

Revista
Española de
Innovación,
Calidad e
Ingeniería del Software



Volumen 4, Número 2 (especial X JICS), septiembre, 2008

Web de la editorial: www.ati.es/reicis

E-mail: editor-reicis@ati.es

ISSN: 1885-4486

Copyright © ATI, 2008

Ninguna parte de esta publicación puede ser reproducida, almacenada, o transmitida por ningún medio (incluyendo medios electrónicos, mecánicos, fotocopias, grabaciones o cualquier otra) para su uso o difusión públicos sin permiso previo escrito de la editorial. Uso privado autorizado sin restricciones.

Publicado por la Asociación de Técnicos de Informática

www.ati.es



Revista Española de Innovación, Calidad e Ingeniería del Software (REICIS)

Editores

Dr. D. Luís Fernández Sanz

Departamento de Ciencias de la Computación, Universidad de Alcalá

Dr. D. Juan José Cuadrado-Gallego

Departamento de Ciencias de la Computación, Universidad de Alcalá

Miembros del Consejo Editorial

Dr. Dña. Idoia Alarcón

Depto. de Informática
Universidad Autónoma de Madrid

Dr. D. José Antonio Calvo-Manzano

Depto. de Leng y Sist. Inf. e Ing. Software
Universidad Politécnica de Madrid

Dra. Dña. Tanja Vos

Instituto Tecnológico de Informática
Universidad Politécnica de Valencia

D. Raynald Korchia

SOGETI

D. Rafael Fernández Calvo

ATI

Dr. D. Oscar Pastor

Depto. de Sist. Informáticos y Computación
Universidad Politécnica de Valencia

Dra. Dña. María Moreno

Depto. de Informática
Universidad de Salamanca

Dr. D. Javier Aroba

Depto de Ing.El. de Sist. Inf. y Automática
Universidad de Huelva

D. Antonio Rodríguez

Telelogic

Dr. D. Pablo Javier Tuya

Depto. de Informática
Universidad de Oviedo

Dra. Dña. Antonia Mas

Depto. de Informática
Universitat de les Illes Balears

Dr. D. José Ramón Hilera

Depto. de Ciencias de la Computación
Universidad de Alcalá

Contenidos

REICIS

Editorial	4
<i>Luís Fernández-Sanz, Juan J. Cuadrado-Gallego</i>	
Presentación	5
<i>Luis Fernández-Sanz</i>	
Hacia la gestión cuantitativa en la gestión de proyectos en el ámbito de las pymes	7
<i>Jose A. Calvo-Manzano, Iván García y Magdalena Arcilla</i>	
Problemas de las pymes en el nivel 2 de madurez. Una muestra sesgada	20
<i>Juan José Cukier</i>	
Mejora de procesos organizativos: análisis estadístico	33
<i>Izaskun Santamaria, Teodora Bozheva, Iñaki Martínez de Marigorta</i>	
Revisiones de código en el contexto del aseguramiento de calidad. Un caso práctico	46
<i>María José Escalona, Manuel Pérez-Pérez, O. González-Barroso, J. Ponce, J. M. Correa, A. I. Merino</i>	
Diagnóstico de la situación de la calidad del software en la industria española	58
<i>Elena Argüelles, Antonio Sepúlveda</i>	
ACCESIBILIDAD WEB: un vistazo a tres webs de administraciones públicas en España	70
<i>Jorge Sánchez, Tanja E.J. Vos</i>	
Infraestructura de pruebas para una plataforma de inteligencia de negocios: lecciones aprendidas de una experiencia académica	82
<i>Ruth Alarcón, Carla Basurto, Abraham Dávila</i>	
Perfiles del ciclo de vida del software para pequeñas empresas: los informes técnicos ISO/IEC 29110	96
<i>José A. Calvo-Manzano, Javier Garzás, Mario Piattini, Francisco J. Pino, Jesús Salillas, José Luis Sánchez</i>	
Estudio experimental de la conversión entre las unidades de medición funcional del software puntos de casos de uso e IFPUG	109
<i>Juan J. Cuadrado-Gallego, María J. Domínguez-Alda, Marian Fernández de Sevilla, Miguel Ángel Lara</i>	

