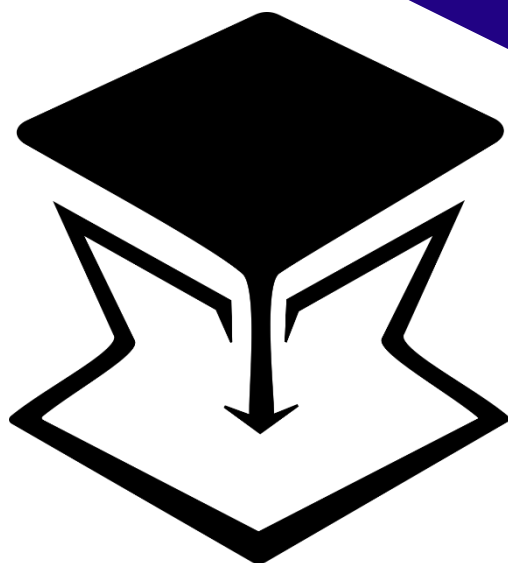




SIGMA

Plan de Estimación

SIGMA

3-BYTE Team

Adrián PORTILLO, Glenda FLORES, Franco ROSAS

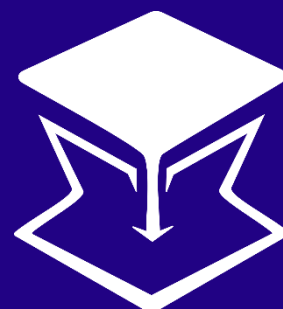
Laboratorio de Desarrollo de Software
Unidad Académica Río Gallegos
Universidad Nacional de la Patagonia Austral

3-BYTE
TEAM



Una parte importante de la toma de decisiones al comenzar un nuevo proyecto de desarrollo de software está dada por el costo que éste tendrá

El método de puntos en casos de uso, que esbozaremos a continuación, es un método de estimación prometedor que se adapta bien al enfoque de caso de uso para la descripción de los requisitos. En sus bases yace el concepto de transacción de caso de uso, la unidad más pequeña de medición. Lamentablemente, hay muchas suposiciones disidentes sobre el concepto.



SIGMA

Tabla de contenido

Introducción	4
<i>Propósito.....</i>	<i>4</i>
<i>Alcance</i>	<i>5</i>
<i>Resumen</i>	<i>5</i>
<i>Objetivos de la administración</i>	<i>5</i>
<i>Referencias</i>	<i>5</i>
Métricas basadas en Casos de Uso	6
<i>Puntos de Casos de Uso Sin Ajustar</i>	<i>6</i>
Peso de los Actores (UAW)	6
Peso de los Casos de Uso (UUCW)	7
Cálculo de los Puntos de Casos de Uso sin Ajustar (UUCP)	8
<i>Puntos de Casos de Uso Ajustados</i>	<i>8</i>
Factores Técnicos.....	9
Factores del entorno.....	10
Cálculo de Puntos de Casos de Uso Ajustados (UCP)	11
<i>Estimación del número de Horas-Hombre.....</i>	<i>12</i>

Plan de Estimación

Introducción

Los casos de uso por sí mismos no permiten efectuar una estimación del tamaño que tendrá el sistema, ni del esfuerzo y el tiempo necesario para implementarlo. Estos permiten documentar los requerimientos del software de una manera compacta y precisa, luego con los puntos de función se puede estimar el tamaño del software a partir de los requerimientos obtenidos de los casos de uso.

Puntos de función de casos de uso consiste en evaluar la complejidad de un sistema de software por medio de una técnica en la que se le asigna una cantidad de puntos de peso, que califican diferentes elementos que componen el sistema de software así como algunos factores del entorno, para obtener una aproximación del tiempo requerido y la cantidad de esfuerzo necesario para la implementación del mismo.

Este proceso se lleva a cabo mediante una serie de pasos que como se mencionó anteriormente evalúan cada factor, empezando por ponderar los casos de uso sin ajustar. Esto quiere decir que únicamente son tomados en cuenta los actores (UAW) y los casos de uso (UUCW). Dicho paso se lleva a cabo dejando por el momento los factores técnicos (TCF) y los factores ambientales (EF), para evaluarlos más tarde. Con el fin de multiplicarlos por el resultado final de los casos de uso sin ajustar. Así, se da el resultado de los casos de uso ajustados, que caracteriza la complejidad del sistema y es usado para obtener una idea del número de horas-persona para un proyecto.

