

Estimación del esfuerzo del proceso de implantación de software basada en el método de puntos de caso de uso

Pablo Vazquez^{1,2}, Marisa Panizzi², Rodolfo Bertone³

¹Programa de Maestría en Ingeniería en Sistemas de Información. Escuela de Posgrado.
Universidad Tecnológica Nacional. Regional Buenos Aires (UTN-FRBA)
Castro Barros 91. (C1178AAA). CABA. Argentina.

²Departamento de Ingeniería en Sistemas de Información. Universidad Tecnológica Nacional.
Regional Buenos Aires (UTN-FRBA).

Medrano 951. (C1179AAQ). CABA, Argentina.

³Instituto de Investigaciones en Informática LIDI. Facultad de Informática. Universidad Nacional
de La Plata (UNLP)
50 y 120 – La Plata, Argentina.

vazpablo@gmail.com, marisapanizzi@outlook.com, bertone@lidi.unlp.edu.ar

Resumen

La estimación precisa del esfuerzo es esencial en cualquier tipo de proyecto de software; esta afirmación se la considera para el proceso de implantación de sistemas informáticos. Si esta estimación falla o se presenta un cambio en la misma puede afectar al proceso de implantación, lo que implicaría incorporar recursos adicionales impactando en el costo del mismo. Una subestimación del proceso puede generar una asignación menor de recursos para cumplir los requisitos del proceso de implantación. Esto generaría la disminución de la calidad o fracaso del proceso. En este trabajo, se presenta una revisión documental de diferentes clasificaciones de métricas. Se plantea y justifica la propuesta de una clasificación de métricas para el proceso de implantación. Se desarrolla la construcción de la métrica "esfuerzo" para el proceso en estudio, la cual se basa en el método de puntos de caso de uso. Por último, se presenta un caso de estudio que sirve como validación inicial para la métrica propuesta.

1. Introducción

En Ingeniería de Software, para obtener un software de calidad es necesario realizar mediciones al inicio del proceso, estimar el tamaño del proyecto, sus costos y el tiempo que requerirá la construcción del producto software. Estas mediciones deben realizarse de manera constante durante todas las fases del proceso de construcción. Las métricas ayudan a comprender estos procesos [1].

Las métricas ayudan a comprender tanto el proceso técnico utilizado para desarrollar el producto, como el propio producto.

En este sentido se puede decir que es habitual confundir el término *métrica* con los términos *medición* y *medida* y tratarlos como sinónimos.

Fenton define *medición* como "el proceso por el cual los números o símbolos son asignados a atributos o entidades en el mundo real tal como son descriptos de acuerdo a reglas claramente definidas" [2].

Pressman sugiere que una *medida* "proporciona una indicación cuantitativa de extensión, cantidad, dimensiones, capacidad y tamaño de algunos atributos de un proceso o producto" [3].

En IEEE 610 "Standard Glossary of Software Engineering Terms", se define *métrica* como: "una medida cuantitativa del grado en que un sistema, componente o proceso posee un atributo dado" [4].

Según Pressman, las métricas son escalas de unidades sobre las cuales puede medirse un atributo cuantificable [3].

Lascano define *indicador* como una métrica o la combinación de varias de ellas. Este indicador proporciona información objetiva que sumados a criterios de decisión definidos, permiten a los participantes de un proyecto medir la ejecución y realizar ajustes tanto en el producto como en los procesos que se emplean [5].

Según Lopez et al. [6], para usar las métricas adecuadamente, no es suficiente con medir los atributos cuantitativamente, sino que es necesario tener en cuenta consideraciones como las unidades que se aplican, el tipo de software al que es aplicable y las condiciones en que se deben recoger los datos.

En Kan [7] y en Basso [1] se propone una de clasificación de métricas, la cual se basa en el contexto o dominio de aplicación y las características o atributos del software. Dicha clasificación define:

- **Métricas del producto:** describen características del producto tales como tamaño, complejidad, características de diseño, rendimiento y nivel de calidad.
- **Métricas del proceso:** pueden ser utilizadas para mejorar el proceso de desarrollo y mantenimiento del software. Algunos ejemplos son: la efectividad de la remoción de defectos durante el desarrollo y el tiempo de respuesta en el proceso de corrección de defectos.
- **Métricas del proyecto:** describen las características y ejecución del proyecto. Algunos ejemplos son: el número de desarrolladores de software, el comportamiento del personal durante el ciclo de vida del proyecto, el costo, el cronograma y la productividad.

