
 DEPARTAMENTO DE TECNOLOGÍAS DE LA INFORMACIÓN	Principios y Fundamentos de la Ingeniería del Software	
	Examen convocatoria de Septiembre 2012 03/09/2012 Duración : 2 horas	2º Grado en Ingeniería Informática

Problema 1. (25% nota). Biblioteca

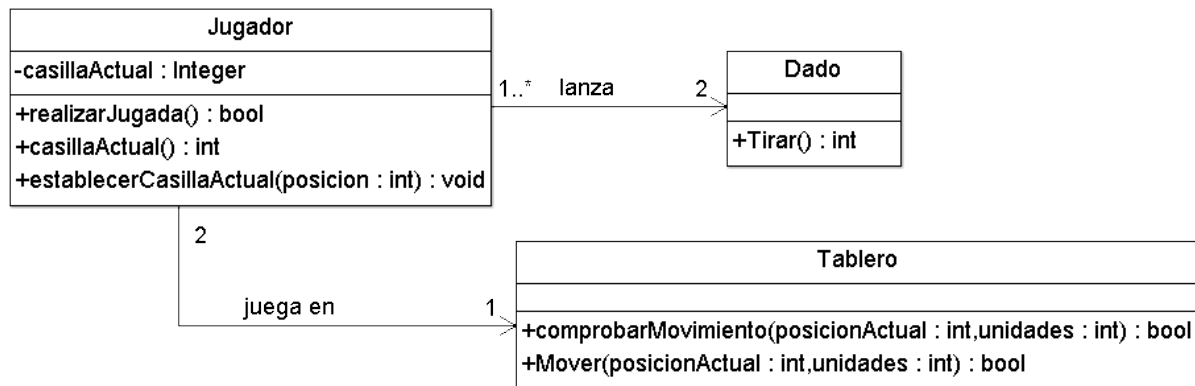
- Una biblioteca contiene un conjunto de copias de libros.
- Los libros se caracterizan por su nombre, tipo (novela, teatro, poesía, ensayo), editorial, año y autor/es.
- Los autores se caracterizan por su nombre, nacionalidad y fecha de nacimiento.
- Cada copia (puede haber más de una por libro) tiene un identificador, y puede encontrarse en uno de los siguientes estados: en biblioteca, prestada, con retraso o en reparación.
- En la biblioteca también se almacena información sobre los lectores, los cuales pueden tener un máximo de 3 libros en préstamo.
- Cada copia se presta un máximo de 30 días, por cada día de retraso, se impone una multa de dos días sin posibilidad de coger un nuevo libro, de manera que se debe poner almacenar la información de dichas multas.

Se pide:

- Realizar un **Diagrama de Clases** que nos permita representar toda la información indicada en el enunciado.
- Completarlo** con los **atributos** que el alumno considere necesarios para el **lector**.
- Añadir los métodos** necesarios para realizar el **préstamo** y **devolución** de libros, teniendo en cuenta que esta última acción puede conllevar una **multa**. Los métodos deben tener su firma lo más completa posible.

Problema 2 (25% nota). Parchís

Partiendo del siguiente Diagrama de Clases, **elegir** para realizar el apartado a) o b):



- a) Realizar el **Diagrama de Secuencia** de la operación **“realizarJugada”** definida en la clase Jugador, para el jugador nº1. Teniendo en cuenta que: **(10% nota).**
- La jugada consiste en lanzar los dos dados, comprobar si es posible desplazarse el número de casillas que resulta del lanzamiento, y en el caso de que sea así realizar dicho movimiento.
 - El método **Tirar()** devuelve el **número** resultante de lanzar el dado.
 - El método **casillaActual()** devuelve **en qué posición del tablero se encuentra la ficha** del jugador en cuestión, antes de hacer el movimiento resultante de tirar los dados.
 - El método **comprobarMovimiento()** devuelve **Verdadero** si el movimiento solicitado (partiendo de **posicionActual** moverse tantas casillas como indique el parámetro **unidades**) se puede realizar, o **Falso** en caso contrario.
 - El método **Mover()** mueve la ficha tantas casillas como hayan indicado los dados y devuelve **Verdadero** si el jugador ha ganado la partida con ese movimiento, o **Falso** en caso contrario.
 - El método **realizarJugada()**, devuelve **Verdadero/Falso** en función de que el Jugador haya ganado o no la partida.
- b) Realizar el **Diagrama de Secuencia** de la operación **“Mover”** definida en la clase Tablero. Teniendo en cuenta que: **(10% nota).**
- Se trata del Diagrama de Secuencia para el movimiento del Jugador nº1, para el escenario en el que el movimiento es posible.
 - Se actualizará la posición de la ficha del jugador nº1 en el tablero.
 - Se comprobará la posición del jugador nº2 por si el jugador nº 1 se lo ha comido, lo que conllevará poner a 0 la posición de la ficha del jugador nº2.
 - El jugador nº1 **se come** al Jugador nº2, cuando la posición de la ficha del jugador nº1 tras su movimiento, coincide con la posición de la ficha del jugador nº2.
 - El método **Mover()** devuelve **Verdadero/Falso**, si el jugador que realiza dicho movimiento ha llegado o no al final.

