

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
ESCUELA DE POSGRADO



“Revisión sistemática de estudios realizados sobre comparaciones de los métodos de estimación de tamaño funcional IFPUG FPA y COSMIC sobre proyectos SOA”

Tesis para optar el grado de Magíster en Informática con mención en Ingeniería de Software

AUTOR

Javier Orlando Campos Maldonado

ASESOR

Dr. José Antonio Pow Sang Portillo

JURADO

Dr. José Antonio Pow Sang Portillo

Dr. Héctor Andrés Melgar Sasieta

Mg. Claudia María Del Pilar Zapata Del Rio

LIMA – PERÚ

2015

A Yshette y Camila...

A mis padres...

A mis hermanos...

A los Loras...

A mi Promo 90 STA...



Agradecimientos

La culminación de esta tesis no ha sido fácil pero agradezco a Dios y al apoyo de muchas personas que me alentaron a no rendirme y llegar hasta el final. Por ello, quiero expresar mi agradecimiento esas personas que de diferentes formas me apoyaron para lograr esta gran meta.

En primer lugar, quiero agradecer a mi asesor, José Antonio Pow Sang Portillo, por todas las sugerencias y observaciones realizadas a lo largo de todo este trabajo, pues me ayudaron a aclarar cuál era el objetivo que debía cubrir con las preguntas de investigación y así encaminar mi trabajo.

A mis padres y hermanos, por todo su apoyo. Gracias mamá por soportar mis amanecidas.

A Isaac Yrigoyen, por sus excelentes consejos en momentos difíciles de este trabajo.

A Iván Caldas, Jennifer Zárate, Geraldo Flores, Rita Álvarez y Emily De la Cruz, por siempre darme ánimo y apoyo para completar esta tesis. Son detalles que nunca olvidaré.

Finalmente, a mi esposa Yshette, por todo su amor y comprensión porque me lo demostraste con tu constante apoyo y me ayudaste con tu presión para que no me rinda y termine esta tesis.

RESUMEN

En la Ingeniería de Software, la estimación de proyectos es considerado un tema importante pues ayuda a la mejora del desarrollo del proyecto. Dentro de las diversas variables a estimar, tres son las más relevantes: el tamaño del software, el esfuerzo y el cronograma. Como la estimación de costos radica básicamente en estimar el tamaño de software así como la cantidad de personas necesarias para desarrollar el producto, se ha decidido centrar el estudio en la estimación del tamaño del software. Ahora, el tamaño de software puede ser cuantificado usando diferentes técnicas, como las líneas de código y los métodos de medición de tamaño funcional, etc. Nosotros nos centraremos en analizar los métodos IFPUG FPA y COSMIC.

Por esta razón, la presente tesis presentará una revisión sistemática de estudios realizados sobre comparaciones de los métodos de estimación de tamaño funcional IFPUG FPA y COSMIC sobre proyectos SOA.

El objetivo será el poder encontrar y analizar los diferentes trabajos que se han realizado para adaptar los métodos de estimación de tamaño funcional IFPUG FPA y COSMIC sobre proyectos SOA. Para lograr ello, se ha desarrollado esta tesis en seis capítulos. En el primero, se plantean las definiciones de los métodos de estimación IFPUG FPA y COSMIC, y el concepto SOA. En el segundo, se incluye la definición de una revisión sistemática así como los trabajos realizados de revisiones sistemáticas aplicadas a proyectos SOA. En el tercero, se presenta la planificación de la aplicación de la revisión sistemática donde se incluyen el desarrollo del protocolo, la formulación de las preguntas de investigación y la estrategia para la búsqueda. En el cuarto, se presenta la aplicación de la revisión sistemática. En el quinto, se presentan los resultados de la revisión, y en el último capítulo se incluyen las conclusiones y los trabajos futuros.

Palabras clave: SOA, IFPUG, COSMIC, Revisión sistemática

Índice de Figuras

Figura 1 Marco de Referencia SOA	3
Figura 2 Necesidad de sintetizar el conocimiento	5
Figura 3 Diferencias más importantes entre revisiones narrativa y sistemática	6

Índice de Tablas

Tabla 1 Criterios PICOC usados en la investigación	10
Tabla 2 Formato de tabla de presentación de documentación	10
Tabla 3 Formato de tabla de extracción de datos	11
Tabla 4 Criterios PICOC usados en el desarrollo de la revisión	12
Tabla 5 Cadenas generales básicas de búsqueda	13
Tabla 6 Cantidad de artículos encontrados en primera fase de búsqueda	14
Tabla 7 Cantidad de artículos encontrados en segunda fase de búsqueda	14
Tabla 8 Formato con información de casos primarios	15
Tabla 9 Resultado pregunta 1	16
Tabla 10 Artículos seleccionados como casos primarios por método de estimación	21
Tabla 11 Artículos encontrados con información relevante para la Pregunta 2(a)	22
Tabla 12 Resumen con información relevante de los artículos seleccionados para la Pregunta 2(a)	22
Tabla 13 Artículos encontrados con información relevante para la Pregunta 2(b)	24
Tabla 14 Resumen con información relevante de los artículos seleccionados para la Pregunta 2(b)	24
Tabla 15 Artículos encontrados con información relevante para la Pregunta 2(c)	26
Tabla 16 Resumen con información relevante de los artículos seleccionados para la Pregunta 2(c)	26

Tabla de contenido

Capítulo 1: Planteamiento del proyecto	1
1.1. Definición del problema	1
1.2. Objetivo General	3
1.3. Objetivos Específicos.....	3
1.4. Resultados esperados	4
1.5. Alcance y limitaciones	4
1.6. Metodología aplicada en el proyecto	4
1.7. Justificación	4
Capítulo 2: Marco Conceptual y Estado del Arte	5
2.1. Marco Conceptual.....	5
2.2. Revisión del estado del arte	6
2.3. Discusión sobre los resultados de la revisión del estado del arte	7
Capítulo 3: Planificación de la Revisión Sistemática	8
3.1. Planificación de la revisión	8
3.1.1 Identificación de la necesidad de la revisión.....	8
3.1.2 Desarrollo de un protocolo de revisión.....	8
3.1.3 Formulación de las preguntas de la investigación	9
3.1.4 Estrategia para la búsqueda y selección de fuentes	10
3.1.5 Selección de los estudios primarios	10
3.1.6 Extracción de datos.....	11
Capítulo 4: Aplicación de la Revisión Sistemática	12
4.1. Desarrollo de la revisión.....	12
4.1.1. Identificación de la necesidad de la revisión.....	12
4.1.2. Desarrollo de un protocolo de revisión.....	12
Capítulo 5: Resultados de la revisión	16
5.1 Resultados	16
5.1.1 Análisis de pregunta 1	16
5.1.2 Análisis de pregunta 2	21
Capítulo 6: Conclusiones y trabajos futuros.....	31
Referencias.....	32
Anexos.....	34



Capítulo 1: Planteamiento del proyecto

El presente capítulo tiene por finalidad exponer la problemática existente en el ámbito de los modelos de estimación de tamaño del software aplicado a proyectos SOA, el cual está teniendo una mayor presencia. Para esta tesis se ha optado por la aplicación de una revisión sistemática pues así podremos incrementar la posibilidad de detectar trabajos elaborados con técnicas de estimación y poder compararlos, y con ello ver la profundidad de la aplicación de estas técnicas en el campo de la Ingeniería del Software. Por ello, se escogió como referencia a las técnicas de estimación de tamaño funcional más usadas: IFPUG FPA y COSMIC.

1.1. Definición del problema

En la Ingeniería de Software, la estimación de proyectos es considerado un tema importante pues ayuda a la mejora del desarrollo del proyecto. Dentro de las diversas variables a estimar, tres son las más relevantes: el tamaño del software, el esfuerzo y el cronograma. Según [27], en esta disciplina de la Ingeniería, la estimación de costos radica básicamente en estimar el tamaño de software así como la cantidad de personas necesarias para desarrollar el producto, por ello, se ha decidido centrar el estudio en la estimación del tamaño del software.

Dentro de los proyectos software existentes tenemos a los orientados a servicios. Estos proyectos fueron mencionados por primera vez en 1996 en dos artículos en Gartner Inc. ("Service Oriented" Architectures, Part 1 y "Service Oriented" Architectures, Part 2), pero recién en el 2003, cuando se publica en [28] y se la introduce en el entorno empresarial mediante los servicios web. Dada su reciente aparición, se desea conocer cómo se ha intentado aplicar la estimación del tamaño de software a estos tipos de proyectos.

Según la Real Academia Española [2], la palabra medición significa acción y efecto de medir. Medir no es otra cosa que comparar una cantidad con su respectiva unidad, con el fin de averiguar cuántas veces la segunda está contenida en la primera. Por lo tanto, la medición debe entenderse como el proceso de obtener información experimentalmente acerca de la magnitud de una cantidad (ISO/IEC 2, 2004).

