

# MEMORIA FINAL DE ACTIVIDADES

## 1. DATOS GENERALES DEL PROYECTO DE INNOVACIÓN DOCENTE

### **DENOMINACIÓN DEL PROYECTO DE INNOVACIÓN DOCENTE**

INNOVACIÓN DOCENTE EN INGENIERÍA DEL SOFTWARE

### **COORDINADOR/A DEL PROYECTO DE INNOVACIÓN DOCENTE**

Nombre y apellidos: ROBERTO RUIZ SÁNCHEZ

Departamento: DEPORTE E INFORMÁTICA

Centro: ESCUELA POLITÉCNICA SUPERIOR

Correo electrónico: robertoruiz@upo.es

Teléfono: 954977591

### **TITULACIÓN, CURSO, MATERIAS Y/O ASIGNATURA/S SOBRE LAS QUE TIENE REPERCUSIÓN**

GRADO EN INGENIERÍA INFORMÁTICA EN SISTEMAS DE INFORMACIÓN. 2º CURSO. INGENIERÍA DEL SOFTWARE 1

## 2. DESCRIPCIÓN GENERAL DEL PROYECTO

La llegada del Espacio Europeo de Educación Superior supone un esfuerzo bidireccional que afecta tanto al profesorado como al alumnado. Se plantea la necesidad de establecer simultáneamente nuevas metodologías de enseñanza y aprendizaje encaminadas a difundir y crear conocimiento. La dificultad que supone un cambio metodológico tan profundo invita a la búsqueda de nuevas estrategias y organizaciones por medio de la experiencia. Así, a través de este proyecto docente, se pretenden llevar a cabo una serie de acciones encaminadas a corroborar el grado de aplicabilidad y complejidad de nuevas metodologías docentes y evaluadoras para mejorar la formación en competencias en la asignatura de Ingeniería del Software I, impartida por primera en el primer semestre del segundo curso del Grado de este curso 2011-2012.

## 3. RESULTADOS Y EVALUACION DE LA ACTIVIDAD DE INNOVACIÓN Y DOCENCIA PROPUESTA

Uno de los objetivos era la implementación de metodologías activas de aprendizaje, y para ello se han puesto en práctica diferentes elementos de innovación docente utilizando técnicas de aprendizaje cooperativo.

### **Planificación de una clase expositiva con aprendizaje cooperativo informal:**

Actualmente, las clases de teoría o enseñanza básica tienen una duración de 90 minutos, así que el plan de la clase expositiva participativa se ha ajustado a esta duración. A continuación se muestra un ejemplo de una de estas clases donde las actividades de aprendizaje cooperativo informal están sombreadas.

|     |  |                   |
|-----|--|-------------------|
| 5'  | Introducción, presentación del guión de la clase.<br>Importancia de la fase de pruebas   | 2 transparencias  |
| 20' | Costes de la realización de pruebas. Cálculo del ROI.<br>Descripción general del proceso. Unidad de prueba.<br>Casos de prueba en positivo y en negativo (ejemplos).<br>Tipos de pruebas de unidad: de caja negra y de caja blanca.<br>Ejemplo de un posible proceso para la realización de pruebas unitarias.                             | 11 transparencias |
| 10' | Pruebas de caja negra. Explicar con un ejemplo.<br>Pruebas de caja blanca. Criterios de cobertura. Ver ejemplo.<br>El criterio de la mutación. Ver ejemplo.  | 10 transparencias |
| 5'  | Preparar una pregunta en grupos de tres alumnos agrupados por cercanía.  |                   |
| 20' | Estrategias de generación de casos.<br>Técnicas para obtener buenos valores: particiones o clases de equivalencia, análisis de valores límite, dominio de salida y conjetura de errores. Ver ejemplos.   | 8 transparencias  |
| 10' | Perspectivas laborales.<br>Habilidades específicas del ingeniero de pruebas.<br>Organización de equipos de pruebas.<br>Conclusiones.   | 4 transparencias  |
| 20' | Propuesta de ejercicios para verificar en grupo: dejar un tiempo para que el alumno plantee la resolución individualmente; a continuación, en grupos de tres alumnos deben comparar las soluciones y ponerse de acuerdo en la propuesta de una solución que deben entregar firmada.<br>El profesor revisará las soluciones fuera de clase. |                   |

