



Gestión de Proyectos

Estimación del esfuerzo en
desarrollo de software



Introducción

- Queremos conocer el coste de desarrollar un sistema (esfuerzo, dinero, etc.)
- Queremos conocer el coste pronto
- Una vez conocido el esfuerzo, hay que asignarlo a las tareas y personas

Métodos para la estimación

Basados en ...	Método
Experiencia	Juicio experto: Puro
	Juicio experto: Delphi, Wideband Delphi
	Estimación multipunto
	Analogía
	Distribución de la utilización de recursos en el ciclo de vida
Recursos	Parkinson
Mercado	Precio para vender
Componentes del producto	Bottom-up
	Top-Down
Algoritmos	Puntos de Función
	COCOMO II



Métodos basados en la experiencia

■ Juicio experto – Puro

- ☐ Un experto estudia la especificación y hace su estimación
- ☐ Se basa fundamentalmente en los conocimientos del experto
- ☐ Si desaparece el experto, la empresa deja de estimar

Métodos basados en la experiencia

■ Juicio experto – Delphi

- ☐ Se dan las especificaciones a un grupo de expertos
- ☐ El grupo debate
- ☐ Cada uno hace su propia estimación y la remite al coordinador
- ☐ El coordinador las revisa y si divergen mucho, las vuelve a enviar al grupo de expertos (anónimas)
- ☐ Se repite el proceso hasta que las estimaciones convergen de forma razonable

Métodos basados en la experiencia

- Juicio experto – Wideband Delphi

- Siguiendo el procedimiento Delphi, los miembros de las estimaciones más optimista y pesimista tienen que explicar (de forma anónima) el porqué.

Métodos basados en la experiencia

■ Estimación multipunto

- Se calcula a partir de una media ponderada de estimaciones: la optimista, la media y la pesimista

$$Estimación = \frac{Optimista + 4xMedia + Pesimista}{6}$$

Métodos basados en la experiencia

■ Analogía

- Consiste en comparar las especificaciones del proyecto con la de otros anteriores
- Hay que tener en cuenta diversos factores:
 - Tamaño: ¿mayor o menor?
 - Complejidad: ¿más complejo de lo habitual?
 - Usuarios: Mayor número de usuarios -> mayor complejidad
 - Otros factores: SO, HW, entorno, personal, etc.

Métodos basados en la experiencia

- Distribución de la utilización de los recursos en el ciclo de vida
 - Normalmente en una organización la estructura de coste es similar entre distintos proyectos
 - Si volvemos a realizar un proyecto con las mismas fases es de esperar que la distribución sea similar
 - Ej. Planificación (20%), Análisis (5%), Diseño (20%)....

Método basado en los recursos

■ Parkinson

- La estimación consiste en ver de cuánto personal y durante cuánto tiempo se dispone
- “El trabajo se expande hasta consumir todos los recursos disponibles” → Ley de Parkinson
- Ejemplo:
 - Si tenemos 12 meses para desarrollar el proyecto y somos 5 personas → el esfuerzo es 60 personas-mes



Método basado en el mercado

■ Precio para vender

- ☐ Lo importante es conseguir el contrato
- ☐ El precio se fija en función de lo que el cliente está dispuesto a pagar
- ☐ Si se usa en conjunción con otros métodos puede ser aceptable
- ☐ Peligroso como único método de estimación



Método basado en los componentes

■ Bottom-Up

- ☐ Se descompone el proyecto en unidades lo menores posibles
- ☐ Se estima cada unidad y se calcula el coste total

■ Top-Down

- ☐ Se estima el proyecto completo
- ☐ Se divide la estimación entre las distintas tareas



Métodos algorítmicos

- Utilizan fórmulas para producir estimaciones.

