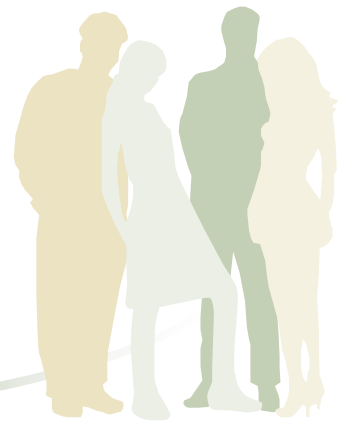


# Tema 06 – Gestión de proyectos

## Ingeniería del Software

Rubén Fuentes Fernández  
Dep. Ingeniería del Software e Inteligencia Artificial  
Facultad de Informática  
Universidad Complutense Madrid

Trabajando con Antonio Navarro, Juan Pavón y Pablo Gervás



## Contenidos

- Introducción
  - Problemática
- Aspectos de la solución
  - Planificación temporal
    - Ámbito del software
    - Descomposición del problema
    - Descomposición del proceso
    - Proyecto
  - Plan de proyecto



## En los clásicos

“... si las estimaciones efectuadas antes de las hostilidades indican victoria, es porque los cálculos muestran que la fortaleza propia es superior a la del enemigo; si indican derrota, es porque los cálculos muestran que es inferior. Con muchos cálculos se puede ganar; con pocos, no. ¡Cuántas menos posibilidades de victoria tiene quien no hace ninguno!”, *El arte de la guerra* (Suntzu, siglo IV a.c)



## Introducción

- Un proyecto se ejecuta con éxito cuando:
  - Se entrega un producto satisfactorio según las necesidades del cliente.
  - Se hace cumpliendo con las restricciones establecidas.
    - Ej. coste, plazos, recursos y fallos.
- La planificación temporal y el seguimiento del proyecto tienen como objetivo primordial evitar los retrasos en las entregas del software.
  - Siempre sin penalizar el resto de los requisitos y restricciones del proyecto.





# Causas de los problemas con el tiempo

- Fechas límite de entrega poco realistas
- Cambios de los requisitos que no se reflejan en la planificación temporal
- Subestimación honesta del esfuerzo y/o recursos
- Riesgos predecibles e impredecibles no considerados
- Falta de comunicación entre la plantilla
- Falta de reconocimiento del retraso en un proyecto
- Falta de medidas para corregir el problema



# Abordar los problemas de fechas

- Realizar estimaciones detalladas
  - Recursos necesarios para producir el sistema
  - Recursos disponibles y productividad en el equipo de desarrollo
- Utilizar modelos incrementales
  - Implementar la funcionalidad crítica mínima anticipadamente
  - Extender el sistema
- Establecer plazos realistas para ejecutar las tareas previstas
  - Considerando los recursos a comprometer y su productividad
  - Precizando qué se necesita, qué se generará y cómo se evaluará
  - Dejando margen para los problemas que seguro surgirán
- Reunirse con los clientes para analizar y explicar el devenir del proyecto





# PLANIFICACIÓN TEMPORAL



## Producto

- Dilema del gestor al inicio del proyecto:
  - Necesita planificar proyecto.
  - No dispone de información sólida.
- La especificación de requisitos puede ser la solución.
  - Frecuentemente dicha especificación no está disponible aún.
  - El plan del proyecto se necesita ya.
- Solución: determinar el *ámbito del software*





# Ámbito del software

- El ámbito del software incluye:
  - Contexto
    - Forma de integrar el sistema en el sistema, producto o contexto de negocios mayor
    - Limitaciones resultantes del contexto
  - Objetivos de información
    - Datos visibles al cliente
    - Datos de entrada requeridos
  - Funciones y rendimiento
    - Funciones realizadas por el software para transformar la información de entrada en salida
    - Características especiales de rendimiento
- El ámbito debe ser:
  - Unívoco
  - Entendible a nivel técnico y de gestión



# Descomposición basada en el producto

- Se lleva a cabo la *partición horizontal*.
  - Se descompone el sistema en *módulos*.
  - Un *módulo* es una agrupación de funciones que llevan a cabo tareas de naturaleza similar.
- Después se refina con una *partición vertical*.
  - Se trata de una descomposición hasta el nivel de *funciones*.
  - Una *función* representa un procesamiento directamente invocable por el usuario, y que transforma información de entrada en información de salida.
  - En el proceso de descomposición pueden definirse nuevos submódulos.



