

Práctica 2. Planificación temporal y estimación del coste del proyecto

Planificación y Gestión de Proyectos Informáticos

*Máster Profesional en Ingeniería Informática
Universidad de Granada
Curso académico 2016/2017*

*Pablo Martín-Moreno Ruíz
María Victoria Santiago Acalá*

*76654141V
76656695H*

Contenidos

Estimación por descomposición funcional.....	3
Estimación por descomposición de actividades.....	4
Estimación del tamaño del proyecto (KLOC, FP o casos de uso).....	5
Estimación con herramientas software (Construx Estimate).....	6
Estimación con modelos empíricos: COCOMO II.....	7
Estimación con modelos empíricos: Modelo de Putnam.....	9
Conclusiones finales.....	10

Estimación por descomposición funcional.

Para realizar esta estimación hemos supuesto que un mes tiene 20 días.

Módulo	Esfuerzo Estimado
Identificación del problema	0'1 meses
Especificación de las necesidades	0'5 meses
Análisis de viabilidad	4'5 meses
Revisar la propuesta con el cliente	1 meses
Diseño general	0'5 meses
Diseño y desarrollo de la base de datos	1'5 meses
Diseño y desarrollo de la aplicación	2'5 meses
Diseño y desarrollo del componente de códigos QR	0'5 meses
Diseño y desarrollo del componente del control de cámara	1 meses
Diseño y desarrollo del componente del control de cámara por tablet	0'5 meses
Total	12'5 meses

Hemos estimado cuánto tardaremos en realizar tareas en meses y nos sale en total 12'5 personas*mes.

Estimación por descomposición de actividades.

	Comunicación con el cliente	Análisis	Diseño	Código	Test	Documentación	Total
Identificación del problema	0,05	0,05	0	0	0	0	0,1
Especificación de las necesidades	0	0,65	0	0	0	0	0,65
Análisis de viabilidad	0	4,4	0	0	0	0	4,4
Revisar la propuesta con el cliente	0,95	0	0	0	0	0	0,95
Diseño general	0	0	0,6	0	0	0	0,6
Diseño y desarrollo de la base de datos	0	0	0,3	0,65	0,2	0,15	1,3
DD. appi	0,25	0	0,3	1,05	0,5	0,2	2,3
DD. QR	0,05	0	0,1	0,175	0,1	0,05	0,475
DD. Cámara	0,15	0	0,1	0,35	0,25	0,05	0,9
DD. Controlador Cámara	0,1	0	0,1	0,225	0,25	0,05	0,725
Total	1,55	5,1	1,5	2,45	1,3	0,5	12,4
%	12,5	41,12903226	12,09677419	19,75806452	10,48387097	4,032258065	

En esta estimación podemos ver que dedicamos aproximadamente un 65% del tiempo en comunicación con el cliente, Análisis y diseño. Esto es debido a que tomamos muy en cuenta su opinión en varias etapas del proyecto, de hecho, en las primera etapas hasta el diseño general dedicamos 6'75 meses que cómo se trabajaría en equipo se resumiría en 3'4 aproximadamente.

En las siguientes etapas del proyecto tendremos algo presente también al cliente y dedicaremos. En esta fase, implementaremos, documentaremos y especificaremos los test. En total en el proyecto la implementación 20% aproximadamente, la documentación un 11% aproximadamente y la especificación del testeto aproximadamente un 4%.

