DEFINITION OF A COORDINATED PROJECT-BASED LEARNING TEACHING GUIDE AT COMPUTER SCIENCE STUDIES

Juan Jesús Izquierdo-Domenech[1], Jaume Jordán[1], Jordi Linares-Pellicer[1], Joan Albert Silvestre-Cerdà[1], Raquel Sanchis[2]

[1]Dpto. de Sistemas Informáticos y Computación, Escuela Politécnica Superior de Alcoy. Universitat Politècnica de València (SPAIN)
[2]Dpto. de Organización de Empresas, Escuela Politécnica Superior de Alcoy. Universitat Politècnica de València (SPAIN)

Abstract

The well-known active methodology of Project-Based Learning (PBL) is being used more and more at different educational levels due to a large number of advantages it presents. For example, PBL has demonstrated that it increases students' motivation, develops their autonomy and capacity for self-criticism, reinforces the ability to exchange ideas and collaborate, and promotes creativity, among other advantages. Due to these benefits, several educational institutions are introducing the PBL methodology in their teaching-learning processes.

The implementation phase of this type of methodology should be planned, managed, and carried out carefully, considering several aspects. One of utmost importance is collecting and registering all the critical information related to the contents, materials and activities of the subjects participating in the collaborative project through the PBL methodology. In this sense, the objective of this paper is to propose the definition of a coordinated PBL Teaching Guide that includes all this relevant information; containing, mainly: (i) PBL Project description; (ii) recommended previous knowledge; (iii) learning objectives and outcomes; (iv) PBL model milestones; (v) PBL model planning; (vi) evaluation; and (vii) bibliography. Furthermore, this proposal will define the formal guidelines for students and lecturers to define and frame all the related aspects to carry out the proposed PBL model.

The definition of the PBL Teaching Guide will be based on a case study that involves the following two subjects from the Computer Engineering Degree that are taught at the *Escola Politècnica Superior d'Alcoi* (EPSA) - *Universitat Politècnica de València* (UPV): "Programming" and "Data Structures and Algorithms". This proposal has been developed in the context of an innovation and educational improvement project applied in the EPSA during the last two years, covering five degrees, 55 subjects, and more than 10 different PBL models.

Keywords: Teaching Guide, Project-Based Learning, Computer Science.

1 INTRODUCTION

The labour market is becoming more and more demanding, and it requires professionals with great abilities to make decisions and solve problems in a quick, flexible and resilient manner. This is why higher education institutions are seeking to develop and use teaching methodologies that transform students into highly skilled professionals.

Authors such as [1] state that to achieve this goal, higher educational institutions should move from a traditional teaching model, in which lecturers are "the transmitters of knowledge" while students are "the receivers of such piece of information", to a new educational paradigm in which students have an active and important role in the teaching-learning process.

One of these active methodologies is the project-based learning (PBL), which has numerous benefits such as increasing students' motivation, developing their autonomy and capacity for self-criticism, reinforcing their ability to exchange ideas and collaborate, and fostering creativity, among other advantages.

In this paper we present a teaching guide that implements a PBL model involving two subjects from the Computer Science degree of the *Escola Politècnica Superior d'Alcoi* (EPSA) - *Universitat Politècnica de València* (UPV): "Programming" and "Data Structures and Algorithms". The rest of the paper is structured as follows. First, Section 2 provides some background knowledge about the PBL

ISBN: 978-84-09-34549-6

methodology and the educational improvement project that frames this work. Next, Section 3 describes the aforementioned PBL teaching guide. Finally, Section 4 provides some concluding remarks.

2 PROJECT BASED LEARNING (PBL)

2.1 PBL Methodology

The PBL methodology consists of setting up a problem as close to reality as possible so that students have to solve it through the development of a project. Other definitions state that PBL is a constructivist pedagogical approach whose main objective is to achieve deep learning in which lecturers use an inquiry-based method to respond to problems that are rich, real and relevant to the area being studied [2].

As stated before, the PBL methodology is being supported by many higher education institutions because of its great advantages. A recent literature review of PBL [3] highlights the following advantages: improvement of content knowledge and skills; enhancement of students' attitude; motivation; and self-efficacy to the subject. However, although the PBL methodology provides many benefits in the acquisition of deep knowledge and skills, there are also some drawbacks. For instance, solving the problem proposed is usually a very time-consuming task. This encourages lecturers to carefully define the theoretical contents that students need to develop during the project, and to plan all the aspects of the PBL model as well. For this reason, it is important to define in detail the teaching guide to provide complete information to students.