# Ejemplo: descomposición basada en el producto

- En un sistema CAD:
  - Partición horizontal
    - Módulo de dibujo
    - Módulo de transformaciones
    - Módulo de archivo
    - Módulo de impresión
  - Partición vertical
    - Módulo de dibujo
      - Dibujo 2D
        - » Líneas
          - ◆ Por centro y radio
          - ◆ Por tres puntos
        - » Cuadrados
        - » Circunferencias
        - » *Splines*
        - » .....
      - Dibujo 3D



## Proceso

- Las actividades comunes al proceso de IS se encuentran presentes en todos los modelos de proceso.
- El problema es aplicar el modelo de proceso más adecuado para el proyecto.
- El gestor elige dicho modelo en base a:
  - El cliente
  - Las características del producto
  - El equipo de desarrollo





# Actividades

- Las Actividades Estructurales (AE) del proceso deben descomponerse en acciones.
  - Dicha descomposición depende, básicamente del tipo de proyecto.
  - Ej. para un proyecto de desarrollo de un producto nuevo, la AE *diseño* puede descomponerse en acciones *análisis*, *diseño* y *evaluación*.
  - Ej. para un proyecto de reingeniería, la AE *diseño* puede descomponerse en las acciones *reingeniería*, *análisis* y *diseño*.
- A su vez, estas acciones se descomponen en conjuntos de tareas distintos para cada proyecto.
  - Ej. la acción *análisis* puede descomponerse en las tareas *realizar diagramas casos de uso*, *realizar diagramas actividades*, *realizar diagramas clases análisis*...

Ver Tema 02 - Proceso de Desarrollo de Software



## Definición de un conjunto de tareas para el proyecto

- Los modelos de proceso no se aplican sin cambios.
- Hay que adaptar el modelo de proceso al proyecto.
  - Variando las acciones y el conjunto de tareas en las que se descomponen las actividades.
- Por supuesto, las Actividades de Protección (AP) están empotradas en las AE.



# Estructura de descomposición del trabajo

- La Estructura de Descomposición del Trabajo (EDT) determina
  - las tareas de trabajo concretas
    - Entradas + Proceso + Salidas + Criterios de evaluación
  - que se asignan a individuos concretos
  - con plazos concretos
  - con el fin de desarrollar un proyecto de software
- Dicha asignación de individuos y plazos es la *planificación temporal*.
- La EDT admite varias representaciones visuales.
  - Ej. árboles, tablas y gráficos de Gantt



## EDT: representación con tabla

AE	AE1				AE2				.....			AEn			
A	a <sub>1.1</sub>	a <sub>1.2</sub>	....	a <sub>1,m1</sub>	a <sub>2.1</sub>	a <sub>2.2</sub>	....	a <sub>2,m2</sub>	....	....	....	a <sub>n.1</sub>	a <sub>n.2</sub>	....	a <sub>n,m1</sub>
M1															
f1.1															
f1.2															
....															
f1.i <sub>1</sub>															
M2															
f2.1															
f2.2															
....															
f2.i <sub>2</sub>															
....										tarea <sub>x</sub> - recursos - fecha inicio - fecha fin - entrega					
Mk															
fk.1															
fk.2															
....															
fk.i <sub>k</sub>															





