

UNIVERSIDAD DE COSTA RICA  
SISTEMA DE ESTUDIOS DE POSGRADO

**COMPARACIÓN DE CUATRO MÉTODOS DE MEDICION DE  
TAMAÑO FUNCIONAL PARA LA ESTIMACION DEL ESFUERZO DE  
DESARROLLO DE APLICACIONES MÓVILES EN UNA STARTUP QUE  
UTILIZA LA METODOLOGÍA ÁGIL SCRUM**

Trabajo final de investigación aplicada sometido a la consideración de la Comisión del Programa de Estudios de Posgrado en Computación e Informática para optar al grado y título de Maestría Profesional en Computación e Informática

Fabián Ugalde Rivera

Ciudad Universitaria Rodrigo Facio, Costa Rica

2020

## **Dedicatoria**

A la memoria de mi madre y mi hermano menor, quienes me inspiraron a llegar acá.

## **Agradecimientos**

A mi profesor guía Marcelo Jenkins Coronas, y a los profesores Christian Quesada López y Alexandra Martínez Porras, por sus consejos, guía y enseñanzas brindadas durante el desarrollo de la investigación.


A mi esposa Maria Laura Aguilar, por su apoyo incondicional durante todo el programa de maestría.

Este trabajo final de investigación aplicada fue aceptado por la Comisión del Programa de Estudios de Posgrado en Computación e Informática de la Universidad de Costa Rica, como requisito parcial para optar al grado y título de Maestría Profesional en Computación e Informática.



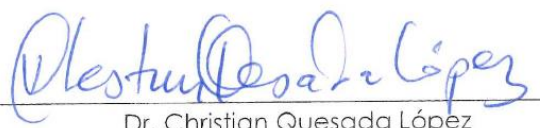
---

Dr. Gustavo López Herrera  
Representante del Decano  
Sistema de Estudios de Posgrado



---

Dr. Marcelo Jenkins Coronas  
Profesor Guía



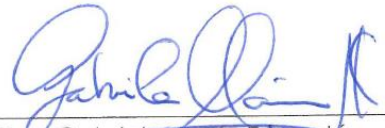
---

Dr. Christian Quesada López  
Asesor



---

Dra. Alexandra Martínez Porras  
Asesora



---

Dra. Gabriela Marín Raventós  
Directora del Programa de Posgrado en Computación e Informática



---

Fabián Ugalde Rivera  
Sustentante

# Tabla de Contenidos

<b>DEDICATORIA .....</b>	<b>II</b>
<b>AGRADECIMIENTOS .....</b>	<b>II</b>
<b>TABLA DE CONTENIDOS .....</b>	<b>III</b>
<b>RESUMEN .....</b>	<b>VII</b>
<b>LISTA DE FIGURAS .....</b>	<b>X</b>
<b>LISTA DE ABREVIATURAS.....</b>	<b>XI</b>
<b>CAPÍTULO 1. INTRODUCCIÓN .....</b>	<b>1</b>
1.1. JUSTIFICACIÓN .....	2
1.2. OBJETIVOS.....	3
1.2.1. <i>Objetivo General</i> .....	3
1.2.2. <i>Objetivos Específicos</i> .....	4
<b>CAPÍTULO 2. MARCO TEÓRICO.....</b>	<b>5</b>
2.1. MÉTODOS DE ESTIMACIÓN BASADOS EN TAMAÑO FUNCIONAL.....	5
2.1.1. <i>Puntos de casos de uso</i> .....	5
2.1.2. <i>Puntos de Historia de usuario</i> .....	7
2.1.3. <i>Puntos de función IFPUG</i> .....	8
2.1.4. <i>Puntos de función COSMIC</i> .....	12
2.2. MÉTRICAS DE EFECTIVIDAD PARA LOS MÉTODOS DE ESTIMACIÓN.....	13
2.2.1. <i>Magnitud media del error relativo (MMRE)</i> .....	13
2.2.2. <i>Media Balanceada del Error Relativo (MBRE)</i> .....	14
<b>CAPÍTULO 3. TRABAJO RELACIONADO.....</b>	<b>15</b>
<b>CAPÍTULO 4. CASO DE ESTUDIO.....</b>	<b>19</b>
4.1. PREGUNTAS DE INVESTIGACIÓN .....	19
4.2. CONTEXTO DEL CASO DE ESTUDIO.....	19
4.2.1. <i>Métodos de estimación utilizados</i> .....	20
4.3. SELECCIÓN DE LA MUESTRA.....	21
4.3.1. <i>Tamaño de la muestra</i> .....	22
4.4. PROCEDIMIENTO PARA EL ANÁLISIS .....	23

4.4.1.	<i>Procedimiento para el análisis de valores atípicos</i> .....	25
<b>CAPÍTULO 5.</b>	<b>RESULTADOS OBTENIDOS</b> .....	<b>26</b>
5.1.	ESTIMACIÓN CON PUNTOS DE CASOS DE USO .....	26
5.1.1.	<i>Identificación de puntos aislados</i> .....	26
5.1.2.	<i>Análisis de regresión lineal</i> .....	29
5.1.3.	<i>Cálculo de la Magnitud de error medio</i> .....	29
5.2.	ESTIMACIÓN CON <i>USER STORY POINTS</i> .....	30
5.2.1.	<i>Identificación de puntos aislados</i> .....	30
5.2.2.	<i>Análisis de regresión</i> .....	33
5.2.3.	<i>Cálculo de la Magnitud de error medio</i> .....	34
5.3.	ESTIMACIÓN CON PUNTOS DE FUNCIÓN IFPUG.....	35
5.3.1.	<i>Identificación de puntos atípicos</i> .....	35
5.3.2.	<i>Análisis de regresión lineal</i> .....	37
5.3.3.	<i>Cálculo del error medio</i> .....	42
5.4.	ESTIMACIÓN CON PUNTOS DE FUNCIÓN COSMIC.....	42
5.4.1.	<i>Identificación de puntos aislados</i> .....	43
5.4.2.	<i>Análisis de regresión</i> .....	45
5.4.1.	<i>Cálculo del MMRE y MBRE</i> .....	49
5.5.	MODELOS DE CONVERSIÓN ENTRE MÉTODOS .....	50
<b>CAPÍTULO 6.</b>	<b>DISCUSIÓN DE RESULTADOS</b> .....	<b>52</b>
6.1.	MÉTODOS MÁS PRECISOS.....	52
6.2.	CORRELACIÓN ENTRE LOS MÉTODOS.....	53
6.2.1.	<i>Estimar con modelo COSMIC a partir de USP</i> .....	54
6.3.	ANÁLISIS DE CAUSA RAÍZ DE VALORES ATÍPICOS .....	55
6.4.	AMENAZAS A LA VALIDEZ .....	60
<b>CAPÍTULO 7.</b>	<b>CONCLUSIONES</b> .....	<b>63</b>
7.1.	LIMITACIONES DEL ESTUDIO .....	64
7.2.	TRABAJO FUTURO.....	64
<b>CAPÍTULO 8.</b>	<b>REFERENCIAS</b> .....	<b>66</b>
<b>CAPÍTULO 9.</b>	<b>ANEXOS</b> .....	<b>68</b>
9.1.	PROTOCOLO DE REVISIÓN DE LITERATURA .....	68
9.1.1.	<i>Preguntas de investigación</i> .....	68

9.1.2.	<i>Cadena de búsqueda</i> .....	68
9.1.3.	<i>Proceso de inclusión y exclusión</i> .....	69
9.1.1.	<i>Proceso evaluación de calidad</i> .....	70
9.1.2.	<i>Proceso de selección de estudios</i> .....	71
9.2.	ANÁLISIS DE REGRESIÓN A NIVEL DE HISTORIAS DE USUARIO.....	72
9.2.1.	<i>Identificación de puntos aislados</i> .....	72
9.2.2.	<i>Análisis de regresión</i> .....	76
9.3.	CONTEO PUNTOS DE FUNCIÓN IFPUG.....	78
9.4.	CONTEO PUNTOS DE CASO DE USO.....	91
9.5.	CONTEO PUNTOS DE FUNCIÓN COSMIC.....	94
9.6.	CONTEO CON PUNTOS DE HISTORIA DE USUARIO.....	107
9.7.	ARTICULO.....	113

## Resumen

Los modelos de estimación de esfuerzo de software basados en el tamaño funcional permiten a las organizaciones de software planificar sus proyectos de desarrollo. Un gran número de organizaciones han adoptado procesos ágiles, pero hay poca evidencia sobre la adopción de métodos de medición funcional para apoyar la estimación del esfuerzo del software en contextos ágiles. En este estudio, comparamos cuatro métodos de medición del tamaño funcional para la estimación del esfuerzo de desarrollo de aplicaciones móviles en una empresa emergente (*startup*) que utiliza la metodología ágil SCRUM. Las mediciones del tamaño del software se tomaron sobre un conjunto de requerimientos de un proyecto de la empresa, y fueron expresadas en puntos de historia del usuario (USP), puntos de caso de uso (UCP), puntos de función IFPUG (UFP) y puntos de función COSMIC (CFP). Los modelos de estimación de esfuerzo se construyeron aplicando análisis de regresión a estas mediciones. La precisión de los modelos se calculó usando las métricas Magnitud Media del Error relativo (MMRE) y Media Balanceada del Error Relativo (MBRE). Para cada método de medición del tamaño funcional, obtuvimos los siguientes resultados: una MMRE de 0,86 para UCP, 0,36 para USP, 0,36 para UFP y 0,22 para CFP, y una MBRE de 0,98 para UCP, 0,45 para USP, 0,53 para UFP y 0,35 para CFP. El modelo de estimación de esfuerzo basado en los puntos de función COSMIC resultó ser el más preciso en el contexto de la organización de software bajo estudio. Además, se generaron modelos de convertibilidad entre medidas de tamaño de software para permitir a la organización expresar sus mediciones históricas en términos de otros métodos de medición, sin tener que realizar el proceso de conteo respectivo desde cero.

## Lista de Tablas

Tabla 1. Matriz de complejidades IFL y ELF .....	10
Tabla 2. Matriz de complejidad de El .....	11
Tabla 3. Matriz de complejidad para EQ y EO .....	11
Tabla 4. Peso de puntos de función según función y complejidad.....	11
Tabla 5. Modelos de conversión entre métodos. ....	24
Tabla 6. Complejidad de Casos de Uso.....	26
Tabla 7. Esfuerzo y productividad de la implementación de los casos de uso.....	27
Tabla 8. Puntos atípicos de puntos de casos de uso. ....	28
Tabla 9. Distribución de Historias de usuario según su tamaño.....	30
Tabla 10. Productividad lograda en la implementación de cada Historia de Usuario. ....	32
Tabla 11. Modelos de regresión para USP. ....	34
Tabla 12. Cálculos de error medio para USP.....	35
Tabla 13. Productividad obtenida en (HP / PF). ....	35
Tabla 14. Puntos aislados IFPUG.....	37
Tabla 15. Modelo de regresión lineal para IFPUG. ....	42
Tabla 16. Error medio de modelos estimación IFUPG. ....	42
Tabla 17. Resumen del conteo COSMIC.....	43
Tabla 18. Productividad de Puntos de función COSMIC. ....	43
Tabla 19. Modelos de regresión para COSMIC. ....	49
Tabla 20. Error Medio de Modelos COSMIC .....	49
Tabla 21. Resumen de Modelos de estimación de esfuerzo. ....	50
Tabla 22. Modelos de conversión entre métodos de estimación. ....	51
Tabla 23. Valores atípicos identificados de la muestra. ....	56
Tabla 24. Cantidad de estudios primarios recuperados por base de datos. .....	69
Tabla 25. Productividad de Historias de usuario. ....	73

Tabla 26. Modelos de regresión y MMRE para USP. ....	77
Tabla 27. Lista de entidades y atributos. ....	78
Tabla 28. Lista de funciones de datos. ....	79
Tabla 29. Medidas de funciones transaccionales. ....	80
Tabla 30. Resumen de resultados de conteo de puntos de función IFPUG. ....	91
Tabla 31. Conteo de puntos de casos de uso. ....	91
Tabla 32. Conteo de puntos de función COSMIC. ....	94
Tabla 33. Conteo con puntos de historia de usuario. ....	107

## Lista de Figuras

Figura 1. Representación de elementos funcionales IFPUG. ....	9
Figura 2. Representación de modelo COSMIC. ....	12
Figura 3. Representación de la fuente de los datos obtenidos. ....	22
Figura 4. Diagrama de caja de productividad según casos de uso (n=32). .....	28
Figura 5. Tamaño en Puntos de Casos de Uso (UCP) vs esfuerzo real. ....	29
Figura 6. Relación entre Historias de Usuario, Requerimientos y Casos de Uso. ....	31
Figura 7. Diagrama de Caja de productividad según Historias de Usuario agrupados por requerimientos (n=32). ....	33
Figura 8. Historias de usuario en tamaño vs esfuerzo real (n=32). ....	34
Figura 9. Gráfico de Cajas de Productividad según UFP. ....	37
Figura 10. Gráfico de dispersión de muestra con DET. ....	38
Figura 11. Gráfico de dispersión de muestra con FTR. ....	39
Figura 12. Gráfico de dispersión de muestra con UFP. ....	39
Figura 13. Gráfico de dispersión DET con poblaciones sugeridas. ....	40
Figura 14. Diagrama de caja productividad CFP. ....	45
Figura 15. Gráfico de dispersión de Entradas (E). ....	46
Figura 16. Gráfico de dispersión de Salidas (X). ....	46
Figura 17. Gráfico de dispersión de Lecturas (R). ....	47
Figura 18. Gráfico de dispersión de Escrituras (W). ....	47
Figura 19. Gráfico de dispersión de Puntos COSMIC (CFP). ....	48
Figura 20. Diagrama causa raíz de R1.1. ....	57
Figura 21. Diagrama de causa raíz de R3.4. ....	59
Figura 22. Diagrama de caja de productividad según historias de usuario (n=119). ....	76
Figura 23. Historias de usuario en tamaño vs esfuerzo real. ....	77

## Lista de Abreviaturas

**AW:** *Actor weight* (peso del actor)

**CFP:** *COSMIC function points* (puntos de función Cosmic)

**DET:** *Data Element Type* (elemento de tipo de dato)

**EFF:** Esfuerzo estimado

**EIF:** *External Interface File* (archivo de interfaz externa)

**HP:** Horas persona

**IFPUG:** *International Function Point Users Group*

**ILF:** *Internal Logical File* (archivo de lógica interna)

**MBRE:** Media Balanceada del Error Relativo

**MMRE:** Magnitud media relativa del error

**RET:** *Record Element Type* (elemento de tipo de registro)

**UCP:** *Use case points* (puntos de caso de uso)

**UFP:** *Unadjusted function point* (puntos de función sin ajustar)

**USP:** *User story points* (puntos de historia de usuario)

**UUCP:** *Unadjusted use case point* (puntos de casos de uso sin ajustar)



**Autorización para digitalización y comunicación pública de Trabajos Finales de Graduación del Sistema de Estudios de Posgrado en el Repositorio Institucional de la Universidad de Costa Rica.**

Yo, Fabián Ugalde Rivera, con cédula de identidad 1 1391 0913, en mi condición de autor del TFG titulado COMPARACIÓN DE CUATRO MÉTODOS DE MEDICION DE TAMAÑO FUNCIONAL PARA LA ESTIMACION DEL ESFUERZO DE DESARROLLO DE APLICACIONES MÓVILES EN UNA STARTUP QUE UTILIZA LA METODOLOGÍA ÁGIL SCRUM

Autorizo a la Universidad de Costa Rica para digitalizar y hacer divulgación pública de forma gratuita de dicho TFG a través del Repositorio Institucional u otro medio electrónico, para ser puesto a disposición del público según lo que establezca el Sistema de Estudios de Posgrado. SI  NO \*

\*En caso de la negativa favor indicar el tiempo de restricción: \_\_\_\_\_ año (s).

Este Trabajo Final de Graduación será publicado en formato PDF, o en el formato que en el momento se establezca, de tal forma que el acceso al mismo sea libre, con el fin de permitir la consulta e impresión, pero no su modificación.

Manifiesto que mi Trabajo Final de Graduación fue debidamente subido al sistema digital Kerwá y su contenido corresponde al documento original que sirvió para la obtención de mi título, y que su información no infringe ni violenta ningún derecho a terceros. El TFG además cuenta con el visto bueno de mi Director (a) de Tesis o Tutor (a) y cumplió con lo establecido en la revisión del Formato por parte del Sistema de Estudios de Posgrado.

**INFORMACIÓN DEL ESTUDIANTE:**

Nombre Completo: Fabián Ugalde Rivera

Número de Carné: B38603 Número de cédula: 1 1391 0913

Correo Electrónico: fugar09@gmail.com

Fecha: 22/09/2020 Número de teléfono: 8840 5111

Nombre del Director (a) de Tesis o Tutor (a): Dr. Marcelo Jenkins Coronas

**FIRMA ESTUDIANTE**

Nota: El presente documento constituye una declaración jurada, cuyos alcances aseguran a la Universidad, que su contenido sea tomado como cierto. Su importancia radica en que permite abreviar procedimientos administrativos, y al mismo tiempo genera una responsabilidad legal para que quien declare contrario a la verdad de lo que manifiesta, puede como consecuencia, enfrentar un proceso penal por delito de perjurio, tipificado en el artículo 318 de nuestro Código Penal. Lo anterior implica que el estudiante se vea forzado a realizar su mayor esfuerzo para que no sólo incluya información veraz en la Licencia de Publicación, sino que también realice diligentemente la gestión de subir el documento correcto en la plataforma digital Kerwá.

## Capítulo 1. Introducción

El uso de modelos de estimación de esfuerzo puede ayudar a las empresas de software a planificar, monitorear y controlar sus esfuerzos en términos de costos y procesos de desarrollo (Lavazza, et al., 2017). Por lo tanto, tener estimaciones realistas en una etapa temprana del ciclo de vida del proyecto, permite a los gerentes controlar los recursos de manera efectiva (Mendes, E., et al., 2003). En proyectos ágiles, se utilizan mayoritariamente estimaciones de esfuerzo basado en experiencia y aplicadas a su contexto, como lo son los puntos de historia (Santana, et al., 2011). Sin embargo, se necesita más evidencia empírica sobre el uso de mediciones de tamaño funcional para apoyar la estimación del esfuerzo del software en contextos ágiles. Según (Lavazza, et al., 2017), la precisión de la estimación del esfuerzo debe evaluarse cuidadosamente antes de ser usada en un entorno de real en la industria.

Por lo tanto, nuestro estudio tiene como objetivo proporcionar evidencia empírica sobre la precisión de las estimaciones del esfuerzo de desarrollo basado en mediciones de tamaño funcional en un contexto ágil. También proporcionamos detalles de los métodos utilizados, para mejorar la replicabilidad del estudio, según sugiere (Kitchenham, et al., 2009). Asimismo, los criterios establecidos por (Kitchenham, et al., 2009) y (Lavazza, et al., 2017), se consideraron para mitigar algunas de las amenazas a la validez.

En esta investigación estudiamos si la elección de las medidas de tamaño funcional tiene un efecto sobre la precisión de los modelos de estimación de esfuerzo, en el contexto de una organización particular. Esta organización es una empresa emergente que desarrolla aplicaciones móviles destinadas a promover estilos de vida saludables.

El propósito de nuestro caso de estudio es comparar la precisión de los modelos de estimación de esfuerzo, basados en cuatro medidas de tamaño funcional: puntos de caso de uso (UCP), puntos de función COSMIC (CFP), puntos de función IFPUG (UFP) y puntos de historia del usuario (USP).

Adicionalmente consideramos la opción de evaluar si convirtiendo entre valores de distintas medidas de tamaño funcional, se puede mejorar la estimación de esfuerzo para este contexto. La capacidad de una medida de tamaño funcional para ser convertida en otro tipo de medida en este contexto, se denomina convertibilidad.

Los cuatro métodos de medición del tamaño funcional se aplicaron al mismo conjunto de requerimientos de un proyecto que desarrolla una aplicación móvil dentro de la organización. Los análisis de regresión se utilizaron para derivar modelos de estimación de esfuerzo a partir de las cuatro medidas de tamaño. La Magnitud media del error relativo (MMRE) y la Media Balanceada del Error Relativo (MBRE) se utilizaron para determinar cuál modelo era más preciso para estimar el esfuerzo. Un valor de MMRE menor a 0,25 y de MBRE menor a 0,35 se considera preciso.

Para guiar este estudio, se definieron dos preguntas de investigación:

**RQ1:** *¿Cuál modelo de estimación de esfuerzo basado en el tamaño funcional produce las estimaciones más precisas en el contexto de la organización?*

**RQ2:** *¿Pueden los modelos de convertibilidad de las medidas de tamaño funcional producir resultados precisos en el contexto de la organización?*

## **1.1. Justificación**

En un estudio de Irlanda del Norte (Wilkie, et al., 2011) que involucraba a más de 56 organizaciones categorizadas como PYMEs

(pequeña y mediana empresa), se determinó que de 15 áreas de la ingeniería de software, el tema de estimaciones fue el más desafiante. En adición, (Quesada-Lopez, et al., 2017) reportaron que en Costa Rica muy pocas organizaciones utilizan los métodos formales de estimación, siendo el juicio experto el método más utilizado para estimar esfuerzo. La empresa en la cual se realizó el presente estudio no es la excepción, de hecho, existe una importante oportunidad de mejora relacionada a la estimación del esfuerzo de desarrollo para el cumplimiento de tiempos de entrega del producto, dado que hasta el momento del estudio las estimaciones se realizaban de manera arbitraria y sin respaldo en datos históricos.

De acuerdo con (Mendes, et al., 2003), tener estimados realistas en etapas tempranas del ciclo de vida del proyecto permite a los tomadores de decisión administrar de manera efectiva los recursos. En el contexto de una empresa emergente (*startup*), esto puede ser determinante para su éxito, ya que los recursos tienden a ser muy limitados.

Teniendo en perspectiva la limitante de recursos y la importancia de las estimaciones, la empresa optó por almacenar datos históricos de forma que, como resultado de este estudio, podamos determinar cuál es el método de estimación de esfuerzo más preciso, para mejorar la gestión de los recursos e incrementar la confianza del cliente al generar expectativas más realistas en cuanto tiempos de entrega.

## **1.2. Objetivos**

### *1.2.1. Objetivo General*

Comparar cuatro métodos de estimación de tamaño funcional utilizados para la estimación del esfuerzo de desarrollo de aplicaciones móviles, en el contexto de una empresa de software emergente que utiliza la metodología ágil Scrum.

### *1.2.2. Objetivos Específicos*

- Caracterizar los métodos de estimación de esfuerzo de desarrollo basados en puntos de función que se ajusten al contexto de la organización de software.
- Comparar la efectividad de modelos de estimación de esfuerzo basados en el tamaño funcional en el contexto de un proyecto de desarrollo de la organización.

La estructura del documento es la siguiente. El Capítulo 2 contiene el marco teórico, donde se definen los métodos de estimación utilizados y las métricas MMRE y MBRE utilizadas para comparar la efectividad de estos métodos. El Capítulo 3 expone el trabajo relacionado encontrado mediante una revisión de literatura. En el Capítulo 4 se describe el caso de estudio diseñado para realizar la comparación entre los métodos de estimación que consideramos en el alcance. El Capítulo 5 muestra los resultados obtenidos del caso de estudio. En el Capítulo 6 se realiza una discusión de los resultados, resaltando hallazgos interesantes y recomendando el método de estimación más preciso. El Capítulo 7 reúne las conclusiones más importantes del estudio y el trabajo futuro.

## Capítulo 2. Marco Teórico

En este capítulo se describen los conceptos teóricos necesarios para entender nuestro estudio. Incluye la descripción teórica de los métodos de estimación basados en el tamaño funcional, así como las métricas de comparación de efectividad entre los métodos de estimación.

### 2.1. Métodos de estimación basados en tamaño funcional

#### 2.1.1. Puntos de casos de uso

##### 2.1.1.1. Casos de uso

Un caso de uso detalla los escenarios que indican la manera en la que el sistema de software debe interactuar con el usuario u otros sistemas para llevar a cabo una acción o una actividad específica (Bente, et al., 2001). Normalmente estas interacciones son iniciadas por un actor principal. Un actor es una entidad externa que mantiene algún tipo de relación con el sistema, puede ser un ser humano o un sistema de software externo.

##### 2.1.1.2. Modelado de casos de uso

El modelado de casos de uso es una técnica popular para capturar y describir los requerimientos funcionales de un sistema de software. Un modelo de caso de uso permite definir el alcance del sistema que va a ser desarrollado, y el alcance servirá para definir las bases de las estimaciones *de general a específico*. Esta técnica se puede utilizar en proyectos de sistemas nuevos, o en proyectos donde se mejora un sistema existente.

### 2.1.1.3. Método de puntos de casos de uso

La estimación de software por puntos de caso de uso es una técnica utilizada para calcular y predecir el tamaño de un sistema en proyectos de desarrollo de software. Esta técnica fue propuesta y desarrollada por (Karner, 1993) mientras laboraba para la empresa Objectory Systems.

La técnica fue diseñada en el contexto de sistemas de software orientado a objetos, que tienen los requerimientos claramente definidos y los casos de uso debidamente creados. Puede ser utilizada para estimar el esfuerzo de desarrollo en una etapa temprana, tanto en sistemas de software nuevos o en el mejoramiento de sistemas existentes.

Para calcular el total de puntos, se requiere contar el total de transacciones en cada caso de uso. Una transacción consiste en un evento que ocurre entre el actor y el sistema. Para que la transacción sea contada, el evento debe ocurrir totalmente o no ocurrir; es decir, no se cuentan eventos parciales.

Los cuatro principales pasos de este método se describen a continuación (Bente, et al., 2001):

1. Se deben categorizar los actores en simples (un sistema con un API definido), promedio (un sistema interactuando a través de un protocolo como TCP/IP), o complejos (un ser humano). Los pesos para cada categoría son 1, 2 y 3, respectivamente.
2. Se deben categorizar los casos de uso dependiendo de la cantidad de transacciones. Un caso de uso simple es aquel que tiene 3 transacciones o menos; un caso promedio tiene entre 4 y 7; y un caso complejo tiene más de 7 transacciones. Los pesos para cada categoría son 5, 10 y 15, respectivamente.
3. Los puntos calculados en los primeros dos pasos son ajustados de acuerdo a una serie de factores técnicos (sistema distribuido,

eficiencia del usuario final, entre otros) y ambientales (experiencia con orientación a objetos, entre otros). Para cada factor se asigna un valor entre 0 y 5, dependiendo del impacto o influencia en el proyecto.

4. Por último, se asignan las horas persona para cada punto de casos de uso. Existen diferentes propuestas para la cantidad de horas. Por ejemplo, (Karner, 1993) propone 20 horas, mientras que (Sparks et al., 1999) proponen entre 15 y 30 horas.

## 2.1.2. Puntos de Historia de usuario

### 2.1.2.1. Historias de usuario

Las historias de usuario son la forma en que se capturan los requerimientos en las metodologías de desarrollo de software ágil (Popli, et al., 2015). Una historia de usuario es independiente, negociable y con valor, estimable, pequeña y comprobable.

### 2.1.2.2. Planning Poker

Las historias de usuario tienen un tamaño de software asignado en términos de puntos de historia de usuario. Uno de los métodos para asignar esta medida de tamaño es por medio de *Planning Poker*.

En *Planning Poker*, a los participantes se les proporcionan mazos especiales de cartas de *Planning Poker* (Gandomani, et al., 2019), donde cada carta (o tarjeta) proporciona un valor único como 0, 1/2, 1, 2, 3, 5, 8, 13, 20, 40 y 100. Otra serie de valores es la secuencia de Fibonacci. Aunque no hay consenso sobre estos valores, la serie de Fibonacci es comúnmente utilizada por los equipos de software.

El primer paso de este método es que el moderador selecciona una historia de usuario (Gandomani, et al., 2019), y luego, el representante del

cliente explica claramente los requerimientos de la historia de usuario seleccionada, para que todos entiendan la historia. Posteriormente, cada participante estima el tamaño de la historia de usuario seleccionada mostrando una tarjeta con un valor. Este valor debe seleccionarse considerando factores como la complejidad y el riesgo. En caso de que exista consenso sobre el tamaño, el tamaño nominado será considerado como el tamaño de la historia de usuario y el equipo continúa eligiendo otra historia de usuario para estimar. De lo contrario, los estimadores del tamaño más alto y más bajo necesitan explicar las razones detrás de su decisión, y defender su sugerencia de valores. Después de eso, el equipo discutirá sobre la historia de usuario y sus diversas especificaciones, requisitos y limitaciones en detalle. Inmediatamente, la estimación debe ser repetida. Este proceso puede repetirse una y otra vez hasta llegar a un consenso.

### *2.1.3. Puntos de función IFPUG*

El análisis de puntos de función es una metodología probada y aceptada para determinar el tamaño de un proyecto de desarrollo de software (Garmus, et al., 2001).

