

# El Proyecto Final de Carrera en Ingeniería Técnica de Informática

Juan Sánchez Díaz; José Hilario Canós Cerdá

Departamento de Sistemas Informáticos y Computación

Universidad Politécnica de Valencia

46021 Valencia

e-mail: {jsanchez, jhcanos}@dsic.upv.es

## Resumen

En el artículo se presenta una propuesta metodológica, utilizada en los últimos cuatro años, para la enseñanza de la asignatura *Proyecto Final de Carrera* (PFC), en la Escuela de Informática (EUI) dentro de la Universidad Politécnica de Valencia (UPV). La propuesta sustituye la clase magistral por un seguimiento tutorizado de los grupos que realizan el proyecto. Los alumnos, dentro de un grupo, juegan roles asociados a las distintas fases del ciclo de desarrollo del software, mientras que los profesores tutores juegan el rol de clientes. El objetivo de las tutorías es asesorar durante las distintas fases de desarrollo. Con respecto a la evaluación ésta la realizan tanto los profesores tutores como los componentes del grupo de proyecto. Consideramos que esta forma de abordar el PFC permite acostumbrar a los alumnos a trabajar y efectuar desarrollos en grupo, fomentando la interacción con los futuros clientes. Un importante porcentaje de los proyectos se realizan en colaboración con empresas externas.

## 1. Introducción

El plan de estudios de las titulaciones de Ingeniero Técnico en Informática (gestión y sistemas) de la EUI de la UPV distingue dos asignaturas de proyecto: Proyecto Final de Carrera 1 (PFC1), obligatoria de 3 créditos, y PFC2, opcional de 12 créditos. La asignatura PFC1 se imparte en el quinto y sexto semestre. Su docencia se la reparten los departamentos: Sistemas Informáticos y Computación (DSIC), Informática de Sistemas y Computadoras (DISCA), Ingeniería de Sistemas y Automática (DISA) y Organización de Empresas (DEO).

Para PFC1 la Junta de Centro decidió que cada departamento de los citados anteriormente tuviese libertad para organizar la docencia de la misma. Algunos de ellos imparten clases teóricas, que se acompañan de la realización de un proyecto en grupo. Estos proyectos, en la mayoría de los casos, no abarcan la fase de implementación. Otros, en cambio, hacen más énfasis en la implementación que en los aspectos de análisis y diseño de un proyecto informático.

Nosotros creemos que la asignatura debe cubrir completamente el ciclo de desarrollo de un proyecto informático y ocuparse tanto de los aspectos metodológicos como de los aspectos de implementación.

En este artículo comentaremos la propuesta metodológica dentro de PFC 1 que se lleva a cabo en el DSIC. Por una parte abarca los aspectos de desarrollo en grupo de un proyecto informático, sin descuidar los aspectos de implementación y documentación que son obligatorios.

Esta forma de trabajo ha permitido que diversos grupos hayan obtenido, durante los últimos años, premios a los mejores proyectos finales de carrera, otorgados por la fundación Bancaja y la Unidad de Prácticas en Empresa de la UPV. También hay que resaltar que uno de los proyectos ha dado lugar a una publicación en el marco de ACM.

El trabajo está estructurado como sigue: en el apartado segundo se enmarca el contexto de la asignatura dentro del plan de estudios. La tercera sección contiene una descripción detallada de la propuesta metodológica. La cuarta una selección de los proyectos más relevantes y finalmente la quinta contiene las conclusiones.

## 2. La asignatura dentro del plan de estudios.

A la hora de plantear una metodología docente y los objetivos de la asignatura, es conveniente saber el nivel de conocimientos que poseen los alumnos.

Cuando se matriculan de PFC1 ya han cursado aquellas asignaturas relacionadas con ingeniería de la programación: Metodología y Tecnología de la Programación (MTP) e Ingeniería del Software (INS). En MTP se presentan diversos ciclos de desarrollo: el ciclo de vida clásico, el ciclo de vida clásico con prototipado y el ciclo de vida en espiral. Un aspecto importante de la citada asignatura es que abarca el ciclo de desarrollo desde el punto de vista estructurado: análisis y diseño estructurado. También se abarca, aunque muy ligeramente, la fase de pruebas de caja blanca y caja negra. Como complemento a la teoría, en las prácticas, los alumnos aprenden a manejar una herramienta CASE (System Architect) que les ayuda a documentar las etapas de análisis y diseño. La tabla 1 contiene el temario de MTP.