Propósito

Durante el desarrollo de un proyecto de software es necesario estimar:

- Cuanto tiempo se tardará en desarrollar el sistema
- Cuantas personas se necesitarán para desarrollarlo

Por lo tanto es necesario cuantificar:

- Complejidad del Sistema
- Funcionalidad
- Complejidad Técnica
- El nivel de experiencia de los integrantes del proyecto
- El tiempo necesario para producir una unidad funcional

En este documento presenta un método de estimación desarrollado por Gustav Karner, perteneciente a Rational Software Corporation, el cual caracteriza la complejidad de un sistema a través de Puntos de Casos de Uso.

Alcance

El método presentado es estimativo y deberá ser utilizado para tomar una idea del número de horas – hombre para el proyecto.

Las fórmulas han sido obtenidas empíricamente, y la experiencia hasta la fecha muestra que las mismas se aplican en buena forma para aplicaciones de negocios, como ser sistemas de información.

Resumen

El presente método se compone de los siguientes pasos:

- Cálculo de Puntos de Casos de Uso Sin Ajustar (UUCP)
- Cálculo de Puntos de Casos de Uso Ajustados (UCP)
- Estimación del número de Horas-Hombre
- Estimación del número de Horas-Hombre refinado

Objetivos de la administración

El objetivo de este plan es obtener un número simple que caracterice completamente al sistema y se corresponda con la productividad en las tareas de ingeniería que ha sido observada.

Referencias

Applying Use Cases: A Practical Guide

Autores: G. Schneider y J. P. Winters

Edición: Segunda – Editorial: Addison Wesley

ISBN: 0-2001-70853-1

Resource Estimation for Objectory Projects (Paper)

G. Kerner - Objectory Systems

The Estimation of Effort Based on Use Cases (Paper)

J. Smith - Rational Software

Carroll, 2005; Clemmons, 2006; Karner, 1993; Nageswaran, 2007

Métricas basadas en Casos de Uso

Puntos de Casos de Uso Sin Ajustar

Para realizar este cálculo es necesario conocer el Peso de los Actores y el Peso de los Casos de Uso.

Peso de los Actores (UAW)

Se consideran los actores del sistema determinando si su intervención es simple, promedio o avanzada de acuerdo a la siguiente tabla:

Tipo de Actor	Descripción
Simple	Otro sistema con una interfaz de aplicación definida.
Promedio	Otro sistema interactuando a través de un protocolo, o un usuario interactuando por medio de una interface basada en texto.
Complejo	Un usuario interactuando a través de una GUI.

Se cuenta la cantidad de actores de cada tipo.

Se multiplica cada tipo por el factor de peso.

Tipo de Actor	Peso
Simple	1
Promedio	2
Complejo	3

Finalmente, se suman estos productos para obtener el total.

Peso de los Casos de Uso (UUCW)

En primer lugar se debe determinar para cada Caso de Uso su tipo: Simple, Promedio o Complejo.

Existen dos alternativas para realizar esta tarea:

- Basado en transacciones
- Basado en Clases de Análisis

Basado en transacciones

Se determina el número de transacciones en un caso de uso, incluyendo los flujos alternativos.

Se entiende por transacción a un set atómico de actividades, que deben ejecutarse completamente o no hacerlo en absoluto.

Tipo de Caso de uso	Descripción – Compuesto por	Factor
Simple	3 transacciones o menos	5
Promedio	4 a 7 transacciones	10
Complejo	Más de 7 transacciones	15

Basado en Clases de Análisis

Se debe observar cuantas clases de análisis se han utilizado para realizar un caso de uso en particular

Tipo de Caso de uso	Descripción – Compuesto por	Factor
Simple	Menos de 5 clases de análisis	5
Promedio	5 a 10 clases de análisis	10

Tipo de Caso de uso	Descripción – Compuesto por	Factor
Complejo	Más de 10 clases de análisis	15

A continuación se debe determinar de qué tipo es cada caso de uso.

Se cuenta cuantos casos de uso de cada tipo se tienen.

Se multiplica la cantidad de cada tipo por el factor

Finalmente, se suman esos productos para obtener el total.