Kan [7] menciona que existen métricas que pertenecen a múltiples categorías; por ejemplo, las métricas de calidad del proceso de un proyecto se pueden aplicar tanto a métricas del proceso como a métricas del proyecto.

Beltrán et al. [8] propone una clasificación de métricas con el fin de describir la conducta del software. Estas métricas miden aspectos como: competencia, calidad, desempeño y complejidad del software.

Tamayo Osorio [9] propone una clasificación de métricas en función de la funcionalidad, complejidad y eficiencia del desarrollo del software, organizándolas en: técnicas, de calidad, de productividad, orientada a la persona, orientada al tamaño, orientada a la función.

Según la norma ISO/IEC 9126 [10][11][12] se pueden clasificar en tres categorías, según sea su naturaleza:

- **Métricas básicas:** son métricas que se obtienen directamente del análisis del código o la ejecución del software. No involucra ningún otro atributo ni depende de otras métricas. Estas métricas se denominan directas. Entre las métricas básicas se tiene la cantidad de líneas de código del programa o de cada módulo, la cantidad de horas de desarrollo, la cantidad de fuentes de datos o tablas a utilizar, la cantidad de atributos y registros de una tabla, entre otras [3].
- **Métricas de agregación:** son métricas compuestas a partir de un conjunto definido de métricas básicas (o directas), generalmente mediante una suma ponderada.
- **Métricas derivadas:** son métricas compuestas por una función de cálculo matemático, que utiliza como variables de entrada el valor de otras métricas. Estas métricas se denominan indirectas. Entre las métricas

derivadas se tiene la cantidad de líneas de código producidas por hora y por persona, el porcentaje de completitud del proyecto, el tamaño promedio de los módulos del software, el tiempo promedio que una persona dedica a corregir los defectos de un módulo, entre otras [3].

En esta línea de investigación, los autores consideran al proceso de implantación de sistemas informáticos como una de las fases o etapas que se realizan en la construcción del software. Esta fase contempla un conjunto de actividades y tareas necesarias que permiten la transferencia del producto software finalizado a su ambiente de utilización.

La investigación realizada por los autores en [13][14] sobre el proceso de implantación o instalación de sistemas informáticos, ha evidenciado la ausencia de métricas como elemento del proceso.

El presente trabajo, se desarrolla bajo la hipótesis propuesta de que las metodologías o estándares de desarrollo de sistemas informáticos proponen, de manera incompleta, los elementos necesarios de un proceso de implantación de software. De esta hipótesis se deriva la pregunta de investigación: ¿Se puede lograr cubrir la vacancia de Métricas del Proceso de implantación de un sistema informático de manera que se integre como parte del proceso? En respuesta a esta pregunta, este trabajo presenta la métrica "esfuerzo".

2. Elementos de trabajo y metodología

Para el desarrollo de este trabajo, se seguirá un enfoque de investigación clásico propuesto por Riveros et al. [15] y Creswell [16] con énfasis en la producción de tecnologías según Sabato y Mackenzie [17]; identificando métodos y materiales necesarios para desarrollar el trabajo:

- **Métodos.** Se utilizarán revisiones sistemáticas y Prototipado Evolutivo Experimental (Método de la Ingeniería). Las revisiones sistemáticas propuestas por Argimón, de artículos científicos siguen un método explícito para resumir la información sobre determinado tema o problema. Se diferencia de las revisiones narrativas en que provienen de una pregunta estructurada y de un protocolo previamente realizado [18]. El prototipado evolutivo experimental expuesto por Basili, consiste en desarrollar una solución inicial para un determinado problema. Esto genera un refinamiento de manera evolutiva por prueba de aplicación de dicha solución a casos de estudio (problemáticas) de complejidad creciente. El proceso de refinamiento concluye al estabilizarse el prototipo en evolución [19].

- **Materiales:** Se ha utilizado la bibliografía existente, la cual ha sido obtenida de repositorios digitales y de libros.

