

Tema 1. Gestión de proyectos

- ✓ Importancia de la gestión
- ✓ Factores que influencian el éxito
- ✓ Personal
- ✓ Problema
- ✓ Proceso
- ✓ Modelos de proceso
- ✓ Actividades de gestión

Bibliografía

[SOM00] Ian Sommerville. "Software Engineering" (7^a ed.). Addison-Wesley. 2005

[PRES98] Roger S. Pressman. Ingeniería del sw. Un enfoque práctico (4^a ed.) Mc Graw-Hill. 1998

Capítulo 2: "El proceso"

Capítulo 3 "Conceptos sobre Gestión de proyectos"

[JAC99] El proceso unificado de desarrollo de software.

Capítulo 1: "El proceso Unificado: dirigido por casos de uso, centrado en la arquitectura, iterativo e incremental".

[PIAT96] Análisis detallado de Aplicaciones informáticas. Mario G. Piattini

Capítulo 3: "Ciclo de vida software"

Importancia de la gestión

Buena gestión = condición **NECESARIA** para el
ÉXITO del proyecto

ES RESPONSABILIDAD DE LOS GESTORES...

- Planificar el proceso de desarrollo
- Hacer un seguimiento del trabajo, de forma que :
 - cumpla los **estándares** establecidos
 - se sigue la **agenda** prevista
 - no se sobrepasa el **presupuesto**

Conceptos sobre gestión (II)

Características especiales del software:

- Producto intangible
- No entendimiento claro sobre el proceso del sw
- Unicidad de los grandes sistemas software

Conceptos sobre gestión (III)

Claves para una **BUENA GESTIÓN**

Personal → Esfuerzo humano intenso

Problema → Minuciosa comunicación con cliente

Proceso → Métodos técnicos y herramientas

Personal

- Instituto de Ingeniería del Software crea “*Modelo de madurez de la capacidad de gestión del personal*”:
- Ayudando a atraer, aumentar, motivar, desplegar y retener el talento.
- Define las siguientes áreas clave:
 - reclutamiento,
 - selección,
 - gestión de rendimiento,
 - entrenamiento,
 - retribución,
 - desarrollo de la carrera,
 - diseño de la organización y del trabajo
 - desarrollo cultural y espíritu de equipo.

Personal

Participantes :

- ✓ Gestores Superiores
- ✓ Gestores Técnicos
- ✓ Profesionales
- ✓ Clientes
- ✓ Usuarios Finales



- Planificar
- Motivar
- Organizar y
- Controlar a los Profesionales

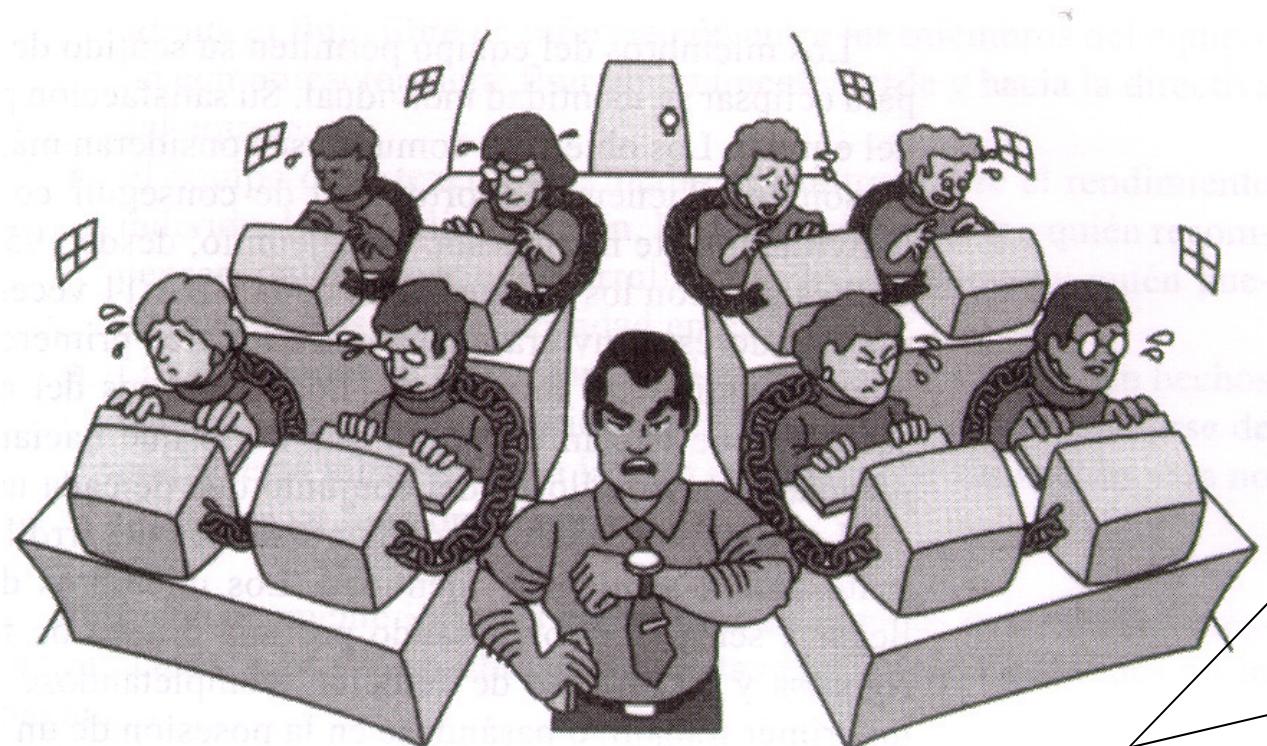
Personal (el jefe de equipo) (I)

Características :

- ✓ Habilidad para motivar
- ✓ Habilidad para moldear procesos (resol. problemas)
- ✓ Habilidad para incentivar la creatividad
- ✓ Dotes de gestión
- ✓ Saber incentivar los logros (increment. productividad)
- ✓ Capacidad para crear un equipo que presente cohesión

Personal (el jefe de equipo) (II)

La forma en que se presente el proyecto determinará si el equipo lo ve como una misión a realizar o como trabajos forzados



“¿Sois un equipo o no?
¿Por qué vamos tan despacio?
Trabajad más duro.”

Personal (el jefe de equipo) (III)



Desarrollador de software bien estimado

Personal (el equipo de software) (I)

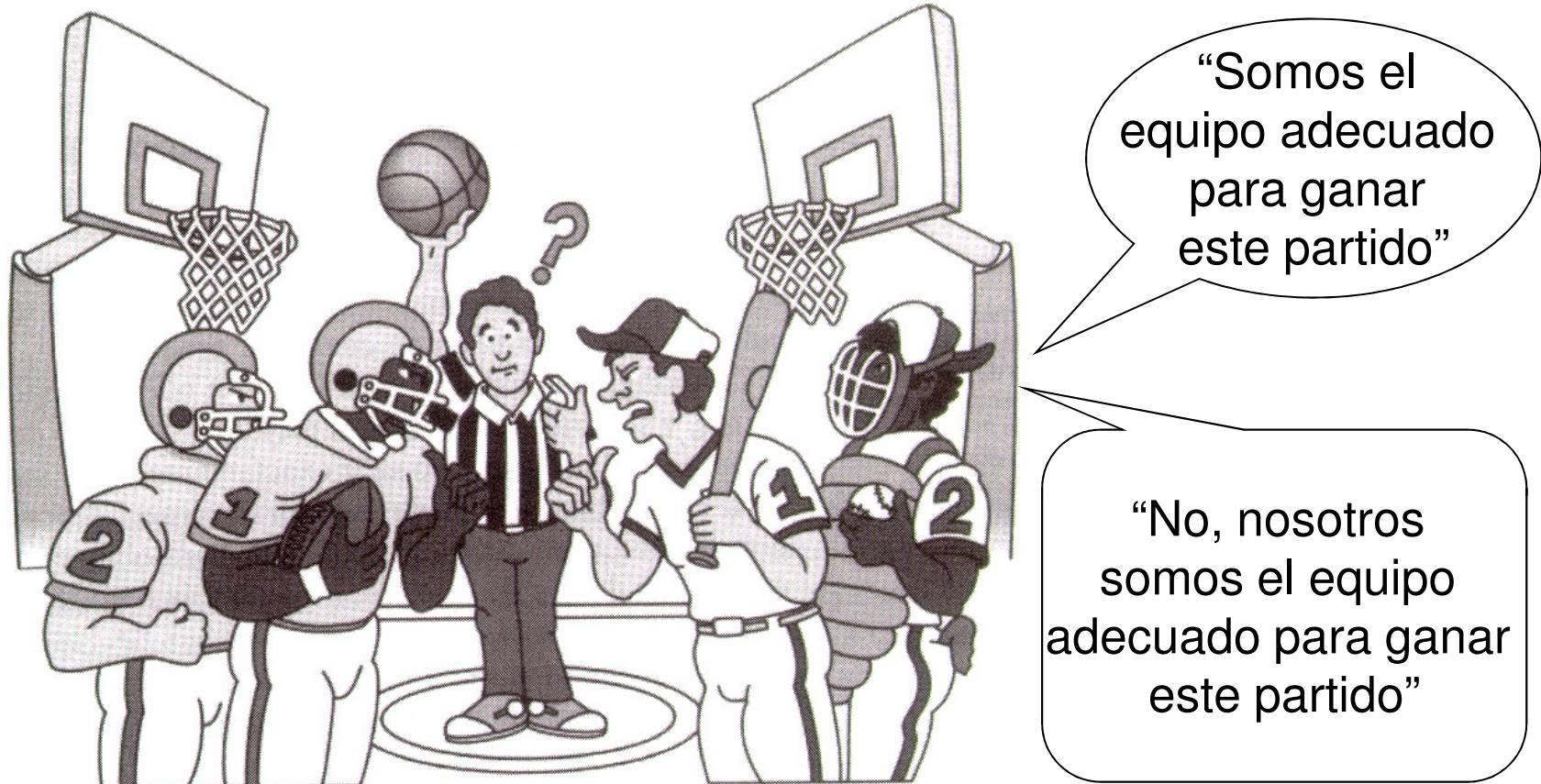
Organización del personal :

- ✓ **N** individuos asignados a **m** tareas funcionales ($m \geq N$)
- ✓ **N** individuos asignados a **m** tareas funcionales ($m < N$)
- ✓ **N** individuos organizados en **t** equipos

Estructuras de equipo :

- ✓ Descentralizado democrático (DD)
- ✓ Descentralizado controlado (DC)
- ✓ Centralizado controlado (CC)

Personal (el equipo de software) (II)



No hay una estructura única de equipo mejor para todos los proyectos

Personal (el equipo de software) (III)

El rendimiento de un equipo es INVERSAMENTE proporcional a la cantidad de comunicación que se deba establecer

El tiempo que los miembros del equipo vayan a “vivir juntos” afecta a la moral del equipo

Factores útiles para seleccionar personal son: experiencia en el dominio, adaptabilidad y personalidad.

Personal (el equipo de software) (IV)

Factores a tener en cuenta:

- ✓ Dificultad y tamaño del problema a resolver
- ✓ Tiempo de vida del grupo
- ✓ Grado de modularización del problema
- ✓ Calidad y fiabilidad requerida del sistema
- ✓ Rrigidez del tiempo de entrega
- ✓ Grado de comunicación requerida por el proyecto

Personal (el equipo de software) (VI)

		DD	DC	CC
DIFICULTAD	ALTA	X		
	PEQUEÑA		X	X
TAMAÑO	GRANDE		X	X
	PEQUEÑO	X		
DURACIÓN DEL EQUIPO			X	X
	CORTO			
	LARGO	X		
MODULARIDAD	ALTA		X	X
	BAJA	X		
FIABILIDAD	ALTA	X	X	
	BAJA			X
FECHA DE ENTREGA				X
	EXTRICTA			
	FLEXIBLE	X	X	
COMUNICACIÓN				
	ALTA	X		
	PEQUEÑA		X	X

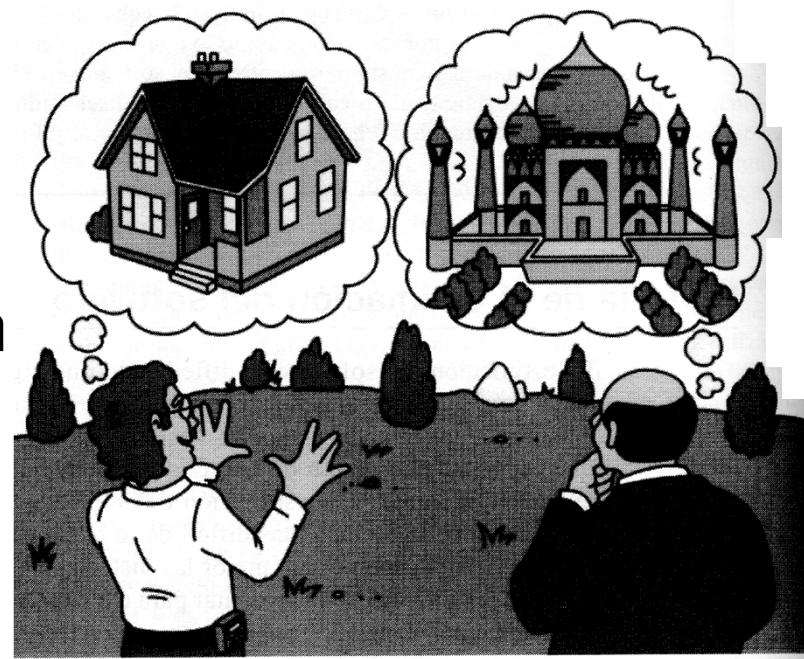
Problema

Para planificar un proyecto se requieren estimaciones
CUANTITATIVAS

Pasos:

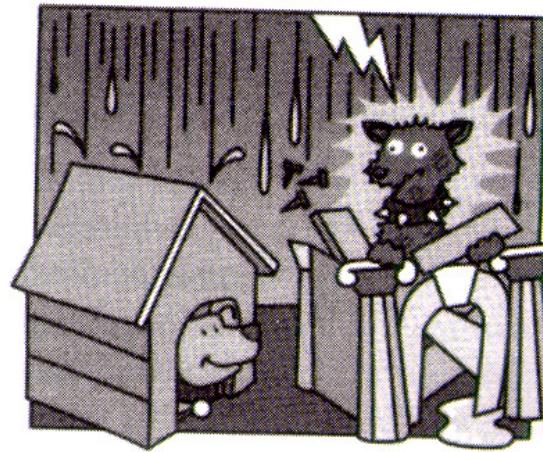
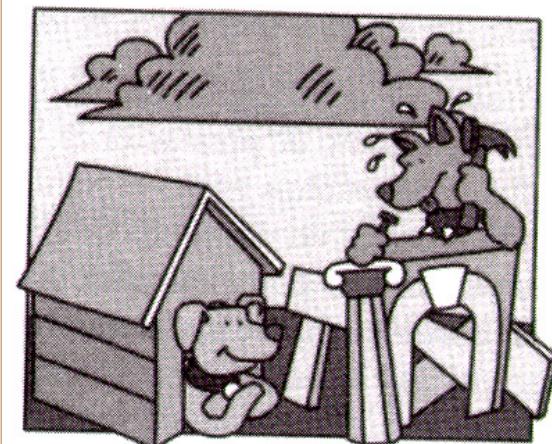
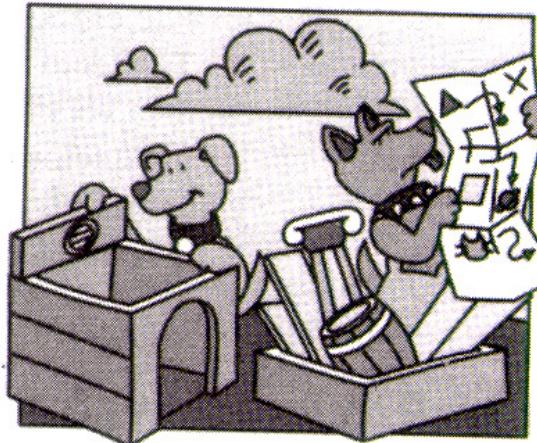
- ✓ Ámbito del software
 - Contexto
 - Objetivos de información
 - Función y rendimiento

- ✓ Descomposición del problema
 - Funcionalidad
 - Proceso



Proceso

Se debe seleccionar el modelo de proceso apropiado

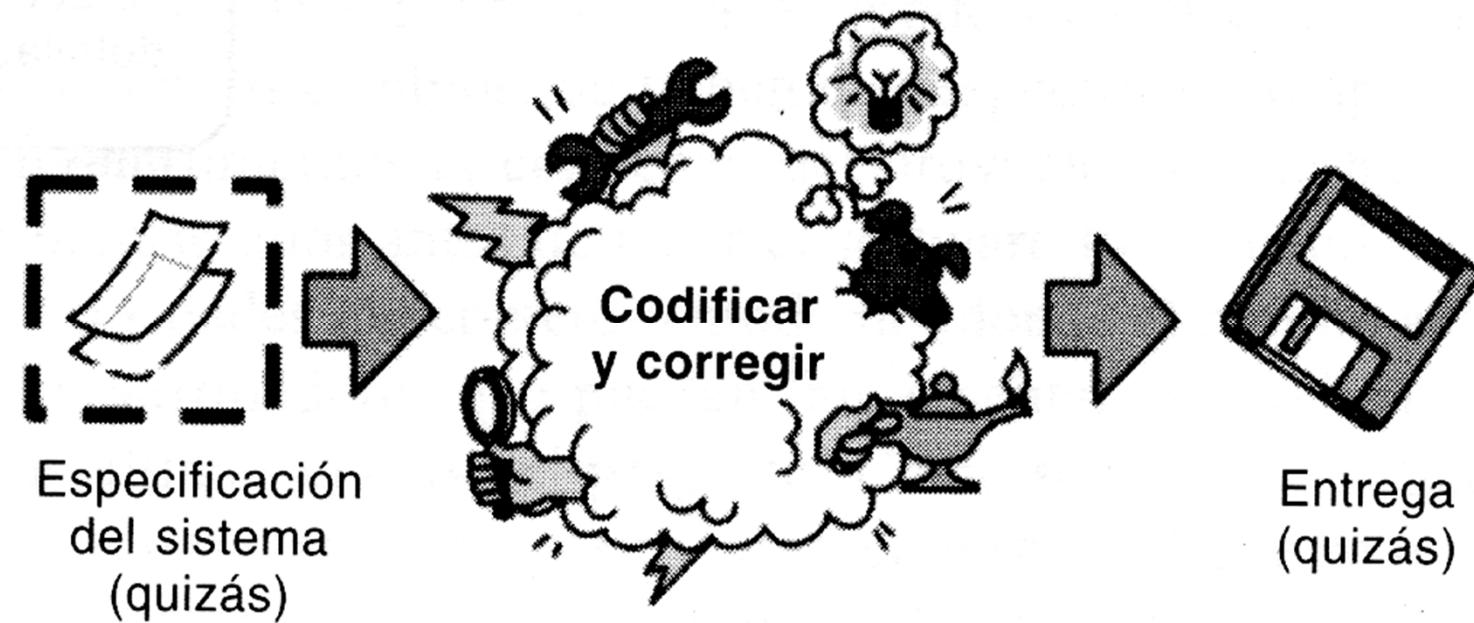


para la ingeniería
del software que
debe aplicar el
equipo del proyecto

Los proyectos
pequeños
necesitan
menos tiempo
para su
desarrollo

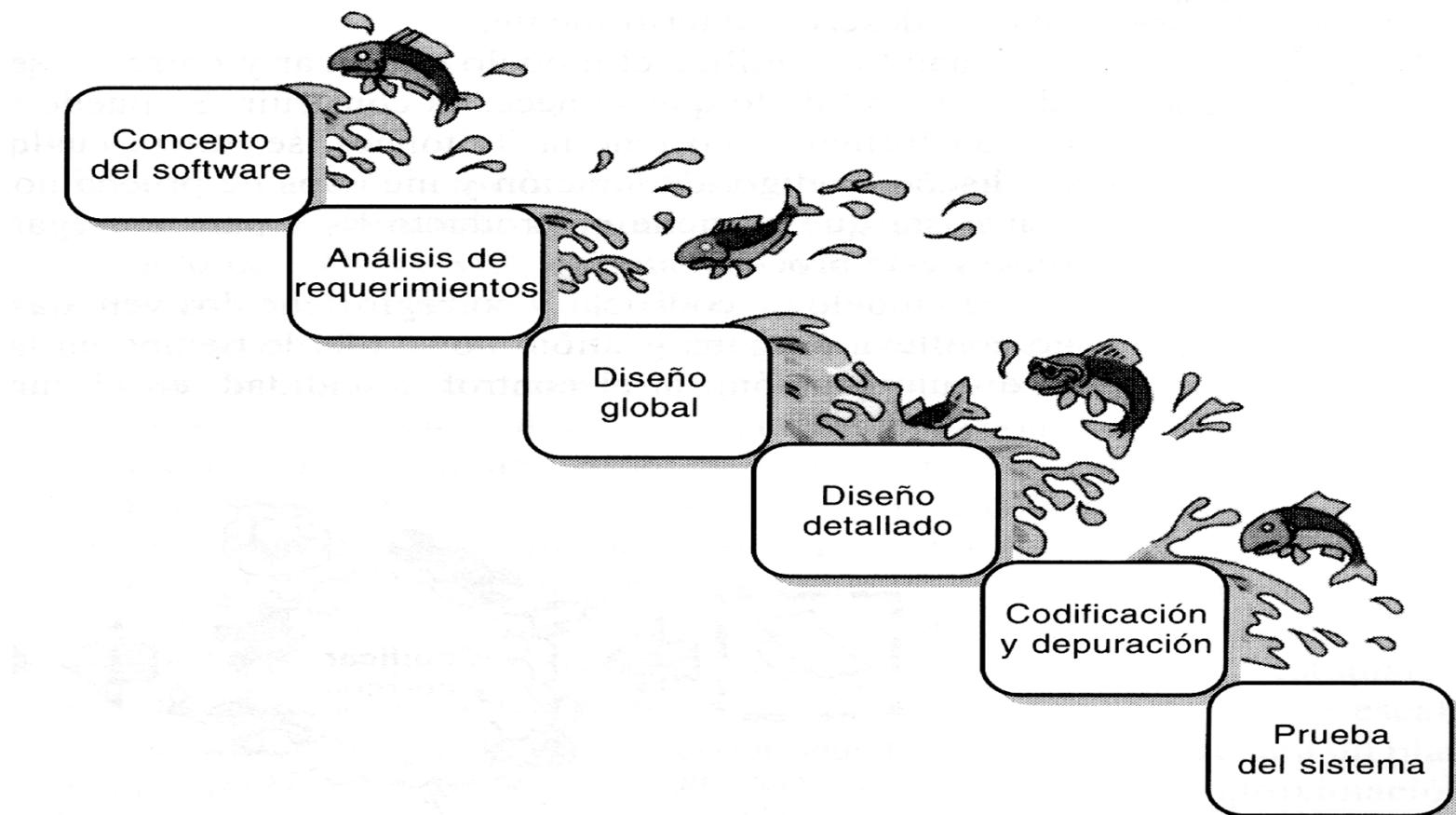
Modelos de Proceso

MODELO CODIFICAR Y CORREGIR



Modelos de Proceso Secuenciales (I)

MODELO DE CICLO DE VIDA EN CASCADA



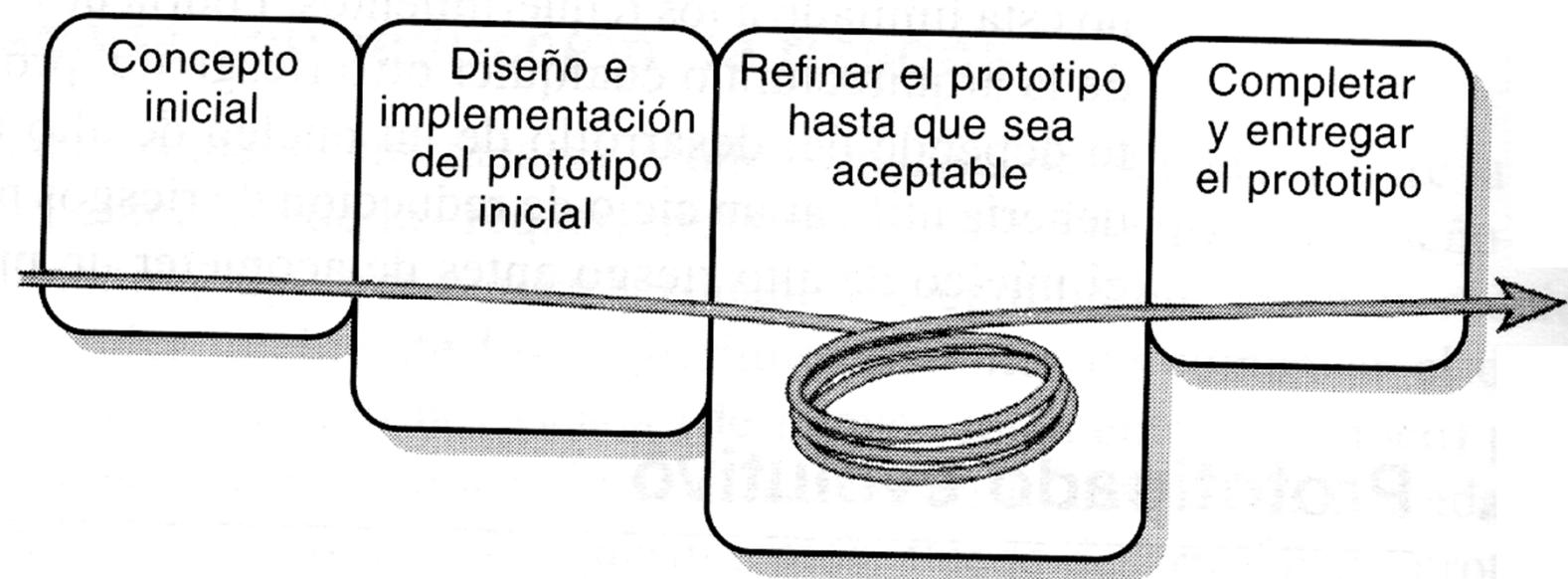
Modelos de Proceso Secuenciales (II)

CRÍTICAS AL MODELO DE CICLO DE VIDA EN CASCADA

- Acentúa el fracaso de la industria software frente al usuario final.
- Se tarda mucho tiempo en pasar por todo el ciclo, dado que hasta que no se finalice una fase no se pasa a la siguiente.
- No refleja el proceso real de desarrollo software. Los proyectos reales raramente siguen este flujo secuencial , puesto que siempre hay iteraciones.

Modelos de Proceso Secuenciales (III)

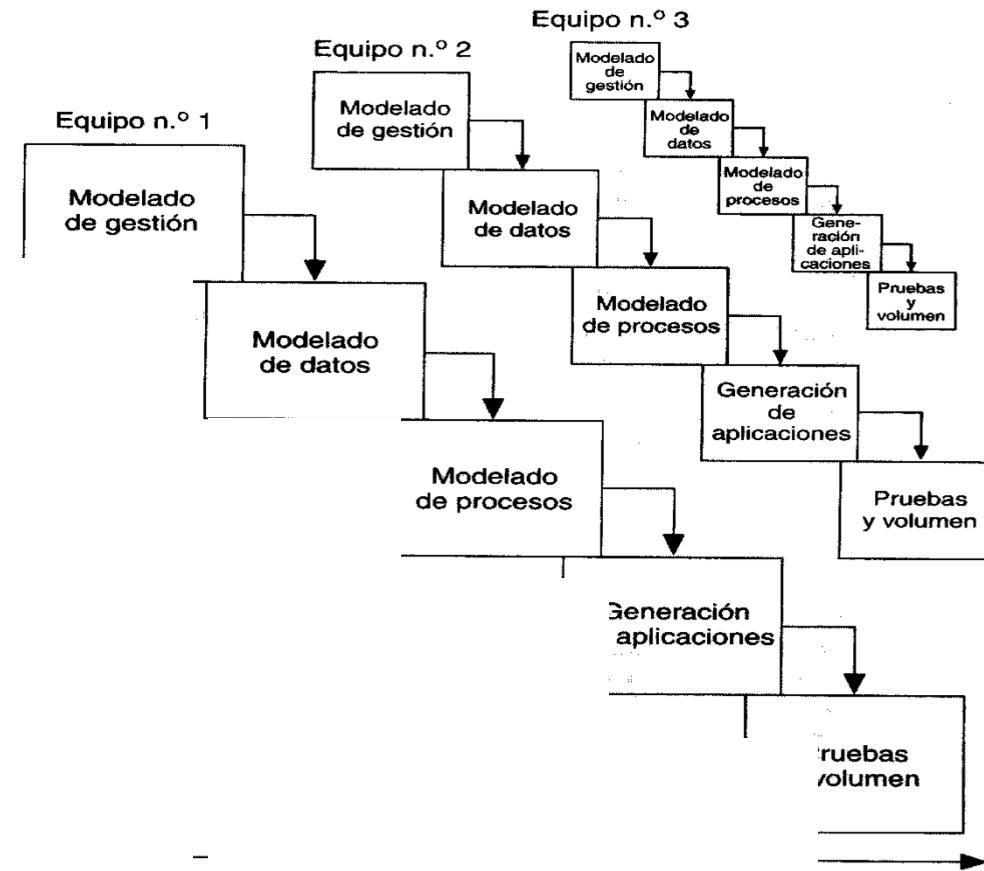
MODELO CONSTRUCCION DE PROTOTIPOS



Modelos de Proceso Secuenciales (VI)

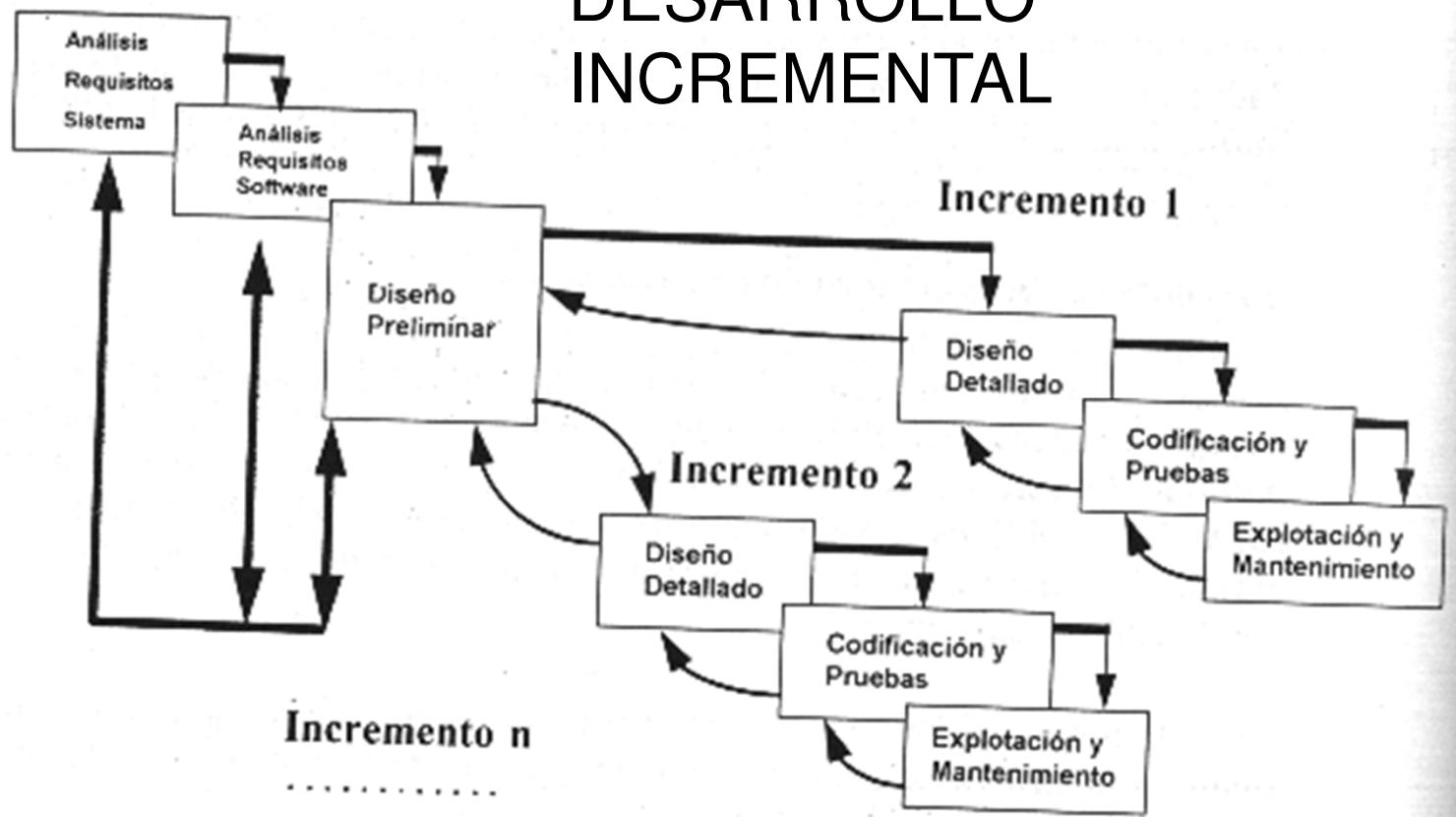
MODELO DE DESARROLLO RAPIDO DE APLICACIONES (DRA)

El modelo DRA



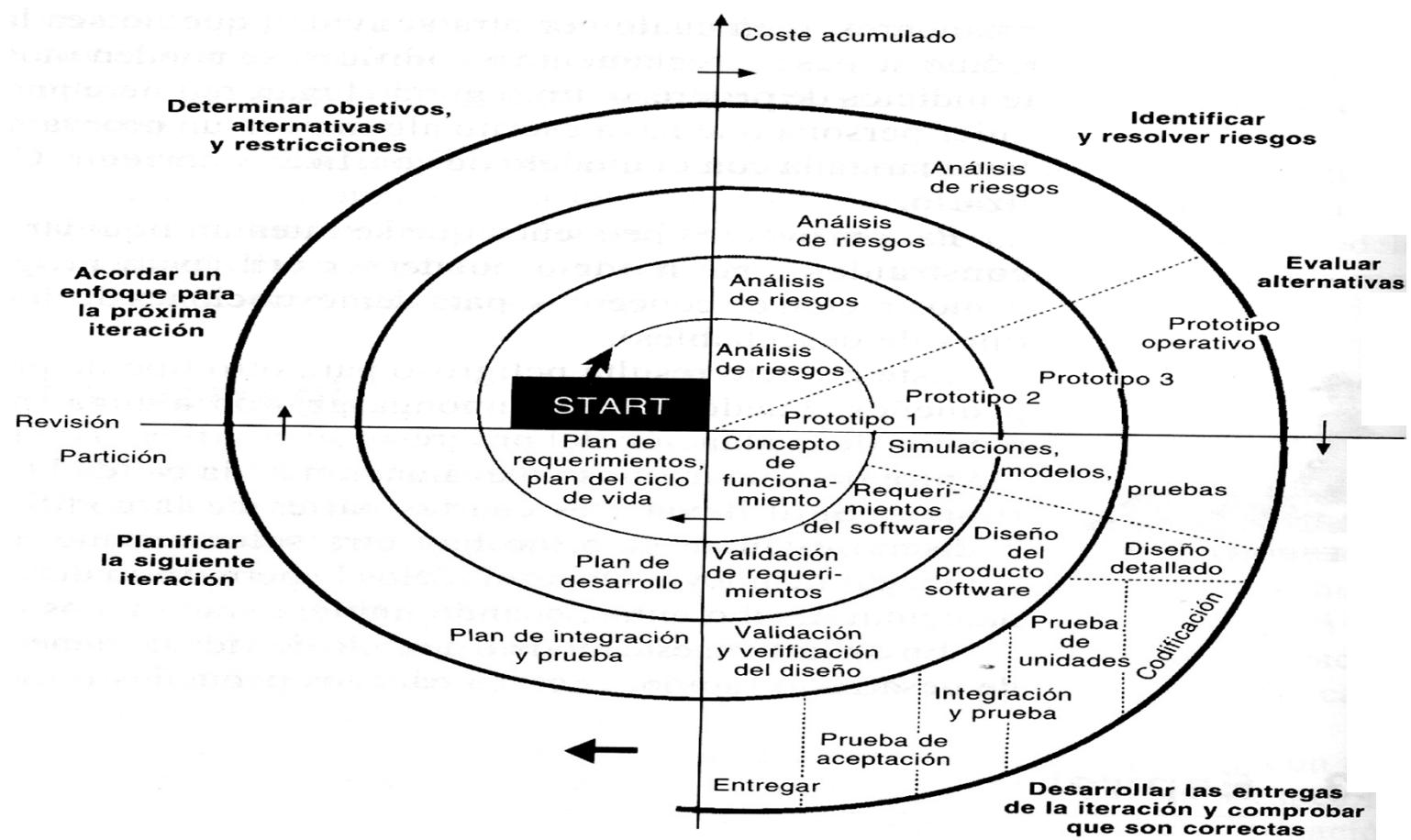
Modelos de Proceso Evolutivos (I)

MODELO DE DESARROLLO INCREMENTAL



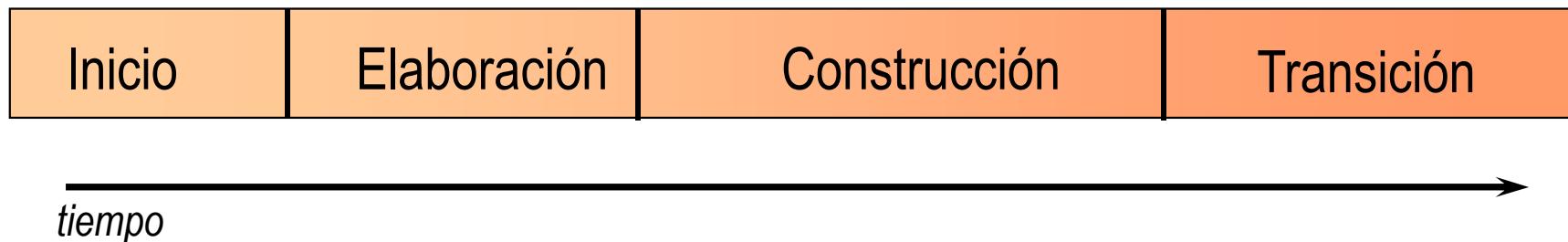
Modelos de Proceso Evolutivos (II)

MODELO EN ESPIRAL



Modelos de Proceso Evolutivos (III). (cont.)

DESARROLLO UNIFICADO

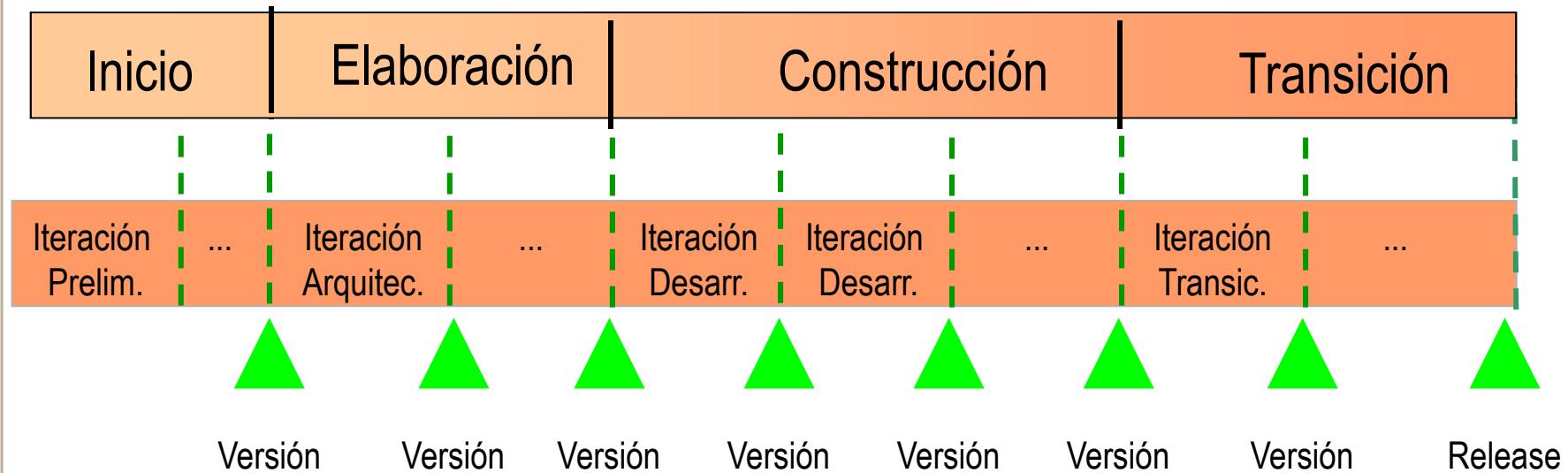


- **Inicio:** Se define el ámbito del proyecto y se desarrollan los casos de uso
- **Elaboración:** Se realiza el plan del proyecto, estimaciones, diseño básico
- **Construcción:** Se implementa en base a iteraciones
- **Transición:** Fase de transición para entregar el producto a los usuarios (p.ej. pruebas beta)

Modelos de Proceso Evolutivos

(III) (cont.)

DESARROLLO UNIFICADO



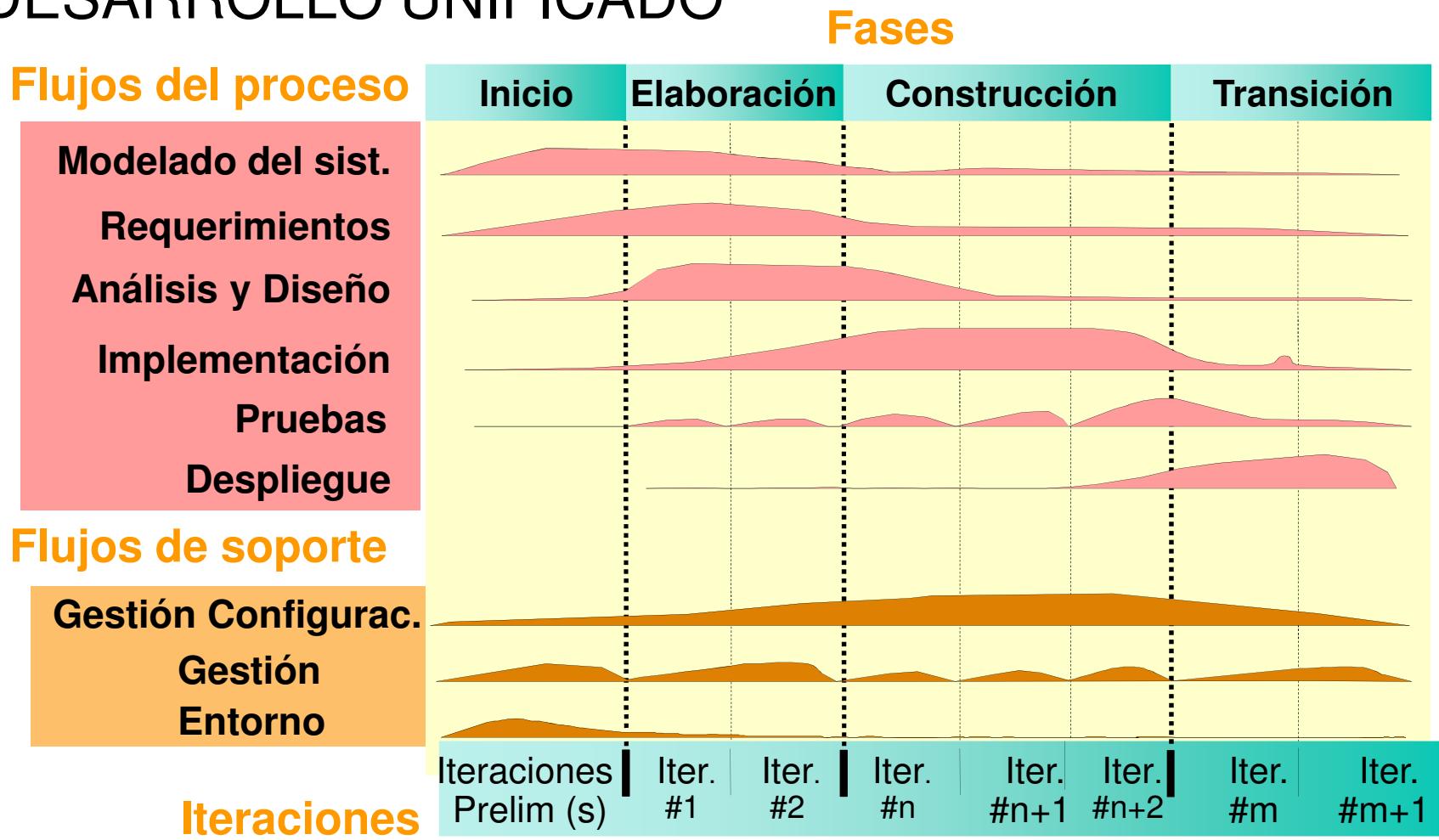
Una iteración es una secuencia de actividades con un plan establecido y un criterio de evaluación, que resulta en una *versión interna nueva (incremento)*

Al final de un ciclo obtenemos una versión para el cliente (release).

Modelos de Proceso Evolutivos

(III) (cont.)

DESARROLLO UNIFICADO



Modelos de Proceso Evolutivos (III)

DESARROLLO UNIFICADO: Conceptos clave

- Fases, Iteraciones
- Flujos del proceso
 - Actividades, pasos
- Artefactos
 - modelos
 - Informes, documentos
- Trabajador: Ingeniero

¿Cuándo tienen lugar?

¿Qué hay que hacer?

¿Qué se produce?

¿Quién lo hace?

Problema y proceso

Actividades del proceso	comunicación cliente	planificación	análisis de riesgos	ingeniería	
Tareas					
Funciones del producto					
Función 1					
Función 2					
Función 3					
...					

Para cada celda, estimar los requisitos de recursos, poner fechas a las tareas y determinar los productos a obtener

Actividades de gestión



- Calidad del producto
- Evaluación del riesgo
- Métricas
- Estimación de costes
- Confección de agendas
- Comunicación cliente
- Personal
- Otros recursos
- Monitorización del proyecto

Tema 2. Estimación de costes

-  Introducción
-  Productividad
-  Técnicas de estimación
-  Modelo algorítmico de costes
-  Duración y personal del proyecto

Bibliografía

Captítulo 26. Software cost estimation. Software
Ingeniering Sommerville 7^a edición.

Capítulo 18. Métricas del software. Ingeniería del
software. 4^a edición. Roger S. Pressman.

Introducción



¿Qué es la estimación de costes?

Consiste en predecir los **recursos** (monetarios, temporales, humanos, materiales, ...) necesarios para llevar a cabo el proceso de desarrollo del software.



Cuestiones fundamentales:

- ¿Cuánto esfuerzo es necesario para completar una actividad?
- ¿Cuánto tiempo se necesita para completar una actividad?
- ¿Cuál es el coste total de una actividad?

Componentes de coste

- Costes hardware y software
- Costes de viajes y aprendizaje
- Costes de **esfuerzo** (factor dominante casi siempre)
 - sueldo ingenieros del proyecto
 - gastos seguros y seguridad social
- Otros costes:
 - costes de alquiler, calefacción y luz
 - costes de redes y comunicaciones
 - costes de recursos compartidos (p.e. librería, personal del restaurante, etc.)

Factores de coste

Factor	Description
Market opportunity	A development organisation may quote a low price because it wishes to move into a new segment of the software market. Accepting a low profit on one project may give the opportunity of more profit later. The experience gained may allow new products to be developed.
Cost estimate uncertainty	If an organisation is unsure of its cost estimate, it may increase its price by some contingency over and above its normal profit.
Contractual terms	A customer may be willing to allow the developer to retain ownership of the source code and reuse it in other projects. The price charged may then be less than if the software source code is handed over to the customer.
Requirements volatility	If the requirements are likely to change, an organisation may lower its price to win a contract. After the contract is awarded, high prices may be charged for changes to the requirements.
Financial health	Developers in financial difficulty may lower their price to gain a contract. It is better to make a small profit or break even than to go out of business.

Productividad (I)

- La productividad de un programador es una medida de la "velocidad" a la que los ingenieros implicados en el desarrollo del software producen dicho **software** y su **documentación** asociada
- Es necesario estimar la productividad:
 - para realizar las estimaciones necesarias en el proyecto
 - para evaluar si un proceso o mejoras en la tecnología son efectivas.

Productividad (II)

$$\text{Productividad} = \frac{\text{atributos del software}}{\text{esfuerzo total de desarrollo}}$$

Tipos de medidas:

- Relacionadas con el tamaño (líneas de código)
- Relacionadas con la funcionalidad (puntos de función, puntos de objeto)

Líneas de código



¿Qué es una línea de código?

- Es una medida propuesta inicialmente cuando los programas se escribían en tarjetas, con una línea por tarjeta
- Actualmente los lenguajes permiten escribir varias sentencias en una línea, o una misma sentencia en varias líneas
- Se debe decidir qué programas deberían contarse como parte del sistema
- Asumen una relación lineal entre el tamaño y el volumen de documentación

Comparaciones de productividad

- Cuanto mayor sea la expresividad del lenguaje, más baja será su productividad aparente.
- Cuánto más líneas de código emplee el programador, mayor será su productividad

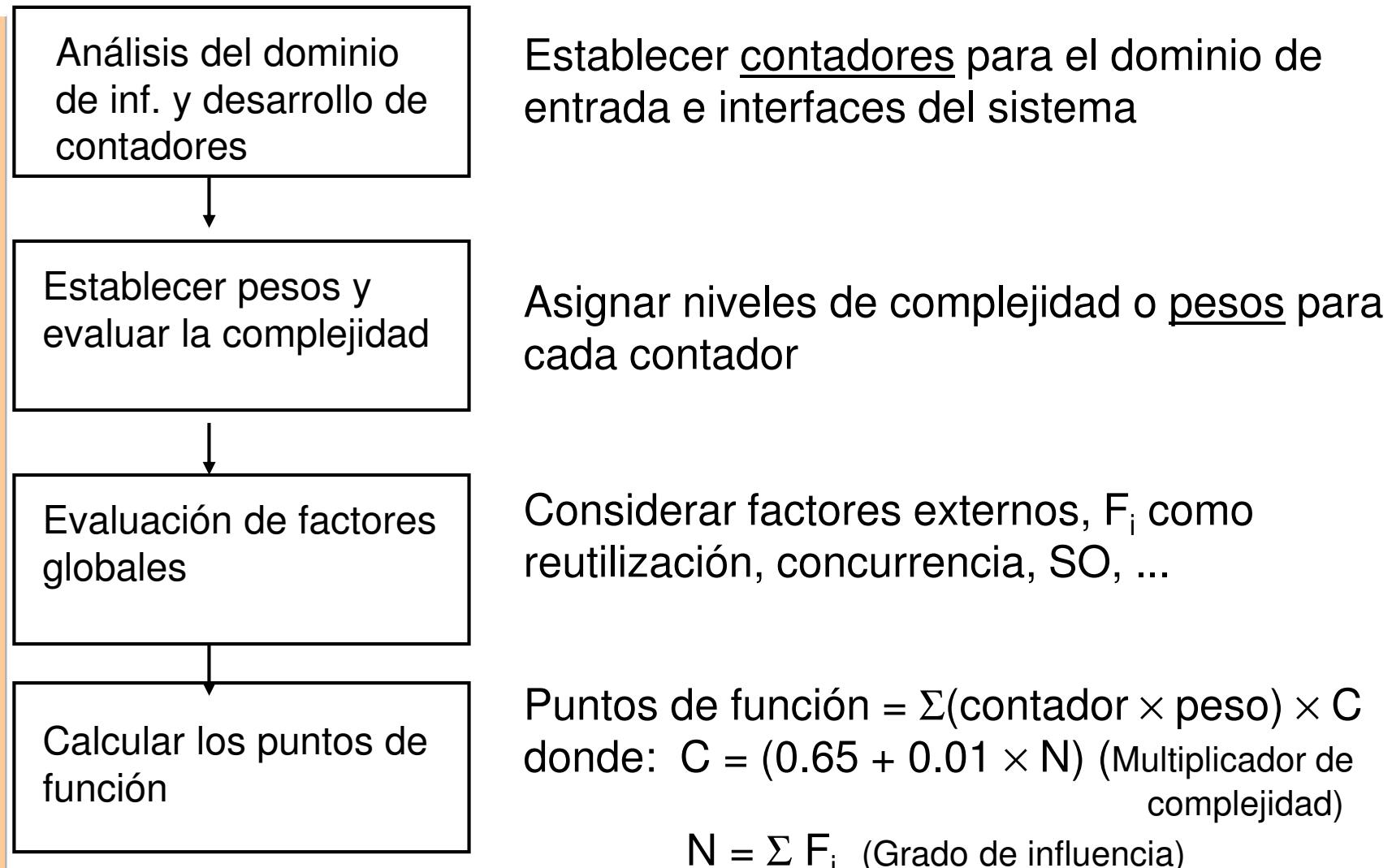
	Analysis	Design	Coding	Testing	Documentation
Assembly code	3 weeks	5 weeks	8 weeks	10 weeks	2 weeks
High-level language	3 weeks	5 weeks	8 weeks	6 weeks	2 weeks
	Size		Effort	Productivity	
Assembly code	5000 lines		28 weeks	714 lines/month	
High-level language	1500 lines		20 weeks	300 lines/month	

Comparar la productividad utilizando lenguajes diferentes de programación puede llevar a conclusiones erróneas respecto a la productividad de los programadores

Puntos de función

-  Basados en una combinación de características del programa
 - entradas y salidas externas
 - interacciones de usuario
 - interfaces externas
 - ficheros usados por el sistema
-  Se asocia un peso con cada uno de ellos
-  Los puntos de función se calculan multiplicando cada factor por su peso y sumando todos ellos

Cálculo de los puntos de función(I)



Cálculo de los puntos de función(II)

<u>Parámetro de medida</u>	<u>Contador</u>	<u>Factor de peso</u>			
		Simp.	Med.	Compl.	
Nº entradas usuario	<input type="text"/>	× 3	4	6	= <input type="text"/>
Nº salidas usuario	<input type="text"/>	× 4	5	7	= <input type="text"/>
Nº peticiones usuario	<input type="text"/>	× 3	4	6	= <input type="text"/>
Nº ficheros	<input type="text"/>	× 3	4	6	= <input type="text"/>
Nº interfaces externas	<input type="text"/>	× 7	10	15	= <input type="text"/>
Total contadores	—————				<input type="text"/>
Multiplicador de complejidad	—————				<input type="text"/>
Puntos de función	—————				<input type="text"/>

Ventajas de los puntos de función

- Son independientes del lenguaje de programación
- Pueden calcularse a partir de la especificación
- Usa información del dominio del problema
- Resulta más fácil a la hora de reusar componentes
- Se encamina a aproximaciones orientadas a objetos

Puntos de función y estimación

- Los puntos de función (FP) pueden usarse para estimar el **número de líneas de código** (LOC) dependiendo del número medio de LCDs por PF para un lenguaje dado (AVC)
 - LOC = AVC * número de puntos de función
 - AVC es un factor dependiente del lenguaje que varía desde 200-300 para lenguaje ensamblador hasta 2-40 para un lenguaje 4GL
- Los puntos de función son muy **subjetivos**.
Dependen del estimador.

Puntos de objeto

- Los puntos de objeto son una medida alternativa relacionada con la funcionalidad cuando se utilizan lenguajes 4GLs o similares para el desarrollo
- Los puntos de objeto **NO** son clases de objetos
- El número de puntos de objeto en un programa es una estimación ponderada de:
 - El número de pantallas que son visualizadas por separado
 - El número de informes que se producen por el sistema
 - El número de módulos 3GL que deben desarrollarse para complementar el código 4GL

Estimación de puntos de objeto

- Son más fáciles de estimar a partir de una especificación que los puntos de función, ya que solamente consideran pantallas, informes y módulos 3GL
- Por lo tanto pueden estimarse en fases tempranas del proceso de desarrollo. En estas etapas resulta muy difícil estimar el número de líneas de código de un sistema

Factores que afectan la product.

-  Experiencia en el dominio de la aplicación
-  Calidad del proceso
-  Tamaño del proyecto
-  Tecnología de soporte
-  Entorno de trabajo

Calidad y productividad

- Todas las métricas basadas en volumen/unidad de tiempo son engañosas debido a que no tienen en cuenta la calidad
- La productividad puede incrementarse generalmente a costa de la calidad
- No está claro cómo la productividad y las métricas de calidad están relacionadas
- Las métricas de productividad deberían usarse únicamente como guía

Técnicas de estimación (I)

 No existe una forma simple de obtener estimaciones exactas del esfuerzo requerido para desarrollar un sistema software

- Las estimaciones iniciales se basan en información incompleta en la definición de requerim. del usuario
- El software puede tener que ejecutarse sobre ordenadores no usuales o usar nuevas tecnologías
- Puede desconocerse a la gente que interviene en el proy.

Técnicas de estimación (II)

 Modelado algorítmico de costes

 Juicio experto

 Estimación por analogía

 Ley de Parkinson

 *Pricing to win*

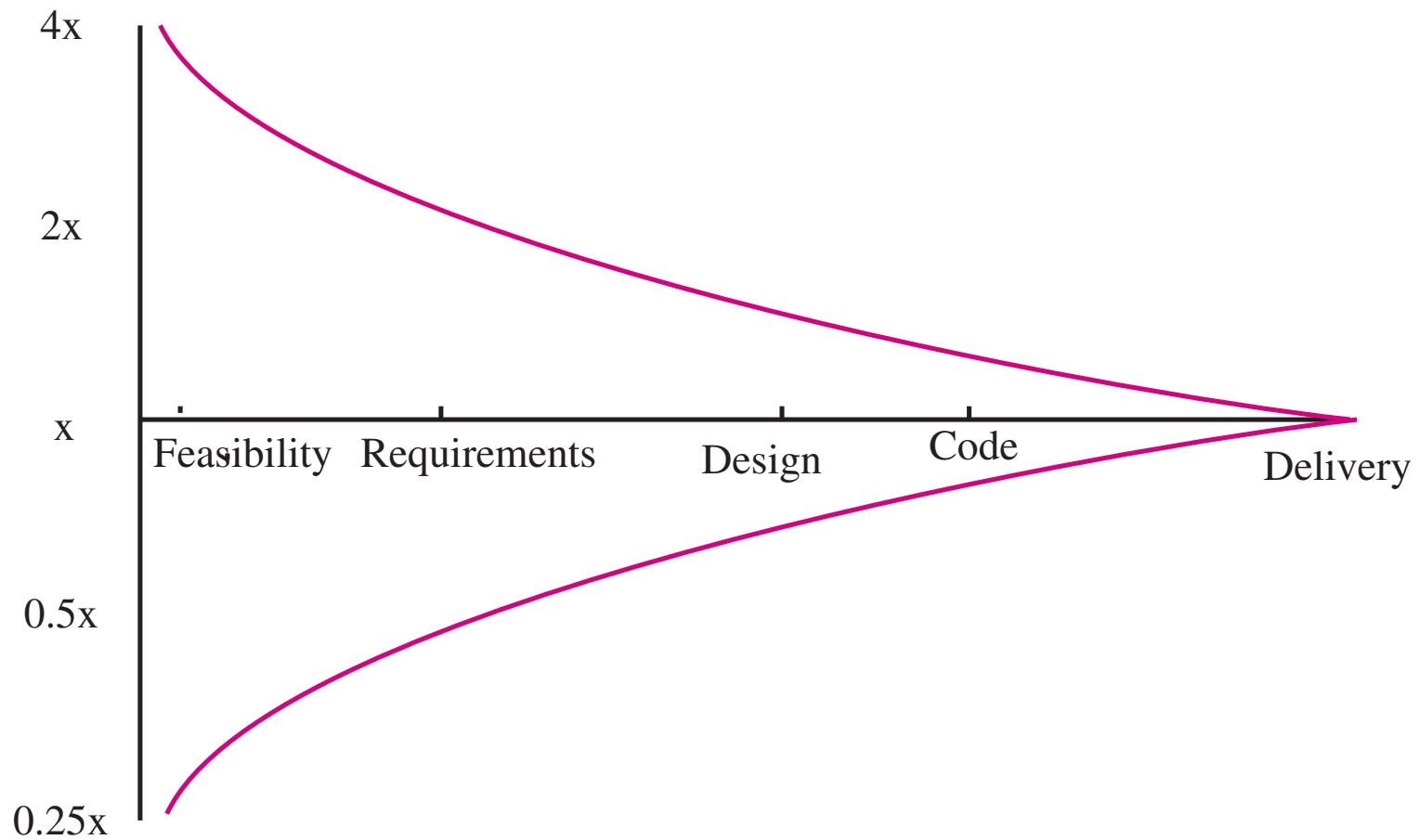
Modelado algorítmico de costes

- Es una aproximación que utiliza fórmulas obtenidas a partir de información histórica. Suele basarse en el tamaño del software
- La mayoría de modelos tienen una componente exponencial (los costes no crecen normalmente de forma lineal con el tamaño del proyecto)
 - $\text{Esfuerzo} = A \times \text{Tamaño}^B \times M$

Exactitud de la estimación

- El tamaño de un sistema software puede conocerse con exactitud solamente cuando está terminado
- Algunos factores que influyen en el tamaño final son:
 - Uso de COTs y componentes
 - Lenguaje de programación utilizado
 - Distribución del sistema
- La estimación del tamaño se realiza de forma más exacta a medida que el desarrollo del sistema progresá

Estimar la incertidumbre



Juicio experto

-  Uno o más expertos, tanto en desarrollo de software como en el dominio de la aplicación usan su experiencia para predecir los costes de software. Se realizan iteraciones hasta que se alcanza un consenso
-  **Ventajas:** Método de estimación relativamente barato. Puede ser bastante exacto si los expertos tienen experiencia directa en sistemas similares
-  **Desventajas:** ¡Muy impreciso si no se dispone de los expertos adecuados!

Estimación por analogía

- El coste de un proyecto se calcula por comparación con proyectos similares en el mismo dominio de aplicación
- **Ventajas:** Bastante preciso si se disponen de datos de proyectos previos
- **Inconvenientes:** Imposible de realizar sin no se han abordado proyectos comparables. Necesita un mantenimiento sistemático de una base de datos

Ley de Parkinson

-  Los costes del proyecto están en función de los recursos disponibles, utilizando todo el tiempo permitido.
-  **Ventajas:** No realiza presupuestos "abultados"
-  **Inconvenientes:** El sistema normalmente no termina

Pricing to win

-  El coste del proyecto está en función de lo que el cliente está dispuesto a pagar
-  **Ventajas:** La empresa desarrolladora consigue el contrato
-  **Inconvenientes:** La probabilidad de que el cliente obtenga el sistema que quiere es pequeña. Los costes no reflejan realmente el trabajo requerido

Estimación ascend. y descend.

- Cualquiera de estas aproximaciones puede utilizarse de forma ascendente o descendente
 - Descendente
 - Comienza a nivel de sistema y evalúa la totalidad de funcionalidades y cómo éstas se subdividen en subsistemas
 - Ascendente
 - Comienza a nivel de componentes y estima el esfuerzo requerido para cada componente. Dichos esfuerzos se añaden a la estimación final

Estimación descendente

- Se puede usar sin conocer la arquitectura ni los componentes que formarán parte del sistema
- Tiene en cuenta costes tales como integración, gestión de configuraciones y documentación
- Puede infra-estimar costes relacionados con la resolución de problemas técnicos de bajo nivel difíciles de resolver

Estimación ascendente

- Se puede usar cuando la arquitectura del sistema es conocida y los componentes han sido identificados
- Proporciona estimaciones bastante exactas si el sistema se ha diseñado con detalle
- Puede infra-estimar costes a nivel de sistema, tales como integración y documentación

Comparando métodos

- Cada método tiene sus ventajas e inconvenientes
- La estimación debería basarse en varios métodos
- Si el resultado de aplicar varios de ellos difiere mucho, es que no se dispone de suficiente información
- Muchas veces el método *Pricing to win* es el único aplicable

Modelo COCOMO

Constructive **C**Ost **M**odel

-  Es un modelo empírico basado en la experiencia con proyectos (grandes)
-  Es un método bien documentado, cuya primera versión se publicó en 1981
-  La última versión, COCOMO 2, tiene en cuenta diferentes aproximaciones de desarrollo, reutilización, etc.

COCOMO 81

Project complexity	Formula	Description
Simple	$PM = 2.4 (\text{KDSI})^{1.05} \times M$	Well-understood applications developed by small teams.
Moderate	$PM = 3.0 (\text{KDSI})^{1.12} \times M$	More complex projects where team members may have limited experience of related systems.
Embedded	$PM = 3.6 (\text{KDSI})^{1.20} \times M$	Complex projects where the software is part of a strongly coupled complex of hardware, software, regulations and operational procedures.

Niveles COCOMO 2

- COCOMO 2 es un modelo de tres niveles que permite estimaciones cada vez más detalladas y que pueden realizarse a la vez que progresá el desarrollo del proyecto
 - Nivel inicial de prototipado
 - Estimaciones realizadas con puntos de objeto y una fórmula simple para el cálculo del esfuerzo
 - Nivel inicial de diseño
 - Estimaciones realizadas con puntos de función convertidas en líneas de código
 - Nivel post-arquitectura
 - Estimaciones basadas en líneas de código fuente

Nivel inicial prototipado

- Soporta proyectos con prototipado y proyectos que hacen uso intensivo de la reutilización
- Basado en estimaciones estándar de la productividad del desarrollador en puntos-objeto/mes
- Tiene en cuenta el uso de herramientas CASE
- La fórmula es:
 - $OPM = (NOP \times (1 - \%reuse/100)) / PROD$
 - OPM es el esfuerzo en personas-mes, NOP es el número de puntos de objeto, y PROD es la productividad

Productiv. de puntos de objeto



La productividad (PROD) depende de:

- La experiencia y capacidad del desarrollador
- Las capacidades de la herramienta CASE utilizada

Developer's experience and capability	Very low	Low	Nominal	High	Very high
ICASE maturity and capability	Very low	Low	Nominal	High	Very high
PROD (NOP/month)	4	7	13	25	50

Nivel inicial de diseño

- Las estimaciones pueden hacerse después de que los requerimientos hayan sido establecidos
- Basado en las fórmulas estándar para métodos algorítmicos
 - $OPM = A \times \text{Tamaño}^B \times M + PM_m$ en donde
 - $O\text{M} = PERS \times RCPX \times RUSE \times PDIF \times PREX \times FCIL \times SCED$
 - $O\text{PM}_m = (\text{ASLOC} \times (\text{AT}/100)) / \text{ATPROD}$
 - $O\text{A} = 2.5$ según la calibración inicial, **Tamaño** se da en KLOC, **B** varía desde 1.1 hasta 1.24 dependiendo de la novedad del proyecto, la flexibilidad del desarrollo, la gestión de riesgos, y la madurez del proceso

Multiplicadores (**M**)

-  Los multiplicadores reflejan la capacidad de los desarrolladores, requerim. no funcionales, la familiaridad con la plataforma de desarrollo, etc.
 - RCPX - fiabilidad de producto y complejidad
 - RUSE - reutilización requerida
 - PDIF - dificultad de la plataforma
 - PREX - experiencia del personal
 - PERS - capacidad del personal
 - SCED - agenda requerida
 - FCIL - facilidades de soporte de grupo
-  **PM** refleja la cantidad de código generada automáticamente

Nivel post-arquitectura

Uso de la misma fórmula que la estimación inicial de diseño

$$\textcircled{O} \quad PM = A \times \text{Tamaño}^B \times M + PM_m$$

Se ajusta la estimación de **tamaño** para que tenga en cuenta

O La volatilidad de los requerimientos

O Grado de posible reutilización

$$\textcircled{O} \quad \text{ESLOC} = \text{ASLOC} \times (\text{AA} + \text{SU} + 0.4\text{DM} + 0.3\text{CM} + 0.3\text{IM})/100$$

ESLOC es el número de líneas de código nuevo. **ASLOC** es el número de líneas de código reusable que debe modificarse, **DM** es el porcentaje de diseño modificado, **CM** es el porcentaje de código que se modifica, **IM** es el porcentaje del esfuerzo original de integración del software reusado.

SU es un factor basado en la interpretación del coste del software, **AA** es un factor que refleja los costes de evaluación iniciales para decidir si el software puede reutilizarse.

El término exponente (**B**)

 Depende de 5 factores de escala. La suma de dichos factores se divide por 100 y se añade a 1.01

 Ejemplo

-  Antecedentes - proyecto nuevo - 4
-  Flexibilidad desarrollo - no implicación cliente - Muy alto - 1
-  Arquitectura/resolución riesgos - No análisis de riesgos - Muy bajo - 5
-  Cohesión del grupo - nuevo grupo - nominal - 3
-  Madurez proceso - algún control - nominal - 3

 El factor de escala es 1.17

Factores de escala de exponente

Scale factor	Explanation	(B)
Precedentedness	Reflects the previous experience of the organisation with this type of project. Very low means no previous experience, Extra high means that the organisation is completely familiar with this application domain.	
Development flexibility	Reflects the degree of flexibility in the development process. Very low means a prescribed process is used; Extra high means that the client only sets general goals.	
Architecture/risk resolution	Reflects the extent of risk analysis carried out. Very low means little analysis, Extra high means a complete a thorough risk analysis.	
Team cohesion	Reflects how well the development team know each other and work together. Very low means very difficult interactions, Extra high means an integrated and effective team with no communication problems.	
Process maturity	Reflects the process maturity of the organisation. The computation of this value depends on the CMM Maturity Questionnaire but an estimate can be achieved by subtracting the CMM process maturity level from 5.	

Multiplicadores (**M**)

 Atributos del producto

 Atributos del ordenador

 Atributos del personal

 Atributos del proyecto

Conductores de coste del proy.

Product attributes			
RELY	Required system reliability	DATA	Size of database used
CPLX	Complexity of system modules	RUSE	Required percentage of reusable components
DOCU	Extent of documentation required		
Computer attributes			
TIME	Execution time constraints	STOR	Memory constraints
PVOL	Volatility of development platform		
Personnel attributes			
ACAP	Capability of project analysts	PCAP	Programmer capability
PCON	Personnel continuity	AEXP	Analyst experience in project domain
PEXP	Programmer experience in project domain	LTEX	Language and tool experience
Project attributes			
TOOL	Use of software tools	SITE	Extent of multi-site working and quality of site communications
SCED	Development schedule compression		

Efectos de los conduct. de coste

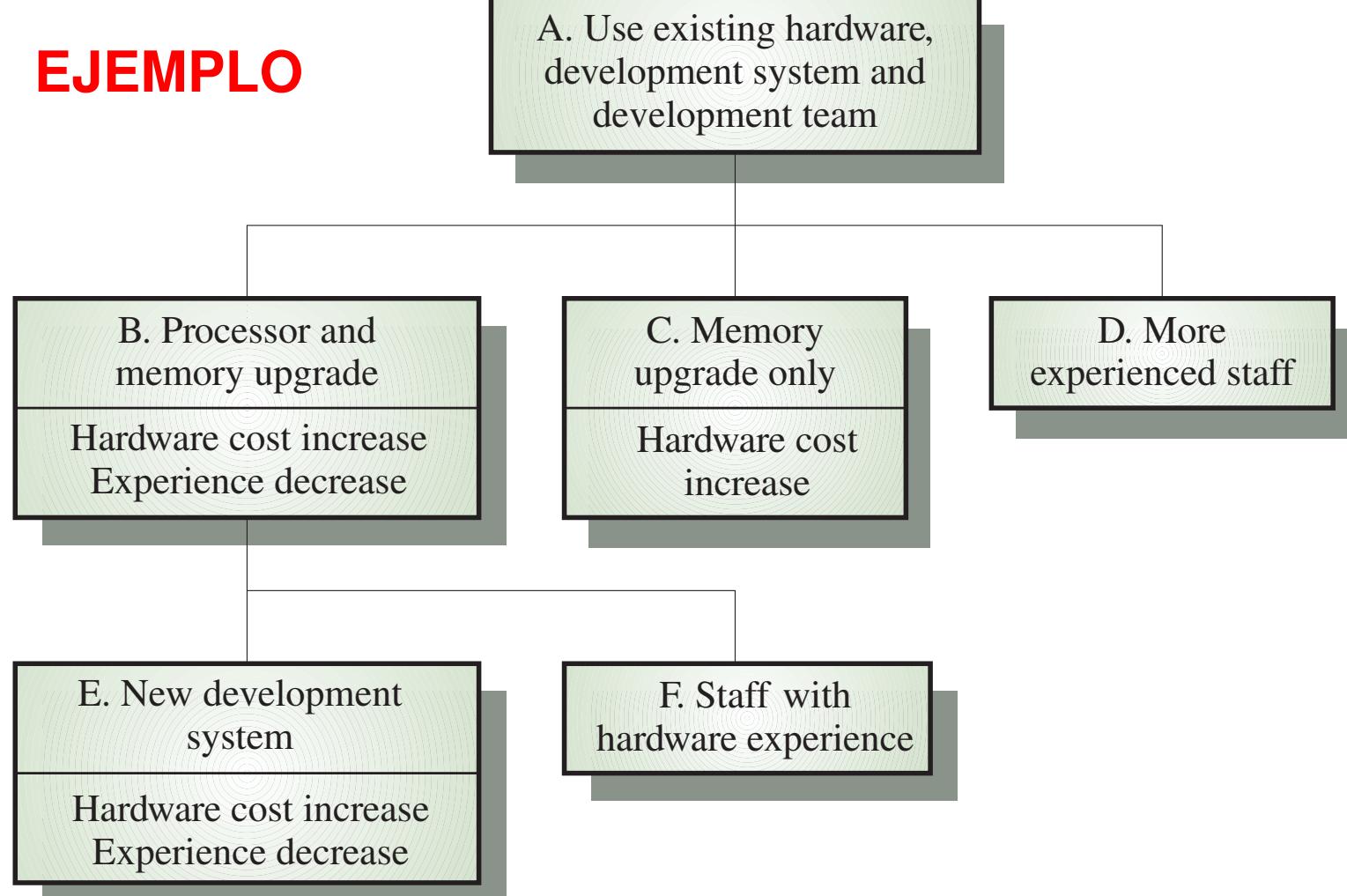
Exponent value	1.17	(M)
System size (including factors for reuse and requirements volatility)	128, 000 DSI	
Initial COCOMO estimate without cost drivers	730 person-months	
Reliability	Very high, multiplier = 1.39	
Complexity	Very high, multiplier = 1.3	
Memory constraint	High, multiplier = 1.21	
Tool use	Low, multiplier = 1.12	
Schedule	Accelerated, multiplier = 1.29	
Adjusted COCOMO estimate	2306 person-months	
Reliability	Very low, multiplier = 0.75	
Complexity	Very low, multiplier = 0.75	
Memory constraint	None, multiplier = 1	
Tool use	Very high, multiplier = 0.72	
Schedule	Normal, multiplier = 1	
Adjusted COCOMO estimate	295 person-months	

Planificación del proyecto

- Los modelos algorítmicos de costes proporcionan una base para la planificación del proyecto en tanto que permiten comparar estrategias alternativas
- Ejemplo: desarrollo de un sistema espacial empotrado
 - Debe ser fiable
 - Debe tener un peso mínimo (número de chips)
 - Multiplicadores de fiabilidad y restricciones del ordenador > 1
- Componentes de coste
 - Hardware destino
 - Plataforma de desarrollo
 - Esfuerzo requerido

Opciones de gestión

EJEMPLO



Gestión de opciones de coste

EJEMPLO

Option	RELY	STOR	TIME	TOOLS	LTEX	Total effort	Software cost	Hardware cost	Total cost
A	1.39	1.06	1.11	0.86	1	63	949393	100000	1049393
B	1.39	1	1	1.12	1.22	88	1313550	120000	1402025
C	1.39	1	1.11	0.86	1	60	895653	105000	1000653
D	1.39	1.06	1.11	0.86	0.84	51	769008	100000	897490
E	1.39	1	1	0.72	1.22	56	844425	220000	1044159
F	1.39	1	1	1.12	0.84	57	851180	120000	1002706

Selección de opciones

 Opción D (usa más personal con experiencia) parece la mejor alternativa

○ Sin embargo tiene un alto riesgo asociado ya que personal con experiencia puede ser difícil de encontrar

 Opción C (actualización memoria) tiene un menor ahorro de costes, pero un riesgo muy bajo

En conjunto, el modelo revela la importancia del personal con experiencia en el desarrollo del software

Duración y personal del proyecto

- Además de la estimación del esfuerzo, se debe estimar el tiempo requerido para terminar el proyecto, así como el personal necesario
- La duración del proyecto puede estimarse mediante la fórmula de COCOMO 2
 - $TDEV = 3 \times (PM)^{(0.33+0.2*(B-1.01))}$
 - PM es el esfuerzo.

La duración es independiente del número de gente que trabaje en el proyecto (depende del esfuerzo total invertido en el proyecto)

Puntos clave

-  Es necesario estimar costes: (esfuerzo, tiempo de desarrollo y número de recursos)
-  La productividad es un factor a tener en cuenta a la hora de realizar estimaciones
-  Existen varias técnicas de estimación de costes. La estimación algorítmica de costes es difícil al necesitar una estimación previa de atributos del producto terminado
-  Los modelos de estimación algorítmicos suponen una opción de análisis cuantitativo
-  El tiempo necesario para completar un proyecto no es proporcional al número de personas que trabajan en el mismo

Tema 3. Planificación de proyectos

- Planificación de proyectos
- Confección de agendas (*scheduling*)
- Gestión de riesgos

Bibliografía

- Captítulo 5. Project Management. Software Engineering Sommerville 7^a edición.
- Capítulo 5. Planificación de Proyectos software. Ingeniería del software. 4^a edición. Roger S. Pressman.
- Capítulo 6. Gestión del riesgo. Ingeniería del software. 4^a edición. Roger S. Pressman.
- Capítulo 7. Planificación temporal y seguimiento del proyecto. Ingeniería del software. 4^a edición. Roger S. Pressman.

Importancia de la gestión

- Se trata probablemente de la actividad de gestión que más tiempo consume
- Es una actividad que se realiza de forma **CONTINUADA** desde el concepto inicial del sistema, hasta su entrega. Los planes se deben revisar continuamente según esté disponible nueva información

Tipos de planes de proyectos

- Plan de calidad
- Plan de validación
- Plan de gestión de configuraciones
- Plan de mantenimiento
- Plan de gestión de personal

Proceso de planificación

Planificar es decidir de antemano:

- **QUÉ** hay que hacer
- **CÓMO** hay que hacerlo
- **CUÁNDO** se va a hacer
- **QUIÉN** lo va a hacer

ALGORITMO:

Establecer **restricciones** proy.

Hacer **evaluaciones** iniciales

Definir **hitos** y **entregas**

Mientras proy no acaba hacer

Confeccionar **agenda**

Iniciar trabajos agenda

Esperar

Revisar el progreso

Revisar estimaciones

Actualizar agenda

Re-negociar restricciones

Solucionar posibles problemas

Fin mientras

Estructura del plan del proyecto

- Introducción
- Organización del proyecto
- Análisis de riesgos
- Requerimientos hardware y software
- Estructura de actividades del proyecto
- Agenda del proyecto
- Mecanismos de monitorización y control

Organización de las actividades

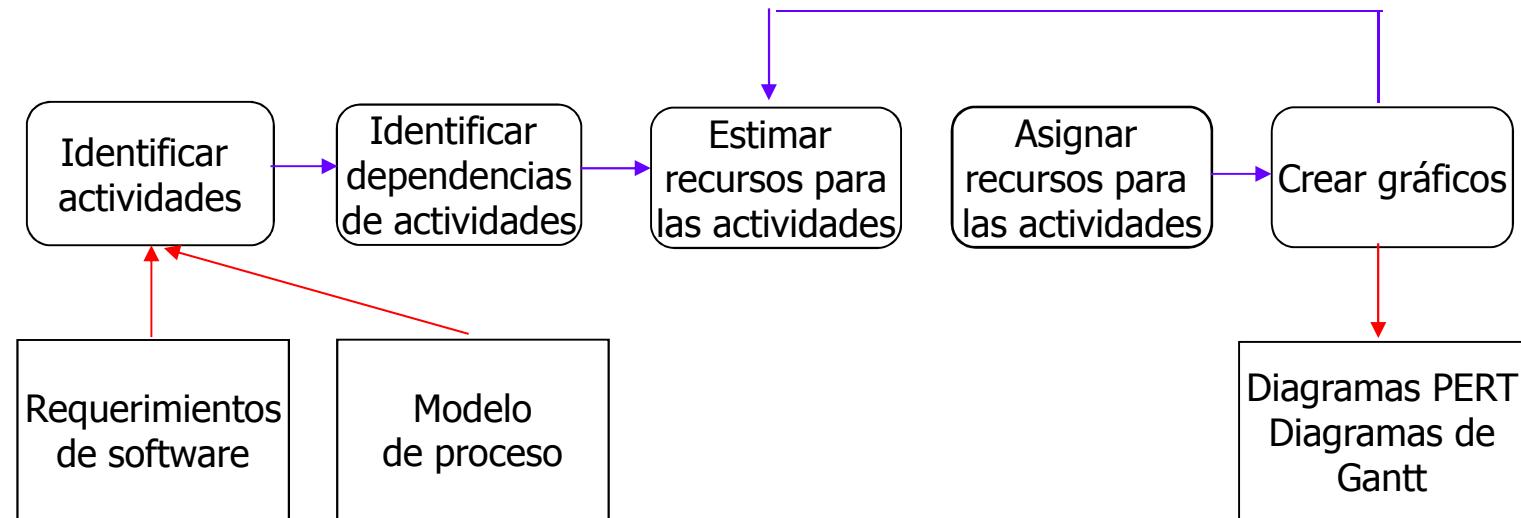
- Las actividades se deben organizar de forma que produzcan salidas "tangibles" válidas para la gestión del progreso del desarrollo
- Los HITOS (*milestones*) marcan el final de una actividad del proceso de desarrollo
- Las ENTREGAS (*deliverables*) son resultados del proyecto que se entregan a los clientes
- El proceso en cascada permite identificar de forma sencilla los **hitos** que marcan el progreso del proyecto

Scheduling

- Consiste en la organización temporal y asignación de recursos a las actividades de un proyecto.

- Pasos a seguir:
 - Determinación de las actividades a realizar
 - Asignación de tiempos estimados
 - Asignación de recursos
 - Organización temporal de las actividades

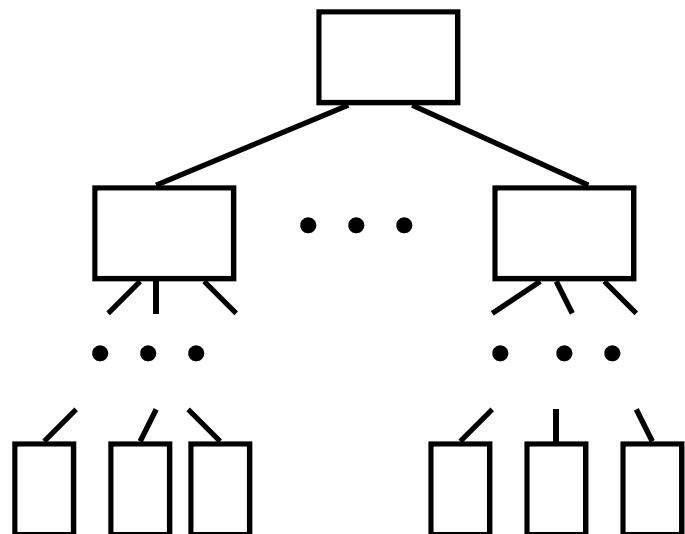
El proceso de *Scheduling*



Estructuración de las actividades

Agrupación de actividades por niveles.

Sirven para situar las actividades dentro de la organización del proyecto



00000 Nivel 0
10000 Subnivel 1
11000 Subnivel 1.1
11100 Actividad 1.1.1
11200 Actividad 1.1.2

20000 Subnivel 2

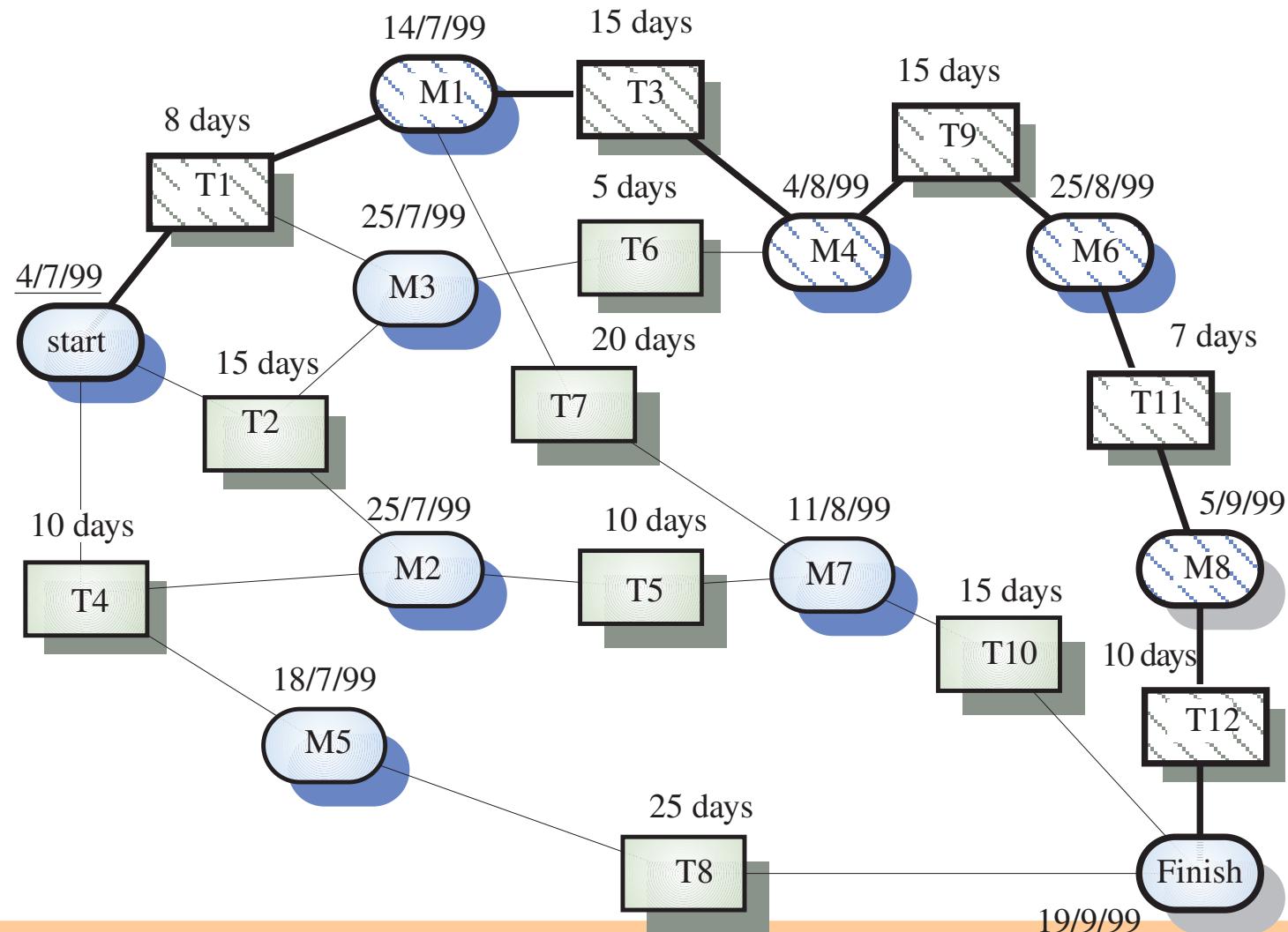
Representaciones gráficas

- Se utilizan para ilustrar la agenda del proyecto
- Permiten mostrar una vista de la división en tareas del proyecto. Las actividades no deberían ser demasiado "pequeñas" (alrededor de una semana o más)
- Los diagramas de actividades muestran las dependencias de las tareas y el camino crítico
- Los diagramas de barras muestran la agenda del proyecto

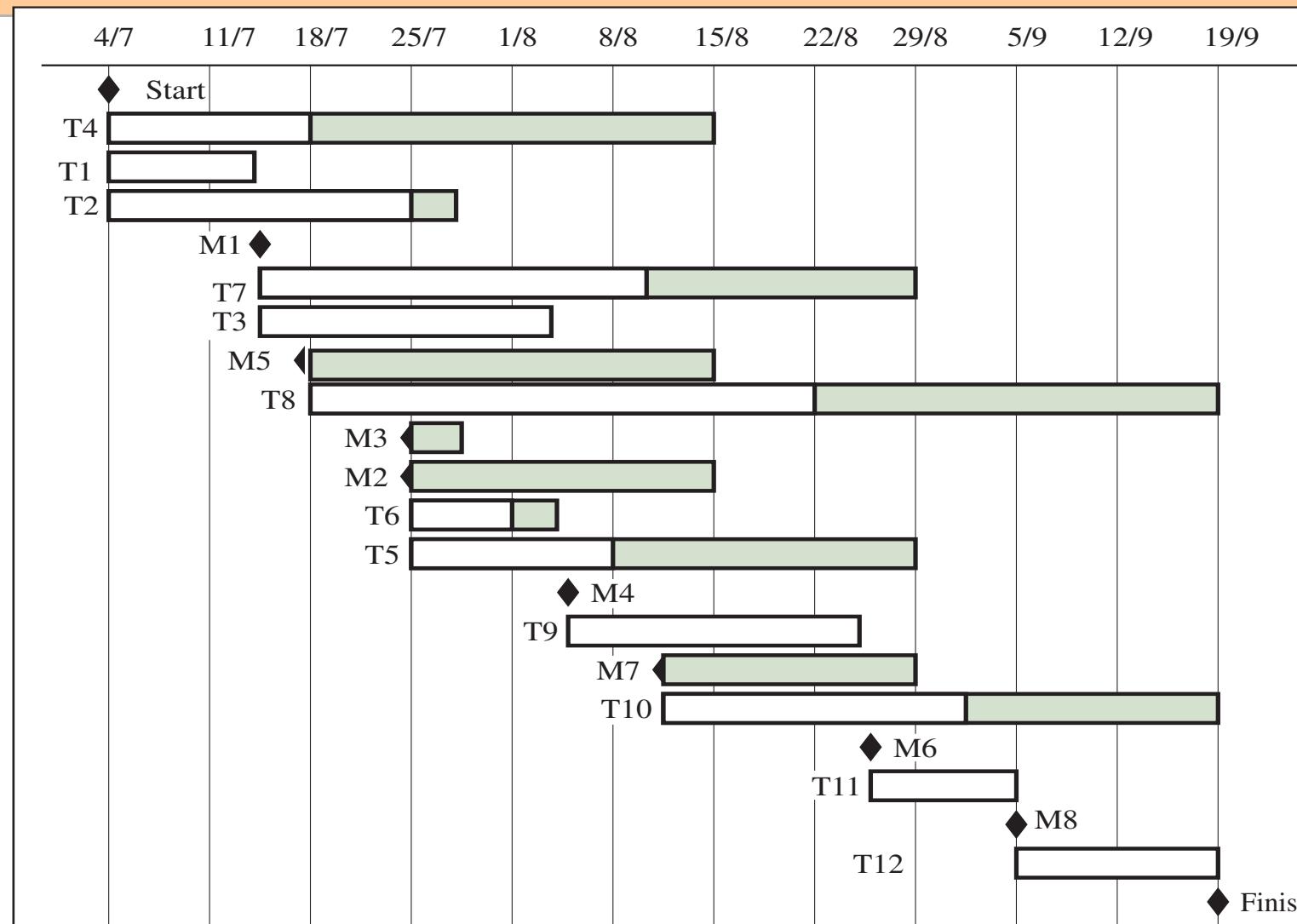
Duración de tareas y dependencias

Task	Duration (days)	Dependencies
T1	8	
T2	15	
T3	15	T1 (M1)
T4	10	
T5	10	T2, T4 (M2)
T6	5	T1, T2 (M3)
T7	20	T1 (M1)
T8	25	T4 (M5)
T9	15	T3, T6 (M4)
T10	15	T5, T7 (M7)
T11	7	T9 (M6)
T12	10	T11 (M8)

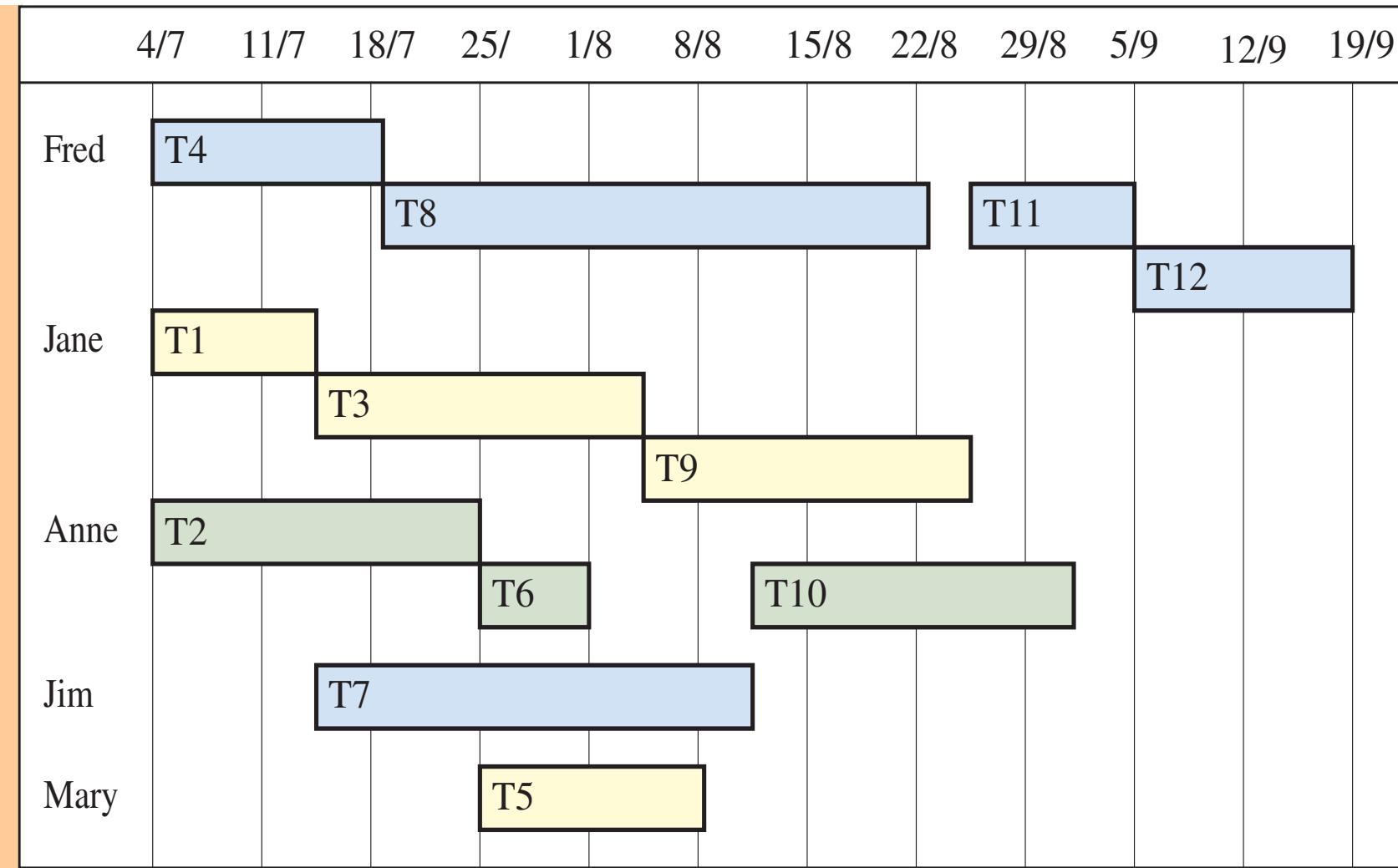
Red de actividades



Secuencia temporal actividades



Asignación de personal



Grafos PERT

Datos de entrada:

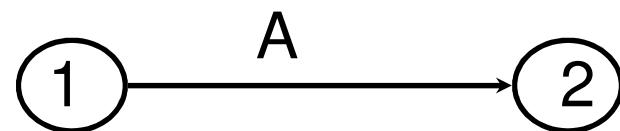
- Lista de precedencias del proyecto
- Asignación de tiempos y recursos a actividades

Proceso a realizar:

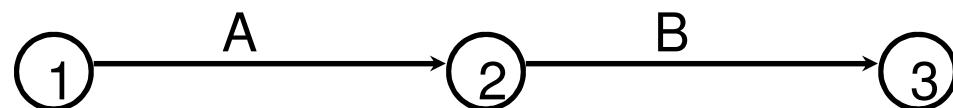
- Representar grafo
- Cálculo de tiempos “early” y “last”
- Cálculo de holguras
- Cálculo del camino crítico
- Confección de agenda

Precedencia entre actividades(I)

$G = (V, E)$, V = cto. de vértices \rightarrow sucesos
 E = cto. de aristas \rightarrow actividades

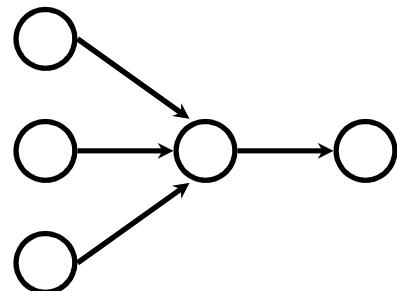


PRELACIÓN LINEAL

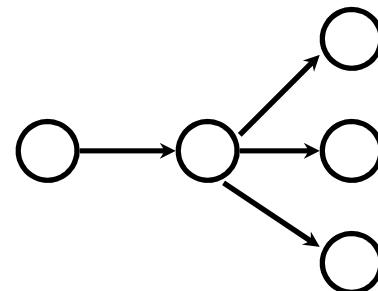


Precedencia entre actividades(II)

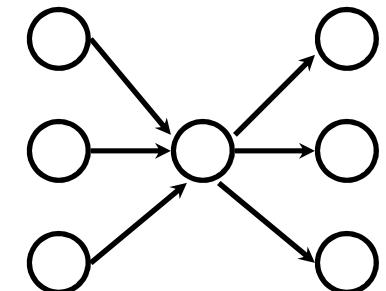
CONVERGENCIA



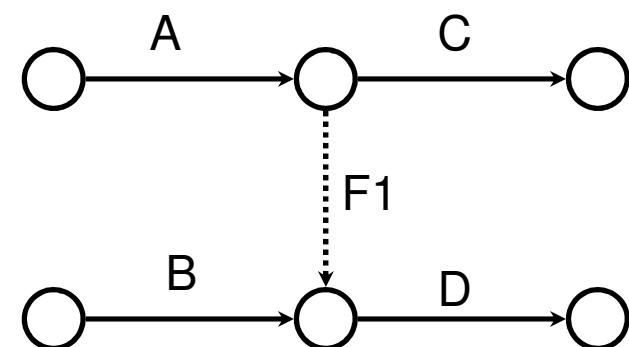
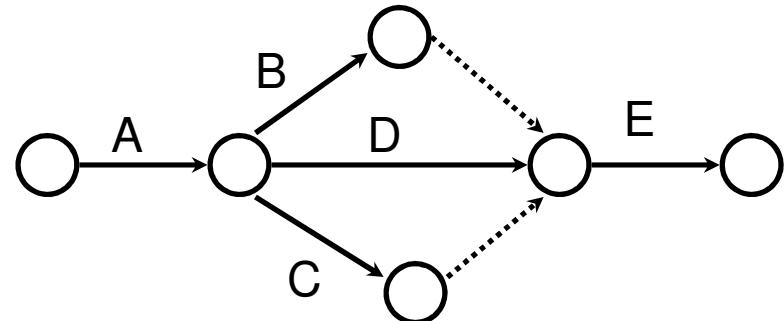
DIVERGENCIA



CONV-DIVER.



