
PLAN DE ESTIMACIÓN DE PROYECTOS SOFTWARE



GESTIÓN DE PROYECTOS SOFTWARE Y METODOLOGÍAS DE DESARROLLO

Facultad de Informática

Universidad Complutense de Madrid

Jonathan Carrero Aranda
José Luis Díaz Minaya
José Francisco García Ruiz
Eloy González Acedo
Daniel Martín Barrios
Javier Mendoza Gómez
Pedro Sánchez Ramírez
David de los Santos Gil

Historial de versiones

Fecha	Versión
17/05/2016	LK-PEPS - v1.0
26/05/2016	LK-PEPS - v1.1

Índice de Contenido:

1. Introducción	4
2. Técnicas de estimación	4
3. Calcular puntos de función sin ajustar	4
4. Estimación usando el modelo COCOMO II	14
5. Modelo de post arquitectura	14
6. Estimación del tamaño del software	14
7. Escala de ahorro-gasto	17
8. Cálculo de esfuerzo	19
9. Multiplicador de esfuerzo	20
10. Cálculo de tiempo de desarrollo	20
11. Plan de presupuesto	21
12. Acrónimos	22

1. Introducción

El objetivo de este documento es predecir las variables involucradas en el proyecto con cierto grado de certeza, así como tratar de aportar una predicción con indicadores de estimación basados en Puntos de Función y COCOMO II. Estimando obtendremos las debilidades que exponen dichos indicadores y se establecerá la idea de la solución. Finalmente, calcularemos el tiempo necesario para desarrollar el proyecto en base al equipo del que disponemos.

2. Técnicas de estimación

Las técnicas de estimación nos permitirán resolver problemas en donde, en la mayoría de los casos, el problema a resolver es demasiado complejo para considerarlo como una sola parte. Por esta razón, descomponemos el problema obteniendo un conjunto de problemas más pequeños. A continuación, se detallan las técnicas utilizadas en el proyecto.

2.1 COCOMO II

Pertenece a la categoría de modelos de subestimaciones basados en estimaciones matemáticas. Está orientado a la magnitud del producto final, midiendo el "tamaño" del proyecto, en líneas de código principalmente. Dentro del modelo de COCOMO II se puede distinguir dos modelos:

- Modelo de diseño preliminar.
- Modelo de post arquitectura.

En nuestro caso particular vamos a utilizar el modelo de post arquitectura.

3. Calcular puntos de función sin ajustar

Primero se calculan los puntos de función sin ajustar para cada tipo de función según la cantidad de los diferentes datos de entrada/salida.

Tipos de Función Procedimientos:

- Salidas externas (EO)
- Consultas externas (EQ)

Tipos de Función Datos:

- Ficheros lógicos internos (ILF)
- Ficheros de interfaz externos (EIF)
- Entradas externas (EI)

1. Módulo Gestor Usuario:

1.1- Alta Usuario:

Ficheros lógicos internos(1):

- DET (Campos): 7, id, nickname, nombre, apellidos, correo, contraseña, tipo de usuario(VIP o no VIP)
- RET (Subgrupos): 1, Lista de usuarios

Este requisito tiene un fichero lógico interno de complejidad baja.

Entradas (1):

- DET (Campos): 6, id, Nickname, nombre, apellidos, correo, contraseña, tipo de usuario(VIP o no VIP).
- FTR (Ficheros): 1, Lista de usuarios.

Este requisito tiene una entrada de complejidad baja.

1.2- Borrar Usuario:

Entradas(1):

- DET (Campos): 1, ID usuario.
- FTR (Ficheros): 1, Lista de usuarios.

Este requisito tiene una entrada de complejidad baja.

1.3- Consultar Usuario:

Consultas(1):

- DET (Campos): 1, ID usuario.
- FTR (Ficheros): 3, Lista de usuarios, Lista de enlaces, Lista de valoraciones.

Este requisito tiene una consulta de complejidad baja.

Entradas(1):

- DET: (Campos); 1, ID usuario.
- FTR: (Ficheros): 3, Lista de usuarios, Lista de enlaces, Lista de valoraciones.

Este requisito tiene una entrada de complejidad media.