<i>Tema</i>	<i>Título</i>
1	Ingeniería del Software: ciclos de vida
2	Análisis Estructurado
3	El diagrama de estructura
4	Métodos de especificación de módulos
5	Acoplamiento
6	Cohesión
7	Guías de diseño
8	Análisis de transacciones
9	Pruebas de caja blanca y caja negra

**Tabla 1. Temario de MTP**

Como complemento, en INS, se aborda el ciclo de desarrollo del software desde el punto de vista de la orientación a objetos. En particular se utiliza una combinación de OMT [7] y del lenguaje unificado de modelado (UML [1]). Los contenidos teóricos abarcan desde el modelado conceptual hasta la implementación en un lenguaje orientado a objetos.

La tabla 2 contiene los distintos temas de la asignatura.

En la parte práctica aprenden y utilizan un lenguaje orientado a objetos como Delphi o C++.

<i>Tema</i>	<i>Título</i>
1	Software e Ingeniería del Software
2	Paradigmas de ciclo de vida.
3	Planificación y control de proyectos.
4	Introducción a la orientación a objetos.
5	Modelado orientado a objetos
6	Diseño orientado a objetos
7	Programación orientada a objetos
8	Técnicas de prueba de software OO.

**Tabla 2. Temario de INS.**

Por consiguiente los alumnos poseen el suficiente bagaje a nivel teórico para abordar desarrollos software de tamaño medio, ya sea desde una perspectiva estructurada o bien desde una perspectiva orientada a objetos.

## 3. Objetivos y metodología en PFC-1

El objetivo primordial de la asignatura PFC1 es conseguir que los alumnos realicen en grupo un proyecto final de carrera totalmente aplicado y lo más próximo posible a un caso real. Este proyecto ha de abarcar todas las fases del desarrollo de software, siendo un aspecto importante del mismo la generación de documentación asociada a cada etapa.

Los enunciados de proyecto son principalmente propuestos por los profesores de la asignatura, aunque, en algunos casos, se admiten proyectos propuestos por los alumnos. La mayor parte de estos últimos se realizan en colaboración con una empresa externa, estando tutorizados por ambas partes.

Un aspecto importante a la hora de seleccionar los objetivos y la metodología de la asignatura ha radicado en el hecho de disponer de un número limitado de créditos (3 en particular). Esto imposibilita que el proyecto se lleve a cabo de manera individual ya que no podrían abarcarse todas las etapas de desarrollo.

### 3.1. Calendario

A principio del cuatrimestre se convoca a los alumnos matriculados y se realiza una presentación de los objetivos, metodología y dinámica de la asignatura.

La asignatura posee una página web (<http://mariachi.dsic.upv.es/pfc1>), donde se encuentran las normas, las propuestas de proyecto y las asignaciones de temas a grupos.

Los alumnos disponen de un tiempo para confeccionar los grupos de trabajo. De la lista de proyectos que propone cada profesor, tienen que elegir tres por orden de preferencia. En la página Web existe un documento Word para que rellenen y envíen por correo electrónico los componentes de cada grupo.

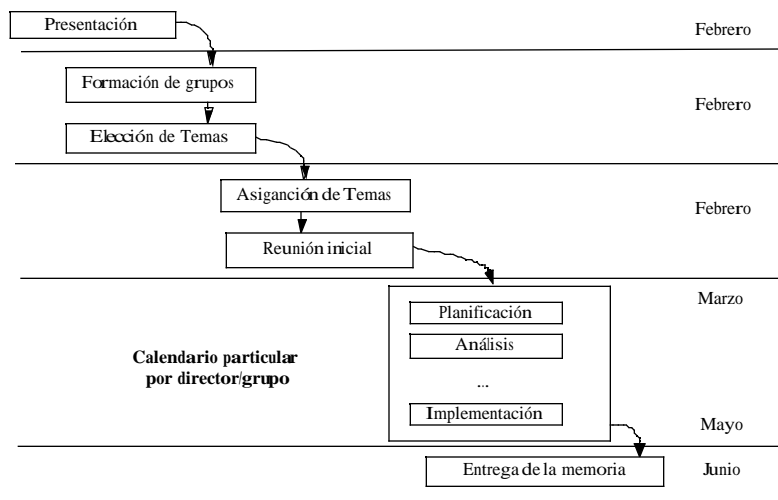
A partir de los correos recibidos y de acuerdo al Plan de Ordenación Docente (POD) del departamento se produce una asignación de grupos a profesores tutores.

