

ESTIMACIÓN DE ESFUERZO Y COSTO EN EL DESARROLLO DE UN SOFTWARE PARA LA DIVULGACIÓN DE INFORMACIÓN A TRAVÉS DEL TELÉFONO CELULAR¹

Álvaro Ignacio Morales González

Candidato a Magíster en Instrumentación Física

Especialista en Instrumentación Física

Ingeniero de Sistemas

Docente asistente Universidad Católica Popular del Risaralda

Docente Catedrático asistente Universidad Tecnológica de Pereira

amorales@ucpr.edu.co

Luis Eduardo Pelaez Valencia

Estudiante de Magíster en Ingeniería del Software

Especialista en Propiedad Intelectual

Ingeniero de Sistemas

Docente asistente Universidad Católica Popular del Risaralda

Docente Especialización en Gerencia Universidad Pontificia Bolivariana

eduardo.pelaez@ucpr.edu.co

Wilson Pinto Romero

Magíster en Ciencias de Computación

Especialista en Ingeniería del Software

Ingeniero de Sistemas

Docente asistente Universidad Católica Popular del Risaralda

wpinto@ucpr.edu.co

Recibido Mayo 29 de 2008 – Aceptado Julio 08 de 2008

RESUMEN

El presente artículo hace parte del proyecto de investigación: “Sistema para la divulgación de información a través del teléfono celular” y con él se pretende dar a conocer la forma de utilizar las técnicas de estimación de esfuerzo y costos en el desarrollo de software y el impacto que puede llegar a tener el uso de estas técnicas en la ejecución normal del

¹ Producot derivado del proyecto de investigación “Sistema para la divulgación de información a través del teléfono celular” avalado por el centro de investigaciones de la universidad Católica popular del Risaralda.

proyecto. El escrito se aborda con una introducción a la estimación de esfuerzo y costos de software; luego se exponen las razones que llevan al ingeniero a utilizar determinada técnica; en un tercer apartado se exponen algunas prácticas de estimación de uso común, llegando a la selección y aplicación de una determinada técnica en el software que hace parte del proyecto de investigación; se finaliza presentando algunas conclusiones que pueden servir de referencia a los equipos de desarrollo de software.

Palabras Clave: estimación, costos de software, métricas, Ingeniería de Software.

EVALUATION OF EFFORT AND COST IN THE DEVELOPMENT OF SOFTWARE FOR THE DIVULGATION OF INFORMATION THROUGH THE MOBILE PHONE

ABSTRACT

This article makes part of the research project: “system for the divulgation of information through the mobile phone” and it tries to make the readers know the ways to use the techniques of evaluation of effort and cost in the development of software, and the impact that the use of these techniques in the normal execution of the project may have. The report starts with an introduction to the evaluation of effort and the cost of software; then, it exposes the reasons why the engineer uses a determined technique; in a third part, some practices of evaluation of common use are presented, afterwards it introduces the selection and application of a determined technique in the software that makes part of the research project; finally some conclusions that can be useful as a reference to the software development teams are given.

Key Words: Evaluation Cost of Software, Metrics, Software Engineering.

1. INTRODUCCIÓN

En noviembre de 2007 el Centro de Investigaciones de la Universidad Católica Popular del Risaralda publicó la respuesta a la convocatoria de proyectos de investigación de esa vigencia, para la cual, el programa de Ingeniería de Sistemas y Telecomunicaciones de la misma institución, propuso un proyecto de desarrollo de software para dispositivos móviles, acompañado de todo el proceso de ingeniería del software y la medición del impacto de este en la solución del problema que se planteaba. Este proyecto fue aprobado y actualmente uno de los temas abordados como etapa de la ingeniería del software es la estimación de esfuerzo y costos del mismo.

No es normal encontrar, al menos en la región y según la experiencia, un software que cumpla con todas las normas de ingeniería que proponen las metodologías aceptadas mundialmente en esta área. Usualmente se encuentran proyectos software que parten de un simple análisis, un reducido diseño y llegan rápidamente a la etapa de implementación, resultado del afán por entregar un prototipo funcional al cliente. El proyecto de investigación propuesto, plantea una forma seria de llegar a un producto software y como parte de esta metodología se encuentra un gran valor en la estimación de esfuerzo y costos, de tal manera que se pueda cumplir, con mayor certeza, un cronograma estipulado desde inicios del proyecto y con un presupuesto más acorde con las necesidades del mismo.

