

# VIRTUAL PROGRAMMING LABS IN THE COMPUTER PROGRAMMING LEARNING PROCESS, PREPARING A CASE STUDY

Marílio Cardoso<sup>1</sup>, Rosa Barroso<sup>1</sup>, António Vieira de Castro<sup>2</sup>, Álvaro Rocha<sup>3</sup>

<sup>1</sup> ISEP – Polytechnic of Porto (PORTUGAL)

<sup>2</sup> GILT & ISEP – Polytechnic of Porto (PORTUGAL)

<sup>3</sup> Universidade de Coimbra (PORTUGAL)

## Abstract

Usually, in course units of programming, the process begins with the teaching of algorithms, and later the knowledge of logic learned being applied to a specific programming language.

In the course of Informatics Engineering of the Informatics Engineering Department of School of Engineering, Polytechnic Institute of Porto (ISEP/IPP) one of the course units that first directly deals with this training is called APROG (unit of Algorithm and Programming). This unit starts with the teaching of algorithm followed by the programming language JAVA.

In recent years, we used the EduScrum teaching methodology, where the student follows a set of procedures focused on the needs and indications of a Product Owner. The subject is structured in: programming fundamentals, programs codification, classes, modular decomposition, arrays and text files manipulation.

During the next year we intend to add to the process a new tool, with the main goal of evaluating its potential for the teaching process of programming. This tool is the VPL (Virtual Programming Lab) that will be incorporated into the Moodle (Modular Object-Oriented Dynamic Learning Environment) of the course unit of APROG. With this study, we intend to analyse the potential of the tool and verify if it is possible to provide to student's validation of their own programs by previously providing a set of data. In this way, we will have an overview about the potential of the tool and correlate the results with its pedagogical potential.

This paper is part of a preliminary study on virtual laboratories in programming teaching.

Keywords: Teaching programming, algorithms, Learning Management System (LMS), EduScrum, Virtual Programming Lab.

## 1 INTRODUCTION

Given the importance and relevance of computers and information technology in contemporary society, teaching programming becomes essential for the development of computer applications and for technology evolution.

The training of professionals in this area is very diversified and one of the fundamental roles remains to be played by Higher Education Institutions linked to the Engineering areas and in particular in the field of Computer Engineering.

This document intends to present a reflection on the teaching process of programming in the School of Engineering, Polytechnic Institute of Porto using for this purpose one of the basic course units, APROG.

In APROG, in the last years, an EduScrum methodology was adopted using LMS (Learning Management System) Moodle (Modular Object-Oriented Dynamic Learning Environment).

This study intends to evaluate the use of virtual programming laboratories in teaching programming, using, for this, some classes of prototype in the next academic year.

### 1.1 Computers and society

Nowadays computers are absolutely essentials to modern societies. Almost everyone have a personal computer, a tablet or smartphone. In the simplest activity that we can imagine, certainly will be involved computer technology. We can find it, e.g., in communications, businesses, banking,

medicine, travel, work, leisure and teaching. With a computer and an internet connection everyone have the world at a click's distance.

One of the greatest impacts of technology on society is the importance reduction of distance, allowing to communicate almost instantaneously, in a reliable and cheap way, with someone thousands of kilometres away. It is possible to buy (almost) everything by internet, at any time and from any source, so that e-commerce is an irreversible reality. That meant a revolution in business models, commerce, market structure and global economy.

Even the relationships and the way we communicate have undergone huge changes with the advent of the internet and social networks. This reality brings new challenges and risks as well as a considerable increase in privacy concerns, highlighting issues of security and criminality.

Computers also had a huge impact on education today, and future growth is still expected. The raise of internet and interactive technologies turn information and communication technologies (ICT) as an important tool for teaching and learning. With internet in their own homes, many students take the initiative to study other subjects doing research and watching tutorial videos on specific topics. Distance learning is not new, because television has been used for decades for schooling. However, computers and the Internet have completely changed the paradigm, being possible to have distance classes with interaction between teacher and students, communicating in a synchronous or asynchronous way, making evaluations, etc. E-learning is an undeniable and growing trend.

So, obviously, we can point to positive and negative impacts on computers. With computers, a huge amount of data and information can be stored in a small space and accessed easily, the work can be done in a more efficient way in very less time, documents can be kept secret and greatly increases multitasking and multiprocessing capabilities of data. However, there are some drawbacks to, such as fast changing computer technology, risks of lost huge important data and information, data privacy issues or illiteracy of computers that can lead to social exclusion. Also, some health issues are associated with the use of computers, such as obesity due to physical inactivity, vision problems or body posture.