Making Software Process Management Agile	122
<i>José Manuel García, José Javier Berrocal, Juan Manuel Murillo</i>	
La norma ISO/IEC 25000 y el proyecto KEMIS para su automatización con software libre	135
<i>José Marcos, Alicia Arroyo, Javier Garzás y Mario Piattini</i>	
Modelo de calidad para herramientas FLOSS que dan apoyo al modelado de procesos del negocio	148
<i>Leslibeth Pessagno, Kenyer Domínguez, Lornel Rivas, María Pérez, Luis E. Mendoza, Edumilis Méndez</i>	

Editorial

The logo for REICIS (Revista Española de Innovación, Calidad e Ingeniería del Software) is displayed in white, bold, uppercase letters within a black rectangular box.

El grupo de Calidad del Software de ATI ha consolidado su posición como principal promotor de la disciplina de ingeniería y calidad del software con la décima edición de las Jornadas sobre Innovación y Calidad del Software (las tradicionales JICS). Estas X JICS pretenden además potenciar la presencia iberoamericana en este foro de promoción de la cultura de la calidad del software y de la innovación en el desarrollo de sistemas y aplicaciones por lo que constituyen la promoción de una I Conferencia Iberoamericana de Calidad del Software (CICS). Por otra parte, las X JICS incorporan la presencia de la ponencia de un destacado experto europeo en la disciplina de ingeniería de software como es Darren Dalcher, Director del UK National Centre for Project Management en la Middlesex University y editor de la revista Software Process Improvement and Practice.

Por otra parte, queremos resaltar la línea de calidad de los trabajos, eminentemente prácticos pero rigurosos, aceptados entre los remitidos en la convocatoria de contribuciones: las ponencias aceptadas (con una tasa de rechazo del 40%) han sido sometidos a un completo proceso de revisión por el comité de programa así como a una cuidadosa labor de revisión de estilo, de terminología y de ortotipografía para garantizar el mejor resultado para nuestros lectores. Por supuesto, no cabe olvidar el apoyo de los patrocinadores (Telelogic, Steria, Deiser, GESEIN y SOGETI) no sólo aportando recursos sino también interesantes presentaciones de experiencias prácticas de sus expertos. Los debates promovidos en las mesas redondas así como la promoción de las actividades de comunicación y *networking* entre los participantes, tanto a nivel presencial como a través de la lista de distribución, los medios electrónicos y la nueva oferta formativa con plataforma *e-learning*. En definitiva, el evento más completo con toda la información disponible en la página del grupo de Calidad del Software (www.ati.es/gtcalidadsoft) acorde a la trayectoria pionera en España que, desde 1997, está proporcionando, a través de la Asociación de Técnicos de Informática, el apoyo para la productividad y la calidad en los proyectos de software. Este perfil ha sido reconocido por el apoyo del Ministerio de Industria, Turismo y Comercio con su apoyo institucional dentro de la convocatoria de la orden ITC/390/2007. Por último, debemos resaltar la aportación de datos de gran importancia no sólo mediante los eventos organizados sino también a través de la realización de estudios específicos (por ejemplo, sobre las prácticas de pruebas, el diseño de casos y los factores que dificultan su implantación eficiente y eficaz en las organizaciones) que permiten un mejor conocimiento de la práctica real de esta disciplina en España.