3. Desarrollo

Para la propuesta de métricas del proceso de implantación, se definió una clasificación basada en la propuesta de ISO/IEC 9126 [10][11][12] y la propuesta definida por Basso [1] y Kan [7]. De la última clasificación se consideran solo las categorías: métricas del proceso y métricas del producto. Se consideran las métricas del proceso debido a que resulta de interés las tareas, actividades y roles que comprenden al proceso de implantación de sistemas informáticos. La métrica del producto se la considera por la necesidad de contemplar características como la complejidad del producto software a instalar, los requisitos de instalación para el producto software, la integración con la infraestructura del cliente, entre otras; estas tendrán impacto en el proceso de implantación estudiado. En la clasificación propuesta se adiciona una tercera categoría: "Persona", dado a la existencia del peopeware y su impacto en el proceso en estudio.

De la clasificación antes mencionada, se define una matriz que permite visualizar como resulta la intersección de las métricas que se propondrán en esta línea de investigación, (Tabla 1).

Tabla 1. Matriz de clasificación propuesta.

Métrica	Proceso	Producto	Persona
Básica			
Agregada			
Derivada			

En este trabajo se presenta la métrica de "Esfuerzo" del proceso de implantación de sistemas informáticos. Esta se clasifica como métrica Derivada y métrica del Proceso.

Dapozo et al. [20] en su trabajo menciona que, los métodos para estimar el esfuerzo de desarrollo se clasifican en: métodos paramétricos donde el proceso de cuantificación del resultado está basado en un proceso mecánico, por ejemplo, la aplicación de una fórmula derivada de los datos históricos. Dentro de este conjunto se encuentran métodos como COCOMO y COCOMO II de Boehm, SLIM de Putnam, y Puntos de Caso de Uso propuesto por Karner. La segunda categoría, representa a los métodos heurísticos, en los cuales la cuantificación del resultado se produce a partir del juicio y/o la experiencia de un experto.

Como método de cálculo del esfuerzo requerido para la implantación del software, se toma como base el método de estimación por Puntos de Caso de Uso propuesto por Karner [21]. La ventaja de estimar sobre Puntos de caso de uso se debe al logro de una estimación temprana sobre un conocimiento cierto de los requerimientos a desarrollar. Esta forma de estimar el esfuerzo medido en horas hombre (HH) requerido para el desarrollo de un determinado producto software, resulta de interés para las empresas que se dedican a la construcción de software. Esto se debe a que es uno de los principales factores para la determinación del costo del producto.

A continuación se describen los pasos propuestos en el método:

1. Se identifican y clasifican los casos de uso según la cantidad de transacciones en tres tipos: simple, medio y complejo con su respectivo factor de peso (ver Tabla 2).

Tabla 2. Clasificación de Caso de Uso y sus pesos

Tipo de Caso de Uso	Descripción	Peso
Simple	Menos de 3 transacciones	5
Medio	de 4 a 7 transacciones	10
Complejo	Más de 7 transacciones	15

2. Se clasifican los actores que interactúan con el sistema informático según la complejidad de la interacción en tres tipos: simple, medio y complejo, con su respectivo factor de peso (ver Tabla 3).

Tabla 3. Clasificación de actores y sus pesos

Tipo de actor	Descripción	Peso
Simple	Otro sistema que interactúa con el sistema a desarrollar mediante una interfaz de programación (API, Application Programming Interface)	1
Medio	Otro sistema que interactúa con el sistema a desarrollar mediante un protocolo o una interfaz basada en texto	2
Complejo	Una persona que interactúa con el sistema mediante una interfaz gráfica	3

3. Para el cálculo del total del peso de los casos de uso sin ajustar (UUCW) se utiliza la siguiente ecuación:

$$UUCW = \Sigma \text{ Casos de uso simples} * WF + \Sigma \text{ Casos de uso medios} * WF + \Sigma \text{ Casos de uso complejos} * WF$$

4. El peso total de los actores (UAW) se calcula con la siguiente ecuación:

$$UAW = \Sigma \text{ Actor simple} * WF + \Sigma \text{ Actor medio} * WF + \Sigma \text{ Actor complejo} * WF$$

5. Una vez obtenidos los valores de UAW y UUCW se los utiliza para calcular los puntos de caso de uso sin ajustar (UUCP) por medio de la siguiente ecuación:

$$UUCP = UAW + UUCW$$

6. Se calcula los puntos de caso de uso ajustados (AUCP) por medio de la siguiente fórmula:

$$AUCP = UUCP * TF * EF$$

Donde TF es el factor de Complejidad Técnica (ver Tabla 4) y EF es el factor del Entorno (ver Tabla 5).

