

UNA PROPUESTA DE ASIGNATURA PARA LA INGENIERÍA DEL SOFTWARE ORIENTADA A OBJETOS.

Miguel Angel Pérez Toledano, Amparo Navasa Martínez, Marisol Sánchez Alonso

Departamento de Informática. Escuela Politécnica.

Universidad de Extremadura.

Avda. de la Universidad s/n.

10071 Cáceres.

Resumen.

En la mayoría de los planes de estudio de la Ingeniería Informática, los temas de orientación a objetos suelen incluirse de forma aislada dentro de las asignaturas de Ingeniería del Software, complementando así la visión del alumno con respecto a los sistemas tradicionales. La creciente utilización de herramientas O.O, unida a las interesantes características que posee este tipo de sistemas y al poco énfasis que se hace del ciclo de vida de aplicaciones orientadas a objetos dentro de los planes de estudio actuales, nos ha movido a ofertar una materia “ Desarrollo de sistemas orientados a objetos”, que intentamos abarque los aspectos de OO que no son tratados con suficiente profundidad en las asignaturas obligatorias de Ingeniería del Software.

1. Introducción

El auge de los sistemas orientados a objetos ha hecho que las asignaturas de Ingeniería del Software, hayan ido evolucionado. Se han introducido paulatinamente en sus temarios referencias al desarrollo de este tipo de aplicaciones. Sin embargo se han tratado como temas aislados dentro de asignaturas concebidas para enseñar el ciclo de vida estructurado. Ahora bien la creciente demanda, por parte de los alumnos, de ampliar sus conocimientos

en el desarrollo de sistemas orientados a objetos; unida a la cada vez mayor expansión que estos sistemas tienen en el mundo empresarial, y a la dificultad de incluir estos temas en las asignaturas clásicas de Ingeniería del Software, por su carácter cuatrimestral; ha hecho que en un principio nos planteásemos la actualización de los temarios, de las asignaturas de Ingeniería del Software como medio de paliar esta necesidad. Sin embargo, hemos llegado a la conclusión de que esta no es la solución. En las organizaciones actuales todavía prima la utilización de métodos de desarrollo que siguen un ciclo de vida estructurado, y aunque bien es cierto que esta situación se prevé pueda cambiar en los próximos años, es clara la necesidad de seguir formando a los alumnos en el desarrollo de este tipo de sistemas. Por ello consideramos que deberemos paliar las deficiencias en el tratamiento de objetos de las actuales asignaturas de Ingeniería del Software, ofertando una nueva asignatura que incluya el desarrollo completo de sistemas orientados a objetos.

Por consiguiente la inclusión de esta nueva asignatura dentro de los planes de estudio, no tiene como objetivo competir con las que ya existen en Ingeniería del Software encargadas de enseñar el ciclo de vida estructurado, sino que la idea es complementar los conocimientos del alumno con una visión más amplia del desarrollo de sistemas orientados a objetos.

2. Objetivos de la materia

A la hora de diseñar la asignatura nos hemos planteado los siguientes objetivos generales:

- ◆ -Afianzar los conocimientos básicos, que ya tiene el alumno, de los sistemas O.O.
- ◆ Estudio, análisis y diseño de sistemas O.O., y su comparación con el ciclo de vida estructurado.
- ◆ Conocimiento de una metodología de desarrollo O.O (UML).
- ◆ Introducción a las bases de datos O.O.
- ◆ Control de calidad de aplicaciones O.O.
- ◆ Métricas orientadas a objetos.
- ◆ Enlace con la programación y visión general de los lenguajes O.O. y sus características.

A la vista de estos objetivos, hemos estructurado la materia dividiendo su contenido en 3 módulos cada uno de ellos con objetivos específicos. El primer módulo “Desarrollo orientado a objetos” tiene como objetivo básico el conocimiento del ciclo de vida de aplicaciones orientadas a objetos, y la utilización de una metodología de desarrollo. El segundo módulo “Control de calidad en sistemas O.O.” pretende dar al alumno las nociones necesarias para la realización de pruebas y métricas que le permitan comprobar la calidad del software que está desarrollando. Y por último el módulo tercero “Programación orientada a objetos” nos permitirá enlazar las fases de análisis y diseño con la programación, dando una visión general de los distintos LOO actuales y de los aspectos de OO que cubren.

3. Características de la asignatura

El nombre de la asignatura será: **Desarrollo de sistemas orientados a objetos**. El número de créditos que proponemos es de 6 (4,5T+1,5P). Su carácter es optativo, y se impartirá en el segundo ciclo como asignatura de 2º cuatrimestre; aconsejando su matriculación en el 5º curso, en el que el alumno habrá ya adquirido los conocimientos relativos a Ingeniería del Software tradicional, y los referentes a gestión del software, métricas y pruebas para sistemas tradicionales.

4. Temario

Modulo I: Desarrollo orientado a objetos.

Tema 1: Introducción al Desarrollo de Sistemas Orientados a Objeto.

- 1.1. - Justificación.
- 1.2. - Conceptos básicos.
- 1.3. - Principios básicos.
- 1.4. - Metodologías emergentes.
- 1.5. - Estado del Arte.