Luis Fernández Sanz
Juan J. Cuadrado-Gallego
Editores

En este número especial de septiembre de 2008 de REICIS, por primera vez en la historia de nuestra revista, esta publicación se convierte en el vehículo de difusión del evento decano en España en el ámbito de la ingeniería y la calidad del software: las Jornadas de Innovación y Calidad del Software (JICS) que alcanzan así su décima edición desde su inicio en 1998. En esta ocasión, el Grupo de Calidad del Software de ATI (www.ati.es/gtcalidadsoft) no sólo ha querido cumplir con esta decena de ediciones sino que ha apostado por una apertura a nuevos retos como la presencia de eminentes ponentes invitados de gran presencia internacional y la potenciación de los vínculos iberoamericanos para convertir a este evento en la referencia sobre calidad del software en la amplia comunidad latina. Los trabajos aceptados han sido sometidos a un completo proceso de revisión por el comité de programa así como a una cuidadosa labor de revisión de estilo, terminología y ortotipografía para garantizar la mejor calidad para nuestros lectores. Este número especial constituye en definitiva la publicación de las actas de las X JICS y, por ello, cuenta con un tamaño mayor del habitual. Esperamos repetir este número especial el próximo año con la undécima edición de las Jornadas de Innovación y Calidad del Software. Agradecemos la labor del comité de programa coordinado por la Dr. M. Idoia Alarcón (Universidad Autónoma de Madrid) y compuesto por la siguiente lista de expertos:

- Antonia Mas (Universitat de les Illes Balears)
- Luis de Salvador (AGPD)
- Ricardo Vargas (Universidad del Valle de Méjico)
- Javier Tuya (Universidad de Oviedo)
- Antonio de Amescua (Universidad Carlos III de Madrid)
- María Moreno (Universidad de Salamanca)
- José Antonio Calvo-Manzano (Universidad Politécnica de Madrid)
- José Antonio Gutiérrez de Mesa (Universidad de Alcalá)
- Isabel Ramos (Universidad de Sevilla)
- Esperança Amengual (Universitat de les Illes Balears)
- José Ramón Hilera (Universidad de Alcalá)
- Mercedes Ruiz (Universidad de Cádiz)
- María Teresa Villalba (Universidad Europea de Madrid)
- Adolfo Vázquez (INSA)
- María José Escalona (Universidad de Sevilla)
- Ana Araújo (Ministerio de Medio Ambiente)
- Antonio Rodríguez (Telelogic)
- Gurutze Miguel (TQS)
- Beatriz Pérez (Centro de Ensayos de Software, Uruguay)
- José Javier Martínez (Universidad de Alcalá)
- José Díaz (SSQTB)

Luis Fernández Sanz

Estudio experimental de la conversión entre las unidades de medición funcional del software puntos de casos de uso e IFPUG

Juan J. Cuadrado-Gallego, María J. Domínguez-Alda,
Marian Fernández de Sevilla, Miguel Ángel Lara
Departamento de Ciencias de la Computación. Universidad de Alcalá
{jjcg, mariajose.dominguez, marian.fernandez}@uah.es

Abstract

The objective of this research is to find a mathematical equation that allows convert software functional size measured with IFPUG unit into Use Cases Points unit. A database built explicitly for this research is used to obtain the equation. Both a lineal and no lineal mathematical equations have been founded, in order to obtain the most accuracy results. Another goal for this research has been to develop a process to obtain this kind of data in an academic environment.

Key words: software engineering, software measurement, IFPUG, Use Case Points.

Resumen

El objetivo de esta investigación es la búsqueda de una función matemática que permita convertir mediciones del tamaño funcional del software realizadas en unidades IFPUG a unidades puntos de casos de uso. Para ello se parte de un conjunto de datos obtenidos en esta investigación. La búsqueda de la función matemática se ha realizado tratando tanto de obtener una ecuación lineal como de buscar ecuaciones no lineales que puedan dar una solución más correcta y precisa al problema. Como parte de esta investigación también se ha abordado el problema de la obtención de datos para este tipo de estudios, proponiéndose un procedimiento repetible y contrastado para la obtención de datos fiables en un entorno académico.

Palabras clave: ingeniería del software, medición del software, IFPUG, puntos de casos de uso.

1. Introducción

La medición del tamaño del software que va a ser desarrollado en un proyecto es una de las magnitudes más importantes para la correcta gestión del mismo. Las primeras unidades que se definieron para medir el tamaño del software fueron las líneas de código fuente. Aunque se trata de una unidad útil cuando se utiliza para analizar diferentes aspectos, como el índice de errores o de productividad de un equipo, presenta el problema de que solo se

puede aplicar una vez que se ha construido el software. Por esta razón, la definición de una magnitud que permitiese medir el tamaño del software antes de construirlo se convirtió en una tarea esencial.

