

# GPSW-Apuntes-Segundo-Parcial.pdf



**CreatorBeastGD**



**Gestión de Proyectos Software**



**4º Grado en Ingeniería del Software**



**Escuela Técnica Superior de Ingeniería Informática  
Universidad de Málaga**

# **Gestión de Proyectos**

## **Software**

**Apuntes del Segundo Parcial**

**Javier Molina Colmenero**

**WUOLAH**

# Tema 7. Cómo iniciar un proyecto

## Consideraciones estratégicas.

Para poder iniciar un proyecto, es recomendable tener en cuenta algunas consideraciones estratégicas, también llamadas motivos para un proyecto, que conllevan a poner en marcha proyectos:

- **Demanda del Mercado** (Una compañía de coches autoriza un proyecto que construirá coches más eficientes).
- **Necesidad u Oportunidad de negocio** (Una compañía de formación crea un curso para incrementar sus ingresos)
- **Necesidades Organizacionales** (Por los altos costes, una empresa puede combinar sus funciones de personal administrativo con los procesos de recepción de pedidos para reducir costes)
- **Peticiones del Cliente** (Una compañía eléctrica autoriza un proyecto para servir un nuevo parque industrial)
- **Avances Tecnológicos** (Autorización de una empresa de un proyecto para desarrollar un portátil más rápido y barato)
- **Requisitos Legales** (Una empresa química autoriza un proyecto para establecer guías para el manejo de un nuevo material tóxico)
- **Impacto Ecológico** (Una empresa autoriza un proyecto para reducir su impacto ambiental)
- **Necesidades Sociales** (Una ONG autoriza un proyecto en un país en desarrollo para proporcionar sistemas de agua potable y educación sanitaria para prevenir el cólera)

Los proyectos tienen que surgir como respuesta a necesidades del "Negocio". Desde el punto de vista organizacional, los proyectos se inician por una "entidad externa" al propio proyecto:

- Un **iniciador** o un **espónsor**.
- Una **Oficina de Gestión de Proyectos o Programas (OGP)**.
- O bien por una **persona autorizada** que gobierna un portfolio o programa de la organización.

El iniciador o espónsor del proyecto debería tener suficiente nivel de autoridad para asegurar financiación y recursos para el proyecto.

## Secuencia natural de los documentos en el arranque del Proyecto

Se requieren hacer 3 documentos, en el inicio, antes y durante el inicio:

- **Statement of Work.** Es el enunciado del trabajo y es el que describe lo que hay que hacer. Debe tener un nivel de detalle suficiente para que todas las partes entiendan lo que se quiere hacer. A partir de este documento se puede hacer el **Request for Proposal (RFP)**, que servirá para explicar a los posibles proveedores qué es lo que se necesita.
- **Business Case.** Es el análisis de coste y beneficio, y describe cuál es el problema de Negocio, y qué se quiere hacer para darle respuesta, además de explicar cómo y cuánto beneficiará la solución a resolver el problema. Adicionalmente, ayuda a tomar la decisión de ejecutar el proyecto.
- **Project Charter.** Es la descripción del proyecto y la justificación de su ejecución. Describe con suficiente nivel de detalle los aspectos importantes del proyecto. Contiene el presupuesto con el que cuenta el proyecto, e informa quién será la persona que asumirá el role de JP. Autoriza formalmente que el proyecto se haga.

Este último documento se realiza al inicio del proyecto y es sumamente importante. También se puede conocer como **Acta de Constitución del Proyecto, Definición del Proyecto o Definición y Encuadre del Proyecto.**

	Entradas	Procesos Iniciación	Salidas	Técnicas
<b>4</b> Gestión de la Integración del Proyecto	.1 Project statement of work (SoW - <b>Enunciado del Trabajo del Proyecto</b> ) .2 <b>Business case</b> (Caso de Negocio) .3 Agreements ( <b>Contratos</b> ) .4 Enterprise environmental factors (Factores ambientales de la Organización) .5 Organizational process <b>assets</b>	4.1 Desarrollar el Acta de Constitución del Proyecto ( <b>Project charter</b> )	.1 <b>Project charter</b> (Acta de Constitución del Proyecto – Definición y Encuadre del Proyecto)	.1 Expert judgment (Opinión de expertos) .2 Técnicas de Facilitación
<b>13</b> Gestión de los Stakeholders del Proyecto	.1 <b>Project charter</b> .2 Procurement documents .3 Enterprise environmental factors .4 Organizational process assets	13.1 Identificar los Stakeholders (Grupos de Interés)	.1 <b>Stakeholder register</b> (registros de los Grupos de Interés)	.1 Stakeholder analysis .2 Expert judgment .3 Meetings

## Posible estructura del Project Charter

El Project Charter puede tener como estructura la siguiente:

- **Encuadre.** Código del programa, nombre del programa y plan.
- **Identificación.** Código del proyecto, nombre, departamento responsable, responsable del proyecto, JP, etc.
- **Objetivos del Proyecto.** Código del objetivo, descripción e importancia, y alineación de cada objetivo con el programa o plan.
- **Necesidad.** Describir la necesidad, problema u oportunidad que cubre el proyecto.
- **Descripción del Proyecto.** En qué consiste en proyecto, cuál es la solución que se va a poner en marcha con este proyecto.
- **Justificación.** Si existían otras alternativas, porqué se llega a ésta. Qué impacto podría tener no realizar el proyecto.
- **Grupos de Interés.** Listarlos y describir los criterios de éxito.
- **Entregables Finales.** Listarlos y describirlos.
- **Alcance de los entregables.** Descripción de los requisitos de los entregables del proyecto.
- **Criterios de Aceptación.** Quién dará el VB a cada entregable y con qué criterios.
- **Suposiciones y Restricciones.** Cualquier suposición o restricción que se considere importante mencionar y bajo la cual el proyecto se va a desarrollar.
- **Límites del Proyecto.** Explícitamente qué está fuera del proyecto.
- **Estándares aplicables.** Cualquier regulación, metodología o política que se deba aplicar en el proyecto, entre otras cosas.
- **Presupuesto.** Presupuesto del proyecto y fuentes de financiación por años.
- **Coste/Beneficio.** Análisis del coste o beneficio si tiene sentido.
- **Metas e indicadores.** Metas e indicadores asociados.
- **Aprobación del Proyecto.** De qué manera y por quién ha sido aprobado este proyecto.

