

Capítulo 9. *La planificación del trabajo del estudiante y el desarrollo de su autonomía en el aprendizaje basado en proyectos*

Miguel Valero-García* y Juan J. Navarro**

(*) Escuela Politécnica Superior de Castelldefels

(**) Facultad de Informática de Barcelona
Universidad Politécnica de Cataluña

1. Introducción

El aprendizaje basado en proyectos (PBL, *Project Based Learning*) es el aprendizaje que se produce como resultado del esfuerzo que hacen los alumnos para desarrollar un proyecto. Se trata, por tanto, de un caso particular de aprendizaje basado en problemas y resulta especialmente adecuado en el contexto de las enseñanzas técnicas, puesto que la realización de proyectos es una labor esencial de los técnicos e ingenieros.

En un escenario de PBL los alumnos, organizados en grupos, reciben del profesor el “pliego de condiciones” del proyecto a realizar, que incluye la definición precisa del producto a obtener y las fechas de entrega de los resultados parciales y finales. Los alumnos deben entonces identificar qué es lo que ya saben y qué deben aprender para desarrollar el proyecto, establecen, con la ayuda del profesor, un plan de aprendizaje y lo llevan a cabo. Este proceso se repite hasta que se han alcanzado los objetivos de aprendizaje. En este contexto, lo que realmente importa es el aprendizaje que se produce en el proceso y no tanto el resultado final del proyecto. Esto es precisamente una de las diferencias importantes entre PBL y la tradicional realización

del proyecto de fin de carrera en los estudios de ingeniería, que pone el énfasis en la calidad profesional del resultado del proyecto.

El modelo PBL tiene sus fundamentos en las teorías del constructivismo (Piaget, 1970) y el aprendizaje social (Vygotsky, 1986), y existen precedentes ya bastante antiguos de su aplicación en la enseñanza universitaria (Woods et al., 1997). No estamos hablando, por tanto, de conceptos o métodos novedosos. No obstante, lo cierto es que el proyecto de convergencia europea (EEES), y en particular la adopción del sistema europeo de créditos (ECTS), está dando gran protagonismo a los métodos activos en general y al PBL en particular. La razón es que la adopción del sistema ECTS pone sobre la mesa del docente los dos retos siguientes:

- Diseñar un programa de actividades para realizar dentro y fuera de clase, de las que el alumno no pueda escapar sin haber aprendido.
- Conseguir que el alumno haga esas actividades.

En este contexto, los métodos PBL nos ofrecen, por una parte, pautas específicas para diseñar esos programas de actividades, es decir, para llenar de actividad significativa todas las horas de clase y fuera de clase que nos han sido asignadas en virtud de los ECTS de nuestra asignatura. Por otra parte, los métodos PBL introducen elementos de motivación que hacen más probable que el alumno recorra el camino que hemos preparado. Es bien sabido que los alumnos están más motivados y persisten más en el esfuerzo cuando trabajan en grupo para realizar un proyecto que perciben que está conectado con su futura actividad profesional. En definitiva, los métodos PBL inciden directamente sobre los dos retos que el sistema ECTS ha puesto sobre nuestra mesa.

Además de lo anterior, el modelo PBL también nos ofrece un marco adecuado para el desarrollo de ciertas habilidades transversales que constituyen otro de los retos más importantes para los docentes en el marco del EEES. El PBL facilita el desarrollo de habilidades tales como la comunicación oral y escrita, el trabajo en grupo y el aprendizaje autónomo.

Por último, en virtud de la naturaleza cooperativa de la actividad que se desarrolla en el proyecto, se enfatizan especialmente habilidades tales como descomponer trabajo en tareas, repartirlas equitativamente e integrar resultados;

comprometerse con unos plazos de entrega; consensuar acuerdos, y gestionar los conflictos que se producen entre los compañeros de grupo.

La dificultad más importante del PBL es que su puesta en marcha tiene un coste elevado de reorganización de la asignatura y de aprendizaje por parte de los profesores y de los alumnos. En particular, los alumnos necesitan una continuidad en el uso de métodos PBL para poder desarrollar las habilidades que pretendemos. Necesitan tener segundas y terceras oportunidades para hacer mejor lo que no hicieron bien en la primera ocasión.

Por este motivo, una implementación de PBL en una asignatura de pocos créditos de forma aislada en un plan de estudios quizá pueda ser un escenario adecuado para motivar a los alumnos en el aprendizaje de los temas del curso, pero no va a conseguir un impacto notable en el desarrollo de las habilidades tales como el trabajo en grupo o el aprendizaje autónomo.

No en vano, las implementaciones más exitosas de los métodos PBL consisten de organizar todo el plan de estudios, o incluso toda la universidad, entorno al modelo PBL, optimizando así el uso de los recursos. Este es el caso, por ejemplo, de la Universidad de Maastricht y de la Universidad de Aalborg (Kjersdam, y Enemark, 1994). En escenarios más modestos, los profesores de una secuencia de asignaturas que compartan el modelo PBL pueden coordinar sus esfuerzos para conseguir que los alumnos vayan progresivamente adquiriendo conocimientos y desarrollando habilidades.

En este trabajo nos interesa precisamente analizar cómo una secuencia de asignaturas, organizadas de acuerdo con el modelo PBL, puede desarrollar la autonomía del estudiante en el aprendizaje, y en particular, en el desarrollo de proyectos en grupo. Consideraremos tres escenarios correspondientes a diferentes niveles del plan de estudio y veremos cómo se despliegan varios elementos claves del modelo PBL para pasar de forma progresiva de un escenario muy guiado a otro en el que el alumno tiene mayores dosis de autonomía. Los modelos y propuestas se fundamentan en la experiencia acumulada por los autores en asignaturas de planes de estudio del ámbito de la informática y las telecomunicaciones, en la Universidad Politécnica de Cataluña.

2. Elementos clave en el diseño PBL y tres escenarios para su estudio

Cuando se organiza una asignatura utilizando el modelo PBL hay una serie de elementos clave que hay que considerar con cuidado. Entre ellos podemos mencionar los siguientes:

- *¿Cómo se forman los grupos de trabajo? ¿De qué tamaño? ¿Con qué criterios?*
- *¿Cómo son los enunciados de los proyectos?*
- *¿Cómo es el plan de trabajo de los alumnos? ¿Quién define los objetivos de aprendizaje? ¿Qué grados de libertad tienen los estudiantes?*

Las respuestas a estas cuestiones de diseño cambian a medida que el alumno está más avanzado en el plan de estudios, para así ayudarle a desarrollar su autonomía. Por ejemplo, el plan de trabajo para los alumnos no es igual en la primera asignatura en la que trabajan con estos métodos, momento en que deben encontrar un plan de trabajo muy guiado, que en asignaturas posteriores en las que el profesor propone un plan de trabajo con mayores dosis de indefinición, para que el alumno pueda desarrollar su autonomía.

Para analizar esta cuestión estableceremos tres escenarios hipotéticos, correspondientes a tres asignaturas que, por los temas tratados, podrían ser del área de la informática, aunque las ideas básicas se pueden aplicar a cualquier ingeniería:

1. una asignatura *inicial*, que los estudiantes cursan nada más entrar en la universidad,
2. otra *intermedia*, que cursan al siguiente cuatrimestre, y por último,
3. una asignatura *avanzada*, que cursan en uno de los últimos cursos.