La respuesta a este problema la dio el investigador de IBM Alan Albrecht [1], que definió una nueva unidad de medida: el tamaño funcional del software, a la que denominó puntos de función. Esta unidad se aplica sobre los documentos que sí existen en las fases iniciales de un proyecto, como son la especificación de requisitos y los modelos de análisis, para obtener una medida de la cantidad de funcionalidad que el software entregará al usuario. Albrecht estableció —y posteriormente se comprobó de forma experimental [2]— que dicha medida estaba directamente relacionada con el número de líneas de código que tendría el software una vez realizado y, por lo tanto, que era una unidad correcta de medida del tamaño del software que, además, se podía obtener al comienzo del proyecto.

El método de casos de uso permite documentar los requisitos de un sistema en términos de actores y casos de uso, proporcionando uno o más escenarios que indican cómo debería interactuar el sistema con el usuario o con otro sistema para conseguir un objetivo específico [3]. Debido a la importancia de esta técnica en el análisis orientado a objetos para capturar y describir los requerimientos funcionales de un sistema, surgió la necesidad de desarrollar un método de cálculo del tamaño funcional específicamente diseñado para esa técnica. Uno de los primeros resultados lo obtuvo en 1993 Gustav Karner [4], que desarrolló el método de puntos de casos de uso, como una extensión de los puntos de función IFPUG.

Este método cuantifica las características del sistema, tanto funcionales como no funcionales, a través de los siguientes pasos:

1. Categorización de los actores del modelo de casos de uso:

- Simple. Sistema externo que interactúa a través de una interfaz de programación definida y conocida (API) (Factor 1).
- Promedio. Sistema externo que interactúa a través de un protocolo (conjunto de reglas que especifican el intercambio de datos u órdenes durante la comunicación entre las entidades que forman parte de una red), como TCP/IP (Factor 2).
- Complejo. Usuario físico que interactúa a través de una interfaz gráfica de usuario (GUI) o sitio web (Factor 3).

Esta categorización da como resultado el número total de UAW (Unadjusted Actor Weight), contando el número de actores existentes en cada categoría, multiplicando cada resultado por el factor de ajuste correspondiente y realizando la suma de los resultados.

2. Categorización de los casos de uso:

- Simple. Tres transacciones o menos (Factor 5).
- Promedio. Entre cuatro y siete transacciones (Factor 10).
- Complejo. Más de siete transacciones (Factor 15).

Las transacciones son un grupo de actividades que se ejecutan de forma completa (éxito) o bien se vuelve al estado previo a la ejecución de la transacción (fracaso), quedando siempre el sistema en un estado consistente.

La categorización de los casos de uso permite conocer el valor del UUCW (Unadjusted Use Case Weights) de manera análoga a como se realizaba en el punto anterior, mediante el conteo del número de casos de uso en cada categoría, multiplicando estos por el factor de ajuste correspondiente y realizando la suma de los resultados.

Utilizando los resultados obtenidos en los dos pasos anteriores, se obtiene el número total de UUCP (Unadjusted Use Case Points) o puntos de casos de uso desajustados, mediante la fórmula:

$$UUCP = UAW + UUCW$$

3. Ajuste del valor UUCP

Mediante el uso de diversos factores técnicos y ambientales (no funcionales), a los cuales se asigna un valor comprendido entre cero y cinco, dependiendo de su influencia en el desarrollo del proyecto, se obtiene el valor TCF (Technical Complexity Factor) y EF (Environmental Factor), mediante las formulas:

$$TCF = 0,6 + (0,01 * TFactor)$$

$$EF = 1,4 + (-0,03 * EFactor)$$

TFactor y EFactor se calculan multiplicando el valor de cada uno de los factores de cada tabla por su peso y obteniendo la suma total.