All these issues have in common computer technology and the need for qualified professionals in several fields, such as programming, computer security or computer networks.

## **1.2 Teaching Programming: Algorithms and programming languages**

Teaching computer science has two distinct strands. The first is in the user's perspective, enabling students to use computer tools to improve their work in a large range of subjects. The other, more important for those who want to develop applications and computer solutions, is programming.

To develop software is a very important activity in this era of information and knowledge society. Although producing software is not just programming, this task is crucial to the process success.

Algorithms are the base of the programming process. To develop algorithms is to define a finite set of rules, well defined (without ambiguity), for the solution of a problem in a finite time, and it should be effective and efficient. The most common example around world of use of algorithms is a cook recipe, where we find a finite sequence of steps that lead to the execution of a task, a sequence of instructions for a specific goal.

To start the learning process should be use a language that can be understand by all. The first approach should be with pseudo-code. A kind of native language used in a shorter way that turns it still understandable to the student, but pointing to a structure more similar of a programming language.

Another way to represent algorithms is using flowchart, that consists in a path guiding by arrows, with a short set of symbols, so self-explanatory as possible, and understandable in a visual way.

When the algorithm is specified it must the encoded (written) in a programming language. A programming language is a formal language with a specific and well-defined set of instructions that allow to develop software. There are a huge set of programming languages, each one with is specific syntax and characteristic and fields of application.

## **1.3 Technology to mediate the teaching and learning process**

Technology-mediated learning is a generic term, related with several different approaches to using computers in learning and teaching. In recent years, technology has been raised its importance in schools at various levels, from the basic to the university. However, on the last years Learning

Management Systems (LMS) became one the most used and powerful tool to promote learning, in which educators can define an environment where learners can access resources, research, drills and assessment tools.

### 1.3.1 LMS

LMS is a framework that interconnects both sides of the learning process simplifying the relationship between teachers and students. LMSs advocate a new teaching paradigm, breaking with a logic of face-to-face and centralized teaching, facilitating access to teaching materials and providing assessment mechanisms [1]. They promote teacher-student communication (synchronous or asynchronous) and the simplification of the learning process by providing resources that reinforce student motivation and an easier understanding of program content. LMS are also a valuable tool to teachers helping them to manage the learning process, identifying and evaluating learning and training goals and collect data to supervise and monitor the process to reach the goals.

In recent years, LMS has been raised its importance in schools at various levels, from the basic to the university linking teachers, students and community stakeholders with a single platform and improving the workflow efficiency and productivity.

### 1.3.2 Moodle

In Portugal Moodle is certainly the most popular and widespread LMS.

Moodle is an open source software under General Public Licenses (GNU), with a business model that allow to use some basic elements for free, but charge for extra options. It is often used for distance education, blended learning, face-to-face classroom and other modes of e-learning in companies, schools and universities.

In order to use Moodle, it is just needed a basic computer, an internet connection and internet navigation skills. So, to the average student is very simply to use Moodle, almost like use Social Media, doing comments, posts, view or share pictures, videos, etc.

## 2 CASE STUDY: APROG (THE FIRST COURSE UNIT OF PROGRAMMING ENGINEERING)

One of the main goals of the course unit of APROG is to initiate students in software production. As a first course unit in this area it begins to develop logical reasoning skills through the development of algorithms and data structuring. Programming is to design algorithms and describe them in a given language. On the case of APROG the language that has been used is JAVA. However, the language is not the main issue to start programming.

On the last 4 years the number of registered students has been between 343 and 383 students.

*Table 1. Number of registered students in APROG in the most recent school years.*

School year	Number of students
2013/2014	351
2014/2015	361
2015/2016	383
2016/2017	343

These numbers are the result of the annual admission of, usually, more than 250 students (from national access contest plus others from special contests), and the remaining ones are related with retention of students from previous years.

### 2.1 Students outcomes

According with the syllabus of APROG course unit, students should be able to:

- 1 Understand and apply the fundamental programming concepts.

- 2 Identify and describe the requirements of a problem, analysing it, design an algorithm for its computational solution and conceive an appropriate test? Plan for its validation.
- 3 Know and understand Java programming language in the perspective essentially procedural and apply it to algorithms? Implementation. Testing the solutions using the appropriate test plan.
- 4 Analyse and designing algorithms as models of computational processes structured in modules, creating and reusing modules and implement them.
- 5 Know, understand and use indexed data structures as well as manipulate text files.
- 6 Apply the acquired knowledge to solve real problem.