# Ejemplo: EDT con tabla

- Tabla EDT con información de planificación temporal

Acción de Ingeniería del software

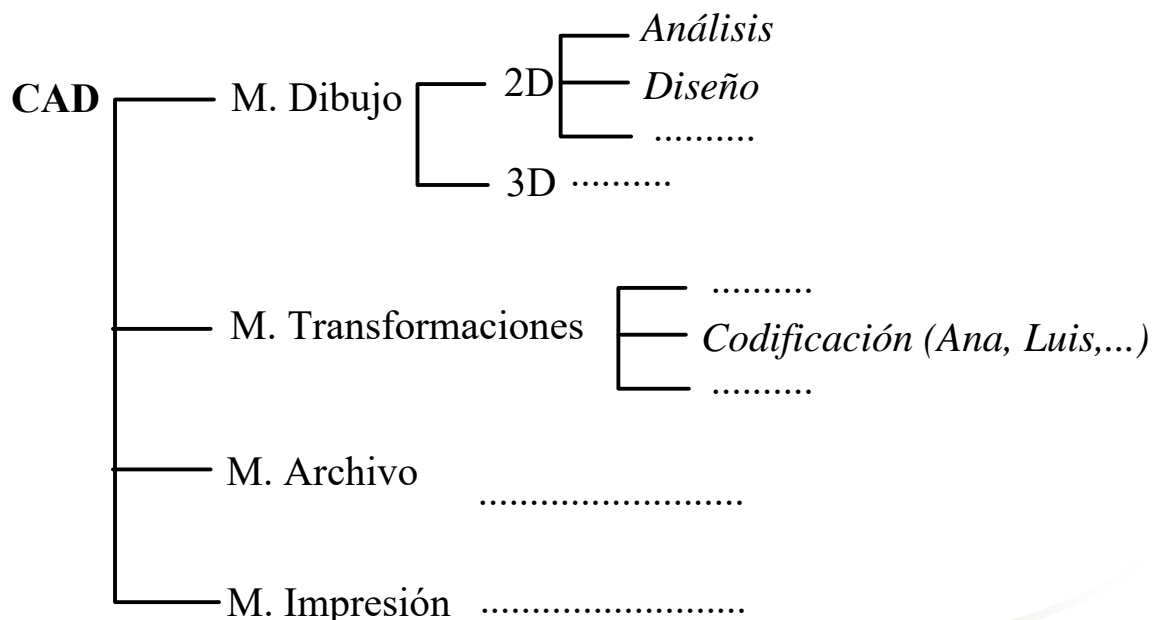
AE	Comunicación Cliente		Planificación		Análisis de riesgos		Ingeniería		Construcción y adaptación			Evaluación Cliente	
Acción	TUE	SRS	Estim.	Planif.	Valor	RSGR	Anal.	Diseño	Codif.	Prueba	Ensam.	Instal.	Eval.
Módulo dibujo													
Módulo transformaciones									Ana, Luis, Paco 13.11.06 17.11.06 Código m. transf.				
Módulo archivo													
Módulo impresión													