Problema 3 (20% nota). Gestión de calificaciones

Se desea desarrollar una aplicación de gestión de las calificaciones de los alumnos. La aplicación deberá cubrir los siguientes aspectos:

El profesor recibe las actas en blanco de las asignaturas de las que es responsable, en formato electrónico. El acta contiene los datos de la asignatura y la lista de alumnos matriculados. Algunas de las acciones que puede hacer el profesor son:



- Completar un acta con las notas de los alumnos.
- Gestionar los alumnos de un acta, pudiendo añadir o borrar un alumno de la misma.
- Integrar las actas de varios grupos de una misma asignatura en una sola acta.

Otras de las opciones que se le exige a la aplicación, para satisfacer las necesidades del profesor, son:

- Permitir la consulta la información académica de cualquier alumno seleccionado
- Obtener una estadística de las calificaciones obtenidas por los alumnos en un determinado grupo de una asignatura.
- Poder visualizar un gráfico indicativo del número de personas que han obtenido una calificación.
- Permitir la importación y exportación de la lista de alumnos con sus calificaciones a un formato compatible con MS Excel.
- Imprimir bien la acta, o bien la lista provisional de calificaciones, dependiendo de las necesidades del profesor en cada momento.

Todas estas acciones requieren de la identificación del profesor.

Se pide realizar el **Diagrama de Casos de Uso relacionado con el actor Profesor.**

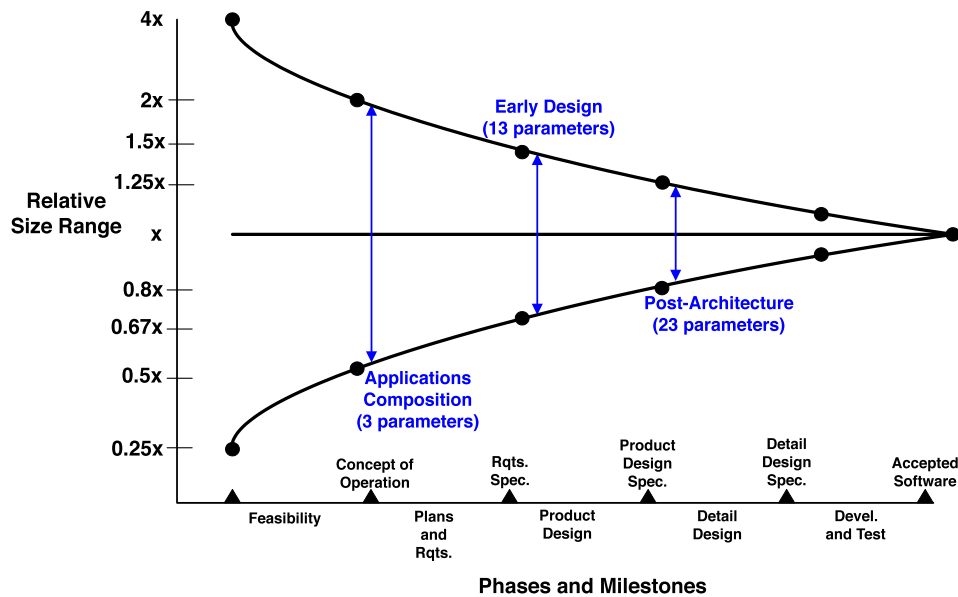
 <p>DEPARTAMENTO DE TECNOLOGÍAS DE LA INFORMACIÓN</p> <p>Universidad de Huelva</p>	<p>Principios y Fundamentos de la Ingeniería del Software</p> <p>Examen convocatoria de Septiembre 2012 03/09/2012 Duración : 30 minutos</p>	 <p>2º Grado en Ingeniería Informática</p>
---	--	---

Teoría (30% nota).