Finalmente, mediante el uso de estos factores de ajuste TCF y EF, se obtiene el valor ajustado de puntos de casos de uso UCP (Use Case Points):

$$UCP = UUCP * TCF * EF$$

Uno de los problemas más importantes que actualmente queda por resolver en el campo de la medición del software es la conversión de unidades de medición funcional del software. El origen del problema reside en que existen muchas industrias en bastantes países que desde hace tiempo utilizan IFPUG como unidad de medida del software, habiendo desarrollado extensas bases de datos que les sirven para realizar correctamente las planificaciones de proyectos futuros. En la generación de esas bases de datos, para que los resultados obtenidos sean fiables, se necesita un gran número de años; por eso, la única solución realista que permitiría la introducción de otros métodos sería encontrar un factor de conversión de las medidas realizadas con IFPUG a puntos de casos de uso, de tal manera que se pudieran convertir automáticamente.

2. Obtención de los datos

2.1. Problemas en la obtención de los datos

Uno de los principales problemas de este tipo de investigaciones radica en la obtención de conjuntos de datos que, tanto en calidad como en tamaño, permitan hacer un análisis fiable y riguroso sobre ellos. El origen del problema se encuentra en el coste de la obtención de los datos, ya que los honorarios de los especialistas en esta materia son elevados (llegan hasta los 200 €/hora). Así, las empresas que han seleccionado una unidad específica para realizar las mediciones no ven fácilmente el retorno de la inversión que obtendrían si volvieran a realizar las mediciones sobre el mismo software con una unidad distinta. La consecuencia de todo esto es que la obtención de los datos procedentes de un entorno industrial es prácticamente imposible.

Para resolver este problema existen dos soluciones: o bien los investigadores, procedentes de otra institución —normalmente una universidad o centro de investigación— se encargan de medir con distintas unidades un conjunto de aplicaciones cedidas por la empresa en virtud de acuerdos suscritos por ambas organizaciones; o bien se pide a un conjunto de estudiantes que han finalizado sus estudios de un curso sobre estas unidades que, como proyecto final, realicen las mediciones necesarias con las distintas unidades estudiadas. En ambos casos el problema del coste de las mediciones se reduce muy significativamente.

Un segundo problema radica en la calidad de los datos obtenidos en las mediciones, cuyas consecuencias son dos: la realización de mediciones incorrectas, a causa de una formación inadecuada de los medidores, y la construcción de una base de datos heterogénea, debida a la inclusión en ella de mediciones realizadas por distintos medidores. Sobre este último aspecto no se ha publicado aún ningún trabajo que cuantifique el impacto que un determinado medidor tiene en la medida, si bien ya se están haciendo trabajos de investigación sobre este aspecto —entre ellos, uno por parte del equipo de Cuadrado-Gallego— que permiten llegar a unas conclusiones preliminares, al menos cualitativas, que establecen que existe un impacto en la medida imputable al medidor.

La solución a este segundo problema descansa en la participación de medidores expertos que, en el primer caso, realicen las mediciones y, en el segundo caso, revisen y hagan homogéneas las mediciones previas procedentes de varios medidores.

2.2. Metodología de obtención de los datos en la investigación actual

El procedimiento utilizado en este trabajo para la obtención de los datos ha sido diferente al usado en los trabajos previos. La forma de abordar la solución de los dos problemas descritos en el punto primero fue la siguiente: para la cuestión del coste de obtención de los datos se adoptó la segunda solución posible, de modo que estos se obtuvieron a través de los trabajos finales de estudiantes sobre medición, utilizando IFPUG y puntos de casos de uso de aplicaciones reales. Esta opción introducía el segundo problema, el de la calidad, tanto a través de la corrección de los datos —ya que los medidores no eran expertos—, como a través de la heterogeneidad, pues cada dato iba a ser obtenido por un medidor distinto.