Ahora, el tamaño de software puede ser cuantificado usando diferentes técnicas, como las líneas de código y los métodos de medición de tamaño funcional, etc. Para el análisis de este proyecto usaremos los métodos: IFPUG FPA (International Function Point Users Group) y COSMIC (Common Software Measurement International Consortium).

El método COSMIC nos proporciona un método estandarizado de medición del tamaño funcional de software para los dominios funcionales comúnmente denominados como Software de Aplicaciones de Negocio (ó *Management Information Systems, MIS*) y Software de Tiempo Real. Para la medición usaremos la adaptación propuesta en el documento *Guideline for sizing Service-Oriented Architecture Software versión 1.0a* [25]. Esta guía explica cómo se debe aplicar COSMIC a un tamaño de software llamado servicio, el cual pertenece a SOA. Además, en esta se define y estandariza los conceptos particulares como capas, componentes de pares, el tamaño ilimitado de un proceso funcional, y las piezas de software que pueden ser usadas entre sí. Finalmente, aplica las 3 fases del método COSMIC pero explicando cómo se realizará la separación de componentes y cómo los medirá usando los Functional User Requirements (FUR).

Acerca del método IFPUG FPA, hemos tomado de su sitio web (<http://www.ifpug.org/>), la siguiente descripción que la explica: “la técnica de FPA cuantifica las funciones contenidas en el software en términos que tengan sentido para los usuarios de software. La medida está directamente relacionada con los requerimientos del negocio que el software está destinada a hacer frente. Por lo tanto, se puede aplicar fácilmente a través de una amplia gama de entornos de desarrollo y durante la vida de un proyecto de desarrollo, a partir de principios de los requisitos de la definición de uso operativo completo. Otras medidas comerciales, como la productividad del proceso de desarrollo y el costo por unidad para apoyar el software, también pueden ser fácilmente derivadas. La medida en puntos función en sí se deriva en una serie de etapas. El uso de un conjunto estandarizado de criterios básicos, cada una de las funciones de la empresa es un índice numérico de acuerdo a su tipo y complejidad. Estos índices se suman para dar una primera medida de tamaño que luego se normalizó mediante la incorporación de una serie de factores relacionados con el software en su conjunto. El resultado final es un número único llamado el índice de punto de función que mide el tamaño y la complejidad del producto de software”.

Como mencionamos al inicio, esta evaluación se realizará sobre proyectos orientados a servicios, así que indicaremos en qué consiste su arquitectura.

La Arquitectura Orientada a Servicios (en inglés *Service Oriented Architecture SOA*), es un concepto de arquitectura de software que define la utilización de servicios para dar soporte a los requisitos del negocio.

Permite la creación de sistemas altamente escalables que reflejan el negocio de la organización, a su vez brinda una forma bien definida de exposición e invocación de servicios (comúnmente pero no exclusivamente servicios web), lo cual facilita la interacción entre diferentes sistemas propios o de terceros.

SOA define las siguientes capas de software:

- Aplicaciones básicas - Sistemas desarrollados bajo cualquier arquitectura o tecnología, geográficamente dispersos y bajo cualquier figura de propiedad;
- De exposición de funcionalidades - Donde las funcionalidades de la capa aplicativa son expuestas en forma de servicios (servicios web);

- De integración de servicios - Facilitan el intercambio de datos entre elementos de la capa aplicativa orientada a procesos empresariales internos o en colaboración;
- De composición de procesos - Que define el proceso en términos del negocio y sus necesidades, y que varía en función del negocio;
- De entrega - donde los servicios son desplegados a los usuarios finales.

SOA proporciona una metodología y un marco de trabajo para documentar las capacidades de negocio y puede dar soporte a las actividades de integración y consolidación.

Modelo de Referencia basado en Arquitectura SOA

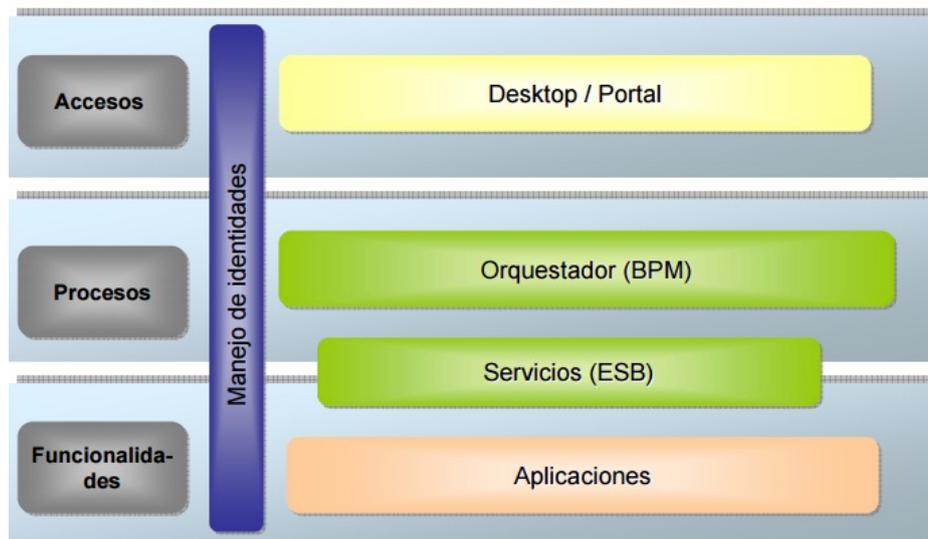


Figura 1. Marco de Referencia SOA [29]

1.2. Objetivo General

Poder encontrar y analizar los diferentes trabajos que se han realizado para adaptar los métodos de estimación de tamaño funcional IFPUG FPA y COSMIC sobre proyectos SOA.

1.3. Objetivos Específicos

- Determinar la existencia de estudios sobre la aplicabilidad del método de estimación de tamaño funcional IFPUG FPA sobre proyectos SOA.
- Determinar la existencia de estudios sobre la aplicabilidad del método de estimación de tamaño funcional COSMIC sobre proyectos SOA.

1.4. Resultados esperados

Los resultados esperados de acuerdo a los objetivos específicos son los siguientes:

- Encontrar evidencia considerable que permita evaluar la aplicación de la técnica COSMIC en proyectos que empleen SOA.
- Encontrar evidencia considerable que permita evaluar la aplicación de la técnica IFPUG FPA en proyectos que emplee SOA.

1.5. Alcance y limitaciones

Esta tesis busca aplicar una revisión sistemática cuantitativa sobre trabajos que muestren técnicas de estimación de software para proyectos que emplean SOA.

Las limitaciones y alcances serán definidos en el protocolo de la revisión sistemática que se aplicará en este estudio.

1.6. Metodología aplicada en el proyecto

Para la realización de esta tesis se aplicará la metodología de aplicación de una revisión sistemática detallada en el capítulo 3.

1.7. Justificación

Existen varios métodos de estimación de tamaño funcional que se han aplicado a diversos tipos de proyectos de Ingeniería de software, pero hay muy pocos trabajos sobre su aplicación sobre proyectos SOA. Se ha podido encontrar trabajos sobre proyectos SOA que se enfocan en analizar los modelos de referencia orientados a servicios, la arquitectura orientada a servicios y solo algunos en pruebas de software, pero no relacionados directamente con métodos de estimación de tamaño funcional. Lo que se espera con esta tesis es poder mostrar la utilidad que una revisión sistemática da para encontrar una mayor información, y de mejor calidad, sobre trabajos realizados sobre el tema indicado.

Capítulo 2: Marco Conceptual y Estado del Arte

El presente capítulo tiene por finalidad presentar todos los términos y conceptos que serán usados a lo largo del desarrollo de la revisión sistemática; además, en el estado del arte, se presentará algunas revisiones sistemáticas que se hayan realizado. Finalmente, se realizará una breve discusión de los resultados obtenidos en la investigación del estado del arte.

2.1. Marco Conceptual

Empezaremos por definir qué es una revisión sistemática.

En [11] se nos presenta una definición bastante práctica y clara sobre ella. Una revisión sistemática de la literatura permite identificar, evaluar, interpretar y sintetizar todas las investigaciones existentes y relevantes en un tema particular de interés particular. Inicialmente las revisiones sistemáticas fueron muy aplicadas en el ámbito de la medicina.

En [16] se nos menciona que hay 2 tipos de revisiones que se aplican en la investigación por revisión de publicaciones: sistemática y no sistemática o “revisión narrativa”. La diferencia entre revisión sistemática y no sistemática está dada porque en esta última falta un protocolo que defina los pasos que siguió el revisor para actualizar el tópico. A su vez dentro de la revisión sistemática existen dos formas: “cuantitativa o meta-análisis” y “cualitativa”. Las diferencias están dadas fundamentalmente por el uso de métodos estadísticos, en el meta-análisis, que permiten la combinación y análisis cuantitativos de los resultados obtenidos en cada estudio.

La figura 2 tomada de [18] nos muestra lo que busca cada tipo de revisión.