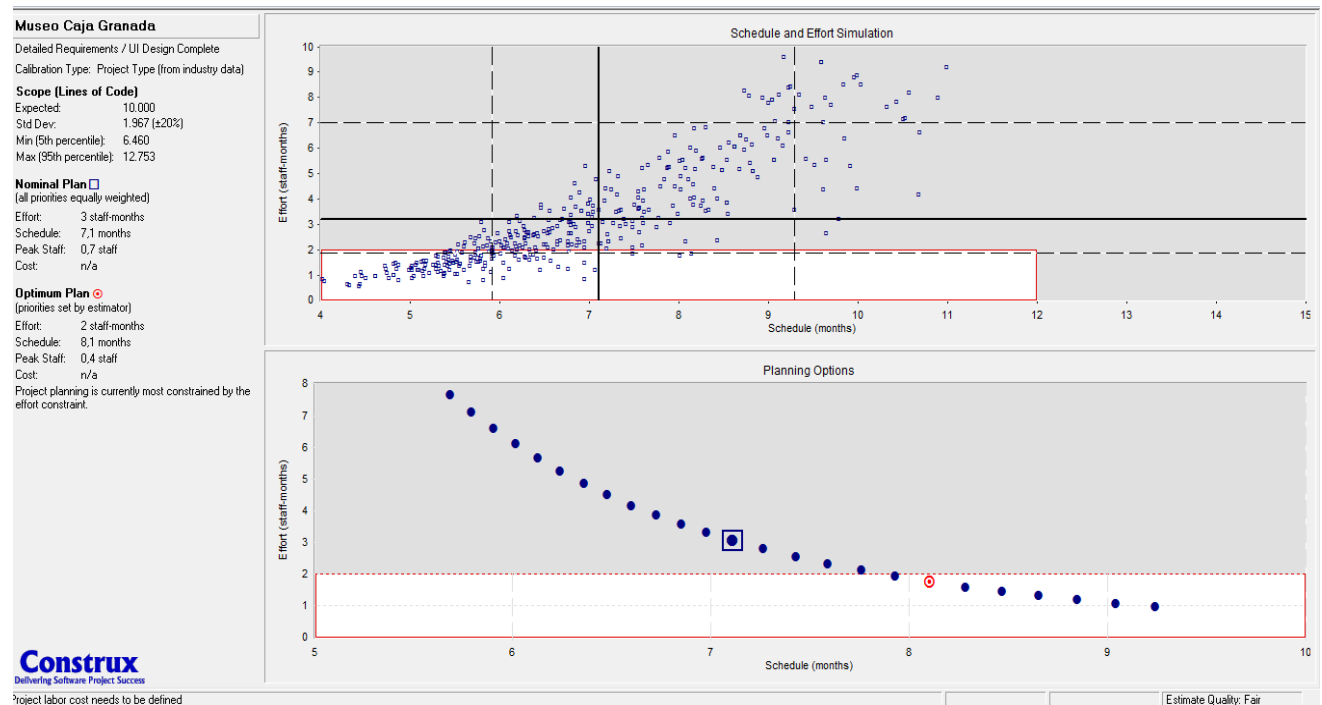
Estimación del tamaño del proyecto (KLOC, FP o casos de uso).

	KLOC
Identificación del problema	0,1
Especificación de las necesidades	0,7
Análisis de viabilidad	4,3
Revisar la propuesta con el cliente	1
Diseño general	0,5
Diseño y desarrollo de la base de datos	1,3
DD. appi	2,3
DD. QR	0,5
DD. Cámara	1
DD. Controlador Cámara	0,7
Total	12,4

Creo que todo el desarrollo en total aproximadamente será de unos 12,4 KLOC. Suponiendo 2 KLOC realizados al mes por persona, estimamos que aproximadamente tardaremos 11,2 meses.

Estimación con herramientas software (Construx Estimate).

Para realizar la estimación hemos supuesto que el tipo es un Sistema software y el subtipo es network internal. Además, hemos supuesto que son 10.000 SLOC.



Podemos ver que este software estima que acabaremos en 8'1 meses siendo dos personas en el equipo. Y siendo una sola persona algo más de 9 meses.

Estimación con modelos empíricos: COCOMO II.

Para el ajuste de esta herramienta hemos supuesto que usamos 10.000 SLOC y para la configuración de los atributos hemos valorado habilidades individuales tanto de conocimientos pasados por en el tema por uno de los miembros, las habilidades en los diversos lenguajes, herramientas y plataforma, Cómo las habilidad dades de coordinación del equipo.

Además, hemos tenido en cuenta la flexibilidad de los timepo, la reusabilidad del producto a desarrollar y la cantidad de documentación.

Se ve reflejado en la siguiente imagen:

Software Size Sizing Method | Source Lines of Code ▼

	<u>SLOC</u>	% Design Modified	% Code Modified	% Integration Required	Assessment and Assimilation (0% - 8%)	Software Understanding (0% - 50%)	Unfamiliarity (0-1)
New	10000						
Reused		0	0				
Modified							

Software Scale Drivers

Precedentedness	Very High ▼	Architecture / Risk Resolution	Nominal ▼	Process Maturity	Low ▼
Development Flexibility	Nominal ▼	Team Cohesion	Extra High ▼		