Veremos:

- ☐ Puntos de Función
- ☐ COCOMO II

Puntos de Función

- Métrica de los puntos de función
 - Es una métrica que se puede aplicar en las primeras fases de desarrollo
 - Se basa en características fundamentalmente “externas” de la aplicación a desarrollar
 - Mide dos tipos de características
 - Elementos de función : entradas, salidas, ficheros, etc.
 - Factores de Complejidad

Puntos de Función

■ Pasos a seguir:

1. Identificación de elementos y clasificación
2. Cálculo de los puntos de función no ajustados (PFNA)
3. Cálculo del factor de ajuste (FA)
4. Cálculo de los puntos de función ajustados (PFA)

Elementos de función

■ Identificación de los elementos de función:

☐ Ficheros Lógicos Internos (FLI)

☐ Ficheros de Interfaz Externos (FIE)

} Relacionados con los datos

☐ Entradas Externas (EE)

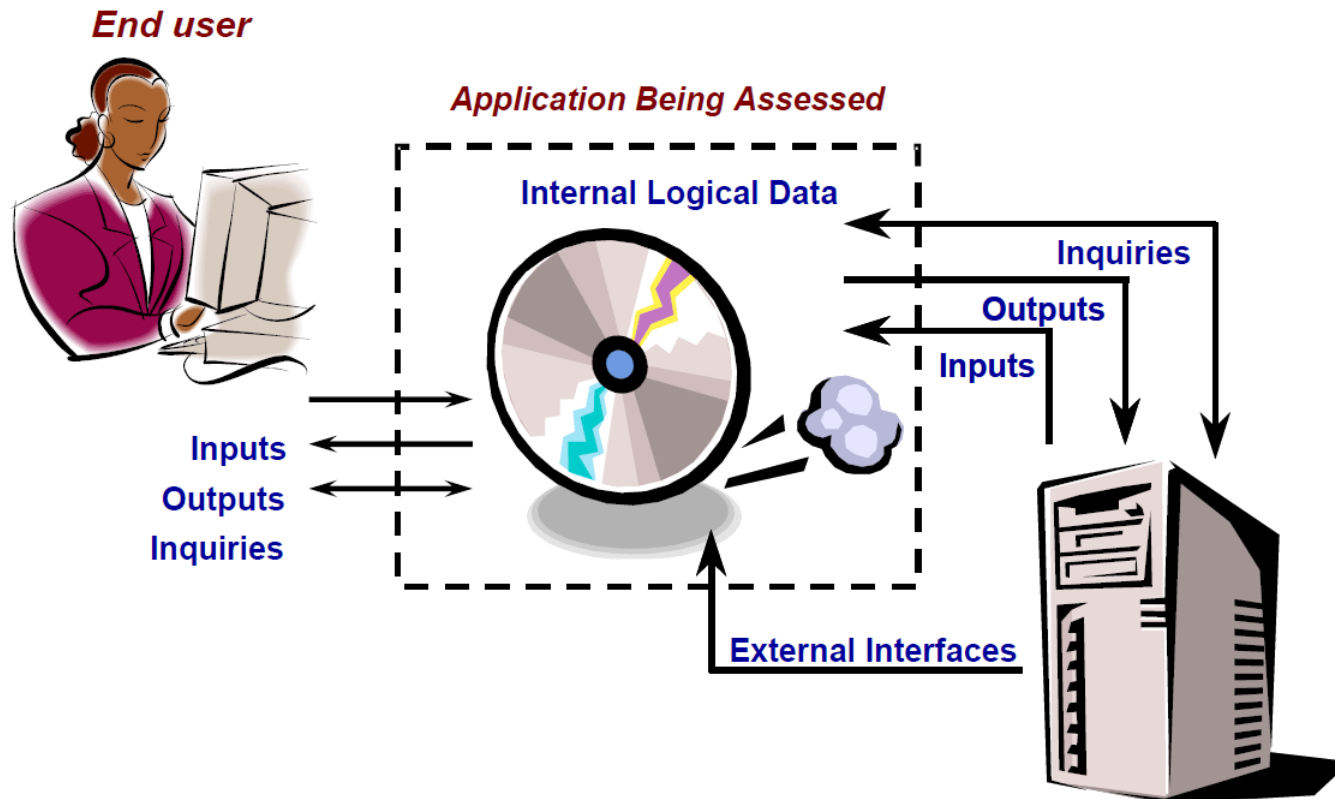
☐ Salidas Externas (SE)

☐ Consultas Externas (CE)

} Relacionados con las transacciones (proceso elemental: unidad mínima de actividad que tiene sentido para el usuario. Ej: realizar una reserva, modificar reserva, borrar reserva)

Elementos de función

- Ficheros Lógicos Internos (FLI)
- Ficheros de Interfaz Externos (FIE)
- Entradas Externas (EE)
- Salidas Externas (SE)
- Consultas Externas (CE)