En este artículo se desarrolla ese componente de los proyectos de desarrollo que causa gran dificultad a las personas o empresas constructoras de software – la estimación de esfuerzo y costos-, esperando sea de gran utilidad y abra una puerta de discusión entre la comunidad

académica en los aspectos que deben ser relevantes en la gestión de productos software con un enfoque ingenieril.

2. EL POR QUÉ DE LA ESTIMACIÓN DE COSTOS Y ESFUERZOS EN EL DESARROLLO DE SOFTWARE

Como se expresa en la parte introductoria, el proceso de estimación de costos y esfuerzo en un proyecto de desarrollo de software es de vital importancia para su exitosa culminación. Desafortunadamente este aspecto de la gerencia, en la gran mayoría de los proyectos, se ha relegado a un mero acto protocolario al comienzo de éstos. Como consecuencia, la planeación y seguimiento se realiza con poco rigor, desencadenando eventos como retraso en la entrega del producto e incremento del presupuesto, hechos considerados por muchos como algo natural.

En la actualidad las compañías dedicadas a prestar servicios de desarrollo de software se encuentran en una constante búsqueda de la calidad, implementando técnicas que les permitan obtener resultados óptimos en el desarrollo de las aplicaciones, esto incluye aspectos como la administración de los recursos humanos y presupuestales, los cuales deben ser tenidos en cuenta en las diferentes etapas del desarrollo, a fin de planear en forma general y también en forma detallada las actividades y el uso de recursos destinados para la realización del proyecto.

Por último, es importante mencionar que la precisión de las estimaciones contribuye a tener presentes aspectos como:

- realizar análisis de costo-beneficios y financieros
- realizar análisis de inversión de Hardware y Software
- proveer la base para la evaluación gerencial de múltiples proyectos
- servir de fundamento para los cronogramas, asignación de personal, gerencia de proyectos y definición de infraestructura
- evitar problemas como la renegociación de contratos, sobre tiempos y/o sobrecostos del proyecto.

3. ALGUNAS TÉCNICAS DE ESTIMACIÓN

Existen diversas técnicas que aportan a la realización de estimaciones de costo y esfuerzo para el desarrollo de productos software. A continuación se describen brevemente algunas de ellas.

3.1. TÉCNICA DE LÍNEAS DE CÓDIGO Y PUNTOS DE FUNCIÓN

Se fundamenta en la estimación del número de instrucciones de lenguaje de programación que tendrá el software, así como el número de funciones o métodos que lo conformarán (Pressman, 1998). Se puede documentar con exactitud sobre esta técnica consultando el texto Ingeniería de Software de Roger Pressman.

3.2. TECNICA COCOMO

Esta técnica se vale de la anterior para calcular el número de líneas de código, insumo con el cual desarrolla la técnica por medio de un conjunto de parámetros de costeo, entre los cuales

se encuentra el modo y el método del proyecto. Esta técnica se desarrolla en detalle en el apartado número 5 de este documento.

3.3. TECNICA COCOMO II

COCOMO II es un modelo que permite estimar el costo, el esfuerzo y el tiempo, cuando se planifica una nueva actividad de desarrollo software, y está asociado a los ciclos de vida modernos. Fue desarrollado a partir de COCOMO, incluyendo actualizaciones y nuevas extensiones más adecuadas a los requerimientos de los ingenieros de software (Boehm, 1999).

3.4. TECNICA COCOTS

Los modelos COCOTS (Constructive Commercial Off-The-Shelf) se utilizan en proyectos caracterizados principalmente por dos aspectos: su código fuente no está disponible para el desarrollador de la aplicación y su evolución no es supervisada por el mismo. El desarrollo de software utilizando COCOTS implica tres actividades, explicadas suficientemente por Álvarez (2000).

3.5. ESTIMACIÓN PARA PROYECTOS DE INGENIERÍA WEB

Los proyectos de ingeniería web adoptan normalmente el modelo de proceso ágil. Por este motivo, es frecuente utilizar una medición de puntos de función modificada en conjunto con los pasos de la estimación en proyectos ágiles (Mendoza, 1997).