In the case of APROG, the course unit used in the case study that carries out, the curriculum unit form, establish this sequence of contents: programming fundamentals, programs codification, data types reference: classes, modular decomposition, data types reference: arrays, text files manipulation and developing applications.

To achieve that goal, in APROG, the students start programming with JAVA language.

## 2.2 How this course unit is structured and organized

This course unit takes place during the first semester and is taught by a team that, usually, have between 8 and 10 teachers. So it has to be very well organized and planned with the maximum detail to ensure that the lessons taught are in all similar and with the same pedagogical objectives.

As shown on Fig. 1, APROG is divided on three different types of classes.

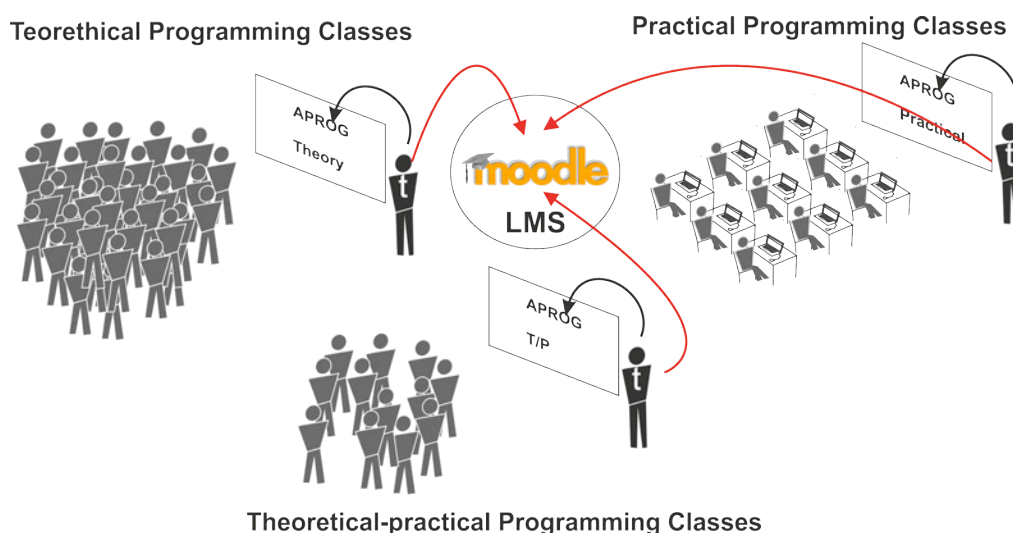


Figure 1 - Types of APROG classes at DEI-ISEP

APROG usually has about 350 students every year. Theoretical classes involve all the students that attend the course unit. This kind of classes are taught at various moments and typically we have about 75 students on each theoretical class which is taught in an amphitheatre. Each one of those classes are an important moment for the process of programming learning and, usually, where the main programming concepts are transmitted.

It is however verified that the students often skip theoretical classes, what could compromise the learning or the acquisition of the fundamental knowledge of the programming by them. Thus, to reinforce the transmission of this knowledge was created theoretical-practical classes.

The main difference between theoretical and theoretical-practical classes are:

- Reduction of the number of students (about 50% less);
- Explanation of some solved exercises related with the theoretical topics of the week.

The third set of classes are the practical ones where students work in a class room with computers, and where they apply their knowledge of theoretical and theoretical-practical classes. The number of

students in each class is about 20 and, usually, they work in pairs. During these classes, the students must codify computer programs in the selected programming language (in this case JAVA) to solve a set of problems proposed by the teacher and available in Moodle. To codify, students use an Integrated Development Environment (IDE) which has been, in the recent years, NetBeans.

**Table 2.** Weekly workload by class type

Class type	Number of classes by week	Hours by class	Total week hours
T	1	1	1
TP	1	1	1
PL	2	2	4

With so many students and hours of classes, several teachers are needed to ensure the teaching. In the last academic year the distribution of teachers by type of class was as shown in the table 3.

**Table 3.** Teachers by class type

Class type	T	TP	PL
Number of teachers	1	3	7

## 2.3 Planning, Contents and Learning Activities

As presented before, it is necessary to plan all the activities with detail. We will just focus the practical-laboratory classes to this study due the intention to engage students with Virtual Programming Labs on the future.

