Project-based Learning Experience on Data Structures Course

Carolina Gallardo Pérez, Pilar Martínez García, Jesús Sánchez López
Department of Information Structure and Organization
Escuela Universitaria de Informática - Universidad Politécnica de Madrid (EUI-UPM)
Madrid, Spain
{cgallardo,pmartin,jsanchez}@eui.upm.es

Abstract—**This paper presents the innovations in the practical work of the Data Structures subject carried out in the last five years, including a transition period and a first year of implantation of the European Higher Education Area. The practical coursework is inspired by a project-based methodology and from 2008/2009 additional laboratory sessions are included in the subject schedule. We will present the academic results and ratios of the mentioned time period which imply a significant improvement on students' performance.**

Keywords- ***data structures; EHEA; laboratory work; active learning; project-based learning***

I. INTRODUCTION

More than a decade ago, in 1992, P. Denning, in his seminal article Eduction a New Enginner [1] anticipated the necessity of deep changes in university and engineering education. The article highlighted different aspects which can be grouped within a common paradigm: *the need to evolve from a technology-oriented approach to a socio-technical one*.

Much has happened since then that have helped to realize the ideas of Denning. Two of them are particularly relevant to what is exposed in this paper:

- The high penetration of Internet in society and the emergence of tools to support asynchronous learning, especially in the form of b-learning (blended learning).
- The creation of the European Higher Education Area (EHEA), which has its origins in the so-called "Bologna Declaration" which, although geographically limited to the European Community, has become an enterprise of global dimension.

The Technical University of Madrid (UPM), one of the most important technical European universities, to address both challenges, has developed two strategic plans: a first one for the development of formal qualifications (degrees) adapted to the EHEA guidelines, and a second one for the incorporation of technology in the teaching and learning processes, along with the adoption of new methodological frameworks for assessing the acquired knowledge, skills and competences of their students. UPM created a specific Educative Innovation unit to support these issues (http://www.upm.es/innovacion/).

This article summarizes the most noteworthy actions framed under the aforementioned context from 2005 till now in the Data Structures course, located in the second semester of the first year course of the different qualifications offered at the School of Informatics of UPM (http://www.eui.upm.es). The time period we are going to describe spans from 2005/2006 to 2009/2010. This period is particularly interesting since, apart from the fact that there is available data, it knew the transition to the new degrees of the European Higher Education Area. Besides, there is also a local factor particularly affecting the Schools of Informatics (EUI) that shaped the transition to EHEA. The last years in EUI were defined by a worrying situation caused by a decreasing number of students, high drop-out student rate and lack of student motivation [2]. As mitigation measures, the governing bodies decided to implement measures such as the reduction of the size of students' groups and implantation of the European Credit Transfer System (ECTS) [3] as preparatory actions previous to the implantation of the new degrees in the academic year 2008/2009.

As paradigmatic of educative innovations, introduction of new technologies and as a shift to a student-centred educative process, this article is focused on the evolution of the practical work of the Data Structures course, as a result of the adaptation to the students' needs as well as the adaptation to the new degrees curricula. Subsequent changes in students' assessment and student-teacher interaction schemas are also analysed. Finally, it is shown the impact of these changes on students' performance, learning objectives and competence achievements, as supported by available data which suggests a subtle improvement in the students' performance ratios (in particular, the increasing percentage of students that pass the subject and the decreasing number of drop-outs) before the implantation of the EHEA degrees in contrast with a significant improvement of these ratios in the first year of the implantation of the new degrees.

II. DESCRIPTION OF PRACTICAL COURSEWORK

The time period at study knows two main programmes of study: Technical Engineering on Computer Science and Degree on Software/Computer Systems Engineering. In the former Technical Engineering qualification, the Data Structure course is allocated with 7.5 credits (4.5 practical ones and 3 for theory, equivalent to 2 hours of theory in a classroom setting and the 3 hours of student practice –not in the classroom –) or 5 ECTS. In EHEA degrees, Data Structures is allocated with 6 ECTS credits (equivalent to 156 hours of study, 2 hours of classroom and 2 hours of laboratory).

In the studied time-span, practical coursework undergoes a number of major changes in its design: from an initial problem-solving methodology [4] to a final situation characterized by a project-based methodology [5].