Suponemos que las tres asignaturas son de cinco créditos ECTS, lo que supone que el alumno debe dedicar unas ocho horas de trabajo semanal durante las 16 semanas que dura el curso cuatrimestral (incluyendo los periodos de exámenes).

Aunque el escenario es hipotético, los autores se basan en su experiencia concreta en cuatro asignaturas que utilizan PBL. Una asignatura inicial, *Introducción a los Computadores*, y otra intermedia, *Laboratorio de Computadores*, ambas pertenecientes a la Ingeniería Técnica de Telecomunicaciones de la Escuela Politécnica

Superior de Castelldefels, y dos asignaturas avanzadas, *Algoritmos Paralelos*, del anterior programa de doctorado del departamento de Arquitectura de Computadores, y *Algoritmos y Modelos de Programación Paralela* del máster Computer Architecture, Networks and Systems. Todas estas asignaturas son de la Universidad Politécnica de Cataluña.

♦ ***Escenario inicial: asignatura de primer cuatrimestre***

El primer escenario es una asignatura obligatoria de introducción a la programación de computadores, que trata temas tales como estructuras de control básicas, ficheros, funciones, algoritmos de búsqueda y recorrido. En esta asignatura, de las ocho horas de trabajo semanal sólo dos de ellas son de clase. Los alumnos se organizan en grupos de clase de 40 alumnos. Por su parte, el proyecto se inicia en la semana 9 y finaliza en la semana 15 (las primeras 9 semanas son de formación inicial en programación y no se comentan aquí, ya que no siguen un modelo PBL).

La Tabla 1 muestra un esquema de la planificación de las sesiones de clase (Ai denota la sesión de trabajo en el aula de la semana i), y del trabajo que realizan los estudiantes en casa después de cada clase (la letra C denota los periodos de trabajo fuera de clase). Esta Tabla describe brevemente las tareas que realizan los alumnos junto con las entregas más importantes (columna de la derecha) y se usará como apoyo para las explicaciones de las secciones siguientes.

♦ ***Escenario intermedio: asignatura de segundo cuatrimestre***

Este es un escenario similar al anterior. Los alumnos profundizan en el aprendizaje de la programación de computadores con una organización docente igual a la de la asignatura anterior. En este caso, los temas tratados son: programación visual, orientada a objetos, algoritmos de búsqueda rápida y ordenación, gestión dinámica de la memoria y estructuras de datos avanzadas.

Tabla 1. Plan de actividades y entregas (semanas de la 9 a la 15) para la asignatura inicial

S		Descripción de las actividades	Entr.
9	A9 2h	<ul style="list-style-type: none"> ▪ <i>Presentación del proyecto.</i> El profesor presenta el enunciado y los criterios de calidad. ▪ <i>Reglas de funcionamiento.</i> Cada grupo discute sus reglas de funcionamiento. ▪ <i>Presentación del puzle.</i> El profesor explica brevemente cada parte y los alumnos se reparten los tres temas. 	#9.2
	C9 6h	<ul style="list-style-type: none"> ▪ <i>Estudio individual.</i> Cada alumno trabaja el tema que le ha sido asignado. 	
10	A10 2h	<ul style="list-style-type: none"> ▪ <i>Reunión de expertos.</i> Alumnos de grupos distintos pero que han trabajado el mismo tema aclaran sus dudas y realizan juntos ejercicios de profundización. 	#10.1
	C10 6h	<ul style="list-style-type: none"> ▪ <i>Preparación de explicación y materiales.</i> Cada alumno prepara ejercicios de aplicación de su tema para sus compañeros, con las soluciones a los ejercicios 	#10.2
		<ul style="list-style-type: none"> ▪ <i>Explicaciones en el grupo.</i> Cada alumno explica su tema (por turnos) y entrega a los compañeros de los ejercicios preparados con las soluciones. ▪ <i>Resolución de ejercicios de compañeros.</i> Cada alumno intenta resolver los ejercicios y prepara un informe de autoevaluación. 	#10.3
11	A11 2h	<ul style="list-style-type: none"> ▪ <i>Resolución de dudas.</i> Los grupos se reúnen para aclararse las dudas sobre el material y los ejercicios realizados. ▪ <i>Práctica guiada.</i> El profesor dirige una práctica que relaciona los tres temas del puzle. ▪ <i>Ejercicio de integración.</i> El profesor propone un ejercicio que requiere de los tres temas. Inicia su resolución en la pizarra. El ejercicio de integración es un primer paso hacia el primer prototipo del proyecto. 	#11.1
	C11 6h	<ul style="list-style-type: none"> ▪ <i>Acabar ejercicio de integración.</i> Los alumnos trabajan juntos para completar el ejercicio que inició el profesor en la pizarra. 	
12	A12 2h	<ul style="list-style-type: none"> ▪ <i>Demostración de ejercicio integración.</i> Los grupos muestran al profesor el correcto funcionamiento del ejercicio. 	#12.1
		<ul style="list-style-type: none"> ▪ <i>Asignación del ejercicio de integración individual.</i> Los alumnos se reparten los tres ejercicios individuales y realizan una primera planificación (cómo es el reparto y cuándo se reunirán para unir las tres partes). 	#12.2
	C12 6h	<ul style="list-style-type: none"> ▪ <i>Trabajo individual.</i> Cada alumno realiza su ejercicio individual. ▪ <i>Puesta en común.</i> El grupo se reúne para unir las tres partes y poner a punto el primer prototipo. ▪ <i>Diseño y planificación del segundo prototipo.</i> Los grupos preparan un documento en el que se reparten el trabajo de segundo prototipo y establecen fechas para las tareas. 	#12.3
13	A13 2h	<ul style="list-style-type: none"> ▪ <i>Demostración del primer prototipo.</i> Los grupos acaban el primer prototipo y hacen una demostración al profesor. ▪ <i>Revisión de planes.</i> El profesor discute con los grupos las planificaciones del segundo prototipo. 	#13.1
	C13 6h	<ul style="list-style-type: none"> ▪ <i>Tareas segundo prototipo.</i> Los grupos realizan las tareas del segundo prototipo, de acuerdo con su planificación. 	
14	A14 2h	<ul style="list-style-type: none"> ▪ <i>Prueba de mínimos.</i> Ejercicio individual en modo examen para verificar los conocimientos mínimos de la asignatura. 	#14.1
	C14 6h	<ul style="list-style-type: none"> ▪ <i>Tareas segundo prototipo.</i> Los grupos realizan las tareas del segundo prototipo, de acuerdo con su planificación. 	
15	A15 4h	<ul style="list-style-type: none"> ▪ <i>Puesta a punto del segundo prototipo.</i> Los grupos acaban el segundo prototipo y hacen la entrega. 	#15.1
		<ul style="list-style-type: none"> ▪ <i>Evaluación cruzada.</i> Cada grupo evalúa el proyecto de otros dos grupos, de acuerdo con los criterios de calidad. 	#15.2
		<ul style="list-style-type: none"> ▪ <i>Ampliación individual.</i> Cada alumno realiza un ejercicio individual de ampliación del proyecto, para demostrar que domina la globalidad del trabajo. 	#15.3
		<ul style="list-style-type: none"> ▪ <i>Revisión del funcionamiento de grupo.</i> Los alumnos rellenan un cuestionario en el que establecen objetivos de mejora para su próximo proyecto juntos. 	#15.4

La diferencia más importante es que en este caso el proyecto se inicia en la primera semana y se extiende hasta el final de curso. Los alumnos pueden empezar directamente con el proyecto desde el primer día de clase ya que disponen de los conocimientos básicos de programación adquiridos en la asignatura inicial. Para esta asignatura no hemos incluido ninguna Tabla con un resumen de actividades y entregas, porque las explicaciones del texto serán suficientes.