Software Cost Drivers

Product		Personnel		Platform	
Required Software Reliability	Nominal ▼	Analyst Capability	High ▼	Time Constraint	Extra High ▼
Data Base Size	Low ▼	Programmer Capability	Very High ▼	Storage Constraint	Extra High ▼
Product Complexity	Nominal ▼	Personnel Continuity	Very High ▼	Platform Volatility	Low ▼
Developed for Reusability	Extra High ▼	Application Experience	High ▼	Project	
Documentation Match to Lifecycle Needs	Very High ▼	Platform Experience	Nominal ▼	Use of Software Tools	Very High ▼
		Language and Toolset Experience	High ▼	Multisite Development	Extra High ▼
				Required Development Schedule	Nominal ▼

Maintenance Off ▼

Software Labor Rates

Cost per Person-Month (Dollars) 3200

Calculate

En la siguiente imagen vemos los resultados obtenidos:

Results

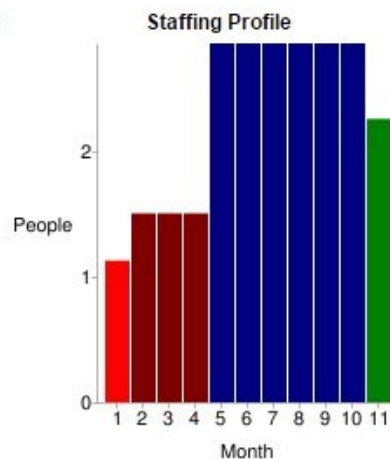
Software Development (Elaboration and Construction)

Effort = 24.9 Person-months
 Schedule = 10.6 Months
 Cost = \$0

Total Equivalent Size = 10000 SLOC

Acquisition Phase Distribution

Phase	Effort (Person-months)	Schedule (Months)	Average Staff	Cost (Dollars)
Inception	1.5	1.3	1.1	\$0
Elaboration	6.0	4.0	1.5	\$0
Construction	19.0	6.6	2.9	\$0
Transition	3.0	1.3	2.3	\$0



Software Effort Distribution for RUP/MBASE (Person-Months)

Phase/Activity	Inception	Elaboration	Construction	Transition
Management	0.2	0.7	1.9	0.4
Environment/CM	0.1	0.5	0.9	0.1
Requirements	0.6	1.1	1.5	0.1
Design	0.3	2.2	3.0	0.1
Implementation	0.1	0.8	6.4	0.6
Assessment	0.1	0.6	4.6	0.7
Deployment	0.0	0.2	0.6	0.9

Your output file is http://csse.usc.edu/tools/data/COCOMO_November_8_2016_17_54_46_122610.txt

Created by Ray Madachy at the Naval Postgraduate School. For more information contact him at rjmadach@nps.edu

Este software estima el tiempo que debemos de dedicar en este proyecto, que en concreto es 24'9 personas mes.

En la gráfica multicolor nos indica el tiempo de cada fase:

- Inicialización que consiste en la la interacción con el cliente. Dónde cocomo asigna a una persona.
- Elaboración que indica la realización de documentación del proyecto. Donde cocomo asigna a una persona y a otra a tiempo parcial.
- Construcción que es la implementación de aplicación. Donde cocomo asigna a unas tres personas.
- Transición que es la fase dónde se ultiman los detalles. Cocomo asigna a dos personas para este último proceso.

La otra tabla nos muestra para cada tarea que cantidad de los antiguos procesos se realiza.

Estimación con modelos empíricos:

Modelo de Putnam.

El modelo de Putnam es un modelo de estimación de esfuerzo software el cual permite realizar estimaciones aplicando la siguiente fórmula:

$$E = B \times \left(\frac{LOC}{P}\right)^3 \times \frac{1}{t^4}$$

Donde tenemos que nuestros parámetros usados para nuestra estimación son los siguientes:

- B: esta variable representa el valor de escala que vamos a tener. Puesto que nuestro proyecto tiene un tamaño que está entre el intervalo [5,15] KLOC, el valor de B es 0.16.
- LOC: el total de LOC son 10000.
- P: el parámetro P depende del tipo de proyecto pudiendo ser un proyecto de sistemas empotrados, software científico, software de sistemas y comunicaciones o aplicación de gestión. Nuestro sistema se identifica más con el grupo formado por software de sistemas y comunicaciones, por lo tanto, el valor de P asociado a este grupo es 10000.
- t: representa el tiempo mínimo en años que va a durar el proyecto, en nuestro caso tiene el valor **0.5** puesto que el proyecto realizado por dos personas tendría la duración de 6 meses.