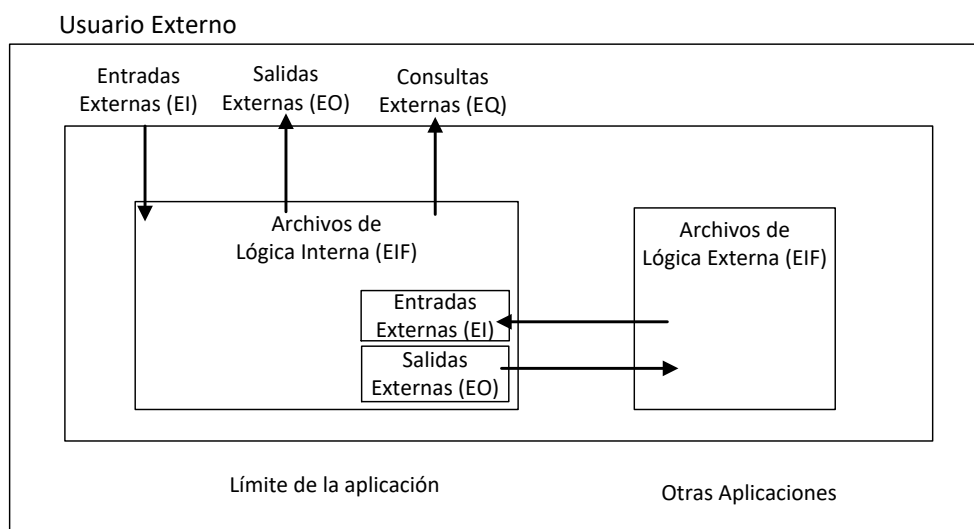
El Comité de Prácticas de Conteo de IFPUG (*International Function Point Users Group*) publica un manual que contiene todas las prácticas y estándares de los procesos de conteo. En este caso de estudio se utiliza como base la versión 4.1 de dicho manual, liberada en 1999.

#### *2.1.3.1. Identificación de puntos de función IFPUG*

El método de puntos de función evalúa la entrega del software y mide su tamaño con base en características o elementos funcionales bien definidos del software, que son (Garmus, et al., 2001):

1. Datos que ingresan a un sistema: entradas externas (EI), como entradas de transacciones lógicas o sistema de alimentación.
2. Datos que salen del sistema: salidas externas (EO) o consultas externas (EQ), tales como pantallas en línea, informes o alimentación a otros sistemas.
3. Datos que se fabrican y almacenan dentro del sistema: archivos lógicos internos (ILF), tales como grupos lógicos de datos definidos por el usuario.
4. Datos que se mantienen en un sistema diferente pero que son necesarios para satisfacer un determinado requisito de proceso: interfaces externas (EIF), como interfaces a otros sistemas.

Estos cuatro elementos funcionales, ilustrados en la Figura 1, se evalúan en función de sus complejidades, y se utilizan para determinar un recuento de puntos de función. Los ILF, EIF, EI, EO y EQ se clasifican en complejidad baja, promedio o alta, utilizando matrices IFPUG.



Tomado de (Garmus & Herron, 2001)

Figura 1. Representación de elementos funcionales IFPUG.

Para establecer la complejidad de ILF o EIF, se deben seguir las siguientes reglas (Cuadrado, et al., 2010):

1. Asigne a cada ILF o EIF identificado una complejidad basada en el número de datos tipos de elementos (DET) y tipos de elementos de registro (RET) asociado al ILF o al EIF.
2. Cuente un DET para cada usuario único reconocible, que es un campo no repetido mantenido o recuperado de la función de datos, a través de ejecución de todos los procesos elementales dentro del alcance de conteo.
3. Cuente un RET para cada función de datos. Cuente un RET adicional para cada uno de los subgrupos lógicos de la función de datos que contiene más de un DET.

La matriz de complejidad para ILF y ELF se muestra en la Tabla 1.

Tabla 1. Matriz de complejidades IFL y ELF

1 RET	1 -19 DET (Bajo)	20 -50 DET (Bajo)	+51 DET (Promedio)
2 -5 RET	1 -19 DET (Bajo)	20 -50 DET (Promedio)	+51 DET (Alto)
+6 RET	1 -19 DET (Promedio)	20 -50 DET (Alto)	+51 DET (Alto)

Tomado de (Garmus & Herron, 2001)

Para establecer la complejidad de EI, EQ, o EO, se deben seguir las siguientes reglas (Cuadrado, et al., 2010):

1. Asigne a cada EI, EQ, o EO identificado una complejidad funcional basada en el número de Tipos de elementos de datos (DET) y Tipos de archivo referenciados (FTR) asociados con la función transaccional.
2. Revise cada DET (campo) que cruza (entra / sale) el límite. Cuente solo una DET por cada usuario reconocible, que es un

atributo no repetido, que cruzó el límite durante el procesamiento de la función transaccional.

3. Cuente un FTR por cada función de datos única a la que accede (leída o escrita) por la función transaccional.

La matriz de complejidad para los EI se muestra en Tabla 2.

Tabla 2. Matriz de complejidad de EI

0 -1 FTR	1 -4 DET (Bajo)	5- 15 DET (Bajo)	+16 DET (Promedio)
2 FTR	1 -4 DET (Bajo)	5- 15 DET (Promedio)	+16 DET (Alto)
+3 FTR	1 -4 DET (Promedio)	5- 15 DET (Alto)	+16 DET (Alto)

Tomado de (Garmus & Herron, 2001)

La matriz de complejidad para EQ y EO esta descrita en la Tabla 3.

Tabla 3. Matriz de complejidad para EQ y EO

0 -1 FTR	1 -5 DET (Bajo)	6- 19 DET (Bajo)	+20 DET (Promedio)
2 -3 FTR	1 -5 DET (Bajo)	6- 19 DET (Promedio)	+20 DET (Alto)
+4 FTR	1 -5 DET (Promedio)	6- 19 DET (Alto)	+20 DET (Alto)

Tomado de (Garmus & Herron, 2001)

Después de establecer las funciones y sus complejidades, los puntos de función IFPUG se obtienen del conteo de pesos según la Tabla 4.

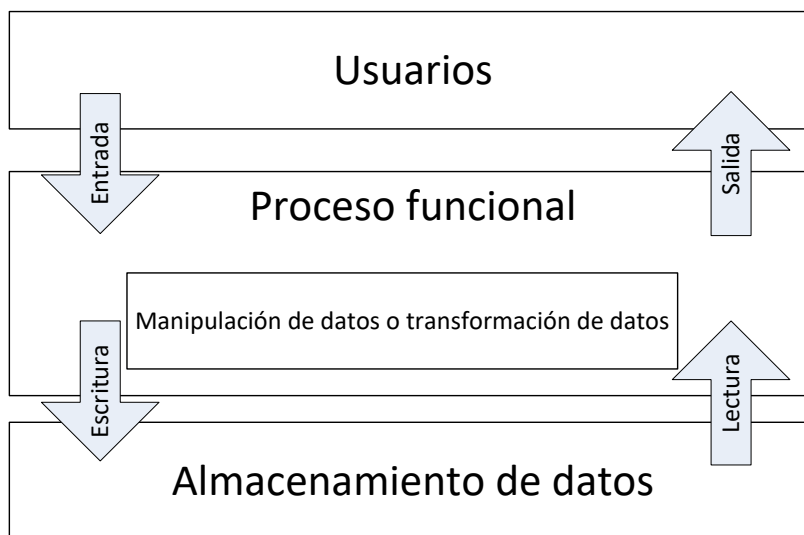
Tabla 4. Peso de puntos de función según función y complejidad

ILF	Bajo (7)	Promedio (10)	Alto (15)
EIF	Bajo (5)	Promedio (7)	Alto (10)
EI	Bajo (3)	Promedio (4)	Alto (6)
EO	Bajo (4)	Promedio (5)	Alto (7)
EQ	Bajo (3)	Promedio (4)	Alto (6)

Tomado de (Garmus & Herron, 2001)

### 2.1.4. Puntos de función COSMIC

COSMIC-FFP es un modelo propuesto por el *Common Software Measurement International Consortium (COSMIC)*, el cual establece que una unidad de datos está compuesta por un grupo de datos que es equivalente a la entidad normalizada en los sistemas de información. El modelo COSMIC FFP se representa en la Figura 2.



Tomado de (Chi-Jui, et al., 2016)

Figura 2. Representación de modelo COSMIC.

Las funciones del sistema se modelan a través de procesos funcionales, que se pueden dividir en subprocesos (Chi-Jui, et al., 2016). Los subprocesos se clasifican en dos tipos: movimiento de datos y manipulación de datos. El subproceso de movimiento de datos es fundamental para el cálculo de los puntos de función, y se divide en cuatro categorías:

1. **Entrada:** el movimiento de datos de un usuario a un funcional proceso.
2. **Salida:** el movimiento de datos de un proceso funcional a un usuario.

3. Lectura: el movimiento de datos de un almacenamiento persistente a un proceso de función. El almacenamiento persistente debe ser parte del sistema.
4. Escritura: el movimiento de datos de un proceso funcional a un almacenamiento persistente. De la misma forma, el almacenamiento persistente debe ser parte del sistema.

Por lo general, se supone que la manipulación de datos está contenida en el movimiento de los datos. En un proceso funcional, los números de movimiento del grupo de datos para las cuatro categorías mencionadas anteriormente se presentan como puntos de función, en una unidad llamada CFSU (unidad de tamaño funcional COSMIC).

## **2.2. Métricas de efectividad PARA los métodos de estimación**

### *2.2.1. Magnitud media del error relativo (MMRE)*

El criterio de evaluación más utilizado para medir el rendimiento de los modelos de predicción de software es la magnitud media del error relativo, MMRE (Foss, et al., 2003). Esto generalmente se calcula siguiendo procesos de evaluación estándar como la validación cruzada. Según (Foss, et al., 2003), un MMRE de 0,25 se considera como aceptable para modelos de predicción de esfuerzo.

La MMRE se usa para muchos propósitos. Uno de ellos es seleccionar el mejor modelo entre dos o más modelos de predicción alternativos. Por ejemplo, para comparar un modelo de estimación por analogía con un modelo de regresión lineal. El modelo que obtiene la MMRE más baja se considera mejor.

El valor de MMRE para cada dato  $i$  de la muestra  $n$  se calcula mediante la siguiente fórmula:

$$MRE_i = \frac{|V_{i \text{ estimado}} - V_{i \text{ observado}}|}{V_{i \text{ observado}}}$$

La magnitud media relativa del error, MMRE, se calcula a partir del promedio de cada  $MRE_i$ , así:

$$MMRE = \frac{\sum_{i=0}^n MRE_i}{n}$$

### 2.2.2. Media Balanceada del Error Relativo (MBRE)

La MBRE es un criterio de evaluación del desempeño para modelos de estimación, medida que ha evidenciado ser muy eficiente (Jørgensen, M., 2007). Un modelo se considera lo suficientemente preciso cuando el valor de MBRE es inferior a 0,35. Según (Miyazaki, 1994), una precisión del 35% podría ser suficiente en las fases tempranas del ciclo de vida del software.

Para calcular el valor de MBRE, primero se calcula el Error Relativo ( $RY_i$ ) para cada una de las muestras analizadas, mediante la fórmula:

$$RY_i = \frac{\text{esfuerzo estimado } (i) - \text{esfuerzo real } (i)}{\text{esfuerzo real } (i)}$$

Después se calcula el Error relativo balanceado ( $R_i$ ), utilizando la siguiente función compuesta:

$$R_i = \begin{cases} \frac{\text{esfuerzo estimado } (i) - \text{esfuerzo real } (i)}{\text{esfuerzo real } (i)}, & \text{esfuerzo estimado } (i) - \text{esfuerzo real } (i) \geq 0 \\ \frac{\text{esfuerzo estimado } (i) - \text{esfuerzo real } (i)}{\text{esfuerzo estimado } (i)}, & \text{esfuerzo estimado } (i) - \text{esfuerzo real } (i) < 0 \end{cases}$$

Finalmente, la Media balanceada del error relativo (MBRE), se calcula a partir del promedio de cada uno de los  $R_i$ , así:

$$MBRE = \frac{1}{N} \times \sum_{i=0}^N R_i$$

Las métricas de MMRE y MBRE son utilizadas en este estudio como indicadores de desempeño de los modelos de predicción generados.

## Capítulo 3. Trabajo relacionado

Este capítulo describe estudios que realizan comparaciones de métodos de estimación de esfuerzo basados en el tamaño del software.

Para identificar los trabajos relacionados, realizamos una revisión de literatura. El protocolo de dicha revisión se detalla en el Anexo 9.1 Protocolo de revisión de literatura. En total se analizaron 14 artículos, de los cuales 12 corresponden a estudios que comparan métodos de estimación de tamaño de software, y 2 corresponden a estudios sobre la convertibilidad entre medidas de tamaño de software.

Primeramente presentamos los estudios comparativos de métodos de estimación de tamaño de software.

El trabajo de (Salmanoglu, et al., 2017) consistió en tres casos de estudio que compararon la efectividad de la estimación del esfuerzo basada en dos métodos de tamaño funcional (CFP y USP) en un contexto ágil. Los autores utilizaron el análisis de regresión para generar modelos estadísticos a partir de mediciones de tamaño de software, y evaluaron su rendimiento utilizando MMRE. La conclusión del estudio fue que CFP produjo el modelo de estimación de esfuerzo más preciso.

Similarmente, (Ungan et al., 2014) compararon la efectividad de dos enfoques para la estimación del esfuerzo en una organización que usaba SCRUM. Ellos compararon la estimación de esfuerzo basada en USP (con *Planning Poker*) contra la estimación de esfuerzo basada en CFP. Para desarrollar los modelos de estimación utilizaron modelos de regresión y redes neuronales artificiales. Los resultados indicaron que las mediciones COSMIC eran una mejor base para la estimación del esfuerzo que los Puntos de historia del usuario, en el contexto del estudio.

En el estudio de (Paz et al., 2014), aplicaron los puntos de función COSMIC incrementales en un entorno ágil y compararon su precisión con

el método del Proceso Rational Unificado (RUP), concluyendo que el modelo basado en COSMIC era mejor. Los autores introdujeron el concepto de Puntos de función COSMIC incrementales como un enfoque para adaptar la medida de la CFP a entornos ágiles. Propusieron que la suma de todos los puntos de función COSMIC incrementales para cada requerimiento o unidad de tamaño de trabajo (historia de usuario en SCRUM), determinara el total de puntos de función COSMIC para todo el sistema.

Por otro lado, (Desharnais, et al., 2011) propusieron un enfoque para pasar de COCOMO (Modelo de costo constructivo) a Historias de usuario utilizando COSMIC como método de medición del tamaño. Ellos crearon un modelo que permite estimar el esfuerzo para cada historia de usuario basado en CFP.

(Sholiq, et al., 2017) compararon UCP contra UFP en un modelo de estimación de esfuerzo, concluyendo que UCP era ligeramente mejor. (Briand, et al., 1999) propusieron un modelo de estimación de esfuerzo basadas en UFP y CFP, utilizando bases de datos externas, y llegaron a la conclusión de que no había diferencias significativas entre estos dos métodos de estimación de tamaño de software.

(Mendes, et al., 2003) estudiaron tres modelos de estimación de costos aplicados a aplicaciones de hipermedia web, utilizando modelos de regresión para respaldar sus estimaciones. Ellos compararon regresiones lineales y escalonadas, y razonamiento basado en casos (CBR), concluyendo que la técnica CBR era la que ofrecía resultados más precisos.

(Wilkie, et al., 2011) exploraron la utilidad de las técnicas de tamaño de software expresado en puntos de función, cuando se aplican a varios niveles de documentación de requerimientos de software, en una organización comercial de desarrollo de software. Los autores evaluaron el

valor (costo / beneficio) que las técnicas de tamaño funcional aportaban a la planificación y gestión de proyectos de software. Ellos concluyeron que "NESMA estimado" es la herramienta más apropiada para la estimación del tamaño en el contexto de la empresa estudiada.

(Phannachitta, P., 2017) hizo una comparación de medidas de similitud en la estimación del esfuerzo de software basado en analogía, mediante un enfoque robusto que involucra MMRE y otros indicadores de comparación. Similarmente, (Mittas, et al., 2013) proponen un método gráfico para evaluar el desempeño de un modelo de estimación de costos, usando un análisis de la característica de error de regresión. Adicionalmente, (Mittas, et al., 2010) creó un algoritmo para agrupar y clasificar modelos de estimación de costos de software a través de múltiples comparaciones.

Otros estudios como (Jørgensen, M, et al., 2009) y (Lenarduzzi, V, et al., 2015) han comparado diferentes métodos de estimación de esfuerzo con estimaciones basadas en expertos. Jørgensen y Boehm (Jørgensen, M, et al., 2009) debatieron sobre qué método de estimación de esfuerzo producía una mayor precisión entre modelos formales y juicios expertos. Ellos concluyeron que no existe un mejor método único, sino diferentes perspectivas que benefician a escenarios particulares. No obstante, destacaron que los modelos de estimación más avanzados probablemente conducen a estimaciones de esfuerzo significativamente más precisas. Del mismo modo, el estudio de (Lenarduzzi, V., et al., 2015) comparó la efectividad de un modelo de estimación de esfuerzo basado en puntos de función IFPUG con una estimación basada en expertos, en un entorno ágil. Ellos mostraron que la precisión de la estimación del esfuerzo proporcionada por el equipo SCRUM era mejor que la obtenida a través de mediciones de tamaño funcional.

Ahora presentamos los estudios que abordaron la convertibilidad entre mediciones del tamaño funcional.

El trabajo de (Lavazza, et al., 2018) presenta una correlación entre IFPUG y los puntos de función COSMIC, proporcionando un conjunto de modelos para convertir recuentos de un método de tamaño de software a otro. Por otro lado, (Santana C, et al., 2011) examinaron si los puntos de función eran compatibles con los puntos de historia de usuario en proyectos ágiles, y encontró una correlación entre ellos.

Todos los estudios mencionados anteriormente se realizaron en empresas medianas y grandes, y se centraron en comparar modelos de estimación de esfuerzo basados en solo dos mediciones de tamaño funcional. Además, sus técnicas alternativas consideran solo proyectos completados con un conjunto definido de requerimientos y un alcance claro. En contraposición, nuestro estudio considera modelos de estimación de esfuerzo basados en cuatro mediciones de tamaño funcional, y se realiza en una empresa de inicio ágil. Este contexto plantea algunos desafíos, como el alcance poco claro del proyecto y la inestabilidad de los requerimientos.

En el siguiente capítulo se presenta el desarrollo del caso de estudio para este contexto, utilizando como referencia los trabajos relacionados mencionados anteriormente.

## Capítulo 4. Caso de Estudio

El caso de estudio presentado, está guiado por el protocolo de casos de estudio (Runeson, et al., 2008). En este capítulo se describen las preguntas de investigación, el contexto del caso de estudio, la selección de la muestra, y los procedimientos para el análisis.

### 4.1. Preguntas de investigación

Nuestro caso de estudio está enfocado en responder las siguientes preguntas de investigación:

- RQ1:** ¿Cuál modelo de estimación de esfuerzo basado en el tamaño funcional produce las estimaciones más precisas en el contexto de la organización?
- RQ2:** ¿Pueden los modelos de convertibilidad de las medidas de tamaño funcional producir resultados precisos en el contexto de la organización?

### 4.2. Contexto del caso de estudio

Nuestro caso de estudio se desarrolla en el contexto de una empresa emergente ("*start-up*") llamada *Koncow LLC*. Esta empresa fue fundada en el estado de Wyoming, Estados Unidos, y se dedica al desarrollo de aplicaciones móviles enfocadas en facilitar un estilo de vida saludable, aplicando el enfoque de red social a Nutrición.

Actualmente el equipo de desarrollo del producto consta de 4 desarrolladores de software y 1 dueño del producto. La aplicación consta de dos capas principalmente:

- Canal externo: Compuestos principalmente por dos aplicaciones de desarrollo "*mobile*" una en Android y otra en

iOS, ambas construidas desde cero y en su lenguaje de programación nativo.

- Canal interno: Comprende toda la plataforma en la nube de servicios Amazon Web Services (AWS) para el manejo de autenticación de usuario, administración de bases de datos y lógica de negocio.

Desde la fundación de la compañía, se ha utilizado SCRUM como marco de referencia para la gestión de los proyectos. Por lo tanto, el trabajo se cuantifica en términos de historias de usuario y su tamaño en puntos de historia de usuario. El proceso de estimación de esfuerzo es por medio de consenso en *Planning Poker*.

Cada uno de los miembros del equipo de manera individual tiene experiencia previa marco el trabajo SCRUM, sin embargo, al momento del estudio el grupo aún estaba en proceso de formación por lo cual la precisión de sus estimaciones de tamaño aún estaba en etapa de ajuste.

#### *4.2.1. Métodos de estimación utilizados*

Tomando como referencia el mismo grupo de requerimientos del mismo producto y equipo de desarrollo, aplicamos los siguientes cuatro métodos de estimación de tamaño de software:

- Puntos de casos de uso
- Puntos de historias de usuario
- Puntos de función IFPUG
- Puntos de función COSMIC

Con excepción de Puntos de historia de usuario, los métodos de estimación fueron seleccionados porque son los métodos de estimación de tamaño funcional más utilizados actualmente en la industria, y porque su método de conteo es familiar para el grupo de investigación. Los

puntos de historias de usuario es el método de conteo original de los requerimientos que utiliza actualmente la empresa.

### **4.3. Selección de la muestra**

Al momento del estudio, el equipo de desarrollo utilizaba el marco de referencia SCRUM para organizar el proyecto, y se usaba la plataforma Jira para crear las especificaciones del producto en forma de historias de usuario. Adicionalmente, como parte del desarrollo se documentan casos de uso y requerimientos de acuerdo a las prácticas recomendadas por el estándar IEEE 830-1998 de Especificación de Requerimientos (Software Engineering Standards Committee of the IEEE Computer Society, 1998).

Debido a lo anterior, se recopilaron todas las historias de usuarios para el producto en desarrollo desde el II Semestre del 2018 hasta el I Semestre del 2019, lo cual representa 6 meses de trabajo y 12 Sprints de desarrollo. Para este caso en particular, los requerimientos y casos de uso tienen una relación de 1 a 1, no obstante, las historias de usuarios son más granulares, por lo que un requerimiento o caso de uso puede tener a su vez una o más historias de usuario relacionadas.

Con respecto a los puntajes asignados relacionados al tamaño de software, los puntos de historia de usuarios fueron extraídos directamente del histórico reflejado en Jira, siendo este puntaje asignado de forma colaborativa por todos los miembros del equipo SCRUM. Sin embargo, para los puntos de casos de uso y puntos de función, uno de los investigadores un conteo manual específico para este caso de estudio.

Por otra parte, dentro de la plataforma Jira, los miembros del equipo SCRUM reportaron las horas de esfuerzo invertidas para entregar cada historia de usuario. La Figura 3 muestra la representación gráfica de la fuente de los datos.

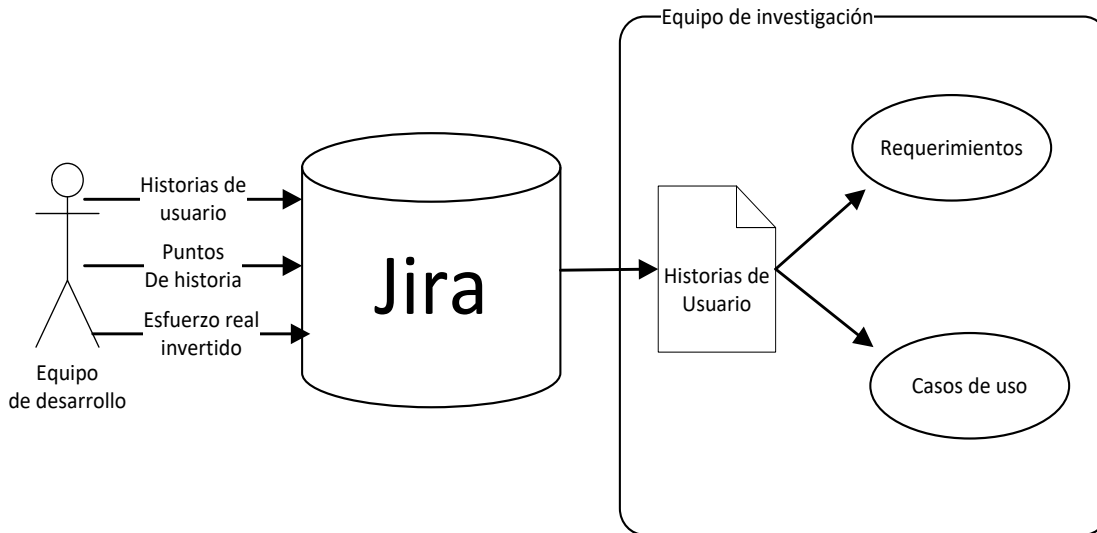


Figura 3. Representación de la fuente de los datos obtenidos.

Todos estos datos se consolidan dentro de una misma hoja de cálculo, que incluye:

- Historias de usuario, con su peso en puntos de historia y el esfuerzo real invertido.
- Casos de uso con su puntaje respectivo de puntos de casos de uso sin ajustar y el esfuerzo real invertido.
- Requerimientos caracterizados con los atributos necesarios para calcular su aporte en términos de puntos de función IFPUG y COSMIC.
- Requerimientos descritos con los atributos necesarios para calcular su aporte en términos de puntos de función COSMIC.

#### 4.3.1. Tamaño de la muestra

La muestra analizada incluye un total de 119 historias de usuario que representan 32 casos de uso, equivalentes a igual número de requerimientos unitarios. Estos registros contemplan la población completa de datos que cumplían con las características de inclusión de la muestra, desde el inicio del proyecto hasta el corte del I Semestre 2019.

Los criterios de inclusión de los datos dentro de la muestra son:

- Historias de usuarios con un puntaje en puntos de historias.
- Historias de usuario completadas
- Historias de usuario ejecutadas por el mismo equipo SCRUM
- Historias de usuario relacionadas al mismo producto desarrollado.

Estos criterios de inclusión pueden suponer una amenaza a la validez interna del estudio al contemplar el riesgo de incluir un sesgo por representatividad de la muestra. Sin embargo, estos criterios tienen el objetivo de homogenizar los datos de forma tal que cumplan con los mismos atributos y por ende puedan ser comparables entre sí.

Adicionalmente extrajimos de la muestra los valores atípicos. Para identificar los valores atípicos se graficaron los valores de productividad de cada dato de la muestra en un diagrama de caja. La productividad según cada método de estimación  $i$ , la calculamos como:

$$Productividad_i = \frac{Horas\ de\ esfuerzo}{Tamaño\ software}$$

La herramienta Minitab 16® fue utilizada para crear los diagramas de caja de productividad, a su vez el software estadístico identificaba estos valores atípicos. Según Minitab 16® los valores atípicos son observaciones que son al menos 1,5 veces el rango intercuartil (Q3 - Q1) desde el borde de la caja.

#### **4.4. Procedimiento para el análisis**

El siguiente paso fue hacer un análisis de varianza de la muestra utilizando un gráfico de dispersión. Este tipo de diagrama nos permitió observar comportamientos particulares de las muestras en términos de tendencia, pendiente y segmentación de población.

Adicionalmente, utilizando Minitab realizamos análisis de regresión similar al ejercicio realizado por (Ungan, et al., 2014), entre el tamaño de software contabilizado por requerimiento y el esfuerzo real reportado por el equipo. Caso por caso determinamos el uso de diferentes tipos de regresión, prevaleciendo siempre el tipo lineal como el de mejor exactitud. Sin embargo, si la muestra en el gráfico de dispersión daba indicio de algún comportamiento interesante que ameritara probarse con algún tipo de regresión no lineal se probaba esta para compararla contra la regresión lineal.

Con el fin de recomendar a la empresa del caso de estudio, un método alternativo para convertir sus unidades de tamaño de software existentes a los otros métodos de estimación incluidos en nuestro estudio. Se calcularon modelos de conversión entre unidades de métodos de estimación. Replicando en nuestro contexto el experimento de (Lavazza, et al., 2018).

Los modelos de conversión incluidos en este estudio están detallados en Tabla 5.

Tabla 5. Modelos de conversión entre métodos.

<b>Conversión de</b>	<b>Hacia</b>
UFP	UCP
	USP
	CFP
UCP	USP
	CFP
	UFP
USP	CFP
	UFP
	UCP
CFP	UFP
	UCP
	USP

Según lo revisado en la literatura, el procedimiento seleccionado para la comparación de métodos es utilizar la magnitud media relativa del

error MMRE, producto de la aplicación de la ecuación predictiva sobre los mismos datos en comparación con el dato real observado.

Las fórmulas para la magnitud del error utilizadas son:

Para Magnitud del error relativo de cada dato  $i$  de la muestra  $n$

$$MRE_i = \frac{|V_{i \text{ estimado}} - V_{i \text{ observado}}|}{V_{i \text{ observado}}}$$

La magnitud media relativa del error, se calcula a partir del promedio de cada MRE calculado.

$$MMRE = \frac{\sum_{i=0}^n MRE_i}{n}$$

El criterio de comparación dispuesto a ser utilizado es que un MMRE  $\leq 0,25$  es un valor aceptable. (Foss , Stensrud, Kitchenham, & Myrtveit, 2003)

#### 4.4.1. Procedimiento para el análisis de valores atípicos

Los valores atípicos que identificamos en las diferentes muestras, los analizamos más fondo utilizando una técnica de causa raíz, tal y como el Diagrama de Ishiwaka. En este, describimos de manera detallada las causas asignables en términos de 4S:

- Circunstancias (*Surroundings*)
- Proveedores (*Suppliers*)
- Habilidades (*Skills*)
- Sistemas (*Systems*)

El objetivo de dicho análisis es encontrar de manera detallada los motivos por los que dicho requerimiento fue considerado como valor atípico, tanto a nivel estadístico como en el contexto en el que ocurrió.