Elementos de función

■ Ficheros Lógicos Internos (FLI)

- Grupo de datos relacionados identificados por el usuario y mantenidos por el sistema
- El grupo de datos es mantenido mediante un proceso elemental
- Ejemplos
 - Tablas de una BD*
 - Ficheros*

*Veremos cuando son FLI y cuándo no

Elementos de función

■ Ficheros de Interfaz Externos (FIE)

- ☐ Grupo de datos relacionados identificados por el usuario y NO mantenidos por el sistema
- ☐ Ficheros generados y mantenidos por otra aplicación
- ☐ Es un Fichero Lógico Interno de la otra aplicación

Elementos de función

- Entradas Externas (EE)
 - Procesos elementales que hacen llegar datos desde el exterior (usuario, otra aplicación)
 - Sólo desde el exterior hacia el interior
 - Siempre se actualiza un FLI
 - Ejemplos:
 - Pantallas para la entrada de datos
 - Lecturas de código de barras, voz, etc.

Elementos de función

■ Salidas Externas (SE)

- ☐ Procesos elementales que hacen llegar datos al exterior (usuario, otra aplicación)
- ☐ Sólo desde el interior al exterior
- ☐ Deben procesar datos (es decir, no únicamente mostrarlos)
- ☐ Ejemplos (si cumplen lo anterior):
 - Pantallas con datos / información
 - Listados
 - Gráficos
 - Transferencias de datos a otras aplicaciones

Elementos de función

■ Consultas Externas (CE)

- Procesos elementales formados por una combinación de entrada y salida que...
 - No genera datos derivados
 - No modifica ningún Fichero Lógico Interno
- Ej: Obtener alumnos matriculados en GP

Puntos de Función No Ajustados

- Se estudia cada elemento de función (cada tabla, cada entrada...) identificando sus componentes:
 - Sus atributos, mensajes de error, opciones...
 - Únicamente los que tienen sentido desde el punto de vista del usuario (ej: si una tabla tiene como clave un autoincremental que no significa nada, no se tiene en cuenta)
- Se calculan los PFNA (Puntos de Función No Ajustados) de cada elemento de función en las siguientes tablas.

Puntos de Función No Ajustados

Complejidad Ficheros Lógicos Internos	Nº de Campos o atributos		
	1-19 Atributos	20-50 Atributos	51 ó más Atributos
	1 Registro lógico Baja 7	Baja 7	Media 10
	2-5 Registros lógicos Baja 7	Media 10	Alta 15
6 ó más Registros lógicos	Media 10	Alta 15	Alta 15

Puntos de Función No Ajustados

□ Imaginemos que la aplicación que gestiona las matriculas utiliza las siguientes tablas:

- Asignatura (codigo, nombre, créditos)
- Alumno (DNI, nombre)
- DireccionAlumno(DNI, calle, portal, piso, ciudad, CP)
- NotaBachillerAlumno(DNI, notaMedia)
- Matricula(DNI, codigoAsignatura)

Son 5 tablas pero son 5 FLI?

Puntos de Función No Ajustados

☐ Son 5 tablas pero...

- Asignatura (codigo, nombre, créditos)

- Alumno (DNI, nombre)

- DireccionAlumno(DNI, calle, portal, piso, ciudad, CP)

- NotaBachillerAlumno(DNI, notaMedia)

- Matricula(DNI, codigoAsignatura)

☐ Por tanto tendremos 3 FLI: Asignatura, DatosDelAlumno, Matricula

- DatosDelAlumno tiene 3 registros lógicos: Alumno, DireccionAlumno y NotaBachillerAlumno

- Asignatura tiene un único registro lógico

- Matricula tiene un único registro lógico

Pueden
verse como
un grupo

Puntos de Función No Ajustados

Complejidad Ficheros de Interfaz Externos	Nº de Campos o atributos		
	1-19 Atributos	20-50 Atributos	51 ó más Atributos
1 Registro lógico	Baja 5	Baja 5	Media 7
2-5 Registros lógicos	Baja 5	Media 7	Alta 10
6 ó más Registros lógicos	Media 7	Alta 10	Alta 10