Por lo que finalmente, tras aplicar la ecuación tenemos que el esfuerzo necesario es de:

B	0.16
P	10000
t	0.5
LOC	10000

$$E = B \times \left(\frac{LOC}{P}\right)^3 \times \frac{1}{t^4} = 0.16 \times (1^3) \times \frac{1}{0.5^4} = 2,56$$

Como podemos observar a partir del resultado obtenido, este modelo no es el más adecuado a un proyecto de poca magnitud como el nuestro.

Por lo que se necesitan 2.56 personas para finalizar en 6 meses según esta estimación.

Conclusiones finales.

Wikipedia define la estimación estadística como “conjunto de técnicas que permiten dar un valor aproximado de un parámetro de una población a partir de los datos proporcionados por una muestra”.

Como muy bien indica esta definición es un valor aproximado basado en una población o lo que es nuestro caso es en el historial de proyectos similares.

Para realizar este proyecto hemos usado 6 estimaciones:

- La estimación funciona que es muy similar al usado de la herramienta diagrama de Gantt (de hecho, esa herramienta asigna un tiempo en calendario a la estimación puesta por el gestor). Esta estimación tiene la ventaja de que es el gestor el que asigna el tiempo que piensa que su proyecto necesitará para realizar cada tareas. Además, de que cada proyecto puede tener una peculiaridades distintas o una forma de trabajar distinta por lo que esta forma se ajusta bastante bien.
- La estimación por actividades es similar a la anterior (de hecho, podría incluso partirse de la anterior para dar un aproximación inicial a esta). En esta se tiene para cada módulo una estimación de cada tipos de tarea que se requiere para cada uno (comunicación con el cliente, Diseño, Implementación, testeo), pudiendo dar un mejor ajuste.
- La estimación por tamaño del proyecto por KLOC. En este caso se asignan para cada módulo el número de líneas aproximadas que necesita invertirse para módulo. Podría ser una opción para empresas que tengan un buen historial de proyecto y gran experiencia para estimar el número de líneas en cada proyecto
- La herramienta construx estimate está basado en la estadística de un gran conjunto de proyectos. Con esta herramienta podemos darle información demasiado general para el proyecto y nos indicará el tiempo estimado y el esfuerzo. Un problema de esa herramienta es que solo nos permite un lenguaje de programación para la estimación, por lo que no es demasiado útil para sistemas basados en componentes que combinen diversos lenguajes.
- La herramienta cocomo dos nos permite introducirle una gran cantidad de parámetros para indicarle cómo será el proyecto, además del número de líneas. El problema de esta estimación es que usa unos datos muy complejos y que en algunos casos necesita un historial de proyectos o el contexto de una empresa para poder estimarlos adecuadamente.
- El modelo de Putnam indica el esfuerzo que necesitan para una determinada arquitectura de proyecto, un número de líneas y el tiempo que crees que debe de

tener el proyecto. Esta estimación podría ser útil en el caso de saber los atributos con certeza para darnos una idea.

En resumen, hemos indicado como estimaciones más adecuadas la funcional y la de actividad, si sólo debemos quedarnos con una sería esta última. Porque estas son las más abiertas, las que permiten gestionar el tiempo de una forma más libre y permite plasmar metodologías iterativas, una arquitectura deseada e incluso la combinación de diversas tecnologías.

La estimación del tamaño del proyecto es algo difícil en número de líneas y, de hecho, es muy similar a la funcional sólo que expresando el tiempo en KLOC y luego traduciendo.

Las estimaciones construx estimate es demasiado ajustada con pocos parámetros de variación. Por ejemplo, nosotros hemos estimado que supuesto varios lenguajes, y una fase de análisis y elaboración muy elevada que no refleja la herramienta.

La herramienta cocomo necesita demasiada experiencia previa y necesitas tener muy claro el número de líneas para la estimación.

Y el modelo de putnam tiene fallos similares a contrax pero además tiene que indicarle el tiempo final del proyecto.

Estimar por líneas lo vemos un error, ya que nunca se sabe el trabajo que lleva detrás un proyecto simplemente por las líneas implementas. Creemos que afecta aún más la experiencia de la empresa, del equipo y los precedentes. Por eso creemos que cada equipo debería de estimar el tiempo que le hace falta realizar una tarea según sus capacidades.