2.2 PBL Innovation and Educational Improvement Project

This educational research has been performed in the context of an Innovation and Educational Project (IEPI) developed in the EPSA-UPV. The main objective of this project is to incorporate the PBL methodology collaboratively among different subjects of the last courses of different bachelor degrees. Figure 1 shows the IEPI implementation framework with the degrees that have incorporated the PBL methodology and the 10 PBL models defined in this educational innovation [4]. These models cover a wide range of knowledge domains and educational areas, from mechanical engineering to finance. Each PBL model involves a different number of subjects, and hence, each one presents a particular casuistry. Readers may find additional information of each PBL model in the following references: PBL Model 2 [5], PBL Model 3 [6], PBL Model 4 [5], PBL Model 5 [7], PBL Model 6 [8-9], PBL Model 7 [10], PBL Model 8 [11], PBL Model 9 [12], and PBL Model 10 [13]. This IEPI has a duration of two academic years and involves more than 55 subjects, 65 lecturers and 700 students.

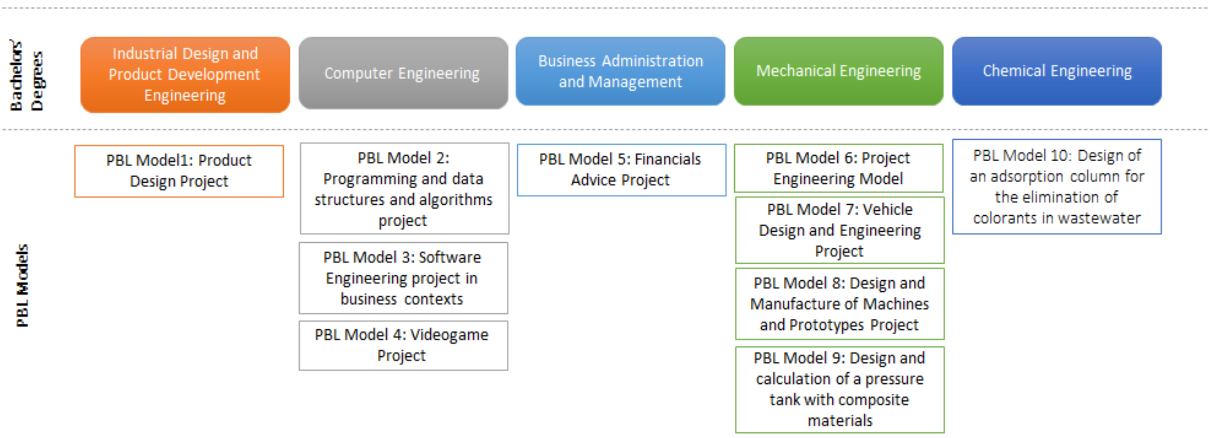

Figure 1. PIME implementation framework.

In this article we focus on the teaching guide that implements the PBL Model 2.

3 A COORDINATED PBL TEACHING GUIDE RESULTS

3.1 Current Teaching Guides

Currently, there are two subjects involved in the PBL Model 2: “Programming” (PRG), and “Data Structures and Algorithms” (DSA).

On the one hand, in PRG, students are introduced to recursive design and the analysis of algorithms, involving aspects such as object-oriented programming (OOP), input/output and exception handling. Moreover, all these contents are applied to the study of linear data structures. The main objective of the subject is to provide students with the necessary knowledge and skills to be able to design, analyse, implement and validate efficient algorithmic solutions for specific problems, in the field of small-scale programming, following the imperative programming model, and with emphasis on fundamental aspects of object-oriented programming. All this will provide students with the necessary aptitudes to build and deploy computer systems. In the programming learning units, a widely used object-oriented programming language is used in coordination with other subjects such as “Languages, Mathematics and Computer Science”, and DSA.PRG is scheduled on first course, semester B (1B), and comprises 6 ECTs. More information about this subject can be found in its complete teaching guide [14].

On the other hand, DSA extends the basic programming skills acquired by students in previous subjects such as PRG. Specifically, the subject has the following objectives for the students: (i) to learn the basics that allow them to analyse, design, and implement abstract types of complex data; (ii) to learn new strategies for designing efficient algorithms, and (iii) to understand the importance of temporal and spatial cost when designing particular solutions. DSA is scheduled on second course, semester B (2B), covering 6 ECTs. More information about this subject can be found in its complete teaching guide [15].