Tabla 4. Factores de complejidad técnica y pesos propuestos (TF).

Factor	Descripción	Peso
T1	Sistema distribuido	2
T2	Objetivos de performance o tiempo de respuesta	1
T3	Eficiencia del usuario final	1
T4	Procesamiento interno complejo	1
T5	El código debe ser reutilizable	1
T6	Facilidad de instalación	0.5
T7	Facilidad de uso	0.5
T8	Portabilidad	2
T9	Facilidad de cambio	1
T10	Concurrencia	1
T11	Incluye objetivos especiales de seguridad	1
T12	Provee acceso directo a terceras partes	1
T13	Se requieren facilidades especiales de entrenamiento a usuarios	1

Tabla 5. Factores del entorno y pesos propuestos (EF).

Factor	Descripción	Peso
E1	Familiaridad con el modelo de proyecto utilizado	1.5
E2	Experiencia en la aplicación	0.5
E3	Experiencia en orientación a objetos	1
E4	Capacidad del analista líder	0.5
E5	Motivación	1
E6	Estabilidad de los requerimientos	2
E7	Personal part-time	-1
E8	Dificultad del lenguaje de programación	-1

A continuación se observa la ecuación correspondiente al Factor de complejidad Técnica (TF):

$$TF = 0.6 + (0.01 * TWF)$$

$$TWF = \Sigma (\text{Peso}_i \times \text{Valor asignado}_i)$$

El Factor del Entorno se obtiene a través de la siguiente ecuación:

$$EF = 1.4 + (-0.03 * EWF)$$

$$EWF = \Sigma (\text{Peso}_i \times \text{Valor asignado}_i)$$

Se cuantifica cada uno de los factores con un valor de 0 a 5, donde 0 significa un aporte irrelevante y 5 un aporte muy importante.

7. Una vez calculados los puntos de caso de uso ajustados (AUCP), se obtiene el esfuerzo (E) del proceso de desarrollo por medio de la siguiente ecuación:

$$E = AUCP \times CF$$

Donde CF corresponde a las horas por persona por punto de caso de uso.

Karner [21] originalmente sugirió que cada Puntos de Caso de Uso requiere un esfuerzo de 20 horas-hombre.

Para el cálculo del esfuerzo del proceso de implantación, se considera un registro histórico de proyectos desarrollados en los últimos 12 años en una consultora de la República Argentina. En esta organización se desarrolló el caso de estudio que se presenta en la sección 4 de esta comunicación.

Como resultado del análisis de la base de proyectos, se propone una distribución del esfuerzo expresado en un rango de porcentajes para las actividades genéricas que conforman un proyecto de software (ver Tabla 6).

Tabla 6. Distribución de esfuerzo para desarrollo de software (rango de porcentajes).

Etapas	Rango
Relevamiento inicial	(2%-5%)
Especificación de requerimientos y análisis	(15%-25%)
Arquitectura y diseño	(10%-15%)
Desarrollo	(25%-40%)
Testing	(10%-20%)
Implantación	(6%-8%)

Es necesario destacar que cada equipo de desarrollo ajusta los porcentajes y las definiciones de las etapas en base a su marco de trabajo, características del producto y proyecto asociado.

8. Como propuesta para el cálculo del esfuerzo requerido para el proceso de implantación se presenta la siguiente ecuación:

$$EPI = (E * \% \text{Implantación}) / \% \text{Desarrollo}$$

Donde,

EPI: Esfuerzo Proceso de Implantación (HH)

4. Caso de estudio

Se presenta un caso de estudio para realizar una validación inicial de la métrica "esfuerzo" para el proceso de implantación de software. Esta se obtiene a partir de la aplicación de un método de estimación basado en puntos de caso de uso. El caso de estudio corresponde a la implantación de un desarrollo de tipo mejora; esta impacta sobre un sistema de gestión de agencias del rubro de la publicidad para Latinoamérica ya implantado. La consultora en la que se desarrolló la experimentación de la métrica de esfuerzo, es una empresa multinacional que se encuentra en la República Argentina y ofrece servicios de consultoría.