Indicar si son Verdaderas (V) o Falsas (F) las siguientes afirmaciones (respuesta correcta 0.1 puntos, respuesta errónea -0.025 puntos)

El Modelo en Cascada es el proceso que mejor se ajusta a un desarrollo de software acelerado, con necesidad de una versión funcional del software en etapas tempranas de desarrollo.	
El Prototipado tiene como finalidad servir de mecanismo para identificar los requerimientos del software pues, aunque en la práctica haya prototipos evolutivos, se recomienda desecharlos.	
El Modelo en Espiral es un modelo de proceso Evolutivo, apropiado para grandes desarrollos. Utiliza Prototipos, y realiza un análisis de riesgos en todas las etapas.	
La elicitación de requisitos es la última etapa de la Ingeniería de Requisitos, tras el análisis y validación de los mismos.	
Los Casos de Uso son una herramienta para el análisis útil solamente para proyectos orientados a objetos.	
UML es un método que marca las pautas para la realización de Análisis y Diseño Orientado a Objetos.	
En una composición (◆) , los componentes no pueden ser compartidos por varios objetos compuestos, la cardinalidad máxima a nivel de objeto compuesto es obligatoriamente 1 y la supresión del objeto compuesto comporta la supresión de los componentes.	
Todos los modelos simplifican la realidad, lo fundamental es no enmascarar ningún detalle importante, y para ello es fundamental centrarse en un solo modelo (punto de vista), pues con uno es suficiente.	
Una versión candidata (Release Candidate) es una versión alfa con potencial para ser un producto final, que se está dispuesto a liberar a menos que aparezcan errores significativos.	
El proceso de gestión del cambio tiene que ver con el análisis de los costos y beneficios de los cambios propuestos.	
El Personal Software Process [™] (PSP [™]) es un modelo de establecimiento y mejora del trabajo en equipo para procesos software, dando directrices para ayudar a un equipo a establecer sus objetivos, a planificar sus procesos y a revisar su trabajo.	
El enfoque de control del proceso “tradicional” ha demostrado ser efectivo y necesario en proyectos de gran tamaño, pero no resulta adecuado para muchos proyectos actuales.	
El Desarrollo Basado en Componentes intenta maximizar el concepto de reutilización mediante la construcción de una aplicación a partir de componentes software compilada, reutilizable, con una interfaz bien definida.	
Con respecto a las técnicas empíricas , podemos afirmar que: Se basan en datos históricos de proyectos; aplican técnicas de regresión lineal; una vez calibrados, son efectivos y rápidos	
Las metodologías Ágiles surgen como respuesta a los estándares basados en proceso y ponen el énfasis en: La satisfacción al cliente; la entrega rápida de software incremental; los equipos pequeños	

Rodee con un círculo la respuesta correcta: (0.125 puntos c/u)

