

# The Use of Gamification for Teaching Algorithms

Luiz Ricardo Begosso<sup>1</sup>

Douglas Sanches da Cunha<sup>2</sup>

João Victor Pereira Pinto<sup>3</sup>

Lucas Teixeira de Lemos<sup>4</sup>

Michel Gargel Nunes<sup>5</sup>

Fundação Educacional do Município de Assis

Centro de Pesquisas em Informática

Assis, São Paulo, Brazil

<sup>1</sup>begosso@femanet.com.br;

<sup>2</sup>professordouglasacunha@gmail.com; <sup>3</sup>p-vittor@hotmail.com;

<sup>4</sup>lucas.lemos97@live.com; <sup>5</sup>michelgargel@hotmail.com

Luiz Carlos Begosso

Fundação Educacional do Município de Assis

Faculdade de Tecnologia de Assis

Assis, São Paulo, Brazil

lbegosso@femanet.com.br

**Abstract**—This paper presents our experience using gamification principles into the free and open-source learning management system Moodle for aiding and abetting our Computer Science students in learning algorithms. In this work, we used the LMS Moodle and we developed a module with gamification features focused on promoting engagement of students in the learning process of basic concepts of algorithms, data structures and pointers. We conducted a deep study about Moodle and how to implement gamification plugins into the environment. We used and configured HotPotatoes, Games, LevelUp and Badges plugins. We defined the lessons about the specific concepts of algorithms and we created them into Moodle environment. Then we implemented several games, like hangman, crosswords, cryptex, snakes and ladders. We used LevelUp with the objective to gamify the students' learning experience by allowing them to earn experience points to level up in their courses. Badges were used in order to motivate students in their achievements and to show their progress in the courses. In our learning environment, it is possible to have different levels of proficiency in the class and rewards are based on different rules the professor can define; the student can earn more points for some activities or fewer points for other activities. A comparative ranking can be displayed so that students can see their progress, and the professor can look at the log to verify the activities students did and the points they have earned. The results showed us that the use of gamification concepts can contribute significantly to the process of teaching-learning programming concepts to students in the early years, as well for teenager's students without any previous knowledge about programming concepts. This study aims to present the methodology used to carry out our experience and the results obtained with the development and implementation of gamification concepts in a free and open-source learning management system.

**Index Terms**—gamification; education; computer programming; software algorithms; computer science education

## I. INTRODUCTION

THE teaching of algorithms and programming concepts to first-year students has become a critical challenge to En-

gineering and Computer Science courses. It is usual that students face difficulties to understand some concepts they are taught for the first time, such as logical thinking, abstraction, algorithms, data structures, formal computer language, and others. This is potentially quite challenging material that is going to form the basis of the rest of their learning. In few weeks students are introduced to data structures, programming resources, binary trees, sorting, which are examples of very important subjects for them, but many students do not learn those concepts appropriately.

Several efforts have been conducted to get more positive results in the learning process of programming concepts [1]. This paper presents our experience using gamification principles into the free and open-source learning management system Moodle for aiding and abetting our Computer Science students in learning algorithms.

The Millennials, also known as Generation Y, and the post-Millennials, also known as Generation Z, use technology as part of their lives, they are digital natives and have more of a positive view of how technology is affecting their lives than any other generation. Learning is a faster and more flexible endeavor for Millennials and post-Millennials if they can use their smartphone or tablet PC. Both those generations are university students right now.

As described by [2], Millennials and post-Millennials are best suited to modern learning methods and prefer learning in a more relaxed environment, expect instant gratification, and value a fun and flexible learning environment where colleagues are friends. They like to have some control over their development and feel comfortable using technology in the classroom.

This affinity with technology encourages the use of additional tools for supporting the teaching-learning process. In this context, the use of games and simulation environments has taken place in the academia and is getting more and more attention of researchers. The process of using game thinking and mechanics to engage an audience and solve problems is

named “Gamification”. The concept of gamification is associated with the use of game elements in generic contexts. Gamification is the use of game-design elements in a particular task, providing more intense interaction on the exchange of information and encouraging the involvement of the public in a playful way. The concept of gamification has been gaining prominence also in the educational area and this work is situated in this context.