A. Initial situation: Problem-based methodology

Up to the 2005/06 academic year, the design of the practical coursework followed the classical problem-solving paradigm. The practicum is divided in three independent parts that are reviewed and scored independtly from each other and it is necessary to pass all three parts. In the first assignment, students have to implement an Abstract Data Type (ADT) according to some given specifications. In the second one, they are asked to implement the stack and queue ADTs with a given technology (as a double linked list with header and sentinel or a circular linked list, for example). The last one consists in the implementation of a binay search tree using dynamic technology. Apart from the implementation of the ADTs (in the Pascal language), the practicum is completed with exam problems, usually from previous years, which they have to solve and run on their own ADTs.

Students must form groups of two people. The estimated time for completion of the three parts is around 100 hours of common work. For assessment, they have to write a final report and deliver the source code of their ADTs and problems.

While the contents and methodology of the theory part of the Data Structure subject are fairly stable, it was decided to drastically change the design of the practical coursework and adopt the project-based learning approach. Several reasons motivated this shift: on the one hand, the incorporation to the faculty members of teachers with previous experience in Software Engineering courses and expertise in project-based methodologies; on the other, the decrease in the number of students and the subsequent reduction of students' groups favoured a more personal relationship between student and teacher. Besides, project-based teching-learning encourages desirable skills in students [6] like analysis, synthesis, knowledge and procedure transference to other contexts, critical thinking, groupal and individual responsibility, oral and written expression, and teamwork planning, organization and decision making.

Therefore, the faculty members decided to adopt thereafter the project-based approach.

B. Target situation: Project-based methodology

As already said, since the 2006/07, practical coursework follows the project-based model, although with certains limitations due to target *audience*: first year students in a very early stage of their training as engineers. The main manifestation of this limitation lies in the process of **planification**. Students do not participate at all in the planning activities (which are made by the faculty members) but are at charge of implementation, (partially) monitoring and closing activites.

The *project* consists in the implementation of an application which makes use of several units or ADTs. Initially, the whole project is divided in three different and independent assignments. In the first one, students are asked to build two ADTs and the application that uses them. Each ADTs is built as a Pascal unit that can be independently compiled and exchanged. The proposed ADTs were *Person* (implemented as a pointer to a record type) and the *Agenda* (implemented as ordered linked list of ADTs Persons) and an application to manage personal agendas. Fig. 1 graphically shows the module design of the whole project.

In the second assignment, the internal implementation of the ADTs has to be modified and the application is enriched with new functionalities. Essential changes are:

- The ADT Person is not a pointer to a record but a linked list of fields (with name and value).
- The ADT Agenda becomes a circular qualified ordered linked list of items of type Person.

Finally, the third assignment modifies the ADT Agenda so that it implements a binary search tree.

From 2007/08, it is considered more interesting to trace a parallel way between theory and practice: first, how to use the ADTs is taught, and then how to build them. Following this scheme, in the first assignment, students are asked to build the application that handles the provided compiled ADTs. In subsequent deliveries, they are asked to build compatible ADTs according to different implementations.

As an important innovation, the figure of the tutor is introduced. From the very beginning of the project, each student team is assigned a single tutor to assist them if needed throughout all the processes, which takes all the semester.

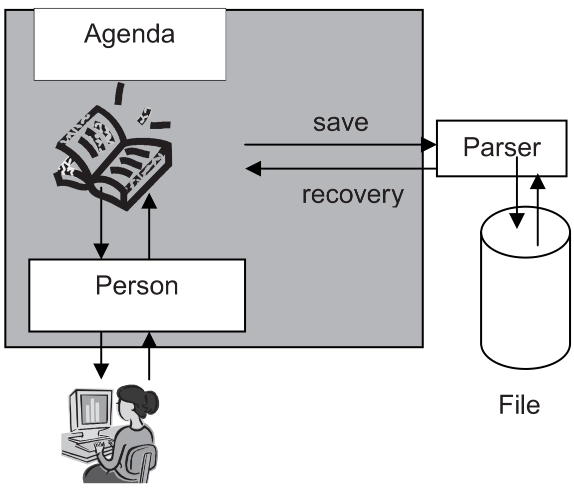

Figure 1. Module architecture for project.