El capítulo siguiente presenta los resultados obtenidos, a partir de la ejecución del caso de estudio diseñado.

## Capítulo 5. Resultados obtenidos

En esta sección presentamos los resultados obtenidos producto de aplicar las fórmulas de regresión derivadas del análisis estadístico para la estimación de tamaño de software. Adicionalmente, estudiamos los datos identificados como puntos aislados y calculamos el error medio para cada uno de los métodos alternativos.

### 5.1. Estimación con Puntos de casos de uso

Modelamos los 32 casos de uso en forma de diagramas de secuencia y flujos de pantallas para realizar el conteo de puntos de casos de uso (UCP), mediante el cual contabilizamos 181,7 UCP en total.

Sin embargo, para efectos de este estudio, estamos interesados en estudiar de manera más granular el trabajo, analizando el valor estimado de esfuerzo de cada caso de uso en comparación con el esfuerzo real requerido para implementarlo.

Tomando como referencia los 32 casos de uso correspondientes a los requerimientos del sistema, aplicamos los criterios de clasificación según complejidad de los casos de uso, del cual obtuvimos la distribución presentada en la Tabla 6.

Tabla 6. Complejidad de Casos de Uso.

Tipo	Complejo	Promedio	Simple
Caso de Uso	13%	25%	63%
Actor	22%	9%	69%

En el Anexo 9.4 se muestra la totalidad de los datos analizados.

#### 5.1.1. Identificación de puntos aislados

Antes de proceder a hacer el análisis de regresión, realizamos un análisis de puntos aislados con la intención de extraer de la muestra los

valores atípicos y minimizar el sesgo. La Tabla 7 muestra los valores de productividad del equipo de desarrollo para cada uno de los puntos de casos de uso y las horas de esfuerzo reales invertidas en implementar cada uno de ellos.

Tabla 7. Esfuerzo y productividad de la implementación de los casos de uso.

Requerimiento	UCP	HP	Productividad (horas / caso de uso)
1.1	18	504,95	28,05
1.2	7	124,00	17,71
2.1	11	128,75	11,70
2.2	11	195,51	17,77
2.3	6	166,50	27,75
3.1	8	96,00	12,00
3.2	6	68,00	11,33
3.3	11	57,75	5,25
3.4	18	459,50	25,53
4.1	11	263,45	23,95
4.2	11	39,50	3,59
4.3	6	21,25	3,54
5.1	8	81,50	10,19
5.2	6	63,00	10,50
6.1	8	32,00	4,00
6.2	6	13,75	2,29
6.3	6	10,00	1,67
6.4	6	12,00	2,00
7.1	18	247,25	13,74
7.2	6	72,00	12,00
8.1	6	60,00	10,00
8.2	6	18,50	3,08
8.3	6	22,00	3,67
8.4	6	24,50	4,08
8.5	6	10,00	1,67
8.6	6	15,00	2,50
8.7	12	142,50	11,88
8.8	18	158,00	8,78
9.1	11	40,00	3,64
9.2	11	71,75	6,52
9.3	6	37,00	6,17
9.4	7	31,50	4,5

Por medio de un diagrama de caja identificamos los puntos que estadísticamente representan valores atípicos. Los puntos aislados se representan con un asterisco en la Figura 4.

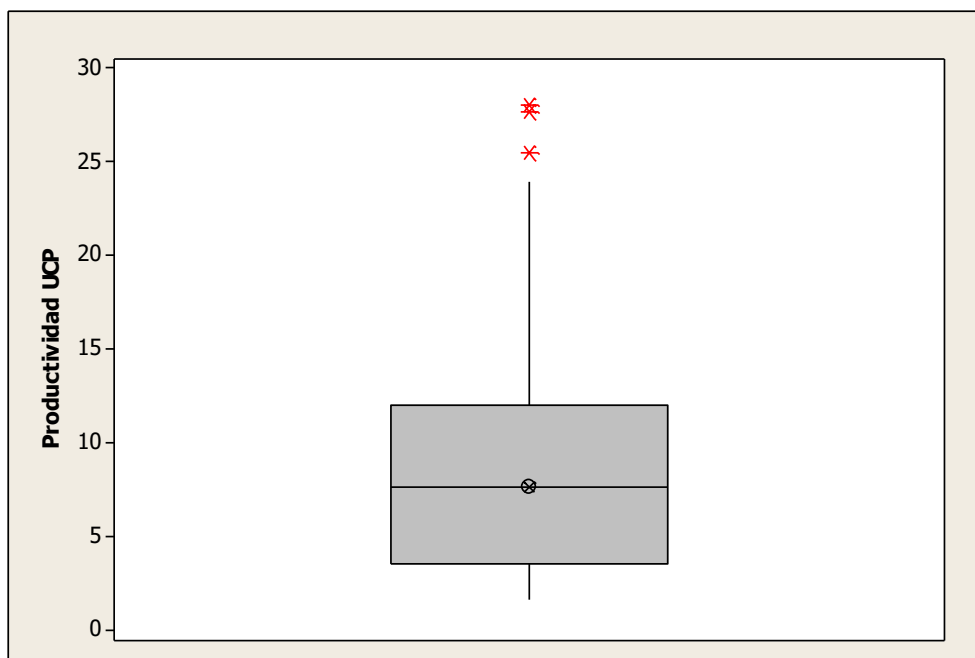


Figura 4. Diagrama de caja de productividad según casos de uso (n=32).

En la Figura 4, se demarcan con un símbolo de asterisco rojo tres valores atípicos, los cuales se detallan en Tabla 8.

Tabla 8. Puntos atípicos de puntos de casos de uso.

Requerimiento	UCP	HP	Productividad
(R1.1) Crear un usuario nuevo en la plataforma usando correo electrónico e información personal	18	504.95	28.05
(R2.3) Mantener sesión del usuario activa, para que el usuario pueda ingresar a la aplicación rápidamente.	6	166.5	27.75
(R3.4) Después de establecer la relación paciente-nutricionista, el usuario puede cancelar dicha conexión	18	459.3	25.53

En el Capítulo 6 se discutirán más a fondo la situación particular de estos valores atípicos y sus causas asignables.

### 5.1.2. Análisis de regresión lineal

Utilizando el software estadístico Minitab 16, se grafica la nueva muestra excluyendo estos valores atípicos en un gráfico de dispersión para poder inferir algún patrón o tendencia. La muestra de puntos de casos de uso la presentamos en la Figura 5.

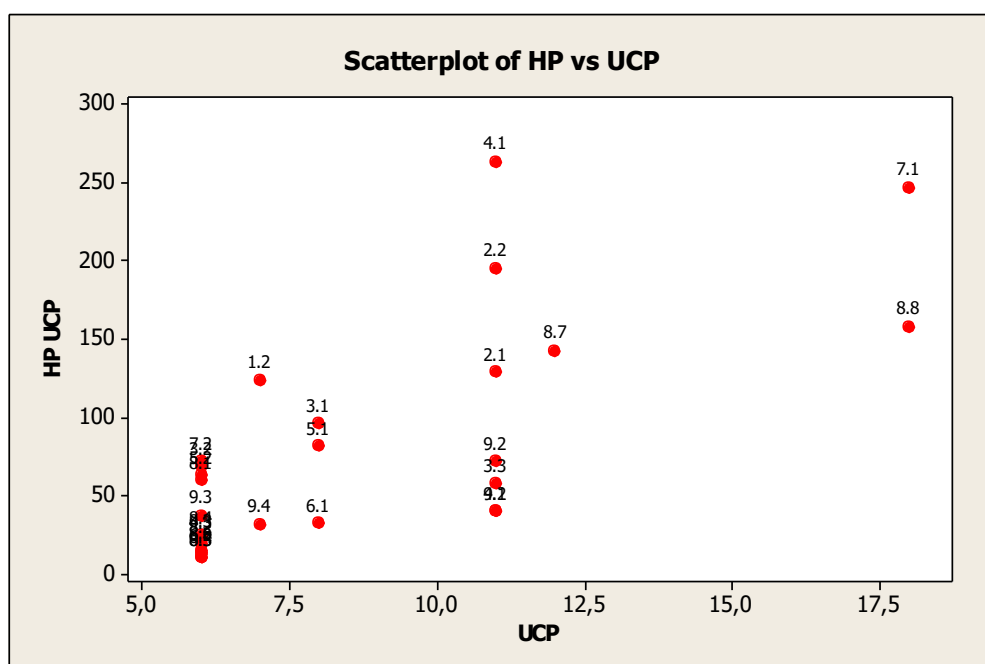


Figura 5. Tamaño en Puntos de Casos de Uso (UCP) vs esfuerzo real.

Después de aplicar un análisis de regresión lineal, se obtiene una ecuación lineal predictiva de esfuerzo:

$$EFF = -51,0 + 14,7 UCP$$

### 5.1.3. Cálculo de la Magnitud de error medio

Utilizando esta ecuación lineal, se predice cuál sería el esfuerzo estimado (EFF), y este se compara contra el esfuerzo real reportado (HP).

Por cada uno de los casos de usos recopilados en la muestra, se calculó la magnitud de error relativa (MRE). El promedio de todos estos MRE resultaron en una magnitud media de error relativo (MMRE) de 0,86. Por otro lado el valor de MBRE calculado es 0,98. La discusión sobre estos resultados la desarrollamos en el Capítulo 6.

## 5.2. Estimación con *User Story Points*

El proyecto utilizó el marco de referencia SCRUM para gestionar el desarrollo, por lo que las historias de usuario fueron documentadas en la plataforma Jira. Con un total de 119 historias de usuario, cada una de ellas está caracterizado por una estimación en User Story Points (USP), un esfuerzo real reportado por historia, y la descripción de la misma. La distribución de las historias según su tamaño en USP se muestra en Tabla 9.

Tabla 9. Distribución de Historias de usuario según su tamaño.

USP	Cantidad
1	18
2	30
3	42
5	20
8	9
Total	119

### 5.2.1. Identificación de puntos aislados

De la misma forma, para esta muestra identificamos los puntos aislados por medio de la productividad lograda para implementar cada historia de usuario.

Según mostramos en la Figura 3, las historias de usuario se documentan en la plataforma Jira. Sin embargo, por definición las historias de usuario son muy pequeñas y específicas, por lo que la empresa documenta sus requerimientos a nivel de historias más grandes o “épicas”, de forma tal que por cada Épica existen 1 o varios requerimientos y/o

casos de uso, a la vez que para desarrollar 1 requerimiento se crearon 1 o varias historias de usuario. La relación entre requerimientos, casos de uso e historias de usuario se representa en la Figura 6.

El agrupamiento de las historias de usuario en términos de requerimientos y casos de uso permite homogenizar la muestra de tal manera que la comparación entre los métodos se haga en condiciones similares. Esta estrategia la utilizamos para mitigar las amenazas a la validez interna, según advertencias de (Kitchenham & Mendes, 2009) cuando se realizan este tipo de estudios comparativos. Por lo tanto, los datos de productividad de las historias de usuario las calculamos agrupadas por requerimiento, tal y como se muestran en la Tabla 10.

Utilizamos el gráfico de caja para descubrir algún posible valor atípico. Sin embargo como se muestra en la Figura 7, no existe evidencia estadística para considerar algún valor atípico, desde la perspectiva de historias de usuario.

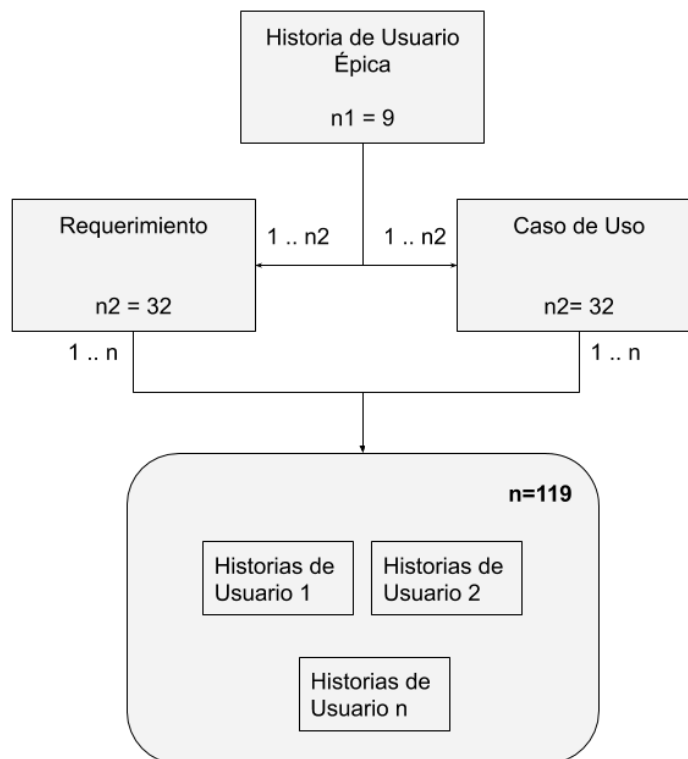


Figura 6. Relación entre Historias de Usuario, Requerimientos y Casos de Uso.

Tabla 10. Productividad lograda en la implementación de cada Historia de Usuario.

<b>Requerimiento</b>	<b>USP</b>	<b>HP</b>	<b>Productividad</b>
1.1	44	504,95	11,48
1.2	8	124,00	15,50
2.1	18	128,75	7,15
2.2	19	195,51	10,29
2.3	29	166,50	5,74
3.1	13	96,00	7,38
3.2	8	68,00	8,50
3.3	6	57,75	9,63
3.4	39	459,50	11,78
4.1	27	263,45	9,76
4.2	7	39,50	5,64
4.3	6	21,25	3,54
5.1	7	81,50	11,64
5.2	7	63,00	9,00
6.1	5	32,00	6,40
6.2	2	13,75	6,88
6.3	4	10,00	2,50
6.4	1	12,00	12,00
7.1	34	247,25	7,27
7.2	12	72,00	6,00
8.1	6	60,00	10,00
8.2	2	18,50	9,25
8.3	3	22,00	7,33
8.4	3	24,50	8,17
8.5	1	10,00	10,00
8.6	1	15,00	15,00
8.7	20	142,50	7,13
8.8	21	158,00	7,52
9.1	4	40,00	10,00
9.2	7	71,75	10,25
9.3	6	37,00	6,17
9.4	6	31,50	5,25

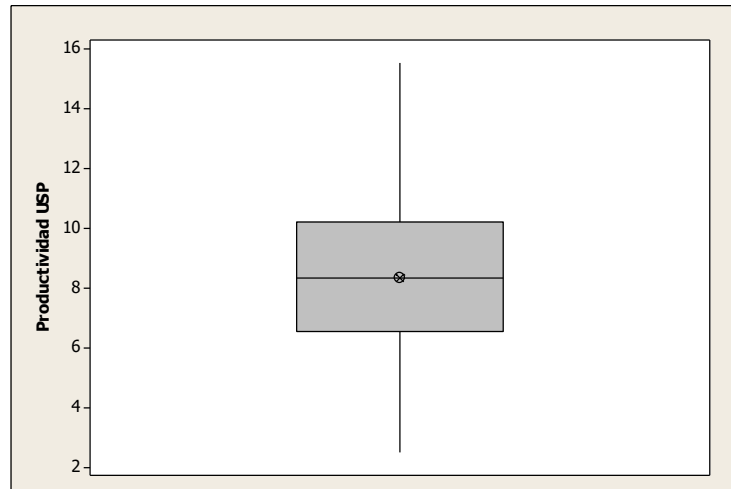


Figura 7. Diagrama de Caja de productividad según Historias de Usuario agrupados por requerimientos (n=32).

### 5.2.2. Análisis de regresión

Las historias de usuario se agruparon por requerimiento propiciando un ambiente en el cual los métodos de estimación sean comparables entre sí. Si los datos no se agruparan de esta forma, el análisis cambiaría su enfoque y de hecho obtuvimos un MMRE mayor, en el Anexo 9.2 se muestra el análisis con el enfoque más granular a nivel de historia de usuario evidenciando MMRE mayor.

Los requerimientos estudiados, con su tamaño en historias de usuario y esfuerzo requerido, se graficaron para evaluar su patrón de tendencia, según se muestra en la Figura 8.

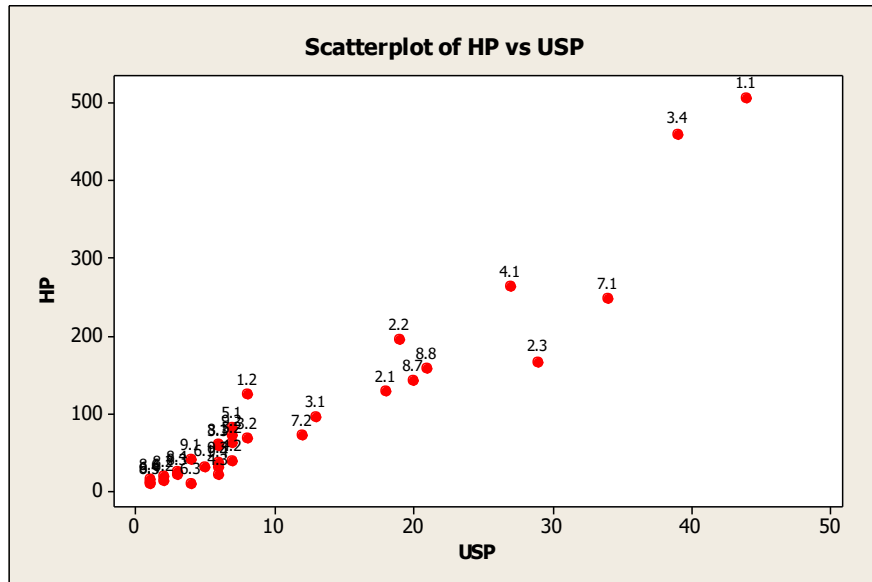


Figura 8. Historias de usuario en tamaño vs esfuerzo real (n=32).

Después de inspeccionar la muestra, consideramos que un análisis de regresión se ajusta a la tendencia lineal que presentan los datos. Sin embargo intentamos además ajustar a los datos a una regresión cuadrática y cúbica, por la tendencia parabólica que observamos en los datos.

Los modelos de regresión estudiados para este método de estimación los presentamos en la Tabla 11.

Tabla 11. Modelos de regresión para USP.

Tipo de regresión	Modelo
Lineal	$EFF = -13,1 + 9,86 USP$
Cuadrada	$EFF = 18,47 + 3,392 USP + 0,1650 USP^2$
Cúbica	$EFF = -8,75 + 12,30 USP - 0,3977 USP^2 + 0,008921 USP^3$

### 5.2.3. Cálculo de la Magnitud de error medio.

Aplicamos la ecuación de predicción de esfuerzo a partir de los puntos de historia de usuario. A la muestra completa dado que para este

método no se identificó ningún valor atípico. Los resultados de MMRE y MBRE para los modelos de regresión de USP los presentamos en la Tabla 12. La discusión sobre estos resultados la desarrollamos en el Capítulo 6.

Tabla 12. Cálculos de error medio para USP.

<b>Modelo</b>	<b>MMRE</b>	<b>MBRE</b>
$EFF = -13,1 + 9,86 USP$	0,45	0,87
$EFF = 18,47 + 3,392 USP + 0,1650 USP^2$	0,41	0,45
$EFF = -8,75 + 12,30 USP - 0,3977 USP^2 + 0,008921 USP^3$	0,36	0,58

### 5.3. Estimación con Puntos de función IFPUG

El proyecto estudiado lo medimos también con el método de estimación de puntos de función IFPUG. Para esto cada requerimiento representa una medida de función transaccional, y la caracterizamos en términos de DET, FTR y UFP.

#### 5.3.1. Identificación de puntos atípicos

Calculamos la productividad de cada uno de los requerimientos, utilizando como criterio de tamaño de software los puntos de función (UFP) y el esfuerzo real reportado para su implementación. En la Tabla 13 se describe la productividad reportada por cada requerimiento.

Tabla 13. Productividad obtenida en (HP / PF).

<b>Requerimiento</b>	<b>UFP</b>	<b>HP</b>	<b>Productividad (HP / PF)</b>
1.1	6	504,95	84,16
1.2	6	124,00	20,67
2.1	6	128,75	21,46

<b>Requerimiento</b>	<b>UFP</b>	<b>HP</b>	<b>Productividad (HP / PF)</b>
2.2	6	195,51	32,59
2.3	5	166,50	33,30
3.1	4	96,00	24,00
3.2	4	68,00	17,00
3.3	4	57,75	14,44
3.4	6	459,50	76,58
4.1	5	263,45	52,69
4.2	3	39,50	13,17
4.3	3	21,25	7,08
5.1	4	81,50	20,38
5.2	6	63,00	10,50
6.1	3	32,00	10,67
6.2	3	13,75	4,58
6.3	3	10,00	3,33
6.4	3	12,00	4,00
7.1	6	247,25	41,21
7.2	4	72,00	18,00
8.1	4	60,00	15,00
8.2	3	18,50	6,17
8.3	3	22,00	7,33
8.4	3	24,50	8,17
8.5	3	10,00	3,33
8.6	3	15,00	5,00
8.7	6	142,50	23,75
8.8	6	158,00	26,33
9.1	3	40,00	13,33
9.2	3	71,75	23,92
9.3	3	37,00	12,33
9.4	3	31,50	10,50

Seguidamente se utilizó un gráfico de cajas para determinar si alguno de los requerimientos podía considerarse un valor atípico según el tamaño funcional y el esfuerzo real reportado.

En la Figura 9 se muestra con asteriscos los puntos determinamos como valores atípicos correspondientes a los requerimientos 1.1, 3.4 y 4.1.

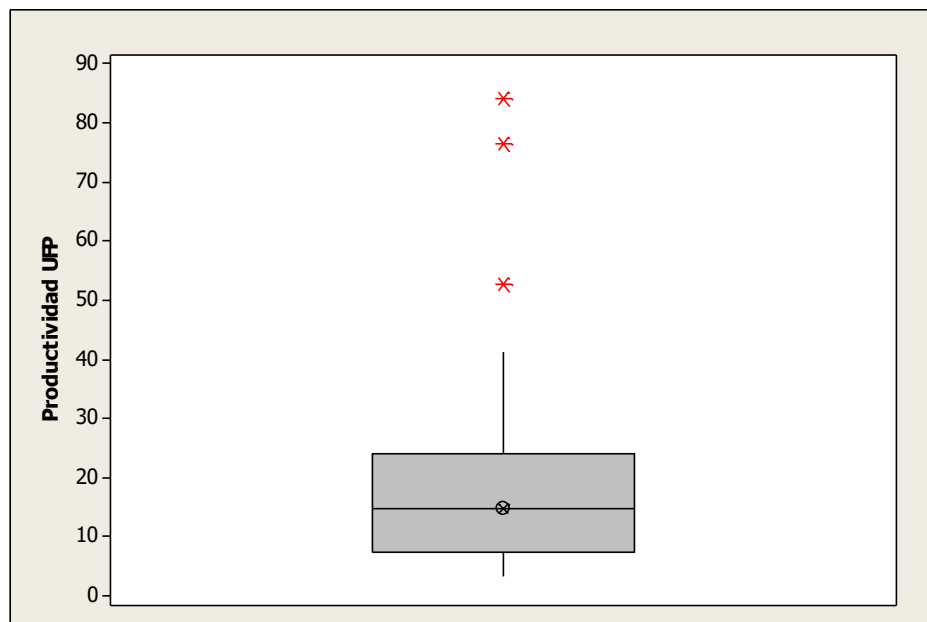


Figura 9. Gráfico de Cajas de Productividad según UFP.

Los requerimientos 1.1 y 3.4 son valores aislados que corresponden a los mismos puntos que fueron identificados para el método de Puntos de Casos de uso, por lo cual aplican las mismas causas asignables. Sin embargo para el punto 4.1 procedimos a destacar su situación, según se muestra en la Tabla 14. En el Capítulo 6 se discutirán más afondo la situación particular de este valor atípico y sus causas asignables.

Tabla 14. Puntos aislados IFPUG.

Requerimiento	Tipo	DET	FTR	UFP	HP
(R4.1) Permitir al usuario continuar con la creación del plan y toma de medidas, sin importar si ya es un usuario registrado.	El	16	3	5	263,45

### 5.3.2. Análisis de regresión lineal

Después de extraer los valores atípicos analizamos nuevamente la muestra y con la intención de identificar algún patrón o tendencia en los datos los graficamos en un diagrama de dispersión, realizamos un gráfico por cada atributo de tamaño de software (DET, FTR y UFP) comparados

contra el esfuerzo real reportado (HP). En la Figura 10 se describe el comportamiento de la muestra utilizando el número de DET como parámetro de tamaño funcional.

En la Figura 11 graficamos el comportamiento de la muestra con base en los FTR, como parámetro de tamaño funcional.

En la Figura 12, se muestra el comportamiento del esfuerzo registrado de tomando los UFP como medida de tamaño.

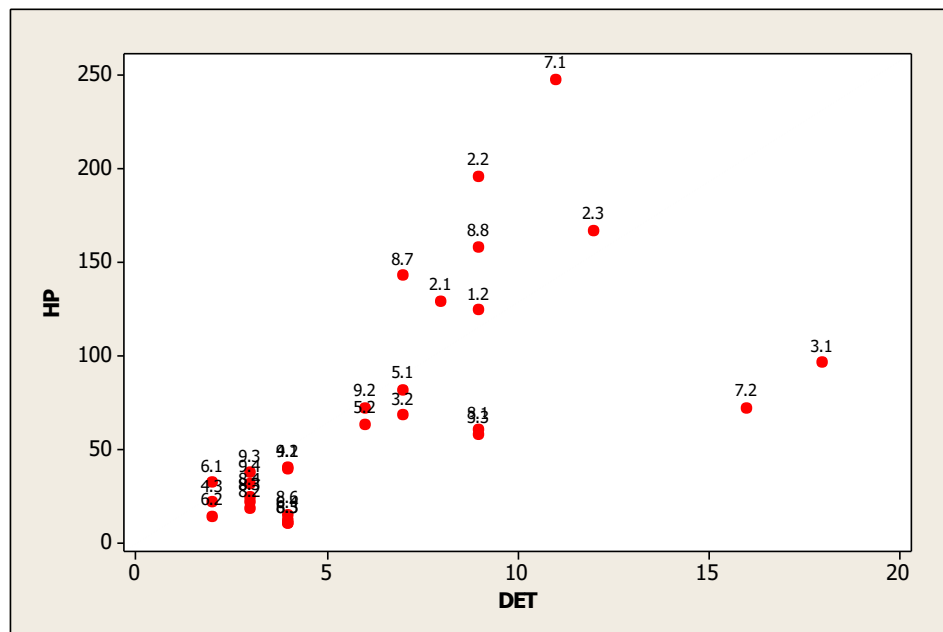


Figura 10. Gráfico de dispersión de muestra con DET.

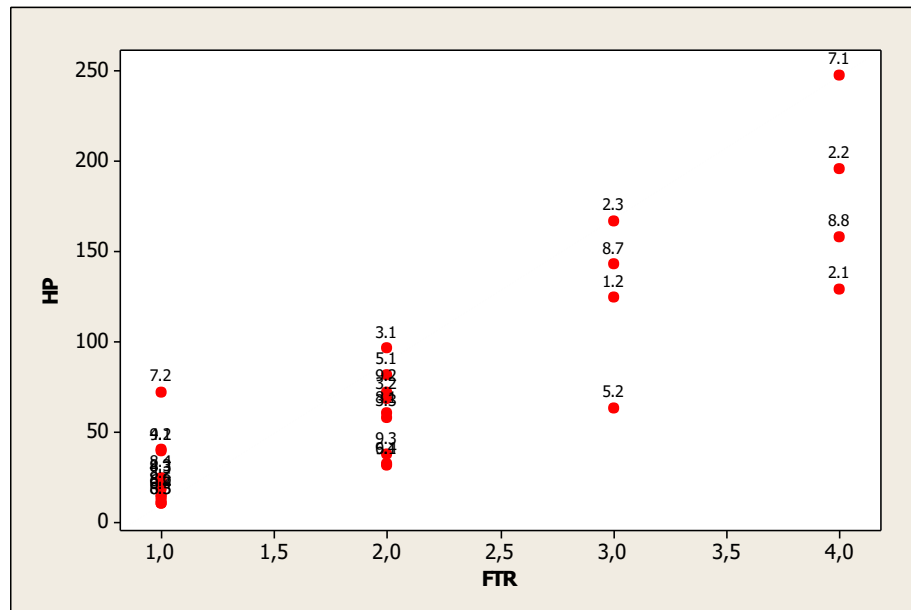


Figura 11. Gráfico de dispersión de muestra con FTR.