# Tema 8. Construir un EDT

## Entregables en un proyecto (Tema 2)

Un entregable es la salida de un proyecto, trabajo o tarea.

Un proyecto puede tener varios entregables:

- **Producto producido:** Es un elemento producido sea o no físico (este último, como el software)
- **Servicio:** Los proyectos tienen capacidad de prestar un servicio, por ejemplo, para la impartición de un curso de formación.
- **Un resultado:** Por ejemplo, en un proyecto de investigación, el resultado es la generación de conocimientos que aportan información sobre si un determinado fármaco puede ser o no beneficioso para la salud.

## Esquema de desglose de trabajo (EDT)

Un **EDT** es una estructura que divide los entregables del proyecto y el trabajo a realizar en el proyecto en componentes más pequeños y manejables. De esta manera se tiene una visión estructurada de lo que se debe entregar. No se puede plantear basado en funciones y zonas en las que se van a realizar los trabajos.

Una estructura se divide en **dos entregables** del proyecto y el trabajo a realizar en el proyecto en componentes más pequeños y manejables.

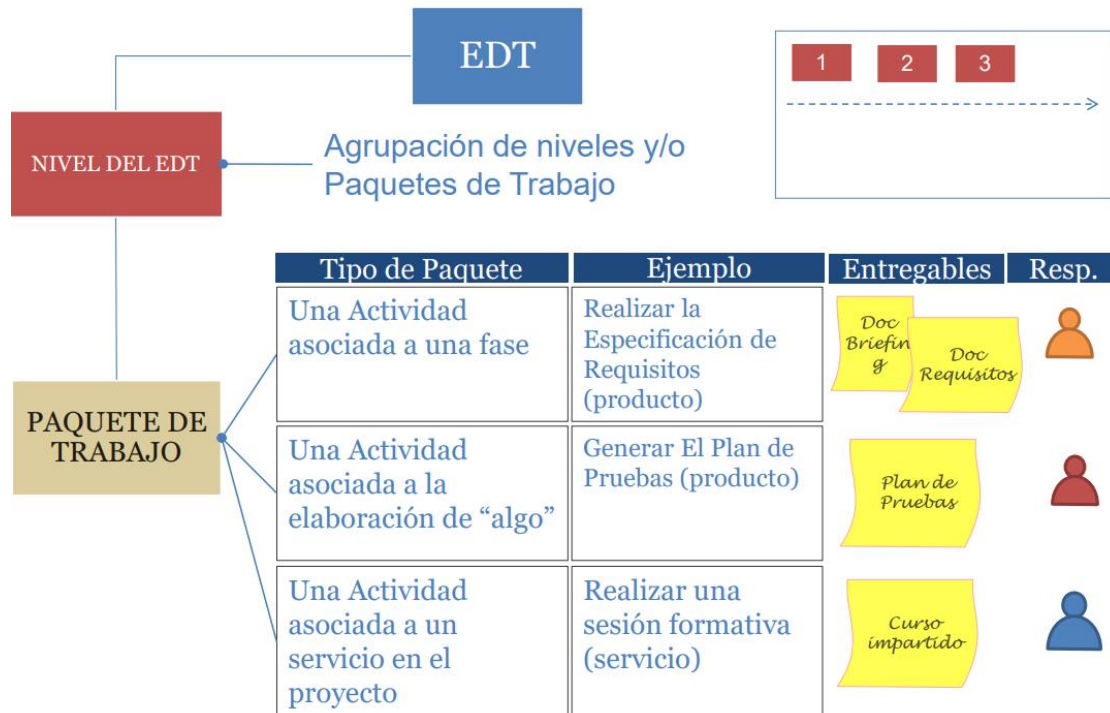
El **Paquete de Trabajo** es el trabajo definido en el nivel más alto del EDT, para el que puede estimarse y gestionarse el coste y la duración.

El **Nivel de Descomposición** es a menudo guiado por el grado del control necesario para gestionar el proyecto de manera efectiva.

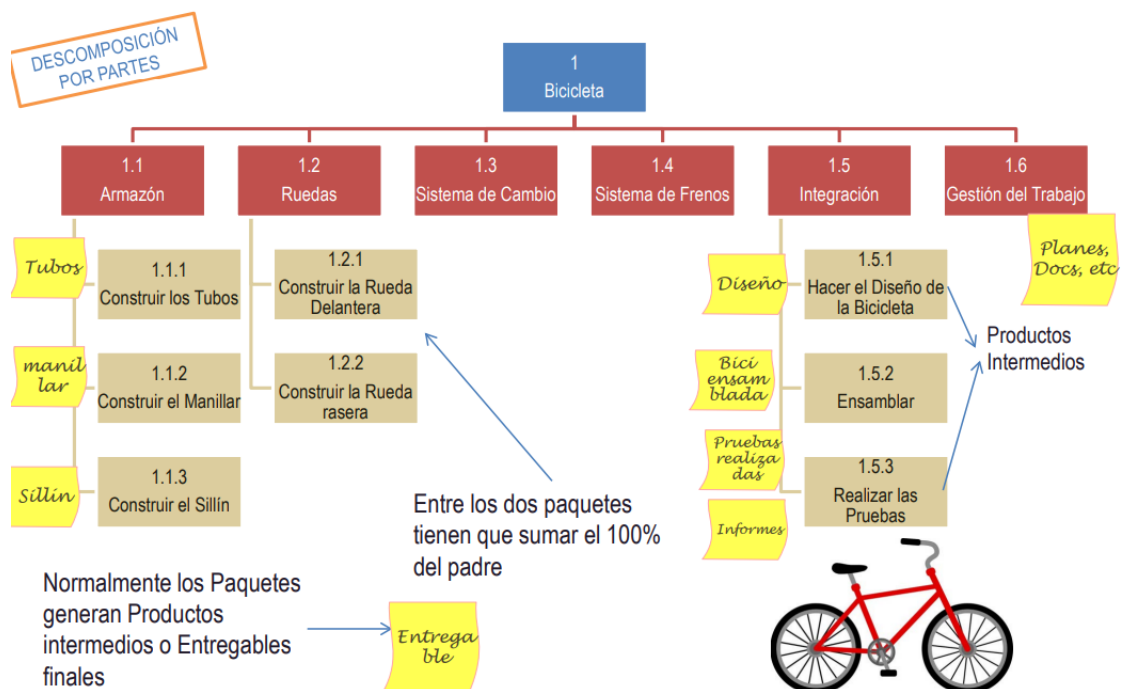
El **Nivel de Detalle** para los paquetes de trabajo variará con el tamaño y complejidad del proyecto.

