceso. La elaboración por lo general sigue el enfoque de arriba hacia abajo. La elaboración final se determina ya sea cuando todos los detalles en las referencias de base son capturados y / o cuando cada elemento en el modelo elaborado alcanza el estado atómico. Un elemento atómico es el elemento más pequeño que no se puede dividir aún más [46]. (2) La *normalización* es un paso que normaliza cada término con el fin de poder comparar en igualdad de condiciones. Cada elemento de proceso necesita ser definido utilizando la misma terminología estándar. Esto se puede hacer usando un diccionario proceso. En esta etapa, SAM, determina de forma automática y reemplaza términos no estándar por sus términos estándar correspondientes del diccionario

proceso previamente definido. Por ejemplo el término "historias de usuario" definidos en XP se sustituye por "necesidades de los usuarios" [46]. (3) La *abstracción*, este es el paso final en la que se combinan todas las actividades similares para proporcionar una visión abstracta de cada modelo de proceso. A través del paso de la abstracción, SAM establece la línea de base para la comparación proceso relacionando elementos normalizados a sus elementos raíz. Los términos normalizados se abstraen más en términos de raíz. Esto permite al ingeniero de procesos para reconocer la raíz de las actividades. Por lo tanto, varios modelos de proceso se pueden comparar [47].

El "*Análisis de Correspondencias de Elementos de Proceso*" que se utiliza en [10] [11] [12] [33] [34] consiste en la realización de un análisis exploratorio, con el fin de resumir una gran cantidad de datos, con la información de menor pérdida posible. El objetivo principal es dar a conocer la estructura de los datos en cada elemento del proceso, a través de una descomposición de los elementos en el nivel más bajo, a continuación, analizar y determinar la relación, literalmente, entre los elementos del proceso de comparación. Para ello, se requiere una comprensión de las estructuras de los dos modelos a comparar, teniendo en cuenta factores clave como objetivos, actividades, entradas, salidas, propósitos, resultados y otros elementos de procesamiento. Estos elementos de procesamiento deben ser descritos en detalle, y luego establecer una correspondencia entre los elementos de cada proceso. Esta correspondencia no siempre implica relación uno a uno entre ellos, ni implica que está completo. En algunos casos, un elemento de procesamiento puede ser dividido en diferentes niveles de abstracción para la correspondencia correcta entre los elementos. Por último, se calcula la cobertura de un modelo de proceso sobre otro de acuerdo a las siguientes dos reglas [5], una de las cuales es la multiplicidad entre los elementos (desde uno o varios elementos de un modelo no cubre necesariamente plenamente a un elemento de otro modelo) y la segunda regla es la cubierta (ver ref. 33, 34), que consiste en la asignación de una puntuación en función al nivel de cobertura alcanzada.

La “*Taxonomía*” [48] [49] [50] [51], se basa en el método de comparación descriptiva obteniendo una lista de las características consideradas como importantes para la descripción de los marcos de SPI (Software Process Improvement) haciendo una presentación tabular corta, lo que requiere que cada característica tenga solamente una breve descripción. Además, la superposición entre algunas de las características es importante para capturar pequeñas variaciones entre los marcos [49]. Las descripciones características se han agrupado en 5 categorías para facilitar la lectura y comprensión. (1) *categoría General*, este grupo de características describe los atributos generales o características de los marcos de SPI. Tales atributos suelen ser específicos para cada marco y con frecuencia relacionado con la manera de cómo se construye o cómo se diseña el marco. (2) *Categoría del proceso*, este grupo de características describe cómo se utiliza el marco SPI (3) *categoría de Organización*, este grupo de características está relacionado con los atributos de la organización y el entorno en el que se utiliza el marco SPI. En otras palabras, la persona que está involucrada y para qué tipo de organización es el marco aplicable. (4) *la categoría de calidad*, esto está relacionado con la dimensión de calidad al señalar aspectos como cómo se mide la progresión, cuya calidad perspectiva es empleado y lo que eso significa en términos de indicadores de calidad y relaciones casuales. (5) *Categoría resultado*, este grupo de características no solo describen los resultados de emplear un marco SPI, sino también los costos de alcanzar estos resultados y los métodos utilizados para validarlos [49]. Por último, se tiene que comparar los marcos SPI entre categorías de cada uno.