◆ ***Escenario avanzado: asignatura de últimos cursos***

Esta es una asignatura optativa de últimos cursos, con un solo grupo de unos 20 alumnos. Se tiene reservada un aula para realizar clases de dos horas de duración, dos días a la semana, durante las 15 semanas de clase del curso, aunque no todas las horas se usan para clase.

En esta asignatura los alumnos, que ya tienen una amplia experiencia en programación de computadores, aprenden a programar un supercomputador usando entornos de programación paralela. El objetivo final del proyecto es conseguir la máxima velocidad de ejecución en la resolución de un problema similar a los que aparecen en el ámbito profesional o académico. Para conseguirlo necesitan primero tener unos conocimientos básicos de programación paralela, tema nuevo para ellos. Por eso, como ocurre en la asignatura inicial, los conocimientos básicos se obtienen en la primera mitad del curso mientras que el proyecto propiamente dicho se realiza en la segunda mitad del curso.

La diferencia más importante con las asignaturas anteriores es que los alumnos tienen mucha más autonomía en la elección de los objetivos de aprendizaje y en la organización del plan de trabajo para desarrollar el proyecto.

La tabla 2 (A y B) muestra el esquema del plan de trabajo de algunas de las 30 sesiones de clase y de trabajo en casa. En la Tabla 2A, por ejemplo, A2.1 denota la primera sesión de clase de la segunda semana. A principio de curso el profesor proporciona la planificación de las actividades y entregas mostradas en la Tabla sin fondo sombreado. El resto de la actividad debe ser planificada por los estudiantes, como por ejemplo, las actividades sombreadas entre la segunda y sexta semana. Las semanas de la 8 a la 12 no se muestran, ya que las planifica cada grupo.

Tabla 2A. Plan de actividades y entregas para la asignatura avanzada (semanas 1-4)

S		Descripción de actividades	Entr.
1	A1.1 2h	<ul style="list-style-type: none"> ▪ <i>Presentación</i> de los profesores, la asignatura y el campus virtual (Moodle). ▪ <i>Presentación</i> de los estudiantes: motivación y experiencia en supercomputación. ▪ <i>Formación de grupos de 3 estudiantes.</i> ▪ <i>Presentación</i> del proyecto y de la planificación prefijada. ▪ <i>Consensuar</i> con el profesor y firmar el contrato de aprendizaje. 	#1.1 #1.2
	C1.1 2h	<ul style="list-style-type: none"> ▪ <i>Estudio</i>: ¿Cómo se definen los objetivos de aprendizaje? + ¿Cómo se trabaja cooperativamente y se gestionan los conflictos? ▪ <i>Hacer mapas conceptuales</i> de la lectura. 	#1.3
	A1.2 2h	<ul style="list-style-type: none"> ▪ <i>Reunión de grupo</i>: puesta en común mapas conceptuales. Realización mapas consensuados. ▪ <i>Presentación</i> del puzle sobre supercomputación. ▪ <i>Reunión de grupo</i>: asignación de cada pieza del puzle. 	#1.4
	C1.2 2h	<ul style="list-style-type: none"> ▪ <i>Búsqueda</i> de documentación de la parte del puzle asignada. ▪ <i>Lectura rápida</i> y selección de la información. ▪ <i>Propuesta de índice</i> del temario de la pieza del puzle asignada 	#1.5 #1.6
2	A2.1 2h	<ul style="list-style-type: none"> ▪ <i>Reunión de expertos</i> (en 3 aulas, una por pieza del puzle). Consensuar la planificación del trabajo del grupo de expertos hasta la sesión A6.2 (excepto entregas prefijadas #3.1 y #4.1). 	#2.1
		<ul style="list-style-type: none"> ▪ <i>Reunión de grupo de expertos</i>. Puesta en común de las entregas #1.5 y #1.6 y de la documentación encontrada. 	#1
		<ul style="list-style-type: none"> ▪ <i>Primera propuesta consensuada</i> de índice del temario y de documentación. 	#2
	C2.1 4h	<ul style="list-style-type: none"> ▪ <i>Búsqueda</i> de más documentación (si es necesario) ▪ <i>Estudio en profundidad</i> de la documentación del tema asignado. 	
	A2.2	<ul style="list-style-type: none"> ▪ <i>No hay clase</i> 	
C2.2 2h	<ul style="list-style-type: none"> ▪ <i>Preparar</i> propuesta de objetivos mínimos y ejercicios tipo del tema asignado (hay dos horas más para esta tarea en C3.2). 		
3	A3.1	<ul style="list-style-type: none"> ▪ <i>No hay clase</i> 	
	C3.1 4h	<ul style="list-style-type: none"> ▪ <i>Preparar y escribir</i> propuesta de objetivos mínimos y ejercicios tipo (con solución y criterios de corrección) del tema asignado. 	#3
	A3.2 2h	<ul style="list-style-type: none"> ▪ <i>Reunión de expertos</i> (en 3 aulas, una por pieza del puzle) ▪ Puesta en común de la nueva documentación encontrada (si necesario) y de la entrega #3. 	
		<ul style="list-style-type: none"> ▪ <i>Reunión de expertos</i> (en 3 aulas, una por pieza del puzle) ▪ <i>Consensuar y terminar</i>: Documentación, objetivos mínimos (consensuados con el profesor, hay 3 profesores en esta sesión, uno por aula), ejercicios tipo con su solución y criterios de corrección. Publicarlo al resto de compañeros en Moodle. 	#3.1
	C3.2 2h	<ul style="list-style-type: none"> ▪ <i>Lectura rápida de la documentación</i>, los objetivos mínimos y los ejercicios tipo que han preparado los expertos de las otras dos piezas del puzle. ▪ <i>Preparar explicación pieza 1 del puzle</i> (para los expertos de la pieza 1). 	
4	A4.1 2h	<ul style="list-style-type: none"> ▪ <i>Reunión de grupo</i>. Planificación del trabajo del grupo hasta la sesión A6.2. 	#4.1
		<ul style="list-style-type: none"> ▪ <i>Explicación y propuesta de trabajo de la pieza 1 del puzle.</i> ▪ <i>Hacer algunos ejercicios tipo individualmente y puesta en común.</i> 	
	C4.1 2h	<ul style="list-style-type: none"> ▪ <i>Estudiar y hacer los ejercicios</i> tipo de la pieza 1 del puzle. ▪ <i>Preparar explicación pieza 2 del puzle</i> (para los expertos de la pieza 2). 	
	A4.2 2h	<ul style="list-style-type: none"> ▪ <i>Reunión de grupo</i>. Explicación y propuesta de trabajo de la pieza 2 del puzle. ▪ <i>Hacer algunos ejercicios</i> tipo individualmente y puesta en común. 	
C4.2 2h	<ul style="list-style-type: none"> ▪ <i>Estudiar y hacer los ejercicios</i> tipo de la pieza 2 del puzle. ▪ <i>Preparar explicación pieza 3 del puzle</i> (para los expertos de la pieza 3). 		