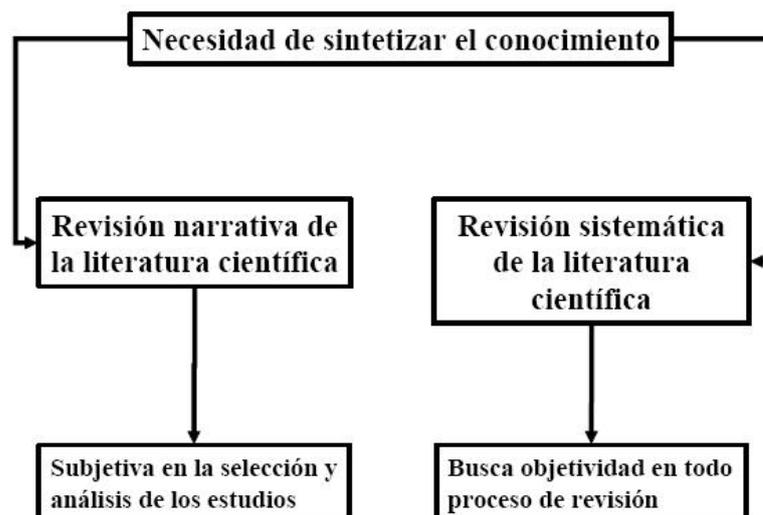


Figura 2. Necesidad de sintetizar el conocimiento

La figura 3 tomada de [18] nos muestra las diferencias más importantes entre estas dos revisiones.

Aspecto	Revisión narrativa	Revisión sistemática
Problema	Amplio	Específico
Fuentes y búsqueda	Generalmente no se especifica	Accesible y estrategia de búsqueda explícita
Selección	Generalmente no se especifica	Criterios definidos Uniforme
Análisis crítico	Variable	Riguroso
Síntesis	Generalmente un resumen cualitativo	Resumen cuantitativo*
Inferencia	Algunas veces basada en evidencias	Siempre basada en evidencias

Figura 3. Diferencias más importantes entre revisiones narrativa y sistemática

De lo presentado en las figuras 2 y 3, se puede concluir que para la búsqueda de información más cercana a la ingeniería, las revisiones sistemáticas son más aplicables pues buscan encontrar un problema concreto y sigue un método establecido haciendo uso de evidencias comprobables.

Luego, Barbara Kitchenham en [10] presentó un método para realizar revisiones sistemáticas en la Ingeniería del Software, el cual será en el que se base la revisión de esta tesis.

Este método propone 3 fases: Planeamiento de la revisión, Desarrollo o ejecución de la revisión y la fase de publicación de los resultados de la revisión y su divulgación.

El detalle de este método será expuesto en el capítulo 3.

2.2. Revisión del estado del arte

Acerca de trabajos realizados de revisiones sistemáticas aplicadas a proyectos SOA, para esta tesis se decidió solo tomar en cuenta las que se refieren a pruebas de software, por ello, estamos considerando solo las siguientes:

- En [19], una revisión cuyo objetivo era realizar un estudio del estado actual de la investigación en pruebas de software para arquitecturas orientadas a servicio, considerando de manera específica el uso de UML como lenguaje de modelado.
- En [20], una revisión cuyo objetivo era identificar y clasificar los trabajos existentes en el campo de las pruebas en SOAs con enlace dinámico, se ha elaborado un protocolo para guiar el desarrollo de revisiones literarias

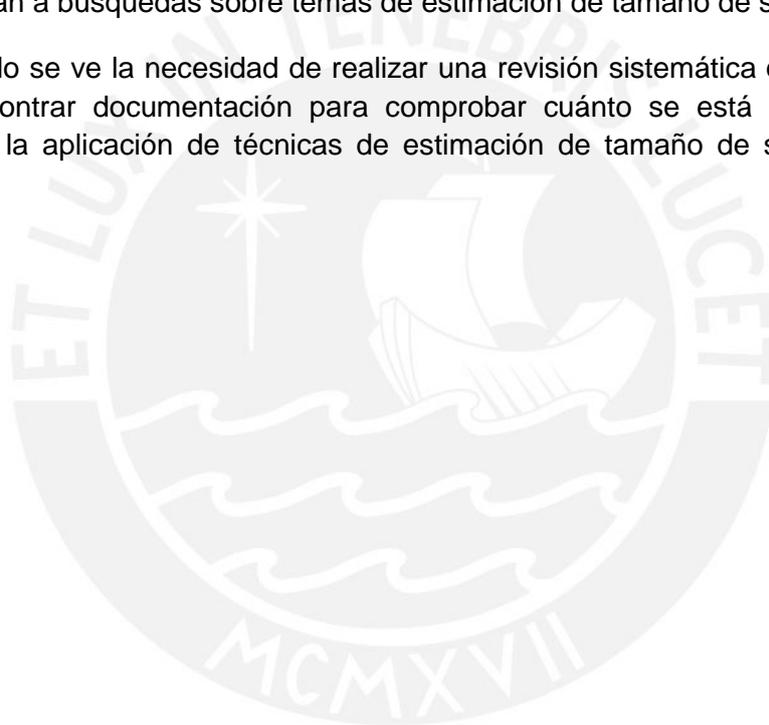
sistemáticas. En el protocolo se especifican tanto las estrategias de búsqueda y selección de los artículos como la posterior extracción y síntesis de la información recopilada. En este artículo se describen todos los elementos del protocolo desarrollado y un resumen de los resultados obtenidos tras la revisión sistemática.

- En [11], una revisión sistemática de la literatura sobre los esfuerzos llevados a cabo en Pymes desarrolladoras de software relacionadas con la mejora de procesos. Lo que buscaba este artículo lo realizado y logrado en este tipo de empresas respecto a la mejora de procesos software.

2.3. Discusión sobre los resultados de la revisión del estado del arte

Al realizar la investigación del estado del arte, obtuvimos como resultado de que existen algunos trabajos sobre revisiones sistemáticas aplicadas a proyectos SOA o algunos otros tipos de procesos de software pero no se orientan a búsquedas sobre temas de estimación de tamaño de software.

Por ello se ve la necesidad de realizar una revisión sistemática que nos ayude a encontrar documentación para comprobar cuánto se está avanzando en afinar la aplicación de técnicas de estimación de tamaño de software sobre SOA.



Capítulo 3: Planificación de la Revisión Sistemática

En este capítulo se presenta de forma detallada como se realizará la aplicación de la revisión sistemática.

Este proceso está basado en la presentación realizada por Kitchenham en [12]. En dicho proceso se han considerado los siguientes puntos: la identificación de la necesidad de la revisión, el desarrollo del protocolo de la revisión, la formulación de las preguntas de investigación y la estrategia para la búsqueda y selección de fuentes.

3.1. Planificación de la revisión

En esta etapa se realizará planificará la revisión sistemática. Tomando como referencia lo indicado en [10], los investigadores deben identificar y revisar cualquier revisión sistemática existente del fenómeno de interés, pues luego se deberá identificar cuáles serán relevantes para el estudio.

3.1.1 Identificación de la necesidad de la revisión

En este apartado se debe identificar la relevancia de la revisión. Para ello, el documento [10] nos sugiere algunas preguntas:

- ¿Cuáles son los objetivos de la revisión?
- ¿Qué fuentes se investigaron para identificar estudios primarios?
¿Hubo alguna restricción?
- ¿Cuáles fueron los criterios de inclusión / exclusión y cómo se aplican?
- ¿Qué criterios se utilizaron para evaluar la calidad de los estudios primarios y cómo fueron aplicados?
- ¿Cómo se extrajeron los datos de los estudios primarios?

3.1.2 Desarrollo de un protocolo de revisión

En este apartado se debe definir un protocolo para la revisión sistemática, la cual especificará los métodos que se utilizarán para llevarla a cabo.

Los componentes de un protocolo deben incluir todos los elementos de la revisión además de alguna información adicional de planificación:

- Los antecedentes para la justificación de la encuesta.
- Las preguntas de investigación que la revisión realizará. Ellas deben ser acorde con toda la metodología de revisión sistemática. La estructura de las preguntas

- La estrategia que se utilizará para buscar los estudios primarios incluidos en los términos de la búsqueda, así como los recursos que se deberán buscar. Estos recursos considerarán bases de datos, revistas específicas, actas de congresos, etc.
- Los criterios y procedimientos de selección de los estudios. Estos criterios determinarán criterios para la inclusión o exclusión de un estudio de la revisión sistemática. Se describirá cómo se aplicarán los criterios para la aceptación de cada estudio primario.
- La estrategia de extracción de datos. Esta debe definir cómo se obtendrá la información requerida de cada estudio primario.

3.1.3 Formulación de las preguntas de la investigación

En este apartado se debe especificar las preguntas de investigación es la parte más importante de cualquier revisión sistemática.

Las preguntas revisadas deben ser acorde con toda la metodología de revisión sistemática y deben ser claras para la búsqueda que debe identificar los estudios primarios que se ocupan de las cuestiones de investigación.

