

Método de Estimación para los proyectos llave en mano de la Universidad de las Ciencias Informáticas

Method of Estimation for the projects key in hand of the University of Informatics Sciences

Dayami Rodríguez Brito *, Ailyn Febles Estrada, Miguel Angel Monagas Reyes,

Ramsés Delgado Martínez

Universidad de las Ciencias informáticas (UCI)

*droduiguez@uci.cu

Resumen

La planificación, en el mundo de la Industria del Software, se ha convertido en uno de los principales retos para la gestión de proyectos y una actividad fundamental para desarrollar software de alta calidad.

Se considera imprescindible para planificar, la aplicación de buenos métodos de estimación, por la creciente influencia que ejercen en el control preciso, predecible y repetido sobre los procesos de producción y los productos de software.

El desarrollo del método de estimación parte de la necesidad en la Universidad de las Ciencias Informáticas (UCI) de estimar el tamaño, costo y esfuerzo requerido para desarrollar los proyectos llave en mano. Dado que la misma no cuenta con una base histórica, se tuvo en cuenta los datos dispersos de algunos proyectos y la evaluación de algunos factores que, según criterio de expertos, pueden influir en las estimaciones de los proyectos.

Para la elaboración del método de estimación se realizó un estudio de otros métodos de la literatura, y algunas buenas prácticas de nuestros proyectos.

Palabras Claves: Estimación, método de estimación, métrica, riesgos.

Abstract

The planning in the world of the Software Industry, has become one of the greatest challenges in project management and a fundamental activity for developing high quality software.

It is considered essential for planning, the implementation of good methods of estimation, by the growing influence they exert on the precise control, predictable and repeated on production processes and software products.

The development of the method of estimation began of the need at the University of the Informatic Sciences (UCI) to estimate the size, cost and effort required to develop turnkey projects. Due to the fact that it has no a historical basis, took into account the data scattered in some projects and assessing some factors that, according to the criterion of experts, can influence the estimates of the projects.

For the development of the method of estimation, was done a study of other methods of literature, and some good practices of our projects.

Key words: Estimates, estimation method, metric, risks.

Introducción

La Universidad de las Ciencias Informáticas (UCI), ha sido creada con un nuevo concepto de universidad productiva. Una de sus misiones es la producción de software y servicios informáticos a partir de la vinculación estudio-trabajo como modelo de

formación, logrando una fuerte relación Universidad-Empresa.

Esta universidad que convierte a la producción en sustento económico, político y social; es productora de grandes soluciones informáticas y está comprometida a ser la vanguardia del desarrollo de las empresas de software en Cuba, llevando la informatización a todos los sectores de la sociedad, de manera tal que convierta a la industria de software en un renglón fundamental de la economía del país.

La UCI tiene retos importantes, dados sus compromisos productivos reales, vinculando a la producción a todos sus estudiantes y profesores, en proyectos de alto valor, ya sea para el mercado nacional, como el internacional, concibiendo la docencia desde la producción.

Como entidad desarrolladora de software, la UCI debe lograr que la planificación se convierta en un elemento esencial a tener en cuenta en el proceso productivo, para obtener productos de alta calidad.

Se considera imprescindible para planificar, la aplicación de buenos métodos de estimación, por la creciente influencia que ejercen en el control preciso, predecible y repetido sobre los procesos de producción y los productos de software. [1]

En la UCI hasta estos momentos no se ha aplicado formalmente un programa de medición de software, por lo que no existen registros históricos en cuanto a tamaño, tiempo, costo, esfuerzo, defectos y recursos de proyectos finalizados, que permitan llevar a cabo un proceso de estimación basado en datos reales. Esto implica una serie de inconvenientes, como que:

- No se definen planificaciones lo más reales posibles, lo que provoca desajustes en los cronogramas previstos, mal empleo de los recursos y que no se pueda controlar el avance en los proyectos llave en mano.
- No se cuantifican los riesgos en dependencia de las actividades en las que afecten los atributos de tamaño, tiempo, costo y esfuerzo.
- No se puedan realizar análisis cualitativos y cuantitativos sobre las condiciones de las facultades para asumir nuevos proyectos.

Dado estos inconvenientes y a la necesidad imperante de solucionarlos, se concibe el Método de estimación UCI.