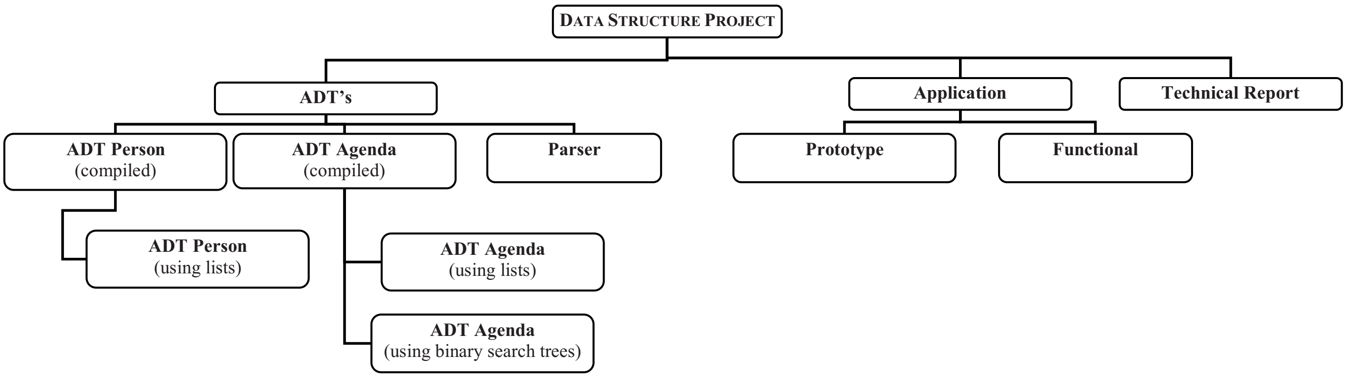

Figure 2. Work Breakdown Structure

From the academic year 2008/09, it was decided to guide students more closely in almost every step and to clear cut the involved tasks. The result is a practical coursework with the following characteristics:

1. Practical coursework is explicitly presented as a project (under the PMBok assumptions [7]) that is carried out along the entire course semester and that is divided in four milestones, the last one occurs as the result of the execution of the project closure process, with the presentation and defence of the deliverable material and public presentation of the results.

2. Work teams are made up of two students. They **do not participate in the planning of the project** (which is done by the teaching staff) but do know the project scope, objectives and specifications, the *Work Breakdown Structure* and the Gantt diagram. They are asked to gather the necessary data in order to incorporate them to the initial planning of the time cost of the activities defined in the Work Breakdown Structure.

   In the time planning, there are pieces of information that are unknown to students that will have to obtain throughout the development of the project, such as the duration of activities (effort in hours of work) and the distribution of the work and roles among the team members, that is, the *Organization Breakdown Structure.* Students are asked to fill in a formulaire (that should be delivered to the tutor on a weekly basis) to note down their dedication time to the project. The structure of the formulaire is shown in table I.

TABLE I. STUDENT FORMULAIRE

Student:				
Date	***Time***	***WP***[a]	***Function***[b]	***Description***

a. WP (Work Package): ADT Agenda, ADT Person; Application ….

b. Function = Analisys; Programming; Testing; Documentation; Communcation; Others.

These data will be used to elaborate the effort metrics as gathered in the table II.

TABLE II. STUDENT METRICS FORMULAIRE

Group			
	(Student 1)	**(Student 2)**	**TOTAL**
[ANL]			
[PRG]			
[TEST]			
[DOC]			
[COM]			
[OTH]			
TOTAL			

3. The software product resulting from the execution of the project is inspired by an object-oriented paradigm:
   a. They are asked to build a non functional prototype according to some specifications.
   b. They are asked to build the functional application with provided ADTs (already compiled).
   c. They substitute the provided ADTs for other built by themselves according to the different techniques explained in the classroom.

   In the light of this, the structure of the complete Work Breakdown Structure is graphically shown in Fig. 2.

4. It is emphasize the importance of the plan and execution processes of software testing in its black box version. The evaluation of software quality is considered as essential. Since in later years of the Degree, this problematic is dealt with more scientific and methodological rigour, our only intention is to raise a sense of awareness for software quality issues in students. To do that, the following tasks are included:
   a. Design of testing plans for each software component (compiled ADTs, built-in ADTs and application) as well as for the integration of components. Each test is documented in a formulaire gathering the following information: identification of the test, test objective,

test program, test data set, and involved software components.
   b. Execution of test plans. The results of the tests are recorded in formulaires.
   c. Elaboration of an executive report. It is a document that summarizes the most relevant conclusions of the process.
5. Awareness of issues related to the product cost. In this sense, students are expected to be able to measure (currently, almost intuitively) the cost of the final product. Number of lines of code and number of number of modules are used as metrics.