According to [3], some examples of gamification have been used in the area of Information Technology, such as Ribbon Hero, which is an application for the corporate management area, serves to educate users of Microsoft Office 2007 and 2010 how to use the ribbon interface. In another example, the Duolingo, which is a language-learning platform, adopts the experience of accumulating experience points to measure the progress of learning a foreign language by the user.

In this context, gamification techniques that use game design and mechanics can be applied or found in many areas, such as education, corporate environment, entertainment, retail trade, among others.

Tasks that tend to become boring or unnoticed are an object of study for gamification, aiming to become more attractive and provide more intense interactions and experiences by applying gaming techniques in non-gaming environments, engaging the involvement of the public in a playful way. According to [4], the term gamification encompasses the use of game elements in activities that are not strictly a game, that is, the individual thinks and uses game systematics and mechanics, but their action does not determine that he is playing a game.

For [5], gamification takes place from characteristics that we like most in games and incorporate them into our daily activities, so that tasks can be carried out in a fun and exciting way.

Using a set of gaming mechanisms and design techniques in a gamified environment, learning can be encouraged as entertainment because it awakens and increases interest and enhances pleasure while performing a task. It can also increase the content retention and improve motivation for learning.

It is worth mentioning that the techniques of gamification include several characteristics, however, it is not mandatory to apply all of them, since the literature defining the term gamification presents differences of interpretation. According to [6], the most common features found in gamified applications are points, levels, rankings, challenges and missions, medals, achievements, integration, engagement, personalization, feedback, rules and narrative.

Games have great potential to improve the learning experience [7]. For these authors, gamification tends to produce improvements in the understanding, commitment and motivation of users.

In this project, we developed a module with gamification features focused on promoting engagement of students in the learning process of basic concepts of algorithms, data structures and pointers. We used the LMS Moodle, a free and open-

source learning management system, widely used in academia. We conducted a deep study about Moodle and how to implement gamification plugins into the environment. We used and configured HotPotatoes, Games, LevelUp and Badges plugins. We defined the lessons about the specific concepts of algorithms and we created them into Moodle environment. Then we implemented several games, like hangman, crosswords, cryptex, snakes and ladders. We used LevelUp with the objective to gamify the students' learning experience by allowing them to earn experience points to level up in their courses. Badges were used in order to motivate students in their achievements and to show their progress in the courses.

This paper is organized into five sections. Section I is the introduction, while section II describes the learning environment we used to implement the project; Section III describes the details of the implementation of the project; the assessment of the learning environment is outlined in Section IV; and Section V presents our conclusions.

## II. THE LEARNING ENVIRONMENT

We decided to use Moodle (Modular Object-Oriented Distance Learning) to apply the gamification concepts because it is a learning management system (LMS) created under the concept of free software that can be installed on different platforms, such as Unix, Linux, Windows and MAC OS. Its development is collaborative by a virtual community, which brings together programmers, designers, administrators, educators and users from all over the world and is available in several languages. The platform has supported Distance Education and face-to-face courses, the formation of study groups, professional training and others [8].

We conducted a deep study about Moodle and how to implement gamification plugins into the environment. We used and configured HotPotatoes, Games, LevelUp and Badges plugins, which will be described below.

HotPotatoes is a plugin created by the Research and Development team at the University of Victoria Humanities Computing and Media Centre, Canada. It contains a package of five tools or authoring programs for the creation of interactive exercises for the Web, named JCloze, JCross, JMatch, JMix and JQuiz; these tools are compiled into one unit, using a sixth application called The Masher. For the implementation of this plugin, it is necessary to understand where the information will be placed (texts, questions, answers, images), since the tools will automatically create the respective webpage for the use of the students.

Games is the second plugin that can be installed in Moodle, in order to provide the creation of educational games. Games plugin is used to simplify the development of gamification concepts in the project. This plugin has several traditional gamification features: hangman, crosswords, cryptex, millionaire, sudoku, snakes and ladders, the hidden picture, book with questions.

