

UNIDAD 3

Alberto Alejandro López Leos

Planificación Del Proyecto Del Software

Actividad:

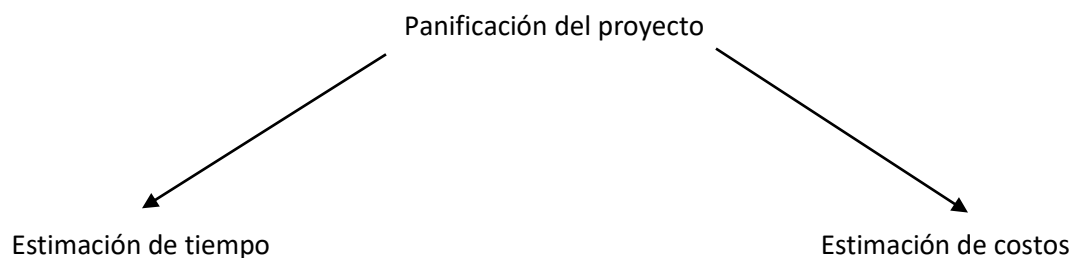
- Manifiesto ágil ----- sierra
- Open Up ----- Alejandro Chávez
- Scrum ----- Leslie
- Extreme programming-----Ricardo
- **Kamban----- Alejandro leos**
- Scrumban----- Edith
- Leam----- Víctor
- Feature-Driven Development----- Carlos
- Test-Driven Development----- Kevin

(Factibilidad técnica, operacional, económica y legal)

Objetivo General: Desarrollar un sistema web con el cual los clientes personalizan sus muebles

Objetivos Específicos: Montar servidor web, desarrollar una base de datos, realiza lista de requerimientos del sistema, desarrollar el sistema en PHP cumpliendo los requerimientos, validar el sistema.

HOSTING	DOMINIO	MATERIALES
\$2290 c/a	\$340 c/a	PC Trabajo



Se han propuesto numerosos métodos de estimación. Entre ellos se pueden contar:

- *Juicio experto
- *Analogía
- *Parkinson
- *Modelos Algoritmos

COCOMO

El modelo constructivo de costes (constructive Cost Model) fue desarrollado por B. W Boehm a finales de los 70 y comienzos de los 80 exponiéndolo en su libro “SOFTWARE ENGINEERING”

Tipos de proyectos que define COCOMO:

- Orgánico
- Semi-acoplado
- Empotrado

Modo orgánico:

En esta clasificación se encuentran proyectos desarrollados en un ambiente familiar y estable. Requiere pocas innovaciones tecnológicas en la que refiere algoritmos y estructuras de software existente.

En los casos de productos más extensos, han sido construidos frecuentemente a partir de software existente.

Ejemplos de software que se encuentran bajo esta clasificación son:

- Modelos de negocios
- Modelos científicos

Los productos tienen un tamaño que llega a 300 KSLOC.

Ejemplos de software que clasifican:

- Sistemas de control de producción
- Sistemas de procesamiento de transacciones
- Administradores de base de datos

Modo Empotrado:

En esta clasificación están incluidos proyectos de gran envergadura que operan en un ambiente complejo con altas restricciones de hardware, software y procedimientos operacionales, tales como los sistemas de tráfico aéreo.

Se espera que el software no solo conforme las especificaciones sino también que sea estable frente a cambios y dificultades producidas en el ambiente.

Modelos que define COCOMO:

- Modelo básico
- Modelo Intermedio

E=Esfuerzo (Persona por mes)

T=Tiempo de duración del proyecto

P=Personas

$$E = a(KDLC)^b$$

$$T = c \cdot (E)^d$$

$$P = \frac{E}{T}$$

Proyecto Software	A	B	C	D
Orgánico	3.2	1.05	2.5	0.38
Semi-Acoplado	3.0	1.12	2.5	0.35

$$KLOC = (\text{Líneas de código} \cdot PF) / 1000$$

Ensamblador 320

C 150

Ctt 64

SQL 12

PRACTICA:

Estimar el costo de programación por los métodos de analogía, juicio experto, y cocomo.

Juicio experto: **\$11,500**

Software con un control de usuarios y roles de los mismos

Analógico:

RIESGOS DE PROYECTO DE SOFTWARE

Implica dos características:

- PERDIDA
- INCERTIDUMBRE



Categorización de riesgos:

- Riesgos Conocidos
- Riesgos Predecibles
- Riesgos Impredecibles

RIESGO	PRIORIDAD	PROBABILIDAD	IMPACTO	CAUSA
No tener tiempo	ALTA	Media	No entregar el software	Otros pendientes
Impacto en el negocio	MEDIA	Media	Error en el software	Otro pendientes
Falta de conocimiento	ALTA	Alta	Nivel bajo de conocimiento	Otro pendientes
Falta de información	MEDIA	Media	Estándar bajo en el proyecto	Otro pendientes
Falta de recursos	MEDIA	Alta	No tener material para realizar el software	Otro pendientes
Definición del proceso	ALTA	Alta	Qué No tenga seguimiento del desarrollo	Otros pendientes
Tecnología a construir	MEDIA	Alta	Complejidad del sistema a construir	Otros pendientes
Tamaño y experiencia de la plantilla	MEDIA	Media	Experiencia técnica	Otros pendientes
Características del cliente	MEDIA	Alta	Satisfacción del cliente y habilidad para usarlo	Otros pendientes

- Xampp
- Codeigniter
- Bootstrap
- JQuery
- Editor de texto
- MySQL Workbench



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	INICIACION	DURACION (DIAS)													
2	ACTIVIDADES	1	2	3	4	5	6	7	8	9	10	11	12	13	14
3	Definir alcance del proyecto														
4	Estimacion de costes y programacion														
5	Definir riesgos														
6	Determinar viabilidad del proyecto														
7	Preparar entorno del proyecto														
8	ELABORACION														
9	ACTIVIDADES	1	2	3	4	5	6	7	8	9	10	11	12	13	14
10	Identificar arquitectura														
11	Validar arquitectura														
12	Desarrollar entorno del proyecto														
13	Determinar equipo														