III. MANAGEMENT ISSUES AND EVALUATION

In essence, when we talk about innovative education we have a mind a collection of instructional techniques and methodologies that imply new patterns and roles in the student-teacher interaction as well as different ways to assess students. In essence, the innovations in the Data Structures course can be viewed as the evolution from almost an anonymous and autonomous process to a personal monitoring of the student.

A. Autonomous process

Up to 2006/2007, the predominant situation is defined by traditional (teacher-centred) lectures for theory and complete depersonalization of the student guidance and assessment. In fact, when students have to be evaluated for the practicum, they are assigned a turn number to be used to request practice evaluation. Each subpart of the practice can be reviewed by different teachers.

During 2006/07, the *Moodle* educational platform is introduced as a repository of materials and as a centralized platform for students to hand out their deliveries. Different forums are opened for students' questions on the subject. Still, evaluation of the practicum is done like in previous years: each subpart can be evaluated by different teachers and the final mark is the average mark of all suparts.

Students' performance results of this period (as representative of a series of previous years of autonomous work) can be seen in Table III. This table shows the percentage of students that passed the subject together with two ratios: passing students to active[1] students, and passing to the total number of enrolled students in the subject.

TABLE III. RESULTS FROM AUTONOMOUS WORK

	2005/06	2006/07
Active students	44 %	32 %
Pass/Active	23 %	21 %
Pass/Total	10 %	7 %
Total number of students	904	672

[1] We consider active students as those who attend the final exam in the ordinary period (usually June exams).

As can be seen, the introduction of a project-based methodology in the subject causes a subtle decrease in the analyzed ratios (including the percentage of students that finish up the practical coursework and students that pass the subject). We consider as possible causes of this initially discouraging results the dramatic change in perspective that the new methodology means, both from a quantitative point of view (completion of the practical coursework demands more time and dedication) and a qualitative point of view (practicum is more than a programming exercise for students, it implies the acquisition of traversal competences). As the main corrective measure, the faculty members commit themselves for a more comprehensive and personalized monitoring of the student in the following, as described in the next section.

B. Monitorized Process

From 2007/08, each student team group is assigned a tutor, who will be responsible for tutoring and assessing the practice. Materials in *Moodle* are expanded, including self-evaluation questionnaires that go *a la par* with the theory contents. Some questions require teacher's feedback, which is given by the tutor of the students group. In addition, students tutorized by the same tutor have at their disposal a forum chaired by their tutor. In this period, the totality of the practical course is assessed by the *same* teacher throughout all the semester, who evaluates and marks the work. The interaction between students and teacher is characterized by a mixture of personal face-to-face interactions and use of the different possibilities supported by b-learning.

In the 08/09 year, three groups out of seven have allocated four weekly classroom hours: 2 hours for traditional lectures and 2 hours in the laboratory. In this way, the teacher is tutor and almost witness of the student work. The relationship between teacher and student becomes much more fluid and personalized. In 09/10, all groups are allocated a 4-hour schedule.

Students' performances data and ratios of this period are shown in table IV. When compared with Table III, it can be observed a substantial imprevement in students' perfomance from year 2007/2008 onwards, and a growing trend over the next two years as long as the tutorization process is improved and a better degree of coordination between teachers is achieved. Needless to say, teachers' effort and dedication to the subject significantly increases.

C. Students and teachers' perceptions

It is also interesenting to analyse the perception of students and faculty members regarding the changes and introduced innovations in the subject. Students (as suggested from message forums, technical reports and personal communication) consider the practical work too demanding, but they point out that the experience has helped to them to understand the need for the division of tasks and teamwork. They also acknowledge that the project-based practical assignments help them to better understand the concepts explained in lectures.