### 3.2. Equipos de trabajo

Los equipos de trabajo o grupos están formados por cuatro o cinco personas. La elección de los grupos es totalmente libre. Dentro de cada grupo los alumnos, según su criterio, deben repartirse los cargos de la siguiente forma:

- 1 Responsable del grupo
- 1 Analista/Diseñador
- 2 Programadores
- 1 Responsable de documentación

El responsable del grupo es la interfaz con el profesor tutor. La reunión inicial se efectúa con todos los componentes del grupo, y el resto de las



**Figura 1. Calendario de reuniones para el segundo semestre**

Una vez producida la asignación cada profesor tutor se encarga de convocar a los grupos que le han correspondido y fijar el calendario para el resto de reuniones (figura 1).

Los alumnos tienen libertad para formar grupos, elegir el método de desarrollo (estructurado, orientado a objetos, etc.), y en algunos casos para seleccionar la plataforma y el lenguaje de implementación.

reuniones con los encargados de cada fase.

Una restricción que imponemos para evitar que los grupos queden descompensados, en el sentido de que puede haber componentes que se retiren del proyecto, perjudicando a sus compañeros, es que la pertenencia a un grupo supone presentarse a la asignatura en la convocatoria correspondiente

### 3.3. Documentación a entregar

La documentación a entregar se corresponde con las etapas del ciclo de vida:

- Planificación.
- Análisis.
- Diseño.
- Implementación.
- Manuales de usuario y de instalación

Se manejan dos tipos de planificación, la ideal que se entrega al inicio del proyecto, en formato MS Project, y la real que se adjunta cuando finaliza el mismo. Los tiempos de la planificación idea, deben adaptarse a los cuatro meses y medio que abarca la realización del proyecto.

La documentación de análisis comprende obligatoriamente la especificación de requisitos, en formato IEEE-830 [1].

<i>Fase</i>	<i>Documentación</i>
<i>Planificación</i>	MS-Project
<i>Análisis</i>	IEEE Diagramas de Flujo Diccionario de Datos Especificación de procesos
<i>Diseño</i>	Diagramas de Estructura Especificación de módulos Modelo relacional
<i>Implementación</i>	Código fuente Base de datos
<i>Manuales</i>	Usuario e instalación

**Tabla 4. Desarrollo clásico y documentación.**

Si se utiliza un método de desarrollo estructurado (tabla 4) como análisis estructurado, la documentación contiene los diagramas de flujo de datos, el diccionario de datos y la especificación de procesos. Para la fase de diseño: diagramas de estructura, especificación de módulos y modelo relacional de la base de datos.

En la etapa de implementación se debe entregar el código fuente, debidamente comentado, y un programa de instalación del producto. Finalmente se confeccionan los manuales de usuario y los procedimientos de instalación.

Para un desarrollo orientado a objetos (tabla 5), habitualmente se utiliza el método Object Modelling Technique (OMT [7]) o bien el

lenguaje unificado de modelado (UML [1]).

<i>Fase</i>	<i>Documentación</i>
<i>Planificación</i>	MS-Project
<i>Análisis</i>	IEEE Modelo de casos de uso Modelo de objetos Modelo dinámico
<i>Diseño</i>	Arquitectura. Modelo relacional
<i>Implementación</i>	Código fuente Base de datos
<i>Manuales</i>	Usuario e instalación

**Tabla 5. Desarrollo OO y documentación.**

En cualquiera de los casos toda la documentación posteriormente se recoge en una memoria individual, que debe presentarse a la hora de obtener el título de Ingeniero Técnico en Informática

### 3.4. Interacción con los profesores

La interacción con los profesores o tutores de proyecto se centra en la discusión de los requisitos de la aplicación a desarrollar y en la evaluación de la documentación aportada en cada fase.

El número de reuniones, siempre con cita previa y para asuntos concretos, disminuye según se avanza hacia la fase de implementación del proyecto. Lógicamente hay un mayor número de reuniones asociadas a la elicitación de requisitos que las que se producen cuando se entra en la fase de pruebas del sistema.

Algunas reuniones también se emplean para evaluar y modificar los modelos, tanto de análisis como de diseño, propuestos por los alumnos.

### 3.5. Evaluación

La evaluación de la asignatura comprende dos partes. Una evaluación global del profesor (Glo) que se centra en la actitud mostrada en las reuniones de seguimiento y en la calidad tanto de la documentación aportada como del producto software. Los componentes del grupo evalúan también, de forma confidencial, a sus compañeros (Ind). La nota se calcula de la siguiente forma:  $Glo*0.5 + Ind*0.5$ .

Este modelo permite penalizar a aquellos

componentes que no hayan realizado el trabajo al que se han comprometido dentro del grupo.