Tarea de trabajo



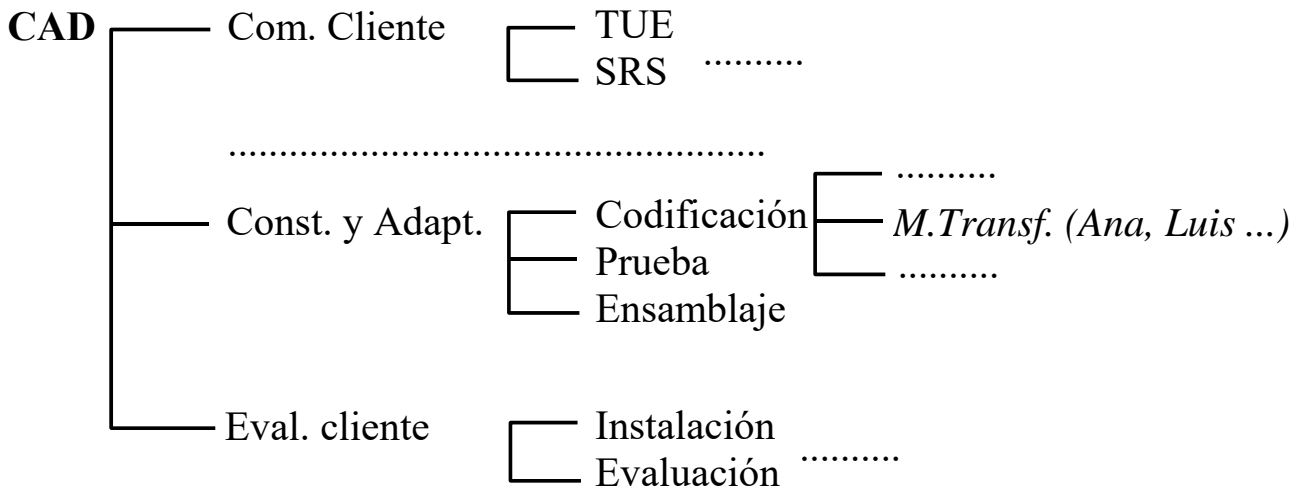
# Ejemplo: EDT con árbol - módulos

- Representación arbórea dirigida por el producto de la anterior tabla EDT y la información planificación temporal.

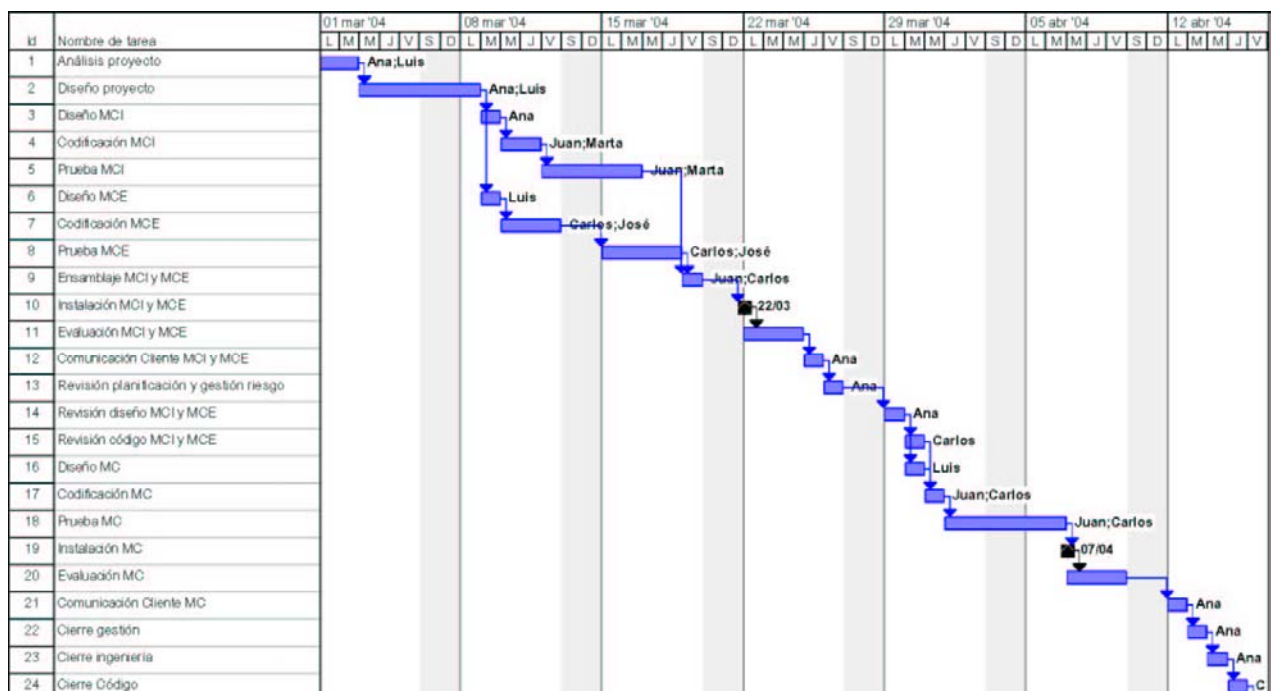


# Ejemplo: EDT con árbol - tareas

- Representación arbórea dirigida por el proceso de la anterior tabla EDT y la información planificación temporal.