El sistema de gestión es un ERP (en inglés, Enterprise Resource Planning), el cual fue desarrollado a medida para la gestión de empresas de publicidad; este cuenta con los siguientes módulos:

- Clientes
- Proveedores
- Contabilidad
- Tesorería
- Administración y parametrización
- Rendiciones de gastos
- Seguridad

La mejora a implementar tiene impacto sobre los módulos de Clientes y Administración y Parametrización. Esta incorpora la modalidad de facturación electrónica,

que requiere de la emisión de un archivo de texto, la generación automática de un archivo XML y un PDF en la generación de cada factura y nota de crédito. Este desarrollo también incorpora un proceso de solicitud de creación de notas de crédito y su posterior aprobación.

Los requisitos funcionales de la mejora a desarrollar han sido capturados en 15 casos de uso, los cuales se enuncian a continuación:

- Modificar información de la empresa
- Generar archivo para ente regulador
- Generar factura automática
- Generar factura manual
- Generar nota de crédito
- Generar factura automática de medios
- Generar factura manual de medios
- Generar nota de crédito de medios
- Generar factura de varios
- Generar nota de crédito de varios
- Generar factura por volumen
- Generar nota de crédito por volumen
- Regenerar archivo para ente regulador
- Solicitud de generación de nota de crédito
- Aprobar de notas de crédito.

A continuación se aplicó el método de estimación del esfuerzo de desarrollo propuesto por Karner al caso de estudio:

1. Se clasificó cada Caso de Uso según el número de transacciones (ver Tabla 7).

Tabla 7. Casos de Uso clasificados.

Tipo de Caso de Uso	Peso	Cantidad de transacciones
Simple	5	8
Medio	10	5
Complejo	15	2

2. Se identificaron y clasificaron los Actores que intervienen en los Casos de Usos detectados. Estos son:

- *Administrador*: encargado de la administración y parametrización de los datos.
- *Facturista*: encargado de generar las facturas y notas de crédito
- *Aprobador de notas de crédito*: encargado de aprobar las solicitudes de notas de crédito.

En este caso, los usuarios constituyen actores de tipo complejo, ya que se trata de personas utilizando el sistema mediante una interfaz gráfica, al cual se le asigna un peso 3. (Ver Tabla 8)

Tabla 8. Actores clasificados.

Tipo de actor	Peso	Cantidad
Simple	1	0
Medio	2	0
Complejo	3	3

3. Se calculó el factor de peso de los Casos de Uso sin ajustar (UUCW).

$$UUCW = 5*8 + 10*5 + 15*2$$

$$UUCW = 120$$

4. Se calculó el factor de peso de los Actores sin ajustar (AUW).

$$AUW = 3*3$$

$$AUW = 9$$

5. A partir de los valores obtenidos de UUCW y AUW, se obtienen los puntos de caso de uso sin ajustar (UUCP).

$$UUCP = UUCW + AUW$$

$$UUCP = 120 + 9 = 129$$

6. Se valorizaron los factores técnicos (ver Tabla 9).

Tabla 9. Factores técnicos valorizados (TF).

Factor	Peso	Valor
T1	2	0
T2	1	1
T3	1	1
T4	1	3
T5	1	2
T6	0,5	3
T7	0,5	3
T8	2	2
T9	1	3
T10	1	2
T11	1	2
T12	1	0
T13	1	1

7. Se calculó el valor TF (Factores de complejidad técnica) con los factores técnicos valorizados:

$$TF = 0.6 + (0.01 * TWF)$$

$$TF = 0.6 + (0.01 * 23)$$

$$TF = 0.6 + 0.23 = 0.83$$

8. Se valorizan los factores del Entorno (ver Tabla 10).

Tabla 10. Factores del entorno valorizados (EF).

Factor	Peso	Valor
E1	1,5	5
E2	0,5	5
E3	1	4
E4	0,5	5
E5	1	2
E6	2	2
E7	-1	0
E8	-1	0

9. Se calculó el valor EF (Factor del Entorno) con los factores de entorno valorizados:

$$EF = 1.4 + (-0.03 * EWF)$$

$$EF = 1.4 + (-0.03 * 22.5)$$

$$EF = 1.4 - 0.675 = 0.725$$

10. Con los valores de UUCP, TF y EF obtenidos, se calculó los Puntos de Caso de Uso ajustados (AUCP).

$$AUCP = UUCP * TF * EF$$

$$AUCP = 129 * 0.83 * 0.725$$

$$AUCP = 77.62$$

11. Se calculó el Esfuerzo (E) del desarrollo con los Puntos de Caso de Uso ajustados (AUCP). Para la variable CF se tomó el valor propuesto por Karner.

$$E = AUCP * CF$$

$$E = 77.62 * 20$$

$$E = 1552.4 \text{ HH}$$

12. Se calculó el esfuerzo del proceso de implantación, para el cual se utilizó el porcentaje real (7%) definido para el caso de estudio (ver Tabla 11).