In APROG there are classes during twelve weeks which are divided according the pedagogical needs. It is also necessary to clearly identify the assessment moments and the organization of all the contents and learning activities.

The planning is made considering the curriculum course unit form and mainly the articulation between the different class types. First there are the theoretical ones, where are presented, analyzed and discussed fundamental concepts, then the theoretical-practical ones and finally the practical – laboratory classes where students apply what they learned and what was demonstrated before. In this last set of classes there are used techniques of active learning, cooperative group work and problem based learning (PBL).

In the practical classes problems are organized in four blocks (sprints) each one with specific purposes, as shown in table 4.

**Table 4.** Sprints organization

Sprint	Number of weeks	Contents
1	4	Algorithms
2	2	Implementation of algorithms in Java and modular decomposition
3	3	Arrays and files
4	3	Integrative application of all contents

### 2.3.1 EduScrum

On the last years on these course unit, we applied EduScrum methodology.

As known, teaching computer programming is a complex and challenging task for any teacher, mainly for those who have beginning students. This activity is very different from teaching any other subject, so the approach should be different to.

Scrum is a framework for developing and sustaining complex products. Therefore, it is widely used in software development and became mainstream in this area and it been used as an iterative and incremental agile software development framework. Companies use the Scrum methodology to promote team work and to foster a productive and enjoyable way of working. Although Information Technology (IT) is the preferred field of application for Scrum, its use has been tested in many areas, including education.

So EduScrum is a variation of Scrum for the educational system where students are responsible for the learning process by teacher delegation [2]. In EduScrum there are six formal events: Team formation, sprint planning, stand-up meeting, sprint review, sprint retrospective and personal reflection.

The scrum team own a board that gives an overview and the structure to make the group's work transparent. The work is organized by sprints, usually of three weeks each. There must be a sprint planning meeting to define and clarify the sprint goals and every next session should start with a short (five minutes) stand-up meeting. In this meeting, it should be remembered what have already been done, defined what should be done in the present session and, if there is any blocking take advices and replace the work. During sessions, the Scrum board must be updated after each changing.

In EduScrum methodology there are roles for team members, such as Product Owner (the teacher), the Scrum Master (one of the students) and the other team members (the other students). Student team should be self-organizing and multi-disciplinary.

The team must be autonomous and responsible, flexible and collaborative, creative and motivated.

### 2.3.2 How APROG uses EduScrum

The pedagogical methodology adopted on the course unit is EduScrum that is related with one of the most actual methods used on the companies that develop computer programs, as a productive method. In APROG, EduScrum methodology is adapted to this unit and to the student's context. That is the first course unit of programming, it takes places at first year, first semester so that's a new reality to the students.

In the practical-laboratorial classes students must be organized in team of two (exceptionally three) and they must work in peer review. That means that all their assignments must be review by a colleague before be assessed by the teacher.

As mentioned before there are four sprints in APROG, witch organization could be seen in table 5.

**Table 5.** Practical classes' organization

<b>Weeks</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>	<b>10</b>	<b>11</b>	<b>12</b>
PL				A			A		A			A

At the beginning of each sprint, there must be a sprint planning meeting, where the teacher presents the team formation (which can have adjustments between sprints), the global learning objectives and the global work planning.

At the beginning of the first class of the week (5 to 10 minutes) the week's work should be planned indicating:

- What is expected of the group: learning objectives and evaluation acceptance criteria;
- Exercises to be solved in the week (each task with an indication of the units of effort), by what order, and by whom, balancing the units of effort per student.

Students must begin to carry out their work according to plan and register the Scrum board with the work situation. Scrum board is an essential and mandatory tool that must be always updated and available for all team members and for the teacher.

At the beginning of the second class of the week students and teacher must have a brief stand up meeting where each student must answer:

- What was done?
- What am I going to do today?
- Is there any blockage?

After an assignment and its verification by another colleague the teacher can ask any team member about to better gauge if students understood what they did. If the assignment have errors or the students couldn't answer in an accurately way, teacher should not accept the assignment order to redone it.

At the end of each sprint there must have a sprint review where students will solve, individually, a set of exercises. One week after the sprint review the teacher should disclose to students the results of sprint evaluation. The final grade of each sprint is calculated be:

- 30% sprint tasks assessment (heavy average of tasks performed successfully);
- 70% mark individual evaluation.

After that students should have a moment to do a sprint retrospective and a personal reflection by each student.