Para nuestro trabajo, las preguntas definidas son las siguientes:

- Pregunta 1: ¿Qué evidencia existe acerca de estudios realizados sobre los métodos de estimación de tamaño funcional IFPUG FPA y COSMIC sobre proyectos SOA?
- Pregunta 2: ¿Qué técnicas o métodos se han realizado para estimar el tamaño funcional de proyectos aplicados a Web o a SOA?
 - Pregunta 2(a): ¿Qué técnicas o métodos se han realizado para las conversiones entre IFPUG y COSMIC para medir proyectos SOA o similares?
 - Pregunta 2(b): ¿Qué técnicas o métodos se han realizado para las comparaciones entre IFPUG y COSMIC para medir proyectos SOA o similares?
 - Pregunta 2(c): ¿Qué otras técnicas o modelos que propongan métodos de estimación para proyectos SOA existen?

En nuestro caso, aplicando lo que Petticrew y Roberts llaman PICOC (Población, Intervención, Comparación, Resultado, Contexto), solo usaremos los dos primeros criterios pues solo se desea conocer un grupo específico (proyectos SOA) y qué técnicas de estimación de tamaño funcional se aplican al grupo específico.

Tabla 1. Criterios PICOC usados en la investigación

Criterio	Descripción
Population (población)	Proyectos SOA (Service oriented architecture)
Intervention (intervención)	Métodos, técnicas IFPUG o COSMIC

3.1.4 Estrategia para la búsqueda y selección de fuentes

Es necesario determinar y seguir una estrategia de búsqueda, ella seguirá lo siguiente:

- Búsquedas iniciales utilizando varias combinaciones de los términos de búsqueda se derivan de la pregunta de investigación.
- Comprobación de las cadenas de investigación con las listas de los estudios primarios ya conocidos.

Algunas de las estrategias de búsqueda se aplicarán en lo siguiente:

- Artículos en bibliotecas digitales
- Actas de congresos o revistas (Journal)
- Digitalización de la literatura gris

La documentación de la búsqueda se presentará en tablas.

A continuación mostramos un formato de las tablas que se usarán para presentar la documentación de las fuentes encontradas.

Tabla 2. Formato de tabla de presentación de documentación

Nombre	
Fuente	
Fecha	
Autores	
Resultados objetivos del estudio	
Metodología	
Resultados	
Resultados subjetivos	

3.1.5 Selección de los estudios primarios

Los criterios de selección del estudio identificarán los estudios primarios que proporcionan pruebas directas acerca de las preguntas de investigación. Los criterios de inclusión y exclusión se basarán también en la misma pregunta.

En nuestro caso, la primera etapa de búsqueda será sobre la información con términos básicos implicados con la búsqueda, y ya en la segunda etapa, con información refinada obtenida en la primera fase, se localizará información más específica, para mejorar las conclusiones.

3.1.6 Extracción de datos

En esta etapa se realiza el diseño de los formularios de extracción de datos para registrar la información obtenida de los estudios primarios.

Los formularios de extracción de datos serán diseñados para recoger toda la información necesaria para poder responder a las preguntas de la revisión. Además se tendrá en cuenta no repetir fuentes encontradas en diversas bases de datos.

A continuación presentamos un formato de tabla a usar para la presentación de la extracción de datos:

Tabla 3. Formato de tabla de extracción de datos

Código	Autores	Criterio1	Criterio2	Criterio3

Capítulo 4: Aplicación de la Revisión Sistemática

En este capítulo aplicaremos la revisión sistemática de estudios realizados sobre comparaciones de los métodos de estimación de tamaño funcional IFPUG FPA y COSMIC sobre proyectos SOA. En dicha revisión se presenta las preguntas de investigación, las cadenas de búsqueda usadas y la selección de los estudios primarios con la cantidad de artículos encontrados en la primera y segunda fase de búsqueda.

4.1. Desarrollo de la revisión

4.1.1. Identificación de la necesidad de la revisión

Luego del análisis previo vemos necesario realizar esta revisión pues necesitamos encontrar la evidencia existente al estudio así como identificar nuevas fuentes en la investigación actual con el objetivo que se continúen en trabajos futuros.

4.1.2. Desarrollo de un protocolo de revisión

En esta etapa desarrollaremos el protocolo de revisión. Para ello nos apoyaremos en [21].

4.1.2.1 Formulación de la pregunta

La pregunta de investigación inicialmente definida fue: ¿Qué evidencia existe acerca de estudios realizados sobre los métodos de estimación de tamaño funcional IFPUG FPA y COSMIC sobre proyectos SOA?

Los términos básicos que usaremos serán: estimation method, IFPUG, COSMIC, SOA, Service oriented architecture

A continuación presentamos una tabla que muestre los términos de acuerdo a PICOC que usaremos.

Tabla 4. Criterios PICOC usados en el desarrollo de la revisión

Criterio	Descripción
Population (población)	SOA, Service oriented architecture, Software measurement
Intervention (intervención)	Estimation, method, IFPUG, COSMIC

Los resultados que esperábamos encontrar eran: aplicaciones de cada método sobre proyectos SOA.

4.1.2.2 Selección de las fuentes

El proceso de búsqueda de información se realizó en dos fases: en la primera se optó por buscar información con términos básicos implicados con la búsqueda, y ya en la segunda, con información refinada obtenida en la primera fase, se decidió por localizar información más específica, para mejorar las conclusiones y hacer un mejor estudio secundario.

De acuerdo a los términos indicados se crearon las siguientes cadenas mostrados en la tabla 5:

Tabla 5. Cadenas generales básicas de búsqueda

	Cadenas generales básicas de búsqueda	Base de datos
1	estimation soa	SCOPUS
2	(((((("SOA" OR "Service oriented architecture") and (("Software measurement") or ("estimation method")) And ("cosmic fpp")OR ("ifpug")))))	ACM Digital Library
3	("estimation method") and (IFPUG or COSMIC) and (soa)	IEEE/IET Electronic Library (IEL)
4	(estimation method) AND Título=(cosmic or ifpug)	Web of Science

La lista de fuentes sobre las cuales se realizaron las búsquedas son:

- SCOPUS
- ACM Digital Library
- IEEE/IET Electronic Library (IEL)
- Web of Science

Esta primera etapa de búsqueda se realizó usando los términos básicos mencionados. Es aquí donde notamos que algunos de los términos usados también se confundían con el de otras ramas de la ingeniería.

Por ello, en la segunda etapa, la consulta se realizó teniendo como restricción solo revisar el campo de estudio referente al ámbito de la ingeniería de software y a la ciencia de la computación.

4.1.2.3 Selección de los estudios primarios

Respecto a los criterios de inclusión de los estudios, se decidió que solo se tomarían en cuenta el título, abstract y KeyWords de los artículos obtenidos en la búsqueda.

Para definir los tópicos para la selección de los casos primarios, se han considerado los siguientes:

- Método de estimación de tamaño funcional IFPUG sobre proyectos SOA.
- Método de estimación de tamaño funcional COSMIC sobre proyectos SOA.
- Propone comparaciones entre los métodos de tamaño funcional IFPUG y COSMIC.
- Propone métodos de estimación para proyectos SOA.

A continuación presentamos 2 tablas con las cantidades de artículos encontrados por base de datos en la primera y segunda fase de búsqueda.

Tabla 6. Cantidad de artículos encontrados en primera fase de búsqueda

Base de datos	Cantidad de artículos encontrados
SCOPUS	211
IEEE/IET Electronic Library (IEL)	19
Web of Science	135
ACM Digital Library	285

Tabla 7. Cantidad de artículos encontrados en segunda fase de búsqueda

Base de datos	Cantidad de artículos encontrados
SCOPUS	73
IEEE/IET Electronic Library (IEL)	12
Web of Science	48
ACM Digital Library	80

De lo mostrado podemos mencionar que en la primera fase de búsqueda se encontraron 650 artículos.

Y en la segunda fase búsqueda, acotando a solo campos del área de ingeniería de software y ciencias de la computación se encontraron 213 artículos, 38 de los cuales eran artículos repetidos.

Luego de ambas búsquedas hemos seleccionado a 21 artículos, los que consideraremos como casos primarios.

4.1.2.4 Extracción de datos

Los resultados de la extracción de cada uno de los casos primarios se muestran en una tabla que tendrá la siguiente estructura:

Tabla 8. Formato con información de casos primarios

Nombre	
Fuente	
Fecha	
Autores	
Resultados objetivos del estudio	
Metodología	
Resultados	
Resultados subjetivos	

Toda la información de cada fuente usada como caso primario se encuentra detallada en el Anexo.



Capítulo 5: Resultados de la revisión

5.1 Resultados

A partir de la información extraída de los estudios realizó un análisis para poder mostrar los datos importantes de la revisión sistemática hecha.

En dicho análisis se presenta, por cada pregunta de investigación, las tablas con los artículos encontrados así como el análisis realizado con dichos artículos.

5.1.1 Análisis de pregunta 1

Pregunta 1: ¿Qué evidencia existe acerca de estudios realizados sobre los métodos de estimación de tamaño funcional IFPUG FPA y COSMIC sobre proyectos SOA?