Puntos de Función No Ajustados

Complejidad Entradas	Nº de Campos o atributos de la entrada		
	1-4 Atributos	5-15 Atributos	16 ó más Atributos
1 Ficheros accedidos	Baja 3	Baja 3	Media 4
2 Ficheros accedidos	Baja 3	Media 4	Alta 6
3 ó más Ficheros accedidos	Media 4	Alta 6	Alta 6

Puntos de Función No Ajustados

Nuevo cliente

Nombre: NIF:

Dirección:

Calle:

Nº: Piso: Puerta:

CP: Ciudad:

Telefono: Email:

Puntos de Función No Ajustados

Complejidad Salidas	Nº de Campos o atributos de la salida		
	1-5 Atributos	6-19 Atributos	20 ó más Atributos
1 Ficheros accedidos	Baja 4	Baja 4	Media 5
2 ó 3 Ficheros accedidos	Baja 4	Media 5	Alta 7
4 ó más Ficheros accedidos	Media 5	Alta 7	Alta 7

Puntos de Función No Ajustados

Complejidad Consultas externas	Nº de Campos o atributos		
	1-5 Atributos	6-19 Atributos	20 ó más Atributos
1 Ficheros accedidos	Baja 3	Baja 3	Media 4
2 ó 3 Ficheros accedidos	Baja 3	Media 4	Alta 6
4 ó más Ficheros accedidos	Media 4	Alta 6	Alta 6

Puntos de Función No Ajustados

■ Casos especiales a conocer*:

- ☐ Pulsar botón/tecla: si origina un proceso, contarlo como atributo
- ☐ Checkbox: cada checkbox cuenta como un atributo
- ☐ RadioButton: el conjunto entero cuenta como un atributo
- ☐ ComboBox: si los datos se cargan de un FLI/FIE contarlo como consulta externa de dos atributos
- ☐ Mensaje de error: cuenta como un atributo

* Hay más casos pero no los trabajaremos

Calcular el Factor de Ajuste (FA)

- Se calcula basado en las catorce características cualitativas generales del software
- A cada característica se le asigna un valor
 - ☐ 0 no está presente o no influye
 - ☐ 1 influencia incidental
 - ☐ 2 influencia moderada
 - ☐ 3 influencia promedio
 - ☐ 4 influencia significativa
 - ☐ 5 influencia fuerte

Calcular el Factor de Ajuste (FA)

■ Las catorce características son:

- | | |
|---|--------------------------------------|
| 1. Comunicación de datos | 8. Actualización en línea |
| 2. Procesamiento distribuido | 9. Procesamiento complejo |
| 3. Prestaciones | 10. Reusabilidad |
| 4. Facilidad de Configuración | 11. Facilidad de instalación |
| 5. Tasa de transacciones | 12. Facilidad operacional |
| 6. Entrada de datos en línea | 13. Adaptabilidad a múltiples sitios |
| 7. Diseño Eficiencia requerida
del usuario final | 14. Facilidad de cambio |

■ Se evalúan en términos de su grado de influencia

6. Entrada de datos en línea

- La toma de datos de la aplicación se realiza en línea. Los posibles valores para este atributo son:
 - 0 ► Todas las transacciones son tratadas por lotes.
 - 1 ► Entre el 1 y el 7% de las funciones son entradas interactivas de datos.
 - 2 ► Entre el 8 y el 15% de las funciones son entradas interactivas de datos.
 - 3 ► Entre el 16 y el 23% de las funciones son entradas interactivas de datos.
 - 4 ► Entre el 24 y el 30% de las funciones son entradas interactivas de datos.
 - 5 ► Más del 30% de las funciones son entradas interactivas de datos.

Calcular el Factor de Ajuste (FA)

- Una vez obtenido el valor de los atributos y sumados se obtiene una cifra comprendida entre 0 y 70, a partir de la cual se obtendrá el factor de ajuste, según la fórmula:

$$FA = 0,65 + (0,01 * SVA)$$

- Siendo:

- ☐ FA: Factor de ajuste
- ☐ SVA: Suma de los valores de los 14 atributos.

Calcular los puntos de función finales

- Por último, se ajustan los Puntos Función mediante la siguiente fórmula:

$$PFA = PFNA * FA$$

□ Siendo:

- PFA: Puntos Función ajustados
- PFNA: Puntos Función no ajustados (suma total)
- FA: Factor de ajuste (calculado anteriormente).