El EDT debería tener en cuenta los entregables externos o subcontractados, y debe estar enfocado en los entregables.

## Conceptos de un EDT



## Ejemplo de un EDT



Aunque el EDT tenga números en cada nodo, no implica que la numeración sea el orden a seguir en las tareas.

## Recomendaciones para crear un EDT

Algunas recomendaciones a la hora de crear un EDT son las siguientes:

- El paquete debería poder ser estimado, planificado, presupuestado y asignado a un responsable.
- Todo el trabajo del proyecto debe estar incluido en el EDT.
- El EDT debe estar enfocado a los entregables.
- En este nivel se estiman las dedicaciones y los costes, de forma que puedan ser fiables.
- El EDT debería crearse con el equipo, al menos debería ser revisado conjuntamente.
- El EDT se refina conforme el proyecto avanza
- El EDT se suele usar para desarrollar la planificación y el presupuesto
- El EDT debería también tener en cuenta entregables externos o subcontratados.



# Tema 9. El Plan de Proyecto

## Términos en la planificación

Antes de profundizar en el tema, debemos tener en cuenta estos términos:

- **Plan de (Gestión) del Proyecto.** Este es todos los documentos que engloban la planificación y que se usan como base para la ejecución y el control.

A menudo se usa incorrectamente para describir la planificación (schedule) o el plan de trabajo.

A veces se piensa equivocadamente que el software usado para planificar es el mismo que se usa para planificar el proyecto.

- **Planificación o Cronograma del Proyecto.** Muestra cuándo el trabajo se realizará, y quién lo realizará. Se usa para conducir la ejecución del proyecto.

Es clave usar alguna herramienta de creación de planificación para generar la primera línea base y sus posteriores, con herramientas como MSPProject.

Existen carencias sobre la creación de planificaciones, por lo que se debería proporcionar formación a los gestores de proyecto.

- **Plan de Trabajo.** Es un término genérico que se usa para referir cualquiera de los otros, pero se refiere normalmente a la **Planificación del Proyecto**.

Lo suelen usar tanto clientes como proveedores, y el término se debe clarificar.

- **EDT (WBS).** Es la representación jerárquica del trabajo a ser ejecutado. Este es creado normalmente por el propio software usado en la Planificación.

Pueden existir plantillas EDT por defecto para diferentes proyectos, las cuales son la base para el EDT de un nuevo proyecto.

## Composición y Líneas Base del Plan de Proyecto

El plan de proyecto puede ser elaborado...

- A nivel de un resumen.
- Con un alto nivel de detalle.

Adicionalmente, el Plan de proyecto puede estar formado...

- Por un único Plan.
- Por diferentes Planes subsidiarios.

Según PMBOK, el Plan de Gestión del proyecto debe tener incluidas **tres Líneas Base**, las cuales deben ser aprobadas de forma explícita, y si se desea cambiar algo, debe ser realizado mediante una petición de cambio y aprobación explícita

- **Línea Base del Alcance.**
- **Línea Base de la Planificación.**
- **Línea Base del Coste.**

La línea base del alcance contiene el diccionario del EDT, el EDT y el enunciado del alcance.

## Planes Subsidiarios en el Plan

Según PMBOK, podemos encontrar los siguientes planes subsidiarios dentro de un Plan de Proyecto.

Por orden, son los siguientes:

- 1. Plan de Gestión de Alcance.**
- 2. Plan de Gestión de los Requisitos.**
- 3. Plan de Gestión de la Planificación.**
- 4. Plan de Gestión del Coste.**
- 5. Plan de Gestión de la Calidad.**
- 6. Plan de Mejoras de los Procesos.**
- 7. Plan de Gestión de los RRHH.**
- 8. Plan de Gestión de las Comunicaciones.**
- 9. Plan de Gestión de los Riesgos.**
- 10. Plan de Gestión de las Compras y Subcontratación.**
- 11. Plan de Gestión de los Grupos de Interés.**

## Otros documentos del proyecto

El Plan de Proyecto es uno de los primeros documentos, pero también existen otros "documentos de proyecto" que no forman parte del plan:

### Otros Documentos de Proyecto

Documentación de requisitos	Documentos de compras y adquisiciones
Requisitos de financiación del proyecto	Enunciado de Compras de Trabajo
Matriz de trazabilidad de requisitos	Criterios de selección de servicio
Lista de Actividades	Propuestas de proveedores
Atributos de Actividad	Previsiones (de coste y de planificación)
Estimaciones de coste de Actividades	Registros (logs) de cambios
Estimaciones de duración de actividades	Peticiones de cambios
Requisitos de recursos de Actividades	Registro de problemas
Bases de Estimaciones	Listas de comprobación (checklists) de calidad
Listas de Hitos	Mediciones de control de calidad
Calendarios del proyecto	Métricas de calidad
Calendarios de recursos	Registro de riesgos
Datos de planificación	Evaluación del desempeño del equipo
Asignaciones de personas	Datos de rendimiento del trabajo
	Información de rendimiento del trabajo
	Informes de rendimiento del trabajo

## Estructura del Plan de Proyecto

El Plan de Proyecto puede tener la siguiente estructura:

- **Identificación.** Introducción, Código y Nombre del Proyecto.
- **Descripción del Alcance.** Descripción del Proyecto y Esquema de Desglose del Trabajo. Describir como se va a realizar el seguimiento del proyecto.
- **Entregables.** Identificación de los Entregables del Proyecto, sus destinatarios, los responsables de su aceptación, etc.
- **Gestión de Requisitos.** Cómo se van a desarrollar y gestionar los requisitos.
- **Gestión de los Cambios.** Cómo se gestionarán, peticionarios, responsables de gestionarlo, autorizarlos, de implementarlos, etc.
- **Hitos Principales.** Identificar los Hitos principales, entregables, fechas, dependencias, criticidad...
- **Gestión de la Configuración.** Cómo se controlarán los diferentes elementos del proyecto, versiones, líneas base, etc.
- **Gestión de la Planificación.** Cómo se gestionará la Planificación, el cronograma, la asignación de tareas, etc.
- **Gestión de los Costes.** Cómo se gestionarán, responsables de identificarlos, estimarlos y seguirlos.
- **Gestión de la Calidad.** Qué actividades se realizan para asegurar la calidad, estándares, revisiones, inspecciones, trazabilidad, auditorías internas o externas, etc.
- **Gestión del Equipo.** Descripción del equipo y roles, y necesidades formativas.