ACTIVIDADES FICTICIAS



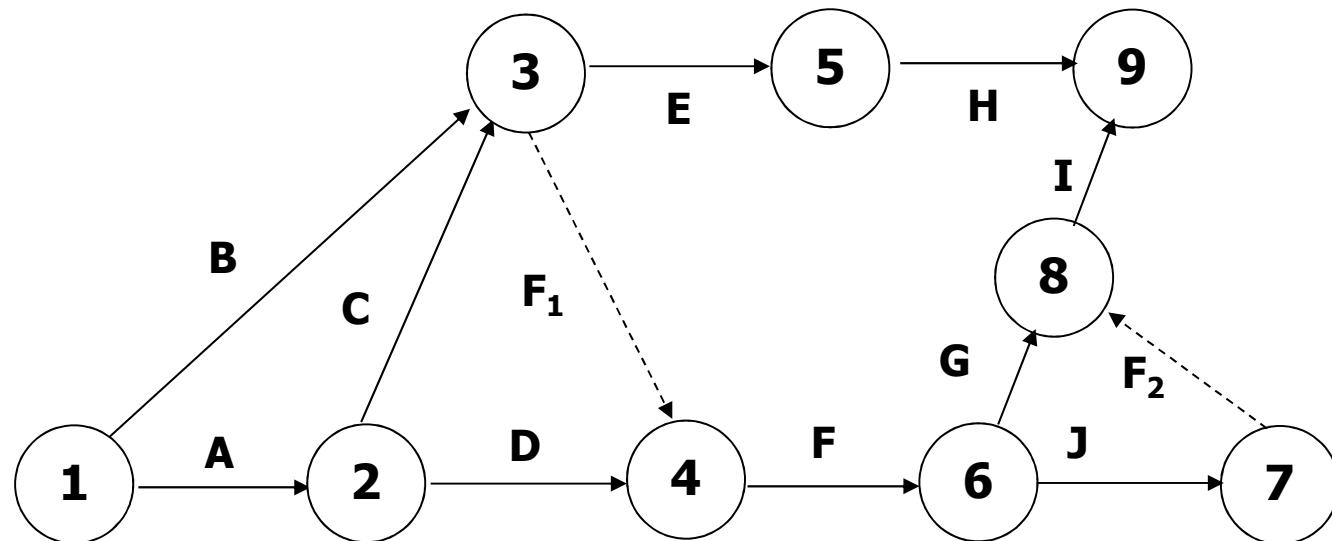
Precedencia entre actividades(III).

Ejercicio: Dibuja el grafo PERT.

Actividades	Precedentes	Duraciones
A	---	2
B	---	3
C	A	7
D	A	8
E	B, C	3
F	B, C, D	9
G	F	8
H	E	2
I	G, J	2
J	F	10

Precedencia entre actividades(VI).

Ejercicio: Dibuja el grafo PERT (SOLUCION)



Tiempos "early" y "last" (I)



Cálculo de tiempos más tempranos de ejecución

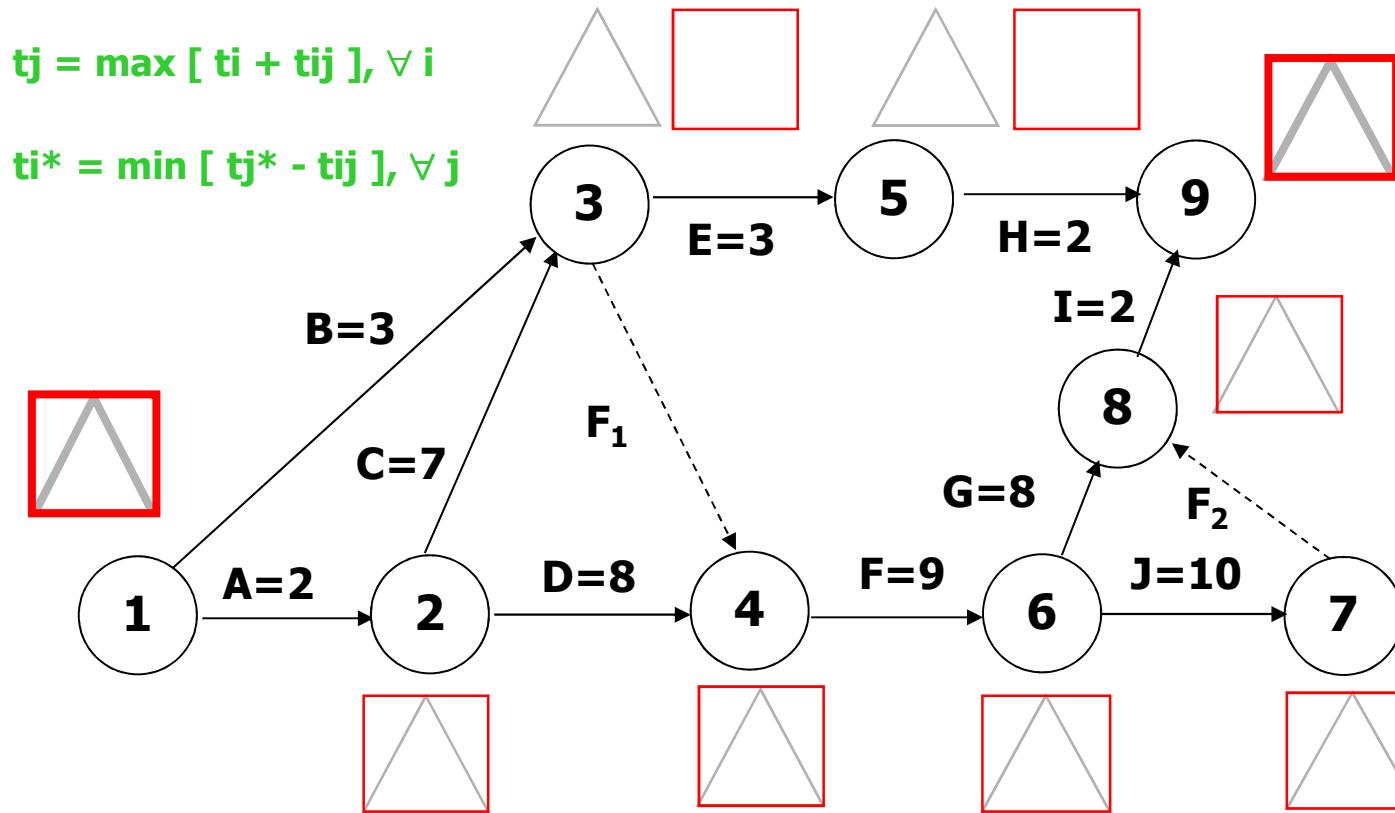
$$t_j = \max [t_i + t_{ij}] \text{ para todo } i \quad \triangle$$

Cálculo de tiempos más tardíos de ejecución

$$t_i^* = \min [t_j^* - t_{ij}] \text{ para todo } j \quad \square$$

Tiempos "early" y "last" (II).

Ejercicio: Calcula los tiempos last y early.

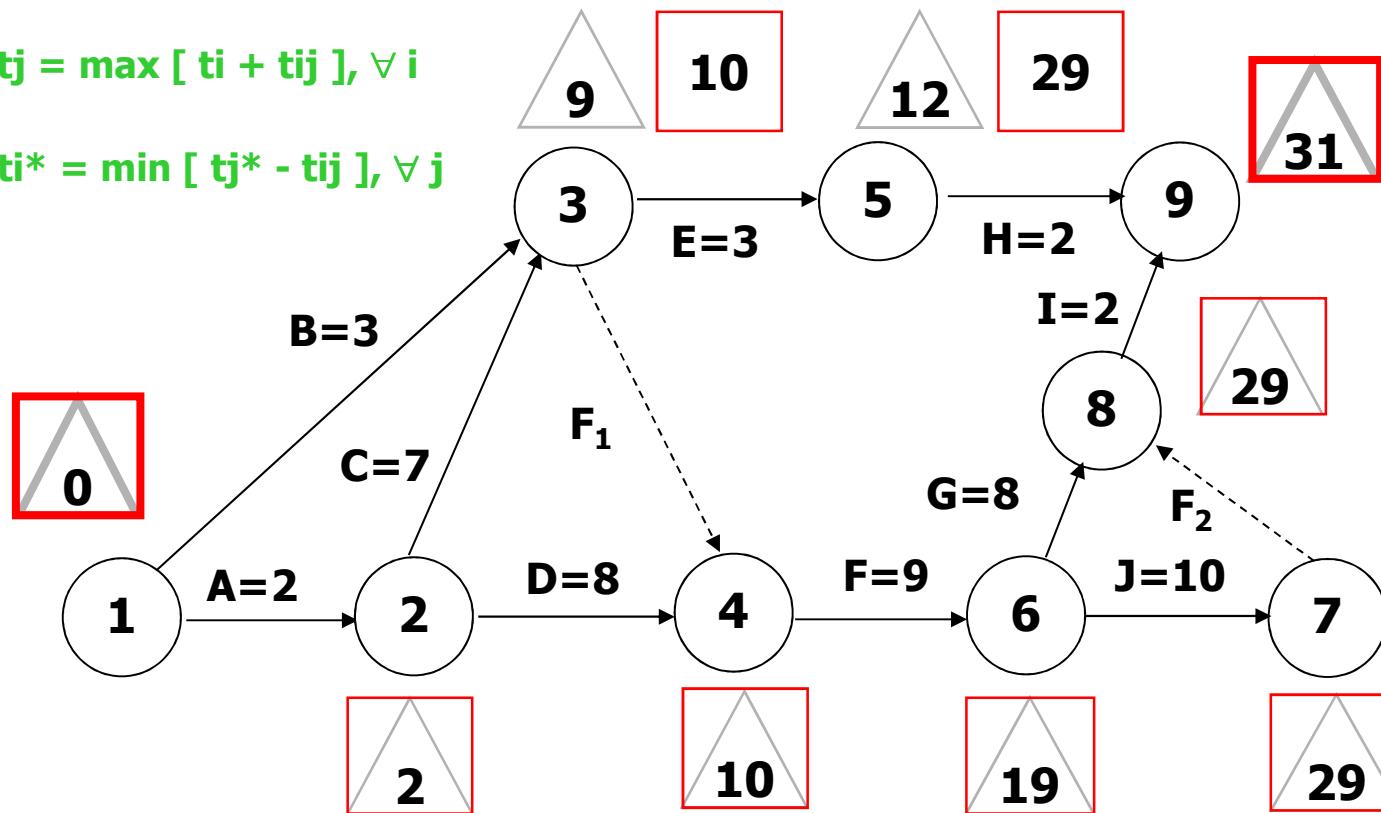


Tiempos "early" y "last" (II).

Ejercicio: Calcula los tiempos last y early (SOLUCION).

$$t_j = \max [t_i + t_{ij}], \forall i$$

$$t_{i^*} = \min [t_{j^*} - t_{ij}], \forall j$$



Holguras y camino crítico (I)

$$H_i = t_i^* - t_i$$

$$H_{ij}^T = t_j^* - t_i - t_{ij} \quad \text{Holgura total}$$

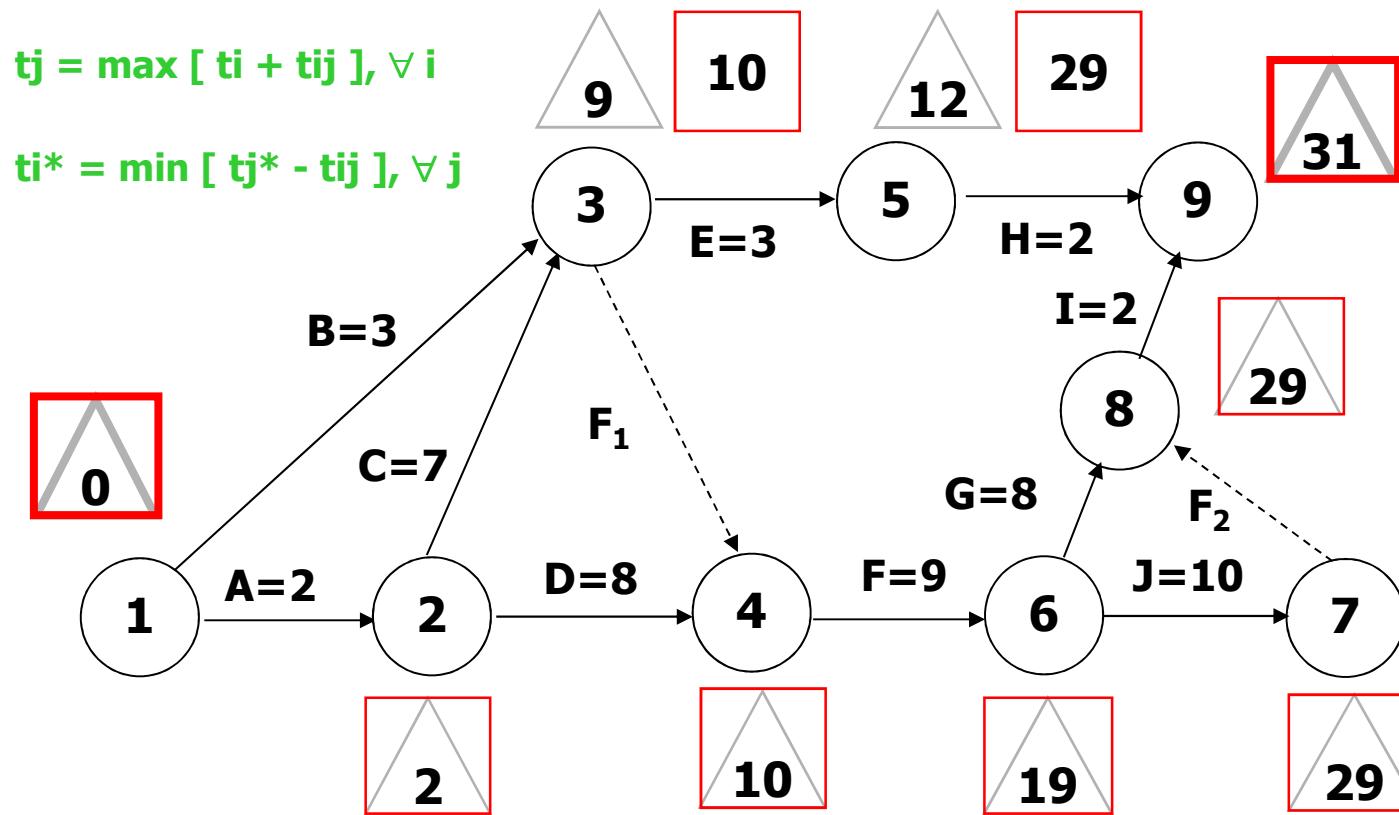
$$H_{ij}^L = t_j - t_i - t_{ij} \quad \text{Holgura libre}$$

Cálculo del camino crítico

- Camino más largo en el grafo
- Viene determinado por el **tiempo early** del suceso fin del proyecto
- Puede haber más de un camino crítico
- **TODAS** las actividades del camino crítico tienen holgura total = 0 (condición necesaria)

Holguras y camino crítico (II)

Ejercicio: Calcula las holguras y establece el camino crítico.

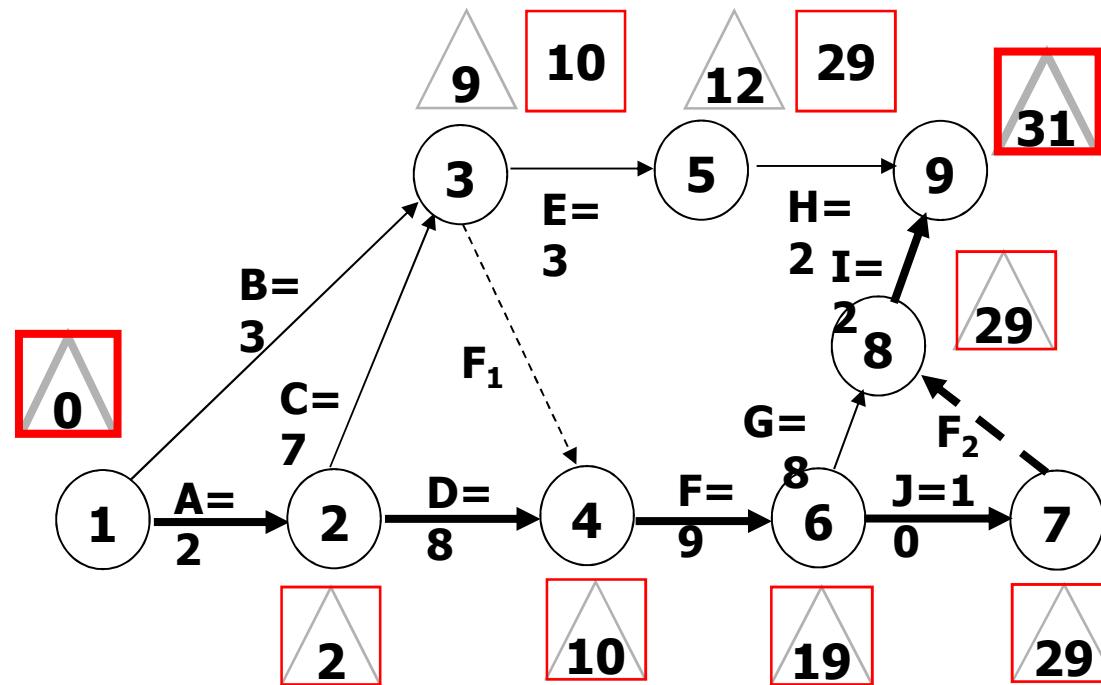


Holguras y camino crítico (III).

Ejercicio: Calcula las holguras y establece el camino crítico (SOLUCION).

$$H^T_{ij} = t_j^* - t_i - t_{ij}$$

ACTIVIDAD (i-j)	A (1-2)	B (1-3)	C (2-3)	D (2-4)	E (3-5)	F (4-6)	G (6-8)	H (5-9)	I (8-9)	J (6-7)
H^T_{ij}	0	7	1	0	17	0	2	17	0	0



Agenda (I)



Fecha de comienzo

$$\triangle_{ij} = t_i$$

Más temprana

$$\triangle_{ij}^* = t_j^* - t_{ij}$$

Más tardía



Fecha de fin

$$\nabla_{ij} = t_i + t_{ij}$$

Más temprana

$$\nabla_{ij}^* = t_j^*$$

Más tardía

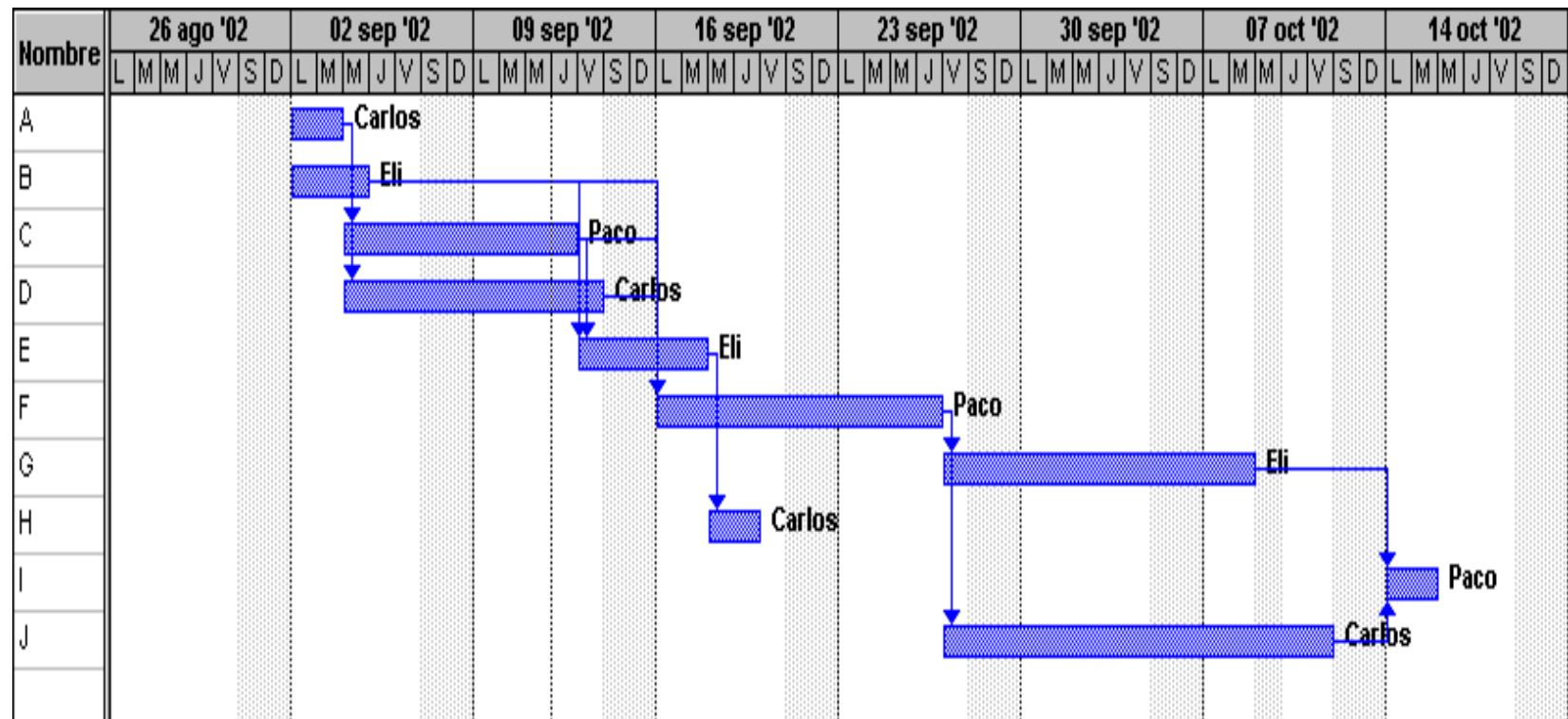
Agenda (II)

Actividades	Fecha inicio	Fecha fin
A		
B		
C		
D		
E		
G		
H		
I		
J		
K		

Diagrama Gant

- Eje de ordenadas
 - Representa Actividades o Recursos
- Eje de abscisas
 - Tiempo
- Permite observar con detalle la evolución del proyecto.
- La herramienta Microsoft Project permite generar automáticamente
 - Diagramas de Gantt
 - Grafos PERT

Diagrama Gant (II)



Ejercicio 1

Tareas	Preced.	Duración.
A	-	2
B	-	3
C	-	2
D	A	3
E	A, B	1
F	A	4
G	B, C	2
H	C	5
N	D, E	3
J	N	2
K	F, J	2
M	F, G	3

Calcular:

1. Tiempo mínimo de duración del proyecto
2. Camino crítico
3. Agenda
4. ¿qué ocurre si G se retrasa en 4 días y M en 3?
5. Si retrasamos G en 1 día, ¿qué le ocurre a M?

Ejercicio 2

Tareas	Preced.	Duración.
A	-	4
B	-	2
C	A , B	1
H	E , C	3
F	B	1
G	B	5
J	F	2
K	F , H	3
L	H , F	6
Q	M , J , G	4
R	Q	5
P	L , K	4
N	K	1
E	B	1
M	H , F	1

Calcular:

1. Tiempo mínimo de duración del proyecto
2. Camino crítico
3. Agenda
4. ¿qué ocurre si J se retrasa en 5 días ?
5. ¿qué ocurre si F se retrasa en 2 días ?

Problemas de *scheduling*

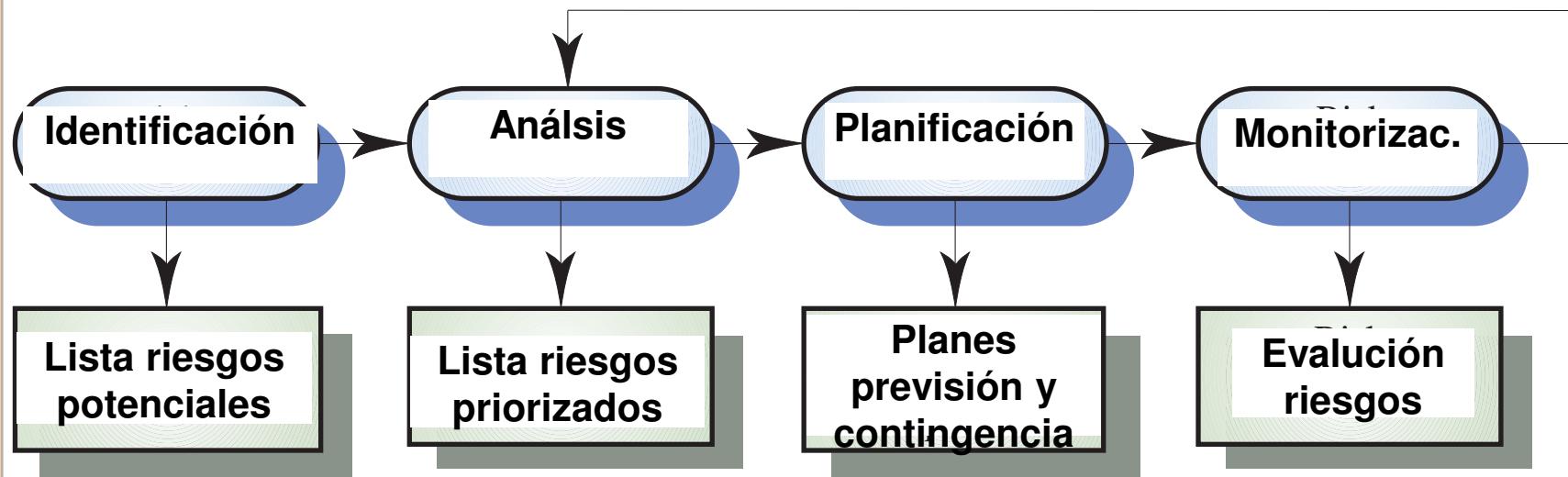
- La tarea de estimar la dificultad del problema y por lo tanto el coste asociado es bastante "dura"
- La productividad no es proporcional al número de gente trabajando en una tarea
- Añadir gente a un proyecto que va con retraso lo retrasará más todavía debido a la sobrecarga en cuanto a las comunicaciones personales
- Lo inesperado SIEMPRE ocurre. Se debe tener esto en cuenta a la hora de planificar

Gestión de riesgos

- Actividad que concierne a la identificación de los riesgos y desarrollo de planes para minimizar sus efectos en un proyecto.
- Un riesgo es una **probabilidad** de que pueda ocurrir alguna circunstancia adversa.
 - Los riesgos del proyecto afectan a la agenda o los recursos
 - Los riesgos del producto afectan a la calidad o realización del desarrollo
 - Los riesgos del negocio afectan a la organización que desarrolla o que gestiona el proyecto

El proceso de gestión de riesgos

- Identificación de riesgos
- Análisis de riesgos
- Planificación de los riesgos
- Monitorización de los riesgos



Identificación de riesgos

TIPO DE RIESGO	POSIBLE RIESGO
TECNOLOGÍA	La Base de Datos utilizada no puede procesar muchas transacciones por segundo como se esperaba.
PERSONAS	Es imposible seleccionar personal con las habilidades requeridas para el proyecto.
ORGANIZACIONAL	Los problemas financieros en la organización causan reducciones en el presupuesto del proyecto.
HERRAMIENTAS	Las herramientas CASE no se pueden integrar.
REQUERIMIENTOS	Se proponen cambios en los requerimientos que suponen rehacer el diseño.
ESTIMACIÓN	El tiempo requerido para desarrollar el software está infraestimado.

Análisis de riesgos

POSIBLE RIESGO	PROBABILIDAD	EFFECTOS
Problemas financieros en la organización.	BAJA	CATASTRÓFICO
Problemas de selección de personal.	ALTA	CATASTRÓFICO
Cambios en los requerimientos.	MODERADA	SERIO
Rendimiento de la Base de Datos.	MODERADA	SERIO
Tiempo de desarrollo infraestimado.	ALTA	SERIO
Herramientas CASE no se pueden integrar.	ALTA	TOLERABLE

Planificación de riesgos

RIESGO	ESTRATEGIA
Problemas financieros en la organización.	Preparar un documento breve para el gestor principal que muestre que el proyecto hace contribuciones muy importantes a los objetivos del negocio.
Problemas de selección de personal.	Alertar al cliente de las dificultades potenciales y las posibilidades de retraso, previsión de cursos de formación.
Cambios en los requerimientos.	Investigar la información para valorar el impacto del cambio en los requerimientos, prototipado y revisión con el cliente.
Rendimiento de la Base de Datos.	Investigar la posibilidad de comprar una base de datos de rendimiento más alto.
Tiempo de desarrollo infraestimado.	Investigar los componentes comprados y la utilización de un generador de programas.

Monitorización de riesgos

TIPO DE RIESGO	IDENTIFICADORES POTENCIALES
TECNOLOGÍA	Entrega retrasada del hardware o del soporte software.
PERSONAS	Personal con moral baja, malas relaciones entre miembros del equipo.
ORGANIZACIONAL	Cotilleos en la organización, pasividad en el gestor principal.
HERRAMIENTAS	Quejas sobre herramientas CASE, peticiones de estaciones de trabajo más potentes.
REQUERIMIENTOS	Peticiones de cambios en muchos requerimientos, quejas del cliente.
ESTIMACIÓN	Fracaso en el cumplimiento de la agenda acordada.

Puntos clave

- Una buena gestión de proyectos es esencial para el éxito del proyecto
- La naturaleza intangible del software causa problemas en la gestión de proyectos
- Los gestores se encargan de varias tareas, pero las más significativas con planificación, estimación y confección de agendas (*scheduling*)
- La planificación y estimación son procesos iterativos y continuados durante todo el desarrollo del proyecto

Tema 4. Gestión de Recursos humanos

-  Límites del pensamiento
-  Introducción
-  Trabajo en grupo
-  Selección y organización del personal
-  El modelo CMM para personal

Bibliografía

- Captítulo 25. Managing people. Software Engineering Sommerville 7^a edición.
- Capítulo 5. Planificación de Proyectos software. Ingeniería del software. 4^a edición. Roger S. Pressman.
- Capítulo 7. Planificación Temporal y seguimiento de proyectos. Ingeniería del software. 4^a edición. Roger S. Pressman.

Introducción (I)

LA GENTE EN EL PROCESO

- La gente es uno de los "bienes máspreciados" de una organización
- Las tareas de un gestor están esencialmente **orientadas a la gente**. A menos que haya algún entendimiento con la gente, la gestión será un fracaso
- La ingeniería del software es fundamentalmente una **actividad cognitiva**. Las limitaciones del conocimiento limitan a su vez el proceso software

Introducción (II)

ACTIVIDADES DE GESTIÓN

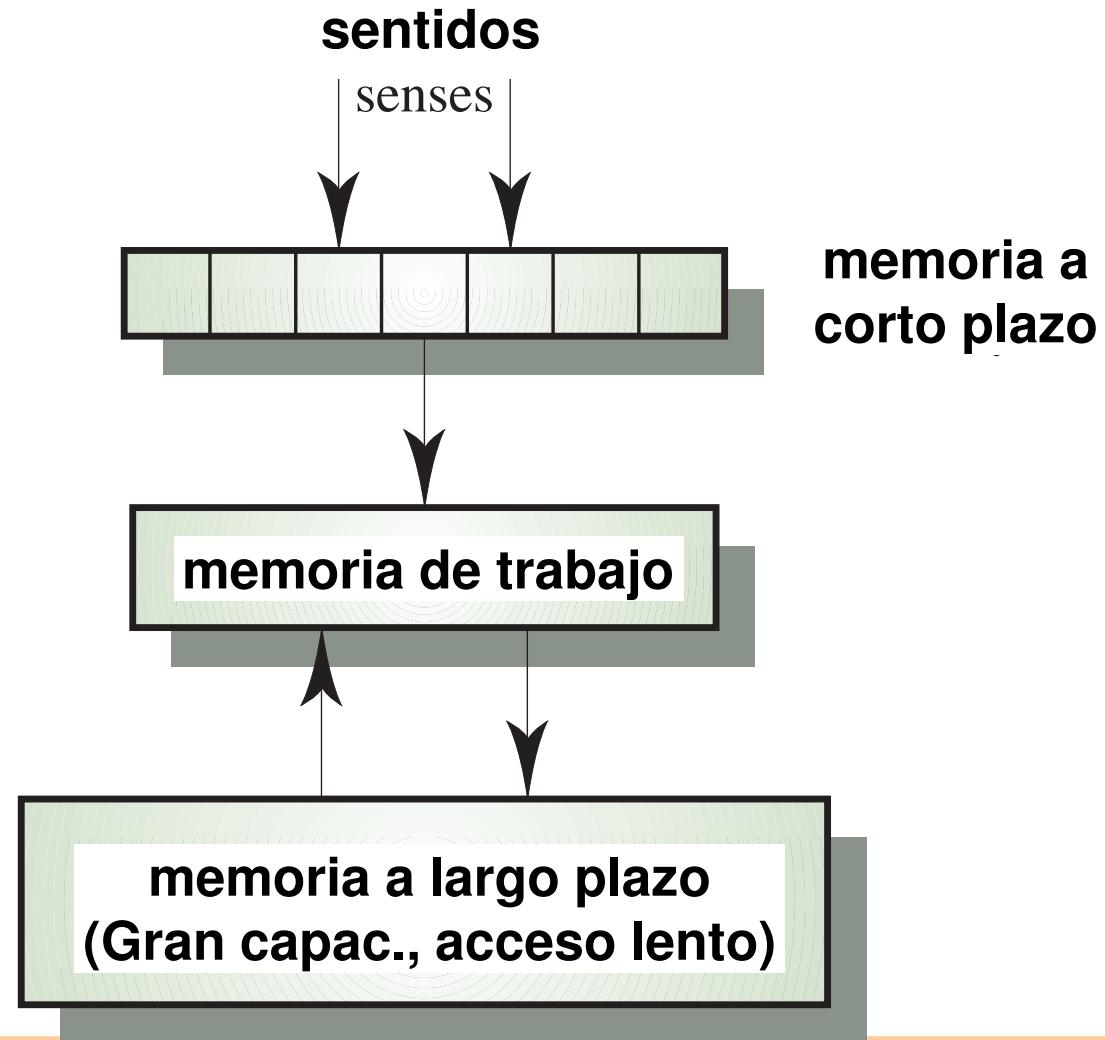
- Solución de problemas (usando la gente disponible)
- Motivación (de la gente que trabaja en el proyecto)
- Planificación (QUÉ tiene que hacer la gente)
- Estimación (CUÁN rápido tiene que trabajar la gente)
- Control (de las actividades de la gente)
- Organización (CÓMO tiene que trabajar la gente)

Límites del pensamiento

- La gente no piensa de la misma manera, pero cada uno está sujeto a restricciones básicas sobre su forma de pensar debido a:
 - La organización de la memoria
 - La representación del conocimiento
 - La influencia de la motivación

- Si comprendemos estas restricciones, podemos comprender cómo éstas afectan a la gente que participa en un proceso software

Organización de la memoria



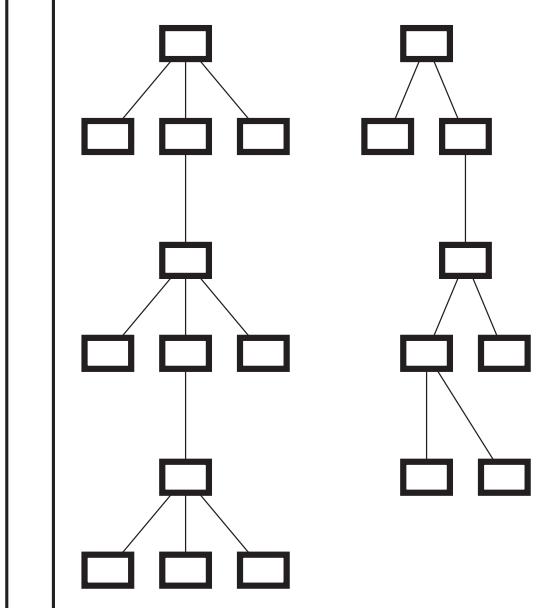
Transferencia de información

- La resolución de problemas requiere normalmente una transferencia de información entre la memoria a corto plazo y la memoria de trabajo
- La información puede perderse o corromperse durante esta transferencia
- El procesamiento de la información ocurre durante la transferencia de información desde la memoria a corto plazo a la memoria a largo plazo

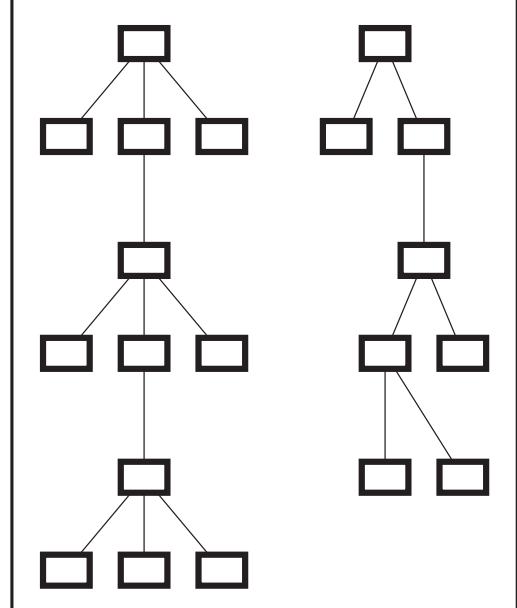
Modelado del conocimiento

- **Conocimiento semántico:** es el conocimiento de los conceptos tales como la operación de asignación, el paso de parámetros, etc.
- **Conocimiento sintáctico:** se refiere a los detalles de representación, por ejemplo un bucle en C.
- El conocimiento semántico se almacena de forma estructurada, independientemente de la representación.

Conocim. sintáctico y semántico

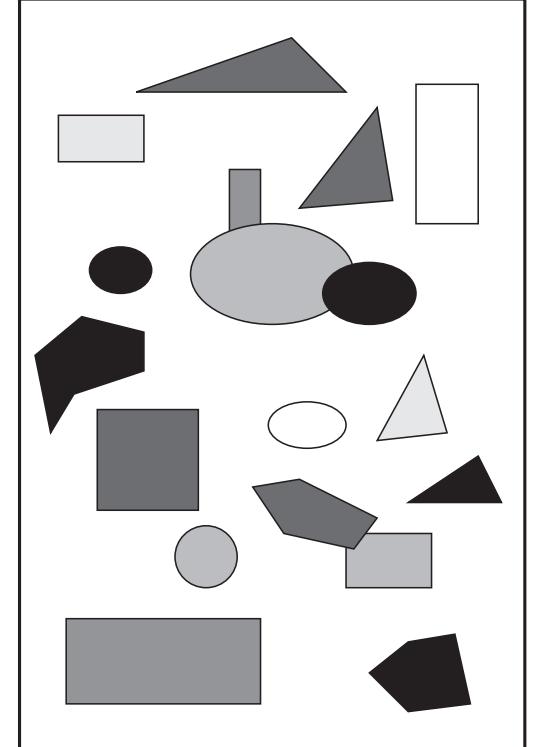


Conocim. de tareas



Conocim. del ordenador

Conocim. semántico



Conocim. sintáctico

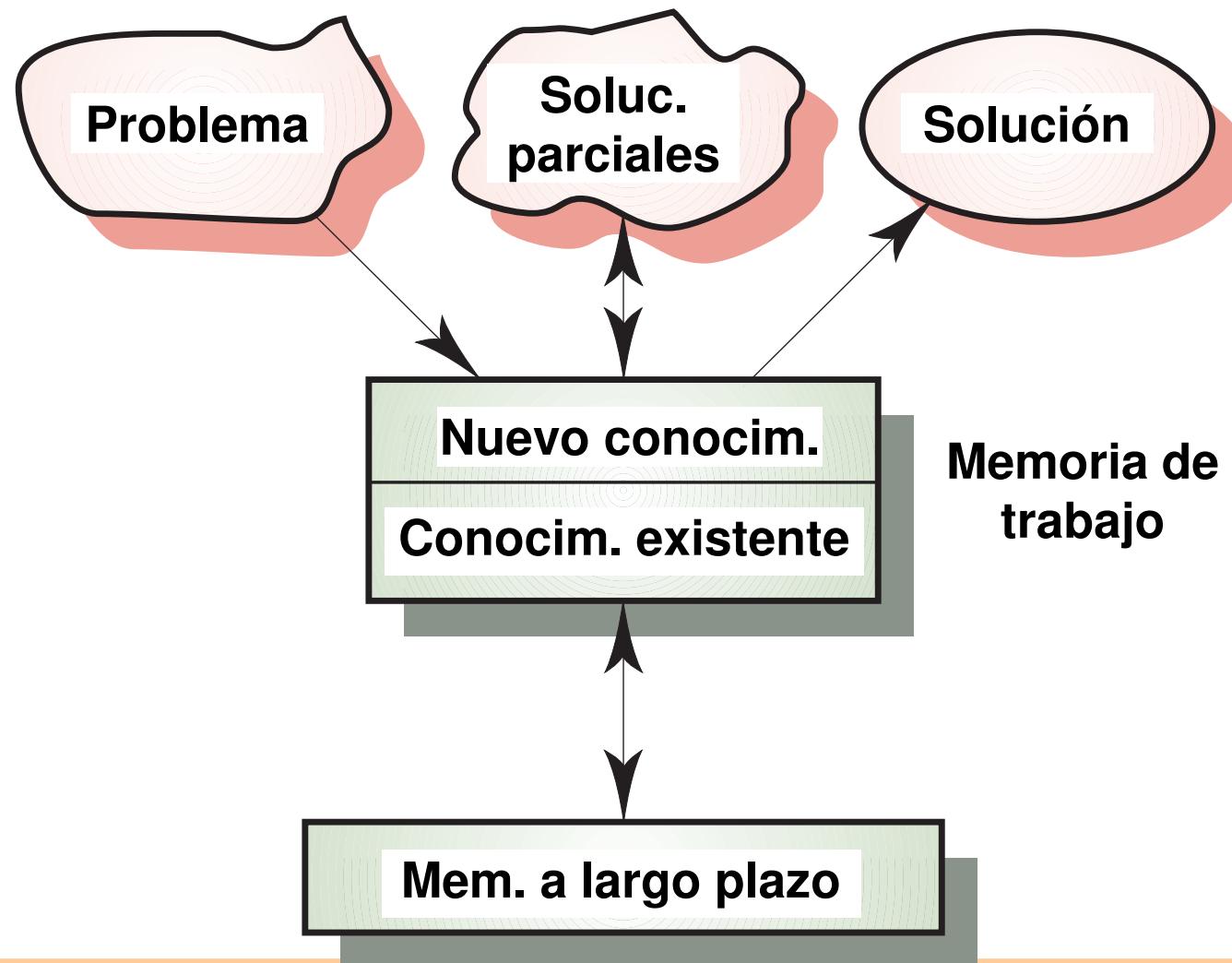
Adquisición del conocimiento

- El conocimiento semántico se adquiere mediante experiencia y aprendizaje activo
- El conocimiento sintáctico se adquiere mediante memorización.
- El nuevo conocimiento sintáctico puede interferir con conocimiento sintáctico ya existente.
 - Suele haber problemas con programadores experimentados en cuanto a mezcla de sintaxis entre diferentes lenguajes de programación

Resolución de problemas (I)

- Es independiente del lenguaje de programación
- Requiere la **integración** de diferentes tipos de conocimiento (del ordenador, de tareas, del dominio, de la organización)
- Implica el desarrollo de un modelo **semántico** de la solución y la **prueba** de dicho modelo contrastándolo con el problema
- Conduce a la representación del modelo en una notación adecuada o lenguaje de programación

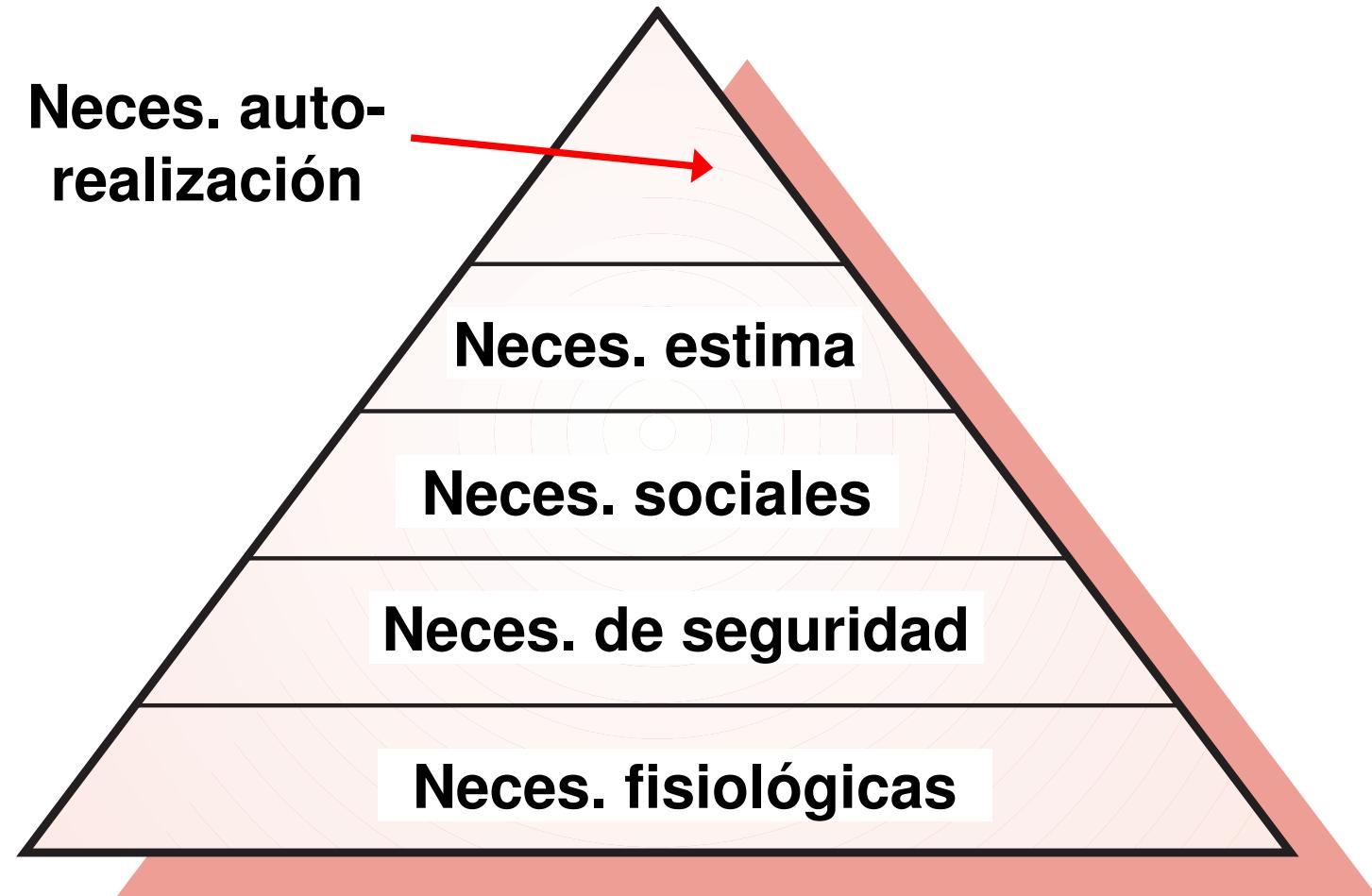
Resolución de problemas (II)



Motivación

- Una tarea importante de un gestor es la de motivar a la gente que trabaja en un proyecto
- La motivación es una tarea compleja. Se considera que hay diferentes tipos de motivación basadas en:
 - Necesidades básicas (ej. comer, dormir, etc.)
 - Necesidades personales (ej. respeto, autoestima)
 - Necesidades sociales (ej. ser aceptado como parte de un grupo)

Jerarquía de necesidades



Tipos de personalidad

- La jerarquía de necesidades es realmente una simplificación de la realidad
- La motivación debería tener en cuenta también los diferentes tipos de personalidad:
 - Orientados a la tarea
 - Orientados a sí mismos
 - Orientados a la interacción

Aspectos sobre motivación

- La motivación se consigue en la medida en que se puedan satisfacer las necesidades de un individuo
- La motivación puede cambiar dependiendo de circunstancias externas al personal o eventos externos
- La gente no solamente está motivada por factores personales, sino por los derivados de formar parte de un grupo y una cultura.

El trabajo en grupo

- La mayor parte del trabajo de ingeniería del software es una actividad de grupo
 - El desarrollo de la agenda para la mayor parte de proyectos no triviales es tal que no puede completarse por una persona trabajando en solitario
- La **interacción** con el grupo es una clave determinante del rendimiento del grupo
- La flexibilidad en la **composición** de un grupo es limitada
 - Los gestores deben intentar trabajar lo mejor posible con la gente disponible

Distribución del tiempo



Composición del grupo

- Un grupo formado por miembros que comparten la misma motivación puede ser problemático
 - Orientados a la tarea - cada uno quiere hacer las cosas según su propio criterio
 - Orientados a sí mismo - cada uno quiere ser el jefe
 - Orientados a la interacción - demasiadas "charlas", no suficiente trabajo
- Un grupo efectivo tiene un equilibrio de todos los tipos
- La mayoría de los ingenieros son orientados a la tarea
- Necesidad de que todos los miembros se impliquen en las decisiones que afecten al grupo

El líder del grupo

- El liderazgo del grupo se debe basar en el **respeto**, no en un título que proporciona un "status"
- Debe haber un líder administrativo y técnico
- Un liderazgo democrático es más efectivo que uno autocrático
- Se debería soportar una trayectoria de carrera profesional basada en la competencia técnica

Cohesión del grupo

- En un grupo cohesivo, los miembros consideran que el grupo es más importante que un individuo del mismo
- Ventajas de un grupo cohesivo:
 - Se pueden desarrollar estándares de calidad del grupo
 - Los miembros del grupo trabajan estrechamente, por lo que se reducen las inhibiciones causadas por la ignorancia
 - Los miembros del grupo aprenden unos de otros y dan a conocer su trabajo entre sus miembros
 - Se puede practicar la "programación sin ego", en la que los miembros se esfuerzan por mejorar el trabajo de los demás

Desarrollo de la cohesividad

- La cohesividad está influenciada por factores tales como la cultura organizacional y las personalidades del grupo
- La cohesividad se puede propiciar mediante
 - Eventos sociales
 - Desarrollar una identidad de grupo y un área propia
 - Actividades explícitas de construcción de grupos
- La sinceridad con la información es una forma sencilla de asegurar que todos los miembros se sientan parte del grupo

Comunicaciones del grupo

- Una buena comunicación es esencial para el trabajo efectivo del grupo
- Se debe intercambiarse información sobre el estado del trabajo, las decisiones de diseño y los cambios en las decisiones previas
- Una buena comunicación fortalece la cohesividad del grupo y promueve un mayor entendimiento
- Factores que influyen:
 - Status de sus miembros
 - Personalidades de sus miembros
 - Composición sexual del grupo
 - Canales de comunicación

Organización del grupo

- Tamaño del grupo relativamente pequeño (menos de ocho personas)
- Dividir los proyectos grandes en múltiples proyectos pequeños
- Los grupos pequeños pueden organizarse de forma informal y democrática
- El jefe de programadores intentará hacer un uso efectivo de las habilidades y experiencia del grupo

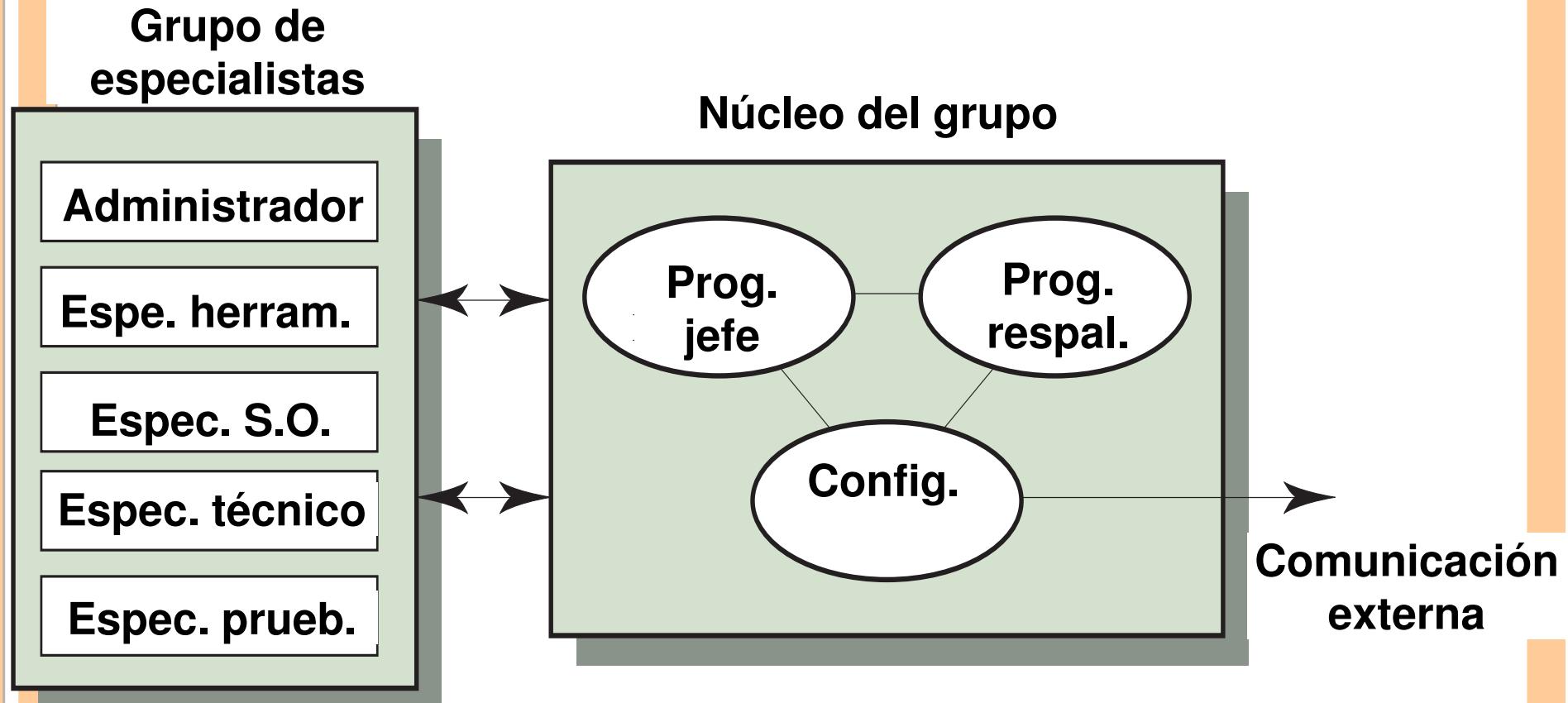
Organización democrática

- El grupo actúa como un todo y las decisiones se toman por consenso
- El líder del grupo sirve como un enlace externo del grupo, pero no realiza asignaciones específicas de trabajo
- El trabajo se discute por el grupo como un todo y las tareas se reparten de acuerdo según la habilidad y experiencia de cada uno
- Esta aproximación tiene éxito en grupos cuyos miembros son todos competentes y con experiencia

Grupos de programac. extrema

- Los grupos de "programación extrema" son una variante de la organización democrática
- En dichos grupos, se toman algunas decisiones de gestión por parte de los miembros del grupo
- Los programadores trabajan por parejas y adquieren una responsabilidad colectiva del código que han desarrollado

Grupos con jefe de trabajo



Problemas

- Los buenos diseñadores y programadores no se encuentran fácilmente.
- Los restantes miembros del grupo pueden verse afectados por el hecho de que el jefe de programadores asuma el éxito de todo el grupo, dificultando su trabajo
- Debido a la estructura impuesta por la organización, puede no ser posible formar este tipo de grupo

Selección y organiz. de personal

- Se trata de una responsabilidad importante del gestor de proyectos
- Normalmente las decisiones se basan en:
 - información proporcionada por el candidato (su currículum)
 - información obtenida mediante una entrevista
 - recomendaciones de otra gente que conoce al candidato
- Algunas compañías utilizan test psicológicos y/o test de aptitud
 - No hay evidencias de si estos tests son o no son realmente útiles

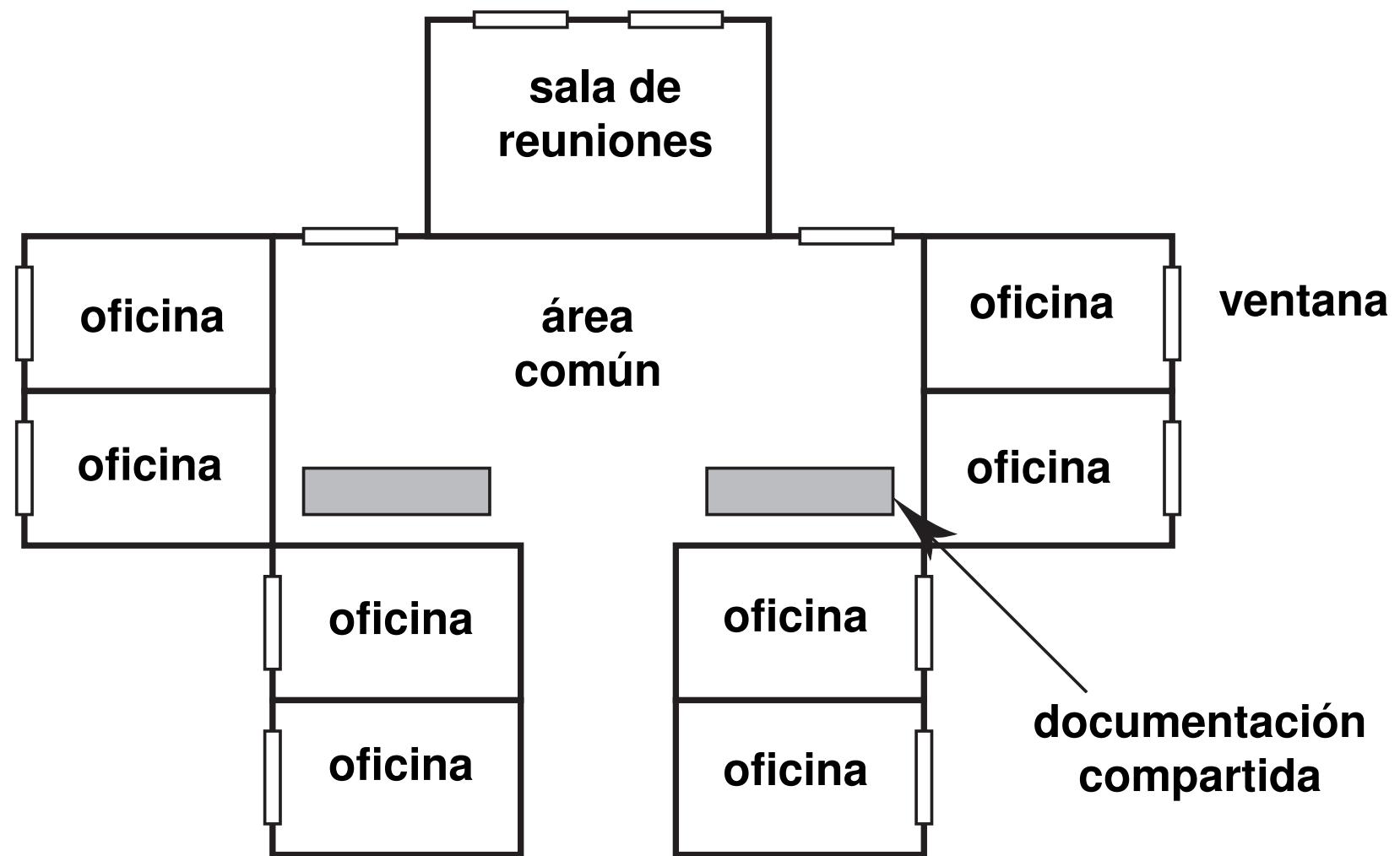
Factores de selección

- Experiencia en el dominio de la aplicación
- Experiencia en la plataforma
- Experiencia en el lenguaje de programación
- Estudios previos
- Capacidad de comunicación
- Adaptabilidad
- Actitud
- Personalidad

Entornos de trabajo

- El entorno físico de trabajo juega un papel importante en la productividad y satisfacción individual
 - Confort
 - Privacidad
 - Otras facilidades
- Las consideraciones acerca de la salud y la seguridad deberían tenerse en cuenta
 - Iluminación
 - Climatización
 - Mobiliario

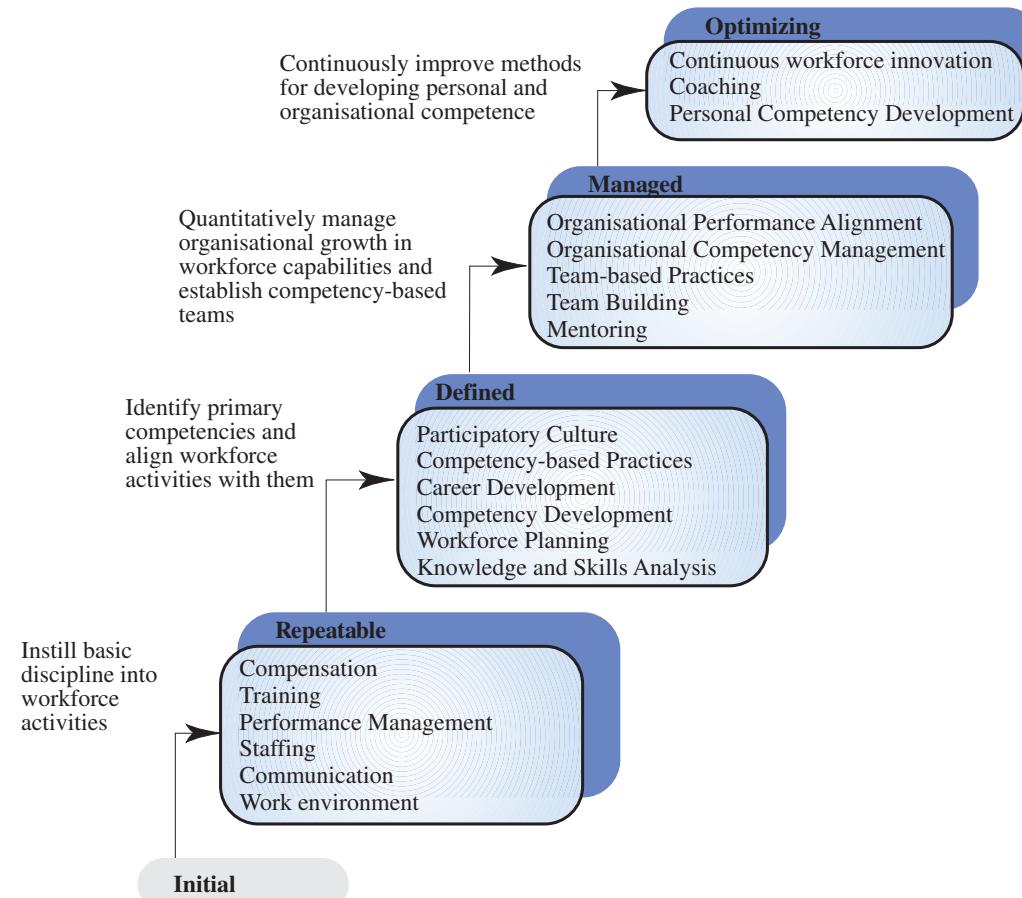
Oficina hipotética



El modelo CMM de personal

- Pretende ser un marco para la **gestión** del trabajo realizado por la gente implicada en el desarrollo del software
- Es un modelo de **cinco** etapas
 - Inicial. Gestión de rec. humanos "ad-hoc"
 - Repetible. Se desarrollan políticas para mejora de las capacidades (aptitudes)
 - Definido. Gestión de rec. humanos estandarizada para la organización
 - Gestionado. Se establecen metas cuantitativas para la gestión de recursos humanos
 - Optimizado. Se realiza un esfuerzo continuado para mejorar la competencia y motivación en el trabajo

El modelo CMM de personal



Objetivos P-CMM

- Mejorar las capacidades de la organización mejorando las capacidades de trabajo de la gente implicada
- Asegurar que las capacidades para el desarrollo del software no conciernen a un número pequeño de individuos
- Igualar la motivación de los individuos con la de la organización
- Ayudar a la "retención" de gente con conocimientos y habilidades críticas

Puntos clave

- Los gestores de software deben comprender algunos de los factores humanos para poder llevar a cabo su trabajo con éxito
- Los factores principales a considerar son:
 - La organización de la memoria
 - La representación del conocimiento
 - La influencia de la motivación
- La composición y comunicación de los grupos de trabajo resulta fundamental para el éxito del proyecto
- El modelo P-CMM proporciona un marco para mejorar las capacidades de los recursos humanos de una organización

Tema 5.

Gestión de la Configuración del Software

-  Definición
-  Identificación
-  Control de versiones
-  Control de cambios
-  Auditorías e informes
-  Construcción del sistema

Bibliografía

Captítulo 29. Gestión de configuraciones. Software Engineering Sommerville 7^a edición.

Capítulo 9. Gestión de configuraciones. Ingeniería del software. 5^a edición. Roger S. Pressman.

Definición

Actividad de autoprotección para:

- Identificar el cambio
- Controlar el cambio
- Garantizar que el cambio se implementa adecuadamente
- Informar del cambio a todos aquellos que les interese

Gestión de configuraciones ≠ Mantenimiento

Planificación de la GC

- Definición de entidades a gestionar y esquema formal de identificación
- Identificación de responsables de los procedimientos de la GC
- Descripción de cómo los registros de documentos del proceso de GC deberían ser mantenidas
- Descripción de las herramientas usadas
- Definición de la base de datos de configuración (BDC) usada para almacenar la información

Identificación de EC

Los elementos de configuración son aquellos documentos que se van a requerir para un futuro mantenimiento

Nombrado jerárquico:

- Asociado a proyectos particulares
- No reusable

No debería cambiar de forma arbitraria

Base de datos de configuraciones (I)

Usada para registrar cualquier información relevante relacionada con las configuraciones

Sirve de apoyo a la evaluación del impacto del cambio

Debe proporcionar respuestas a preguntas cómo:

- (1) ¿A qué clientes se les ha entregado una versión particular del sistema?
- (2) ¿Qué hw y SO son necesarios para ejecutar una determinada versión?
- (3) ¿Cuántas versiones del sistema se han creado y cuáles son sus fechas de creación?

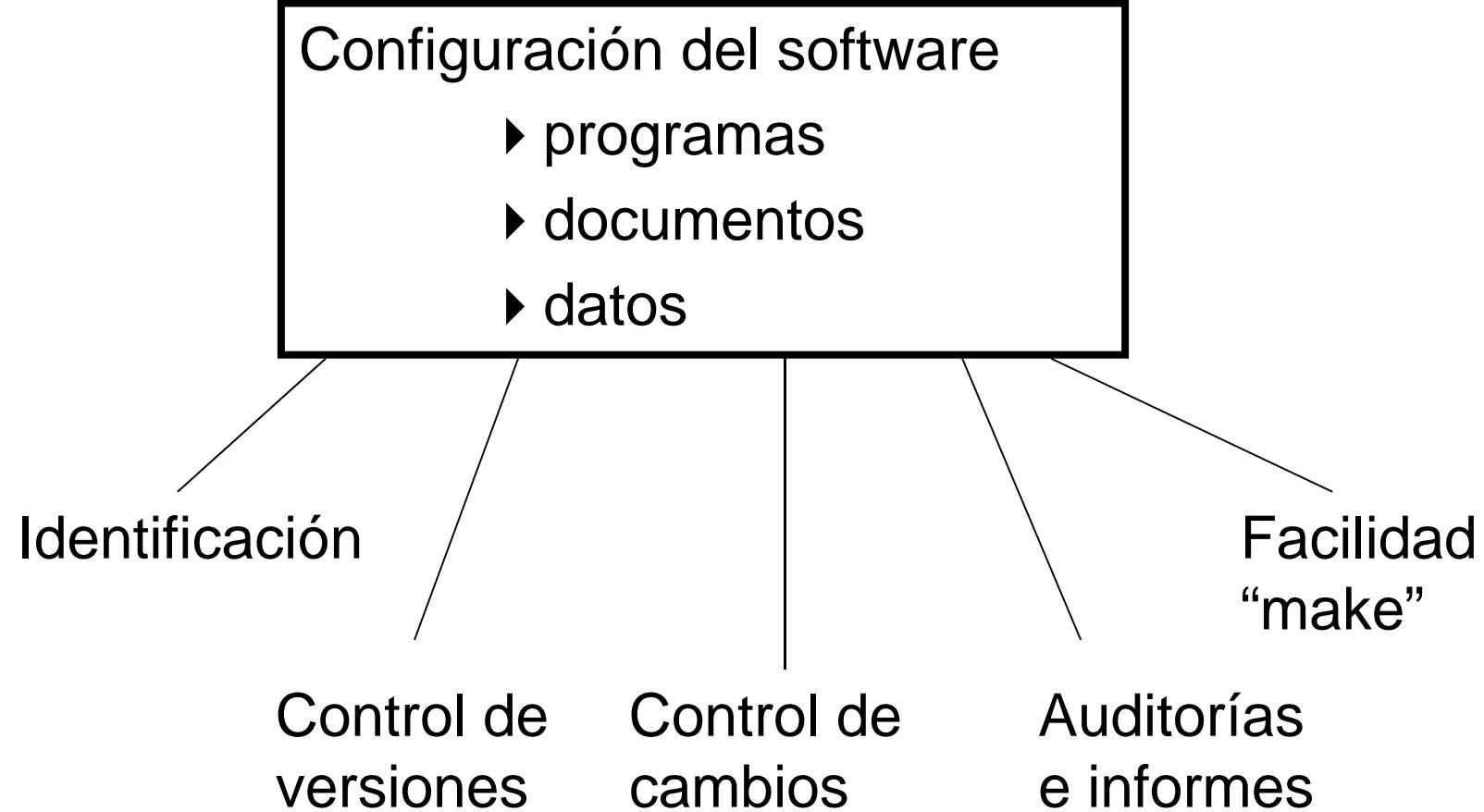
Base de datos de configuraciones (II)

- (4) ¿Qué versiones de un sistema podrían verse afectadas si una componente particular es cambiada?
- (5) ¿Cuántas peticiones de cambio se han hecho sobre una determinada versión?
- (6) ¿Cuántos fallos se han registrado para una versión particular?

Puede ser creada como:

- Un sistema separado
- Integrada con la gestión de versiones y sistema de control que almacena los documentos formales del proyecto

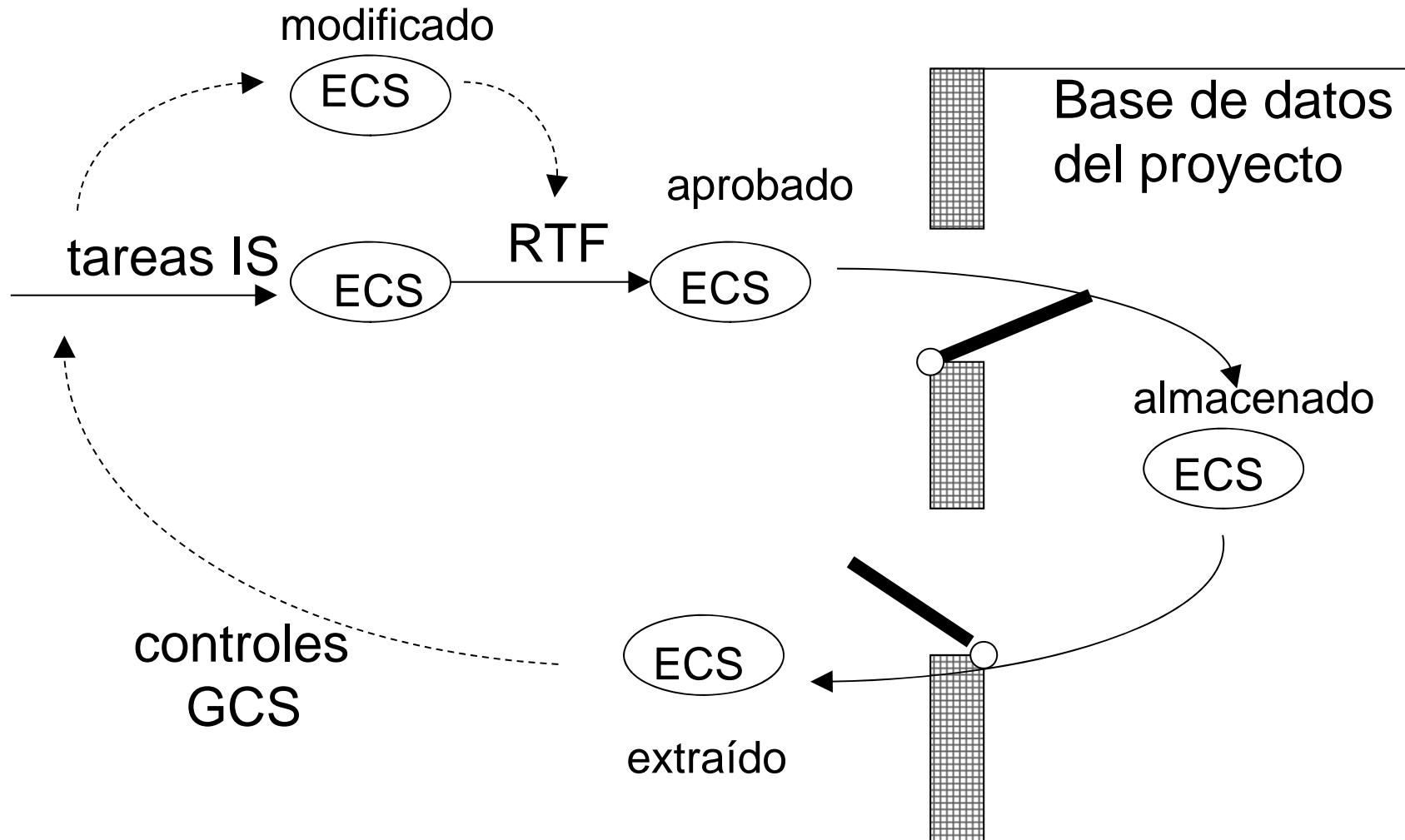
Gestión de configuraciones



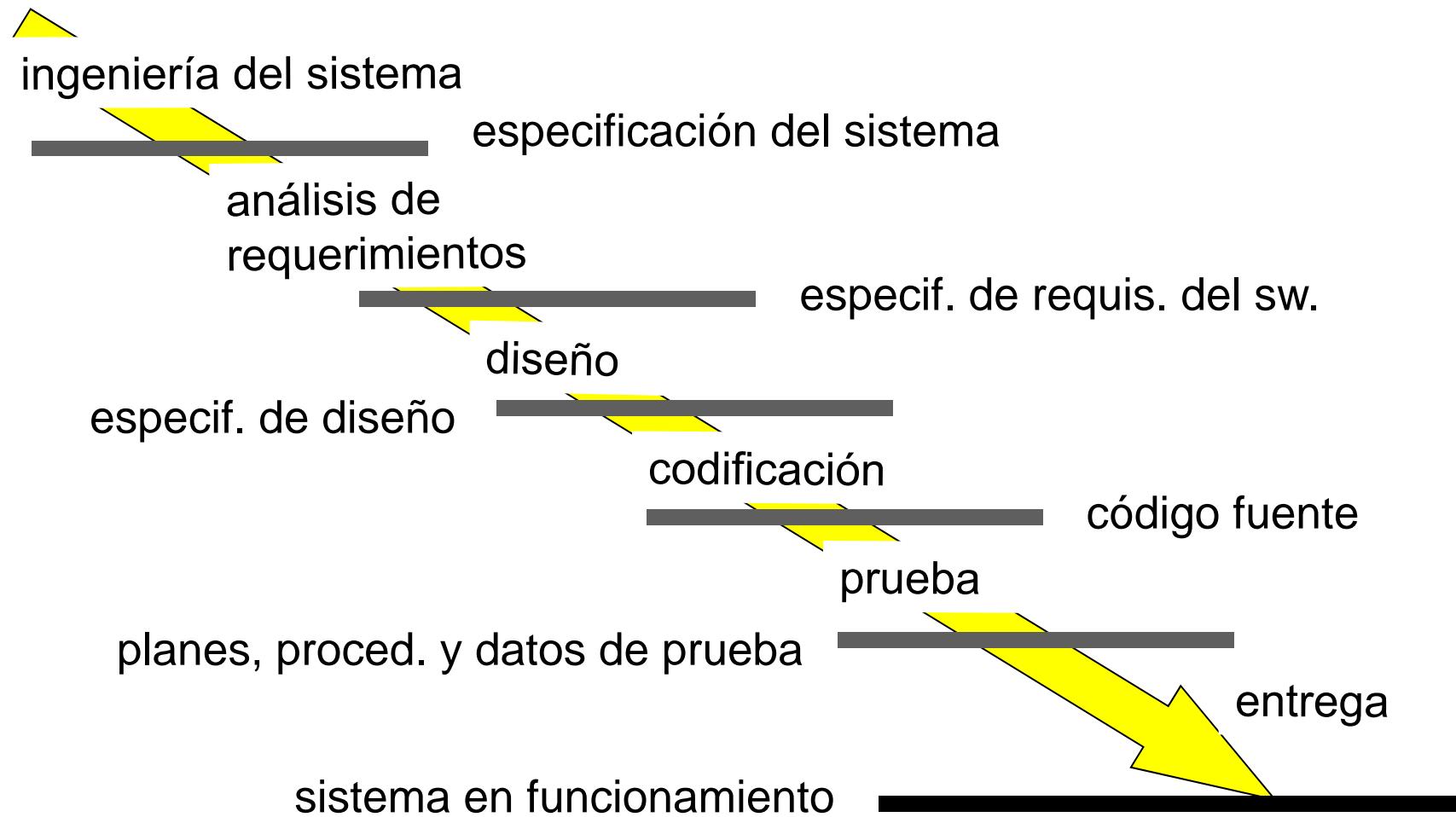
ECS'S típicos

-  Especificación del sistema
-  Plan del proyecto software
-  Especificación de requerimientos del software
-  Manual de usuario preliminar
-  Especificación de diseño
-  Diseño preliminar y detallado
-  Listados de código fuente
-  Planes y procedimientos de prueba
-  Manual de instalación, operación y usuario
-  Ejecutables software

Líneas base (I)



Líneas base (II)



Control de versiones (I)

Implica la identificación y seguimiento de diferentes versiones y “*releases*” del sistema

- Diseño de procedimientos para asegurar que diferentes versiones del sistema puedan recuperarse cuando se requieran, y evitar que no sean cambiadas accidentalmente
- Hay un contacto con el cliente para planificar cuándo deberían distribuirse nuevas *releases*

Control de versiones (II)

VERSION: instancia de un sistema que difiere de algún modo de otras instancias (funcionalidad, rendimiento, fallos...)

RELEASE: es una versión del sistema distribuida a los clientes (funcionalidades nuevas o nueva plataforma)

Incluye:

- Ficheros de configuración
- Ficheros de datos
- Instalación del programa
- Documentación electrónica y sobre papel describiendo el sistema

Control de versiones (III)

Nueva versión ➔ nuevo fuente + construcc. stma. (1)

Nueva release ➔ (1) + fich. datos y conf + nueva docum.

El grupo de GC debe decidir cuándo los componentes afectados por un cambio deben ser reconstruidos en una nueva versión o una nueva release

Decisión forzada a veces por fallos descubiertos por el cliente
➔ parcheo del código objeto (mejor solución: versión nueva sin documentación)

Herramientas de gest. de versiones

Unix : SCCS, RCS

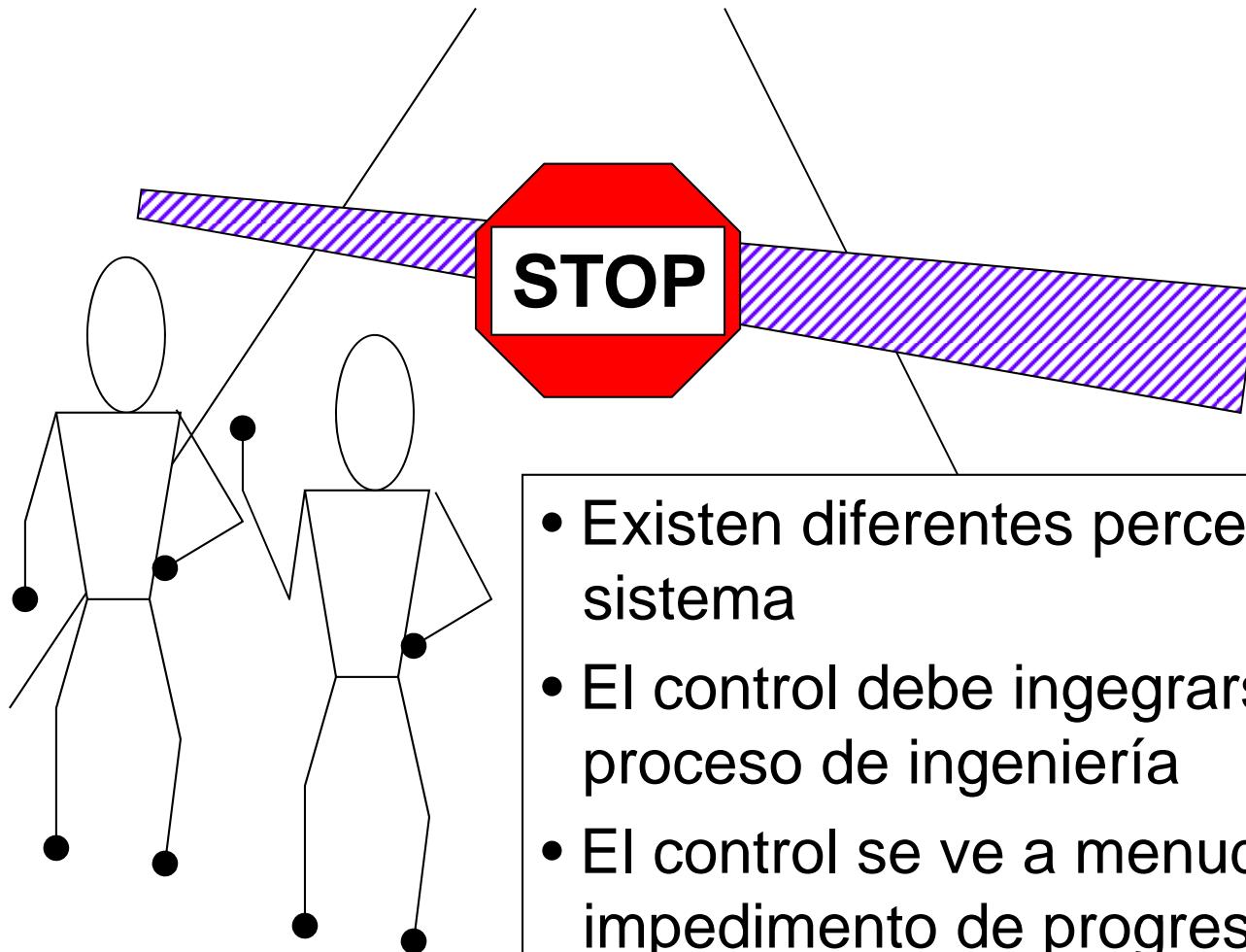
- Identificación de versiones y releases
- Cambios controlados
- Gestión de almacenamiento
- Registro de la historia de cambios

RCS:

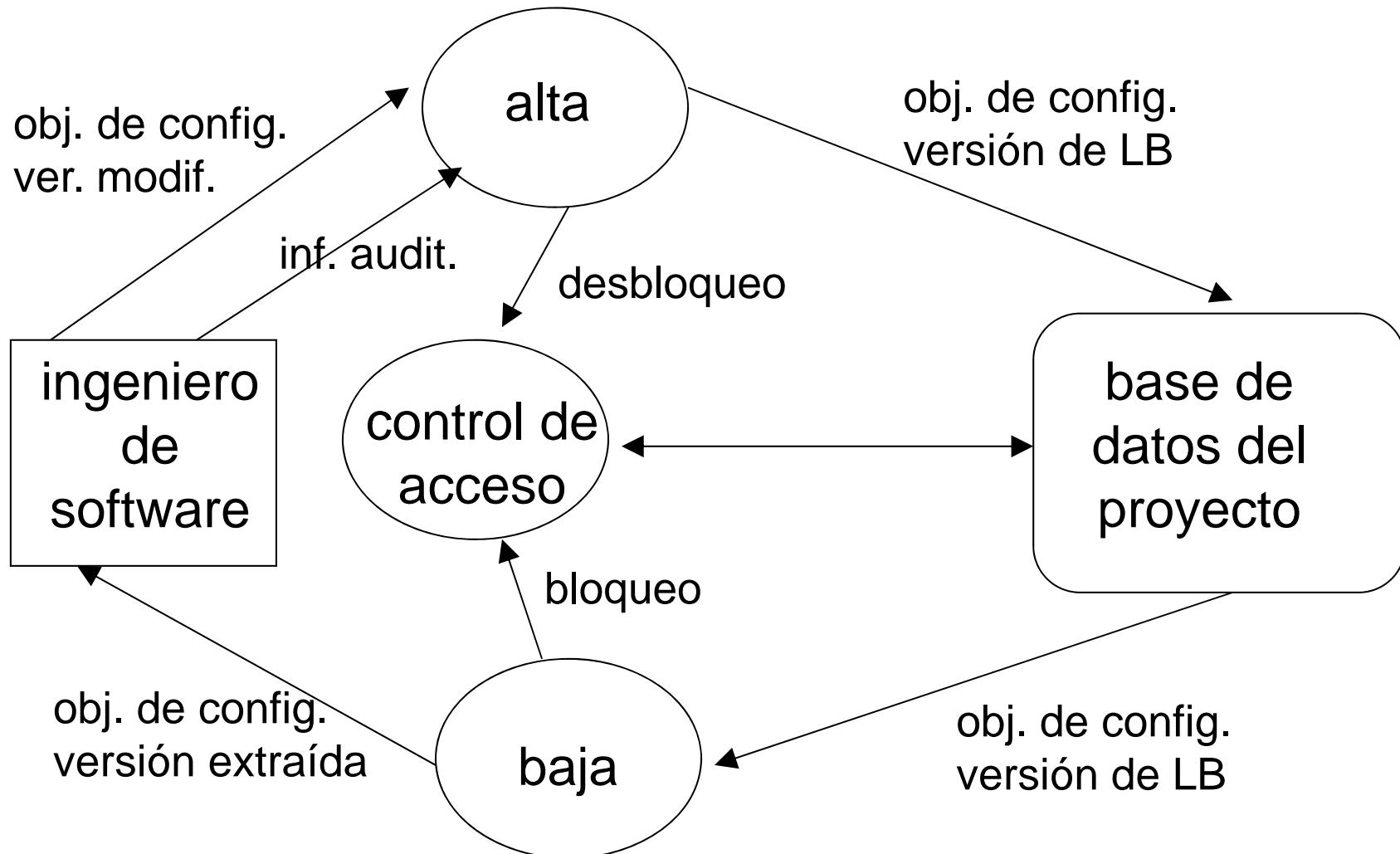
- Graba el código fuente de la versión más reciente
- Soporta el desarrollo paralelo de diferentes releases
- Capacidad de “mezcla” de versiones

Diseñados para trabajar con texto ASCII

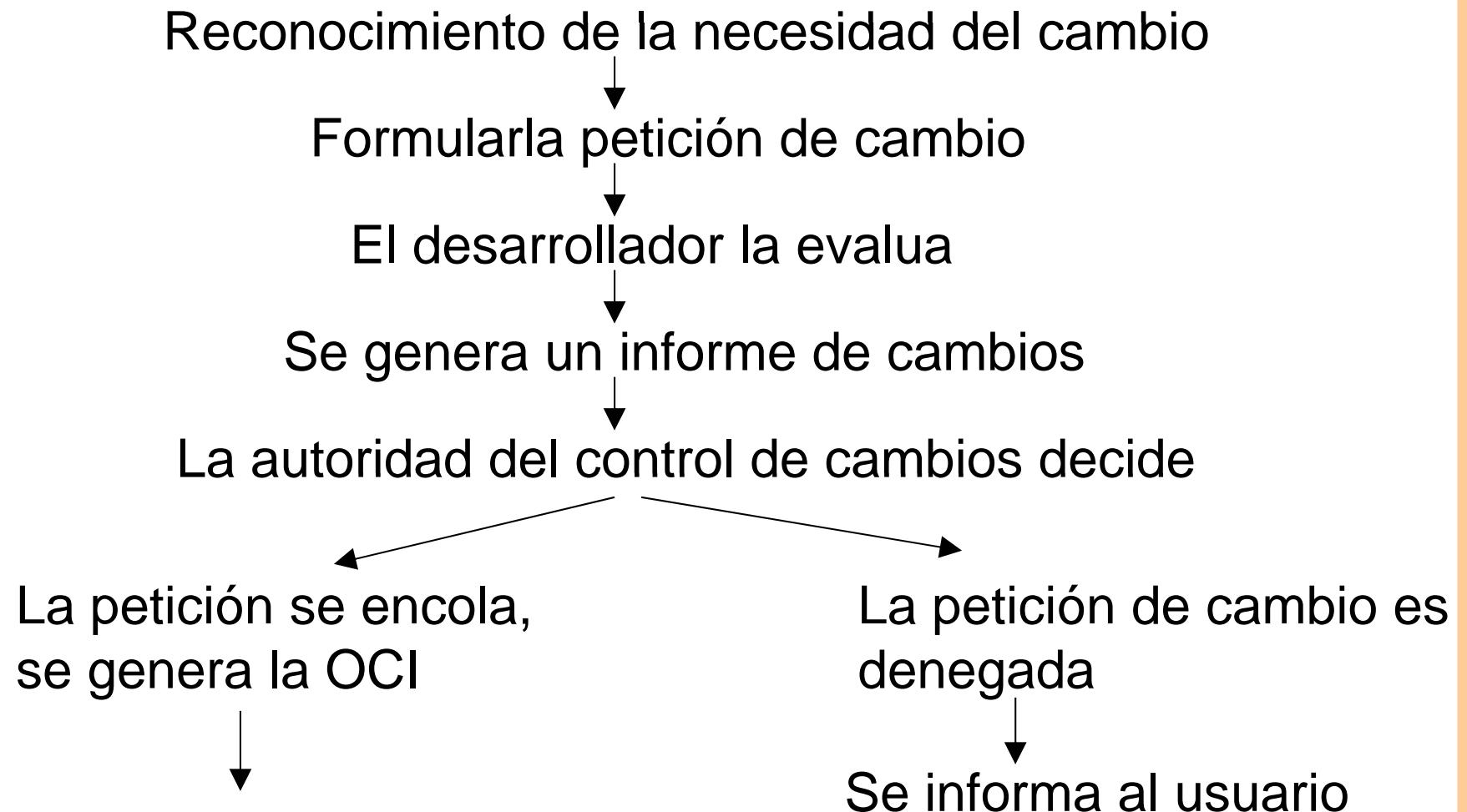
Control de Cambios



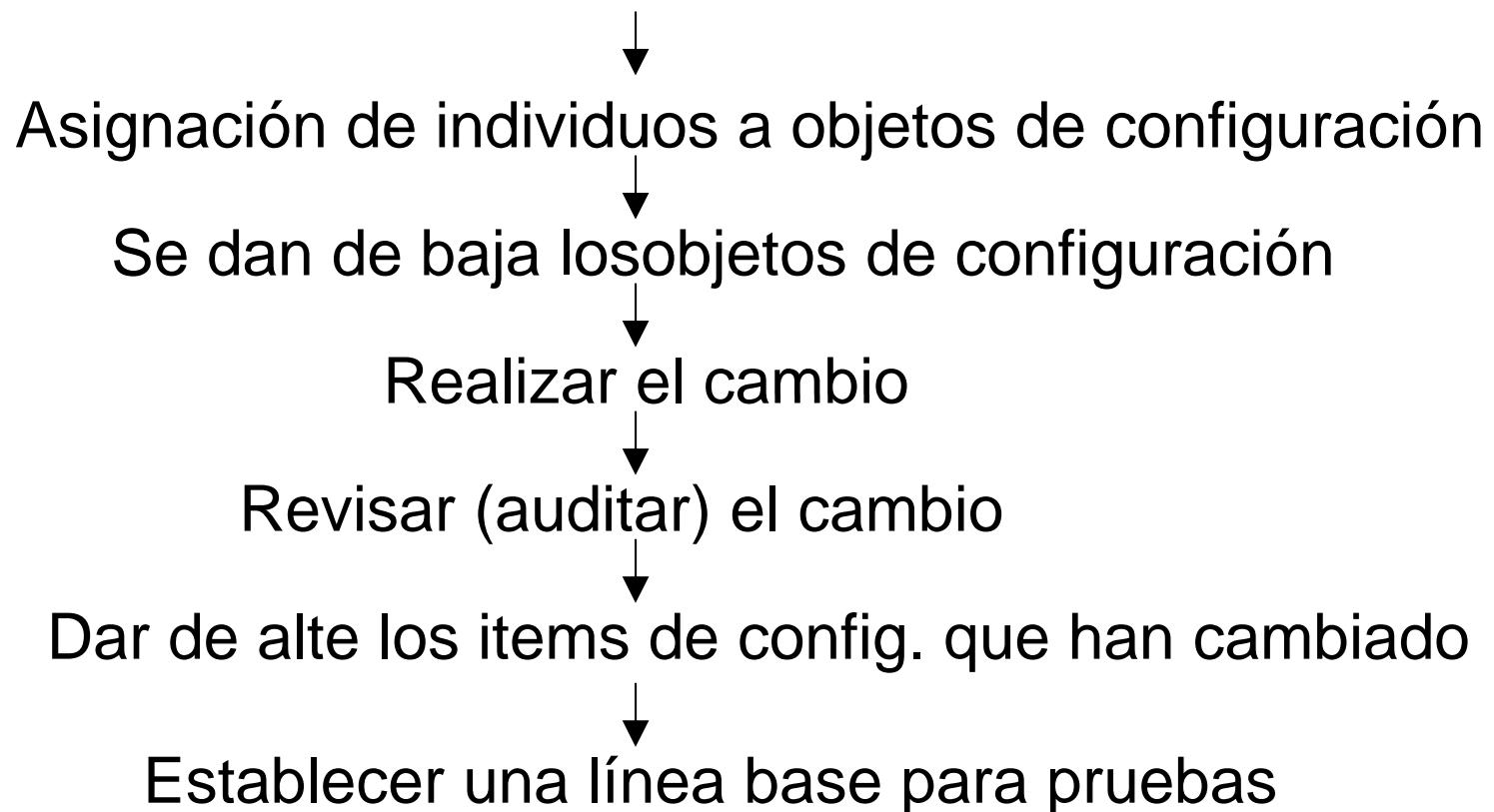
Control de Acceso



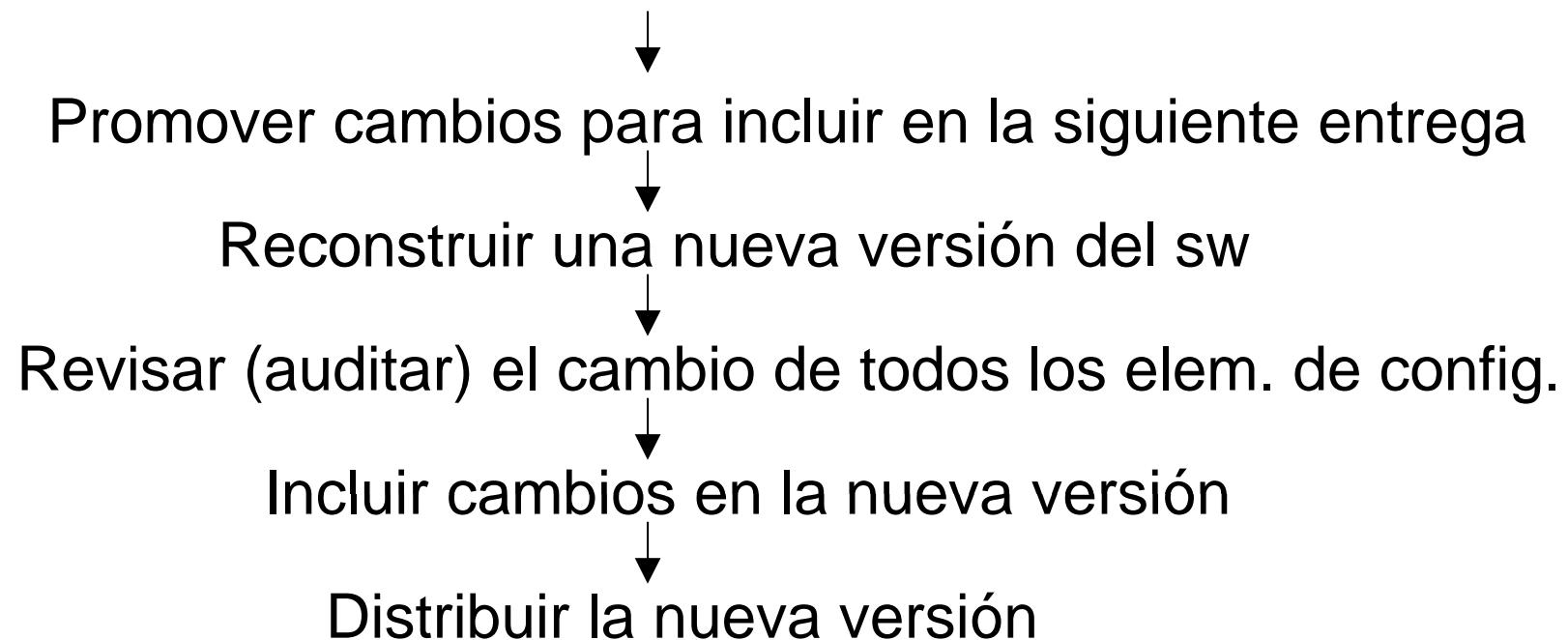
Proceso de control de cambios (I)



Proceso de control de cambios (II)



Proceso de control de cambios (III)



Auditoría de la Configuración

¿Cómo podemos asegurar que el cambio se ha implementado correctamente?