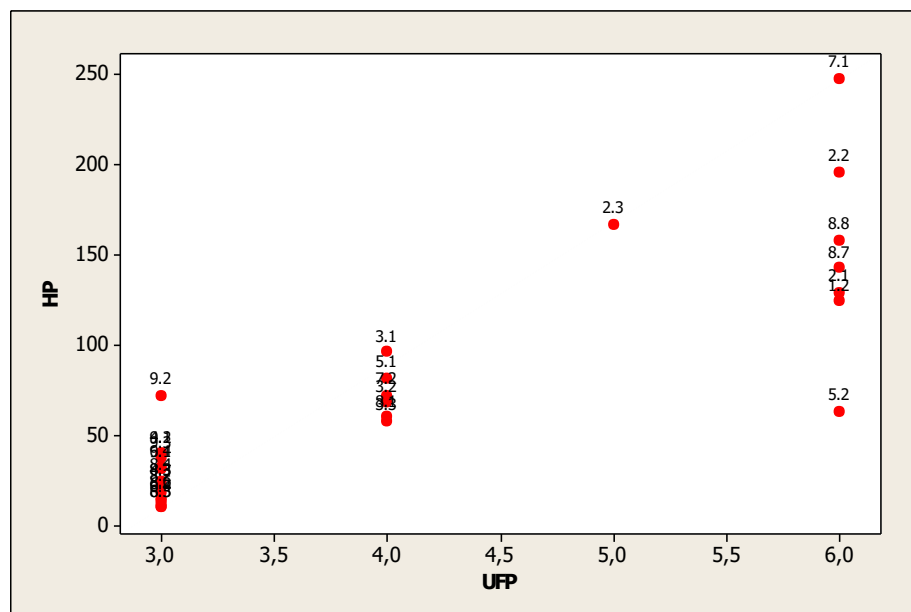


Figura 12. Gráfico de dispersión de muestra con UFP.

Según se observa en la Figura 11 y la Figura 12, el comportamiento de los datos denota una distribución horizontal debido a que los

requerimientos tienden a compartir los mismos valores en cuestión de FTR y UFP.

Sin embargo, los requerimientos en términos de DET tienen una gama más amplia de valores asignados, y es interesante destacar que por observación simple determinamos conglomerados de información, lo que sugiere muestras de poblaciones distintas según la cantidad de DET.

En la Figura 13 se muestra en un recuadro verde y rojo las dos poblaciones sugeridas, según la cantidad de DET contados en el requerimiento. Observamos entonces dos posibles poblaciones: la primera con requerimientos de 2 a 6 DET, y la segunda con un número de DET de 7 a 18. Reiteramos que la conglomeración de datos que segmentaron la muestra en dos poblaciones se realizó por medio de observación simple del gráfico de dispersión mostrado en Figura 13. Como trabajo futuro se puede incluir una prueba estadística que respalde esta observación.

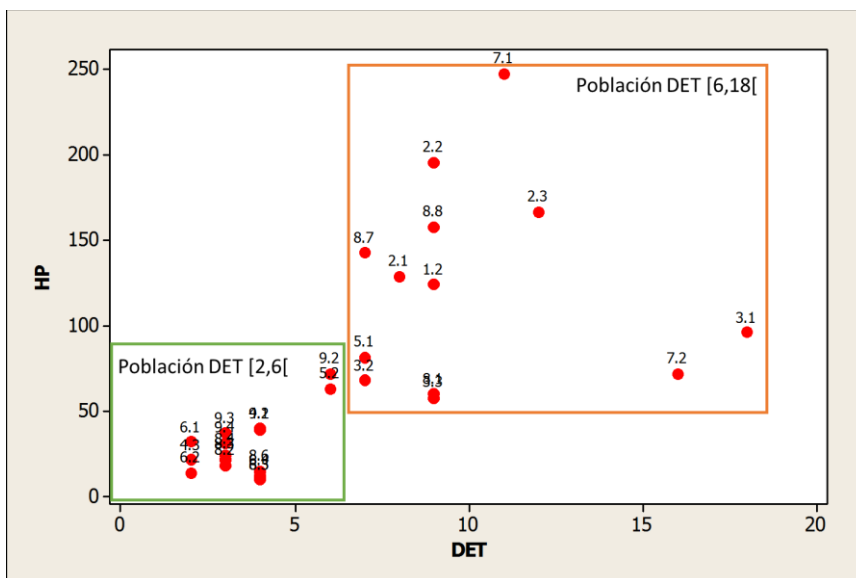


Figura 13. Gráfico de dispersión DET con poblaciones sugeridas.

Después de analizar la muestra desde las diferentes perspectivas, determinamos aplicar análisis de regresión lineal de dos formas:

- Con la muestra completa: creamos un modelo que se predice el esfuerzo a partir de los UFP y a un nivel más granular contemplando los DET y RET que componen ese valor UFP
- Con la muestra segmentada: creamos un modelo que predice el esfuerzo pero aplicado a los contextos poblacionales, considerando un comportamiento diferente para valores de DET de 2 a 6 , y la segunda con DET de 7 a 18

Debido a lo anterior, los análisis de regresión realizados para este método comprendieron los modelos descritos en la Tabla 15.

Tabla 15. Modelo de regresión lineal para IFPUG.

Segmentación muestra	Variable de Respuesta	Dependientes		Modelo
Muestra completa	Esfuerzo estimado (EFF)	UFP		$EFF = - 100 + 42,9 UFP$
Muestra completa	Esfuerzo estimado (EFF)	DET	FTR	$EFF = - 46,2 + 4,51 DET + 44,0 FTR$
Muestra DET [2,6[	Esfuerzo estimado (EFF)	DET	FTR	$EFF = 1,7 + 1,47 DET + 13,9 FTR$
Muestra DET [6,18]	Esfuerzo estimado (EFF)	DET	FTR	$EFF = - 86,6 + 5,93 DET + 53,3 FTR$

### 5.3.3. Cálculo del error medio

Debido a que contemplamos diferentes perspectivas para estimar el esfuerzo, calculamos el indicador de MMRE y MBRE por cada uno de los modelos, los que se muestran en la Tabla 16.

Tabla 16. Error medio de modelos estimación IFUPG.

Segmentación	Modelo	MMRE	MBRE
Muestra completa	$EFF = - 100 + 42,9 UFP$	0,48	0,53
Muestra completa	$EFF = - 46,2 + 4,51 DET + 44,0 FTR$	0,36	0,54
Muestra DET [2,6[	$EFF = 1,7 + 1,47 DET + 13,9 FTR$	0,31	0,45
Muestra DET [6,18]	$EFF = - 86,6 + 5,93 DET + 53,3 FTR$	0,23	0,36

## 5.4. Estimación con Puntos de función Cosmic

Los mismos requerimientos estudiados los contabilizamos desde la perspectiva de COSMIC, resultando en total 174 CFP (puntos de función COSMIC). De manera resumida en la Tabla 17, se muestra la descripción de la muestra a razón de los atributos COSMIC.

Tabla 17. Resumen del conteo COSMIC.

<b>Atributo</b>	<b>Cantidad</b>
E (Entradas)	53
X (Salidas)	41
W (Escrituras)	42
R (Lecturas)	38
CFP	174

#### 5.4.1. Identificación de puntos aislados

Tomando como referencia de tamaño de software los puntos de función COSMIC (CFP) por requerimiento, calculamos la productividad de cada uno. Los resultados se presentan en la Tabla 18.

Tabla 18. Productividad de Puntos de función COSMIC.

<b>Requerimiento</b>	<b>CFP</b>	<b>HP</b>	<b>Productividad HP / CFP)</b>
1.1	13	504,95	38,84
1.2	7	124,00	17,71
2.1	7	128,75	18,39
2.2	8	195,51	24,44
2.3	7	166,50	23,79
3.1	7	96,00	13,71
3.2	6	68,00	11,33
3.3	5	57,75	11,55
3.4	10	459,50	45,95
4.1	9	263,45	29,27
4.2	4	39,50	9,88

<b>Requerimiento</b>	<b>CFP</b>	<b>HP</b>	<b>Productividad HP / CFP)</b>
4.3	3	21,25	7,08
5.1	6	81,50	13,58
5.2	5	63,00	12,60
6.1	4	32,00	8,00
6.2	2	13,75	6,88
6.3	2	10,00	5,00
6.4	2	12,00	6,00
7.1	9	247,25	27,47
7.2	6	72,00	12,00
8.1	5	60,00	12,00
8.2	3	18,50	6,17
8.3	3	22,00	7,33
8.4	3	24,50	8,17
8.5	2	10,00	5,00
8.6	3	15,00	5,00
8.7	7	142,50	20,36
8.8	7	158,00	22,57
9.1	5	40,00	8,00
9.2	6	71,75	11,96
9.3	4	37,00	9,25
9.4	4	31,50	7,88

Seguidamente graficamos los datos de productividad en un diagrama de caja con el fin de identificar algún valor atípico. En la Figura 14, se determina que los requerimientos 1.1 y 3.4 son valores que deben ser extraídos de esta muestra, denotados con asteriscos.

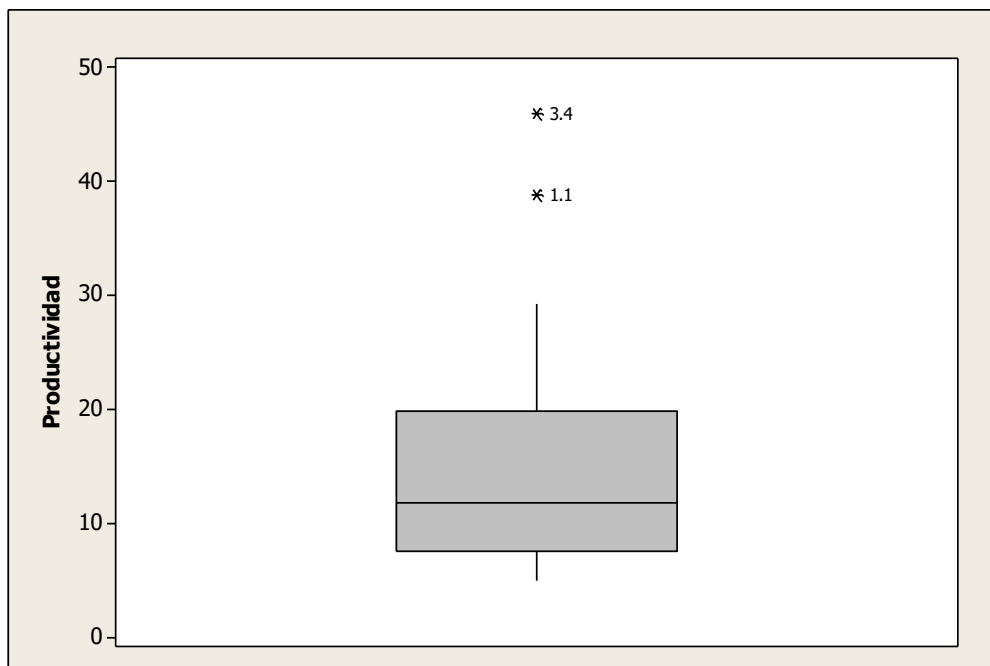


Figura 14. Diagrama de caja productividad CFP.

Los requerimientos 3.4 y 1.1 los habíamos identificado como valores atípicos tanto como para los modelos de UCP y UFP, por lo que su análisis de causas asignables correspondientes lo cubrimos previamente.

#### 5.4.2. Análisis de regresión

Realizamos un escrutinio de la muestra para dejar por fuera los valores que contemplamos como valores atípicos. A partir de esa nueva muestra se estudia su comportamiento en términos de sus atributos (E,X,W,R y CFP) en contraposición del esfuerzo real invertido (HP). En la Figura 15 se observa el comportamiento de muestra en términos de la cantidad de Escrituras (E).

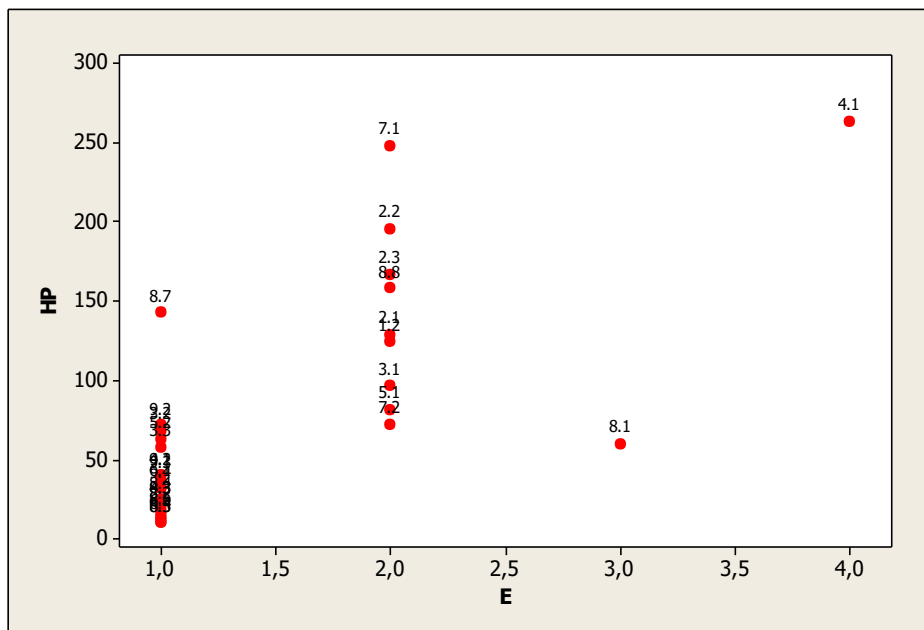


Figura 15. Gráfico de dispersión de Entradas (E).

En la Figura 16, se muestran los requerimientos y su dispersión con respecto a las salidas que se identificaron.

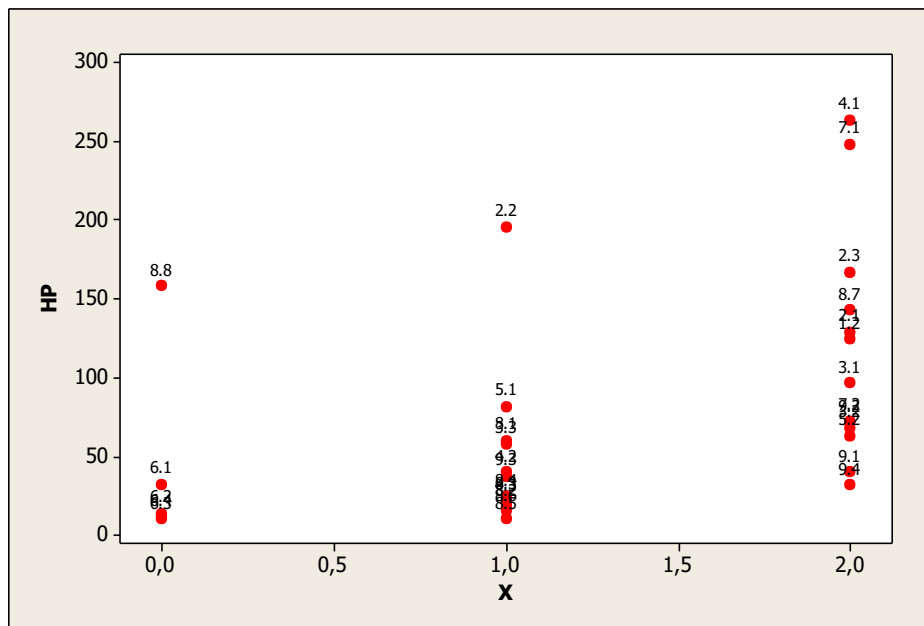


Figura 16. Gráfico de dispersión de Salidas (X).

En la Figura 17 se muestran los datos tomando como referencia la cantidad de lecturas de cada requerimiento.

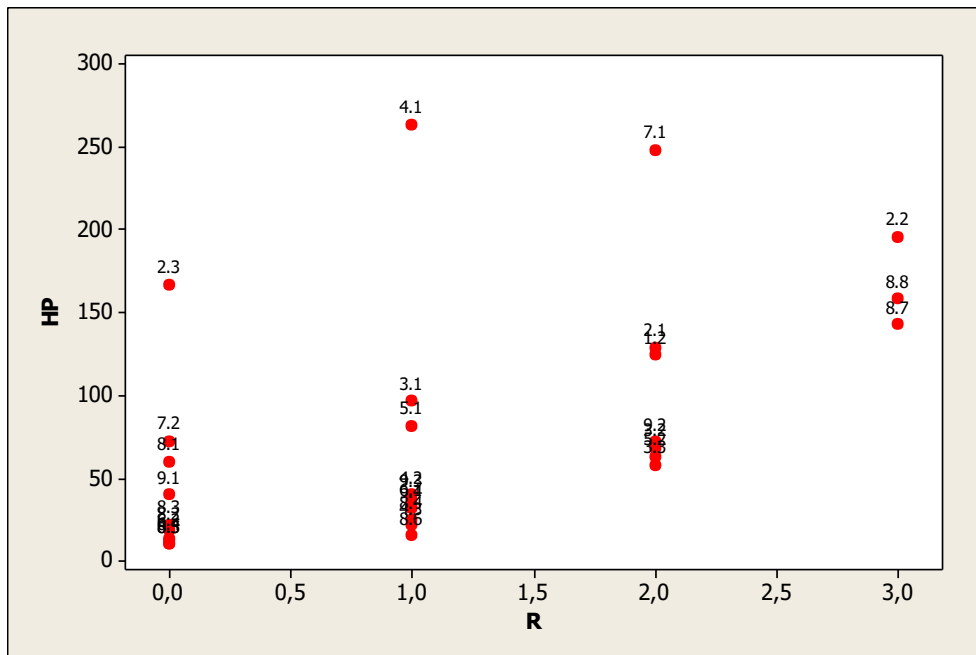


Figura 17. Gráfico de dispersión de Lecturas (R).

En la Figura 18, se observamos la muestra desde la perspectiva de escrituras (W).

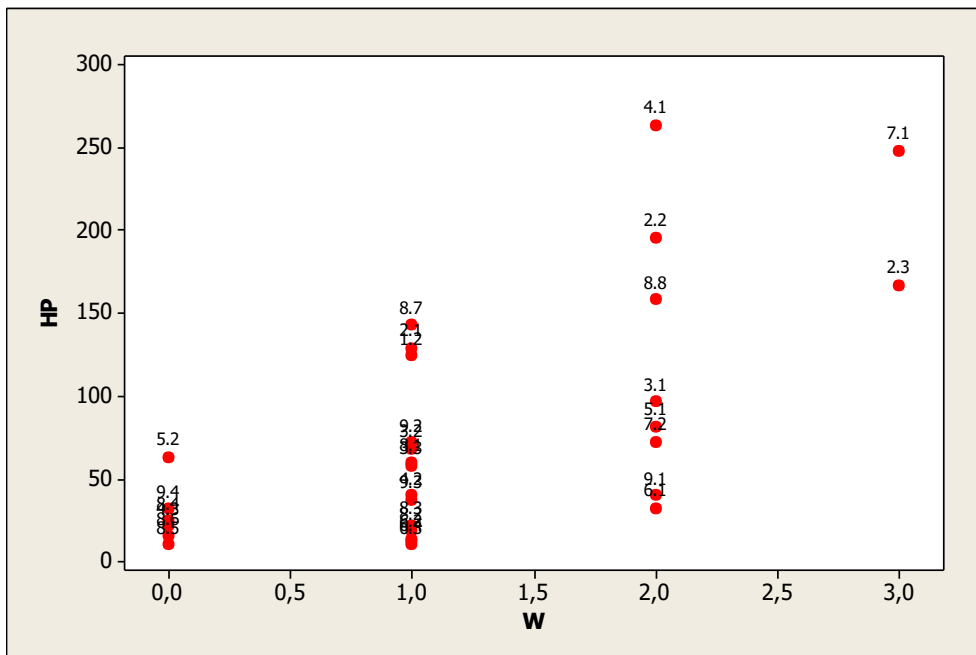


Figura 18. Gráfico de dispersión de Escrituras (W).

En la Figura 19 graficamos la muestra pero utilizando los CFP totales que involucran los cuatro atributos de COSMIC (E,X,R,W).

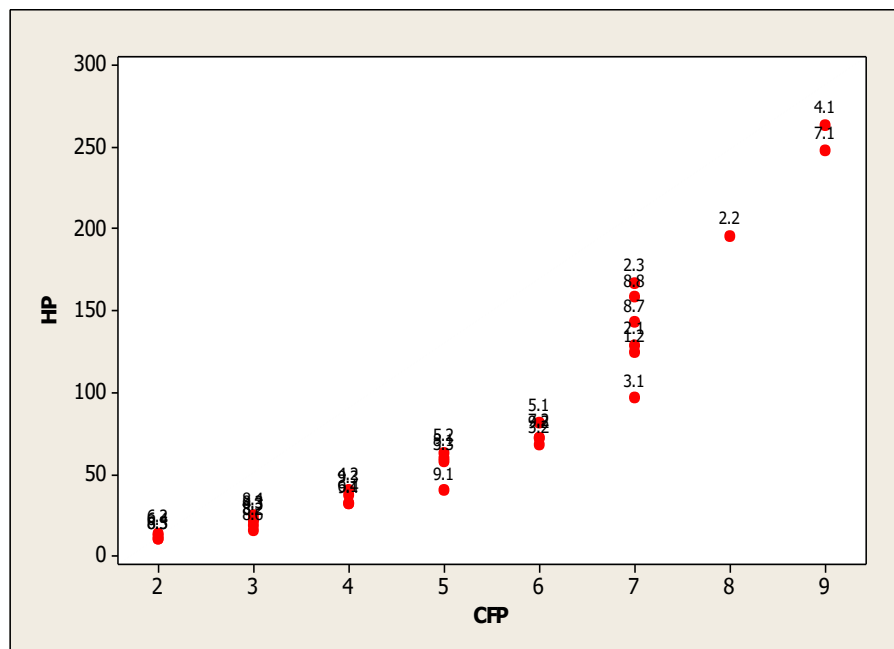


Figura 19. Gráfico de dispersión de Puntos COSMIC (CFP).

Los gráficos de dispersión de cada uno de los atributos separados, es decir la Figura 15, Figura 16, Figura 17 y Figura 18, no reflejan una tendencia o conglomeración particular que nos dieran indicio para hacer algún análisis adicional entorno alguna de esas variables aisladas.

Sin embargo, la Figura 19 muestra una distribución interesante puesto que además de tener una pendiente positiva, su curvatura sugería un ajuste con análisis de regresión no lineal de tipo exponencial. Por lo cual, para este modelo en particular incluimos un análisis de regresión no lineal con ajuste a un modelo exponencial. Al no notar este indicio de comportamiento exponencial en los otros modelos, COSMIC fue el único con el que se probó este tipo de ajuste.

Debido a lo anterior los modelos propuestos para este método de estimación fueron los descritos en la Tabla 19.

Tabla 19. Modelos de regresión para COSMIC.

Tipo Regresión	Variable de Respuesta	de Dependientes	Modelo
Lineal	Esfuerzo estimado (EFF)	CFP	$EFF = - 78,6 + 31,0 CFP$
Lineal	Esfuerzo estimado (EFF)	E, X, R, W	$EFF = - 79,4 + 45,1 E + 31,4 R + 30,8 W + 15,1 X$
No lineal (Exponencial)	Esfuerzo estimado (EFF)	CFP	$EFF = 0,789382 * CFP ^{2,63169}$

#### 5.4.1. Cálculo del MMRE y MBRE

Como siguiente paso, calculamos el indicador de MMRE por cada uno de los modelos, lo que se muestran en la Tabla 20.

Tabla 20. Error Medio de Modelos COSMIC

Modelo	MMRE	MBRE
$EFF = - 78,6 + 31,0 CFP$	0,58	0,51
$EFF = - 79,4 + 45,1 E + 31,4 R + 30,8 W + 15,1 X$	0,47	0,79
$EFF = 0,789382 \times CFP^{2,63169}$	0,22	0,35

En Tabla 21, se muestra un consolidado de los modelos de estimación creados, junto con los las métricas de error MMRE y MBRE de cada uno. La discusión sobre estos resultados la desarrollamos en el Capítulo 6.

Tabla 21. Resumen de Modelos de estimación de esfuerzo.

#	Modelo	MMRE	MBRE
1	$EFF = -51,0 + 14,7 UCP$	0,86	0,98
2	$EFF = -13,1 + 9,86 USP$	0,45	0,87
3	$EFF = 18,47 + 3,392 USP + 0,1650 USP^2$	0,41	0,45
4	$EFF = -8,75 + 12,30 USP - 0,3977 USP^2 + 0,008921 USP^3$	0,36	0,58
5	$EFF = -100 + 42,9 UFP$	0,48	0,53
6	$EFF = -46,2 + 4,51 DET + 44,0 FTR$	0,36	0,54
7	$EFF = 1,7 + 1,47 DET + 13,9 FTR$	0,31	0,45
8	$EFF = -86,6 + 5,93 DET + 53,3 FTR$	0,23	0,36
9	$EFF = -78,6 + 31,0 CFP$	0,58	0,51
10	$EFF = -79,4 + 45,1 E + 31,4 R + 30,8 W + 15,1 X$	0,47	0,79
11	$EFF = 0,789382 \times CFP 2.63169$	0,22	0,35

### 5.5. Modelos de conversión entre métodos

En esta sección presentamos los resultados de magnitud de error media que obtuvimos al aplicar ecuaciones de regresión lineal para convertir tamaño de software entre los diferentes métodos de estimación incluidos en esta investigación.

Estos modelos de conversión ofrecen un método alternativo para la empresa estudiada, pues para obtener el tamaño de software de los requerimientos en las distintas unidades consideradas en nuestro caso de estudio, se pueden utilizar las formulas de conversión, evitando así ejecutar todo el protocolo de conteo específico para cada método de estimación de tamaño.

De esta forma, la empresa puede convertir datos históricos expresados por ejemplo en USP y transformarlos a CFP, para realizar algún estudio adicional o bien obtener una mejor estimación de esfuerzo. En el

Capítulo 6 se describe cómo se relacionan estos modelos de conversión con el estudio de (Lavazza & Liu, 2018) y las implicaciones de su uso.

Tomando en como referencia el estudio de (Lavazza & Liu, 2018), en el contexto de este estudio se define como: “convertibilidad entre tamaño de software” a la posibilidad de generar un modelo de conversión entre medidas de tamaño de software con errores aceptables de MMRE y MBRE, según los parámetros establecidos en el Capítulo 2.

En la Tabla 22 se muestra el resultado de aplicar regresión lineal a nuestra muestra para derivar los modelos de conversión entre los diferentes métodos de estimación de tamaño analizados en este estudio. La discusión sobre estos resultados la desarrollamos en el Capítulo 6.

Tabla 22. Modelos de conversión entre métodos de estimación.

#	Convertir a	desde	Modelo	MMRE	MBRE
1	UFP	UCP	$UFP = 2,37 + 0,195 UCP$	0,21	0,24
2	UFP	USP	$UFP = 2,96 + 0,113 USP$	1,57	4,01
3	UFP	CFP	$UFP = 1,52 + 0,506 CFP$	0,14	0,15
4	UCP	USP	$UCP = 5,57 + 0,324 USP$	0,18	0,20
5	UCP	CFP	$UCP = 3,09 + 1,09 CFP$	0,21	0,24
6	UCP	UFP	$UCP = 2,24 + 1,57 UFP$	0,24	0,28
7	USP	CFP	$USP = - 10,3 + 4,05 CFP$	4,73	4,73
8	USP	UFP	$USP = - 16,8 + 6,88 UFP$	8,37	8,37
9	USP	UCP	$USP = - 8,56 + 2,25 UCP$	4,72	4,72
10	CFP	UFP	$CFP = - 0,435 + 1,36 UFP$	0,38	0,38
11	CFP	UCP	$CFP = 1,72 + 0,392 UCP$	0,35	0,39
12	CFP	USP	$CFP = 3,05 + 0,203 USP$	0,23	0,25

En el siguiente capítulo se discuten estos resultados, contestado las dos preguntas de investigación planteadas en este estudio.

## Capítulo 6. Discusión de resultados

En esta sección analizamos en detalle los resultados de nuestro estudio y realizamos recomendaciones para la empresa contemplada en el estudio.

### 6.1. Métodos más precisos

Con respecto a la **RQ1**, encontramos que según el MMRE y MBRE el modelo 11 de la Tabla 21 (basado en los Puntos de Función COSMIC) produce las estimaciones de esfuerzo más precisas. Al comparar la precisión de los modelos COSMIC, observamos que el modelo de estimación basado en CFP (modelo 11 en la Tabla 21) funciona ligeramente mejor que el modelo basado en UFP (modelo 8 en la Tabla 21). Sin embargo, el modelo 8 es aplicable solo a los requisitos con seis o más elementos de datos (DET). Por lo tanto, solo el modelo basado en puntos de función COSMIC demostró ser lo suficientemente preciso y aplicable a cualquier requisito, independientemente de su tamaño o número de elementos de datos.

Estos resultados corroboran los hallazgos de (Salmanoglu et al., 2017), quienes concluyeron que un modelo de estimación de esfuerzo basado en puntos de función COSMIC era más preciso que otro modelo basado en User Story Points. También utilizaron MMRE como el indicador de exactitud para los modelos, lo que hace que nuestros resultados sean comparables con los de ellos.