La comparación usando “*Composition Trees (CT)*” usado en [35] se basa en una técnica adaptada de la ingeniería de requisitos para describir gráficamente la composición de un componente de un sistema de software. Este método proporciona, de forma gráfica, la información de resumen, muy útil, incluyendo los estados, los atributos y las relaciones en el sistema de software [9]. Por lo tanto, tiene una manera fácil de identificar las similitudes y diferencias entre los procesos relacionados. El proceso seguido es el siguiente [9]: En primer

lugar los principales elementos de un modelo de proceso están identificados, que son los propósitos y resultados. En segundo lugar, leer los propósitos y los resultados, y hacer una lista completa y coherente de los nombres y siglas, que son generalmente los componentes o atributos de los componentes. En tercer lugar, a partir del estado de finalidad del proceso, identificar los componentes y su estado y dibujar la CT inicial. Finalmente, se lee cada resultado uno por uno, para identificar los componentes, los estados, las relaciones y los atributos y luego integrar la información en la CT, la cual tiene toda la información integrada en una sola gráfica, por lo que las relaciones entre las diferentes partes se hacen visibles. La información de cada componente está dispuesta en un lugar de modo que será más fácil de recuperar y, en general, la versión gráfica del proceso puede tener menos ambigüedad, puede ser más fácil de entender y más fácil para las personas identificar los defectos de proceso [9].

En la Tabla 08, se muestran las principales características de cada método o técnica, que le permite ver las diferencias entre los métodos encontrados. De esta manera se pueden identificar las ventajas posibles de un método sobre otro.

Tabla 08. Principales características de los métodos encontrados

Métodos y técnicas	Necesidad de juicio experto	Tipo de comparación	Comparaciones simultaneas	Metodología sistemática
Método de comparación descriptiva	Sí	Unidireccional	2	No
Método de análisis de software (SAM)	No necesariamente	Unidireccional	2	Sí
Análisis de correspondencia de elementos de proceso	No necesariamente	Unidireccional	2	Sí
La taxonomía	Sí	Bidireccional	2	Sí
Comparison Composition Trees (CCT)	No necesariamente	Bidireccional	2 o más	Sí

En primer lugar, el método de comparación descriptiva y la taxonomía se caracterizan por la necesidad de un juicio o interpretación de los expertos acerca de los modelos de procesos que están comparando. Sin embargo, el SAM, el método de análisis de correspondencias de elementos de proceso y la CCT no necesitan necesariamente un experto en los modelos a comparar, porque hay un proceso sistemático confirmado en su metodología.

En segundo lugar, el método de comparación descriptiva, SAM y el método de análisis de correspondencias de elementos de procesamiento permiten una comparación unidireccional, es decir, si se quiere comparar un modelo A y el modelo B, primero debe hacerse comparando el modelo A respecto al modelo B, a continuación, una comparación de modelo B respecto al modelo A. En contraste, el método de la taxonomía y CCT permite una comparación bidireccional, es decir, si se quiere comparar el modelo A y el modelo B, puede hacerlo utilizando sólo una vez algoritmo de la CCT, con el cual obtendrá un resultado de la comparación global sin volver a aplicar el método.

En tercer lugar, el número de comparaciones simultáneas que se pueden realizar se limitan a un total de dos para el caso del método de comparación descriptiva, SAM y para el análisis de correspondencia de elementos de proceso. Sin embargo, el método de CCT tiene una gran versatilidad ya que permite comparar dos o más métodos simultáneamente de manera gráfica.

Por último, cada método de comparación se analiza para determinar un nivel diferente de la objetividad en el resultado de la comparación final, ya que depende del nivel de la interpretación de los datos analizados. Por otro lado, es importante considerar si la metodología utilizada proporciona un enfoque sistemático para usar. En este caso, metodología descriptiva no proporciona un enfoque sistemático bien definido. Sin embargo, SAM, los métodos de análisis de la correspondencia de los elementos del proceso, la taxonomía y CCT proporcionar un enfoque sistemático, ya que se basan en el concepto de

granularidad. En otras palabras, tratar de descomponer los elementos en el nivel más bajo posible.