- ❑ Revisiones técnicas formales
 - «corrección técnica»
- ❑ Auditorías de configuración
 - «responde a las siguientes preguntas:
 - ❑ ¿se ha hecho el cambio especificado?
 - ❑ ¿se han seguido los estándares de IS?
 - ❑ ¿se han seguido procedimientos para señalar el cambio, registrarlo y divulgarlo?
 - ❑ ¿se han actualizado adecuadamente todos los ECS relacionados? »

Informes de estado

Responde a las siguientes preguntas:

- ❑ ¿Qué pasó?
- ❑ ¿Quién lo hizo?
- ❑ ¿Cuándo pasó?
- ❑ ¿Qué más se vió afectado?

Construcción del sistema (I)

Proceso de combinar componentes en un programa que se ejecuta sobre una configuración destino particular

Implica:

- Compilación
- Linkado

Hay que llevar especial cuidado con sistemas desarrollados con un sistema distinto del de destino.

Construcción del sistema (II)

Factores a considerar:

- ¿Han sido incluidas todas las componentes?
- ¿Tienen la versión adecuada?
- ¿Están disponibles todos los ficheros de datos?
- ¿Los datos tienen el mismo nombre en la componente y en la máquina destino?
- ¿La versión del compilador es la adecuada?

El proceso se refiere normalmente a componentes físicos.

Herram. de construcc. del sistema

- La herramienta más ampliamente usada para sistemas UNIX es MAKE.
- Mantiene una correspondencia entre el código fuente y las versiones de código objeto de un sistema
- El usuario especifica las dependencias de los componentes y MAKE fuerza automáticamente la recompilación de los ficheros cuyo código fuente haya cambiado después de que el código objeto fuese creado

Limitaciones de “make”

- Basado en un modelo físico de dependencias (no lógico)
- Las especificaciones de dependencias (Makefiles) crecen rápidamente, llegando a ser complejas, difíciles de comprender y “caras” de mantener
- Usa simplemente un modelo de cambio basado en fechas de actualización de ficheros. Puede que cambios en el código fuente no necesiten recompilación
- No permite (fácilmente) especificar la versión de utilización de herramientas como el compilador
- No “finamente” enlazado con herramientas de gestión de configuraciones como RCS

Beneficios GCS

- Reduce el esfuerzo necesario para gestionar y realizar el cambio - mejora la productividad
- Conduce a una mejora de la integridad y seguridad del software - incremento de la calidad
- Genera información sobre el proceso - mejora de la gestión del control
- Mantiene una base de datos de desarrollo de software - mejor registro y seguimiento de informes

Condiciones de uso de la videoconferencia que se va a utilizar

- a) La sesión va a ser grabada con el objeto de facilitar al alumnado, con posterioridad, el contenido de la sesión docente.
- b) Se recomienda a las personas asistentes que desactiven e inhabiliten la cámara de su dispositivo si no desean ser visualizados por el resto de las personas participantes.
- c) Queda prohibida la captación y/o grabación de la sesión, así como su reproducción o difusión, en todo o en parte, sea cual sea el medio o dispositivo utilizado. Cualquier actuación indebida comportará una vulneración de la normativa vigente, pudiendo derivarse las pertinentes responsabilidades legales.

Tema 6.

Monitorización y control.

- Monitorización de una agenda
- Métricas de seguimiento y control: holguras y EVA
- Control de una agenda

Se puede mejorar!

- El 70% de los proyectos:
 - Cuestan más de lo presupuestado
 - Se entregan más tarde de lo planificado
- El 52% de los proyectos:
 - Se entregan con un 189% de lo presupuestado
- Y otros, después de invertir tiempo y dinero, simplemente nunca se entregan



Monitorización de una agenda

- Monitorizar (hacer un seguimiento) una agenda consiste en comprobar si la agenda real se ajusta a la planificada
 - A la agenda creada inicialmente la denominaremos agenda planificada (muestra nuestra intención inicial, a partir de información planificada)
 - A medida que el proyecto progresá se creará una agenda real (muestra lo que realmente está ocurriendo, a partir de información real)
- Para monitorizar la agenda necesito hacer uso de diversas **MÉTRICAS** del proyecto
 - Una métrica es cualquier tipo de medición que proporciona un valor **cuantitativo** para indicar el grado en el que un sistema, componente o proceso posee un determinado atributo
 - Por ejemplo la holgura libre de una actividad indica cuánto puedo retrasar esa actividad sin afectar a las siguientes ni a la finalización del proyecto

Métricas de seguimiento del proyecto

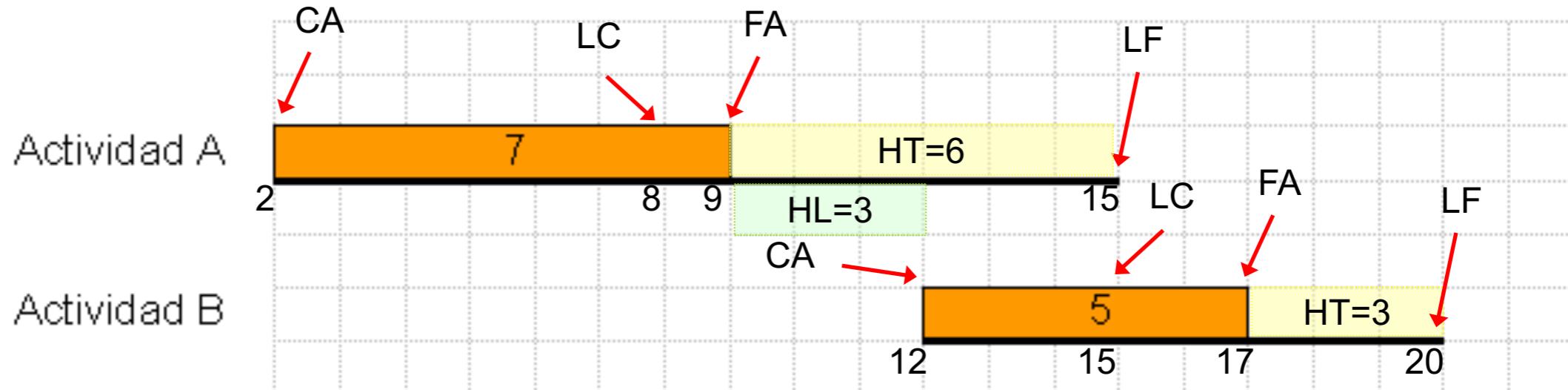
- Fechas de inicio/fin
 - Duraciones
 - Holguras totales y libres
 - Análisis del Valor Acumulado (EVA)
-
- La cuestión es: ¿cómo utilizo las métricas anteriores para MONITORIZAR Y CONTROLAR el proyecto?
 - Recordemos que nuestro objetivo es TOMAR LAS DECISIONES adecuadas que garanticen el éxito del proyecto:
 - ✿ cumplir con los plazos temporales, costes y expectativas del usuario PREVISTOS

Programación de la agenda prevista

- Una vez que tenemos las entradas: **actividades, dependencias, duraciones y recursos**, se realizan los siguientes **CÁLCULOS** (entre otros):
 - Fechas de inicio y fin más tempranas y tardías de las actividades
 - Holguras totales y libres de las actividades
 - ★ Tiempo que una actividad puede retrasarse sin retrasar el proyecto (**HOLGURA TOTAL**: HT)
 - ★ Tiempo que una actividad puede retrasarse permitiendo que las actividades siguientes puedan comenzar lo más pronto posible (**HOLGURA LIBRE**: HL)
 - Camino crítico
 - ★ Secuencia de actividades con holgura total 0
 - Cálculos de EV (Earned Value)

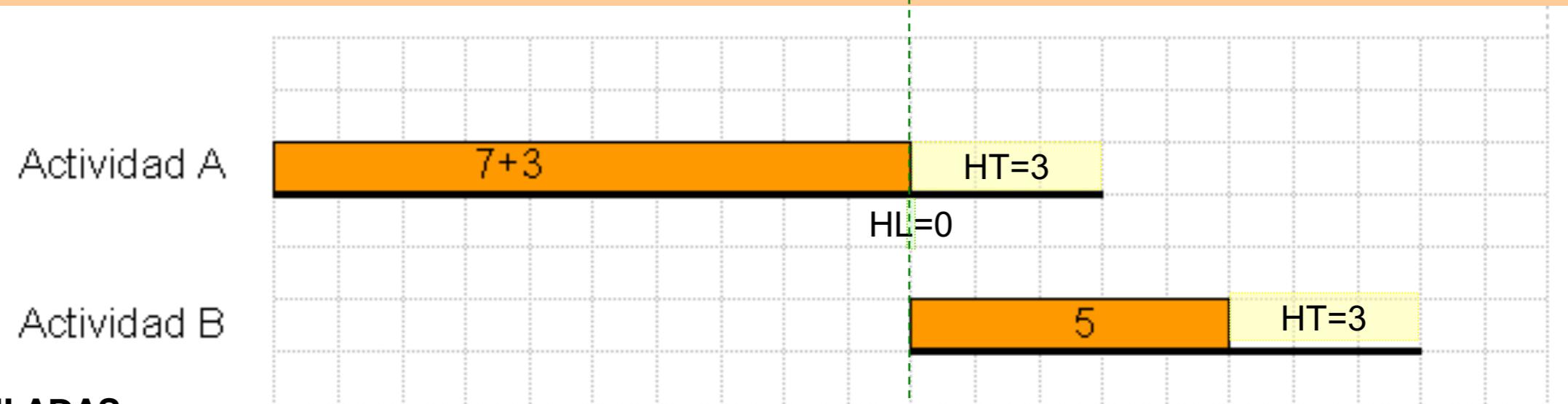
Para tener nuestra agenda sólo nos faltaría establecer una
FECHA DE INICIO

Cálculo de Holguras: veamos un ejemplo gráfico



- $H_T(A) = 15 - 2 - 7 = 6; H_L(A) = 12 - 2 - 7 = 3$
- $H_T(B) = 20 - 12 - 5 = 3;$
- ¿Qué pasa si A se retrasa 3 días?

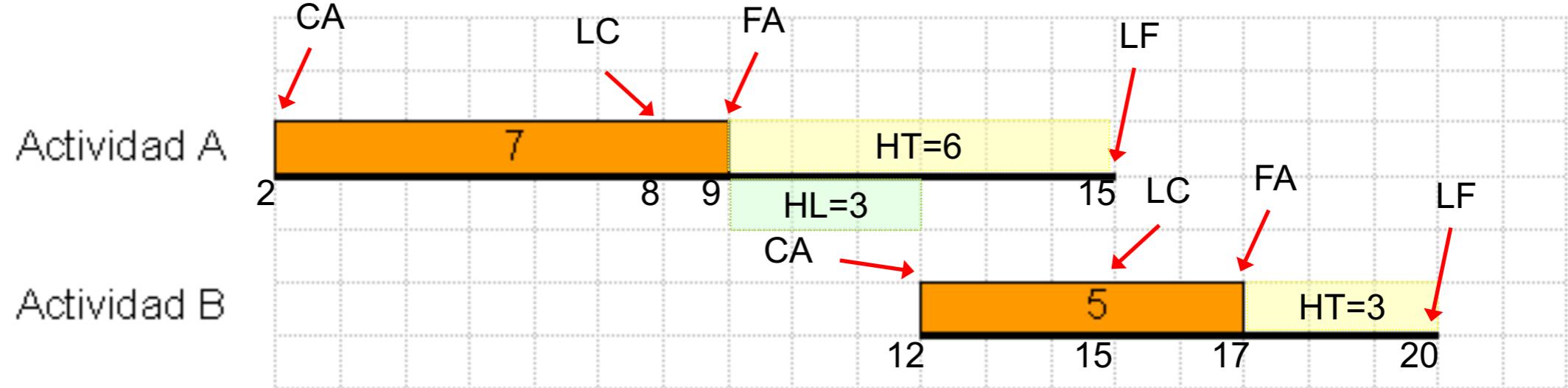
✓ El proyecto NO se retrasa
✓ B puede comenzar lo antes posible



FECHAS CALCULADAS

CA: Comienzo anticipado (fecha más temprana de comienzo) FA: Fin anticipado (fecha más temprana de fin)
LC: Límite de comienzo (fecha más tardía de comienzo) LF: Límite de fin (fecha más tardía de fin)

Holguras: cómo afectan las actividades siguientes

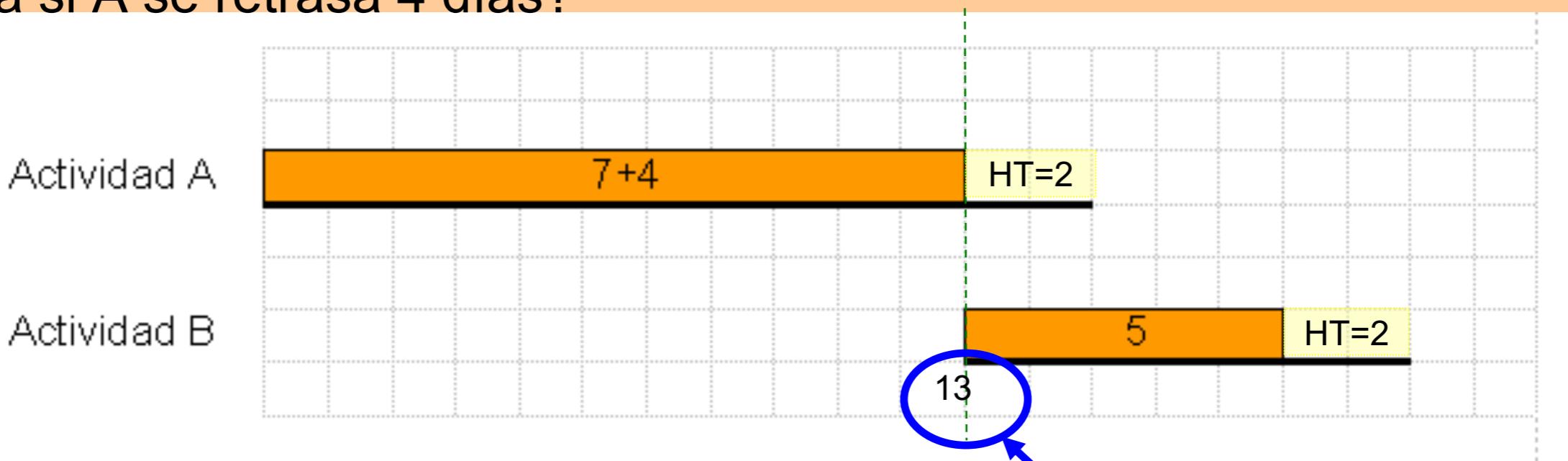


■ $H_T(A) = 15 - 2 - 7 = 6; H_L(A) = 12 - 2 - 7 = 3$

■ $H_T(B) = 20 - 12 - 5 = 3;$

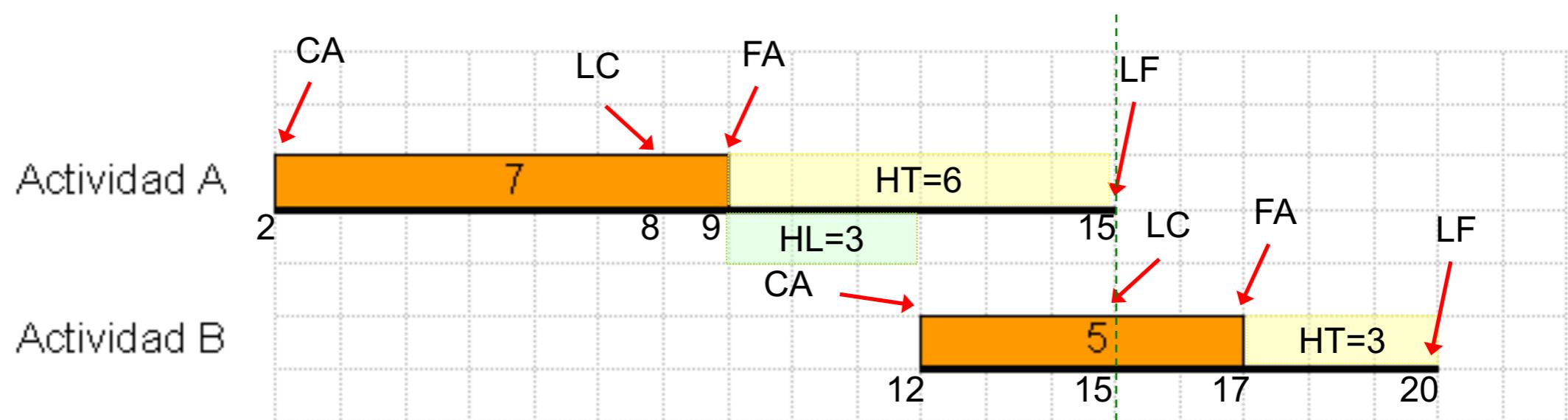
■ ¿Qué pasa si A se retrasa 4 días?

- ✓ El proyecto NO se retrasa
- ✓ B reduce su HT en 1 unidad



B tiene que empezar 1 día más tarde

Holguras: nuevos caminos críticos

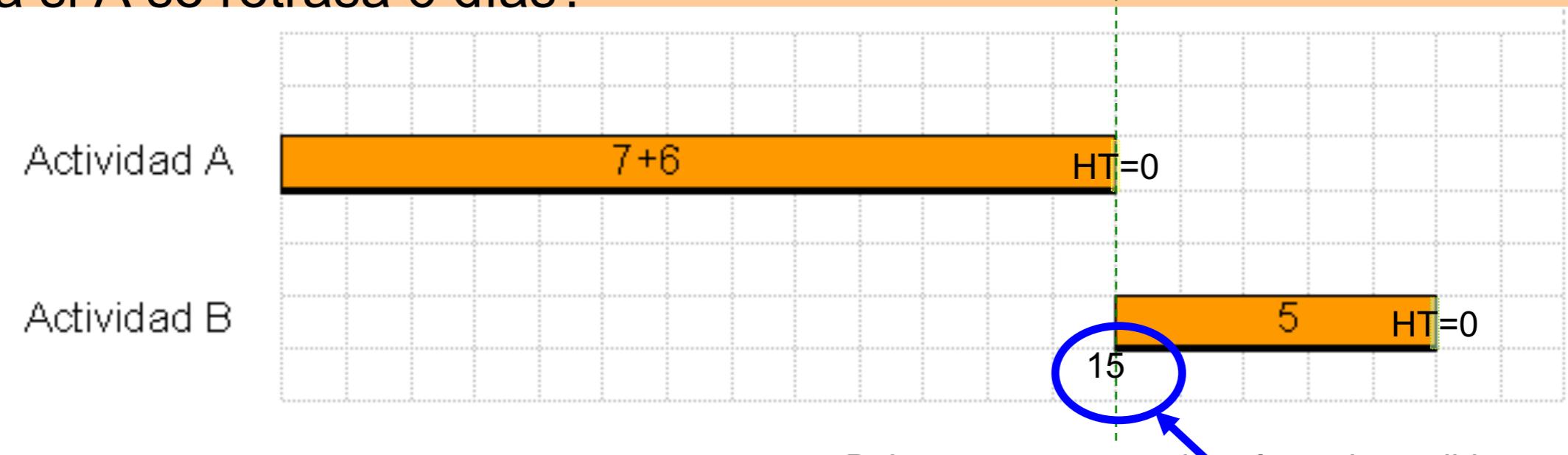


■ $H_T(A) = 15 - 2 - 7 = 6; H_L(A) = 12 - 2 - 7 = 3$

■ $H_T(B) = 20 - 12 - 5 = 3;$

■ ¿Qué pasa si A se retrasa 6 días?

- ✓ El proyecto NO se retrasa
- ✓ B se convierte en CRITICA



Ms Project y Fechas/Holguras programadas

Campos de FECHA:

- Comienzo anticipado, Límite de comienzo
- Fin anticipado, Límite de finalización

	Nombre de tarea	Comienzo anticipado	Límite de comienzo	Fin anticipado	Límite de finalización
1	- Proyecto 1	jue 06/03/08	jue 06/03/08	jue 03/04/08	jue 03/04/08
2	A	jue 06/03/08	vie 07/03/08	lun 17/03/08	mar 18/03/08
3	B	jue 06/03/08	jue 06/03/08	vie 14/03/08	vie 14/03/08
4	C	jue 06/03/08	mar 18/03/08	vie 07/03/08	mié 19/03/08
5	D	mar 18/03/08	mié 19/03/08	mié 19/03/08	jue 20/03/08
6	E	jue 17/03/08	jue 17/03/08	jue 20/03/08	jue 20/03/08

Campos de HOLGURA:

- Demora permisible (Holgura libre)
- Margen de demora total (Holgura total)

	Nombre de tarea	Demora permisible	Margen de demora total
1	- Proyecto 1	0 días	0 días
2	A	0 días	1 día
3	B	0 días	0 días
4	C	5 días	8 días
5	D	1 día	1 día
6	E	0 días	0 días
7	~	0 ~	0 ~

La vista de Gantt detallado muestra gráficamente las holguras libres

Earned Value Analysis (EVA)

- El EV (*earned value*) o valor acumulado es una métrica que proporciona una información CUANTITATIVA del PROGRESO de un proyecto
 - A cada tarea se le asigna un valor devengado basado en su porcentaje estimado del valor total
- Permite vislumbrar dificultades en la agenda antes de que éstas puedan ser aparentes.
 - Esto permite al gestor del proyecto tomar acciones correctivas antes de que el proyecto "entre en crisis"



A menos que realicemos un seguimiento del EV, realmente no tendremos idea de qué está pasando en nuestro proyecto!!!

EVA: Parámetros

- EVA básicamente compara tres tipos de información:
 - ¿Cuánto trabajo (del que se ha planificado) debería haberse completado hasta el momento? (Valor planificado: BCWS)
 - ¿Cuánto se ha gastado hasta el momento? (Coste real: ACWP)
 - ¿Cuál es el valor, en términos del coste de línea base, del trabajo realizado hasta el momento? (Valor acumulado: BCWP)

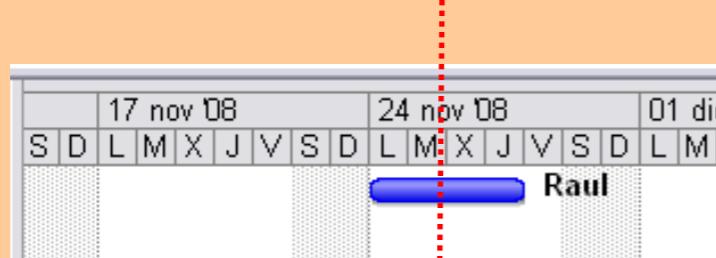


¡IMPORTANTE!

Un análisis de valor acumulado
SIEMPRE se hace tomando como
referencia un instante de tiempo
concreto del desarrollo del proyecto!!!

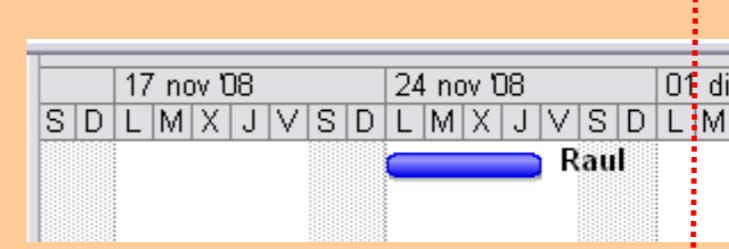
BCWS- Ejemplo

- Budgeted Cost of Work Scheduled
- Supongamos una tarea X con una duración de 4 días (desde el lun 24-Nov-2008 hasta el jue 27-Nov-2008), con un recurso asignado cuyo salario es 100€/dia
 - El coste previsto para la tarea, por lo tanto, es de 400€ (100×4)
 - Si calculamos BCWS el día 25-Nov-2008 tendrá un valor de 200€, puesto que la tarea debería estar medio realizada
 - Si calculamos BCWS el día 1-Dic-2008 tendrá un valor de 400€, puesto que la tarea debería estar completada



BCWS=200€

25-Nov-2008

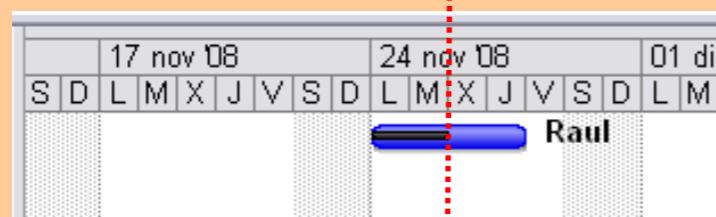


BCWS=400€

1-Dic-2008

BCWP- Ejemplo

- Budgeted Cost of Work Performed
- Supongamos que la tarea X anterior solamente se ha "completado" hasta la mitad (se ha hecho el 50% de lo que se había pedido):
 - El coste previsto para la tarea sigue siendo de 400€ (100×4)
 - Si calculamos BCWP el día 25-Nov-2008 tendrá un valor de 200€: el 100% del valor planificado
 - Si calculamos BCWP el día 1-Dic-2008 tendrá un valor de 200€: el 50% del valor planificado



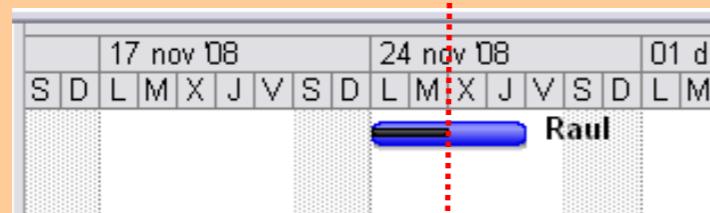
BCWS=200€
BCWP=200€ 25-Nov-2008



BCWS=400€
BCWP=200€ 1-Dic-2008

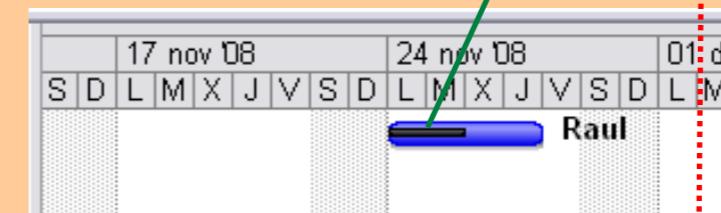
ACWP- Ejemplo

- Actual Cost of Work Performed
- Supongamos que el recurso asignado a la tarea X realmente recibe 110€/dia para esta tarea (en lugar de los 100€ inicialmente previstos):
 - El coste previsto para la tarea sigue siendo de 400€ (100×4)
 - Si calculamos ACWP el día 25-Nov-2008 tendrá un valor de 220€ (110×2)
 - Si calculamos ACWP el día 1-Dic-2008 tendrá un valor de 220€ (110×2)



BCWS=200€
BCWP=200€
ACWP=220€

25-Nov-2008



BCWS=400€
BCWP=200€A
CWP=220€

1-Dic-2008

¿Cómo funciona EVA?

- Indicadores de **PROGRESO**:
 - Schedule Variance (SV) = BCWP - BCWS
 - Schedule Performance Index (SPI) = BCWP / BCWS
 - Si BCWP > BCWS la tarea/proyecto va adelantada según la agenda planificada
 - P.ej. un SPI de 1,5 significa que sólo ha utilizado el 67% del tiempo planeado para completar una parte de la tarea en un determinado periodo de tiempo (BCWS = 0,67 BCWP)
- Indicadores de **PRODUCTIVIDAD**
 - Cost Variance (CV) = BCWP - ACWP
 - Cost Performance Index (CPI) = BCWP / ACWP
 - Si BCWP > ACWP la tarea/proyecto está gastando menos de lo planificado
 - P.ej. un CPI de 0,8 significa que se está gastando un 25% más de lo que estaba planificado (por cada euro presupuestado se está gastando 1,25€) (ACWP = 1,25 BCWP)
- Si tenemos una buena productividad, y un progreso lento: NOS FALTA GENTE!!!

Ejemplo EVA: BAC y BCWS

- BAC: Budget at Completion (Cantidad de trabajo planificado al final del proyecto/tarea)

	Task Name	BAC	BCWS
0	Manual Project	19,400.00	14,530.00
1	+ Weekly meetings	0.00	0.00
10	Content	10,450.00	8,900.00
11	Design structure	2,750.00	2,750.00
12	Write body text	6,150.00	6,150.00
13	Set page layouts	1,550.00	0.00
14	Exercises	8,950.00	5,630.00
15	Create exercises	5,300.00	5,300.00
16	Test exercises	1,650.00	330.00
17	Create contents & index	2,000.00	0.00
18	Manual completed	0.00	0.00

Se debería haber realizado hasta fecha un 75% del trabajo planificado del proyecto

Tarea que debería haberse completado (BCWS=BAC)

Todavía no ha terminado (BCWS < BAC)

Tarea que no ha comenzado (BCWS=0)

EVA: BCWS y BCWP

BCWP: Budget Cost of Work Performed (*Earned Value!!!!*)

- BCWP = Baseline Cost * Percent Complete

	Task Name	BCWS	BCWP
0	Manual Project	14,530.00	13,700.96
1	+ Weekly meetings	0.00	0.00
10	- Content	8,900.00	6,750.96
11	Design structure	2,750.00	2,750.00
12	Write body text	6,150.00	4,000.96
13	Set page layouts	0.00	0.00
14	- Exercises	5,630.00	6,950.00
15	Create exercises	5,300.00	5,300.00
16	Test exercises	330.00	1,650.00
17	Create contents & index	0.00	0.00
18	Manual completed	0.00	0.00

El proyecto lleva un **retraso** de trabajo de 829.04 (SV=-829.04), ha utilizado un 6,4% más del tiempo planificado (SPI=0,94)

Tarea con un **retraso** de 2149.04 sobre lo planificado hasta la fecha, ha consumido un 54% más del trabajo planificado (SPI =0,65)

Tarea con un **adelanto** de 1320 sobre lo planificado hasta la fecha, sólo ha utilizado el 20% del tiempo planificado para realizar la parte de tarea planificada (SPI =5)

- SI BCWP > BCWS: La tarea/proyecto va ADELANTADA según la agenda
- SI BCWP < BCWS: La tarea/proyecto va con RETRASO según la agenda

EVA: BCWP y ACWP

ACWP: Actual Cost of Work Performed

	Task Name	BCWP	ACWP
0	Manual Project	13,700.96	14,037.50
1	+ Weekly meetings	0.00	0.00
10	+ Content	6,750.96	8,047.50
11	Design structure	2,750.00	2,750.00
12	Write body text	4,000.96	5,297.50
13	Set page layouts	0.00	0.00
14	+ Exercises	6,950.00	5,990.00
15	Create exercises	5,300.00	4,660.00
16	Test exercises	1,650.00	1,330.00
17	Create contents & index	0.00	0.00
18	Manual completed	0.00	0.00

El proyecto va **por encima** de lo presupuestado ($CV=-336.54$), ha gastado un 2,4% más del presupuesto planificado ($CPI=0,98$)

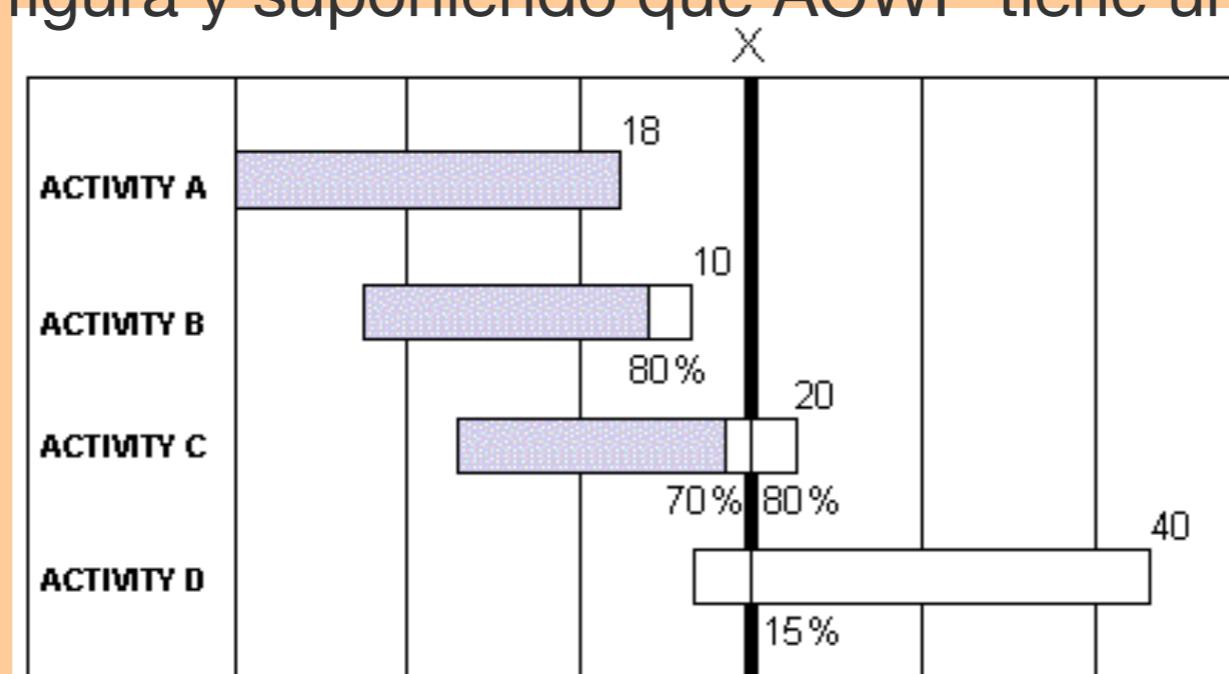
La tarea va **por encima** de lo presupuestado en 1296,54€, por cada euro presupuestado se ha gastado 1,33€ ($CPI =0,75$)

La tarea va **por debajo** de lo presupuestado en 320€, por cada euro presupuestado sólo se ha gastado 0,80€ ($CPI =1,24$)

- SI BCWP > ACWS: La tarea/proyecto está por DEBAJO de lo presupuestado
- SI BCWP < ACWS: La tarea/proyecto está por ENCIMA de lo presupuestado

EVA: Ejemplo

- Determina BCWS, BCWP y realiza un análisis EVA, a partir de la información de la siguiente figura y suponiendo que ACWP tiene un valor de 45\$ información:



- Valor planificado (BCWS) = $18 + 10 + 16 + 6 = \$50$

- Valor earned (BCWP) = $18 + 8 + 14 + 0 = \$40$

- Coste actual (ACWP) = 45\$

- SV = $40 - 50 = -\$10$

- SPI = $40 / 50 = 0.8$

- CV = $40 - 45 = -\$5$

- CPI = $40 / 45 = 0.89$

El proyecto necesita un 25% más del tiempo planificado para completar el proyecto

El proyecto está gastando $\approx \$1.13$ por cada \$1.00 de trabajo presupuestado que se completa

Ejercicio propuesto

- Realiza un análisis cuantitativo del progreso de las siguientes tareas:

BAC (CPF)	BCWS (CPTP)	BCWP (CPTR)	ACWP (CRTR)	SV (VP)	CV (VC)	SPI (IRP)	CPI (IRC)	Tareas
1000	1000	1000	1200					T1
1500	1500	1000	1500					T2
6000	5000	4000	4500					T3
1000	500	0	0					T4
1000	500	500	750					T5
2000	1000	1500	1500					T6

- Suponiendo que nuestro proyecto está formado exclusivamente por las tareas de la tabla anterior, ¿podríamos deducir que falta gente en el proyecto?

Ejercicio propuesto: guía para la solución

- A continuación mostramos el análisis para la Tarea T1:

BAC (CPF)	BCWS (CPTP)	BCWP (CPTR)	ACWP (CRTR)	SV (VP)	CV (VC)	SPI (IRP)	CPI (IRC)	Tareas
1000	1000	1000	1200	0	-200	1	0,8333	T1

La Tarea (1) debería haber terminado (ya que BAC = BCWS).

La tarea se está ejecutando según la agenda (ya que SV=0), de hecho ya ha terminado. Hemos gastado 200 de más (ya que CV es -200)

Pero hemos gastado 1,20 euros por cada 1 euro de coste presupuestado ($1/0,83=1,20$, o también $1200/1000$).

Ms Project y EVA

- BCWS = CPTP (Costo Presupuestado del Trabajo Programado)
- ACWP = CRTR (Costo Real del Trabajo Realizado)
- BCWP = CPTR (Costo Presupuestado del Trabajo Realizado)
- Tablas:
 - Valor acumulado
 - Indicadores de costo del valor acumulado

Datos de monitorización y control en Ms Project

- Con Ms Project, se utilizan cinco tipos de información para poder analizar el progreso al realizar el seguimiento de las tareas de un proyecto:
 - ✿ duración, trabajo, fecha de comienzo, fecha de fin, y costo
- Los cambios en cada uno de estos campos permiten evaluar el progreso:
 - ✿ **Planificado:** información programada de los campos anteriores. Un plan de línea base es el plan original que se guarda y se utiliza para monitorizar y controlar el progreso
 - ✿ **Programado:** información actual más actualizada de los campos anteriores (duración, trabajo, fechas de comienzo y fin, y costos PROGRAMADOS)
 - ✿ **Real:** información de lo que ha ocurrido realmente de los campos anteriores (duración, trabajo, fechas de comienzo y fin, y costos REALES)
 - ✿ **Restante:** información programada - información real (de trabajo, costo y duración)

Ms Project y seguimiento agenda

- Para hacer un seguimiento tenemos que:
 - ✿ Guardar una **LÍNEA BASE** del proyecto
 - ★ Proyecto→Herramientas→Establecer línea de base
 - ★ Los campos: duración, trabajo, comienzo, fin, costo se guardan como duración, trabajo,..., costo PREVISTOS
 - ✿ Establecer una **FECHA DE ESTADO**
 - ★ Proyecto→Información del proyecto →Fecha de estado
 - ✿ Introducir la **INFORMACIÓN REAL** del proyecto
 - ★ Herramientas→Seguimiento→Actualizar tareas
 - ★ Campos: duración, trabajo, comienzo, fin, costo REALES
 - ✿ Comparar el **PROGRESO** con una vista "Gantt de Seguimiento"
 - ★ Compara la programación de la línea base con la programación real

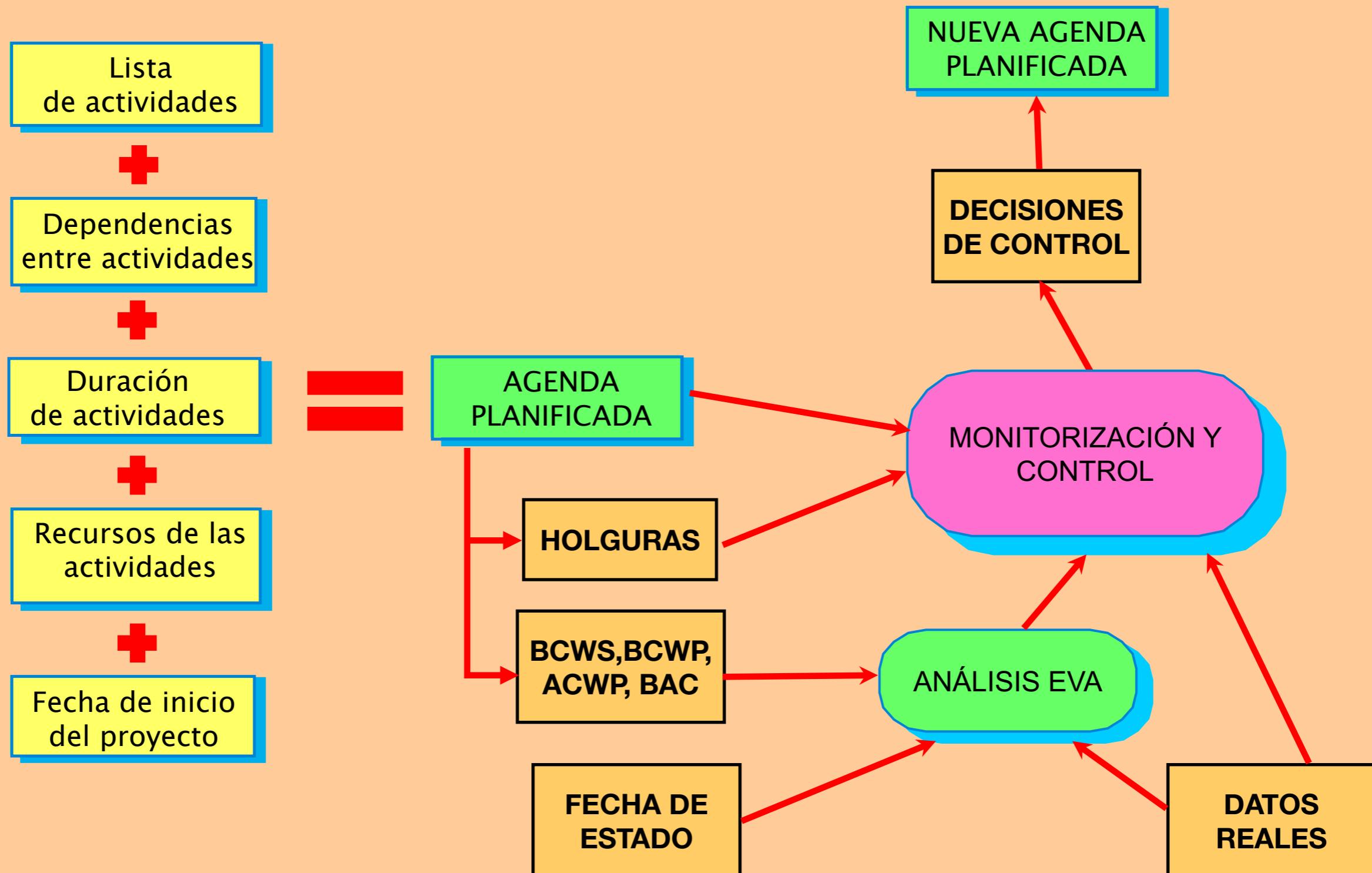
Control de una agenda: holguras

- Controlar una agenda consiste en realizar las acciones necesarias para que las discrepancias entre la agenda real y la planificada sean mínimas
- Las actividades críticas NO se pueden retrasar
- Las actividades no críticas pueden retrasarse siempre y cuando el retraso no supere su holgura total
 - Si una actividad consume toda (o parte) de su holgura total, puede afectar a la holgura total de la(s) actividad(es) siguiente(s)
 - La(s) actividad(es) siguiente(s) puede(n) convertirse en crítica(s)
 - Por ejemplo: A precede B
 - ★ $HT(A) = 3$ y $HL(A)=2$
 - ★ $HT(B) = 1$
 - ★ Si A se retrasa 3 unidades, B se convierte en crítica

Control de una agenda: EVA

- Como hemos visto, los indicadores de valor acumulado: variaciones (SV, CV) o ratios (SPI, CPI):
 - ✿ pueden ayudar a determinar si queda suficiente dinero en el presupuesto y si el proyecto acabará a tiempo
- Las variaciones:
 - ✿ positivas: pueden permitir reasignar dinero y recursos de tareas con variaciones positivas a otros con variaciones negativas
 - ✿ negativas: pueden indicar que quizá convenga aumentar el presupuesto o aceptar márgenes de beneficio reducidos
- Los ratios:
 - ✿ SPI es un indicador de progreso; CPI es un indicador de productividad
 - ✿ Si $CPI > 1$ y $SPI < 1$ necesitamos contratar a más gente
 - ✿ Si $CPI < 1$ puede que estemos haciendo trabajo no planificado, o que hayamos estimado mal

RESUMEN



Referencias

- PERT: holguras
 - <http://www.adeak.com/2010/01/what-is-float-or-slack-in-a-project-schedule/>
- Earned Value Analysis
 - <http://www.projectlearning.net/pdf/I2.1.pdf>
 - http://www.hyperhot.com/pm_cscs.htm

Test de autoevaluación modelo EVA

Puntos totales 10/10

Analizando los valores de los parámetros EVA de la imagen...

4. Dados los siguientes datos de un proyecto. Explica cuál es la situación del proyecto para cada una de las tareas. ¿La tarea 1 se ha terminado? ¿Cómo podemos solucionar los problemas del proyecto? (2,5p)

Tarea	BCWS	BCWP	ACWP
Tarea R	2800	1800	2600
Tarea 1	600	600	1000
Tarea 2	1200	0	0
Tarea 3	1000	1200	800
Tarea 4	0	0	800

✓ 1.- ¿Qué podemos decir del proyecto en su conjunto? *

2/2

- a) El proyecto va adelantado y su productividad es baja
- b) El proyecto va atrasado y su productividad es alta
- c) El proyecto va adelantado y su productividad es alta
- d) El proyecto va atrasado y su productividad es baja

✓ 2.- ¿Qué podemos decir de la tarea 1? *

2/2

- a) No podemos saber si ha terminado sin conocer su BAC
- b) Ha gastado menos de lo previsto
- c) Va adelantada respecto a la agenda
- d) Va atrasada respecto a la agenda

✓ 3.- ¿Qué podemos decir de la tarea 2? *

2/2

- a) No ha comenzado
- b) Debería haber comenzado
- c) Va retrasada
- d) Todas son ciertas

✓ 4.- ¿Qué podemos decir de la tarea 3? *

2/2

- a) No ha comenzado
- b) Va retrasada
- c) Está gastando menos de lo previsto
- d) Todas son falsas

✓ 5.- ¿Qué podemos decir de la tarea 4? *

2/2

- a) Debería haber comenzado y no lo ha hecho
- b) El valor BCWP es erróneo
- c) El valor ACWP es erróneo
- d) No debería haber comenzado pero sí lo ha hecho

Este formulario se creó en Universitat d'Alacant / Universidad de Alicante.

Google Formularios



Test de autoevaluación Propiedad Intelectual

Puntos totales 5/5

- ✓ Según la LPI, ¿Cuál de las siguientes figuras tiene un plazo de protección indefinido? * 1/1

- Marcas
- Secretos comerciales ✓
- Patentes
- Derechos de Autor

- ✓ ¿Cuál de las siguientes no es una causa legal para poder transmitir los derechos de propiedad intelectual? * 1/1

- mortis causa
- En virtud de relación laboral
- inter vivos
- En virtud de relación mercantil ✓

- ✓ ¿Cuál de los siguientes tipos de licencia no es propia del software libre? * 1/1

- GPL
- Creative Commons
- BSD
- OEM ✓

- ✓ El plazo de protección de los derechos de autor según la LPI es: * 1/1

- 20 años a partir de la fecha de registro
- 70 años a partir del fallecimiento del autor ✓
- 70 años a partir de la fecha de registro
- 20 años a partir del fallecimiento del autor

- ✓ ¿Cuál de los siguientes tipos de licencia no es propia del software privativo? * 1/1

- EULA
- Apache ✓
- ROK
- PKC

Este formulario se creó en Universitat d'Alacant / Universidad de Alicante.

Google Formularios

Test de autoevaluación Protección de datos

Puntos totales 5/5

- ✓ Según la LOPD ¿A partir de qué edad puede un menor expresar el consentimiento en términos de protección de datos? *

1/1

16 años

18 años

14 años ✓

El padre o tutor legal será el que lo haga en su nombre mientras forme parte de la unidad familiar

- ✓ Según el RGPD, el RAT (Registro de Actividades de Tratamiento) es obligatorio para: *

1/1

Cualquier empresa pública o privada

Solo para la Administración Pública

Empresas de 250 empleados o más ✓

Empresas de menos de 250 empleados y PYMES

- ✓ La LOPD no será de aplicación en el caso de: *

1/1

Materias consideradas clasificadas

Datos de personas fallecidas

Datos ya excluidos en el RGPD

Todas son ciertas ✓

- ✓ La normativa vigente en España en materia de protección de datos es: * 1/1

LOPD 15/1999 y RGPD UE de 2016

RGDP UE de 2018 y LOPD 3/2018

RGDP UE de 2016 y LOPD 3/2018 ✓

Actualmente en nuestro Estado solo nos afecta la LOPD 3/2018

- ✓ De los siguientes derechos regulados en la LOPD, ¿Cuál es un derecho digital? *

1/1

Derecho de supresión

Derecho a la intimidad en el ámbito laboral ✓

Derecho de rectificación

Derecho de oposición

Este formulario se creó en Universitat d'Alacant / Universidad de Alicante.

Google Formularios

Chapter

2

ITERATIVE, EVOLUTIONARY, AND AGILE

You should use iterative development only on projects that you want to succeed.

—Martin Fowler

Objectives

- Provide motivation for the content and order of the book.
- Define an iterative and agile process.
- Define fundamental concepts in the Unified Process.

Introduction

Iterative development lies at the heart of how OOA/D is best practiced and is presented in this book. Agile practices such as Agile Modeling are key to applying the UML in an effective way. This chapter introduces these subjects, and the Unified Process as a relatively popular *sample* iterative method.

What's Next?

Having introduced OOA/D, this chapter explores iterative development. The next introduces the case studies that are evolved throughout the book, across three iterations.



2 – ITERATIVE, EVOLUTIONARY, AND AGILE

Iterative and evolutionary development—contrasted with a sequential or “**waterfall**” lifecycle—involves early programming and testing of a partial system, in repeating cycles. It also normally assumes development starts before all the requirements are defined in detail; feedback is used to clarify and improve the evolving specifications.

We rely on short quick development steps, feedback, and adaptation to clarify the requirements and design. To contrast, waterfall values promoted big up-front speculative requirements and design steps before programming. Consistently, success/failure studies show that the waterfall is strongly associated with the highest failure rates for software projects and was historically promoted due to belief or hearsay rather than statistically significant evidence. Research demonstrates that iterative methods are associated with higher success and productivity rates, and lower defect levels.

2.1 What is the UP? Are Other Methods Complementary?

A **software development process** describes an approach to building, deploying, and possibly maintaining software. The **Unified Process** [JBR99] has emerged as a popular *iterative* software development process for building object-oriented systems. In particular, the **Rational Unified Process** or **RUP** [Kruchten00], a detailed refinement of the Unified Process, has been widely adopted.

Because the Unified Process (UP) is a relatively popular iterative process for projects using OOA/D, and because some process must be used to introduce the subject, the UP shapes the book’s structure. Also, since the UP is common and promotes widely recognized best practices, it’s useful for industry professionals to know it, and students entering the workforce to be aware of it.

test-driven development and refactoring p. 385

The UP is very flexible and open, and encourages including skillful practices from other iterative methods, such as from **Extreme Programming (XP)**, **Scrum**, and so forth. For example, XP’s **test-driven development**, **refactoring** and **continuous integration** practices can fit within a UP project. So can Scrum’s common project room (“war room”) and daily Scrum meeting practice. Introducing the UP is not meant to downplay the value of these other methods—quite the opposite. In my consulting work, I encourage clients to understand and adopt a blend of useful techniques from several methods, rather than a dogmatic “my method is better than your method” mentality.

The UP combines commonly accepted best practices, such as an iterative lifecycle and risk-driven development, into a cohesive and well-documented process description.

To summarize, this chapter includes an introduction to the UP for three reasons:

1. The UP is an *iterative* process. Iterative development influences how this

WHAT IS ITERATIVE AND EVOLUTIONARY DEVELOPMENT?

- book introduces OOA/D, and how it is best practiced.
2. UP practices provide an example *structure* for how to do—and thus how to explain—OOA/D. That structure shapes the book structure.
 3. The UP is flexible, and can be applied in a lightweight and *agile* approach that includes practices from other agile methods (such as XP or Scrum)—more on this later.

This book presents an introduction to an agile approach to the UP, but not complete coverage. It emphasizes common ideas and artifacts related to an introduction to OOA/D and requirements analysis.

What If I Don't Care About the UP?

The UP is used as an *example* process within which to explore iterative and evolutionary requirements analysis and OOA/D, since it's necessary to introduce the subject in the context of some process.

But the central ideas of this book—how to think and design with objects, apply UML, use design patterns, agile modeling, evolutionary requirements analysis, writing use cases, and so forth—are independent of any particular process, and apply to many modern iterative, evolutionary, and agile methods, such as Scrum, Lean Development, DSDM, Feature-Driven Development, Adaptive Software Development, and more.

2.2 What is Iterative and Evolutionary Development?

A key practice in both the UP and most other modern methods is **iterative development**. In this lifecycle approach, development is organized into a series of short, fixed-length (for example, three-week) mini-projects called **iterations**; the outcome of each is a tested, integrated, and executable *partial* system. Each iteration includes its own requirements analysis, design, implementation, and testing activities.

The iterative lifecycle is based on the successive enlargement and refinement of a system through multiple iterations, with cyclic feedback and adaptation as core drivers to converge upon a suitable system. The system grows incrementally over time, iteration by iteration, and thus this approach is also known as **iterative and incremental development** (see Figure 2.1). Because feedback and adaptation evolve the specifications and design, it is also known as **iterative and evolutionary development**.

Early iterative process ideas were known as spiral development and evolutionary development [Boehm88, Gilb88].

2 – ITERATIVE, EVOLUTIONARY, AND AGILE

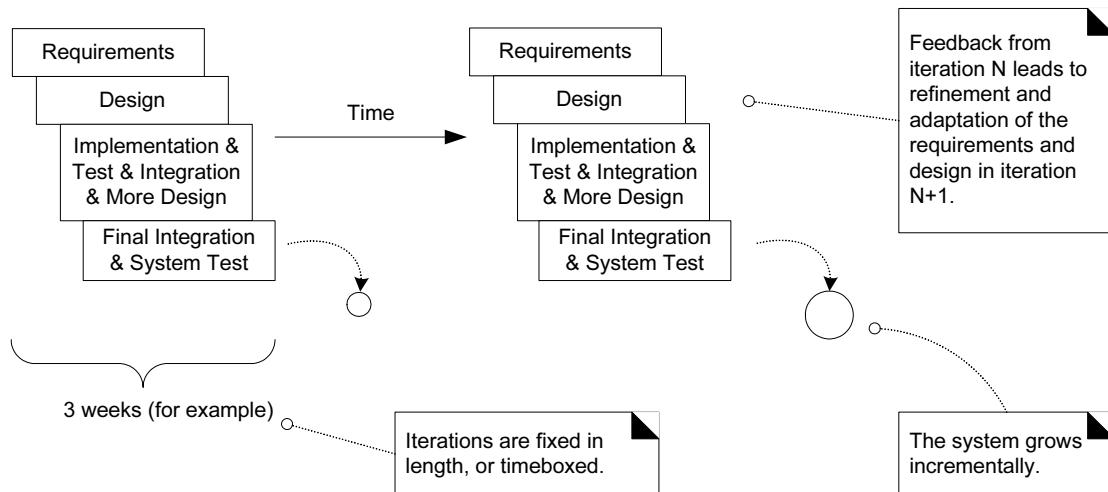


Figure 2.1 Iterative and evolutionary development.

Example

As an example (not a recipe), in a three-week iteration early in the project, perhaps one hour Monday morning is spent in a kickoff meeting with the team clarifying the tasks and goals of the iteration. Meanwhile, one person reverse-engineers the last iteration's code into UML diagrams (via a CASE tool), and prints and displays noteworthy diagrams. The team spends the remainder of Monday at whiteboards, working in pairs while agile modeling, sketching rough UML diagrams captured on digital cameras, and writing some pseudocode and design notes. The remaining days are spent on implementation, testing (unit, acceptance, usability, ...), further design, integration, and daily builds of the partial system. Other activities include demonstrations and evaluations with stakeholders, and planning for the next iteration.

Notice in this example that there is neither a rush to code, nor a long drawn-out design step that attempts to perfect all details of the design before programming. A “little” forethought regarding the design with visual modeling using rough and fast UML drawings is done; perhaps a half or full day by developers doing design work UML sketching in pairs at whiteboards.

The result of each iteration is an executable but incomplete system; it is not ready to deliver into production. The system may not be eligible for production deployment until after many iterations; for example, 10 or 15 iterations.

WHAT IS ITERATIVE AND EVOLUTIONARY DEVELOPMENT?

The output of an iteration is *not* an experimental or throw-away prototype, and iterative development is not prototyping. Rather, the output is a production-grade subset of the final system.

How to Handle Change on an Iterative Project?

The subtitle of one book that discusses iterative development is *Embrace Change* [Beck00]. This phrase is evocative of a key attitude of iterative development: Rather than fighting the inevitable change that occurs in software development by trying (unsuccessfully) to fully and correctly specify, freeze, and “sign off” on a frozen requirement set and design before implementation (in a “waterfall” process), iterative and evolutionary development is based on an attitude of embracing change and adaptation as unavoidable and indeed essential drivers.

This is not to say that iterative development and the UP encourage an uncontrolled and reactive “feature creep”-driven process. Subsequent chapters explore how the UP balances the need—on the one hand—to agree upon and stabilize a set of requirements, with—on the other hand—the reality of changing requirements, as stakeholders clarify their vision or the marketplace changes.

Each iteration involves choosing a small subset of the requirements, and quickly designing, implementing, and testing. In early iterations the choice of requirements and design may not be exactly what is ultimately desired. But the act of swiftly taking a small step, before all requirements are finalized, or the entire design is speculatively defined, leads to rapid feedback—feedback from the users, developers, and tests (such as load and usability tests).

And this early feedback is worth its weight in gold; rather than *speculating* on the complete, correct requirements or design, the team mines the feedback from realistic building and testing something for crucial practical insight and an opportunity to modify or adapt understanding of the requirements or design. End-users have a chance to quickly see a partial system and say, “Yes, that’s what I asked for, but now that I try it, what I really want is something slightly different.”¹ This “yes...but” process is not a sign of failure; rather, early and frequent structured cycles of “yes...buts” are a skillful way to make progress and discover what is of real value to the stakeholders. Yet this is not an endorsement of chaotic and reactive development in which developers continually change direction—a middle way is possible.

In addition to requirements clarification, activities such as load testing will prove if the partial design and implementation are on the right path, or if in the next iteration, a change in the core architecture is required. Better to resolve and *prove* the risky and critical design decisions early rather than late—and iterative development provides the mechanism for this.

Consequently, work proceeds through a series of structured build-feedback-adapt cycles. Not surprisingly, in early iterations the deviation from the “true

1. Or more likely, “You didn’t understand what I wanted!”

2 – ITERATIVE, EVOLUTIONARY, AND AGILE

path” of the system (in terms of its final requirements and design) will be larger than in later iterations. Over time, the system converges towards this path, as illustrated in Figure 2.2.

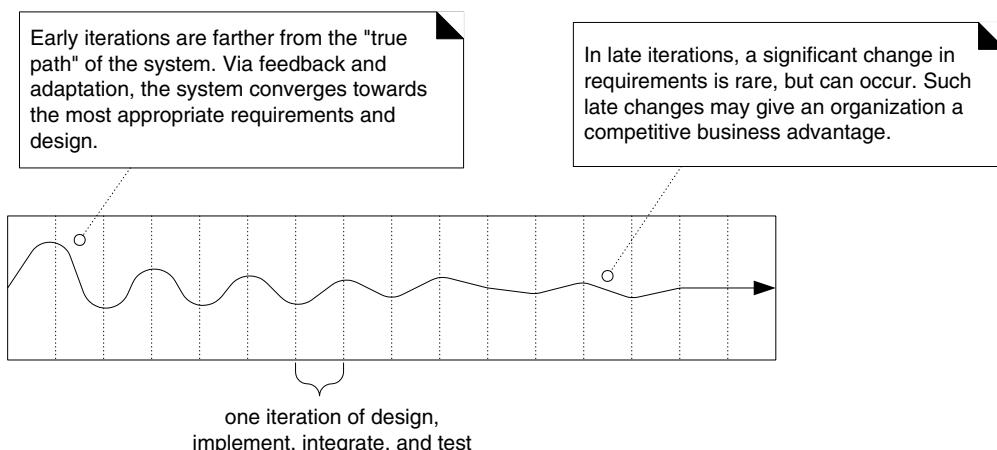


Figure 2.2 Iterative feedback and evolution leads towards the desired system. The requirements and design instability lowers over time.

Are There Benefits to Iterative Development?

Yes. Benefits include:

- less project failure, better productivity, and lower defect rates; shown by research into iterative and evolutionary methods
- early rather than late mitigation of high risks (technical, requirements, objectives, usability, and so forth)
- early visible progress
- early feedback, user engagement, and adaptation, leading to a refined system that more closely meets the real needs of the stakeholders
- managed complexity; the team is not overwhelmed by “analysis paralysis” or very long and complex steps
- the learning within an iteration can be methodically used to improve the development process itself, iteration by iteration

How Long Should an Iteration Be? What is Iteration Timeboxing?

Most iterative methods recommend an iteration length between two and six weeks. Small steps, rapid feedback, and adaptation are central ideas in iterative development; long iterations subvert the core motivation for iterative development and increase project risk. In only one week it is often difficult to complete

WHAT ABOUT THE WATERFALL LIFECYCLE?

sufficient work to get meaningful throughput and feedback; more than six weeks, and the complexity becomes rather overwhelming, and feedback is delayed. A very long timeboxed iteration misses the point of iterative development. Short is good.

A key idea is that iterations are **timeboxed**, or fixed in length. For example, if the next iteration is chosen to be three weeks long, then the partial system *must* be integrated, tested, and stabilized by the scheduled date—date slippage is illegal. If it seems that it will be difficult to meet the deadline, the recommended response is to de-scope—remove tasks or requirements from the iteration, and include them in a future iteration, rather than slip the completion date.

2.3 What About the Waterfall Lifecycle?

In a **waterfall** (or sequential) lifecycle process there is an attempt to define (in detail) all or most of the requirements before programming. And often, to create a thorough design (or set of models) before programming. Likewise, an attempt to define a “reliable” plan or schedule near the start—not that it will be.

Warning: Superimposing Waterfall on Iterative

If you find yourself on an “iterative” project where most of the requirements are written before development begins, or there is an attempt to create many thorough and detailed specifications or UML models and designs before programming, know that waterfall thinking has unfortunately afflicted the project. It is not a healthy iterative or UP project, regardless of claims.

Research (collected from many sources and summarized in [Larman03] and [LB03]) now shows conclusively that the 1960s and 1970s-era advice to apply the waterfall was—ironically—a poor practice for most software projects, rather than a skillful approach. It is strongly associated with high rates of failure, lower productivity, and higher defect rates (than iterative projects). On average, 45% of the features in waterfall requirements are never used, and early waterfall schedules and estimates vary up to 400% from the final actuals.

*feature use
research p. 56*

In hindsight, we now know that waterfall advice was based on *speculation* and *hearsay*, rather than evidence-based practices. In contrast, iterative and evolutionary practices are backed by evidence—studies show they are less failure prone, and associated with better productivity and defect rates.

Guideline: Don’t Let Waterfall Thinking Invade an Iterative or UP Project

I need to emphasize that “waterfall thinking” often incorrectly still invades a so-called iterative or UP project. Ideas such as “let’s write all the use cases before starting to program” or “let’s do many detailed OO models in UML before starting to program” are examples of unhealthy waterfall thinking incorrectly super-

2 – ITERATIVE, EVOLUTIONARY, AND AGILE

imposed on the UP. The creators of the UP cite this misunderstanding—big up-front analysis and modeling—as a key reason for its failed adoption [KL01].

Why is the Waterfall so Failure-Prone?

There isn't one simple answer to why the waterfall is so failure-prone, but it is strongly related to a key false assumption underlying many failed software projects—that the specifications are predictable and stable and can be correctly defined at the start, with low change rates. This turns out to be far from accurate—and a costly misunderstanding. A study by Boehm and Papaccio showed that a typical software project experienced a 25% change in requirements [BP88]. And this trend was corroborated in another major study of thousands of software projects, with change rates that go even higher—35% to 50% for large projects—as illustrated in Figure 2.3 [Jones97].

These are *extremely* high change rates. What this data shows—as any experienced developer or manager is painfully aware—is that software development is (on average) a domain of high change and instability—also known as the domain of **new product development**. Software is not usually a domain of predictable or mass manufacturing—low-change areas where it is possible and efficient to define all the stable specifications and reliable plans near the start.

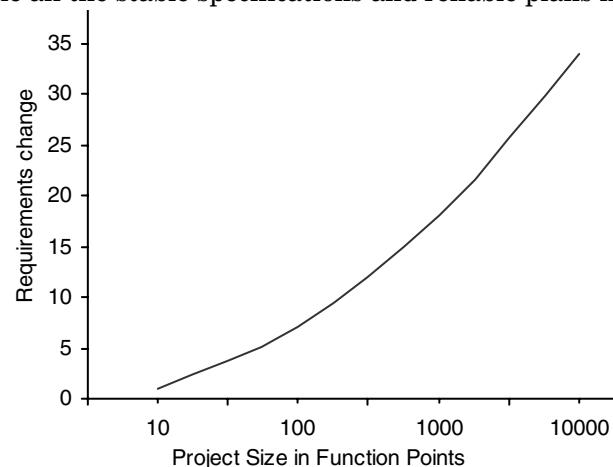


Figure 2.3 Percentage of change on software projects of varying sizes.

Thus, any analysis, modeling, development, or management practice based on the assumption that things are long-term stable (i.e., the waterfall) is fundamentally flawed. *Change* is the constant on software projects. Iterative and evolutionary methods assume and embrace change and adaptation of *partial and evolving* specifications, models, and plans based on feedback.

HOW TO DO ITERATIVE AND EVOLUTIONARY ANALYSIS AND DESIGN?

The Need for Feedback and Adaptation

In complex, changing systems (such as most software projects) feedback and adaptation are key ingredients for success.

- Feedback from early development, programmers trying to read specifications, and client demos to refine the requirements.
- Feedback from tests and developers to refine the design or models.
- Feedback from the progress of the team tackling early features to refine the schedule and estimates.
- Feedback from the client and marketplace to re-prioritize the features to tackle in the next iteration.

2.4 How to do Iterative and Evolutionary Analysis and Design?

This introduction may have given the impression that there is no value in analysis and design before programming, but that is a misunderstanding as extreme as thinking that “complete” up-front analysis is skillful. There is a middle way. Here’s a short *example* (not a recipe) of how it can work on a well-run UP project. This assumes there will ultimately be 20 iterations on the project before delivery:

1. Before iteration-1, hold the first timeboxed requirements workshop, such as exactly two days. Business and development people (including the chief architect) are present.
 - On the morning of day one, do high-level requirements analysis, such as identifying just the names of the use cases and features, and key non-functional requirements. The analysis will not be perfect.
 - Ask the chief architect and business people to pick 10% from this high-level list (such as 10% of the 30 use case names) that have a blending of these three qualities: 1) architecturally significant (if implemented, we are forced to design, build, and test the core architecture), 2) high business value (features business really cares about), and 3) high risk (such as “be able to handle 500 concurrent transactions”). Perhaps three use cases are thus identified: UC2, UC11, UC14.
 - For the remaining 1.5 days, do intensive detailed analysis of the functional and non-functional requirements for these three use cases. When finished, 10% are deeply analyzed, and 90% are only high-level.
2. Before iteration-1, hold an iteration planning meeting in which a subset from UC2, UC11, and UC14 are chosen to design, build, and test within a specified time (for example, four-week timeboxed iteration). Note that not

2 – ITERATIVE, EVOLUTIONARY, AND AGILE

all of these three use cases can be built in iteration-1, as they will contain too much work. After choosing the specific subset goals, break them down into a set of more detailed iteration tasks, with help from the development team.

3. Do iteration-1 over three or four weeks (pick the timebox, and stick to it).
 - On the first two days, developers and others do modeling and design work in pairs, sketching UML-ish diagrams at many whiteboards (along with sketching other kinds of models) in a common war room, coached and guided by the chief architect.
 - Then the developers take off their “modeling hats” and put on their “programming hats.” They start programming, testing, and integrating their work continuously over the remaining weeks, using the modeling sketches as a starting point of inspiration, knowing that the models are partial and often vague.
 - Much testing occurs: unit, acceptance, load, usability, and so forth.
 - One week before the end, ask the team if the original iteration goals can be met; if not, de-scope the iteration, putting secondary goals back on the “to do” list.
 - On Tuesday of the last week there’s a code freeze; all code must be checked in, integrated, and tested to create the iteration baseline.
 - On Wednesday morning, demo the partial system to external stakeholders, to show early visible progress. Feedback is requested.
4. Do the second requirements workshop near the end of iteration-1, such as on the last Wednesday and Thursday. Review and refine all the material from the last workshop. Then pick another 10% or 15% of the use cases that are architecturally significant and of high business value, and analyze them in detail for one or two days. When finished, perhaps 25% of the use cases and non-functional requirements will be written in detail. They won’t be perfect.
5. On Friday morning, hold another iteration planning meeting for the next iteration.
6. Do iteration-2; similar steps.
7. Repeat, for four iterations and five requirements workshops, so that at the end of iteration-4, perhaps 80% or 90% of the requirements have been written in detail, but only 10% of the system has been implemented.
 - Note that this large, detailed set of requirements is based on feedback and evolution, and is thus of much higher quality than purely speculative waterfall specifications.
8. We are perhaps only 20% into the duration of the overall project. In UP terms, this is the end of the **elaboration phase**. At this point, estimate in

WHAT IS RISK-DRIVEN AND CLIENT-DRIVEN ITERATIVE PLANNING?

detail the effort and time for the refined, high-quality requirements. Because of the significant realistic investigation, feedback, and early programming and testing, the estimates of what can be done and how long it will take are much more reliable.

9. After this point, requirements workshops are unlikely; the requirements are stabilized—though never completely frozen. Continue in a series of three-week iterations, choosing the next step of work adaptively in each iteration planning meeting on the final Friday, re-asking the question each iteration, “Given what we know today, what are the most critical technical and business features we should do in the next three weeks?”

Figure 2.5 illustrates the approach for a 20-iteration project.

In this way, after a few iterations of early exploratory development, there comes a point when the team can more reliably answer “what, how much, when.”

2.5 What is Risk-Driven and Client-Driven Iterative Planning?

The UP (and most new methods) encourage a combination of **risk-driven** and **client-driven** iterative planning. This means that the goals of the early iterations are chosen to 1) identify and drive down the highest risks, and 2) build visible features that the client cares most about.

Risk-driven iterative development includes more specifically the practice of **architecture-centric** iterative development, meaning that early iterations focus on building, testing, and stabilizing the core architecture. Why? Because not having a solid architecture is a common high risk.

Book Iterations vs. Real Project Iterations

Iteration-1 of the case studies in this book is driven by learning goals rather than true project goals. Therefore, iteration-1 is not architecture-centric or risk-driven. On a real project, we would tackle difficult and risky things first. But in the context of a book helping people learn fundamental OOA/D and UML, that's impractical—we need to start with problems illustrating basic principles, not the most difficult topics and problems.

2.6 What are Agile Methods and Attitudes?

Agile development methods usually apply timeboxed iterative and evolutionary development, employ adaptive planning, promote incremental delivery, and include other values and practices that encourage *agility*—rapid and flexible response to change.

2 – ITERATIVE, EVOLUTIONARY, AND AGILE

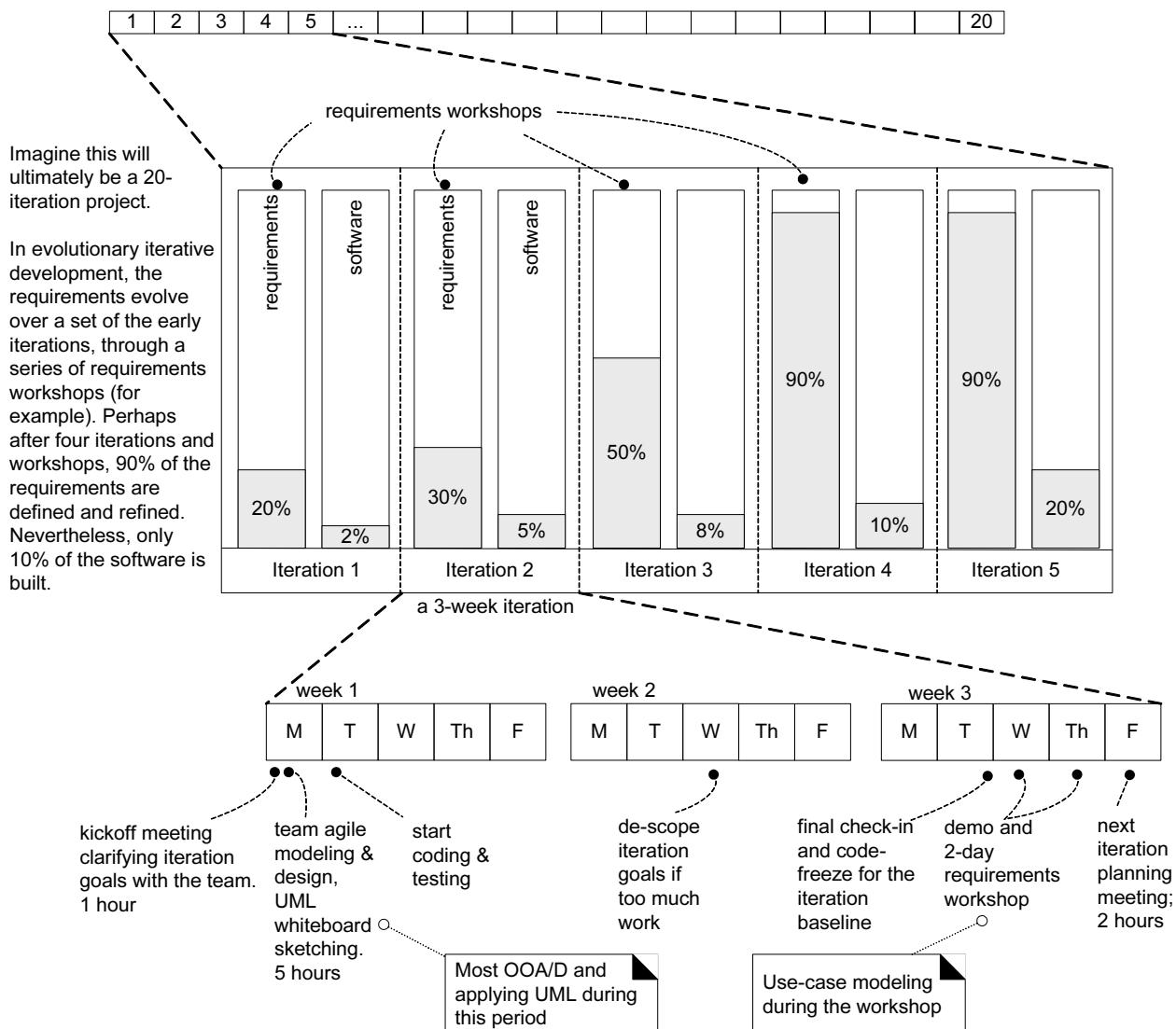


Figure 2.4 Evolutionary analysis and design—the majority in early iterations.

It is not possible to exactly define **agile methods**, as specific practices vary widely. However, short timeboxed iterations with evolutionary refinement of plans, requirements, and design is a basic practice the methods share. In addition, they promote practices and principles that reflect an agile sensibility of simplicity, lightness, communication, self-organizing teams, and more.

WHAT ARE AGILE METHODS AND ATTITUDES?

TDD p. 385

Example practices from the Scrum agile method include a *common project workroom* and *self-organizing teams* that coordinate through a daily stand-up meeting with four special questions each member answers. Example practices from the Extreme Programming (XP) method include *programming in pairs* and **test-driven development**.

Any iterative method, including the UP, can be applied in an agile spirit. And the UP itself is flexible, encouraging a “whatever works” attitude to include practices from Scrum, XP, and other methods.

The Agile Manifesto and Principles

The Agile Manifesto

<i>Individuals and interactions</i>	<i>over processes and tools</i>
<i>Working software</i>	<i>over comprehensive documentation</i>
<i>Customer collaboration</i>	<i>over contract negotiation</i>
<i>Responding to change</i>	<i>over following a plan</i>

The Agile Principles

- 1. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
- 2. Welcome changing requirements, even late in development. Agile processes harness change for the customer’s competitive advantage.
- 3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter time scale.
- 4. Business people and developers must work together daily throughout the project.
- 5. Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
- 6. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
- 7. Working software is the primary measure of progress.
- 8. Agile processes promote sustainable development.
- 9. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
- 10. Continuous attention to technical excellence and good design enhances agility.
- 11. Simplicity—the art of maximizing the amount of work not done—is essential.
- 12. The best architectures, requirements, and designs emerge from self-organizing teams.
- 13. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

In 2001 a group interested in iterative and agile methods (coining the term) met to find common ground. Out of this came the Agile Alliance (www.agilealliance.com) with a manifesto and statement of principles to capture the spirit of agile methods.

2 – ITERATIVE, EVOLUTIONARY, AND AGILE

2.7 What is Agile Modeling?

more on agile modeling p. 214 Experienced analysts and modelers know the *secret of modeling*:

The purpose of modeling (sketching UML, ...) is primarily to *understand*, not to document.

That is, the very act of modeling can and should provide a way to better understand the problem or solution space. From this viewpoint, the purpose of “doing UML” (which should really mean “doing OOA/D”) is *not* for a designer to create many detailed UML diagrams that are handed off to a programmer (which is a very un-agile and waterfall-oriented mindset), but rather to quickly explore (more quickly than with code) alternatives and the path to a good OO design.

This view, consistent with agile methods, has been called **agile modeling** in the book (amazingly called) *Agile Modeling* [Ambler02]. It implies a number of practices and values, including:

- Adopting an agile method does not mean avoiding any modeling; that’s a misunderstanding. Many agile methods, such as Feature-Driven Development, DSDM, and Scrum, normally include significant modeling sessions. Even the XP founders, from perhaps the most well-known agile method with the least emphasis on modeling, endorsed agile modeling as described by Ambler—and practiced by many modelers over the years.
- The purpose of modeling and models is primarily to support understanding and communication, not documentation.
- Don’t model or apply the UML to all or most of the software design. Defer simple or straightforward design problems until programming—solve them while programming and testing. Model and apply the UML for the smaller percentage of unusual, difficult, tricky parts of the design space.
- Use the simplest tool possible. Prefer “low energy” creativity-enhancing simple tools that support rapid input and change. Also, choose tools that support large visual spaces. For example, prefer sketching UML on whiteboards, and capturing the diagrams with a digital camera.²
 - This doesn’t mean UML CASE tools or word processors can’t be used or have no value, but especially for the creative work of discovery, sketching on whiteboards supports quick creative flow and change. The key rule is ease and agility, whatever the technology.

2. Two whiteboard sketching tips: **One:** If you don’t have enough whiteboards (and you should have many large ones), an alternative is “whiteboard” plastic cling sheets which cling to walls (with a static charge) to create whiteboards. The main product in North America is Avery Write-On Cling Sheets; the main product in Europe is Legamaster Magic-Chart. **Two:** Digital photos of whiteboard images are often poor (due to reflection). Don’t use a flash, but use a software “whiteboard image clean up” application to improve the images, if you need to clean them (as I did for this book).

WHAT IS AN AGILE UP?

- Don't model alone, model in pairs (or triads) at the whiteboard, in the awareness that the purpose of modeling is to discover, understand, and share that understanding. Rotate the pen sketching across the members so that all participate.
- Create models in parallel. For example, on one whiteboard start sketching a dynamic-view UML interaction diagram, and on another whiteboard, start sketching the complementary static-view UML class diagram. Develop the two models (two views) together, switching back and forth.
- Use "good enough" simple notation while sketching with a pen on whiteboards. Exact UML details aren't important, as long as the modelers understand each other. Stick to simple, frequently used UML elements.
- Know that all models will be inaccurate, and the final code or design different—sometimes dramatically different—than the model. Only tested code demonstrates the true design; all prior diagrams are incomplete hints, best treated lightly as throw-away explorations.
- Developers themselves should do the OO design modeling, for themselves, not to create diagrams that are given to other programmers to implement—an example of un-agile waterfall-oriented practices.

Agile Modeling in this Book: Why the Snapshots of UML Sketches?

UML-sketchn modeling on whiteboards is a practice I—and many developers—have enthusiastically coached and practiced for years. Yet most of the UML diagrams in this book give the impression I don't work that way, because they've been drawn neatly with a tool, for readability. To balance that impression the book occasionally includes digital snapshot pictures of whiteboard UML sketches. It sacrifices legibility but reminds that agile modeling is useful and is the actual practice behind the case studies.

For example, Figure 2.5 is an unedited UML sketch created on a project I was coaching. It took about 20 minutes to draw, with four developers standing around. We needed to understand the inter-system collaboration. The act of drawing it together provided a context to contribute unique insights and reach shared understanding. This captures the feel of how agile modelers apply the UML.

2.8 What is an Agile UP?

The UP was not meant by its creators to be heavy or un-agile, although its large *optional* set of activities and artifacts have understandably led some to that impression. Rather, it was meant to be adopted and applied in the spirit of adaptability and lightness—an **agile UP**. Some examples of how this applies:

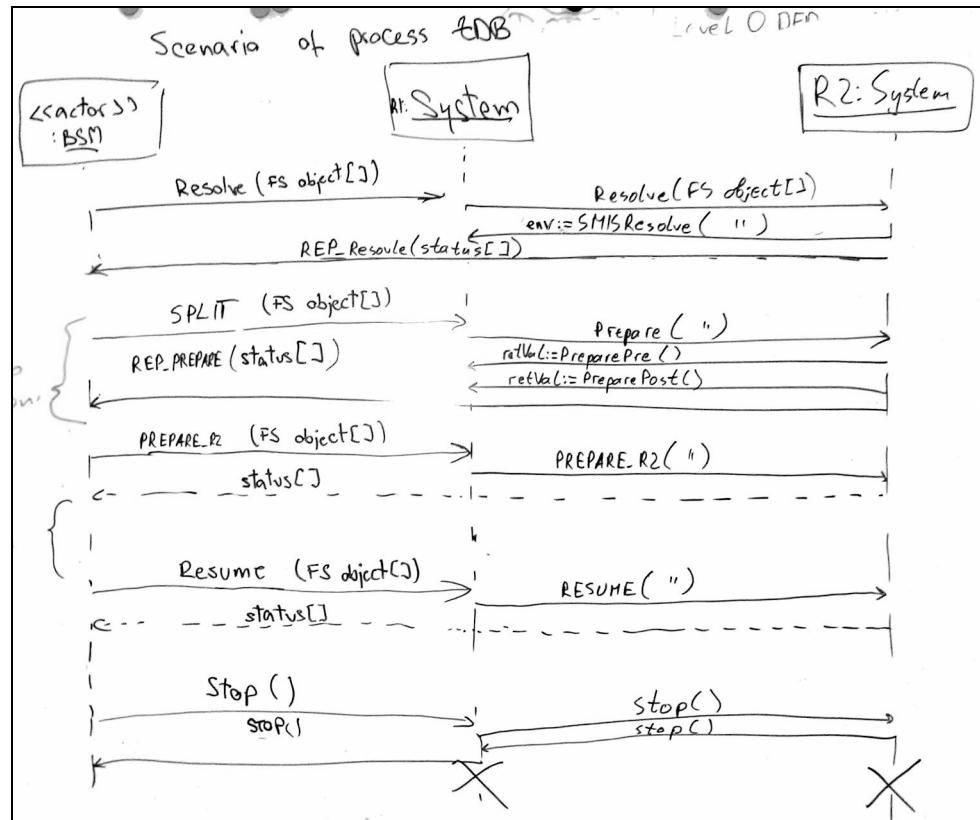


Figure 2.5 A UML sketch of a sequence diagram from a project.

customizing UP
p. 37

- Prefer a *small* set of UP activities and artifacts. Some projects will benefit more than others, but, in general, keep it simple. Remember that all UP artifacts are optional, and avoid creating them unless they add value. Focus on early programming, not early documenting.

evolutionary A&D
p. 25

- Since the UP is iterative and evolutionary, requirements and designs are not completed before implementation. They adaptively emerge through a series of iterations, based on feedback.

agile models p. 30

- Apply the UML with agile modeling practices.

agile PM p. 673

- There isn't a *detailed* plan for the entire project. There is a high-level plan (called the **Phase Plan**) that estimates the project end date and other major milestones, but it does not detail the fine-grained steps to those milestones. A detailed plan (called the **Iteration Plan**) only plans with greater detail one iteration in advance. Detailed planning is done adaptively from iteration to iteration.

ARE THERE OTHER CRITICAL UP PRACTICES?

The case studies emphasize a relatively small number of artifacts, and iterative development, in the spirit of an agile UP.

2.9 Are There Other Critical UP Practices?

The central idea to appreciate and practice in the UP is short timeboxed iterative, evolutionary, and adaptive development. Some additional best practices and key concepts in the UP:

- tackle high-risk and high-value issues in early iterations
- continuously engage users for evaluation, feedback, and requirements
- build a cohesive, core architecture in early iterations
- continuously verify quality; test early, often, and realistically
- apply use cases where appropriate
- do some visual modeling (with the UML)
- carefully manage requirements
- practice change request and configuration management

2.10 What are the UP Phases?

A UP project organizes the work and iterations across four major phases:

1. **Inception**—approximate vision, business case, scope, vague estimates.
2. **Elaboration**—refined vision, iterative implementation of the core architecture, resolution of high risks, identification of most requirements and scope, more realistic estimates.
3. **Construction**—iterative implementation of the remaining lower risk and easier elements, and preparation for deployment.
4. **Transition**—beta tests, deployment.

These phases are more fully defined in subsequent chapters.

This is *not* the old “waterfall” or sequential lifecycle of first defining all the requirements, and then doing all or most of the design.

Inception is not a requirements phase; rather, it is a feasibility phase, where just enough investigation is done to support a decision to continue or stop.

Similarly, elaboration is not the requirements or design phase; rather, it is a phase where the core architecture is iteratively implemented, and high-risk issues are mitigated.

Figure 2.6 illustrates common schedule-oriented terms in the UP. Notice that

2 – ITERATIVE, EVOLUTIONARY, AND AGILE

one development cycle (which ends in the release of a system into production) is composed of many iterations.

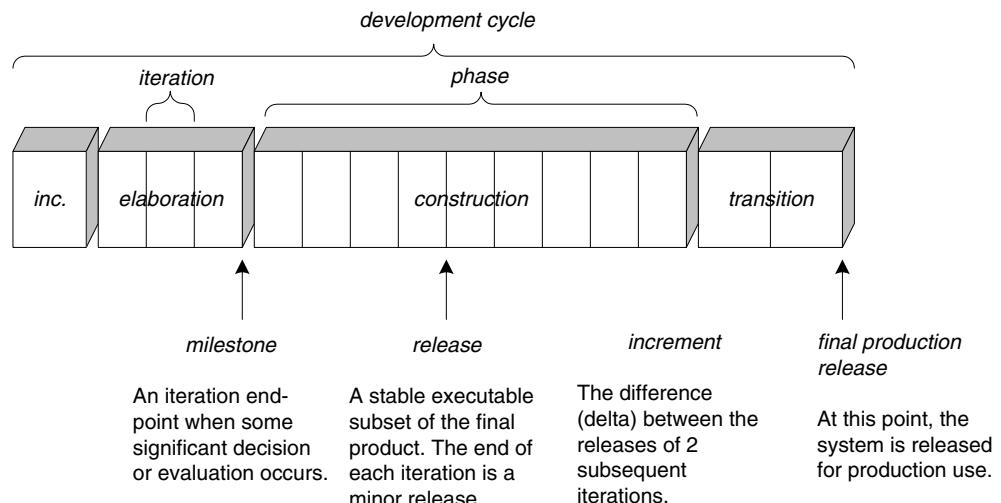


Figure 2.6 Schedule-oriented terms in the UP.

2.11 What are the UP Disciplines?

The UP describes work activities, such as writing a use case, within **disciplines**—a set of activities (and related artifacts) in one subject area, such as the activities within requirements analysis. In the UP, an **artifact** is the general term for any work product: code, Web graphics, database schema, text documents, diagrams, models, and so on.

There are several disciplines in the UP; this book focuses on some artifacts in the following three:

- **Business Modeling**—The Domain Model artifact, to visualize noteworthy concepts in the application domain.
- **Requirements**—The Use-Case Model and Supplementary Specification artifacts to capture functional and non-functional requirements.
- **Design**—The Design Model artifact, to design the software objects.

A longer list of UP disciplines is shown in Figure 2.7.

In the UP, **Implementation** means programming and building the system, not deploying it. The **Environment** discipline refers to establishing the tools and customizing the process for the project—that is, setting up the tool and process environment.

WHAT ARE THE UP DISCIPLINES?

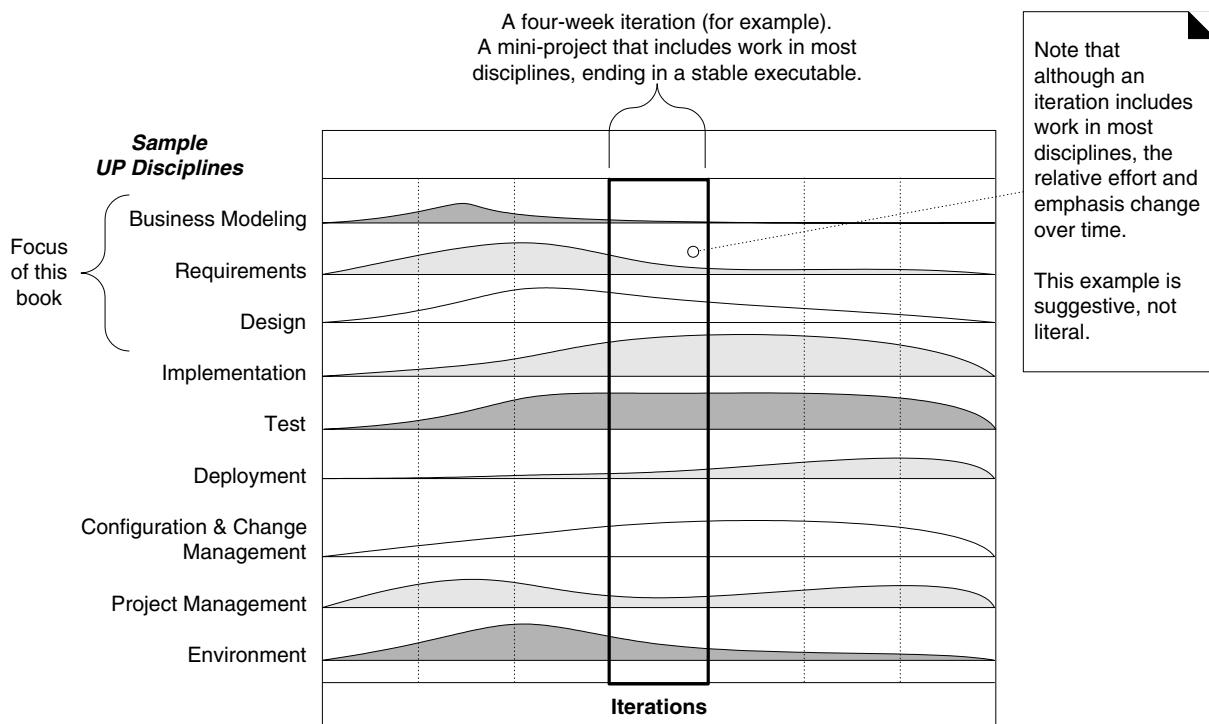


Figure 2.7 UP disciplines.

What is the Relationship Between the Disciplines and Phases?