- **Gestión de los Grupos de Interés.** Cómo se realizará, identificar a los grupos, su relación con el proyecto, y los criterios de éxito.
- **Gestión de las Comunicaciones.** Medios, maneras y canales mediante los cuales el proyecto interactúa con sus Grupos de Interés.
- **Gestión de los Riesgos.** Cómo se realizará, y los responsables de su identificación, análisis, tratamiento y comunicación
- **Gestión de las Compras.** Actividades, responsabilidades, procedimientos, etc.
- **Aceptación.** Quiénes deben comunicar y aceptar el Plan o parte.

NO se incluye la descripción de la arquitectura del software que se va a construir.

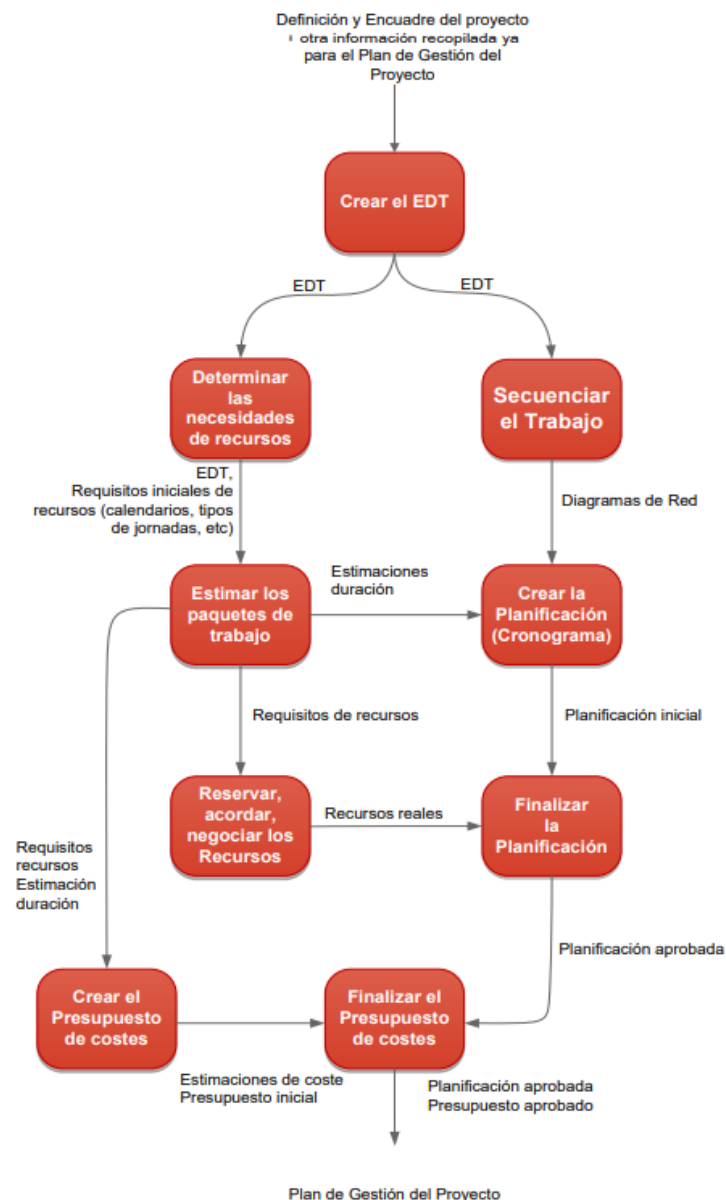
# Tema 10. Planificación o Cronograma

La planificación es un instrumento clave para gestionar la ejecución del Proyecto. En esta se deberán mostrar cuándo y por quién se realizará el trabajo.

Para crear la planificación, **se utiliza el EDT ya creado**. Pero la planificación incluye muchos más detalles, por ejemplo:

- **Dependencias entre Tareas.** Un EDT no las muestra, pero sí una planificación.
- **Tareas planificadas.** Una planificación muestra el inicio y fin de cada tarea, pero el EDT no lo hace.
- **Asignación de Tareas.** Una planificación muestra también los recursos asignados a cada tarea, como personas involucradas, el coste, etc.

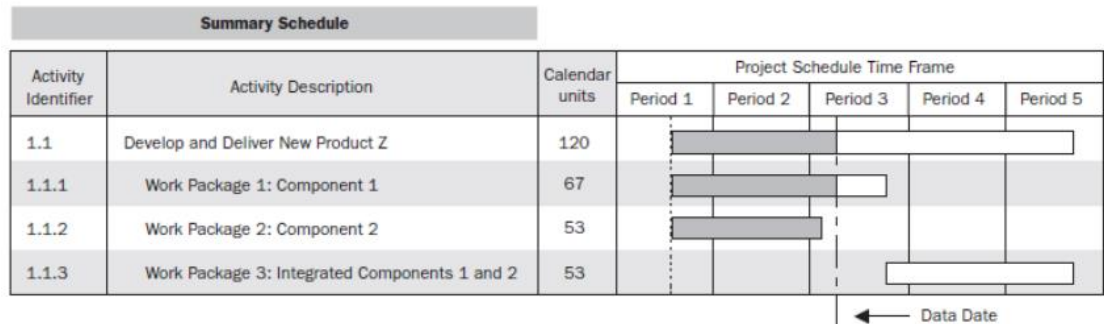
Normalmente, la primera Planificación del Proyecto y el propio Plan de Gestión se realizan de manera paralela, esto hace que la información que se especifica en uno será necesaria para el otro, y viceversa.