Desarrollo

El primer paso del método es la identificación de la cantidad de módulos que potencialmente tendrá el sistema a desarrollar. Estos módulos engloban una serie de funcionalidades y estarán compuestos por una determinada cantidad de CU que será estimada por el método. Estos módulos se clasificarán según la complejidad atendiendo al tamaño.

Tabla 1: Cantidad de CUU según la complejidad de los módulos

Cantidad de CUU Máximos y mínimos según complejidad del módulo		
Complejidad	Cota Mínima	Cota Máxima
Pequeño	3	20
Mediano	21	40
Grande	41	70

Métricas de Tamaño

Para unificar los casos de uso según la clasificación de los módulos, se aplicará una métrica donde influye el Factor Cliente (FC) (Epígrafe 2.3.1). Quedando:

- $TMG_{cu} = TotMG * CMaxG * FC * 100$ donde,
TMG_{cu}: Tamaño módulos grandes (cantidad de CUU)
TotMG: Total de módulos grandes
- $TMM_{cu} = TotMM * CMaxM * FC * 100$ donde,
TMM_{cu}: Tamaño módulos Medianos (cantidad de CUU)
TotMM: Total de módulos medianos
- $TMP_{cu} = TotMP * CMaxP * FC * 100$ donde,
TMP_{cu}: Tamaño módulos pequeños (cantidad de CUU)
TotMP: Total de módulos pequeños

Obtenido el valor del Tamaño para cada tipo de Módulo se procede a calcular el tamaño total:

- TotalCUU: $\sum(TMG_{cu} + TMM_{cu} + TMP_{cu})$

Métricas de Tiempo

Para determinar el tiempo de desarrollo por cada etapa identificada se realizó un análisis de lo propuesto por el método de Casos de Uso ajustándolo a la UCI. Los ajustes fueron validados y consultados por algunos expertos, definiéndose los porcentos que representan el total de los tiempos dedicados a cada actividad. Ver Tabla 2.

Tabla 2 Porcentaje de tiempo por actividades estándares de desarrollo

Porcentaje de tiempo por actividad respecto al total de tiempo de desarrollo	
Captura de requisitos	20.00%
Análisis y diseño	16.00%
Implementación	33.00%
Pruebas internas	20.00%
Otras	11.00%

Para calcular tiempo total de desarrollo se inicia por el cálculo de tiempo de la fase de Implementación (T_{Imp}), teniendo en cuenta que el tiempo más fácil de estimar es el de implementación de un CU.

$$T_{Imp} = T_{ImpUnitario} * CUU$$

Donde:

T_{ImpUnitario}: Constante que identifica el Tiempo de Implementación de 1 CU

$$T_{ImpUnitario} = 33$$

Después se calcula el tiempo de Análisis y Diseño (TAD) utilizando los porcentos definidos para cada fase en la tabla 2, agregándole el factor de valor agregado (FAG) (ver epígrafe 2.3.2) multiplicado por la cantidad de CUU.

$$TAD = T_{Imp} * \%AD / \%Imp$$

Donde:

%AD: % de tiempo correspondiente a la fase de Análisis y Diseño

El tiempo dedicado a la captura de requisitos, las pruebas internas y otras actividades se mantienen calculándose sobre la base del tiempo de Implementación (TImp).

- Tiempo dedicado a la Captura de Requisitos (TCR)= (TImp * % Captura de Requisitos)/ % TImp
- Tiempo dedicado a las pruebas internas (TPI)= (TImp * % Pruebas Internas)/ % TImp.
- Tiempo dedicado a las otras actividades (TOA)= (TImp * % Otras)/ % TImp.

El tiempo real de desarrollo (TRD) se calcula utilizando el Esfuerzo de desarrollo (E) (ver epígrafe 2.1.3) y el Total de Hombres (TH) (ver epígrafe 2.1.3)

- $TRD = E/TH$

Métrica del esfuerzo

Para calcular el esfuerzo de desarrollo se tendrán en cuenta todos los tiempos dedicados a las etapas.

- $E = \sum (TCR + TAD + TImp + TPI + TOA)$

Para calcular el Esfuerzo máximo de desarrollo (EMax) se tendrán en cuenta el tiempo dedicado a trabajar diariamente en el proyecto que como promedio es de 8 horas, la cantidad de días laborables a la semana que son 5 y el TLS de proyecto.

- $EMax = HD * DL * TLS$

Donde:

HD: Tiempo de trabajo diario

DL: Cantidad de días laborables a la semana

Métricas de recursos

El total de hombres (TH) máximo para el desarrollando se obtendrá a partir del esfuerzo de desarrollo y el esfuerzo máximo de desarrollo el tiempo real de desarrollo (TRD).