As illustrated in Figure 2.7, during one iteration work goes on in most or all disciplines. However, the relative effort across these disciplines changes over time. Early iterations naturally tend to apply greater relative emphasis to requirements and design, and later ones less so, as the requirements and core design stabilize through a process of feedback and adaptation.

Relating this to the UP phases (inception, elaboration, ...), Figure 2.8 illustrates the changing relative effort with respect to the phases; please note these are suggestive, not literal. In elaboration, for example, the iterations tend to have a relatively high level of requirements and design work, although definitely some implementation as well. During construction, the emphasis is heavier on implementation and lighter on requirements analysis.

2 – ITERATIVE, EVOLUTIONARY, AND AGILE

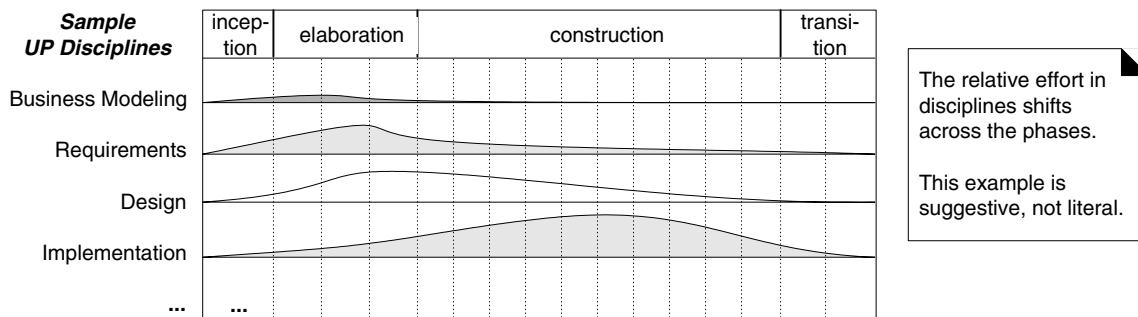


Figure 2.8 Disciplines and phases.

How is the Book Structure Influenced by UP Phases and Disciplines?

With respect to the phases and disciplines, what is the focus of the case studies?

The case studies emphasize the inception and elaboration phase. They focus on some artifacts in the Business Modeling, Requirements, and Design disciplines, as this is where requirements analysis, OOA/D, patterns, and the UML are primarily applied.

The earlier chapters introduce activities in inception; later chapters explore several iterations in elaboration. The following list and Figure 2.9 describe the organization with respect to the UP phases.

1. The inception phase chapters introduce the basics of requirements analysis.
2. Iteration 1 introduces fundamental OOA/D and assignment of responsibilities to objects.
3. Iteration 2 focuses on object design, especially on introducing some high-use “design patterns.”
4. Iteration 3 introduces a variety of subjects, such as architectural analysis and framework design.

HOW TO CUSTOMIZE THE PROCESS? THE UP DEVELOPMENT CASE

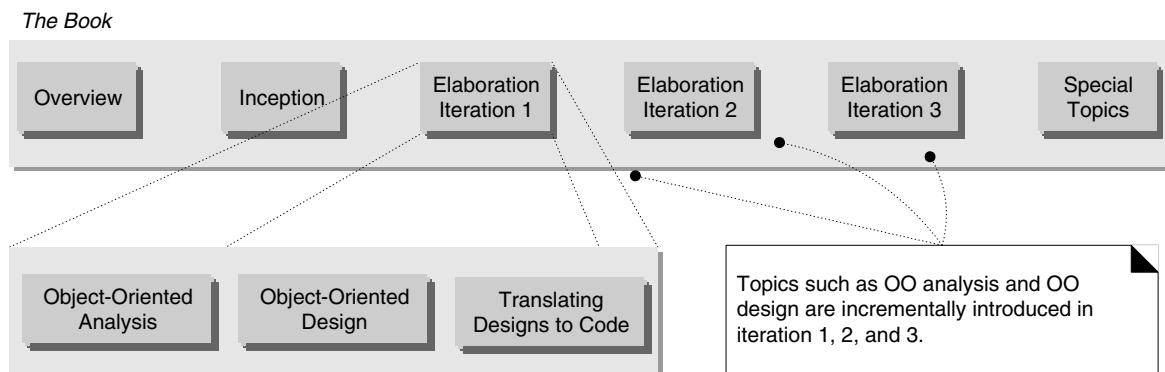


Figure 2.9 Book organization is related to the UP phases and iterations.

2.12 How to Customize the Process? The UP Development Case

Are There Optional Artifacts or Practices in the UP?

Yes! Almost everything is optional. That said, some UP practices and principles are invariant, such as iterative and risk-driven development, and continuous verification of quality.

However, a key insight into the UP is that all activities and artifacts (models, diagrams, documents, ...) are *optional*—well, maybe not the code!

Analogy

The set of possible artifacts described in the UP should be viewed like a set of medicines in a pharmacy. Just as one does not indiscriminately take many medicines, but matches the choice to the ailment, likewise on a UP project, a team should select a small subset of artifacts that address its particular problems and needs. In general, focus on a *small* set of artifacts that demonstrate high practical value.

Definition: What is the Development Case?

The choice of practices and UP artifacts for a project may be written up in a short document called the **Development Case** (an artifact in the Environment discipline). For example, Table 2.1 could be the Development Case for the “Next-Gen Project” case study explored in this book.

Subsequent chapters describe the creation of some of these artifacts, including the Domain Model, Use-Case Model, and Design Model.

2 – ITERATIVE, EVOLUTIONARY, AND AGILE

The example practices and artifacts presented in this case study are by no means sufficient for, or suitable for, all projects. For example, a machine control system may benefit from many state diagrams. A Web-based e-commerce system may require a focus on user interface prototypes. A “green-field” new development project has very different design artifact needs than a systems integration project.

Discipline	Practice	Artifact Iteration→	Incep. I1	Elab. E1..En	Const. C1..Cn	Trans. T1..T2
Business Modeling	agile modeling req. workshop	Domain Model		s		
Requirements	req. workshop vision box exercise dot voting	Use-Case Model	s	r		
		Vision	s	r		
		Supplementary Specification	s	r		
		Glossary	s	r		
Design	agile modeling test-driven dev.	Design Model		s	r	
		SW Architecture Document		s		
		Data Model		s	r	
Implementation	test-driven dev. pair programming continuous integration coding standards	...				
Project Management	agile PM daily Scrum meeting	...				
...						

Table 2.1 Sample Development Case. s - start; r - refine

2.13 You Know You Didn’t Understand Iterative Development or the UP When...

Here are some signs that you have not understood what it means to adopt iterative development and the UP in a healthy agile spirit.

- You try to define most of the requirements before starting design or implementation. Similarly, you try to define most of the design before starting implementation; you try to fully define and commit to an architecture before iterative programming and testing.
- You spend days or weeks in UML modeling before programming, or you think UML diagramming and design activities are a time to fully and accu-

HISTORY

rately define designs and models in great detail. And you regard programming as a simple mechanical translation of these into code.

- You think that inception = requirements, elaboration = design, and construction = implementation (that is, superimposing the waterfall on the UP).
- You think that the purpose of elaboration is to fully and carefully define models, which are translated into code during construction.
- You believe that a suitable iteration length is three months long, rather than three weeks long.
- You think that adopting the UP means to do many of the possible activities and create many documents, and you think of or experience the UP as a formal, fussy process with many steps to be followed.
- You try to plan a project in detail from start to finish; you try to speculatively predict all the iterations, and what should happen in each one.

2.14 History

For the full story and citations, see “Iterative and Incremental Development: A Brief History” (*IEEE Computer*, June 2003, Larman and Basili), and also [Larman03]. Iterative methods go back farther than many realize. In the late 1950s, evolutionary, iterative, and incremental development (IID), rather than the waterfall, was applied on the Mercury space project, and in the early 1960s, on the Trident submarine project, in addition to many other large systems. The first published paper promoting iterative rather than waterfall development was published in 1968 at the IBM T.J. Watson Research Center.

IID was used on many large defense and aerospace projects in the 1970s, including the USA Space Shuttle flight control software (built in 17 iterations averaging about four weeks each). A dominant software engineering thought-leader of the 1970s, Harlan Mills, wrote at that time about the failure of the waterfall for software projects, and the need for IID. Tom Gilb, a private consultant, created and published the IID Evo method in the 1970s, arguably the first fully-formed iterative method. The USA Department of Defense had adopted a waterfall standard in the late 1970s and early 1980s (DoD-2167); by the late 1980s they were experiencing significant failure (estimates of at least 50% of software projects cancelled or unusable), and so it was dropped, and eventually (starting in 1987) replaced by IID method standards—although the legacy of waterfall influence still confuses some DoD projects.

Also in the 1980s, Dr. Frederick Brooks (of *Mythical Man-Month* fame), a major software engineering thoughtleader of that decade, wrote and spoke about the shortcomings of the waterfall and the need to instead use IID methods. Another 1980s milestone was the publication of the spiral model risk-driven IID method by Dr. Barry Boehm, citing the high risk of failure when the waterfall was applied.

2 – ITERATIVE, EVOLUTIONARY, AND AGILE

By the early 1990s, IID was widely recognized as the successor to the waterfall, and there was a flowering of iterative and evolutionary methods: UP, DSDM, Scrum, XP, and many more.

2.15 Recommended Resources

A readable introduction to the UP and its refinement in the RUP is *The Rational Unified Process—An Introduction* by Philippe Kruchten. Also excellent is *The Rational Unified Process Made Easy*, by Kruchten and Kroll.

Agile and Iterative Development: A Manager's Guide [Larman03] discusses iterative and agile practices, four iterative methods (XP, UP, Scrum, and Evo), the evidence and history behind them, and the evidence of failure for the waterfall.

For other iterative and agile methods, the **Extreme Programming** (XP) series of books [Beck00, BF00, JAH00] are recommended, such as *Extreme Programming Explained*. Some XP practices are encouraged in later chapters of this book. Most XP practices (such as test-driven programming, continuous integration, and iterative development) are compatible with—or identical to—UP practices, and I encourage their adoption on a UP project.

The **Scrum** method is another popular iterative approach that applies 30-day timeboxed iterations, with a daily stand-up meeting with three special questions answered by each team member. *Agile Software Development with Scrum* is recommended reading.

Agile Modeling is described in *Agile Modeling*, by Scott Ambler.

IBM sells the online Web-based RUP documentation product, which provides comprehensive reading on RUP artifacts and activities, and templates for most artifacts. An organization can run a UP project just using mentors and books as learning resources, but some find the RUP product a useful learning and process aid.

For Web resources:

- www.agilealliance.com—Collects many articles specifically related to iterative and agile methods, plus links.
- www.agilemodeling.com—Articles on agile modeling.
- www.cetus-links.org—The Cetus Links site has specialized for years in object technology (OT). Under “OO Project Management—OOA/D Methods” it has many links to iterative and agile methods, even though they are not directly related to OT.
- www.bradapp.net—Brad Appleton maintains a large collection of links on software engineering, including iterative methods.
- www.iturls.com—The Chinese front page links to an English version, with a search engine referencing iterative and agile articles.

Chapter

3

CASE STUDIES

Few things are harder to put up with than a good example.

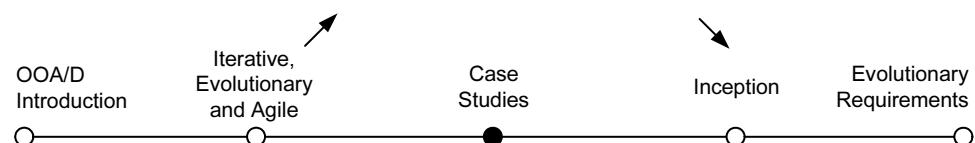
—Mark Twain

Introduction

These case study problems (starting on p. 43) were chosen because they're familiar to many people, yet rich with complexity and interesting design problems. That allows us to concentrate on learning fundamental OOA/D, requirements analysis, UML and patterns, rather than explaining the problems.

What's Next?

Having introduced iterative development, this chapter summarizes the case studies and our focus on the application logic layer. The next chapter introduces the inception phase of the case studies, emphasizing that inception is *not* the waterfall phase of "full" early requirements analysis.



3.1 What is and isn't Covered in the Case Studies?

Generally, applications include UI elements, core application logic, database access, and collaboration with external software or hardware components.

Although OO technology can be applied at all levels, this introduction to OOA/D focuses on the *core application logic layer*, with some secondary discussion of the other layers.



Reglamento General de Protección de Datos (RGPD)

Gestión de Proyectos Informáticos

Proyecto Realizado por:
Daniela Cercel, Imanol
Fránco, Jordi Galiana, Jaime
Monerris, Manuel Peiró,
Wilson Monta.

GPI, L-13-7A Universidad de Alicante

Índice

I. Introducción	2
II. Conceptos Generales/Definiciones	3
III. Participantes	6
IV. Derechos	10
V. Procedimiento y documentos a emplear para el registro de datos.....	13
VI. Obligaciones y Sanciones	17
VII. Observaciones y Conclusiones	20
VIII. Referencias de Información	21

I. Introducción

Somos el grupo L-13-7A de la asignatura Gestión de Proyectos Informáticos teoría, nuestro grupo está formado por seis integrantes: Daniela Cercel, Imanol Fránco, Jordi Galiana, Jaime Monerris, Manuel Peiró y Wilson Monta. En el siguiente documento vamos a tratar el Reglamento de General de Protección de Datos del cual veremos los siguientes aspectos:

- **Conceptos generales:** en donde explicaremos los principales conceptos, como la privacidad de las personas, la Ley Orgánica de Protección de Datos y Garantía de Derechos Digitales, sobre la política de privacidad, etc.
- **Participantes:** donde conoceremos los diferentes participantes en el proceso del tratamiento de datos.
- **Derechos:** donde explicaremos los derechos que hay como, por ejemplo, la transparencia, el derecho al olvido y la limitación de uso de datos.
- **Procedimiento y Documentos a emplear para el Registro de Datos:** mostrando las diferentes medidas para tratar el riesgo que pueden ver expuestos los datos, explicando el procedimiento que se debe llevar para registrar los datos, junto los documentos que son necesarios para cumplir el RGPD.
- **Obligaciones y sanciones:** explicando las diferentes obligaciones que tiene toda persona física o jurídica que tenga que almacenar dicha datos personales, y con los diferentes niveles de sanciones por el incumplimiento de dichas obligaciones.

II. Conceptos Generales/Definiciones

La actual **Ley Orgánica de Protección de Datos y Garantía de Derechos Digitales (LOPDGDD)**, nace al amparo del **Reglamento General de Protección de Datos (RGPD)** y varias leyes tanto nacionales como europeas que ponen de manifiesto el derecho de todas las personas a la protección de toda la información de carácter personal que se va generando con la implantación de la informática en nuestro sistema social y económico. Todo ello deriva de la Declaración Universal de los Derechos Humanos, 10 de diciembre de 1948 en París cuyo artículo 12, que nos dice:



"Nadie será objeto de injerencias arbitrarias en su vida privada, su familia, su domicilio o su correspondencia, ni de ataques a su honra o reputación. Toda persona tiene derecho a la protección de la ley contra tales injerencias o ataques"

y que nace con el propósito de proteger a la persona ante el sistema. Pero esta ley en un principio no contemplaba el incremento y tratamiento de los datos por parte de las empresas y de los gobiernos. Por todo ello la primera Ley Orgánica que va a intentar regular el tratamiento de los datos personales de la población en nuestro país, es la conocida como Ley Orgánica 5/1992, de 29 de octubre, de Regulación del Tratamiento Automatizado de los Datos de Carácter Personal (LORTAD), que nace como extensión del artículo 18 de la Constitución Española de 1978, y establece el marco jurídico de protección en el que se basa la actual legislación española en cuanto a la Protección de Datos de Carácter General.

Tras la ampliación de dicha ley que tuvo lugar en 1999 con el Real Decreto 994/1999, de 11 de junio: Reglamento de Medidas de Seguridad de los Ficheros Automatizados que contengan Datos de Carácter Personal (RMS), se aprobó Ley Orgánica 15/1999, de 13 de diciembre, de Protección de Datos de Carácter Personal (LOPD), vigente en la actualidad, que adaptaba dicha ley a la Directiva Europea, necesaria para entrar en la Unión Europea.

La privacidad de la persona supone un valor en sí mismo para la protección de la dignidad humana y el desarrollo de la persona como tal.

El Reglamento General de Protección de Datos entra en vigor en mayo de 2016, y no fue de total aplicación hasta dos años después, permitiendo una moratoria en el tiempo, para que las empresas, Organismos, Instituciones y Gobiernos se pudiesen adaptar a su cumplimiento. Tras esta y en el mes de diciembre de 2018 entra en vigor la **Ley Orgánica de Protección de Datos y Garantía de Derechos Digitales (LOPDGDD)**. El objetivo de la LOPDGDD es adaptar la legislación española a la normativa europea, definida por el Reglamento General de Protección de Datos (RGPD), vigente desde el 25 de mayo de 2018. Así que, al hablar de protección de datos en España, la Ley de referencia es la LOPDGDD.

Su **Alcance** no solo está limitado a la Unión europea (UE), en cuanto a la obtención y tratamiento de los datos, sino que se amplía a todas las organizaciones de fuera de la UE que recopile cualquier dato de sus residentes (nombre, domicilio, imágenes, correo electrónico, detalles bancarios, publicaciones de redes sociales, información sobre la salud médica o dirección IP de sus dispositivos informáticos).

Los datos solo pueden ser utilizados por los interesados si existe **Una Base Legal** para hacerlo, como que exista un consentimiento expreso del particular, o que se exprese un interés legítimo en la utilización de la información.

Por favor, acepta nuestra Política de Privacidad *

He leído y acepto la Política de Privacidad

ENVIAR

El **consentimiento**, debe ser libre, informado, específico e inequívoco con carácter general y la aceptación no debe de interpretarse con el silencio del ciudadano o usuario en sí. A todos en alguna ocasión nos han “obligado” a aceptar condiciones que nunca nos paramos a leer, para acceder a una aplicación de móvil, o a algún formulario web. Con ello los interesados obtendrán datos sobre nuestro ocio, nuestra movilidad o gustos. Cualquier aplicación gratuita que diariamente instalamos en nuestros dispositivos, por simple que parezca está diseñada para captar datos y revenderlos como forma de financiación de los desarrolladores, aunque nos parezca que su servicio se realiza de forma altruista.

En cuanto a **La Política de Privacidad**, decir que está destinada a advertirnos de si los datos que se recaban son o no utilizados con el consentimiento del usuario, o si van a ser cedidos a terceros. Ejemplo de esta advertencia es la página de la Agencia Española de Protección de datos, donde

Política de privacidad y aviso legal

Política de privacidad

A través de este sitio web no se recaban datos de carácter personal de los usuarios sin su conocimiento, ni se ceden a terceros.

nos advierten de que con la finalidad de ofrecer un mejor servicio y de facilitarnos el uso, nos analizan el número de páginas web que visitamos, así como la actividad que realizamos en ellas y la frecuencia de utilización. Todo ello con el noble propósito de

facilitarnos la vida, y elaborar información estadística, para su posterior utilización. Por último, puntualizar que la política de privacidad es un acuerdo contractual que cada organización establece con el usuario, a través de la aceptación de este de dicho contrato, para el manejo de la información que va a disponer.

La RAE define **dato personal** como “Cualquier información numérica, alfabética, gráfica, fotográfica, acústica o de otro tipo concerniente a personas físicas identificadas o identificables.” Según la ley española identifica estos datos en **tres niveles en función del nivel de seguridad que el tomador de ellos debe de adoptar para su protección**. Estos tres niveles son:

- Datos de **nivel Básico**, que tratan de datos identificativos, circunstancias sociales o familiares, tipo de empleo, o historial de trabajo.
- Datos de **nivel Medio**, que son los que hacen referencia al nivel de solvencia financiera, datos legales y judiciales, y los que incluyen entre otros aspectos sobre la personalidad del sujeto, o su comportamiento.
- Los datos de nivel más sensible o datos de **nivel Alto** incluyen todo lo que hace referencia a la libertad de pensamiento de la persona, religión, condición sexual, afinidad política, o los que obran en posesión de las autoridades policiales.

La **toma de datos** se produce a cada movimiento que se producimos a través de la red, número de visitas a un determinado producto, tiempo en el que estamos viendo un determinado artículo, una foto, la geolocalización de una app, etc. Todo son datos que se captan por los llamados graneros de datos y que luego son vendidos y utilizados por las empresas para desarrollar campañas de marketing, o para bombardearnos con publicidad personalizada en la red, todo ello para hacer negocio.

Reglamento General de Protección de Datos (RGPD)

En **nuestro proyecto**, al ser una aplicación web de gestión de la unidad de negocio de un autónomo, nuestra base de datos va a incluir datos personales de dos tipos, de nivel Básico, como va a ser el tipo de trabajo que efectúa el usuario, si está casado o no, a la hora de la declaración del IRPF, los datos identificativos de este, nombre y apellidos, domicilio, DNI, NIF y, por último los datos de acceso a la aplicación, usuario y contraseña. También nuestra aplicación va a almacenar datos de nivel Medio, como pueden ser datos bancarios, solvencia del usuario, o situaciones de legalidad y compromiso con la tributación.

El tratamiento es toda operación o conjunto de estas llevadas a cabo sobre los datos personales. Dentro de este y por orden de operaciones efectuadas sobre de ellos encontramos, recogida, registro, organización, estructuración, organización, estructuración, conservación, adaptación o modificación, extracción, consulta, utilización, y por último una vez que cumplen su cometido, supresión o eliminación. Los datos personales se organizan en **ficheros** de forma estructurada, y que resultan accesibles según criterios preestablecidos. Los más comunes son: centralizado, descentralizado, de manera funcional o geográfica, etc.

A la hora de aplicar esta ley, cada país tiene jurisdicción propia para juzgar las infracciones, y estas podrán ser alzadas a instancias superiores, como El Tribunal Europeo de Derechos Humanos. En el caso de **las infracciones**, en este Real Decreto-Ley, están consideradas en tres tipos diferentes, según el grado de incumplimiento, leves, graves y muy graves.

- **Las leves** se caracterizan por apertura de expediente sancionador al sujeto, en la mayoría de los casos, y su causa puede ser el incumplimiento de no informar al afectado cuando este lo haya solicitado, o la no transparencia en la información.
- **Las graves** incluyen penas administrativas que varían hasta 10.000.000 de € como máximo o, en su caso si es una empresa hasta el 4% de su volumen de negocio del año anterior al ejercicio de la infracción, lo que puede llegar a ser una cantidad considerable. Aquí se incluyen infracciones como recabar datos de un menor sin su conocimiento, o la adopción de medidas organizativas y técnicas para hacer efectiva al máximo la protección de los datos.
- En el caso de las **Muy Graves** se mantiene la sanción a las empresas en el 4% del volumen de negocio, pero en el caso de que no tenga identidad jurídica el implicado, la sanción puede llegar a ser de 20.000.000 de €. En estas infracciones están la transferencia entre naciones de datos sin ninguna garantía, el uso de los datos con finalidad distinta a la pactada, o no poder acceder a los datos propios, si no se efectúa un pago, el cuál es exigido por el poseedor de estos.

III. Participantes

Interesado

La persona física identificada o identifiable. Se considerará persona física identifiable toda persona **cuya identidad pueda determinarse**, directa o indirectamente, en particular, mediante un identificador, como por ejemplo un nombre, un número de identificación, datos de localización, un identificador en línea o uno o varios elementos propios de la identidad física, fisiológica, genética, psíquica, económica, cultural o social de dicha persona.

El tratamiento de los datos de los **menores de catorce años**, fundado en el consentimiento, sólo será lícito si consta el del **padre/madre/tutor**, con el alcance que determinen el padre/madre/tutor. Para los datos de un menor, **mayor de catorce años**, su consentimiento ya podrá considerarse **válido**.

Uno de los principios básicos de la protección de datos según la LOPD, es el consentimiento, el cual debe cumplir con 4 características para considerarse válida: Toda manifestación de **voluntad libre, específica, informada e inequívoca** por la que el interesado acepta, ya sea mediante una declaración o una clara acción afirmativa, el tratamiento de datos personales que le conciernen (Definición del Reglamento Europeo de Protección de Datos).

Además, el consentimiento debe cumplir las siguientes condiciones para poder verificarse:

- Demostrable
- Distinguible
- Revocable
- Libertad de consentimiento

Responsable del tratamiento o responsable

La persona física o jurídica, autoridad pública, servicio u otro organismo que, solo o junto con otros, **determine los fines, contenidos y medios del tratamiento**. Cuando dos o más responsables determinen conjuntamente los objetivos y los medios del tratamiento serán considerados corresponsables del tratamiento. Los interesados podrán ejercer los derechos que les reconoce el presente Reglamento frente a, y en contra de, cada uno de los responsables.

Sus **deberes** son, llevar un **registro de los tratamientos** de datos personales que se realizan y **atender las peticiones** de los ciudadanos sobre las consultas relacionadas con el ejercicio de sus derechos e **informar de forma transparente** el procedimiento de suministro de información los interesados, en cuanto al ejercicio de sus derechos. Además de **determinar el nivel de seguridad** de los datos recopilados.

El responsable deberá adoptar medidas apropiadas, incluida la **elección de encargados**, de forma que garantice y esté en condiciones de demostrar que el tratamiento se realiza conforme el RGPD (principio de responsabilidad activa). Los responsables deberán de elegir únicamente encargados que, ofrezcan garantías suficientes, para aplicar medidas técnicas y organizativas apropiadas, de manera que el tratamiento sea conforme con los requisitos de la ley. Esta previsión se extiende también a los encargados cuando subcontraten operaciones de tratamiento.

Reglamento General de Protección de Datos (RGPD)

En cuanto al **contenido del contrato para el encargo** de tratamiento de datos:

- Las **relaciones entre el responsable y el encargado**, que deben formalizarse en un contrato o acto jurídico que vincule al encargado con el responsable.
- Una **Regulación minuciosa del contenido mínimo** del contrato, debiendo preverse aspectos como:
 - Objetivo, duración, naturaleza y la finalidad del tratamiento.
 - Tipo de datos personales y categorías de interesados.
 - Obligación del encargado de tratar los datos personales únicamente siguiendo instrucciones documentadas del responsable.
 - Condiciones para que el responsable pueda dar su autorización previa, específica o general, a las subcontrataciones.
 - Asistencia al responsable, siempre que sea posible, en la atención al ejercicio de derechos de los interesados...
- Los contratos de encargo concluidos con anterioridad a la aplicación del Reglamento europeo o Ley española, es decir, en mayo de 2018 deben modificarse y adaptarse para respetar este contenido.

Encargado del tratamiento o Encargado

La persona física o jurídica, autoridad pública, servicio u otro organismo **que trate datos personales por cuenta del responsable** del tratamiento.

En determinadas materias los encargados tienen **obligaciones propias** que establece el RGPD, que no se circunscriben al ámbito del contrato que los une al responsable, y que pueden ser supervisadas separadamente por las autoridades de protección de datos. Por ejemplo:

- Deben **mantener** un **registro** de actividades de tratamiento.
- Deben **determinar** las **medidas de seguridad** aplicables a los tratamientos que realizan.
- Deben **designar** a un **Delegado de Protección de Datos** en los casos previstos por el RGPD.

Para demostrar que los encargados o subencargados ofrecen las **garantías** exigidas por el RGPD, éstos podrán adherirse a **códigos de conducta** o certificarse dentro de los esquemas previstos por el RGPD.

Destinatario

La persona física o jurídica, autoridad pública, servicio u otro organismo **al que se comuniquen datos personales**, se trate o no de un tercero.

No obstante, las autoridades públicas a las que se comunican datos personales en virtud de una obligación legal, para el ejercicio de su misión oficial, no deben considerarse destinatarios de datos si reciben datos personales que son necesarios para llevar a cabo una investigación concreta de interés general, para la Unión o los Estados miembros. El tratamiento de tales datos por dichas autoridades públicas será conforme con las normas en materia de protección de datos aplicables a los fines del tratamiento.

Tercero

Un tercero es aquella persona física o jurídica, autoridad pública, servicio u organismo **distinto del interesado, del responsable, del encargado y de las personas autorizadas** para tratar los datos personales bajo la autoridad directa del responsable o del encargado.

Autoridad de control

En un marco europeo, encontramos las diferentes autoridades de control para cada estado. Una Autoridad de Control, establecida por un Estado miembro se define y configura como **una autoridad pública independiente al tratamiento**, cuya función principal será supervisar la aplicación del Reglamento, con el fin de **proteger los derechos y las libertades** fundamentales de las personas físicas y de facilitar la libre circulación de datos personales en la Unión.

La autoridad de control del establecimiento principal del responsable o del encargado, será competente para actuar como autoridad de control principal para el tratamiento transfronterizo realizado.

Cada autoridad de control será competente para tratar una reclamación que le sea presentada o una posible infracción del Reglamento situado en su Estado miembro o afecte de manera sustancial a interesados en su Estado miembro.

Autoridad de control interesada

Dentro de las autoridades de control tenemos a las autoridades de control interesadas, que son las **autoridades de control a la que les afecta el tratamiento de datos personales** debido a alguna o varias de las siguientes condiciones:

- a) El **responsable** o el **encargado** del tratamiento está establecido en el territorio del Estado miembro de esa autoridad de control.
- b) Los **interesados** que residen en el Estado miembro de esa autoridad de control se ven o es probable que se vean sustancialmente afectados por el tratamiento.
- c) Se ha presentado una **reclamación** ante esa autoridad de control.

Todo esto quiere decir que existe una autoridad de control principal, que, en principio, es en la que se establezca el Responsable, es la autoridad competente, encargada de controlar y supervisar los tratamientos, sin embargo, la autoridad de control interesada es en la que residen el Encargado o el Interesado, y no es la encargada de supervisar la operación, pero si debe participar y cooperar en el procedimiento de investigación.

Representante (del responsable o encargado)

El Representante es la **Persona** física o jurídica, establecida en la Unión, que, habiendo sido **designada**, por escrito, **por el responsable o el encargado del tratamiento**, represente al responsable o al encargado en lo relacionado con sus respectivas obligaciones. Sin embargo, **la designación del representante no afecta a la responsabilidad, ni del responsable, ni del encargado**.

Por esto mismo, dicho representante debe desempeñar sus funciones **conforme al mandato recibido del responsable** o del encargado, incluida la cooperación con las autoridades de control competentes en relación con cualquier medida que se tome para garantizar el cumplimiento del Reglamento.

IV. Derechos

Se debe garantizar una serie de derechos básicos. De este modo, el interesado puede acceder a revisar cuáles son sus datos recogidos, el motivo, y si este se corresponde con el original de su colección. Las reclamaciones de derechos y solicitudes de información para los interesados, como revisión de los anteriores derechos **ARCO**, se resumen en los siguientes puntos, donde se utilizan los conceptos de **interesado** y **responsable** definidos anteriormente:

1) **Transparencia** de información, **Art.12.**

Define la forma en la que se debe presentar la información al interesado.

El interesado debe poder reclamar la información sobre sus datos personales y con qué fines se utiliza. Esta información se debe proporcionar con claridad y de forma concisa; en un formato limpio e inteligible, con un lenguaje sencillo. Se debe ofrecer la información de modo que pueda comprenderse sin necesidad de tener conocimientos legales. Se deben tener en cuenta estas consideraciones con especial cuidado cuando el interesado sea menor.

Algunas prácticas recomendadas son: evitar jerga legal y/o especializada que pueda ser ambigua; unificar la información o bien en formato físico o electrónico, pero evitar utilizar ambos a la vez; en caso de solicitudes excesivas en repetición, se podrá cobrar una comisión de gestión administrativa, o incluso negar el servicio.

2) **Acceso** a la información, **Art.15.**

Define el tipo de contenido al que puede acceder el interesado, relativo a sus datos personales. Amplia la información definida anteriormente.

La solicitud permite conocer los fines de tratamiento y las categorías de estos, plazos y destinatarios que van a tratarlos tanto directamente como a través de terceros interesados; e información sobre otros derechos como la oposición, limitación, y presentar reclamaciones a la autoridad de control pertinente. También permite acceder a las garantías legales de los responsables en caso de transferencias internacionales.

El interesado tendrá derecho a acceder a sus datos siempre que lo necesite, pero no entre en conflicto con lo definido a continuación.

El interesado tendrá acceso a sus datos, en las distintas variantes definidas en los puntos más abajo, dentro de los marcos de tiempo en los que duren los contratos con los responsables y las condiciones del producto o servicio ofrecido. La normativa especifica un máximo de un mes para cumplir con una solicitud de acceso.

El derecho tampoco podrá ejercerse cuando se opongan motivos de interés público o se dé prioridad a la libertad de expresión e información.

3) **Rectificación** de los datos, **Art.16.**

Ante la posible situación de que los datos almacenados contengan inconsistencias o información directamente errónea, el interesado podrá acceder a los mismos en calidad de corrector. Podrá modificar aquellos apartados concretos que desee en un tiempo razonable.

4) **Supresión / derecho al olvido** de la información, **Art.17.**

También llamado derecho a borrar. Supone el fin de la relación del interesado con el servicio o producto. Salvo razones de peso, la empresa borrará todo dato sobre el interesado y, a su vez, informará a todos los terceros pertinentes de que tomen las mismas acciones. Razones principales para la retirada de los datos:

DERECHO AL OLVIDO



GRUPO ÁTICO34

5) **Oposición, Art.21.**

Se garantiza al interesado la opción de retirada de los datos personales para las siguientes situaciones características, por motivos personales u oposición a la mercadotecnia:

- Oposición al **tratamiento automático**: Cuando la solicitud del interesado especifica que no se puede tratar la situación única del cliente mediante métodos automáticos.
- **Retirar el consentimiento**: Cuando se da el caso de ejercer el derecho de supresión o limitación posteriormente al momento de aceptación de las condiciones iniciales.

6) **Limitación de uso de datos, Art.18.**

En casos concretos el interesado podrá solicitar al responsable de recolección y tratamiento de los datos que se limiten estas acciones. En este supuesto, la limitación de los datos delimita que uso quedara restringido al establecimiento de reclamaciones legales, protección de derechos de otras tercera personas físicas o jurídicas, o razones de interés público de la UE o estados miembro.

Estos casos son los detallados a continuación y relativos a los descritos anteriormente:

Oposición: En el caso de que el interesado haya solicitado ejercer el derecho de oposición, y mientras dure el tiempo durante el cual el responsable del tratamiento decida si los motivos son suficientes y lícitos.

Illicitud: Cuando el contenido de los datos de la información personal no sea lícito y el interesado requiera retirarlos. Se ofrece como alternativa a la retirada de los datos por completo.

Reglamento General de Protección de Datos (RGPD)

Inexactitud: Mientras dure el proceso de corrección de los datos vía solicitud del interesado. Se procesa desde el lado del responsable del tratamiento, y mientras se comprueba la exactitud de los mismos.

Reclamaciones: Cuando se pierde el interés original de recolectar los datos por parte del responsable, pero el interesado todavía precisa de esa recolección por motivos de ejecución o defensa de reclamaciones con el mismo.

7) Portabilidad de datos, Art.20.

Se establece, dentro de un formato convenido, la transferencia de los datos del interesado entre distintos responsables o hacia el mismo solicitante y por solicitud del mismo.

Se debe garantizar que el solicitante puede solicitar la trasferencia de sus datos entre las plataformas de distintos interesados en su tratamiento. Esto se debe llevar a cabo cuando sea técnicamente posible para los responsables, y a través de medios electrónicos de lectura mecánica, como los que se encuentran en los pasaportes electrónicos.

El caso más común es cuando se realiza una portabilidad de servicio entre compañías telefónicas, móviles y/o de internet. Otra situación habitual es en el caso de cambios de seguro del hogar, automóvil, etc.

V. Procedimiento y Documentos a emplear para el Registro de Datos

Procedimiento para el ejercicio de los derechos del interesado.

Las entidades responsables deben revisar los procedimientos que se ponen a disposición del interesado para ejercer sus derechos dentro del marco de la RGPD. Estos quedan bajo la responsabilidad de cada entidad interesada en el tratamiento de datos personales, y siempre cumpliendo con los derechos anteriormente definidos.

La entidad interesada debe proporcionar sistemas que permita, a través de una serie de formularios sencillos y accesibles (intuitivos), a todos sus usuarios proceder al ejercicio de sus derechos como interesados. Estos sistemas estarán sujetos a los análisis propios del desarrollo en cuanto a costo/beneficio de los servicios de atención y asistencia en línea.

En resumen, es la entidad interesada la responsable de proveer de la infraestructura necesaria para que el interesado pueda ejercer sus derechos, y como tal tiene la libertad de hacer un diseño acorde a su negocio, pero siempre dentro del marco del RGPD.

Evaluación del Impacto en Protección de Datos (EIPD)

El RGPD exige que los responsables del tratamiento implementen medidas de control adecuadas para demostrar que se garantizan los derechos y libertades de las personas y la seguridad de los datos, teniendo en cuenta entre otros, los “riesgos de diversa probabilidad y gravedad para los derechos y libertades de las personas físicas” (**artículo 24.1**) y aplicando las medidas oportunas. Para ello, el responsable del tratamiento debe considerar desde el inicio, en la fase de diseño, las acciones preventivas suficientes para poder identificar, evaluar y tratar los riesgos asociados al tratamiento de datos personales, y así, poder asegurar los principios de protección de los datos garantizando los derechos y libertades de los interesados.

Es importante describir en detalle todas las actividades u operaciones que se llevan a cabo sobre los datos de carácter personal con el objetivo de entender los posibles riesgos a los que se pueden ver expuestos los datos. La evaluación del impacto deberá incluir como mínimo:

- Una descripción sistemática de la actividad de tratamiento previstas
- Una evaluación de la necesidad y proporcionalidad del tratamiento respecto a su finalidad
- Una evaluación de los riesgos
- Las medidas previstas para afrontar los riesgos, incluidas garantías, medidas de seguridad y mecanismos que garanticen la protección de datos personales.

El nivel de riesgo se puede tratar con el objetivo de reducir o mitigar el mismo, en función de la medida que se adopte. Existen cuatro medidas diferentes para tratar el riesgo:

- **Reducción del riesgo:** Para reducir el nivel de riesgo, se deben establecer medidas de control que reduzcan los niveles de probabilidad y/o impacto asociados al riesgo inherente.
- **Retención del riesgo:** Si el nivel de riesgo inherente es inferior al nivel de riesgo considerado como aceptable, no existe necesidad de implementar controles adicionales.

Reglamento General de Protección de Datos (RGPD)

- **Transferencia del riesgo:** Consiste en compartir un riesgo con una organización externa. Se puede transferir el riesgo a una aseguradora que afronte las posibles consecuencias materiales. Sin embargo, se ha de considerar que, en ocasiones, la transferencia de riesgos puede generar otros riesgos. Por ello, la transferencia puede generar la necesidad de análisis adicionales.
- **Anulación del riesgo:** Si el riesgo es muy elevado y no se quiere asumir el mismo, se puede decidir abandonar la actividad de tratamiento.

Las medidas de control tienen como objetivo mitigar o minimizar el riesgo asociado a una operación de tratamiento. Es importante destacar que el objetivo principal de una EIPD no es eliminar completamente el riesgo asociado a las actividades de tratamiento, lo que se pretende es reducir el mismo hasta un nivel aceptable para poder llevar a cabo las mismas garantizando los derechos y libertades de los interesados.

Durante el proceso de definición de las medidas de control se debe considerar de forma independiente cada riesgo identificado y establecer tantas medidas de control como sean necesarias hasta lograr un nivel de riesgo aceptable. Existen diversos tipos de medidas de control, por ejemplo:

- **Organizativas:** Medidas asociadas a procedimientos, a la organización y gobierno de la entidad. En esta tipología de medidas se pueden incluir los procedimientos para ejercer los derechos de los interesados, protocolos para gestionar vulnerabilidades e incidentes, etc.
- **Legales:** Medidas asociadas al cumplimiento normativo. Por ejemplo, cláusulas para recogida de consentimientos expresos, etc.
- **Técnicas:** Medidas que permiten velar por la seguridad física y lógica de los activos de información. Por ejemplo, controles de acceso, cifrado, etc.

Como último paso en la realización de una evaluación de impacto, se debe elaborar un plan de acción donde se describan todas las medidas de control definidas para tratar los riesgos identificados y concluir con respecto al resultado obtenido. Un plan de acción es el conjunto de iniciativas que se deben llevar a cabo para implantar los controles que ayudan a reducir el riesgo de una actividad de tratamiento hasta un nivel considerado aceptable. Se recomienda que el plan de acción incluya al menos los siguientes campos de información:

- Control
- Descripción del control
- Responsable de implantación
- Plazo de implantación

Como hemos dicho anteriormente la EIPD es una herramienta con carácter preventivo que debe realizar el responsable del tratamiento para poder identificar, evaluar y gestionar los riesgos a los que están expuestas sus actividades. Ahora bien, se exige una EIPD cuando el tratamiento pueda entrañar un alto riesgo para los derechos y libertades de las personas. Se exige una EIPD por lo menos en los tres casos siguientes:

- La evaluación sistemática y exhaustiva de aspectos personales de una persona, incluida la elaboración de perfiles
- El tratamiento a gran escala de datos sensibles
- La observación sistemática a gran escala de una zona pública

Reglamento General de Protección de Datos (RGPD)

Documentación.

Para cumplir totalmente con el RGPD se necesitan todos los documentos expuestos a continuación:

- **Política de Protección de Datos Personales (Artículo 24):** es un documento de alto nivel para la gestión de privacidad en su empresa, que define lo que quiere alcanzar y cómo.
- **Aviso de Privacidad (Artículos 12, 13, y 14):** es un documento (que puede ser también publicado en la página web de la empresa) explica en palabras sencillas cómo se tratarán los datos personales de clientes, visitantes del sitio web, y otros.
- **Aviso de Privacidad de los Empleados (Artículos 12, 13 y 14):** explica que la empresa va a tratar datos personales de sus empleados (que pueden incluir registros de salud, antecedentes penales, etc.).
- **Política de Retención de Datos (Artículos 5, 13, 17, y 30):** describe el proceso de decidir cuánto tiempo va a guardarse un tipo de datos personales en particular, y cómo se destruirá de forma segura.
- **Programa de Retención de Datos (Artículo 30):** enumera todos los datos personales y describe cuánto tiempo cada tipo de datos se conservará.
- **Formulario de Consentimiento del Interesado (Artículos 6, 7, y 9):** esta es la forma más común de obtener el consentimiento de un interesado/a para tratar sus datos personales.
- **Formulario de Consentimiento Paterno (Artículo 8):** si el interesado tiene menos de 16 años, entonces el padre necesita proporcionar el consentimiento para tratar los datos personales.
- **Registro de EIPD (Artículo 35):** aquí es donde registra todos los resultados de la Evaluación de Impacto de Protección de Datos.
- **Acuerdo de Tratamiento de Datos del Proveedor (Artículos 28, 32, y 82):** se necesita este documento para regular la protección de datos con un encargado de tratamiento o cualquier otro proveedor.
- **Procedimiento de Respuesta a la Violación de Seguridad de Datos (Artículos 4, 33, y 34):** describe que hacer antes, después, durante y después de una violación de seguridad de datos.
- **Registro de Violación de Seguridad de Datos (Artículo 33):** aquí es donde registrará todas las violaciones de seguridad de datos.
- **Formulario de Notificación de Violación de Seguridad de Datos a la Autoridad de Control (Artículo 33):** en caso de tener una violación de seguridad de datos, se tendrá que notificar formalmente a la Autoridad de Control.
- **Formulario de Notificación de Violación de Seguridad de Datos a los Interesados (Artículo 34):** en caso de una violación de seguridad de datos, tendrá existente la obligación de notificar a los interesados de manera formal.

Documentos necesarios bajo ciertas condiciones:

- **Descripción del Puesto de Delegado de Protección de Datos (Artículos 37, 38, y 39):** se necesitará disponer de un Delegado de Protección de Datos (DPD) si (a) el tratamiento es llevado a cabo bajo una autoridad u organismo público, a excepción de los tribunales que actúan en su capacidad judicial; o (b) las actividades centrales consisten en operaciones de tratamiento que precisan un seguimiento regular y sistemático de los interesados a gran escala; o (c) las actividades centrales tratan una categoría especial a gran escala de datos y datos personales relacionados con condenas y delitos penales.
- **Listado de Actividades de Tratamiento (Artículo 30):** documento es obligatorio si (a) la empresa tiene más de 250 empleados; o (b) el tratamiento que lleva a cabo la empresa puede resultar en un riesgo para los derechos y libertades de los interesados; o (c) el tratamiento no es ocasional; o

Reglamento General de Protección de Datos (RGPD)

(d) el tratamiento incluye categorías especiales de datos; o (e) el tratamiento incluye datos personales relacionados con condenas y delitos penales.

- **Cláusulas Contractuales Estándar para la Transferencia de Datos Personales a Responsables (Artículo 46):** obligatorio si se transfieren datos personales a un responsable fuera del Área Económica Europea (AEE) y se ampara en cláusulas modelo como la base legal para las transferencias transfronterizas de datos.
- **Cláusulas Contractuales Estándar para la Transferencia de Datos Personales a Encargados (Artículo 46):** obligatorio si se transfieren datos personales a un encargado fuera del Área Económica Europea (AEE) y se ampara en cláusulas modelo como la base legal para las transferencias transfronterizas de datos.

VI. Obligaciones y Sanciones

El 19 de enero de 2008 se publicó en el Boletín Oficial del Estado, el Real Decreto 1720/2007, de 21 de diciembre, en el que se aprobaba la Ley Orgánica 15/1999, de 13 de diciembre, encargada de regular la Protección de datos de carácter personal.

En dicho reglamento, en el art. 80 RLOPD, se establecían tres niveles de cumplimiento; Bajo, Medio y Alto. Siendo cada uno de ellos acumulativo del anterior, es decir, si se desea alcanzar el nivel Medio es necesario cumplir todos los requisitos del nivel Bajo, lo mismo pasaría con el nivel Alto, en el que se debería cumplir todos y cada uno de los requisitos de los niveles nombrados anteriormente.

- a. El nivel **Alto** debe ser de obligada aplicación para todo aquel que almacene la información más sensible y confidencial de sus usuarios, algunos ejemplos pueden ser los que manifiesten las ideologías políticas y creencias religiosas, las afiliaciones a sindicatos, origen racial, y por supuesto los datos de la salud y la vida sexual del usuario.
- b. Para el nivel **Medio**, se reserva para los datos referentes a sanciones administrativas, informes de solvencia (ficheros de impagadores), datos de Hacienda, y aquellos datos que permitan estudiar la personalidad del sujeto, es decir todos aquellos que puedan crear un perfil tipo, a partir de información sobre gustos, aficiones, y demás factores del estilo de vida del sujeto en cuestión.
- c. Por último, en el nivel **Bajo**, se refiere al resto de los datos para identificar a los sujetos, como puede ser nombre, apellidos, DNI, número de teléfono, etc.

Una vez aclarado los distintos tipos de niveles, para toda persona física o jurídica que tenga que almacenar dicha información debe cumplir con las siguientes obligaciones:

a. Calidad de los datos

Solo deben recogerse aquellos datos que sean adecuados y que cumplan la finalidad para la que el afectado los está cediendo, cualquier uso innecesario o excesivo debe ser cancelado. Si un dato deja de ser necesario en un momento posterior también debe ser cancelado.

b. Consentimiento del afectado

El tratamiento de los datos de carácter personal que son regulados en el reglamento anteriormente nombrado debe cumplir con el consentimiento inequívoco del afectado. Dicho consentimiento podrá ser retirado si existe una causa justificada.

c. Datos especialmente protegidos

Los datos de nivel medio y alto, es aconsejables no solicitarlos de no ser totalmente imprescindible para la actividad. De acuerdo con ello está prohibido solicitarlos para exclusivamente tenerlos almacenados.

A demás, en el apartado 2 del artículo 16 de la Constitución Española, se nombra el derecho a que nadie sea obligado a revelar su ideología o religión.

Reglamento General de Protección de Datos (RGPD)

Por tanto, se deberá insistir a lo hora de recabar esta información del usuario a su derecho de no proporcionarlo.

d. Deber de secreto

Al igual que otros profesionales independientes, en el momento de recabado de la información de carácter personal, se debe guardar el secreto profesional del usuario.

e. Comunicación o cesión de datos

Para la cesión de datos a terceros, deberá ser imprescindible el cumplimiento de los deberes anteriormente nombrados, haciendo hincapié en el consentimiento del afectado, que será debidamente comunicado que su información personal podrá ser cedida a terceros.

Además, esta cesión debe ser absolutamente necesaria para fines directamente relacionados a la cesión de los datos.

f. Derechos de las personas

Se debe tener en cuenta los siguientes derechos mencionados en el punto 3 de este documento, haciendo especial hincapié en:

- i. Derecho de acceso:
- ii. Derecho de rectificación y modificación
- iii. Derecho de cancelación
- iv. Derecho de oposición

g. Seguridad de los datos

Se deberán adaptar las mecánicas de naturaleza organizativas y/o técnica, para garantizar la seguridad de los datos de carácter personal. Todo ello se llevará a cabo con el fin de que no sean alterados, se sufran pérdidas, o se lleve a cabo un acceso no autorizado.

h. Notificación de ficheros

Se deberá notificar a la Agencia de Protección de Datos Española (AGPD) los ficheros que se utilizaran para almacenar los datos, así como su estructura y una descripción. Pero nunca incluyendo la propia información de los usuarios en esta notificación.

En el caso de que, por el motivo que fuere, se cometa una infracción el reglamento de RGPD contempla, en sus artículos 72, 73 y 74; tres tipos de infracciones:

i. Infracciones consideradas muy graves:

Dichas infracciones serán sancionadas con hasta 20 millones de Euros, y en el caso de empresas como máximo un 4% del volumen de cifra de negocios del ejercicio contable inmediatamente anterior.

Este tipo de infracción tipifica la ausencia de responsabilidad en materia de principios básicos, donde estos son incumplidos.

Reglamento General de Protección de Datos (RGPD)

Si los derechos de los interesados son vulnerados, también se considera una infracción muy grave.

ii. Infracciones consideradas graves:

Dichas infracciones serán sancionadas con hasta 10 millones de Euros, y en el caso de empresas como máximo un 2% del volumen de cifra de negocios del ejercicio contable inmediatamente anterior.

Se considera una infracción grave el incumplimiento de las obligaciones del responsable y la falta de compromiso de las administraciones certificadoras.

iii. Infracciones consideradas leves:

Se considerarán leves las restantes infracciones de los artículos mencionados en los apartados 4 y 5 del artículo 83 del Reglamento (UE) 2016/679.

Las sanciones impuestas por aplicación del RGPD tendrán un periodo de prescripción dependiente de la valía de la sanción, los plazos son los siguientes:

- i. Las sanciones por importe igual o inferior a 40.000 euros prescriben en el plazo de un año.
- ii. Las sanciones por importe comprendido entre 40.001 y 300.000 euros prescriben a los dos años.
- iii. Las sanciones por un importe superior a 300.000 euros prescriben a los tres años.

VII. Observaciones y Conclusión

Una vez llegados a este punto queremos añadir una parte de crítica a esta ley y a su aplicación. El sistema digital está desarrollado para copar de necesidades al usuario. Hoy en día se puede conseguir todo desde detrás de una pantalla y una conexión a la red. Por ello, y ante un mundo tan grande creado para satisfacer todas nuestras necesidades, hay grandes compañías que encapsulan condiciones inaceptables sobre la utilización de nuestros datos personales, en extensos formularios que la mayoría de los usuarios aceptan ante la complejidad de este o la prisa por acceder a algo que sabemos cierto que nos recompensara. Por este motivo creemos que estamos indefensos, no ante una ley que verdaderamente debe de protegernos, pero si ante un sistema que comercia con un poder creciente y que verdaderamente nadie sabe aún donde están sus límites. Por ello, como el delito no está aún definido es difícil que los que deberían hacer cumplir la ley se impliquen en ello. Se deja hacer mientras que todos consigamos la parte que demandamos. Mientras tanto los derechos de la persona se van perdiendo.

Esta ley, como tal, no protege de la comercialización de las grandes empresas sobre los datos, ni el uso que se hace de ellos, o los robos que se producen, o las pérdidas que se declaran, si nos protege del uso de estos por las pequeñas empresas que están cercanas a nosotros. Al menos hay medio camino recorrido.

VIII. Bibliografía

Conceptos Generales/Definiciones y portada.

<https://sites.google.com/site/wwwtecnoinformaticadmcom/blog/5gestiondeproyectosinformaticos>

<https://oposicionestic.blogspot.com/2017/12/breve-historia-de-la-proteccion-de.html>

<http://www.injuve.es/convivencia-y-salud/noticia/celebra-el-70-aniversario-de-la-declaracion-universal-de-derechos-humanos>

<https://reduce.es/se-clasifican-los-datos-personales-segun-la-lopd/>

<https://dej.rae.es/lema/tratamiento-de-datos-personales>

<https://ayudaleyprotecciodatos.es/2019/02/19/sanciones-rgpd-lopd-2019/>

<https://protecciodatos-lopd.com/empresas/nueva-ley-proteccion-datos-2018/>

<https://www.iberley.es/temas/glosario-definiciones-rgpd-62717>

Participantes

<https://www.aepd.es/sites/default/files/2019-09/guia-rgpd-para-responsables-de-tratamiento.pdf>

<https://www.iberley.es/legislacion/reglamento-ue-2016-679-27-abr-doue-reglamento-general-europeo-proteccion-datos-gdpr-rgpd-24473701>

<https://www.iberley.es/temas/glosario-definiciones-rgpd-62717>

<https://sedeagpd.gob.es/sede-electronica-web/vistas/infoSede/detallePreguntaFAQ.jsf?idPregunta=FAQ%2F00096>

<https://www.aepd.es/sites/default/files/2019-10/guia-ciudadano.pdf>

Derechos

<https://protecciodatos-lopd.com/empresas/nueva-ley-proteccion-datos-2018/>

<https://advisera.com/eugdpracademy/es/knowledgebase/8-derechos-del-interesado-segun-el-rgpd/>

<https://blog.signaturit.com/es/gdpr-nuevos-derechos-que-disfrutaremos-todos>

Procedimiento y Documentos a emplear para el Registro de Datos

<https://advisera.com/eugdpracademy/es/knowledgebase/lista-de-documentos-obligatorios-requeridos-por-el-rqpd-de-la-ue/>

<https://advisera.com/eugdpracademy/es/knowledgebase/5-fases-evaluacion-impacto-proteccion-datos-RGPDUE/>

<https://www.aepd.es/sites/default/files/2019-09/guia-evaluaciones-de-impacto-rqpd.pdf>

https://ec.europa.eu/info/law/law-topic/data-protection/reform/rules-business-and-organisations/obligations/when-data-protection-impact-assessment-dpia-required_es

Obligaciones y Sanciones

<https://www.boe.es/buscar/act.php?id=BOE-A-2008-979&p=20120308&tn=0#reglamento>

<https://www.aepd.es/es/derechos-y-deberes/cumple-tus-deberes/principios>

<https://www.crear-empresas.com/obligaciones-proteccion-datos>

<https://www.aepd.es/es/derechos-y-deberes/cumple-tus-deberes/medidas-de-cumplimiento/delegado-de-proteccion-de-datos/certificacion>

<https://www.boe.es/buscar/doc.php?id=BOE-A-2018-16673>

TEMA 8:

LA PROPIEDAD INTELECTUAL

Ricardo González Torres

Ana Romero Lobato

Carlos Garrido Marín

Pablo García Tolosa

Pablo Pérez Mora

Andrés García Fernández

Javier Segovia Antón



ÍNDICE

REGISTRO DE PROGRAMAS	4
1.1. Propiedad Intelectual	4
1.2. Ley que regula la PI	4
1.3. El Porqué de la necesidad de PI	4
1.4. Infringir los derechos sobre Propiedad Intelectual	5
1.5. Figuras involucradas, derechos y su duración	7
1.6. Necesidad de proteger proyectos mediante PI	8
1.7. Procedimiento, impresos necesarios y tasas a pagar.	8
1.7.1. La vía telemática	9
1.7.2. La vía presencial	9
1.7.3. Europa	10
1.8. Transmisión de la propiedad intelectual	10
USO DE RECURSOS.	11
2.1. Excepciones	11
2.2. Dónde buscar recursos de terceros	12
2.2.1. Imágenes	14
2.2.2. Música	14
2.2.3. Software	14
2.2.4. Otros Recursos	15
SOFTWARE LIBRE	15
3.1. Definición	15
3.2. Ventajas y Desventajas	16
3.3. Libertades del software libre	17
3.4. Tipos de Licencias	18
3.4.1. Licencias GPL	18
3.4.2. Licencia AGPL	18
3.4.3. Licencia BSD	19
3.4.4. Licencia Apache	19
3.4.5. Licencias Creative Commons	19
SOFTWARE PRIVATIVO	20
4.1. Definición	20
4.2. Ventajas e inconvenientes del software propietario	21
4.3. Tipos de software privativo	22

4.4. Tipos de licencias de software privativo	24
4.4.1. Licencias OEM	24
4.4.2. Licencias EULA	24
4.4.3. Licencias ROK	25
4.4.4. Licencias PKC	26
NUEVAS FORMAS DE USO	26
5.1. Modelos de negocio	26
5.1.1. Modelos de negocio en Internet:	27
5.1.2. Nuevos modelos basados en la comercialización y marketing en Internet:	29
5.2. Relación Proyecto Autónomos	32
5.2.1. Modelos Posibles	32
5.2.2. Consideraciones	33

1. REGISTRO DE PROGRAMAS

1.1. Propiedad Intelectual

Cuando hablamos de la propiedad intelectual nos podemos referir tanto a ideas, información y conocimiento que están bajo diferentes derechos legales que protegen la autoría de su creador. Su objetivo es estimular la creatividad en beneficio de todos al garantizar que las ventajas derivadas de la explotación de una creación beneficien al creador. Esto al final lo que consigue es el fomento de la actividad creativa y permite que el creador pueda sacar beneficio económico sin que otros puedan hacerlo hasta un determinado tiempo.

Hoy en día la propiedad intelectual es un derecho poco respetado en cuanto al software informático se refiere debido en gran parte a la piratería y al poco control que se puede ejercer ante esto.

1.2. Ley que regula la PI

En el caso de España la propiedad intelectual es respaldada por el Real Decreto Legislativo 1/1996, de 12 abril, que aprueba el texto refundido de la Ley de Propiedad Intelectual. Esta ley abarca diferentes puntos como: derechos de autor, sujeto, objeto y contenido de los derechos de autor, derechos de explotación y otros derechos de los creadores y autores.

1.3. El Porqué de la necesidad de PI

Es necesario ya que protege jurídicamente los desarrollos industriales e intelectuales a través de distintos mecanismos: marcas, secretos comerciales, patentes y derechos de autor. Los recursos más utilizados para la protección del software por su utilidad jurídica son los derechos de autor y las patentes industriales y sus semejantes, aunque el uso de marcas va en aumento. A pesar de ello, y como ya se ha comentado anteriormente, estos mecanismos tienen limitaciones (temporales, de ámbito, etc.) y su uso depende de los intereses de los autores.

Si no se protegiese la propiedad intelectual, los esfuerzos económicos y humanos estarían desprotegidos del plagio y demás prácticas desleales. Además, estos derechos salvaguardan los intereses creativos de los autores y sus obras, tanto de su literalidad como de sus ideas e ideas subyacentes.

1.4. Infringir los derechos sobre Propiedad Intelectual

La protección de los derechos de autor viene recogida en el artículo 20 de la Constitución Española y, como consecuencia, estos derechos se protegen también desde la perspectiva del Derecho Penal, al encontrarse en el Código Penal diversos delitos que castigan a los que violen dicho derecho, como por ejemplo el delito del artículo 270 del Código Penal, en el que se castiga con una pena de prisión de seis meses a cuatro años al que: “con ánimo de lucro, plagie, reproduzca, distribuya o comunique públicamente, en parte o en su totalidad, una obra literaria, artística o científica”. Los derechos sobre las dos últimas vienen determinados por la Ley de Propiedad Intelectual.

En ésta, podemos encontrar a partir del artículo 138 y en adelante, varios medios de protección. Según estos artículos, el titular de los derechos reconocidos en la Ley de Propiedad Intelectual puede pedir que cesen las actividades ilícitas de cualquier infractor, que afecten a sus derechos (Artículo 139) y exigir la correspondiente indemnización por daños y perjuicios (Artículo 140). Además de ello, podrá solicitar la adopción de medidas cautelares urgentes (Artículo 141). Dichas medidas de cese y cautelares, se podrán solicitar también contra los intermediarios a los que se recurra por un tercero para llevar a cabo la infracción de los derechos de propiedad intelectual.

El **Artículo 139** contempla el cese de diversas actividades, las cuales son:

1. Suspensión de la actividad explotadora o de la actividad de infracción, incluyendo todos aquellos actos o actividades a los que se refieren los artículos 196 y 198.
2. Prohibición de reanudar la explotación o la actividad de infracción.
3. Retirar del comercio los productos ilícitos o su destrucción, lo cual se llevará a cabo a cargo del infractor.
4. Retirada de los circuitos comerciales, inutilización o destrucción de los moldes, planchas y demás elementos que se hayan utilizado para la reproducción, creación o fabricación de los productos ilícitos.

5. Remoción o precinto de los aparatos utilizados en la comunicación pública no autorizada de las obras o prestaciones, así como de aquellas en las que se haya suprimido o alterado sin autorización la información para la gestión electrónica de derechos, en los términos previstos en el artículo 198, o a las que se haya accedido eludiendo su protección tecnológica, en los términos previstos en el artículo 196.
6. Comiso, inutilización o destrucción de los elementos con cargo al infractor, de los elementos que se hayan utilizado para la neutralización o supresión de cualquier dispositivo técnico de protección de un sistema informático, para proteger un programa de ordenador. Las mismas medidas podrán adoptarse en relación con los dispositivos, productos o componentes para la elusión de medidas tecnológicas a los que se refiere el artículo 196 y para suprimir o alterar la información para la gestión electrónica de derechos a que se refiere el artículo 198.
7. Remoción o precinto de los instrumentos utilizados para facilitar la neutralización o supresión de cualquier dispositivo técnico utilizado para proteger obras o prestaciones.
8. La supresión de servicios prestados por intermediarios a terceros que se sirvan de ellos para infringir derechos de propiedad intelectual.

La indemnización a la que alude el **Artículo 140** se deberá fijar de acuerdo con alguno de los tres criterios previstos en dicho artículo, que son los siguientes:

- a) Las consecuencias económicas negativas, entre ellas la pérdida de beneficios. Y en caso de daño moral se deberá indemnizar también.
- b) La cantidad que como remuneración hubiera percibido el perjudicado.
- c) La acción para reclamar los daños y perjuicios prescribe a los cinco años.

Las medidas cautelares se solicitarán según el **Artículo 141**. Según este artículo, dichas medidas se adoptarán en casos urgentes y pueden consistir en:

- 1º La intervención de los ingresos obtenidos por la actividad ilícita de que se trate.
- 2º La suspensión de la actividad ilícita.

3º El secuestro de los ejemplares producidos o utilizados y del material empleado para la infracción.

4º Secuestro de los instrumentos, dispositivos, productos y componentes referidos en los artículos 102.c) y 196.2 y de los utilizados para la supresión o alteración de la información para la gestión electrónica de los derechos referidos en el artículo 198.2.

5º El embargo de los equipos, aparatos y soportes materiales utilizados para la comisión del acto ilícito.

6º Y la suspensión de los servicios prestados por intermediarios a terceros para poder violar la propiedad del autor

1.5. Figuras involucradas, derechos y su duración

Cuando hablamos de la propiedad intelectual podemos definir diferentes derechos en función lo que se quiera proteger.

Los derechos más relevantes son:

- **Marcas:** Son signos que permiten diferenciar los productos o servicios de una empresa de las demás. Las marcas pueden consistir en palabras, letras, números, dibujos, fotos, formas, colores, logotipos, etiquetas o combinaciones de estos elementos que se empleen para diferenciar el origen de los productos o servicios.

En el caso de España la marca se obtiene mediante un registro y con una protección de 10 años.

- **Secretos comerciales:** Es información comercial confidencial que otorga una ventaja competitiva a una empresa. La apropiación indebida, la divulgación o el uso no autorizado de dicha información se considera una práctica desleal y una violación del secreto comercial.

La protección de los secretos comerciales continúa de manera indefinida siempre que el secreto no se revele al público.

- **Patentes:** La patente es un derecho exclusivo otorgado por el Estado para una invención que es nueva, implica una actividad inventiva (o no es evidente) y es susceptible de aplicación industrial (o es útil). La patente otorga a su titular el derecho exclusivo de impedir que otros fabriquen, utilicen, ofrezcan para la venta, vendan o importen la invención patentada sin la autorización del titular.

La patente es concedida por la oficina nacional o regional de patentes, y es válida durante un período de tiempo limitado, que suele ser de 20 años a partir de la fecha de presentación

- **Derecho de autor y derechos conexos:** El derecho de autor otorga a los autores, artistas y otros creadores (por ejemplo, las empresas de programas informáticos, los productores de multimedios y los diseñadores de sitios Web) protección jurídica para sus creaciones literarias y artísticas.

El derecho de autor tiene un plazo de protección general de 70 años contados desde el fallecimiento del creador o autor.

1.6. Necesidad de proteger proyectos mediante PI

El Artículo 1 de la Ley de Propiedad Intelectual (RDL 1/1996) establece que “*La propiedad intelectual de una obra literaria, artística o científica corresponde al autor por el sólo hecho de su creación*”. En un principio la ley define quién presupone que es el autor de dicha obra.

Sin embargo, para proteger su autoría (una medida de protección más), es necesario un registro de la propiedad intelectual. Así lo indica el Artículo 101 de dicha ley: “*Los derechos sobre los programas de ordenador (...) podrán ser objeto de inscripción en el Registro de la Propiedad Intelectual.*”.

A la cuestión de si es necesario registrarla para proteger la propiedad intelectual, conforme lo dispuesto en la ley, no es necesario, pero sí recomendable.

1.7. Procedimiento, impresos necesarios y tasas a pagar.

A nivel nacional hay dos formas de realizar la inscripción. El registro es de competencia autonómica y depende directamente del Ministerio de Cultura.

1.7.1. La vía telemática

Tiene los siguientes pasos:

1. Tener un certificado digital.
2. Acceder a la página del Ministerio de Cultura y seleccionar nuestra comunidad autónoma.
3. Rellenar la solicitud, los formularios requeridos y presentar el fichero que contenga la creación.
4. Abonar la tasa de Tramitación de expediente de solicitud (13,46 €) y firmar digitalmente la solicitud.

1.7.2. La vía presencial

Tiene los siguientes pasos:

1. Acudir a cualquier Registro Territorial u Oficinas Delegadas o a las Oficinas Provinciales del Registro Central. En las Comunidades Autónomas de Madrid, La Rioja, Asturias y Murcia se presenta en la propia sede del Registro Territorial.
2. Cumplimentar la solicitud y acompañarla del ejemplar de la obra a registrar.
3. Abonar la tasa de Tramitación de expediente de solicitud (13,46 €).

En Alicante, la sede presencial se encuentra en la Avda. de Aguilera, nº 1, 2º planta.

Hay 5 impresos asociados a dicha solicitud:

- [\[MODELO A\] SOLICITUD REGISTRO PROPIEDAD INTELECTUAL](#)
- [\[MODELO A2\] RELACIÓN DE AUTORES](#)
- [\[MODELO B\] OBJETO/S DE PROPIEDAD INTELECTUAL QUE DESEA PROTEGER](#)
- [\[MODELO A3\] RELACIÓN DE TITULARES DE DERECHO](#)
- [SOLICITUD GENERAL - PROCEDIMIENTOS](#)

Dichos impresos se pueden encontrar en la propia web de la Generalitat Valenciana (en el caso de vivir en la Comunidad Valenciana).

1.7.3. Europa

A nivel europeo, la EUIPO administra los derechos sobre marcas. Estos derechos complementan los adquiridos a nivel nacional.

Básicamente el registro consta de 4 fases: examen, oposición, registro y recurso.

- En el examen la oficina examina la solicitud en busca de errores y ambigüedades.
- En la fase de oposición, terceros tienen un plazo de tres meses para oponerse porque piensen que existe un derecho anterior al nuestro y sea suyo o por motivos absolutos, que son observaciones de errores gratuitas por parte de estos terceros.
- Si no hay oposición o se resuelve favorablemente, se realiza el registro y se publica.
- La fase de recurso es para que las partes perjudicadas por esta resolución final puedan presentar un recurso.

La tasa mínima es de 850€.

La sede de la EUIPO se encuentra precisamente aquí en Alicante, en la Avenida de Europa, 4.

1.8. Transmisión de la propiedad intelectual

Los derechos de propiedad intelectual son transmisibles a excepción de los derechos morales del autor de una obra. La normativa sólo permite transmitir los derechos de propiedad intelectual de estas dos formas:

- **Mortis causa:** Consiste en la transferencia de la titularidad de la propiedad intelectual de una persona física a otra persona física o jurídica derivada de la muerte o declaración de fallecimiento de aquella con el fin de la explotación ([art 42 LPI](#)).
- **Intervivos:** Se basa en la transmisión de la cesión de la titularidad de un derecho a un tercero, normalmente como consecuencia de un contrato expreso o una presunción legal

vinculada a un contrato con el objetivo de la explotación de la creación del creador, ya sea, produciendo, distribuyendo y/o comunicando su creación. ([art 43 LPI](#))

2. USO DE RECURSOS.

En este apartado vamos a hablar de los diferentes materiales de terceros que podemos utilizar en un proyecto, tales como imágenes, música, programas software, etc, así como la manera en la que están protegidos estos recursos y algunas excepciones en su protección.

2.1. Excepciones

La ley de Propiedad intelectual recoge los recursos protegidos por los derechos de autor. Esta ley según su Artículo 2 es el autor el que tiene el derecho exclusivo de explotación de la obra, a excepción de algunos casos que recoge esta misma ley, como: [\[Real Decreto Legislativo 1/1996, 12 de Abril\]](#)

- **Artículo 32. Citas y reseñas e ilustración con fines educativos o de investigación científica:** Es lícita la inclusión de recursos de naturaleza escrita, sonora o audiovisual, así como la de obras aisladas de carácter plástico o fotográfico figurativo, siempre que se trate de obras ya divulgadas. Su inclusión debe ser a través de citas o, comentario o juicio crítico. Además, se indicará la fuente y el nombre del autor de la obra utilizada.
- **Artículo 38. Actos oficiales y ceremonias religiosas:** La ejecución de obras musicales en el curso de actos oficiales del Estado, de las Administraciones públicas y ceremonias religiosas no requerirá autorización de los titulares de los derechos, siempre que el público pueda asistir a ellas gratuitamente y los artistas que en las mismas intervengan no perciban remuneración específica por su interpretación o ejecución en dichos actos.
- **Artículo 39. Parodia:** No será considerada transformación que exija consentimiento del autor la parodia de la obra divulgada, mientras no implique riesgo de confusión con la misma ni se infiera un daño a la obra original o a su autor.

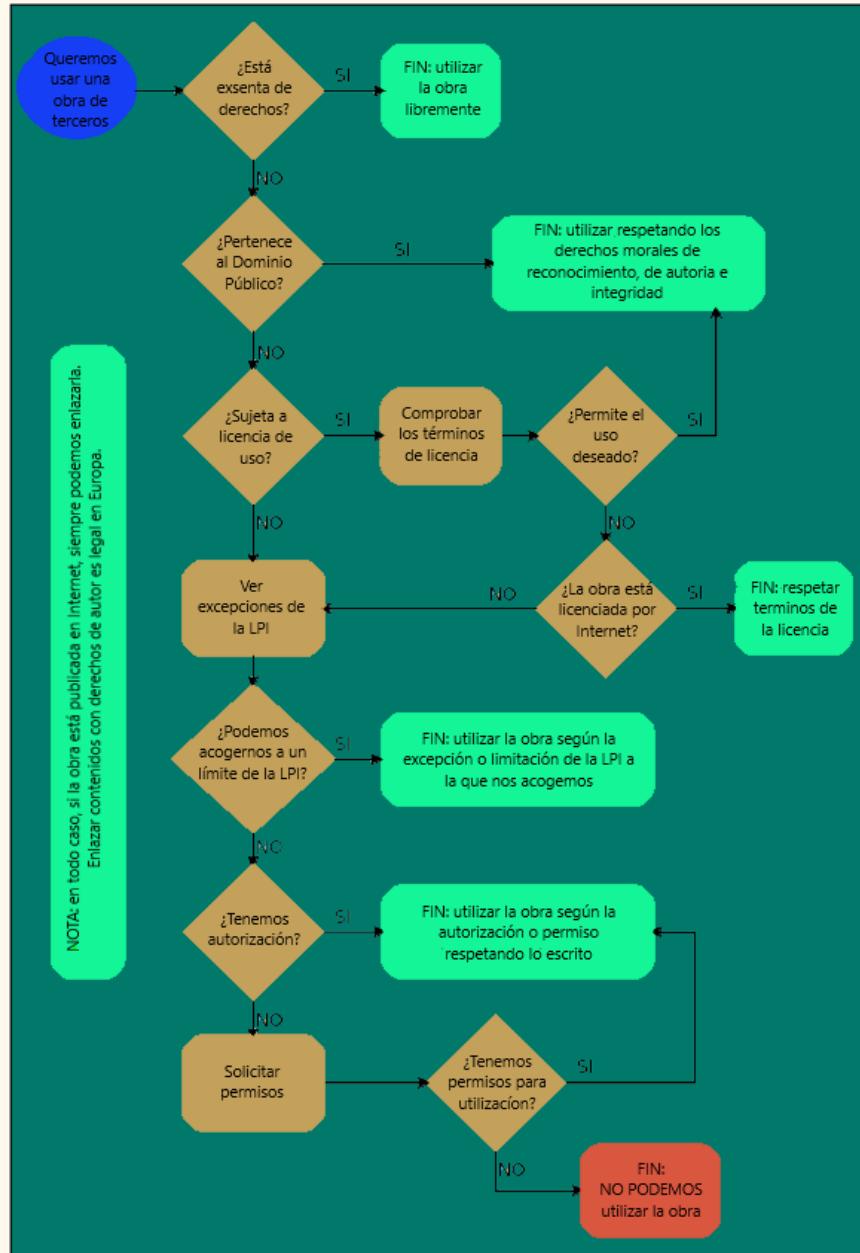
En la mayoría de los proyectos, se utilizan recursos de terceros, ya sean imágenes, sonidos, videos, software, o simplemente código (librerías) y la mayor parte de estos recursos normalmente están protegidos tanto los derechos de autor convencionales, como Copyright, o por licencias como **Creative Commons** o **CopyLeft**, aunque también se pueden encontrar recursos de Dominio Público que se pueden utilizar libremente, tomando ciertas consideraciones.

Si hablamos de Copyright, siempre se deberá llegar a un acuerdo con el autor, o persona que tenga los derechos del recurso que queremos utilizar, para obtener la autorización conveniente. Normalmente esta autorización se consigue al realizar un pago por la compra o alquiler del derecho de uso, aunque el propietario del derecho siempre puede negar su autorización, y ante la negativa no hay forma de hacer nada y no se podrá hacer uso del recurso.

En el caso de recursos protegidos por licencias, lo que se puede o no se puede hacer con ese recurso varía según la licencia a la que esté suscrita esta.

2.2. Dónde buscar recursos de terceros

La mayoría de recursos de terceros los podemos encontrar en Internet, y estos como ya hemos hablado antes pueden estar protegidos por derechos de autor o licencias.



En nuestro caso hablaremos de recursos tales como imágenes, música y software y una mezcla de recursos a veces menos utilizados.

2.2.1. Imágenes

La ley diferencia las Imágenes en grupos diferentes ya sean obras fotográficas o una fotografía como objeto individual.

Si hablamos del caso de una obra fotográfica, en ellas se consideran la originalidad y personalidad del creador implícitamente. Una obra es protegida durante toda la vida del autor y hasta 70 años después de su muerte.

En el caso de una mera fotografía, esta solo está protegida durante 25 años desde que esta fue tomada.

Podemos encontrar webs con imágenes que no estén protegidas por Copyright, y que sean de uso libre, como por ejemplo: [Flickr](#) (usa CC) o [Pixabay](#) y otras páginas donde pagar por imágenes para su uso como [GettyImages](#).

2.2.2. Música

Como en las imágenes, la música también está protegida por los derechos de autor y si queremos utilizar música protegida con Copyright lo más probable es que tengamos que pedir permiso al autor, lo que se suele traducir en la compra del derecho de uso. El pago puede oscilar de 2 a 2000 euros.

La compra del derecho normalmente es para un solo uso a no ser que se especifique de alguna manera en el contrato.

Para buscar música de libre uso podemos utilizar los siguientes sitios: Youtube Music Library, [Jamendo](#) o [Freesound.org](#).

2.2.3. Software

Los programas software o librerías que queremos incluir en proyectos normalmente están protegidos por licencias. Si por ejemplo encontramos una licencia en un recurso que queremos introducir, podremos hacerlo sin problemas siempre y cuando nuestro proyecto no tenga fines comerciales y la licencia lo impida. Además, las licencias nos obligan a reconocer al

autor del recurso y a incluir la licencia del recurso en el proyecto. Si es una licencia [CopyLeft](#), estaremos también obligados a que nuestro proyecto tenga la misma licencia que el recurso que incluimos.

Los recursos software como hemos visto antes pueden ser programas en sí o el propio código fuente y si hablamos del código fuente, en él se pueden ver diferentes consideraciones, como por ejemplo, según qué licencia proteja el código, podremos modificarlo cuanto queramos o utilizar partes del código para crear algo completamente distinto a la función principal en la que se tenía pensado su uso, o que no se permita la modificación de este en ningún concepto.

Podemos encontrar diferentes librerías y recursos software en páginas como [GitHub](#).

2.2.4. Otros Recursos

Otros recursos que se pueden utilizar en proyectos son modelos 3D, texturas, materiales, estos se pueden encontrar en páginas como Sketchfab o Squid Pro, donde hay tanto una selección gratuita de estos recursos protegidos por licencias, o la opción de comprar los recursos para su uso en los proyectos que queramos.

3. SOFTWARE LIBRE

3.1. Definición

El software libre se refiere al programa informático en el que el usuario tiene libertades sobre el mismo, pudiendo ser usado, copiado, modificado y distribuido como el usuario quiera. Su potencial es muy grande, ya que permite a millones de programadores aplicar y ampliar su crecimiento, y es que, la disponibilidad y el acceso global a este tipo de software permite la expansión del conocimiento depositado en cada pieza desarrollada.

Algunas de sus ventajas más importantes es que el hecho de poder acceder a su código fuente, ofrece cierta independencia tecnológica permitiendo desarrollar tu propia versión del software.

No hay que confundir tampoco, el software libre con el software gratuito, ya que el software libre puede ser gratuito o de pago, pero el usuario siempre será libre de usarlo como desee. El software gratuito se podría denominar como freeware: en este caso, a pesar de que el software es gratuito, está restringido con licencias, con lo que el usuario no es libre de actuar como con el software libre.

3.2. Ventajas y Desventajas

Hay muchas ventajas del uso del software libre para las pymes, destacando algunas bastante relevantes:

- Descargar y probar sin coste y de forma ilimitada en el tiempo: Puedes descargar y probar cualquier software libre antes de incorporarlo al ecosistema de aplicaciones de tu empresa. Sin restricciones, probando bien todas sus características antes de tomar la decisión final de usarlo.
- Soporte gratis y bueno en general: Detrás de las aplicaciones de software libre hay una comunidad de desarrollo que se encarga de resolver las dudas y cuestiones técnicas, y de definir su evolución. Aquí hay que ser cautos, las aplicaciones de cierta importancia tienen una comunidad fuerte detrás y el soporte es muy bueno, pero si la aplicación no es de uso extendido, el soporte puede ser precario.
- Seguridad muy por encima del software privativo: El hecho de que se tenga acceso al código fuente hace que muchos ojos revisen y prueben ese código y sobre todo sea difícil guardar secretos. Esto provoca que la seguridad sea muy superior en general al software privativo donde el código solo es observado por unos pocos y es, por tanto, más fácil que tenga más agujeros de seguridad.

Las desventajas del software libre son inferiores a sus ventajas, pero podemos destacar algunas que conviene tener en cuenta:

- Soporte técnico urgente: Salvo que se tenga contratado directamente con una empresa, el soporte técnico del software libre es muy bueno, pero lógicamente puede que no se ajuste a lo que se necesita en casos de emergencia. Esto es algo normal. Si se requiere un soporte con unos SLA (Service Level Agreement) determinados por necesidades de negocio, no significa que haya que prescindir del software libre, sino que conviene contratar ese soporte a una empresa que lo proporcione y se ajuste a lo que queremos.

- Usabilidad de las aplicaciones: Esto es algo que va cambiando pero existe. Las aplicaciones suelen tener una usabilidad no muy trabajada en general, habiendo ciertas excepciones.
- Conocimiento técnico: No es necesario ser un técnico para usar el software libre. Pero sí es conveniente que si vas a incorporarlo a tu empresa, tengas una persona que conoce ese mundo y te ayude a analizar y seleccionar las herramientas que necesitas, valorando bien la madurez de los desarrollos y la comunidad que hay detrás en cada caso.

3.3. Libertades del software libre

Para que un programa sea de software libre, este debe contar con cuatro libertades:

Libertad 0: de usar el programa con cualquier propósito

La libertad de ejecutar el programa con cualquier propósito se trata de que todo usuario es libre de utilizar el software en:

- cualquier sistema de computación;
- con cualquier tipo de trabajo y finalidad;
- sin obligación de comunicar a ninguna entidad específica o al programador.

En otras palabras, al usuario no se le prohíbe o se le impide ejecutarlo ya que lo que importa es su propósito y no el del programador.

Libertad 1: de estudiar cómo funciona el programa y poder modificarlo

Para tener las libertades 1 y 3 es necesario tener acceso al código fuente, por lo que esta es una condición fundamental para el software libre.

Ahora bien, la Libertad 1 constituye la libertad de usar la versión modificada (según los deseos y necesidades del usuario) en lugar de la original.

Libertad 2: de distribuir copias del programa a cualquier persona u organización

La libertad para distribuir el programa (Libertades 2 y 3), consiste en que los usuarios tienen la libertad de redistribuir y compartir copias con o sin modificaciones, sin necesidad de pedir ni pagar algún permiso para hacerlo.

Además, se puede distribuir los programas gratuitamente o cobrando alguna tarifa por ello.

Libertad 3: de mejorar el programa y compartir las mejoras en beneficio de todos

La Libertad 3 permite mejorar el programa — y, al igual que la libertad 2 — posibilita compartir y liberar las versiones modificadas como software libre y, de tal forma, beneficiar con ello a las personas u organizaciones que lo utilicen.

En conclusión, muchas empresas que han ganado mucho dinero haciendo software libre y utilizándolo pero para hacer el proyecto de gestión de autónomos y entregárselo personalmente al cliente hemos preferido que nuestro proyecto no sea software libre.

3.4. Tipos de Licencias

3.4.1. Licencias GPL

Es una de las más utilizadas y se suele denominar como **GNU GPL**. Con esta licencia el desarrollador conserva los derechos de autor, pero permite su **libre distribución, modificación y uso** siempre y cuando, en el caso de que el software se modifique, el nuevo software que se desarrolle como resultado quede obligatoriamente con la misma licencia. Está considerada la primera licencia [copyleft](#) y, bajo esta filosofía, cualquier código fuente licenciado bajo GPL, debe estar disponible y accesible, para copias ilimitadas y a cualquier persona que lo solicite. De cara al usuario final, el software licenciado bajo GPL es **totalmente gratuito**, pudiendo pagar únicamente por gastos de copiado y distribución.

3.4.2. Licencia AGPL

Se engloba dentro de las licencias destinadas a modificar el derecho de autor derivadas de GNU. La novedad de AGPL es que, aparte de las cláusulas propias de una GNU GPL, ésta obliga a que se distribuya el software que se destine a dar servicios a través de una red de ordenadores, es decir, si se quiere usar como parte del desarrollo de un nuevo software, éste quedaría obligado a su **libre distribución**.

3.4.3. Licencia BSD

Es un buen ejemplo de una **licencia permisiva** que casi no impone condiciones sobre lo que un usuario puede hacer con el software. El software bajo esta licencia es la menos restrictiva para los desarrolladores, ya que, por ejemplo, el software puede ser vendido y no hay obligaciones de incluir el código fuente. Además, una aplicación licenciada con **BSD** permite que otras versiones pueden tener otros tipos de licencias, tanto libres como propietarias; un buen ejemplo de ello es el conocido sistema operativo Mac OS X, desarrollado bajo esta licencia. También, **BSD** permite el cobro por la distribución de objetos binarios.

3.4.4. Licencia Apache

El software bajo este tipo de licencia permite al usuario distribuirlo, modificarlo, y distribuir versiones modificadas de ese software pero debe conservar el copyright y el disclaimer. La **licencia Apache** no exige que las obras derivadas (las versiones modificadas) se distribuyan usando la misma licencia, ni siquiera que se tengan que distribuir como software libre, sólo exige que se informe a los receptores que en la distribución se ha usado código con la licencia Apache. En este sentido, al crear nuevas piezas de software, los desarrolladores deben incluir dos archivos en el directorio principal de los paquetes de software redistribuidos: una copia de la licencia y un documento de texto que incluya los avisos obligatorios del software presente en la distribución.

3.4.5 Licencias Creative Commons

Su definición se basa en cuatro condiciones:

- ‘**Atribución**’, con la cual se puede distribuir, exhibir, representar... siempre y cuando se reconozca y se cite a su autor
- ‘**No comercial**’, que no permite usar el software con fines comerciales
- ‘**No derivadas**’, con la cual no se puede modificar dicha obra
- ‘**Compartir igual**’, que incluye la creación de obras derivadas siempre que mantengan la licencia original.

Bajo las licencias Creative Commons, existen otros tipos de licencia las cuales se diferencian entre sí con el “juego” de la combinación de las cuatro condiciones que acabamos de ver.

Con estas licencias, el desarrollador tiene la posibilidad de elegir lo que considere más conveniente para su trabajo. Esta decisión implica la renuncia a algunos derechos reservados inherentes al derecho de autor, tales como el derecho de reproducción y el derecho de

transformación que, por otra parte, puede contribuir a la divulgación de su trabajo y permitir el derecho de acceso por parte del público a sus obras.

Tipo de Licencia	Descripción	Atribución del autor o del tipo de licencia
GNU GPL	Libre, abierta, gratuita	Sí
AGPL	Libre, abierta, gratuita	Sí
BSD	Libre, abierta, permisiva, gratuita Con cláusula de advertencia	No
Apache	Libre, abierta, gratuita Con copyright y disclaimer	Si
Creative Commons	Libre, gratuita Con propiedad intelectual	Si

En nuestro proyecto pondremos la licencia GPL ya que conservamos los derechos de autor, pero permitimos su **libre distribución, modificación y uso** siempre y cuando, en el caso de que el software se modifique, el nuevo software que se desarrolle como resultado quede obligatoriamente con la misma licencia.

4. SOFTWARE PRIVATIVO

4.1. Definición

El software privativo, o también conocido como software propietario o de código cerrado, es aquel cuya principal característica es estar limitado tanto en uso, como modificación o la distribución de este. Unas reglas que suelen ser estipuladas por el desarrollador o empresa detrás de este software; y que a menudo es considerado como el antónimo del software libre que sigue reglas completamente distintas.

Este tipo de software es considerado el más popular en todo el mundo; ya que, a pesar de las buenas intenciones de algunas fundaciones que promueven el software libre, el software privativo o comercial es el más recurrido debido al ingreso que este representa.

Este ingreso es debido a la existencia de la licencia. Es decir, el contrato entre el titular y los usuarios, que estipula lo que éstos pueden hacer con su obra: uso, copia, redistribución, modificación, etc., y en qué condiciones.

Las diferencias entre las distintas licencias, aunque en principio puedan parecer pequeñas, suelen suponer condiciones de uso y redistribución totalmente diferentes, dando lugar en los últimos años a una nueva forma de entender la informática, que ha supuesto además el desarrollo de modelos de negocio que van más allá de los desarrollos a medida y las licencias de uso.

La historia de este tipo de licencia comenzó al final de los años 70'. después de que el gobierno de los Estados Unidos obliga a IBM para hacer distinción entre la fabricación de software y hardware. Con esta acción, comienzan a desarrollarse los primeros programas cerrados y que posteriormente se convertiría en la licencia de software propietario que hoy conocemos.

Específicamente, el software propietario cuenta a menudo con las siguientes características:

- Es imposible de copiar y/o modificar.
- Su protección es respaldada por el Copyright o la protección.
- Normalmente cuentan con un nivel o estándares de calidad superior al del software libre.
- Puedes localizarlo fácilmente con retailer o en tiendas especializadas.
- El soporte o mantenimiento es exclusivo del propietario de su licencia.

4.2. Ventajas e inconvenientes del software propietario

El software propietario es desarrollado por grandes corporaciones que emplean grandes recursos en su proceso y por lo tanto someten a sus productos a un periodo de validación más

largo antes de su implementación definitiva, lo que hace que este tipo de software sea más estable desde su lanzamiento al mercado.

El software propietario tiene un uso mayoritario, suele ser utilizado por organismos gubernamentales, universidades, empresas, aunque poco a poco el software libre se está abriendo camino entre los ámbitos oficiales.

Otra ventaja del uso de software propietario es el soporte que ofrece a sus clientes y la estabilidad con la que cuentan al saber que hay una gran empresa detrás del software que asegure su estabilidad y perdurabilidad en el tiempo.

Por otro lado, el software propietario es un producto cerrado que no puede ser modificado ni mejorado, son los usuarios y las empresas las que deben adaptarse a él y no al contrario.

En contra del software libre, que cuenta con el apoyo de una comunidad que en su mayoría no tiene fines comerciales, el software propietario depende de una empresa. Como todas las empresas están sujetas a intereses económicos, por lo tanto, pueden quebrar o ser absorbidas por empresas mayores, que dejen de continuar con esa línea de desarrollo.

Caso muy parecido es el que ocurrió con el conocido software de diseño vectorial FreeHand, su empresa, Macromedia, fue absorbida por Adobe y el programa dejó de actualizarse hasta quedar en desuso en favor del software que ya comercializaba Adobe, Adobe Illustrator.

Lo importante es elegir el software que mejor atienda tus necesidades y que cuente con una comunidad activa detrás, ya sea propietario o no.

4.3. Tipos de software privativo

En la actualidad el software propietario se presenta en la mayoría de los casos como un software comercial (aunque no necesariamente el software comercial tiene que ser de carácter propietario, también puede ser libre).

Podemos encontrar ejemplos de lo que es el software propietario en todos los ámbitos y segmentos del mundo del software.

Si empezamos por lo más básico, el sistema operativo, tenemos excelentes ejemplos en las dos principales plataformas de escritorio: macOS de Apple, y Windows de Microsoft.

Si bien parece que estos “nos los regalan” cuando adquirimos una computadora nueva, para utilizarlos debemos aceptar una licencia de uso que restringe nuestra capacidad de uso del sistema: para macOS solamente podemos utilizarlo en una computadora Apple, para la cual ya hemos pagado el precio del sistema con su compra.

De la misma forma, también pagamos la licencia de Windows cuando adquirimos una computadora PC, y aunque podemos “trasladar” la licencia de una computadora a otra -y en según qué condiciones y el tipo de licencia adquirida-, sólo podemos utilizarla en una computadora a la vez.

Tampoco podemos compartir el sistema con otro usuario, ni hacer copias para distribuirlo. Es todo lo contrario a lo que nos permiten las plataformas libres como GNU/Linux o FreeBSD entre otras.

Otros ejemplos de software comercial más populares del mundo:

- La suite de oficina Microsoft Office, que incluye en su paquete al procesador de texto Word, la planilla de cálculo Excel, y el creador de presentaciones Power Point, entre otros.
- El famoso Adobe Photoshop, una de las mejores, quizás la mejor, aplicación para la manipulación de imágenes a nivel profesional.
- Los softwares de diseño gráfico vectorial Corel Draw o Adobe Illustrator, ambos estándares de la industria.
- El software de diseño CAD AutoCAD, de la firma Autodesk.
- Los softwares de compresión de archivos WinZIP o WinRAR.

4.4. Tipos de licencias de software privativo

4.4.1. Licencias OEM

Las licencias que más solemos encontrar por Internet, y las más baratas además, son las licencias OEM. Estas licencias son las que suelen venir cuando compramos un ordenador, portátil o de sobremesa, ya montado, y es que, cuando nos conectamos a Internet y activamos Windows, el hardware de nuestro ordenador queda ligado a la licencia y, mientras no cambiemos los componentes esenciales del mismo, no tendremos problemas.

Con las licencias OEM, generalmente, podemos cambiar la memoria RAM o la tarjeta gráfica sin tener problemas de activación, sin embargo, si cambiamos otros componentes como el disco duro, el procesador y, sobre todo, la placa base, la licencia quedará ya inservible y, para volver a activar Windows, tendremos que comprar otra.

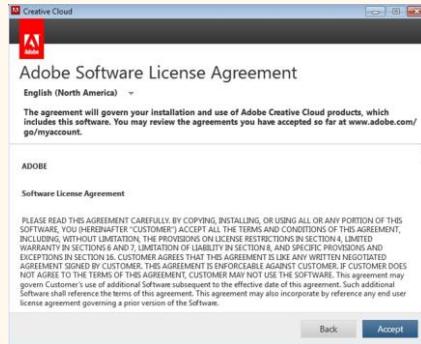


4.4.2. Licencias EULA

Un acuerdo de licencia de usuario final (End-User Licence Agreement) es un contrato legal acordado entre el desarrollador o vendedor de software y el usuario del software, a menudo cuando el software ha sido comprado por el usuario a un intermediario, como un minorista.

Una licencia EULA especifica en detalle los derechos y restricciones que se aplican al uso del software.

Muchos de estos contratos sólo están contenidos en forma digital, y sólo se presentan a un usuario como un click a través del cual el usuario debe "aceptar". Cómo es posible que el usuario no vea el acuerdo hasta después de haber comprado el programa informático, estos documentos pueden ser contratos de adhesión.



4.4.3. Licencias ROK

Estas licencias son un caso especial de las licencias OEM. Son una versión más económica que ya viene instalada de fábrica en los servidores de HP, Dell, Lenovo, etc. Su uso está muy extendido en la pequeña empresa.

No se recomienda su uso en sistemas virtuales, ya que si instalamos una licencia de este tipo la máquina estará vinculada a un fabricante y hardware en concreto, y perdemos la principal ventaja de la virtualización, que es el independizar el software del hardware.



4.4.4. Licencias PKC

Hace años las licencias físicas se vendían en formato caja e incluían un CD, pero cayeron en desuso y ya no se utilizan prácticamente.

Hoy en día la versión física es sinónimo de licencia PKC. Si la compras recibes un paquete (caja, sobre, ...) y una tarjeta con el código de activación.

Ésta es una licencia perpetua que se puede desinstalar y volver a instalar en otro equipo.



5. NUEVAS FORMAS DE USO

En la época actual estamos viviendo una revolución tecnológica que está obligando a las empresas a reestructurarse y a evolucionar para mantenerse en el mercado. Es por ello, que durante los últimos años y las próximas décadas los modelos de negocio, es decir, los modelos esquemáticos que describen la manera en que las empresas crean y producen valor para sus clientes y la recompensa que obtienen de ello, experimentarán o están experimentando una reestructuración total para adaptarse a la nueva sociedad.

5.1. Modelos de negocio

Existen multitudinarios modelos de negocio en la era digital. Los podríamos dividir en:

- **Modelos de negocio en Internet:** Modelos que se basan directamente en la obtención de riqueza a través del pago por una prestación o un conjunto de prestaciones en internet y no tienen ningún matiz de marketing o comercialización.

- **Modelos basados en la comercialización y el marketing en Internet:** Modelos que sí que tienen matices de marketing y comercialización y tienen, al menos, como uno de sus objetivos dar a conocer o hacer llegar un bien o servicio en concreto.