Método de los Puntos de Función

■ Ya tenemos los PFA, ¿y ahora qué?

Nombre del Proyecto	PFA	Lenguaje	Esfuerzo (horas/persona)
Deskontplan	300	PHP	220
Tringes	210	Java	180
Júpiter	500	Java	350
Mercurio	430	C++	390

☐ $\text{Esfuerzo} = \text{PFA} * \text{Promedio (Lenguaje)}$

☐ ¿Otros factores? SO, HW, BBDD, equipo de proyecto, “Ojo clínico” del experto

COCOMO II

- COnstructive COst Model
- Es el modelo de estimación de costes más utilizado
- En 1995 se publicó la versión COCOMO II
- Existen múltiples variantes: COSCOMO, COINCOMO, DBA COCOMO,...

COCOMO II

■ Objetivos:

- ☐ Desarrollar un modelo de estimación de tiempo y de coste del software de acuerdo con los ciclos de vida más usados
- ☐ Construir una BD de proyectos de SW que permita la calibración continua del modelo y así incrementar la precisión de la estimación

COCOMO II

- COCOMO II está compuesto por 3 modelos:
 - Composición de Aplicación (ACM)
 - Diseño Temprano (EDM)
 - Post-Arquitectura (PAM)
- Los 3 modelos se adaptan al tipo y a la cantidad de información disponible en cada etapa del ciclo de vida

COCOMO II. ACM

- Indicado en la etapa de planificación
- Cuando se tiene un esbozo de cómo va a ser el sistema
- Para proyectos contruidos con herramientas de construcción de interfaces gráficos
- Utiliza Puntos Objeto

COCOMO II. ACM

■ Puntos Objeto

- 1º.- Estimar la cantidad de pantallas, informes y componentes 3GL que contendrá la aplicación
- 2º.- Clasificar cada instancia de un objeto según su nivel de complejidad

COCOMO II. ACM

<i>Pantallas</i>	Número de Tablas de Datos		
Número de vistas	<4	<8	>=8
<3	Sencillo	Sencillo	Medio
3-7	Sencillo	Medio	Difícil
>=8	Medio	Difícil	Difícil

COCOMO II. ACM

<i>Informes</i>	Número de Tablas de Datos		
Número de secciones	<4	<8	>=8
0 ó 1	Sencillo	Sencillo	Medio
2 ó 3	Sencillo	Medio	Difícil
>=4	Medio	Difícil	Difícil

COCOMO II. ACM

■ Puntos Objeto

- 3º.- Dar el peso a cada objeto según el nivel de complejidad

<i>Tipo de Objeto</i>	Sencillo	Medio	Difícil
Pantallas	1	2	3
Informes	2	5	8
3GL			10

COCOMO II. ACM

■ Puntos Objeto

- 4º.- Determinar la cantidad de Puntos Objeto, sumando todos los pesos de las instancias (PO)
- 5º.- Estimar el porcentaje de código a reusar. Calcular los Nuevos Puntos Objeto a desarrollar (NPO).

$$NPO = \frac{(PO \times (100 - \%reusado))}{100}$$

COCOMO II. ACM

■ Puntos Objeto

- 6º.- Calcular el ratio de la productividad según la siguiente tabla:

Experiencia de los desarrolladores	Muy Baja	Baja	Normal	Alta	Muy Alta
Experiencia con las herramientas	Muy Baja	Baja	Normal	Alta	Muy Alta
PROD (ratio)	4	7	13	25	50

$$\text{PROD}(\text{total}) = (\text{PROD}(\text{exp. Trabajadores}) + \text{PROD}(\text{exp. Herramientas}))/2$$

COCOMO II. ACM

■ Puntos Objeto

- 7º.- El esfuerzo (en meses-persona) viene dado por:

$$PM = \frac{NPO}{PROD}$$



COCOMO II. EDM

- Se utiliza en las primeras etapas del proyecto
- Cuando se conoce muy poco sobre el tamaño del producto, la plataforma y el personal
- Basado en Puntos de Función No Ajustados
- Una vez calculados los PFNA los convierte a líneas de código
- Utiliza 7 conductores/parámetros (drivers) de ajuste