Tabla 2B. Plan de actividades y entregas para la asignatura avanzada (semanas 5-15)

S		Descripción de actividades	Entr.
5	A5.1 2h	<ul style="list-style-type: none"> ▪ <i>Reunión de grupo.</i> Explicación y propuesta de trabajo de la pieza 3 del puzle. ▪ <i>Hacer algunos ejercicios</i> tipo individualmente y puesta en común. 	
	C5.1 2h	<ul style="list-style-type: none"> ▪ <i>Estudiar y hacer los ejercicios</i> tipo de la pieza 3 del puzle. 	
	A5.2	<ul style="list-style-type: none"> ▪ <i>No hay clase.</i> 	
	C5.2 4h	<ul style="list-style-type: none"> ▪ <i>Preparar examen de objetivos mínimos</i> (ejercicios tipo). ▪ <i>Consultas</i> a través del foro del grupo. 	
6	A6.1	<ul style="list-style-type: none"> ▪ <i>No hay clase.</i> 	
	C6.1 4h	<ul style="list-style-type: none"> ▪ <i>Preparar examen de objetivos mínimos</i> (ejercicios tipo). ▪ <i>Consultas</i> a través del foro del grupo. 	
	A6.2 2h	<ul style="list-style-type: none"> ▪ <i>Examen de objetivos mínimos</i> (ejercicios tipo). 	#6.1
	C6.2 2	<ul style="list-style-type: none"> ▪ <i>Preparar individualmente</i> ideas sobre las bases del concurso del mejor código. 	#6.2
7	A7.1 2h	<ul style="list-style-type: none"> ▪ <i>Reunión de grupo.</i> Consensuar en el grupo una propuesta de bases del concurso. Nombrar coordinador de grupo. ▪ <i>Reunión de coordinadores de grupo.</i> Consensuar entre todos y el profesor las bases del concurso del mejor código (criterios de valoración). 	#7.1
	C7.1 2h	<ul style="list-style-type: none"> ▪ <i>Preparar individualmente</i> ideas sobre la planificación del proyecto, hasta la primera mitad de la semana 13. 	#7.2
	A7.2 2h	<ul style="list-style-type: none"> ▪ <i>Reunión de grupo.</i> Consensuar en el grupo una propuesta de planificación del proyecto (46 horas de trabajo de cada estudiante del grupo) hasta la primera mitad de la semana 13. 	#7.3
8 a 12		<ul style="list-style-type: none"> ▪ <i>Desarrollo del trabajo de acuerdo con el plan de cada grupo</i> (en algún momento de este periodo se debe entregar un prototipo del código y un esquema de la memoria del proyecto) 	
13	A13.2 2h	<ul style="list-style-type: none"> ▪ <i>Entrega final del código</i> (con los scripts para su ejecución y obtención de medidas) y de la memoria del proyecto. ▪ <i>Asignación por el profesor</i> de las entregas #13.1 de los dos grupos que debe evaluar cada grupo. ▪ <i>Reunión de grupo.</i> Planificación del trabajo de evaluación de los dos grupos asignados, que se finalizará en la sesión A14.1. 	#13.1 #13.2
	C13.2 2h	<ul style="list-style-type: none"> ▪ <i>Preparar evaluación</i> para la sesión A14.1. 	#13.3
	A14.1 2h	<ul style="list-style-type: none"> ▪ <i>Evaluación entre compañeros.</i> Cada grupo evalúa la entrega final #13.1 de dos grupos, de acuerdo con las rúbricas. 	#14.1
14	C14.1 2h	<ul style="list-style-type: none"> ▪ <i>Preparar exposición</i> para la sesión A14.2, si corresponde. ▪ <i>Preparar examen</i> de objetivos mínimos, si corresponde. 	
	A14.2 2h	<ul style="list-style-type: none"> ▪ <i>Exposiciones del proyecto</i> de la primera mitad de los grupos. ▪ <i>Preguntas.</i> ▪ <i>Evaluación por los compañeros</i> (aplicando la rúbrica) 	#14.2 #14.3
	C14.2 2h	<ul style="list-style-type: none"> ▪ <i>Preparar exposición</i> para la sesión A15.1, si corresponde. ▪ <i>Preparar examen</i> de objetivos mínimos, si corresponde. 	
	A15.1 2h	<ul style="list-style-type: none"> ▪ <i>Exposiciones del proyecto</i> de la segunda mitad de los grupos. ▪ <i>Preguntas.</i> ▪ <i>Evaluación por los compañeros</i> (aplicando la rúbrica) 	#15.1 #15.2
15	C15.1 2h	<ul style="list-style-type: none"> ▪ <i>Preparar examen</i> de objetivos mínimos, si corresponde. 	
	A15.2 2h	<ul style="list-style-type: none"> ▪ <i>Examen de objetivos mínimos</i> (ejercicios tipo) para los que no superaran el de la sesión A12. 	#15.1
	C15.2	<ul style="list-style-type: none"> ▪ <i>Los alumnos no hacen nada.</i> ▪ <i>Los profesores hacen la evaluación final y sacan las notas</i> 	

La evaluación

Los tres escenarios tienen el mismo esquema de evaluación que se basa en tres componentes:

- *Entregas.* Puede asignarse hasta un 20% de la calificación por el mero hecho de realizar todas las entregas del curso a tiempo.
- *El proyecto.* Puede tener asignado un alto porcentaje de la calificación (50%), distribuido en dos entregas clave que se califican por separado (por ejemplo, un primer prototipo del proyecto y una versión final). La calificación es la misma para todos los miembros del grupo.
- *Conocimientos mínimos.* En las tres asignaturas se han establecido una lista de conocimientos mínimos (ejercicios típicos) que los alumnos deben demostrar durante el curso para aprobar la asignatura, con independencia del proyecto o de las entregas. Deben demostrar todos los conocimientos mínimos (por ejemplo, mediante un examen individual) y tienen varias oportunidades durante el curso. Una parte de la calificación (alrededor del 30%) se determina en función de la calidad de las respuestas y del número de oportunidades que se han necesitado para superar los mínimos. Pero en todo caso, si no se superan todos los conocimientos mínimos la asignatura está suspendida, aunque el proyecto esté bien y se hayan realizado todas las entregas a tiempo.

2.1. Formación de grupos

Cuando se organiza una asignatura con el modelo PBL debe prestarse especial atención a la formación de los grupos de alumnos (los equipos que realizarán los proyectos), porque esos grupos van a tener que trabajar a lo largo de varias semanas, dentro y fuera de clase, y comprometerse con unas tareas y unas fechas de entrega. De hecho, uno de los aprendizajes más valiosos en estos escenarios es justamente el aprender a trabajar en equipo y resolver los conflictos interpersonales que se producen por el camino. Veamos entonces algunos de los aspectos clave de la formación de grupos en función del escenario de aplicación.

◆ *Escenario inicial*

En la asignatura inicial los grupos se forman el primer día de clase, de forma aleatoria. Si bien es cierto que los alumnos siempre prefieren elegir ellos mismos a sus compañeros de grupo, también es cierto que suelen aceptar sin problemas que los grupos los forme el profesor si se les explica bien las razones de ello.