For the hangman feature, a keyword is chosen from a glossary or quiz short answer questions and generates a hangman

puzzle. The teacher can set the number of words that each game contains, configure if it shows the first or last letter, or if it shows the question or the answer at the end. Students will need to deduce which word will be explored within the rules of the game, based on the content that was previously studied.

For the crosswords feature, words are taken from either a Glossary or quiz short answer questions and it generates a random crossword puzzle. Teacher can set the maximum number of columns/rows or words that it contains. Students can press the button “Check crossword” to check if the answers are correct. Every crossword is dynamic so it is different for every student.

For the Cryptex feature, it is like the crossword but the answers are hidden inside a random cryptex and the student needs to deduce them, based on the content previously studied.

The Millionaire feature takes words from multiple choice quiz questions and creates a “Who wants to be a Millionaire” style game complete with the three lifelines. Students must answer each question correctly to proceed.

For the Sudoku feature, a sudoku puzzle is presented to the students with not enough numbers to allow it to be solved. For each question the student correctly answers an additional number is slotted into the puzzle to make it easier to solve.

The hidden picture feature randomly grabs an image from a glossary and hides it behind panels. When the student answers a question correctly, a portion of the image is revealed.

For the Snakes and Ladders feature, students have to traverse a traditional “Snakes and Ladders” board by answering questions taken from either a Glossary or quiz short answer questions. As they get an answer right, the dice are rolled and a random number displayed. The game piece is moved ahead of that many squares. If the game piece is in the bottom of a ladder and the answer is correct, it goes to the top. If the game piece is in the head of the snake and the answer is wrong, it goes back to the tail.

Book with questions feature controls the progression of the student, he can go to the next chapter only if he answers the questions correctly.

We also used the LevelUp plugin, with the objective to gamify the students’ learning experience by enabling learners to gain experience points for participating in their courses, increasing engagement and participation by motivating students to progress towards the next level and rewarding their efforts by congratulating them for reaching the next level. LevelUp plugin allows the teacher to use the leaderboard to leverage competitiveness while keeping it friendly and motivating, unlock access to course content when a certain level is reached, and substitute experience points for other images to make the learning process even more attractive for the student.

We used the Badges plugin to allow the teacher to show student’s progress awarding him with badges, based on a variety of chosen criteria.

### III. PROJECT IMPLEMENTATION

Our institution enrolls approximately 2,200 students in 11 programs. STEM-related degree programs are the most traditional and we have around 330 students enrolled at Computer Science course, with a typical first-year class of around 100 students. In the first year, the Department of Computer Science offers an introductory programming course: Algorithms I, which is the expected starting point for students majoring in our Department. It is a full-year typical introductory course with no previous programming experience required. In the second year, we offer the course Algorithms II, closing the fundamentals for logic and programming basics.

From our previous experience, we know that it is difficult for students to understand some concepts they are taught for the first time, such as logical thinking, abstraction, algorithms, data structures, formal computer language, and others. This is potentially quite challenging material that is going to form the basis of the rest of their learning. In few weeks students are introduced to data structures, programming resources, binary trees, sorting, which are examples of very important subjects for them, but many students do not learn those concepts appropriately.

Our project is divided into five phases, as described in Figure 1.

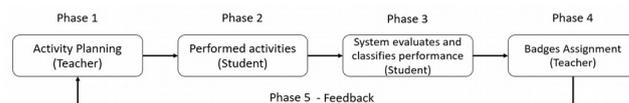


Figure 1 –Phases of the Project

During Phase 1, teacher defines the learning outcomes that students should be able to know in order to complete a study stage successfully; the teacher must decide which concepts and skills students should have at the end of a given learning period.

When Phase 2 begins, students access the learning environment and perform the activities proposed by the teacher.

On Phase 3, the system evaluates and classifies students’ performance, based on the activities performed by each student and on the results achieved by each student.

During Phase 4, the teacher assigns a badge to the students, according to their performance and experience points gained.

For the feedback phase, the teacher can check what aspects students presented most difficult, based on reports generated by the learning environment about students’ performance, and make a new plan of activities, returning to Phase 1.