Luego de la búsqueda se ha encontrado cuatro artículos que menciona algunos puntos a tomar en cuenta de ambas técnicas respecto de proyectos SOA, aunque más orientados a COSMIC.

La siguiente tabla indica los datos de dichos artículos:

Tabla 9. Resultado pregunta 1

Nro	Autor	Título	Tipo de publicación	Año
P1	Santillo, Luca	Seizing and sizing SOA applications with COSMIC Function Points	Conferencia	2007
P2	Rivas Germán; Quintana Mauricio	Estimation in Service Oriented Architecture	Paper	2010
P3	Jayakumar K R	WHY YOU MUST CHANGE TO COSMIC FOR SIZING AND ESTIMATION	Paper	2010
P4	Ashish Seth, Himanshu Agarwal, Ashim Raj Singla	Techniques for Evaluating Service Oriented Systems: A Comparative Study	Journal	2014

En P1 se nos presenta de forma detallada los principios de SOA, su tipo de servicio y niveles, el modelo de referencia de SOA empresarial y la integración con el software, y los Servicios Web, también llamados Web Services. Luego, presenta los métodos IFPUG y COSMIC, indicando las posibles ventajas o desventajas que podrían tener ambos métodos con este tipo de proyectos.

Por ejemplo indica que el límite de tamaño que tiene IFPUG complica la estimación, además no toma en cuenta la reutilización de software, y que ningún proceso interno puede ser identificado y medido, salvo que otro proceso le envíe o reciba datos desde fuera del límite del sistema. En cambio menciona que COSMIC podría adaptarse mejor a SOA pues se puede aplicar a cualquier arquitectura en una capa o en múltiples capas, incluso dentro de una capa.

Por ello, el autor, al presentar la forma de dividir el software para su medición indica que mientras el método IFPUG, por su forma de conteo, no ofrece soluciones específicas para este tipo de proyectos, el método COSMIC podría ajustar mejor su medición a proyectos SOA. Lamentablemente las opiniones vertidas no tienen un caso de estudio de apoyo a sus conjeturas.

En P2 se presentan 2 métodos para estimar proyectos SOA. Estos son: SMART y COSMIC.

Acerca de SMART, este documento basa su análisis en [26].

Este método fue desarrollado por el Software Engineering Institute (SEI) para ayudar a las organizaciones en el análisis de las capacidades heredadas que serán usadas como servicios en una SOA.

Acerca del método COSMIC, este documento se basa en [25], en el cual se presenta la métrica completa, más no tiene la adaptación más reciente aplicada a los proyectos SOA.

Asimismo como añadido al documento, se presenta la información general del método Linthicum, el cual está basado en [24].

En P3 se presenta los beneficios del método COSMIC como una mejor alternativa en la forma de estimar proyectos de software, entre ellos a los proyectos SOA.

Se presentan como ventajas las siguientes:

- Aplicable para todo tipo de proyectos
- Produce resultados estadísticamente significativos
- Se puede aplicar desde el inicio del ciclo de vida de desarrollo
- Es compatible con las últimas metodologías de ingeniería de software - OOAD, UML, casos de uso, y otras metodologías tradicionales de desarrollo
- COSMIC puede ser utilizado bajo un enfoque ágil para el desarrollo de software usando "Scrum" o desarrollo basado en pruebas
- Se puede utilizar en aplicaciones Datawarehouse
- Se puede utilizar en aplicaciones de arquitectura orientada a servicios (SOA), con lo cual se estaría preparando para poder estimar en un futuro aplicaciones en la nube.

Además, según el documento, COSMIC se diferencia de otros métodos de estimación por lo siguiente:

- Totalmente de código abierto. Sus manuales, directrices, casos de estudio y documentos son de libre acceso desde el portal web.
- Es el más simple entre los métodos existentes para la medición de tamaño funcional y es aplicable a todos los tipos de software.

- Fue el primer método que se hizo una norma ISO (IEC 19761), y es el único método 100% compatible con la norma ISO 14143, la cual es un estándar para los métodos de tamaño funcional.

- ISBSG publica informes de punto de referencia en base a las mediciones de tamaño COSMIC.

- COSMIC está siendo adoptado cada vez más por muchos proveedores de software de todo el mundo, incluyendo empresas de software de la India.

Al igual que los documentos anteriores, también menciona que hay poca información relacionada con la estimación de proyectos SOA.

En P4 se parte de la premisa que a pesar haber muchos métodos y herramientas para evaluar los sistemas de software, lamentablemente no funcionan bien para los sistemas que tienen servicios. Una de las grandes dificultades tiene que ver los servicios pues sólo son interfaces. La falta de acceso al código fuente hace muy complicado el poder medir tanto la estructura del código como el flujo de datos.

Por ello, se analizan distintas herramientas de pruebas y métodos de evaluación disponibles para aplicaciones basados en servicios, resume sus limitaciones y beneficios a su aplicación al contexto de las arquitecturas orientadas a servicios.

De lo más relevante en el documento está la parte III: Trabajo relacionado a la evaluación de costos para sistemas SOA.

Aquí se mencionan los siguientes métodos:

- Fórmula de Dave Linthicum

Él propone la siguiente fórmula para analizar el costo de un proyecto SOA:

$$\text{Costo de SOA} = (\text{Costo de la complejidad de los Datos}) + (\text{Costo de la complejidad del Servicio}) + (\text{Costo de la complejidad del Proceso}) + (\text{Soluciones para habilitar la tecnología})$$

Se espera que haya una variación del 10% al 20% en el costo.

Como limitación se indica que este enfoque no es una métrica exactamente.

- Modelo Cocomo II

Este modelo tiene un gran número de coeficientes, tales como multiplicadores de esfuerzo y los factores de escala.

COCOMO II considera dos tipos de componentes reutilizables, los de caja negra y de caja blanca.

Sin embargo, este modelo es insuficiente para poder cubrir todas las necesidades de estimación de costo y esfuerzo que se necesitan para un proyecto SOA.

- Modelo IFPUG

IFPUG propone matrices de rango simple para la evaluación de costos. Sin embargo, las medidas IFPUG conducen a cantidades similares para unidades diferentes de software.

- Modelo COSMIC

Este modelo COSMIC ofrece un rango de escalas amplio con la finalidad de tomar en cuenta la mayor cantidad de funciones de alta complejidad.

Este enfoque supone que cubrirá los tamaños típicos de todos los aspectos de un software basado en SOA.

Sin embargo, al tener un amplio marco de directrices para la aplicación práctica de la medición, debe seguirse probando y experimentando.

- Modelo SLOC

Este modelo está basado en punto de función (IFPUG / COSMIC) y las líneas de código fuente.

El punto de función mide el tamaño del software a través de la cuantificación de la cantidad de funcionalidad que proporciona al usuario en términos de la cantidad de entradas, salidas, consultas y archivos.

Sin embargo, el esfuerzo para usar el código y datos heredados para trabajar con los servicios no pueden ser asignados a cualquier tamaño funcional y además aún faltan directrices para la aplicación práctica de COSMIC en un contexto SOA.

- Método Liu

Este método considera que la estimación del tamaño de software se basa en la suma de los tamaños de cada servicio ($Tamaño = \sum (P_i * P)$) donde P_i es un factor cuyo valor es empírico y está relacionado con la infraestructura y tecnología del software.

En este método el tipo de servicio es muy determinante y el valor de P representa el tamaño estimado de un único servicio específico que varía respecto del tipo de servicios que se analice.

- Framework SMAT-AUS

Este framework considera que una aplicación genérica SOA podría comprenderse como una combinación de tipos de proyectos, dividiendo el proyecto en partes más manejables.

Además considera que toda aplicación SOA se clasifican en uno de los siguientes tipos de proyectos SOA: Servicio Minería, Servicio de Desarrollo, Integración de Servicios y Desarrollo de aplicaciones.

Pero tanto la especificación de cómo todas estas partes se estiman y el procedimiento necesario para la estimación práctica de los costes de desarrollo de software para sistemas basados en SOA todavía se está desarrollando.

- Método SMART (Service-Oriented Migration and Reuse Technique)

Este método puede ser adoptado para la estimación de costos de Service Mining

Algunos costos de estimación relacionadas con los tipos de proyectos se puede encontrar a través de una revisión de la literatura.

Lamentablemente, se considera que no hay otros sistemas de medición adecuados para los diferentes proyectos que el framework SMAT-AUS.

- Método GQM

Este método supone que SOA sigue diferentes objetivos en diferentes niveles de abstracción EA y que es posible identificar ciertos tipos de proyectos SOA y ciertos factores de contexto que se pueden combinar a las situaciones.

Sin embargo la identificación de los tipos de proyectos y los factores contextuales no son claros.

- Método Divide-and-Conquer (D&C)

Este método plantea descomponer de forma recursiva el problema en subproblemas más pequeños hasta que todos los subproblemas sean lo suficientemente simples, y en ese momento darle solución a los subproblemas. Finalmente, las soluciones resultantes se recomponen para formar la solución global del problema.