Aunque, desde nuestro punto de vista, el tamaño ideal para los grupos es de tres alumnos, en el escenario inicial formamos grupos de cuatro, porque es habitual que durante el curso algunos alumnos abandonen. Si formamos grupos de cuatro, estos abandonos hacen que vayan quedando grupos de tres, que pueden continuar trabajando. Se minimiza así la frecuencia con la que hay que rehacer grupos para evitar que queden grupos de sólo dos alumnos.

En el momento en que se forman los grupos los alumnos, que llevan pocas horas en la Universidad, tienen unos minutos en clase para conocerse y rellenar un cuestionario en el que deben identificar las franjas de tiempo libre que tienen en común fuera de clase y que podrán usar para reunirse cuando sea necesario. El grupo sólo se da por válido si sus miembros pueden identificar al menos cuatro horas semanales para esas reuniones. Si no es posible entonces se realizan los cambios de grupo necesarios. Esta es una cuestión clave porque nuestra experiencia indica que los grupos que no tienen tiempo para reunirse fuera de clase tienen muchas más dificultades para conseguir el éxito en el proyecto. Además, en ese cuestionario inicial los alumnos deben indicar su procedencia (bachillerato o ciclos formativos) de forma que podamos verificar que hay una sana mezcla de procedencias en todos los grupos. Esta verificación podría tener como consecuencia algún cambio de grupos para mejorar la heterogeneidad.

Durante la primera mitad del curso, en la que no se trabaja todavía con el proyecto, los grupos se usan esencialmente para tareas en clase (por ejemplo, ejercicios en grupo). Lo cierto es que cuando comienza el proyecto, en la novena semana, la situación de los grupos ya se ha estabilizado, de manera que todos los grupos son de tres o de cuatro alumnos.

◆ *Escenario intermedio*

En la asignatura intermedia los grupos se forman también el primer día de clase, y de nuevo una condición esencial para que el grupo pueda constituirse es que los miembros del grupo encuentren franjas de tiempo libre en común durante la semana para reunirse fuera de clase. No obstante, existen dos diferencias importantes en la formación de grupos en relación al escenario inicial.

En primer lugar, los grupos de trabajo son de tres alumnos. En este escenario los abandonos son menos frecuentes que en el anterior y el problema de que vayan quedando grupos de dos es menor.

En segundo lugar, los alumnos eligen ahora sus grupos de trabajo. Ya han tenido una experiencia en el escenario inicial y conviene que los grupos que han funcionado bien puedan seguir trabajando juntos para perfeccionar sus habilidades (y en particular, introducir las mejoras de funcionamiento identificadas por ellos mismos durante la experiencia previa). Por otra parte, los alumnos que no han tenido una buena actitud en el escenario inicial van a tener dificultades para encontrar compañeros de grupo, lo cual es una presión saludable para provocar un cambio en su actitud.

◆ *Escenario avanzado*

En la asignatura avanzada se matriculan estudiantes con algunas diferencias en su formación, ya sea porque han cursado algunas asignaturas optativas diferentes o porque han tenido diferentes experiencias de aprendizaje por motivos laborales o de interés personal. Por ello, y porque en esta asignatura hay menos estudiantes, antes de formarse los grupos se hace una rueda de presentación en la que cada uno indica su experiencia previa en la programación paralela y su interés por el tema. Si hay varios estudiantes con experiencia previa en supercomputación, el profesor se asegura de que esos estudiantes se repartan entre diferentes grupos. Excepto por esto, se deja a los alumnos que formen los grupos a su gusto (muchos ya se conocen y quieren ir juntos).

En esta asignatura los alumnos disponen de un aula reservada en horario lectivo para realizar dos sesiones de clase de dos horas cada semana. No obstante, en muchas ocasiones el aula no se usará para clase sino que está disponible para que los alumnos puedan hacer sus reuniones de grupo (de esta forma los alumnos tienen pocos

problemas para reunirse). Los grupos también son de tres estudiantes, ya que los abandonos son prácticamente nulos. Los grupos que se forman el primer día se usan tanto para el trabajo cooperativo de formación en supercomputación previo al proyecto, como para realizar el proyecto en la segunda mitad del curso.

2.2. Enunciado de los proyectos

El enunciado del proyecto es un elemento en el que ya puede graduarse el nivel de autonomía del estudiante, porque podemos ir pasando de enunciados cerrados en los que queda bien establecido lo que hay que aprender y lo que hay que producir, a escenarios más abiertos en los que los alumnos pueden elegir incluso una parte de los objetivos de aprendizaje. Veamos algunos detalles en el contexto de los escenarios objeto de nuestro análisis.

◆ *Escenario inicial*

En la asignatura inicial el proyecto es cerrado porque establece todo lo que hay que hacer, sin apenas grados de libertad. En concreto, el proyecto consiste en desarrollar un programa de ordenador que gestione información sobre personas, ofreciendo al usuario el típico menú de alta/baja/modificación/consulta de datos de personas. El enunciado del proyecto establece con detalle cuál es la información que debe tener asociada cada persona (nombre, edad, etc.) y cuáles son exactamente las opciones que debe tener la aplicación.

El enunciado también establece dos entregas básicas para el proyecto. La primera entrega es un primer prototipo que debe implementar un subconjunto de las opciones, y que debe estar listo en la semana 13 (entrega #13.1 de la Tabla I). La segunda entrega es la versión final con todas las opciones previstas inicialmente funcionando perfectamente. Esta entrega debe estar lista la semana 15 (entrega #15.1).

También como parte del enunciado se entregan los criterios de calidad para la evaluación del proyecto, en formato de rúbrica. La Tabla 3 muestra esa rúbrica. La columna de la izquierda indica los criterios de calidad y en cada fila se describen los tres niveles de calidad para cada criterio. La rúbrica contiene incluso los criterios que se usarán para determinar la calificación del proyecto.

Tabla 3. Criterios de calidad (en formato de rúbrica) para la evaluación y calificación del resultado de los proyectos (programas de ordenador)