Apart from any subjective question, although the amount of student work commited to the realization of the practical work could be in principle objectivelly measured, it is not so easy in practice. Students are asked to write down the time devoted to the realization of the different tasks in the project, but they do not seem to do so: it even seems that at the end they note down a random metric, as suggested in Tables V and VI. Table V shows the summary of the metrics of a student group in 2007/2008, while Table VI shows the data from students surveys about the time devoted to the whole Data Structure course (with only 41 observations). As can be seen, presented data is quite unrepresentative: Table V shows an extremely high standard deviation, with too distant minimum and maximum values. From this extremely high deviation, we can only infer that the student is not methodic when collecting (own) data for metrics. Thus, we cannot rely on elicited from students.

TABLE IV. RESULTS FROM MONITORIZED WORK.

	2007/08	2008/09	2009/10
Active Students	44 %	44 %	64 %
Pass/Active	51 %	73 %	73 %
Pass/Total	23 %	32 %	47 %
Total number of students	503	350	220

On the other hand, the faculty consider the practicum as quite demanding but quite effective to consolidate core concepts like recursion, use of ADTs and dynamic technologies. Besides, students that successfully finish the practical coursework manage to successfully pass the theory exam.

From another perspective, students respond quite well to the project-based methodology, but they present recurrent errors in the testing phases and documentation.

TABLE V. STATISTICS FROM A STUDENTS GROUP

	Alleged hours
Average	212,5
Maximum value	560
Minimum value	70
Standard Deviation	109,07

TABLE VI. RESULTS FROM STUDENTS' SURVEY A.C. 2009/2010

Alleged dedication	Abs F	Rel F
Less than de 25 hours	4	9,76
Between 25 and 50 hours	13	31,7
Between 50 and 75 hours	11	26,83
Between 75 y 100 hours	5	12,19
More than 100 hours	8	19,51
TOTAL	41	100

IV. PILOT GROUP: TRANSITION TO EHEA

Actually, the Data Structure course has been in permanent evolution in the last years disregarding of the compulsory changes due to the EHEA implantation, always seeking for students' excellence. In fact, 2008-2009 supposes a bridge between the extinction of old technical engineering degrees and EHEA adapted degrees. In this year, the ECTS credit system was introduced in three groups of the extinct pogrammes of study, hereafter referenced as *pilot groups*.

In the pilot groups, the 2+2 scheme is introduced: 2 hours of traditional lectures + two hours of laboratory work). These changes affect the organization of the course in the following aspects:

- Students carried out the practicum in the laboratory with the presence of the teacher, thus, work is fully guided from the beginning.
- Although usually an impopular policy, students' attendance is tracked in order to get the student more involved in the subject.
- Students are provided with exercises to be individually done and receive teacher feedback.

The most remarkable issue of the pilot groups is the chosen methodology for evaluation based on continuous assessment. Final evaluation of the student takes into account the final mark on the practicum and theory exam (as always) together with self-questionnaires in *Moodle*, exercises proposed by the teacher in the classroom and lectures attendance.

Despite using the same methodology in the three pilot groups, results are quite heterogeneous: some present good results while others can be considered a *disaster*. Table VII is a summary of the results obtained by the pilot groups.

As can be seen, there was a high dispersion among the three groups allowing for the following hypothesis:

- We cannot rule out the improvement in academic performance as a consequence of the implementation of the described measures, so we cannot exclude the possibility to adopt them on a general basis in subsequent courses.
- We have to analyse the causes of the worst results and introduce the appropriate corrective measures.
- It is considered a good idea to introduce the pilot experience since, in case of keeping the same proportions in a generalized situation; we would have arrived at an undesirable situation.

TABLE VII. RESULTS OF PILOT GROUPS

	Pilot 1	Pilot 2	Pilot 3	Average
Pass/Active	88%	68%	25%	71%
Pass/Total	54%	41%	13%	47%
% Class attendance	61%	47%	47%	52%
% Class attendance of students that pass	71%	60%	77%	67%
% Class attendance of inactive student	47%	35%	24%	34%

TABLE VIII. RESULTS OF ALL STUDENT GROUPS IN 2009/2010

	C1[1]	C2	S1[2]	S2	S3	Aver.
Pass/Active	82%	74%	52%	69%	52%	73%
Pass/Total	41%	57%	29%	44%	29%	47%
% Class attendance	49%	66%	47%	62%	60%	57%
% Class attendance of pass students	71%	82%	87%	78%	80%	83%
% Class attendance of inactive student	29%	38%	10%	46%	34%	29%

1. *C* stands for groups belonging to the Degree on Computer Systems Engineer group.
2. *S* stands for groups of the degree on Software Engineer.