Nuestros resultados también corroboran la evidencia existente de (Ungan et al., 2014), quienes también descubrieron que el modelo de esfuerzo basado en puntos de función COSMIC tuvo un mejor desempeño que el modelo basado en User Story Points. Además del análisis de

regresión, incluyeron un enfoque de redes neuronales artificiales para construir sus modelos, pero obtuvieron resultados similares a los nuestros.

Adicionalmente, nuestro estudio se asemeja al trabajo de (Paz et al., 2014) en dos aspectos. Primero, obtuvimos resultados de exactitud similares ya que determinamos que el modelo de estimación de esfuerzo más preciso se basa en puntos de función COSMIC como medida de tamaño funcional. En segundo lugar, seguimos un enfoque similar para estimar el tamaño del software: ambos usamos puntos de función COSMIC incrementales, principalmente porque la aplicación se desarrolló utilizando una metodología SCRUM, y este enfoque permite aplicar CFP en entornos ágiles.

También para contestar la **RQ1**, encontramos que el modelo 1 de la Tabla 21 (basado en puntos de casos de uso) produce las estimaciones de esfuerzo menos precisas. Este hallazgo difiere de los resultados obtenidos por (Sholiq et al., 2017), ya que encontraron un desempeño ligeramente mejor utilizando los puntos de caso de uso. Sin embargo, hay dos diferencias principales entre los estudios que podrían haber afectado los resultados. Primero, nosotros utilizamos un análisis de regresión para construir los modelos de estimación del esfuerzo, mientras que ellos utilizaron un factor de productividad (calculado por ellos) para ese escenario específico. En segundo lugar, ellos midieron el tamaño de los proyectos ya completados, mientras que nosotros medimos el tamaño de un proyecto ágil en curso (piezas incrementales de software).

## **6.2. Correlación entre los métodos**

Con respecto a la **RQ2**, pudimos obtener modelos de convertibilidad precisos para algunas mediciones de tamaño funcional, pero no para cada par de métodos de medición de tamaño. Los modelos de convertibilidad que pueden usarse de manera segura en el contexto de la

organización son los modelos 1, 2, 3, 7, 9 o 12 de la Tabla 22, dado que su error de estimación medido como MMRE es menor que 0,25 y MBRE es menor que 0,35. Estos resultados se asemejan con los hallazgos de (Lavazza et al. 2018) donde presentaron un modelo que permite la convertibilidad de CFP a UFP (en nuestro estudio, es el modelo 9 de la Tabla 22). Sin embargo, en el caso de la convertibilidad de UFP a CFP, (Lavazza et al. 2018) mostró evidencia que lo permite, mientras que nosotros no pudimos encontrar un modelo preciso para ello (el modelo 10 de la Tabla 22 no fue lo suficientemente preciso).

### *6.2.1. Estimar con modelo COSMIC a partir de USP*

Según describimos en el Capítulo 1, la empresa donde realizamos este estudio utiliza SCRUM como marco de referencia para administrar sus proyectos. Dentro de la ceremonia de planificación se utilizan puntos de historias de usuario para medir el tamaño de software. Debido a lo anterior, un modelo que permita transformar USP en CFP es lo más conveniente para después determinar el esfuerzo.

De acuerdo a lo que mostramos en la Tabla 22, el modelo de conversión de USP hacia puntos COSMIC es:

$$CFP = 3,05 + 0,203 USP$$

Que tiene una magnitud de error relativa (MMRE) de 0,23, lo cual sugiere que es un modelo con relativamente buena exactitud.

Por lo tanto, en caso de que la organización quisiera estimar el esfuerzo sin necesariamente hacer un proceso formal de conteo CFP, puede utilizar los puntos USP del mismo equipo de trabajo que estimó las historias de esta muestra y transformarlas a CFP. Seguidamente, podría predecir el esfuerzo requerido en horas utilizando el modelo exponencial CFP, definido como:

$$EFF = 0,789382 \times CFP^{2,63169}$$

### 6.2.1.1. Consideraciones de la estimación con conversión previa

Contemplando los modelos propuestos que dimos a la organización en términos de estimación, existen dos alternativas sugeridas:

1. Estimar esfuerzo utilizando directamente los USP.
2. Estimar esfuerzo realizando previamente una conversión a CFP.

Según nuestra investigación, la primera opción tiene asociado una magnitud de error relativo MMRE del 0,36.

En contraposición, la segunda opción ofrece lograr una estimación con una magnitud de error de 0,22, a partir de los CFP contabilizados. Sin embargo, debe pasarse por primero por un proceso de conversión de USP a CFP teóricos, que aporta una magnitud de error de 0,23 adicional.

Debido a lo anterior, aplicamos la ley de aditiva o regla de la suma de eventos mutuamente excluyentes para determinar cuál sería la magnitud de error total esperada aplicando ambos modelos:

$$\text{Sea } A: USP \rightarrow CFP; B: CFP \rightarrow EFF$$

$$P(A \cup B) = P(A) + P(B)$$

$$P(A \cup B) = 0,23 + 0,22 = 0,45$$

Por lo tanto, la magnitud de error esperada al aplicar ambos modelos de manera consecutiva es de 0,45, siendo un error mayor al que ofrece la opción 1 de hacer la estimación de manera directa utilizando USP.

## 6.3. Análisis de causa raíz de valores atípicos

Los distintos análisis de regresión propuestos evidenciaron ciertos valores atípicos que en esta sección vamos a revisar más a fondo, para lograr fundamentar la situación particular de esos requerimientos.

En la Tabla 23 se muestra el compilado de los requerimientos que fueron estadísticamente identificados como valores atípicos. Adicionalmente, se describe desde cuál método fue identificado, ya sea UCP, UFP o CFP.

Tabla 23. Valores atípicos identificados de la muestra.

<b>Requerimiento</b>	<b>HP</b>	<b>Identificado</b>
(R1.1) Crear un usuario nuevo en la plataforma usando correo electrónico e información personal	504,95	UCP, UFP, CFP
(R2.3) Mantener sesión del usuario activa, para que el usuario pueda ingresar a la aplicación rápidamente.	166,5	UCP
(R3.4) Después de establecer la relación paciente-nutricionista, el usuario puede cancelar dicha conexión	459,3	UCP, UFP, CFP
(R4.1) Permitir al usuario continuar con la creación del plan y toma de medidas, sin importar si ya es un usuario registrado.	16	UFP

Los requerimientos R2.3 y R4.1 son valores considerados atípicos desde la perspectiva de un solo método de estimación, es decir (R2.3) desde el punto de vista UCP y (R4.1) desde el aspecto UFP son descritos como puntos aislados. Sin embargo, al no existir una recurrencia de anomalía en los otros dos métodos, consideramos que la causa asignable es circunstancial y se le atribuye al contexto del método de medición utilizado, ya sea un error en el conteo de UCP o UFP.

Los requerimientos (R1.1) y (R3.4) los analizamos con la herramienta de causa raíz de Ishikawa, según se muestra en la Figura 20 y Figura 21.

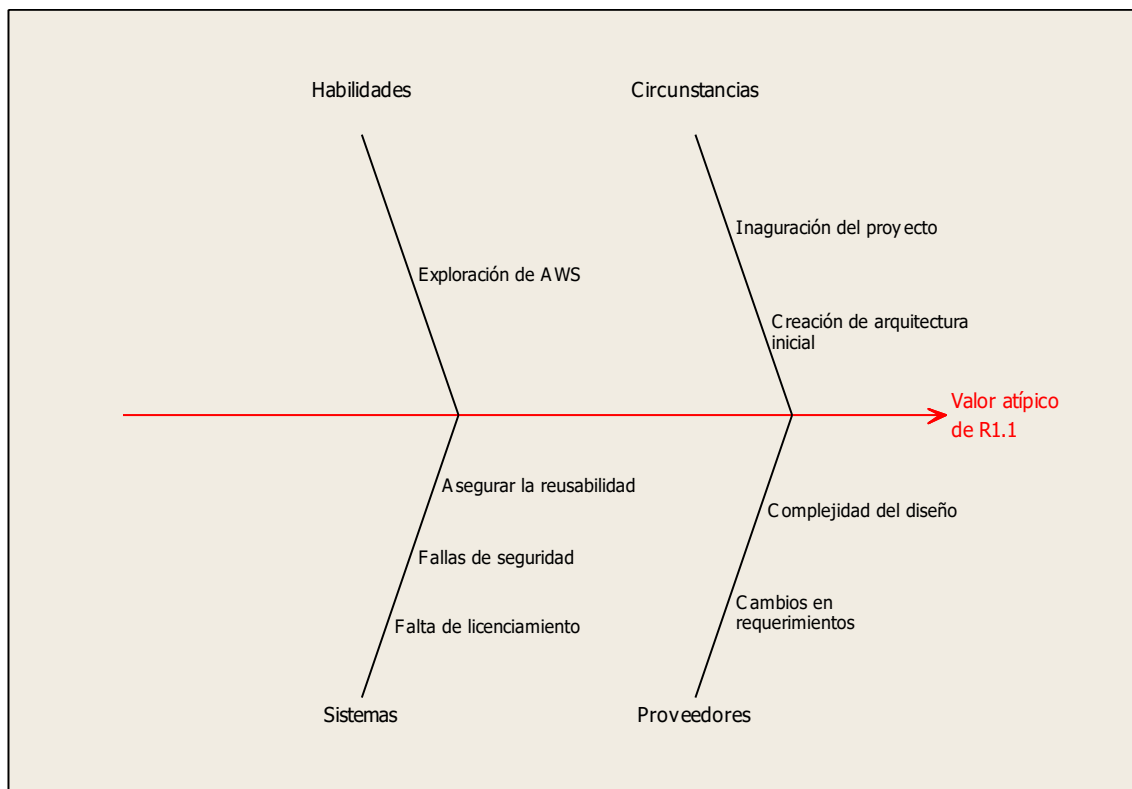


Figura 20. Diagrama causa raíz de R1.1.

El requerimiento R1.1 fue analizado desde cuatro aspectos que comprenden:

- **Habilidades:** el equipo de desarrollo no contaba con experiencia previa en la plataforma de servicios de la nube AWS, por cual esto implicó una curva de aprendizaje importante para determinar cómo construir la plataforma deseada desde el inicio con un enfoque diferente al habitual.
- **Circunstancias:** Este fue el primer requerimiento desarrollado por el equipo, por lo cual todo el ajuste desde el punto de vista administrativo que implica la inauguración de un proyecto y fue asumido en esta primera entrega. Adicional, existió un costo técnico de diseñar la arquitectura ideal para el alcance del proyecto a desarrollar, incluyendo detalles de bases de datos, patrones de diseño, interacciones entre

sistemas, e integraciones con terceras partes que se debían de contemplar.

- **Sistemas:** Desde el punto de vista de tecnología, existieron retos costos iniciando por el licenciamiento requerido, a medida que se iban incluyendo nuevas partes del código el equipo viendo la necesidad de incluir algunas herramientas que implicaban adquirir licenciamiento. Iniciar primero con plataformas abiertas para luego migrar a plataformas pagadas produjo un retrabajo. Adicionalmente, existieron factores de seguridad que se fueron evidenciando a medida que se avanzaba con el desarrollo y que, al no considerarse desde el inicio, agregaron complejidad al diseño original, temas relacionados a autenticación de usuarios e certificados. Otro factor que se descubrió fue la reusabilidad de los componentes, al ser el primer requerimiento mucho del código que se creó no era de uso único sino que estaba pensado para ser reutilizado en futuros componentes. por lo que el costo del desarrollo de estas partes abstractas se sumó a este primer requerimiento.
- **Proveedores:** Al tratarse de la primera experiencia con el producto y la plataforma, los requerimientos se estuvieron ajustando en el camino. Constantes cambios de requerimientos se experimentaron durante el desarrollo de este primer requerimiento. Adicional, se subcontrató el diseño de las interfaces de usuario a un tercero, que sumó más cambios a los requerimientos inicialmente planteados.

El análisis de requerimiento R3.4 se describe en la Figura 21 , de los cuales se detalla:

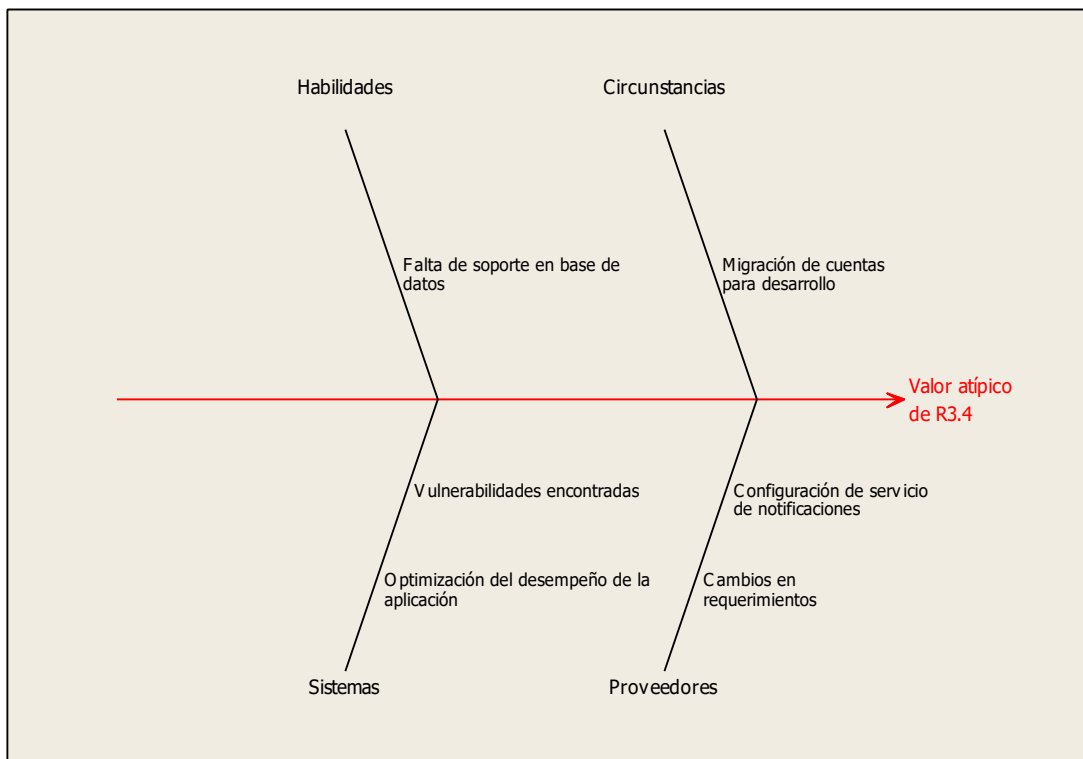


Figura 21. Diagrama de causa raíz de R3.4.

- **Habilidades:** Este requerimiento implicó un cambio a nivel de bases de datos considerablemente grande, el cual no se anticipó en el diseño de manera inicial. Incluir este cambio en base de datos para soportar la funcionalidad del requerimiento R3.4 reflejó un retrabajo considerable pues se debió de ajustar varias tablas, campos y relaciones que fue costoso para el equipo. La habilidad de pensar en diseños escalables es un factor que fue considerado desde este punto en adelante.
- **Circunstancias:** En el contexto del desarrollo del requerimiento R3.4 se cambiaron las cuentas, debido a un cambio estratégico del nombre de la compañía. Implicó migrar contenido entre cuentas y pruebas de regresión para descartar inyección de defectos por esta causa.
- **Sistemas:** el enfoque de los sistemas utilizados en la nube conllevó a optimizar el factor de desempeño de la plataforma pues el cobro se

da por procesamiento, por lo que un ahorro en el consumo de datos beneficia las finanzas de la empresa. Cambios a nivel de desarrollo fueron requeridos para mejorar este aspecto. Adicionalmente, este requerimiento evidenció vulnerabilidades a nivel de seguridad que fueron reparadas de inmediato.

- Proveedores: En este requerimiento se experimentó una cantidad inusual de cambios en los requerimientos, durante las ceremonias de Revisión y validación con el usuario final se cambiaron varios elementos que contemplaron trabajo adicional importante. Agregado a esto, el requerimiento implicaba la integración por primera vez de un sistema externo para notificaciones, lo que implicó validación de opciones y licenciamiento adicional.

Debido a las causas asignadas que descubrimos en esos requerimientos, se justifican los requerimientos R1.1 y R3.4 como valores atípicos de la muestra y se respalda su extracción de todas las muestras. Para el caso de los requerimientos R2.3 y R4.1 se consideran como puntos aislados únicamente dentro del dominio del método de conteo utilizado, es decir R2.3 en UCP y R4.1 en UFP.

#### **6.4. Amenazas a la validez**

En esta sección se contemplan los factores de confiabilidad de los resultados obtenidos. A través de este caso de estudio se han expuesto los diferentes factores que podrían amenazar el estudio, sin embargo, a continuación se presentan los hechos de frente y se analizan a profundidad.

Evaluamos la validez de nuestro estudio con base en las pautas de (Runeson, et al., 2008), considerando la validez de constructo, la validez interna, la validez externa y la confiabilidad, como se muestra a

continuación. Las amenazas a la validez se mencionan brevemente a continuación.

- *Validez del constructo*: en nuestro estudio de caso, creamos un conjunto de modelos de estimación de esfuerzo basados en la medición del tamaño funcional para un contexto organizacional específico. Evaluamos su desempeño de precisión utilizando MMRE y MBRE, métricas que permiten comparar entre sí modelos y determinar el modelo más exacto para hacer la estimación en el contexto de la empresa.
- *Validez interna*: una amenaza de validez es el pequeño tamaño de muestra utilizado (que solo tiene 32 requisitos para analizar). Además, podría haberse introducido un sesgo en el proceso de selección de la muestra. Además, algunos datos se descartaron del análisis ya que se identificaron como valores atípicos. Otra amenaza es que solo un investigador realizó la medición del tamaño funcional para cada requisito. Sin embargo, este investigador tiene experiencia comprobada con estos métodos de conteo. Finalmente, MMRE como una medida estadística para evaluar el desempeño de un modelo a veces no ha logrado mostrar evidencia estadística, como lo aconsejaron (Kitchenham et al., 2009) y (Lavazza et al., 2017). Intentamos mitigar esta amenaza comparando también la precisión en combinación con MBRE, que ha demostrado más eficiencia, según (Jørgensen M., 2007)
- *Validez externa*: los modelos de estimación de esfuerzo son aplicables solo a la organización descrita en este estudio.

Estos modelos no están destinados a ser utilizados en otros contextos además de este.

- *Fiabilidad*: dado que el conjunto de datos de la organización creció con el tiempo, existe una amenaza de validez de que este estudio de caso presente resultados no repetibles, como lo señalaron (Lavazza et al., 2017) A medida que la organización se amplía, los resultados pueden variar.

## Capítulo 7. Conclusiones

En esta sección se reúnen los principales hallazgos del caso de estudio, complementado con la relación del estudio con la evidencia existente en la literatura, el impacto e implicaciones, limitaciones del estudio y trabajo futuro.

Describimos un estudio de caso que compara la precisión de los modelos de estimación de esfuerzo derivados del análisis de regresión a partir de cuatro mediciones de tamaño funcional: puntos de historia de usuario (USP), puntos de caso de uso (UCP), puntos de función IFPUG (UFP) y puntos de función COSMIC (CFP).

Adicionalmente, presentamos un conjunto de modelos de convertibilidad entre mediciones de tamaño funcional, que se pueden usar para convertir fácilmente las mediciones históricas de la organización en diferentes medidas de tamaño de software.

Nuestros resultados confirman los hallazgos de estudios previos como (Ungan, E., et al 2014), (Paz, F., et al, 2014) y (Salmanoglu, et al, 2017), que concluyeron que los modelos de estimación de esfuerzo basados en CFP muestran una mayor precisión que los métodos alternativos en comparación con sus estudios.

Por otra parte, ninguno de los modelos de convertibilidad es lo suficientemente preciso como para estimar el tamaño del software expresado en USP. Por lo tanto, en la organización en estudio, las ceremonias de la metodología ágil actuales deben continuar realizándose para estimar el tamaño del trabajo.

Nuestros modelos de convertibilidad reafirman evidencia previa de (Lavazza et al., 2018), ya que presentaron un escenario en el que UFP y

CFP se correlacionaron, y se propone un modelo para convertir de un método de dimensionamiento de software a otro.

A pesar de que el criterio de juicio experto sigue siendo generalmente preferido como método de estimación en entornos ágiles (Usman, et al., 2015), nuestra recomendación a la organización en estudio es adoptar el conteo COSMIC (CFP) como su método de estimación de tamaño funcional, especialmente a medida que crece el número de proyectos y la empresa aumenta de tamaño. Adicional a los resultados obtenidos en esta investigación, hay abundante evidencia en la literatura de que el uso de CFP podría mejorar las estimaciones de una organización (Desharnais, J., et al, 2011), (Ungan, et al, 2014), (Paz, et al, 2014) y (Salmanoglu, et al., 2017).

### **7.1. Limitaciones del estudio**

- La cantidad y tipo de datos considerados en el estudio. La muestra que se obtuvo puede ser considerada pequeña, como para inferir resultados para toda la población.
- Solo se consideraron 4 métodos de estimación de esfuerzo, que fueron seleccionados basado en la experiencia del investigador encargado de los conteos con protocolos de cada uno, sin embargo, en la literatura existen más métodos que podrían reflejar mejor la situación de esta empresa.

### **7.2. Trabajo futuro**

Como trabajo futuro, planificamos replicar este estudio utilizando un conjunto de datos más grande con una muestra ampliada de requisitos funcionales. Otra posible extensión de nuestro trabajo sería realizar un análisis estadístico para comparar los modelos de estimación de esfuerzo, incluyendo las pruebas estadísticas sugeridas por (Lavazza, et al., 2017)

como las pruebas de T pareada de MRE o las pruebas pareadas de los residuos absolutos.

Producto de esta investigación derivamos un artículo (ver Anexo 9.7) que fue aceptado en la conferencia CIBSE 2020, celebrada en Curitiba, Brasil.

## Capítulo 8. Referencias

1. Bente, A., Hege, D., Sjøberg, D., & Magne, J.: Estimating Software Development Effort Based on Use Cases - Experiences from Industry: The Unified Modeling Language. *Modeling Languages, Concepts, and Tools* (2001).
2. Briand, L., Maxwell, K.: An Assessment and Comparison of Common Software Cost Estimation Modeling Techniques: *ICSE '99* (1999).
3. Chi-Jui, L., & Yeh, D.-M.: A Software Maintenance Project Size Estimation Tool Based On Cosmic Full Function Point: *2016 International Computer Symposium (ICS)* (2016).
4. Desharnais, J., Buglione, L. & Kocaturk, B.: Using the COSMIC Method to Estimate Agile User Stories: *Proceedings of the 12th International Conference on Product Focused Software Development and Process Improvement* (2011).
5. Foss, T., Stensrud, E., Kitchenham, B., Myrtveit, I.: A Simulation Study of the Model Evaluation Criterion MMRE: *IEEE Transactions on Software Engineering* (2003).
6. Gandomani, T. J., Mahsa Radnejad, H. F. Planning Poker in cost estimation in Agile methods: Averaging Vs. Consensus: *IEEE 5th International Conference on Knowledge-Based Engineering and Innovation* (2019).
7. Garmus, D., Herron, D.: *Function Point Analysis: Measurement Practices for Successful Software Projects*: Addison-Wesley (2001)
8. Jørgensen, M., Boehm, B., Rifkin, S. Software development effort estimation: Formal models or expert judgment? : *IEEE software* (2009).
9. Jørgensen, M.: A critique of how we measure and interpret the accuracy of software development effort estimation: *Proceedings of 1st International Workshop on Software Productivity Analysis and Cost Estimation* (2007)
10. Karner, G. *Metrics for Objectory* Diploma thesis: University of Linköping, (1993)
11. Kitchenham, B., Mendes, E.: Why comparative effort prediction studies may be invalid. In *Proceedings of the 5th International Conference on Predictor Models in Software Engineering: PROMISE '09* (2009).
12. Lavazza, L., & Liu, G.: A Study of the Correlation between Functional Size Measures and Object-oriented Measures from UML Requirements Models. *IWSM Mensura*, (2018)
13. Lavazza, L., & Morasca, S.: On the Evaluation of Effort Estimation Models. *EASE'17*. (2017)
14. Lenarduzzi, V., Lunesu, I., Matta, M, Taibi, D.: Functional Size Measures and Effort Estimation in Agile Development: A Replicated Study. In Suomalainen, T. *Continuous Strategy Process in the context of Agile and Lean Software Development*, Springer (2015)
15. Mendes, E., Watson, I., Triggs, C.: A Comparative Study of Cost Estimation Models for Web Hypermedia Applications. *Empirical Software Engineering* (2003).
16. Mittas N., Angelis, L.: 2010. Visual comparison of software cost estimation models by regression error characteristic analysis: *J. Syst. Softw.* 83, 4 (2010)

17. Mittas, N., & Angelis, L.: Ranking and Clustering Software Cost Estimation Models through a Multiple Comparisons Algorithm: *IEEE Transactions on Software Engineering* Vol. 39, No. 4, (2013)
18. Miyazaki, Y., et al., Robust regression for developing software estimation models: *Journal of Systems and Software*, (1994)
19. Paz, F., Zapata, C., Pow-Sang, J. A.: An Approach for Effort Estimation in Incremental Software Development using COSMIC Function Points. *ESEM'14* (2014).
20. Petersen, K., Vakkalanka, S., Kuzniarz, L., Guidelines for conducting systematic mapping studies in software engineering: An update, *Information and Software Technology*, (2015).
21. Phannachitta, P. Robust Comparison of Similarity Measures in Analogy-Based Software Effort Estimation: *11th International Conference on Software, Knowledge, Information Management and Applications (SKIMA)* (2017)
22. Popli, R., Chauhan, N. Managing Uncertainty of Story-Points in Agile Software: 2nd International Conference on Computing for Sustainable Global Development (INDIACom)(2015)
23. Quesada-López, C. & Jenkins, M.: Estudio sobre las prácticas de la ingeniería del software en Costa Rica: Resultados preliminares.: Proceedings of the XX Ibero-American Conference on Software Engineering (CibSE 2017). (2017)
24. Runeson, P., & Höst, M: Guidelines for conducting and reporting case study research in software engineering: *Empiric Software Eng.* (2008)
25. Salmanoglu, M., Hacaloglu, T., Demirors, O.: Effort Estimation for Agile Software Development: Comparative Case Studies Using COSMIC Functional Size Measurement and Story Points, *IWSM/Mensura '17*, (2017)
26. Santana C., Leoneo F., Vasconcelos A., Gusmão C. Using Function Points in Agile Projects. (2011)
27. Sholih, Renny, S., & Pribadi, A. A Comparative Study of Software Development Size Estimation Method: UCPabc vs Function Points: *4th Information Systems International Conference 2017*, (2017)
28. Urgan, E., Çizmeli, N., & Demirörs, O.: Comparison of Functional Size Based Estimation and Story Points, Based on Effort Estimation Effectiveness in SCRUM Projects: *Euromicro Conference on Software Engineering and Advanced Applications*, (2014)
29. Usman, M., Mendes, E., Börstler, J. Effort Estimation in Agile Software Development: A Survey on the State of the Practice. *EASE'15*, (2015)
30. Wilkie, F., McChesney, I., Morrow P., Tuxworth, C., Lester, N.: The value of software sizing: *Information and Software Technology* (2011).

## Capítulo 9. Anexos

### 9.1. Protocolo de revisión de literatura

Este anexo describe el protocolo seguido para lograr la revisión de la literatura. El protocolo seguido es la versión abreviada de la propuesta por (Petersen, Vakkalanka, & Kuzniarz, 2015) y está compuesta por la definición de preguntas de investigación, cadena de búsqueda, criterios de inclusión y exclusión, y criterios de calidad. Además de los artículos encontrados con la ejecución del protocolo, se incorporaron posteriormente algunos estudios de interés durante el desarrollo de la investigación.

#### *9.1.1. Preguntas de investigación*

Las preguntas de investigación planteadas para la revisión de literatura fueron las siguientes:

- **RQ1** ¿Cómo se han comparado los métodos de estimación de tamaño de software en la literatura?
- **RQ2** ¿Cuáles estudios han abordado la convertibilidad entre medidas de tamaño de software?

La primera pregunta ayuda a entender el estado del arte en cuanto a comparaciones previas de métodos de estimación de software. La segunda pregunta identifica los estudios existentes sobre modelos de conversión entre diferentes medidas de tamaño de software.

#### *9.1.2. Cadena de búsqueda*

La cadena de búsqueda utilizada fue:

“software” AND (“estimation methods” or “sizing” or “cost estimation” or “estimation models”) AND (“comparison” or “correlation” or “evaluation” or “comparative”)

Las bases de datos sobre las cuales se ejecutó la cadena fueron:

- SpringerLink
- ACM
- IEEE Xplore Digital Library
- Proquest
- Science Direct

La Tabla 24, muestra la cantidad de estudios recuperados en cada base de datos.

Tabla 24. Cantidad de estudios primarios recuperados por base de datos.