Finalmente, es necesario aclarar que existen otras técnicas (Top Down, Bottom Up, Delphi, Desarrollo ágil, entre otras), pero las mencionadas anteriormente, hacen parte de las más usadas en los procesos de estimación en la actualidad.

4. CÓMO SE SELECCIONA UNA TÉCNICA DE ESTIMACIÓN

Cada técnica que se quiera utilizar, requiere de un número y de un modelo determinado de variables para su estimación. Una de las razones importantes al momento de elegir una técnica, es conocer la cantidad de variables que se solicitan, cuáles están disponibles y luego, para elegir la técnica, es necesario evaluar una serie de aspectos relacionados con el proyecto, tales como: el tipo de aplicación a desarrollar, el tamaño de la aplicación, la experiencia de los integrantes del proyecto en desarrollo, los recursos tecnológicos disponibles y el presupuesto entre otros.

Es importante recalcar además, que el proceso de estimación es iterativo, es decir, a medida que se avanza en el proyecto se debe hacer un seguimiento para verificar el cumplimiento de lo planeado de acuerdo con las estimaciones inicialmente realizadas. Esto permite que la dirección del proyecto ajuste su planeación con base en las nuevas estimaciones, y por tanto la reasignación de los recursos disponibles.

Las características del proyecto de investigación y las especificaciones del software llevan al grupo de investigadores y desarrolladores a inclinarse por el método estimación de esfuerzo y costos COCOMO, dando, como se mencionó anteriormente, las variables utilizadas por éste para los cálculos respectivos y su concordancia con la disponibilidad de las mismas en el proyecto.

5. DESCRIPCIÓN DE LA TÉCNICA SELECCIONADA - Método de estimación COCOMO

Este método se aplica al desarrollo que sigue el ciclo de vida en cascada, el cual consta de 9 etapas (Moreno, 2007): Planificación y Definición de Requisitos, Diseño de Producto, Diseño Detallado, Codificación y Pruebas, Integración y Pruebas, Implantación, Verificación y Validación y Gestión de Configuración.

De acuerdo con ciertas características del proyecto, el método de COCOMO define algunas ecuaciones para realizar las estimaciones. Se establecen para ello 3 Modos de desarrollo: orgánico, semilibre y rígido; la clasificación del proyecto en uno de ellos dependerá de las características de la aplicación y del entorno de desarrollo. Para cada uno de los modos podría aplicar uno de los 3 métodos de estimación: básico, intermedio y detallado.

- MODOS DE DESARROLLO DE SOFTWARE

Modo Orgánico: Esta categoría se emplea en proyectos cuyas características coinciden con las enunciadas a continuación.

- Desarrollo por un equipo relativamente pequeño
- El personal que participa es experimentado en sistemas similares al que posee la organización y conoce la importancia que tiene el proyecto para alcanzar los objetivos de la empresa.
- El personal, por su conocimiento y experiencia, puede contribuir al proyecto en las diferentes etapas sin generar mayor sobrecarga de comunicación.

- El entorno de desarrollo facilita la participación activa de los usuarios para que la aplicación cumpla con las especificaciones.
- Condiciones de desarrollo estables, donde los avances tecnológicos tengan un impacto mínimo en las especificaciones del proyecto.

Modo Semilibre: Los proyectos que cumplen con las condiciones relacionadas en este apartado se clasifican generalmente bajo esta modalidad.

Equipos de desarrollo integrados por personal que tiene un nivel medio de experiencia en proyectos similares, donde existe personal experto e inexperto, donde hay elementos con o sin experiencia en aspectos del proyecto y donde el tamaño del producto llega a las 300.000 instrucciones, se enmarcan en el modo semilibre en el proceso de desarrollo.

Modo Rígido. Existen proyectos que por su misma naturaleza tienen unos niveles de exigencia muy altos. A continuación se relacionan las características que permiten clasificar un proyecto de desarrollo en esta categoría:

- Limitaciones muy estrictas para el desarrollo del proyecto.
- El sistema debe desarrollarse bajo estrictas especificaciones de hardware y software, así mismo, a nivel normativo y de procedimientos operativos, tales como aplicaciones de transferencia electrónica de fondos o control de tráfico aéreo.
- Proyectos donde el cambio de algunas características implicarían altos costos, por lo que se consideran inmodificables, por tanto el sistema debe implementarse bajo especificaciones estrictas.