Este método supone que la estimación de costos por el desarrollo de software basada en SOA puede ser separada en áreas más pequeñas con las métricas correspondientes.

Sin embargo este método se concentra principalmente en la estimación de costos para la integración de servicios, y esto complica la estimación pues la clasificación de servicio es distinta dependiendo de su propósito, por lo cual, no hay una forma estándar para clasificar los servicios.

- Enfoque Work Breakdown Structure (WBS)

Este enfoque está basado en el principio en D&C y lo que permite es simplificar la complejidad de la estimación de costos de SOA.

Mediante la conmutación de diferentes tipos de métricas, este enfoque supone que podría satisfacer las diferentes necesidades de estimación de costos de software para proyectos SOA, sin embargo no se explican claramente cómo se usarían las métricas.

Esta comparación realizada para proyectos SOA es un gran alcance para poder analizar tanto lo que se está trabajando en la estimación de costo y esfuerzo en estos proyectos, como también en la gran dificultad que se tiene la adecuación debido a las diferentes arquitecturas y la complejidad de la implementación de los servicios.

Ahora, debido a no obtener suficiente información para la pregunta inicial de la búsqueda se decidieron generar una segunda pregunta, la cual fuera respondida, por los casos primarios seleccionados.

5.1.2 Análisis de pregunta 2

Pregunta 2: ¿Qué técnicas o métodos se han realizado para estimar el tamaño funcional de proyectos aplicados a Web o a SOA?

Se escogió aplicaciones Web pues debido a su separación de capas puede tener una semejanza en lo que se espera que se analice con SOA.

De acuerdo a esta pregunta, se generaron las siguientes subpreguntas:

- Pregunta 2(a): ¿Qué técnicas o métodos se han realizado para las conversiones entre IFPUG y COSMIC para medir proyectos SOA o similares?
- Pregunta 2(b): ¿Qué técnicas o métodos se han realizado para las comparaciones entre IFPUG y COSMIC para medir proyectos SOA o similares?
- Pregunta 2(c): ¿Qué otras técnicas o modelos que propongan métodos de estimación para proyectos SOA existen?

En primer lugar presentaremos una tabla que nos muestra la información obtenida de los casos primarios encontrados.

Tabla 10. Artículos seleccionados como casos primarios por método de estimación

Método de estimación	Tipo de proyectos		
	SOA	Web	Otros
IFPUG		2	
COSMIC		1	
Comparaciones IFPUG-COSMIC			7
Otros	2		5

A continuación, presentaremos los resultados obtenidos para cada subpregunta de la pregunta 2:

- Pregunta 2(a): ¿Qué técnicas o métodos se han realizado para las conversiones entre IFPUG y COSMIC para medir proyectos SOA o similares?

La siguiente tabla nos muestra los artículos encontrados que tienen información relevante respecto de este punto:

Tabla 11. Artículos encontrados con información relevante para la Pregunta 2(a)

Nro	Autores	Título	Tipo de publicación	Año
A1	Rabbi, M.F. ; Natraj, S. ; Kazeem, O.B.	Evaluation of Convertibility Issues between IFPUG and COSMIC Function Points	Conferencia	2009
A2	Gencel, C. ; Bideau, C.	Exploring the Convertibility between IFPUG and COSMIC Function Points: Preliminary Findings	Conferencia	2012
A3	Cuadrado-Gallego, J.J. ; Buglione, L. ; Rejas-Muslera, R.J. ; Machado-Piriz, F.	IFPUG-COSMIC Statistical Conversion	Conferencia	2008
A4	Lavazza, L. ; del Bianco, V. ; Geng Liu	Analytical Convertibility of Functional Size Measures: A Tool-based Approach	Conferencia	2012
A5	Abualkishik, Abedallah Zaid; Desharnais, Jean-Marc; Khelifi, Adel; et ál.	An exploratory study on the accuracy of FPA to COSMIC measurement method conversion types	Journal	2012
A6	Cuadrado-Gallego, Juan J.; Machado-Piriz, Fernando; Aroba-Paez, Javier	On the conversion between IFPUG and COSMIC software functional size units: A theoretical and empirical study	Journal	2008

A continuación presentamos una tabla resumen con la información relevante extraída de los artículos antes seleccionados.

Tabla 12. Resumen con información relevante de los artículos seleccionados para la Pregunta 2(a)

Código	Autores	Propone nuevo método	Realiza validación del estudio	Técnica aplicada formalizada	Uso de casos de estudio	Uso de herramientas y/o software
A1	Rabbi, M.F. ; Natraj, S. ; Kazeem, O.B.			X	X	
A2	Gencel, C. ; Bideau, C.	X	X	X	X	
A3	Cuadrado-Gallego, J.J. ; Buglione, L. ; Rejas-Muslera,			X		

	R.J. ; Machado-Piriz, F.					
A4	Lavazza, L. ; del Bianco, V. ; Geng Liu	X	X	X	X	
A5	Abualkishik, Abedallah Zaid; Desharnais, Jean-Marc; Khelifi, Adel; et ál.		X	X		X
A6	Cuadrado-Gallego, Juan J. ; Machado-Piriz, Fernando; Aroba-Paez, Javier		X	X	X	

En primer lugar, se vio la necesidad de buscar conversiones entre los dos métodos, pues en muchos casos, ante su posible aplicación a un tipo de proyecto de arquitectura diferente del que fueron inicialmente diseñados, lo más probable es que se busque compararlos bajo una misma escala.

En A1 se presenta como problemática que debido al cambio en los tipos de software se hace necesario encontrar algunas técnicas para estimar su esfuerzo y costo que puedan adaptarse a ellos. Por ello, evalúa los más conocidos en puntos de función: IFPUG y COSMIC. Se afirma que COSMIC cubre varios aspectos que IFPUG no puede. Sin embargo, al no haber una herramienta para medir proyectos con COSMIC es que se busca una forma de comparar ambos métodos.

La metodología usada en A1 fue hecha en 2 pasos. Primero, se hizo una revisión literaria para conocer la aplicación de ambos métodos. Como segundo paso, se realizó aplicó ambos en 2 casos: PC GEEK (aplicación de negocio) y Locator (aplicación orientado a eventos). Finalmente, se concluye que aún se necesita más investigación al respecto.

Como información importante, adicional a lo presentado, podemos mencionar que como parte de su técnica de conversión, en dos de los artículos se menciona el uso de regresiones lineales (A2 basó su técnica de conversión en base a 3 regresiones lineales (OLSs, LTSs y Log-Log) mientras que A4 usó OLSs), en cambio los otros artículos señalan como instrumento el uso de fórmulas o funciones matemáticas derivadas de sus estudios.

- Pregunta 2(b): ¿Qué técnicas o métodos se han realizado para las comparaciones entre IFPUG y COSMIC para medir proyectos SOA o similares?

La siguiente tabla nos muestra los artículos encontrados que tienen información relevante respecto de este punto:

Tabla 13. Artículos encontrados con información relevante para la Pregunta 2(b)

Código	Autores	Título	Tipo de publicación	Año
B1	Edagawa, T., Akaike, T., Higo, Y., Kusumoto, S., Hanabusa, S., Shibamoto, T.	Function point measurement from Web application source code based on screen transitions and database accesses	Journal	2011
B2	Cândido, E.J.D., Sanches, R.	Estimating the size of web applications by using a simplified function point method	Conferencia	2004
B3	Umbers, P., Miles, G.	Resource estimation for Web applications	Conferencia	2004
B4	Santillo, L.	Easy Function Points -- 'Smart' Approximation Technique for the IFPUG and COSMIC Methods	Conferencia	2012
B5	Santillo, L. ; Conte, M. ; Meli, R.	Early & Quick function point: sizing more with less	Conferencia	2005
B6	L. De Marco, F. Ferrucci, C. Gravino, F. Sarro	A COSMIC measurement procedure for sizing web applications developed using the OO-H method	Journal	2010

A continuación presentamos una tabla resumen con la información relevante extraída de los artículos antes seleccionados.

Tabla 14. Resumen con información relevante de los artículos seleccionados para la Pregunta 2(b)

Código	Autores	Propone nuevo método o técnica	Relación con método de estimación		Aplica método completo de medición	Crea aplicaciones para medición	Tecnología usada en aplicaciones web	Cantidad de casos de estudios evaluados
			IPFUG	COSMIC				
B1	Edagawa, T., Akaike, T., Higo, Y., Kusumoto, S., Hanabusa, S., Shibamoto, T.	X	X		X	X	JAVA	2
B2	Cândido, E.J.D., Sanches, R.		X		X		PHP, HTML, JAVA	20
B3	Umbers, P., Miles, G.	X		X	X	X	J2EE	2

B4	Santillo, L.	X	X	X	X			
B5	Santillo, L. ; Conte, M. ; Meli, R.	X	X	X	X	X		
B6	L. De Marco, F. Ferrucci, C. Gravino, F. Sarro	X		X	X			1

Respecto al tema de las comparaciones se pudo detectar dos grupos, los que analizaban a IFPUG o a COSMIC con aplicaciones web y las que los comparaban con otros tipos de mediciones.