<b>Base de datos</b>	<b>Número de estudios</b>
SpringerLink	5
ACM	7
IEEE Xplore Digital Library	12
Proquest	8
Science Direct	5

### *9.1.3. Proceso de inclusión y exclusión*

#### Criterios de inclusión:

- El artículo se enfoca explícitamente en comparar 2 o más métodos de estimación de software.
- En artículo está escrito en inglés o español.

#### Criterios de exclusión:

- El estudio se encuentra fuera del dominio de la ingeniería de software.
- El estudio no aborda la comparación entre métodos de estimación de software

### 9.1.1. Proceso evaluación de calidad

Se definieron los siguientes criterios de calidad, con su respectivo puntaje según afinidad con las preguntas de investigación:

#### A. Tipo de estudio:

- Investigación de validación (0 pts): Técnicas de comparación de métodos de estimación de software utilizadas no se han implementado, y son utilizadas con experimentos.
- Evaluation Research (1 pt): Las técnicas de comparación de métodos de estimación de software se implementan en la práctica y se realiza una evaluación de la técnica. Eso significa que se muestra cómo se implementa la técnica en la práctica (implementación de la solución) y cuáles son las consecuencias de la implementación en términos de beneficios e inconvenientes (evaluación de la implementación).
- Propuesta de solución (2 pts): Se propone una solución para comparar métodos de estimación de software, la solución puede ser novedosa o una extensión significativa de una técnica existente. Los beneficios potenciales y la aplicabilidad de la solución se muestran con un pequeño ejemplo o una buena línea de argumentación.
- Artículos de experiencia (3 pts): explican qué y cómo se ha hecho las comparaciones de métodos de estimación de software de en la práctica. Interpretado desde la experiencia personal del autor.

#### B. Relación con métodos de comparación:

- No se presenta ningún método de comparación (0 pts).: Artículo no se expone la metodología del método utilizado cuantitativo o cualitativo para comparar métodos de estimación de software. Incluye opiniones propias del autor sobre los mismos.

- Se propone un método de comparación (1 pts). Se presenta una metodología sistematizada y reproducible para comparar métodos de estimación de software.
  - Se evalúan dos o más métodos de comparación (2 pts). Se presenta una metodología sistematizada y reproducible para comparar entre sí, métodos que comparan los métodos de estimación de software.
- C. Robustez de métricas:
- Ninguna métrica se usa para comparar los modelos de estimación propuestos (0 pts)
  - Se incluyen métricas para minimizar error (1pt).
  - Se incluyen métricas de precisión de predicción (2 pts).
  - Se incluyen métricas de precisión de predicción y minimización de las métricas de error (3pts).
- D. Validación empírica:
- No presenta validación empírica (0 pts): Artículo no hace referencia a experimentos, o aplicaciones empíricas de las técnicas propuestas. Por lo tanto la argumentación es basada en teorías únicamente.
  - Presenta validación empírica (1 pt): Artículo presenta evidencia empírica sobre la aplicación del método de comparación propuesto.
- E. Criterios de las conclusiones:
- Las conclusiones no muestran evidencia suficiente para declarar que un modelo de estimación es competente (0 pts)
  - Las conclusiones declaran un modelo de estimación como proficiente (1 pt)
  - Las conclusiones declaran qué modelo de estimación es mejor y qué método de comparación es competente. (2 pts)

### 9.1.2. Proceso de selección de estudios

Después de identificar los estudios primarios iniciales, el proceso de selección de estudios lo realizamos siguiendo estos pasos:

1. Se realizaron búsquedas en los estudios primarios en las bases de datos mencionadas y se descartaron los estudios duplicados.
2. Se analizaron el título y el resumen de cada documento y se aplicaron los criterios de inclusión y exclusión.
3. Se asignaron puntajes a los estudios según los criterios de calidad definidos
4. Los artículos se ordenaron de mayor a menor puntaje de calidad
5. Los estudios con puntaje menor a 6pts se clasifican como Rechazados o Pendientes.
6. Los estudios con el estado "Pendiente" se leen a fondo para determinar si pueden ser aceptados. Si es así, el estado cambia a Aceptado.
7. Todos los estudios "aceptados" se procesaron para extraer los datos correspondientes para responder las preguntas de investigación.

## **9.2. Análisis de regresión a nivel de historias de usuario**

Tomando como muestra las 119 historias de usuario se realiza el mismo análisis de regresión, para predecir esfuerzo a partir de los USP.

### *9.2.1. Identificación de puntos aislados*

Identificamos los valores atípicos de esta muestra de 119 historias, calculando la productividad de cada historia de usuario.

Tabla 25. Productividad de Historias de usuario.

Historia Usuario	USP	Reported Hours	Productividad
1	3	11,00	3,67
2	2	8,00	4,00
3	1	15,00	15,00
4	5	49,25	9,85
5	1	6,00	6,00
6	3	29,50	9,83
7	3	16,00	5,33
8	2	8,00	4,00
9	3	61,50	20,50
10	5	56,00	11,20
11	2	45,00	22,50
12	3	35,00	11,67
13	2	52,00	26,00
14	3	42,50	14,17
15	3	20,00	6,67
16	3	50,20	16,73
17	8	124,00	15,50
18	5	53,00	10,60
19	3	20,00	6,67
20	5	18,50	3,70
21	5	37,25	7,45
22	2	9,75	4,88
23	2	9,00	4,50
24	2	13,00	6,50
25	8	131,26	16,41
26	3	17,00	5,67
27	2	15,50	7,75
28	5	55,00	11,00
29	2	12,00	6,00
30	3	10,00	3,33
31	8	45,00	5,63
32	8	32,50	4,06
33	3	12,00	4,00
34	2	14,00	7,00

<b>Historia Usuario</b>	<b>USP</b>	<b>Reported Hours</b>	<b>Productividad</b>
35	3	26,00	8,67
36	5	42,00	8,40
37	2	9,00	4,50
38	1	5,00	5,00
39	3	35,00	11,67
40	2	10,00	5,00
41	3	23,00	7,67
42	3	37,00	12,33
43	3	20,75	6,92
44	8	75,25	9,41
45	5	114,25	22,85
46	8	102,50	12,81
47	2	14,00	7,00
48	5	56,00	11,20
49	2	15,00	7,50
50	5	63,75	12,75
51	3	8,00	2,67
52	1	10,75	10,75
53	2	17,00	8,50
54	3	10,00	3,33
55	8	108,70	13,59
56	5	48,00	9,60
57	5	44,50	8,90
58	1	8,00	8,00
59	2	4,50	2,25
60	1	22,75	22,75
61	1	2,00	2,00
62	2	7,25	3,63
63	1	2,50	2,50
64	3	27,75	9,25
65	2	14,00	7,00
66	1	3,25	3,25
67	3	4,00	1,33
68	2	8,00	4,00
69	5	73,50	14,70
70	2	14,00	7,00
71	5	49,00	9,80
72	3	14,00	4,67

<b>Historia Usuario</b>	<b>USP</b>	<b>Reported Hours</b>	<b>Productividad</b>
73	2	18,00	9,00
74	2	13,75	6,88
75	3	4,00	1,33
76	1	6,00	6,00
77	1	12,00	12,00
78	1	4,00	4,00
79	8	41,00	5,13
80	1	5,00	5,00
81	3	20,75	6,92
82	2	17,75	8,88
83	3	35,00	11,67
84	3	30,50	10,17
85	3	25,00	8,33
86	2	11,00	5,50
87	8	57,25	7,16
88	3	27,00	9,00
89	3	24,00	8,00
90	5	14,00	2,80
91	1	7,00	7,00
92	3	47,00	15,67
93	3	13,00	4,33
94	2	18,50	9,25
95	3	22,00	7,33
96	3	24,50	8,17
97	1	10,00	10,00
98	1	15,00	15,00
99	2	17,50	8,75
100	3	25,50	8,50
101	5	28,00	5,60
102	5	31,00	6,20
103	3	37,50	12,50
104	2	3,00	1,50
105	5	30,00	6,00
106	3	23,00	7,67
107	2	43,00	21,50
108	3	15,00	5,00
109	3	22,00	7,33
110	5	25,00	5,00
111	1	19,00	19,00
112	3	21,00	7,00
113	2	24,00	12,00

Historia Usuario	USP	Reported Hours	Productividad
114	5	47,75	9,55
115	3	26,00	8,67
116	3	11,00	3,67
117	1	1,00	1,00
118	3	13,00	4,33
119	2	17,50	8,75

Utilizamos el gráfico de caja para identificar los valores atípicos de la muestra, tal y como se muestra en la Figura 22.

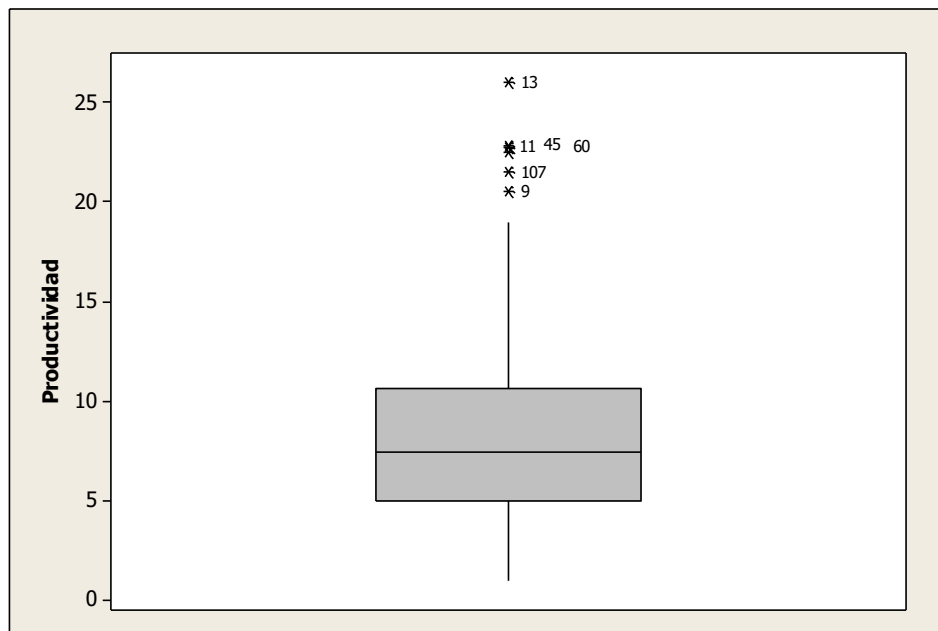


Figura 22. Diagrama de caja de productividad según historias de usuario (n=119).

### 9.2.2. Análisis de regresión

Excluyendo de la muestra los valores atípicos, los datos se graficaron en un diagrama de dispersión, para valorar su patrón de comportamiento de datos a nivel visual.

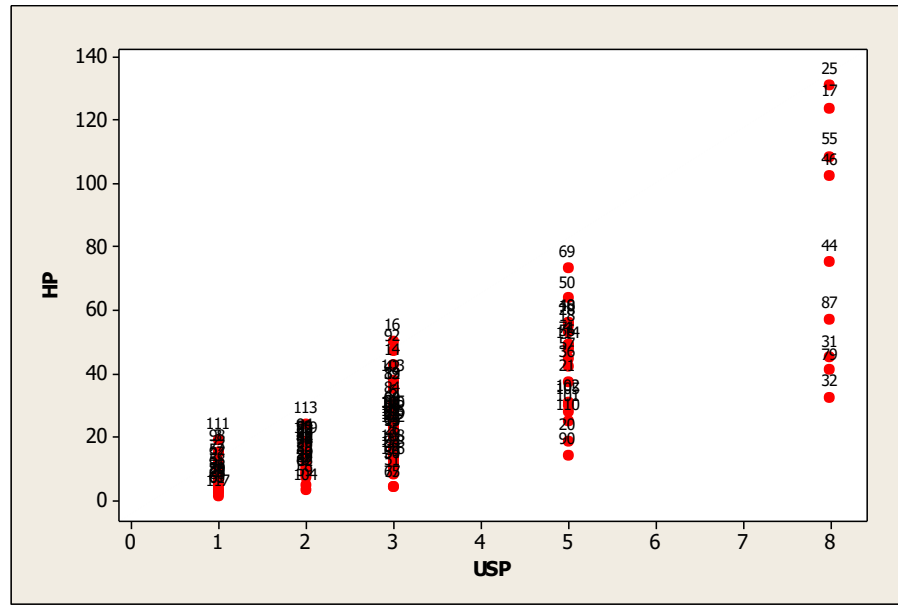


Figura 23. Historias de usuario en tamaño vs esfuerzo real.

Después de inspeccionar el comportamiento de los datos de la muestra, se determina que los datos tienen una tendencia incremental y ejecutamos el análisis de regresión lineal, además la cuadrática y cúbica para replicar el mismo análisis que con los datos agrupados en el Capítulo 5.2.2. Los modelos de regresión conjunto a sus respectivos MMRE y MBRE los presentamos en la Tabla 26.

Tabla 26. Modelos de regresión y MMRE para USP.

Modelo	MMRE	MBRE
$EFF = - 6,935 + 10,34 USP$	0.56	1.58
$EFF = - 0,036 + 6,064 USP + 0,4945 USP^2$	1.49	1.49
$HP = 3,251 + 2,573 USP + 1,478 USP^2 - 0,0757 USP^3$	0.66	2.13

### 9.3. Conteo Puntos de función IFPUG

La lista de entidades y atributos de la aplicación analizada la presentamos en la Tabla 27.

Tabla 27. Lista de entidades y atributos.

Entidad	Atributos	Sistema
User	"Email": "User@nutri-fi.com", "Gender": 1, "DateOfBirth": "1992-01-05" "TemporalCode": "9EIUYTBB"	AWS
Nutritionist	"NationalId": "258910365", "Phone": "84527566", "Country": 1, "FirstName": "Hernaldo", "LastName": "Rivas", "DateOfBirth": "1988-10-10", "Gender": 1	AWS
Patient	idPatient, fkPerson, fkUser, TemporalCode	AWS
Clinic	idClinic, fkCity, ClinicName, StreetAddress, Phone, email	AWS
education	idEducation, fkDoctor, MedLicience, Description	AWS
bodymeasur ement	idBodymeasurement, fkIdPatient, height, idealWeight,fatPercentaje, MusclePercentaje, weight, currentDate	AWS
measure	idMeasurement, fkBodymeasurement, fatPercentaje, MusclePercentaje	AWS
nutritionpla n	idNutritionplan, fkPatient, createdDate, enabled, Name	AWS
meal	idMeal, fkNutritionplan, fkday, description, hour	AWS
ingredientp ermeal	fkIngredient, fkMeal, portion	AWS
ingredient	idIngredient, fkFoodGroup, Ingredient	AWS
foodGroup	idFoodGroup, fkIngridient	AWS
allergy	fkPatient, fkIngredient	AWS
country	idCountry, country	AWS
gender	idGender, gender	AWS
person	idPerson, fkCountry, fkGender, fistName, lastname, secondname, date, phone, email, streetaddress	AWS

La lista de funciones de datos los describimos en la Tabla 28.

Tabla 28. Lista de funciones de datos.

Entidad	Uso primario	Tipo	DET	DET	RET	RET	Complejidad	UF P
User	Third party	EIF	Email, Gender, DOB, TemporalCode	4	Users, Nutritionist, Patient	3	Low	5
Nutritionist	Third party	EIF	NationalId, Phone, CountryId, FirstName, LastName, DOB, Gender	7	Nutritionist	1	Low	5
Patient	Third party	EIF	idPatient, fkPerson, fkUser, TemporalCode	4	Patient	1	Low	5
Clinic	Third party	EIF	idClinic, fkCity, ClinicName, StreetAddress, Phone, email	6	Primary clinic, secondary clinic	2	Low	5
education	Third party	EIF	idEducation, fkDoctor, MedLience, Description	4	education	1	Low	5
bodymeasurement	Third party	EIF	idBodymeasurement, fkIdPatient, height, idealWeight, fatPercentage, MusclePercentage, weight, currentDate	8	bodymeasurement	1	Low	5
measure	Third party	EIF	idMeasurement, fkBodymeasurement, fatPercentage, MusclePercentage	4	measure	1	Low	5
nutritionplan	Third party	EIF	idNutritionplan, fkPatient, createDate, enabled, Name	5	nutritionplan	1	Low	5
meal	Third party	EIF	idMeal, fkNutritionplan, fkday, description, hour	5	meal	1	Low	5
ingredient	Third party	EIF	idIngredient, fkFoodGroup, Ingredient	3	ingredient, foodGroup	2	Low	5
foodGroup	Third	EIF	idFoodGroup,	2	foodGroup	1	Low	5

Entidad	Uso primario	Tipo	DET	DET	RET	RET	Complejidad	UFP
	party		fkIngredient					
allergy	Third party	EIF	fkPatient, fkIngredient	2	allergy	1	Low	5
country	Third party	EIF	idCountry, country	2	country	1	Low	5
gender	Third party	EIF	idGender, gender	2	gender	1	Low	5
person	Third party	EIF	idPerson, fkCountry, fkGender, fistName, lastname, secondname, date, phone, email, streetaddress	10	person, patient, nutritionist	3	Low	5

Las medidas de funciones transaccionales las contabilizamos en la Tabla 29.

Tabla 29. Medidas de funciones transaccionales.

Función	Uso primario	Tipo	DET	DET	FTR	FTR	Complejidad	UFP
(R1) Sign in for new user								
(R1.1) User create a new user in Nutri Fi by using its email and some personal information	Maintain	EI	Create new account, Email, Password, Sign up, Phone, Send Code, Code, Verify Code, Country, Next, CPN Code, Verify Code, Gender, Next,	18	Users, Nutritionist, Patient, Clinic, Education, person	6	High	6
(R1.2) Terms and conditions are mandat	Maintain	EI	T/C, Accept, Permissions, Finish, email, idPatient, NationalID, fkPerson, idPerson	9	Users, Nutritionist, Patient	3	High	6

Función	Uso primario	Tiempo	DET	DET	FTR	FTR	Complejidad	UFP
ory to be part of Nutri Fi								
(R2) Login for existing user								
(R2.1) Allows the user to get in the application by entering its credentials, whenever the user session expired.	Maintain	EI	Email, Password, Log In, idPerson, NationalId, fkPerson, fkUser	8	Users, Nutritionist, Patient, person	4	High	6
(R2.2) Allow the user to recover its password.	Maintain	EI	Email, Password, Log In, idPerson, NationalId, fkPerson, fkUser, RecoverButton	9	Users, Nutritionist, Patient, person	4	High	6
(R2.3) User session must remain active, hence the user can access to the application as fast as it select the app in its	Display	EQ	Email, Password, Log In, idPerson, NationalId, fkPerson, fkUser, Phone, Country, CPN Code, idBodymeasurement, idMealPlan	12	Users, Bodymeasurement, NutritionPlan	3	Average	5

Función	Uso primario	Tipo	DET	DET	FTR	FTR	Complejidad	UFP
mobile.								
(R3) Connect nutritionist with a patient								
(R3.1) Allow the nutritionist to add a new patient, by enable the option to search it by its emails address	Maintain	EI	idPerson, fkCountry, fkGender, fistName, lastname, secondname, date, phone, email, streetaddress, Email, DOB, gender, create user, search	15	person, user	2	Average	4
(R3.2) Allow the nutritionist to realize this particular patient have been added into its contact list previously.	Display	EQ	firstname, lastname, email, clinic, gender, age, status	7	Patient, person	2	Average	4
(R3.3) Once the	Display	EQ	firstname, lastname, email, clinic, status, DOB, country, job,	9	person, user	2	Average	4

Función	Uso primario	Tipo	DET	DET	FTR	FTR	Complejidad	UFP
Invitation have been accepted. The nutritionist is able to access its patient profile information.			streetaddress					
(R3.4) Once the nutritionist-patient relationship have been in place, the nutritionist is able to cancel this relationship.	Maintain	EI	idPerson, idNutritionPlan, idBodymeasurement, firstName, lastname, secondname, date, phone, email, streetaddress, Email, DOB, gender, cancel	14	Patient, user, nutritionist, bodymeasurement, nutritionplan	5	High	6
(R4) Create a temporary user								
(R4.1) Allow the user to move forward with the body measurements and meal	Display	EO	idBodymeasurement, fkIdPatient, height, idealWeight, fatPercentage, MusclePercentage, weight, currentDate, idMeasurement, fkBodymeasurement, fatPercentage, MusclePercentage, addmeasurement, done, cancel, back	16	Patient, user, bodymeasurement	3	Average	5

Función	Uso primario	Tipo	DET	DET	FTR	FT R	Complejidad	UFP
plan creation , regardless if the patient is already a Nutri Fi user.								
(R4.2) A temporary user will remain active as long as the Patient haven't authenticate it, and therefore created an actual Nutri Fi user.	Maintain	EI	authenticationtoken, Email, Gender, DOB	4	user	1	Low	3
(R4.3) If the Patient haven't confirm the temporary user, and therefore an actual Nutri Fi user using this email haven't been created. After	Display	EQ	pusktokennotification, message	2	user	1	Low	3

Función	Uso primario	Tipo	DET	DET	FTR	FT R	Complejidad	UFP
each 3 days, a push notification will be delivered to the Nutritionist letting the user know that this Patient haven't access its information yet.								
(R5) Patient profile information								
(R5.1) Allow the nutritionist to see all the patient profile information, as long as they remain in a nutritionist-patient relationship.	Display	EQ	firstname, lastname, email, clinic, status, DOB, country, job	7	Person, Patient	2	Average	4
(R5.2) Allows the nutritionist to interact	Maintain	EI	firstname, lastname, email, clinic, status, DOB	6	Person, Patient, user	3	High	6

<b>Función</b>	<b>Uso primario</b>	<b>Ti po</b>	<b>DET</b>	<b>D ET</b>	<b>FTR</b>	<b>F TR</b>	<b>Compl ejidad</b>	<b>U F P</b>
with a particular patient profile, meaning to create body measurements, create a meal plan, schedule an appointment, etc.								
(R6) Nutritionist profile information								
(R6.1) Allow nutritionist to edit its profile information.	Maintain	EI	clinic, status	2	Person, nutritionist	2	Low	3
(R6.2) Allow the nutritionist to add a new office location	Maintain	EI	clinic name, location	2	clinic	1	Low	3
(R6.3) Allow the nutritionist to edit an existing office location	Maintain	EI	clinic name, location, phone, email	4	clinic	1	Low	3

Función	Uso primario	Tipo	DET	DET	FTR	FTR	Complejidad	UFP
(R6.4) Allow the nutritionist to remove an existing office location	Maintain	El	clinic name, location,select, remove	4	clinic	2	Low	3
(R7) Create set of new body measurements								
(R7.1) Possible body measurement must be shown to the user, so it can be entitled to add a new value related for the body measurement selected.	Maintain	El	idBodymeasurement, fkIdPatient, height, idealWeight,fatPercentage, MusclePorcentaje, weight, fatPorcentaje, MusclePorcentaje, addmeasurement, done	11	user, patient,bodymeasurement, measurement	4	High	6
(R7.2) Allow the user to enter a new body measurement value.	Maintain	El	idBodymeasurement, fkIdPatient, height, idealWeight,fatPercentage, MusclePorcentaje, weight, currentDate, idMeasurement, fkBodymeasurement, fatPorcentaje, MusclePorcentaje, addmeasurement,	16	bodymeasurement,	1	Average	4

Función	Uso primario	Ti po	DET	D ET	FTR	F TR	Compl ejidad	U F P
			done, cancel, back					
(R8) Create a new meal plan								
(R8.1) Create plan name	Maintain	EI	idNutritionplan, fkPatient, createdDate, enabled, status Name, create, cancel,back, clear	9	nutritionplan, patient	2	Average	4
(R8.2) Adjust plan factor	Maintain	EI	physicalfactor, thermalfactor, stressfactor	3	nutritionplan	1	Low	3
(R8.3) Calculate the basal metabolic rate and energy requirement rate	Maintain	EI	energyrequirement,prescribedrequirement, next	3	nutritionplan	1	Low	3
(R8.4) Adjust the protein requirement factor	Maintain	EI	proteinrequirement, dailyrequirement, currentweight	3	nutritionplan	1	Low	3
(R8.5) Allocate macronutrients distribution	Maintain	EI	carbs, protein,fat, next	4	meal	1	Low	3
(R8.6) Select nutritional	Maintain	EI	idIngredient,idNutritionPlan,standard, next	4	ingredient,	1	Low	3

<b>Función</b>	<b>Uso primario</b>	<b>Ti po</b>	<b>DET</b>	<b>D ET</b>	<b>FTR</b>	<b>F T R</b>	<b>Compl ejidad</b>	<b>U F P</b>
standard plan								
(R8.7) Allocate nutrients "trade-off" (swapping process)	Maintain	EI	calories,remaining, assigned, carbs, protein,fat, ingredient	7	ingredient, foodgroup, allergy	3	High	6
(R8.8) Create a daily meal plan	Maintain	EI	mealtime, calories,mealname, calories,remaining, assigned, carbs, protein,fat, ingredient	9	nutritionplan, meal, day, ingredient, foodgroup, allergy	6	High	6
(R9) Patient history trace								
(R9.1) Everytime a new set of body measurements or a new meal plan are created, those have to be logged for future tracking.	Display	EQ	date, measurementname, measurevalue, idPatient	4	bodymeasurement,	1	Low	3
(R9.2) Allow the nutritionist to consult what was the previous state for a particular	Display	EQ	patient, mealPlan, carbs, protein,fat, patientName	6	person,nutrition plan	2	Low	3

Función	Uso primario	Tipo	DET	DET	FTR	FT R	Complejidad	UFP
ar meal plan displayed at the Patient history trace								
(R9.3) Allow to create a new meal plan by editing a previous meal plan done by its own.	Maintain	EI	date, patient, mealtime	3	patient, nutritionplan	2	Low	3
(R9.4) Allow the nutritionist to consult what was the previous state for a particular set of body measurements displayed at the Patient history trace	Display	EQ	idPatient, idBodymeasurement, measurevalue	3	patient, bodymeasurement	2	Low	3

El resumen de los resultados de puntos de función IFPUG que capturamos para esta aplicación se presenta en la Tabla 30.

Tabla 30. Resumen de resultados de conteo de puntos de función IFPUG.

Total ( $\Sigma$ )	EI	99
Total ( $\Sigma$ )	EO	5
Total ( $\Sigma$ )	EQ	29
Total ( $\Sigma$ )	TF	133
Total ( $\Sigma$ )	ILF	0
Total ( $\Sigma$ )	EIF	75
Total ( $\Sigma$ )	DF	75
	Total UFP $\Sigma$	208

#### 9.4. Conteo Puntos de caso de uso

En la Tabla 31, se muestran los resultados del conteo de puntos de casos de uso para aplicación analizada en nuestro caso de estudio.

Tabla 31. Conteo de puntos de casos de uso

Caso de uso	Descripción	Tipo caso de uso	UUCP	Tipo actor	AW	UCP
1.1	User create a new user in Nutri Fi by using its email and some personal information	Complex	15	Complex	3	18
1.2	Terms and conditions are mandatory to be part of Nutri Fi	Simple	5	Average	2	7
2.1	Allows the user to get in the application by entering its credentials, whenever the user session expired.	Average	10	Simple	1	11
2.2	Allow the user to recover its password.	Average	10	Simple	1	11
2.3	User session must remains active, hence the user can access to the application as fast as it select the app in its mobile.	Simple	5	Simple	1	6
3.1	Allow the nutritionist to add a new patient, by enable the option to	Simple	5	Complex	3	8

Caso de uso	Descripción	Tipo caso de uso	UUCP	Tipo actor	AW	UCP
	search it by its emails address					
3.2	Allow the nutritionist to realize this particular patient have been added into its contact list previously.	Simple	5	Simple	1	6
3.3	Once the Invitation have been accepted. The nutritionist is able to access its patient profile information.	Average	10	Simple	1	11
3.4	Once the nutritionist-patient relationship have been in place, the nutritionist is able to cancel this relationship.	Complex	15	Complex	3	18
4.1	Allow the user to move forward with the body measurements and meal plan creation, regardless if the patient is already a Nutri Fi user.	Average	10	Simple	1	11
4.2	A temporal user will remain active as long as the Patient haven't authenticate it, and therefore created an actual Nutri Fi user.	Average	10	Simple	1	11
4.3	If the Patient haven't confirm the temporal user, and therefore an actual Nutri Fi user using this email haven't been created. After each 3 days, a push notification will be delivered to the Nutritionist letting the user known that this Patient haven't access its information yet.	Simple	5	Simple	1	6
5.1	Allow the nutritionist to see all the patient profile information, as long as they remain in a nutritionist-patient relationship.	Simple	5	Complex	3	8
5.2	Allows the nutritionist to interact with a particular patient profile, meaning	Simple	5	Simple	1	6

Caso de uso	Descripción	Tipo caso de uso	UUCP	Tipo actor	AW	UCP
	to create body measurements, create a meal plan, schedule an appointment, etc.					
6.1	Allow nutritionist to edit its profile information.	Simple	5	Complex	3	8
6.2	Allow the nutritionist to add a new office location	Simple	5	Simple	1	6
6.3	Allow the nutritionist to edit an existing office location	Simple	5	Simple	1	6
6.4	Allow the nutritionist to remove an existing office location	Simple	5	Simple	1	6
7.1	Possible body measurement must be shown to the user, so it can be entitled to add a new value related for the body measurement selected.	Complex	15	Complex	3	18
7.2	Allow the user to enter a new body measurement value.	Simple	5	Simple	1	6
8.1	Create plan name	Simple	5	Simple	1	6
8.2	Adjust plan factor	Simple	5	Simple	1	6
8.3	Calculate the basal metabolic rate and energy requirement rate	Simple	5	Simple	1	6
8.4	Adjust the protein requirement factor	Simple	5	Simple	1	6
8.5	Allocate macronutrients distribution	Simple	5	Simple	1	6
8.6	Select nutritional standard plan	Simple	5	Simple	1	6
8.7	Allocate nutrients"trade-off" (swapping process)	Average	10	Average	2	12
8.8	Create a daily meal plan	Complex	15	Complex	3	18
9.1	Everytime a new set of body measurements or a new meal plan are created, those have to be logged for future tracking.	Average	10	Simple	1	11
9.2	Allow the nutritionist to consult what was the previous state for a	Average	10	Simple	1	11

Caso de uso	Descripción	Tipo caso de uso	UUCP	Tipo actor	AW	UCP
	particular meal plan displayed at the Patient history trace					
9.3	Allow to create a new meal plan by editing a previous meal plan done by its own.	Simple	5	Simple	1	6
9.4	Allow the nutritionist to consult what was the previous state for a particular set of body measurements displayed at the Patient history trace	Simple	5	Average	2	7
		Total	240	Total	49	289

## 9.5. Conteo Puntos de función Cosmic

En la Tabla 32, se muestra el conteo de puntos de función Cosmic para los requerimientos de la aplicación analizada.