Salidas(1):

- DET: (Campos): 2, Perfil usuario, enlaces y valoraciones asociadas.
- FTR: (Ficheros): 3, Lista de usuarios, Lista de enlaces, Lista de valoraciones.

Este requisito tiene una salida de complejidad baja.

1.4-Modificar Usuario:

Consultas(1):

- DET (Campos): 6, nickname, nombre, apellidos, correo, contraseña, tipo de usuario(VIP o no VIP) y la privacidad.
- FTR (Ficheros): 1, Lista de usuarios.

Este requisito tiene una consulta de complejidad baja.

Entradas(1):

- DET (Campos): 6, ID_usuario, correo, nombre, apellidos, privacidad, contraseña.
- FTR (Ficheros): 1, Lista de usuarios.

Este requisito tiene una entrada de complejidad baja.

Salidas(1):

- DET (Campos): 6, nickname, nombre, apellidos, correo, contraseña, tipo de usuario(VIP o no VIP) y la privacidad.
- FTR (Ficheros): 1, Lista de usuarios.

Este requisito tiene una salida de complejidad baja.

1.5- Añadir Amigo:

Ficheros lógicos internos(1):

- DET (Campos): 7, id, Nickname, nombre, apellidos, correo, contraseña, tipo de usuario(VIP o no VIP).
- RET (Subgrupos): 1, Lista de usuarios.

Este requisito tiene un fichero lógico interno de complejidad baja.

Entradas (1):

- DET (Campos):2, ID usuario, ID amigo.
- FTR (Ficheros): 1, Lista de usuarios.

Este requisito tiene una entrada de complejidad baja.

1.6- Eliminar Amigo:

Entradas(1):

- DET (Campos): 1, ID_usuario, ID_amigo
- FTR (Ficheros): 1, Lista de usuario.

Este requisito tiene una entrada de complejidad baja.

1.7- Bloquear Usuario:

Ficheros lógicos internos(1):

- DET (Campos): 7, Nickname, nombre, apellidos, correo, contraseña, tipo de usuario(VIP o no VIP) y privacidad.
- RET (Subgrupos): 1, Lista de usuarios.

Este requisito tiene un fichero lógico interno de complejidad baja.

Entradas (1):

- DET (Campos): 2, ID usuario, ID amigo.
- FTR (Ficheros): 1, Lista de usuarios.

Este requisito tiene una entrada de complejidad baja.

1.8- Compartir Enlace:

Ficheros lógicos internos(1):

- DET (Campos):8, id, nickname, nombre, apellido, correo, contraseña, tipo de usuario, id enlace.
- RET (Subgrupos): 1, Lista de enlaces.

Este requisito tiene un fichero lógico interno de complejidad baja.

Entradas (1):

- DET (Campos): 2, nickname, id enlace
- FTR (Ficheros): 2, lista de usuarios, lista de enlaces del usuario.

Este requisito tiene una entrada de complejidad baja.

1.9- Buscar Enlace:

Consultas(1):

- DET (Campos): 1, enlace.
- FTR (Ficheros): 1, Lista de enlaces.

Este requisito tiene una consulta de complejidad baja.

Entradas(1):

- DET (Campos): 2, ID_usuario, ID_enlace.
- FTR (Ficheros): 2, Lista de usuarios, Lista de enlaces.

Este requisito tiene una entrada de complejidad baja.

Salidas(1):

- DET (Campos): 1, Enlace.

- FTR (Ficheros): 1, Lista de enlaces.

Este requisito tiene una salida de complejidad baja.

1.10- Listar Enlaces de un Usuario:

Salidas(1):

- DET (Campos): 6, ID_enlace, valoración media, URL, Propietario, Tags, Público.
- FTR (Ficheros): 1, Lista de enlace.

Este requisito tiene una salida de complejidad baja.

1.11- Listar Valoraciones de un Usuario:

Salidas(1):

- DET (Campos): 5, ID, ID_Enlace, ID_Usuario, puntuación, comentario.
- FTR (Ficheros): 1, Lista de valoraciones.

Este requisito tiene una salida de complejidad baja.

1.12- Pagar VIP:

Ficheros lógicos internos(1):

- DET (Campos): 6, Nickname, nombre, apellidos, correo, contraseña, tipo de usuario(VIP o no VIP).
- RET (Subgrupos): 1, Lista de usuarios.