Criterios para la evaluación de los prototipos			
	Nivel de calidad		
Criterio	Bien	Mejorable	Mal (0 puntos)
<i>Correcto (4 puntos)</i>	<ul style="list-style-type: none"> • La aplicación realiza correctamente todas las funciones previstas en el enunciado, y ha funcionado bien con todas las pruebas que he hecho (ha fallado como máximo una vez), y eso que he hecho muchas pruebas (4 puntos). 	<ul style="list-style-type: none"> • Falta una de las funciones previstas, o bien la aplicación ha fallado en tres o cuatro ocasiones (2 puntos). 	<ul style="list-style-type: none"> • Faltan dos o más de las funciones previstas, o bien la aplicación falla con mucha frecuencia. Está claro que no está bien.
<i>Robusto (1,5 puntos)</i>	<ul style="list-style-type: none"> • La aplicación resiste sin bloquearse todos los errores típicos que pueden aparecer. No he conseguido que se cuelgue (1,5 puntos). 	<ul style="list-style-type: none"> • Es razonablemente robusto. No es fácil que se quede colgado, pero en uno o dos casos se bloqueó (0,75 puntos). 	<ul style="list-style-type: none"> • La aplicación no es robusta en absoluto. Se queda colgada con frecuencia ante errores típicos.
<i>Amigable (1,5 puntos)</i>	<ul style="list-style-type: none"> • El usuario no tiene ninguna duda, en ningún momento, sobre cómo interactuar con la aplicación, qué datos debe entrar y cómo, y cómo interpretar los resultados y mensajes de la aplicación (1,5 puntos). 	<ul style="list-style-type: none"> • Los mensajes e información que da la aplicación son suficientes para trabajar bien. Sin embargo, en alguna ocasión he tenido algunas dudas sobre lo que hay que hacer o cómo hay que hacerlo (0,75 puntos). 	<ul style="list-style-type: none"> • El usuario tiene dudas constantes sobre lo que le está pidiendo la aplicación, y es difícil interpretar los resultados y mensajes en pantalla.
<i>Bien organizado y documentado (3 puntos)</i>	<ul style="list-style-type: none"> • El código está bien organizado. Es muy fácil encontrar el punto de la aplicación que hay que tocar para realizar alguna modificación en la funcionalidad. • Cada procedimiento y función tiene un comentario inicial que explica lo que hace, y cuáles son los parámetros. Además, los puntos del código especialmente complicados tienen un comentario suficientemente clarificador. • Las variables, procedimientos y funciones tienen nombres que ayudan a comprender para qué se usan. • El código está bien indentado (3 puntos). 	<ul style="list-style-type: none"> • Está razonablemente bien organizado y documentado, aunque en algún caso, la estructuración en procedimientos y funciones podría ser mejor. • Los comentarios son suficientes, aunque echo de menos alguna aclaración más en algún punto del código (1,5 puntos). 	<ul style="list-style-type: none"> • La estructura del código no tiene lógica, y no hay comentarios (o los que hay no clarifican nada). • El código no está bien indentado. • Sería incapaz de modificar este código para añadir alguna funcionalidad nueva o arreglar algún error.

Establecer los criterios de calidad desde el comienzo (en formato rúbrica o en cualquier otro) es esencial porque ayuda a los alumnos y a los profesores a ser más eficaces en su trabajo. En particular, es esencial evaluar el primer prototipo del proyecto en base a los criterios de calidad de forma que los alumnos reciban rápidamente información precisa sobre lo que deben mejorar de cara a la versión final. Por otra parte, los criterios de calidad se usan también en la evaluación de la versión final, que se realiza mediante la técnica de la evaluación entre compañeros (unos grupos evalúan el proyecto de otros) (Bangert y Peer, 2001).

◆ *Escenario intermedio*

En la asignatura intermedia el proyecto es mucho más ambicioso, no en vano se extiende a lo largo de todo el curso. De hecho, uno de los objetivos es presentar a los alumnos, el primer día, un reto que les parezca muy ambicioso (incluso fuera de sus posibilidades), porque sabemos que los retos ambiciosos son elementos de gran motivación. Por ejemplo, los proyectos utilizados en los últimos años han sido: una aplicación informática que ayude a resolver sudokus (incluyendo una opción de resolución automática), o una aplicación para gestionar las reservas de asientos en aviones. En ambos casos, los alumnos deben aprender a diseñar interfaces de usuario que les van a permitir hacer proyectos visualmente espectaculares y de aspecto profesional (otro elemento clave para la motivación).

Una novedad importante en el enunciado del proyecto de esta asignatura intermedia es que sólo se fijan a priori una parte de las funciones de la aplicación. Los grupos deben elegir algunas funciones adicionales que hagan que su proyecto sea más atractivo que los demás. Encontramos aquí, por tanto, unas primeras dosis de autonomía que ayudan a que los grupos se motiven más como resultado de esa adaptación de una parte del proyecto a sus intereses personales.

El proyecto también tiene dos entregas esenciales: el primer prototipo y la versión final, que juegan el mismo papel que en el escenario inicial. Los requerimientos del primer prototipo están ya prefijados en el enunciado pero el segundo prototipo recoge esas funciones adicionales que cada grupo ha elegido para hacer que su proyecto

sea el más atractivo. La rúbrica para la evaluación del proyecto es muy similar a la de la Tabla 3 pero tiene un criterio más que hace referencia a la calidad de las funcionalidades adicionales incorporadas por cada grupo.

◆ *Escenario avanzado*

El enunciado del proyecto de la asignatura avanzada es muy escueto. Se formula un problema típico de supercomputación y el objetivo es escribir un código que se ejecute sobre un supercomputador dado en el menor tiempo posible. Para aumentar la motivación de los estudiantes se realiza un concurso: ganará el concurso el grupo que obtenga mejores resultados. Las bases del concurso son inicialmente muy simples:

- Se formula el problema (por ejemplo: un problema numérico como la resolución de un sistema grande de ecuaciones por un método concreto, directo o iterativo, o un problema no numérico como la comparación de secuencias de aminoácidos mediante el algoritmo de Smith-Waterman¹).
- Se da el tamaño del problema y posiblemente los datos para un juego de pruebas.
- Se fija el supercomputador sobre el que se ejecutará indicando el número máximo de procesadores disponibles.
- Se determina el software de base que se puede usar (lenguajes, librerías, colas de ejecución, etc.)
- Por último, se establece el criterio para determinar qué código gana el concurso, que inicialmente es el mínimo tiempo de ejecución.

Además del concurso, también se evalúa la calidad del código y de la memoria del proyecto, para lo que se entregan, junto con el enunciado del proyecto, las rúbricas correspondientes. No obstante, los estudiantes pueden modificar las bases del concurso y las rúbricas siempre y cuando lo hagan por consenso (esto se hace en la entrega #7.1 de la semana 7 del curso, tal y como indica la Tabla 2). Por ejemplo, podrían acordar que gana el que obtiene el menor valor $8xT_8 + 16xT_{16} + 64xT_{64}$ siendo T_k el tiempo de

¹ En algunos casos se les da un código secuencial que resuelve el problema para que puedan comparar el resultado de su código paralelo y verificar que coincide con el de la ejecución secuencial.

ejecución usando k procesadores y ejecutando siempre el mismo código que se compila sólo una vez.

2.3. La planificación

Sin duda es en la planificación de las tareas del proyecto donde más claramente se ven las diferencias entre los distintos escenarios en lo que respecta a la autonomía del estudiante, porque pasaremos de un escenario en el que el plan de trabajo está completamente preestablecido por el profesor a escenarios en los que el plan de trabajo lo deciden los grupos (qué tareas van a hacer, cuándo las van a hacer y que van a entregar como producto de cada tarea). La autonomía de los estudiantes llega a su punto álgido cuando, en la asignatura avanzada, los alumnos formulan ellos mismos los objetivos de aprendizaje con los que serán evaluados.

2.4.1. La técnica del puzle para trabajo cooperativo

Un elemento que comparten los tres escenarios que estamos estudiando, y que conviene explicar de forma genérica antes de ver los detalles, es el uso de la técnica del puzle para la planificación de una parte de las actividades del proyecto.

El puzle consiste en dividir el material de estudio (nuevos temas que se requieren para avanzar en el proyecto) en tres partes razonablemente independientes de forma que cada miembro del grupo se convierta en un experto de uno de esos temas (en los grupos de cuatro alumnos habrá dos expertos de uno de los temas). El procedimiento genérico es el siguiente:

1. Los miembros del grupo se reparten los temas.
2. Cada miembro estudia de forma individual el contenido del tema asignado.
3. El profesor organiza una reunión de expertos en la que miembros de grupos distintos pero que han trabajado el mismo tema comparten (en grupos de dos o tres) sus dudas sobre el tema y hacen juntos algunos ejercicios de profundización.
4. Cada alumno prepara una explicación de su tema para los compañeros de grupo. Prepara también materiales de apoyo para los compañeros (por ejemplo,

ejercicios sobre el tema con soluciones para que sus compañeros puedan autoevaluarse).