- Proyectos que no admiten cambios en los requisitos e interfaces de usuario, así el esfuerzo en validación y verificaciones es alto.
- Este tipo de proyectos se desarrollan en áreas desconocidas lo que conlleva a los equipos de trabajo –analistas- a una sobrecarga de comunicación durante el desarrollo.
- Culminada la fase de diseño, es necesario constituir un equipo grande de programación para realizar el diseño detallado, codificación y, pruebas. En esta etapa se presenta un mayor consumo de esfuerzo que en los otros niveles.

- MODELOS COCOMO

Los modelos expuestos a continuación son descritos por Moreno (2007), allí se presentan algunos lineamientos que permiten realizar la estimación de acuerdo con las características específicas del proyecto en desarrollo.

Modelo Básico: Se emplea en el desarrollo de aplicaciones pequeñas o medianas, normalmente por un grupo de personas de la misma compañía.

Las ecuaciones definidas en este modelo para la estimación de tiempos de desarrollo y esfuerzo para cada modo de desarrollo se presentan a continuación:

Modo Orgánico: Esfuerzo = $2,4 (KDSI)^{1,05}$
 Duración = $2,5(\text{Esfuerzo})^{0,38}$

Modo Semilibre: Esfuerzo = $3,0 (KDSI)^{1,12}$
 Duración = $2,5(\text{Esfuerzo})^{0,35}$

Modo Rígido: Esfuerzo = 3,6 (KDSI) ^{1,20}
Duración = 2,5(Esfuerzo) ^{0,32}

KDSI: Representa el número de instrucciones de código (K: Miles).

Es necesario mencionar la existencia de otros modelos (Intermedio, Avanzado), los cuales no se consideran en este documento por no ser relevantes en el contexto del proyecto en desarrollo.

6. APLICACIÓN DE LA TÉCNICA COCOMO PARA LA ESTIMACIÓN DE COSTO Y ESFUERZO DE UN SOFTWARE PARA LA DIVULGACIÓN DE INFORMACIÓN A TRAVÉS DEL TELÉFONO CELULAR

Aunque en la actualidad existen herramientas informáticas como CA-ESTIMACS, BYL (Before You Leap) o COSTAR, que facilitan la aplicación de una técnica de estimación de costos, a continuación y de manera procedimental, la forma de calcular el costo del software y la estimación de esfuerzos se aplicará de manera manual (Peláez, 2007).

- Método de estimación seleccionado: COCOMO, por las razones expuestas en el apartado número 4.
- Clasificación del proyecto según modo y modelo:
 - a. Modo SEMILIBRE, porque está caracterizado al interior del proyecto de investigación así:

- i. El equipo del proyecto de investigación tiene un nivel medio de experiencia en proyectos similares; esta integrado por personal experto a termino medio (Profesores que tienen experiencia en desarrollo de proyectos similares) y personal inexperto (Profesores y Estudiantes que no han participado de desarrollo de proyectos de este tipo)
 - ii. Algunos miembros del proyecto tienen experiencia en la temática tratada en el software y otros no.
 - iii. El proyecto software requiere interfaces poco rigurosas y no exige una plataforma rígida (grandes máquinas computacionales) especial.
 - iv. El tamaño del producto tiene, a lo sumo, 300 KDSI (Miles de instrucciones de código)
- b. Modelo BÁSICO, porque es el modelo que suele aplicarse a los desarrollos de software de tamaño pequeño o mediano y que serán probados y utilizados por personal de la propia organización donde se desarrolló el producto.
- o Determinar la ecuación utilizada para la estimación de esfuerzo y tiempo de desarrollo para el modo y modelo elegido. Para el caso que nos ocupa es:

$$MM = 3,0 * (KDSI)^{1,12}$$

$$TDEV = 2,5 * (MM)^{0,35}$$

Donde,

KDSI = número de instrucciones de código en miles

MM = esfuerzo medido en meses/hombre

TDEV = duración en meses

De acuerdo con esto:

1. Se ha estimado que el software componente de este proyecto tendrá alrededor de 5000 (5 KDSI) instrucciones, entonces calculamos esfuerzo, tiempo y recursos requeridos para desarrollar dicha aplicación así:

a. Esfuerzo: $MM = 3.0 * (5)^{1.12} = 20.75$ Meses/Hombre

b. Tiempo: $TDEV = 2.5 * (20.75)^{0.35} = 3.98$

c. Promedio de personas utilizadas para el proyecto: $PP = 20,75/3,98 = 5,21$ personas

d. Valor Mes/Hombre: $VMH = 800.000$ pesos colombianos

2. ¿Cuál sería el esfuerzo, tiempo y recursos requeridos para desarrollar el software?

a. Esfuerzo: 20,75 Meses/Hombre

b. Tiempo: 3,98 Meses

c. Costo del software: $PP * VMH * TDEV = 5,21 * 800.000 * 3,98 = 16,588,640$ pesos colombianos

7. CONCLUSIONES

Si bien existen muchos factores a la hora de estimar un proyecto software, su duración y costo cobran cada vez más importancia, sobre todo en nuestra región, cuando la industria informática ha logrado ser estigmatizada por presupuestos financieros que siempre se desfasan y plazos que pocas veces se cumplen.

Sin importar la variedad de técnicas de estimación existentes, la mayoría coinciden en las variables que tienen en cuenta para los cálculos finales.

Entre las técnicas de estimación de esfuerzo y costo se eligió COCOMO por la posibilidad de converger otras técnicas dentro de ella misma como el cálculo de líneas de código y puntos de función.

La consolidación de métricas de software ayuda a construir presupuestos mas ajustados a la realidad y cronogramas de actividades con mayores posibilidades de cumplimiento, pues éstas obedecen a fórmulas que son el resultado de múltiples desarrollos, que han sido probadas en proyectos de diferentes tamaños, obteniendo un margen de error muy bajo.

La reutilización del código juega un papel importante en la estimación del costo. Esto permite reducir la cantidad de instrucciones (KDSI) y por defecto bajar el número de personas requeridas para el desarrollo.

Para disminuir costos en el desarrollo de software es recomendable vincular al proyecto, personal con un perfil más técnico que profesional, con el acompañamiento de un experto que lidere el proceso.

En la actualidad existen herramientas software (utilitarios) que permiten facilitar el proceso de estimación para cualquier proyecto de este tipo y de cualquier tamaño.

El no tener antecedentes en la estimación de esfuerzo para software similar al que se está tratando en este proyecto, dificulta este proceso, haciendo que los resultados obtenidos sean algo inciertos.

BIBLIOGRAFÍA

ALVAREZ, J. (2008, abril 15). Ingeniería de Software, Técnicas de Estimación. Disponible en: http://sinbad2.ujaen.es/~mafer/drupal-5.1/files/pfc/pfc_jesus.pdf.

BOEHM, B. (2008, abril 25). *COCOMO II Model Definition Manual*. Disponible en: <http://sunset.usc.edu/research/COCOMOII>. USA, 1999.

LONDEIX, B. (1997). *Cost Estimation for Software Development*. Addison-Wesley Publishers Company.

MENDOZA, L. (2008, mayo 7). *Planificación de la Implementación, Técnicas de Estimación*. Disponible en: <http://prof.usb.ve/lmendoza/Documentos/PS-6116/Teoría PS6116 Planificación de la Implementación.pdf>.

MORENO, A. (2007). *Estimación de Proyectos Software. Centro de Ingeniería del Software e Ingeniería del Conocimiento. Buenos Aires (Argentina)*: Instituto Tecnológico de Buenos Aires.

PELÁEZ, L. (2007). *Estimación de Proyectos de Software, texto No. 4, control de unidad.* Maestría en Ingeniería del Software. Buenos Aires (Argentina): Instituto Tecnológico de Buenos Aires.

PRESSMAN, R. (1998). *Ingeniería del Software. Un Enfoque Práctico.* Cuarta Edición. 581 páginas. Editorial Mc Graw-Hill. ISBN 84-481-1186-9.

RUMBAUGH, J., BOOCH, G. (1999). *The Unified Modeling Language: Reference Manual.* Addison Wesley.

SYMONS, C. (1998). *Function Point Analysis Difficulties and Improvements.* *IEEE Transactions on Software Engineering.*