Este requisito tiene un fichero lógico interno de complejidad baja.

Entradas (1):

- DET (Campos): 1, ID usuario.
- FTR (Ficheros): 1, Lista de usuarios.

Este requisito tiene una entrada de complejidad baja.

1.13- Listar seguidores de un Usuario:

Salidas(1):

- DET (Campos): 1, ID_Usuario.
- FTR (Ficheros): 1, Lista de usuarios.

Este requisito tiene una salida de complejidad baja.

1.14- Listar usuarios a los que sigue otro usuario:

Salidas(1):

- DET (Campos): 1, ID_Usuario.
- FTR (Ficheros): 1, Lista de usuarios.

Este requisito tiene una salida de complejidad BAJA.

2. Módulo Gestor de Enlaces

2.1- Alta Enlace:

Ficheros lógicos internos(1):

- DET: 6, ID_enlace, Valoración media, URL, Propietario, Tags, Público.
- RET: 1, Lista de enlaces.

Este requisito tiene un fichero lógico interno de complejidad baja.

Entradas(1):

- DET: 4, ID_usuario, URL, Tags, Público.
- FTR: 1, Lista de enlaces.

Este requisito tiene una entrada de complejidad baja.

2.2- Baja Enlace:

Entradas(1):

- DET: 1, URL.
- RTF: 1, Lista de enlaces.

Este requisito tiene una entrada de complejidad baja.

2.3- Modificar Enlace:

Consultas(1):

- DET: 3, Tags, Valoración media, tipo.
- RTF: 1, Lista de enlaces.

Este requisito tiene una consulta de complejidad baja.

Entradas(1):

- DET: 1, URL.
- RTF: 1, Lista de enlaces.

Este requisito tiene una entrada de complejidad baja.

Salidas(1):

- DET: 3, Tags, Valoración media, tipo.
- RTF: 1, Lista de enlaces.

Este requisito tiene una salida de complejidad baja.

2.4- Consultar Enlace:

Consultas(1):

- DET (Campos): 1, ID enlace.
- FTR (Ficheros): 1, Lista de enlaces.

Este requisito tiene una consulta de complejidad baja.

Entradas(1):

- DET: 1, ID enlace.
- RTF: 1, Lista de enlaces.

Este requisito tiene una entrada de complejidad baja.

Salidas(1):

- DET: 5, Valoración media, URL, Propietario, Tags, tipo.
- RTF: 1, Lista de enlaces.

Este requisito tiene una salida de complejidad baja.

2.5- Reportar Enlace:

Entradas(1):

- DET: 2, URL, comentario sobre la queja.
- RTF: 1, Lista de enlaces.

Este requisito tiene una entrada de complejidad baja.

2.6- Calcular Valoración:

Salidas(1):

- DET: 1, Valoración media.
- RTF: 1, Lista de enlaces.

Este requisito tiene una salida de complejidad baja.

3. Módulo Gestor de Valoraciones

3.1- Alta Valoración:

Ficheros lógicos internos(1):

- DET (Campos): 5, ID, ID_Enlace, ID_Usuario, puntuación, comentario.
- RET (Subgrupos): 1, Lista de valoraciones.

Este requisito tiene un fichero lógico interno de complejidad baja.

Entradas (1):

- DET (Campos): 5, ID, ID_Enlace, ID_Usuario, puntuación, comentario.
- FTR (Ficheros): 1, Lista de valoraciones.

Este requisito tiene una entrada de complejidad baja.

3.2- Modificar Valoración:

Consultas(1):

- DET (Campos): 2, puntuación, comentario.
- FTR (Ficheros): 1, Lista de valoraciones del usuario.

Este requisito tiene una consulta de complejidad baja.

Entradas(1):

- DET (Campos): 1, ID valoración.
- FTR (Ficheros): 1, Lista de valoraciones del usuario.

Este requisito tiene una entrada de complejidad baja.

Salidas(1):

- DET (Campos): 2, puntuación, comentario.
- FTR (Ficheros): 1, Lista de valoraciones del usuario.

Este requisito tiene una salida de complejidad baja.

3.3- Consultar Valoración:

Consultas(1):

- DET (Campos): 1, ID_valoración.
- FTR (Ficheros): 1, Lista de valoraciones.