COCOMO II. EDM

■ Procedimiento

- 1º.- Calcular los PFNA
- 2º.- Convertir los PFNA a miles de líneas de código
 - $(KSLOC = PFNA * Factor / 1000)$

Lenguaje	Factor
Ada	71
Basic	91
C	128
C++	29
Cobol 85	100
Ensamblador	320
Visual Cobol	29
Java	23
Lisp	64
Pascal	91
Prolog	64

COCOMO II. EDM

■ Procedimiento

□ 3º.- El esfuerzo viene dado por:

$$PM_{Nominal} = A \times (Tamaño)^B$$

□ Donde:

- A= Constante de calibración (2,45)
- Tamaño= KSLOC
- $B = 0.91 + 0.01 \times \sum Fe_j$ (j= 1..5) (Fe=Factor de escala)

COCOMO II. EDM

■ B es el factor exponencial de escala

- Si $B < 1.0$ significa que si un proyecto aumenta el doble su tamaño, el esfuerzo es menos del doble (es decir, aumenta la productividad)
- Si $B = 1.0$ es un modelo lineal. Es lo habitual en proyecto pequeños
- Si $B > 1.0$, si el esfuerzo necesario aumenta más que el tamaño (es decir, disminuye la productividad)

COCOMO II. EDM

■ Factores de escala

PREC	Experiencia previa de la organización con este tipo de proyectos.
FLEX	Refleja el grado de flexibilidad en el proceso de desarrollo.
RESL	Refleja la amplitud de análisis de riesgo que se lleva a cabo.
TEAM	Refleja la relación entre los miembros del equipo de desarrollo.
PMAT	Refleja la madurez del proceso de la organización.

COCOMO II. EDM

■ Factores de escala

Factores de Escala (SF _j)	Muy Bajo	Bajo	Nominal	Alto	Muy Alto	Extra Alto
PREC	Completamente sin precedentes	Prácticamente sin precedentes	Casi sin precedentes	Algo familiar	Muy familiar	Completamente familiar
FLEX	Riguroso	Relajación ocasional	Algo de relajación	Conformidad general	Algo de conformidad	Metas generales
RESL*	Poco (20%)	Algo (40%)	A menudo (60%)	Generalmente (75%)	En su mayor parte (90%)	Por completo (100%)
TEAM	Interacciones muy difíciles	Algo de dificultad en las interacciones	Interacciones básicamente cooperativas	Bastante cooperativo	Altamente cooperativo	Completas interacciones
PMAT	Peso medio de respuestas "Sí" para el cuestionario de Madurez CMM					

COCOMO II. EDM

■ Cuestionario de Madurez CMM

Key Process Areas	Almost Always (>90 %)	Frequently (60-90 %)	About Half (40-60 %)	Occasional ly (10-40 %)	Rarely If Ever (<10 %)	Does Not Apply	Don't Know
1 Requirements Management	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2 Software Project Planning	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3 Software Project Tracking and Oversight	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4 Software Subcontract Management	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
5 Software Quality Assurance	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
6 Software Configuration Management	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
7 Organization Process Focus	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
8 Organization Process Definition	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
9 Training Program	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
10 Integrated Software Management	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
11 Software Product Engineering	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
12 Intergroup Coordination	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
13 Peer Reviews	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
14 Quantitative Process Management	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
15 Software Quality Management	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
16 Defect Prevention	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
17 Technology Change Management	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
18 Process Change Management	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

COCOMO II. EDM

■ Factores de escala

	Muy Bajo	Bajo	Nominal	Alto	Muy Alto	Extra Alto
Precedentes (PREC)	6,2	4,96	3,72	2,48	1,24	0
Flexibilidad (FLEX)	5,07	4,05	3,04	2,03	1,01	0
Arquitectura/resolución del riesgo (RESL)	7,07	5,65	4,24	2,83	1,41	0
Cohesión del Equipo (TEAM)	5,48	4,38	3,29	2,19	1,10	0
Madurez del Proceso (PMAT)	7,80	6,24	4,68	3,12	1,56	0

COCOMO II. EDM

■ Procedimiento

□ 4º.- El esfuerzo ajustado será:

$$PM_{Ajustado} = PM_{Nominal} \times \prod_{i=1}^7 FA_i$$

□ Donde:

■ FA = Factores de ajuste

COCOMO II. EDM

■ Factores de ajuste

Factor	Descripción
RCPX	Fiabilidad y complejidad del producto
RUSE	Requerimientos de reusabilidad
PDIF	Dificultad de la plataforma
PERS	Capacidad del personal
PREX	Experiencia del personal
FCIL	Facilidades para el desarrollo
SCED	Esfuerzo de calendario

COCOMO II. EDM

■ Factores de ajuste

	Extra Bajo	Muy Bajo	Bajo	Nominal	Alto	Muy Alto	Extra Alto
RCPX	0.73	0.81	0.98	1	1.3	1.74	2.38
RUSE	-	-	0.95	1	1.07	1.15	1.24
PDIF	-	-	0.87	1	1.29	1.81	2.61
PERS	2.12	1.62	1.26	1	0.83	0.63	0.5
PREX	1.59	1.33	1.12	1	0.87	0.71	0.62
FCIL	1.43	1.30	1.10	1	0.87	0.73	0.62
SCED	-	1.43	1.14	1	1	1	-



COCOMO II. PAM

- Es el modelo más detallado
- Se usa cuando se ha desarrollado por completo la arquitectura del proyecto
- Tiene 17 factores de ajuste

COCOMO II. PAM

■ Procedimiento

- 1º - 3º.- Los pasos similares al modelo EDM
- 4º.- El esfuerzo ajustado será:

$$PM_{Ajustado} = PM_{Nominal} \times \prod_{i=1}^{17} FA_i$$

COCOMO II. PAM

■ Factores de ajuste

Factor EDM	Factores PAM
RCPX	RELY, DATA, CPLX, DOCU
RUSE	RUSE
PDIF	TIME, STOR, PVOL
PERS	ACAP, PCAP, PCON
PREX	AEXP, PEXP, LTEX
FCIL	TOOL, SITE
SCED	SCED

COCOMO II. PAM

- Los factores de coste ahora son 17, y son de cuatro clases:
 - Producto: se refieren a las características requeridas del producto de software a desarrollar.
 - Plataforma: son restricciones impuestas sobre el software o la plataforma de hardware.
 - Personal: son multiplicadores que toman en cuenta la experiencia y capacidades de las personas que trabajan en el proyecto.
 - Proyecto: se refieren a las características particulares del proyecto de desarrollo de software.

COCOMO II. PAM

- PRODUCTO:

- **(RELY). Fiabilidad Requerida de Software**

Esta es la medida de hasta qué punto el software debe realizar su función esperada durante un periodo de tiempo. Si el efecto de un fracaso es sólo una molestia ligera entonces RELY es Bajo. Si un fallo arriesgase vidas humanas entonces RELY es Muy Alto.

- **(DATA). Medida del Volumen de Datos**

Esta medida intenta capturar lo que afecta en el desarrollo del producto unos requerimientos de datos grandes. La medida se determina calculando D/P. La razón por la que es importante considerar el tamaño de la Base de Datos es por el esfuerzo necesario para generar datos de prueba que se usarán para ejecutar el programa.

- **(CPLX). Complejidad del Producto**

La complejidad se decide en 5 áreas: Funcionamiento de control, Funcionamiento computacional, Funcionamiento de Dispositivos dependientes, Funcionamiento del sector de datos y Funcionamiento del Gestor de Interfaz de Usuario.

COCOMO II. PAM

- PRODUCTO:
- **(RUSE). Reutilización Requerida**

Es el esfuerzo adicional necesario para construir componentes pensados para ser reutilizados en proyectos presentes ó futuros. Este esfuerzo se consume en crear un diseño más genérico del software, documentación más detallada y pruebas más extensas, para asegurar que los componentes están listos para utilizar en otras aplicaciones.

- **(DOCU). Documentación Asociada a las Necesidades del Ciclo de Vida**

Nivel de documentación requerida. En COCOMO II la escala de medida para el driver de coste se evalúa en términos de la adecuación de la documentación del proyecto a las necesidades de su ciclo de vida. La escala de valores va desde Muy Bajo (muchas necesidades del ciclo de vida sin cubrir) hasta Muy Alto (excesiva para las necesidades del ciclo de vida).