5.1.1. Modelos de negocio en Internet:

- **Micropagos – contenido fraccionado:** Los micropagos nacen en el entorno de Internet y, en cierto modo, tienen mucho que ver con los contenidos fraccionados y con el consumo por contenido. Se suele entender por micropago una transacción, por lo general entre menos de 1€ y los 5€ (Paypal o Visa defienden que pueden llegar hasta los 10€ o 20€), para acceder a algún tipo de contenido. Este tipo de modelo se puede ver en una multitud de aplicaciones para smartphones y va muy ligada al modelo de Freemium que se comentará seguidamente.
- **Pago por Consumo – Streaming/Pay per view:** Pay per View es un modelo de negocio que nació en el sector de la televisión. Como indica su nombre se trata de un sistema de pago en el que el usuario paga sólo por lo que ve, aunque fuera del medio audiovisual sería el “Pago por Consumo”. En lo referente a este modelo podemos nombrar “Canal Plus” o “Netflix” como ejemplo, dado que en esta plataforma pagas por “consumir” o ver cierto contenido. Este modelo está muy ligado a la suscripción, teniendo en cuenta que en sistemas de *Streaming* se estila el pago de una suscripción por el uso de la misma para ver contenido, en el caso de Canal Plus, audiovisuales.
- **Suscripción:** La economía de la suscripción no hace sino seguir esta senda de modelos flexibles. Aunque hemos hablado algo en cuanto al paradigma “Netflix”, debemos señalar algunos aspectos del modelo suscripción. Una de las ventajas de este modelo es la posibilidad de trabajar con una base de clientes fija en un tiempo concreto (semanal, mensual, anual) y, por tanto, contar con un flujo de ingresos también fijos, es decir, positivo, ya que los suscriptores pagan por adelantado. En un auténtico pago por consumo no sería exactamente así. Además, para la planificación del negocio este sistema ofrece una visión más clara de las necesidades reales del mismo.

- **Membresía:** La membresía, el ser miembro de un servicio, es una modalidad de la que se comienza a hablar en los últimos tiempos, sobre todo en el sector de las comunicaciones. También se podría considerar un tipo de suscripción, aunque en términos generales, en realidad ser miembro o ser suscriptor son cosas algo diferentes. La membresía supone pertenecer a un grupo que, en términos de intercambio, puede ser también cualquier tipo de empresa que oferte servicios o contenidos. Por tanto, la suscripción implica el pago regular, mientras que ser miembro supone pertenencia a algo: club, comunidad, etc. Lleva implícita una relación más directa y de filiación. Es el caso de servicios web como Match.com, la plataforma para encontrar pareja, que hacen que sus usuarios no sean meros suscriptores del servicio, sino miembros de una comunidad a las que se les ofrece ventajas que son los que realmente impulsan el negocio.
- **Freemium – Premium:** Los modelos de negocio Freemium llevan años funcionando en la Red. Este modelo consiste en ofrecer algún producto o contenido de manera gratuita mientras que para tener acceso a otra parte de los contenidos hay que pagar: el servicio llamado Premium. En ocasiones, este tipo de modelo incluye publicidad o marketing agregado a dichos contenidos, esperando que este tipo de ingresos sustenten el negocio junto a los consumidores Premium, a los que se tarda un tiempo en convencer. Sería el caso de Spotify o de muchas aplicaciones móviles.
- **Por publicidad insertada:** Este modelo en realidad es otra versión del modelo Freemium/Premium. Consiste en ofrecer contenido gratuito, pero con publicidad insertada, frente al Premium, que va sin publicidad además de obtener otras ventajas por el pago. Este modelo va muy ligado al Freemium. El ejemplo más claro para este modelo es Spotify. También se estila en muchas aplicaciones móviles.
- **Acceso abierto (*Open Access*):** Hace referencia a todo tipo de acceso sin necesidad de suscripción previa o pago. Este modelo suele utilizarse más para ofrecer material de tipo educativo, científico o académico que está en relación directa con la gestión de adquisiciones y préstamos en las Bibliotecas. Hablamos de servicios como son

bibliotecas digitales como la Virtual Miguel de Cervantes en España, Internet Archive u Open Library, a nivel internacional. En nuestro país, por ejemplo, publicaciones y repositorios científicos, como las del CSIC, se basan en el Acceso Abierto.

- **P2P – MOOC's:** El peer-to-peer (o P2P; “de igual a igual”, literalmente) tiene un origen tecnológico. El germen de este modelo está en las redes P2P, un sistema de conexión directamente entre ordenadores, es decir, de ordenador a ordenador. Esto permite la conexión vertical entre particulares a través de la red, sin necesidad de utilizar servidores fijos. Lo que en un principio servía para compartir de manera rápida y eficaz documentos en el trabajo, se convirtió en un método popular para compartir todo tipo de archivos entre personas de todo el mundo a través de los denominados programas P2P.

Los principales ejemplos de este modelo se encuentran en el sector educativo así como la University of the People (UoPeople), primera institución del mundo de contenidos académicos gratuitos online dedicada a la promoción y democratización mundial de la educación superior.

5.1.2. Nuevos modelos basados en la comercialización y marketing en Internet:

- **“Paga lo que quieras” – PWYW:** Este modelo lleva al extremo las políticas de precios dinámicos. En este modelo, los clientes optan por pagar el precio sugerido por la empresa, un mínimo casi simbólico incluso en casos muy extremos nada en absoluto (básicamente son casos de puro marketing). Se considera más un modelo para ganar visibilidad que algo sostenible a largo plazo. Un ejemplo muy claro es el de *Gumroad*, una plataforma de comercio electrónico en la que los usuarios pueden vender directamente a los compradores estableciendo precios muy bajos.
- **Bundle:** Se trata de la oferta de paquetes de productos. Pueden ser del mismo tipo o un combinado de productos o servicios diferentes, pero de la misma naturaleza. Es un modelo ideal para vender diversos productos que el cliente percibe como más asequibles que si los comprara por separado. No obstante, hay casos en los que los productos del bundle o paquete son inseparables, es decir, el conjunto es el producto en sí. Es un modelo muy utilizado en el mundo de las telecomunicaciones y de la informática

(software), como es el caso de Microsoft Office, que vende varias aplicaciones en una: Excel, Word, PowerPoint, etc.

- **Crowdfunding:** Es un modelo que tiene gran popularidad y visibilidad, sobre todo en el medio online. Se basa en la financiación masiva de un proyecto, servicio, contenido, plataforma, productor, obra. En definitiva, cualquier cosa susceptible de ser financiada entre varios mecenas que voluntariamente quieran participar y hacer posible la creación o puesta en marcha de una idea.

Se trata de la búsqueda de financiación recurriendo a la buena voluntad de las personas, pero lo que hace del crowdfunding un modelo de negocio establecido es su relación con la red y su vocación global. En cuanto a plataformas de crowdfunding que estén especializadas por sectores: moda (Fanstylers), audiovisuales (TheCrowdTelevision), solidarios (Migranodearena), etc.

- **Gamificación:** Hace referencia a la inclusión de mecánicas de juego en aspectos y contextos que no tienen que ver con el juego. Con esto se pretende potenciar y motivar el desarrollo de labores, ideas o algún interés, en definitiva, actuar sobre el comportamiento de forma lúdica y en cualquier ámbito: cultural (museos, teatro, libros), de empresa, de ocio, de comercio, etc.

Por lo general, este tipo de iniciativas se basan en un modelo de recompensas. Es un paso más allá de otros modelos, como la recompensa por valoraciones o votaciones sobre objetos de consumo (ya sean productos en tiendas online, videojuegos ...). El aprovechamiento del valor de la experiencia que supone el juego, el hecho de superar un desafío y el discernimiento que le son propios pueden atraer a un público determinado hasta entonces ajeno a determinados productos. Un ejemplo muy claro es el de Airbnb o BlaBlaCar en las que quién hace funcionar las aplicaciones son una comunidad que ofrece sus servicios ya sea como huésped ofreciendo su casa p como conductor ofreciendo su coche. En estas aplicaciones es posible opinar sobre otros usuarios para incentivar o impulsar una percepción sobre cada uno.

- **Venta directa:** La venta directa es la comercialización y venta de productos directamente a los consumidores, sin ningún tipo de intermediario o distribuidor (modelo B2C). En realidad, este modelo es previo a la era digital, pero Internet y el e-

commerce le dan un nuevo significado y valor a la posibilidad de vender cualquier tipo de producto directamente al cliente, y más si se trata de un contenido digital, cuyo acceso es inmediato. En cuanto a ejemplos, hoy en dia, cualquier marca de ropa ofrece sus servicios de venta online en sus páginas web.

- **Autoedición:** Se trata de la auto-publicación de material literarios. A lo largo de los dos últimos años, desde importantes editoriales nacionales e internacionales hasta librerías y bibliotecas están apostando firmemente por la autoedición como complemento a su actividad principal. Añadir una plataforma de autoedición al negocio “tradicional” redefine los límites hasta ahora establecidos entre escritores y aspirantes a escritores en el mundo del libro.

En cuanto a algunos ejemplos, hay varias entidades del sector del libro que están apostando por la autoedición: HarperCollins, Roca Editorial, Apple, Casa del Libro, entre otras, han lanzado diversas iniciativas para adentrarse en este nuevo modelo de negocio, incluso Amazon con Amazon Kindle Direct Publishing.

- **Préstamo de contenidos digitales en bibliotecas:** En este ámbito nos encontramos con varias posibilidades de modelos. Desde el modelo de licencia individual no concurrente (pago por préstamo de una obra en formato digital a un usuario no simultáneo), con el que la empresa que ofrece el servicio de préstamo digital da a la biblioteca acceso a su catálogo, en la mayoría de los casos previo pago de una suscripción, y esta ha de pagar por cada préstamo de un título que se efectúe (un modelo que constituye una traslación del servicio de préstamo tradicional al entorno digital), hasta encontrarse con opciones más amplias y diversas desde el modelo de suscripción anual, al catálogo al modelo de pago por consumo referido al pago de un contenido específico, que ha sido ampliamente definido en este estudio; pasando por el modelo de licencia anual por título, que ofrece a las bibliotecas la posibilidad de prestar un ebook de forma ilimitada no simultánea durante su periodo de duración y la opción de compra perpetua de títulos.

También hay modelos más singulares que tienen una presencia más residual pero que ponen en evidencia las enormes posibilidades que se presentan. En cuanto a ejemplos, sin ir más lejos, la Biblioteca de la Universidad de Alicante que ofrece un servicio electrónico de préstamo de libros.

5.2. Relación Proyecto Autónomos

5.2.1. Modelos Posibles

De entre los modelos anteriormente podríamos considerar para nuestro proyecto:

- **Suscripción:** Considerando la suscripción el usuario (en este caso el autónomo) deberá pagar una cantidad por el uso de la aplicación funcional durante un tiempo limitado, normalmente para software de este tipo sería de 1 año.
- **Membresía:** Otra opción a la suscripción es la membresía. Esto sería similar a la suscripción, pero con el matiz de que con este modelo el usuario formaría parte de una comunidad en la recibiría las ventajas de ello, como la interacción con otros miembros.
- **Freemium – Premium (publicidad):** Esta opción se refiere a la posibilidad de hacer entrega al cliente de la aplicación de forma gratuita, pero con publicidad insertada con lo que se ofrecerá la posibilidad de pagar una cuantía por la versión de la aplicación sin publicidad. En el caso de nuestra aplicación para autónomos haríamos entrega de la aplicación gratis, aunque mostrando publicidad durante su uso y en el momento en el que el cliente quiera una versión sin publicidad deberá pagar una cantidad.
- **Micropagos:** Este modelo consistiría en hacer disponible la aplicación de forma gratuita, pero sin algunas funcionalidades o aspectos por las que se deberá pagar una pequeña cantidad por cada una de ellas, “fraccionando” de esta forma la aplicación. Enfocado a nuestro proyecto se trataría de hacer llegar la aplicación de forma gratuita sin algunas funcionalidades, así como la posibilidad de relacionarse con otros usuarios, por las que deberán pagar una pequeña cantidad para poder disponer de la aplicación completamente funcional. Este modelo podrá ir ligado al Freemium.
- **Gamificación:** Se podría contemplar la posibilidad de intercambiar comentarios entre usuarios en cuanto a aspectos de su actividad, o inventar un sistema de puntuación por el que opinar y así incentivar las relaciones entre autónomos. Este modelo se podría considerar ligado al modelo de membresía, dado que de esta forma se le da valor añadido a la existencia de una comunidad.

5.2.2. Consideraciones

Sería interesante que nuestro proyecto sea software privativo, concretamente software comercial, ya que para que el autónomo pueda contar con el derecho de uso y acceso a todas las utilidades que el programa presenta. Será necesario que abone una licencia anual.

Teniendo en cuenta esto para nuestro proyecto de una aplicación web para autónomos vería la posibilidad de usar el modelo de negocio de la membresía dadas sus ventajas de la existencia de una comunidad, además de que se podría combinar con la gamificación.

Se deberán de redactar ciertos puntos para la consulta del usuario y se pedirá además que se lean detenidamente antes de continuar con la obtención del software, ya que establece las condiciones de uso:

1. CONCESIÓN DE LA LICENCIA: Se le otorga el derecho limitado, no exclusivo y no transferible del uso de la aplicación al usuario.
2. DESCRIPCIÓN DE LOS DERECHOS Y LAS LIMITACIONES: Enumeración de lo que el usuario en su condición no debe hacer.
3. DERECHOS DE PATENTE; PROPIEDAD: Todos los derechos, títulos e intereses incluidos y relacionados con la aplicación, así como cualquier copia de ellos, son propiedad exclusiva de la compañía que proporciona el software o de sus proveedores.
4. VIGENCIA; TERMINACIÓN DEL CONTRATO: El contrato permanecerá en vigor hasta su finalización además se informa de las obligaciones del usuario en el caso de incumplimiento alguno de las condiciones del contrato.

5. AUSENCIA DE GARANTÍAS: Esto quiere decir el tipo de responsabilidad que la compañía asume con respecto a la precisión o la integridad de la información, los textos, los gráficos, los enlaces o cualquier otro elemento que la aplicación pueda contener.
6. LIMITACIÓN DE RESPONSABILIDAD: Se redacta la limitación de la responsabilidad de la compañía en lo referente a diversos casos de daño indirecto o mal uso de la aplicación por parte de usuarios para perjudicar a terceros.
7. USO DE LA INFORMACIÓN: Términos de uso de la información que los usuarios faciliten a la aplicación, por ejemplo, nombre, fecha de nacimiento, email, etc.
8. EXPORTACIÓN: El usuario debe cumplir todas las restricciones y regulaciones de (control de) exportación y reexportación. No debe transferir, promover, fomentar ni autorizar la transferencia de la aplicación o de su información o tecnología subyacente a un país prohibido, ni infringir de cualquier otra manera tales restricciones y regulaciones.
9. LEGISLACIÓN Y JURISDICCIÓN VIGENTES: Se hará referencia a la ley y jurisdicción por la que se debe regir, analizar e interpretar el contrato.
10. ASIGNACIÓN: El usuario no puede asignar ni transferir los derechos ni las obligaciones que se le otorgan en el presente Contrato mediante la aplicación de la ley, ni de ninguna otra manera, sin el consentimiento previo por escrito de la compañía. La compañía puede asignar sus derechos u obligaciones a otras partes sin su consentimiento.
11. SOFTWARE DE TERCEROS: La compañía proporciona el siguiente aviso de derechos de autor para el software de terceros que se incluye en la aplicación. Este aviso lo exige el

respectivo propietario de los derechos de autor y no modifica su licencia para utilizar este software.

Bloque II: Gestión de Proyectos Informáticos.

Tema 9: Profesión Informática.

Grupo 03 de teoría
martes 15:00 – 17:00

Integrantes del equipo:

Alejandro Culiáñez Llorca
Héctor Mateo Pastor Pérez
Wilmer Fabricio Bravo Shuiria
Manuel García Cremades
Víctor Navarro Martínez-Reina



Índice

1.	Introducción	4
2.	Regularización frente a mercado libre	4
2.1.	Profesiones reguladas y no reguladas	4
2.2.	Situación actual	5
2.3.	Ventajas e inconvenientes de la regularización de Ingeniería Informática	5
3.	Papel de los colegios profesionales	7
3.1.	¿Qué son los colegios profesionales?	7
3.2.	Beneficios de estar colegiado	7
3.3.	Los colegios profesionales relacionados con la Ingeniería Informática	8
3.4.	Los colegios profesionales en ingeniería técnica informática	9
3.5.	El convenio entre ANECA, CONCITI y el CCII	9
3.6.	Papel de los colegios profesionales de Ingeniería Informática con respecto a sus colegiados/as	10
3.7.	Requisitos para formar parte del colegio profesional de ingeniería informática	11
4.	Competencias profesionales. Perfiles y competencias	12
4.1.	Competencias profesionales en Ingeniería informática	12
4.1.1.	Competencias básicas	12
4.1.2.	Competencias transversales	13
4.1.3.	Competencias específicas	15
4.2.	Perfiles profesionales de los/as ingenieros/as en Informática	22
5.	Situación europea e internacional de los informáticos	39
5.1.	Estudiar Ingeniería Informática en España y en el extranjero	39
5.2.	Situación profesional nacional en Ingeniería Informática	39
5.3.	Índice de colocación de los egresados de la titulación	40
5.4.	Empleos más demandados a nivel nacional	40
5.5.	Situación profesional internacional en Ingeniería Informática	40
5.6.	Empleos más demandados a nivel internacional	40
6.	Responsabilidad	41
6.1.	Instalación y uso de software	41
6.2.	Garantía software	42
6.3.	Venta de software	43
6.4.	Desarrollo de software	44
6.5.	Pérdida de información	45
7.	Ética informática	47
7.1.	Definición	47

7.2.	Historia de la ética informática	47
7.3.	Discusión actual	48
7.4.	Responsabilidades como profesionales	48
7.4.1.	Sociedad	48
7.4.2.	Cliente y empresario	49
7.4.3.	Producto	49
7.4.4.	Persona	49
7.5.	Ejemplos cuestiones éticas	50
7.6.	Cuestiones éticas en nuestro proyecto	50
8.	Bibliografía	51

1. Introducción

En el presente documento se explicará cual es la situación actual de la profesión de Ingeniero Informático respecto a la regulación de la profesión, el papel que desempeñan los colegios profesionales y más concretamente los colegios relacionados con la ingeniería informática. Despues, definiremos cuales son las competencias y los perfiles de los profesionales de la ingeniería informática. También, mencionaremos cual es la situación europea e internacional de este sector. Y, por último, comentaremos una serie de aspectos relacionados con la responsabilidad del informático y con la ética laboral de estos.

2. Regularización frente a mercado libre

Existe una gran diferencia entre profesiones reguladas y no reguladas. A continuación, se explicará este tipo de profesiones, cual es la situación actual en la profesión de Ingeniero Informático, y qué ventajas e inconvenientes traería la regularización de dicha profesión.

2.1. Profesiones reguladas y no reguladas

En primer lugar, en el mundo laboral, existen dos tipos de profesiones: las reguladas y las no reguladas.

Una **profesión regulada** es aquella que, para tener acceso a ella, o a alguna de sus modalidades, es necesario directa o indirectamente estar en posesión de un título académico o superar la prueba de aptitud correspondiente. Tienen como **objetivo** garantizar la protección de la ciudadanía que recibe estos servicios profesionales. Algunas de estas profesiones reguladas son accesibles mediante el título universitario de Grado, mientras que en otras profesiones es necesario estar en posesión de un Máster. Además, para ejercicio la profesión será necesario estar colegiado en el Colegio Profesional correspondiente.

Una profesión libre es aquella para la no existe una regulación concreta, y por consecuente, no es necesario estar en posesión de ningún título, ni superar ninguna prueba de aptitud, ni estar colegiado para acceder a ésta.

2.2. Situación actual

La situación actual de la profesión de Ingeniería Informática dentro del territorio español es especialmente impactante.

En el año 2008, tras la aprobación del Real Decreto 1837/2008, se incorpora al ordenamiento jurídico español la Directiva 2005/36/CE, del Parlamento Europeo, relativa al reconocimiento de cualificaciones profesionales.

Esta directiva europea establece que las profesiones reguladas son aquellas actividades o conjunto de actividades profesionales para cuyo acceso, ejercicio de alguna de sus modalidades de ejercicio es necesario directa o indirectamente estar en posesión de un título y constituyan una profesión en un Estado miembro de la Unión Europea.

En el Anexo VIII de este Real Decreto es donde se recoge la relación de profesiones y actividades que se benefician de dicha regulación en el territorio español. En esta lista aparece un gran número de ingenierías, pero la Ingeniería Informática no está entre ellas. También, se establece que las profesiones y actividades no reguladas se entiende que son de ejercicio libre, y, por tanto, no requieren ningún reconocimiento.

En 2015, tras varias charlas en universidades y manifestaciones delante del Ministerio por parte de los profesionales del sector y gran asistencia del sector estudiantil, el Congreso de los Diputados aprobó por unanimidad una proposición no de ley en la que se rogaba al Gobierno a instaurar una regulación a la profesión de Ingeniería Informática. A pesar de ello, desde entonces no se han realizado acciones para solventar esta situación.

Como consecuencia de todo esto, **la profesión de Ingeniera Informática no está actualmente regulada en España y es de ejercicio libre.**

2.3. Ventajas e inconvenientes de la regularización de Ingeniería Informática

La regularización de la profesión de Ingeniería Informática traería una serie de ventajas para unos, pero también una seria de desventajas para otros. Esto es así porque no todos los que actualmente están trabajando como Ingeniero Informático tienen título oficial que les acredite como profesional en caso de que hubiera una regularización de la profesión.

Como primer punto, hablaremos sobre el **reconocimiento** profesional. Hoy en día el ingeniero/a informático apenas goza de reconocimiento profesional. Se piensa que realizar un programa es relativamente sencillo, pero las personas que han estudiado la carrera o quienes son grandes expertos saben que para realizar un programa es necesario tomar una serie de decisiones no arbitrarias que pueden marcar el correcto funcionamiento de ese software. Concretamente, si se regularizara la profesión, se obtendrían programas de mayor calidad y por consecuente la profesión gozaría de un mayor reconocimiento.

Después, podemos hablar sobre el **contenido del grado de Ingeniería Informática** en las universidades. Actualmente, a las universidades españolas se les permite diseñar el plan de estudios y los contenidos del grado a su manera, y, por lo tanto, puede que en una universidad

no se esté enseñando lo mismo que en otras a pesar de pertenecer al mismo grado. Si se regularizara la profesión, existiría unos mismos contenidos para todo el alumnado y la preparación de estos sería más equilibrada.

También, el **intrusismo** laboral está muy acentuado en esta profesión. Personas que no se hayan en propiedad del título de Ingeniero Informático ejercen como tal. Este aspecto es determinante a la hora de la contratación de personal en las diferentes empresas, ya que el sueldo de un ingeniero informático es superior al de otra persona sin título. Si se hiciera la regularización el ingeniero se vería afectado positivamente, y el intruso se vería afectado negativamente.

Por otro lado, existe una serie de **puestos de trabajo como ingeniero informático en la administración pública** que no requiere de la titulación específica. Si se realizase la regularización, el ingeniero informático tendría más posibilidades de obtener el trabajo ya que habría menos demandantes del puesto.

Otro aspecto importante que debemos tener en cuenta sería el de la **movilidad internacional**. Si en España se regularizase la profesión, resultaría más fácil acceder a un puesto de trabajo, sobre todo dentro de la Unión Europea, en un país donde también éste regularizado.

Por último, según los expertos, debido al alto intrusismo laboral existente en la profesión, la regularización laboral podría traer consecuencias negativas al sector desembocando en una crisis de este.

3. Papel de los colegios profesionales

3.1. ¿Qué son los colegios profesionales?

Los colegios profesionales son corporaciones de derecho público formadas por personas de una misma titulación. Los miembros que la constituyen son llamados colegiados.

Se crearon con el objetivo de tener un control de la actividad profesional, asegurando que los profesionales cumplan una serie de normas que permitieran a las personas poder ejercer sus derechos con plena garantía.

Hay que destacar que la pertenencia a un colegio profesional no es obligatoria y se considera voluntaria para ciertas titulaciones, como el colegio de periodistas, y cuya finalidad es la de respaldar a sus colegiados. En el caso de la colegiación en el colegio de ingeniería técnica en informática (la que nos habilita el título de grado en ingeniería informática) de la comunidad valenciana, si es obligatorio la inscripción en el colegio para ejercitar la profesión, siempre que se tenga domicilio profesional principal en la comunidad valencia. Se excluyen a los profesionales que trabajan en administraciones públicas.

Los colegios profesionales están encargados de regular la práctica de su profesión garantizando a los clientes que el colegiado cumple con una serie de estándares que le aporta su formación, además, tienen como tarea la defensa de sus colegiados y el asegurarse que cumplen un código de conducta.

3.2. Beneficios de estar colegiado

Una de las ventajas claras de pertenecer a un colegio profesional es el de demostrar tu formación profesional y que cumples con los estándares para realizar una actividad profesional. Otra ventaja es la de dar a conocer al público tu profesión y los beneficios que esta aporta a la sociedad.

Además de estas ventajas los colegios profesionales ofrecen a sus miembros una serie de servicios como:

- Dar a conocer salidas profesionales y participar en una bolsa de empleo.
- Inscribirse en el TAP, donde podrás prestar tus servicios a otras personas o empresas.
- Complementos para la formación, con descuentos a los miembros colegiados.
- Certificado de títulos y acreditación de cualificaciones.
- Asesoría laboral y fiscal.
- Dar a conocer eventos relacionados con la profesión.
- Descuentos en algunos servicios como seguros, bancos, compras de material propio de la profesión, etc.
- Espacios de trabajo para desarrollar tu actividad.
- Acceso a publicaciones sobre tu profesión.

Por último, debemos mencionar, que el colegio se encarga de la defensa y protección de sus miembros, lo cual consideramos una ventaja.

3.3. Los colegios profesionales relacionados con la Ingeniería Informática

Antes de comenzar a tratar los distintos colegios profesionales en España, debemos señalar que existe una organización colegial compuesta por un **consejo general de colegios oficiales de ingeniería en Informática (CCII)** y los distintos **colegios autonómicos de ingeniería informática** de España.

El **CCII**, es la entidad que representa a la profesión de ingeniería informática en el ámbito nacional, integrando los distintos colegios autonómicos de la ingeniería informática españoles. Fue constituida en el 2010 y aprobada en el congreso de los diputados.

El **CCII** está regido por una serie de organismos que citaremos textualmente de su misma página web: “*CCII está regido por Constitución Española, la Ley 2/1974, de 13 de febrero, sobre Colegios Profesionales, por la legislación estatal básica o de aplicación directa o general que afecte a los colegios y consejos profesionales y por los Estatutos Generales de los Colegios Oficiales de Ingeniería en Informática y de su Consejo General, aprobados en el Real Decreto 518/2015, de 19 de junio, publicados en el Boletín Oficial del Estado de 11 de julio de 2015.*”

Los distintos **colegios oficiales con ámbito autonómico de la profesión de ingeniería informática** y que constituyen el **CCII** son:

- **CPIIA** - Colegio Profesional de Ingenieros en Informática de Andalucía.
- **COIIPA** - Colegio Oficial de Ingenieros en Informática del Principado de Asturias.
- **COIIE** - Colegio Oficial de Ingenieros en Informática del País Vasco.
- **COIICLM** - Colegio Oficial de Ingenieros en Informática de Castilla-La Mancha.
- **COEINF** - Colegio Oficial de Ingenieros en Informática de Catalunya.
- **CPIIRM** - Colegio Profesional de Ingenieros en Informática de la Región de Murcia.
- **COIICV** - Colegio Oficial de Ingenieros en Informática de la Comunidad Valenciana.
- **CPIICyL** - Colegio Profesional de Ingenieros en Informática de Castilla y León.
- **CPIIEX** - Colegio Profesional de Ingenieros en Informática de Extremadura.
- **CPEIG** - Colegio Profesional de Ingenieros en Informática de Galicia.
- **COEIB** - Colegio Oficial de Ingeniería en Informática de las Islas Baleares.
- **CPIICM** - Colegio Profesional de Ingenieros en Informática de la Comunidad de Madrid.
- **CPIIR** - Colegio Profesional de Ingenieros en Informática de La Rioja.
- **CPIINA** - Colegio Profesional de Ingenieros en Informática de Navarra.
- **CPIIC** - Colegio Profesional de Ingenieros en Informática de Cantabria.
- **CPIIAR** - Colegio Profesional de Ingenieros en Informática de Aragón.

Cada una de estas entidades está formada por los ingenieros informáticos e ingenieras informáticas de cada comunidad.

3.4. Los colegios profesionales en ingeniería técnica informática

No hay que olvidar que la titulación de grado en ingeniería informática nos habilita a la realización de la profesión de ingeniería técnica en informática y nos proporciona el acceso a colegio de ingeniería técnica en informática y no al de ingenieros informáticos.

El **Consejo General de Colegios Oficiales de Ingeniería Técnica en Informática (CONCITI)**, es una institución que integra todos los colegios de ingeniería técnica en informática y grado en ingeniería informática. Fue constituido en el 2009 al ser aprobada por el congreso y senado español.

El 11 de julio de 2015, se aprueban por ley en el BOE los Colegios Oficiales de Ingeniería Técnica en Informática y de su Consejo General.

Los distintos **colegios oficiales con ámbito autonómico de ingeniería técnica informática y grado en informática** y que constituyen el CONCITI son:

- Consejo General de Colegios Oficiales de Ingeniería Técnica en Informática.
- Colegio Profesional de Ingenieros Técnicos en Informática y Graduados en Ingeniería Informática de la Región de Murcia.
- Colegio Oficial de Ingeniería informática de Cataluña.
- Colegio Oficial de Ingenieros Técnicos en Informática del Principado de Asturias.
- Colegio Oficial de Ingenieros Técnicos en Informática de Castilla-La Mancha.
- Colegio Oficial de Ingenieros Técnicos en Informática de las Islas Baleares.
- Colegio Profesional de Ingenieros Técnicos en Informática de Andalucía.
- Colegio Oficial de Ingenieros Técnicos en Informática de la Comunidad Valenciana.
- Colegio Profesional de Ingenieros Técnicos en Informática de Canarias.
- Colegio Profesional de Ingenieros Técnicos en Informática de Galicia.
- Colegio Profesional de Ingenieros Técnicos en Informática de la Comunidad de Madrid.
- Colegio Profesional de Ingenieros Técnicos en Informática de La Rioja.
- Colegio Profesional de Ingenieros Técnicos en Informática de Navarra.
- Colegio Profesional de Ingenieros Técnicos en Informática de Cantabria.
- Colegio Profesional de Ingenieros Técnicos en Informática de Aragón.

3.5. El convenio entre ANECA, CONCITI y el CCII

El 18 de noviembre de 2013 **ANECA**, el **CONCITI** y el **CCII** firmaron un convenio para poder acreditar títulos oficiales de Grado y Máster del ámbito de la informática cumplieran los estándares de la **EQANIE (European Quality Assurance Network for Informatics Education)** y concederles de esta forma la mención **EURO-INF** en el marco del programa «Acredita Plus» de ANECA.

La **ANECA** es el órgano encargado de realizar actividades de evaluación, certificación y acreditación del sistema universitario español.

El **EURO-INF** es una mención que reconoce estudios que cumplen con las exigencias profesionales en el ámbito de la informática, según una serie de estándares definidos, de acuerdo con los principios de calidad, relevancia, transparencia, reconocimiento y movilidad contemplados por la **EQANIE**.

ANECA obtuvo la autorización para poder otorgar el sello **EURO-INF** por parte de la **EQANIE** el 13 de octubre de 2014, autorización vigente hasta el 12 de octubre de 2019.

Este convenio fue renovado en el BOE el 27 de abril de 2018.

3.6. Papel de los colegios profesionales de Ingeniería Informática con respecto a sus colegiados/as

Podemos resumir las finalidades de cada uno de estos colegios profesionales de informática en las siguientes:

1. **Representación y ordenación** de la profesión en el ámbito autonómico.
2. Asegurar el **cumplimiento de los preceptos constitucionales, ética profesional y normas deontológicas** en el uso de la informática.
3. **Defensa** de los colegiados.
4. **Asegurar el nivel de calidad** a través de mejoras en la formación de los colegiados.
5. **Fomentar la investigación y el desarrollo e innovación (I+D+i)**.

Podemos resumir las funciones del **CONCITI** y el **CCII** respecto a sus colegios profesionales y siempre que no se entre conflicto con estos:

1. **Cumplir y hacer cumplir a los colegiados** las leyes, estatutos y reglamentos de régimen, así como las normas y decisiones adoptadas por el colegio.
2. **Realización de estudios**, emisión de informes, elaboración de estadísticas y otras actividades encomendadas por la administración pública.
3. Participar en los órganos consultivos de la administración **en representación de la profesión**.
4. **Participar en la elaboración de los planes de estudio y organizar actividades formativas y culturales**.
5. **Velar por la ética, dignidad profesional y por el respeto**.
6. **Evitar el intrusismo profesional**.
7. **Resolver discrepancias** que puedan surgir sobre el cumplimiento de las obligaciones por parte de los colegiados en algún trabajo.
8. **Atender las solicitudes de información** sobre los colegiados (por ejemplo, en el caso de inspecciones para sancionar a integrantes).

3.7. Requisitos para formar parte del colegio profesional de ingeniería informática

Para colegiarse es necesario tener alguno de los siguientes títulos:

- Licenciado en informática.
- Máster en ingeniería informática, reconocido por **ANECA** como título habilitante para ejercer la profesión de Ingeniero en Informática.
- Título extranjero equivalente, debidamente convalidado.

Por tanto, el grado en ingeniería informática da acceso a la profesión de ingeniero técnico en ingeniería informática y a la colegiación en el colegio de ingeniería técnica en informática y no permite la colegiación en los colegios profesionales de la ingeniería informática. Por otra parte, el grado si habilita a cursar el máster en ingeniería informática que es la titulación asociada a la profesión de ingeniero en informática y a la colegiación en el colegio oficial de ingenieros informáticos.

Puede **precolegiarse** cuando se matricule de 4º de Grado o 1º de Máster (habilitante) de cualquier universidad española.

En el **BOE de 4 de agosto de 2009** se indican cuáles son las competencias que se adquieren en las titulaciones que dan acceso a ambas profesiones.

4. Competencias profesionales. Perfiles y competencias

4.1. Competencias profesionales en Ingeniería informática

Se entiende por “competencia” la definición establecida por el Marco Europeo de Cualificaciones (European Qualification Framework - EQF) como la capacidad demostrada para utilizar conocimientos, destrezas y habilidades personales, sociales y metodológicas, en situaciones de trabajo o estudio y en el desarrollo profesional y personal.

Las últimas leyes a las que hacen referencia las competencias del profesional de la ingeniería informática son las siguientes:

- Acuerdo del Consejo de Universidades, sobre la propuesta por las universidades de memorias de solicitud de títulos oficiales en el ámbito de la Ingeniería Informática e Ingeniería Técnica Informática, BOE de 9 de agosto de 2009.
- Ley 20/2009, de 4 de diciembre, de creación del Consejo General de Colegios Oficiales de Ingeniería en Informática, BOE de 5 de diciembre de 2009.
- Real Decreto 518/2015, de 19 de junio, por el que se aprueban los Estatutos Generales de los Colegios Oficiales de Ingeniería en Informática y de su Consejo General, BOE de 11 de julio de 2015.

Realizar una mención a que la mayoría de las competencias específicas han sido obtenidas en el Consejo General de Colegios Profesionales – CCII, ya que prácticamente su fecha de creación coincide con la implantación de los grados en las universidades, con lo que las competencias de la UA en lo referente a la Ingeniería informática son prácticamente las mismas que las del CCII.

Tipos de competencias

Existen tres tipos de competencias:

- Básicas (para todos los trabajadores, obligatoria)
- Transversales (para todos los trabajadores, recomendable)
- Técnicas (específicas para un tipo de trabajadores, como nuestra profesión). Dentro de estas también se pueden dividir entre básicas, transversales y técnicas (las cuales harán referencia a las especialidades).

4.1.1. Competencias básicas

Son los conocimientos básicos que permiten acceder a cualquier trabajo.

Ejemplos de competencias básicas:

- Comunicación oral y escrita: Necesario para el buen entendimiento de las partes emisor-receptor a la hora de comunicarnos con otras personas, sea físicamente u online.
- Resolución de problemas: Poseer las competencias que se demuestren por medio de la elaboración y defensa de argumentos y la resolución de problemas dentro de su área de trabajo.

- Lengua materna: Conocimiento y dominio de la lengua materna.
- Educación básica: Saber aplicar los conocimientos adquiridos básicos de una forma profesional.

4.1.2. Competencias transversales

Son las habilidades y aptitudes que hacen que un trabajador desarrolle de forma eficaz su trabajo. Sirven para desarrollar cualquier profesión y que se hayan adquirido en diferentes contextos, sean laborales o no.

Algunas de estas competencias hacen referencia a conductas y actitudes de las personas. Son transversales porque afectan a muchos puestos de trabajo, no solo a un tipo determinado como pasa con las competencias técnicas.

Ejemplos de competencias transversales:

- Dominio de un idioma extranjero: A la hora de trabajar en una empresa, normalmente trabajará con otras empresas o clientes que sean de otras nacionalidades, con lo que necesitará de otro idioma vehicular para realizar correctamente la comunicación. Normalmente este idioma es el **inglés**.
- Análisis numérico: Capacidad para analizar, organizar y resolver cuestiones numéricas, datos financieros, estadísticas y similares.
- Capacidad de resolución de problemas: Disposición y habilidad para enfrentarse y dar solución a una situación determinada mediante la organización y/o aplicación de una estrategia o secuencia operativa (identificación del problema, diagnóstico, formulación de soluciones y evaluación) definida o no para encontrar la solución.
- Capacidad de organización del trabajo: Disposición y habilidad para crear las condiciones adecuadas de utilización de los recursos humanos o materiales existentes para desarrollar las tareas con el máximo de eficacia y eficiencia.
- Capacidad de trabajar en equipo: Disposición y habilidad para colaborar de manera coordinada en la tarea realizada conjuntamente por un equipo de personas para conseguir un objetivo propuesto.
- Responsabilidad en el trabajo: Disposición para implicarse en el trabajo, considerándola la expresión de la competencia profesional y personal y cuidando de que el funcionamiento de los recursos humanos y materiales sea el adecuado.
- Autonomía: Capacidad de realizar una tarea de forma independiente, ejecutándola de principio a fin, sin necesidad de recibir ninguna ayuda o apoyo. Esta capacidad de trabajar de forma autónoma no quiere decir, no obstante, que en ciertas etapas o tareas concretas no pueda ser asesorado.
- Capacidad de iniciativa: Habilidad y disposición para tomar decisiones sobre propuestas o acciones.
- Capacidad de aprendizaje: Habilidad y disposición para aprender nuevos conocimientos que permitan el desarrollo correcto de la actividad en la empresa.
- Capacidad de innovación: Habilidad y disposición para mejorar el proceso productivo, el servicio a los clientes o el producto.
- Relación interpersonal: Capacidad para comunicarse con los demás utilizando un trato adecuado, prestando atención y con simpatía.

- Adaptabilidad: Capacidad para permanecer eficaz dentro de un medio cambiante, así como a la hora de enfrentarse con nuevas tareas, retos y personas.
- Automotivación: Se traduce en la importancia de trabajar por satisfacción personal. Necesidad alta de alcanzar un objetivo con éxito.
- Atención al cliente: Detectar las expectativas del cliente, asumiendo compromiso en la identificación de cualquier problema y proporcionar las soluciones más idóneas para satisfacer sus necesidades.
- Control: Capacidad para tomar decisiones que aseguren el control sobre métodos, personas y situaciones.
- Capacidad crítica: Habilidad para la evaluación de datos y líneas de acción para conseguir tomar decisiones lógicas de forma imparcial y razonada.
- Creatividad: Capacidad para proponer soluciones imaginativas y originales. Innovación e identificación de alternativas contrapuestas a los métodos y enfoques tradicionales.
- Comunicación verbal y no verbal persuasiva: Capacidad para expresarse claramente y de forma convincente con el fin de que la otra persona asuma nuestros argumentos como propios.
- Delegación: Distribución eficaz de la toma de decisiones y responsabilidades hacia el subordinado más adecuado.
- Desarrollo de subordinados: Potenciar las habilidades de las personas a nuestro cargo mediante la realización de actividades (actuales y futuras).
- Tolerancia al estrés: Mantenimiento firme del carácter ante acumulación de tareas o responsabilidades, lo cual se traduce en respuestas controladas frente a un exceso de cargas.
- Espíritu comercial: Capacidad para entender aquellos asuntos del negocio que afectan a la rentabilidad y crecimiento de una empresa con el fin de maximizar el éxito.
- Escucha: Capacidad para detectar la información importante de la comunicación oral. Recurriendo, si fuese necesario, a las preguntas y a los diferentes tipos de comunicación.
- Energía: Capacidad para crear y mantener un nivel de actividad adecuado. Muestra el control, la resistencia y la capacidad de trabajo.
- Flexibilidad: Capacidad para modificar el comportamiento adoptar un tipo diferente de enfoque sobre ideas o criterios.
- Independencia: Actuación basada en las propias convicciones sin deseo de agradar a terceros, en cualquier caso. Disposición para poner en duda un criterio o línea de acción.
- Integridad: Capacidad para mantenerse dentro de una organización o grupo para realizar actividades o participar en ellos.
- Impacto: Causar buena impresión a otros que perdure en el tiempo.
- Liderazgo: Utilización de los rasgos y métodos interpersonales para guiar a individuos o grupos hacia la consecución de un objetivo.
- Meticulosidad: Resolución total de una tarea o asunto, de todas sus áreas y elementos, independientemente de su insignificancia.
- Niveles de trabajo: Establecimiento de grandes metas u objetivos para uno mismo, para otros o para la empresa. Insatisfacción como consecuencia de bajo rendimiento.
- Planificación y organización: Capacidad para realizar de forma eficaz un plan apropiado de actuación personal o para terceros con el fin de alcanzar un objetivo.
- Resistencia: Capacidad para mantenerse eficaz en situaciones de rechazo.

- Sensibilidad organizacional: Capacidad para percibir e implicarse en decisiones y actividades en otras partes de la empresa.
- Sensibilidad interpersonal: Conocimiento de los otros, del grado de influencia personal que se ejerce sobre ellos. Las actuaciones indican el conocimiento de los sentimientos y necesidades de los demás.
- Sociabilidad: Capacidad para mezclarse fácilmente con otras personas. Abierto y participativo.
- Tenacidad: Capacidad para perseverar en un asunto o problema hasta que quede resuelto o hasta comprobar que el objetivo no es alcanzable de forma razonable.

4.1.3. Competencias específicas

Actualmente, los Ingenieros Informáticos deben tener una serie de competencias a la hora de desempeñar sus funciones:

Competencias básicas en Ingeniería Informática

Son los conocimientos básicos que permiten acceder a un trabajo como Ingeniero Informático.

Estos son algunos ejemplos de nuestra profesión:

- Capacidad para la resolución de los problemas matemáticos que puedan plantearse en la ingeniería. Aptitud para aplicar los conocimientos sobre: álgebra lineal, cálculo diferencial e integral, métodos numéricos, algorítmica numérica, estadística y optimización.
- Comprensión y dominio de los conceptos básicos de campos y ondas y electromagnetismo, teoría de circuitos eléctricos, circuitos electrónicos, principio físico de los semiconductores y familias lógicas, dispositivos electrónicos y fotónicos, y su aplicación para la resolución de problemas propios de la ingeniería.
- Capacidad para comprender y dominar los conceptos básicos de matemática discreta, lógica, algorítmica y complejidad computacional, y su aplicación para la resolución de problemas propios de la ingeniería.
- Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería.
- Conocimiento de la estructura, organización, funcionamiento e interconexión de los sistemas informáticos, los fundamentos de su programación, y su aplicación para la resolución de problemas propios de la ingeniería.
- Conocimiento adecuado del concepto de empresa, marco institucional y jurídico de la empresa. Organización y gestión de empresas.
- Tener la capacidad de reunir e interpretar datos relevantes para emitir juicios que incluyan una reflexión sobre temas relevantes de índole social, científica o ética.
- Capacidad para transmitir información, ideas, problemas y soluciones a un público tanto especializada como no especializado.
- Competencias básicas en el uso de las TIC.

Competencias técnicas en Ingeniería Informática

Son las competencias adquiridas después de realizar una formación específica y son específicas de un puesto de trabajo en concreto. Aparte de las comunes a todos los profesionales de informática, el resto lo dividiremos en cinco bloques, tal como lo tiene establecido la UA: Ingeniería del Software, Ingeniería de Computadores, Computación, Sistemas de Información y tecnologías de la Información.

Competencias técnicas comunes

- Capacidad para diseñar, desarrollar, seleccionar y evaluar aplicaciones y sistemas informáticos, asegurando su fiabilidad, seguridad y calidad, conforme a principios éticos y a la legislación y normativa vigente.
- Capacidad para planificar, concebir, desplegar y dirigir proyectos, servicios y sistemas informáticos en todos los ámbitos, liderando su puesta en marcha y su mejora continua y valorando su impacto económico y social.
- Capacidad para comprender la importancia de la negociación, los hábitos de trabajo efectivos, el liderazgo y las habilidades de comunicación en todos los entornos de desarrollo de software.
- Capacidad para elaborar el pliego de condiciones técnicas de una instalación informática que cumpla los estándares y normativas vigentes.
- Conocimiento, administración y mantenimiento sistemas, servicios y aplicaciones informáticas.
- Conocimiento y aplicación de los procedimientos algorítmicos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos.
- Conocimiento, diseño y utilización de forma eficiente de los tipos y estructuras de datos más adecuados a la resolución de un problema.
- Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.
- Capacidad de conocer, comprender y evaluar la estructura y arquitectura de los computadores, así como los componentes básicos que los conforman.
- Conocimiento de las características, funcionalidades y estructura de los Sistemas Operativos y diseñar e implementar aplicaciones basadas en sus servicios.
- Conocimiento y aplicación de las características, funcionalidades y estructura de los Sistemas Distribuidos, las Redes de Computadores e Internet y diseñar e implementar aplicaciones basadas en ellas.
- Conocimiento y aplicación de las características, funcionalidades y estructura de las bases de datos, que permitan su adecuado uso, y el diseño y el análisis e implementación de aplicaciones basadas en ellos.
- Conocimiento y aplicación de las herramientas necesarias para el almacenamiento, procesamiento y acceso a los Sistemas de información, incluidos los basados en web.
- Conocimiento y aplicación de los principios fundamentales y técnicas básicas de la programación paralela, concurrente, distribuida y de tiempo real.
- Conocimiento y aplicación de los principios fundamentales y técnicas básicas de los sistemas inteligentes y su aplicación práctica.
- Conocimiento y aplicación de los principios, metodologías y ciclos de vida de la ingeniería de software.
- Capacidad para diseñar y evaluar interfaces persona computador que garanticen la accesibilidad y usabilidad a los sistemas, servicios y aplicaciones informáticas.

- Conocimiento de la normativa y la regulación de la informática en los ámbitos nacional, europeo e internacional.

Competencias técnicas de Ingeniería del Software

- Capacidad para desarrollar, mantener y evaluar servicios y sistemas software que satisfagan todos los requisitos del usuario y se comporten de forma fiable y eficiente, sean asequibles de desarrollar y mantener y cumplan normas de calidad, aplicando las teorías, principios, métodos y prácticas de la Ingeniería del Software.
- Capacidad para valorar las necesidades del cliente y especificar los requisitos software para satisfacer estas necesidades, reconciliando objetivos en conflicto mediante la búsqueda de compromisos aceptables dentro de las limitaciones derivadas del coste, del tiempo, de la existencia de sistemas ya desarrollados y de las propias organizaciones.
- Capacidad de dar solución a problemas de integración en función de las estrategias, estándares y tecnologías disponibles.
- Capacidad de identificar y analizar problemas y diseñar, desarrollar, implementar, verificar y documentar soluciones software sobre la base de un conocimiento adecuado de las teorías, modelos y técnicas actuales.
- Capacidad de identificar, evaluar y gestionar los riesgos potenciales asociados que pudieran presentarse.
- Capacidad para diseñar soluciones apropiadas en uno o más dominios de aplicación utilizando métodos de la ingeniería del software que integren aspectos éticos, sociales, legales y económicos.

Competencias técnicas de Ingeniería de Computadores

- Capacidad de diseñar y construir sistemas digitales, incluyendo computadores, sistemas basados en microprocesador y sistemas de comunicaciones.
- Capacidad de desarrollar procesadores específicos y sistemas empotrados, así como desarrollar y optimizar el software de dichos sistemas.
- Capacidad de analizar y evaluar arquitecturas de computadores, incluyendo plataformas paralelas y distribuidas, así como desarrollar y optimizar software para las mismas.
- Capacidad de diseñar e implementar software de sistema y de comunicaciones.
- Capacidad de analizar, evaluar y seleccionar las plataformas hardware y software más adecuadas para el soporte de aplicaciones empotradas y de tiempo real.
- Capacidad para comprender, aplicar y gestionar la garantía y seguridad de los sistemas informáticos.
- Capacidad para analizar, evaluar, seleccionar y configurar plataformas hardware para el desarrollo y ejecución de aplicaciones y servicios informáticos.
- Capacidad para diseñar, desplegar, administrar y gestionar redes de computadores.

Competencias técnicas de Computación

- Capacidad para tener un conocimiento profundo de los principios fundamentales y modelos de la computación y saberlos aplicar para interpretar, seleccionar, valorar, modelar, y crear nuevos conceptos, teorías, usos y desarrollos tecnológicos relacionados con la informática.
- Capacidad para conocer los fundamentos teóricos de los lenguajes de programación y las técnicas de procesamiento léxico, sintáctico y semántico asociadas, y saber aplicarlas para la creación, diseño y procesamiento de lenguajes.

- Capacidad para evaluar la complejidad computacional de un problema, conocer estrategias algorítmicas que puedan conducir a su resolución y recomendar, desarrollar e implementar aquella que garantice el mejor rendimiento de acuerdo con los requisitos establecidos.
- Capacidad para conocer los fundamentos, paradigmas y técnicas propias de los sistemas inteligentes y analizar, diseñar y construir sistemas, servicios y aplicaciones informáticas que utilicen dichas técnicas en cualquier ámbito de aplicación.
- Capacidad para adquirir, obtener, formalizar y representar el conocimiento humano en una forma computable para la resolución de problemas mediante un sistema informático en cualquier ámbito de aplicación, particularmente los relacionados con aspectos de computación, percepción y actuación en ambientes o entornos inteligentes.
- Capacidad para desarrollar y evaluar sistemas interactivos y de presentación de información compleja y su aplicación a la resolución de problemas de diseño de interacción persona computadora.
- Capacidad para conocer y desarrollar técnicas de aprendizaje computacional y diseñar e implementar aplicaciones y sistemas que las utilicen, incluyendo las dedicadas a extracción automática de información y conocimiento a partir de grandes volúmenes de datos.

Competencias técnicas de Sistemas de Información

- Capacidad de integrar soluciones de Tecnologías de la Información y las Comunicaciones y procesos empresariales para satisfacer las necesidades de información de las organizaciones, permitiéndoles alcanzar sus objetivos de forma efectiva y eficiente, dándoles así ventajas competitivas.
- Capacidad para determinar los requisitos de los sistemas de información y comunicación de una organización atendiendo a aspectos de seguridad y cumplimiento de la normativa y la legislación vigente.
- Capacidad para participar activamente en la especificación, diseño, implementación y mantenimiento de los sistemas de información y comunicación.
- Capacidad para comprender y aplicar los principios y prácticas de las organizaciones, de forma que puedan ejercer como enlace entre las comunidades técnica y de gestión de una organización y participar activamente en la formación de los usuarios.
- Capacidad para comprender y aplicar los principios de la evaluación de riesgos y aplicarlos correctamente en la elaboración y ejecución de planes de actuación.
- Capacidad para comprender y aplicar los principios y las técnicas de gestión de la calidad y de la innovación tecnológica en las organizaciones.

Competencias técnicas de Tecnologías de la Información

- Capacidad para comprender el entorno de una organización y sus necesidades en el ámbito de las tecnologías de la información y las comunicaciones.
- Capacidad para seleccionar, diseñar, desplegar, integrar, evaluar, construir, gestionar, explotar y mantener las tecnologías de hardware, software y redes, dentro de los parámetros de coste y calidad adecuados.
- Capacidad para emplear metodologías centradas en el usuario y la organización para el desarrollo, evaluación y gestión de aplicaciones y sistemas basados en tecnologías de la información que aseguren la accesibilidad, ergonomía y usabilidad de los sistemas.
- Capacidad para seleccionar, diseñar, desplegar, integrar y gestionar redes e infraestructuras de comunicaciones en una organización.

- Capacidad para seleccionar, desplegar, integrar y gestionar sistemas de información que satisfagan las necesidades de la organización, con los criterios de coste y calidad identificados.
- Capacidad de concebir sistemas, aplicaciones y servicios basados en tecnologías de red, incluyendo Internet, web, comercio electrónico, multimedia, servicios interactivos y computación móvil.
- Capacidad para comprender, aplicar y gestionar la garantía y seguridad de los sistemas informáticos.

Competencias técnicas de Tecnologías Informáticas

- Capacidad para modelar, diseñar, definir la arquitectura, implantar, gestionar, operar, administrar y mantener aplicaciones, redes, sistemas, servicios y contenidos informáticos.
- Capacidad para comprender y saber aplicar el funcionamiento y organización de Internet, las tecnologías y protocolos de redes de nueva generación, los modelos de componentes, software intermediario y servicios.
- Capacidad para asegurar, gestionar, auditar y certificar la calidad de los desarrollos, procesos, sistemas, servicios, aplicaciones y productos informáticos.
- Capacidad para diseñar, desarrollar, gestionar y evaluar mecanismos de certificación y garantía de seguridad en el tratamiento y acceso a la información en un sistema de procesamiento local o distribuido.
- Capacidad para analizar las necesidades de información que se plantean en un entorno y llevar a cabo en todas sus etapas el proceso de construcción de un sistema de información.
- Capacidad para diseñar y evaluar sistemas operativos y servidores, y aplicaciones y sistemas basados en computación distribuida.
- Capacidad para comprender y poder aplicar conocimientos avanzados de computación de altas prestaciones y métodos numéricos o computacionales a problemas de ingeniería.
- Capacidad para diseñar y desarrollar sistemas, aplicaciones y servicios informáticos en sistemas empotrados y ubicuos.
- Capacidad para aplicar métodos matemáticos, estadísticos y de inteligencia artificial para modelar, diseñar y desarrollar aplicaciones, servicios, sistemas inteligentes y sistemas basados en el conocimiento.
- Capacidad para utilizar y desarrollar metodologías, métodos, técnicas, programas de uso específico, normas y estándares de computación gráfica.
- Capacidad para conceptualizar, diseñar, desarrollar y evaluar la interacción persona-ordenador de productos, sistemas, aplicaciones y servicios informáticos.
- Capacidad para la creación y explotación de entornos virtuales, y para la creación, gestión y distribución de contenidos multimedia.

Competencias técnicas de Dirección y Gestión

- Capacidad para la integración de tecnologías, aplicaciones, servicios y sistemas propios de la Ingeniería Informática, con carácter generalista, y en contextos más amplios y multidisciplinares.
- Capacidad para la planificación estratégica, elaboración, dirección, coordinación, y gestión técnica y económica en los ámbitos de la ingeniería informática relacionados, entre otros, con: sistemas, aplicaciones, servicios, redes, infraestructuras o instalaciones informáticas y centros o factorías de desarrollo de software, respetando el adecuado

cumplimiento de los criterios de calidad y medioambientales y en entornos de trabajo multidisciplinares.

- Capacidad para la dirección de proyectos de investigación, desarrollo e innovación, en empresas y centros tecnológicos, con garantía de la seguridad para las personas y bienes, la calidad final de los productos y su homologación.

Competencias transversales en Ingeniería Informática

- Capacidad para proyectar, calcular y diseñar productos, procesos e instalaciones en todos los ámbitos de la ingeniería informática.
- Capacidad para la dirección de obras e instalaciones de sistemas informáticos, cumpliendo la normativa vigente y asegurando la calidad del servicio. Capacidad para dirigir, planificar y supervisar equipos multidisciplinares.
- Capacidad para el modelado matemático, cálculo y simulación en centros tecnológicos y de ingeniería de empresa, particularmente en tareas de investigación, desarrollo e innovación en todos los ámbitos relacionados con la Ingeniería en Informática.
- Capacidad para la elaboración, planificación estratégica, dirección, coordinación y gestión técnica y económica de proyectos en todos los ámbitos de la Ingeniería en Informática siguiendo criterios de calidad y medioambientales.
- Capacidad para la dirección general, dirección técnica y dirección de proyectos de investigación, desarrollo e innovación, en empresas y centros tecnológicos, en el ámbito de la Ingeniería Informática.
- Capacidad para la puesta en marcha, dirección y gestión de procesos de fabricación de equipos informáticos, con garantía de la seguridad para las personas y bienes, la calidad final de los productos y su homologación.
- Capacidad para la aplicación de los conocimientos adquiridos y de resolver problemas en entornos nuevos o poco conocidos dentro de contextos más amplios y multidisciplinares, siendo capaces de integrar estos conocimientos.
- Capacidad para comprender y aplicar la responsabilidad ética, la legislación y la deontología profesional de la actividad de la profesión de Ingeniero en Informática.
- Capacidad para aplicar los principios de la economía y de la gestión de recursos humanos y proyectos, así como la legislación, regulación y normalización de la informática.
- Capacidad para concebir, redactar, organizar, planificar, desarrollar y firmar proyectos en el ámbito de la ingeniería en informática que tengan por objeto, de acuerdo con los conocimientos adquiridos según lo establecido en los apartados específicos posteriores, la concepción, el desarrollo o la explotación de sistemas, servicios y aplicaciones informáticas.
- Capacidad para dirigir las actividades objeto de los proyectos del ámbito de la informática de acuerdo con los conocimientos adquiridos según lo establecido en los apartados específicos posteriores.
- Capacidad para diseñar, desarrollar, evaluar y asegurar la accesibilidad, ergonomía, usabilidad y seguridad de los sistemas, servicios y aplicaciones informáticas, así como de la información que gestionan.
- Capacidad para definir, evaluar y seleccionar plataformas hardware y software para el desarrollo y la ejecución de sistemas, servicios y aplicaciones informáticas, de acuerdo con los conocimientos adquiridos según lo establecido en los apartados específicos posteriores.
- Capacidad para concebir, desarrollar y mantener sistemas, servicios y aplicaciones informáticas empleando los métodos de la ingeniería del software como instrumento para el aseguramiento de su calidad, de acuerdo con los conocimientos adquiridos según lo establecido en los apartados específicos posteriores.

- Capacidad para concebir y desarrollar sistemas o arquitecturas informáticas centralizadas o distribuidas integrando hardware, software y redes de acuerdo con los conocimientos adquiridos según lo establecido en los apartados específicos posteriores.
- Capacidad para conocer, comprender y aplicar la legislación necesaria durante el desarrollo de la profesión de Ingeniero Técnico en Informática y manejar especificaciones, reglamentos y normas de obligado cumplimiento.
- Conocimiento de las materias básicas y tecnologías, que capaciten para el aprendizaje y desarrollo de nuevos métodos y tecnologías, así como las que les doten de una gran versatilidad para adaptarse a nuevas situaciones.
- Capacidad para resolver problemas con iniciativa, toma de decisiones, autonomía y creatividad.
- Capacidad para saber comunicar y transmitir los conocimientos, habilidades y destrezas de la profesión de Ingeniero Técnico en Informática.
- Conocimientos para la realización de mediciones, cálculos, valoraciones, tasaciones, peritaciones, estudios, informes, planificación de tareas y otros trabajos análogos de informática, de acuerdo con los conocimientos adquiridos según lo establecido en los apartados específicos posteriores.
- Capacidad para analizar y valorar el impacto social y medioambiental de las soluciones técnicas, comprendiendo la responsabilidad ética y profesional de la actividad del Ingeniero Técnico en Informática.
- Conocimiento y aplicación de elementos básicos de economía y de gestión de recursos humanos, organización y planificación de proyectos, así como la legislación, regulación y normalización en el ámbito de los proyectos informáticos, de acuerdo con los conocimientos adquiridos según lo establecido en los apartados específicos posteriores.

4.2. Perfiles profesionales de los/as ingenieros/as en Informática

El perfil profesional es un conjunto de capacidades y competencias que identifican la formación de una persona para sumir en condiciones óptimas las responsablemente las funciones y tareas de una determinada profesión o trabajo.

En el caso de la Ingeniería Informática existen una serie de perfiles profesionales a los cuales podemos llegar a optar.

Algunos de estos perfiles, agrupados por bloques, pueden ser:

Dirección

Director de informática

- Participar en el Comité de Dirección.
- Poner en marcha la estrategia de la empresa a nivel informático.
- Garantizar las relaciones entre los departamentos de la empresa. Primordial para una buena acogida de las evoluciones del sistema informático.
- Cuidar la coherencia del sistema de información con respecto a la organización de la empresa y a su evolución.
- Definir las políticas, características técnicas y la adecuación de los sistemas hardware y de red, así como las características de los sistemas de comunicaciones.
- Definir el presupuesto y gestionar los medios materiales y humanos.
- Definir los planes de formación y de reciclaje profesional.
- Definir la política informática de la empresa a medio y largo plazo.
- Establecer el alineamiento de los objetivos informáticos con los objetivos de la empresa y velar por su cumplimiento.
- Evaluar los Riesgos Empresariales asociados a los Sistemas Informáticos y establece las orientaciones y directrices para mitigarlos.
- Evaluar los Riesgos Empresariales asociados a los Sistemas Informáticos y establece las orientaciones y directrices para mitigarlos.
- Evaluar los Riesgos Empresariales asociados a los Sistemas Informáticos y establece las orientaciones y directrices para mitigarlos.

Director de organización e informática

- Principalmente en las PYMEs o en filiales de grandes grupos.
- Garantizar la fiabilidad, la coherencia y la evolución del sistema informático desde un punto de vista técnico y funcional.
- Tomar a su cargo el conjunto de los proyectos informáticos de la PYME.
- Participar en la definición de la estrategia y los objetivos informáticos.
- Organizar el departamento informático tanto a nivel humano como a nivel material.
- Definir y controlar el presupuesto o una parte de este en función de su importancia.
- Gestionar las relaciones con otros departamentos de la PYME para establecer los planes de acción y en ciertos casos, los libros contables.
- Gestionar las relaciones con los proveedores externos y los eventuales prestadores de servicio.
- Asegurar una unidad tecnológica y representar una fuerza de apoyo para la Dirección.

Director de proyecto informático

- La dirección de proyecto puede ser ejecutada por una única persona, independiente de los responsables de las fases principales, o por un comité integrado o no por dichos miembros.
- Gestionar el conjunto de proyectos informáticos asignados por el Director de Informática, o bien relacionados con funciones de negocio, divisiones organizativas, entre otros.
- Concebir las aplicaciones. Controlar y gestionar los proyectos informáticos.
- Controlar la introducción y los parámetros de los sistemas integrados (ERP, CRM) o de los sistemas abiertos de arquitectura empresarial definidos en las políticas informáticas de la empresa (Frameworks u otros).
- Participar en la elaboración de los esquemas directivos.
- Organizar y distribuir el trabajo de los equipos de análisis y de desarrollo (jefes de proyectos, responsables de aplicación).
- Vigilar la coherencia del sistema de Información.
- Participar en la elaboración de los esquemas directivos y vigilar la coherencia del sistema de información.
- Tomar a su cargo las relaciones con los prestadores del servicio y ciertos proveedores externos.
- Gestionar la conexión entre los departamentos usuarios.
- Vigilar la tecnología y definir las orientaciones técnicas (metodología, calidad, herramientas de desarrollo, entre otros).
- Concreción de los objetivos de cualquier Sistema Informático.
- Planificación del desarrollo de un Proyecto Informático.
- Estudio de Rentabilidad de los Sistemas Informáticos.
- Estudio de los Riesgos de los Sistemas Informáticos.
- Redacción, para la Dirección de la Empresa y la Dirección de Informática, de los informes que se precisan para el seguimiento del proyecto.
- La coordinación de las fases en las que se ha estructurado el proyecto. Es una labor que requiere experiencia y aplicación de amplios conocimientos de Ingeniería y dirección.

Jefe de proyecto

- Ejecutar los proyectos informáticos asignados por el Director de Proyecto, dirigiendo y coordinando el proyecto de desarrollo y mantenimiento de aplicaciones, supervisando las funciones y recursos de análisis funcional, orgánico y programación, asegurando la adecuada explotación de las aplicaciones.
- Capaz de dominar todas las etapas de la vida de un proyecto (análisis de concepción, análisis técnico, programación, pruebas, documentación y formación de usuarios).
- Dirigir el equipo de trabajo compuesto por Analistas Funcionales, Analistas de aplicaciones, Programadores. Gestión de los RRHH de los componentes del proyecto (evaluaciones, desempeño, motivación).
- Control y seguimiento de plazos, indicadores económicos y de calidad.
- Supervisar y coordinar el desarrollo completo de aplicaciones y administrar la introducción de los sistemas de gestión.
- Controlar las aplicaciones en explotación, minimizando las consecuencias negativas sobre las operaciones en producción y desarrollo de aplicaciones.

- Analizar y recoger nuevas técnicas y herramientas del mercado estudiando su viabilidad y necesidad. Posibilidad de contratar recursos externos.
- Control y Gestión del Desarrollo del Proyecto Informático.
- Redacción, para la Dirección de Informática y para la Dirección del Proyecto de los informes que se precisan para el seguimiento del proyecto.
- Responsabilidad de la gestión económica del proyecto. Elaboración de propuestas, relación con el cliente a nivel de proyecto.

Sistemas y redes

Jefe de sistemas

- Planificar, supervisar y coordinar el mantenimiento de sistemas operativos, software de mercado y propio, básico o de soporte.
- Definir y actualizar software básico.
- Actualizar y decidir la alternativa óptima de software de mercado a adquirir.
- Diseñar, en conexión con la Dirección de Informática, la política de hardware, redes y comunicaciones, respecto a adquisiciones, sustituciones, entre otros.
- Resolver y coordinar las incidencias de los sistemas.
- Dirigir las actividades y recursos, técnicos, materiales y los equipos de soporte en materia de sistemas operativos, bases de datos y comunicaciones.

Jefe de redes

- Dirigir, planificar y coordinar la gestión de la infraestructura de redes y comunicaciones.
- Gerente de la fiabilidad, de la coherencia y de la evolución de la arquitectura de la Red y de las Telecomunicaciones utilizadas por los Sistemas Informáticos de la Empresa.
- Gestión de grandes redes corporativas y/u operadores de telecomunicaciones, redes de acceso, redes de transmisión de voz, datos, imágenes, conmutación, gestión de tráfico, así como de todos los aspectos de las redes WAN y las estrategias ligadas a Internet.
- Poner en marcha las redes tanto a nivel material como logístico.
- Desarrollar y mantener dichas redes. Elección de los elementos HW y SW para la optimización de los servicios de redes de comunicaciones.
- Gestionar las relaciones con los proveedores y negociar los contratos.
- Seguimiento de los presupuestos, los costes y las inversiones.
- Mantenimiento y evolución de los sistemas de gestión de las Telecomunicaciones.
- Enmarcar los participantes internos y externos en los proyectos de Telecomunicaciones.
- Escoger y gestionar los contratos con los operadores.
- Dirección Técnica y planificación de proyectos de implantación de soluciones y servicios asociados a las redes de comunicaciones.
- Gestión del conocimiento en inteligencia de negocio en grandes sistemas de redes de comunicaciones en datos y voz y sus servicios de valor añadido.
- Gestión de grandes proyectos de cableado de redes, y las infraestructuras parejas, suelos y techos técnicos, electricidad, entre otros.

Ingeniero de sistemas y redes (Administrador de red y/o Sistemas)

- Diseñar la arquitectura de comunicaciones de un entorno de complejidad media o baja (diseño lógico y físico). Definir las necesidades de la arquitectura.
- Dimensionamiento, segmentos, capacidades. Topologías de red, protocolos (TCP/IP, entre otros).
- Políticas de: ampliación de red, administración de red, mantenimiento de red, acceso, gestión y mantenimiento de usuarios, gestión y mantenimiento de seguridad.
- Supervisión de la implantación de la infraestructura de red telemática.
- Instalar las redes (hardware, middleware y software), gestionar las intervenciones externas, si es preciso.
- Administrar la infraestructura de red LAN/WAN), proporcionando la asistencia técnica: mantenimiento general de red, resolución y gestión de incidencias, administración de equipos de monitorización, administración y mantenimiento del hardware de comunicaciones, monitorización de red (estadísticas, incidencias, ataques...).
- Soporte técnico y formación de base a usuarios.
- Desarrollar pequeñas aplicaciones específicas para la gestión o mantenimiento de las redes.
- Producir y analizar estadísticas de aplicación.
- Garantizar la conexión de las redes (puentes, routers, multiplexores).
- Seguimiento y optimización de costes de conexiones y participación en la elaboración del presupuesto de telecomunicaciones.
- Instalación, Administración, Configuración, Mantenimiento y Gestión servidores: servidores de impresoras, servidores web, servidores de mensajería, entre otros. Administración de cuentas de usuario, grupo e impresoras. Administración del sistema de archivos. Seguridad de recursos y carpetas compartidas. Administración de discos. Recursos de Red.
- Servicios de Terminales e Instalaciones Remotas.
- Conexión a Internet. Administración de Servicios de Internet. Configuración de los servicios de correo. Conexión de redes locales a Internet.

Técnico de redes

- Implantar y administrar sistemas informáticos en entornos de baja complejidad (alta y baja de usuarios, servidor de impresión, entre otros).
- Implantar y administrar redes locales de baja complejidad y gestionar la conexión del sistema informático a redes extensas.
- Implantar y facilitar la utilización de paquetes informáticos de propósito general y aplicaciones específicas.
- Proponer y coordinar cambios para mejorar la explotación del sistema y las aplicaciones.
- Mantenimiento de equipos informáticos, impresoras, y otros periféricos de la red local.

Operador de sistemas y redes

- Su función consistiría en monitorizar sistemas y redes, resolver incidencias muy simples, y escalar la incidencia a un administrador si no puede ser resuelta en un plazo de tiempo muy corto.
- Muy común en las grandes empresas, sobre todo en servicios que tienen que estar operativos todo el día, todos los días de la semana.

Seguridad**Jefe de seguridad**

- Definición de las políticas y procedimientos de seguridad.
- Implantación práctica de la seguridad en la organización.
- Mecanismos de registro de actividad y recuperación de errores.
- Velar por mantener un adecuado nivel de riesgo en la organización y establecer el riesgo residual latente.
- Velar por la adecuación de los planes de continuidad del negocio.
- Verificar los incidentes de seguridad y proponer medidas correctoras.

Ingeniero de seguridad

- Analizar y Gestionar los riesgos del sistema informático, determinar sus vulnerabilidades y establecer las medidas de salvaguarda que garanticen la confidencialidad, integridad y disponibilidad de la información de acuerdo con un riesgo residual asumido por la organización.
- Definir, de acuerdo con los objetivos de seguridad establecidos en la organización, la seguridad de los sistemas informáticos.
- Políticas de seguridad.
- Organización de la seguridad y clasificación de los recursos.
- Seguridad física y del entorno.
- Protección y control de acceso al sistema.
- Seguridad en las Comunicaciones.
- Seguridad en la operación y producción.
- Seguridad en el software tanto de los sistemas operativos, bases de datos y aplicaciones.
- Definir las especificaciones de seguridad para que los sistemas informáticos cumplan la legislación y normas estándar de seguridad nacionales e internacionales.
- Diseñar la seguridad del sistema informático según las especificaciones establecidas.
- Dirigir los proyectos de Seguridad basados en las leyes y normas estándar que permiten a la organizaciones Públicas y Privadas validar (o certificar) su cumplimiento y obtener las acreditaciones de seguridad exigidas por ley y normas adoptadas.
- Gestionar el Plan de Seguridad Informática y mantenerlo actualizado, muy especialmente el plan de continuidad del negocio.
- Velar por el cumplimiento legal de los sistemas informáticos utilizados en la organización: datos personales, propiedad intelectual, software legal, etc.

- Colaborar con la Dirección en la resolución de incidentes de seguridad y especialmente en aquellos que puedan dar origen a delitos y faltas tipificados en el derecho Penal, Civil, Convenios internacionales, etc.
- Colaborar con la Autoridad Judicial si los incidentes de seguridad acaecidos lo exigen en defensa de los intereses de la organización, o si son críticos para el Estado: seguridad nacional, seguridad de las personas, medio ambiente, etc.

Técnico de seguridad

- Implantar e implementar las soluciones diseñadas por el responsable de seguridad.
- Seguridad en Internet.
- Implementación práctica de firewall. Funcionalidades, componentes, instalación y configuración del Firewall. Servicio de Nombres y de Correo electrónico a través o no de un Firewall.
- Establecer túneles IP seguros a través de Internet.
- Mecanismos de registro de actividad y recuperación de errores.
- Registro de actividad y recuperación de errores.
- Redes privadas virtuales.
- Dominios TCP/IP.

Calidad**Jefe de calidad**

- Será el encargado de crear el plan de calidad del producto. Diseñará y gestionará el despliegue necesario para ejecutar y coordinar las pruebas y revisiones del software.
- Definir las normas de desarrollo en colaboración con la Dirección de Informática.
- Motivar y coordinar los equipos de desarrollo en el marco de aplicación de las normas y métodos en vigor.
- Intermediario y consejero de cara a los desarrollos que se realicen.
- Asegurar la definición de las directrices de calidad, su aplicación, así como la estandarización.
- Responsable de la adecuación entre los desarrollos realizados y las directrices establecidas.
- Poner en marcha los procedimientos de prueba y de control de calidad.
- Asegurar la coherencia y la coordinación de su trayectoria con la política global de la empresa.
- Tomar a su cargo la campaña de las pruebas de cara al conjunto de los usuarios finales.
- Participar en la distribución de las ediciones originales de las aplicaciones y de los documentos a las entidades de producción garantizando un alto nivel de calidad.
- Garantizar una calidad permanente a través de los procedimientos y de las herramientas.
- Apoyar las demandas cotidianas de los usuarios.

Ingeniero de calidad

- Colaborará con el jefe de calidad en la creación y ejecución del plan de calidad.

Enseñanza**Profesor/a de universidad**

- Las propias del tipo de enseñanza.
- Cabe la posibilidad que también tenga el perfil profesional de **Investigador**. En este caso, aparte de enseñar los conocimientos necesarios para futuras generaciones de informáticos, avanza en los conocimientos necesarios para un mejor entendimiento y avance de las tecnologías actuales.

Profesor/a de ciclo formativo grado superior – bachiller

- Las propias del tipo de enseñanza.

Profesor/a de ciclo formativo de grado medio

- Las propias del tipo de enseñanza.

Base de datos**Ingeniero de base de datos (Administrador de base de datos)**

Este perfil normalmente existe en las grandes empresas y en la Administración, y su responsabilidad es muy alta, ya que la seguridad de los datos es imprescindible. En una pequeña y mediana empresa, no suele tener mucho sentido y normalmente no existe como tal.

- Administrar un sistema de bases de datos, interpretando su diseño y estructura, y realizando la adaptación del modelo a los requerimientos del sistema gestor de bases de datos (SGBD), así como la configuración y administración de este a nivel físico y lógico, a fin de asegurar la integridad, disponibilidad y confidencialidad de la información almacenada.
- Desarrollo y construcción de las bases de datos. Asegurar la coherencia y la adaptación a las necesidades de la empresa.
- Gestionar las autorizaciones de acceso para los usuarios.
- Responsabilidad e de la integridad de los datos y de la existencia de Backups.
- Estimación de volúmenes de las estructuras de datos, definiendo mecanismos de migración y carga inicial de datos.
- En producción se ocupa de la gestión y operativa asociada a las bases de datos y al software en el que están implementadas.
- Este perfil es independiente de la tecnología de Base de Datos, jerárquica, relacional, orientada a objetos, nativa XML, o cualquier otra.
- Implantación de las medidas de seguridad (ejemplo reglamentos de desarrollo de la LOPD).

Arquitectura

Arquitecto de Software

- Gestión de los requisitos no funcionales y definición de la *Arquitectura de Software*.
- Selección de la tecnología adecuada.
- Mejora continua de la arquitectura.
- Líder y formador de otros trabajadores.
- Aseguramiento de la calidad necesaria del producto.

Diseñador de arquitecturas de software

- Identifica y define las responsabilidades, operaciones, atributos y relaciones de los elementos de diseño.
- Se asegura de que el diseño sea coherente con la arquitectura de software, y que esté detallado hasta un punto en que pueda proceder la implementación.

Análisis

Jefe de análisis

- Coordinar el proceso, permitiendo una ejecución fluida. Gestionará y facilitará la interacción con los clientes.
- Será responsable de realizar/coordinar/supervisar los productos obtenidos en el análisis.

Ingeniero de requisitos

- Obtener y gestionar los requisitos del software a construir. Para ello se emplearán técnicas de análisis y licitación de requisitos.
- Durante el proceso de construcción verificará la trazabilidad y cumplimiento de los requisitos establecidos.

Ingeniero funcional

- Modelar el software a desarrollar, comprendiendo y considerando los requisitos.
- Coordinará el trabajo con expertos en el área específica que cubra el desarrollo.
- Participará en el proceso de estimación y planificación.
- Análisis de los nuevos sistemas informáticos y de los cambios en los existentes.
- Diseño de las soluciones informáticas relacionadas con los cambios en los sistemas existentes o con los nuevos sistemas.
- Dirección y asesoramiento a los Técnicos de Programación en la realización de los programas.
- Creación de los test de pruebas para verificar que los sistemas informáticos cumplen los requisitos y especificaciones de análisis y diseño.
- Verificación de la documentación, tanto de los cambios en los sistemas existentes, como la de los nuevos sistemas para garantizar que está completa y al día.

- Asesoramiento a Usuarios, Programadores y Jefe de Estudios en la redacción de la Documentación de Usuario, Instalación y Explotación.
- Dirección del arranque o "lanzamiento" de un nuevo sistema.
- Asesora al Responsable de Estudios en la elaboración de los criterios que permiten la mejor explotación de los nuevos sistemas.
- Ayuda al Área de Estudios en la resolución de los fallos que se producen en los Sistemas en Producción.
- Evalúa nuevos productos informáticos que pueden aportar mejoras tanto en los sistemas existentes, como para el desarrollo de nuevos sistemas.
- Asesora a los Usuarios para mejor utilizar los sistemas existentes.
- Dirige y Coordina el Desarrollo de Reuniones relacionadas con temas que afectan a los Sistemas Informáticos.
- Estudio de métodos, técnicas y herramientas de análisis y diseño.
- Estudio de la evolución de las nuevas tecnologías, sobre todo de aquellas que pueden aportar mejoras importantes en los sistemas utilizados en la empresa.

Ingeniero de Aplicaciones (Analista de aplicaciones)

- Interpretar las especificaciones funcionales encaminadas al desarrollo de las aplicaciones informáticas.
- Realizar el análisis y el diseño detallado de las aplicaciones informáticas.
- Definir la estructura modular y de datos para llevar a cabo las aplicaciones informáticas que cumplan con las especificaciones funcionales y restricciones del lenguaje de programación.
- Definición y descripción de procedimientos e interfaz de usuario.
- Realizar pruebas que verifiquen la validez funcional, la integridad de los datos y el rendimiento de las aplicaciones informáticas.
- Elaborar y mantener documentación descriptiva de la génesis, producción y operatividad de las aplicaciones informáticas.
- Diseñar servicios de presentación que faciliten la explotación de las aplicaciones.
- Estudiar el sistema actual existente y analizar e idear mejores medios para llevar a cabo los mismos objetivos u otros adicionales.
- Participar en el diseño de nuevos sistemas informáticos como consecuencia de la informatización de áreas de la empresa que utilizan para el desarrollo de sus tareas métodos y procesos manuales.
- Integrar sistemas informáticos existentes susceptibles de interrelacionarse.
- Escuchar y asesorar a los usuarios en la resolución de los problemas que se les plantean con el uso de los sistemas informáticos.
- Asesorar a los programadores en los problemas que se les plantean con la programación de los sistemas.
- Colaborar con los responsables de estudios y explotación en la resolución de los fallos que se originen en los sistemas en producción.
- Mantenerse al día en técnicas, métodos y herramientas de análisis y diseño.

Ingeniero web

- Analizar las necesidades de la empresa y definir una estrategia de comunicación online.
- Definir la estructura de información en el seno del sitio, tanto sobre un plano lógico como visual.
- Crear una plataforma técnica en relación con la dirección de la explotación y definir, con la dirección de los estudios, los desarrollos específicos.
- Responsabilidad de la transmisión de la cultura de Internet de los usuarios y desarrollar una política de alimentación del sitio.
- Asegurar una vigilancia técnica de los desarrollos online.
- Asegurar el desarrollo coherente de las actividades Internet de la empresa y gestionar la evolución en función de las orientaciones estratégicas de la empresa.

Diseño**Jefe de diseño**

- Coordinará la labor de diseño.
- Gestionará la planificación del proceso de diseño.

Ingeniero de diseño (Diseñador de sistemas informáticos)

- Convertir un modelo abstracto del sistema a construir en uno concreto, teniendo en cuenta tanto los requisitos del sistema como las restricciones específicas necesarias para la construcción.
- Crear la estructura del software a construir.
- Generar el diseño de software, para lo cual ha de dominar el área de Ingeniería de Software que cubre la fase de diseño.
- Colaborar con expertos en una tecnología o tecnologías concretas, para producir un diseño construible.
- Seleccionar las tecnologías y plataformas que se usarán para implementar el diseño.
- Poseer capacidades avanzadas de implementación de software para los siguientes puntos:
 - Implementar pruebas de concepto con las tecnologías elegidas para verificar su viabilidad.
 - Desarrollar prototipos de partes críticas del software.
 - Crear pruebas de concepto para guiar el proceso de construcción.
 - Asesorar técnicamente al equipo de análisis para propiciar la viabilidad de este.
 - Dominar a nivel técnico la plataforma o plataformas objetivo con el fin de poder tomar decisiones de diseño adecuadas.
 - Considerar restricciones de rendimiento concretas del entorno al tomar decisiones en la elaboración del diseño.

Construcción

Jefe de construcción

- Coordinará el proceso de construcción. Será responsable de colaborar en la creación del plan de proyecto, realizar y gestionar la creación de estimaciones y supervisar el proceso de calidad.
- Coordinar las operaciones de los miembros de su equipo, colaborando con la dirección de proyecto.
- Apoyar técnica y organizativamente a los miembros del equipo.
- Asegurar que se cumplen los objetivos del equipo.
- Supervisar la aplicación del proceso de desarrollo: metodología, guías de estilo, registro de operaciones, entre otras.

Ingeniero de desarrollo

- Participa en el proceso de construcción aplicando técnicas de Ingeniería de Software.
- Realizar estimaciones del software a construir, colaborando en el trabajo de planificación con el jefe o responsable de construcción.
- Interpretar el diseño y realizar adaptaciones cuando sea necesario.
- Colaborar en la generación y supervisar el cumplimiento de la guía de estilo de programación del proyecto.
- Controlar factores de calidad en la producción del código fuente (conceptos como cohesión, acoplamiento, construcciones del lenguaje a utilizar y evitar, porcentaje y formato de comentarios).
- Aplicar e interpretar métricas sobre el software informando al responsable de construcción y tomando medidas correctivas o preventivas.
- Implementar código crítico, considerándose como tal según criterios de valor estratégico, urgencia, rendimiento o dificultad técnica.
- Dominar las tecnologías y lenguajes empleados para la construcción del software, sirviendo de apoyo a los programadores.
- Dominar técnicas de depuración de errores (debugging). La localización y corrección sistemática de errores en el código fuente exige un profundo conocimiento del lenguaje utilizado, las tecnologías empleadas y el sistema operativo, así como de conceptos fundamentales de bases de datos, teoría de sistemas operativos y redes, entre otras.

Ingeniero técnico de desarrollo (Analista programador)

Un analista-programador es la combinación de dos roles que se complementan, por un lado, es un profesional con capacidad para analizar las necesidades que demandan los clientes y los requisitos de una aplicación, es capaz de abstraer y sintetizar una solución óptima, definir la estructura de base de datos que dará soporte a la solución, y finalmente programar con uno o varios lenguajes de programación la solución.

Técnico de programación (Programador)

- Desempeñará toda tarea de desarrollo que no requiera de la cualificación de un ingeniero de desarrollo.

- Desarrollo de aplicaciones. Construir el código que dará lugar al producto resultante en base al diseño realizado, utilizando lenguajes y bases de datos. Generando el código asociado a los procedimientos de migración y carga inicial de datos.
- Realización de las pruebas unitarias y participa en las pruebas de conjunto de la aplicación.
- Configurar y explotar sistemas informáticos.
- Programar bases de datos relacionales no corporativas.
- Desarrollar componentes software en lenguajes de programación.
- Construcción de páginas web.

Jefe de gestión de configuración

La gestión de la configuración es imprescindible en un proyecto de ingeniería de software. El responsable del área será el encargado de diseñar el plan de gestión de configuración, participando en la creación de la planificación, establecimiento de hitos y entregas.

Ingeniero gestión de configuración

Encargado de ejecutar el plan de configuración. Entre sus labores estarán la creación de versiones, realización de integraciones, documentación de versión, entre otros.

Técnico de pruebas

En los sistemas que requieran de pruebas no automatizables será necesario disponer de personal dedicado a ejecutar los casos de prueba. La cualificación requerida variará según la naturaleza del sistema y las pruebas a ejecutar.

Explotación**Jefe de informática**

Se encuentra principalmente esta función en las PYMEs o en filiales de los grandes grupos.

- Garantizar la fiabilidad, la coherencia y la evolución del sistema informático desde un punto de vista técnico y funcional.
- Tomar a su cargo el conjunto de los proyectos informáticos de la PYME.
- Participar en la definición de la estrategia y los objetivos informáticos.
- Organizar el departamento informático tanto a nivel humano como a nivel material.
- Definir y controlar el presupuesto o una parte de este en función de su importancia.
- Gestionar las relaciones con otros departamentos de la PYME para establecer los planes de acción y en ciertos casos, los libros contables.
- Gestionar las relaciones con los proveedores externos y los eventuales prestadores de servicio.
- Asegurar una unidad tecnológica y representar una fuerza de apoyo para la Dirección.
- Responsabilizarse de las bases de datos, comunicaciones, entre otros, de acuerdo con la pequeña escala de la empresa en cuestión.

Jefe de explotación

- Responsabilidad de garantizar el adecuado nivel de servicio en la explotación de los sistemas informáticos y teleinformáticos.
- Responsabilidad en la atención de usuarios y microinformática.
- Responsabilidad en la verificación de la corrección en las medidas de seguridad que requiera la política de seguridad de la organización.
- Garantizar la continuidad de la función de negocio.
- Planificar, supervisar y coordinar el desarrollo, implantación y mantenimiento de los sistemas operativos, bases de datos, software de mercado y propio, básico o de soporte.
- Definir y actualizar el software básico.
- Analizar y decidir la alternativa óptima de software de mercado a adquirir.
- Diseñar la política de hardware, respecto a adquisiciones, sustituciones, entre otros.
- Resolver y coordinar las incidencias de los sistemas.
- Dirigir las actividades y recursos técnicos, materiales y los equipos de soporte en materia de sistemas operativos, bases de datos y comunicaciones.

Ingeniero de explotación

- Asegurar el buen funcionamiento físico de los sistemas informáticos (automatización de copias de seguridad y la seguridad de datos).
- Administrar las incidencias y asegurar las soluciones.
- Organizar y supervisar el trabajo de su equipo de los técnicos de mantenimiento y los ingenieros de sistemas y redes.
- Administrar tanto los abastecimientos como las relaciones con los proveedores y los constructores.
- Responsable del buen funcionamiento del sistema informático y sus resultados.
- Colaboración con el Responsable de Desarrollo para que el sistema de arquitectura pueda responder a las exigencias de las aplicaciones desarrolladas.
- Definir los procesos, los documentos y ejecutar su control.

Jefe de microinformática

- Dirigir el equipo ofimático.
- Definir la arquitectura de la red local y su puesta en marcha.
- Tomar a su cargo la microinformática a nivel material y logístico.
- Definir y poner en marcha la política de mantenimiento del parque ofimático.
- Atender las necesidades de los usuarios, poner en marcha el soporte de primer nivel y gestionar el buen funcionamiento de los puestos de trabajo.
- Gestionar la formación de los usuarios.
- Asegurar las relaciones con los proveedores y los prestadores de servicios externos.
- Supervisar el desarrollo de pequeñas aplicaciones, participar eventualmente en el desarrollo.
- Participar en la elaboración de directrices y en la elección estratégica relativa a la evolución de los puestos de trabajo.
- Asegurar un control tecnológico.
- Gestionar las redes locales y realizar los desarrollos específicos y las interconexiones con las redes WAN.
- Ser responsable de los aspectos ligados a la seguridad del material y los datos del conjunto del parque informático.
- Gestión y seguimiento de las incidencias informáticas y de seguridad.

Técnico en microinformática

- Instalar, configurar y mantener sistemas microinformáticos.
- Conectar, configurar y mantener sistemas microinformáticos en red.
- Instalar, configurar y mantener paquetes informáticos de propósito general y aplicaciones específicas.
- Soporte y apoyo al usuario.
- Gestión de Incidencias.

Operador

- Gestionar las incidencias, el mantenimiento y los consumibles.
- Asistir a los usuarios en los incidentes de orden material y logístico (soporte de primer nivel).
- Asegurar la solución de las incidencias.
- Mantener buenas relaciones con los usuarios teniendo en cuenta que es el primer contacto con el departamento informático.

Ingeniero de soporte

Realizar soporte a otros usuarios y ayudarles con el uso del producto. En este perfil hay distintos tipos de soporte:

- Técnica de Sistemas.
- Microinformática.
- Gestión de red y comunicaciones.
- Administración base de datos.
- Relacionadas con el soporte a la configuración, evaluación y explotación del sistema informático.

Ventas y Marketing**Ingeniero comercial**

Los Ingenieros comerciales son muchas veces antiguos Ingenieros de Desarrollo que después de haber realizado proyectos informáticos, se han interesado por la relación con los clientes.

- Gestionar los clientes o el área geográfica asignada según la organización de la empresa.
- Analizar los proyectos y las necesidades y proponer soluciones en el plano técnico, humano y financiero.
- Redactar las propuestas comerciales que pueden implicar soluciones estándar o a medida.
- Negociar los contratos.
- Desarrollar el volumen de negocios y asegurar la gestión administrativa.
- Poner en marcha la estrategia comercial elaborada con la dirección.
- Asegurar el seguimiento de los proyectos y su realización.

Ingeniero preventa

- La mayoría han ocupado puestos técnicos, pero desean orientar su carrera hacia puestos más cerca de la venta.
- Apoyar a los ingenieros comerciales en las entrevistas con los clientes.
- Ayuda a definir la necesidad, presentar la solución o el producto en un plano técnico.
- Definir con mayor precisión la necesidad técnica del cliente.
- Elaborar la parte técnica de la propuesta.
- Gestionar la implantación de la solución asumiendo la gestión del proyecto en su integridad, o asegurar una transferencia de competencia hacia los equipos de implantación.
- Asegurar la comunicación entre los usuarios y el departamento de I+D para adaptar o evaluar el producto.

Técnico comercial

- Ayuda en las tareas del ingeniero comercial y muchas veces realizan los primeros contactos con los clientes.

Ingeniero gestor de producto

- Identificación de nuevos productos / servicios, y oportunidades de desarrollo de negocio. Análisis de modelos de negocio asociados a la definición de los nuevos productos / servicios.
- Coordinar y participar en el proceso de marketing para el desarrollo de productos / servicios.
- Colaboración en los estudios de investigación de mercado.
- Colaborar en la definición de la estrategia evolutiva del producto.
- Seguimiento del rendimiento de los procesos y resultados comerciales, identificando y documentando incidencias técnicas y de aplicación.
- Analizar las necesidades de los clientes, el mercado y la competencia, participando activamente en la planificación y desarrollo de estrategias de ventas y marketing.

Legal y consultivo

Consultor

- Las personas que trabajen en este perfil son expertos de las problemáticas más frecuentes en cada sector y de las soluciones informáticas disponibles.
- El estudio de las necesidades funcionales y técnicas de los clientes.
- Conocimiento y dominio de todas las actividades del análisis de sistemas y experiencia como analistas de sistemas en la solución de las problemáticas objeto de la consultoría.
- Análisis, mejora, adaptación y diseño de los procesos específicos afectados en las problemáticas a solucionar.
- La implantación y parametrización de los sistemas.
- Analizar, proponer políticas de desarrollo, así como desarrollo y potenciación de los recursos humanos, estructura organizativa, con objeto de optimizar los procesos y procedimientos implicados.
- Planificación y estimación de recursos y costes de los proyectos implantados o a implantar.
- Conocimientos de las restricciones legales aplicables a cada proyecto.
- Dominio de toda la normativa del sector.

Auditor

- Recoger, agrupar y evaluar evidencias para efectuar diagnóstico de los sistemas informáticos desde diferentes ángulos: técnico, organizativo, funcional, económico, legal, normativo y humano.
- Determinar si en el análisis, diseño, construcción y explotación (producción) de un sistema informático se han considerado y evaluado los riesgos y si se han establecido las salvaguardas adecuadas para mantener la confidencialidad, integridad y disponibilidad de la información.
- Determinar si un sistema informático salvaguarda los activos y lleva a cabo los fines encomendados para la organización a la que da servicio.
- Determinar la eficiencia y eficacia del sistema informático teniendo en cuenta los objetivos de la organización con los que debe de estar alineado y los recursos utilizados.
- Verificar la conformidad del sistema informático con la legislación aplicable, o con normas estándar nacionales e internacionales exigidas por las organizaciones públicas o privadas tanto en su construcción, como en su explotación (producción) posterior.
- Emitir informes de auditoría informática para las administraciones públicas y para las organizaciones privadas y especialmente las establecidas por las leyes y reglamentos vigentes.
- Realizar Informes de Auditoría sobre aspectos concretos de los sistemas informáticos: planificación y gestión; acceso a los sistemas; datos y bases de datos; desarrollo y calidad del software; canales de distribución, eficacia; eficiencia de su gestión, entre otros.
- Proponer las soluciones de mejora, en base a los informes emitidos y controlar su implantación.

Perito

- Redactar y firmar informes, dictámenes, peritaciones con validez oficial ante las administraciones públicas, tribunales de justicia, y corporaciones oficiales, así como para empresas privadas y usuarios particulares, en los asuntos relacionados con la informática.

Arbitraje e intermediación

Considerando el arbitraje como un proceso vinculante de toma de decisiones basadas en evidencia de documentos. Testimonio y alegaciones de las partes. La intermediación es la herramienta principal del mismo.

La intermediación permite negociar un acuerdo en lugar de hacer que una autoridad decida el interés de una de las partes. La intermediación da poder a los involucrados en una disputa para decidir cuestiones basadas en sus intereses y valores sin que una autoridad decida las cuestiones por ellos.

- Participar como árbitros en las controversias en que el motivo de discordia sea cualquier tema relacionado con la Informática.
- Decidir ante un nombramiento la competencia o no de su labor.
- Determinar medidas cautelares.
- Determinar, si fuera preciso, el lugar donde ha de tener lugar el arbitraje.
- Participar en las reuniones privadas con las partes.
- Fijar el Inicio del arbitraje.
- Determinar el Idioma del Arbitraje y si fuera preciso, ordenar su traducción.
- Nombrar, si fuera preciso, peritos Informáticos para que intervengan en el laudo.
- Decidir sobre la controversia que motivó el arbitraje.
- Firmar los laudos redactados por las partes.
- Registrar los laudos en el colegio profesional correspondiente.
- Notificar el laudo a las partes.
- Participar como testigo, si fuera preciso, ante los tribunales, en el caso de anulación de un laudo en el que haya participado.

5. Situación europea e internacional de los informáticos

5.1. Estudiar Ingeniería Informática en España y en el extranjero

En España, se puede estudiar en prácticamente cualquier universidad. Aparece también en multitud de dobles grados, combinada con otras carreras como Matemáticas, Administración y Dirección de Empresas, Biotecnología o criminología, entre otras.