El planteamiento para hacer frente a estos dos problemas tuvo una doble vertiente: en primer lugar, una rigurosa selección de los medidores entre los estudiantes que demostrasen un mayor conocimiento de la materia; en segundo lugar, la solución propuesta en el apartado primero, consistente en la revisión de todos los trabajos por parte de un medidor experto. A continuación se describen en detalle los pasos seguidos:

Enseñanza del proceso de puntos de casos de uso

Se dirige a estudiantes de la titulación de Ingeniería Técnica en Informática de Gestión, dentro de la asignatura obligatoria de tres créditos, de tercer curso, Laboratorio de Planificación y Gestión de Sistemas. Se imparte un total de 24 horas de clases teóricas

sobre los métodos IFPUG 4.1 y puntos de casos de uso, ocho horas de cada uno de los métodos, repartidas en clases de dos horas de duración. En dos grupos se impartió primero IFPUG y a continuación puntos de casos de uso; y en otros dos grupos se comenzó por puntos de casos de uso y posteriormente se trató IFPUG, con el objetivo de evitar los efectos que pudiera tener aprender antes unas unidades u otras. Las versiones elegidas de las unidades fueron IFPUG 4.1 y, en cuanto a puntos de casos de uso, la traducción al castellano del método de puntos de casos de uso de Gustav Karner.

Selección de los alumnos participantes en las mediciones

Para poder participar en la realización de las medidas, los estudiantes debían cumplir dos requisitos:

- Asistir al 90% de las clases; esto es, se podía faltar a una sola clase.
- Tener una nota superior a 8 sobre 10 en una prueba escrita de conocimientos, realizada en la undécima semana de clase (hicieron esta prueba todos los alumnos, tanto si iban a participar en las mediciones como si no lo iban a hacer).

Medición de una aplicación real con las unidades de puntos de casos de uso

A cada alumno seleccionado se le asignan las especificaciones de una aplicación real, distinta para cada uno, medida en cursos anteriores de la asignatura mediante el método IFPUG y que hubiera sido desarrollada previamente por la Universidad de Alcalá en la realización de algún proyecto de colaboración con una empresa y respecto a la cual la Universidad tuviera o compartiera la propiedad intelectual.

Cada alumno debe proceder a la medición de la aplicación asignada con las unidades de medida.

Corrección y uniformización de las mediciones hechas por los estudiantes

Un medidor experto revisa las mediciones de puntos de casos de uso realizadas por los estudiantes. Las tareas de revisión consistieron en: comprobar la existencia de errores en la identificación de los actores existentes en el sistema y en la identificación de los casos de uso; comprobar la existencia de errores en la identificación de las transacciones de los casos de uso; comprobar la existencia de errores en el cómputo de los puntos de actores, transacciones y puntos de casos de uso desajustados; comprobar la existencia de errores en la obtención de los factores de peso, de complejidad técnica y ambiental, y en el cómputo de puntos de casos de uso.

Los pasos descritos se aplicaron durante un curso, obteniéndose los siguientes resultados:

1. Generación de la muestra de puntos de casos de uso *m1a08*.

- Se cuenta con un total de 74 alumnos, repartidos en cuatro grupos, tres de ellos tienen 18 alumnos por clase y uno tiene 20 alumnos.
- El número final de alumnos que cumplieron la norma de 90% de asistencia a clase fue de 63.
- De los 63 alumnos aptos para realizar las mediciones, superaron la prueba 56, con la siguiente distribución de notas: 16-aprobado, 23-notable, 11-notable con nota igual o superior a 8 y 6-sobresaliente.

Por lo tanto, se seleccionó a 17 alumnos para realizar las medidas.

2. Generación de la muestra IFPUG *jjcg06*.

- Se parte de un total de 83 alumnos, repartidos en cuatro grupos, tres de ellos de 20 alumnos por clase y uno de 23 alumnos.
- El número final de alumnos que cumplieron la norma de 90% de asistencia a clase fue de 71.
- De los 71 alumnos aptos para realizar las mediciones, superaron la prueba 63, con la siguiente distribución de notas: 18-aprobado, 34-notable, 10-notable con nota igual o superior a 8 y 11-sobresaliente.

Por lo tanto, se seleccionó para realizar las medidas a 21 alumnos.

3. Generación de la muestra de pruebas *jjcg06-m1a08*.

Mediante la combinación de las muestras *jjcg06* y *m1a08* se obtuvo una tercera muestra *jjcg06-m1a08*, combinatoria de los métodos IFPUG y puntos de casos de uso, dado que los proyectos utilizados en la medición IFPUG efectuada en *jjcg06* fueron los usados para medir los puntos de casos de uso de la muestra *m1a08*.