3.3.6. CASOS O EXPERIENCIAS DE COMPARACIÓN DE MODELOS DE PROCESOS SOFTWARE

En la Tabla 09, se muestran las experiencias de comparación que se encontraron, y donde se utiliza cada método. En la referencia [12] se realiza una comparación directa entre la ISO 9001 y CMM, lo que demuestra que los modelos de procesos de software con una alta tasa de adopción en contextos para los que fueron creados no necesariamente tienen una correspondencia directa de las similitudes, lo cual se verifica en este trabajo. En la referencia [33], IBM hace el mismo trabajo, pero entre Rational Unified Process y Microsoft Solutions Framework, tratando de encontrar las similitudes y diferencias, ventajas y desventajas de cada uno. En las referencias [46] [47] se muestra un método interesante que se utiliza para comparar dos modelos de procesos de software, descriptivamente, con un enfoque sistemático. En las referencias [11] [34], con la proliferación de modelos para las pequeñas organizaciones aparece la necesidad de comparar estos nuevos modelos. En este caso, se compara MoProSoft contra RUP y el modelo MPS.BR, respectivamente. Las similitudes son muy prominentes, y muestran una clara evidencia de que puede aplicarse a diferentes contextos y mercados. Por cierto, el método de la taxonomía es realmente utilizada para comparar la mejora de procesos de software [48][49][50][51]. En este caso, se compara CMM contra ISO 9000. Es un método interesante, pero es necesario contar con algunas herramientas como un diccionario de datos para el proceso. Por último, el nuevo método formal de notación gráfica, llamada Composition Trees (CCT), tiene básicamente un trabajo con una aplicación, que puede ser analizada en la referencia [35]. Ese documento pretende validar el perfil de entrada de la ISO/IEC 29110 frente a su contraparte en la norma ISO/IEC 12207. Esta validación se realiza mediante el método de la Composition Trees, a través de una

comparación directa de las notaciones gráficas, haciendo hincapié en la claridad y la gran contribución a no mostrar ambigüedad en el resultado de la comparación.

Tabla 09. Experiencias de comparación por tipo de método o técnica de comparación

Métodos y técnicas	Experiencias de comparación
Método de comparación descriptiva	ISO 9001 vs. CMM [12] Rational Unified Process (RUP) vs. Microsoft solutions Framework (MSF) [33]
Método de análisis de Software (SAM)	Waterfall Model vs Spiral Model[46][47]
Análisis de correspondencia de elementos de proceso	Rational Unified Process (RUP) vs. MoProSoft [11] MPS.BR vs. MoProSoft [34]
La taxonomía	TQM, CMM v1.1, ISO 9000, ISO/IEC15504(SPICE), IP/EF/GQM, and SPIQ [48][49][50][51]
Comparison Composition Trees (CCT)	ISO/IEC 29110-5-1-1 Basic Profile vs ISO/IEC 12207 [35]

4. CONCLUSIONES Y TRABAJOS FUTUROS

En este trabajo se obtuvieron los resultados de una revisión sistemática de 22 artículos, como estudios primarios, y además se analizaron de manera profunda teniendo en cuenta otros artículos relevantes relacionados con el tema de investigación. La investigación presenta una lista de las características de los estudios de investigación publicados por año, tipo de investigación, métodos y técnicas utilizadas, que es una práctica que se ha estudiado poco. Además, como se señaló anteriormente, es necesario realizar un trabajo en mayor profundidad sobre este tema para que se pueda desarrollar un esquema formal y de trabajo automatizado en la medida de lo posible, sobre la base de las Composition Trees y reglas de cobertura. Como trabajos futuros se proponen los siguientes temas a desarrollar.

- Se obtuvo una buena base de justificaciones de necesidad de un modelo o técnica de comparación de modelos de procesos software y/o determinación de cobertura que no dependa necesariamente de un juicio experto. Lo cual generaría un gran impacto en el crecimiento de las PYMES o pequeñas empresas, a través de la mejora y/u optimización de sus procesos de desarrollo de software. Dado que, en la mayoría de casos, es poco probable que las PYMES puedan invertir en un experto que les pueda decir este u otro modelo de proceso es el adecuado para su caso. Sería interesante que las mismas PYMES puedan realizar ese análisis utilizando un modelo o técnica sistemática que les permita llegar a una decisión acertada al momento de escoger un modelo de proceso en el que deban basar su compañía.
- La realización de muchos trabajos de comparación y/o determinación de cobertura han sido realizados de manera manual, a través de un enfoque sistemático, que en la mayoría de casos dependían de un juicio experto para validar que lo que se está realizando es correcto. Sería muy interesante poder contar con una herramienta software que permita realizar esta comparación de manera automática, ingresando

ciertos parámetros y determinar qué modelo es mejor para tal empresa. Ahora que se conoce el potencial que podría tener automatizar las Composition Trees para realizar comparación de modelos de procesos, sería un buen punto de partida para poder desarrollar un modelo automático que te permita comparar no solo dos modelos a la vez, sino varios, permitiendo ahorrar tiempo y dinero, para ganar crecimiento en un menor tiempo.