En Europa, también se puede encontrar en cualquier país alguna universidad donde se oferte la carrera.

Según la clasificación de Shanghái de 2019, una de las clasificaciones más conocidas mundialmente que valora la calidad de los estudios y las universidades, las diez mejores universidades europeas para estudiar informática son las siguientes:

- Escuela Politécnica Federal de Zúrich, Suiza.
- Universidad de Oxford, Reino Unido.
- University College, Londres, Reino Unido.
- Universidad de Edimburgo, Reino Unido.
- Universidad de Copenhague, Dinamarca.
- Universidad de Cambridge, Reino Unido.
- Escuela Politécnica Federal de Lausana, Suiza.
- Imperial College Londres, Reino Unido.
- Universidad Aalto, Finlandia.
- Universidad de Oslo, Noruega.

Mientras que, en España, las diez mejores universidades serían:

- Universidad de Barcelona.
- Universidad Complutense de Madrid.
- Universidad Pompeu Fabra.
- Universidad de Granada.
- Universidad Autónoma de Barcelona.
- Universidad Autónoma de Madrid.
- Universidad del País Vasco en Vizcaya.
- Universidad Politécnica de Valencia.
- Universidad de Santiago de Compostela.
- Universidad de Valencia.

5.2. Situación profesional nacional en Ingeniería Informática

Actualmente la carrera de ingeniería informática es la carrera más demandada en España, seguida de la Ingeniería de Telecomunicaciones, siendo también, uno de los perfiles más solicitados en el mercado laboral nacional.

Los lugares donde más demanda hay de Ingenieros Informáticos, son Madrid y Barcelona, que representan casi la mitad de las ofertas de trabajo del territorio nacional en este campo.

5.3. Índice de colocación de los egresados de la titulación

El índice de inserción laboral para los titulados en Ingeniería Informática era a noviembre de 2019, de un 1,19. Si bien solo el 63,3% de estos, ocupa un puesto de trabajo cualificado, la media de graduados que encuentran trabajo pasados cuatro años de haber terminado la carrera se encuentra en un 90,9%.

5.4. Empleos más demandados a nivel nacional

Los empleos más demandados relacionados con informática según LinkedIn son:

- Especialista en Inteligencia Artificial
- Desarrollador Salesforce
- Especialista en Customer Success
- Ingeniero de robótica
- Especialista en ciberseguridad
- Agile Coach
- Consultor de Cloud
- Desarrollador de Python
- Científico de datos
- Desarrollador de Big Data

5.5. Situación profesional internacional en Ingeniería Informática

Al igual que en España, la ingeniería informática tiene mucha demanda en el extranjero, donde se necesitan muchos profesionales debido a la creciente aparición de empresas relacionadas con la informática.

5.6. Empleos más demandados a nivel internacional

Los empleos más demandados a nivel internacional son muy parecidos a los de España. Siendo estos:

- Arquitecto de IA
- Business Intelligence Analyst
- Arquitecto de Cloud
- Analista /científico/ingeniero de datos.
- Desarrollador
- Ingeniero DevOps
- Administrador de red/cloud
- Especialista en ciberseguridad
- Administrador de sistemas
- Agente Servicio de Soporte

6. Responsabilidad

A la hora de desarrollar un producto software, la entrega de este no es nuestra única responsabilidad. Debemos ser conscientes de que tenemos que cumplir con una serie de aspectos de vital importancia, como puede ser la entrega de un software de calidad y útil a lo largo de un tiempo. Para ello, explicaremos más en profundidad estos aspectos que debemos tener en cuenta.

6.1. Instalación y uso de software

A la hora de instalar y hacer uso de un producto software debemos saber el tipo de licencia que tiene asociada ese software. Estas licencias son una especie de contrato entre el proveedor del programa y el usuario final, y contienen los términos y cláusulas que el usuario debe cumplir para hacer uso de él.

Existen varios tipos de licencias, y a continuación las explicaremos brevemente las más comunes:

- **Software Libre o Free Software.** Estos programas pueden ser utilizados, copiados, modificados y distribuidos por cualquiera, aunque en ocasiones requiera pagar cierta cantidad por ello.
- **Software de Dominio Público.** Los programas bajo esta licencia no están sujetos al copyright, por lo que algunos tipos de copia o versiones de estos pueden no ser libres si el autor impone restricciones adicionales.
- **Software Semi-libre.** Se trata de software que no es libre, pero puede ser utilizado, copiado, distribuido y modificado por particulares (no empresas) sin intenciones de lucro.
- **Software Propietario.** Es aquel software que su copia, redistribución o modificación están prohibidas por el propietario. Para hacerlo se debe pedir permiso al propietario.
- **Software Freeware.** No debemos confundirlo con Free Software, ya que los programas bajo esta licencia permiten la redistribución, pero no la modificación.

También, para hacer un correcto uso del software debe, además de cumplir con los términos y cláusulas de la licencia correspondiente, es recomendable que comprobemos la compatibilidad del sistema operativo y el hardware con el programa que se va a usar e instalar las actualizaciones correspondientes tanto para el sistema operativo como para otros programas ya instalados.

6.2. Garantía software

La garantía software se define como el nivel de certeza en que el software está libre de vulnerabilidades, ya sea que se hayan sido diseñadas intencionalmente en el software o insertada accidentalmente en cualquier fase de su ciclo de vida, además de que el software funcione como se tiene previsto.

El principal objetivo de la garantía de software es asegurarse de que los procesos, procedimientos y productos utilizado para producir y mantener el software cumplen con los requerimientos y estándares especificados para manejar los procesos, procedimientos y productos.

Algunas de las características software son:

Confiabilidad: No existen vulnerabilidades explotables, maliciosas o insertadas no intencionalmente.

Ejecución predecible: Confianza de que el software, cuando se ejecute, funcione como debe de hacerlo.

Conformidad: Conjunto planeado y sistemático de actividades multidisciplinarias que garanticen los procesos de software y que los productos cumplan con los requisitos, normas y procedimientos.

Cuando un usuario adquiere nuestro software realizamos un contrato de la licencia de usuario final y garantía limitada. Aparte de dar el software también tenemos que incluir productos complementarios, componentes, correcciones, actualizaciones y mejoras, como así también todo software, medio de soporte de datos o documentación complementaria tanto impresa como en línea.

Garantizamos de que el software estará libre de defectos de materiales y fabricación durante un plazo de noventa días a partir de la fecha de adquisición.

En el caso de producirse un error durante el período de cobertura de la garantía, el usuario deberá informar de inmediato y devolver el software. Las distintas opciones de proceder son las siguientes:

- La devolución del precio de compra
- La reparación o reemplazo del software que no cumpla con la garantía limitada

La garantía limitada carece de validez si el error del Software se produce como consecuencia de un accidente o el uso abusivo o inadecuado del producto. En caso de reemplazo, los nuevos medios de soporte de datos del software estarán garantizados durante el resto del periodo de cobertura de la garantía original o, de tratarse de un plazo mayor, durante un período de treinta días.

6.3. Venta de software

Existen multitud de productos de software a la venta al público, ya sea por internet o en tiendas, empresas y centros especializados en informática. Algunos de ellos pueden ser descargados libremente de manera online y ser usados por cualquier usuario gratuitamente o con restricciones de contenido o tiempo.

Podemos diferenciar el software según su distribución en:

- **Software enlatado o empaquetado:** Software con unas funcionalidades predefinidas y estándar que suele vender una empresa a usuarios que no necesitan una solución específica o “a medida”.

Son vendidos a multitud de usuarios por lo que la cantidad de errores suele ser menor que la del software hecho a medida y suelen ser mejorados, corregidos y recibir nuevos añadidos con el tiempo. Otra característica que debemos destacar es que son más baratos que el software hecho a medida ya que sus funcionalidades no son específicas para una empresa.

Al tener funcionalidades menos específicas y más generales para lograr venderlo a una cantidad mayor de usuarios, suele pasar que no todas sus funcionalidades son usadas por algunos usuarios.

- **Software desarrollado a medida:** Es software producido a pedido para un cliente dando una solución a los requerimientos de este, por lo que no suele ser útil para la mayoría de los clientes.

Su análisis, diseño y desarrollo puede llevar varios meses/años y suelen tener más problemas que el software empaquetado.

Además, tiene un coste superior ya que sus funcionalidades suelen ser específicas para dar solución a los problemas de un único cliente.

A diferencia del software empaquetado, el usuario si suele usar todas las funciones de este.

- **Freeware:** Software de dominio público al que cualquiera puede acceder de manera gratuita.
- **Shareware:** Software inicialmente gratuito en los que el usuario debe pagar una cuantía después de un tiempo específico.

El proveedor del software tiene ciertas responsabilidades después de la venta de este, a continuación, vamos a explicarlas:

Primero hay que señalar que el cliente y la empresa realizan un **contrato** por la venta del software, especificando una serie de términos como las contraprestaciones, precio, si su uso es exclusivo, mantenimiento, qué hacer en caso de defectos, etc.

El proveedor tiene como obligación principal el mantenimiento del sistema informático dentro de unos plazos acordados. No obstante, las partes deberían definir en el contrato el alcance específico de esta obligación.

El precio es fijado en el contrato por ambas partes y es su pago es obligación del cliente en los contratos de licencia, desarrollo o mantenimiento de software.

Debemos distinguir si estamos ante una licencia a usuario final, en cuyo caso el usuario abona un precio único o periódico por su uso, o si estamos ante las llamadas licencias de empresa. Estas últimas se caracterizan por no vincularse a un usuario o dispositivo concreto, sino al volumen de uso que se haga del software, es decir, se limita su uso a ciertos usuarios y su precio varía según el número de descargas o instalaciones.

También se suelen tener en cuenta modificaciones de precio por devaluación de la moneda o problemas que puedan surgir en el desarrollo que produzcan un retraso en la entrega.

La industria del software utiliza con frecuencia cláusulas de penalizaciones en los contratos, de tal forma que las partes saben con certeza qué puede ocurrir en caso de incumplimiento. Estas cláusulas suelen referirse a defectos o incumplimiento de plazos de entrega.

Otro punto a pactar es donde se llevará a cabo el servicio por parte del proveedor.

6.4. Desarrollo de software

A la hora de desarrollar un producto software, hay varias personas dentro de distintos departamentos que se implican en el desarrollo de este. Así que se puede “generalizar” el hecho de que todos los miembros del grupo de trabajo son responsables del desarrollo del producto a fabricar (sea, por ejemplo, una aplicación o un juego). Al ser, de alguna manera, parecido a una cadena de montaje, cada uno de los puestos tiene una responsabilidad concreta con esa parte del desarrollo de software.

Una buena forma de comprobar la calidad del software viene determinada por un profesional de testeo de software. Por ello, si se utilizará este profesional, sería el mayor responsable, dado que sería el último que revisa si el producto tiene la calidad suficiente como para lanzarse al mercado. Este profesional debería apoyarse bien en su metodología, técnicas y herramientas para cumplir con la calidad requerida en términos de efectividad, eficiencia, usabilidad y satisfacción.

Se puede afirmar el hecho de que la responsabilidad no termina una vez se ha entregado el producto, sino que esa entrega ha tenido una calidad y utilidad adecuados a las expectativas del cliente. De esta forma, no valdría con entregar un proyecto rápido al terminar el sprint, sino ir más allá y terminado con la calidad adecuada. No solo eso, además se deben cumplir las expectativas funcionales y no funcionales.

Una vez realizado todos estos cambios, aún tenemos la responsabilidad de ver la opinión del cliente, el cual nos puede informar de bugs o funcionalidades que no son acordes al producto y arreglarlo hasta alcanzar la buena opinión de los clientes.

Es importante recalcar el hecho de que el contrato es una figura muy importante, ya que lo que figure en dicho contrato será lo que al final valga. Por ello, en muchas empresas deciden realizar tipos de contrato que firman los usuarios que vayan a utilizar su producto. Así gracias a los términos y condiciones de uso, garantías y responsabilidades se puede llegar a asentar un buen entendimiento entre el proveedor y el cliente.

6.5. Pérdida de información

Toda empresa o autónomo que recoja datos de personas físicas se encuentra obligada a cumplir con dos normativas:

- Reglamento General de Protección de Datos (Europa)
- Ley de Protección de Datos y Garantía de Derechos Digitales (España)

Dadas estas normativas, debemos realizar una serie de tareas para garantizar que estamos cubiertos ante cualquier pérdida o robo de información, ya que en caso de que especifiquemos en los términos de uso de nuestra aplicación que no nos hacemos responsables de cualquier pérdida causada por el uso del software, nos estamos eximiendo de cualquier responsabilidad ajena a las citadas según esas dos normativas.

Para cumplir estas dos normativas, debemos:

1. Pedir consentimiento al usuario: El cliente debe aceptar voluntariamente que va a ceder sus datos a la aplicación.
2. Notificar brechas de seguridad: Estamos obligados a notificar cualquier brecha o violación de seguridad si los datos personales de los usuarios han sido afectados, o en caso de que la filtración constituya un riesgo para los derechos y las libertades de las personas físicas.
3. Cláusulas e información: Al usuario le debe quedar claro quién trata sus datos, como los trata y por qué los trata.
 - a. Informar a los propietarios de los datos: Se debe informar a los afectados por el tratamiento de al menos:
 - i. Nombre del responsable
 - ii. Legitimación para la recogida de los datos
 - iii. Para qué se usan
 - iv. Cómo ejercitar sus derechos
 - b. Plazo de conservación de datos: Se debe informar del plazo de conservación de los datos. No existe un plazo mínimo de conservación único, dependiendo de la documentación que se trate el período determinado es uno u otro. Hay que incluir cláusulas de elección a los proveedores que nos realizan servicios, ya que ellos también están obligados a cumplir la normativa.
4. Delegado de protección de datos: Para determinados tratamientos de datos que supongan riesgos para los derechos y libertades de las personas, como datos o históricos psicológicos, es obligatoria la existencia de un delegado de protección de datos.
5. Evaluación de impacto en la protección de datos personales: Se debe realizar un análisis previo de los riesgos que conlleva para las personas dejar los datos en nuestra aplicación.
6. Códigos de conducta y certificados: Podemos acogernos a códigos de conducta y certificados que acrediten que cumplimos la normativa. Muchas empresas solicitan esto para trabajar con otra empresa.
7. Atención de derechos solicitados por los afectados: Debemos proporcionar ciertos derechos a los clientes, estos derechos son acceso, rectificación, cancelación, oposición, limitación y portabilidad de los datos.

8. Ficheros: Debemos identificar los tratamientos que contengan datos de carácter personal, cada tratamiento se considera como un fichero. Por ejemplo, datos sobre empleados o clientes, se considerarán ficheros o datos.
9. Registro interno de las actividades de tratamiento: Solo ciertas actividades que traten datos específicos (datos referentes a salud o menores, etc.) deberán llevar a cabo un registro interno de estos ficheros.
10. Documento de seguridad: Es un documento en el cual se resume todo aquello relativo al tratamiento de datos personales. En este documento se incluyen, por ejemplo:
 - a. Contratos de encargo de tratamiento
 - b. Empleados que acceden a los datos
 - c. Registro de incidencias
11. Formación: El responsable del fichero, es decir, la persona encargada del tratamiento de los datos deberá tener una mínima formación en la materia para tratar los datos de acuerdo con la normativa.
12. Auditorías periódicas: Se deben realizar periódicamente todos estos apartados para corroborar que cumplimos las normativas. Estos informes son revisados por el Responsable de Seguridad, que lleva las conclusiones al responsable del fichero para que elija las medidas correctoras necesarias. En caso de ser autónomos, estas dos figuras somos nosotros mismos.

Podemos enfrentarnos en caso de no cumplir esta normativa, a:

- Responsabilidades tanto legales como civiles.
- Multas relacionadas con la protección de datos.
- Daños de la reputación de la empresa.
- Pérdidas monetarias (indemnizaciones o inversiones para subsanar los fallos).

7. Ética informática

7.1. Definición

La ética informática es la disciplina que analiza problemas éticos que son creados por la tecnología de los ordenadores, también los que son transformados o agravados por la misma. El origen es debido a la introducción masiva de los ordenadores en muchos ámbitos de nuestra vida social.

Según Moore (2005) la ética informática se define como la disciplina que identifica, analiza la naturaleza y el impacto social de las tecnologías de la información y la comunicación en los valores humanos y sociales; estos son: Salud, riqueza, trabajo, libertad, privacidad, seguridad o la autorrealización personal, democracia, conocimiento, entre otros. Además, involucra la formulación y justificación de políticas para dirigir nuestras acciones y hacer un uso ético de estas tecnologías.

7.2. Historia de la ética informática

El estudio de la ética informática fue estudiado por Norbert Wiener entre el año 1948 y 1963. Wiener denominó “cibernética” a la ciencia responsable del estudio de dichos sistemas de información retroalimentados, y trato sobre este nuevo campo y sus preocupaciones éticas relacionadas en su libro de 1948, *Cybernetics*. En el año 1950, Wiener publicó su segundo libro, llamado “The Human Use of Human Beings”, para profundizar en los asuntos éticos que rodean la tecnología de la información y exponer los orígenes básicos de la ética computacional.

Un año más tarde, se cometió el primer delito informático. Un programador fue capaz de diseñar un código para evitar que su cuenta bancaria estuviera marcada como sobregirada.

En el año 1976 Walter Maner desde el punto de vista de la ética médica, observó cómo los problemas éticos tradicionales se veían alterados con el uso de los ordenadores.

Creó el término **Computer ethics** para hacer referencia al área de estudio que examina “los problemas éticos agravados, transformados o creados por la tecnología informática”.

Deborah Johnson colaboró con Walter Maner. Ella no pensaba que los ordenadores generaran nuevos problemas éticos, sino que planteaban problemas tradicionales de forma nueva y en otras áreas que no se habían aplicado antes. El trabajo de ella se consideró una referencia para diversos problemas éticos. Algunos de los puntos importantes de su trabajo fueron los siguientes:

- Propiedad del software y propiedad Intelectual, computación y privacidad
- Responsabilidad de los profesionales de la informática
- Acceso no autorizado a ordenadores
- Impacto de internet en la democracia.

Hoy en día la informática es parte de nuestra sociedad, los dilemas éticos provocadas por el uso de la informática son muy elevadas pudiendo llegar a cambiar nuestro estilo de vida.

7.3. Discusión actual

El problema actual es que hay una falta de reglamentación en cómo utilizar estas nuevas tecnologías que posibilitan nuevas actividades para las cuales no hay o no se perciben con nitidez principios de actuación claros. Por lo tanto, es difícil identificar problemas éticos cuando surgen, así como una forma de actuación.

Las personas con responsabilidades en el área de diseño o gestión de sistemas de información cada vez han de tomar más decisiones sobre problemas que no se resuelven con lo legal y lo casi-legal, sino que rozan lo ético mismo.

El avance rápido de la tecnología requiere una revisión constante de las formas de proceder por lo que es muy difícil de listar y catalogar cada una de ellas.

Los temas más destacados actualmente son:

- **Brecha digital**
- **Contenido.** Debate acerca de cuándo los contenidos son buenos, aportan valor o son ejemplos para imitar.
- **Conocimientos y propiedad.** Relación entre quien posee la propiedad de la información y acerca de los peligros de la comercialización de esta información.
- **Participación y reglas de juego.** Discusión acerca del gobierno y la democracia digital. Así como las reglas del juego que lo regulan.
- **Hackers, piratas informáticos y terrorismo.** Discusión sobre la forma de actuar de estas personas, así como la criminalización de sus actos.
- **Protección de datos personales.** Debate sobre cómo proteger los derechos individuales en internet, cuánto tiempo y qué tipo de datos deberían almacenar los proveedores de servicios de internet y si esta protección de datos contrapone con amenazas virtuales.

La complejidad radica en la facilidad de acceso a los dispositivos tecnológicos y como estos pueden ser usadas en contra de los derechos de las personas. Tampoco existen penas de cárcel para las nuevas formas de crímenes que se producen en internet, así como la dificultad de buscar responsables.

7.4. Responsabilidades como profesionales

La tecnología se encuentra en todos los ámbitos de nuestra vida diaria y avanza a un ritmo muy rápido. Los ingenieros saben que un proyecto informático puede llegar a una cantidad considerable de personas, así como procesar una cantidad enorme de información. El impacto del trabajo del ingeniero informático puede ser enorme en nuestra sociedad, pudiendo llegar a cambiarla. Por lo que las responsabilidades a las que se enfrenta un informático son variadas.

7.4.1. Sociedad

Como profesionales, debemos actuar por un interés general. Es nuestra responsabilidad de que nuestro proyecto satisfaga a los distintos componentes de nuestra sociedad, así como empresarios, clientes, etc.

Por otro lado, somos conscientes del impacto (tanto positivo como negativo) que puede llegar a generar nuestro proyecto informático, por lo que es responsabilidad nuestra supervisar y determinar el trayecto de nuestro proyecto, así como aclarar a las personas y autoridades cualquier peligro real o potencial para el usuario. Las amenazas a la privacidad es uno de los temas más conocidos en materia de ética informática, otros temas muy recurrentes son decisiones tomadas por computadoras (IA), propiedad intelectual, la confianza y la veracidad de la información que está siendo comunicada, etc.

7.4.2. Cliente y empresario

Los ingenieros informáticos debemos actuar de manera en que se representen los mejores intereses para nuestros clientes y empresarios, consistentemente con el interés general.

Como ingenieros proporcionamos servicios dentro de nuestra área de competencia. Tenemos que ser honestos con nuestra formación y experiencia laboral.

No debemos utilizar software obtenido de manera ilegal o no ética. Utilizamos información de nuestros clientes siempre que se haya autorizado de forma legal.

Cualquier información confidencial o situación indebida que haya surgido durante el desarrollo del trabajo, se documentara para que se tenga constancia siempre y cuando esté dentro del marco legal.

7.4.3. Producto

Como ingenieros debemos garantizar que nuestro proyecto y las modificaciones relacionadas cumplen los estándares más elevados posibles.

Promover la máxima calidad, coste aceptable, y un plazo razonable. Dentro del desarrollo de nuestro proyecto tenemos que garantizar que los objetivos son alcanzables, identificar, definir y examinar temas éticos, económicos, culturales, legales y medioambientales relacionados con el proyecto.

Garantizar estimaciones cuantitativas realistas de coste, plazos, personal y proporcionar una evaluación de la incertidumbre de esas estimaciones.

Garantizar pruebas, depuraciones y revisiones de nuestro proyecto.

Garantizar la confidencialidad de aquellos usuarios que van a verse afectados por nuestro proyecto. Manteniendo la integridad de los datos.

Tratar todas las formas del mantenimiento de nuestro proyecto informático.

7.4.4. Persona

Como ingenieros informáticos tenemos que seguir aprendiendo y adaptarnos a las nuevas tecnologías que van surgiendo. Promover un enfoque ético en la práctica de la profesión, así como promover unos valores éticos dentro del campo de la informática, respetando la formación, personalidad y estatus social de los demás compañeros de profesión.

Mejorar nuestras capacidades para realizar proyectos de calidad, seguro, fiables y útil con un coste razonable.

7.5. Ejemplos cuestiones éticas

Caso Edward Snowden. Edward Joseph Snowden es un consultor tecnológico estadounidense, informante y antiguo empleado de la CIA Y la NSA.

En el año 2013, a través de los periódicos The Guardian y The Washington Post, Snowden hizo públicos documentos clasificados como alto secreto sobre varios programas de la NSA, incluyendo los programas de vigilancia masiva PRISM Y XKeyscore.

Caso FC Barcelona.

El Barcelona contrató una empresa que utilizaba cuentas falsas en las redes para criticar a jugadores y opositores. La empresa principalmente se dedicaba a crear estados de opinión en las redes sociales a través de numerosas cuentas. La polémica surge porque también se dedicaba a erosionar la reputación de jugadores y otras personalidades del entorno del club para generar un clima de adhesión a la junta directiva.

7.6. Cuestiones éticas en nuestro proyecto

Nuestra práctica consiste en la gestión de una empresa autónoma, por lo que la información de las empresas que contraten nuestros servicios, así como los datos de particulares y terceros es de suma importancia. La confidencialidad de los datos es importante, así como su integridad, se debe informar a las empresas que contraten nuestro producto de todos los movimientos de datos que vamos a realizar.

Tampoco tenemos que hacer negocio con los datos de nuestros clientes, ya que hay información sensible como cuentas bancarias, información de particulares, etc. Puede darse el caso de que alguna empresa tenga problemas judiciales o bancarios en cuyo caso pueden pedirnos información de la empresa para revisar cualquier movimiento cuestionable. Si esto llega a suceder, la información se proporcionará a través de una orden judicial.

Puede darse el caso de que la empresa decida realizar estudios sobre las empresas para evaluar su rendimiento, su satisfacción de clientes, su capital y facturación, cantidad de clientes, etc. Esto puede ayudar a la empresa a enfocar su manera de negocio, captación de empresas, socios potenciales, etc. Pero accedemos a la información de nuestros clientes por lo que debemos tener consentimiento por parte de ellos para poder realizar los estudios con sus datos.

En un caso hipotético de un ataque informático tenemos que informar de los gastos, daños y repercusiones. Al mismo tiempo, somos responsables de dar solución a los problemas que pueda generar este tipo de situaciones, sin dejar de lado al cliente.

Avisaremos a nuestros clientes de todo lo sucedido con el ataque informático y solucionando todos los problemas que puedan causar a nuestros clientes, también avisar de los futuros riesgos que pueden llegar a tener en caso de que su información haya caído en malas manos.

Cualquier fallo que pueda tener nuestro proyecto informático es responsabilidad de la empresa hacerse cargo, dando soluciones efectivas y en el menor tiempo posible para que las repercusiones económicas no sean graves

8. Bibliografía

Agencia Estatal, Boletín oficial del Estado – Real Decreto 1837/2008.

<https://www.boe.es/eli/es/rd/2008/11/08/1837>

Xataka – Regulación y actualidad Ingeniería Informática.

<https://www.xataka.com/legislacion-y-derechos/los-ingenieros-informaticos-se-suben-al-ring-por-la-regulacion-del-sector-a-favor-y-en-contra>

Economiapedia – Mercado libre.

<https://economipedia.com/definiciones/mercado-libre.html>

Colegio oficial de Ingeniería Informática de la Comunitat Valenciana – Colegio COIICV. <https://www.coii cv.org/colegio/que-es-el-colegio>

Consejo General Colegios Profesionales – Servicios.

<https://www.ccii.es/servicios>

Consejo General Colegios Profesionales – Beneficios de colegiarse.

<https://www.ccii.es/colegiarse/beneficios-colegiarse>

Consejo General Colegios Profesionales – Colegios autonómicos de Ingeniería Informática. <https://www.ccii.es/quienes-somos/colegios-oficiales>

Wikipedia – Consejo General de Colegios Profesionales de Ingeniería Informática.

https://es.wikipedia.org/wiki/Consejo_General_de_Colegios_Profesionales_de_Ingenier%C3%A1_Inform%C3%A1tica

Wikipedia – Colegio profesional.

https://es.wikipedia.org/wiki/Colegio_profesional

Elblogsalmón – Colegios profesionales.

<https://www.elblogsalmón.com/conceptos-de-economia/que-son-los-colegios-profesionales>

Consejo General del Trabajo Social – Colegios profesionales

http://www.cgtrabajosocial.es/app/webroot/files/consejo/files/GUIA_COLEGIACION.pdf

Colegio Profesional de Ingenieros en Informática de la Región de Murcia – Estudios requeridos. <https://cii-murcia.es/master-habilitante-de-la-profesion/>

Wikipedia – Licencias de Software.

https://es.wikipedia.org/wiki/Licencia_de_software

World Intellectual Property Organization – Licencias Software.

<https://www.wipo.int/export/sites/www/amc/en/docs/denaeguia2018.pdf>

Evaluando – Responsabilidad del proveedor.

<https://www.evaluandoerp.com/contratos-de-software-cual-es-la-responsabilidad-del-proveedor/>

Delvy – Licencias Software. <https://delvy.es/contratos-de-software-abogado-especialista-contratos-software/>

Colegio oficial de ingenieros en informática en el País Vasco - COIIE. Perfiles profesionales en informática. https://www.eiieo.eus/wp-content/uploads/perfiles-informatica_noviembre05.pdf

Consejo general de colegios profesionales – CCII. Competencias profesionales. <https://www.ccii.es/ejercicio-profesional-informatica/competencias-ingenieria-informatica>

Educaweb. Las competencias profesionales.

<https://www.educaweb.com/contenidos/laborales/nuevas-profesiones/competencias-profesionales/>

Universidad de Alicante. Competencias en Ingeniería Informática.

<https://web.ua.es/es/grados/grado-en-ingeneria-informatica/como-es-el-grado.html#Competencias>

CUESTIONES TEMA 2

1. La necesidad dees una de las mayores razones para la comunicación en un proyecto.
 - a. Optimización
 - b. Integridad
 - c. Integración
 - d. Diferenciación
2. Un proyecto está lleno de modificaciones al acta de constitución ¿Quién tiene la responsabilidad principal de decidir si estos cambios son necesarios?
 - a. El PM
 - b. El equipo del proyecto
 - c. El patrocinador
 - d. Los interesados
3. El patrocinador está por comenzar el proceso de inicio de un proyecto. Todos los siguientes ítems son necesarios antes de comenzar este procesos, a excepción de:
 - a. El enunciado del trabajo.
 - b. Estándares de la industria.
 - c. El director del proyecto.
 - d. Requisitos del negocio.
4. ¿Por qué es tan importante que los interesados participen en los distintos procesos de un proyecto?:
 - a. Evita cambios en el alcance.
 - b. Mejora la probabilidad de un cliente satisfecho.
 - c. Definen las restricciones del alcance.
 - d. Sirve para una comunicación más fluida.
5. El PM y su equipo estuvieron dos meses involucrados en el proceso de inicio del proyecto. Después de identificar a los interesados, dieron por finalizado el proceso de inicio. Indique qué se acaba de finalizar y qué está a punto de comenzar:
 - a. Plan para la dirección del proyecto/La ejecución del proyecto.
 - b. Acta de constitución/Planificación del proyecto
 - c. Ejecución del proyecto/Seguimiento y Control.
 - d. Acta de constitución/Ejecución del proyecto.
6. ¿Cuál de las siguientes opciones es verdadera sobre el desarrollo del acta de constitución del proyecto?
 - a. El patrocinador crea el acta de constitución del proyecto y el director la aprueba
 - b. El equipo del proyecto crea el acta de constitución y la Oficina de proyectos la aprueba
 - c. El gerente ejecutivo crea el acta de constitución y el gerente funcional la aprueba
 - d. El director del proyecto crea el acta de constitución y el patrocinador la aprueba
7. ¿Cuál de los siguientes se incluye en el acta de constitución del proyecto?
 - a. Una estrategia de gestión de los riesgos
 - b. Una estimación de los diferentes paquetes de trabajo
 - c. Una estimación detallada de los recursos

“ El Máster en Data Science de CUNEF es específico para el sector financiero y tiene como elemento diferenciador la combinación de ciencia (modelos y técnicas) y experiencia (conocimiento del negocio de las entidades financieras).”

JUAN MANUEL ZANÓN
Director - CRM & Commercial
Intelligence Expert

YGROUP



Convierte el desafío en oportunidad y especialízate en Data Science.

Más de 1.600 acuerdos con empresas

POSTGRADO EN DATA SCIENCE

CUNEF

Excelencia,
futuro, éxito.

- d. El caso de negocio del proyecto
8. Has creado el acta de constitución del proyecto, pero no se pudo aprobar. Tu director y su jefe pidieron que el proyecto se iniciara inmediatamente ¿Cuál de las siguientes es la mejor opción?
 - a. Crear un procesos de control de cambios integrados
 - b. Mostrarle a tu director el impacto de proceder sin una aprobación
 - c. Comenzar el trabajo solo con las actividades de la ruta crítica
 - d. No comenzar el proyecto
9. El PM tiene por primera vez un problema con uno de los miembros del equipo. ¿Cuál es la mejor forma de comunicación para abordar el problema?
 - a. Comunicación formal escrita
 - b. Comunicación formal verbal
 - c. Comunicación informal escrita
 - d. Comunicación informal verbal
10. El resultado más probable de la falta de comunicación en un proyecto es:
 - a. El proyecto se retrase
 - b. Aumente el nivel de confianza
 - c. Se produzcan conflictos
 - d. La gerencia se disguste



Departamento de Lenguajes y Sistemas Informáticos

E.T.S. Ingeniería Informática. Universidad de Sevilla

Avda Reina Mercedes s/n. 41012 Sevilla



e.t.s. ingeniería

informática

CUESTIONES TEMA 3

1. El sistema de numeración de una EDT permite al equipo del proyecto:
 - a. Estimar de forma sistemática los costes de los elementos de la EDT
 - b. Proporcionar la justificación del proyecto
 - c. Identificar el nivel en el que se encuentra cada actividad
 - d. Usarlo en una herramienta software para la dirección de proyectos
2. ¿Cuál de las siguientes es una salida clave del proceso de verificar el alcance?
 - a. Un plan para la gestión del alcance más complejo
 - b. La aceptación del cliente de los entregables del proyecto
 - c. Disponer de estimaciones del cronograma mejorada
 - d. Un sistema de información para la mejora de la dirección de proyectos
3. Durante la ejecución del proyecto, un miembro del equipo se acerca al PM porque no está seguro sobre qué trabajo debe realizar en el proyecto. ¿Cuál de los siguientes documentos contiene descripción detallada de los paquetes de trabajo?:
 - a. Diccionario de la EDT
 - b. Lista de actividades
 - c. Enunciado del alcance del proyecto
 - d. Plan de gestión del Alcance
4. ¿Cuándo debería realizarse el proceso de verificar el alcance?:
 - a. Al final del proyecto
 - b. Al inicio del proyecto
 - c. Al final de cada fase del proyecto
 - d. Durante los procesos de planificación
5. Después de varios años de trabajar en proyectos, te acabas de integrar en la oficina de la dirección de proyectos. Una de las cosas que quieras introducir en tu organización es el valor de la creación y utilización de EDT. Algunos de los directores de proyectos están en desacuerdo porque les entienden que les supondrá un trabajo añadido más. ¿Cuál de las siguientes sería la mejor de las respuestas posibles a dichos directores para convencerlos de usar EDT?:
 - a. LAS EDT evitarán que el trabajo se omita
 - b. Las EDT sólo son necesarias en proyectos grandes
 - c. Las EDT sólo serán necesarias si el proyecto conlleva más de un contrato
 - d. Las EDT son la única manera de identificar los riesgos del proyecto
6. Un PM nuevo te ha pedido un consejo sobre la creación de una EDT. Después de explicarle el proceso, te pregunta qué software debe usar para crear la EDT y que debe hacer con la EDT cuando esté completada. Quizás responda que la imagen no es el resultado más valioso de crear una EDT, sino que es:
 - a. Un diagrama de barra
 - b. La aceptación del equipo
 - c. Las definición de las actividades
 - d. Disponer de una lista de riesgos

POSTGRADO EN DATA SCIENCE

Lidera tu futuro y realiza
prácticas como
científico de datos.

Más de 1.600
acuerdos con
empresas



Departamento de Lenguajes y Sistemas Informáticos

E.T.S. Ingeniería Informática. Universidad de Sevilla

Avda Reina Mercedes s/n. 41012 Sevilla



e.t.s. ingeniería
informática

7. Un PM acaba de ser asignado a un nuevo proyecto y se le ha entregado el acta de constitución del proyecto aprobada. Lo primero que el PM debe hacer es:
 - a. Crear el enunciado del alcance del proyecto
 - b. Confirmar que todos los interesados conocen y aceptan el alcance
 - c. Analizar los riesgos del proyecto
 - d. Iniciar el proyecto en base a un plan para la dirección del proyecto
8. ¿Cuál de las siguientes puede crear el mayor malentendido en el enunciado de la definición del alcance del proyecto?:
 - a. Utilizar un lenguaje impreciso
 - b. Utilizar una estructura y un orden cronológico deficiente
 - c. Pequeñas variaciones en el tamaño de los paquetes de trabajo
 - d. Demasiado detalle
9. El cliente ha entregado el enunciado del alcance del proyecto ¿Cuál será el próximo paso?:
 - a. Ejecutar los paquetes de trabajo.
 - b. Realizar el Plan para la dirección del proyecto.
 - c. Completar la verificación del alcance.
 - d. Realizar el control integrado de cambios.

amazon

McKinsey&Company

KPMG

accenture

pwc

Morgan Stanley

CUNEF

Excelencia,
futuro, éxito.

WUOLAH



**Planificación y Gestión de
Proyectos Informáticos**
04-06-2018
Examen Tipo A

Nombre:

DNI:

TEORÍA (3 PUNTOS). Pregunta correcta: +0,2. Pregunta incorrecta: -0,1

- 1) La holgura de una tarea:
 - a. Siempre es cero
 - b. Es el tiempo máximo que puede durar una tarea en caso de comenzar en su inicio temprano y concluir en su fin tardío
 - c. **Es el tiempo que disponemos para jugar con el inicio de una tarea sin afectar al proyecto**
- 2) La representación gráfica del WBS:
 - a. Nos muestra un grafo no ordenado donde los nodos son las tareas
 - b. **Nos muestra un árbol en que las hojas son las tareas**
 - c. Ninguna de las dos anteriores es correcta
- 3) El objetivo básico del estudio de viabilidad es:
 - a. Determinar las actividades principales dentro del proyecto y desglosarlas en paquetes de trabajo componentes
 - b. **Determinar si la organización cuenta con los recursos y capacidades requeridas para llevar a cabo el proyecto**
 - c. Incorporar el proyecto con las metas estratégicas de la empresa
- 4) Tras la elaboración del correspondiente grafo PERT, una empresa calcula que el camino crítico de uno de sus proyectos es A-F-G. Si la duración de las actividades del camino crítico es 10, 19 y 10 semanas respectivamente, podemos asegurar que:
 - a. **La duración total del proyecto es de 39 semanas**
 - b. La duración total del proyecto es de 19 semanas, ya que es la máxima duración de una de las actividades del camino crítico
 - c. Estas actividades son las únicas que pueden retrasarse sin que se retrase la entrega final del proyecto
- 5) La duración del proyecto es de 15 semanas, la varianza del camino crítico es 3,11 y su desviación estándar 1,76 semanas. La probabilidad de que el proyecto termine en 16 semanas es:
 - a. 62,5%
 - b. **71,2%**
 - c. 77,3%
- 6) Tu empresa está evaluando tres proyectos de inversión mutuamente excluyentes entre sí. El proyecto A tiene un 70% de probabilidad de ganar 5.000€ y un 30% de ganar 10.000€. El proyecto B tiene un 50% de probabilidad de ganar 70.000€ y un 50% de perder 10.000€. El proyecto C tiene un 25% de probabilidad de ganar 20.000€ y un 75% de perder 35.000€. En base al valor monetario esperado, ¿Qué proyecto es el mejor para tu empresa?
 - a. Proyecto A
 - b. **Proyecto B**
 - c. Proyecto C
- 7) Propiedades que se pueden medir en un software para evaluar su calidad:
 - a. **Características operativas, adaptabilidad a nuevos entornos y capacidad para soportar cambios**
 - b. Cumplimiento normativo, planificación temporal y adaptabilidad a nuevos entornos
 - c. Cumplimiento de la planificación temporal y de los requisitos funcionales del proyecto
- 8) Indique cual de las siguientes sentencias es falsa:
 - a. La SQA es una actividad de protección que se aplica en todo el proceso de ingeniería del software
 - b. **Entre las actividades de la SQA no se encuentra la realización de las revisiones técnicas formales**
 - c. Entre las actividades de la SQA se encuentra el control de cambios

“ El Máster en Data Science de CUNEF es específico para el sector financiero y tiene como elemento diferenciador la combinación de ciencia (modelos y técnicas) y experiencia (conocimiento del negocio de las entidades financieras).”

JUAN MANUEL ZANÓN
Director - CRM & Commercial Intelligence Expert

YGROUP



Convierte el desafío en oportunidad y especialízate en Data Science.

Más de 1.600 acuerdos con empresas



POSTGRADO EN DATA SCIENCE

CUNEF

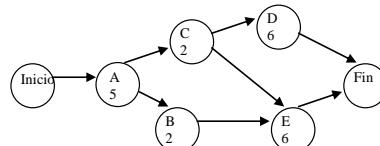
Excelencia, futuro, éxito.

- 9) El estudio de viabilidad del proyecto se realiza desde cuatro perspectivas diferentes:
 - a. Económico, normativo, legal y social
 - b. Económico, técnico, operativo y legal**
 - c. Económico, técnico, operativo y social
- 10) Sobre la gestión de riesgos:
 - a. Para todo riesgo se debe determinar un conjunto de acciones para reducir o eliminar el riesgo
 - b. Los riesgos y las decisiones se registran en un plan de gestión de riesgos al comienzo del proyecto, que se debe actualizar y monitorear a lo largo del mismo**
 - c. En la detección de riesgos, solo se deben identificar y atacar todos los riesgos específicos del proceso
- 11) Para medir el tamaño de la funcionalidad de una aplicación en Puntos de Función se elaboró el cuadro siguiente en el que se indica la cantidad de cada una de las características que aportan Puntos de Función y a su vez, se discrimina cada una de acuerdo al grado de complejidad.

Evaluación					Ponderadores		
Características	Cantidad	Baja	Media	Alta	Baja	Media	Alta
Entradas	10	0	5	5	3	4	6
Salidas	8	3	4	1	4	5	7
Consultas	15	5	3	7	3	4	6
Archivos Internos	3		2	1	7	10	15
Archivos de Interfaz	3	1	2		5	7	10

Siendo 1,2 el Factor de Ajuste de la Complejidad obtenido considerando los distintos factores de ajuste:

- a. Los puntos de función sin ajustar son 212
- b. a) y los puntos de función ajustados son 258,8
- c. a) y los puntos de función ajustados son 254,4**
- 12) El código de ética y práctica profesional de la ACM-IEEE los ingenieros de software se adhieren a los ocho principios siguientes:
 - a. Sociedad, cliente, producto, empresario, juicio, administración, profesión y personal.
 - b. Cliente y empresario, producto, gestión, compañeros, juicio, administración, personal y computadores.
 - c. Sociedad, cliente y empresario, producto, valoración, gestión, profesión, compañeros y personal**
- 13) COCOMO versus PUNTOS DE FUNCIÓN:
 - a. COCOMO estima costes, PUNTOS DE FUNCIÓN no aporta directamente una estimación de costes.**
 - b. Son dos técnicas distintas pero persiguen exactamente los mismos objetivos: dimensionar proyectos, plazos y costes.
 - c. Ninguna de las anteriores es correcta
- 14) La estimación con puntos de función:
 - a. Es una técnica de estimación empírica, igual que PERT
 - b. Es independiente del lenguaje de implementación**
 - c. No es un modelo adecuado para proyectos de tipo “gestión”
- 15) Sea el grafo de precedencia:



¿Cuál de las siguientes afirmaciones es CORRECTA?

- a. No hay caminos críticos
- b. El proyecto tiene siete actividades reales, las tareas precedentes de E son B y C
- c. Tiene más de un camino crítico**



**Planificación y Gestión de
Proyectos Informáticos**
23-06-2017
Examen Tipo A

Nombre:

DNI:

TEORÍA (3 PUNTOS). Pregunta correcta: +0,2. Pregunta incorrecta: -0,1

- 1) Según el modelo de estimación de “Puntos de función”, ¿Cuál de las siguientes afirmaciones es incorrecta?
 - a. **Depende del lenguaje de programación.**
 - b. Es un modelo adecuado para proyectos de gestión.
 - c. Realiza la estimación a partir de los requisitos.
- 2) Para medir el tamaño de la funcionalidad de una aplicación en Puntos de Función se elaboró el cuadro siguiente en el que se indica la cantidad de cada una de las características que aportan Puntos de Función y a su vez, se discrimina cada una de acuerdo al grado de complejidad.

Evaluación					Ponderadores		
Características	Cantidad	Baja	Media	Alta	Baja	Media	Alta
Entradas	13	0	6	7	3	4	6
Salidas	8	5	3		4	5	7
Consultas	17	5	5	7	3	4	6
Archivos Internos	3	1	2		7	10	15
Archivos de Interfaz	3	1	2		5	7	10

Siendo 1,1 el Factor de Ajuste de la Complejidad obtenido considerando los distintos factores de ajuste:

- a. Los puntos de función sin ajustar son 224
- b. a) y los puntos de función ajustados son 242
- c. **a) y los puntos de función ajustados son 246,4**

- 3) Como resultado de la Revisión Técnica Formal:
 - a. Se obtendrá un informe sumarial sobre consideraciones vistas únicamente desde el punto de vista económico, ya que es la parte que realmente condiciona terminar el producto a tiempo.
 - b. **Se obtendrá una lista con los sucesos de la revisión y un informe sumarial de dicha revisión.**
 - c. Se llevan a cabo constantemente a lo largo del proyecto, y únicamente determinan si se aprobará o no la salida del producto.
- 4) El método de planificación PERT es:
 - a. **Una definición en base a una red (grafo dirigido) de dependencias de tareas que llevan al final del proyecto.**
 - b. Una definición en forma de grafo no dirigido de tareas que ayudan a visualizar el Plan del Proyecto.
 - c. Una definición en forma de barras que dan un aspecto visual temporal del Plan del Proyecto.
- 5) En un grafo de actividades (PERT):
 - a. No se consideran relaciones de precedencia entre las actividades.
 - b. **La holgura total de una actividad indica cuánto se puede atrasar su comienzo sin afectar la duración total del proyecto.**
 - c. El fin más tardío de una actividad es lo más tarde que puede comenzar una actividad, afectando con su retraso a la duración total del proyecto.
- 6) Tu empresa está evaluando tres proyectos de inversión mutuamente excluyentes entre sí. El proyecto A tiene un 50% de probabilidad de ganar 50.000€ y un 50% de perder 10.000€. El proyecto B tiene un 30% de probabilidad de ganar 40.000€ y un 70% de ganar 10.000€. El proyecto C tiene un 45% de probabilidad de ganar 50.000€ y un 55% de perder 5.000€. En base al valor monetario esperado, ¿Qué proyecto es el peor para tu empresa?
 - a. Proyecto A
 - b. **Proyecto B**
 - c. Proyecto C

POSTGRADO EN DATA SCIENCE

Lidera tu futuro y realiza prácticas como científico de datos.

Más de 1.600 acuerdos con empresas



amazon

McKinsey&Company

KPMG

accenture

pwc

Morgan Stanley

CUNEF

Excelencia,
futuro, éxito.

- 7) Estudio de viabilidad en un proyecto software:
 - a. Debe hacerse el estudio de viabilidad cuando el coste supere el umbral Boehm.
 - b. **Debe hacerse un estudio de viabilidad como una de las etapas iniciales del ciclo de vida.**
 - c. Solo debe realizarse el estudio de viabilidad cuando el cliente lo solicita.
- 8) Tras la elaboración del correspondiente grafo PERT, una empresa sabe que las holguras de las 4 actividades de uno de sus proyectos son las siguientes: H(A)=3 ; H(B)=2; H(C)=0 ; H(D)=4. Con esa información indica cuál de las siguientes afirmaciones es correcta:
 - a. El máximo retraso que puede experimentar la actividad A sin que se modifique la duración final del proyecto es 2.
 - b. La duración final del proyecto será de 9.
 - c. **Si la duración de la actividad D se reduce en una unidad no se alteraría la duración final del proyecto.**
- 9) Rol del gerente funcional:
 - a. Responsable de la gestión coordinada de proyectos relacionados.
 - b. **Persona clave que desempeña el rol de gestor en un área de la empresa, como recursos humanos, finanzas, contabilidad o adquisiciones.**
 - c. Persona relacionada con la producción y el mantenimiento de los productos o servicios que vende la empresa.
- 10) La ACM y la IEEE ha cooperado para crear un código de ética y práctica profesional. Este código se basa en ocho principios, uno de estos principios es la Valoración, el cual hace referencia a:
 - a. El ingeniero de software mantendrá la valoración de su juicio personal y profesional a lo largo de todo el proyecto.
 - b. **El ingeniero de software mantendrá la integridad e independencia en sus valoraciones profesionales**
 - c. El ingeniero de software tiene la obligación de informar al cliente, teniendo en cuenta sus valoraciones personales.
- 11) En relación con la técnica PERT podemos afirmar que:
 - a. En una actividad el tiempo early es el tiempo máximo en el que se puede iniciar las actividades siguientes sin que ello retrase la duración del proyecto.
 - b. **Una actividad comprendida entre dos actividades no tiene por qué ser crítica.**
 - c. En una actividad cualquiera el tiempo last podría ser menor que el tiempo early de dicha actividad.
- 12) En un proyecto software:
 - a. **Los riesgos suelen ir cambiando a lo largo del mismo, por lo que en instancias predefinidas conviene rehacer las evaluaciones de riesgos.**
 - b. Los riesgos se establecen con precisión en la fase de planificación, ya que suelen ser estables a lo largo del mismo.
 - c. Los riesgos identificados se suelen reducir sistemáticamente, es decir, se desarrollan acciones que permitan aumentar la probabilidad de que un riesgo ocurra y/o disminuya el impacto negativo que produciría.
- 13) La Ley Orgánica 15/1999, de Protección de Datos de Carácter Personal (en adelante LOPDPC) tiene por objeto garantizar y proteger, en lo que concierne al tratamiento de los datos personales:
 - a. Las libertades públicas de las personas físicas.
 - b. Las libertades públicas y los derechos fundamentales de las personas físicas y jurídicas.
 - c. **Las libertades públicas y los derechos fundamentales de las personas físicas.**
- 14) En relación a la patente software en España se protege:
 - a. El código fuente, el código objeto y los manuales de instrucciones.
 - b. En España solo se protege el código fuente mediante patentes.
 - c. **En España no se aceptan las patentes de software.**
- 15) Indica cual de las siguientes afirmaciones es VERDADERA:
 - a. Uno de los grandes inconvenientes de las revisiones técnicas formales es que fomentan la inseguridad en el proyecto.
 - b. Como resultado de una Revisión Técnica Formal se obtiene un informe sobre consideraciones vistas únicamente en la codificación del software, ya que es la parte que condiciona realmente el funcionamiento de la aplicación.
 - c. **La Garantía de Calidad del Software (SQA) es una actividad que se aplica en todo el proceso de ingeniería del software.**

TEORÍA (4 puntos) (Encierre en un círculo la respuesta correcta. Las que haya que contestar utilice exclusivamente los renglones dedicados a ello debajo de la pregunta. Cada respuesta correcta vale 0,4 puntos y cada respuesta incorrecta resta 0,2).

1.- La estimación con puntos de función:

- a) Es una técnica de estimación empírica, al igual que PERT.
- b) Es independiente del lenguaje de implementación.**
- c) No es un modelo adecuado para proyectos “tipo gestión”.

2- Los diagramas de GANTT...

- a)... se usan para la estimación
- b)... se usan para la planificación temporal**
- c)... son unos diagramas de red que permiten calcular los caminos y tiempos límite.

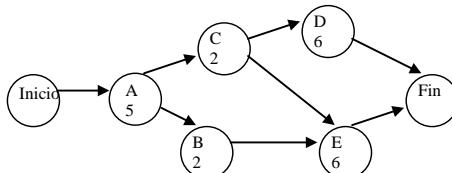
3- El Work Breakdown Structure (WBS):

- a) Es un diagrama que se utiliza para estudiar distintas alternativas, y plasmarlas en forma de árbol.
- b) Se suele utilizar junto con otras técnicas de estimación, por ejemplo las de descomposición.
- c) Identifica las tareas, que luego podremos incluir en diagramas de PERT.**

4- El método de planificación PERT es:

- a) Una definición en base a una red (grafo dirigido) de dependencias de tareas que llevan al final del proyecto.**
- b) Una definición en forma de barras que dan un aspecto visual temporal del Plan del Proyecto.
- c) Una definición en forma de grafo no dirigido de tareas que ayuda a visualizar el Plan del Proyecto.

5- Sea el grafo de precedencia:



¿Cuál de las siguientes afirmaciones es FALSA?

- a) No hay caminos críticos.**
- b) El proyecto tiene cinco actividades reales, las tareas precedentes de E son B y C.
- c) Tiene más de un camino crítico.

6- Como resultado de la Revisión Técnica Formal:

- a) No se obtendrá ningún documento, únicamente se aprobará o no la salida del producto.
- b) Se obtendrá un informe sobre consideraciones vistas únicamente en la codificación del software, ya que es la parte que realmente condiciona el funcionamiento de la aplicación.
- c) Se obtendrá una lista con los sucesos de la revisión y un informe sumarial de dicha revisión.**

7- ¿Cuál de los siguientes NO es un modelo de estimación del software?

- a) WBS (Work Breakdown Structure)**
- b) COCOMO
- c) Técnica Delphi

8- Un proyecto tiene una duración de 3 semanas si trabajan dos personas en su realización. ¿Significa esto que una única persona necesitaría 6 semanas para realizar el mismo proyecto? Razonar adecuadamente la respuesta.

No. Es posible que estas actividades no se puedan realizar en paralelo. Podría incluso darse el caso de que una sola persona tardara lo mismo que dos, si ninguna de estas actividades se pueden realizar en paralelo (puede que no haya recursos suficientes para que se realicen en paralelo).

9- ¿Cuál de las siguientes afirmaciones es correcta?

- a) Los riesgos de un proyecto están vivos durante todo el proyecto y hay que adaptar el plan de riesgos permanentemente**
- b) Los riesgos no cambian durante todo el proyecto
- c) Los riesgos se definen y se gestionan de cualquier manera

Indica cuál sería la respuesta adecuada a las siguientes cuestiones añadiendo una breve justificación de la elección para cada una de ella.

NOTA: La justificación se comentará en clase.

1. Una empresa está haciendo un esfuerzo por mejorar el rendimiento de sus proyectos y crear registros históricos de proyectos pasados ¿Cuál es la mejor manera de lograrlo?
 - a. Crear planes para la dirección de proyectos.
 - b. Crear registros lecciones aprendidas.**
 - c. Crear diagramas de red.
 - d. Crear informes de estado.
2. Usted está trabajando en un proyecto de investigación y desarrollo para lanzar un nuevo producto al mercado. Le quedan pocos días para finalizar el proyecto y está teniendo muchos problemas con los usuarios, el cliente, el patrocinador y los miembros del equipo, entre otros ¿Qué debería haber realizado primero para mitigar este tipo de problemas?
 - a. Determinar los requisitos y expectativas de los interesados.
 - b. Identificar a todos los interesados.**
 - c. Crear una matriz para la gestión de conflictos.
 - d. Mantener una comunicación fluida con los interesados.
3. Usted trabaja como responsable de un proyecto en una empresa con una estructura funcional. Su nivel de autoridad en el proyecto será:
 - a. Alto.
 - b. Medio.
 - c. Muy alto.
 - d. Bajo.**
4. Un PM ha sido asignado a un nuevo proyecto del que dispone de poca experiencia y es bastante más grande de otros proyectos en los que ha trabajado. El PM conoce a un amigo que ya ha dirigido proyectos similares ¿Qué debería hacer el PM?:
 - a. Contactar con el PM del proyecto anterior y pedirle consejos.
 - b. Esperar a la ejecución del proyecto para ver si necesitará ayuda.
 - c. Obtener registros históricos desde la oficina de gestión de proyectos.**
 - d. Asegurarse de que todos los interesados acepten el alcance del proyecto.
5. Cuándo tienen los interesados una mayor influencia en el proyecto:
 - a. Al inicio.**
 - b. En la mitad.
 - c. Al final.
 - d. A lo largo del proyecto.
6. En su proyecto ya se ha finalizado con la identificación de los interesados agregando información sobre sus roles, áreas de conocimiento, necesidades, intereses y expectativas ¿Cuál debería ser el siguiente paso a realizar?:
 - a. Clasificar a los interesados según su influencia e intereses.**
 - b. Evaluar cómo podrían influir los interesados en el proyecto.
 - c. Controlar la participación de los interesados en el proyecto.
 - d. Definir la mejor estrategia de comunicaciones con cada grupo de interesados.

“ El Máster en Data Science de CUNEF es específico para el sector financiero y tiene como elemento diferenciador la combinación de ciencia (modelos y técnicas) y experiencia (conocimiento del negocio de las entidades financieras).”

JUAN MANUEL ZANÓN
Director - CRM & Commercial
Intelligence Expert

YGROUP



Convierte el desafío en oportunidad y especialízate en Data Science.

Más de 1.600 acuerdos con empresas

POSTGRADO EN DATA SCIENCE

CUNEF

Excelencia,
futuro, éxito.

7. Durante la ejecución del proyecto, un miembro del equipo se acerca al PM porque no está seguro sobre qué trabajo debe realizar en el proyecto ¿Cuál de los siguientes documentos contiene descripción detallada de los paquetes de trabajo?:
 - a. **Diccionario de la EDT**
 - b. Lista de actividades
 - c. Enunciado del alcance del proyecto
 - d. Plan de gestión del Alcance
8. Durante una reunión con algunos de los interesados del proyecto, se le pide al PM añadir más trabajo al alcance del proyecto. El PM revisó la correspondencia con el patrocinador antes que se firmara el acta de constitución y éste se negó aportar más financiación para el trabajo que supone modificar el alcance mencionado por los interesados. Lo mejor que el PM puede hacer es:
 - a. Dejar que el patrocinador se entere de la solicitud de los interesados
 - b. Evaluar el impacto sobre el proyecto de añadir más trabajo al alcance
 - c. **Decirle a los interesados que no se puede modificar ese aspecto del alcance**
 - d. Añadir el trabajo si el cronograma del proyecto lo permite
9. Tras recoger los requisitos del producto y definir el alcance del proyecto procederás a hacer:
 - a. Una descomposición a nivel de actividades para estimar el coste del proyecto.
 - b. **Una organización del trabajo del equipo en paquetes de trabajo que ayudará en la comunicación con los interesados.**
 - c. La estructura desglosada de tiempo para estimar la duración del proyecto.
 - d. Un listado de los paquetes de trabajo que una vez realizados completen los entregables
10. ¿Cuál de las siguientes afirmaciones relativas a la EDT no es correcta?:
 - a. La EDT representa todo el trabajo que hay que realizar para ejecutar el proyecto y entregar el producto o servicio solicitado.
 - b. **La EDT incluye todas las tareas necesarias para realizar el proyecto salvo las relacionadas con la gestión del proyecto.**
 - c. La suma del trabajo de todos los niveles inferiores debe corresponder al trabajo total de sus niveles inmediatamente superiores.
 - d. Lo que no aparece en la EDT no forma parte del proyecto y no se ejecuta salvo que se haya integrado y aprobado en el control de cambios del proyecto.

2019

1. El código de ética y práctica profesional de SISTEDES se adhieren a los ocho principios siguientes:
 - a. Sociedad, cliente, valoración, empresario, juicio, gestión, profesión y personal.
 - b. Cliente y empresario, producto, gestión, compañeros, juicio, administración, colegas y uso de computadores.
 - c. Sociedad, cliente y empresario, producto, juicio, administración, profesión, colegas y personal
2. Indica cuál de las siguientes afirmaciones sobre dirección de proyectos es incorrecta:
 - a. El objetivo principal de la dirección de proyectos es cumplir con los requisitos del proyecto.
 - b. Los cinco grupos de procesos en la dirección de proyectos son: iniciación, planificación, ejecución, seguimiento y control y cierre.
 - c. La dirección de proyectos no es la encargada de equilibrar las restricciones contrapuestas del proyecto.
3. Los tipos básicos de relaciones entre fases de un proyecto son:
 - a. Relación secuencial, relación de superposición y relación iterativa.
 - b. Relación de gobernabilidad, relación con los stakeholders y relación bilateral.
 - c. Relación secuencial, relación bilateral y relación de superposición.
4. Una editorial solicita a una empresa de informática que desarrolle un software para un nuevo libro electrónico. El presidente de la empresa de informática asigna a un director de proyecto para que se haga cargo del desarrollo del proyecto. En este ámbito tenemos:
 - a. Patrocinador: Presidente de la empresa de informática. Cliente: Editorial. Usuario: Persona que compra el libro electrónico en el mercado.
 - b. Patrocinador: Director de Proyecto. Vendedor/Socio de Negocio: Editorial.
 - c. Gerente de operaciones: Editorial. PMO: Presidente de la empresa de informática. Usuario: persona que compra el libro en el mercado.
5. La Deontología Profesional:
 - a. Está recogida en normas y códigos deontológicos.
 - b. Está regulada por el Estado.
 - c. No son exigibles a los profesionales de un determinado colectivo.
6. ¿Quién debe cumplir la Ley Orgánica de Protección de Datos?
 - a. El responsable del fichero y quien decide sobre la finalidad, contenido y uso del tratamiento.
 - b. El responsable del fichero y el encargado del tratamiento.
 - c. Persona o personas responsables de los ficheros.
7. Obligaciones de las empresas respecto a los datos personales:
 - a. Proteger los ficheros para preservar la confidencialidad, legitimar los datos personales y declarar los ficheros con datos de carácter personal a la Agencia de Protección de Datos.
 - b. Proteger los ficheros para preservar la confidencialidad e implantar un responsable de seguridad junto con una Auditoría interna o externa.
 - c. Desarrollar un Reglamento de Medidas de Seguridad e informar al Registro General de Protección de Datos sobre los datos de carácter personal sensible.
8. Desarrollamos un software para jugar al ajedrez. Para proteger este programa de la mejor forma posible:
 - a. Lo registramos mediante una Patente del Software.
 - b. Lo registramos en el Registro de la Propiedad Intelectual.
 - c. Las dos anteriores son correctas, ya que ambas ofrecen igual grado de protección.

POSTGRADO EN DATA SCIENCE

Lidera tu futuro y realiza prácticas como científico de datos.

Más de 1.600 acuerdos con empresas

9. Una entrevista se puede estructurar de los siguientes modos:
- Pirámide, TFEA, Cerradas y Abiertas.
 - Especificación Preliminar Individual y Unificada.
 - Pirámide, Embudo y Diamante.**

10. El estudio de viabilidad en un proyecto consiste en:
- Estudio que garantice que el proyecto es factible y es la mejor solución.
 - Estudio que garantice que el proyecto se puede integrar en otros sistemas de la empresa y se puede hacer en plazo y coste.
 - Las dos anteriores son correctas.**

11. Para medir el tamaño de la funcionalidad de una aplicación en Puntos de Función se elaboró el cuadro siguiente en el que se indica la cantidad de cada una de las características que aportan Puntos de Función y a su vez se discrimina cada una de acuerdo al grado de complejidad.

Evaluación	Características	Cantidad	Ponderadores		
			Baja	Media	Alta
	Entradas	13	2	5	6
	Salidas	8	5	3	
	Consultas	17	5	5	7
	Archivos Internos	3	1	2	
	Archivos de Interfaz	3	1	2	

Siendo 1,1 el Factor de Ajuste de la Complejidad obtenido considerando los distintos factores de ajuste:

- Los puntos de función sin ajustar son 220
- a) y los puntos de función ajustados son 242**
- 252 y los puntos de función ajustados son 277,2

12. El software utilizado para controlar una fotocopiadora avanzada requiere 32.000 líneas de código C y 4.200 líneas de código en un lenguaje especializado de Cuarta Generación (4GL). Teniendo en cuenta que en los lenguajes 4GL se cumple que el ratio LDC/PF es 20 y en código C el ratio es 128 LDC/PF, ¿Cuál es el número de puntos de función del software de gestión de la fotocopadoras?

- 1633
- 260
- 460**

Otros

- La estimación con puntos de función:
 - Es una técnica de estimación empírica, al igual que PERT
 - Es dependiente del lenguaje de implementación
 - Es un modelo adecuado para proyectos “Tipo Gestión”**
- ¿Qué modelo COCOMO utilizarías si el proyecto a desarrollar es muy complejo, restrictivo e innovador?
 - Organico
 - Semiacoplado
 - Empotrado**

amazon

McKinsey&Company

KPMG

accenture

pwc

Morgan Stanley

CUNEF

Excelencia,
futuro, éxito.

WUOLAH

3. Indica si las siguientes afirmaciones son verdaderas o falsas. En caso de que sean falsas, indique el por qué.
- a. La garantía de calidad del software (SQA) es una actividad de protección que se aplica en todo el proceso de ingeniería del software. **Verdadero.**
- b. Los proyectos software se diferencian de los proyectos de otras ingenierías en que en el software no hay fase de “fabricación” propiamente dicha, además de que en la mayoría del software se construye “a medida”. **Verdadero.**
- c. Dentro de la formalización de proyectos, OBS (Organization Breakdown Structure) es la técnica que describe la estructura de descomposición del trabajo total en subtrabajos. **Verdadero.**
- d. El camino crítico en el grafo PERT es el cálculo del camino mínimo para llegar a un hito. **Falso, El camino crítico es la secuencia de tareas con dependencias de inicio y fin que condicionan el logro final del proyecto.**



Grado en Ingeniería Informática en Tecnologías Informáticas

PLANIFICACIÓN Y GESTIÓN DE PROYECTOS INFORMÁTICOS

APELLIDOS: _____ NOMBRE: _____

1. Lo mejor sería que las lecciones aprendidas fueran completadas por:
 El jefe de proyecto. El equipo. El patrocinador. Los interesados.

2. La consideración de las operaciones del día a día y del mantenimiento es de crucial importancia para los productos de los proyectos. Las operaciones del día a día y el mantenimiento deben:
 Estar incluidos en las actividades que habrán de llevarse a cabo durante el cierre del proyecto.
 Tener una fase distinta en el ciclo de vida del proyecto, pues se gasta una parte importante de los costos del ciclo de vida en mantenimiento y operaciones.
 No ser vistos como parte del proyecto.
 Ser vistos como un proyecto separado.

3. Una compañía está haciendo un esfuerzo por mejorar el rendimiento de su proyecto y crear registros históricos de proyectos pasados. ¿Cuál es la MEJOR manera de lograrlo?
 Crear planes para la dirección del proyecto. Crear lecciones aprendidas.
 Crear diagramas de red. Crear informes de estado.

4. El equipo del proyecto acaba de completar el cronograma inicial del proyecto y el presupuesto. Lo SIGUIENTE que debes hacer es:
 Identificar los riesgos. Comenzar las iteraciones.
 Determinar los requisitos de las comunicaciones. Crear un diagrama de Gantt.

5. El patrocinador del proyecto acaba de firmar el acta de constitución del proyecto. ¿Qué es lo SIGUIENTE que debemos hacer?
 Comenzar a completar los paquetes de trabajo.
 Validar el alcance.
 Comenzar el control integrado de cambios.
 Empezar a crear los planes de gestión.

6. Un director de proyectos no tiene mucho tiempo para hacer la planificación, pues ya se aproxima la fecha límite para comenzar. Por lo tanto, busca planificar de la manera más efectiva posible. ¿Qué consejo ofrecerías?
 Asegurarse de que tenga un acta de constitución del proyecto firmada y comenzar con la EDT.
 Crear una lista de actividades antes de crear un diagrama de red.
 Documentar todos los riesgos conocidos antes de documentar los supuestos de alto nivel.
 Finalizar el plan de gestión de calidad antes de determinar las métricas de calidad.

POSTGRADO EN DATA SCIENCE

Lidera tu futuro y realiza prácticas como científico de datos.

Más de 1.600 acuerdos con empresas

7. Un miembro del equipo notifica al director del proyecto que las actividades que comprenden un paquete de trabajo ya no resultan apropiadas. Lo MEJOR para el director del proyecto sería estar en qué parte del proceso de la dirección de proyectos:
- Acción correctiva Control integrado de cambios
 Seguimiento y control. Cierre del proyecto.
8. Durante una reunión del equipo, un miembro hace una pregunta respecto de las medidas que se utilizarán en el proyecto para evaluar el desempeño. El miembro del equipo siente que algunas de las medidas relacionadas con las actividades que le han sido asignadas no son medidas válidas. ¿Es MEJOR considerar el proyecto en que parte de los grupos de procesos de dirección de proyectos?
- Cierre. Seguimiento y control. Ejecución. Iniciación.
9. ¿Cuál de las siguientes acciones es la MÁS adecuada durante el grupo de procesos de iniciación?
- Crear una descripción detallada de los entregables del proyecto.
 Familiarizarse con la cultura de la compañía y su estructura en relación con el proyecto.
 Identificar la causa raíz de los problemas.
 Asegurarse de que todos los procesos de dirección de proyectos hayan sido completados.
10. ¿Qué grupo de procesos de dirección de proyectos es el que generalmente consume MÁS tiempo y recursos del proyecto?
- Planificación. Diseño. Integración. Ejecución.

amazon

McKinsey&Company

KPMG

accenture

pwc

Morgan Stanley

CUNEF

Excelencia,
futuro, éxito.

WUOLAH

Por si alguien abandona todos trabajamos en todo (utilizamos una estructura democrática).

Para detectar errores en las estimaciones iremos reestimando el proyecto en cada iteración. Preveremos unos márgenes de error para las dificultades encontradas. Debido al poco tiempo de desarrollo seleccionaremos las funcionalidades principales que necesita el cliente.

En el plan de contingencia se planifican las tareas para prevenir los riesgos y minimizar el impacto en el caso de que aparezcan los riesgos.

En el plan de seguimiento se van a establecer las actuaciones para detectar si un riesgo aparece o no.

Cuestiones rápidas

1. Un plan de proyecto ...

- a. Proporciona visibilidad al progreso del proyecto
- b. Es una representación abstracta de un proceso que proporciona información parcial sobre el proceso software
- c. Es un conjunto de actividades que tienen como objetivo el desarrollo de un producto software
- d. Todas las afirmaciones anteriores son falsas

2. El modelo basado en prototipos ...

- a. Reduce las horas de implementación.
- b. Permite reducir riesgos y costes de desarrollo
- c. El sistema resultante es un prototipo
- d. Es posible utilizarlo sólo con lenguajes orientados a objetos

3. La gestión de proyectos ...

- a. Tiene como objetivo determinar qué hay que hacer, cómo hay que hacerlo y quién tiene que hacerlo.

- b. Está fuertemente condicionada por la implementación del sistema
- c. Está fuertemente condicionada el diseño del sistema
- d. Tiene como objetivo el confeccionar una agenda del proyecto

4. Tenemos tres actividades: A, B y C. A precede a B. Las holguras totales son: HT(A)= 4, HT(B)2, HT(C)/3. Las holguras libres son: HL(A)= 3, HL(B)-2, HL(C)/F3. Si A se retrasa 4 días, ¿qué les ocurre a B y C?

- a. Nada, ya que no sobrepasamos la holgura total de A
- b. La holgura total de B y C pasa a ser de 1 y 2 días respectivamente
- c. B se convierte en crítica**
- d. Ninguna de las anteriores es cierta

5. Ms Project es ...

- a. Una herramienta de seguimiento de proyectos**
- b. Una herramienta de modelado de proyectos
- c. Una herramienta de estimación de costes de proyectos
- d. Una herramienta de desarrollo de proyectos

6. El cono de la incertidumbre ...

- a. Dice que conforme avanza el proyecto menos errores de estimación tenemos.
- b. Dice que conforme avanza el proyecto más errores de estimación descubrimos.
- c. No puedo aplicarlo sino hago el seguimiento del proyecto.**
- d. Todas las anteriores son falsas.

7. ¿Cuál de los siguientes elementos NO es un elemento de configuración (EC)?

- a. Los ejecutables de la aplicación
- b. La estructura de directorios.**
- c. La definición de las tablas de la BD de la aplicación
- d. Todos los anteriores son elementos de configuración.

8. Make es una herramienta case que soporta ...

- a. la planificación de una gestión de configuraciones

- b. el control de cambios
- c. la construcción del sistema**
- d. la gestión de versiones y releases

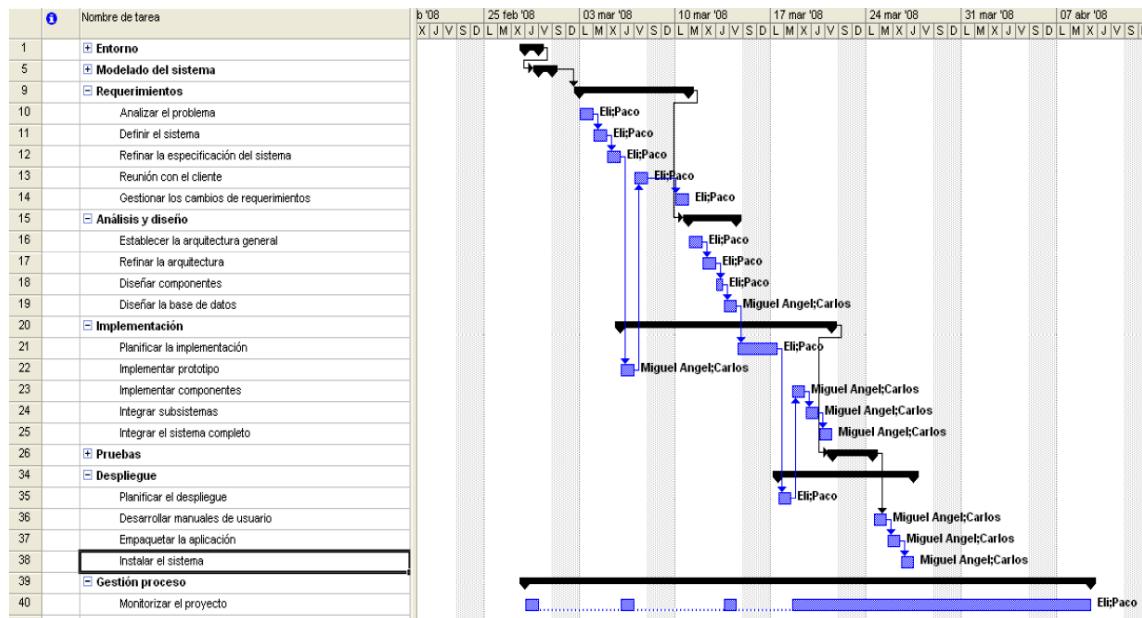
9. Las versiones de un sistema.

- a. Tienen un coste menor que las releases
- b. Suelen superar en número a las releases
- c. Incluyen documentación técnica
- d. Todas son ciertas.**

10. En el modelo UP.

- a. Hay una versión por iteración.
- b. Puede haber más de una versión por iteración.**
- c. Tenemos una release por fase.
- d. Todas las anteriores son ciertas.

11. En el siguiente diagrama Gantt ¿qué modelo de proceso se está siguiendo?



- a. Desarrollo Rápido de aplicaciones.
- b. Cascada

c. Prototipado.

d. Todos los anteriores.

12. En la pirámide de jerarquía de necesidades donde se encuentran la mayoría de jefes de proyecto.

a. No les afecta esta jerarquía, es sólo para empleados.

b. Necesidades sociales.

c. Necesidades seguridad.

d. Necesidades autorrealización.

13. ¿Cuál es la distribución de tiempo dentro de un equipo de desarrollo.

a. 20% Trabajando sólo, 30% actividades no productivas, 50% Interactuando con otra gente.

b. 30% Trabajando sólo, 20% actividades no productivas, 50% Interactuando con otra gente.

c. 30% Trabajando sólo, 20% actividades de gestión, 50% Interactuando con otra gente.

d. 20% Implementando, 30% Interactuando con otra gente y 50% en actividades de gestión.

Analizando los valores de los parámetros EVA de la imagen...

4. Dados los siguientes datos de un proyecto. Explica cuál es la situación del proyecto para cada una de las tareas. ¿La tarea 1 se ha terminado? ¿Cómo podemos solucionar los problemas del proyecto? (2,5p)

Tarea	BCWS	BCWP	ACWP
Tarea R	2800	1800	2600
Tarea 1	600	600	1000
Tarea 2	1200	0	0
Tarea 3	1000	1200	800
Tarea 4	0	0	800

1.- ¿Qué podemos decir del proyecto en su conjunto? *

2 puntos

- a) El proyecto va adelantado y su productividad es baja
- b) El proyecto va atrasado y su productividad es alta
- c) El proyecto va adelantado y su productividad es alta
- d) El proyecto va atrasado y su productividad es baja

2.- ¿Qué podemos decir de la tarea
1? *

2 puntos

- a) No podemos saber si ha terminado sin conocer su BAC
- b) Ha gastado menos de lo previsto
- c) Va adelantada respecto a la agenda
- d) Va atrasada respecto a la agenda

3.- ¿Qué podemos decir de la tarea 2? *

2 puntos

- a) No ha comenzado
- b) Debería haber comenzado
- c) Va retrasada
- d) Todas son ciertas

4.- ¿Qué podemos decir de la tarea 3? *

2 puntos

- a) No ha comenzado
- b) Va retrasada
- c) Está gastando menos de lo previsto
- d) Todas son falsas

5.- ¿Qué podemos decir de la tarea 4? *

2 puntos

- a) Debería haber comenzado y no lo ha hecho
- b) El valor BCWP es erróneo
- c) El valor ACWP es erróneo

Test de autoevaluación Protección de datos

*Obligatorio

Según la LOPD ¿A partir de qué edad puede un menor expresar el consentimiento en términos de protección de datos? *

1 punto

- 16 años
- 18 años
- 14 años
- El padre o tutor legal será el que lo haga en su nombre mientras forme parte de la unidad familiar

Según el RGPD, el RAT (Registro de Actividades de Tratamiento) es obligatorio para: *

1 punto

- Cualquier empresa pública o privada
- Solo para la Administración Pública
- Empresas de 250 empleados o más
- Empresas de menos de 250 empleados y PYMES

La LOPD no será de aplicación en el caso de: *

1 punto

- Materias consideradas clasificadas
- Datos de personas fallecidas
- Datos ya excluidos en el RGPD
- Todas son ciertas



La normativa vigente en España en materia de protección de datos es: * 1 punto

- LOPD 15/1999 y RGPD UE de 2016
- RGDP UE de 2018 y LOPD 3/2018
- RGDP UE de 2016 y LOPD 3/2018
- Actualmente en nuestro Estado solo nos afecta la LOPD 3/2018

De los siguientes derechos regulados en la LOPD, ¿Cuál es un derecho digital * 1 punto

- Derecho de supresión
- Derecho a la intimidad en el ámbito laboral
- Derecho de rectificación
- Derecho de oposición

Enviar

Este formulario se creó en Universitat d'Alacant / Universidad de Alicante. [Notificar uso inadecuado](#)

Google Formularios



Test de autoevaluación Propiedad Intelectual

*Obligatorio

Según la LPI, ¿Cuál de las siguientes figuras tiene un plazo de protección indefinido? * 1 punto

- Marcas
- Secretos comerciales
- Patentes
- Derechos de Autor

¿Cuál de las siguientes no es una causa legal para poder transmitir los derechos de propiedad intelectual? * 1 punto

- mortis causa
- En virtud de relación laboral
- inter vivos
- En virtud de relación mercantil

¿Cuál de los siguientes tipos de licencia no es propia del software libre? * 1 punto

- GPL
- Creative Commons
- BSD
- OEM



El plazo de protección de los derechos de autor según la LPI es: *

1 punto

- 20 años a partir de la fecha de registro
- 70 años a partir del fallecimiento del autor
- 70 años a partir de la fecha de registro
- 20 años a partir del fallecimiento del autor

¿Cuál de los siguientes tipos de licencia no es propia del software privativo? *

1 punto

- EULA
- Apache
- ROK
- PKC

Enviar

Este formulario se creó en Universitat d'Alacant / Universidad de Alicante. [Notificar uso inadecuado](#)

Google Formularios



4. Dados los siguientes datos de un proyecto. Explica cuál es la situación del proyecto para cada una de las tareas. ¿La tarea 1 se ha terminado? ¿Cómo podemos solucionar los problemas del proyecto? (2,5p)

Tarea	BCWS	BCWP	ACWP
Tarea R	2800	1800	2600
Tarea 1	600	600	1000
Tarea 2	1200	0	0
Tarea 3	1000	1200	800
Tarea 4	0	0	800

1.- ¿Qué podemos decir del proyecto en su conjunto? * 2 puntos

- a) El proyecto va adelantado y su productividad es baja
- b) El proyecto va atrasado y su productividad es alta
- c) El proyecto va adelantado y su productividad es alta
- d) El proyecto va atrasado y su productividad es baja

2.- ¿Qué podemos decir de la tarea 1? * 2 puntos

- a) No podemos saber si ha terminado sin conocer su BAC
- b) Ha gastado menos de lo previsto
- c) Va adelantada respecto a la agenda
- d) Va atrasada respecto a la agenda

3.- ¿Qué podemos decir de la tarea 2? * 2 puntos

- a) No ha comenzado
- b) Debería haber comenzado
- c) Va retrasada
- d) Todas son ciertas

4.- ¿Qué podemos decir de la tarea 3? * 2 puntos

- a) No ha comenzado
- b) Va retrasada
- c) Está gastando menos de lo previsto
- d) Todas son falsas

5.- ¿Qué podemos decir de la tarea 4? * 2 puntos

- a) Debería haber comenzado y no lo ha hecho
- b) El valor BCWP es erróneo
- c) El valor ACWP es erróneo
- d) No debería haber comenzado, pero sí lo ha hecho

Test de autoevaluación Protección de datos

*Obligatorio

Según la LOPD ¿A partir de qué edad puede un menor expresar el consentimiento en términos de protección de datos? *

1 punto

16 años

18 años

14 años

El padre o tutor legal será el que lo haga en su nombre mientras forme parte de la unidad familiar

Según el RGPD, el RAT (Registro de Actividades de Tratamiento) es obligatorio para: *

1 punto

Cualquier empresa pública o privada

Solo para la Administración Pública

Empresas de 250 empleados o más

Empresas de menos de 250 empleados y PYMES

La LOPD no será de aplicación en el caso de: *

1 punto

Materias consideradas clasificadas

Datos de personas fallecidas

Datos ya excluidos en el RGPD

Todas son ciertas

La normativa vigente en España en materia de protección de datos es: *

1 punto

LOPD 15/1999 y RGPD UE de 2016

RGDP UE de 2018 y LOPD 3/2018

RGDP UE de 2016 y LOPD 3/2018

Actualmente en nuestro Estado solo nos afecta la LOPD 3/2018

De los siguientes derechos regulados en la LOPD, ¿Cual es un derecho digital *

1 punto

Derecho de supresión

Derecho a la intimidad en el ámbito laboral

Derecho de rectificación

Derecho de oposición

Según la LPI, ¿Cuál de las siguientes figuras tiene un plazo de protección indefinido? *

1 punto

Marcas

Secretos comerciales

Patentes

Derechos de Autor

¿Cuál de las siguientes no es una causa legal para poder transmitir los derechos de propiedad intelectual? *

1 punto

mortis causa

En virtud de relación laboral

inter vivos

En virtud de relación mercantil

¿Cuál de los siguientes tipos de licencia no es propia del software libre? *

1 punto

GPL

Creative Commons

BSD

OEM

El plazo de protección de los derechos de autor según la LPI es: *

1 punto

20 años a partir de la fecha de registro

70 años a partir del fallecimiento del autor

70 años a partir de la fecha de registro

20 años a partir del fallecimiento del autor

¿Cuál de los siguientes tipos de licencia no es propia del software privativo? *

1 punto

EULA

Apache

ROK

PKC

1. En el registro de la propiedad intelectual de un software bajo una relación laboral, ¿para qué sirve el documento de declaración de autor asalariado de un empleado?
 - a. Para la cesión de los derechos de explotación del software del empleado a la empresa.
2. En el análisis EVA, la tarea/proyecto va con retraso según la agenda si:
 - a. BCWP < BCWS
3. Según la LOPDGDD ¿A partir de qué edad puede un menor expresar el consentimiento en términos de protección de datos?
 - a. 14
4. Según la Ley de Propiedad Intelectual, ¿Cuál de las siguientes figuras tiene un plazo de protección indefinido?
 - a. Secretos Comerciales
5. En el modelo UP
 - a. Las iteraciones pueden tener duraciones distintas
 - b. Puede haber más de una versión por iteración
6. ¿Qué tarea no podemos realizar en Microsoft Project?
 - a. La planificación detallada de todo el proyecto
7. La cadena de actividades que determina la duración global mínima de un proyecto se conoce como
 - a. Camino Crítico
8. En el análisis EVA, el indicador CPI se calcula como:
 - a. BCWP / ACWP
9. ¿Cuál de los siguientes parámetros ECA nos informa de la cantidad de trabajo planificado al final del proyecto/tarea?
 - a. BAC
10. En el análisis EVA, el indicador SV se calcula como:
 - a. BCWP - BCWS
11. ¿Qué aspecto ha sido el más importante a la hora de establecer los objetivos de las fases de elaboración de construcción en el modelo UP seguido para la planificación del software de gestión de autónomos?
 - a. Los requerimientos funcionales y no funcionales.
12. En el análisis EVA, la tarea/proyecto está por debajo de lo presupuestado si:
 - a. BCWP > ACWS
13. En el análisis EVA, la tarea/proyecto está por encima de lo presupuestado si:
 - a. BCWP < ACWS
14. Un plan de proyecto
 - a. Proporciona visibilidad al progreso del proyecto
15. ¿Cuál es la distribución de tiempo dentro de un desarrollo de proyecto?
 - a. 30% Trabajando sólo, 20% actividades no productivas, 50% Interactuando con otra gente.
16. ¿Qué técnica de estimación de costes se basa en contabilizar pantallas, informes y módulos de cálculo?
 - a. Puntos objeto
17. Una de las recomendaciones que se deben tener en cuenta durante la estimación de recursos y tiempo de un proyecto informático es
 - a. Identificar las tareas que consumen más recursos.
- 18.

19. ¿Cuál de las siguientes no es una causa legal para poder transmitir los derechos de Propiedad Intelectual?
- En virtud de relación mercantil.
20. En una iteración del Plan detallado se establece una actividad de revisión de los riesgos del proyecto. Dicha actividad consiste en:
- Revisar los identificadores potenciales que figuran en el plan
21. La LOPDGDD no será de aplicación en caso de
- Materias consideradas clasificadas
 - Datos de personas fallecidas
 - Datos ya excluidos en el RGPD
22. En caso de producirse un retraso en una actividad, ¿cuál de las siguientes medidas sería la que menos afecta a la agenda del proyecto?
- Retrasarla según su holgura libre.
23. El modelo basado en prototipos
- Permite reducir riesgos y costes de desarrollo
24. ¿Cuál de las siguientes no es una técnica de estimación de costes de un proyecto?
- Gantt
25. Monitorizar una agenda tiene como objetivo fundamental
- consiste en comprobar si la agenda real se ajusta a la planificada
26. ¿Cuál de los siguientes modelos de proceso es evolutivo?
- Modelo en espiral, Desarrollo Unificado
27. Las releases de un sistema
- Incluye documentación técnica
 - Coste mayor que las versiones.
 - Son menos en número que las versiones
28. Cuando en el desarrollo del plan del proyecto existen restricciones de tiempo y presupuesto establecidas por el cliente, estas deben
- Deben ser consideradas evaluando su impacto en el proyecto.
29. ¿Hasta qué momento durante la ejecución de un proyecto, la contingencia debe ser considerada?
- Hasta que el tiempo y los avances del proyecto demuestran que el riesgo se ha reducido lo suficiente o desaparecido
30. ¿Qué importancia tiene el RGPD?
- Establece un marco europeo relativo a la protección de las personas físicas en lo que respecta al tratamiento de sus datos personales y a la libre circulación de estos
31. ¿Qué podemos decir de una tarea si conocemos los valores BCWS=BCWP=600 y ACWP=1000?
- no podemos saber si ha terminado sin conocer su BAC
32. Muchas organizaciones fortalecen sus equipos de gestión de negocios, para que estos sean identificados a priori, pero lo que ocurre realmente es
- a.
33. ¿Qué podemos decir de una tarea si conocemos los valores BCWS=1200 y BCWP=ACWP?
- a.
34. If BCWS 1000, BCWP 1200, and ACWP 1300, the project is:
- Ahead of schedule and over budget

35. Dado que un riesgo siempre va a existir hay que
- Establecer mecanismos de seguimiento y prevención.
36. El Plan detallado de un proyecto
- Todas son falsas
37. Respecto a la protección de datos del software de gestión de autónomos, indica cuál de las siguientes sería una acción obligatoria para cumplir con la legislación actual en el caso de desarrollarla como aplicación web
- Incluir un apartado de aviso legal en la web.
38. En la pirámide de jerarquía de necesidades donde se encuentran la mayoría de jefes de proyecto
- Necesidades autorrealización.
39. Las características del software libre son
- Se distribuye libremente y es de código abierto
40. ¿Qué podemos decir de una tarea si conocemos los valores $BCWS=BCWP=0$ y $ACWP=800$?
- El valor ACWP es erróneo.
41. ¿Cuál de los siguientes tipos de licencia no es propia del software privativo
- Apache
42. En el análisis EVA, la tarea/proyecto va adelantada según la agenda si
- $BCWP > BCWS$
43. En el caso de que hayamos desarrollado un software de gestión de autónomos en forma de aplicación de escritorio, ¿quién debería encargarse del cumplimiento de la legislación a nivel de protección de datos?
- Nosotros como desarrolladores de la aplicación.
44. Tenemos un proyecto con los siguientes datos del proyecto EVA, cual de las siguientes afirmaciones es falsa.

Tarea	BCWS	BCWP	ACWP	BAC
Tarea0	2800	1800	1800	2600
Tarea1	600	600	600	1200
Tarea2	1200	0	600	2600
Tarea3	1000	1200	800	2000
Tarea4	0	0	0	600

La tarea 0 va retrasada y consumiendo más de lo previsto



GESTIÓN DE PROYECTOS INFORMATICOS

Convocatoria de JULIO. TEORIA

6 de julio de 2016

Apellidos, Nombre: _____

Grupo de prácticas: _____

- 1.- Explica qué es la visibilidad de un proceso software y por qué es importante que un proceso sea visible. Pon tres ejemplos que den visibilidad a un proyecto (2.5 p)

La visibilidad de un proceso software consiste en conocer en cada momento exactamente qué es lo que se está haciendo en cada momento del desarrollo. Dado que el software es un producto intangible, la única forma de hacer que el proceso de desarrollo muestre el estado del producto en construcción es mediante la documentación generada durante el proceso.

La posibilidad de "ver" en qué estado del desarrollo se encuentra un producto software resulta de vital importancia a la hora de gestionar un proyecto software, de forma que se pueda hacer una planificación efectiva, y monitorizar y controlar el desarrollo de dicho software.

Ejemplos concretos pueden ser: utilizar 3 niveles en el plan, utilización de estándares, hitos (objetivos definidos), Gantt de seguimiento, informe de iteración, etc.

- 2.- Explica en que consiste la estimación con puntos de función. Indica las ventajas de la estimación de puntos de función respecto a líneas de código y de la estimación mediante puntos objeto respecto a puntos de función (2.5 p).

La técnica de estimación de puntos de función se basa en la contabilización de unos contadores (características del programa):

entradas y salidas externas.

interacciones de usuario.

interfaces externas.

ficheros usados por el sistema.

Se asocia un peso con cada uno de estos contadores y los puntos de función se calculan multiplicando cada factor por su peso y sumando todos ellos

Puntos de función frente a líneas de código.

Son independientes del lenguaje de programación

Pueden calcularse a partir de la especificación

Usa información del dominio del problema

Resulta más fácil a la hora de reusar componentes

Se encamina a aproximaciones orientadas a objetos

Puntos objeto frente a puntos de función.

- Son más fáciles de estimar a partir de una especificación que los puntos de función, ya que solamente consideran pantallas, informes y módulos 3GL

- Por lo tanto pueden estimarse en fases tempranas del proceso de desarrollo. En estas etapas resulta muy difícil estimar el número de líneas de código de un sistema.

3.- ¿Qué es el modelo P-CMM? ¿Qué etapas tiene? Explica cada una de ellas. ¿Cuáles son sus objetivos (2,5p)

Pretende ser un marco para la gestión del trabajo realizado por la gente implicada en el desarrollo del software

Es un modelo de cinco etapas

Inicial. Gestión de rec. humanos "ad-hoc"

Repetible. Se desarrollan políticas para mejora de las capacidades (aptitudes)

Definido. Gestión de rec. humanos estandarizada para la organización

Gestionado. Se establecen metas cuantitativas para la gestión de recursos humanos

Optimizado. Se realiza un esfuerzo continuado para mejorar la competencia y motivación en el trabajo

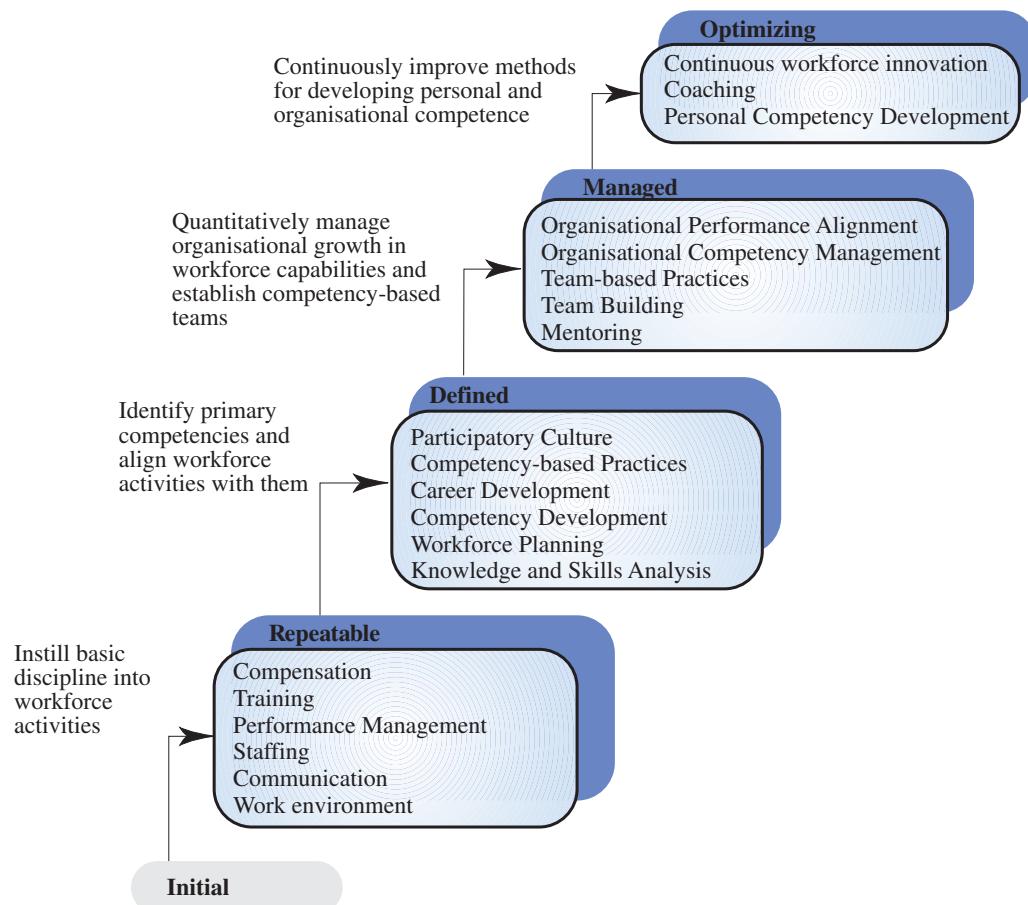
Sus objetivos son:

Mejorar las capacidades de la organización mejorando las capacidades de trabajo de la gente implicada

Asegurar que las capacidades para el desarrollo del software no conciernen a un número pequeño de individuos

Igualar la motivación de los individuos con la de la organización

Ayudar a la "retención" de gente con conocimientos y habilidades críticas



4.- ¿Qué significa que en una fecha determinada una actividad de un proyecto tiene un SPI de 1,5? (0,5 p)

Vamos muy bien respecto al tiempo en el plan. Necesitamos el 50% menos de tiempo para completar el proyecto.

- ¿Qué significa que en una fecha determinada una actividad de un proyecto tiene un CPI de 0,8? (0,5 p)

Que tenemos una productividad baja. Estamos gastando un 25% más de lo que toca.

- Estamos realizando un análisis EVA de un proyecto y resulta que en una fecha determinada el indicador SPI es menor que 1 y el indicador CPI es mayor que 1. ¿Qué significan estos valores? ¿Qué debemos de hacer?

(0,75 p)

Que tenemos una productividad alta, pero vamos retrasados. Debemos de contratar a más gente.