3.2 Proposal of a Coordinated PBL Teaching Guide

As an alternative to the original approach, where the two subjects are scheduled in different academic courses, we propose to allocate them in the same course (2^{nd} course), but splitting contents into three different subjects: PRG, DSA, and Practical Programming Foundations (PPF).The objective is based on changing the original courses by making them just based on a theoretical approach, where the basis of programming, data structures and algorithms are taught in two sequential subjects: PRG on the second course, semester A (2^{nd}A), and DSA on the second course, semester B (2^{nd}B). Hence, the practical contents of the two former subjects become part of PPF, a new PBL-based subject, scheduled on the whole second course, this is, on both semesters (2^{nd}AB).

As DSA is based on the previous knowledge in PRG, merging their practical contents into one single subject (PPF), and making it overlap with them, allows lecturers to better create a learning methodology based on incremental projects in which students will be able to improve and add new features based on the theoretical contents they learn.

Previously, there was a discontinuity in the practical work developed in both subjects. With this new approach, this discontinuity disappears, and the natural flow based on projects can be redesigned in order to better transmit how software development can be improved by iterating and adding new features and knowledge to different projects.

✦ *PBL Project description*

As mentioned before, the PBL project considers two already existing subjects (PRG and DSA) and adds a new subject called PPF. By placing PRG and DSA in the same year in consecutive semesters, we can add a new subject where all the theoretical contents of PRG and DSA can be applied into a more practical and realistic approach. PPF is scheduled as a full-year subject, parallel to PRG and DSA. The main idea is to develop an incremental project, divided into several steps, from a first and inefficient solution to a more complex and efficient one. Again, one of the goals of introducing this subject is to apply theoretical concepts in real problems to understand the complexity of real-life issues and consider aspects like complexity and decision-making.

✦ *Recommended previous knowledge*

To make students take advantage of the full potential of PPF, recommended prerequisite subjects are: “Statistics”, “Introduction to Computer Science and Programming”, “Computer Fundamentals”, “Computer Technology”, “Mathematical Analysis”, “Discrete Mathematics” and “Programming Languages, Technologies and Paradigms”.

✦ *Learning objectives and outcomes*

Along with these three subjects, the student is expected to acquire the following knowledge:

1 To analyse and comprehend algorithms in terms of spatial and temporal cost. The student can dissect iterative and recursive algorithms and the implications of inefficient approaches in big-scale programming by applying mathematical concepts.

2 To model solutions for complex problems through OOP.

3 To decide a better implementation for a specific data structure when facing particular problems.

4 To generalize, through OOP concepts such as generics, concrete data structures to more abstract ones.

5 To learn and understand the difference between linear and non-linear data structures in terms of implementations and cost and the implications of choosing an inadequate data structure in a specific problem.

6 To optimize resources on expensive algorithms through techniques such as divide and conquer or dynamic programming.

✦ *PBL model milestones*

To achieve these objectives, the PBL project is divided into several milestones:

1 Students' ability of temporal and spatial analysis with different practical examples both recursive and iterative.

2 Design of an application with OOP techniques.

3 Ability to make decisions in different practical examples to decide the best implementation of data structures during the development of the project.

4 Perform different generalizations in the project to be developed by means of OOP techniques.

5 Analysis of linear and non-linear structures by means of different implementations in the project for the detection of the most suitable ones in each case.

6 Implementation of efficient algorithms based on divide and conquer techniques and dynamic programming for various project tasks.

✦ *PBL model planning*

On the one hand, these are the different units the student must undertake to follow up with the PPF subject. These didactical units will be developed in PRG and DSA.

- **Didactic Unit I: Mathematical properties of algorithms: complexity and correctness**

  Theoretical Contents (15 h): Introduction to the analysis of algorithms and main concepts. Time and space cost of computer programs, growth of functions and asymptotic notation. Analysis of iterative and recursive algorithms, as well as sorting algorithms.

- **Didactic Unit II: Introduction to OOP**

  Theoretical Contents (15 h): Main concepts of OOP. Objects, classes, reference variables. Creation and destruction of objects. Overloading, parameter passing, attributes and static methods. Main Java libraries and primitive types.

- **Didactic Unit III: Linear data structures: stacks, queues and lists**

  Theoretical Contents (15 h): Representations of linked sequences. Structure, representation and different implementations of stacks, queues and lists.

- **Didactic Unit IV: Introduction to Data Structures**

  Theoretical Contents (3 h): Introduction to basic data structures in java language.