Este requisito tiene una consulta de complejidad baja.

Entradas(1):

- DET: (Campos); 1, ID_valoración.
- FTR: (Ficheros): 2, Lista de valoraciones

Este requisito tiene una entrada de complejidad baja.

Salidas(1):

- DET: (Campos): 2, Puntuación, comentario.
- FTR: (Ficheros): 1, Lista de valoraciones.

Este requisito tiene una salida de complejidad BAJA.

3.4- Listar Valoraciones:

Entradas(1):

- DET: (Campos); 1, ID_enlace.
- FTR: (Ficheros): 2, Lista de valoraciones

Este requisito tiene una entrada de complejidad baja.

Salidas(1):

- DET (Campos): 4, Comentario, Puntuación, Usuario, Enlace.
- FTR (Ficheros): 1, Lista de valoraciones de un enlace.

Este requisito tiene una salida y una consulta de complejidad baja.

3.5- Baja Valoración:

Entradas(1):

- DET (Campos): 1, ID_valoración.
- FTR (Ficheros): 1, Lista de valoraciones del usuario.

Este requisito tiene una entrada de complejidad baja.

Puntos de función:

Parámetro	Complejidad	Peso	Total PF
Fichero Lógico Interno	ALTA(0)	15	49
	MEDIA(0)	10	
	BAJA(7)	7	
Fichero de Interfaz Externo	ALTA(0)	10	0
	MEDIA(0)	7	
	BAJA(0)	5	
Entrada	ALTA(0)	6	58
	MEDIA(1)	4	
	BAJA(18)	3	
Salida	ALTA(0)	7	52
	MEDIA(0)	5	
	BAJA(13)	4	
Consulta	ALTA(0)	6	21
	MEDIA(0)	4	
	BAJA(7)	3	
Total			180

4. Estimación usando el modelo COCOMO II

Ahora se va a hacer la estimación del proyecto Linker usando el modelo de estimación paramétrica COCOMO II, en concreto se va a usar el modelo de post arquitectura.

Los resultados de la estimación son sacados de los documentos <<Main>> y <<Phase>> obtenidos con la aplicación de cocomo ii.

5. Modelo de post arquitectura

Es un modelo basado en la contabilización de puntos de función sin ajustar.

6. Estimación del tamaño del software

LINKER

Total Size = 8268

Calculamos KSLOC: $8268/1000 = 8,268$ KSLOC

Total Effort = 30.006312

Coste = $30 * 7800 = 234000$ €

	Programar (%)	Programar (Months)	Esfuerzo (%)	Esfuerzo	Personal
Planes y Requisitos	18.022333%	1.950483	7.000000%	2.100442	1.076883
Diseño de producto	25.011167%	2.706855	17.000000%	5.101073	1.884502
Programación	51.955333%	5.622911	60.966500%	18.293798	3.253439
Integración y Test	23.033500%	2.492820	22.033500%	6.611441	2.652193

GESTOR USUARIOS

Total Size = 4823

Calculamos KSLOC: $4823/1000 = 4,823$ KSLOC

Total Effort = 17.503682

Coste = $17.5 * 7800 = 136500$ €

	Programar (%)	Programar (Months)	Esfuerzo (%)	Esfuerzo	Personal
Planes y Requisitos	18.022333%	1.950483	7.000000%	1.225258	0.628182
Diseño de producto	25.011167%	2.706855	17.000000%	2.975626	1.099293
Programación	51.955333%	5.622911	60.966500%	10.671382	1.897840
Integración y Test	23.033500%	2.492820	22.033500%	3.856674	1.547113

GESTOR ENLACES

Total Size = 1802

Calculamos KSLOC: $1802/1000 = 1,802$ KSLOC

Total Effort = 6.539837

Coste = $6.5 * 7800 = 50700$ €

	Programar (%)	Programar (Months)	Esfuerzo (%)	Esfuerzo	Personal
Planes y Requisitos	18.022333%	1.950483	7.000000%	0.457789	0.234705
Diseño de producto	25.011167%	2.706855	17.000000%	1.111772	0.410725
Programación	51.955333%	5.622911	60.966500%	3.987110	0.709083
Integración y Test	23.033500%	2.492820	22.033500%	1.440955	0.578042