## 2.4 APROG contents – Moodle

In ISEP, the LMS adopted is Moodle, so it used also in the course unit of APROG. It is used to provide contents, to present the course unit and the schedule of the classes throughout the academic period, to allow the submission of assignments and to present its evaluation, as well as to provide examples of exams of previous years, allowing to notify students and publish news and announcements.

The screenshot displays the Moodle interface for the APROG course unit. The top header shows the ISEP logo and the user 'Marlio Cardoso JOG (Sair)'. The main content area is titled 'Geral' and contains a welcome message, exam correction links, and a list of weekly topics. The right sidebar shows 'Informação Disciplina' with links to 'Programa', 'Bibliografia', and 'Avaliação'. The user is identified as 'Marlio Cardoso JOG (Sair)'.

Figure 2. APROG Moodle contents

In central area of Fig. 2 we can see how the contents are available to the students, organized by a first section (general) that contains examples of exams, announcements, news and a welcome message for students. Below are sections (per week) with content for each of the weeks, by class type, and may contain other materials, suggestions for studies, messages, or link for submission of assignments for the evaluation.

On the left side are available some functionalities like request print services, the list of all participants in the course unit, or access to the contents distribution by week.



Figure 3. APROG Moodle assignments submissions

In Fig. 3 there is an example of information about the assignments statistics, namely the number of students enrolled in Moodle, the number of assignments submitted, the deadline for submission and the time remaining for the submission limit.

So, Moodle is a simple and efficient way to provide contents and promote communication between teachers and students.

### 3 NEW APPROACH ON APROG TEACHING BASED ON VIRTUAL PROGRAMMING LABORATORIES

Due the quantity of projects developed by the students and naturally the need to evaluate them a problem was found. With the current teaching model, with several assignments during the academic period, teachers spend a lot of time to evaluate all the projects and to grade school assignments.

On the last years teachers referred that they need to spend lots of time with this assessment process (basically every 3 weeks they should evaluate all the sprint works for all the students)

With a big set of proposed exercises, it is not possible to evaluate all of them in a deep way, thus the solution is evaluated by sampling. This means that feedback to students about their own assignments is neither complete nor given quickly.

It was also perceived that students cannot develop their work and improve their programming skills as fast as desirable and they don't are quite sure if what they did is correct.

So to achieve goals in an efficient and fast way teachers referred that classes must have less students or should be considered less assignments moments.

A solution must be identified to solve the excess of teacher's assessment work and, at the same time, provide to the students a mechanism that help them to evaluate their programming code with a set of data given by the teachers.

With this goal in mind we looked for a new solution to help us to support the process, which allows a faster assessment of the projects.

We identified Virtual Programming Labs (VPL) [3], a Moodle module, as a possible solution to solve some of the identified problems and we are considering to implement it on the next year on a few practical classes maintaining the actual EduScrum methodology

#### 3.1 VPL

VPL is an open source software product developed by the Department of Computer Science and Systems, University of Las Palmas de Gran Canarias, Spain. This tool allows the management of programming assignments on Moodle, and the incorporation of the software development environment in a virtual classroom. The architecture contains a Moodle module, a source code editor applet and a special Linux process that runs as background process, commonly called a daemon, which allows the remote execution of programs in a secure way. VPL helps out in the follow-up and personalized and continuous orientation of the student's learning process, thus contributing to address the difficulties that the student faces in carrying out programming activities. It intent to provide a simple and intuitive development environment, in particular to new students entering the programming courses.



Nowadays, ethical and moral issues are very important and matters like plagiarism are crucial to the institutions credibility [4]. VPL includes a tool which allows to detect plagiarism [5] among source code searching for similarity between files. The tool has the main goal to detect plagiarism among submissions for the same task by different students in a course. However, it can include other sources, like submissions for the same task in previous years or similar tasks from other sources.

Learning to program is a difficult process to many students. To learn to programme computers student must have motivation and to program a lot. That is a hard task that demands a lot of effort and training. The assignments may present specific difficulties for the students, mainly for those you never programmed before. So that requires a constant monitoring by the teacher. Usually, until the work is evaluated, the students do not know if they did correctly their tasks. In addition, assessment requires time and effort by the teacher due to the number of students, the number of submissions required and their complexity. The availability of an automatic grading tool [5, 6] to assist personalized monitoring and guidance in a continuous learning process, will reduce the initial difficulties faced by the student, improving students' programming skills and optimizing the teaching staff time [7].