## Representaciones de la planificación

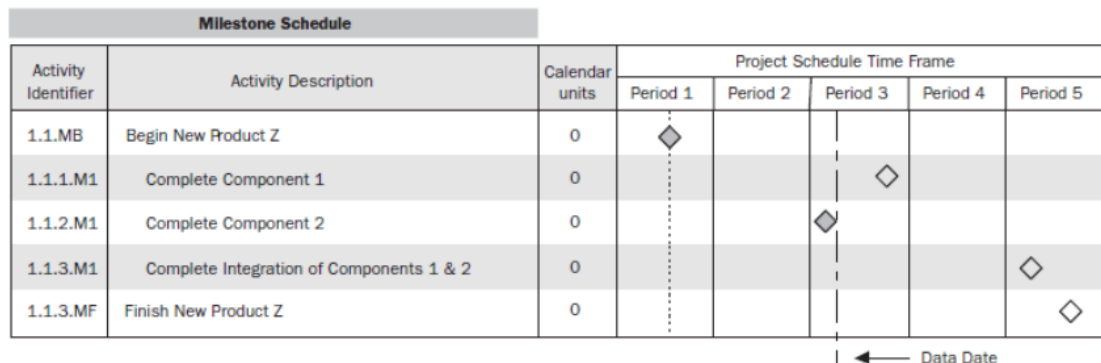
Las planificaciones se pueden representar de diferentes maneras. Entre estas encontramos:

- **Diagramas de Barra.** Son conocidos también como diagramas de Gantt. Son relativamente fáciles de leer, y se usan frecuentemente para presentaciones de seguimiento

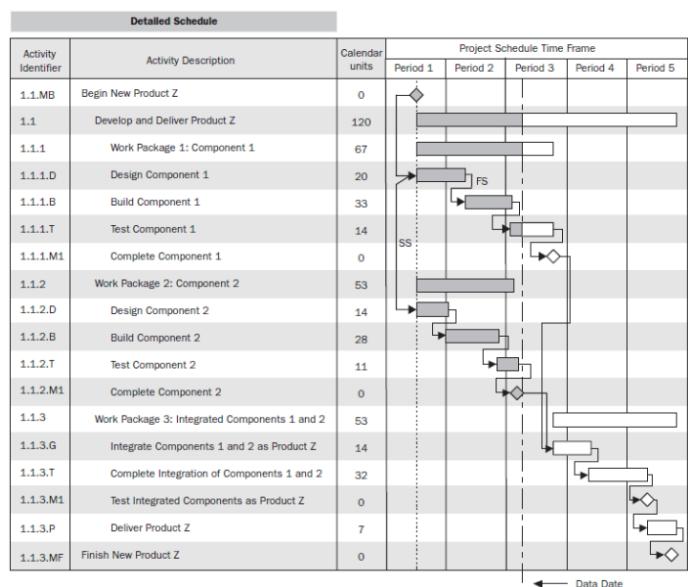


- **Diagrama de Hitos.** Son similares a los de barra, pero únicamente destacan los hitos.

Un hito no es una tarea, sino un movimiento destacado en el proyecto que se quiere destacar, ya que es relevante en el mismo. Este hito puede identificar el inicio o fin de los entregables principales, o la fecha en la que ocurrirá algo a destacar, como una reunión, presentación, entrega, etc. También es un punto que se establece en el que se desea alcanzar algo.



- **Diagramas de Red de la Planificación.** Estos diagramas se presentan de forma gráfica visualizando las actividades mediante barras situadas a lo largo del tiempo, conectadas según las dependencias.



# Tema 11. Estimación de un Proyecto Software

## Horas y Precio

### Necesidades para poner precio a un Proyecto

Para definirlos, tenemos en cuenta que debemos definir diversos costes directos:

- Horas de dedicación al Proyecto, haciendo énfasis en la estimación.
- Precio de cada hora que dedica alguna persona del Equipo del proyecto.

Otros costes directos del proyecto son:

- Material físico que hay que comprar para el proyecto.
- Ordenadores y otros equipos a comprar **específicos para el proyecto**.
- Licencias de software.
- Servicios de uso de recursos en la nube.
- Viajes, costes de desplazamiento, alojamiento, dietas, etc.
- Formación que es necesaria recibir para realizar el proyecto.
- Otros costes más.

### Búsqueda del precio por hora

Tenemos que identificar también los costes indirectos del Proyecto. La empresa tiene muchos costes que no están directamente relacionados al Proyecto. Pueden ser:

- Resto de personas que no trabajan en proyectos software, como los encargados en RRHH, marketing, administración, comerciales, directores, etc.
- Costes generales, como los suministros de luz, agua y gas, alquileres, seguros, bancos, **impuestos**, asesoría, etc.

Para calcular cuanto es el **precio por hora** que debemos usar para un proyecto software, tenemos que pensar en un **salario medio** de una persona en la actividad de desarrollo software, que es aproximadamente de unos 30000€.

Por lo tanto, el coste para la empresa por persona de desarrollo de software es el siguiente:

- 30000€ del salario medio.
- La Seguridad Social que pagar por la persona. Pensando en que es un 33% del salario, entonces es 9900€.
- En total, redondeando sería **unos 40000€**.

Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.



## Técnicas de estimación más conocidas

Algunas son

- **Modelo COCOMO.**
- **Puntos de Función.**
- **Casos de Uso.**
- **Estimación ágil.**

Hablaremos de cada una de las técnicas de estimación

## Modelo COCOMO

En 1981, Barry Boehm publica el modelo COCOMO (**CO**nstructive **CO**st **MO**del), acorde a las prácticas de desarrollo software de la época. Durante la década de los 80, el modelo se continuó perfeccionando y consolidando, siendo el modelo de estimación de costos muy utilizado.

El modelo COCOMO se centra en el uso de las Líneas de Código del proyecto.