In this project, we decided to apply games concepts in the Moodle free software platform, to verify if our students would have more success in learning those important concepts. So, we implemented this course covering the following subjects: Declaration and Manipulation of Variables, Repetition Structures, Data Structures, Pointers, and List Structures. We integrated the HotPotatoes, Games, LevelUp and Badges plugins into Moodle platform, using textual and videos resources fo-

cusing on the practical concepts of the contents, and the games were used for engaging students in the learning process.

We implemented four packages from HotPotatoes: JCloze, JCross, JMatch and JQuiz. The JCloze package creates gap-fill exercises, for example, student need to complete sentences about some taught concept. Unlimited correct answers can be specified for each gap, and the student can ask for a hint and see a letter of the correct answer. The JCross package creates crossword puzzles which can be completed online, with words from theory taught; the teacher can configure grids of any size and the system places the words in columns automatically. The JMatch package creates matching exercises, where students can match vocabulary to pictures or translations, or ordering sentences to form a sequence or a conversation. The JQuiz package creates multiple-choice and short-answer quizzes; specific feedback can be provided both for right answers and predicted wrong answers.

For the LevelUp plugin, there are general and specific configurations the instructor needs to do. Initial settings include General, Ranking, Cheat Guard, and Logging settings, and there are different tabs, like Ladder, Report, Log, Level, Rules, Visual and Settings. At Settings tab, teacher can enable the student to gain Experience Points and reach new levels. In the configuration of the Student Ranking, instructor controls whether participants can see each other's name and avatar. Neighbors are the participants ranked above and below the current user. For instance, when choosing 'Display two neighbors', only the two participants ranked directly higher and lower than the current user will be displayed. The rank is the absolute position of the current user in the ladder. The relative rank is the difference in experience points between the user and their neighbors.

At Cheat Guard setting, teacher can enable it to limit the maximum number of actions that will count for Experience Points during the time frame given to the student. Any subsequent action is ignored. Teacher can configure the minimum time required between identical actions (an action is considered identical if it was placed in the same context and object; for example, reading a forum post will be considered identical if the same post is read again).

At Visual tab, teacher can configure images that are used as badges, which are assigned to each student, according to the level they reach the experience points, showing their specific performance. Figure 2 presents the badge images we configured for each level.

At Infos tab, teacher can see all the values used to identify each level and its respective image used to customize the badges. This tab records all the actions performed by students, so the teacher can create motivational phrases, which are attributed to each student when he reaches a new level and receives a new badge.

Finally, at Rules tab, teacher defines the rules to compose all the activities that will be evaluated by the LevelUP plugin. In this tab, the teacher can configure several rules considering all the activities students have to do, setting values for experi-

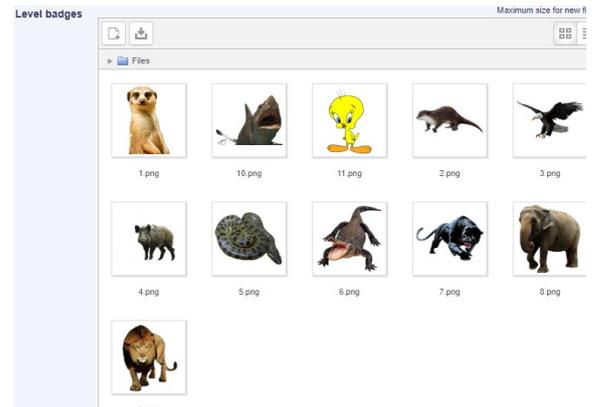


Figure 2 –Badge images used for each level

ence points and identifying which events are triggered as students perform actions in the course. Once each rule is included by the teacher, he can add or remove activities and resources which LevelUP plugin will record as an action to be taken on the student's experience point.

When students access their dashboard for the first time, on the left side they have information about grades and level reached, on the right side they have access to the lessons, exercises (traditional and games) and homework. Figure 3 presents the student view of the learning environment.