Tabla 32. Conteo de puntos de función COSMIC.

Modulo	Requerimiento	Descripción	Usuario Funcional	Subproceso	Datos	Objeto	Tipo de movimiento	CFP
Sign in for new user	1.1	User create a new user in Nutri Fi by using its email and some personal information	User	Enter credentials	Credentials	User	E	1
			*	Validate user	Credentials	Nutritionist	R	1
			*	Validate password	Credentials	User	R	1
			*	Request a verification code	Credentials	User	E	1
				Send verification code to user	Credentials	User	X	1
			User	Enter verification code	Verification code	User	E	1
			*	Validate verification code	Verification code	User	R	1

Modulo	Requerimiento	Descripción	Usuario Funcional	Subproceso	Datos	Objeto	Tipo de movimiento	C F P
			User	Enter personal information	Profile information	Nutritionist	E	1
			User	Enter personal information	Profile information	Paciente	E	1
			User	Enter personal information	Profile information	Clinic	E	1
			User	Enter personal information	Profile information	Education	E	1
			User	Create user	User information	Person	W	1
			*	Create profile	Personal information	Nutritionist	W	1
	1.2	Terms and conditions are mandatory to be part of Nutri Fi	*	Get T/C	T/C	User	R	1
			*	Get data policy	T/C	User	R	1
			User	Display T/C	T/C	User	X	1
			User	Display data policy	T/C	User	X	1
			User	Accept T/C	User information	Nutritionist	E	1
			User	Accept Data policy	User information	Nutritionist	E	1
			*	Activate user	User information	Paciente	W	1
Login for existing user	2.1	Allows the user to get in the application by entering its credentials, whenever the user	User	Enter credentials	Credentials	User	E	1
			*	Validate user	Credentials	Nutritionist	R	1
				Validate password	Credentials	Person	R	1
			User	Forgot pwd request	Credentials	User	E	1
			*	log event track	Credentials	event log	W	1
			User	Display results	Credentials	User	X	1
			User	Access app	Credentials	Paciente	X	1

Modulo	Requerimiento	Descripción	Usuario Funcional	Subproceso	Datos	Objeto	Tipo de movimiento	C F P
		session expired.						
	2.2	Allow the user to recover its password.	User	Request pwd	Credentials	User	E	1
			*	Get temp pwd	Credentials	Person	R	1
			*	Send temp pwd	Credentials	Nutritionist	X	1
			User	Enter temp pwd	Credentials	User	E	1
			*	Validate user	Credentials	Nutritionist	R	1
			*	Validate password	Credentials	Person	R	1
			*	Track recover password event	Credentials	event log	W	1
			*	Reactivate user	User information	Paciente	W	1
	2.3	User session must remains active, hence the user can access to the application as fast as it select the app in its mobile .	*	expiration alert	Credentials	User	E	1
			*	Renew user token	Credentials	User	W	1
			User	Keep user in app location	Credentials	User	X	1
			User	Access app in background	Credentials	User	E	1
			*	Renew measurement service	Body measurement	Body measurement	W	1
			*	Renew nutrition plan	Meal plan	Meal plan	W	1
			User	Display refresh message	Credentials	User	X	1
Connect nutritionist with a	3.1	Allow the nutritionist to add a	User	Search patient	person data	Person	E	1
			*	Get patient	person data	Person	R	1

Modulo	Requerimiento	Descripción	Usuario Funcional	Subproceso	Datos	Objeto	Tipo de movimiento	C F P
patient		new patient, by enable the option to search it by its emails address	*	Display results	person data	Person	X	1
			User	Send invitation	User information	User	E	1
			*	Request approval	User information	User	X	1
			*	Add patient to user	User information	User	W	1
			*	Create connection	User information	User	W	1
	3.2	Allow the nutritionist to realize this particular patient have been added into its contact list previously.	User	Search patient	person data	Person	E	1
			*	Get patient	person data	Person	R	1
				Get appointment details	person data	Person	R	1
			*	Display results	person data	Person	X	1
			*	Add patient to user	User information	User	W	1
			*	Display patient	User information	User	X	1
	3.3	Once the Invitation have been accepted. The nutritionist is able to access its patient	User	Search patient	person data	Person	E	1
			*	Get patient	person data	Person	R	1
			*	Display results	person data	Person	X	1
			User	Access patient profile	User information	User	R	1
*			Store event tracking	User information	User	W	1	

Modulo	Requerimiento	Descripción	Usuario Funcional	Subproceso	Datos	Objeto	Tipo de movimiento	C F P
		profile information.						
	3.4	Once the nutritionist-patient relationship have been in place, the nutritionist is able to cancel this relationship.	User	Search patient	person data	Person	E	1
			*	Get patient	person data	Person	R	1
			*	Display results	person data	Person	X	1
			User	cancel connection	patient data	Paciente	E	1
			*	Get recent connection status	person data	Person	R	1
			*	delete relationship	User information	User	W	1
			*	delete user history data	User information	Nutritionist	W	1
			*	delete measurements	Body measurement	Body measurement	W	1
			*	delete meal plan	Meal plan	Meal plan	W	1
			User	Display profile status	person data	Person	X	1
Crear a temporal user	4.1	Allow the user to move forward with the body measurements and meal plan creation, regardless if the patient is already a	User	Search patient	person data	Person	E	1
			*	Get patient	person data	Person	R	1
			User	Display patient status	person data	Person	X	1
			User	Enter temp user	person data	Paciente	E	1
			*	Create temp user	User information	User	W	1
			User	Display temp user	User information	User	X	1
			User	add measurement	Body measurement	Body measurement	E	1
			*	store measurement	Body measurement	Paciente	W	1
			*	Initiate	User	User	E	1

Modulo	Requerimiento	Descripción	Usuario Funcional	Subproceso	Datos	Objeto	Tipo de movimiento	C F P
		Nutri Fi user.		appointment	informati on	informati on		
	4.2	A tempo ral user will remai n active as long as the Patien t haven' t authe nticat e it, and theref ore create d an actual Nutri Fi user.	*	get temporal user	User informati on	User	R	1
	User		update user status	User informati on	User	E	1	
	*		send new status update	User informati on	User	X	1	
	*		Create actual user	User informati on	User	W	1	
	4.3	If the Patien t haven' t confir m the tempo ral user, and theref ore an actual Nutri Fi user using this email haven' t been	*	BD trigger alert	User informati on	User	E	1
			*	Get temp users	User informati on	User	R	1
			*	Send notification	User informati on	User	X	1

Modulo	Requerimiento	Descripción	Usuario Funcional	Subproceso	Datos	Objeto	Tipo de movimiento	C F P
		created. After each 3 days, a push notification will be delivered to the Nutritionist letting the user know that this Patient hasn't accessed its information yet.						
Patient profile information	5.1	Allow the nutritionist to see all the patient profile information, as long as they remain in a nutritionist-	User	Search patient	patient data	Paciente	E	1
			*	Validate connection	User information	User	R	1
			*	log event track	User information	event log	W	1
			*	Display results	patient data	Paciente	X	1
			User	Search patient	patient data	Paciente	E	1
			*	log event track	User information	event log	W	1

Modulo	Requerimiento	Descripción	Usuario Funcional	Subproceso	Datos	Objeto	Tipo de movimiento	C F P
		patient relationship.						
	5.2	Allows the nutritionist to interact with a particular patient profile, meaning to create body measurements, create a meal plan, schedule an appointment, etc.	*	Validate connection	User information	User	R	1
			User	search measurements	Body measurement	Person	E	1
			*	Display results	patient data	Paciente	X	1
			*	get meal plan	Meal plan	Person	R	1
			*	Display results	patient data	Paciente	X	1
Nutritionist profile information	6.1	Allow nutritionist to edit its profile information.	*	Get profile information	User information	Nutritionist	R	1
			*	enter updated values	User information	Nutritionist	E	1
			User	store updated values	User information	Person	W	1
			*	update profile in cognito	User information	Person	W	1
	6.2	Allow the nutritionist to add a	User	enter updated values	User information	Nutritionist	E	1
			*	store updated values	User information	Clinic	W	1

Modulo	Requerimiento	Descripción	Usuario Funcional	Subproceso	Datos	Objeto	Tipo de movimiento	C F P
		new office location			on			
	6.3	Allow the nutritionist to edit an existing office location	User	enter updated values	User information	Nutritionist	E	1
			*	store updated values	User information	Clinic	W	1
	6.4	Allow the nutritionist to remove an existing office location	User	enter updated values	User information	Nutritionist	E	1
			*	store updated values	User information	Clinic	W	1
Create set of new body measurements	7.1	Possible body measurement must be shown to the user, so it can be entitled to add a new value related for the body measurement	*	Get measurements	Measurement	Measurement	R	1
			*	Filter measurement to user	Measurement	user	R	1
			User	add new value	Measurement	Paciente	E	1
			*	Store new measurement	Body measurement	Body measurement	W	1
			User	Display measurement to user	Measurement	Paciente	X	1
			User	update measurement list	Measurement	Paciente	E	1
			*	Store new measurement in patient	Body measurement	Body measurement	W	1
			User	Display measurement to user	Measurement	Paciente	X	1

Modulo	Requerimiento	Descripción	Usuario Funcional	Subproceso	Datos	Objeto	Tipo de movimiento	C F P
		remem t select ed.	*	Track measurement change event	Measure ment	event log	W	1
	7.2	Allow the user to enter a new body measu remen t value.	User	add new value	Measure ment	Paciente	E	1
			*	Store new measurement	Body measure ment	Body measure ment	W	1
			User	Display available measurements	Body measure ment	Body measure ment	X	1
			User	Enter measurement value	Measure ment	Paciente	E	1
			*	Store updated measurement	Body measure ment	Body measure ment	W	1
			User	Display results	Body measure ment	Body measure ment	X	1
Creat e a new meal plan	8.1	Creat e plan name	User	Request plan to patient	patient data	Paciente	E	1
			*	Create plan to patient	patient data	Paciente	W	1
			User	enter plan name	Meal plan	Meal plan	E	1
				store plan name	Meal plan	Meal plan	E	1
			User	display name	Meal plan	Meal plan	X	2
	8.2	Adjust plan factor	User	enter factors	Meal plan	Meal plan	E	1
			*	store factors	Meal plan	Meal plan	W	1
			*	display factors validated	Meal plan	Meal plan	X	
	8.3	Calcul ate the basal metab olic rate and energ y requir ement	User	enter rates	Meal plan	Meal plan	E	1
			*	store rates	Meal plan	Meal plan	W	1
			*	display rates	Meal plan	Meal plan	X	1

Modulo	Requerimiento	Descripción	Usuario Funcional	Subproceso	Datos	Objeto	Tipo de movimiento	C F P
		rate						
	8.4	Adjust the protein requirement factor	*	get suggested protein factor	Meal plan	Meal plan	R	1
			*	display factor suggested	Meal plan	Meal plan	X	1
			User	enter protein factor	Meal plan	Meal plan	E	1
	8.5	Allocate macro nutrients distribution	User	enter distribution	Meal	Meal	E	1
			*	Display results	Meal	Meal	X	1
	8.6	Select nutritional standard plan	*	Get supported standards	Ingredients	Ingredients	R	1
			User	Select standard	Meal plan	Meal plan	E	1
			*	Display standards	Meal plan	Meal plan	X	1
	8.7	Allocate nutrients "trade-off" (swapping processes)	User	Allocate nutrient	Food group	Food group	E	1
			*	Validate allergies	Allergy	Allergy	R	1
			*	Validate prescribed values	Ingredients	Ingredients	R	1
			User	display partial result	Food group	Food group	X	1
			*	Store nutritient	Food group	Food group	W	1
			*	Calculate calories	Food group	Food group	R	1
			User	display final result	Food group	Food group	X	1
	8.8	Create a daily meal plan	*	Get allocated nutrients	Food group	Food group	R	1
			*	Create default meal times	meal day	meal day	W	1
			User	Allocate nutrient in meal plan	Meal plan	Meal plan	E	1
			*	Validate nutritient	Ingredients	Ingredients	R	1
				Validate allergies	Allergy	Allergy	R	1
			User	Allocate meal	Day	day	E	1

Modulo	Requerimiento	Descripción	Usuario Funcional	Subproceso	Datos	Objeto	Tipo de movimiento	C F P
				day				
			*	Store meal plan	Meal plan	Meal plan	W	1
Patient history trace	9.1	Everytime a new set of body measurements or a new meal plan are created, those have to be logged for future tracking.	*	Trigger event	Body measurement	event log	E	1
			*	Store measurement event	Body measurement	event log	W	1
			User	display event in historical events	Body measurement	event log	X	1
			*	Store meal plan	Meal plan	event log	W	1
			User	display event in historical events	Meal plan	event log	X	1
	9.2	Allow the nutritionist to consult what was the previous state for a particular meal plan displayed at the Patient history trace	User	search meal plan	User information	Person	E	1
			*	get user history	User information	Person	R	1
			*	get meal plan	Meal plan	Meal plan	R	1
			*	Store meal plan event	Meal plan	event log	W	1
			User	Display results	Meal plan	Meal plan	X	1
			*	Notify user revision	User information	Person	X	1

Modulo	Requerimiento	Descripción	Usuario Funcional	Subproceso	Datos	Objeto	Tipo de movimiento	C F P
	9.3	Allow to create a new meal plan by editing a previous meal plan done by its own.	*	get meal plan	Meal plan	Meal plan	R	1
			User	update meal plan	User information	Paciente	E	1
			*	store updated plan	Meal plan	Meal plan	W	1
			*	Display results	Meal plan	Meal plan	X	1
	9.4	Allow the nutritionist to consult what was the previous state for a particular set of body measurements displayed at the Patient history trace	User	search measurements history	Body measurement	Body measurement	E	1
			*	Get measurements	Body measurement	Body measurement	R	1
			*	Display results	patient data	Paciente	X	1
			*	Display historical chart	patient data	Paciente	X	1
							Total	174

## 9.6. Conteo con puntos de historia de usuario

En la Tabla 33, se muestra el consolidado de puntos de historia de usuario (USP) con su correspondiente tiempo en horas reportado (HP).

Tabla 33. Conteo con puntos de historia de usuario.

EPICA	Descripción de EPIC	Historia de usuario	USP	HP
1	As a Nutritionist I want to be created as a user so that I can leverage all the app features	Design user login screens	3	11,00
		Create login screens to request user information	2	8,00
		Store user information in database	1	15,00
		Create cognito user authentication process	5	49,25
		Integrate authentication process with screen	1	6,00
		Database model creation	3	29,50
		Validate user information	3	16,00
		Validate user information	2	8,00
		Populate parametric list using single service	3	61,50
		Populate parametric list using single service	5	56,00
		User token auth validation	2	45,00
		User token auth validation	3	35,00
		Retrieve user data from database	2	52,00
		Lifecycle event management	3	42,50
		Reuse UI models for Login process	3	20,00
		Manipulate user data as a reusable dataset	3	50,20
		Create architectural model for project	8	124,00
2	As a Nutritionist I want to log into the application so that I can use the features in my daily duties	Allow the user to login using its credentials	5	53,00
		Authenticate the user against the cognito service	3	20,00
		Create password recovery screens	5	18,50
		Generate temporal password and update new password	5	37,25
		Forgot password screen	2	9,75
		Forgot password validation	2	9,00
		Forgot password validation	2	13,00
		Enhance poor input validation	8	131,26

EPICA	Descripción de EPIC	Historia de usuario	USP	HP
		Lifecycle event management	3	17,00
		Manipulate new user session	2	15,50
		Touch ID login screen	5	55,00
		Touch ID configuration	2	12,00
		Touch ID user validation	3	10,00
		Touch ID entitlement	8	45,00
		Renew user session token	8	32,50
		Implement app lifecycle event to keep user session active while app is in background	3	12,00
3	As a nutritionist I want to connect with my patient so that I can access his/her information and start its medical consultation	Design user connection screens	2	14,00
		Add a new patient to the Contact List	3	26,00
		Add a new patient to the Contact List	5	42,00
		Allow the nutritionist to realize this particular patient have been added into its contact list previously.	2	9,00
		Allow the nutritionist to realize this particular patient have been added into its contact list previously.	1	5,00

EPICA	Descripción de EPIC	Historia de usuario	USP	HP
		Once the Invitation have been accepted. The nutritionist is able to access its patient profile information.	3	35,00
		Once the Invitation have been accepted. The nutritionist is able to access its patient profile information.	2	10,00
		Once the nutritionist-patient relationship have been in place, the nutritionist is able to cancel this relationship.	3	23,00
		Update user relationship status	3	37,00
		Update user relationship status	3	20,75
		Disconnect user relationship	8	75,25
		Disconnect user relationship	5	114,25
		Cancel user relationship	8	102,50
		Cancel user relationship	2	14,00
		Request a user invitation	5	56,00
		Request a user invitation	2	15,00
		User invitation accepted notification	5	63,75
		User invitation accepted notification	3	8,00

EPICA	Descripción de EPIC	Historia de usuario	USP	HP
		Once the nutritionist-patient relationship have been in place, the nutritionist is able to cancel this relationship.	1	10,75
4	As a nutritionist, I want to create a temporal user so that I can attend my patients regardless if they are previously enrolled in Nutri Fi	Temporal user screen design	2	17,00
		Allow the user to move forward with the body measurements and meal plan creation, regardless if the patient is already a Nutri Fi user.	3	10,00
		Allow the user to move forward with the body measurements and meal plan creation, regardless if the patient is already a Nutri Fi user.	8	108,70
		Temporal user expiration	5	48,00
		Create body measurement to temporal user	5	44,50
		Create body measurement to temporal user	1	8,00
		Update body measurement to temporal user	2	4,50
Update body measurement to	1	22,75		

EPICA	Descripción de EPIC	Historia de usuario	USP	HP
		temporal user		
		Delete body measurement to temporal user	1	2,00
		Delete body measurement to temporal user	2	7,25
		Create meal plan to temporal user	1	2,50
		Create meal plan to temporal user	3	27,75
		Renew temporal user expiration date	2	14,00
		Temporal user notifications	1	3,25
		Temporal user notifications	3	4,00
5	As nutritionist I want to see my patient profile information so I can check any relevant facts that might affect in my prescription.	Patient profile screen design	2	8,00
		Display patient profile information	5	73,50
		Create user nutrition goals	2	14,00
		Create user nutrition goals	5	49,00
6	As a nutritionist I want to create and edit my professional profile information so I that I could offer my services better to my potential customers	Update user nutrition goals	3	14,00
		Update user nutrition goals	2	18,00
		Delete user nutrition goals	2	13,75
		Delete user nutrition goals	3	4,00
		Display summary result chart	1	6,00
		Populate result chart service	1	12,00
7	As a nutritionist I want to create new body measurements to my patient so that I can trace its progress against the meal plan prescribed	Design body measurement screens	1	4,00
		Maintanance body measurement information	8	41,00
		Display all supported body measurements to add a new value.	1	5,00
		Validate required body measurements are entered	3	20,75
		Body measurement configuration set	2	17,75
		Populate supported body measurement	3	35,00
		Update a body measurement	3	30,50
		Update a body measurement	3	25,00
		Delete a body measurement	2	11,00
		Delete a body measurement	8	57,25
		Parametric list population for body measurement	3	27,00
		Goal compare for body measurement	3	24,00
		Goal compare for body measurement	5	14,00
		Add a new value for each body measurement	1	7,00

EPICA	Descripción de EPIC	Historia de usuario	USP	HP
8	As a nutritionist I want to create a new meal plan to my patient so that he/she can follow the meal plan prescribed everyday.	Meal plan screen design	3	47,00
		Create plan name	3	13,00
		Adjust plan factor	2	18,50
		Calculate the basal metabolic rate and energy requirement rate	3	22,00
		Adjust the protein requirement factor	3	24,50
		Allocate macronutrients distribution	1	10,00
		Select nutritional standard plan	1	15,00
		Allocate nutrients"trade-off" (swapping process)	2	17,50
		Adjust each parameter for calculation	3	25,50
		Assign proper icon asset for each nutrient	5	28,00
		Manipulate dataset for chart display	5	31,00
		Add/Update/Delete nutrient allocation	3	37,50
		Backward navigation in flow issue	2	3,00
		Optimize meal plan creation time response	5	30,00
		Validate each meal plan parameter is entered	3	23,00
		Calories progress bar implementation	2	43,00
		Calories progress bar implementation	3	15,00
		Meal day screen optimization	3	22,00
		Meal day screen optimization	5	25,00
9	As nutritionist I want to check my patient history trace in terms of measurements and meal plan so that I can understand what should be adjusted to achieve its goals.	Historical data screen design	1	19,00
		Retrieve list of historical user events	3	21,00
		Display historical user events	2	24,00
		Get further details for a single event	5	47,75
		Display drill down screen for a single event	3	26,00
		Access the previous state for a meal plan prescribed	3	11,00
		Access the previous state for a meal plan prescribed	1	1,00
		Access the previous state for a particular set of body measurements	3	13,00
		Access the previous state for a particular set of body	2	17,50

EPICA	Descripción de EPIC	Historia de usuario	USP	HP
		measurements		
			376	3287,41

## 9.7. Artículo

A continuación se detalla el contenido del artículo aceptado para publicación en la conferencia CibSE 2020, XXIII Ibero-American Conference on Software Engineering, que se realizará en noviembre del 2020.

- Ugalde, F., Quesada-López, C., Martínez, A., Jenkins, M. (2020) *A comparative study on measuring software functional size to support effort estimation in agile*. Proceeding of the XXIII Ibero-American Conference on Software Engineering (CibSE 2020)

# A comparative study on measuring software functional size to support effort estimation in agile

Fabián Ugalde, Christian Quesada-López, Alexandra Martínez, and Marcelo Jenkins

University of Costa Rica, San Pedro, Costa Rica  
{fabian.ugalde, cristian.quesadalopez, alexandra.martinez,  
marcelo.jenkins}@ucr.ac.cr

**Abstract.** Software effort estimation models based on functional size allow software organizations to plan their development projects. A large number of organizations have adopted agile processes, but there is little evidence on the adoption of functional sizing methods to support software effort estimation in agile contexts. In this study, we compare four functional size estimation methods as the basis for effort estimation in the context of a startup company that develops mobile applications using an agile methodology. Measurements of software size, expressed in User Story Points (USP), Use Case Points (UCP), IFPUG Function Points (UFP), and COSMIC Function Points (CFP), were taken for a set of requirements from one project in the company. Effort estimation models were then derived from these measurements, using regression, and their accuracy was determined by the Mean Magnitude of Relative Error (MMRE) and Mean Balanced Relative Error (MBRE). We obtained the following MMRE results for each functional sizing method: 0,86 for UCP, 0,36 for USP, 0,36 for UFP and 0,22 for CFP and following MBRE results: 0,98 for UCP, 0,45 for USP, 0,53 for UFP and 0,35 for CFP. The effort estimation model based on COSMIC function points turned out to be the most accurate in the context of the software organization under study. Additionally, convertibility models between sizing measurements were generated to allow the organization to convert its historical measurements into any other software size measure, without having to perform the counting process of the target method.

**Keywords:** Functional size, software effort estimation, function points, IFPUG FPA, COSMIC FFP, empirical study.

## 1. Introduction

The use of effort estimation models can help software companies plan, monitor, and control their efforts in terms of costs and development processes [8]. Therefore, having realistic estimates at an early stage of the project life cycle, allows managers to control resources effectively [10]. In agile projects, effort estimations based on functional size are being incorporated to provide more accurate estimates [18]. However, more empirical evidence is needed on the use of functional size measurements to support software effort estimation in agile contexts. According to Lavazza et al [8], the accuracy of effort estimation must be carefully evaluated before its use in a real life environment.

Hence, our study aims to provide empirical evidence on the accuracy of effort estimation based on functional size measurements in an agile context. It also provides enough details of the methods used –as advised by Kitchenham et al. [6] – so that other researchers can replicate the study in different contexts, thus widening the body of evidence.

In this study, we investigated whether the choice of functional size measurements has an effect on the accuracy of effort estimation models in the context of an agile organization. This organization is a startup company that develops mobile applications aimed at promoting healthy lifestyles, through the application of the social network approach to nutritional concepts.

The purpose of our case study is to compare the accuracy of effort estimation models built upon four different functional size measurements: Use Case Points (UCP), COSMIC Function Points (CFP), IFPUG Function Points (UFP), and User Story Points (USP). The four functional size estimation methods were applied to the same set of requirements from a mobile application project within the organization. Regression methods were then used to derive effort estimation models from the four size measurements. The Mean Magnitude of Relative Error (MMRE) and Mean Balanced Relative Error (MBRE) were used to determine

which model was more accurate for predicting effort. Criteria set out by [6, 8] were considered to mitigate some threats to validity. To guide this study, two research questions were defined:

**RQ1:** Which functional size-based effort estimation model yields the most accurate estimates in the context of the organization under study?

**RQ2:** Can convertibility models of functional size measurements yield accurate results in the context of the organization under study?

The rest of the paper is organized as follows. Section 2 explores the functional size estimation methods used in our study: Use Case Points, User Story Points, COSMIC Function Points and IFPUG Function Points. Section 3 presents the related work. Section 4 describes the case study performed in the startup company. Section 5 shows the results obtained from this case study. In Section 6 we discuss and summarize the results. And section 7 presents our conclusions and future work.

## 2. Background

Here we describe each of the functional size estimation methods used in this study: Use Case Points, User Story Points, IFPUG Function Points and COSMIC Function Points.

### 2.1. Use Case Points

A *use case* details a scenario of how the software system should interact with the user –or other systems– to carry out a specific action or activity [23]. Software estimation by *use case points* is a technique to calculate and predict the size of a system for software development projects [24]. To calculate the total points, it is necessary to count the total transactions in each use case.

Use cases and its actor are categorized into simple, average and complex. For each category, a weighted point is assigned. The unadjusted use case points are the sum of all weighted points assigned.

### 2.2. User Story Points

*User stories* are the way in which requirements are captured in agile software development methodology. A *user story* is independent, negotiable, valuable, estimated, small and verifiable [25]. One method to estimate the size of a user story is through *planning poker*, which uses a Fibonacci scale. In planning poker, each team member assigns user story points for a requirement, based on its expert criteria. Then the team discusses until a single size value is agreed on, and that will be the *user story points* assigned to that requirement [26].

### 2.3. IFPUG Function Points

*Function Point Analysis* (FPA) is a proven and accepted methodology to determine the size of a software development project [27]. FPA measures software size in terms of the following attributes [27]:

- Data entering a system: external inputs (EI) as logical transaction inputs.
- Data leaving the system: external outputs (EO) or external queries (EQ) such as online screens, reports, or inputs to other systems.
- Data created and stored within the system: internal logical files (ILF) such as user-defined logical groups of data.
- Data maintained within a different system but necessary to satisfy a certain process requirement: external interfaces (EIF) as interfaces to other systems.

There are set of counting rules that assign *IFPUG Function Points* (UFP) to each EI/EO/ILF/EIF, based on the number of Data Element Types (DET) and File Type Referenced (FTR) associated with the transactional function. Such rules can be found in the work of Gandomani et al. [28].

## 2.4. COSMIC Function Points

System functions are modeled through functional processes, which can be divided into sub-processes [29]. Sub-processes are classified in two types: data movement and data manipulation. The data movement sub-process is the main goal for the calculation of the function point, which is categorized into four categories:

- Input (I): the movement of a user's data to a functional process.
- Output (O): the movement of data from a functional process to a user.
- Read (R): the movement of data from persistent storage to a function process. Persistent storage must be part of the system.
- Write (W): the movement of data from a functional process to persistent storage. Persistent storage must be part of the system.

In a functional process, the data movement quantity for the four categories mentioned above are presented as *COSMIC function points* (CFP), in a unit called CFSU (COSMIC functional size unit).