Consequently, it was decided to implement the innovations introduced in the pilot groups for all groups in the EHEA context, once taken the appropriate remedial actions. Table VIII shows the greatly improved results obtained in all groups in 2009/10.

As happened in the pilot groups, there exists a direct relation between class attendance and the number of passing students.

V. RESULTS AND CONCLUSIONS

As the main objective measure for success, we are considering the pass and active ratios. Table IX shows the complete evolution of the ratios of the time period at study, they are graphically shown in Fig. 3. This table shows a strong upward tendency in the two ratios under consideration.

TABLE IX. RESULTS FROM STUDENTS' SURVEY A.C. 2009/2010.

	05/06	06/07	07/08	08/09	09/10*	Media
% Pass	0,44	0,32	0,44	0,44	0,64	0,46
% Pass/Active	0,23	0,21	0,51	0,73	0,73	0,49
% Pass/Total	0,1	0,07	0,23	0,32	0,47	0,24

The table shows a fairly stable line on the number of presented ration (around 43%) with the exception of the last academic year (increase of 20%). At this point, we can establish a logical turning point between the pre-EHEA and EHEA period due to the strong supervision and the system of continuous assessment imposed over students.

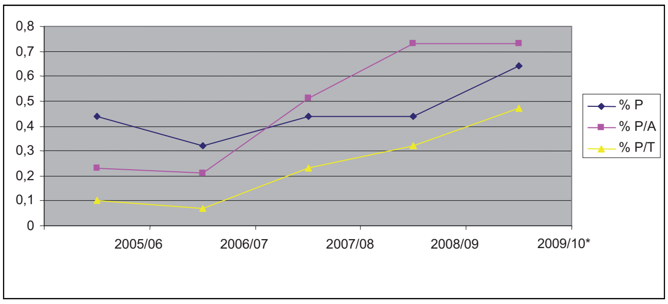

Figure 3. Results from students' survey A.C. 2009/2010

On the other hand, it is important for us to establish to what extent the introduced innovations in the practical coursework (as well as in evaluation and student-teacher interaction) *positively* influence students' performance. That is, we have to ask ourselves about the direct implications between the laboratory work and success rate. From Table IX, we cannot establish a direct effect on the rate of **active** students, crudely speaking: a higher implication of the teacher and instructional innovations do not mitigate students' drop-out rates. However, from the point of view of pass rates, from the year 2005/2006 there is a clear upward tendency. Students feel more motivated and involved in the subject because of the introduced changes with targets at the personalization of the student coursework.

These data points at (but not prove) a direct cause between the innovation in the practical coursework and increase in the success rates. Several factors have to be taken into account:

- External to the subject: Reduction of group size and implantation of EHEA
- Internal: tutorial action, project orientation, insertion of b-learning platforms, changes in the evaluation system.

Aware of the significance of external factors, we are not in the position to evaluate their impact on students' performance. On the other hand, internal factor, although traversal to the whole subject, have a direct impact on the practical coursework and having into account that the theoretical part of the subject has not changed/evolved as the practical one.

So, are we on the right way? It seems so.

REFERENCES

[1] P. J. Denning. "Educating a New Engineer". ACM Communications, vol. 35, issue 12, pp 82-97, 1992.

[2] J. Uceda. "Informe del curso académico 2009/2010". Universidad Politécnica de Madrid. Madrid, Spain, 2010.

[3] Directorate-General for Education and Cultures. "ECTS users' guide" European Comission, 2004.

[4] Servicio de Innovación Educativa (UPM). "Aprendizaje basado en Problemas". 2008. Available at: http://innovacioneducativa.upm.es/guias/Aprendizaje_basado_en_problemas.pdf

[5] Servicio de Innovación Educativa (UPM). "Aprendizaje orientado a Proyectos". 2008. Available at: http://innovacioneducativa.upm.es/guias/AP_PROYECTOS.pdf

[6] M . de Miguel (ed). "Metodologías de enseñanza para el desarrollo de competencias. Orientaciones para el profesorado universitario ante el Espacio Europeo de Educación Superior". Madrid, Spain: Alianza Editorial, 2005.

[7] Project Management Institute. "A Guide to the Project Management Body of Knowledge (PMBOK® Guide)". 2008.