- **Didactic Unit V: Data structures: trees, hash tables, graphs**

  Theoretical Contents (27 h): Study of the fundamental data structures. On the one hand, the different types of trees are studied, such as generic trees and their representation, heaps, binary search trees, and balanced trees. On the other hand, hash tables, their different representations and implementations, and their use are shown. Finally, the different types of directed and undirected graphs, their different representations, and graph traversal are studied.

- **Didactic Unit VI: Algorithmic techniques: divide and conquer, greedy algorithms, dynamic programming**

  Theoretical Contents (15 h): Study of the main algorithmic techniques and their most representative algorithms. Mergesort and quicksort sorting algorithms using the divide and conquer technique are explained. On the other hand, the greedy algorithm technique is explained by the currency exchange problem and the knapsack problem. Finally, the dynamic programming technique to obtain optimal solutions to the coin exchange problem and the knapsack problem is explained.

On the other hand, and in parallel, the PPF subject (4,5 ECTS) will be composed by the following units/projects:

- **Incremental project 1, Complexity analysis (5 h):** By developing and analysing the proposed algorithms, the students can better understand the consequences of why temporal and spatial costs must always be considered. Students can analyse empirically what the theoretical approach proves regarding the efficiency of algorithms.
- **Incremental project 2, OOP basics (10 h):** Once students are familiar with the foundations of OOP, students are introduced to a real-life problem in which they have to reason about how to model it into an OOP language.
- **Incremental project 3, Linear data structures (10 h):** By adding the notion of data collections to the problem in project 2, the student must choose and implement a concrete linear data structure and decide the best implementation for a particular context.
- **Incremental project 4, Generalization (5 h):** By applying more advanced concepts such as universal polymorphism, the student faces the problem of generalising the already-implemented data structures to benefit from code reuse.
- **Incremental project 5, Non-linear data structures (10 h):** Having learnt the costs of the most popular non-linear data structures such as hash-maps, trees and graphs, the student is proposed to replace previously used linear data structures with non-linear ones and take advantage in terms of efficiency.
- **Incremental project 6, Optimization (5 h):** Since several algorithms applied during the project are still very inefficient (e.g., recursive and the use of the call-stack), the student can apply different approaches such as divide and conquer or dynamic programming in order the increment the overall performance of the project.

✦ <u>Evaluation</u>

The evaluation of this subject is made exclusively on the incremental project. Specifically, each part of the incremental project is evaluated separately. This allows students to make corrections to their project and thus not be disadvantaged in the following incremental parts of the project. This approach is more similar to the way software companies work with incremental prototypes in which new tools, functionalities and improvements are introduced. Thus, it is an efficient method and closer to the professional reality in which students will find themselves in the future.

✦ <u>Bibliography</u>

The following books are recommended as sources of complementary information:

- The Java Programming Language. Arnold, Ken; Gosling, James; Holmes, David.
- Data Structures and Algorithm Analysis in Java. Weiss, Mark Allen.
- A complete Java training course. Java how to program. Deitel, Harvey M.; Deitel, Paul J.
- Introduction to algorithms. Cormen, Thomas H.; Leiserson, Charles Eric; Rivest, Ronald; Stein, Clifford.
- Fundamentals of Algorithmics. Brassard, Gilles; Bratley, Paul.

4 CONCLUSIONS

In this paper, we have presented a PBL approach to better teach the basic foundations of the "Programming" and "Data Structures and Algorithms" subjects of the Computer Science degree from the EPSA-UPV. In this proposal, we suggest an alternative to having two isolated subjects by unifying the practical contents of both subjects into one new overlapping PBL-based subject called "Practical Programming Foundations". Merging the practical contents of the two courses, let us define and create a sequence of incremental problems, project-based, towards a final objective where the knowledge of both courses is combined, and a final project is developed with the synergy of both courses.

The theoretical content remains sequential, and the new subject runs simultaneously with the two theoretical courses. Finally, a complete teaching guide for this new subject has been proposed.

ACKNOWLEDGEMENTS

This article has been supported by the Vice-rectorate for Digital Resources and Documentation (Vicerrectorado de Recursos Digitales y Documentación) and Vice-Rectorate for Studies, Quality and Accreditation (Vicerrectorado de Estudios, Calidad y Acreditación) under the Call for Learning + Teaching (Convocatoria Aprendizaje + Docencia (A+D 2019)) and Project Code: A-157. The authors would like to acknowledge the support of the Institute of Educational Sciences (Instituto de Ciencias de la Educación) of Universitat Politècnica de València, the Evaluation and Monitoring Commission for Educational Innovation and Improvement Projects (Comisión de Evaluación y Seguimiento de Proyectos de Innovación y Mejora Educativa - CESPIME) and Escuela Politécnica Superior de Alcoy.