- Estamos realizando un análisis EVA de un proyecto y en una fecha determinada tenemos los siguientes valores para una actividad:

BAC	BCWP	BCWS	ACWP
2000 €	1500 €	1000 €	1500 €

En base a estos datos, ¿Qué podemos afirmar respecto al progreso y coste de esta actividad? Debes justificar tu respuesta realizando los cálculos que consideres necesarios. (0,75 p)

Podemos afirmar que vamos adelantados(BCWP) y gastando justamente lo presupuestado (ACWP). En concreto hemos hecho un 50% más de lo esperado, llevamos hecho un 75% de la tarea cuando planificamos llevar el 50%



GESTIÓN DE PROYECTOS INFORMATICOS

Convocatoria de JULIO. PRÁCTICAS

6 de julio de 2016

Apellidos, Nombre: _____

Grupo de prácticas: _____

1.- Explica los pasos seguidos en prácticas dentro Microsoft Project para asignar recursos. (0,5p). Explica las diferentes formas que existen en Microsoft Project para hacer el seguimiento de un proyecto. (0,5p)

- a) Crearíamos los recursos.
- b) Asignaríamos los recursos a cada una de las tareas usando el filtro de recursos para facilitar la tarea.
- c) Comprobaríamos el uso de recursos mediante el informe de “uso de recursos” con el fin de verificar que no existen sobreasignaciones.

Para hacer el seguimiento de un proyecto en Microsoft Project tenemos las siguientes posibilidades:

- a) Hacer uso de los tiempos early, last y las holguras libres y totales, añadiendo las columnas a la rejilla de datos.
- b) Ver el diagrama Pert.
- c) Visualizando los valores EVA.
- d) Mostrando el Gant de seguimiento.

2.- A la hora de registrar software, ¿Qué diferencia hay entre los impresos de autores1 A-T, Autores TIV y Autores TMC? Pon un ejemplo de uso relacionado con software donde se utilice el impreso A-T, y otro donde se utilice el TIV. (0,5p)

- Autores 1 A-T: a cumplimentar por los titulares de derechos de propiedad intelectual que sean autores y titulares de una obra.

- Autores 1 TIV: a cumplimentar por los titulares que hubieran adquirido los derechos por transmisión inter vivos, ya sea por contrato de cesión o por relación laboral, y siempre que se trate de una primera inscripción de derechos.

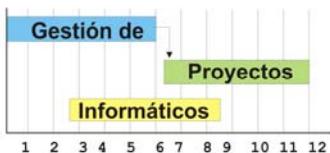
- Autores 1 TMC: a cumplimentar por los herederos que hubieran adquirido los derechos por transmisión mortis causa del autor, siempre que se trate de una primera inscripción de derechos.

Ejemplos: Un profesional autónomo crea un programa software y para registrararlo utiliza el impreso A-T, mientras que una empresa de desarrollo con empleados utilizaría el formulario TIV.

¿En el caso de informática, entre que 3 tipos de impresos (autores 2) tenemos? ¿En nuestra práctica cual se eligió? (0.5 p)

- Programas de ordenador.
- Base de datos.
- Páginas Web.

En nuestra práctica se eligió el primero de ellos, programas de ordenador.



GESTIÓN DE PROYECTOS INFORMATICOS

Convocatoria de JUNIO. TEORIA

9 de junio de 2016

Apellidos, Nombre: _____

Grupo de prácticas: _____

- Indica 5 factores que afecten a la estimación de costes de un proyecto. Propón una medida para ayudar a minimizar el impacto de cada uno de estos factores. (2,5p)

Oportunidad de mercado: A través de calidad y precio. Para evitar que otras compañías entren en mi mercado diferencia mi producto de la competencia. Haría promociones para que a esos nuevos desarrolladores les cueste más competir y entrar.

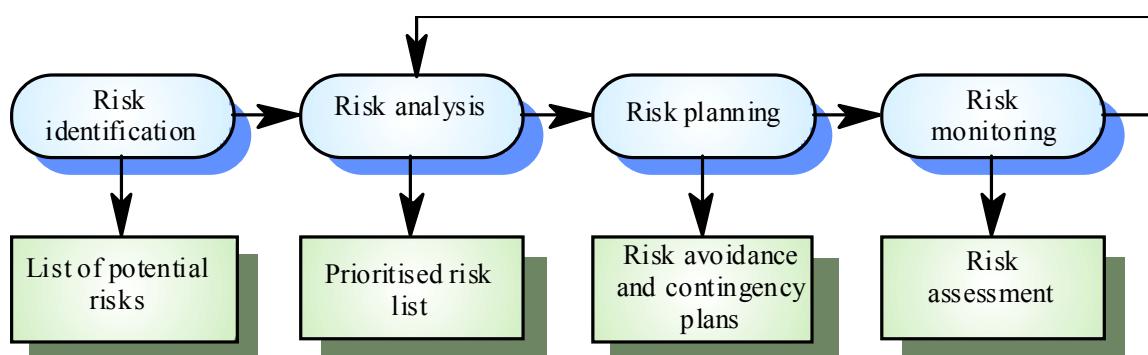
Incertidumbre en la estimación de costes: La parte clave es guardar un histórico de las estimaciones y llevar una contabilización de los costes reales de cada proyecto. En segundo lugar tendría en cuenta márgenes de error en las estimaciones. Luego propondría medidas con el fin de ir estudiando en cada proyecto los errores cometidos, respecto a las estimaciones, tratando de que estos fuesen menores.

Términos contractuales: Para minimizar los costes intentaría que la venta fuese a través de licencias, manteniendo nosotros la propiedad del software y de sus derechos comerciales.

Volatilidad de los requerimientos: Lo que haría es mantener una comunicación con el cliente a lo largo del proyecto. Le haría validar los requerimientos, utilizaría un prototipo para comprobar que hemos entendido bien el problema, etc.

Salud financiera: Para minimizar el impacto cobraría por adelantado parte del proyecto, crearía un fondo para autofinanciar futuros proyectos, estudiaría fórmulas de financiación, pediría subvenciones, etc.

- Crea un plan de riesgos para un proyecto ficticio (identifica sólo 3 riesgos). Indica los pasos seguidos y crea los documentos oportunos. Detalla como harías el seguimiento y control de dicho proyecto atendiendo a los riesgos identificados. (2,5p).



:

Identificar riesgos:

- a. Es imposible seleccionar personal con las habilidades requeridas para el proyecto.
- b. Baja en el personal.
- c. Los problemas financieros en la organización causan reducciones en el presupuesto del proyecto.

Analizar riesgos:

- a. Probabilidad alta. Efecto catastrófico.
- b. Probabilidad media. Efecto serio.
- c. Probabilidad baja. Efecto catastrófico.

Plan de prevención y minimización.

- d. Hacer una campaña de selección donde se difundan muchos los puestos ofertados dentro de personal especializado. Prever cursos de formación.
- e. Utilizar una estructura organizativa democrática descentralizada, donde todos hagamos todo tipo de tareas.
- f. Preparar un informe para justificar la importancia de nuestro proyecto.

Seguimiento:

- a. Revisar número de candidatos en otras convocatorias. Trabajar conjuntamente con las organizaciones educativas con el fin de conocer los perfiles.
- b. Llevar un control de ausencias. Ver el ánimo del grupo. Hacer actos sociales con mi personal para conocer su situación.
- c. Mantener contacto con directivos de la empresa, gestores de otros proyectos de nuestra empresa y mercado con el fin de ir conociendo el estado del sector.

3. ¿Qué relación guarda un modelo de proceso con la agenda de un proyecto y con la estimación de costes? Pon un ejemplo que justifique la respuesta. (2,5p)

El modelo de proceso condiciona qué actividades y en qué orden se incluirán en la agenda del proyecto, necesaria para poder conocer el tiempo y el coste total de desarrollo, y que son responsabilidad del gestor de un proyecto.

El modelo de proceso facilita la subdivisión de actividades en otras más "pequeñas", ya que proporciona el conjunto de actividades que hay que realizar. Estas actividades pueden subdividirse y estructurarse convenientemente, por ejemplo utilizando representaciones WBS que son muy útiles para estimar los costes, en donde, además, se favorece el uso de una estimación ascendente o descendente.

Las estadísticas de proyectos anteriores nos van a ayudar en las estimaciones, más aún cuando el proyecto se haya desarrollado con el mismo modelo de proceso. En el caso de UP incluso nos podría decir los % de las estimaciones que se usan en cada fase.

Si se sigue un modelo de proceso UP nos dará lugar a las 4 fases (inicio, elaboración, construcción y transición), así como a un conjunto de n iteraciones y a una serie de tareas de cualquier disciplina dentro de las iteraciones. Por otro lado las relaciones de precedencia de dichas actividades se corresponden con la secuencia temporal de las actividades que se llevan a cabo en el modelo UP.

4. Dados los siguientes datos de un proyecto. Explica cuál es la situación del proyecto para cada una de las tareas. ¿La tarea 1 se ha terminado? ¿Cómo podemos solucionar los problemas del proyecto? (2,5p)

Tarea	BCWS	BCWP	ACWP
Tarea R	2800	1800	2600
Tarea 1	600	600	1000
Tarea 2	1200	0	0
Tarea 3	1000	1200	800
Tarea 4	0	0	800

La situación del proyecto es que va retrasado y su productividad es baja (gastando de más).

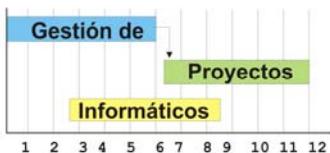
La tarea 1 va bien respecto a la agenda, pero en ella se ha gastado más de lo previsto. No podemos saber si se ha terminado, para ello nos haría falta conocer su BAC.

La tarea 2 debería de haber comenzado, pero todavía no ha hecho.

La tarea 3 ha comenzado, pero va retrasada, sin embargo su productividad es buena (se está gastando menos de lo previsto).

La tarea 4 no debería haber comenzado y no lo ha hecho. El valor ACWP es erróneo, sino ha empezado a realizarse no debería de tener imputados gastos.

Para resolver los problemas del proyecto podríamos asignar recursos de la tarea 3 a 2, ya que la 3 va adelantada y la 2 retrasada. También podríamos contratar a más gente para la 2 ya que en la 3 estamos gastando menos de lo esperado.



GESTIÓN DE PROYECTOS INFORMATICOS

Convocatoria de JUNIO. PRÁCTICAS

9 de junio de 2016

Apellidos, Nombre: _____

Grupo de prácticas: _____

1.- Dentro de un plan de proyecto planificado en Microsoft Project. ¿Cómo podemos definir los objetivos (metas)? (0,5p)

Los podemos definir con tareas e duración 0 (hitos). Otra alternativa es en el propio título de la fase o iteración. Y la última definiendo una nota.

¿Se podrían fijar los objetivos de las fases y las iteraciones en la planificación inicial? Pon los objetivos de las fases definidos a lo largo de tus prácticas de la asignatura (0,5p)

Fase de inicio. Estimaciones y plan inicial. Estudio de viabilidad del proyecto.

Fase de elaboración. Prototipo de la interfaz gráfica de la aplicación (formularios)

Fase de Construcción. Implementar la capa de lógica de negocio y acceso a datos.

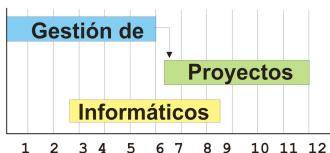
Fase de transición. Pruebas del sistema. Creación de soportes de distribución y manuales.

2.- Dentro de la LPOD, que son los códigos tipo? Nombra 3 códigos tipo. (0,5p)

Los códigos tipo son formularios NOTA prerellenados y simplificados para poder dar de alta un fichero de datos dentro de la agencia de protección de datos de una manera muy sencilla. Así mismo, permite notificar de forma simplificada, los ficheros de titularidad privada de comunidades de propietarios, clientes, libro recetario de las oficinas de farmacia, pacientes, gestión escolar, nóminas, recursos humanos y videovigilancia y los de titularidad pública de recursos humanos, gestión del padrón, gestión económica o control de acceso.

A la hora de registrar software, ¿Qué dos alternativas tenemos? ¿Cuál se suele utilizar? ¿Para esta alternativa a través de quién debemos de hacerlo y que impresos debemos de rellenar? (0,5p)

Tenemos la opción de hacerlo a través de patente de software y de registro de la propiedad intelectual. El más común (99%) es a través del registro de propiedad intelectual. Esto se hace a través del ministerio de Educación, cultura y Deporte. Hay que llenar los impresos de autores o impreso de otros titulares originarios, y el modelo de programas de ordenador.



GESTIÓN DE PROYECTOS INFORMÁTICOS

Convocatoria de JUNIO. TEORIA

10 de julio de 2017

Apellidos, Nombre: _____

Grupo de prácticas: _____

1. Explica las diferencias entre un presupuesto y un documento de estimación de costes. ¿Qué información debe de contener cada uno de ellos? (2 p)

Las estimaciones de coste son un apartado del plan del proyecto, y es una información interna de la empresa. Indica el esfuerzo necesario para realizar un proyecto (personas/mes), el coste temporal y económico del proyecto.

El presupuesto es un documento que se va a entregar al cliente para informarle sobre los precios de venta, tiempos de entrega y funcionalidades de la aplicación.

La diferencia fundamental es que la estimación habla de costes, mientras que el presupuesto habla en términos de precios de venta (coste+margin error+beneficio) y fechas de entrega (entrega estimada+margin de error).

El documento de estimación de costes debe de tener las estimaciones mediante distintas técnicas, por ejemplo: Parkinson, Pricing to Win, juicio experto, puntos de función y puntos objeto, así como una comparativa de los resultados obtenidos mediante las distintas técnicas.

El presupuesto debe de tener:

Los datos de proyecto, empresa y cliente (portada).

Una descripción del producto a desarrollar (requerimientos).

Los requisitos mínimos que necesita el programa (tanto SW como HW)

Las disposiciones legales.

Las retribuciones y plazos de pago.

Podría contener un anexo donde se describan productos o servicios adicionales (HW, mantenimiento, migraciones, etc).

Dependen del problema, de modelo de proceso y del personal que tengamos. El problema (requerimiento funcionales y no funcionales) sugerirá una arquitectura que será uno de los condicionamientos más fuertes para determinar cómo vamos a construir el sistema.

Un ejemplo de objetivos de la fase de construcción podrían ser los que hemos tenido en prácticas: dotar de funcionalidad a las pantallas creadas en la fase de elaboración.

Suponiendo una duración media de 3 semanas por iteración, y suponiendo que tenemos sólo 1 iteración para elaboración, podríamos planificar 1 o 2. Lo normal sería no tener ninguna ya que la fase de elaboración suele tener más de 1 iteración.

2. ¿Qué unidades de medida se emplean para dar los datos de las estimaciones de costes? ¿Qué ventajas presentan estas unidades respecto a otras? ¿Qué unidades se emplean en los presupuestos? (2 p)

Todas las técnicas de estimación estiman el esfuerzo necesario para completar un proyecto. El esfuerzo se mide en **Personas/mes**.

Tiene la ventaja de permitirnos **comparaciones** entre proyectos de forma **atemporal** y entre **distintos lugares** (empresas/proyectos).

En los presupuestos utilizamos unidades **monetarias** (precio venta) y de **tiempo** (fecha entrega).

3.- Para un proyecto que se planifica con el modelo en cascada. ¿Qué estructura organizativa recomendarías?

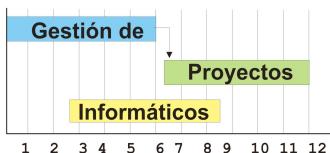
Dependiendo de las características del proyecto utilizare cualquiera de las 3 estructuras organizativas. Por ejemplo si es un proyecto con tamaño grande, dificultas baja y gran modularidad aplicaría el descentralizado controlado o el centralizado controlado.

		DD	DC	CC
DIFICULTAD	ALTA	X		
	PEQUEÑA		X	X
TAMAÑO	GRANDE		X	X
	PEQUEÑO	X		
DURACIÓN DEL EQUIPO	CORTO		X	X
	LARGO	X		
MODULARIDAD	ALTA		X	X
	BAJA	X		
FIABILIDAD	ALTA	X	X	
	BAJA			X
FECHA DE ENTREGA	EXTRICTA			X
	FLEXIBLE	X	X	
COMUNICACIÓN	ALTA	X		
	PEQUEÑA		X	X

4.- ¿Qué significa que tengamos un valor de BCWP menor que BCWS? ¿Qué pasos deberíamos de seguir en este caso, en qué orden y por qué? ¿Qué significa si tenemos un valor de ACWP mayor que BCWS? ¿Qué deberíamos de hacer en este caso? (2 p).

Si BCWP es menor que BCWS significa que el proyecto va retrasado. Lo primero que deberíamos de hacer es ver los indicadores de productividad. Si ACWP es menor que BCWP significa que no hemos utilizado los recursos previstos y habrá que aumentar la carga de trabajo a la tarea retrasada. Si ACWP es mayor o igual que BCWP en líneas generales podríamos restimar la duración de la tarea consumiendo su holgura libre, a continuación la holgura total, o si no es posible otra solución reasignar recursos o recortar incluso funcionalidades.

Comparar ACWP con BCWS no sirve para nada. Tendríamos que comparar con BCWP. En el caso de que ACWP sea menor que BCWP significa que nuestra productividad es buena (estamos gastando menos de lo presupuestado) y atendiendo al progreso, contrataremos a más gente si el este el lento, o no tendremos que aplicar acciones correctivas en caso contrario. En el caso de ser ACWP mayor que BCWP significa que estamos gastando más de lo presupuestado y tendremos que aplicar planes de contingencia relacionados con la mejora de la productividad.



GESTIÓN DE PROYECTOS INFORMÁTICOS

Convocatoria de JUNIO. PRÁCTICAS

10 de julio de 2017

Apellidos, Nombre: _____

Grupo de prácticas: _____

1.- ¿Qué son los formularios tipo? ¿Qué dos formas hay para registrar programas informáticos? (0,5p)

Los formularios tipos son formularios simplificados (que llevan los datos básicos rellenados) para dar de alta ficheros en la agencia de protección de datos.

Las dos formas son a través de patente y de derechos de autor. La primera de ellas se realizaría a través del ministerio de industria y es más complejo. La segunda es mucho más sencilla y se hace a través del ministerio de cultura y deporte.

2.- ¿Qué tipos de planes hemos hecho en prácticas? Justifica la respuesta. (0,75)

Hemos hecho un plan general al principio del proyecto y un plan detallado de las 2 primeras iteraciones. En el modelo UP al principio del proyecto no tenemos toda la información (es un modelo iterativo), por lo tanto sólo desarrollamos un plan general. Conforme avanza el proyecto vamos concretando el plan, detallando una serie más de iteraciones.

3.- ¿Qué diferencias hay entre un plan de 3 y 4 niveles? ¿Qué estructura de niveles hemos utilizado en prácticas? ¿cómo hemos solventado los problemas que presenta esa estructura? (0,75)

En un plan en 3 niveles tenemos los niveles de fase, iteración y tarea. En un plan de 4 tenemos fase, iteración, disciplina y actividad. En prácticas hemos utilizado el de 3 ya que presenta una mejor visibilidad. Para evitar sus desventajas hemos utilizado un esquema de nombrado (verbo+sustantivo) y una tormenta de ideas. La primera medida iba orientada a saber exactamente que íbamos a hacer y de qué disciplina se trataba. La segunda era para no olvidar tareas, o al menos no dejar una disciplina sin actividades.



GESTIÓN DE PROYECTOS INFORMÁTICOS

Convocatoria de JUNIO. TEORIA

30 de mayo de 2017

Apellidos, Nombre: _____

Grupo de prácticas: _____

1. Explica que es un presupuesto y explica cuales son sus 3 objetivos. Propón una medida para cumplir con cada uno de estos objetivos. (2,5p)

Es un documento contractual entre el cliente y proveedor que sirve para establecer los productos y o servicios, la fecha de entrega del servicio así como el precio de los mismos. Los objetivos del presupuesto son: Vender, fortalecer las relaciones entre cliente y proveedor y crear oportunidad de negocio.

Para vender haríamos en presupuesto bien diseñado, con una presentación clara, información detallada y unos precios ajustados al mercado. Para fortalecer las relaciones haremos que los productos y o servicios estén bien definidos con el fin de evitar discrepancias futuras. Y para crear oportunidades de negocio ofreceremos servicios adicionales tales como hardware, migración datos, mantenimiento, formación.

2. Explica la diferencia entre WBS, plan general y plan detallado de un proyecto. ¿Qué diferencia existe entre el plan general y detallado en los modelos UP y cascada. (2,5p)

El WBS es un esquema de tareas estructuradas. El plan general fija las fases e iteraciones (siempre que el modelo tenga), así como sus objetivos. El plan detallado concreta las tareas, establece el orden y precedencias de las tareas y asigna duraciones y recursos a las tareas y establece las fechas.

La primera diferencia es que el modelo en cascada no tiene iteraciones. La segunda diferencia es que en el modelo en cascada el plan detallado se establece al principio del proyecto, mientras que en el modelo up se va detallando poco a poco. Otra diferencia son las fases, en cascada son análisis, diseño, implementación, integración y prueba, mientras que en el modelo UP son inicio, elaboración, construcción y transición.

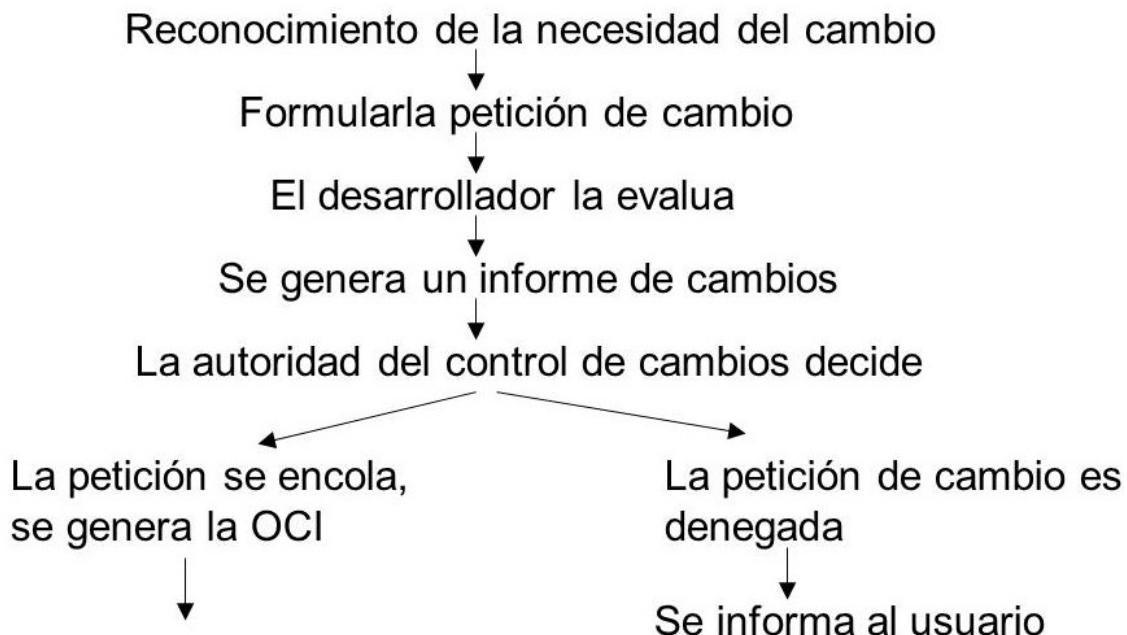
3. Describe 5 multiplicadores que nos afectan a las estimaciones de coste de un proyecto. (2,5p)

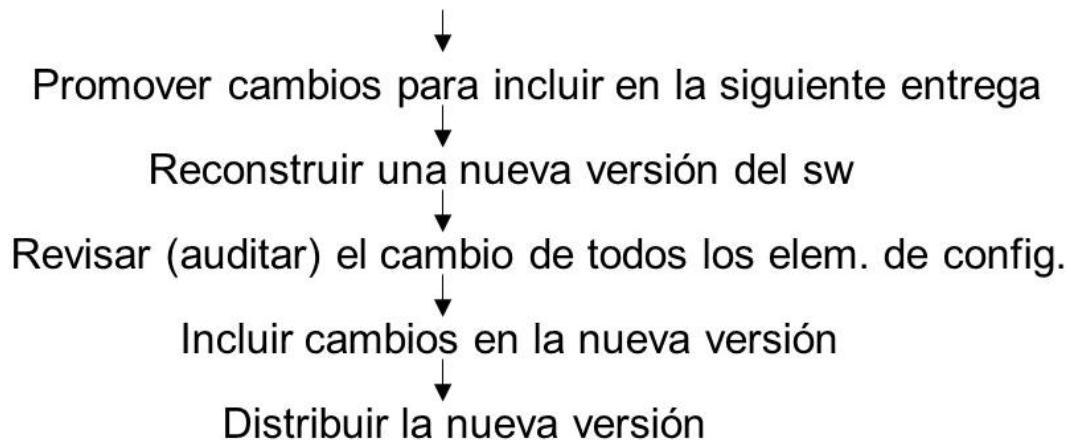
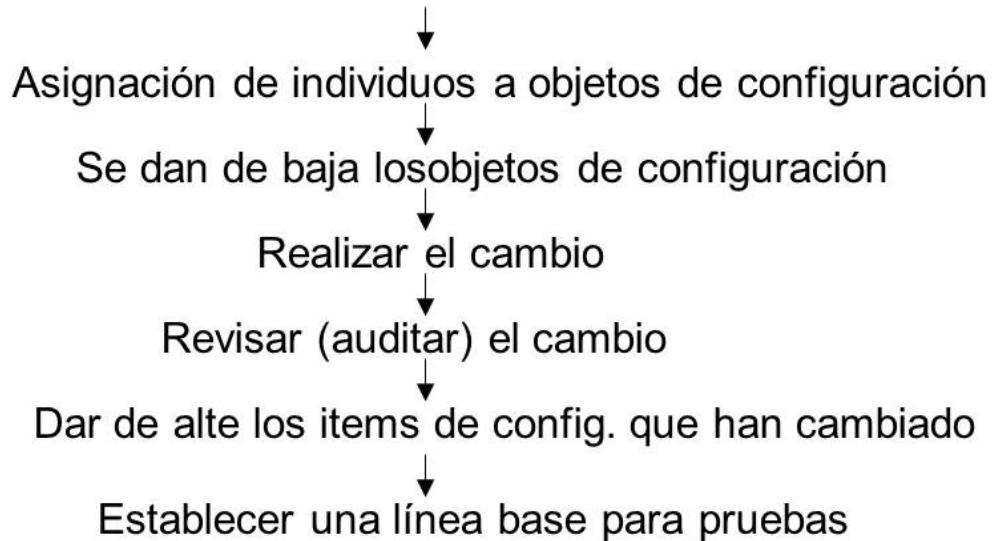
- RCPX - fiabilidad de producto y complejidad
- RUSE - reutilización requerida
- PDIF - dificultad de la plataforma
- PREX - experiencia del personal
- PERS - capacidad del personal
- SCED - agenda requerida
- FCIL - facilidades de soporte de grupo

Cuando un producto requiere **fiabilidad** su estimación de costes crecerá. Sin embargo si hablamos de **reutilización** tendremos un coste de integración de los componentes, pero siempre menor a la de su implementación. Si vemos que vamos a desarrollar con el fin de lo que desarrollemos lo aprovechamos (reutilicemos) en otros proyectos el coste sin embargo aumentará. Dependiendo de la **plataforma** los costes pueden variar, hay plataformas más sencillas, mejor preparadas para programar en ellas y con mayor tiempo en el mercado. Cuando hablamos del personal tenemos los términos de experiencia de capacidad. Lo interesante es que el personal tenga **experiencia**, disminuyendo así los costes del proyecto. El término **capacidad** va en la línea de no tener experiencia pero que los empleados sean capaces de asumir las nuevas tareas. Esto es muy típico en los ingenieros recién titulados, no tienen experiencia, pero si muchísima capacidad. Si hablamos de agenda poco flexible y apretada en el tiempo va provocar costes mayores.

4. Explica la diferencia entre gestión de configuraciones y mantenimiento. Escribe el algoritmo de proceso de mantenimiento ¿En qué fase del proceso UP se realiza el mantenimiento?

La gestión de configuración es una actividad de autoprotección que se lleva a cabo a lo largo de todo el ciclo de vida del software. El mantenimiento es un subconjunto de la gestión de configuraciones que tiene lugar desde el momento que el software se pone en producción.





El modelo de proceso UP sólo abarca el desarrollo del software hasta su puesta en producción, no modela el mantenimiento.



GESTIÓN DE PROYECTOS INFORMÁTICOS

Convocatoria de JUNIO. PRÁCTICAS

30 de mayo de 2017

Apellidos, Nombre: _____

Grupo de prácticas: _____

- 1.- Elige un perfil profesional dentro de la informática y dibuja el diagrama de proceso de una competencia de dicho perfil. (0,75p)

Un perfil podría ser el técnico microinformático

La competencia reinstalar un ordenador de un cliente. En lugar del diagrama lo vamos a mostrar en forma de tabla, con entradas y salidas. En el diagrama de proceso representaríamos las acciones con círculos y los artefactos con rectángulos.

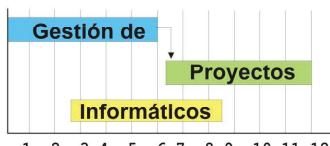
Entrada	Acción	Salida
	Recogida de datos del cliente	Listado de servicios, programas y configuraciones
Especificaciones fabricante	Comprobar requisitos mínimos	
Licencias aportadas por cliente	Comprobar licencias	
	Hacer imagen de la instalación actual	Imagen disco
	Verificar imagen	
	Exportar datos	Datos cliente
	Particionar disco	
	Instalar operativo	
Listado de servicios necesarios	Configurar servicios	
Listado de programas y configuraciones	Instalar y configurar programas	
	Verificar instalación	

2.- Explica la vinculación de la estimación realizada con Puntos Objeto con Pricing to Win.
(0,5p)

La técnica de Puntos Objeto estima el coste del proyecto, mientras que Pricing to Win estima el precio de venta. Al comparar ambos resultados nos dará el número de licencias de software que tenemos que vender para cubrir los costes. Esto implicará hacer un estudio de mercado para ver las ventas esperadas y el tiempo necesario para recuperar la inversión.

3.- ¿Qué dos formas hemos empleado en prácticas para realizar el seguimiento de un proyecto? ¿Qué ventajas presenta cada una de ellas? (0,75p)

Diagrama Gantt de Seguimiento y Modelo Eva. El Gantt de seguimiento permite realizar el seguimiento de una manera visual y rápida. El modelo Eva nos permite hacer un seguimiento objetivo y cuantitativo (con datos numéricos).



Ejercicios Repaso

Cuestiones rápidas

1. Un plan de proyecto ...
 - a. Proporciona visibilidad al progreso del proyecto
 - b. Es una representación abstracta de un proceso que proporciona información parcial sobre el proceso software
 - c. Es un conjunto de actividades que tienen como objetivo el desarrollo de un producto software
 - d. Todas las afirmaciones anteriores son falsas

2. El modelo basado en prototipos ...
 - a. Reduce las horas de implementación.
 - b. Permite reducir riesgos y costes de desarrollo
 - c. El sistema resultante es un prototipo
 - d. Es posible utilizarlo sólo con lenguajes orientados a objetos

3. La gestión de proyectos ...
 - a. Tiene como objetivo determinar qué hay que hacer, cómo hay que hacerlo y quién tiene que hacerlo.
 - b. Está fuertemente condicionada por la implementación del sistema
 - c. Está fuertemente condicionada el diseño del sistema
 - d. Tiene como objetivo el confeccionar una agenda del proyecto

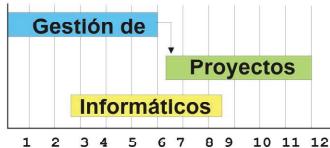
4. Tenemos tres actividades: A, B y C. A precede a B. Las holguras totales son: HT(A)= 4, HT(B)=2, HT(C)=3. Las holguras libres son: HL(A)= 3, HL(B)=2, HL(C)=3. Si A se retrasa 4 días, ¿qué les ocurre a B y C?
 - a. Nada, ya que no sobrepasamos la holgura total de A
 - b. La holgura total de B y C pasa a ser de 1 y 2 días respectivamente
 - c. B se convierte en crítica
 - d. Ninguna de las anteriores es cierta

5. Ms Project es ...
 - a. Una herramienta de seguimiento de proyectos
 - b. Una herramienta de modelado de proyectos
 - c. Una herramienta de estimación de costes de proyectos
 - d. Una herramienta de desarrollo de proyectos

6. El cono de la incertidumbre ...
 - a. Dice que conforme avanza el proyecto menos errores de estimación tenemos.
 - b. Dice que conforme avanza el proyecto más errores de estimación descubrimos.
 - c. No puedo aplicarlo sino hago el seguimiento del proyecto.
 - d. Todas las anteriores son falsas.

7. ¿Cuál de los siguientes elementos NO es un elemento de configuración (EC)?
 - a. Los ejecutables de la aplicación
 - b. La estructura de directorios.
 - c. La definición de las tablas de la BD de la aplicación
 - d. Todos los anteriores son elementos de configuración.

8. Make es una herramienta case que soporta ...
 - a. la planificación de una gestión de configuraciones
 - b. el control de cambios
 - c. la construcción del sistema
 - d. la gestión de versiones y releases



Ejercicios Repaso

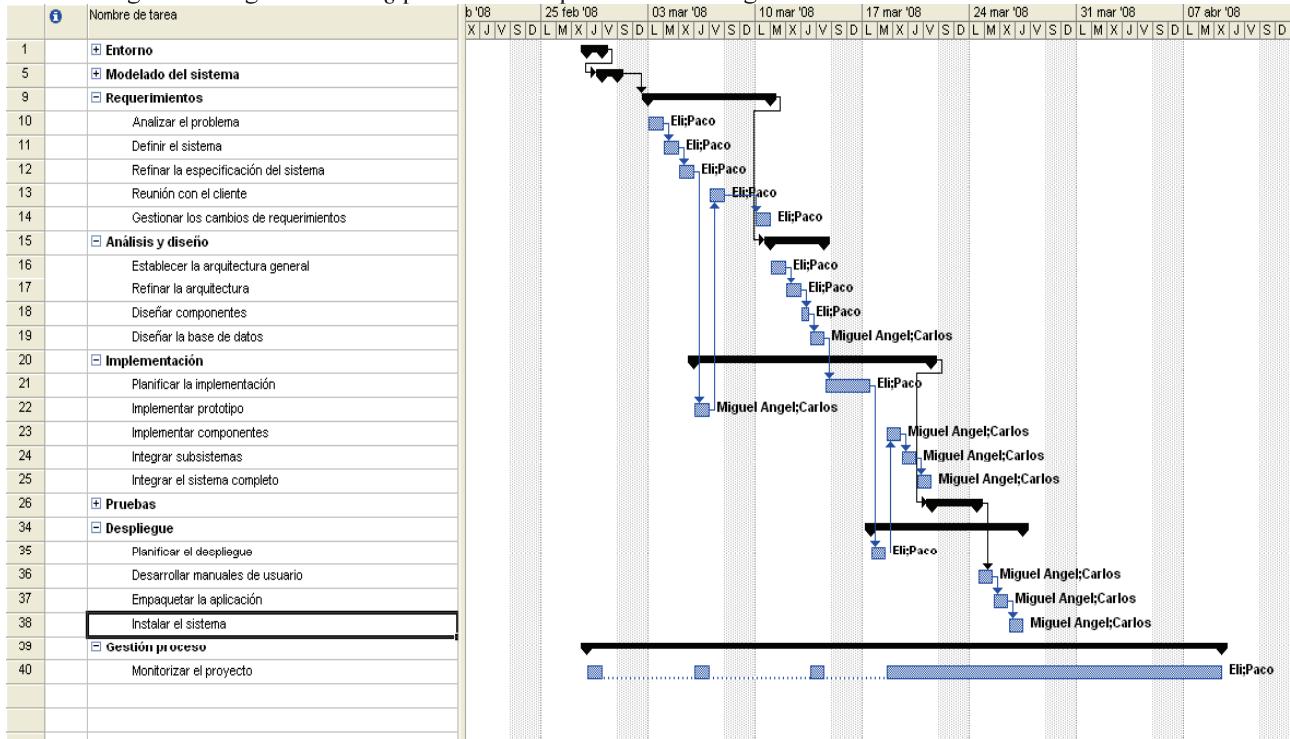
9. Las versiones de un sistema.

- a. Tienen un coste menor que las releases
- b. Suelen superar en número a las releases
- c. Incluyen documentación técnica
- d. Todas son ciertas.**

10. En el modelo UP.

- a. Hay una versión por iteración.
- b. Puede haber más de una versión por iteración.**
- c. Tenemos una release por fase.
- d. Todas las anteriores son ciertas.

11. En el siguiente diagrama Gantt ¿qué modelo de proceso se está siguiendo?



- a. Desarrollo Rápido de aplicaciones.
- b. Cascada
- c. Prototipado.
- d. Todos los anteriores.**

12. En la pirámide de jerarquía de necesidades donde se encuentran la mayoría de jefes de proyecto.

- a. No les afecta esta jerarquía, es sólo para empleados.
- b. Necesidades sociales.
- c. Necesidades seguridad.
- d. Necesidades autorrealización.**

13. ¿Cuál es la distribución de tiempo dentro de un equipo de desarrollo.

- a. 20% Trabajando sólo, 30% actividades no productivas, 50% Interactuando con otra gente.
- b. 30% Trabajando sólo, 20% actividades no productivas, 50% Interactuando con otra gente.**
- c. 30% Trabajando sólo, 20% actividades de gestión, 50% Interactuando con otra gente.
- d. 20% Implementando, 30% Interactuando con otra gente y 50% en actividades de gestión.

1. En el registro de la propiedad intelectual de un software bajo una relación laboral, ¿para qué sirve el documento de declaración de autor asalariado de un empleado?
 - a. Para la cesión de los derechos de explotación del software del empleado a la empresa.
2. En el análisis EVA, la tarea/proyecto va con retraso según la agenda si:
 - a. BCWP < BCWS
3. Según la LOPDGDD ¿A partir de qué edad puede un menor expresar el consentimiento en términos de protección de datos?
 - a. 14
4. Según la Ley de Propiedad Intelectual, ¿Cuál de las siguientes figuras tiene un plazo de protección indefinido?
 - a. Secretos Comerciales
5. En el modelo UP
 - a. Las iteraciones pueden tener duraciones distintas
 - b. Puede haber más de una versión por iteración
6. ¿Qué tarea no podemos realizar en Microsoft Project?
 - a. La planificación detallada de todo el proyecto
7. La cadena de actividades que determina la duración global mínima de un proyecto se conoce como
 - a. Camino Crítico
8. En el análisis EVA, el indicador CPI se calcula como:
 - a. BCWP / ACWP
9. ¿Cuál de los siguientes parámetros ECA nos informa de la cantidad de trabajo planificado al final del proyecto/tarea?
 - a. BAC
10. En el análisis EVA, el indicador SV se calcula como:
 - a. BCWP - BCWS
11. ¿Qué aspecto ha sido el más importante a la hora de establecer los objetivos de las fases de elaboración de construcción en el modelo UP seguido para la planificación del software de gestión de autónomos?
 - a. Los requerimientos funcionales y no funcionales.
12. En el análisis EVA, la tarea/proyecto está por debajo de lo presupuestado si:
 - a. BCWP > ACWS
13. En el análisis EVA, la tarea/proyecto está por encima de lo presupuestado si:
 - a. BCWP < ACWS
14. Un plan de proyecto
 - a. Proporciona visibilidad al progreso del proyecto
15. ¿Cuál es la distribución de tiempo dentro de un desarrollo de proyecto?
 - a. 30% Trabajando sólo, 20% actividades no productivas, 50% Interactuando con otra gente.
16. ¿Qué técnica de estimación de costes se basa en contabilizar pantallas, informes y módulos de cálculo?
 - a. Puntos objeto
17. Una de las recomendaciones que se deben tener en cuenta durante la estimación de recursos y tiempo de un proyecto informático es
 - a. Identificar las tareas que consumen más recursos.
- 18.

19. ¿Cuál de las siguientes no es una causa legal para poder transmitir los derechos de Propiedad Intelectual?
- En virtud de relación mercantil.
20. En una iteración del Plan detallado se establece una actividad de revisión de los riesgos del proyecto. Dicha actividad consiste en:
- Revisar los identificadores potenciales que figuran en el plan
21. La LOPDGDD no será de aplicación en caso de
- Materias consideradas clasificadas
 - Datos de personas fallecidas
 - Datos ya excluidos en el RGPD
22. En caso de producirse un retraso en una actividad, ¿cuál de las siguientes medidas sería la que menos afecta a la agenda del proyecto?
- Retrasarla según su holgura libre.
23. El modelo basado en prototipos
- Permite reducir riesgos y costes de desarrollo
24. ¿Cuál de las siguientes no es una técnica de estimación de costes de un proyecto?
- Gantt
25. Monitorizar una agenda tiene como objetivo fundamental
- consiste en comprobar si la agenda real se ajusta a la planificada
26. ¿Cuál de los siguientes modelos de proceso es evolutivo?
- Modelo en espiral, Desarrollo Unificado
27. Las releases de un sistema
- Incluye documentación técnica
 - Coste mayor que las versiones.
 - Son menos en número que las versiones
28. Cuando en el desarrollo del plan del proyecto existen restricciones de tiempo y presupuesto establecidas por el cliente, estas deben
- Deben ser consideradas evaluando su impacto en el proyecto.
29. ¿Hasta qué momento durante la ejecución de un proyecto, la contingencia debe ser considerada?
- Hasta que el tiempo y los avances del proyecto demuestran que el riesgo se ha reducido lo suficiente o desaparecido
30. ¿Qué importancia tiene el RGPD?
- Establece un marco europeo relativo a la protección de las personas físicas en lo que respecta al tratamiento de sus datos personales y a la libre circulación de estos
31. ¿Qué podemos decir de una tarea si conocemos los valores $BCWS=BCWP=600$ y $ACWP=1000$?
- no podemos saber si ha terminado sin conocer su BAC
32. Muchas organizaciones fortalecen sus equipos de gestión de negocios, para que estos sean identificados a priori, pero lo que ocurre realmente es
- a.
33. ¿Qué podemos decir de una tarea si conocemos los valores $BCWS=1200$ y $BCWP=ACWP$?
- a.
34. If BCWS 1000, BCWP 1200, and ACWP 1300, the project is:
- Ahead of schedule and over budget

35. Dado que un riesgo siempre va a existir hay que
- Establecer mecanismos de seguimiento y prevención.
36. El Plan detallado de un proyecto
- Todas son falsas
37. Respecto a la protección de datos del software de gestión de autónomos, indica cuál de las siguientes sería una acción obligatoria para cumplir con la legislación actual en el caso de desarrollarla como aplicación web
- Incluir un apartado de aviso legal en la web.
38. En la pirámide de jerarquía de necesidades donde se encuentran la mayoría de jefes de proyecto
- Necesidades autorrealización.
39. Las características del software libre son
- Se distribuye libremente y es de código abierto
40. ¿Qué podemos decir de una tarea si conocemos los valores $BCWS=BCWP=0$ y $ACWP=800$?
- El valor ACWP es erróneo.
41. ¿Cuál de los siguientes tipos de licencia no es propia del software privativo
- Apache
42. En el análisis EVA, la tarea/proyecto va adelantada según la agenda si
- $BCWP > BCWS$
43. En el caso de que hayamos desarrollado un software de gestión de autónomos en forma de aplicación de escritorio, ¿quién debería encargarse del cumplimiento de la legislación a nivel de protección de datos?
- Nosotros como desarrolladores de la aplicación.
44. Tenemos un proyecto con los siguientes datos del proyecto EVA, cual de las siguientes afirmaciones es falsa.

Tarea	BCWS	BCWP	ACWP	BAC
Tarea0	2800	1800	1800	2600
Tarea1	600	600	600	1200
Tarea2	1200	0	600	2600
Tarea3	1000	1200	800	2000
Tarea4	0	0	0	600

La tarea 0 va retrasada y consumiendo más de lo previsto

Derechos LOPDGDD (A.R.C.O.)

Acceso

Permite que una persona pueda solicitar información al responsable de un fichero sobre si sus datos personales están siendo tratados.

Rectificación

Permite a la persona afectada solicitar la modificación de datos que sean inexactos o incompletos.

Cancelación

El afectado puede solicitar la supresión de los datos que resulten inadecuados o excesivos, sin perjuicio del deber de bloqueo.

Oposición

Se trata del derecho de una persona a oponerse al tratamiento de sus datos personales.

Derechos RGPD (P.O.L.)

Portabilidad

Permite la posibilidad de solicitar al responsable del tratamiento que se le faciliten los datos personales en un formato estructurado y elaros a otro responsable.

Olvido o supresión

Facilita al interesado la posibilidad de requerir que sus datos personales "desaparezcan" o "no dejen rastro en la red".

Limitación del tratamiento

Permiten que cualquier persona tenga derecho a exigir al responsable del tratamiento la limitación del tratamiento de sus datos personales.

1. ¿Cuál de los siguientes modelos de proceso es evolutivo?

- a. Cascada.
- b. Unificado.
- c. Espiral.
- d. Las respuestas b y c son correctas.

2. ¿Qué tarea no podemos realizar en Microsoft Project?

- a. La planificación detallada de todo el proyecto.
- b. El seguimiento del proyecto.
- c. La estimación del proyecto.
- d. Todas las tareas anteriores se pueden realizar en Microsoft Project.

3. En el modelo UP...

- a. El número de iteraciones es igual en todas las fases.
- b. La fase de inicio siempre tiene 1 iteración.
- c. Las fases de elaboración y construcción tienen el mismo número de iteraciones.
- d. Todas son falsas.

4. ¿Cuál de las siguientes afirmaciones es falsa?

- a. El conocimiento sintáctico se adquiere mediante experiencia y aprendizaje activo.
- b. El conocimiento semántico se adquiere mediante memorización.
- c. El nuevo conocimiento semántico puede interferir con conocimiento semántico ya existente.
- d. Todas son falsas.

5. Las releases de un sistema.

- a. Tienen un coste menor que las versiones.
- b. Superan en número a las versiones.
- c. Incluyen documentación técnica.
- d. Todas son ciertas.

6. En el proceso unificado de desarrollo de software:

- a. Los requisitos se establecen en la primera fase y no se permite ningún cambio en las fases siguientes.
- b. La tarea de establecer los requisitos es más intensa en la primera fase aunque en las fases siguientes los requisitos se pueden modificar.
- c. El establecimiento de requisitos requiere del mismo esfuerzo en todas las fases.
- d. Los requisitos siempre se establecen a partir de prototipos.

7. Las metodologías ágiles se caracterizan

- a. Simplificar la complejidad de otras metodologías.
- b. Por inventarse a finales del S XIX en la revolución industrial.
- c. Por hacer más énfasis en la documentación a entregar.
- d. Ponen en el centro el proyecto a entregar no al cliente.

8. ¿Qué beneficios tiene las metodologías ágiles?

- a. Rapidez, documentación muy completa y trabajar de forma intensa la última semana antes de la entrega.
- b. Eficacia, rapidez y visión cliente.
- c. Cliente, proveedores y cumplir el presupuesto.
- d. Individualismo.

9. Indica cuál de las siguientes metodologías son ágiles.

- a. Cobol, c++, Java.
- b. Uml , PPo, itil.
- c. Kanban, Scrum, Extrem Programming.
- d. Ninguna de las anteriores

10. Indica cuál de las siguientes características aplican al modelo Canvas de negocio.

- a. Es una herramienta nueva que surge después de la segunda guerra mundial.
- b. Se basa en existencia de una propuesta de valor que tenemos que comunicar al cliente mediante un modelo canvas y rellenando todos los apartados.
- c. Se utiliza solo para proyectos de robótica.
- d. Se puede dejar apartados sin llenar, al menos con llenar el 50 por ciento es válido.

11. Señala cuál de las siguientes características corresponden a la metodología SCRUM.

- a. Es incremental, reuniones diarias y se basa en roles.
- b. Reuniones diarias, y trabajo individual.
- c. Lo importante es el trabajo individual y la documentación que se realice sobre la aplicación.
- d. Se basa en roles y pasa a segundo plano la calidad del proyecto.

12. Las características del software libre son:

- a. Se distribuyen libremente y es de código abierto.
- b. Están disponibles en una página web que se llama www.softlibre.com.
- c. Sólo se pueden usar en la enseñanza e investigación.
- d. Tiene limitaciones de usuario.

13. ¿Qué importancia tiene el RGPD?

- a. Establece un marco para el registro de patentes de empresas.
- b. Establece un marco para regular a las empresas sobre las llamadas que un ciudadano puede recibir para venderle productos.
- c. Establece un marco europeo relativo a la protección de las personas físicas en lo que respecta al tratamiento de sus datos personales y a la libre circulación de estos datos.
- d. Es una recomendación europea que está en estudio para su redacción definitiva.

14. Monitorizar una agenda tiene como objetivo fundamental

- a. Presentar al cliente las desviaciones producidas en el proyecto para que sea consciente de que se va a retrasar la entrega.
- b. Consiste en comprobar si la agenda real se ajusta a la planificada
- c. Se usa como seguimiento interno.
- d. Sirve para justificar las horas dedicadas a un proyecto.

15. ¿Cuál de estas características no es de la metodología SCRUM?

- a. Metodología iterativa e incremental.
- b. Reuniones diarias.
- c. Descanso cada 15 días para renovar las ideas.
- d. Demo al cliente al final de cada iteración.

16. ¿Cuál de los siguientes modelos de proceso es evolutivo?

- a. Cascada.
- b. Desarrollo rápido de aplicaciones.

c. Espiral.

d. Las respuestas b y c son correctas.

17. ¿Cuál de los siguientes libros es el principal recomendado en la asignatura?

- a. Ingeniería del software" 7^a Edición. Ian Sommerville.
- b. "Técnicas de Programación y Control de Proyectos". C.Romero López. Pirámide
- c. Ingeniería del Software: un enfoque práctico. 5^o edición.Roger S. Pressman.
- d. "Desarrollo y Gestión de Proyectos Informáticos". Steve McConnell.

18. ¿Qué multiplicador con valor alto reduciría el esfuerzo estimado en COCOMO?

- a. Fiabilidad de producto y complejidad.
- b. Reutilización requerida.
- c. Dificultad de la plataforma.
- d. Experiencia del personal.

19. ¿Cuál de las siguientes afirmaciones es cierta?

- a. El conocimiento sintáctico se adquiere mediante experiencia y aprendizaje activo.
- b. El conocimiento semántico se adquiere mediante memorización.
- c. El nuevo conocimiento semántico puede interferir con conocimiento semántico ya existente.
- d. Ninguna de las anteriores.

20. Un plan de proyecto ...

- a. Proporciona visibilidad al progreso del proyecto.
- b. Es una representación abstracta de un proceso que proporciona información parcial sobre el proceso software.
- c. Es un conjunto de actividades que tienen como objetivo el desarrollo de un producto software.
- d. Todas las afirmaciones anteriores son falsas.

21. La gestión de proyectos ...

- a. Tiene como objetivo determinar qué hay que hacer, cómo hay que hacerlo y quién tiene que hacerlo.
- b. Está fuertemente condicionada por la implementación del sistema.
- c. Está fuertemente condicionada el diseño del sistema.
- d. Tiene como objetivo el confeccionar una agenda del proyecto

22. El cono de la incertidumbre ...

- a. Dice que conforme avanza el proyecto menos errores de estimación tenemos.
- b. Dice que conforme avanza el proyecto más errores de estimación descubrimos.
- c. No puedo aplicarlo sino hago el seguimiento del proyecto.
- d. Todas las anteriores son falsas.

23. ¿Cuál es la distribución de tiempo dentro de un equipo de desarrollo.

- a. 20% Trabajando sólo, 30% actividades no productivas, 50% Interactuando con otra gente.
- b. 30% Trabajando sólo, 20% actividades no productivas, 50% Interactuando con otra gente.
- c. 30% Trabajando sólo, 20% actividades de gestión, 50% Interactuando con otra gente.
- d. 20% Implementando, 30% Interactuando con otra gente y 50% en actividades de gestión.

24. Las versiones de un sistema.

- a. Tienen un coste menor que las releases.
- b. Suelen superar en número a las releases.
- c. Incluyen documentación técnica.
- d. Todas son ciertas.

25. En el proceso unificado de desarrollo de software:

- a. Los requisitos se establecen en la primera fase y no se permite ningún cambio en las fases siguientes.
- b. La tarea de establecer los requisitos es más intensa en la primera fase aunque en las fases siguientes los requisitos se pueden modificar.
- c. El establecimiento de requisitos requiere del mismo esfuerzo en todas las fases.
- d. Los requisitos siempre se establecen a partir de prototipos.

26. Las metodologías de desarrollo de software tienen como objetivo:

- a. Establecer los requisitos de un sistema.
- b. Desarrollar un sistema dentro de un tiempo apropiado y con unos costes aceptables.
- c. Proporcionar sistemas que satisfagan a todas las personas afectadas por el mismo.
- d. Todas son ciertas.

27. En eXtreme programming:

- a. No es necesario que el cliente esté in situ, ya que, se pueden utilizar especificaciones escritas para la siguiente iteración.
- b. No es necesario que se escriban las pruebas antes de comenzar a programar.
- c. Cada iteración debe acabar con una versión lista para su producción.
- d. Ninguna de las anteriores.

28. En la fase de exploración de eXtreme programming:

- a. El cliente establece la prioridad de cada historia de usuario.
- b. El equipo de desarrollo establece la velocidad del proyecto.
- c. Se prueba la tecnología y se construye un prototipo.
- d. Se realizan tareas de soporte para el cliente.

29. En Scrum, la gráfica “Burn-Down” indica:

- a. La velocidad de desarrollo del equipo.
- b. La cantidad de trabajo pendiente en un sprint.
- c. Los posibles impedimentos del equipo.
- d. Todas son ciertas

30. En Scrum, un Sprint:

- a. Tiene una duración prefijada.
- b. Puede alargarse el tiempo necesario hasta acabar todas las tareas pendientes.
- c. Tiene la duración que el cliente indique.
- d. Ninguna de las anteriores.

31. Un impedimento en SCRUM:

- a. Se refleja como una línea horizontal en el gráfico burn-down.
- b. Se refleja como una línea vertical en el gráfico burn-down.
- c. Se refleja como una línea discontinua en el gráfico burn-d.own.
- d. Ninguna de las anteriores.

32. Los modelos de SCRUM y XP

- a. Tienen unos roles predefinidos.
- b. Utilizan historias de usuario para definir requisitos.
- c. Ofrecen resultados al cliente al finalizar cada iteración.
- d. Todas son ciertas.

33. ¿Qué es la refactorización en XP?

- a. Las acciones que realizan los programadores para pasar las pruebas unitarias
- b. Las acciones que realizan los testers para definir los casos de prueba
- c. La simplificación del código sin modificar su comportamiento
- d. Ninguna de las anteriores

Indica 5 factores que afecten a la estimación de costes de un proyecto. Propón una medida para ayudar a minimizar el impacto de cada uno de estos factores.

- Oportunidades de mercado: Realizar un proyecto empresarial, con un estudio claro de los costes y de la inversión, así como de la recuperación de los mismos. Crear proyectos diferenciados y mantener los secretos de nuestro software con el fin de evitar que la competencia entre en nuestro mercado.
- Incertidumbre en las estimaciones: Guardar datos de proyectos y trabajar para que en el futuro nuestras desviaciones del presupuesto sean menores. Estimaríamos con varias técnicas y compararíamos los resultados.
- Términos contractuales: Fijar los términos de forma clara mediante un contrato escrito.
- Volatilidad de los requerimientos: Firmar un documento donde el cliente revise y acepte las especificaciones del sistema a construir.
- Salud financiera: Cobrar un anticipo en la firma del contrato. Guardar dinero de proyectos anteriores con el fin de tener autofinanciación total o parcial.

Explica qué es la pirámide de necesidades. ¿Por qué es importante conocerla como gestores de proyectos? Pon un par de ejemplos de su uso.

- La pirámide de necesidades es una simplificación de la realidad, donde nos indica distintas necesidades que tenemos las personas.
- En la base de la pirámide están las necesidades principales que debemos de cubrir:
 - o necesidades fisiológicas (comer, dormir, etc.)
 - o necesidades de seguridad,
 - o necesidades sociales (ser aceptado como parte de un grupo)
 - o necesidades de estima (del reconocimiento de nuestro trabajo)
 - o necesidades de autorealización.

- Es importante como gestores de proyectos por que dentro de nuestras tareas están las relacionadas con el personal, en concreto las tareas de motivación y retención del personal serían las más vinculadas con la pirámide.
- Como ejemplo de uso:
 - o podríamos poner seleccionar a personal cuyas necesidades podamos cubrir a corto plazo y formar equipos donde sus necesidades sean satisfechas (si todos quieren lo mismo podría ser imposible satisfacer a todos).
 - o el de retención, haciendo que el personal vaya cubriendo sus expectativas (necesidades) a lo largo de su vida laboral.

Describe los roles y funciones de cada uno de ellos de la Metodología SCRUM.

Scrum diferencia claramente la responsabilidad y funciones de cada uno de los Roles:

- Dueño del producto (Product Owner):
 - o Define las funcionalidades del producto
 - o Decide sobre las fechas y contenidos de los releases
 - o Es responsable por la rentabilidad del producto (ROI)
 - o Prioriza funcionalidades de acuerdo al valor del mercado/negocio
 - o Ajusta funcionalidades y prioridades en cada iteración si es necesario
 - o Acepta o rechaza los resultados del trabajo del equipo
- Scrum Master:
 - o Gestiona y supervisa el proyecto
 - o Responsable de promover los valores y prácticas de Scrum
 - o Elimina impedimentos Se asegura de que el equipo es completamente funcional y productivo
 - o Permite la estrecha cooperación en todos los roles y funciones
 - o Escudo del equipo ante interferencias externas
- Equipo de desarrollo o Scrum Team
 - o Típicamente de 5 a 9 personas
 - o Multi-funcional:
 - Programadores, testers, analistas, diseñadores, etc
 - o Los miembros deben ser full-time Los equipos se auto-organizan
 - o Sólo puede haber cambio de miembros entre los Sprints

¿Qué relación guarda un modelo de proceso con la agenda de un proyecto y con la estimación de costes? Indica como el modelo de proceso utilizado en prácticas ha marcado el plan.

- El modelo de proceso condiciona qué actividades y en qué orden se incluirán en la agenda del proyecto, necesaria para poder conocer el tiempo y el coste total de desarrollo, y que son responsabilidad del gestor de un proyecto.
- El modelo de proceso facilita la subdivisión de actividades en otras más "pequeñas", ya que proporciona el conjunto de actividades que hay que realizar. Estas actividades

pueden subdividirse y estructurarse convenientemente, por ejemplo utilizando representaciones WBS que son muy útiles para estimar los costes, en donde, además, se favorece el uso de una estimación ascendente o descendente. Las estadísticas de proyectos anteriores nos van a ayudar en las estimaciones, más aún cuando el proyecto se haya desarrollado con el mismo modelo de proceso. En el caso de UP incluso nos podría decir los % de las estimaciones que se usan en cada fase.

- Si se sigue un modelo de proceso UP nos dará lugar a las 4 fases (inicio, elaboración, construcción y transición), así como a un conjunto de n iteraciones y a una serie de tareas de cualquier disciplina dentro de las iteraciones. Por otro lado las relaciones de precedencia de dichas actividades se corresponden con la secuencia temporal de las actividades que se llevan a cabo en el modelo UP. Al utilizar prototipado ha condicionado los objetivos de las fases, teniendo como objetivo al final de la fase de elaboración la interfaz gráfica de la aplicación.

Indica las tareas que hemos realizado dentro de Microsoft Project. Indica los pasos que debemos de seguir para hacer el seguimiento del proyecto.

- WBS (esquema actividades), mediante la creación de una plantilla
- Plan temporal general, estableciendo iteraciones y objetivos (tanto de fases como de iteraciones).
- Plan temporal detallado de un par de iteraciones (duraciones y precedencias de las tareas y asignación de recursos)
- Seguimiento mediante Gantt de seguimiento y valores EVA.

Para realizar el seguimiento debemos de establecer una fecha inicial de proyecto. A continuación crearemos el plan real estableciendo una línea base. Luego debemos de ir introduciendo los porcentajes de completitud de cada tarea. Para visualizar el estado podemos utilizar el Gantt de seguimiento o bien establecer la fecha de estado y analizar los valores EVA.

Explica las ventajas del modelo democrático y cuando es aconsejable aplicarlo. ¿Sería posible utilizarlo junto con un proyecto que utilice el modelo en cascada? Razona la respuesta.

Las ventajas del modelo democrático son:

- Incentiva la creatividad y la comunicación (comunicación horizontal y sin jefes) permitiendo resolver problemas de dificultad alta. Permite una programación sin ego donde unos aprenden de otros y todos se sienten responsables y parte del proyecto, obteniendo una fiabilidad alta en los programas y una alta productividad. Si hay una baja entre los miembros del equipo es más fácil suplirla ya que todos trabajamos en las diferentes disciplinas.

Es aconsejable utilizarlo en problemas de dificultad alta, de tamaño pequeño, donde la duración del equipo sea larga, modularidad baja, fiabilidad alta, fecha de entrega flexible y comunicación alta.

Sería posible y aconsejable si se cumpliesen las características explicadas antes, pero suele emplearse más con modelos que emplean prácticas ágiles (informales) que con modelos pesados como el modelo en cascada. El modelo en cascada se suele utilizar en problemas grandes, con modularidad alta y dificultad y fiabilidad (media-baja), donde no se requiere una alta comunicación.

Los siguientes conceptos pueden pertenecer a una o varias metodologías. Indica a qué metodologías pertenece cada uno: XP, SCRUM o RUP.

- Programación en pareja: XP
- Reuniones diarias obligatorias : SCRUM
- Refactorización: XP
- Iteración: SCRUM, XP y RUP
- Coach: XP
- Casos de Uso: RUP
- Historias de Usuario: XP y SCRUM
- Iterativo e incremental: XP, SCRUM y RUP

En relación a los términos BAC (Budget At Completion), BCWS (Budgeted Cost of Work Scheduled), CPI (Cost Performance Index), SPI (Schedule Performance Index) indica el significado de las siguientes situaciones en un proyecto:

- Si $BAC \neq BCWS$ entonces la tarea todavía no ha terminado. Hasta la fecha se habrá realizado un % del trabajo planificado en el proyecto.
- Si $BAC = BCWS$ entonces la tarea se debería haber completado
- Si $BCWS = 0$ entonces la tarea todavía no ha comenzado
- Si $CPI > 1$ vamos bien de coste, gastando menos de lo estimado
- Si $SPI < 1$ vamos mal en tiempo, vamos retrasados

¿Qué derechos tienen las/os interesadas/os por nuestro proyecto respecto al tratamiento de sus datos de carácter personal? ¿Cuándo se considera que los datos deben de tener un nivel de seguridad alto teniendo que hacer Evaluaciones de Impacto en Protección de Datos.?

- Información y Acceso, Rectificación, Supresión (Olvido), Limitación, Portabilidad, Oposición y decisiones individuales automatizadas.
- Se consideran de nivel alto cuando tienen datos de salud, genéticos y/o biométricos, condenas o infracciones penales, ideología política y/o sexual, datos de menores o vulnerables con riesgo de exclusión social, tratamientos de datos a gran escala, creación de perfiles, etc.

¿Qué diferencia existe en la pirámide de Maslow entre las necesidades de autorealización, estima y necesidad social?

La necesidad social hace referencia a la necesidad del individuo de sentirse reconocido como miembro de un grupo. La necesidad de estima es la necesidad del individuo de que se reconozca su trabajo y sus logros. La autorealización es la necesidad psicológica más elevada del ser humano, se halla en la cima de la pirámide, y viene marcada por el carácter vocacional y el desarrollo personal.

¿Microsoft Project permite detectar las actividades críticas del proyecto? En caso afirmativo, de qué forma o formas permite hacerlo. ¿Qué es la línea base de un proyecto? ¿Qué información nos aporta una tarea resumen del proyecto?

- Las actividades críticas se pueden ver poniendo la visualización en formato “Gráfico de actividades” y las marca en color rojo o viendo las holguras totales (que son 0).
- La línea base establece una copia del proyecto en un momento dado para poder hacer luego comparaciones. Por ejemplo, se puede utilizar para establecer el plan previsto y luego ir haciendo comparaciones con él.
- La tarea resumen muestra el resumen de un conjunto de actividades, aunando los esfuerzos, duraciones, valores EVA, etc.

MOHA

- 1. En el registro de la propiedad intelectual de un software bajo una relación laboral ¿para qué sirve el documento de declaración de autor asalariado de un empleado?**
 - a. Para la cesión de los derechos de explotación del software del empleado a la empresa
 - b. Para la cesión de los derechos de autoría del software del empleado a la empresa
 - c. Para la cesión de los derechos de explotación del software de la empresa al empleo
 - d. Para la cesión de los derechos de autoría del software de la empresa al empleado
- 2. Tenemos 3 actividades: A, B, y C. A precede a B. Las holguras totales son: HT(A)=4, HT(B)=2, HT(C)=3. Las holguras libres son: HL(A)=3, HL(B)=2, HL(C)=3. Si A se retrasa 4 días, ¿Qué les ocurre a B y C?**
 - a. Nada, ya que no sobrepasamos la holgura total de A
 - b. La holgura total de B y C pasa a ser 1 y 2 días respectivamente
 - c. B se convierte en crítica
 - d. Ninguna de las anteriores es cierta
- 3. Tenemos 3 actividades: A, B, y C. A precede a B. Las holguras totales son: HT(A)=2, HT(B)=4, HT(C)=3. Las holguras libres son: HL(A)=3, HL(B)=2, HL(C)=3. Si A se retrasa 2 días, ¿Qué les ocurre a B y C?**
 - a. Nada, ya que no sobrepasamos la holgura total de A
 - b. La holgura total de B y C pasa a ser 2 y 1 días respectivamente
 - c. B se convierte en crítica
 - d. Las condiciones iniciales no se pueden dar
- 4. En el análisis EVA, la tarea/proyecto va con retraso según la agenda si:**
 - a. BCWP>BCWS
 - b. BCWP<ACWP
 - c. BCWP>ACWP
 - d. BCWP<BCWS
- 5. Tenemos una tarea en MicrosoftProject de 2 días de duración con un recurso asignado a tiempo completo ¿Qué ocurre si le asignamos otro recurso adicional también a tiempo completo?**
 - a. La duración de la tarea se reducirá a la mitad
 - b. El trabajo necesario para realizar la tarea se duplicará al ser dos recursos
 - c. Nada, simplemente tiene en cuenta que la tarea la realizan 2 recursos
 - d. Depende del tipo de tarea que tenga establecido
- 6. Según la LOPD ¿A partir de qué edad puede un menor expresar el consentimiento en términos de protección de datos?**
 - a. El padre o tutor legal será el que lo haga en su nombre mientras forme parte de la unidad familiar
 - b. 16
 - c. 18
 - d. 14

- 7. Según la ley de propiedad intelectual ¿Cuál de las siguientes figuras tiene un plazo de protección indefinido?**
- a. Derechos de autor
 - b. Secretos comerciales**
 - c. Patentes
 - d. Marcas
- 8. La cadena de actividades que determina la duración global mínima de un proyecto se conoce como....**
- a. Cronograma
 - b. Línea base
 - c. Camino crítico**
 - d. Diagrama de Gantt
- 9. En el análisis EVA, el indicador CPI se calcula como:**
- a. BCWP/ACWP**
 - b. BCWP-ACWP
 - c. BCWP/BCWS
 - d. BCWP-BCWS
- 10. ¿Cuál de los siguientes parámetros EVA nos informa de la cantidad de trabajo planificado al final del proyecto/tarea?**
- a. BAC**
 - b. BCWP
 - c. ACWP
 - d. BCWS
- 11. En el análisis EVA, el indicador SV se calcula como:**
- a. BCWP-BCWS**
 - b. BCWP/BCWS
 - c. BCWP/ACWP
 - d. BCWP-ACWP
- 12. ¿Qué aspecto ha sido el más importante a la hora de establecer los objetivos de las fases de elaboración y construcción en el modelo UP seguido para la planificación del software de gestión de autónomos?**
- a. Los requerimientos funcionales y no funcionales
 - b. La arquitectura elegida**
 - c. Precisamente el modelo de proceso
 - d. Los recursos empleados
- 13. En el análisis EVA, la tarea/proyecto está por debajo de lo presupuestado si:**
- a. BCWP<ACWP
 - b. BCWP>BCWS
 - c. BCWP>ACWP**
 - d. BCWP<ACWP
- 14. ¿Cuál es la distribución de tiempo dentro de un desarrollo de equipo?**
- a. 20% Trabajando sólo, 30% actividades no productivas, 50% Interactuando con otra gente.
 - b. 30% Trabajando sólo, 20% actividades no productivas, 50% Interactuando con otra gente.**
 - c. 30% Trabajando sólo, 20% actividades de gestión, 50% Interactuando con otra gente.

- d. 20% Implementando, 30% Interactuando con otra gente y 50% en actividades de gestión

15. ¿Qué técnica de estimación de costes se basa en contabilizar pantallas, informes y módulos de cálculo?

- a. Pricing to win
- b. Estimación por analogía
- c. Puntos de función
- d. Puntos objeto**

16. Una de las recomendaciones que se deben tener en cuenta durante la estimación de recursos y tiempo de un proyecto informático es...

- a. Asignar tareas en el mismo periodo, así si una no se puede realizar se realizará otra
- b. Identificar las tareas que consumen más recursos**
- c. Todas las respuestas son validas
- d. Planificar siempre como si los recursos estuvieran asignados al 100%

17. ¿Cuál de las siguientes no es una causa legal para poder transmitir los derechos de propiedad Intelectual?

- a. Mortis causa
- b. En virtud de relación mercantil**
- c. En virtud en relación laboral
- d. Inter vivos

18. En una iteración del Plan detallado se establece una actividad de revisión de los riesgos del proyecto. Dicha actividad consiste en:

- a. Revisar los identificadores potenciales que figuran en el plan**
- b. Volver a analizar la probabilidad y los efectos de todos los riesgos del plan
- c. Identificar más riesgos para el proyecto
- d. Revisar las estrategias establecidas en el plan de riesgos

19. La LOPDGDD no será de aplicación en el caso de

- a. Datos de personas fallecidas
- b. Datos ya excluidos en el RGPD
- c. Materias consideradas clasificadas
- d. Todas son ciertas**

20. En el caso de producirse un retraso en una actividad ¿Cuál de las siguientes medidas sería la que menos afectaría a la agenda del proyecto?

- a. Retrasarla según su holgura libre**
- b. Eliminar alguna funcionalidad
- c. Reemplazar a los recursos asignados a la misma
- d. Retrasarla según su holgura total

21. El modelo basados en prototipos

- a. Reduce horas de implementación
- b. Es posible utilizarlo solo con lenguajes orientados a objetos
- c. El sistema resultante es un prototipo
- d. Permite reducir riesgos y costes de desarrollo**

22. ¿Cuál de las siguientes no es una técnica de estimación de costes de un proyecto?

- a. Ley de Parkinson
- b. Gantt**
- c. Modelado algorítmico de costes
- d. Juicio Experto

23. Monitorizar una agenda tiene como objetivo fundamental

- a. Consiste en comprobar si la agenda real se ajusta a la planificada
- b. Sirve para justificar las horas dedicadas a un proyecto
- c. Presentar al cliente las desviaciones producidas en el proyecto para que sea consciente de que se va a retrasar la entrega
- d. Se usa como seguimiento interno

24. Cuando en el desarrollo del plan de proyecto existen restricciones de tiempo y presupuesto establecidas por el cliente, estas deben...

- a. Ser incluidas y ajustar el cronograma
- b. Deben ser consideradas evaluando su impacto en el proyecto
- c. Ser descartadas ya que hay que ignorar completamente al cliente
- d. Ser descartadas ya que no se podrán hacer ajustes en el presupuesto

25. ¿Hasta qué momento durante la ejecución de un proyecto, la contingencia debe ser considerada?

- a. Hasta que el tiempo y los avances del proyecto demuestren que el riesgo se ha reducido lo suficiente o desaparecido
 - b. Hasta que el proyecto se dore de mas presupuesto
 - c. Hasta ejecutar un 70% del presupuesto asignando a la contingencia
 - d. Hasta el fin del proyecto
- 26. ¿Qué importancia tiene el RGPD?**
- a. Establece un marco para el registro de patentes de empresas
 - b. Establece un marco para regular a las empresas sobre las llamadas que un ciudadano puede recibir para vender productos
 - c. Es una recomendación europea que esta en estudio para su redacción definitiva
 - d. Establece un marco europeo relativo a la protección de las personas físicas en lo que respecta al tratamiento de sus datos personales y a la libre circulación de estos datos

27. ¿Qué podemos decir de una tarea si conocemos los valores BCWS=BCWP=600 y ACWP=1000?

- a. Va adelantada respecto a la agenda
- b. Ha gastado menos de lo previsto NO
- c. Va atrasada respecto a la agenda NO
- d. No podemos saber si ha terminado sin conocer su BAC

28. Muchas organizaciones fortalecen sus equipos de gestión de riesgos, para que estos sean identificados a priori, pero lo que ocurre realmente es...

- a. Es imposible identificar los riesgos a priori
- b. Lo mas usual es identificar los riesgos de ejecución
- c. No se identifican a tiempo
- d. Identifican normalmente riesgos muy poco probables

29. ¿Qué podemos decir de una tarea si conocemos los valores BCWS=1200 y BCWP=ACWP?

- a. Va retrasada
- b. Todas son ciertas
- c. No ha comenzado
- d. Debería haber comenzado

30. Dado que un riesgo siempre va a existir hay que....

- a. No es necesario tener planes de contingencia porque la experiencia demuestra que no es posible contener los riesgos en su totalidad
- b. Contar con un proceso de gestión de proyectos adecuado para reducir los riesgos en su totalidad
- c. Es falso, los riesgos no siempre existen
- d. Establecer mecanismos de seguimiento y prevención

31. El plan detallado de un proyecto...

- a. Contiene todas las actividades incluidas en el plan general
- b. Se hace siempre de dos en dos iteraciones
- c. Solo corresponde a una fase del proyecto
- d. Todas son falsas

32. Respecto a la protección de datos del software de gestión de autónomos, indica cuál de las siguientes sería una acción obligatoria para cumplir con la legislación actual en el caso de desarrollaría como aplicación web:

- a. Incluir un apartado de aviso legal en la web
- b. Nombrar un delegado de protección de datos
- c. Ninguna de las otras respuestas es correcta
- d. Resgitrar los ficheros con los datos de carácter personal en la AEPD

33. En la pirámide de jerarquía de necesidades donde se encuentran la mayoría de jefes de proyecto

- a. No les afecta esta jerarquía, es solo para empleados
- b. Necesidades autorrealización
- c. Necesidades sociales
- d. Necesidades seguridad

34. Las características del software libre son:

- a. Tiene limitaciones de usuario
- b. Solo se pueden usar en la enseñanza e investigación
- c. Están disponibles en una pagina web que se llama www.softlibre.com
- d. Se distribuye libremente y es de código abierto

35. ¿Qué podemos decir de una tarea si conocemos los valores BCWS=BCWP=0 y ACWP=800?

- a. El valor BCWP es erroneo
- b. El valor ACWP es erroneo
- c. Debería haber comenzado y no lo ha hecho
- d. No debería haber comenzado, pero si lo ha hecho

36. ¿Cuál de los siguientes tipos de licencia no es propia del software privativo?

- a. Apache
- b. PKC
- c. ROK
- d. EULA

37. ¿Cuál de los siguientes tipos de licencia no es propia del software libre?

- a. GPL
- b. Creative Commons
- c. BSD
- d. OEM

38. Tenemos un proyecto con los siguientes datos del modelo EVA

Tarea	BCWS	BCWP	ACWP	BAC
Tarea0	2800	1800	1800	2600
Tarea1	600	600	600	1200
Tarea2	1200	0	600	2600
Tarea3	1000	1200	800	2000
Tarea4	0	0	0	600

- a. La tarea 1 ha comenzado y esta realizada al 50%, va desarrollándose en el tiempo previsto y gastando también lo previsto
- b. La tarea 3 va adelantada y consumiendo menos de lo previsto
- c. La tarea 4 aun no ha comenzado
- d. La tarea 0 va retrasada y consumiendo mas de lo previsto**

39. En el análisis EVA, la tarea/proyecto va adelantada según la agenda si

- a. BCWP<BCWS
- b. BCWP<ACWP
- c. BCWP>ACWP
- d. BCWP>BCWS**

40. En el caso de que hayamos desarrollado un software de gestión de autónomos en forma de aplicación de escritorio ¿Quién debería de encargarse del cumplimiento de la legislación a nivel de protección de datos?

- a. Nosotros como desarrolladores de la aplicación**
- b. El cliente o clientes que hayan comprado nuestro software**
- c. Depende del presupuesto firmado por ambas partes**
- d. No sería necesario al tratarse de una aplicación del escritorio

41. En el modelo UP

- a. Tenemos un reléase por fase
- b. Todas son ciertas
- c. Hay una versión por iteración**
- d. Puede haber más de una versión por iteración**

42. La normativa vigente en España en materia de protección de datos es:

- a. LOPD 15/1999 y RGPD UE de 2016
- b. RGDP UE de 2018 y LOPD 3/2018
- c. RGDP UE de 2016 y LOPD 3/2018**
- d. Actualmente en nuestro Estado solo nos afecta la LOPD 3/2018

43. De los siguientes derechos regulados en la LOPD ¿Cuál es un derecho digital?

- a. Derecho de supresión
- b. Derecho a la intimidad en el ámbito laboral**
- c. Derecho de rectificación
- d. Derecho de oposición

44. Según el RGPD, el RAT (registro de Actividades de Tratamiento) es obligatorio para:

- a. Cualquier empresa pública o privada
- b. Solo para la Administración Pública
- c. Empresas de 250 empleados o más**
- d. Empresas de menos de 250 empleados y PYMES

45. El plazo de protección de los derechos de autor según la LPi es:

- a. 20 años a partir de la fecha de registro
- b. 70 años a partir del fallecimiento del autor**
- c. 70 años a partir de la fecha de registro
- d. 20 años a partir del fallecimiento del autor

- 1- En el registro de la propiedad intelectual de un software bajo una relación laboral, ¿para qué sirve el documento de declaración de autor asalariado de un empleado?
- Para la cesión de los derechos de explotación del software del empleado a la empresa
 - Para la cesión de los derechos de autoría del software del empleado a la empresa
 - Para la cesión de los derechos de explotación del software de la empresa al empleado
 - Para la cesión de los derechos de autoría del software de la empresa al empleado
- 2- Tenemos la actividad A B y C. A precede a B. Las holguras totales son $HT(A)=2, HT(B)=4, HT(C)=3$. Las holguras libres son $HL(A)=3, HL(B)=2, HL(C)=3$. Si A se retrasa 2 días. ¿qué les ocurre a B y C?
- Nada, ya que no sobrepasamos la holgura total de A.
 - La holgura total de B y C pasa a ser de 2 y 1 días respectivamente.
 - B se convierte en crítica
 - Las condiciones iniciales no se pueden dar.
- 3- En el análisis EVA, la tarea/proyecto va con retraso según la agenda si:
- $BCWP < ACWP$
 - $BCWP > BCWS$
 - $BCWP > ACWP$
 - $BCWP < BCWS$
- 4- Tenemos una tarea en Microsoft Project de 2 días de duración con un recurso asignado al tiempo completo. ¿Qué ocurre si le asignamos otro recurso adicional también a tiempo completo?
- La duración de la tarea se reducirá a la mitad
 - El trabajo necesario para realizar la tarea se duplicará al ser de dos recursos
 - Nada, simplemente tiene en cuenta que la tarea la realizan 2 recursos
 - Depende del tipo de tarea que tenga establecido

- 5- Según LOPDGDD ¿A partir de qué edad puede un menor expresar el consentimiento en términos de protección de datos?
- El padre o tutor legal será el que lo haga en su nombre mientras forme parte de la unidad familiar
 - 16
 - 18
 - 14**
- 6- Según la ley propiedad intelectual, ¿Cuál de las siguientes figuras tiene un plazo de protección indefinido?
- Derecho de autor (creo)
 - Secretos comerciales**
 - Patentes
 - Marcas
- 7- En el modelo UP...
- La fase de Transición suele ser la de mayor duración
 - Las iteraciones pueden tener duraciones distintas (creo)**
 - La fase de inicio siempre consta de una única iteración
 - El numero de iteraciones es igual en todas las fases
- 8- ¿Qué tarea no podemos realizar en Microsoft Project?
- Todas las tareas de estas contestaciones se pueden realizar en Microsoft Project
 - La planificación detallada de todo el proyecto
 - La estimación del proyecto**
 - El seguimiento del proyecto
- 9- La cadena de actividades que determina la duración global mínima de un proyecto se conoce como:
- Cronograma (no)**
 - Línea base
 - Camino crítico**
 - Diagrama de Gantt

10- En el análisis EVA, el indicador CPI se calcula como:

- a. BCWP / ACWP
- b. BCWP – ACWP
- c. BCWP / BCWS
- d. BCWP – BCWS

11- ¿Cuál de los siguientes parámetros EVA nos informa de la cantidad de trabajo planificado al final del proyecto/tarea?

- a. BAC
- b. BCWP
- c. ACWP
- d. BCWS

12- En el análisis EVA el indicador SV se calcula como:

- a. BCWP – BCWS
- b. BCWP/BCWS
- c. BCWP/ACWP
- d. BCWP – ACWP

13- ¿Qué aspecto ha sido el más importante a la hora de establecer los objetivos de las fases de elaboración y construcción en el modelo UP seguido para la planificación del software de gestión de autónomos?

- a. Los requerimientos funcionales y no funcionales
- b. La arquitectura elegida
- c. Los recursos empleados
- d. Precisamente el modelo de proceso

14- En el análisis EVA, la tarea/proyecto está por debajo de lo presupuestado si:

- a. BCWP < ACWP
- b. BCWP > ACWS

c. BCWP > ACWP

d. BCWP < ACWP

15- Un plan de proyecto...

- a. Proporciona visibilidad al progreso del proyecto
- b. Es una representación abstracta de un proceso que proporciona información parcial sobre el proceso software
- c. Es un conjunto de actividades que tienen como objetivo el desarrollo de un producto software
- d. Todas las afirmaciones son falsas.

16-¿Cuál es la distribución de tiempo dentro de un desarrollo de proyecto?

- a. 30% Trabajando solo, 20% actividades de gestión, 50% interactuando con otra gente
- b. 30% Trabajando solo, 20% actividades no productivas, 50% interactuando con otra gente
- c. 20% Trabajando solo, 30% actividades no productivas, 50% interactuando con otra gente
- d. 20% Implementando, 30% interactuando con otra gente, 50% en actividades de gestión
- e. No sabe/ no contesta

17-¿Qué técnica de estimación de costes se basa en contabilizar pantallas, informes y módulos de cálculo?

- f. Pricing to Win
- g. Estimados por analogía
- h. Puntos de función
- i. No sabe/ no contesta
- j. Puntos objeto

18- Una de las recomendaciones que se deben tener en cuenta durante la estimación de recursos y tiempo de un proyecto informático es:

- k. Asignar varias tareas en el mismo periodo, así si una no se puede realizar se realizara otra
- l. Identificar las tareas que consumen más recursos

- m. Todas las respuestas son válidas
- n. No sabe/ no contesta
- o. Planificar siempre como si los recursos estuvieran asignados al 100%

19- ¿Cuál de las siguientes no es una causa legal para poder transmitir los derechos de Propiedad Intelectual?

- p. Vortis causa
- q. **En virtud de la relación mercantil**
- r. En virtud de la relación laboral
- s. No sabe/ no contesta
- t. Inter vivos

20-En una iteración de Plan detallado se establece una actividad de revisión de riesgos del proyecto. Dicha actividad consiste en:

- u. **Revisar los identificadores potenciales que figuran en el plan**
- v. Volver a analizar la probabilidad y los efectos de todos los riesgos del plan
- w. No sabe/ no contesta
- x. Identificar más riesgos para el proyecto
- y. Revisar las estrategias establecidas en el plan de riesgos

21- La LOPDGDD no será de aplicación en el caso de:

- a. Datos de personas fallecidas
- b. Datos ya excluidos en el RGPD
- c. No sabe/ No contesta
- d. Materias consideradas clasificadas
- e. **Todas son ciertas**

22- En el caso de producirse un retraso en una actividad, ¿cuál de las siguientes medidas sería la que menos afectaría a la agenda del proyecto?

- a. No sabe/ no contesta
- b. **Retrasarla según su holgura libre**
- c. Eliminar alguna funcionalidad
- d. Reemplazar a los recursos asignados a la misma

- e. Retrasarla según su holgura total

23- El modelo basado en prototipos...

- a. Reduce las horas de implementación
- b. Es posible utilizarlo solo con lenguajes orientados a objetos
- c. El sistema resultante es un prototipo(no)
- d. No sabe/ no contesta
- e. Permite reducir riesgos y costes de desarrollo

24- ¿Cuál de las siguientes no es técnica de estimación de costes de un proyecto?

- a. Ley de Parkinson
- b. Gantt
- c. Modelado algorítmico de costes
- d. No sabe/ no contesta
- e. Juicio Experto

25- Monitorizar una agenda tiene como objetivo fundamental:

- a) Consiste en comprobar si la agenda real se ajusta a la planificada.
- b) Sirve para justificar las horas dedicadas a un proyecto
- c) No sabe
- d) Presentar al cliente las desviaciones en el proyecto para que sea consciente de que se va a retrasar la entrega
- e) Se usa como seguimiento interno

26- ¿Cuál de los siguientes modelos de proceso es evolutivo?

- a) No sabe
- b) Cacada
- c) Espiral y unificado
- d) Espiral
- e) Unificado

27-Las releases de una sistema:

- a) Superan en numero a las versiones
- b) **Todas son ciertas**
- c) Tienen un costo menor que las versiones
- d) No sabe
- e) Incluyen documentación técnica

28- Cuando el desarrollo del plan del proyecto existen restricciones de tiempo y presupuesto establecidas por el cliente, estos deben..

- a) **Ser incluidas y ajustar el programa**
- b) Deben ser consideradas evaluando su impacto en el proyecto
- c) Ser descartadas ya que hay que ignorar completamente al cliente
- d) No sabe
- e) Ser descartadas ya que no se podrán hacer ajustes en el presupuesto

29-En el modelo UP:

- a) El numero de iteraciones es igual en todas las fases
- b) No sabe
- c) Las fases de elaboración y construcción tienen el mismo número de iteraciones
- d) **Todas son falsas**
- e) La fase de inicio siempre tiene 1 iteración

30- ¿Hasta que momento durante la ejecución de un proyecto, la contingencia debe ser considerada?

- a) Hasta que el tiempo y los avances del proyecto demuestren que el riesgo se ha reducido lo suficiente o desaparecido.
- b) Hasta que el proyecto se dote de más presupuesto.
- c) Hasta ejecutar un 70 % de presupuesto asignado a la contingencia.
- d) No sabe
- e) **Hasta el fin del proyecto.**

31- ¿Qué importancia tiene el RGPD?

- a) Establece un marco para el registro de patentes de empresas.
- b) Establece un marco para regular a las empresas sobre las llamadas que un ciudadano puede recibir para venderte productos.

- c) No sabe
- d) Es una recomendación europea que está en estudio para su redacción definitiva.
- e) **Establece un marco europeo relativo a la protección de las personas físicas en lo que respecta al tratamiento de sus datos personales y a la libre circulación de estos datos.**

32- ¿Qué podemos decir de una tarea si conocemos los valores BCWS=BCWP=600 y ACWP=1000?

- a) Va adelantada respecto a la agenda
- b) Ha gastado menos de lo previsto
- c) Va atrasada respecto a la agenda
- d) **No podemos saber si ha terminado sin conocer su BAC**
- e) No sabe

33-Muchas organizaciones fortalecen sus equipos de gestión de riesgos, para que estos sean identificados a priori, pero lo que ocurre realmente es..

- a) Es imposible identificar riesgos a priori
- b) No sabe
- c) **Lo más usual es identificar los riesgos de ejecución**
- d) No se identifican a tiempo
- c) Identifican normalmente riesgos muy poco probables

34- ¿Qué podemos decir de una tarea si conocemos los valores BCWS=1200 y BCWP=ACWP?

- a) Va retrasada
- b) No sabe
- c) **Todas son ciertas**
- d) No ha comenzado
- e) Debería haber comenzado

35- Dado que un riesgo siempre va a existir hay que...

- a) No es necesario tener planes de contingencia porque la experiencia demuestra que no es posible contener los riesgos en su totalidad.
- b) No sabe
- c) Contar con un proceso de gestión de proyectos adecuado para reducir los riesgos en su totalidad.
- d) Es falso, los riesgos no siempre existen.
- e) **Establecer mecanismos de seguimiento y prevención.**

36. El Plan detallado de un proyecto... Seleccione una:

- a) No sabe / No contesta
- b) Contiene todas las actividades incluidas en el plan general**
- c) Todas son falsas
- d) Se hace siempre de dos en dos iteraciones
- e) Solo corresponde a una fase del proyecto

37. Respecto a la protección de datos del software de gestión de autónomos, indica cuál de las siguientes sería una acción obligatoria para cumplir con la legislación actual en el caso de desarrollarla como aplicación web. Seleccione una:

- a) Incluir un apartado de aviso legal en la web
- b) Nombrar un delegado de protección de datos**
- c) Ninguna de las otras respuestas es correcta
- d) Registrar los ficheros con los datos de carácter personal en la AEPD
- e) No sabe / No contesta

38. En la pirámide de jerarquía de necesidades donde se encuentran la mayoría de jefes de proyecto:

- a) No les afecta esta jerarquía, es solo para empleados.**
- b) No sabe / No contesta.
- c) Necesidades autorrealización.
- d) Necesidades sociales
- e) Necesidades seguridad

39. Las características del software libre son:

- a) Tiene limitaciones de usuario
- b) No sabe / No contesta
- c) Solo se pueden usar en la enseñanza e investigación
- d) Están disponibles en una página web que se llama www.softlibre.com
- e) Se distribuyen libremente y es de código abierto**

40. ¿Qué podemos decir de una tarea si conocemos los valores $BCWS=BCWP=0$ y $ACWP=800$?

- a) No sabe / No contesta
- b) El valor BCWP es erróneo

- c) El valor ACWP es erróneo
- d) Debería haber comenzado y no lo ha hecho
- e) No debería haber comenzado, pero sí lo ha hecho

41. ¿Cuál de los siguientes tipos de licencia no es propia del software privativo?

- a) Apache → libre
- b) PKC
- c) ROK
- d) No sabe / No contesta
- e) EULA

42. Tenemos un proyecto con los siguientes datos del modelo EVA. ¿Cuál de las siguientes afirmaciones es falsa?

Tarea	BCWS	BCWP	ACWP	BAC
Tarea0	2800	1800	1800	2600
Tarea1	600	600	600	1200
Tarea2	1200	0	600	2600
Tarea3	1000	1200	800	2000
Tarea4	0	0	0	600

- a) La tarea 1 ha comenzado y está realizada al 50%, va desarrollándose en el tiempo previsto y gastando también lo previsto → VERDADERA
- b) No sabe / No contesta
- c) La tarea 3 va adelantada y consumiendo menos de lo previsto → VERDADERA
- d) La tarea 4 aún no ha comenzado → VERDADERA
- e) La tarea 0 va retrasada y consumiendo más de lo previsto → FALSA

43. En el análisis EVA la tarea/proyecto va adelantada según la agenda si:

- a) BCWP < BCWS
- b) BCWP < ACWP
- c) No sabe / No contesta
- d) BCWP > ACWP
- e) BCWP > BCWS

44. En el caso de que hayamos desarrollado un software de gestión de autónomos en forma de aplicación de escritorio, ¿quién debería encargarse del cumplimiento de la legislación a nivel de protección de datos?

- a) Nosotros como desarrolladores de la aplicación
- b) Los clientes o clientes que hayan comprado nuestro software
- c) Depende del presupuesto firmado por ambas partes
- d) No sabe / No contesta
- e) No sería necesario al tratarse de una aplicación de escritorio

45. En el modelo UP:

- a) No sabe / No contesta
- b) Tenemos un release por fase
- c) Todas son ciertas
- d) Hay una versión por iteración**
- e) Puede haber más de una versión por iteración

1. ¿Cuál de los siguientes modelos de proceso es evolutivo?

- a. Cascada.
- b. Unificado.
- c. Espiral.
- d. Las respuestas b y c son correctas.**

2. ¿Qué tarea no podemos realizar en Microsoft Project?

- a. La planificación detallada de todo el proyecto.
- b. El seguimiento del proyecto.
- c. La estimación del proyecto.**
- d. Todas las tareas anteriores se pueden realizar en Microsoft Project.

3. En el modelo UP...

- a. El número de iteraciones es igual en todas las fases.
- b. La fase de inicio siempre tiene 1 iteración.
- c. Las fases de elaboración y construcción tienen el mismo número de iteraciones.
- d. Todas son falsas.**

4. ¿Cuál de las siguientes afirmaciones es falsa?

- a. El conocimiento sintáctico se adquiere mediante experiencia y aprendizaje activo.
- b. El conocimiento semántico se adquiere mediante memorización.
- c. El nuevo conocimiento semántico puede interferir con conocimiento semántico ya existente.
- d. Todas son falsas.**

5. Las releases de un sistema.

- a. Tienen un coste menor que las versiones.
- b. Superan en número a las versiones.
- c. **Incluyen documentación técnica.**
- d. Todas son ciertas.

6. En el proceso unificado de desarrollo de software:

- a. Los requisitos se establecen en la primera fase y no se permite ningún cambio en las fases siguientes.
- b. **La tarea de establecer los requisitos es más intensa en la primera fase aunque en las fases siguientes los requisitos se pueden modificar.**
- c. El establecimiento de requisitos requiere del mismo esfuerzo en todas las fases.
- d. Los requisitos siempre se establecen a partir de prototipos.

7. Las metodologías ágiles se caracterizan

- a. **Simplificar la complejidad de otras metodologías.**
- b. Por inventarse a finales del S XIX en la revolución industrial.
- c. Por hacer más énfasis en la documentación a entregar.
- d. Ponen en el centro el proyecto a entregar no al cliente.

8. ¿Qué beneficios tiene las metodologías agiles?

- a. Rapidez, documentación muy completa y trabajar de forma intensa la última semana antes de la entrega.
- b. **Eficiencia, rapidez y visión cliente.**
- c. Cliente, proveedores y cumplir el presupuesto.
- d. Individualismo.

9. Indica cuál de las siguientes metodólogas son ágiles.

- a. Cobol, c++, Java.
- b. Uml , PPo, itil.
- c. **Kanban, Scrum, Extrem Programming.**
- d. Ninguna de las anteriores.

10. Indica cuál de las siguientes características aplican al modelo Canvas de negocio.

- a. Es una herramienta nueva que surge después de la segunda guerra mundial.
- b. **Se basa en existencia de una propuesta de valor que tenemos que comunicar al cliente mediante un modelo canvas y llenando todos los apartados.**
- c. Se utiliza solo para proyectos de robótica.
- d. Se puede dejar apartados sin llenar, al menos con llenar el 50 por ciento es válido.

11. Señala cuál de las siguientes características corresponden a la metodología SCRUM.

- a. **Es incremental, reuniones diarias y se basa en roles.**
- b. Reuniones diarias, y trabajo individual.
- c. Lo importante es el trabajo individual y la documentación que se realice sobre la aplicación.
- d. Se basa en roles y pasa a segundo plano la calidad del proyecto.

12. Las características del software libre son:

- a. **Se distribuyen libremente y es de código abierto.**
- b. Están disponibles en una página web que se llama www.softlibre.com.
- c. Sólo se pueden usar en la enseñanza e investigación.
- d. Tiene limitaciones de usuario.

13. ¿Qué importancia tiene el RGPD?

- a. Establece un marco para el registro de patentes de empresas.
- b. Establece un marco para regular a las empresas sobre las llamadas que un ciudadano puede recibir para venderle productos.
- c. **Establece un marco europeo relativo a la protección de las personas físicas en lo que respecta al tratamiento de sus datos personales y a la libre circulación de estos datos.**
- d. Es una recomendación europea que está en estudio para su redacción definitiva.