COCOMO describe tres modelos diferentes, el cual cada uno de ellos ofrece niveles crecientes de detalle y precisión:

- **Modelo Básico.** Es el modelo COCOMO básico.
- **Modelo Intermedio.** Añade un conjunto de 15 variables de predicción agrupadas en 4 categorías (atributos del **producto software**, del **hardware**, del **personal** y del **proyecto software**).
- **Modelo Detallado.** Se hacen factores de ajuste según módulos y subsistemas y según sus tipos, búsquedas, entrada de datos, etc.

COCOMO permite estimar el **esfuerzo** y el **tiempo** necesarios para desarrollar un proyecto software, y, además, indica como debe distribuirse el tiempo entre las distintas fases del desarrollo. (Análisis de requisitos, diseño del producto, programación, diseño detallado, codificación y pruebas, e integración y pruebas).

## Modos de desarrollo en COCOMO

Un factor primordial que influye en el esfuerzo y tiempo es el **modo de desarrollo**.

- **Modo Orgánico.** El producto a desarrollar es relativamente pequeño, no requiere innovaciones y usa tecnologías conocidas. El personal tiene gran experiencia.
- **Modo Semiacoplado.** El producto es de tamaño y complejidad media, algunos programadores no tienen la suficiente experiencia, y algún requisito tiene un nivel de exigencia alto.
- **Modo Empotrado.** Proyecto complejo a integrar en un entorno crítico con altas restricciones de hardware y software. El equipo tiene experiencia, y se abordan problemas novedosos y se usan tecnologías recientes.

## Modelo Básico COCOMO'81

Estimación del esfuerzo en programadores por mes (PM):

$$PM = A \times (KLOC)^B$$

- KLOC es el tamaño estimado del software en Miles de Líneas de Código (K Lines of Code)
- A y B son los coeficientes que dependen el modo de desarrollo (orgánico, semiacoplado o empotrado).

Número de meses necesarios para terminar el proyecto (TDEV):

$$TDEV = C \times (PM)^D$$

- C y D son los coeficientes que dependen del modo de desarrollo (orgánico, semiacoplado o empotrado).

Los coeficientes para cada modo de desarrollo son los siguientes:

MODO	A	B	C	D
Orgánico	2.40	1.05	2.50	0.38
Semi - Orgánico	3.00	1.12	2.50	0.35
Empotrado	3.60	1.20	2.50	0.33

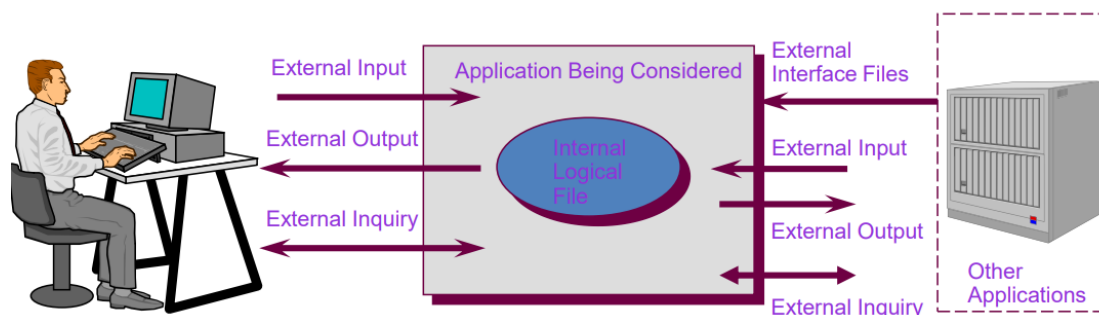
En los años 90, con el cambio del paradigma a la orientación a objetos, el modelo es reinventado y aparece **COCOMO II**.

Se evoluciona generando diferentes versiones del modelo para tipos y fases del proyecto.

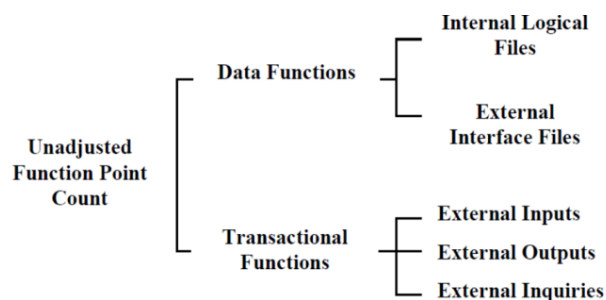
## Modelo de Puntos de Función

### Funcionalidad desde el punto de vista del usuario

La funcionalidad se visualiza como desde el punto de vista del usuario.



Dentro de los distintos puntos de función, podemos distinguir las "funciones de manejo de datos", que serían los **Data Functions**, y las funciones transaccionales o **Transactional Functions**.



## Tipos de Transacciones

- **External Input (EI): Mantiene los archivos de datos (ILF)** mediante creación, actualización y borrado de datos. Son transacciones que llevan datos desde fuera de la aplicación hasta el interior de la aplicación.

Por ejemplo, un botón de OK en una interfaz puede dar lugar a una External Input, puesto que maneja "información de negocio".

- **External Output (EO):** Es una salida formateada de datos de la aplicación. **Los datos deben tener algún procesamiento**, no son simplemente datos existentes en la aplicación.

Por ejemplo, al ver una gráfica con diferentes datos de la base de datos diferenciando diferentes elementos, o tomando distintas mediciones de temperaturas, etc. Se ve que hay un procesamiento de los datos mostrados. Por lo tanto, estamos en un External Output.

Un ejemplo es mostrar las ventas por meses según los sabores de helados, con totales por meses y por sabores.

- **External Inquiry (EQ):** Es una salida de datos formateados **que no tiene ningún procesamiento**, son simplemente datos existentes en la aplicación.

Cuando estamos en una página donde se muestra la información de un elemento de una base de datos, entonces estamos en una External Inquiry.