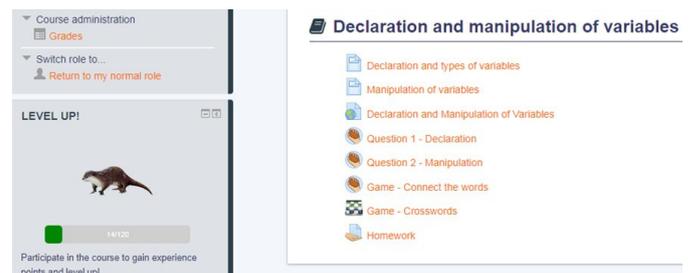


Figure 3 –Student view of the learning environment

On the dashboard of the course, students have access to the concepts taught, for example, Figure 3 illustrates the concepts about declaration and manipulation of variables. Below the first concepts, students have questions to answer and the first game: crosswords. On the left side of the dashboard, students have their level, which changes to different images as they participate in the course and gain experience points to level up.

Instructors and teachers have access to students' progress dashboard and they can verify the reports about students' progress, students' log of use, change the rules for reaching different levels and change general settings. Figure 4 illustrates how the instructors can see students' report, with students' name, the level reached, experience points gained and progress. All the environment is configured by the instructor, so we can check which level the student reached, how many

points he has earned until that moment, and the student’s progress compared to other students.

| First name / Surname | Level | Experience points | Progress  |
|----------------------|-------|-------------------|-----------|
| [Redacted]           | 10    | 1190              | 1100/1190 |
| [Redacted]           | 10    | 1354              | 1354/1354 |
| [Redacted]           | 10    | 2150              | 2100/2150 |
| [Redacted]           | 10    | 1301              | 1301/1301 |
| [Redacted]           | 9     | 1071              | 108/121   |
| [Redacted]           | 8     | 917               | 79/121    |
| [Redacted]           | 7     | 747               | 20/121    |
| [Redacted]           | 7     | 821               | 100/121   |
| [Redacted]           | 7     | 749               | 20/121    |
| [Redacted]           | 6     | 670               | 70/121    |
| [Redacted]           | 6     | 695               | 65/121    |

Figure 4 –Students’ progress dashboard

Figure 5 illustrates some of the games students need to participate for gaining experience points, the first game is the crosswords, the second game is the hangman, the third game is a quiz and the fourth game is the cryptex.

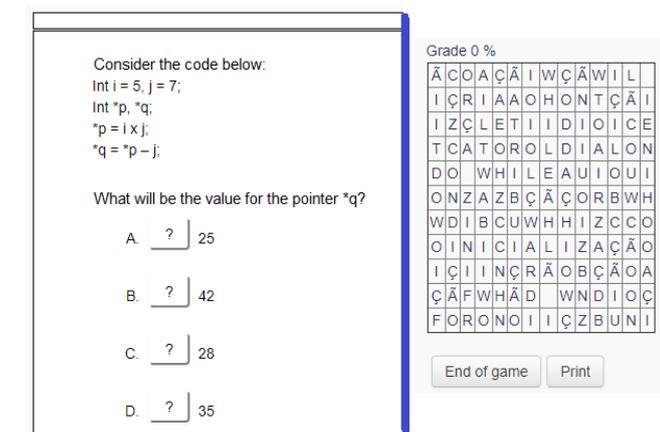


Figure 5 –Examples of games implemented in the project

For each game, students have a specific number of tries, configured by the teacher, and successful results take students to the next level.

#### IV. ASSESSMENT OF THE ENVIRONMENT

In order to know better the effectiveness of our gamification environment for promoting engagement of students in the learning process of basic concepts of algorithms, we applied it for our first-year students in the Computer Science course. The study was conducted with two groups, each one of them with 22 students (n = 22), of both sexes. The first group was called by “Game Group” and the second group was called by “Test Group”.

The Game Group explored the gamified learning environment during two weeks, and then they performed the activities planned in the environment. At the end of the experience, students of this group were tested for the concepts discussed in the environment.

At the same time, the Test Group was being taught the same concepts by professors using the traditional methods. At the end of this stage, students were tested for the same concepts of the first group.

The analysis of the data was performed with a paired t-test on the same sample unit and the objective was to verify if there was a significant difference between the two groups. Table I presents the results each group reached on this test.