Es así que, en B1 analiza al método IFPUG con el método que propone aplicándolo en cinco pasos: extraer Actions, construir el árbol SQL, identificar data functions (DFs), identificar transaction functions (TFs) y contar function points (FPS); en cambio B2 analiza también a IFPUG con NESMA, comparando ambos resultados antes de simplificar los métodos que considerará para el estudio.

Ahora, B3 analiza al método COSMIC con el método que propone aplicándolo en relación con el Diseño de patrones pero solo para la Vista Developer.

En B4, al momento de comparar ambos métodos, se indica tienen diferentes niveles conceptuales pues mientras IFPUG necesita saber los procesos elementales y los archivos lógicos, y la cantidad de rangos (DET y RET) para asignar los puntos de función, COSMIC trabaja con componentes a un nivel inferior al IFPUG (movimientos de datos entre grupos de datos).

IFPUG tiene escalas definidas para los valores de los puntos de función (escala baja, media y alta) para cualquier componente funcional base, en cambio COSMIC asigna un 1 punto de función a cada tipo de movimiento y no hay límite para cantidad de movimientos de datos, teniendo en cuenta que el tamaño mínimo de un proceso funcional en COSMIC tendrá 2 puntos de función.

Un punto importante visto en los artículos que comparan con aplicaciones web es que solo han analizado aplicaciones en Java.

Sobre las que los comparaban con otros tipos de mediciones podemos mencionar que se basan en propuestas propias, A4 se basa en SMART con aproximación probabilística, A5 se basa en una clasificación por componentes en pasos de 3 categorías, y A6 se basa en un modelo que contempla 2 modelos conceptuales: Diagrama de clases UML, que representara el contenido y los requerimientos, y los diagramas de Navegación, para los requerimientos de navegación.

- Pregunta 2(c): ¿Qué otras técnicas o modelos que propongan métodos de estimación para proyectos SOA existen?

La siguiente tabla nos muestra los artículos encontrados que tienen información relevante respecto de este punto:

Tabla 15. Artículos encontrados con información relevante para la Pregunta 2(c)

Código	Autores	Título	Tipo de publicación	Año
C1	Silvia Abrahão, Emilia Mendes, Jaime Gomez, Emilio Insfran	A model-driven measurement procedure for sizing web applications: design, automation and validation	Conferencia	2007
C2	Emilia Mendes, Nile Mosley, Steve Counsell	Investigating Web size metrics for early Web cost estimation	Journal	2005
C3	Nusiba Abdalrhman Ahmed, Abdelgaffar Hamed Ahmed	Enabling Complexity Use Case Function point on Service-oriented Architecture	Conferencia	2013
C4	Ashish Seth, Himanshu Agarwal, Ashim Raj Singla	Testing and Evaluation of Service Oriented Systems	Paper	2013
C5	L. O'Brien	A Framework for Scope, Cost and Effort Estimation for Service Oriented Architecture (SOA) Projects	Conferencia	2009

A continuación presentamos una tabla resumen con la información relevante extraída de los artículos antes seleccionados.

Tabla 16. Resumen con información relevante de los artículos seleccionados para la Pregunta 2(c)

Código	Autores	Tipo de medición analizada			Método aplicado	Existe análisis teórico	Existe aplicación práctica	Limitaciones del análisis	Número de casos de estudio
		Costo	Tamaño	Otros					
C1	Silvia Abrahão, Emilia Mendes, Jaime Gomez, Emilio Insfran		X		OO-HFP	X	X	Validación de esfuerzo se dará sobre la comparación de predicciones	
C2	Emilia Mendes, Nile Mosley, Steve Counsell	X	X		Encuestas y análisis estadístico (regresión multivarianza)	X	X	Solo trabaja con resultados de encuestas en 2 etapas	1

								para un total de 180 encuestas	
C3	Nusiba Abdalrhman Ahmed, Abdelgaffar Hamed Ahmed		X	X	UCP	X	X	Solo aplicable a modelos de procesos adecuados a principios SOA	1
C4	Ashish Seth, Himanshu Agarwal, Ashim Raj Singla	X	X			X		Cuadro de información sobre método, solución propuesta, supuestos y limitaciones	
C5	L. O'Brien	X	X		SMAT-AUS	X	X	Presenta un enfoque que aún no se termina de desarrollar.	1

Al revisar la información encontrada en estos artículos podemos notar que hay muy poco trabajo práctico realizado a analizar tanto métodos de estimación para proyectos SOA como los de costeo, siendo dos de ellos presentados en el año 2013, los que dan buenas aproximaciones al igual que C5.

Salvo C1, que aplica el método OO-HFP, el cual es un procedimiento de medición de tamaño funcional para proyectos web desarrollados utilizando el método orientado a objetos hipermedia (OO- H) y que nos permite obtener una medida de tamaño una vez que se ha completado el modelo conceptual de una aplicación web, y C3 con UCP, el cual basa su medición en 3 diagramas UML (Casos de uso, Clases y Actividades) a los cuales le asigna pesos, no hay mucha información sobre iniciativas aplicadas que estén trabajando en la estimación de proyectos SOA, aunque en C5 se presenta un framework que está en formación pero con buenas expectativas en su análisis sobre SOA.

Respecto a C5, este documento revisa con bastante detalle los diversos tipos de proyectos SOA. Además propone un marco o framework llamado SMAT-AUS para la

recolección y el uso de los datos sobre varios aspectos de los proyectos de SOA y así poder determinar el alcance, el costo estimado y el esfuerzo.

En C5 se describen las distintas dimensiones del marco (framework) que tendrá un impacto en el alcance, el costo y el esfuerzo que incluye la técnica y aspectos organizativos, como por ejemplo, la madurez de la organización en términos de su experiencia en proyectos SOA.

Finalmente nos mostrará este framework mediante un caso práctico.

Este documento empieza tomando como punto de partida que es necesario saber qué tipo de proyecto SOA se desea aplicar. Por ello, se hace una clasificación de los proyectos SOA:

- Service Mining

Este tipo se orienta a identificar cuáles de los servicios ya existentes se podrán reutilizar en el proyecto.

Para realizar esa identificación se sugieren tomar en cuenta las siguientes actividades:

- Analizar los sistemas existentes e identificar los componentes potenciales que podrían satisfacer las necesidades de servicio.
- Determinar aquellos componentes a migrar y volver a usar como servicios. Esto permitirá revisar la creación de nuevos componentes.
- Determinar el costo y el esfuerzo de la creación y reutilización de los componentes.

Un método para apoyar la determinación del alcance, el esfuerzo y el costo de estimación en la minería y la reutilización de los componentes existentes es the Software Engineering Institute's Service Migration and Reuse Technique (SMART).

- Service Development

Este tipo se orienta crear servicios que cumplan con las necesidades de calidad estipulados. Para desarrollarlos se debe seguir pasos muy similares con la creación de un software cualquiera (Determinar los requisitos del servicio, desarrollar la arquitectura de los servicios así como elaborar un diseño detallado para cada uno, implementar los servicios y finalmente crear las pruebas de los mismos).

- Application Development

Este tipo se deberá tomar en cuenta el desarrollo de las aplicaciones ya sea por parte de la organización así como por otros proveedores externos. En cualquiera de los casos se deberá tomar en cuenta la integración completa mediante una aplicación.

- Service Integration

En este tipo se analizan la integración de los servicios con los sistemas de la organización.

Para ello, se deberá determinar qué sistemas deben integrarse con los servicios, lo cual implica el pleno conocimiento de los cambios que se necesitan para realizar la integración. Luego, se deberá determinar el alcance del proyecto, y con ello poder estimar los costos y el esfuerzo. Además se recomienda no dejar de lado el impacto de las diferentes alternativas de arquitectura así como los requerimientos indicados en los contratos con los proveedores de servicios.

- SOA Infrastructure

En este tipo se determina la infraestructura de SOA que usará la organización.

Para ello, primero se determina los requisitos de la arquitectura SOA, luego se evalúa cuál sería el o los proveedores más adecuados. Finalmente, escoge la infraestructura y se analiza la personalización que se deberá hacer para adecuarla totalmente a la organización.

Es por este motivo que los costos involucrados y el esfuerzo para realizar el proyecto pueden ser mucho mayores al contemplado inicialmente.

- SOA Governance

En este punto se debe establecer el gobierno del proyecto SOA. Se debe determinar el desarrollo de la estrategia y los objetivos del proyecto, la forma de financiación y definir los indicadores que mostrarán el progreso del proyecto. Es aquí donde se debe determinar cuál será el modelo que usar para la medición.

- SOA Architecture Analysis

En este punto se puede examinar y determinar el cumplimiento de los requisitos de calidad de servicio, como el rendimiento, la escalabilidad, la seguridad, etc.