GESTOR VALORACIONES

Total Size = 1643

Calculamos KSLOC: $1643/1000 = 1,643$ KSLOC

Total Effort = 5.962793

Coste = $6 * 7800 = 46800$ €

	Programar (%)	Programar (Months)	Esfuerzo (%)	Esfuerzo	Personal
Planes y Requisitos	18.022333%	1.950483	7.000000%	0.417395	0.213996
Diseño de producto	25.011167%	2.706855	17.000000%	1.013675	0.374484
Programación	51.955333%	5.622911	60.966500%	3.635306	0.646517
Integración y Test	23.033500%	2.492820	22.033500%	1.313812	0.527038

7.Escala de ahorro-gasto

Lo primero que se va a hacer es estimar los porcentajes de cumplimiento de los KPA.

Áreas de procesos claves	Casi siempre (90%)	A menudo (60-90%)	La mitad de las veces (40-60%)	Ocasionalmente (10-40%)	Casi nunca (<10%)	No se aplica	No se conoce
Administración de requerimientos		75%					
Planificación del proyecto software	90%						
Seguimiento y supervisión del proyecto software		75%					
Administración de subcontratos						X	
Aseguramiento de la calidad			50%				
Administración de la configuración			40%				
Objetivo del proceso de organización			60%				
Definición del proceso de organización		80%					
Programa de entrenamiento							X
Administración integrada de software						X	
Ingeniería del producto			50%				
Coordinación entre grupos		70%					
Revisiones por pares			50%				
Administración cuantitativa				30%			

Administración de la calidad		70%					
Prevención de defectos			45%				
Administración de las tecnologías de cambio					10%		
Administración de los procesos de cambio					10%		

Se calcula la madurez del proceso (PMAT) utilizando los % establecidos en la anterior tabla.

$$PMAT = 5 - [(1324/100) * (5/18)] = 1.322$$

Ahora se va a establecer la tabla de escala de factores para el modelo de diseño preliminar.

Factor de Escala Wi	Muy bajo	Bajo	Nominal	Alto	Muy alto	Extra Alto
Precedencia PREC	Completamente sin precedentes (6.20)	Ampliamente sin precedentes (4.96)	Algún precedente (3.72)	Generalmente familiar (2.48)	Ampliamente familiar (1.24)	Completamente familiar (0.00)
Flexibilidad en el desarrollo FLEX	Rigurosa (5.07)	Relajación ocasional (4.05)	Alguna relajación (3.04)	Conformidad en general (2.03)	Alguna conformidad (1.01)	Metas generales (0.00)
Arquitectura/ Resolución del riesgo RESL	Poca (20%) (7.07)	Alguna (40%) (5.65)	Siempre (60%) (4.24)	Generalmente (75%) (2.83)	Principalmente (90%) (1.41)	Completo (100%) (0.00)
Cohesión de Equipo TEAM	Interacciones difíciles (5.48)	Interacciones con alguna dificultad (4.38)	Interacciones básicamente cooperativas (3.29)	Ampliamente cooperativas (2.19)	Altamente cooperativas (1.10)	Interacciones sin fisuras (0.00)

Con los datos establecidos en las anteriores tablas se va a calcular el factor escalar (B), que indica el grado de economía en el esfuerzo a realizar o bien variaciones en la productividad.

$$B = 0,91 + 0,01 * (6.20 + 2.03 + 1.41 + 3.29 + 2.65) = 1,0658$$

El proyecto presenta gastos de escala al ser $B > 1.0$

Esto puede deberse a la cantidad de personal involucrada en el proyecto Linker, ya que para ponerse de acuerdo se tiene que hacer un gasto en comunicaciones como internet, teléfono o asistiendo a reuniones.