El campo de aplicación de *jjcg06-m1a08* (C) se obtuvo mediante discriminación de un número de proyectos igual a la diferencia de muestras en *jjcg06* (P) y el número de alumnos seleccionados para realizar las mediciones en *m1a08* (A).

$$C = \min(A,P)$$

$$C = \min(17,21)$$

$$C = 17$$

Fueron 17 los proyectos evaluados de manera satisfactoria, tanto con el método IFPUG como con el método de puntos de casos de uso.

3. Análisis de los datos

3.1. Búsqueda de las ecuaciones lineales

El primer paso del análisis que se realizó sobre los datos recogidos consistió en la búsqueda de las ecuaciones lineales que permitiesen la conversión entre los puntos de función IFPUG y puntos de casos de uso para cada una de las muestras.

1. Muestra *am04*

La ecuación lineal obtenida realizando un ajuste de mínimos cuadrados sobre sus 15 datos es:

$$UCP = 20,23 + 0,394 I$$

Con un $R^2 = 0,6$.

2. Muestra *jjgc06-mla08*

La ecuación lineal resultante haciendo un ajuste de mínimos cuadrados sobre sus 17 datos es:

$$UCP = -0,62 + 0,94 I$$

Con un $R^2 = 0,78$.

3. Muestra *am04- jjgc06-mla08*

La ecuación lineal obtenida haciendo un ajuste de mínimos cuadrados sobre sus 32 datos es:

$$UCP = -15,26 + 0,93 I$$

Con un $R^2 = 0,86$.

La Figura 1 muestra gráficamente los conjuntos de datos utilizados y las ecuaciones de regresión obtenidas para ellos.

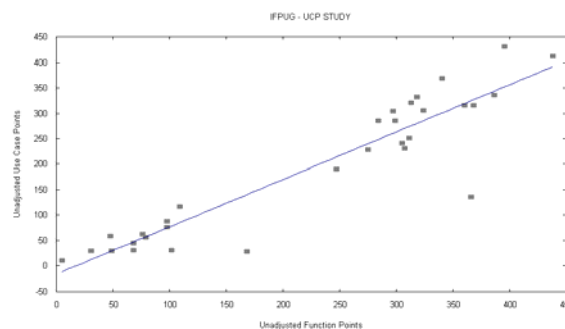


Figura 1. Ecuaciones lineales para *am04-jjgc06-mla08*.

3.2. Análisis de las ecuaciones lineales

Un primer análisis de los indicadores estadísticos de las ecuaciones muestra que todos ellos están dentro de los márgenes considerados aceptables; si bien es cierto que los de la muestra *am04* están ligeramente fuera de los límites, las distancias no son suficientemente grandes como para despreciar los resultados obtenidos.

Cuando se analizan los parámetros de las ecuaciones se observa que las tres, aunque con diferencias notables en los términos independientes, muestran una tendencia marcada en las pendientes de las rectas, hacia el entorno de 1; si se toma la intersección, se obtiene el intervalo (0,7; 0,8). Esto muy significativo, ya que se puede observar la favorable evolución de la recta a mayor número de proyectos medidos con ambos métodos.

3.3. Búsqueda de las ecuaciones no lineales

Aunque en la bibliografía existente no se encuentra ningún estudio de la conversión de unidades de medición del tamaño funcional del software mediante el uso de ecuaciones no lineales, dos importantes razones indican que tal estudio podría proporcionar resultados interesantes:

- El análisis realizado utilizando ecuaciones lineales ha permitido constatar que tanto *jjcg06-mla08* como *am04-jjcg06-mla08* presentan una pendiente similar, en el entorno de uno, lo que parece indicar que sería posible encontrar un factor de conversión entre las unidades de medida IFPUG y puntos de casos de uso en ausencia de un término independiente.
- Un aspecto fundamental en la búsqueda del factor de conversión es el hecho de que la ecuación debería verificar el origen de coordenadas, ya que si hubiera un software con una medida de 0 IFPUG (es decir, sin tamaño), debería tener una medida de 0 puntos de casos de uso y en las ecuaciones lineales presentadas el tamaño que tendría sería el correspondiente al término independiente. Este aspecto incluso se complica más cuando tal término independiente es negativo, ya que daría lugar a un software con un tamaño x , positivo, en IFPUG, que daría lugar no solo a

un tamaño 0 en puntos de casos de uso, sino a un tamaño negativo, lo cual es un absurdo conceptual.