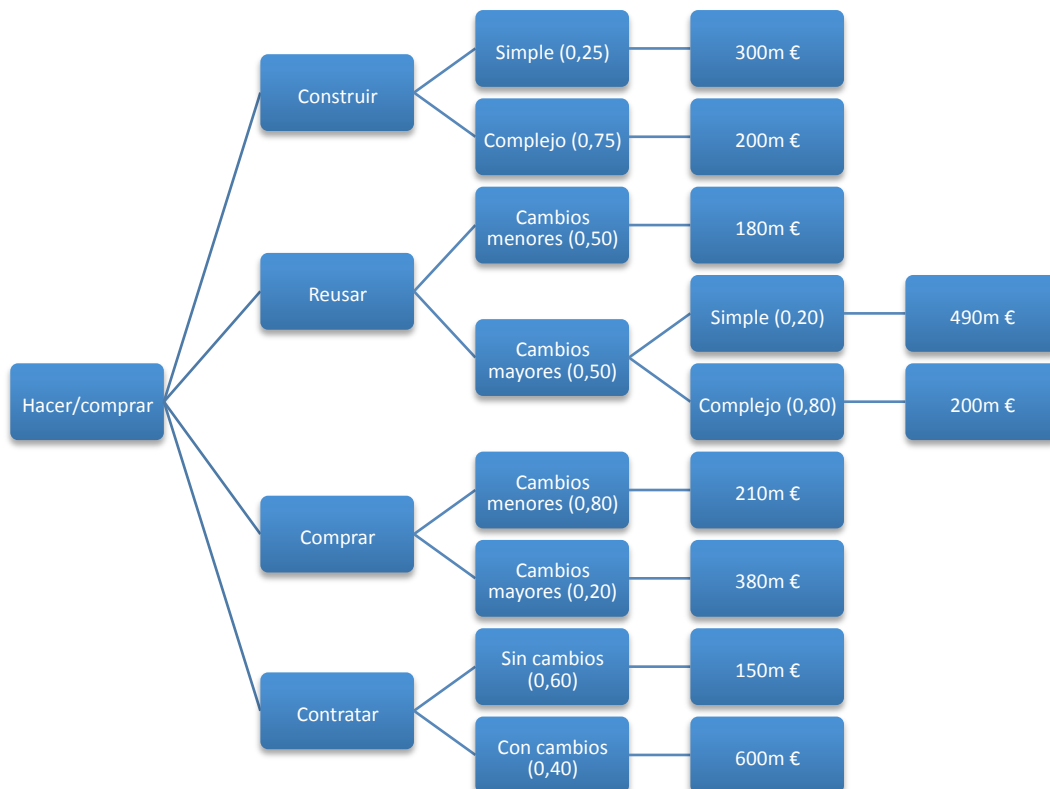


- Al hablar de **COCOMO** se estudió este gráfico que nos resumía:
 - Que COCOMO se divide en tres submodelos, Modelo de composición de aplicación, Modelo de fase de diseño previo y Modelo de post-arquitectura.
 - Que a medida que avanza el proyecto disminuye la incertidumbre y tenemos más parámetros con los que estimar.
 - Que a medida que avanza el proyecto disminuye el rango de tamaño relativo.
 - Ninguna de las anteriores respuestas es correcta.
- La **Exposición al Riesgo** ($ER = P \times C$) donde P es la probabilidad de que el riesgo ocurra y C es el coste para el proyecto si el riesgo ocurre, mide:
 - La influencia de los riesgos de negocio en el proyecto.
 - La valoración económica ponderada por la incertidumbre.
 - La importancia del riesgo en función de la probabilidad de que ocurra
 - La importancia del riesgo en función del coste de solventarlo.
- Un **componente** es:
 - Parte física y reemplazable de un sistema que implementa un conjunto de interfaces.
 - Colección de operaciones que se utiliza para especificar un servicio.
 - Elemento físico que existe en tiempo de ejecución y representa un recurso computacional.
 - Empaquetamiento lógico de elementos físicos.
- Indique qué conjunto de **patrones** son de tipo **GRASP**:
 - Experto en Información, Singleton y Creador.
 - Singleton, Memento y View Helper.
 - Creador, Controlador y Observer.
 - Experto en Información, Creador y Controlador.

Rodee con un círculo la respuesta correcta: (0.5 puntos c/u)

- Dado el siguiente árbol de decisión, qué opción le parecería más justificada:

- a) Construir
- b) Reusar
- c) Comprar
- d) Contratar



- Empresa que actualmente lleva la facturación mediante sistema cliente/servidor decide implantar un sistema informático basado en “Computación en la nube” para un proceso más rápido y fiable. Actualmente dispone de un ordenador servidor (renting renovable anualmente, a razón de 1100€/año durante próximos 3 años) con un contrato de mantenimiento anual de 350€/año, una persona que dedica 60 horas/mes a su gestión (sueldo global 1600€/mes). El nuevo sistema requeriría un upgrade de la línea de datos (80€/mes) y un coste del proveedor del servicio de 170€ / año. La misma persona dedicada a gestión habrá de dedicar un 50% más de tiempo, con una formación del nuevo sistema de una semana. Cuantifique el beneficio al primer año del cambio.

a)	Año 1
Ingresos/Ahorros	
renting	1100
mantenimiento	350
gestion (60 h.)	600
Total ingresos	2050
Costes/Gastos	
upgrade	960
coste cloud	170
Gestión (+50%)	900
Formación (1600/4)	400
Total costes	2430
Beneficio	-380

b)	Año 1
Ingresos/Ahorros	
renting	3300
mantenimiento	350
gestion (60 h.)	600
Total ingresos	4250
Costes/Gastos	
upgrade	80
coste cloud	170
Gestión (+50%)	900
Formación (1600/4)	400
Total costes	1550
Beneficio	2700

c)	Año 1
Ingresos/Ahorros	
renting	3300
mantenimiento	350
gestion (60 h.)	600
Total ingresos	4250
Costes/Gastos	
upgrade	960
coste cloud	170
Gestión (+50%)	900
Formación (1600/4)	400
Total costes	2430
Beneficio	1820