14. Monitorizar una agenda tiene como objetivo fundamental

- a. Presentar al cliente las desviaciones producidas en el proyecto para que sea consciente de que se va a retrasar la entrega.
- b. **Consiste en comprobar si la agenda real se ajusta a la planificada**
- c. Se usa como seguimiento interno.
- d. Sirve para justificar las horas dedicadas a un proyecto.

15. ¿Cuál de estas características no es de la metodología SCRUM?

- a. Metodología iterativa e incremental.
 - b. Reuniones diarias.
 - c. **Descanso cada 15 días para renovar las ideas.**
 - d. Demo al cliente al final de cada iteración.
1. ¿Cuál de los siguientes modelos de proceso es evolutivo?
- a. Cascada.
 - b. Desarrollo rápido de aplicaciones.
 - c. **Espiral.**
 - d. Las respuestas b y c son correctas.

2. ¿Cuál de los siguientes libros es el principal recomendado en la asignatura?

- a. **Ingeniería del software" 7^a Edición. Ian Sommerville.**
- b. "Técnicas de Programación y Control de Proyectos". C.Romero López. Pirámide
- c. Ingeniería del Software: un enfoque práctico. 5^o edición.Roger S. Pressman.
- d. "Desarrollo y Gestión de Proyectos Informáticos". Steve McConnell.

3. ¿Qué tarea no hemos realizado en Microsoft Project?

- a. La planificación detallada de todo el proyecto.
- b. El seguimiento del proyecto.
- c. **La estimación del proyecto.**
- d. No hemos realizado ninguna de las anteriores en Microsoft Project.

4. En el modelo UP...

- a. El número de iteraciones es igual en todas las fases.
- b. La fase de inicio siempre tiene 1 iteración.
- c. Las fases de elaboración y construcción tienen el mismo número de iteraciones.
- d. **Todas son falsas.**

5. ¿Qué multiplicador con valor alto reduciría el esfuerzo estimado en COCOMO?

- a. Fiabilidad de producto y complejidad.
- b. Reutilización requerida.
- c. Dificultad de la plataforma.
- d. **Experiencia del personal.**

6. ¿Cual de las siguientes afirmaciones es cierta?

- a. El conocimiento sintáctico se adquiere mediante experiencia y aprendizaje activo.
- b. El conocimiento semántico se adquiere mediante memorización.
- c. El nuevo conocimiento semántico puede interferir con conocimiento semántico ya existente.
- d. **Ninguna de las anteriores.**

7. Un plan de proyecto ...

- a. **Proporciona visibilidad al progreso del proyecto.**
- b. Es una representación abstracta de un proceso que proporciona información parcial sobre el proceso software.
- c. Es un conjunto de actividades que tienen como objetivo el desarrollo de un producto software.
- d. Todas las afirmaciones anteriores son falsas.

8. La gestión de proyectos ...

- a. Tiene como objetivo determinar qué hay que hacer, cómo hay que hacerlo y quién tiene que hacerlo.
- b. Está fuertemente condicionada por la implementación del sistema.
- c. Está fuertemente condicionada el diseño del sistema.
- d. Tiene como objetivo el confeccionar una agenda del proyecto.

9. El cono de la incertidumbre ...

- a. Dice que conforme avanza el proyecto menos errores de estimación tenemos.
- b. Dice que conforme avanza el proyecto más errores de estimación descubrimos.
- c. No puedo aplicarlo sino hago el seguimiento del proyecto.
- d. Todas las anteriores son falsas.

10. ¿Cuál es la distribución de tiempo dentro de un equipo de desarrollo?

- a. 20% Trabajando sólo, 30% actividades no productivas, 50% Interactuando con otra gente.
- b. 30% Trabajando sólo, 20% actividades no productivas, 50% Interactuando con otra gente.
- c. 30% Trabajando sólo, 20% actividades de gestión, 50% Interactuando con otra gente.
- d. 20% Implementando, 30% Interactuando con otra gente y 50% en actividades de gestión.

11. Las versiones de un sistema.

- a. Tienen un coste menor que las releases.
- b. Suelen superar en número a las releases.
- c. Incluyen documentación técnica.
- d. Todas son ciertas.

12. En el proceso unificado de desarrollo de software:

- a. Los requisitos se establecen en la primera fase y no se permite ningún cambio en las fases siguientes.
- b. La tarea de establecer los requisitos es más intensa en la primera fase aunque en las fases siguientes los requisitos se pueden modificar.
- c. El establecimiento de requisitos requiere del mismo esfuerzo en todas las fases.
- d. Los requisitos siempre se establecen a partir de prototipos.

13. Las metodologías de desarrollo de software tienen como objetivo:

- a. Establecer los requisitos de un sistema.
- b. Desarrollar un sistema dentro de un tiempo apropiado y con unos costes aceptables.
- c. Proporcionar sistemas que satisfagan a todas las personas afectadas por el mismo.
- d. Todas son ciertas.

14. En eXtreme programming:

- a. No es necesario que el cliente esté in situ, ya que, se pueden utilizar especificaciones escritas para la siguiente iteración.
- b. No es necesario que se escriban las pruebas antes de comenzar a programar.
- c. Cada iteración debe acabar con una versión lista para su producción.
- d. Ninguna de las anteriores.

15. En la fase de exploración de eXtreme programming:

- a. El cliente establece la prioridad de cada historia de usuario.
- b. El equipo de desarrollo establece la velocidad del proyecto.
- c. Se prueba la tecnología y se construye un prototipo.
- d. Se realizan tareas de soporte para el cliente.

16. En Scrum, la gráfica “Burn-Down” indica:

- a. La velocidad de desarrollo del equipo.
- b. La cantidad de trabajo pendiente en un sprint.
- c. Los posibles impedimentos del equipo.
- d. Todas son ciertas

17. En Scrum, un Sprint:

- a. Tiene una duración prefijada.
- b. Puede alargarse el tiempo necesario hasta acabar todas las tareas pendientes.
- c. Tiene la duración que el cliente indique.
- d. Ninguna de las anteriores.

18. Un impedimento en SCRUM:

- a. Se refleja como una línea horizontal en el gráfico burn-down.
- b. Se refleja como una línea vertical en el gráfico burn-down.
- c. Se refleja como una línea discontinua en el gráfico burn-d.own.
- d. Ninguna de las anteriores.

19. Los modelos de SCRUM y XP

- a. Tienen unos roles predefinidos.
- b. Utilizan historias de usuario para definir requisitos.
- c. Ofrecen resultados al cliente al finalizar cada iteración.
- d. Todas son ciertas.

20. ¿Qué es la refactorización en XP?

- a. Las acciones que realizan los programadores para pasar las pruebas unitarias
- b. Las acciones que realizan los testers para definir los casos de prueba
- c. La simplificación del código sin modificar su comportamiento
- d. Ninguna de las anteriores

1- La normativa vigente en España en materia de protección de datos es:

- a. LOPD 15/1999 y RGPD UE de 2016
- b. RGDP UE de 2018 y LOPD 3/2018
- c. RGDP UE de 2016 y LOPD 3/2018
- d. Actualmente en nuestro estado solo nos afecta la LOPD 3/2018

2- De los siguientes derechos regulados en la LOPD, ¿Cuál es un derecho digital?

- e. Derecho de supresión
- f. Derecho de la intimidad en el ámbito laboral
- g. Derecho de rectificación
- h. Derecho de oposición

3- Según la LPI, ¿Cuál de los siguientes figuras tienen un plazo de protección indefinido?

- i. Marcas
- j. Secretos comerciales
- k. Patentes
- l. Derecho de autor

4- ¿Cual de las siguientes no es una causa legal para poder transmitir los derechos de propiedad intelectual?

- m. Mortis causa
- n. En virtud de relación laboral
- o. Inter vivos
- p. En virtud de relación mercantil

5- ¿Cuál de los siguientes tipos de licencia no es propia del software libre?

- q. GPL
- r. Creative Commons
- s. BSD
- t. OEM

6- El plazo de protección de los derechos de autor según la LPI es:

- u. 20 años a partir de la fecha de registro
- v. 70 años a partir del fallecimiento del autor
- w. 70 años a partir de la fecha de registro
- x. 20 años a partir del fallecimiento del autor

7- ¿ Cual de los siguientes tipos de licencia no es propia del software privativo?

- y. EULA
- z. Apache
- aa. ROK
- bb. PKC

Analizando los valores de los parámetros EVA de la imagen...

4. Dados los siguientes datos de un proyecto. Explica cuál es la situación del proyecto para cada una de las tareas. ¿La tarea 1 se ha terminado? ¿Cómo podemos solucionar los problemas del proyecto? (2.5p)

Tarea	BCWS	BCWP	ACWP
Tarea R.	2800	1800	2600
Tarea 1	600	600	1000
Tarea 2	1200	0	0
Tarea 3	1000	1200	800
Tarea 4	0	0	800

1- ¿ Que podemos decir del proyecto en su conjunto?

- cc. El proyecto va adelantado y su productividad es baja
- dd. El proyecto va atrasado y su productividad es alta
- ee. El proyecto va adelantado y su productividad es alta
- ff. El proyecto va atrasado y su productividad es baja

2- ¿Qué podemos decir de la tarea 1?

- gg. No podemos saber si ha terminado sin conocer su BAC
- hh. Ha gastado menos de lo previsto
- ii. Va adelantada respecto a la agenda
- jj. Va atrasada respecto a la agenda

3- ¿Qué podemos decir de la tarea 2?

- kk. No ha comenzado
- ll. Debería haber comenzado
- mm. Va retrasada
- nn. Todas son ciertas

4- ¿Qué podemos decir de la tarea 3?

- oo. No ha comenzado
- pp. Va retrasada
- qq. Está gastando menos de lo previsto
- rr. Todas son falsas

5- ¿Qué podemos decir de la tarea 4?

- ss. Debería haber comenzado y no lo ha hecho
- tt. El valor BCWP es erróneo
- uu. El valor ACWP es erróneo
- vv. No debería haber comenzado per si lo ha hecho

1- Según el RGPD, el RAT (registro de actividades de tratamiento) es obligatorio para:

- ww. Cualquier empresa pública o privada
- xx. Solo para administraciones públicas
- yy. Empresas de 250 empleados o mas**
- zz. Empresas de menos de 250 empleados y PYMES

2- La LOPD no será de aplicación en caso de

- a. Materias consideradas clasificadas
- b. Datos de personas fallecidas
- c. Datos ya excluidos en RGPD
- d. Todas son ciertas**

Resumen Estimación

```
58 Tecnicas de estimacion de costes
59 -Modelado algoritmico de costes
60     Aproximacion basada en tamano software
61     Los costes crecen de forma exponencial con tamano
62     Como el tamano solo se conoce al final del desarollo, a medida que avanza
63     el proyecto la estimacion se hace mas precisa
64 -Juicio experto
65     Expertos en el campo predicen el coste. Iteracion hasta consenso
66     Muy barato y exacto. Cuesta encontrar expertos
67 -Estimacion por analogia
68     Comparar con proyectos similares
69     Preciso si hay info. Imposible si no hay proyectos parecidos
70 -Ley de Parkinson
71     Coste en funcion de recursos disponibles, usando todo el tiempo
72     Presupuesto NO abultados. Suele faltar tiempo
73 -Pricing to win
74     Coste cuanto quiera pagar cliente
75     Costes no reflejan trabajo requerido. Raro obtener programa que cumpla lo que pides
76
77     .Estimacion Descendente
78         Se puede usar sin conocer arquitectura ni componentes
79         Tiene en cuenta integracion, gestion de config y documentacion
80         Infra-estima costes de resolucion problemas tecnicos dificiles
81
82     .Estimacion Ascendente
83         Se puede usar con arquitectura y componentes conocidos
84         Estimaciones bastante exactas si se diseña con detalle
85         Infra-estima costes de sistema, integracion y documentacion
86
87     Estimacion deberia basarse en varios metodos
88     Si resultados de varios metodos difiere mucho, no hay suficiente info.
89     Muchas veces Pricing to Win una opcion
90
91
```

Resumen EVA

```
6 Tema 6 EVA
7
8     HL Holgura Libre: cuánto puedo retrasar una actividad sin afectar
9         a las siguientes ni al proyecto
10    HT Holgura Total   Cuanto se puede retrasar una actividad sin
11        afectar a la finalización del proyecto.
12
13    BCWS: Cuanto trabajo debería haberse completado hasta el momento
14    ACWP: Cuanto se ha gastado hasta el momento
15    BCWP: Cual es el valor del trabajo realizado hasta el momento
16    BAC: Cantidad de trabajo planificado al final del proyecto
17
18    Progreso
19        SV: Scheduled Variance           BCWP - BCWS
20        SPI: Scheduled Performance Index BCWP / BCWS
21        if BCWP > BCWS      proyecto adelantado segun ajenda
22
23    Productividad:
24        CV: Cost Variance             BCWP - ACWP
25        CPI: Cost Performance Index BCWP / ACWP
26        if BCWP > ACWP      proyecto gasta menos de lo planeado
27
28    BCWS = BAC  Tarea debería haber terminado
29    BCWS < BAC  Tarea no Terminada
30    BCWS = 0   Tarea no empezada
31    BCWS = BAC != BCWP      Deberíamos haber terminado pero no lo hemos hecho
32    BCWS = BCWP = BAC      Hemos terminado.
33    BCWS < BAC && BAC = BCWP  Hemos terminado y antes de lo esperado.
34
35    BCWP > BCWS Tarea Adelantada
36    BCWP < BCWS Tarea atrasada
37    BCWP > ACWP El proyecto esta por DEBAJO de lo presupuestado
38          GASTA MENOS de lo planeado
39    BCWP < ACWP El proyecto esta por ENCIMA de lo presupuestado
40          GASTA MAS de lo planeado
41
42    Si CPI > 1 && SPI < 1 necesitamos contratar a más gente
43    Si CPI < 1 puede que estemos haciendo trabajo no planificado, o que hayamos estimado mal
```



Ejercicios Repaso

Cuestiones rápidas

1. Un plan de proyecto ...
 - a. Proporciona visibilidad al progreso del proyecto
 - b. Es una representación abstracta de un proceso que proporciona información parcial sobre el proceso software
 - c. Es un conjunto de actividades que tienen como objetivo el desarrollo de un producto software
 - d. Todas las afirmaciones anteriores son falsas

2. El modelo basado en prototipos ...
 - a. Reduce las horas de implementación.
 - b. Permite reducir riesgos y costes de desarrollo
 - c. El sistema resultante es un prototipo
 - d. Es posible utilizarlo sólo con lenguajes orientados a objetos

3. La gestión de proyectos ...
 - a. Tiene como objetivo determinar qué hay que hacer, cómo hay que hacerlo y quién tiene que hacerlo.
 - b. Está fuertemente condicionada por la implementación del sistema
 - c. Está fuertemente condicionada el diseño del sistema
 - d. Tiene como objetivo el confeccionar una agenda del proyecto

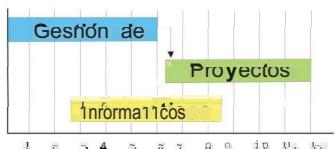
4. Tenemos tres actividades: A, B y C. A precede a B. Las holguras totales son: HT(A)= 4, HT(B)=2, HT(C)=3. Las holguras libres son: HL(A)= 3, HL(B)=2, HL(C)=3. Si A se retrasa 4 días, ¿qué les ocurre a B y C?
 - a. Nada, ya que no sobrepasamos la holgura total de A
 - b. La holgura total de B y C pasa a ser de 1 y 2 días respectivamente
 - c. B se convierte en crítica
 - d. Ninguna de las anteriores es cierta

5. Ms Project es ...
 - a. Una herramienta de seguimiento de proyectos
 - b. Una herramienta de modelado de proyectos
 - c. Una herramienta de estimación de costes de proyectos
 - d. Una herramienta de desarrollo de proyectos

6. El cono de la incertidumbre ...
 - a. Dice que conforme avanza el proyecto menos errores de estimación tenemos.
 - b. Dice que conforme avanza el proyecto más errores de estimación descubrimos.
 - c. No puedo aplicarlo sino hago el seguimiento del proyecto.
 - d. Todas las anteriores son falsas.

7. ¿Cuál de los siguientes elementos NO es un elemento de configuración (EC)?
 - a. Los ejecutables de la aplicación
 - b. La estructura de directorios.
 - c. La definición de las tablas de la BD de la aplicación
 - d. Todos los anteriores son elementos de configuración.

8. Make es una herramienta case que soporta ...
 - a. la planificación de una gestión de configuraciones
 - b. el control de cambios
 - c. la construcción del sistema
 - d. la gestión de versiones y releases



Ejercicios Repaso

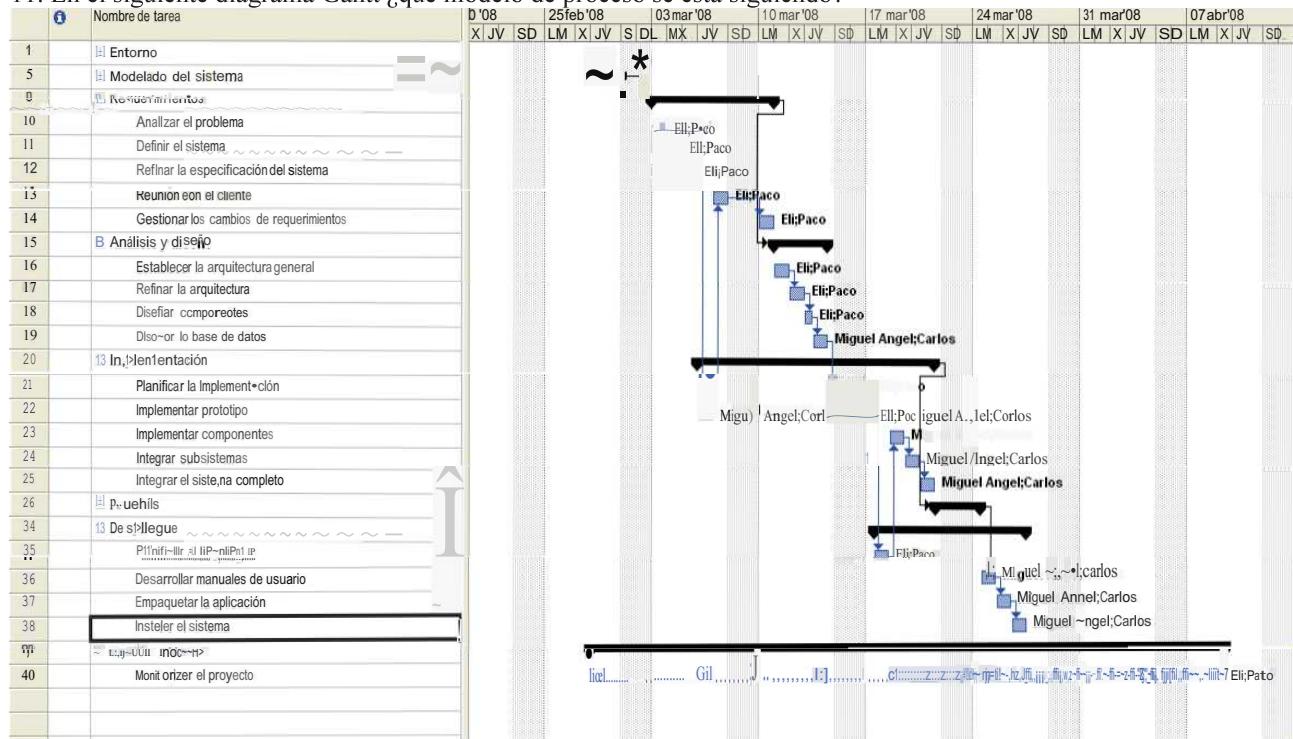
9. Las versiones de un sistema.

- a. Tienen un coste menor que las releases
- b. Suelen superar en número a las releases
- c. Incluyen documentación técnica
- d. Todas son ciertas.**

10. En el modelo UP.

- a. Hay una versión por iteración.
- b. Puede haber más de una versión por iteración.**
- c. Tenemos una release por fase.
- d. Todas las anteriores son ciertas.

11. En el siguiente diagrama Gantt ¿qué modelo de proceso se está siguiendo?



- a. Desarrollo Rápido de aplicaciones.
- b. Cascada
- c. Prototipado.
- d. Todos los anteriores.**

12. En la pirámide de jerarquía de necesidades donde se encuentran la mayoría de jefes de proyecto.

- a. No les afecta esta jerarquía, es sólo para empleados.
- b. Necesidades sociales.
- c. Necesidades seguridad.
- d. Necesidades autorrealización.**

13. ¿Cuál es la distribución de tiempo dentro de un equipo de desarrollo.

- a. 20% Trabajando sólo, 30% actividades no productivas, 50% Interactuando con otra gente.
- b. 30% Trabajando sólo, 20% actividades no productivas, 50% Interactuando con otra gente.**
- c. 30% Trabajando sólo, 20% actividades de gestión, 50% Interactuando con otra gente.
- d. 20% Implementando, 30% Interactuando con otra gente y 50% en actividades de gestión.

Tema 7.

Reglamento General de Protección de Datos



Gestión de Proyectos Informáticos: Grupo GPI 43
Sergio Suescun Ferrández
Andrii Shybaiev
Valiera Catalá Khudzhadze
Judith Vilella Cantos

Índice

- 1. Conceptos generales y Definiciones**
 - 1.1. Conceptos generales**
 - 1.2. Definiciones**
- 2. Participantes**
- 3. Derechos**
- 4. Procedimientos y documentos a emplear para el registro de datos.**
 - 4.1. Procedimientos para la toma de datos personales**
 - 4.2. Documentos requeridos por el RGPD**
- 5. Obligaciones y Sanciones**
 - 5.1. Obligaciones**
 - 5.2. Sanciones**
- 6. Referencias**

1. Conceptos generales y definiciones

1.1 Conceptos generales

El RGPD (Reglamento General de Protección de Datos) es un reglamento emitido por el parlamento y consejo europeo el 27 de abril de 2016. Este entró en vigor el 25 de mayo de 2018 y otorga a los ciudadanos seguridad y mayor control sobre sus datos de carácter personal.

Los datos de carácter personal son toda información sobre una persona física identificada o identifiable, ya sea un nombre, un DNI, datos de localización, un identificador en línea o uno o varios elementos propios de la identidad física, fisiológica, genética, psíquica, económica, cultural o social de dicha persona.

El Comité Europeo de Protección de Datos (CEPD) es un organismo independiente que garantiza la aplicación coherente de las normas de protección de datos en toda la Unión Europea. El CEPD se creó en virtud del Reglamento General de Protección de Datos (RGPD).

La AEPD también conocida como Agencia Española de Protección de Datos es la organización pública encargada de velar por el cumplimiento de la Ley Orgánica de Protección de Datos Personales y Garantía de los Derechos Digitales en España. El objetivo de la LOPDGDD es adaptar la legislación española a la normativa europea, definida por el Reglamento General de Protección de Datos.

La AEPD controla que los responsables de los ficheros (empresas, entidades públicas, etc) cumplan con el reglamento. Sus funciones son las siguientes:

1. Velar por la Ley Orgánica de Protección de Datos Personales y Garantía de los Derechos Digitales.
2. Sancionar el incumplimiento de la ya mencionada ley.
3. Requerir medidas de corrección.
4. Cesar el uso de tratamiento de datos en caso de ilegalidad.
5. Autorizar las transferencias internacionales de datos.
6. Recabar información sobre los responsables de ficheros cuanta necesite para el desempeño de sus funciones.
7. Atender a peticiones y reclamaciones.
8. Representación de España en los foros internacionales en la materia.
9. Control y observancia de lo dispuesto en la ley reguladora de la Función Estadística Pública.
10. Cooperación con organismos internacionales y órganos de las Comunidades Europeas en materia de protección de datos.

1.2 Definiciones

Para el entendimiento en profundidad de lo que es este reglamento, hay una serie de definiciones que debemos conocer con anterioridad. Estos conceptos principales son los siguientes:

- **Datos personales.** Toda la información con respecto al interesado (una persona física e identificable).
- **Tratamiento.** Cualquier operación o conjunto de operaciones realizada sobre los datos personales, sea dicha operación automatizada o no. Un ejemplo sería la extracción de la información o la utilización de los datos.
- **Responsable.** Persona física o jurídica, autoridad pública u otro organismo que determina cuáles son los fines y los medios del tratamiento de los datos.
- **Interesado.** Persona física la cual está siendo sometida al tratamiento de sus datos personales.
- **Encargado (del tratamiento).** La persona física o jurídica, autoridad pública, servicio u otro organismo que trate datos personales por cuenta del responsable del tratamiento.
- **Limitación del tratamiento.** El marcado de los datos de carácter personal conservados con el fin de limitar su tratamiento en el futuro.
- **Fichero.** Cualquier conjunto estructurado y accesible de datos personales.
- **Elaboración de perfiles.** Cualquier tratamiento automatizado de los datos personales que consista en utilizar datos personales para evaluar determinados aspectos personales de una persona física. Concretamente, para analizar o predecir aspectos relativos al rendimiento profesional, situación económica, salud, preferencias personales, intereses, fiabilidad, comportamiento, ubicación o movimientos de dicha persona física.
- **Destinatario.** La persona física o jurídica, autoridad pública, servicio u otro organismo al que se comuniquen datos personales, se trate o no de un tercero. Sin embargo, no se considerarán destinatarios las autoridades públicas que puedan recibir datos personales en el marco de una investigación concreta de conformidad con el Derecho de la Unión o de los Estados miembros. El tratamiento de tales datos por dichas autoridades públicas será conforme con las normas en materia de protección de datos aplicables a los fines del tratamiento.
- **Tercero.** Persona física o jurídica, autoridad pública, servicio u organismo distinto del interesado, del responsable del tratamiento, del encargado del tratamiento y de las personas autorizadas para tratar los datos personales bajo la autoridad directa del responsable o del encargado.

- **Representante.** Persona física o jurídica establecida en la Unión que, habiendo sido designada por escrito por el responsable o el encargado del tratamiento con arreglo al artículo 27, represente al responsable o al encargado en lo que respecta a sus respectivas obligaciones en virtud del presente Reglamento.
- **Personal autorizado.** Persona o personas autorizadas para el tratamiento de los datos personales del interesado, bajo la autoridad directa del responsable o encargado (con los compromisos que esto conlleva).
- **Delegado de protección de datos.** Figura obligatoria para determinadas empresas, el cual se encargará de informar y asesorar a los responsables y encargados.
- **Autoridad de control.** Autoridad pública independiente la cual supervisa la aplicación del RGPD.
- **Violación de la seguridad de los datos personales.** Toda violación de la seguridad que ocasione la destrucción, pérdida o alteración accidental o ilícita de datos personales transmitidos, conservados o tratados de otra forma, o la comunicación o acceso no autorizados a dichos datos.
- **Tipos de datos.**
 - **Datos genéticos.** Datos personales relativos a las características genéticas heredadas o adquiridas de una persona física que proporcionen una información única sobre la fisiología o la salud de esa persona, obtenidos en particular del análisis de una muestra biológica de tal persona.
 - **Datos biométricos.** Datos personales obtenidos a partir de un tratamiento técnico específico, relativos a las características físicas, fisiológicas o conductuales de una persona física que permitan o confirmen la identificación única de dicha persona, como imágenes faciales o datos dactiloscópicos
 - **Datos relativos a la salud.** Datos personales relacionados con la salud física o mental de una persona física, incluida la prestación de servicios de atención sanitaria, que revelen información sobre su estado de salud.
- **Normas corporativas vinculantes.** Las políticas de protección de datos personales asumidas por un responsable o encargado del tratamiento establecido en el territorio de un Estado miembro para transferencias o un conjunto de transferencias de datos personales a un responsable o encargado en uno o más países terceros, dentro de un grupo empresarial o una unión de empresas dedicadas a una actividad económica conjunta.
- **Objeción pertinente y motivada.** La objeción a una propuesta de decisión sobre la existencia o no de infracción del presente Reglamento, o sobre la conformidad con el presente Reglamento de acciones previstas en relación con el responsable o el encargado del tratamiento, que demuestre claramente la importancia de los riesgos que entraña el proyecto de decisión para los derechos y libertades fundamentales de los interesados y, en su caso, para la libre circulación de datos personales dentro de la Unión.

2. Participantes.

Existen dos participantes principales respecto al tratamiento de datos de carácter general, el responsable y el interesado.

El responsable de datos personales en el RGPD es una persona física o jurídica, autoridad pública u otro organismo que determine cuáles son los fines y los medios del tratamiento de los datos.

El interesado es cualquier persona física la cual está siendo sometida al tratamiento de sus datos personales.

El responsable debe seguir los principios generales de la protección de datos, estos son:

- **Lealtad con el interesado.** Los datos siempre deben recogerse a través de medios legítimos, no se deben usar engaños.
- **Transparencia.** Se debe respetar el derecho de información relativa al tratamiento de datos personales.
- **Licitud o legitimación del tratamiento.** Para considerar legítimo el tratamiento de los datos se debe contar con el consentimiento expreso del interesado.
- **Limitación de la finalidad.** Sólo se pueden tratar los datos con fines determinados, explícitos y legítimos.
- **Minimización.** Solo deben recogerse los datos a tratar, es decir, no se pueden recoger aquellos datos que no sean necesarios para el fin para el que se realiza el tratamiento.
- **Exactitud.** Los datos tendrán que ser exactos y corregidos en caso alguno de error.
- **Limitación del plazo de conversación.** Los datos personales no pueden conservarse o mantener más tiempo del estrictamente necesario para los fines del tratamiento.
- **Integridad y confidencialidad.** Se debe garantizar la seguridad de los datos, es decir, que se impida el acceso o uso no autorizado. Cualquier persona involucrada en el tratamiento de datos está sujeta a guardar la confidencialidad de los datos.

Se tendrán en cuenta ciertos aspectos a la hora de ofrecer el consentimiento del tratamiento de los datos de carácter personal:

- El responsable del tratamiento deberá obtener el consentimiento del interesado.
- A la hora de pedir el consentimiento del interesado para la cesión de sus datos, se le debe informar de forma clara y precisa la finalidad con la que se recabarán sus datos.
- Correspondrá al responsable del tratamiento la prueba de la existencia del consentimiento del interesado por cualquier medio de prueba admisible en derecho.

En el caso de que sean datos relativos a menores de edad:

- Sólo podrán proceder al tratamiento si el menor es mayor de catorce años de edad y se requerirá el consentimiento de los padres o tutores.
- No podrán recabarse datos del menor que permitan obtener información sobre los demás miembros del grupo familiar, o sobre las características del mismo. No obstante, podrán recabarse los datos de identidad y dirección del padre, madre o tutor con la única finalidad de recabar la autorización.
- Cuando el tratamiento se refiera a datos de menores de edad, la información dirigida a los mismos deberá expresarse en un lenguaje que sea fácilmente comprensible por aquéllos
- Correspondrá al responsable del fichero o tratamiento articular los procedimientos que garanticen que se ha comprobado de modo efectivo la edad del menor y la autenticidad del consentimiento prestado en su caso, por los padres, tutores o representantes legales

3. Derechos

Para ayudar a los interesados a asegurar la protección y la privacidad de sus datos personales, el RGPD otorga varios derechos a los interesados. Mediante estos derechos, los interesados pueden realizar una solicitud específica y tener la seguridad de que los datos personales no se utilizan de forma incorrecta para fines que no sean el objetivo legítimo del propósito para el que se proporcionaron originalmente.

- **Derecho de información.** Este derecho proporciona al interesado la capacidad de solicitar a una empresa información sobre qué datos personales (sobre él o ella) están siendo tratados y la justificación de dicho tratamiento. Por ejemplo, un consumidor puede preguntar por la lista de encargados con los que se comparten sus datos personales.
- **Derecho de acceso.** Este derecho proporciona al interesado la capacidad de obtener acceso a sus datos personales que estén siendo tratados. Esta solicitud proporciona el derecho del interesado a ver o consultar sus propios datos personales, así como de solicitar copias de sus datos personales.
- **Derecho de rectificación.** Este derecho proporciona al interesado la capacidad de solicitar modificaciones de sus datos personales en caso de que el interesado crea que estos datos personales no estén actualizados o no sean precisos.
- **Derecho a retirar el consentimiento.** Este derecho proporciona al interesado la capacidad de retirar un consentimiento otorgado previamente para tratar sus datos con una determinada finalidad. La solicitud luego precisa que la empresa detenga el tratamiento de los datos personales sobre los cuales se basaba el consentimiento proporcionado anteriormente.
- **Derecho de oposición.** Este derecho proporciona al interesado la capacidad de oponerse al tratamiento de sus datos personales. Normalmente, esto sería lo mismo que el derecho a retirar el consentimiento, si el consentimiento se solicitó de forma adecuada y no se realiza otro tratamiento que no sea con fines legítimos. Sin embargo, un escenario concreto sería cuando un cliente solicita que sus datos personales no sean tratados para determinados fines, mientras que una disputa legal esté en curso en los tribunales.
- **Derecho de oposición al tratamiento automático.** Este derecho proporciona al interesado la capacidad de oponerse a una decisión basada en el tratamiento automático. Utilizando este derecho, el cliente puede pedir su solicitud (por ejemplo, una solicitud de préstamo) que se revisará manualmente, ya que él o ella cree que el tratamiento automático de su préstamo no puede considerarse la situación única del cliente.

- **Derecho a ser olvidado.** También conocido como derecho a borrar, este derecho proporciona al interesado la capacidad de eliminar sus datos. Esto generalmente se aplica a situaciones donde la relación con un cliente ha finalizado. Es importante destacar que este no es un derecho absoluto, y que depende del programa de retención y el periodo de retención de acuerdo con otras leyes de aplicación.
- **Derecho a la portabilidad de datos.** Este derecho proporciona al interesado la capacidad de solicitar la transferencia de sus datos personales. Como parte de dicha solicitud, el interesado puede solicitar que sus datos personales sean devueltos (a él o ella) o transferidos a otro responsable. Al hacerlo, los datos personales deben ser proporcionados o transferidos en un formato electrónico de lectura mecánica.
- **Derecho a la limitación del tratamiento.** El derecho a la limitación del tratamiento es el derecho que tienen los interesados a que no se apliquen sus datos personales a las operaciones de tratamiento que en cada caso corresponderían. Como responsable del tratamiento, durante el tiempo en que dure la limitación usted, o su empresa, sólo podrá tratar los datos afectados, más allá de su conservación, con el consentimiento del interesado, para la formulación y el ejercicio o defensa de reclamaciones y para proteger los derechos de otra persona física o jurídica.

4. Procedimiento y documentos a emplear para el registro de datos.

La premisa sobre la que parte el RGPD es que la información privada de cada usuario debe ser utilizada de forma lícita y leal. A continuación se detallan los procedimientos que han de seguir las empresas a la hora de tomar datos de carácter personal.

4.1 Procedimientos para la toma de datos personales

Toda persona física ha de saber que su información personal está siendo recogida, utilizada, consultada y tratada. Además, debe ser consciente exactamente de cuál es la que está siendo empleada por la compañía.

En este sentido, el usuario debe dar su consentimiento explícito para ello. Ya no basta con indicar una casilla al final de un contrato o de un proceso de registro que afirme que se han leído los términos y condiciones de uso de un servicio. Es necesario preguntar específicamente al interesado si consiente que la empresa en cuestión haga uso de su información personal y hasta qué punto.

Gracias al RGPD, las preguntas acerca del tratamiento de la información personal otorgada por el usuario deben aparecer en un lenguaje claro y sencillo que pueda ser bien interpretado por cualquier persona. Usar oraciones excesivamente complejas con el propósito de confundir al interesado está totalmente prohibido y puede ser motivo de graves sanciones.

El RGPD especifica claramente que los fines del uso de la información personal de cualquier sujeto físico deben quedar claramente especificados en el momento del consentimiento del usuario.

Por su parte, la empresa que va a proceder al uso de los datos del usuario también debe presentarse adecuadamente. Esto quiere decir que ha de exponer claramente su identidad jurídica y sus intenciones respecto a ellos.

La empresa ha de informar de las normas, los riesgos y los derechos que le asisten a toda persona física que ceda su información personal. También debe especificarse cuál es la forma adecuada para hacer valer dichos derechos si el interesado cree que la información no está siendo tratada de la forma adecuada o pactada anteriormente.

La información personal de un usuario debe estar limitada al uso que la empresa va a hacer de ella en función de las tareas que realice la empresa. A esto hay que añadir, además, que la información no puede ser conservada de forma ilimitada, es decir, al cabo de un tiempo ha de ser eliminada tanto si el usuario lo solicita como si no.

4.2 Documentos requeridos por el RGPD

Para poder cumplir con la ley vigente a la hora de tratar con datos de carácter personal tendremos que elaborar los siguientes documentos:

- **Autorización para el uso de datos personales**, este documento permite que una persona física (el interesado) autorice a un tercero (cualquier persona física o jurídica) a utilizar sus datos personales. Esta autorización estará formada con los siguientes elementos:
 - Datos personales tanto del interesado como del responsable de los datos personales.
 - Los usos o finalidades (es decir, dónde y para qué se van a utilizar los datos personales).
 - La posibilidad de cesión a terceros de los datos personales.
- **Registro de actividades**, documento en el cual se mostrará los flujos de datos personales que ocurren dentro de la empresa u organización. Se estructurará de la siguiente manera:
 - Nombre y datos de contacto del responsable, y del delegado de protección de datos.
 - Los fines del tratamiento.
 - Una descripción de las categorías de interesados y de las categorías de datos personales.
 - Las categorías de destinatarios a quienes comunicas o comunicarás los datos, incluidos los destinatarios en terceros países u organizaciones internacionales.
 - Las transferencias de datos personales a un tercer país o una organización internacional, con mención del destinatario y, en el caso de que justifiques las transferencias en tus intereses legítimos imperiosos, la documentación de garantías adecuadas.
 - Los plazos previstos para la supresión de las diferentes categorías de datos.
 - Una descripción general de las medidas de seguridad que incluya.
- **Contrato de encargo de tratamiento de datos**, en este documento se vinculará al responsable y al encargado del tratamiento y regula las relaciones entre ambos respecto al tratamiento de los datos personales. En el RGPD se establece también la posibilidad de que exista un acto jurídico unilateral del responsable del tratamiento que regule esta relación y precise la posición del encargado del tratamiento. Como mínimo el documento tendrá que incluir los siguientes apartados:
 - Servicio que el encargado va a prestar, las instrucciones otorgadas por el responsable al encargado.
 - Deber de confidencialidad, tanto el encargado del tratamiento como todas aquellas personas que estén autorizadas a tratar los datos personales deben establecer su expreso compromiso a guardar secreto sobre esa información a la que tienen acceso.
 - Naturaleza, finalidad y duración del tratamiento, los motivos y usos del tratamiento de datos.

- Tipo de datos personales a los que se va a acceder, con qué datos se van a tratar.
 - Derechos y obligaciones del responsable.
- **Política de Protección de Datos Personales**, este es un documento de alto nivel para la gestión de privacidad en una empresa, que define lo que ésta quiere alcanzar y cómo.
 - Finalidad de la política: Esta parte de la política describe por qué se utiliza esta política, y por qué es importante para la empresa.
 - Definiciones de los términos principales: Esta parte de la política define los términos principales como datos personales, categorías especiales de datos, etc., en el contexto de la empresa.
 - Principios y finalidad del tratamiento: Esta parte de la política define los principios del tratamiento de datos personales, y las actividades para las que los datos personales pueden ser tratados. Por ejemplo, esto puede incluir la planificación de las actividades de la empresa con fines legítimos definidos en el RGPD.
 - Principales requisitos o controles: Esta parte de la política enumera los principales requisitos que se deben cumplir para considerar que se da cumplimiento a la política.
 - Funciones clave y sus responsabilidades: Esta parte de la política define las funciones / partes interesadas principales y sus responsabilidades para asegurar su cumplimiento con esta política. Esta sección además describe las responsabilidades de cada uno de los principales interesados. Es importante tener en cuenta que las responsabilidades de los empleados también deben de establecerse de manera explícita, de modo que los empleados se sientan parte de estas.
 - Nombramiento de la Autoridad Supervisora Principal: Esta parte de la política establece a quién se considera (desde la perspectiva de su empresa) la Autoridad Supervisora Principal.
- **Evaluación de Impacto de la Protección de Datos**, será obligatoria para tratamientos que conlleven un alto riesgo para los derechos y libertades de las personas. Esta herramienta se usa para evaluar un tratamiento de datos personales con el fin de identificar las medidas de seguridad necesarias y eliminar los riesgos asociados. Además, puede demostrar el correcto cumplimiento de las normativas de protección de datos. La EIPD está formada de los siguientes apartados:
 - Una descripción sistemática de la actividad o actividades de tratamiento previstas.
 - Una evaluación de la necesidad y proporcionalidad del tratamiento respecto a su finalidad.
 - Una evaluación de los riesgos.
 - Las medidas previstas para afrontar los riesgos, incluidas garantías, medidas de seguridad y mecanismos que garanticen la protección de datos personales.

5. Obligaciones y sanciones

5.1 Obligaciones

Los responsables del tratamiento de los datos personales tienen una serie de obligaciones para asegurar el correcto uso y la seguridad de los datos. Las principales obligaciones de los responsables son:

- Articular procedimientos que permitan fácilmente que los interesados puedan acreditar que han ejercido sus derechos por medios electrónicos.
- Cuando el interesado ejerza su derecho a rectificación el responsable deberá informar al interesado sobre las actuaciones derivadas de su petición en el plazo de un mes.
- Cuando el interesado ejerza su derecho a acceso los responsables deberán tomar medidas para verificar la identidad de quienes soliciten acceso a datos.
- Los responsables deben adoptar medidas que garanticen que solo se traten los datos necesarios en lo relativo a la cantidad de datos tratados, la extensión del tratamiento, los períodos de conservación y la accesibilidad a los datos.
- Desde el inicio, los responsables deben tomar medidas organizativas y técnicas para integrar en los tratamientos garantías que permitan aplicar de forma efectiva los principios del RGPD.
- Todos los responsables deberán realizar un análisis del riesgo de los tratamientos que realicen, a fin de poder establecer qué medidas deben aplicar y cómo deben hacerlo. El tipo de análisis variará en función de:
 - los tipos de tratamiento,
 - la naturaleza de los datos,
 - el número de interesados afectados,
 - la cantidad y variedad de tratamientos que una misma organización lleve a cabo.
- Los responsables de tratamiento deberán realizar una Evaluación de Impacto sobre la Protección de Datos (EIPD) con carácter previo a la puesta en marcha de aquellos tratamientos que sea probable que conlleven un alto riesgo para los derechos y libertades de los interesados.
- El RGPD establece que el responsable del tratamiento está obligado a bloquear los datos en los casos en que se deba proceder a su rectificación o supresión.
- Cuando se produzca una violación de la seguridad de los datos, el responsable debe notificarla a la autoridad de protección de datos competente, a menos que sea improbable que la violación suponga un riesgo para los derechos y libertades de los afectados.

- Los responsables deben documentar todas las violaciones de seguridad.
- El RGPD requiere que los responsables hagan esfuerzos razonables, teniendo en cuenta la tecnología disponible, para verificar que, para los niños menores de la edad que se fije como límite, el consentimiento se ha dado o se ha autorizado por los padres o tutores del menor.
- El RGPD establece la figura del Delegado de Protección de Datos (DPD), que será obligatorio en:
 - Autoridades y organismos públicos
 - Responsables o encargados que tengan entre sus actividades principales las operaciones de tratamiento que requieran una observación habitual y sistemática de interesados a gran escala.
 - Responsables o encargados que tengan entre sus actividades principales el tratamiento a gran escala de datos sensibles.

5.2 Sanciones

Las sanciones varían con respecto al tipo de infracción. Las infracciones pueden ser de tres tipos:

- Leves (multas de hasta 40.000 €):
 - Incumplimiento del principio de transparencia de la información o el derecho de información del afectado por no facilitar toda la información que exigen.
 - No atender los derechos de acceso, rectificación, supresión, limitación del tratamiento o portabilidad de los datos cuando no se requiera la identificación del afectado.
 - Incumplimiento de las obligaciones de los responsables y encargados del tratamiento.
 - Que el registro de actividades de tratamiento no contenga toda la información que exige el RGPD.
 - No proceder a borrar los datos de una persona fallecida.
 - Informar tarde o de forma incompleta de una brecha de seguridad.
- Graves (multas de 40.000 € hasta 300.000 €).
 - Brechas de seguridad ocurridas por no haber adoptado las medidas adecuadas de seguridad.
 - Trata de datos de menores de edad (sin el consentimiento de los padres o del menor, cuando pueda darlo).
 - No acreditar esfuerzos razonables para verificar la validez del consentimiento del menor o de sus padres o tutores.
 - No atender de forma reiterada u obstaculizar la solicitud de los derechos de acceso, rectificación, supresión, limitación del tratamiento o portabilidad de los datos en los tratamientos en los que no se requiere la identificación del afectado.
 - No adoptar las medidas técnicas y organizativas necesarias para garantizar la seguridad de los datos.
 - El encargo del tratamiento de los datos a un tercero sin el correspondiente contrato.
 - No disponer del registro de actividades.
 - No permitir que el delegado de protección de datos cumpla con sus funciones.
 - No atender las solicitudes de las agencias de protección de datos.
 - No informar al afectado de una violación de seguridad de datos personales.
 - Llevar a cabo el tratamiento de datos sin realizar una evaluación de impacto cuando esta es exigible.
 - No designar a un delegado de protección de datos cuando sea obligatorio.

- Muy graves (multas de 300.000 € hasta 20.000.000 €).
 - Vulnerar el deber de confidencialidad.
 - Tratamiento de datos personales que vulneren las garantías y principios establecidos en el RGPD.
 - Incumplir con la obligación del bloqueo de datos.
 - Transferencias internacionales de datos personales sin las debidas garantías.
 - Utilizar los datos personales recogidos con una finalidad diferente para la que se dio el consentimiento.
 - La omisión del deber de informar al afectado sobre el tratamiento de sus datos personales.
 - No facilitar el acceso a los datos personales, información, locales, equipos y medios de tratamiento cuando así sea requerido durante una investigación.
 - Obstruir en una inspección.

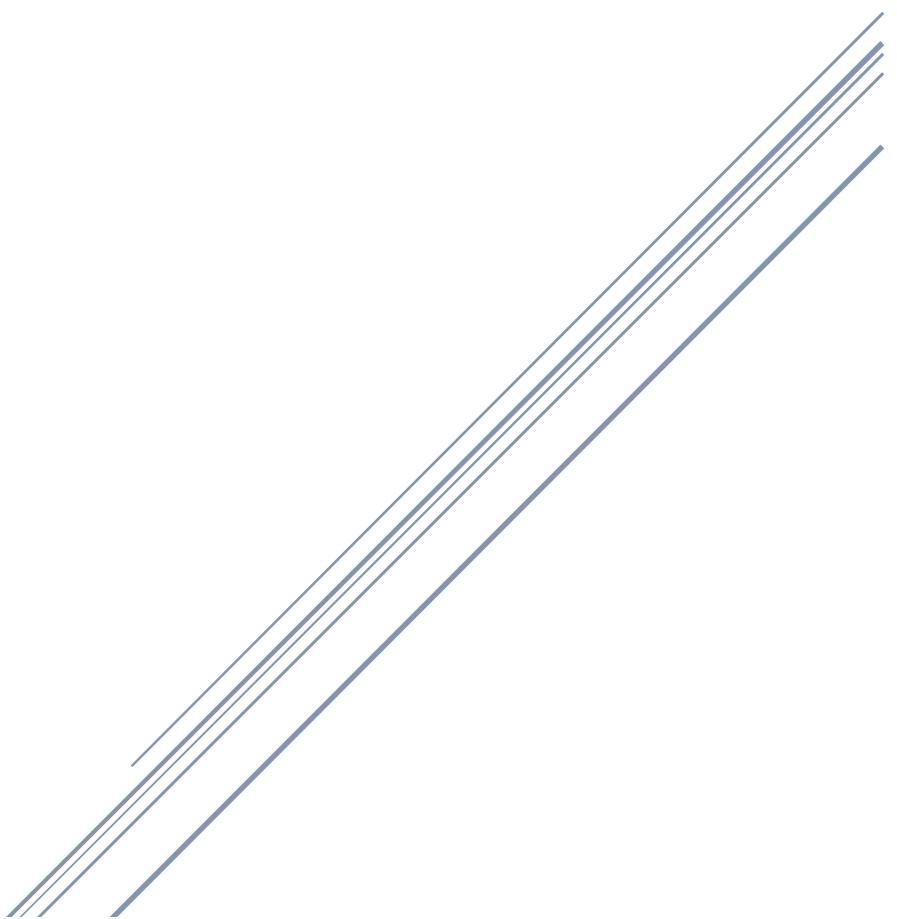
La cuantía de estas sanciones se valora según los derechos personales afectados, los beneficios obtenidos, reincidencia, intencionalidad y cualquier indicio que nos lleve a determinar la culpabilidad. En España, estas sanciones son impuestas por la LOPDGDD.

6. Referencias

- Documentos requeridos por el RGPD de la UE (<https://advisera.com/eugdpracademy/es/knowledgebase/lista-de-documentos-obligatorios-requeridos-por-el-rgpd-de-la-ue/>), visitado por última vez el 29/04/2021.
- Documents you need to comply the gdpr (<https://www.itgovernance.co.uk/blog/the-documents-you-need-to-comply-with-the-gdpr>), visitado por última vez el 29/04/2021.
- Guía RGPD para responsables de tratamiento (<https://www.aepd.es/sites/default/files/2019-09/guia-rgpd-para-responsables-de-tratamiento.pdf>), visitado por última vez el 02/05/2021.
- Reglamento general de protección de datos (RGPD) (<http://www.dpoitlaw.com/reglamento-general-de-proteccion-de-datos-rgpd/>), visitado por última vez el 02/05/2021.
- Reglamento del parlamento europeo y del consejo (RGPD) (<https://www.boe.es/doue/2016/119/L00001-00088.pdf>), visitado por última vez el 03/05/2021.
- Infracciones y sanciones LOPDGDD y RGPD (<https://proteccióndatos-lopd.com/empresas/infracciones-sanciones-lopdgdd-rgpd/>), visitado por última vez el 04/05/2021.
- Biggest GDPR fines 2020 (<https://www.tessian.com/blog/biggest-gdpr-fines-2020/>), visitado por última vez el 04/05/2021.

TEMA 8

Ley de propiedad intelectual



Francisco José Villena Charcos

Alejandro Palomares

Ricardo Moncayo

Jorge Vázquez López

David Molina

Grupo X15-D

Índice

1.- Registro de programas.	3
1.1.- Propiedad Intelectual	3
1.2.- Ley de propiedad intelectual	3
1.3.- Infringir los derechos sobre Propiedad Intelectual	3
1.3.- Derechos y figuras involucradas.	4
1.4.- Por qué proteger la propiedad intelectual	5
1.5.- Forma de conseguir la protección de la propiedad intelectual	6
1.6.- Traspaso de la propiedad intelectual	6
2.- Uso de recursos (imágenes, música, etc).	6
2.1.- Derechos de explotación.	7
2.2.1.- Reproducción.	7
2.2.2.- Distribución.	7
2.2.3.- Comunicación pública.	8
2.2.4.- Transformación.	8
2.2.- Excepciones o límites a los derechos de autor	9
2.2.1.- Copia privada	9
2.2.2.- Préstamo	9
2.2.3.- Parodia	9
2.2.4.- Libertad de panorama	10
2.2.5.- Citación	10
2.2.6.- Ilustración con finalidad docente o de investigación	10
2.3.- Dónde buscar recursos de terceros	11
2.3.1.- Imágenes	11
2.3.2.- Música	11
2.3.3.- Software	11
2.3.4.- Otros Recursos	12
3.- Software libre (freeware, shareware, copyleft, etc).	13
3.1.- Definición de Software Libre	13
3.2.- Ventajas y desventajas del Software Libre	14
3.3.- Libertades del software	16
3.4.- Tipos de Licencias de Software Libre	17

3.4.1.- Licencias GPL	17
3.4.2.- Licencias AGPL	17
3.4.3.- Licencias BSD	17
3.4.4.- Licencias Apache	17
3.4.5.- Licencias Creative Commons	17
4.- Software privativo.	18
4.1. Definición	18
4.1.1.- Características	18
4.2. Ventajas e inconvenientes del software propietario	18
4.3. Algunos ejemplos de software privativo	19
4.4. Tipos de licencias de software privativo	19
4.5. Acuerdo de Licencia con el Usuario Final (EULA)	20
5.- Nuevas formas de uso (licencias del software).	21
5.1. Modelos de negocio en internet	21
5.2. Plataformas en las que publicar un proyecto	23
5.2.1. Publicar software libre mediante las licencias GNU	23
6.- Bibliografía:	24

1.- Registro de programas.

1.1.- Propiedad Intelectual

Si hablamos de la ley de propiedad intelectual debemos saber que es cualquier creación del intelecto: desde las obras de arte hasta las invenciones, los programas informáticos, las marcas y otros signos comerciales. Estas "ideas" están bajo diferentes derechos legales que protegen la autoría de su creador. Su objetivo es estimular la creatividad en beneficio de todos al garantizar que las ventajas derivadas de la explotación de una creación beneficien al creador.

Hoy en día la propiedad intelectual es un derecho poco respetado en cuanto al software informático se refiere debido en gran parte a la piratería y al poco control que se puede ejercer ante esto.

1.2.- Ley de propiedad intelectual

La propiedad intelectual en España es respaldada por el Real Decreto Legislativo 1/1996, de 12 abril, que aprueba el texto refundido de la Ley de Propiedad Intelectual. Esta abarca diferentes puntos como: derechos de autor, sujeto, objeto y contenido de los derechos de autor, derechos de explotación y otros derechos de los creadores y autores.

La ley de propiedad intelectual es necesaria pues protege los derechos a través de distintos mecanismos: marcas, secretos comerciales, patentes y derechos de autor. Los recursos más utilizados para la protección del software por su utilidad jurídica son los derechos de autor y las patentes industriales y sus semejantes. Estos mecanismos tienen limitaciones (temporales, de ámbito, etc.) y su uso depende de los intereses de los autores.

Si no se aplicase la ley la propiedad intelectual, los esfuerzos económicos y humanos estarían desprotegidos del plagio y demás prácticas. Estos derechos salvaguardan los intereses creativos de los autores y sus obras, tanto de su literalidad como de sus ideas e ideas subyacentes.

1.3.- Infringir los derechos sobre Propiedad Intelectual

La protección de los derechos de autor está recogida en la constitución española por lo que estos derechos están protegidos por una perspectiva del derecho penal. Uno de los delitos más claros es el intento de plagio, reproducción o comunicación pública de una obra. Este delito es castigado de seis meses a 4 años de cárcel.

A continuación pondremos varios artículos donde podemos ver su protección:

- Artículo 138, acciones y medidas cautelares urgentes. Esto comprende:
 - El titular de los derechos reconocidos en esta ley, sin perjuicio de otras acciones que le correspondan, podrá instar el cese de la actividad ilícita del infractor y exigir la indemnización de los daños materiales y morales causados, en los términos previstos en los artículos 139 y 140. También podrá instar la publicación o difusión, total o parcial, de la resolución judicial o arbitral en medios de comunicación a costa del infractor.

- Artículo 139, cese de actividad ilícita. Esto podrá comprender:
 - La suspensión de la explotación o actividad infractora, incluyendo todos aquellos actos o actividades a los que se refieren los artículos 196 y 198.
 - La prohibición al infractor de reanudar la explotación o actividad infractora.
 - La retirada del comercio de los ejemplares ilícitos y su destrucción, incluyendo aquellos en los que haya sido suprimida o alterada sin autorización.
 - La retirada de los circuitos comerciales, la inutilización, y, en caso necesario, la destrucción de los moldes, planchas, matrices, negativos y demás elementos materiales, equipos o instrumentos destinados principalmente a la reproducción.
 - La remoción o el precinto de los aparatos utilizados en la comunicación pública no autorizada de obras o prestaciones.
 - La suspensión de los servicios prestados por intermediarios a terceros que se valgan de ellos para infringir derechos de propiedad intelectual.

- Artículo 140, Indemnización. Esto comprende:
 - La indemnización por daños y perjuicios debida al titular del derecho infringido comprenderá no sólo el valor de la pérdida que haya sufrido, sino también el de la ganancia que haya dejado de obtener a causa de la violación de su derecho.
 - La indemnización por daños y perjuicios se fijará, a elección del perjudicado.
 - La acción para reclamar los daños y perjuicios a que se refiere este artículo prescribirá a los cinco años desde que el legitimado pudo ejercitara.

1.3.- Derechos y figuras involucradas.

Los derechos de la propiedad intelectual dependen de lo que se quiera defender. Entre estos derechos encontramos los siguiente:

- a) **Secretos comerciales:** Es información comercial confidencial que otorga una ventaja competitiva a una empresa. La apropiación indebida, la divulgación o el uso no autorizado de dicha información se considera una práctica desleal y una violación del secreto comercial.

La protección de los secretos comerciales continúa de manera indefinida siempre que el secreto no se revele al público.
- b) **Marcas:** Son iconos que permiten diferenciar los productos o servicios de una empresa de las demás. Las marcas pueden consistir en palabras, letras, números, dibujos, fotos, formas, colores, logotipos, etiquetas o combinaciones de estos elementos que se empleen para diferenciar el origen de los productos o servicios.

- c) **Patentes:** La patente es un derecho exclusivo otorgado por el Estado para una invención que es nueva, implica una actividad inventiva y es susceptible de aplicación industrial. La patente otorga a su titular el derecho exclusivo de impedir que otros fabriquen,

utilicen, ofrezcan para la venta, vendan o importen la invención patentada sin la autorización del titular.

La patente es concedida por la oficina nacional o regional de patentes, y es válida durante un período de tiempo limitado, que suele ser de 20 años a partir de la fecha de presentación.

- d) **Derechos de autor:** El derecho de autor otorga a los autores, artistas y otros creadores protección jurídica para sus creaciones literarias y artísticas.

El derecho de autor tiene un plazo de protección general de 70 años contados desde el fallecimiento del creador o autor.

En cuanto a las figuras involucradas podemos encontrar al autor la cual es la persona natural que crea alguna obra literaria, artística o científica. La condición de autor tiene un carácter irrenunciable; no puede transmitirse "inter vivos" ni "mortis causa", no se extingue con el transcurso del tiempo así como tampoco entra en el dominio público ni es susceptible de prescripción.

Otros sujetos involucrados son:

- **Artistas intérpretes o ejecutantes.** Se entiende por tal a la persona que represente, cante, lea, recite o intérprete en cualquier forma una obra. A esta figura se asimila la de director de escena y de orquesta.
- **Productores de fonogramas.** Persona natural o jurídica bajo cuya iniciativa y responsabilidad se realiza por primera vez la fijación exclusivamente sonora de la ejecución de una obra o de otros sonidos.
- **Productores de grabaciones audiovisuales.** Persona natural o jurídica que tiene la iniciativa y asume la responsabilidad de la grabación audiovisual.
- **Entidades de radiodifusión.** Personas jurídicas bajo cuya responsabilidad organizativa y económica se difunden emisiones o transmisiones.
- **Creadores de meras fotografías.** Persona que realice una fotografía u otra reproducción obtenida por procedimiento análogo a aquélla, cuando ni una ni otra tengan el carácter de obras protegidas en el Libro I de la Ley de Propiedad Intelectual.
- **Protección de determinadas producciones editoriales.** Hace referencia a las obras inéditas en dominio público y a determinadas obras no protegidas por las disposiciones del Libro I del TRLPI.

1.4.- Por qué proteger la propiedad intelectual

La protección de los derechos de **Propiedad Intelectual**, tiende a ser considerada por la sociedad en general una carga, algo de carácter secundario y no algo que verdaderamente genera valor, cuando en realidad es una de las herramientas fundamentales de innovación y crecimiento para el negocio. Es más, se reconoce como el activo más importante que poseen muchas de las empresas más grandes y poderosas, algo que debería ser considerado la clave de su dominio y de su rentabilidad. En muchas ocasiones, es el objetivo preponderante en las fusiones y adquisiciones.

La propiedad intelectual no solo resalta la innovación o la historia personal y humana de las personas que han tenido curiosidad, determinación y perseverancia; también, y principalmente, protege a todo aquel que ha realizado un esfuerzo intelectual y la inversión económica necesaria para convertir sus ideas en algo tangible

1.5.- Forma de conseguir la protección de la propiedad intelectual

Para que la propiedad intelectual abarque tu trabajo y pueda defenderse en caso de copia, plagio, comunicación pública,... Debes llenar una serie de formularios y registrar tu obra. Para ello hay dos vías:

- Vía telemática:

Si se recurre por la vía telemática primero se debe tener un certificado digital, acceder a la página del ministerio de Cultura diciendo que comunidad autónoma está, llenar la solicitud y formularios y por último abonar una tasa firmando digitalmente la solicitud.

- Vía presencial:

Para la vía presencial hay que acudir a cualquier registro u oficina de la provincia donde se esté, cumplir con la solicitud y abonar la tasa.

1.6.- Traspaso de la propiedad intelectual

Los derechos de propiedad intelectual son transmisibles a excepción de los derechos morales del autor de una obra. La normativa sólo permite transmitir los derechos de propiedad intelectual de estas dos formas:

- Mortis causa: Consiste en la transferencia de la titularidad de la propiedad intelectual de una persona física a otra persona física o jurídica derivada de la muerte de aquella con el fin de la explotación.
- Inter Vivos: Se basa en la transmisión de la cesión de la titularidad de un derecho a un tercero, normalmente como consecuencia de un contrato expreso o una presunción legal.

2.- Uso de recursos (imágenes, música, etc).

Un recurso pertenece a su autor y es este el que tiene el uso exclusivo de la explotación de su obra en cualquier forma de la misma. El autor tiene los derechos de distribución, reproducción, transformación y comunicación pública, que no podrán ser realizados sin la autorización del mismo a excepción de los casos previstos por la ley de propiedad intelectual. Los derechos de explotación no son infinitos, cuando estos se acaban, la obra se vuelve al dominio público. A continuación hablaremos de los derechos de explotación.

2.1.- Derechos de explotación.

2.2.1.- Reproducción.

La reproducción es el concepto nuclear de los derechos de explotación pues es el primer paso para la mayoría de las modalidades de explotación. Su definición legal se ha adaptado al entorno digital dotando al concepto de la mayor amplitud con el fin de que el derecho permite al titular controlar todos los supuestos.

El derecho de Reproducción es de los más importantes de los derechos patrimoniales. El derecho de reproducción como uno de sus derechos exclusivos que la Ley de Propiedad Intelectual reconoce a los autores sobre la explotación de sus obras. Ello quiere decir que la obra de autor sólo puede ser explotada por el autor mismo o por aquellas personas que hayan recibido una autorización expresa y previa a la explotación que se desee realizar. El autor tiene derecho a decidir si la obra puede ser copiada o no, exceptuando algunos casos previstos.

La reproducción exige la preexistencia de la obra, y conlleva su fijación total o parcial de cualquier forma y en cualquier medio (material o inmaterial), desde su original ("directa") o desde una copia ("indirecta"), sea la fijación provisional o permanente, pero siempre que permita su reproducción y/o su comunicación pública.

Un ejemplo claro de reproducción sería la fotografía que se obtiene de una pintura o dibujo.

2.2.2.- Distribución.

Consiste en poner a disposición del público, es decir, al alcance de una pluralidad indeterminada de personas, el original o copias de la obra en un soporte tangible, con independencia de que los lleguen a adquirir o no. Tal puesta a disposición puede realizarse a través de distintos cauces jurídicos, ya sea mediante la venta, mediante alquiler o mediante un préstamo. A través de la distribución el autor u otro titular derivativo tiene la posibilidad de, controlar el destino de las fijaciones de su obra una vez que se han introducido en el comercio, estableciendo las formas o modalidades de su distribución. Un ejemplo claro sería las fotos que se reparten en un museo de alguna obra.

2.2.3.- Comunicación pública.

Se entiende por comunicación pública de las obras protegidas por el derecho de autor, cualquier acto que permita el acceso de estas obras a un conjunto de personas sin que sea necesario para ello la distribución de ejemplares a cada una ellas.

La Ley de Propiedad Intelectual no considera que existe una explotación de las obras en forma de comunicación pública de éstas, cuando la difusión se realice dentro de un ámbito estrictamente doméstico, que no esté integrado o conectado a una red de difusión.

La comunicación pública de las creaciones visuales puede realizarse de muy diferentes maneras:

- Mediante la proyección o exhibición pública de estas obras incorporadas en obras cinematográficas y otras obras audiovisuales.
- Mediante su emisión por cualquier medio de difusión inalámbrica.
- Mediante su difusión vía satélite.
- Mediante su transmisión al público por hilo, cable, fibra óptica u otro procedimiento análogo, sea o no mediante abono.
- Mediante su emisión, transmisión, o retransmisión por cualquier medio.
- Mediante su exposición pública, tanto del original como de sus reproducciones.
- Mediante la puesta a disposición al público de estas obras por procedimientos alámbricos o inalámbricos.
- Mediante el acceso público en cualquier forma de estas obras incorporadas en una base de datos.

Un ejemplo de distribución pública podría ser las publicaciones de fotos de obras de arte en internet.

2.2.4.- Transformación.

El derecho de transformación consiste básicamente en el derecho de autorizar o prohibir modificaciones en una obra (obra preexistente) de las que resulten una obra diferente (obra derivada). Es decir, la transformación implica la creación de una obra nueva a partir de otra obra, que se llama original, para diferenciarla de la nueva, que se llama derivada, y que genera unos nuevos derechos para el autor de esta última. Así, son transformaciones la traducción de un libro, la adaptación cinematográfica de una novela, por ejemplo, y requieren de la autorización del autor de la obra preexistente al de la obra derivada

Los derechos de propiedad intelectual sobre el resultado de la transformación de la obra original corresponderá al autor de esta última, sin perjuicio del derecho del autor de la obra preexistente de autorizar, durante todo el plazo de protección de sus derechos sobre ésta, la explotación de esos resultados en cualquier forma y especialmente mediante su reproducción, distribución, comunicación pública o nueva transformación.

Una vez la obra es creada, hay un periodo de tiempo en el cual los derechos de la propiedad intelectual pertenecen al autor o a quien este se los haya cedido. Después de un tiempo, la obra pasa a ser de dominio público. Dependiendo del país donde se esté, el periodo de tiempo cambia.

2.2.- Excepciones o límites a los derechos de autor

Como ya hemos dicho anteriormente los derechos de autor tienen límites y encontramos excepciones de las cuales se pueden hacer uso de obras sin tener que pedir permiso al autor o propietario de la misma. Entre los cuales queremos destacar la copia privada, el préstamo, la parodia, la libertad de panorama y, especialmente remarcables en el ámbito educativo, la citación y la ilustración con finalidad docente o de investigación científica.

2.2.1.- Copia privada

La copia privada es una limitación al derecho exclusivo que la ley concede al autor y al propietario de contenidos a hacer copias de ellos, que permite a una persona realizar la copia de una obra para uso privado sin ánimo de lucro no profesional ni empresarial, sin finalidades directas ni indirectamente comerciales, siempre que haya tenido acceso legítimo al original. Como ejemplo de esto podemos decir que podemos hacer una grabación de una película o un documental, podemos copiar un CD musical al disco duro, etc. pero, de estas copias sólo se puede hacer un uso privado, no se pueden reproducir en público, ni siquiera con finalidad educativa.

2.2.2.- Préstamo

No necesitarán de la autorización de los titulares de derechos por los préstamos de obras que realicen los siguientes establecimientos: museos, archivos, bibliotecas, hemerotecas, fonotecas o filmotecas de titularidad pública o que pertenezcan a entidades de interés general de carácter cultural, científico o educativo sin ánimo de lucro, o a instituciones docentes integradas en el sistema educativo español. Por ejemplo podemos decir los préstamos de libros de las bibliotecas. En algunos casos los titulares de los establecimientos tienen que remunerar a los autores.

2.2.3.- Parodia

No es necesario el consentimiento del autor de una obra ya divulgada para parodiarla cuando la parodia:

- No implique riesgo de confusión con la obra original.
- No infiera un daño a la obra original.
- No cause daño al autor de la obra original.

Es importante destacar que estas excepciones no podrán interpretarse de manera tal que permitan su aplicación de forma que causen un perjuicio injustificado a los intereses legítimos del autor o que vayan en detrimento de la explotación normal de las obras a que se refieran

2.2.4.- Libertad de panorama

La libertad de panorama limita el derecho de los propietarios de las obras a emprender medidas legales por violación de derechos contra el fotógrafo o cualquier persona que distribuya la imagen resultante. Es una excepción a la regla general que el propietario tiene el derecho exclusivo para autorizar la creación y distribución de trabajos derivados.

El artículo de la ley de propiedad intelectual 35.2 dice que “Las obras situadas permanentemente en parques, calles, plazas u otras vías públicas pueden ser reproducidas, distribuidas y comunicadas libremente por medio de pinturas, dibujos, fotografías y procedimientos audiovisuales».

En algunas leyes europeas no existe este límite en la propiedad intelectual y, por lo tanto, no se pueden divulgar sin permiso del autor fotografías de paisajes donde aparezcan obras.

2.2.5.- Citación

La inclusión en una obra propia de fragmentos de obras ajenas no necesita la autorización del autor de la obra citada, siempre que se cumplan todas las condiciones, que:

- El fragmento que se incluya corresponde a una obra ya divulgada.
- Su inclusión se realice a título de cita o para su análisis, comentario o juicio crítico.
- Se realice con fines docentes o de investigación.
- Se indiquen la fuente y el nombre del autor de la obra utilizada.

Las recopilaciones periódicas efectuadas en forma de reseñas o revistas de prensa tienen la consideración de citas.

2.2.6.- Ilustración con finalidad docente o de investigación

En la ley de propiedad intelectual el artículo 32.3 dice que “El profesorado de la educación reglada impartida en centros educativos, no necesitarán autorización por el autor o editor para realizar actas de reproducción, distribución y comunicación pública de fragmentos de obras, cuando se cumplan las siguientes reglas:

- Que estas acciones se hagan para la ilustración de sus actividades.
- Que se trate de obras ya divulgadas.
- Que las obras no tengan condición de libro de texto a no ser que se trate de actas de reproducción para la comunicación pública o actas de distribución de copias para el personal investigador.
- Que incluya el nombre del autor y fuente, a no ser que sea imposible.”

2.3.- Dónde buscar recursos de terceros

La mayoría de recursos de terceros los podemos encontrar en Internet, y estos pueden estar protegidos por derechos de autor o licencias.

A continuación hablaremos de recursos tales como imágenes, música y software y una mezcla de recursos a veces menos utilizados.

2.3.1.- Imágenes

La ley diferencia las Imágenes en grupos diferentes ya sean obras fotográficas o una fotografía como objeto individual.

Si hablamos de una obra fotográfica, se consideran la originalidad y personalidad del creador implícitamente. Una obra es protegida durante toda la vida del autor y hasta 70 años después de su muerte.

En el caso de una mera fotografía, esta solo está protegida durante 25 años desde que esta fue tomada.

Podemos encontrar webs con imágenes que no estén protegidas por Copyright, y que sean de uso libre, como por ejemplo: Flickr (usa CC) o Pixabay.

2.3.2.- Música

La música también está protegida por los derechos de autor y si queremos utilizar música protegida con Copyright lo más probable es que tengamos que pedir permiso al autor, lo que se suele traducir en la compra del derecho de uso.

La compra del derecho normalmente es para un solo uso a no ser que se especifique de alguna manera en el contrato.

Para buscar música de libre uso podemos utilizar los siguientes sitios: Youtube Music Library, Jamendo o Freesound.org.

2.3.3.- Software

Los programas software o librerías que queremos incluir en proyectos normalmente están protegidos por licencias. Si por ejemplo encontramos una licencia en un recurso que queremos introducir, podremos hacerlo sin problemas siempre y cuando nuestro proyecto no tenga fines comerciales y la licencia lo impida. Si es una licencia CopyLeft, estaremos también obligados a que nuestro proyecto tenga la misma licencia que el recurso que incluimos.

Los recursos software pueden ser programas en sí o el propio código fuente y si es este último, se pueden ver diferentes consideraciones, como por ejemplo, según qué licencia proteja el código, podremos modificarlo cuanto queramos o utilizar partes del código para crear algo completamente distinto a la función principal en la que se tenía pensado su uso, o que no se permita la modificación de este en ningún concepto.

Podemos encontrar diferentes librerías y recursos software en páginas como GitHub.

2.3.4.- Otros Recursos

Otros recursos que se pueden utilizar en proyectos son modelos 3D, texturas, materiales, estos se pueden encontrar en páginas como Sketchfab o Squid Pro, donde hay tanto una selección gratuita de estos recursos protegidos por licencias, o la opción de comprar los recursos para su uso en los proyectos que queramos.

3.- Software libre (freeware, shareware, copyleft, etc).

Antes de hablar sobre lo que es el software libre, debemos hablar primero sobre qué son los **derechos de autor del software** o el **software copyright**, el cual es la extensión de la ley de derechos de autor al campo del lenguaje máquina.

Los derechos de autor sobre el software son principalmente utilizados por los Desarrolladores de Software y las compañías propietarias del mismo para prevenir copias no autorizadas de su Software.

En España el software está protegido mediante la Ley de Propiedad Intelectual, que lo considera como la categoría programas de ordenador.

Según el Artículo 96 del Texto Refundido de la Ley de Propiedad Intelectual, se definen a los programas de ordenador como:

"toda secuencia de instrucciones o indicaciones destinadas a ser utilizadas, directa o indirectamente, en un sistema informático para realizar una función o una tarea o para obtener un resultado determinado, cualquiera que fuere su forma de expresión y fijación".

Por tanto, en España la ley **no protege al programa ejecutable en sí, sino al código fuente**. Además, la expresión programas de ordenador incluye la documentación técnica y manuales de uso de dicho programa.

El Artículo 97 considera autor al creador del programa o a la persona jurídica que sea contemplada como titular de los derechos de autor. En el caso de obras colectivas, al ente que la edite y divulgue bajo su nombre. Si el programa es fruto de la colaboración de varios autores, la propiedad intelectual corresponderá a todos en la proporción que acuerden.

La duración de los derechos de autor del software es de setenta años para personas jurídicas desde el momento de su creación. Para personas físicas durará hasta setenta años después de su muerte o declaración de fallecimiento. En el caso de obras colectivas, durará hasta setenta años después del fallecimiento del último coautor.

3.1.- Definición de Software Libre

Respecto a la definición de software libre, lo que llamamos "Software Libre" estipula los criterios que se tienen que cumplir para que un programa sea considerado libre. De vez en cuando modificamos esta definición para clarificarla o para resolver problemas sobre cuestiones delicadas.

«Software libre» es el software que respeta la libertad de los usuarios y la comunidad. A grandes rasgos, significa que los usuarios tienen la libertad de ejecutar, copiar, distribuir, estudiar, modificar y mejorar el software. Es decir, el «software libre» es una cuestión de libertad, no de precio. Para entender bien el concepto, piense en «libre» como en «libre expresión», no como en «barra libre». En inglés, a veces en lugar de «free software» decimos «libre software», empleando ese adjetivo francés o español, derivado de «libertad», para mostrar que no queremos decir que el software es gratuito.

No confundir con «Open source» (Código abierto) es algo distinto: su filosofía es diferente y está basada en otros valores. No se centran tanto en el hecho que los programas derivados mantengan las características, sino en fomentar la apertura del código que utilizan los programas para que todos puedan colaborar y beneficiarse.

Con el Software Libre promovemos estas libertades porque todos merecen tenerlas. Con estas libertades, los usuarios (tanto individualmente como en forma colectiva) controlan el programa y lo que este hace. Cuando los usuarios no controlan el programa, decimos que dicho programa «no es libre», o que es «privativo».

En conclusión, podemos decir que una buena definición de Software Libre es:

- Programa informático en el que el usuario tiene libertades sobre el mismo, pudiendo ser usado, copiado, modificado y distribuido como el usuario quiera.

3.2.- Ventajas y desventajas del Software Libre

Hay muchas ventajas del uso del software libre, destacando algunas bastante relevantes:

- Puedes descargar y probar cualquier software libre antes de incorporarlo al ecosistema de aplicaciones de tu empresa. Sin restricciones, probando bien todas sus características antes de tomar la decisión final de usarlo.
- Detrás de las aplicaciones de software libre hay una comunidad de desarrollo que se encarga de resolver las dudas y cuestiones técnicas, y de definir su evolución de forma totalmente gratuita.
- Seguridad muy por encima del software privativo: El hecho de que se tenga acceso al código fuente hace que muchos ojos revisen y prueben ese código y sobre todo sea difícil guardar secretos.
- Uso de estándares. Los desarrollos de software libre suelen ser muy fieles a los estándares, lo que significa que suelen comunicarse bien con otras aplicaciones que cumplan con esos mismos estándares.

Las desventajas del software libre son inferiores a sus ventajas, pero podemos destacar algunas que conviene tener en cuenta:

- Soporte técnico urgente: Salvo que se tenga contratado directamente con una empresa, el soporte técnico del software libre es muy bueno, pero lógicamente puede que no se ajuste a lo que se necesita en casos de emergencia.
- Conocimiento técnico: No es necesario ser un técnico para usar el software libre. Pero sí es conveniente que si vas a incorporarlo a tu empresa, tengas una persona que conozca ese mundo y te ayude a analizar y seleccionar las herramientas que necesitas, valorando bien la madurez de los desarrollos y la comunidad que hay detrás en cada caso.
- El sistema operativo de software libre no ha alcanzado las ventajas del privativo. su sistema operativo para escritorio basado en GNU/Linux, quizás el más maduro para un usuario no técnico sea Ubuntu.
- Usabilidad de las aplicaciones. Las aplicaciones suelen tener una usabilidad no muy trabajada en general, aunque con algunas excepciones. No es un problema importante dependiendo de la aplicación que sea.

3.3.- Libertades del software

Para que un programa sea de software libre, este debe contar con cuatro libertades:

Usar el programa con cualquier propósito:

La libertad de ejecutar el programa con cualquier propósito se trata de que todo usuario es libre de utilizar el software en cualquier sistema de computación, con cualquier tipo de trabajo y finalidad y sin obligación de comunicar a ninguna entidad específica o al programador.

Estudiar cómo funciona el programa y poder modificarlo:

Para este tipo de libertad es necesario tener acceso al código fuente, por lo que esta es una condición fundamental para el software libre.

Ahora bien, se constituye la libertad de usar la versión modificada en lugar de la original.

Distribuir copias del programa a cualquier persona u organización:

La libertad para distribuir el programa, consiste en que los usuarios tienen la libertad de distribuir y compartir copias con o sin modificaciones, sin necesidad de pedir ni pagar algún permiso para hacerlo.

Además, se puede distribuir los programas gratuitamente o cobrando alguna tarifa por ello.

Mejorar el programa y compartir las mejoras en beneficio de todos:

Esta libertad permite mejorar el programa y posibilita compartir y liberar las versiones modificadas como software libre de tal forma que se beneficia con ello a las personas u organizaciones que lo utilicen.

En conclusión, muchas empresas que han ganado mucho dinero haciendo software libre y utilizándolo pero para hacer el proyecto de gestión de autónomos y entregárselo personalmente al cliente hemos preferido que nuestro proyecto no sea software libre.

3.4.- Tipos de Licencias de Software Libre

3.4.1.- Licencias GPL

Con esta licencia el desarrollador conserva los derechos de autor, pero permite su libre distribución, modificación y uso siempre y cuando, en el caso de que el software se modifique, el nuevo software que se desarrolle como resultado quede obligatoriamente con la misma licencia.

3.4.2.- Licencias AGPL

Se engloba dentro de las licencias destinadas a modificar el derecho de autor derivadas de GNU. La novedad de AGPL es que, aparte de las cláusulas propias de una GPL, ésta obliga a que se distribuya el software que se destine a dar servicios a través de una red de ordenadores.

3.4.3.- Licencias BSD

El software bajo esta licencia es la menos restrictiva para los desarrolladores ya que el software puede ser vendido y no hay obligaciones de incluir el código fuente. Además, una aplicación licenciada con BSD permite que otras versiones pueden tener otros tipos de licencias, tanto libres como propietarias.

3.4.4.- Licencias Apache

El software bajo este tipo de licencia permite al usuario distribuirlo, modificarlo, y distribuir versiones modificadas de ese software pero debe conservar el copyright y el disclaimer. La licencia Apache no exige que las obras derivadas (las versiones modificadas) se distribuyan usando la misma licencia, sólo exige que se informe a los receptores que en la distribución se ha usado código con la licencia Apache.

3.4.5.- Licencias Creative Commons

Su definición se basa en cuatro condiciones:

- **Atribución:** con la cual se puede distribuir, exhibir, representar... siempre y cuando se reconozca y se cite a su autor
- **No comercial:** que no permite usar el software con fines comerciales
- **No derivadas:** con la cual no se puede modificar dicha obra
- **Compartir igual:** que incluye la creación de obras derivadas siempre que mantengan la licencia original.

Con estas licencias, el desarrollador tiene la posibilidad de elegir lo que considere más conveniente para su trabajo. Esta decisión implica la renuncia a algunos derechos reservados inherentes al derecho de autor, tales como el derecho de reproducción y el derecho de transformación que, por otra parte, puede contribuir a la divulgación de su trabajo y permitir el derecho de acceso por parte del público a sus obras.

4.- Software privativo.

4.1. Definición

Conocemos como software privativo, propietario o de código cerrado, a aquel que está limitado tanto en su uso y distribución como en su modificación. Las limitaciones a la hora de realizar estas acciones las suele marcar el desarrollador o la empresa (persona física o jurídica) que posee los derechos de autor. Generalmente el usuario sólo tiene derecho a ejecutar el programa bajo las condiciones fijadas por el proveedor. La Fundación para el Software Libre (FSF) considera que cualquier software no libre es software privativo.

El origen de este tipo de software se remonta a 1979 cuando el gobierno de los EEUU obliga a IBM a distinguir entre software y hardware, que hasta entonces no se hacía. Esto dió lugar a los primeros programas con el código cerrado, lo que ha evolucionado hasta llegar al software propietario.

En general el software privativo es más utilizado, pero la tendencia actual es el aumento de usuarios de software libre, con la principal ventaja del ahorro de dinero que supone.

Existen diferentes tipos de licencias, que suelen suponer condiciones de uso y distribución totalmente diferentes, dando lugar en los últimos años a una nueva forma de entender la informática, implicando la aparición de nuevos modelos de negocio que van más allá de las licencias de uso y los desarrollos a medida.

4.1.1.- Características

Generalmente este tipo de software presenta las siguientes características:

- No se pueden realizar modificaciones en el código fuente.
- No se puede distribuir sin permiso del propietario.
- Se puede localizar en tiendas.
- Está protegido por Copyright.
- Se ofrece soporte por parte del propietario.

4.2. Ventajas e inconvenientes del software propietario

Generalmente el software propietario es creado por grandes empresas que invierten una gran cantidad de recursos en su desarrollo, sometiendo su producto a más pruebas de validación antes de su salida al mercado haciendo que sean programas más estables desde su salida al mercado.

Una de las principales ventajas del software propietario es que los proveedores ofrecen soporte a los usuarios, asegurando su estabilidad y teniendo la posibilidad de dar apoyo en la resolución de los posibles problemas que puedan surgir. Esto hace que sea muy utilizado por organismos gubernamentales, universidades y empresas.

También supone una ventaja la facilidad de adquirir este tipo de productos, puesto que se puede encontrar en las tiendas y su instalación suele ser más sencilla. Por otra parte, las interfaces suelen estar mejor diseñadas y ser más sencillo su uso.

Una de las desventajas que presenta el software privativo es que el código fuente no es libre y no puede ser modificado por los usuarios, por tanto, son los propios usuarios los que se deben adaptar al software, dando menos posibilidades de personalización. Por otra parte ofrece menos

transparencia, y la posibilidad de que haya errores no detectados y que únicamente lo podrá hacer la empresa propietaria.

También presenta la desventaja de que el coste de los productos suele ser elevado, además de no siempre estar disponible para todas las plataformas. Otro de los problemas es que al ser un producto de una empresa cuyo único objetivo es el obtener beneficios económicos, el programa está sujeto a intereses económicos y cabe la posibilidad de que se deje de dar soporte al producto.

[**4.3. Algunos ejemplos de software privativo**](#)

El software privativo se presenta normalmente como software comercial, que no necesariamente tiene que ser privativo. Se puede encontrar software privativo en todos los ámbitos del mundo del software.

Por ejemplo hay muchos sistemas operativos operativos que son software propietario:

- Windows
- Chrome OS
- macOS

Algunos ejemplos de software de antivirus serían:

- Norton
- Kasperski o Panda

O algunas aplicaciones comerciales como:

- Google Drive
- Skype
- Microsoft Edge

[**4.4. Tipos de licencias de software privativo**](#)

A continuación podemos ver las diferentes licencias de software privativo que nos podemos encontrar:

- **Licencias OEM**

Las licencias OEM son las que da el fabricante del software al fabricante de los equipos para instalar el programa. Lo normal al adquirir un equipo es que ya venga instalado el sistema operativo con este tipo de licencia, aun así se puede adquirir por separado, quedando el software ligado a los componentes del equipo. Esto supone que en caso de cambiar alguno de los componentes principales, será necesario volver a adquirir el software. Este tipo de licencias suele ser por lo general el tipo más barato.

- **Licencias Retail**

Son el tipo de licencias que los usuarios, por lo general, pueden adquirir directamente de los proveedores del software o a través de tiendas. Este tipo de licencias permite vincular el software a una cuenta.

- **Licencias PKC**

Es un tipo de licencia perpetua, en la cual puedes desinstalar el software de un equipo y volver a instalarlo en otro. Hace años se adquirían las licencias físicas en formato CD, actualmente se recibe un código que permite activar el producto.

[**4.5. Acuerdo de Licencia con el Usuario Final \(EULA\)**](#)

El EULA (End-User License Agreement) es un contrato donde se indican las condiciones y limitaciones que se deben aceptar por parte del usuario para utilizar un software propietario.

Se trata de un contrato donde se comprometen y protegen el usuario y el fabricante delimitando los derechos y deberes del usuario sobre el producto adquirido. Suelen incluir cláusulas relacionadas con leyes de propiedad intelectual e industrial e información sobre la responsabilidad de la empresa.

Puede presentarse en un documento en papel en caso de que se adquiera de forma física. Actualmente, lo normal es que aparezca una ventana en el proceso de instalación que deba ser aceptada para finalizarlo. Generalmente es necesario que se acepten las condiciones para poder hacer uso del software final.

Lo más habitual es que no se preste atención al documento, puesto que suele emplear terminología muy técnica y en algunos casos su redacción es confusa.

5.- Nuevas formas de uso (licencias del software).

En los últimos años el abrumador avance tecnológico está obligando a las empresas a transformarse y adaptarse a los nuevos tiempos. Estas transformaciones afectan a los modelos de negocio, la forma en que las empresas crean valor, suponiendo cambios en la forma en que se ofrecen los productos y servicios.

5.1. Modelos de negocio en internet

A continuación se muestran los modelos de negocio más comunes y empleados en internet, así como los más recientes, para la creación de valor a partir de productos y servicios.

- **Suscripción**

En el modelo de suscripción, el cliente paga una cantidad por mantener, un tiempo determinado, acceso a productos o servicios. Presenta la ventaja de poder trabajar con una base de datos de clientes fija en un tiempo concreto y, por tanto, con una cantidad de ingresos también fijo para ese periodo de tiempo. Para que este modelo de negocio sea eficaz, es necesario mantener actualizado el producto, de lo contrario es posible que no se cumplan los objetivos económicos. Un ejemplo lo podríamos encontrar en plataformas como Netflix, o Spotify.

- **Micropagos**

Con micropago se entienden los sistemas empleados por los usuarios para realizar pequeños pagos. Este concepto nace del entorno de Internet, y está muy relacionado con el modelo de Freemium, es un modelo muy empleado en juegos y aplicaciones de smartphones. Según las fuentes se consideran micropagos transacciones por valor desde unos céntimos hasta unos 20€.

- **Pago por Consumo**

Se trata de un sistema de pago, surgido en el sector televisivo, en el que el usuario paga únicamente por lo que consume. Se podría calificar como una forma de micropagos. Por ejemplo, en los ciertos periódicos online, hay ciertas noticias que están disponibles para suscriptores, pero haciendo un pequeño pago puedes disfrutar, este sería un ejemplo de pago por consumo.

- **Freemium/Premium**

Este modelo de negocio consiste en ofrecer un producto de manera gratuita, mientras que para poder acceder al resto de contenidos es necesario realizar un pago. Suele estar compaginado con el modelo por publicidad insertada, que se verá a continuación además de con el modelo de suscripción. Este modelo se puede observar, por ejemplo, en una gran cantidad de juegos para smartphone o también en aplicaciones como Spotify donde se puede acceder a la música con ciertas limitaciones a la hora de cambiar de canción y teniendo anuncios, en caso de que no se quiera tener estas limitaciones es necesario realizar una suscripción.

- **Por publicidad insertada**

Se trata de uno de los modelos más frecuentes que es utilizado generalmente en aplicaciones móviles, periódicos, webs... Consiste en incluir avisos publicitarios en él.

- **Paga lo que quieras**

Se trata de un modelo en el que los clientes pagan lo que consideran por el producto, generalmente se propone un precio por parte de la empresa, pero son los usuarios los que deciden cuánto pagar. Se emplea generalmente como estrategia de marketing puesto que normalmente no es un modelo de negocio sostenible a largo plazo. Un ejemplo podría ser la editorial independiente con sede en Varsovia OpenBooks.com, que permite el acceso a una gran parte de sus productos y decidan después lo que desean pagar, de lo que el 70% corresponde al autor.

- **Bundled**

Se trata de un modelo que consiste en poner a la venta paquetes de productos del mismo tipo o no, a un precio menor que si fueran adquiridos por separado. Es un buen modelo para dar salida a productos que no se venden mucho. Es un modelo que se suele observar en productos software de grandes empresas, como Microsoft, con los paquetes de oficina, o en algunos juegos en los que se agrupan varios ediciones del mismo juego para venderlas conjuntamente en un paquete.

- **Crowdfunding**

El crowdfunding, también conocido como micromecenazgo, es un modelo de financiación de proyectos desarrollado sobre las nuevas tecnologías. Consiste en que una persona busca financiación para llevar a cabo un proyecto y lo publica, recibiendo pequeños préstamos de una multitud de personas a cambio de lo que ofrezca el creador o únicamente por simpatizar con la causa. Este modelo ha sufrido un incremento notable desde el año 2017. Existen diversas plataformas para crowdfunding como Hatch a Dream, Kickstarter o Ulule.

- **Venta directa**

Se trata de un modelo que actualmente está bastante normalizado y es común, pero que gracias a las nuevas tecnologías y a internet se le ha dado un nuevo significado permitiendo que cualquier empresa o persona que cree un producto pueda venderlo directamente a los clientes sin necesidad de intermediarios, como las tiendas.

5.2. Plataformas en las que publicar un proyecto

Actualmente existen multitud de plataformas en las que se puede publicar un proyecto software, a continuación se muestran algunas de las más destacadas.

- **Github**

Github es una plataforma de control de versiones y desarrollo colaborativo basado en Git. Normalmente el código de los proyectos alojados en esta plataforma se almacena de forma pública. Tiene como características que cada proyecto posee una wiki y una página web, existen gráficos para comprobar el flujo de trabajo y además incluye ciertas funcionalidades típicas de redes sociales, como, por ejemplo, la posibilidad de tener seguidores. Tiene la desventaja de que las versiones más avanzadas y flexibles son de pago.

- **SourceForge**

Se trata de un sitio web para la colaboración de proyectos software. Provee una amplia gama de servicios útiles para los procesos de desarrollo de software e integra un amplio número de aplicaciones de software libre. Sourceforge controla y gestiona varios proyectos de software libre y actúa como un repositorio de código fuente. Desde 2013, cuando pasa a ser comercializado por Dice Holdings, incluye en los instaladores software publicitario, con el objetivo de generar ingresos para los desarrolladores, lo que ha generado numerosas críticas.

5.2.1. Publicar software libre mediante las licencias GNU

Para publicar un proyecto con la licencia GNU, habría que realizar las siguientes acciones:

- **Conseguir una declaración de renuncia de copyright de su empleador o institución académica.**

Se trata de prevenir que posteriormente la empresa en la que trabajamos o centro en el que estudiamos no pueda alegar posteriormente que el copyright les pertenece y que nosotros no estamos autorizados a publicar el programa.

- **Incluir en cada archivo las notas de copyright apropiadas.**

El aviso de copyright deberá incluir el año en el que finalizó la preparación de la publicación, y el nombre de las personas que colaboraron escribiendo el código. Para el software con diversas publicaciones a lo largo de los años se puede emplear un rango de años. El símbolo de copyright (©) se puede incluir si se desea, pero no es necesario.

- **Añadir un archivo COPYING que contenga una copia de la GPL o AGPL de GNU. Añadir un archivo COPYING:LESSER que contenga una copia de la LGPL de GNU, en caso de utilizar esta licencia.**

- **Incluir una nota de licencia en cada archivo.**

A continuación de los avisos de copyright debe aparecer la declaración sobre el permiso de copia de cada archivo. Es necesario hacer esto para garantizar los derechos que les damos a los usuarios, para que el propósito del software libre no se vea frustrado.

- Hacer que el programa muestre al inicio una nota al respecto, de forma opcional.
- Hacer que el programa ofrezca copias del código fuente, en caso de utilizar la AGPL.

6.- Bibliografía:

- <http://www.rajyl.es/diccionario-juridico-cultura/voces/distribucion>
- <https://www.vegap.es/reproduccion>
- <https://www.vegap.es/comunicacion-publica#:~:text=De%20acuerdo%20con%20la%20Ley,cada%20una%20de%20estas%20personas>
- https://noticias.juridicas.com/base_datos/Admin/rdleg1-1996.l3t1.html
- https://www.supercontable.com/informacion/ley_gestion/Art. 139. R.D.Leg. 1-1996-Ley_de_Propiedad_Intelectual.html
- https://www.supercontable.com/informacion/ley_gestion/Art. 140. R.D.Leg. 1-1996-Ley_de_Propiedad_Intelectual.html#:~:text=Art.-,140.,la%20violaci%C3%B3n%20de%20su%20derecho.
- <https://www.culturaydeporte.gob.es/cultura/propiedadintelectual/la-propiedad-intelectual/sujetos.html>
- <http://www.a2estudiolegal.com/importancia-proteger-los-derechos-de-propiedad-industrial/>
- <https://www.cedro.org/sitefinity/status?ReturnUrl=https%3A%2F%2Fwww.cedro.org%2Fblog%2Farticulo%2Fblog.cedro.org%2F2018%2F04%2F05%2Fpasos-para-inscribir-una-obra>
- http://www.dosdoce.com/evolucion_nuevos_modelos_negocio_en_la_era_digital_v2.pdf
- <https://www.infoautonomos.com/ideas-de-negocio/modelos-de-negocio-en-internet/>
- <http://andreitamedina.blogspot.com/2012/04/ventajas-y-desventajas-del-software.html>
- <https://colaboratorio.net/glosario/cuatro-libertades-del-software-libre/>
- <https://www.obsbusiness.school/blog/ley-de-propiedad-intelectual-resumen-de-lomas-importante#:~:text=La%20Ley%20de%20Propiedad%20Intelectual%20recoge%20una%20serie%20de%20derechos,de%20Propiedad%20Intelectual%20o%20no.>
- <https://www.corima.mx/por-que-es-importante-la-propiedad-intelectual/>
- <https://www.legalbono.com/guias/cesion-de-derechos-propiedad-intelectual/>
- https://www.eoi.es/wiki/index.php/Los_derechos_de_explotaci%C3%B3n_en_Propiedad_intelectual
- <https://www.cedro.org/propiedad-intelectual/limites-y-excepciones#:~:text=Las%20excepciones%20son%20limitaciones%20al,por%20dichos%20actos%20de%20explotaci%C3%B3n.>

<https://rockcontent.com/es/blog/software-libre/#:~:text=La%20libertad%20de%20ejecutar%20el,entidad%20espec%C3%ADfica%20o%20al%20programador.>

<https://es.wikipedia.org/wiki/GitHub>

<https://es.wikipedia.org/wiki/SourceForge>

<https://www.gnu.org/licenses/gpl-howto.es.html>