Estas dos importantes razones llevaron, en el marco de las investigaciones presentadas aquí, a analizar el uso de ecuaciones no lineales en busca del factor de conversión. El nuevo análisis con ecuaciones no lineales se realizó, no solo para la muestra de datos obtenida en esta investigación, sino que también se aplicó a la muestra procedente de trabajos previos existentes. La Tabla 1 recoge los resultados obtenidos, una vez realizada la conversión inversa de los parámetros.

Estudio	a	b	R ²
<i>am04</i>	3,86	0,59	0,63
<i>jjcg06-mla08</i>	1,5	0,92	0,84
<i>am04-jjcg06-mla08</i>	1,05	0,94	0,82

Tabla 1. Parámetros de las ecuaciones no lineales.

3.4. Análisis de las ecuaciones no lineales

Un primer análisis de los indicadores estadísticos de las tres ecuaciones muestra que todas ellas se sitúan dentro de los márgenes considerados estadísticamente aceptables.

Cuando se analizan los parámetros de las ecuaciones se observa que las muestras aceptadas presentan un comportamiento tendente, en el valor del exponente, hacia un entorno cercano a 1, lo cual es positivo para el estudio, dado que un exponente con estas características favorece una conversión unívoca entre dos unidades de medida. Igualmente, es destacable la tendencia en el valor del término independiente, el cual marca la proporcionalidad de la conversión. Esta tendencia es hacia la unidad, lo cual es muy significativo e interesante. Si se toma la intersección, analizando los intervalos más probables para el exponente, se obtiene el intervalo (0,7; 1,2), que es un resultado muy esperanzador, observándose que se ha acotado su dispersión respecto a la que se observaba para el término independiente de las ecuaciones lineales.

Si lo anteriormente expuesto se combina con los resultados específicos de la muestra *am04-jjcg06-mla08*, que es la que mayor número de datos tiene y, que en consecuencia, se podría considerar como la más significativa, y observando la manifiesta tendencia en el término independiente y en el exponente hacia su compensación en un entorno de 1, se puede apreciar que están en consonancia. Esto significaría que un punto de

función IFPUG equivaldría, aproximadamente, a un punto de función UCP, con un margen actual de precisión en un entorno de un 25%.

4. Conclusiones y trabajos futuros

Como resultado de las investigaciones realizadas, se ha podido concluir que el factor de conversión entre las unidades IFPUG y puntos de casos de uso está en un entorno en el que a un punto de función IFPUG corresponde un punto de casos de uso. No se ha concluido que sea una conversión exacta, sino aproximada. Así, en determinados casos, el rango de variación podría situarse en torno a un 25%. Se proponen los siguientes trabajos futuros: realizar nuevos análisis sobre nuevos conjuntos de datos más amplios, obtenidos con el doble objetivo de verificar los resultados de este estudio y reducir el intervalo de confianza para el factor de conversión, y contrastar los resultados obtenidos en este tipo de investigaciones, basadas en datos empíricos sobre la conversión de unidades de medición funcional del software, con las que se están llevando a cabo y que se basan en análisis cualitativos.

Referencias

- [1] Albrecht, A. J., "Measuring Application Development Productivity", en *Proceedings Joint SHARE, GUIDE and IBM Application Development Symposium*, IBM, pp. 83-92, 1979.
- [2] Capers, J., "Backfiring: Converting Lines-of-Code to Function Points", *Computer*, vol. 28, nº 11, pp. 87-88, 1995.
- [3] Jacobson, I., *Object-Oriented Software Engineering*, Addison Wesley Professional, 1992.
- [4] Karner, G., *Use Case Points. Resource Estimation for Objectory Projects*, Objective Systems SF AB, 1993.