COCOMO CVs

# Ajuste de punto de función

## Fi:

¿Requiere el sistema copias de seguridad y de recuperación fiables? Moderado 2

¿Se requiere comunicación de datos? Moderado 2

¿Existen funciones de procesamiento distribuido? Moderado 2

¿Es crítico el rendimiento? Incidental 1

¿Se ejecutaría el sistema en un entorno operativo existente y fuertemente utilizado? Moderado 2

¿Requiere el sistema entrada de datos interactiva? Moderado 2

¿Requiere la entrada de datos interactiva que las transacciones de entrada se lleven a cabo sobre múltiples pantallas u operaciones?

Moderado 2

¿Se actualizan los archivos maestros de forma interactiva? Incidental 1

¿Son complejas las entradas, las salidas, los archivos o las peticiones? Incidental 1

¿Es complejo el procesamiento interno? Moderado 2

¿Se ha diseñado el código para ser reutilizable? Moderado 2

¿Están incluidas en el diseño la conversión y la instalación?

Moderado 2

¿Se ha diseñado el sistema para soportar múltiples instalaciones en diferentes organizaciones?

Sin influencia 0

¿Se ha diseñado la aplicación para facilitar los cambios y para ser fácilmente utilizada por el usuario?

Medio 3

Sumatoria de Fi=24

# Parámetros de medición:

Número de entradas de usuario: 21 x 3 = 63 Número de salidas de usuario: 2 x 4 = 8 Número de peticiones de usuario: 5 x 3 = 15 Número de archivos: 5 x 7 = 35

Número de interfaces externas: 0 x 5 = 0

Cuenta total: 121

FP = 121 x [0.65 + 0.01 x 24]

FP = 107.69

# Estimación con método de Cocomo:

Ocupamos calcular el esfuerzo, pero primero tenemos que hallar la variable KDLC, donde los PF son 107.69 y las líneas por cada PF

equivalen a 47 por ser en JavaScript.

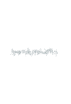
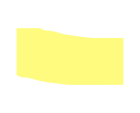
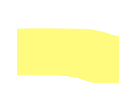
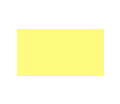
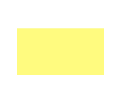
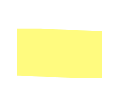
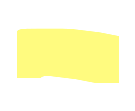
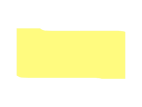
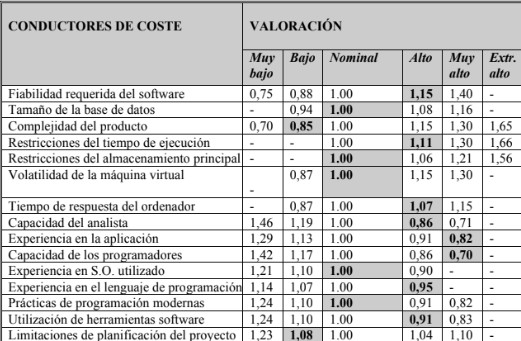
KDLC = (PF \* Líneas de código por cada PF)/1000 = (107.69\*47)/1000= 5.06143 KDLC

En este caso nuestro proyecto por tener menos de 50 KDLC es un proyecto **Orgánico** por tener 5.06143 KDLC líneas de código.

Y al ser orgánico nuestro proyecto tenemos que tomar en cuenta las siguientes variables:



Por otro lado, también tenemos que hallar la variable FAE que se obtiene mediante la multiplicación de los siguientes conductores:



FAE = 1.058778401

# Cálculo del esfuerzo del desarrollo:

E = a KLDC^e \* FAE = 3.2 (5.06143) ^1.05 \* 1.058778401= 18.596 personas

/mes

# Calculo de tiempo de desarrollo:

T = c Esfuerzo^d = 2.5 (18.596)^0.38 = 7.59 meses

# Productividad:

PR = LDC/Esfuerzo = 5061/18.596 = 272.155 LDC personas/mes

# Personal promedio:

P = E/T = 18.596/7.59 = 2.45 personas

Lo que ocupamos de personas para acabar en mes y medio son:

18.596/1.5= 12.397 personas

## Fiabilidad requerida del software

El sistema no es crítico. Si falla, se puede volver a intentar sin

consecuencias graves. Por eso no se necesita una fiabilidad tan alta.

## Tamaño de base de datos

No se guardan millones de datos. Solo se manejan archivos temporales o datos simples como productos, cantidades y PDFs, así que el tamaño es pequeño.

## Complejidad del producto

El sistema hace tareas claras: leer un archivo, extraer información y

generar otro archivo. No tiene funciones avanzadas ni muchos módulos diferentes.

**Restricciones del tiempo de ejecución**

**No es necesario que funcione en tiempo real ni al instante. Con que tarde unos segundos está bien, no hay urgencia extrema.**

**Restricciones del almacenamiento principal**

Corre en computadoras normales. No necesita mucha memoria RAM ni espacio de disco para trabajar bien.

## Volatilidad de la máquina virtual

El sistema corre en una computadora o en el NAS, pero no cambia mucho el entorno donde se usa. Por eso se considera estándar.

## Tiempo de respuesta del ordenador

No necesitamos que el equipo sea rápido como una supercomputadora. Una laptop común puede correrlo sin problema.

## Capacidad del analista

Somos estudiantes y aunque no somos expertos, entendemos lo básico para analizar y diseñar lo que necesitamos.

## Experiencia en la aplicación

Es un sistema nuevo para nosotros y estamos aprendiendo mientras lo hacemos. No tenemos mucha experiencia con este tipo de proyecto.

## Capacidad de los programadores

Es un sistema nuevo para nosotros y estamos aprendiendo mientras lo hacemos. No tenemos mucha experiencia con este tipo de proyecto.

## Experiencia en el S.O utilizado

Sabemos usar Windows (u otro S.O. que estemos usando), pero no somos expertos. Lo usamos en la vida diaria y eso es suficiente.

## Experiencia en el lenguaje de programación

Conocemos el lenguaje (como Python o JavaScript) a nivel básico o medio. Podemos programar lo que necesitamos, aunque a veces investigamos.

## Prácticas de programación modernas

Aún no aplicamos muchas prácticas profesionales porque estamos aprendiendo. No usamos pruebas automáticas ni control de versiones todo el tiempo.

## Utilización de herramientas de software

Usamos algunas herramientas básicas como editores de código y navegadores, pero no herramientas avanzadas como IDEs profesionales o automatización.

## Limitaciones de planificación del proyecto

Como somos estudiantes, el tiempo es limitado. El proyecto avanza conforme podemos entre clases y otras actividades, así que hay pocas horas por semana.