# Ejemplo: EDT con gráfico de Gantt





# Principios básicos de la planificación temporal

- **Compartimentalización**
  - Número de tareas manejables
  - Obtenidas mediante la EDT
- **Interdependencia**
  - Dependencias entre las tareas identificadas
- **Asignación de esfuerzo**
  - Cada tarea debe tener asignadas un cierto número de unidades de trabajo, así como fecha de inicio y de fin
- **Validación de esfuerzo**
  - No se debe sobreasignar recursos
- **Responsabilidades definidas**
  - Cada tarea debe tener asignados miembros específicos
- **Resultados definidos**
  - Cada tarea debe tener un resultado definido.
- **Hitos definidos**
  - Las tareas deberían asociarse a hitos del proyecto
  - Se consigue un hito cuando se aceptan uno o más productos tras revisar su calidad



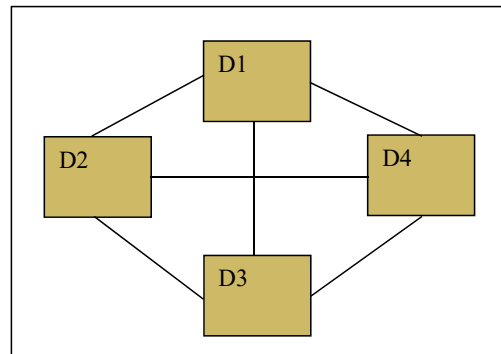
# Relación entre personas

- Desde el punto de vista del esfuerzo da igual tener tres personas trabajando dos meses que dos personas trabajando tres meses.
- Lo que no da igual es añadir personas de manera incontrolada.
- Personas → canales de comunicación → descenso en la productividad
  - Cuanta más gente haya en un equipo, menor será su productividad.
  - Incorporar gente nueva implica comunicaciones más costosas.
- Esto puede tener consecuencias nefastas si se añade gente al final de un proyecto.



## Relación entre personas: comunicación (1/2)

- Supongamos 4 desarrolladores con una productividad individual de 600 LDC/PM.
- Cuando trabajan juntos, se abren seis posibles vías de comunicación.



## Relación entre personas: comunicación (2/2)

- Cada vía de comunicación requiere un tiempo que podría emplearse en desarrollo.
- Si suponemos una reducción en 20 LDC/vía-mes debido al gasto en comunicación, la productividad real es:
  - $600 \text{ LDC/PM} * 4 \text{ personas} - 20 \text{ LDC/vía-mes} * 6 \text{ vías} = 2280 \text{ LDC/mes}$
  - $2280 \text{ LDC/mes} / 4 \text{ personas} = 570 \text{ LDC/PM}$
- Es decir, una merma del 5% en la productividad.





# Ecuación del software

- $E = B * (LDC / P)^3 * (1/t^4)$

- $E = (B * (LDC / P)^3) / t^4$

- donde:

- E = Esfuerzo (medido normalmente en PM)

- B = Factor especial de destrezas

- LDC = Líneas de código (tamaño del proyecto)

- P = Parámetro de productividad

- t = duración del proyecto (medido normalmente en meses)

Ver Tema 07 – Estimación de proyectos



## Relación entre personas y ecuación del software

- Supongamos un proyecto con 50000 LDC, B=0,28 y P=10000

- $E = 35/t^4$

- Si t = 12 meses = 1 año

- E = 35 personas-año

- Para acabar en 1 año → equipo ≈ 3 personas

- Si t = 13 meses = 1,083 años

- E = 25,4 personas-año

- Para acabar en 1 año → equipo ≈ 2 personas

- Es decir, es más rentable menos personas más tiempo...

- O eso, o estamos forzando la ecuación del software.