Cálculo de los Puntos de Casos de Uso sin Ajustar (UUCP)

(Obtenido a través de uno de los dos mecanismos mencionados).

$$\text{UUCP} = \text{UAW} + \text{UUCW}$$

Puntos de Casos de Uso Ajustados

Este método incorpora factores técnicos y del entorno, de manera de obtener una estimación más adecuada para el tipo de sistema a desarrollar.

Factores Técnicos

Referencias

Factor Técnico	Descripción	Peso
T1	Sistema Distribuido	2
T2	Objetivos de alta performance	1
T3	Eficiencia para el usuario final	1
T4	Procesamiento interno complejo	1
T5	El código debe ser reusable	1
T6	Facilidad de instalación	0,5
T7	Facilidad de uso	0,5
T8	Portabilidad	2
T9	Facilidad al cambio	1
T10	Concurrencia	1
T11	Incluye características especiales de seguridad	1
T12	Provee acceso directo tripartito	1
T13	Entrenamiento de usuarios especial	1

Peso de los factores técnicos (Tfactor)

En base a la tabla anterior, calificar cada ítem de 0 a 5, en donde:

- Un puntaje de 0 significa que el factor es irrelevante
- Un puntaje de 5 significa que el factor es esencial
- Un puntaje de 3 significa que el factor es promedio

Se multiplica cada factor por su calificación.

Se suman estos productos para obtener el total TFactor

Para obtener la valoración final, se utiliza la siguiente fórmula:

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

Factores del entorno

Referencias

Factor del Entorno	Descripción	Peso
E1	Experiencia en el uso de RUP	1,5
E2	Experiencia en la aplicación	0,5
E3	Experiencia en el desarrollo Orientado a Objetos	1
E4	Capacidad del Líder	0,5
E5	Motivación	1
E6	Requisitos estables	2
E7	Trabajadores part-time	-1
E8	Dificultades en el lenguaje de programación	-1

Peso de los factores del entorno (EF)

En base a la tabla anterior, calificar cada ítem de 0 a 5, de acuerdo a lo siguiente:

Para los factores E1 a E4

Un puntaje de 0 significa que no hay experiencia en ese punto

Un puntaje de 5 significa experto

Un puntaje de 3 significa promedio

Para el factor E5 (Motivación)

Un puntaje de 0 significa que no hay motivación en el grupo de desarrollo

Un puntaje de 5 significa una alta motivación

Para el factor E6 (Requisitos estables)

Un puntaje de 0 significa requisitos extremadamente inestables

Un puntaje de 5 significa requisitos sin cambios

Para el factor E7 (Trabajadores part-time)

Un puntaje de 0 significa que no hay trabajadores part-time

Un puntaje de 5 significa que todos los trabajadores son part-time

Para el factor E8 (Dificultades en el lenguaje de programación)

Un puntaje de 0 significa que el lenguaje de programación es de uso sencillo

Un puntaje de 5 significa que el lenguaje de programación presenta muchas dificultades

Se multiplica cada factor por su calificación.

Se suman estos productos para obtener el total EFactor

Para obtener la valoración final, se utiliza la siguiente fórmula:

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

Cálculo de Puntos de Casos de Uso Ajustados (UCP)

Los puntos Casos de Uso Ajustados se obtienen de acuerdo a la siguiente fórmula:

$$UCP = UUCP * TCF * EF$$

Estimación del número de Horas-Hombre

En general, Karner sugiere utilizar 20 horas-hombre por UCP. Sin embargo, a los fines del desarrollo de SIGMA, y teniendo en cuenta la información provista por los docentes de la cátedra Laboratorio de Desarrollo de Software (basada en anteriores proyectos de la materia) se utilizará el factor 4 horas-hombre por UCP.

$$\text{Total Hombres Hora} = \text{UCP} * 4$$

Cabe destacar, que el valor obtenido corresponde solamente a la codificación, por lo que para obtener el esfuerzo total del proyecto es necesario realizar el cálculo a partir de la siguiente tabla:

Actividad	Porcentaje
Análisis	10%
Diseño	20%
Codificación	40%
Pruebas	15%
Sobrecarga (otras actividades)	15%

Finalmente, el valor total obtenido se divide entre la cantidad de integrantes del grupo (en este caso 3) y la dedicación diaria de cada uno, para obtener la cantidad de semanas de desarrollo.