5. El grupo se reúne para que cada alumno explique su tema a los compañeros
6. El grupo hace un ejercicio de integración propuesto por el profesor, que requiere conocimientos de los tres temas y, por tanto, las aportaciones de los tres expertos.
7. Cada alumno realiza una prueba individual evaluable en la que debe demostrar que ha adquirido unos conocimientos mínimos de cada uno de los tres temas (lo que incluye los dos temas de los que no es experto).

Este procedimiento, que se usa en los tres escenarios (aunque con diferencias que se explicarán en breve) pone de manifiesto un aspecto importante del modelo PBL: no todos los alumnos van a aprender lo mismo. Es obvio que en cada grupo habrá alumnos que sabrán más de unos temas (de aquellos en los que son expertos) que de otros (de los que los expertos son sus compañeros). Esto es imprescindible para que los grupos puedan hacer un proyecto ambicioso. Si resulta que todos los miembros del grupo estudian y aprenden lo mismo, entonces la percepción del alumno puede ser: “esto no es un trabajo en equipo, en un grupo de personas que hacemos tres veces lo mismo”.

No obstante, tal y como se ha explicado antes, el modelo incluye la posibilidad de que para cada tema del curso se establezcan unos conocimientos mínimos que deben ser adquiridos por todos los alumnos, incluso en los temas de los que no son expertos. Pero, insistimos, habrá que aceptar que no todos los alumnos podrán alcanzar los mismos objetivos de aprendizaje en todos los temas.

▪ *Escenario inicial*

El escenario inicial es un ejemplo en el que el plan de trabajo para desarrollar el proyecto está prefijado por el profesor, con poco margen para desviaciones, tal y como muestra la Tabla 1. Después de la presentación del proyecto (en la sesión A9) se pone en marcha un puzzle que abarca varias semanas de trabajo dentro y fuera de clase, siguiendo el patrón descrito antes. Los tres temas implicados en este puzzle son: funciones, ficheros y estructuras.

El ejercicio de integración, que se inicia en la semana 11, es en realidad una versión muy sencilla del primer prototipo del proyecto que requiere conocimientos de los tres temas abarcados en el puzle. El profesor inicia en clase (por ejemplo, con una explicación en la pizarra) la resolución del ejercicio de integración, tomando las decisiones esenciales en cuanto a estructuras de datos y funciones a realizar, y resolviendo algunas de estas funciones. Los alumnos deben reunirse durante la semana 11 para completar juntos el ejercicio de integración (esencialmente, añadir al código realizado por el profesor algunas funciones más de complejidad similar). El ejercicio debe estar listo en la sesión A12.

Acabado el ejercicio de integración, cada miembro del grupo, de forma individual, realiza un ejercicio de ampliación, a elegir entre tres (uno por cada miembro del grupo). Este ejercicio consiste en programar varias funciones de complejidad similar a las que se hicieron en grupo. La suma del ejercicio de integración más las tres ampliaciones individuales es prácticamente el primer prototipo que se entrega en la sesión A13. La primera planificación a la que se hace referencia en la sesión A12 consiste básicamente en decidir cuál de las tres tareas individuales hace cada miembro del grupo, cuándo estará lista y cuándo se reunirá el grupo (antes de la sesión A13) para integrar las tres partes y encarrilar el primer prototipo, que acabarán durante la primera mitad de la sesión A13.

Los grupos realizan la planificación de la versión final del proyecto durante la semana 12. Esta planificación está también muy guiada porque el profesor entrega una plantilla con la lista de tareas a realizar. Cada grupo debe decidir qué modificaciones deben hacer en las estructuras de datos para la versión final, quién hace cada tarea (esencialmente, qué funciones va a programar cada uno), y para cuándo debe estar lista cada tarea.

■ *Escenario intermedio*

El plan de trabajo en la segunda asignatura se inicia con un ejercicio intenso de programación en grupo (debe realizarse en una semana), que denominamos proyecto 0. Este proyecto se realiza con los conocimientos adquiridos en la asignatura anterior. Su función principal es ayudar a que los alumnos refresquen esos conocimientos, aunque

contiene ya algunos elementos que serán útiles de cara al proyecto importante del curso, que también se presenta en la primera sesión de clase.

En la segunda semana se inicia un puzzle que no tiene diferencias importantes con el del escenario inicial, excepto, naturalmente, los tres temas a estudiar (programación visual, orientación a objetos y estructuras de datos avanzadas). Tal y como ocurre en el escenario inicial, el ejercicio de integración es una versión preliminar del primer prototipo y consiste en aplicar los conocimientos de los tres temas nuevos para construir una nueva versión del proyecto 0 que hicieron en la primera semana. En esta versión nueva los alumnos incorporan una interfaz gráfica atractiva, aplican la filosofía de programación orientada a objetos e incorporan alguna funcionalidad que requiere el uso de pilas o colas (estos son los temas introducidos en el puzzle). En este caso, el ejercicio de integración se extiende a lo largo de la quinta y sexta semana. El profesor entrega una planificación en la que ya se han decidido las estructuras de datos a usar y la lista de tareas. Los grupos deben repartirse las tareas y decidir cuándo estará lista cada tarea. Es, por tanto, un ejercicio de planificación similar al que se hizo en el escenario inicial para la versión final del proyecto.

Una vez entregado el ejercicio de integración en la sexta semana, los grupos empiezan a trabajar con una mayor dosis de autonomía por lo que respecta a la planificación de su trabajo. En particular, para la elaboración del primer prototipo cada grupo va a decidir cómo descompone el trabajo en tareas (recordemos que el trabajo a realizar está preestablecido en el enunciado del proyecto), quién va a hacer cada tarea y para cuando estará. De nuevo, no hay mucho tiempo que planificar, porque el primer prototipo debe estar listo en la décima semana (la séptima semana se dedica casi exclusivamente a preparar y hacer el primer examen de conocimientos mínimos y la octava semana no hay actividades de la asignatura). Por tanto, hay poco margen para desviaciones y el profesor puede controlar bastante bien la situación. Lo importante es que en el plan de trabajo realizado por los grupos quede bien claro qué tareas deben estar concluidas durante la novena semana de manera que el profesor pueda, en la sesión de clase de esa semana, verificar que cada grupo está cumpliendo el plan que estableció.

El régimen de mayor libertad de planificación se intensifica para la elaboración de la versión final, que se extiende a lo largo de las últimas cinco semanas de curso. Recordemos que es en este momento en el que los grupos incorporan al proyecto las

funcionalidades que han elegido (además de otras prefijadas en el enunciado). De nuevo deben decidir qué tareas deben hacer, quién las hace y qué tendrán listo cada una de las semanas. La labor del profesor, a partir de ese momento, consiste en verificar semana a semana que cada grupo está cumpliendo su plan, sugiriendo revisiones de esos planes a medida que se vayan detectando desviaciones, y naturalmente, ayudando a los grupos cuando aparezcan dificultades y proponiendo ampliaciones ambiciosas a los grupos más avanzados. En definitiva, durante esta fase, la actitud del profesor es la misma que tendría en una empresa dirigiendo a sus equipos de ingenieros: *“no podemos permitirnos el lujo de que los proyectos acaben mal porque en ese caso el que suspende (quizá pierda el puesto de trabajo) soy yo”*.