# Grado de rigor del EDT

- Los conjuntos de tareas se eligen en función del grado de rigor que deseemos aplicar en un proyecto.
- A su vez el grado de rigor depende fundamentalmente del tipo de proyecto:
  - Proyectos de desarrollo del concepto
    - Exploración de nuevos conceptos de negocio o aplicación de nuevas tecnologías
  - Proyectos de desarrollo de una nueva aplicación
    - Encargo de un cliente específico
  - Proyectos de mejora de aplicaciones
    - Modificaciones relevantes de un software existente que afectan a su funcionamiento, rendimiento o interfaces, y que son observables externamente
  - Proyectos de mantenimiento de aplicaciones
    - Correcciones, adaptaciones o ampliaciones de un software existente de maneras que pueden no ser evidentes para el usuario final
  - Proyectos de reingeniería
    - Reconstrucción de un sistema existente en su totalidad o parte



# Criterios de adaptación

- Los criterios de adaptación se emplean para determinar el grado de rigor.
- Matizan la importancia del tipo de proyecto.
- Hay doce criterios de adaptación.
- Se puntúan de 1 (poca rigidez) a 5 (muchísima rigidez).



# Selector del conjunto de tareas

Criterios de Adaptación	Grado	Peso	Multiplicador de punto de entrada					Producto
			Concur.	Ndev.	Mejora	Mant.	Reing.	
Tamaño del proyecto	2	1,2	—	1	—	—	—	2,4
Número de usuarios	3	1,1	—	1	—	—	—	3,3
Importancia para el negocio	4	1,1	—	1	—	—	—	4,4
Antigüedad	3	0,9	—	1	—	—	—	2,7
Estabilidad de los requisitos	2	1,2	—	1	—	—	—	2,4
Facilidad de Comunicación	2	0,9	—	1	—	—	—	1,8
Madurez de la tecnología	2	0,9	—	1	—	—	—	1,8
Limitaciones de rendimiento	3	0,8	—	1	—	—	—	2,4
Empotrado/no empotrado	3	1,2	—	1	—	—	—	3,6
Personal del proyecto	2	1,0	—	1	—	—	—	2,0
Interoperabilidad	4	1,1	—	1	—	—	—	4,4
Factores de reingeniería	0	1,2	—	0	—	—	—	0,0
<b>Selector de conjunto de tareas (SCT)</b>								
TABLA 7.2 CÁLCULO DEL SCT								2,8



## Interpretación del SCT

Valor del SCT	Grado de rigor
$SCT < 1,2$	Casual
$1,0 < SCT < 3,0$	Estructurado
$SCT > 2,4$	Estricto
Solapamiento	Libertad entre opciones solapadas