### 3.2 New model for APROG practical classes

Moodle has been used in APROG for several years, and it has an obvious added value in the communication between teacher and students and in the availability of didactic resources. Thus, using the potential of Moodle, we intend to add a new VPL module to facilitate the process of learning programming, provide students with much faster feedback about finished assignments, and reducing the teacher's mechanical work, allowing him/her to focus on other tasks.

The VPL module will be used in practical classes as a pilot experience with a limited number of students. With this option, we intent to not change radically everything, due to allow to compare the results between the two groups (those who will use VPL and the others) and, eventually, with grades from past academic years.

### New Practical Programming Classes

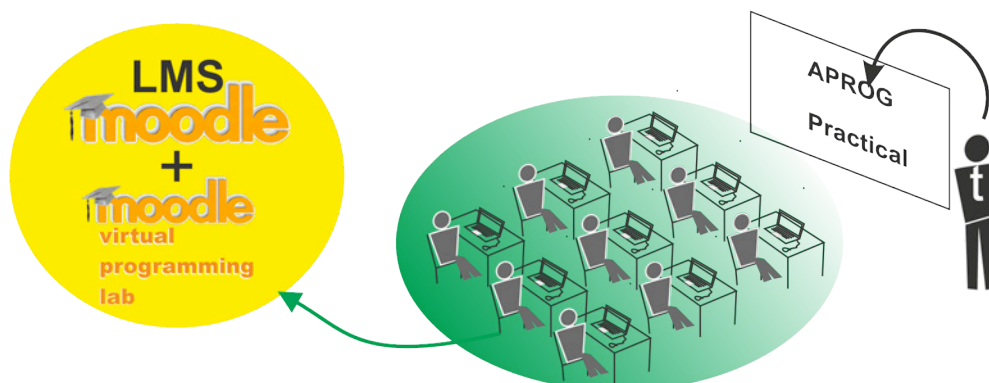


Figure 4 - Moodle with VPL.

All students will have the same assignments, and will be evaluated in the same way. The expectation is that those who used VPL will be able to practice more because they will need less supervision from teachers to verify the results.

With this model is expected that teachers will spend less time assessing assignments. However, in the first instance, they will need more time to prepare the environment to use VPL, as well as building exercises and validation data so that they can be evaluated in an automatic way.

## 4 CONCLUSIONS

In this paper, we present a work in progress that is being carried out in Informatics Engineering Department of School of Engineering, Polytechnic Institute of Porto and where combining technologies and methodologies is expected to provide an even more effective teaching learning process to students of Computer Engineering.

With this preliminary study related to Virtual Programming Labs in laboratories in programming, we intend to study the impact of virtual laboratories on learning. We will use a real unit course on a real course of Computer Engineering to promote and support his case study.

We have the expectation that the process will lead to an improvement in the programming learning process. If that happens, the use of the VPL should be extended to all APROG students and, if possibly, to other course units or even to other courses.

## ACKNOWLEDGEMENTS

This work was partially supported by the European Union's Horizon 2020 research and innovation programme under Grant Agreement No. 732049 – Up2U.

## REFERENCES

- [1] E. Dahlstrom, D. Christopher Brooks, and Jacqueline Bichsel. The Current Ecosystem of Learning Management Systems in Higher Education: Student, Faculty, and IT Perspectives. Research report. Louisville, CO: ECAR, 2014. Retrieved from <http://www.educause.edu/ecar>.
- [2] J. Sutherland, Scrum: The Art of Doing Twice the Work in Half the Time, Crown Business, 2014.
- [3] J. Rodríguez-del-Pino, E. Rubio-Royo and Z. Hernández-Figueroa, "Uses of VPL". 5th International Technology, Education and Development Conference (INTED), pp. 743-748, 2011.
- [4] V. Crittenden, R. Hanna, and R. Peterson. "The cheating culture: A global societal phenomenon.", Business Horizons pp. 337-346, 2009.
- [5] J. Rodríguez-del-Pino, E. Rubio-Royo and Z. Hernández-Figueroa, "A Virtual Programming Lab for Moodle with automatic assessment and anti-plagiarism features," International Conference on e-Learning, e-Business, Enterprise Information Systems, & e-Government, 2012.
- [6] A. Patil, "Automatic grading of programming assignments", 2010.
- [7] D. Thiébaud, "Automatic evaluation of computer programs using Moodle's virtual programming lab (VPL) plug-in.", Journal of Computing Sciences in Colleges 30.6, pp. 145-151, 2015.