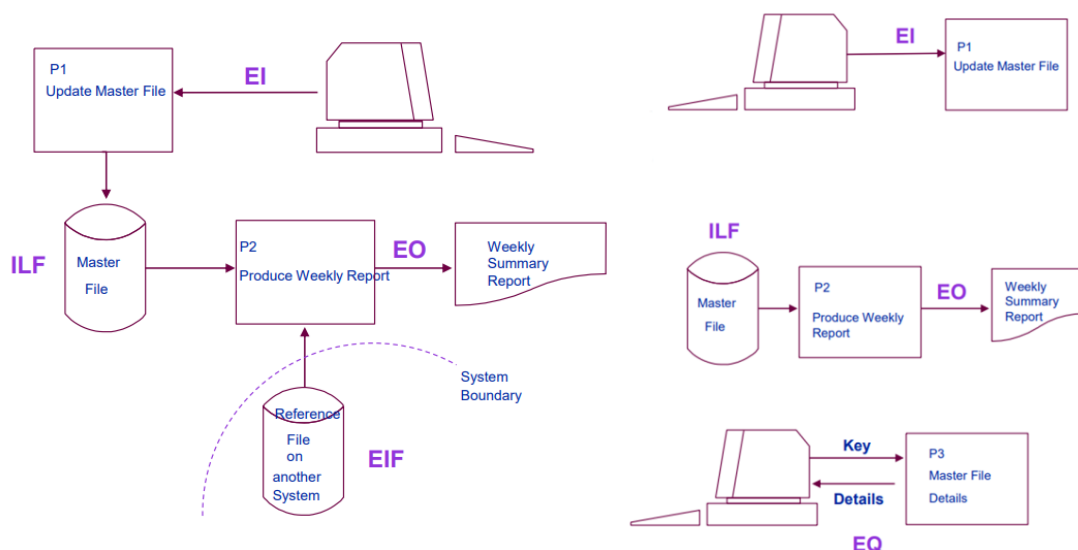
## Almacenamiento de Datos

- **Internal Logical File. (ILF)** Es un grupo lógico de datos que **son gestionados** (mediante un CRUD) por la aplicación.

Por ejemplo, los clientes, pedidos, etc. Son gestionados por la aplicación de la empresa.

- **External Interface File (EIF).** Grupo lógico de datos que se referencia desde la aplicación, pero que **no son gestionados por la aplicación**.

Por ejemplo, en el caso de una administración de hospital, la información clínica del cliente está en otra aplicación.



## Cálculo de los Puntos de Función

Se deben identificar todas las transacciones y archivos de datos en la aplicación. Se clasifican también según su complejidad (baja, media y alta), y se calculan los **Puntos de Función No Ajustados (UFP)** según los pesos de la tabla.

Tipo de Función	Nº de tipo Complejidad Baja	Peso Baja Low	Nº de tipo Complejidad Media	Media Average	Nº de Complejidad Alta	Alta High	Total
EI (External Input)		x 3		x 4		x 6	
EO (External Output)		x 4		x 5		x 7	
EQ (External Query)		x 3		x 4		x 6	
ILF (Internal Logical File)		x 7		x 10		x 15	
EIF (External Interface File)		x 5		x 7		x 10	
Puntos de Función No Ajustados =							

Se debe definir posteriormente un **factor de ajuste (VAF)** que se basa en 14 características generales que sirven para evaluar la complejidad de desarrollar el sistema. Esta tabla es lo que dice el método original de puntos de función, pero lo importante es que **la empresa defina las características que quiere evaluar**.

General System Characteristic		Brief Description
1.	Data communications	How many communication facilities are there to aid in the transfer or exchange of information with the application or system?
2.	Distributed data processing	How are distributed data and processing functions handled?
3.	Performance	Did the user require response time or throughput?
4.	Heavily used configuration	How heavily used is the current hardware platform where the application will be executed?
5.	Transaction rate	How frequently are transactions executed daily, weekly, monthly, etc.?
6.	On-Line data entry	What percentage of the information is entered On-Line?
7.	End-user efficiency	Was the application designed for end-user efficiency?
8.	On-Line update	How many ILF's are updated by On-Line transaction?
9.	Complex processing	Does the application have extensive logical or mathematical processing?
10.	Reusability	Was the application developed to meet one or many user's needs?
11.	Installation ease	How difficult is conversion and installation?
12.	Operational ease	How effective and/or automated are start-up, back up, and recovery procedures?
13.	Multiple sites	Was the application specifically designed, developed, and supported to be installed at multiple sites for multiple organizations?
14.	Facilitate change	Was the application specifically designed, developed, and supported to facilitate change?

Teniendo los puntos, se evalúa cada característica de 0 (sin influencia o no presente) a 5 (gran influencia) y hacemos el cálculo para obtener el valor de ajuste:

$$VAF = 0,65 + \frac{1}{100} \sum_{i=1}^{14} C_i$$

- $C_i$  es el grado de influencia de cada una de las tareas.

Con estos dos valores, tenemos lo siguiente:

$$FP = UFP * VAF$$

Una vez tenemos los puntos de función, debemos calcular el número de horas de dedicación para hacer el trabajo. Para ello debemos tener el valor de horas por cada PF, que es un valor que la empresa decide después de observaciones de proyectos anteriores.

A partir de esto, se calcula en número de horas totales:

$$N^{\circ} \text{ Horas} = \text{Horas}/PF * PF \text{ Totales}$$

Y el precio:

$$\text{Precio Total} = N^{\circ} \text{ Horas} * \text{Precio Medio por hora}$$

## Modelo de Puntos de Casos de Uso

El modelado mediante casos de uso es una técnica muy extendida y aceptada para capturar los procesos de negocio y los requisitos de un proyecto de aplicación software.

Ya que proporcionan un alcance funcional del proyecto, analizarlos nos da una vista muy valiosa para calcular el esfuerzo y el tamaño necesario para diseñar e implementar un proyecto.

En general, los proyectos con mayores y complicados casos de uso necesitan más esfuerzo para su diseño e implementación que los proyectos pequeños. Además, el tiempo requerido para completarlo es afectado por:

- La **extensión de un caso de uso**.
- El **número y complejidad** de los actores.
- Los **requisitos técnicos** de un caso de uso, como concurrencia, rendimiento o seguridad.
- **Varios factores del entorno**, como la experiencia del equipo de desarrollo.

