

# PROYECTOS DINÁMICOS PARA EL APRENDIZAJE EN LOS PRIMEROS CURSOS DE LAS ENSEÑANZAS DE PROGRAMACIÓN INFORMÁTICA

**Alfonso E. Márquez-Chamorro<sup>1</sup>, Gualberto Asencio-Cortés<sup>1</sup>, Raúl Giráldez Rojo<sup>1</sup>, Alicia Troncoso Lora<sup>1</sup>**

*<sup>1</sup>Escuela Politécnica Superior. Universidad Pablo de Olavide,  
{amarcha,guasecor,rgirroj,atrolor}@upo.es*

## Resumen

Hoy en día, el Espacio Europeo de Educación Superior ha propiciado la modificación de las estructuras rígidas, que estaban establecidas en todas las universidades españolas, para convertir a los estudiantes en el centro del proceso de enseñanza-aprendizaje que en las mismas se lleva a cabo. En particular, en las ingenierías siempre ha existido un número muy elevado de abandonos y los estudiantes en su mayoría consiguen graduarse en un número de años muy superior al número de años de la titulación. Esto hace especialmente interesante la aplicación de nuevas metodologías en el aula para lograr mejorar el aprendizaje, y a su vez conseguir mejores tasas de rendimiento, eficiencia y graduación, necesarias para la acreditación de los títulos. En este contexto, se plantea una propuesta basada en proyectos dinámicos para el aprendizaje en asignaturas relacionadas con la programación informática que suelen estar en los primeros cursos de titulaciones de ingeniería y que suelen ser bastante áridas para los estudiantes que inician estos estudios sin ningún tipo de conocimiento en programación. La propuesta consiste en el diseño de un proyecto global en la asignatura que pueda ser desarrollado de forma secuencial en las clases de enseñanzas prácticas. De esta forma, el estudiante va desarrollando el proyecto conforme va adquiriendo los conocimientos necesarios en clases de teoría y prácticas, y no al final del semestre cuando ya tiene todos los conocimientos. Para llevar esta experiencia a cabo es necesario diseñar miniproyectos, los cuales deben basarse los unos en los otros de forma secuencial.

Palabras clave: innovación; proyecto, aprendizaje, asignatura programación, ingeniería informática, primer curso.

## 1 INTRODUCCIÓN

La importancia de las asignaturas de programación informática estriba en el desarrollo de las habilidades que se adquieren al aprender los fundamentos algorítmicos junto a un lenguaje de programación concreto. Reforzar el pensamiento lógico mediante el aprendizaje de cómo plantear un problema, cómo organizar la solución del mismo en una secuencia lógica de pasos y cómo formular la toma de decisiones, es la principal competencia que subyace en el objetivo de este tipo de asignaturas.

Por otro lado, estas asignaturas normalmente suponen para el estudiante un esfuerzo superior al del resto de asignaturas, debido principalmente al hecho de utilizar conceptos y términos difíciles de comprender e interpretar por los alumnos de cursos iniciales, ya que no se han enfrentado previamente a ellos. Este hecho, provoca altas tasas de abandono en las titulaciones donde se imparten las citadas asignaturas y un bajo rendimiento académico general en las mismas [1]. Es por tanto necesario encontrar alternativas estratégicas de aprendizaje para las asignaturas de programación de cursos iniciales, tales como las metodologías de aprendizaje basado en proyectos [2] [3]. En este artículo se presenta una propuesta de estrategia metodológica para aplicar en dichas asignaturas.

La propuesta de este artículo está basada en el aprendizaje basado en proyectos, y describe la implantación de un proyecto incremental dividido en miniproyectos para asignaturas de programación de los primeros cursos. Este proyecto será independiente del lenguaje de programación, de la asignatura y de la titulación donde se imparta la misma, ya sea una ingeniería u otra titulación.

Es importante que la temática de los proyectos esté relacionada con la asignatura en cuestión. El tema ha de ser atractivo, motivante y significativo para los alumnos, además de innovador y actual.

Asimismo, ha de resolver problemas reales de la actualidad de manera que el alumno se sienta motivado con el mismo.

## 2 METODOLOGÍA

Las titulaciones que incorporan asignaturas de programación tienen en su plan de estudio un número significativo de créditos de enseñanzas prácticas para estas asignaturas, debido a la naturaleza de las competencias a adquirir en las mismas. En este contexto, como se ha comentado anteriormente, esta propuesta es aplicable a cualquier tipo de plan de estudios, número de créditos de la asignatura y titulación.