### **Incorporación de aprendizaje cooperativo formal, la técnica del puzzle:**

La asignatura de Ingeniería del Software 1 se presta totalmente a este tipo de estrategias de aprendizaje. El objetivo principal de la asignatura es que los alumnos analicen y diseñen software con orientación a objetos, usando principios y patrones de diseño, empleando como notación el lenguaje UML. En la actualidad, en todo el proceso de aprendizaje se fomenta el intercambio de ideas entre los alumnos para llegar a los modelos software resultantes.

El primer día de clase, en la presentación de la asignatura, se pidió a los alumnos que se

agruparan de tres en tres, con la intención de mantener estos grupos hasta el final del semestre. La intención es la de poder aplicar la técnica del puzzle que se describe a continuación al aprendizaje de algunos temas.

En primer lugar, se selecciona un material del curso que puede dividirse en tres partes razonablemente independientes y cada grupo debe decidir quien se encarga de cada parte (Principios de Diseño OO+Patrón estrategia+Patrón observador); fuera de clase, cada alumno prepara la parte que le ha tocado y las dudas que se le planteen; en una clase posterior, se realiza una reunión de grupos de expertos para discutir las dudas sobre el material trabajado y comparar las soluciones de los ejercicios, y se realizan ejercicios individuales sobre el tema trabajado; fuera de clase, los alumnos preparan individualmente la presentación al grupo base e inventan ejercicios sobre el tema; en la siguiente sesión en clase, se realizará la explicación de las partes en el grupo base; fuera de clase los alumnos resuelven individualmente los ejercicios de los otros dos temas de los compañeros y preparan dudas; finalmente en clase, los grupos base se reúnen, discuten dudas y comparan soluciones, y a continuación, el profesor explica los aspectos destacables del material del puzzle y propone la resolución de un ejercicio que primero lo ven individualmente y posteriormente con el grupo base.

La evaluación se llevará a cabo con la demostración del ejercicio de grupo en clase y con la propuesta de ejercicios sobre cualquier tema del puzzle que los alumnos deben solucionar individualmente. En la demostración del grupo, para forzar que todos los alumnos dominen cada una de las partes, se les avisa que el profesor elegirá que parte defiende cada alumno. Por otra parte, se puede reforzar la dinámica de grupo incentivando si todos los participantes del grupo superan un umbral.

En esta asignatura, en cualquier momento es útil incorporar los grupos base para la resolución de ejercicios mediante diagramas UML, o para la preparación de patrones de diseño software no explicados por el profesor.

### **Resultados:**

En general, se ha observado que los alumnos mejoran en la actitud y la motivación, y que las horas de trabajo en casa también pueden permitir avanzar en el temario. También ha ayudado el uso de problemas y ejemplos de la vida real para poner en práctica lo aprendido.

Se han solicitado numerosas entregas, donde algunas eran ejercicios a realizar con los conceptos vistos en clase, y otras eran evaluables, en este apartado se incluyen los exámenes. De esta manera, los alumnos pueden comprobar cual va siendo su nivel de aprendizaje.

Dado que estamos en estas semanas evaluando a los alumnos, no disponemos de los resultados finales de la aplicación de las acciones llevadas a cabo.

Nuestra intención es ir perfeccionando la implantación del modelo y afinando los tiempos de dedicación a cada actividad con la intención de mejorar la calidad de la enseñanza. Además, aplicar los conocimientos adquiridos con la experiencia de este proyecto en varias asignaturas de Ingeniería del software del grado que se impartirán posteriormente. Está claro que debemos mejorar algunos detalles, perfeccionando la implantación de estas metodologías, pero estoy convencido de que siguiendo este modelo se puede mejorar la calidad de la enseñanza en este tipo de asignaturas.