5. REFERENCIAS BIBLIOGRÁFICAS

- [1] RELAIS (2010), “Red Latinoamericana de la Industria de Software”– www.relais.mx
- [2] Competisoft (2006), “COMPETISOFT: Proyecto de mejora de procesos para fomentar la competitividad de la pequeña y mediana industria del software de Iberoamerica”, Diciembre, 2006.
- [3] ISO/IEC (2011) “ISO/IEC 29110-5-1-2:2011 Software engineering – Lifecycle profiles for Very Small Entities (VSEs) – Part 5-1-2: Management and engineering guide: Generic profile group: Basic profile”, Geneva, January.
- [4] Moore, J.W. (1999). Integrated collection of software engineering standards. IEEE Software. 16(6), 51-57.
- [5] S.Sheard, “Evolution of the Frameworkd Quagmire” , IEEE Computer Magazine Vol. 34, N° 7. Pp. 96-98, 2001.
- [6] E. Ericsson, P. Gustafsson, D. Hook, Marcks Von L., W. Rocha, “Process improvement framework evaluation”, Management Science and Engineering International Conference, pp. 319-326, 2010.
- [7] S. Sheard, “The Frameworks Quagmire”, Crosstalk: The Journal of Defense Software Engineering, vol. 10, no. 9, September 1997.
- [8] L. Ferreira, R. Machado, M. Paulk, Size and Complexity Attributes for Multimodel Improvement Framework Taxonomy, 36th EUROMICRO, 2010.
- [9] Wen, L., Tuffley, D., Rout, T., Using Composition Trees to Model and Compare software Process, in Proceedings of 11th International conference, SPICE2011, pp1-15. 2011

- [10] Charboneau, S. (2004) "Software Project Management" - A Mapping between RUP and the PMBOK" in: IBM Rational Edge. 14 May.
- [11] Cánepa, K. y Dávila, A. (2008) "Mapeo de los procesos de RUP respecto a MoProSoft, in Proc. JIISIC, 2008, pp. 139-146.
- [12] Paulk, M. (1995) "How ISO 9001 compares with the CMM". In: IEEE Software January, pp 74-83.
- [13] Sheard, S. A. , The frameworks quagmire, a brief look, Proceedings of the 7th Annual International INCOSE, Symposium('97), 1997.
- [14] Podorozhny, R.M., Perry, D.E. and Osterweil, L. J., Artifact-based functional comparison of software processes, 4th International Workshop on Software Process Simulation and Modeling, May 2003, pp V .29.1-10
- [15] Robert H. Martin, David Raffo; A Comparison of Software Process Modeling Techniques. 1997.
- [16] Mi, P. and Scacchi, W., "Modeling Articulation Work in Software Engineering Process," Proceedings of the First International Conference on the Software Process, Redondo Beach, CA, IEEE Computer Society Press, October 21-22, 1991, pp. 188-201.
- [17] Dieters, W. and Gruhn, V., "Software Process Model Analysis Based on FUNSOFT Nets," Mathematical Modeling and Simulation, Vol. 8,1991.
- [18] Kellner, M.I. and Hansen, G.A., "Software Process Modeling: A Case Study," Proceedings of the 22nd Annual Hawaii International Conference on System Sciences-Vol ZZ -Software Track, IEEE Computer Society Press, 1989, pp. 175-188.
- [19] Fuggetta, A. (2000). Software process: a roadmap (Eds.).Proceedings of the Conference on The Future of Software Engineering. pp. 25-34. ACM, Limerick, Ireland.

- [20] Walston, C.E. and Felix, C.P., A Method of Programming Measurement and Estimation, IBM System Journal 16, 1 (1970): 54-73
- [21] Boehm, B. "Software Engineering Economics", Prentice-Hall, 1981, ISBN 0-13-822122-7
- [22] Breu, R., Huber, W., Scwerin, W. (2001) A conformity model of software processes. Information and Software Technology. Volume 43, Issue 5, Pages 339–349.
- [23] W.W. Royce, Managing the development of large software systems, Tutorial: Software Engineering Project Management, IEEE Computer Society, Washington, DC, 1970, pp. 188-127.
- [24] B. Boehm, A spiral model of software development and enhancement, Software Engineering Project Management, 1987, pp. 128-142.
- [25] H. Balzert, Lehrbuch der Software-Technik 1/2, Spektrum Akademie, 1996.
- [26] A.S. Hornby, Oxford Advanced Learner's Dictionary of Current English, 3rd ed., Oxford University Press, Oxford, 1974.
- [27] B. Rumpe, Formale Methodik des Entwurfs verteilter objektorientierter Systeme, Herbert Utz, Wissenschaft, 1996.
- [28] OMG: Unified Modeling Language Specification, Version 1.3 alpha R5, <http://www.omg.org/>, March 1999.
- [29] Feiler P. H., Humphrey. W.S. (1992) . Software Process Development and Enactment, Software Engineering Institute, Pittsburgh, CMU/SEY-92-TR-04 p 11
- [30] Maheshwari, S., Dinesh Ch. Jain (2012) A Comparative Analysis of Different types of Models in Software Development Life Cycle. International Journal of Advanced Research in Computer Science and Software Engineering. Volume 2, Issue 5.