#### 4. Algunos proyectos realizados

De los proyectos realizados en los últimos años citaremos a continuación algunos de los más relevantes:

- Desarrollo de un portal Web para una empresa inmobiliaria, con los servicios de compra-venta, alquiler de inmuebles y promoción de nuevas viviendas. Este proyecto recibió un premio de la fundación Bancaja, a uno de los mejores proyectos del curso 1999-2000. Fue desarrollado utilizando la metodología OMT. Actualmente se encuentra en explotación en la siguiente dirección <http://www.gesinco.es/>.
- Generador de citas bibliográficas dentro de documentos Word (Bibword). Bibword está formado por un conjunto de herramientas que permiten la inserción de referencias bibliográficas, contenidas en una base de datos, dentro de un documento Word. Posee pasarelas que permiten transformar referencias del formato utilizado por Latex (BiBTeX) al formato utilizado por el programa. La herramienta puede descargarse de la dirección <http://mariachi.dsic.upv.es/bibword/index.htm>. Puede consultarse en [2] una publicación internacional de la misma.
- Sistema para el tratamiento de pedidos electrónicos para suministrar componentes a fabricas de Ford Europa. Este proyecto también fue galardonado con un premio Bancaja [6] durante el curso 99/2000. Desarrollado con OMT, se encuentra operativo en un proveedor de Ford España.
- Desarrollo de un portal WAP para una empresa del sector textil. Se trata de un portal al que pueden acceder tanto los proveedores como los clientes de la compañía para consultar estado de pedidos, estado de suministros, etc. En la actualidad no se encuentra operativo.

#### 5. Conclusiones

Se ha presentado una propuesta metodológica que permite abordar el desarrollo de una asignatura tan importante para los alumnos como es proyecto final de carrera, ya que ésta puede ser para ellos un primer contacto con su posterior actividad profesional.

Entre los aspectos positivos de la misma cabe citar:

- Fomento del trabajo en grupo: donde cada componente debe jugar un rol específico (analista, diseñador, etc).
- Negociación y trato directo con el cliente, en este caso o el profesor tutor o bien la empresa para la que se hace el proyecto.
- Utilización práctica de métodos de análisis y diseño, obligando a los alumnos a seguir estrictamente el ciclo de desarrollo.
- Realización de un proyecto aplicado y realista que en algunos casos se utiliza dentro de entornos empresariales.

La experiencia ha demostrado también algunos aspectos negativos:

- Dificultad de que algunos alumnos, sobre todos los del segundo semestre, se adapten a la planificación del proyecto, debido principalmente a la carga docente que arrastran del primer cuatrimestre.
- Quejas de los alumnos con respecto a los proyectos asignados. Esto se debe al proceso de selección en sí. Cada grupo elige tres proyectos por orden de preferencia, por cuestiones de POD los grupos se tienen que dividir entre los profesores que imparten la asignatura. Esto ocasiona que en la mayor parte de los casos no les corresponda el proyecto que han elegido.
- Algunas fases del ciclo de desarrollo, como la fase de pruebas no se abordan, por cuestiones de tiempo, de una manera un poco superficial.

Es interesante indicar que los alumnos muestran un mayor grado de preferencia hacia los proyectos que tienen que ver con las nuevas

tecnología de información, como puede ser la tecnología WAP y las aplicaciones que se desarrollan para la Web. Con respecto a los lenguajes de programación, los alumnos muestran una gran predisposición a emplear el lenguaje Java.

Finalmente, queremos resaltar el alto grado de motivación de los alumnos por aprender por ellos mismos las tecnologías necesarias para llevar a cabo los proyectos. Hay que tener en cuenta que muchas veces es necesario utilizar lenguajes que no conocen, protocolos que no han estudiado, etc.

## **Referencias**

- [1] Booch G; et al “El lenguaje unificado de modelado”. Addison-Wesley 2000.
- [2] Canos J; The Summer 2000. Issue of ACM Crossroads. <http://www.acm.org/crossroads>.
- [3] IEEE-830. Guía del IEEE para la especificación de requisitos del software. (ANSI/IEEE Std. 830-1984)
- [4] Molina A; et al. “Metodología y Tecnología de la Programación”. Servicio Publicaciones. Universidad Politécnica de Valencia.
- [5] Page-Jones, M. “The Practical Guide to Structured Systems Design”. Prentice Hall 1989.
- [6] Premios Bancaja. Unidad de Prácticas en Empresa. <http://www.sie.upv.es>.
- [7] Rumbaugh J; et al “Modelado y Diseño Orientado a Objetos: metodología OMT”. Prentice Hall 1992.
- [8] Yourdon E. “Análisis Estructurado Moderno”. Prentice Hall 1989.