Tema 2: Análisis de Sistemas Orientado a Objeto.

- 2.1. - Enfoque convencional vs enfoque Orientado a Objetos.
- 2.2. - Métodos de análisis.
- 2.3. - Análisis Orientado a Objeto. Una metodología.

Tema 3: Diseño de Sistemas Orientado a Objeto

- 3.1. - Enfoque convencional vs enfoque Orientado a Objetos.
- 3.2. - Métodos de diseño.
- 3.3. - Diseño Orientado a Objeto. Una metodología.

Tema 4: Bases de Datos Orientadas a Objeto

- 4.1. - El modelado de los datos. Bases de datos convencionales
- 4.2. - Bases de datos Orientado a Objeto.
 - 4.2.1. - Definición,
 - 4.2.2. - Ventajas e inconvenientes.
 - 4.2.3. - Situación actual.

Tema 5: Interfaces de Usuario O.O.

- 5.1. - Características de las I.U.
- 5.2. - Diseño de I.U. Orientado a Objeto
- 5.3. - Herramientas CASE.

Modulo II: Control de calidad en sistemas orientados a objetos.**Tema 6: Pruebas Orientadas a Objeto.**

- 6.1. - Definición, necesidad de la realización de pruebas.
- 6.2. - Las pruebas en el desarrollo tradicional.
- 6.3. - Modelos de pruebas para AOO y DOO.
- 6.4. - Estrategias de pruebas.
- 6.5. - Diseño de casos de prueba para software Orientado a Objeto.

Tema 7: Métricas para Sistemas Orientados a Objeto

- 7.1. - Objetivo de las métricas, características.
- 7.2. - Aplicación de las métricas durante el desarrollo Orientado a Objeto.

Tema 8: Gestión de proyectos de software Orientado a Objeto.

- 8.1. - Introducción.
- 8.2. - Estimaciones y planificación de proyectos Orientados a Objeto.
- 8.3. - Seguimiento de un proyecto Orientado a Objeto.

Modulo III. Programación orientada a objetos.**Tema 9: Programación Orientado a Objeto.**

- 9.1. - Programación tradicional vs programación Orientada a Objeto.
- 9.2. - Lenguajes de programación Orientados a Objeto.
- 9.3. - Desarrollo tradicional y programación Orientada a Objeto.
- 9.4. - Desarrollo Orientado a Objeto y programación tradicional.

5. Prácticas

Para la realización de las prácticas contaremos con el siguiente material:

- ◆ Herramienta de desarrollo: Rational Rose.
- ◆ Laboratorio de trabajo: 30 puestos conectados en red para trabajo en grupo.
- ◆ Contenido de la práctica: desarrollo de un supuesto práctico en las horas de clase utilizando las herramientas anteriores.
- ◆ Carácter de las prácticas: obligatorio.

El objetivo de la parte práctica consiste en desarrollar un supuesto que incluya las fases de análisis y diseño (e interfaz de usuario) aplicando la metodología estudiada. En cuanto a la elección de la metodología de trabajo (UML) hemos optado por aquella que recoge los estándares de las anteriores metodologías de análisis y diseño OO como OMT, Jacobson y Booch.

6. Bibliografía

La bibliografía que proponemos para la materia a impartir la hemos dividido en 2 partes:

Bibliografía de trabajo:

- Booch G., Rumbaugh J., Jacobson I.
"Unified Modeling Language User Guide".
Editorial: Addison Wesley
- Booch G., Rumbaugh J., Jacobson I.
"Unified Modeling Language Reference Manual"
Editorial: Addison Wesley
- Pressman Roger
"Ingeniería del software. Un enfoque práctico
4ª edición" Editorial Mc Graw Hill.
- Referencia:
<http://www.rational.com/uml/>

Bibliografía de apoyo:

- Texel y Williams

“Use Cases Combined with Booch, Omt, Uml. Process and products”.

Editorial Prentice Hall.

- Grahan Ian

“Métodos orientados a objetos”.

Editorial Addison Wesley.

- Leiberherr. K.

“Adaptative Object-Oriented Software. The Demeter method with propagation patterns”.

Editorial PWS Publishing Company.

- Martin y Kendall

“UML Distilled: applying the standard Object Modeling Language”.

Editorial Addison Wesley.

- Revista Novática. Números 118, 121, 125, 128.

- Rumbaugh J., Blaha M., Premerlani, W., Eddy F., Lorensen. W.

“Modelado y diseño orientado a objetos. Metodología OMT”.

Editorial Prentice Hall.

- Martin y Odell.

“Análisis y diseño orientado a objetos”.

Editorial Prentice Hall.

- Jacobson

“Object Oriented Software Engineering”.

Editorial Addison Wesley.

- Booch G.

“Object Oriented Annalysis and Design”.

Editorial Benjamin / Cummings Publishing Company Inc.

7. Conclusión

La asignatura propuesta permite adentrarnos en el ciclo de vida de los desarrollos O.O., centrándonos en este tipo de sistemas, cuya visión y contenido global no se consigue con las asignaturas de carácter obligatorio que cursan los estudiantes en la carrera. Se suple así la carencia y falta de profundidad de estos conocimientos que tenían hasta ahora los alumnos de la Ingeniería Informática.