[31] Ashwini Mujumdar, Gayatri Masiwal, P. M. Chawan (2012) Analysis of various Software Process Models. International Journal of Engineering Research and Applications (IJERA). Vol. 2, Issue 3, pp.2015-2021.

[32] Vohra, P., Singh, A. (2013) A Contrast and Comparison of Modern Software Process Models International Conference on Advances in Management and Technology (iCAMT - 2013) Proceedings published in International Journal of Computer Applications® (IJCA) (0975 – 8887).

[33] Segi, S. (2007) “Comparing the Rational Unified Process and Microsoft Solutions Framework”. In: IBM Developerworks.<http://www.ibm.com/developerworks/rational/library/apr07/santos/index.html>

[Revisado 2014-05-05]

[34] Alvarado, R., Delgado, L., Dávila, A. “Mapeo y evaluación de la cobertura de los procesos de MPS.Br a los procesos de la categoría de Operación de MoProSoft” XI Simpósio Brasileiro de Qualidade de Software.

[35] Wen, L., Rout, T., Using Composition Trees to Validate an Entry Profile of Software Engineering Lifecycle Profiles for Very Small Entities (VSEs), 12th International Conference, SPICE 2012, May 29-31, Spain. pp. 38-50

[36] Dromey, R. G., System Composition: Constructive Support for the Analysis and De-sign of Large Systems, SETE-2005, Systems Engineering/Test and Evaluation Conference , Brisbane, Australia, 2005

[37] Dromey, R.G., “Formalizing the Transition from Requirements to Design”, in Mathematical Frameworks for Component Software, Models for Analysis and Synthesis, Chapter 6, World Scientific, 2006, Ed. Liu, Z., and He, J., ISBN 981-270-017-X, pp. 173-206

[38] Behavior Engineering Web Site, <http://www.behaviorengineering.org/>

- [39] Babar Muhammad; Zhang He; Systematic Literature Review in Software Engineering: Preliminary Results from Interviewers with Researches
- [40] B. Kitchenham and S. Charters ; Guidelines for performing Systematic Literature Reviews in software engineering; Keele University and Durham University joint report ; 2007
- [41] T. Dyba, T. Dingsoyr y G. K. Hanssen, «Applying systematic reviews to diverse study types: An experience report,» de Proceedings ESEM, 2007.
- [42] C. D. Mulrow, Systematic reviews: synthesis of best evidence for health care decisions, American College of Physicians, 1998.
- [43] J. A. Pow Sang Portillo, Técnicas para la Estimación y Planificación de Proyectos de Software con Ciclos de Vida Incremental y Paradigma Orientado a Objetos, 2012.
- [44] C.M.C. Santos, C.A.M Pimenta and M.R.C. ; Nobre, A estratégia PICOC para a construção da pergunta de pesquisa e busca de evidências. Revista Latino-Americana de Enfermagem, v.15, n.3, pp. 508-511,2007
- [45] R. Prikladnicki, S. Marczak, T. Conte, C. DE Souza, J. L. N. Audy, J. Kroll, A. B. Marques, and R. A. D. Orsoletta. A Evolução e o Impacto da Pesquisa em Desenvolvimento Distribuído de Software no Brasil. In: XXV Simpósio Brasileiro de Engenharia de Software, São Paulo, 2011.
- [46] Lachana Inchaiwong, A.S.M. Sajeev, A comparison of Software Process Models, SERP 2008 International Conference on Software Engineering Research and Practice.
- [47] Lachana Inchaiwong, A.S.M. Sajeev, Sakgasit Ramingwong, SAM: An analysis Method of Software Process Models, IASTED Conf. on Software Engineering.
- [48] C. Halvorsen, R. Conradi, A taxonomy for SPI Frameworks, Proc. 24th NASA Software Engineering 1999.

[49] C. Halvorsen, R. Conradi, A taxonomy to compare SPI Frameworks, Software Process Technology, 2001 – Springer

[50] M. Zarour, J. Desharnais and A. Abran, “A Framework to Compare Software Process Assessment Methods Dedicated to Small and Very Small Organizations”, International Conference on the Software Quality - ICSQ'07, (2007).

[51] M. C. Paulk "A taxonomy for improvement frameworks", presented at the World Congr. Softw. Qual., 2008