Tabla 11. Porcentajes definidos para el caso de estudio.

Etapas	Rango	Porcentaje caso de estudio
Relevamiento inicial	(2%-5%)	3%
Especificación de requerimientos y análisis	(15%-25%)	20%
Arquitectura y diseño	(10%-15%)	15%
Desarrollo	(25%-40%)	35%
Testing	(10%-20%)	20%
Implantación	(6%-8%)	7%

$$EPI = (E * \%Implantación) / \%Desarrollo$$

$$EPI = (1552.4 * 7%) / 35\%$$

$$EPI = 310.48HH$$

En la tabla 12, se presentan los valores resultantes de la aplicación del método de Karner [21]. Los valores de la estimación de esfuerzo están expresados en horas/hombre. Se comparan los valores de estimación del esfuerzo real y los obtenidos por la aplicación del método.

Tabla 12. Comparación del esfuerzo real estimado vs esfuerzo estimado por el método

Caso de estudio	Esfuerzo Real Estimado	Esfuerzo Estimado por el método
Esfuerzo desarrollo	590	1552,4
Esfuerzo Implantación	118	310,48

La estimación del esfuerzo para el proceso de implantación de sistemas informáticos con el método de cálculo del esfuerzo por puntos de caso de uso, presenta un desvío significativo respecto del esfuerzo real estimado. Se puede considerar que, un factor de influencia en el desvío se debe al nivel de experiencia de los responsables en la aplicación del método. La falta de experiencia y subjetividad del estimador impacta en los valores asignados a los factores técnicos y del entorno, como así también en el momento de clasificar los niveles de complejidad de los casos de usos y los actores.

Si bien se ha utilizado un método paramétrico para la estimación del proceso en estudio, los autores reflexionan sobre la necesidad de evaluar otros métodos de estimación mediante cálculos que permitan aproximar brecha detectada entre la estimación real y la obtenida.

5. Conclusiones

Se logró definir una matriz de clasificación de métricas aplicable a la fase de implantación de proyectos

de software en organizaciones de la República Argentina, la cual permite avanzar en la propuesta de otras métricas para el fortalecimiento del proceso en estudio.

Para el cálculo del esfuerzo del proceso de implantación, se tomó como base el método por puntos de caso de uso propuesto por Karner, este método en comparación con otros permite realizar una estimación temprana del esfuerzo del desarrollo.

La disponibilidad de datos de proyectos anteriores, contribuyó a la determinación de la distribución del esfuerzo total requerido para el ciclo de vida del proyecto, y por medio de una extrapolación porcentual se obtuvo el esfuerzo necesario para llevar a cabo el proceso de implantación.

Se presentó una validación inicial de la métrica de esfuerzo en un caso de estudio de un proyecto de desarrollo, la cual presentó una desviación considerable en la estimación. Esta se debe a la sensibilidad de los factores técnicos y del entorno, donde se pone en evidencia la subjetividad introducida por el estimador, aunque esta no invalida el método.

Como trabajo futuro se identifica: (a) refinar la métrica de "esfuerzo" basándose en otros métodos de estimación (b) experimentar la métrica de "esfuerzo" en otros casos de estudio; (c) avanzar con la construcción de otras métricas para el proceso de implantación de sistemas informáticos considerando la clasificación propuesta en la sección 3 de este artículo (básicas, agregadas y derivadas).