La realización de los proyectos tendrá lugar durante algunas de las sesiones de enseñanzas prácticas. Estas sesiones se dividirán en dos partes:

1. Parte ordinaria: explicación magistral por parte del profesor de los conceptos y resolución de los problemas del boletín de prácticas correspondiente.
2. Desarrollo y/o entrega de los miniproyectos de manera autónoma e individual, los cuales serán puntuables dentro del modelo de evaluación continua aplicado. La planificación docente para la entrega de los miniproyectos así como la del proyecto final estará acorde al número de créditos de la asignatura en cuestión.

A medida que los elementos conceptuales de fundamentos de la programación son introducidos en la asignatura (tipos de datos, variables, tablas, cadenas de caracteres, construcciones algorítmicas tales como sentencias condicionales y bucles, tratamiento de ficheros ...) el alumno tendrá que utilizarlos en los miniproyectos. De esta forma, se permite al estudiante ir desarrollando un trabajo por partes.

El fraccionamiento de los contenidos de la asignatura debe definir la cantidad de miniproyectos y su secuenciación, de tal manera que sean realizables durante las clases de enseñanzas prácticas, o en su defecto en el tiempo de trabajo autónomo correspondiente a la sesión de prácticas. Las horas totales invertidas por el estudiante en la realización del miniproyecto no deben sobrepasar las horas estimadas que alumno debe dedicar al estudio de la asignatura en una semana, ya que si los miniproyectos son de una mayor extensión, se pierde el interés por parte del alumno y no se logran los propósitos planteados.

Una ventaja de las asignaturas de programación es la capacidad de poder dividir en módulos casi cualquier solución. En el contexto de esta propuesta, esta ventaja permite dividir el proyecto global en miniproyectos independientes y evaluables por separado pero cuyos conceptos inherentes siempre incluyen los del miniproyecto anterior, tal y como se ilustra en la Fig. 1. Es evidente que este modelo incremental conlleva un esfuerzo importante de planificación por parte del equipo docente, ya que el proyecto final debe tener unas características que permitan la implementación de dicho modelo, además de estar completamente definido antes de la primera sesión de prácticas que incluya el desarrollo del primer miniproyecto.

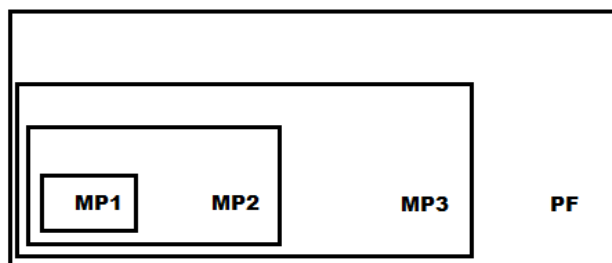


Figura 1. Representación gráfica de la propuesta, donde MP representan a los distintos miniproyectos y PF representa el proyecto final.

### 3 APLICACIÓN DE LA METODOLOGÍA A UN EJEMPLO CONCRETO

Para probar esta metodología, se va a describir un ejemplo propuesto para una asignatura de programación orientada a objetos tipo de primer curso de una titulación de *Grado en Ingeniería Informática*. Los principales objetivos de dicha asignatura tipo pueden ser: aprender una metodología para el diseño, implementación y documentación de programas; apreciar el papel fundamental que juega la abstracción en la tarea de programar; conocer estructuras de datos, algoritmos y esquemas de uso general; introducir el paradigma de la programación orientada a objetos; y aprender JAVA como lenguaje de programación.

Para este ejemplo, el objetivo del proyecto global es implementar un juego de ajedrez. En el primer miniproyecto, se pretende que los alumnos sean capaces de aplicar los conceptos básicos de la programación orientada a objetos tales como: objetos y clases, encapsulamiento, herencia y polimorfismo. En este contexto, este miniproyecto requerirá la definición, de manera genérica, de las clases y subclases de objetos así como las interfaces principales que intervendrán en el juego, por ejemplo, las clases Tablero, Pieza, Reloj, Jugador, etc. El objetivo de este miniproyecto es familiarizarse con el lenguaje de programación Java de mano de un programa que implementa el movimiento de piezas del juego del ajedrez.