- $TH = E/EMax$

Métricas de Costo

Para calcular el costo de desarrollo de software (CS) se necesita partir del tiempo que llevará el desarrollo y las tarifas asociadas a la cantidad de roles clasificados en el proyecto.

Los roles identificados serán clasificados en: Técnico, Profesional o Consultor de Alto Nivel a los que se les asocia una tarifa presupuestada con una cota mínima y otra máxima. Tabla 3

Tabla 3 Tarifas por rol

Tarifa horaria por rol			
Rol	Mínimo (TMín)	Máximo(TMax)	Porcentage (%)
Técnico	4.47	6.35	15%
Profesional	6.58	10.53	60%
Consultor de Alto Nivel	11.05	15.79	25%

El costo del software se calcula a partir del TRD

- $CS = (TRD * TotalConsultores * TarifaConsultores) + (TRD * Totaltécnicos * Tarifatécnicos) + (TRD * TotalProfesinales * TarifaProfesionales)$

Donde TotalConsultores, Totaltécnicos y TotalProfesinales se calculan a partir del % que representa cada rol en el total de miembros del equipo.

Límites.

Se han establecido algunos límites para evaluar la factibilidad del desarrollo del software. El software se realiza si:

$TH < 80$. En caso contrario se debe redefinir el alcance del proyecto, el % del Factor Ambiente (FA) (ver epígrafe 3.3.3) es menor del 50%. De lo contrario se considera que la Facultad no tiene las condiciones para desarrollar el proyecto.

Factores de Complejidad que influyen en las métricas

Los factores de complejidad definidos se listan dándoseles un peso determinado en una escala de 0.5 - 1.5 en dependencia del nivel de importancia. La complejidad se da en un primer momento como una valoración cualitativa de cada factor y luego internamente se cuantifica en:

1. Escala de 0-5.

Evaluación	Valoración
Alto (A)	5
Medio Alto (MA)	4
Medio (M)	3
Medio Bajo (MB)	2
Bajo (B)	1

2. Escala de 0-1.

Evaluación	Valoración
Si	1
No	0

Luego de clasificar la complejidad por cada factor se procede a calcular el resultado final para cada uno al multiplicar el peso predeterminado para cada factor y el valor equivalente que toma cada valoración cualitativa.

El resultado final será la sumatoria del valor calculado para todos los criterios llevándose a una escala de conveniencia en función de su uso en la fórmula final del método. Los máximos se determinan a partir de configuraciones totalmente positivas o negativas.

Factor del Cliente:

Permite determinar un nivel de incertidumbre asociado a la madurez que tiene el cliente en materia de asimilación de proyectos informáticos, se utilizará para determinar la cantidad de casos de uso que se dejan como reserva.

Factor	Peso	Valoración	P*V
Existe sistema anterior.	0.5	(Si, NO)	
Existen resultados de informatización.	1	(Si, NO)	
Existe Dirección de Informatización o Informática.	1	(Si, NO)	
Existe infraestructura tecnológica en la organización.	1	(Si, NO)	
Existe base legal en la organización.	1	(Si, NO)	
Es el primer proyecto con la organización	0.5	(Si, NO)	
Existe una estructura clara en la	1	(Si, NO)	

organización.			
Existe un especialista para atender al proyecto	1	(Si, NO)	
Están definidas las funciones de las áreas.	1.5	(Si, NO)	
Estabilidad de los requisitos.	1	(Si, NO)	
El cliente es el usuario de la aplicación.	0.5	(Si, NO)	

El Factor Cliente se calcula como: uno más, la sumatoria del producto de cada factor individual por su respectivo peso convertido a una escala entre 1 y 1.5 mediante la fórmula

$$FC = (FO*0.5)/FP$$

Donde:

FO: factor obtenido

FP: factor obtenido asignando valoraciones pesimistas.

Factor de valor agregado

Factor de Valor Agregado (VA): determina algunas acciones que pueden representar esfuerzo adicional al desarrollo de software, incluye en todas las etapas fundamentalmente en la de Análisis y Diseño. Se debe evitar seleccionar estas opciones en aquellos casos donde el nivel de desarrollo sea alto.