6. Referencias

- [1] Basso, D., *Propuesta de Métricas para Proyectos de Explotación de Información*, 2014. <http://sistemas.unla.edu.ar/sistemas/redisla/ReLAIS/relais-v2-n4-157-218.pdf>. Último acceso el 14/12/2017.
- [2] Fenton, E., *Software Metrics A Rigorous approach*, Chapman & Hall, Primera Edición, 1991.
- [3] R. Pressman, *Ingeniería del software: Un enfoque práctico*, Mc Graw Hill, 7ma Edición, 2010.
- [4] IEEE 610. *IEEE Standard Glossary of Software Engineering Terminology*. IEEE Std 610.12-1990, 1990.
- [5] Lascano, N., *Un Conjunto de Métricas para Proyectos de Transición de Software Offshore*, Tesis Magister en Ingeniería de Software, Facultad de Informática, Universidad Nacional de La Plata, 2013. <http://sedici.unlp.edu.ar/handle/10915/32513>. Página veinte al 26/06/2017.
- [6] López, A., Sánchez, A., Montejano, G., *Definición de Métricas de Calidad para Productos de Software*. XVIII Workshop de Investigadores en Ciencias de la Computación (WICC 2016, Entre Ríos, Argentina). Red de Universidades con Carreras en Informática (RedUNCI). ISBN: 978-950-698-377-2, 2016. <http://sedici.unlp.edu.ar/handle/10915/53444>. Página veinte al 16/12/2017.

- [7] Kan, S., *Metrics and models in software quality engineering*, second edition, Addison Wesley, 95-456, ISBN 0-201-72915-6, 2002, Referenciado en 126, 128, 135
- [8] Beltrán, N., Castro León, G., *Gestión de calidad en desarrollo de software*, Universidad Nacional Mayor de San Marcos, Facultad de Ingeniería de Sistemas e Informática. ISSN 1816-3823, 2011.
- [9] Tamayo Osorio, P., *Métricas de calidad para esquemas conceptuales Una revisión crítica*. Revista Reune No. 1 pp. 35-60, 2016.
- [10] ISO/IEC 9126-2, *Software engineering - Product quality - Part 2 External metrics*, 2003, <http://www.iso.org/iso/home.html>.
- [11] ISO/IEC 9126-3. 2003. *Software engineering - Product quality - Part 3 Internal metrics*. <http://www.iso.org/iso/home.html>.
- [12] ISO/IEC 9126-4. 2004. *Software engineering - Product quality - Part 4 Quality in use metrics*. <http://www.iso.org/iso/home.html>.
- [13] Panizzi, M., Bertone R., Hossian A., *Proceso de Implantación de Sistemas Informáticos – Identificación de vacancias en Metodologías Usuales*, Departamento de Ingeniería en Sistemas de Información, Universidad Tecnológica Nacional, Facultad Regional Buenos Aires, Actas de las conferencias Iadis Iberoamericanas (CIACA 2017), ISBN: 978-989-8533-70-8, 2017.
- [14] Vazquez, P., Panizzi, M., *Implantación de Sistemas Informáticos – Identificación de Elementos del Proceso*, 5º Congreso Nacional de Ingeniería Informática / Sistemas de Información Ing. de Sistemas, Ing. de Software y Gestión de Proyectos (CONAIISI 2017), ISSN: 2347-0372, 2017.
- [15] Riveros, H. y Rosas, L., *El Método Científico Aplicado a las Ciencias Experimentales*, Editorial Trillas, México, ISBN 96-8243-893-4, 1985.
- [16] Creswell, J., *Educational Research: Planning, Conducting, and Evaluating Quantitative and Qualitative Research*, Prentice Hall, ISBN 10: 01-3613-550-1, 2002.
- [17] Sábado, J. y Mackenzie, M., *La Producción de Tecnología*, Editorial Nueva Imagen, México, ISBN 968-429-348-8, 1982.
- [18] Argimón, J. 2004. *Métodos de Investigación Clínica y Epidemiológica*. Elsevier España. 84-8174-709-2.
- [19] Basili. *The Experimental Paradigm in Software Engineering*. En *Experimental Software Engineering Issues: Critical Assessment and Future Directions* (Ed. Rombach, H., Basili, V., Selby, R.). Lecture Notes in Computer Science, Vol. 706. (1933). ISBN 978-3-540-57092-9.
- [20] Dapozo, G., Greiner, C., Medina, Y., Ferraro, M., Pedrozo Petrazzini, G., Lencina, B., *Métodos de estimación de software. Un análisis desde un enfoque evolutivo*, de III Jornadas de Investigación en Ingeniería del NEA y Países Limítrofes, Resistencia, 2014.
- [21] Karner, G., *Metrics for Objectory*, Degree thesis, Universidad de Linkoping, Suecia, 1993.