casual < estructurado < estricto

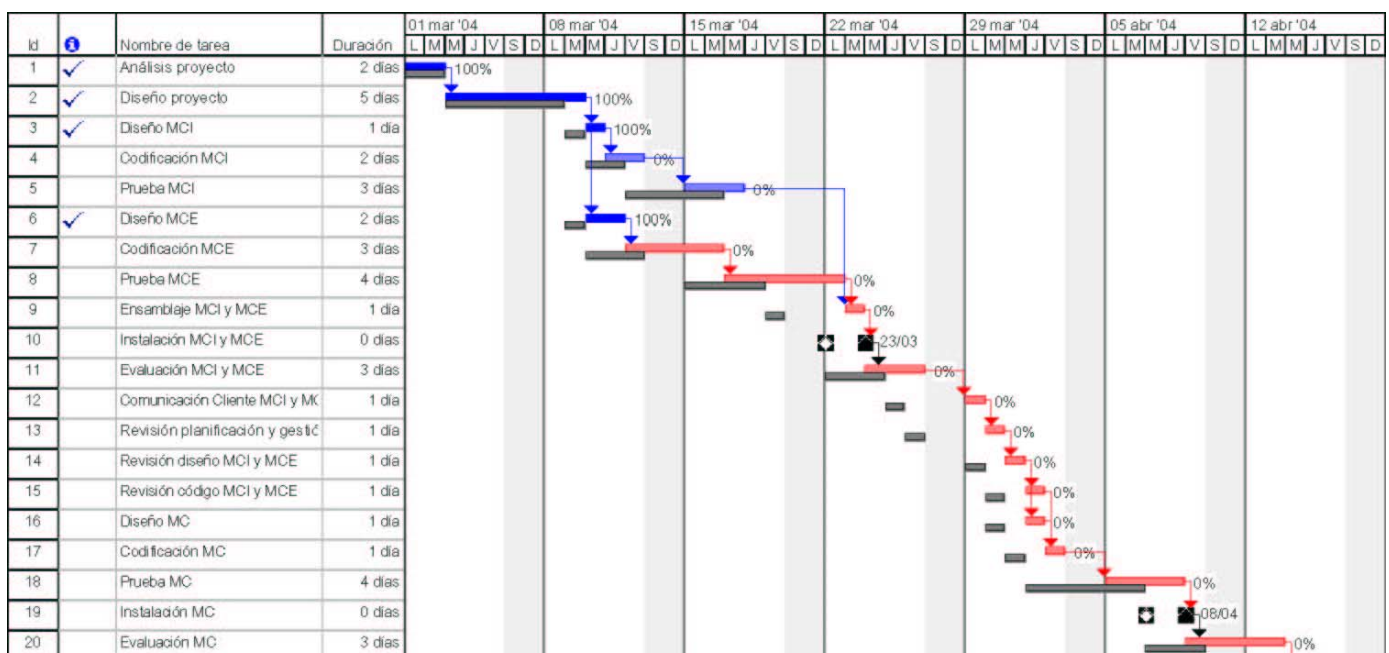


# Planificación temporal: seguimiento de la planificación

- El seguimiento de la planificación temporal es un aspecto clave de la ejecución de un proyecto.
- Hay distintas formas de implementarlo:
  - Realizando reuniones periódicas
  - Evaluando los resultados de las revisiones
  - Determinando si se han conseguido los hitos
  - Recabando las opiniones subjetivas del equipo
  - Comparando la fecha real de inicio con las previstas
  - Utilizando el análisis del valor ganado
- En el seguimiento merece especial atención el *camino crítico*.
  - Formado por la cadena de tareas de mayor duración del proyecto.
  - Un retraso en una tarea crítica implica automáticamente un retraso en el proyecto.



## Seguimiento de la planificación con diagrama de Gantt







# Orientaciones de distribución del esfuerzo

- Regla del 40-20-40
  - 40% esfuerzo en análisis y diseño
  - 20% esfuerzo en codificación
  - 40% esfuerzo en pruebas
- Otra posibilidad:
  - 2-3% en planificación
  - 10-25% en análisis de requisitos
  - 20-25% diseño
  - 15-20% en codificación
  - 30-40% en pruebas
- Nunca aplicar sin reflexionar.
  - Cada proyecto es un mundo.



# Señales de peligro

- Para gestionar un proyecto se debe comprender qué puede ir mal.
- Reel identifica diez señales que indican que un proyecto está en peligro:
  - Los desarrolladores no comprenden las necesidades del cliente.
  - El ámbito del producto está definido pobremente.
  - Los cambios están mal realizados.
  - La tecnología elegida cambia.
  - Las necesidades del negocio cambian o están mal definidas.
  - Las fechas de entrega no son realistas.
  - Los usuarios se quejan.
  - Se pierden recursos económicos.
  - El equipo carece de personal adecuado.
  - Gestor y equipo no aplican prácticas de IS.





# PLAN DE PROYECTO



## Introducción

- El plan del proyecto del software contiene toda la información relativa a la gestión del proyecto.
  - Se produce cuando se terminan las tareas de planificación.
- Objetivos:
  - Comunicar el ámbito y recursos al personal de desarrollo y al cliente.
  - Definir los riesgos y sugerir técnicas de control del riesgo.
  - Definir los costes y planificación temporal para la revisión de la gestión.
  - Proporcionar un enfoque general del desarrollo del software para todo el personal relacionado con el proyecto.
  - Describir cómo se garantizará la calidad y se gestionarán los cambios.
- El plan del proyecto no debe ser un documento largo ni complejo.
  - Guía el desarrollo del software