Factor	Complejidad	P*V
Ayuda Integrada.	0	0
Mantener una interfaz familiar al negocio.	0	0
Aplicación multidioma.	0	0

Factor ambiente

Factor de Ambiente (FA): es interno, se determina a través de analizar las condiciones de la facultad en la que se desarrolla en proyecto, no influye en el precio final, pero si en las valoraciones para la realización del proyecto y se utilizará como indicador para saber si la facultad ha creado condiciones para cumplir los compromisos. Se aplicará en el precio de negociación y previo a la contratación.

Factor	Peso	Complejidad	P*V
Temática conocida por la facultad.	1.5		
La facultad tiene conocimientos sobre Base de Datos.	1		
La facultad tiene conocimientos sobre Leguaje de Programación.	1		
La facultad tienes conocimientos sobre el Control de Versiones.	1		
La facultad tiene conocimientos sobre la Gestión de Proyectos.	1		
Proyecto de continuidad.	2		

Disponibilidad de la facultad de líder de Proyecto.	1		
---	---	--	--

Factor de Complejidad Técnica

Es asociado a la complejidad de la tecnología y requerimientos no funcionales del sistema, influye directamente en el esfuerzo y fundamentalmente en las etapas de implementación.

Factor	Peso	Evaluación	P*V
Rendimiento de la aplicación.	1	(A, MA, M, MB, B)	Peso * Valoración
Rendimiento de la Plataforma.	1	(A, MA, M, MB, B)	
Complejidad de Diseño.	1	(A, MA, M, MB, B)	
Procesamiento interno complejo.	1	(A, MA, M, MB, B)	
Seguridad.	0.5	(A, MA, M, MB, B)	
Reutilización.	0.5	Según Fórmula	
Integración.	1	Según Fórmula	
Asimilación de Sistemas.	1	Según Fórmula	
Asimilación de Dispositivos.	0.5	Según Fórmula	
Asimilación de Estándares.	1	Según Fórmula	

En el caso específico de este factor algunos de los valores para calcular la complejidad están dados por las siguientes métricas, para la cual existe una escala para llevarla a la ecuación cualitativa:

- Porcentaje de Reutilización = $CMR / CMD * 100$
 - Cantidad de Módulos con Reutilización (CMR).
 - Cantidad de Módulos a Desarrollar (CMD).
- Porcentaje de Integración = $CSI / CMS * 100$
 - Cantidad de Sistemas a Integrar o Legar (CSI).
 - Cantidad de Módulos del Sistema (CMS).
- Porcentaje de Asimilación de Sistemas = $CSI-D / CSI * 100$
 - Cantidad de Sistemas a Integrar o Legar (CSI).
 - Cantidad de Sistemas a Integrar o Legar Desconocidos (CSI-D).
- Porcentaje de Asimilación de Dispositivos = $(CDE - DRV) / CDE * 100$
 - Cantidad de Dispositivos Externos (CDE).
 - Cantidad de Dispositivos Externos con Driver o SDK (DRV).
- Porcentaje de Asimilación de Estándares = $CEE / CEC * 100$
 - Cantidad de Estándares y Normas a cumplir (CEC).
 - Cantidad de Estándares y Normas específicas (CEE).

El valor del factor para cada caso será la proporción entre las medidas sin ser convertido a valores porcentuales, por lo que para estos factores no aplican los criterios cuantitativos.

Conclusiones

Realizar la estimación de un proyecto software no es tarea fácil, máxime cuando no existe un método único viable para que cada tipo de proyecto.

Como conclusión del presente trabajo se obtuvo un método matemático basado en métodos reconocidos internacionalmente para la estimación de proyectos, adaptados a las características específicas de la Universidad, agregando factores que posibiliten a la toma de decisiones asociadas a la ejecución del proyecto.

El método ha sido aplicado en proyectos en estado postmortem corroborando los resultados con los factores obtenidos por el proyecto.

Referencias Bibliográficas

- Cao, M.I.J.I. *"Principios para un método de estimación de proyectos de software basado en los escenarios principales"*.
- Nuchera, A.H., *"Una introducción a la gestión de riesgos tecnológicos"*. 2004.
- Pressman, R.S., *"Ingeniería de Software. Un enfoque práctico"*. 2005.
- Kan., S.H., *"Metrics and Models in Software Quality Engineering"*. 2000.
- Ovejero, I.J.D. *"Estimación de proyectos para sistemas basados en conocimiento"*.
- Hernández, S.E.B. (2002) *"Métricas de estimación de tamaño Puntos de Caso de Uso"*.