

Estudio Comparativo de Técnicas para la Estimación del Esfuerzo de las Iteraciones de Proyectos Software

José Antonio Pow-Sang Portillo

Pontificia Universidad Católica del Perú, Sección Ingeniería Informática
Av. Universitaria cda 18, Lima 32, Perú
jpow@pucp.edu.pe

Resumen. El presente artículo muestra un estudio comparativo de tres técnicas que se podrían emplear para la estimación del esfuerzo de cada una de las iteraciones de la construcción de software. Las técnicas se basan en Puntos de Función y COCOMO II. Una de estas técnicas ha sido probada en proyectos desarrollados por una persona con resultados satisfactorios. Este estudio utiliza los resultados obtenidos en dos proyectos, uno formado por un equipo de 10 y otro de 11 alumnos de pre-grado. Las técnicas se han aplicado al esfuerzo real utilizado por cada uno de los equipos de desarrollo (esfuerzo en horas-hombre) para determinar cuál de ellas pudo haber sido la más adecuada para realizar la estimación del esfuerzo de cada una de las iteraciones.

1 Introducción

En las últimas décadas se han desarrollado algunas técnicas para la estimación del esfuerzo de proyectos software completos, tales como Puntos de Función [1][7] y COCOMO II [2]. Aún así, estas técnicas fueron concebidas para su aplicación en sistemas basados en el paradigma estructurado con un ciclo de vida clásico o en cascada de Royce [6], y aún es difícil emplearlas en desarrollos orientados a objetos y ciclos de vida iterativo-incrementales, tan en boga en los últimos años.

Teniendo en cuenta la problemática planteada, el presente artículo muestra un estudio realizado a dos equipos de desarrollo de 10 y 11 personas que han utilizado un modelo iterativo-incremental para el desarrollo de un sistema software. De acuerdo a estos resultados, se puede concluir que técnica es la más adecuada para realizar la estimación del esfuerzo para cada iteración.

Este documento se ha estructurado de la siguiente manera: la sección 2, muestra las tres técnicas motivo del estudio; la sección 3, muestra las características del proyecto; la sección 4, presenta los resultados obtenidos al aplicar dichas técnicas; y, finalmente, se presentan las conclusiones obtenidas de este trabajo.

2. Técnicas para Estimar el Esfuerzo de las Iteraciones de Construcción

La principal dificultad para estimar el esfuerzo necesario para el proyecto se encuentra en el cálculo de puntos de función sin ajustar (PFsA) [7] para cada una de las iteraciones. Para ello, se han propuesto, para este estudio, el empleo de tres técnicas las cuales se diferencian en el cálculo de los PFsA referentes a los ficheros lógicos internos (ILF) y ficheros de interfaz externa (EIF).

La primera técnica es propuesta en [4] y consiste en determinar los PFsA correspondientes a los ILFs y EIFs para cada caso de uso [3], calculando una proporción de uso entre todos los casos de uso a los que les hacen referencia.

La segunda técnica es una variación de la primera y corresponde a la hipótesis en las que sólo se debe calcular el proporcional de uso de los ILFs sólo entre las Entradas Externas (EI) que les hacen referencia, ya que es en este tipo de transacciones en las que se deben crear las estructuras de la base de datos correspondiente a esos ILFs. En las otras transacciones (EO, EQ), básicamente, sólo se realizan consultas de la información almacenada y cuya estructura ya ha sido construida.

La tercera técnica se relaciona a la hipótesis en la que no es necesario considerar los PFsA de los ficheros (ILF, EIF) y sólo se deben utilizar los PFsA relacionados a las transacciones.

Es importante resaltar que para este estudio se planificaron las iteraciones en base al diagrama de precedencias de casos de uso propuesto en [4].

3. Características del Proyecto Materia del Estudio

El proyecto de software se realizó dentro de una asignatura del cuarto año de la carrera de Ingeniería Informática. La duración total fue de un semestre académico (14 semanas); la metodología de desarrollo, RUP [5]; el proyecto, un sistema para una cadena de restaurantes tipo “fast food”; el tiempo por iteración en construcción, 2 a 3 semanas; la herramienta de desarrollo, Delphi 5.0; el manejador de base de datos, Microsoft SQL Server 2000 y el estilo arquitectónico, cliente/servidor de dos capas.

En cuanto a los conocimientos de los alumnos, en semestre anteriores aprendieron los lenguajes de programación Java, C#, Pascal y C. Además, dominan técnicas de análisis y diseño orientado a objetos y estructurado.

Cabe resaltar, que a los estudiantes no se les explicó ninguna de las técnicas mostradas en la sección anterior (sección 2 de este artículo) para evitar algún posible sesgo en los tiempos utilizados para el desarrollo del software. Sólo se les indicó que se repartieran el trabajo en base a los puntos de función sin ajustar correspondientes a las transacciones (EI,EO,EQ).

4. Resultados Obtenidos

Según lo conversado con los equipos luego de finalizado el proyecto y utilizando los drivers de coste de COCOMO II [2], se pudo determinar el factor de ajuste EAF de cada iteración, los cuales se muestran en la tabla 1. El factor EAF es utilizado para ajustar el esfuerzo en horas-hombre; es decir que este valor es proporcional al esfuerzo requerido en una iteración.

Tabla 1. Factor de ajuste “EAF” de COCOMO II por iteración y equipo.

Iteración de Construcción	Factor “EAF” del Equipo 1	Factor “EAF” del Equipo 2
1	1.46	1.20
2	0.80	1.20
3	0.80	0.80

Debido a que los integrantes de los equipos habían trabajado de manera conjunta en proyectos de asignaturas de semestres anteriores, el contexto correspondiente a las relaciones interpersonales fue la misma en todas las iteraciones. Es por ello que todos los drivers de coste se tomaron con valor “nominal”, a excepción de los drivers AEXP y LTEX que se relacionan a la experiencia de desarrollo de aplicaciones, de la herramienta y del lenguaje de programación.