## IEEE Std. 1058-1998: tabla de contenidos (1/5)

- Página de título
- Página de firma
- Historial de cambios
- Prefacio
- Tabla de contenidos
- Lista de figuras
- Lista de tablas



## IEEE Std. 1058-1998: tabla de contenidos (2/5)

1. Descripción
  - 1.1. Resumen del proyecto
    - 1.1.1. Propósito, alcance y objetivos
    - 1.1.2. Supuestos y restricciones
    - 1.1.3. Entregas del proyecto
    - 1.1.4. Resumen de planificación y presupuesto
  - 1.2. Evolución del plan
2. Referencias
3. Definiciones
4. Organización del proyecto
  - 4.1. Interfaces externas
  - 4.2. Estructura interna
  - 4.3. Papeles y responsabilidades





## IEEE Std. 1058-1998: tabla de contenidos (3/5)

### 5. Planes de proceso de gestión

#### 5.1. Plan de comienzo

5.1.1. Plan de estimación

5.1.2. Plan de plantilla

5.1.3. Plan de adquisición de recursos

5.1.4. Plan de formación de la plantilla del proyecto

#### 5.2. Plan de trabajo

5.2.1. Actividades de trabajo

5.2.2. Asignación de planificación

5.2.3. Asignación de recursos

5.2.4. Asignación de presupuesto



## IEEE Std. 1058-1998: tabla de contenidos (4/5)

#### 5.3. Plan de control

5.3.1. Plan de control de requisitos

5.3.2. Plan de control de planificación

5.3.3. Plan de control de presupuesto

5.3.4. Plan de control de calidad

5.3.5. Plan de informes

5.3.6. Plan de recolección de métricas

#### 5.4. Plan de gestión del riesgo

#### 5.5. Plan de cierre

### 6. Planes de proceso técnico

6.1. Modelo de proceso

6.2. Métodos, herramientas y técnicas

6.3. Plan de infraestructura

6.4. Plan de aceptación del producto





# IEEE Std. 1058-1998: tabla de contenidos (5/5)

## 7. Planes de soporte del proceso

- 7.1. Plan de gestión de la configuración
- 7.2. Plan de verificación y validación
- 7.3. Plan de documentación
- 7.4. Plan de garantía de calidad
- 7.5. Revisiones y auditorías
- 7.6. Plan de resolución de problemas
- 7.7. Plan de gestión de la subcontratación
- 7.8. Plan de mejora del proceso

## 8. Planes adicionales

Apéndices

Índice



# CONCLUSIONES





# Conclusiones

- La planificación del proyecto tiene como objetivo evitar retrasos.
- La planificación consiste en establecer tareas para llevar a cabo el proyecto y relacionarlas.
  - Asignarles fechas de inicio y fin, recursos y esfuerzo.
  - Adaptadas al grado de rigor requerido por el proyecto.
- Existen algunas pautas generales para la planificación.
  - Estimación por descomposición del producto y el proceso
  - Regla del 40-20-40
  - Mejor equipos pequeños que grandes
- La planificación es inútil sin un seguimiento y ajuste de la misma a lo largo del proyecto.



# Glosario

- 2D = 2 Dimensiones
- 3D = 3 Dimensiones
- AE = Actividad Estructural
- AP = Actividad de Protección
- CAD = *Computer-Aided Design*
- EDT = Estructura de Descomposición del Trabajo = WBS
- IS = Ingeniería del Software
- SCT = Selector del Conjunto de Tareas
- WBS = *Work Breakdown Structure* = EDT





# Referencias

- R. Pressman: Ingeniería del Software. Un enfoque práctico, 7ª edición. McGraw-Hill, 2010.
  - Capítulo 3
- I. Sommerville: Ingeniería del Software, 7ª edición. Addison Wesley, 2007.
  - Capítulo 22
- IEEE: IEEE Std. 1058-1998, IEEE standard for software project management plans. IEEE, 1998.