## Variables de PuntosCU

La ecuación de los Puntos de Casos de Uso (PCUs) se compone de 3 variables:

- **Puntos No Ajustados de Casos de Uso (PuntosNCU).**
- **Factor de Complejidad Técnica (FCT).**
- **Factor de Complejidad del Entorno (FCE).**

A su vez, para poder calcular los **PuntosNCU**, necesitamos las siguientes variables:

- **Peso No ajustado de los Actores (PesoNAtores).** Se calcula contando en primer lugar cuántos actores hay, identificándolos en actores simples, medios y complejos, y se multiplica el número de actores de cada equipo por el peso de cada complejidad, y finalmente se suma todo.
- **Peso No ajustado de los Casos de Uso (PesoNCU).** Se calcula contando en primer lugar cuántos casos de uso hay, identificándolos en casos de uso simples, medios y complejos, y se multiplica el número de actores de cada grupo por el peso de cada complejidad, y finalmente se suma todo.

$$\text{PuntosNCU} = \text{PesoNCU} + \text{PesoNAtores}$$

$$\text{PuntosNCU} * FCT * FCE = PCUs$$

Para calcular las horas de desarrollo, se usa un **Factor de Productividad (FP)**. Se usa un valor entre 15 y 30 para el FP dependiendo de la experiencia del equipo y de logros anteriores. Este factor es simplemente una indicación que debe determinar cada empresa. Estas horas se suelen considerar de "desarrollo".

$$\text{Horas} = FP * PCUs$$

## Factor de Complejidad Técnica

Para calcular el **Technical Complexity Factor (TCF)**, se evalúa la complejidad de 0 a 5 de los factores de complejidad técnica, luego se multiplican por los pesos correspondientes, y sale el Factor (TTF) de la fórmula siguiente:

$$TCF = 0.6 + 0.01 * TTF = [0.6 \dots 1.3]$$

Technical Factor	Description	Weight	Perceived Complexity (0-5)	Calculated Factor
T1	Distributed System	2	2	4
T2	Performance	1	1	1
T3	End User Efficiency	1	2	2
T4	Complex Internal Processing	1	2	2
T5	Reusability	1	3	3
T6	Easy to Install	0.5	2	1
T7	Easy to Use	0.5	2	1
T8	Portability	2	0	0
T9	Easy to Change	1	2	2
T10	Concurrency	1	0	0
T11	Special Security Features	1	1	1
T12	Provides Direct Access for Third Parties	1	2	2
T13	Special User Training Facilities Are Required	1	0	0
TECHNICAL TOTAL FACTOR (TTF)				19
TCF Technical Total Factor = $0.6 + 0.01 * TTF$				0.79

## Factor de Complejidad de Entorno

Se evalúan los factores de entorno, se multiplican por sus pesos, y lo que salga se aplica en la siguiente fórmula:

$$ECF = 1.4 - 0.03 * ETF = [0.425 \dots 1.7]$$

## Estimación Ágil

En la estimación ágil se toma en cuenta el concepto de **velocidad**, que es el trabajo realizado por tiempo. Con esto, obtenemos "**Puntos de Esfuerzo**" que el equipo es capaz de producir en una unidad de tiempo. La unidad de tiempo puede ser el sprint o la semana.

Normalmente, Scrum determina el grado de avance del proyecto no por el trabajo realizado, sino **por lo que le queda por realizar**. Esto tiene algunas dificultades al hacer estimaciones

- El trabajo necesario para realizar una historia de usuario o una tarea no se puede prever de forma absoluta.
- No es realista hablar de la cantidad de trabajo que realiza una persona por unidad de tiempo, ya que dos personas son diferentes entre sí
- Una misma tarea puede requerir diferentes tiempos en situaciones distintas
- La complejidad de las técnicas de estimación crece exponencialmente en la medida que se quiere incrementar la fiabilidad y precisión de los resultados, o aumentar el tamaño del trabajo.

Para trabajar con la gestión ágil, se siguen las siguientes estrategias

- Trabajar con estimaciones aproximadas.
- Estima con la técnica de "juicio de expertos".
- Con la participación del equipo.
- Descomponer las tareas en subtarear más pequeñas, especialmente las estimaciones mayores.
- Revisión diaria del progreso del proyecto.

## El punto como unidad de medida

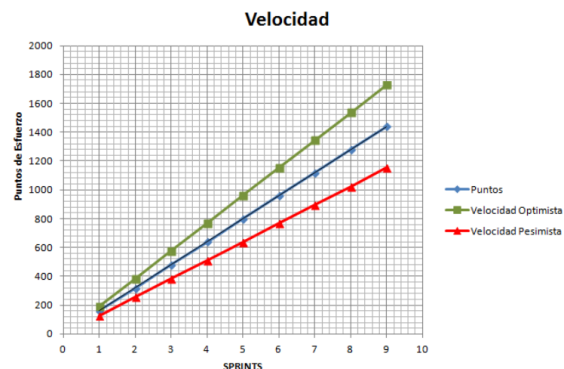
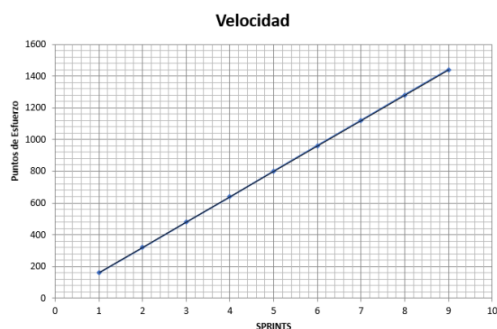
Se usan puntos de historia de usuario, puntos de esfuerzo, puntos de valor proporcionado, etc.

Podemos definir el punto comparando el tamaño de las historias unas con otras.

Es importante que todo el equipo conozca bien las instrucciones y criterios para estimar.

## Velocidad

La velocidad prevista se puede pintar. Por ejemplo, con los siguientes datos (Sprint = 1 semana, equipo de 4 personas, 8 puntos al día, un Sprint de 160 puntos en total) vemos la evolución los puntos respecto a los sprints. A la derecha, un gráfico con velocidad optimista y pesimista (+/-20%)



## El Burndown

Puede pasar que durante el Burndown la cantidad de tareas baje muy rápido o muy lentamente. Esto pasa principalmente si la historia que se esté ejecutando actualmente sea de un gran tamaño (entonces baja lentamente), o que no lo sea (baja rápidamente, pero todavía queda ejecutar las grandes).

Lo recomendable es equilibrar las tareas, y no ejecutar ni la más grande del tirón, ni todas las pequeñas a la vez.