▪ **Escenario avanzado**

La gran diferencia de la asignatura avanzada respecto a las anteriores es que aquí la autonomía de los estudiantes respecto de la elección de objetivos de aprendizaje y la planificación del trabajo es mucho mayor.

No obstante, la asignatura avanzada tiene un aspecto parecido a la inicial, en la que los estudiantes llegan sin saber nada de programación de computadores. En la asignatura inicial se usa la primera mitad del curso para estudiar, usando métodos docentes activos (pero no PBL), los rudimentos de la programación, antes de comenzar el proyecto propiamente dicho. En la asignatura avanzada los estudiantes no saben nada de programación paralela, por lo que también se usa la primera mitad del curso para que adquieran una formación básica en el tema, antes de comenzar el proyecto. A diferencia de la asignatura inicial, aquí se organiza un puzzle a gran escala para este objetivo. Los tres temas del puzzle están directamente relacionados con el proyecto (código paralelo, mezclando los modelos de programación OpenMP y MPI, para su ejecución en el mínimo tiempo en un supercomputador dado). Los tres temas del puzzle son:

- Arquitectura del supercomputador, software de base, medidas de rendimiento.
- Modelo de programación OpenMP.
- Modelo de programación MPI.

Este puzle termina, en la sexta semana, con un examen individual de objetivos mínimos sobre los tres temas del puzle. La integración de conocimientos se lleva a cabo en el proyecto que se inicia a continuación.

Un aspecto nuevo en esta asignatura es que los estudiantes consensuan entre ellos y con el profesor los objetivos de aprendizaje del puzle: lo que deben saber, que es lo mismo que se les preguntará en el examen de conocimientos mínimos en la sexta semana (o en la última semana de curso si no lo han superado en la sexta). Los objetivos mínimos de aprendizaje los consensuan en la tercera semana (Tabla 2: sesión A3.2, entrega #3.1), después de haber tenido una pequeña formación de cómo formular objetivos operativos (específicos, evaluables) en la primera semana, haber estudiado cada uno su tema del puzle, y haber preparado documentación y ejercicios sobre el tema para sus compañeros en la segunda semana.

El nivel de autoaprendizaje en esta asignatura también es mayor que en las anteriores. Aquí los estudiantes expertos en cada tema deben buscar, clasificar y elegir la documentación que usaran para estudiar. También deben preparar ejercicios tipo (con su solución) para que sus compañeros se ejerciten en los objetivos mínimos (los ejercicios del examen individual serán similares a estos).

Por último, en esta asignatura, se aumenta el nivel de autonomía respecto a la planificación del trabajo por parte del estudiante. El estudiante recibe planificadas el primer día de clase únicamente las actividades y las entregas que se muestran sin fondo sombreado en la Tabla 2. Las actividades y entregas de la segunda semana y primera mitad de la tercera, que se muestran en la tabla con el fondo sombreado, son un ejemplo de planificación de un grupo de expertos, que han realizado ellos mismos al inicio de la sesión A2.1. De igual forma, las actividades sombreadas desde la cuarta semana hasta la primera mitad de la sexta son un ejemplo de las actividades planificadas por un grupo para terminar el puzle. Estos ejemplos de planificación se han incluido aquí para documentar mejor este trabajo, pero inicialmente no se les darían a los alumnos.

Las únicas exigencias de planificación del proyecto y entregas obligatorias son las de la semana 7, para consensuar las bases del concurso, y la planificación inicial que debe hacer cada grupo sobre cómo desarrollará su proyecto y las de las últimas dos semanas y media del curso para evaluar el proyecto (ver Tabla 2). La fase de realización del proyecto (semanas de la 8 a la primera mitad de la 13) queda a criterio de cada grupo, y por eso no se muestran en la Tabla. De nuevo, la tarea del profesor durante

estas semanas consiste en supervisar el trabajo de los grupos, verificar que avanzan de acuerdo con los planes establecidos y ofrecer ayuda cuando sea necesario.

Finalmente, en las últimas semanas se realizan las tareas correspondientes a la presentación oral de los proyectos realizados y a la evaluación entre grupos de esos proyectos.

3. Conclusiones

¿No estaremos siendo excesivamente paternalistas? Esta es una crítica habitual entre el profesorado a la vista de planes de asignatura tan guiados y detallados como el mostrado en la Tabla 1 para el caso de la asignatura inicial. Es un debate que se ha puesto sobre la mesa desde el momento en el sistema de crédito europeo nos ha hecho centrar nuestra atención en la planificación del trabajo del alumno.

Naturalmente, nuestra conclusión es que, si queremos conseguir de nuestros alumnos niveles de autonomía como los que se ponen en evidencia en escenarios como la asignatura avanzada descrita en nuestro trabajo, es imprescindible empezar con planes de trabajo muy guiados como los de la asignatura inicial. El proceso es el mismo que cuando, al plantar un árbol joven, lo sujetamos a una estaca bien fuerte para que crezca recto durante sus primeros años, de manera que podamos retirar más adelante la estaca con la confianza de que ya no se va a torcer.

En nuestro trabajo hemos centrado la atención en el desarrollo, dentro del marco del modelo PBL, de la autonomía del estudiante en la planificación del trabajo o la decisión de los objetivos de aprendizaje. Exactamente igual habría que proceder en el desarrollo de su autonomía en otros ámbitos tales como la preparación de informes escritos o la realización de presentaciones orales en público.

Naturalmente, escenarios como el descrito en este artículo requieren altos niveles de coordinación entre el profesorado, a lo largo del plan de estudios, de manera que se compartan criterios y métodos que ayuden a los alumnos a mejorar en una asignatura lo que no hicieron bien en la anterior. Desgraciadamente, en general, tenemos esta capacidad de coordinación poco desarrollada, porque no ha sido un requerimiento especialmente importante en los modelos tradicionales de enseñanza universitaria. Ahora todos tenemos una oportunidad inmejorable para revisar los

modelos organizativos con la excusa del diseño de los nuevos títulos de grado, basados en competencias y adaptados al sistema ECTS. Esperemos que esta oportunidad no se desaproveche.

Referencias

- Bangert, A.W. y Peer, A.W. (2001). Assessment: A Win-Win Instructional Strategy for Both Students and Teachers. *Journal of Cooperation and Collaboration in College Teaching*, 10 (2), 77-84.
- Kjersdam, K. y Enemark, S. (1994). The Aalborg Experiment. Project Innovation in University Education. *Aalborg University Press*. Disponible en <http://adm.aau.dk/fak-tekn/aalborg/engelsk>
- Página web sobre la técnica del puzle: <http://jigsaw.org/>
- Página web sobre rúbricas: <http://rubistar.4teachers.org/index.php>
- PBL en la Universidad de Maastricht: <http://www.unimaas.nl/pbl/>
- Piaget, J. (1970). *The Science of Education and the Psychology of the Child*. New York: Grossman.
- Vygotsky, L. (1986). *Thought and language*. Massachusetts: M.I.T., Massachusetts.
- Woods, D.R., et al. (1997). Developing problem-solving skills: TheMcMaster problem solving program. *Journal of Engineering Education*, 86 (2), 75-91.