En el segundo miniproyecto se introducen elementos básicos del lenguaje JAVA [4]. En nuestro ejemplo, sería necesario escribir el código que describe los movimientos y toda la casuística de las distintas piezas del ajedrez: caballo, torre, reina, alfil, peón y rey; utilizando variables, tipos de datos, sentencias condicionales y bucles.

En el tercer miniproyecto, se incorporarían elementos tales como colecciones o estructuras ordenadas, así como instrucciones básicas de la entrada y salida en un lenguaje orientado a objetos. Asimismo se incorporaría la lógica del juego: orden de las jugadas, activación/desactivación de los relojes, ejecución de movimientos, etc. Para el uso de entrada y salida por pantalla, se mostraría el tablero de ajedrez después de cada movimiento, tal como indica la Fig. 2. El programa puede recibir también un fichero de entrada, que representa el tablero de ajedrez con la posición específica de las piezas en un momento determinado de la partida. Asimismo podrá almacenar el estado del juego igualmente en un fichero de salida.

```
TimeLimit[0] = 0
TimeLimit[1] = 0
1. Nf3

black KQkq
r n b q k b n r
p p p p p p p
. . . . .
. . . . .
. . . . .
. . . . N . .
P P P P P P P
R N B Q K B . R

Thinking...

white KQkq
r n b q k b . r
p p p p p p p
. . . . N . .
. . . . .
. . . . .
. . . . N . .
P P P P P P P
R N B Q K B . R

My move is : Nf6
white (2) :
```

Figura 2. Ejemplo de propuesta para el miniproyecto 3 explicado en el apartado 3.

El cuarto miniproyecto se centraría en la utilización de paquetes de interfaces gráficas de usuario que permita al alumno desarrollar la habilidad de utilizar sistemas de ventanas, componentes, estructuras visuales y desarrollar un sistema gráfico de interacción con el usuario. En este caso, se deberá implementar una interfaz gráfica para la aplicación del juego de ajedrez desarrollada en los tres miniproyectos anteriores. La interfaz permitirá visualizar el estado actual de la partida, así como introducir las acciones de los jugadores, ya sea en formato textual o través de la interfaz gráfica utilizando manipulación directa. También se capturarán los eventos correspondientes.

De este modo, tras la ejecución de los cuatro miniproyectos se obtiene como resultado un proyecto final con la implementación de un juego de ajedrez completo con interfaz gráfica tal y como muestra la Fig. 3.



Figura 3. Ejemplo de propuesta para el proyecto final explicado en el apartado 3.

## 4 CONCLUSIONES

El objetivo principal de este trabajo es proponer una metodología de enseñanza que palíe la alta tasa de suspensos y abandono durante los primeros cursos en las asignaturas de programación. La falta de motivación, el hecho de enfrentarse a una materia totalmente nueva, así como la dificultad para entender y desarrollar el concepto de abstracción, pueden ser algunos de los motivos de dichos resultados negativos. Así, se presenta un innovador método de aprendizaje basado en proyectos dinámicos que permiten la adquisición incremental de conocimientos y habilidades necesarios para la programación. Este tipo de estrategias motiva a los estudiantes, les hace partícipes y más activos en su propio proceso de aprendizaje.

La posibilidad de aplicar esta metodología en estas asignaturas se debe al carácter versátil y modularizable de la programación. Otra ventaja que presenta es su adaptabilidad a cualquier plan de estudios, número de créditos y titulación.

Tras la implantación de la experiencia piloto de estos procedimientos en la titulación de Grado en Biotecnología, se obtuvieron resultados muy satisfactorios que avalan nuestra propuesta para ser difundida.

## **REFERENCIAS**

- [1] Cernuda del Río A. Un estudio sobre el absentismo y el abandono en asignaturas de programación. *ReVisión*, Vol. 6(1), 2013.
- [2] Gayo JEL., Lanvin DF., Salvador JC., et. al. Una experiencia de aprendizaje basado en proyectos utilizando herramientas colaborativas de desarrollo de software libre. XII Jornadas de Enseñanza Universitaria de la Informática (JENUI 2006).
- [3] Aznar Gregori F., Arques Corrales P., Molina Carmona R., et al. Cambiando la metodología docente al aprendizaje basado en proyectos: la evaluación global. VIII Jornadas de Redes de Investigación en Docencia Universitaria "Nuevas titulaciones y cambio universitario", Alicante, 8-9 Julio 2010.
- [4] Meyer B. Construcción de software orientado a objetos. Prentice Hall, 1999.