8. Cálculo de esfuerzo

Para calcular los meses en función a la persona se aplicará la siguiente fórmula:

$$PM(\text{nominal}) = A * (\text{tamaño})^B$$

A = constante (actualmente, está calibrada en 1.87)

tamaño = cantidad de líneas de código (medido en miles de unidades)

$$PM = 1.322 * (8,268)^{1,0658} = 12,56.$$

$$PM = 12,56 * 0.85 = 10,1.$$

9. Multiplicador de esfuerzo

	Symbol	VL	L	N	H	VH	XH
RELY	EM1	0.82	0.92	1	1.1	1.26	
DATA	EM2		0.9	1	1.14	1.28	
CPLX	EM3	0.73	0.87	1	1.17	1.3	1.74
RUSE	EM4		0.95	1	1.07	1.15	1.24
DOCU	EM5	0.81	0.91	1	1.11	1.23	
TIME	EM6			1	1.11	1.29	1.63
STOR	EM7			1	1.05	1.17	1.46
PVOL	EM8		0.87	1	1.15	1.3	
ACAP	EM9	1.42	1.19	1	0.85	0.71	
PCAP	EM10	1.34	1.15	1	0.88	0.79	
PCON	EM11	1.29	1.12	1	0.9	0.81	
APEX	EM12	1.22	1.1	1	0.88	0.81	
PLEX	EM13	1.19	1.09	1	0.91	0.85	
LTEX	EM14	1.2	1.09	1	0.91	0.84	
TOOL	EM15	1.17	1.09	1	0.9	0.78	
SITE	EM16	1.22	1.09	1	0.93	0.86	0.8
SCED	EM17	1.43	1.14	1	1	1	

$$PM_a = PM_n * (\pi EM_i)$$

πEM_i = Multiplicación de los 17 valores del driver calculados previamente.

$$\pi EM_i = 1.00$$

$$*0.90*0.87*0.95*1.00*1.00*1.00*1.00*1.00*1.00*1.22*1.09*1.00*1.00*1.00*0.86*1.00 = 0.85.$$

10. Cálculo de tiempo de desarrollo

Para realizar este cálculo se ha utilizado la herramienta Cocomo II, y se han obtenido los siguientes resultados:

Esfuerzo = 10.1 Personas-mes.

Planificación = 8.2 Meses.

11. Plan de presupuesto

Personal:

Puntos de Función	Esfuerzo Personas / Mes	Duración en meses	Coste total
180	10.1	8.2	234000 €

Inventario:

Recurso	Coste / Unidad	Nº de unidades	Coste total
PCs	750€ / ud.	9 uds.	6750€
Alquiler local	1300€ / mes	9 uds	11700€
Servidor	9100€ /año	1 ud.	9100€

Otros gastos:

Recurso	Coste / Unidad	Nº de unidades	Coste total
Actividades de prevención de errores	70€ / hora	13 uds	910€
Desplazamientos	40€ / mes	5 uds	200€

Total:

Recurso	Total
Personal	234000 €
Inventario	27550€
Otros gastos	1110€
TOTAL	262660€

12. Acrónimos

AFP: Puntos de función del proyecto.
AF: Factor de ajuste.
FP: Puntos de función sin ajustar del proyecto.
KPA: Key process Areas
PREC: Precedentes.
FLEX: Flexibilidad.
RESL: Resolución de riesgos.
TEAM: Cohesión de equipo.
PMAT: Madurez del proceso.
PM: Meses-persona.
MM: Esfuerzo Medio.
PF: Puntos de Función.
RCPX: Fiabilidad del producto y complejidad
RUSE: Reutilización requerida.
PDIF: Dificultad de la Plataforma.
PERS: Capacidad Personal.
PREX: Experiencia personal.
FCIL: Facilidades.
SCED: Ajuste a la planificación.
RELY: nivel de confiabilidad para realizar la función esperada.
DATA: medida del volumen de datos.
CPLX: complejidad del producto.
RUSE: grado de reusabilidad requerida para otras aplicaciones.
DOCU: documentación requerida de acuerdo al ciclo de vida.
TIME: restricciones del tiempo de ejecución.
STOR: restricciones del almacenamiento principal.
PVOL: volatilidad de la plataforma HW-SW de base.
ACAP: capacidad de los analistas para trabajar en equipo.
PCAP: capacidad de los programadores para trabajar en equipo.
AEXP: experiencia en las aplicaciones.
PEXP: experiencia en la plataforma.
LTEX: experiencia en lenguajes y herramientas.
PCON: continuidad del personal.
TOOL: uso de herramientas de software.
SITE: desarrollo en sitios múltiples.
SCED: restricciones en más/menos impuestas al plan del proyecto.