REFERENCES

[1] B. Alorda, K. Suenaga, P. Pons. "Design and evaluation of a microprocessor course combining three cooperative methods: SDLA, PBL and CnBL", *Computers & Education*, vol. 57, no. 3, pp. 1876–1884, 2011.

[2] T. Markham, Project based learning; Design Coaching Guide. California: Wilsted & Publishing Service, 2012.

[3] P. Guo, N. Saab, L.S. Post, W. Admiraal, W. "A review of project-based learning in higher education: Student outcomes and measures", *International Journal of Educational Research*, vol. 102, 101586, 2020.

[4] R. Sanchis, J. Mula, B. Cantó Colomina, D. Garcia-Sanoguera, J.I. Torregrosa López, "Diseño de la Incorporación del Aprendizaje Basado en Proyectos en las Titulaciones de Grado del Campus de Alcoy de la Universitat Politècnica de València", *VI Congreso Nacional de Innovación Educativa y Docencia en Red*, Editorial Universitat Politècnica de València, 2020.

[5] J. Linares-Pellicer, J. Orta-López, J. Salavert-Torres, M.J. Segura Flor, J.A. Silvestre Cerdà, R. Sanchis, "Towards Inter-Subject Project-Based Learning in Programming-Related Courses at Computer Science Studies", *EDULEARN Proceedings*, pp. 3973 – 3978, 2020.

[6] J. Esparza Peidro, E.J. Golf Laville, J.J Izquierdo-Doménech, J.V. Tomas Miquel, R. Sanchis, "Definition of Project-Based Learning Models in the Computer Engineering Degree", *EDULEARN Proceedings*, pp. 3966 – 3972, 2020.

[7] J. Capó Vicedo, A. Benito, R. Sanchis. "Project-Based Learning in Financials Advice", *EDULEARN Proceedings,* pp. 7522-7528, 2021.

[8] M. Pérez-Sánchez, M. Díaz-Madroñero Boluda, J. Mula, R. Sanchis, "The Sustainable Development Goals (SDGs) Applied to Higher Education. A Project Based Learning Proposal Integrated with the SDGs in Bachelor Degrees at the Campus Alcoy (UPV)", *EDULEARN Proceedings*, pp. 3997 – 4005, 2020.

[9] M. Pérez-Sánchez, R. Sanchis, "Sustainable Development Goals Integrated in Project-Based Learning in the Mechanical Engineering Degree", *Proceedings of INTED2021*, pp. 5965 – 7480 - 7487, 2021.

[10] D. Garcia-Garcia, N. Montanes, L. Quiles-Carrillo, J. Ivorra-Martinez, R. Sanchis, "Implementation of the Project Based Learning Methodology in the "Materials. Design and Restyling" Subject", Proceedings of INTED2021, pp. 9883– 9891, 2021.

[11] S. Montava-Jorda, V. Colomer Romero, A.V. Martínez Sanz, M.J. Reig Pérez, F. López Esteve. Implementation of Project-Based Learning at a Multidisciplinary Level of the Specialization in Design and Manufacture of Machines and Prototypes in the Degree of Mechanical Engineering, *EDULEARN Proceedings*, pp. 7477 – 7483, 2021.

[12] R. Balart, N. Montanes, L. Quiles-Carrillo, S. Jordá-Gisbert, H.C. Sanchis-Gomis, Sanchis, R., “Model Design and Calculation with Polymeric and Composite Materials through Project-Based Learning”, *EDULEARN Proceedings*, pp. 3979 – 3986, 2020.

[13] C. Carbonell, S.C. Cardona, I. Domínguez, V. Fombuena, M.F. López-Pérez, J. Lora, R. Sanchis, “Project-Based Learning as a Coordination Methodology between Subjects in a Chemical Engineering Degree”, Proceedings of INTED2021, pp. 5965 – 5974, 2021.

[14] Universitat Politècnica de València. “Programming Teaching Guide”, 2021. Retrieved from URL.

[15] Universitat Politècnica de València. “Data Structures and Algorithms Teaching Guide”, 2021. Retrieved from URL.