## 3. Related work

We performed a literature review and found several studies on comparing software effort estimation models, but only a few studies on convertibility of size measurements. We will first present the studies related to estimation model comparison [1, 3, 5, 9, 10, 11, 13, 15, 17, 19, 20, 22], and then the studies about convertibility between functional size measurements [7, 18].

The work by Salmanoglu et al. [17] performed three case studies that compared the effectiveness of effort estimation based on two functional size methods (CFP and USP) in an agile context. They used regression analysis to generate statistical models from software size measurements and evaluated their performance using MMRE. They concluded that CFP produced the most accurate effort estimation model.

Similarly, Ungan et al. [20] compared the effectiveness of two approaches for effort estimation in an organization that uses SCRUM. They compared USP-based effort estimation (with Planning Poker) to CFP-based effort estimation. They used regression models and artificial neural networks to develop the estimation models. They showed that COSMIC measurements are a better basis for effort estimation than User Story Points, in their context.

Paz et al. [13] applied Incremental COSMIC Function Points in an agile environment, and compared its accuracy against the Rational Unified Process (RUP) method, concluding that the COSMIC-based model was better. They introduced the concept of *Incremental COSMIC Function Points* as an approach for adapting the CFP measure to agile environments. They proposed that the sum of all Incremental COSMIC function points for each requirement or work size unit (user story, from SCRUM standpoint), will determine the total COSMIC function points for the whole system.

Desharnais et al. [3] proposed an approach to move from COCOMO (Constructive Cost Model) to User Stories using COSMIC as the size measurement method. They showed a model that allows estimating the effort for each user story based on CFP. Sholiq et al. [19] compared UCP against UFP in an effort estimation model, concluding that UCP was slightly better. Brian et al. [1] propose an effort estimation model using external databases based on UFP and CFP, reaching the conclusion that there was no significant difference between these two software size estimation methods.

Mendes et al. [10] studied three cost estimation models applied to web hypermedia applications, using regression models to support their estimations. They compared linear and stepwise regressions, and case-based reasoning (CBR), concluding that CBR technique gave the most accurate results.

Wilkie et al. [22] explored the utility of function point software sizing techniques when applied at two levels of software requirements documentation, in a commercial software development organization. They appraised the value (cost/benefit) that functional sizing techniques can bring to the planning and

management of software projects, concluding that “Estimated NESMA” is the most appropriate tool to use for size estimation in the context of the studied company.

Phannachitta [15] did a comparison of similarity measures in analogy-based software effort estimation, using a robust approach involving MMRE and some other comparison indicators. Mittas et al. [11] propose a graphical method to evaluate the performance of a cost estimation model by using an analysis of regression error characteristic. Later, Mittas et al. [12] created an algorithm to cluster and rank software cost estimation models through multiple comparisons.

Other studies [5, 9] have compared different effort estimation methods to expert-based estimates. Jørgensen and Boehm [5] debated over which effort estimation method produces better accuracy, comparing formal models against expert judgment. Elaborating on evidence that backup both approaches, they agreed there is no single *best* method, but different perspectives that benefit particular scenarios. Nonetheless, they highlight that more advanced estimation models will likely lead to significantly more accurate effort estimates. Similarly, the study by Lenarduzzi et al. [9] compared the effectiveness of an effort estimation model based on IFPUG Function Points to an expert-based estimation, in an agile environment. They showed that the accuracy of the effort estimate provided by the SCRUM team was better than the obtained through functional size measurements.

Finally, two studies [7, 18] addressed the issue of convertibility between functional size measurements. Lavazza et al. [7] presented a correlation study between IFPUG and COSMIC function points. They provide a set of models to convert counts from one software size method to the other. Santana et al. [18] examined whether function points are compatible with user story points on agile projects, and found a correlation between them. They stated that their results should respect the units of measurement and the reality of each organization.

All previously mentioned studies were performed in mid to large companies, and focused on comparing effort estimation models based on just *two* functional size measurements. Furthermore, their competing techniques consider only completed projects with a defined set of requirements and a clear scope. In contrast, our study considers effort estimation models based on *four* functional size measurements, and is performed in an agile startup company. This context poses some challenges such as unclear project scope and unstable requirements.

#### 4. Our Case Study

We conducted a case study in a software startup company mainly dedicated to the development of mobile applications. It has four developers, who use the SCRUM methodology. The organization has been using an agile metric collection tool, hence both estimated and real effort data were available for every user requirement (use case). We obtained most of the project data from this repository. Although the requirements were developed using SCRUM, there were formal specification documents that we used to count COSMIC, IFPUG Function Points, and Use Case Points.

##### 4.1. Goal and object selection

The goal of our case study was to accurately estimate the effort by generating models based on software functional size measurements. We scoped our work to four functional size estimation methods: Use Case Points, User Story Points, IFPUG Function Points, and COSMIC Function Points. A secondary and related goal of the study was to seek accurate convertibility models between functional size measurements. Both goals are applicable to the case study only, therefore its results are limited to this company studied.

Since the organization where the study was performed used SCRUM, we selected nine epics from one project, for this case study. The epics were broken down into 32 requirements and use cases, which were in turn decomposed into 119 user stories.

##### 4.2. Data collection procedure

We created a spreadsheet to record the functional size measurements as well as the actual effort time. To collect the functional size measurements expressed in UCP, UFP, and CFP, one of the researchers (previously trained on these size estimation methods) performed the counting corresponding to each sizing

method. On the other hand, for each user story, we recorded both its actual effort, and its estimated effort obtained with planning poker.

We identified outliers by calculating the productivity index (**PI**) of each requirement under each sizing method as **PI= actual effort/size**, and then generating box-plots with this information. Outliers indicated in boxplots were then discarded.

### 4.3. Analysis procedure

The analysis procedure for generating the effort estimation models was based on a regression analysis. To select the regression analysis approach to follow (linear, multiple, polynomial, exponential), the dataset was analyzed with a scatterplot. Depending on the pattern revealed by the scatterplot, different regression approaches were chosen. By default, linear regression was used. In all models, the dependent variable was the estimated effort time (EFF) and the independent variables were a subset of the software functional size measurements (i.e., UCP, USP, UFP or CFP).

On the other hand, the procedure for calculating the MMRE was to apply Equation 1 to each generated model. Case study wise, the lowest MMRE would determine the best effort estimation model. Nevertheless, a model that exhibits an MMRE value less than 0,25 is considered accurate enough.

$$MMRE = \frac{100}{T} \times \sum_{i=0}^T \frac{|estimated\ eff(i) - actual\ eff(i)|}{actual\ effort(i)} \quad (1)$$

In addition, the MBRE calculation procedure was first to apply Equation 2 to get the Relative Error ( $RY_i$ ).

$$RY_i = \frac{estimated\ eff(i) - actual\ eff(i)}{actual\ effort(i)} \quad (2)$$

Then calculate the Balance Relative Error ( $R_i$ ) applying Equation 3.

$$R_i = \begin{cases} \frac{estimated\ eff(i) - actual\ eff(i)}{actual\ effort(i)}, & estimated\ eff(i) - actual\ eff(i) \geq 0 \\ \frac{estimated\ eff(i) - actual\ eff(i)}{estimated\ eff(i)}, & estimated\ eff(i) - actual\ eff(i) < 0 \end{cases} \quad (3)$$

Equation 4 is applied to get the MBRE value.

$$MBRE = \frac{1}{N} \times \sum_{i=0}^N R_i \quad (4)$$

The evaluation criteria for MBRE metric consider a model accurate enough when  $MBRE < 0,35$ , as Miyazaki [30] explained, 35% accuracy might be enough in the early phase of the software life cycle.

### 4.4. Threats to validity

We evaluated the validity of our study based on Runeson's guidelines [16], considering construct validity, internal validity, external validity, and reliability, as shown below. The threats to validity are briefly mentioned next.

*Construct validity:* in our case study, we created a set of effort estimation models based on functional size measurement for a specific organizational context, and evaluated their accuracy performance using MMRE and MBRE.

*Internal validity:* one validity threat is the small sample size used (having only 32 requirements to analyze). Additionally, a bias could have been introduced in the sample selection process. Also, some data were discarded from the analysis as they were identified as outliers. Another threat is that only one researcher performed the functional size measurement for each requirement, however this researcher have proven experience with these counting rules methods. Finally, MMRE as a statistical measure to evaluate a model's performance has sometimes failed to show statistical evidence, as advised by Kitchenham et al. [6] and Lavazza et al. [8], we attempt to mitigate this threat by comparing also accuracy in combination of MBRE, that has demonstrate more efficiency as Jørgensen recommended [31]

*External validity:* the effort estimation models are applicable only to the organization described in this study. These models are not intended to be used in other contexts besides this one.

*Reliability:* since the organization's dataset grew with time, there is a validity threat that this case study presents non-repeatable results, as pointed out by Lavazza et al. [8]. As the organization scales up, results might vary.

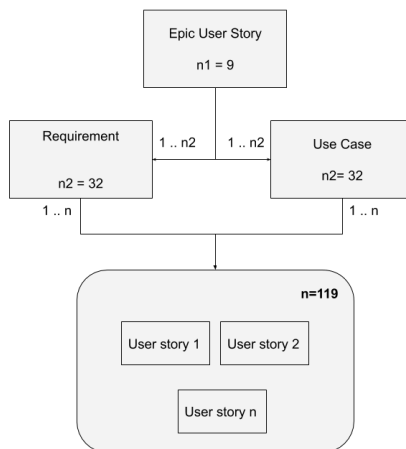
## 5. Analysis of Results

In this section we present the results of our case study, addressing each of the research questions.

### 5.1. Accuracy of effort estimation models based on functional size (RQ1)

Our first research question aimed at finding the most accurate estimation model based on functional size measurements, in the context of the organization, and using MMRE and MBRE as the accuracy performance measurements.

Figure 1 describes the relationship in between: requirements, use cases and user stories, for this case study we gather 9 Epic User Stories, those Epics were broken down into 32 Requierements and same amount of Use Case. Then during the development phase, those Requirements and Use Cases were structured as 119 User Stories, these User Stories were delivered by the team, having the actual time tracked against each User Story. Requierements assets were used for UFP and CFP functional size measurements, Use Cases were used for UCP and User Story items for USP functional size measurements.



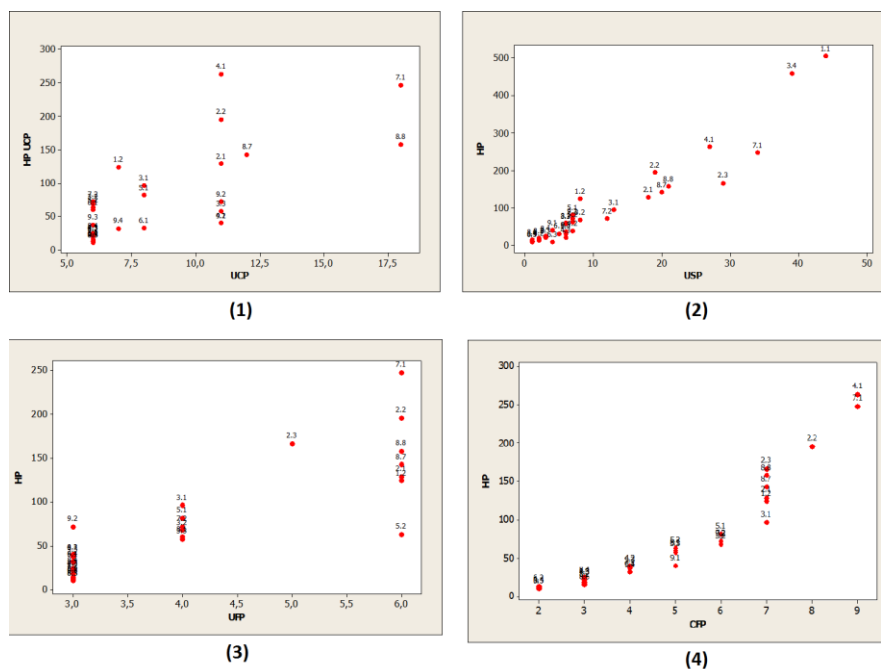
**Fig. 1.** Requirements detail structure

Using regression, we developed a series of parametric statistical models to describe the relationship between size and effort. We applied different kinds of regression analyses to build the models, the model types were selected by using a visul pattern scan in the scatterplot for each functional measurement. Simple, multiple, polynomial, and exponential regression methods were applied to UCP, UFP, USP, CFP counts to render several effort estimation models. We generated scatterplots for each functional size method to decide

on the type of regression to apply, based on the dataset pattern. Figure 1 shows scatterplots of actual effort vs. size for each sizing measurement: (1) UCP, (2) UFP, (3) USP, and (4) CFP. The Y-axis represents actual effort, while the X-axis represents the functional size measurements, for each method.

In total, we developed 11 effort estimation models based on four different functional size measures. Table 1 summarizes these models, their input parameters, dependent parameters, analysis method, MMRE and MBRE

The accuracy of each effort estimation model was assessed by its Mean Magnitude of Relative Error (MMRE) and Mean Balanced Relative Error (MBRE), depicted in Equation 1 and Equation 4. The accuracy of an estimation model increases as its MMRE gets closer to 0, thus a small MMRE value indicates that the model is a good predictor. Following [4][30], a model with an MMRE value less than 0,25 and a MBRE value less than 0,35 was deemed accurate enough to be used in the organization under study.



**Fig. 2.** Scatterplots of actual effort vs. size for (1) UCP, (2) UFP, (3) USP and (4) CFP.

**Table 1.** Effort estimation models based on functional size measurements.

#	Analysis Method	Parameter Dependents	Model	MMRE	MBRE	
1	Linear	UCP	Effort	$EFF = -51,0 + 14,7 UCP$	0,86	0,98
2	Linear	USP	Effort	$EFF = -13,1 + 9,86 USP$	0,45	0,87
3	Polynomial	USP	Effort	$EFF = 18,47 + 3,392 USP + 0,1650 USP^2$	0,41	0,45
4	Polynomial	USP	Effort	$EFF = -8,75 + 12,30 USP - 0,3977 USP^2 + 0,008921 USP^3$	0,36	0,58
5	Linear	UFP	Effort	$EFF = -100 + 42,9 UFP$	0,48	0,53
6	Linear	DET, FTR	Effort	$EFF = -46,2 + 4,51 DET + 44,0 FTR$	0,36	0,54

		FTR			
7	Linear	DET,	Effort	$EFF = 1,7 + 1,47 \text{ DET} + 13,9 \text{ FTR}$	0,31 0,45
		FTR			
8	Linear	DET,	Effort	$EFF = -86,6 + 5,93 \text{ DET} + 53,3 \text{ FTR}$	0,23 0,36
		FTR			
9	Linear	CFP	Effort	$EFF = -78,6 + 31,0 \text{ CFP}$	0,58 0,51
10	Linear	E,R,WX	Effort	$EFF = -79,4 + 45,1 \text{ E} + 31,4 \text{ R} + 30,8 \text{ W} + 15,1 \text{ X}$	0,47 0,79
11	Exponential	CFP	Effort	$EFF = 0,789382 \times \text{CFP}^{2,63169}$	0,22 0,35

Model 11 from Table 1, which uses an exponential regression on COSMIC Function Points, is the most accurate effort estimation model, with an MMRE value of 0,22 and MBRE value of 0,35. Revisiting chart (4) from Figure 1, we confirm that this dataset follows an exponential pattern, which explains why model 11 provided the lowest MMRE and MBRE of all models.

On the other hand, model 1 from Table 1 is the least accurate estimation model, and it uses a linear regression on UCP. Revisiting chart (1) from Figure 1, there is no clear pattern in this dataset, which might explain why model 1 showed a high MMRE and MBRE.

## 5.2. Accuracy of convertibility models for functional size measurements (RQ2)

Our second research question sought to find accurate convertibility models of functional size measurements, in the context of the organization, using MMRE and MBRE as the accuracy performance measurements.

Further analysis was done to provide a set of models that allow convertibility between different functional size measurements (e.g. from USP to CFP), without having to execute the formal counting protocol for each size estimation method. As previously stated, these convertibility models are applicable only in the context of the organization under study, and are therefore not intended to be widely used across industry.

Twelve parametric statistical models were built, one for each possible convertibility combination between functional size measurements. We used linear regression analysis, and then we measured the accuracy by calculating the MMRE and MBRE for each model. These convertibility models are shown in Table 2, together with their analysis method, dependent variable, parameters (independent variables), MMRE and MBRE. Only models with MMRE less than 0,25 and MBRE less than 0,35 are accurate enough to be considered for common use in our company context.

**Table 2.** Convertibility models of software functional size measurements.

#	Analysis Method	Dependent Variable	Parameter	Model	MMRE	MBRE
1	Linear	UCP	USP	$UCP = 5,57 + 0,324 \text{ USP}$	0,18	0,20
2	Linear	UCP	CFP	$UCP = 3,09 + 1,09 \text{ CFP}$	0,21	0,24
3	Linear	UCP	UFP	$UCP = 2,24 + 1,57 \text{ UFP}$	0,24	0,28
4	Linear	USP	CFP	$USP = - 10,3 + 4,05 \text{ CFP}$	4,73	4,73
5	Linear	USP	UFP	$USP = - 16,8 + 6,88 \text{ UFP}$	8,37	8,37
6	Linear	USP	UCP	$USP = - 8,56 + 2,25 \text{ UCP}$	4,72	4,72
7	Linear	UFP	UCP	$UFP = 2,37 + 0,195 \text{ UCP}$	0,21	0,24
8	Linear	UFP	USP	$UFP = 2,96 + 0,113 \text{ USP}$	1,57	4,01

9	Linear	UFP	CFP	$UFP = 1,52 + 0,506 CFP$	0,14	0,15
10	Linear	CFP	UFP	$CFP = - 0,435 + 1,36 UFP$	0,38	0,38
11	Linear	CFP	UCP	$CFP = 1,72 + 0,392 UCP$	0,35	0,39
12	Linear	CFP	USP	$CFP = 3,05 + 0,203 USP$	0,23	0,25

Two interesting findings arise from Table 2:

1. Given the MMRE and MBRE results for models 4, 5 and 6, there is no linear model that offers an acceptable accuracy for USP convertibility from any other software size measure. Therefore, for this organization to get software size estimates in USP, they would need to perform SCRUM ceremonies (like Planning Poker) to provide them.
2. Using model 12 from Table 2, the organization would be able to convert their current USP estimates into CFP, and then use model 11 from Table 1 to estimate the effort. This approach could improve accuracy, rather than using the model 4 from Table 1. However, this approach would lead the company to have less accuracy performance since for probability addition rule, the MMRE must be added to get the total MMRE for the whole effort estimation process of 0,45, the same for MBRE, in this case would lead to get a total MBRE for the complete effort estimation process of 0,60

## 6. Discussion

Regarding RQ1, we found that model 11 from Table 1 (based on COSMIC Function Points) yields the most accurate effort estimates. Comparing the accuracy of COSMIC models, we observe that the CFP-based estimation model (model 11 in Table 1) performs slightly better than the UFP-based model (model 8 in Table 1). However, model 8 is applicable only to requirements with six or more data elements (DETs). Hence, only the COSMIC-based model proved to be accurate enough and applicable to any requirement, regardless of its size or number of data elements.

These results support the findings of Salmanoglu et al. [17], who concluded that an effort estimation model based on COSMIC Function Points was more accurate than another model based on User Story Points. They also used MMRE as the performance indicator for the models, making our results quite relatable to theirs.

Our results also relate to existing evidence by Ungan et al. [20], who also found that the effort model based on COSMIC Function Points performed better than the model based on User Story Points. Besides regression analysis, they included an artificial neural networks approach to build their models, yet they obtained similar results to ours.

It is worth mentioning that our study relates to the work of Paz et al. [13] in two aspects. First, we obtained similar results, as we determined that the most accurate effort estimation model was based on COSMIC function points. Second, we followed a similar approach to estimate the software size: we both used *Incremental COSMIC Function Points*, mostly because the application was developed using a SCRUM methodology, and this approach allows us to apply CFP in agile environments.

Also for RQ1, we found that model 1 from Table 1 (based on Use Case Points) yields the least accurate effort estimates. This finding differs from the results obtained by Sholiq et al. [19], as they found a slightly better performance using Use Case Points. There are, however, two main differences among the studies that could have affected the results. First, we used a regression analysis to build the effort estimation models, while they used instead a productivity factor (calculated by them) for that specific scenario. Second, they measured the size of already completed projects, whereas we measured the size of an ongoing agile project (incremental pieces of software).

Regarding RQ2, we were able to obtain accurate convertibility models for some functional size measurements, but not for every pair. The convertibility models that can be safely used in the organization context are models 1, 2, 3, 7, 9 or 12 from Table 2, given that their estimation error measured as MMRE is

less than 0,25 and MBRE is less than 0,35. These results relate to the findings of Lavazza et al. [7], as both presented a model that allow convertibility from CFP to UFP (in our study, it is model 9 from Table 2). However, in the case of the convertibility from UFP to CFP, Lavazza et al. [7] showed evidence that allows it, while we were not able to find an accurate model for it (model 10 from Table 2 was not accurate enough).

## 7. Conclusions

In this paper, we described a case study that compares the accuracy of effort estimation models derived through regression from four functional size measurements: Story Points, Use Case Points (UCP), IFPUG Function Points (UFP), and COSMIC Function Points (CFP). Additionally, we derived a set of convertibility models between functional size measurements, which can be used to easily convert the organization's historical measurements into different software size measures.

Our results confirm findings from previous studies such as [20], [13], and [17], which concluded that effort estimation models based on CFP show better accuracy than alternative methods compared in their studies.

However, compared to counting USP, none of the convertibility models are accurate enough to estimate software size. Thus, in the organization under study, current agile ceremonies must continue to be performed in order to estimate work size.

Our convertibility models reaffirm previous evidence by Lavazza et al. [7], as they presented a scenario where UFP and CFP were correlated, and a model is proposed to convert from one software sizing method to the other. Although expert judgment is still generally preferred as the estimation method in agile environments [21], our recommendation to the organization under study is to adopt CFP as their sizing method, especially as the number of projects grows and the company scales up. There is ample evidence in the literature that using CFP could improve an organization's estimates [3], [20], [13], and [17].

As future work, we plan to replicate this study using a larger dataset with an extended sample of functional requirements. Another possible extension to our work would be to perform a statistical analysis to compare the effort estimation models, including the statistical tests suggested by Lavazza et al. [8] such as paired t-tests of the MRE, or paired tests of the absolute residuals.

## References

31. Briand, L., & Maxwell, K. An Assessment and Comparison of Common Software Cost Estimation Modeling Techniques. *ICSE '99*, 1999, Pages: 313-323.
32. Chi-Jui, L., & Yeh, D.-M. A Software Maintenance Project Size Estimation Tool Based On Cosmic Full Function Point. *2016 International Computer Symposium (ICS)*, 2016, Pages: 555-560.
33. Desharnais, J., Buglione, L. & Kocaturk, B., Using the COSMIC Method to Estimate Agile User Stories. Proceedings of the *12th International Conference on Product Focused Software Development and Process Improvement*, 2011.
34. Foss, T., Stensrud, E., Kitchenham, B., & Myrtveit, I. A Simulation Study of the Model Evaluation Criterion MMRE. *IEEE Transactions on Software Engineering*, Volume: 29, Issue: 11, 2003, Pages: 985-995.
35. Jørgensen, M., Boehm, B., & Rifkin, S. (2009). Software development effort estimation: Formal models or expert judgment? *IEEE software*, 26(2), 14-19.
36. Barbara Kitchenham and Emilia Mendes. 2009. Why comparative effort prediction studies may be invalid. In *Proceedings of the 5th International Conference on Predictor Models in Software Engineering (PROMISE '09)*. ACM, New York, NY, USA, , Article 4 , 5 pages. DOI=<http://dx.doi.org/10.1145/1540438.1540444>
37. Lavazza, L., & Liu, G.. A Study of the Correlation between Functional Size Measures and Object-oriented Measures from UML Requirements Models. *IWSM Mensura 2018*, Pages:54-69
38. Lavazza, L., & Morasca, S. On the Evaluation of Effort Estimation Models. EASE'17. 2017.
39. Lenarduzzi, V, Lunesu, I, Matta, M & Taibi, D., Functional Size Measures and Effort Estimation in Agile Development: A Replicated Study. In Suomalainen, T. *Continuous Strategy Process in the context of Agile and Lean Software Development*. Springer, 2015.
40. Mendes, E., Watson, I., Triggs, C. et al. A Comparative Study of Cost Estimation Models for Web Hypermedia Applications. *Empirical Software Engineering* (2003) 8: 163. <https://doi.org/10.1023/A:1023062629183>
41. Mittas, N., & Angelis, L. Ranking and Clustering Software Cost Estimation Models through a Multiple Comparisons Algorithm. *IEEE Transactions on Software Engineering* Vol. 39, No. 4, 2013, Pages: 537- 551.

42. Nikolaos Mittas and Lefteris Angelis. 2010. Visual comparison of software cost estimation models by regression error characteristic analysis. *J. Syst. Softw.* 83, 4 (April 2010), 621-637. DOI=<http://dx.doi.org/10.1016/j.jss.2009.10.044>
43. Paz, F., Zapata, C., & Pow-Sang, J. A. An Approach for Effort Estimation in Incremental Software Development using COSMIC Function Points. *ESEM'14*, 2014.
44. Kai Petersen, Sairam Vakkalanka, Ludwik Kuzniarz, Guidelines for conducting systematic mapping studies in software engineering: An update, *Information and Software Technology*, Volume 64, 2015, Pages 1-18,
45. Phannachitta, P. Robust Comparison of Similarity Measures in Analogy-Based Software Effort Estimation. *11th International Conference on Software, Knowledge, Information Management and Applications (SKIMA)*, 2017.
46. Runeson, P., & Höst, M. Guidelines for conducting and reporting case study research in software engineering. *Empiric Software Eng.* 2008. Pages: 131–164.
47. Salmanoglu, M., Hacaloglu, T., & Demirors, O. Effort Estimation for Agile Software Development: Comparative Case Studies Using COSMIC Functional Size Measurement and Story Points, *IWSM/Mensura '17*, 2017.
48. Santana C., Leoneo F., Vasconcelos A., Gusmão C. (2011) Using Function Points in Agile Projects. In: Sillitti A., Hazzan O., Bache E., Albaladejo X. (eds) *Agile Processes in Software Engineering and Extreme Programming. XP 2011. Lecture Notes in Business Information Processing*, vol 77. Springer, Berlin, Heidelberg
49. Sholiq, Renny, S., & Pribadi, A. A Comparative Study of Software Development Size Estimation Method: UCPabc vs Function Points. *4th Information Systems International Conference 2017*, Pages 470-477.
50. Ugan, E., Çizmeli, N., & Demirörs, O. Comparison of Functional Size Based Estimation and Story Points, Based on Effort Estimation Effectiveness in SCRUM Projects. *Euromicro Conference on Software Engineering and Advanced Applications*, 2014, Pages. 77-80.
51. Usman, M., Mendes, E., & Börstler, J. Effort Estimation in Agile Software Development: A Survey on the State of the Practice. *EASE'15*, 2015.
52. F.G. Wilkie, I.R. McChesney, P. Morrow, C. Tuxworth, N.G. Lester, The value of software sizing, *Information and Software Technology*, Volume 53, Issue 11, 2011, Pages 1236-1249
53. Bente, A., Hege, D., Sjøberg, D., & Magne, J. Estimating Software Development Effort Based on Use Cases - Experiences from Industry. *The Unified Modeling Language. Modeling Languages, Concepts, and Tools*, 2001, Pages: 487-502.
54. Karner, G. Metrics for Objectory. Diploma thesis. University of Linköping, No. LiTH-IDA-Ex-9344:21, 1993
55. Popli, R., & Chauhan, N. Managing Uncertainty of Story-Points in Agile Software. *2nd International Conference on Computing for Sustainable Global Development (INDIACom)*, 2015, Pages: 1357-1361.
56. Gandomani, T. J., & Mahsa Radnejad, H. F. Planning Poker in cost estimation in Agile methods: Averaging Vs. Consensus. *2019 IEEE 5th International Conference on Knowledge-Based Engineering and Innovation (KBEL)*, 2019, Pages: 66-71.
57. Garmus, D., & Herron, D. *Function Point Analysis: Measurement Practices for Successful Software Projects*. Addison-Wesley, 2001
58. Gandomani, T. J., & Mahsa Radnejad, H. F. Planning Poker in cost estimation in Agile methods: Averaging Vs. Consensus. *2019 IEEE 5th International Conference on Knowledge-Based Engineering and Innovation (KBEL)*, 2019, Pages: 66-71.
59. Chi-Jui, L., & Yeh, D.-M. A Software Maintenance Project Size Estimation Tool Based On Cosmic Full Function Point. *2016 International Computer Symposium (ICS)*, 2016, Pages: 555-560.
60. Miyazaki, Y., et al., Robust regression for developing software estimation models. *Journal of Systems and Software*, 1994. 27(1): Pages. 3-16.
61. M. Jørgensen. A critique of how we measure and interpret the accuracy of software development effort estimation. *Proceedings of 1st International Workshop on Software Productivity Analysis and Cost Estimation*, 2007, Pages 15–22.