COCOMO II. PAM

- PLATAFORMA:
- **(TIME). Restricción del Tiempo de Ejecución**

Las medidas se expresan en términos de porcentaje de tiempo de ejecución disponible que se espera que sea usado por el sistema que consume el recurso de tiempo de ejecución. Los valores van desde nominal, menos del 50% de recursos de tiempo de ejecución utilizados, hasta Extra Alto, 95% del recurso de tiempo de ejecución consumido.

- **(STOR). Restricción de Almacenamiento Principal**

Esta medida representa el grado de restricción de almacenamiento principal impuesto a un sistema ó subsistema software. Los valores van desde nominal, menos que el 50%, a Extra Alto, 95%.

- **(PVOL). Volatilidad de la Plataforma**

“Plataforma” se usa aquí para significar la complejidad del Hardware y Software que el producto software necesita para realizar sus tareas. Si el software a desarrollar es un sistema operativo, entonces la plataforma es el hardware del ordenador. Si se desarrolla un Gestor de Base de Datos, entonces la plataforma es el hardware y el sistema operativo ETC. Los valores van desde Bajo donde cada 12 meses hay un cambio importante, hasta Muy Alto, donde hay un cambio importante cada 2 semanas.

COCOMO II. PAM

- PERSONAL:

- **(ACAP). Habilidad del Analista**

Los analistas son personal que trabaja en los requisitos de diseño de alto nivel y en diseño detallado. Los atributos principales que deben considerarse en esta medida son la habilidad de análisis y diseño, la eficiencia y minuciosidad y la habilidad para comunicar y cooperar. La medida no debe considerar el nivel de experiencia del analista, eso se mide con AEXP. Los analistas que caen en el 90 avo percentil se evalúan como Muy Alto

- **(PCAP). Habilidad del Programador**

La evaluación debe basarse en la capacidad de los programadores como un equipo , más que individualmente. La habilidad del programador no debe considerarse aquí, eso se mide con AEXP. Unos valores Muy Bajos del equipo de programadores es el 15 avo percentil y Muy Alto en el 90 avo percentil.

- **(AEXP). Experiencia en las Aplicaciones**

Esta medida depende del nivel de experiencia en aplicaciones del equipo de proyecto al desarrollar sistemas ó subsistemas software. Un valor muy bajo para experiencia en aplicaciones es menor que 2 meses. Un valor muy alto es por experiencia de 6 años ó más.

COCOMO II. PAM

- PERSONAL:
- **(PEXP). Experiencia en la Plataforma**

El modelo postarquitectónico amplía la influencia en productividad de PEXP, reconociendo la importancia de entender el uso de plataformas más poderosas, incluyendo más interfaces gráficos de usuario, redes y capacidades de middleware distribuido.

- **(LTEX). Experiencia en la Herramienta y en el Lenguaje**

Esta es una medida del nivel de experiencia en el lenguaje de programación y en la herramienta software del equipo de proyecto que desarrolla el sistema ó subsistema software. Tener una experiencia de menos de 2 meses da un valor Bajo. Si es de 6 ó más años el valor es Muy Alto.

- **(PCON). Continuidad del Personal**

La escala de valores de PCON se mide en términos del movimiento de personal del proyecto anualmente: desde 3%, Muy Alto, hasta el 48%, Muy Bajo.

COCOMO II. PAM

- PROYECTO
- **(TOOL). Uso de Herramientas Software**

Las herramientas software han mejorado significativamente desde los proyectos de los 70 usados para calibrar COCOMO. Los valores para la herramienta van desde edición y código simple, Muy Bajo, hasta herramientas integradas de gestión del ciclo de vida, Muy Alto.

- **(SITE). Desarrollo Multilugar**

Dada la frecuencia creciente de desarrollos multilugar e indicaciones de que los efectos del desarrollo multilugar son significantes. Determinar la medida del driver de coste incluye el cálculo y la medida de 2 factores: Localización del lugar y soporte de comunicación.

- **Requerido (SCED). Calendario de Desarrollo**

Este valor mide las restricciones de horario impuestas al equipo de proyecto que desarrolla el software. Los valores se definen en términos de porcentaje de aceleración ó alargamiento sobre el calendario respecto de un calendario nominal para un proyecto que requiere una cantidad de esfuerzo dada. Una compresión del calendario del 74% se evalúa como Muy Bajo. Un alargamiento del 160% se valora como Muy Alto.