A continuación se comentan algunos enfoques relativos a los costos y el esfuerzo.

Entre ellos están los siguientes: los modelos algorítmicos o paramétricos, los que son por analogía, los juicios de expertos y los de punto de función.

En este punto el documento coincide conmigo acerca que hay poca documentación que trate acerca de la estimación del esfuerzo sobre proyectos SOA.

Además se indica que principalmente los intentos de medición se han tratado mediante modelos algorítmicos o por puntos de función. Sobre puntos de función se cita a Santillo sobre la dificultad de la definición de la unidad de tamaño debido al alcance de las partes del software.

También se cita a Linthicum que en [24] da algunas pautas para la determinación del costo de un proyecto SOA.

A continuación se presenta las características principales del framework SMAT-AUS.

Este marco contempla que en cada tipo de proyecto SOA hay un conjunto de métodos, plantillas, funciones de costo y factores de costo específico.

Además se menciona que no debe dejarse de lado la madurez de la organización así como dos dimensiones importantes: la dimensión técnica que implica el trabajo técnico implicado y la dimensión sociocultural que implica el socio / cultural / cuestiones de organización dentro de un proyecto SOA.

Finalmente se presenta un caso práctico de este framework, el cual se aplica sobre un proyecto SOA que integrará un ESB con varias aplicaciones de una agencia gubernamental de Australia.

Al momento de aplicarlo se detectaron varios inconvenientes, como por ejemplo que existían varias áreas que no estaban preparadas para la integración, además no se había analizado el costo y el esfuerzo de la parte técnica de la integración de los sistemas, ni tampoco los involucrados en los cambios en las operaciones de los usuarios finales.

Capítulo 6: Conclusiones y trabajos futuros

Luego de revisar las fuentes presentadas hemos podido encontrar que la aplicación de los métodos de estimación de tamaño funcional IFPUG FPA y COSMIC sobre proyectos SOA está en una fase inicial, esto debido a que se ha investigado poco sobre cómo adaptar ambos métodos a un tipo de software que tiene una arquitectura en base a capas como lo es SOA. COSMIC es el que ha avanzado más, incluso ya ha propuesto una primera guía, la cual se puede revisar en [25].

La mayoría de los artículos encontrados se basan en mencionar la conversión entre las medidas de ambos métodos, pero hay muy poco trabajo en su relación con los proyectos web y SOA. Muchos de ellos se centran en convertir IFPUG a COSMIC y discutir la validez de ello, pero no buscan adaptarlos a los nuevos tipos de proyectos como aplicaciones Web o SOA. Esta adaptación sería importante pues permitiría tanto a IFPUG como a COSMIC, el poder ampliar el uso de su medición. Aunque los artículos revisados sí coinciden en que el enfoque dado por COSMIC es más flexible para adaptarse a los diferentes tipos de proyectos.

Respecto a las comparaciones entre IFPUG y COSMIC para medir proyectos SOA o similares, se han encontrado trabajos que están intentando crear o aplicar métodos para la estimación tanto de esfuerzo como de tamaño funcional de aplicaciones web pero coinciden que es algo complicado debido a como se estructuran estos proyectos. Sin embargo, estas apreciaciones no toman en cuenta la guía de COSMIC orientada a SOA recientemente presentada.

Respecto al tema de encontrar otras técnicas o modelos que propongan métodos de estimación para proyectos SOA, hemos podido ver que hay muy poco lo que aun se ha hecho, a pesar de ello C3, C4 y C5 nos dan pautas muy buenas sobre adonde direccionar los estudios para poder aproximar métodos de estimación sobre proyectos SOA. Cabe resaltar que el framework SMAT-AUS presentado en C5 es una muy buena propuesta pues está pensada en muchos tipos de proyectos SOA, aunque como aún está en construcción, habrá que seguir evaluándolo en más casos prácticos.

Respecto a las fuentes revisadas, hemos notado que hay pocos trabajos publicados acerca de estimación en proyectos SOA. La mayoría de ellos se han presentado en Conferencias, sin embargo, se necesita aumentar su presencia en journals.

Proponemos como trabajos a futuros realizar estimaciones a proyectos SOA con el método COSMIC aplicando casos de estudio de los diferentes tipos de proyectos SOA que sigan la última versión del método propuesto en [25] y que además se asegure que cumplan todos los principios que SOA define, esto con la finalidad de reducir una de las limitaciones que se le adjudican por tener poca información histórica.

También sugerimos que los trabajos que se hagan no solo se ciñan a proyectos en Java sino que también analicen casos que usen .Net

Referencias

- [1] Manual de Medición, Método COSMIC Versión 3.0.1 (traducción al español)
- [2] RAE. Diccionario de la Lengua Española. Madrid: Real Academia Española
<http://www.rae.es/>
- [3] <http://www.cosmicon.com/> The official website of the Common Software Measurement International Consortium , the 'COSMIC' organization
- [4] Santillo L., Seizing and Sizing SOA Applications with COSMIC Function Points, SMEF, 9-11 mayo 2007, Roma
- [5] Artículo Wikipedia "Arquitectura orientada a servicios", Wikipedia La enciclopedia libre,
http://es.wikipedia.org/wiki/Arquitectura_orientada_a_servicios , acceso 20 de mayo del 2010
- [6] A COSMIC-FFP Based Method to Estimate WebApplication Development Effort
- [7] Kumar, B.V.: Implementing SOA using Java EE
- [8] Erl, Thomas: SOA Principles of Service Design.
- [9] Canto M., Pereda D., Seguro A.: Service Oriented Architecture (SOA)
- [10] Kitchenham: Procedures for Performing Systematic Reviews 2004
- [11] Pino: Revisión sistemática de mejora de procesos software en micro, pequeñas y medianas empresas 2006
- [12] Kitchenham: Guidelines for performing Systematic Literature Reviews in Software Engineering 2007
- [13] Jorgensen: A Systematic Review of Software Development Cost Estimation Studies 2007
- [14] Artículo Infobiblio " Software para gestión de citas bibliográficas", Infobiblio,
<http://www.infobiblio.es/software-para-gestion-de-citas-bibliograficas>, acceso 10 de octubre del 2013
- [15] <http://www.ifpug.org/> The official website of IFPUG (International Point Función Grupo de Usuarios)
- [16] Ortiz, Z. ¿Qué son las revisiones sistemáticas? Publicaciones CIE, World Wide Web: <http://www.epidemiologia.anm.edu.ar/> Julio 2005. CIE. Academia Nacional de Medicina, Buenos Aires.
- [17] Melgar, Andrés: Revisión sistemática Grupo de Reconocimiento de Patrones e Inteligencia Artificial aplicada
- [18] Dr. Jorge Alarcón V: CAPITULO 10. REVISION SISTEMÁTICA Y METANÁLISIS. http://www.epiredperu.net/epired/cursos/epidemiologia_bas-mh/epibas_text10.pdf
- [19] M^a del Pilar Romay, Fernández-Sanz y Roberto Tejedor: Revisión Sistemática: Pruebas de Software para SOA con UML, Actas de los Talleres de las Jornadas de Ingeniería del Software y Bases de Datos, Vol. 4, o. 5, 2010
- [20] Marcos Palacios, José García-Fanjul, Javier Tuya: Protocolo para la revisión sistemática de estudios sobre pruebas en SOAs con enlace dinámico, Actas de los Talleres de las Jornadas de Ingeniería del Software y Bases de Datos, Vol. 4, o. 5, 2010
- [21] Biolchini: Systematic Review in Software Engineering 2005 Systems Engineering and Computer Science Departament COPRE / UFRJ

- [22] Pérez, M: Medición del tamaño del software en aplicaciones SOA con puntos de función COSMIC
<http://www.cc.uah.es/ie/mt4SOA/publications/SOAYPF.pdf>
- [23] Pow Sang Portillo, José Antonio: Tesis doctoral: Técnicas para la Estimación y Planificación de Proyectos de Software con Ciclos de Vida Incremental y Paradigma Orientado a Objetos, Marzo 2012. Universidad Politécnica de Madrid
- [24] Linthicum, D., “How Much Will Your SOA Cost?”
<http://www.soainstitute.org/articles/article/article/how-muchwill-your-soa-cost.html> , Julio 2014
- [25] COSMIC_Method_v3.0.1_SOA_Software_Guideline_v1.0a
- [26] Lewis, G., Morris, E., O'Brien, L., Smith, D., and Wrage, L., “SMART: The Service-Oriented Migration and Reuse Technique”, CMU/SEI-2005-TN-029, Software Engineering Institute, USA.
- [27] Suárez, D.: Modelos de estimación para Proyectos de Software
- [28] Gartner Inc.: Predicts 2003: SOA Is Changing Software
<http://www.gartner.com/resources/111900/111987/111987.pdf>, Diciembre 2002
- [29] Bolsa de Valores de Colombia: El Modelo de Integración SOA
http://www.bvc.com.co/recursos/Files/Bus_de_Integracion/CTPEI_Modelo_de_Integracion_SOA_V2.pdf