La tabla 2 resume el resultado del equipo 1 (en este artículo no se incluye la tabla con los resultados del equipo 2). La primera columna de la tabla muestra el número de iteración; la segunda, el esfuerzo real utilizado en horas-hombre; la tercera, cuarta y quinta, los puntos de función sin ajustar por técnica; y la sexta, séptima y octava, el esfuerzo real en horas hombre por punto de función sin ajustar de cada una de las técnicas.

Tabla 2. Resultados para el equipo 1

Iter. Const.	Esfuerzo Real (horas)	TPFsA_1 (Téc. 1)	TPFsA_2 (Téc. 2)	TPFsA_3 (Téc. 3)	Esf_Real por PfsA (Téc. 1)	Esf_Real por PfsA (Téc. 2)	Esf_Real por PfsA (Téc. 3)
1	465	114.05	116.50	78.00	4.08	3.99	5.96
2	715	282.32	290.25	187.00	2.53	2.46	3.82
3	334	129.63	119.25	72.00	2.58	2.80	4.64

De acuerdo a la tabla 1, en los resultados para el grupo 1 se debería haber obtenido el mismo esfuerzo real por PfsA en la segunda y tercera iteración. Esto no se observa en ninguna de las tres técnicas en la tabla 2. Sin embargo, la técnica 1 es la que obtiene menor diferencia entre ambas iteraciones (sólo 0.01), en cambio las técnicas 2 y 3 muestran mucha diferencia (más de 0.3). Por tanto, para el grupo 1, podríamos afirmar que la técnica 1 da una mejor precisión para el cálculo del esfuerzo de cada una de las iteraciones.

En cuanto al grupo 2 y de acuerdo a la tabla 1, se debería haber obtenido el mismo esfuerzo real por PfsA en la primera y segunda iteración. Esto no se observó en ninguna de las tres técnicas. No obstante, la técnica 1 es la que tiene menor diferencia entre ambas iteraciones (sólo 0.02), en cambio las técnicas 2 y 3 muestran mucha diferencia, incluso, para estas dos últimas, la segunda iteración necesita de un mayor esfuerzo por PfsA que en la primera. En consecuencia, para el grupo 2, podríamos afirmar que la técnica 1 da una mejor precisión para el cálculo del esfuerzo de cada una de las iteraciones.

La tabla 3 muestra el coeficiente de correlación lineal entre el factor EAF mostrados en la tabla 1 y el esfuerzo x PfsA de las técnicas 1, 2 y 3 de ambos equipos. En

esta tabla se puede observar que, para los dos grupos, la correlación entre el factor EAF y el esfuerzo por PFsA es mayor para la primera de las tres técnicas.

Tabla 3. Coeficiente de correlación de regresión lineal de EAF vs Esfuerzo por PFsA

Equipo	Coeficiente Correlación	Coeficiente Correlación	Coeficiente Correlación
	Técnica 1	Técnica 2	Técnica 3
1	0.9997	0.9777	0.9259
2	0.9998	0.5461	0.9969

En base a los resultados mostrados en las tablas anteriores, se puede observar que la técnica 1 ofrece una mayor precisión para determinar el esfuerzo x PFsA para cada iteración que las otras técnicas. Por consiguiente, se puede afirmar que la técnica 1 es la más adecuada de las tres para realizar las estimaciones de esfuerzo de las iteraciones de proyectos software.

5. Conclusiones y Trabajo Futuro

La mayoría de las aproximaciones actuales para estimar proyectos, aún definiéndose como independientes de la tecnología y modelos de ciclo de vida, tienen un carácter fuertemente influido por los ciclos de vida en cascada. A pesar de que son válidas para otras aproximaciones, como los ciclos de vida iterativos-incrementales, por ejemplo, en general no ofrecen ninguna guía para acometer dicha adaptación.

El presente trabajo muestra que la propuesta dada en [4] se podría utilizar para proyectos con ciclos de vida iterativos-incrementales y con casos de uso, en equipos de desarrollo en los que participan más de una persona.

En un futuro, una vez que la técnica esté estabilizada, será aplicada a nuevos desarrollos de características más amplias, como equipos de desarrollo más numerosos o aplicaciones más complejas. También, se determinará la viabilidad de modificar la técnica de puntos de función de manera que se puedan reemplazar los ficheros lógicos internos y ficheros de interfaz externa por clases y diagramas de clases, para que dicha técnica pueda adaptarse completamente al enfoque orientado a objetos.

Bibliografía

1. Albrecht, A. J. *Measuring Application Development Productivity*, IBM Applications Development Symposium, Monterey, CA, USA, 1979.
2. Boehm, B., *COCOMO II Model Definition Manual*, <http://sunset.usc.edu/research/COCOMOII>, USA, 1999.
3. Jacobson, I., *Object-Oriented Software Engineering, A Use Case Driven Approach*, Addison-Wesley, USA, 1992.
4. Pow-Sang, J., Imbert R., Estimación y Planificación de Proyectos Software con Ciclo de Vida Iterativo-Incremental y empleo de Casos de Uso, Proceedings IDEAS 2004, ISBN 9972-9876-1-2, Arequipa-Perú, 2004.
5. Rational Software, *Rational Unified Process version 2001A.04.00.13*, USA, 2001.
6. Royce, W. W., *Managing the Development of Large Software Systems: Concepts and Techniques*. Proceedings WESCON, 1970.
7. The International Function Point User Group (IFPUG), *Function Point Counting Practices Manual-Release 4.1*, USA, 1999.