Table I - t-Test: Paired Two Sample for Means (p < 0,05)

|            | Number of Individuals | Mean Score | Standard Deviation | P-value |
|------------|-----------------------|------------|--------------------|---------|
| Game Group | 22                    | 8.0        | 2.07               | 0.0015  |
| Test Group | 22                    | 5.5        | 3.26               |         |

We can see in Table I that the Game Group obtained a mean score significantly higher than the Test Group ( $8.0 \pm 2.07$  vs  $5.5 \pm 3.26$ ,  $p < 0.05$ ). These results were considered very positive because the gamified learning environment increased the content retention and improved motivation for learning.

After having these encouraging results, we applied new questionnaires for students to verify their perception about the qualitative results for the learning environment.

Students explored the content of the modules and, at the end of each stage, they were asked to respond to a questionnaire to evaluate several aspects of the environment. Figure 6 illustrates what students thought about the usability of the tool.

We can see in Figure 6 that 35.3% of the students answered that the tool is very easy to use; 23.5% of the students answered that the tool is easy to use; 23.5% of the students answered that the tool is moderately easy to use. At the other end, only 17.6% of the respondents answered that the tool was reasonably moderate-difficult to use.

Then, we asked the students if the tool can help in the learning process of basic concepts of algorithms. Figure 7 illustrates their answer about this question and we can see that 76.5% of the students answered that the tool has a very great

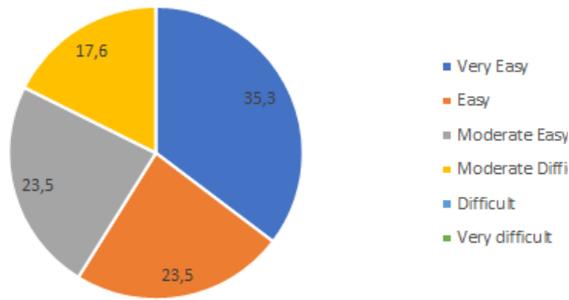


Figure 6 –Degree of difficult in using the tool

contribution to the learning of basic concepts of Programming; 17.6% of the students answered that the tool has a great contribution to the learning of basic concepts of Programming. At the other end, but not least important, only 5.9% answered that the tool has a moderate contribution for their learning process.

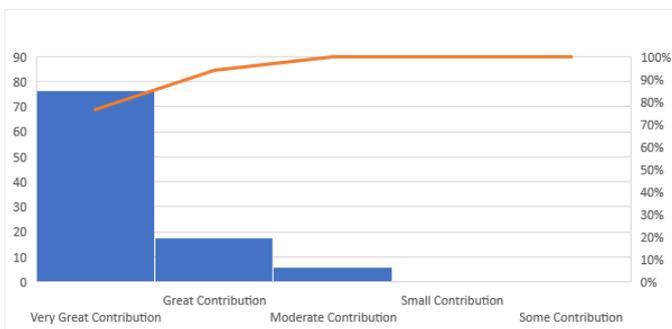


Figure 7 –Contribution of the tool for the learning process

We also asked students to answer which approach they prefer better to learn, traditional or traditional with gamification tools. Figure 8 illustrates their answer about this question and we can see that 82.4% of students prefer to learn by a traditional approach with motivational tools using games; only 11.8% answered that they learn best with traditional classes and 5.8% answered that they prefer learning with other methods.

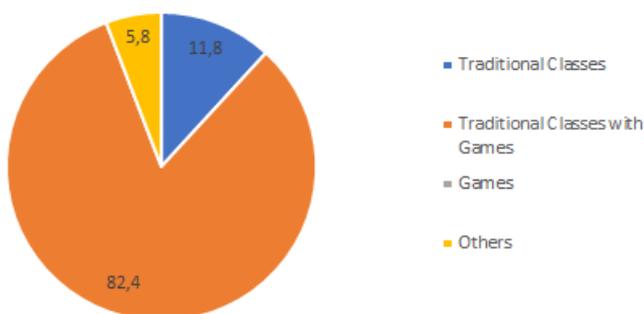


Figure 8 – Students' preference for learning approach

## V. CONCLUSIONS

This work aimed to present our experience in using gamification features in the LMS Moodle, focused on promoting engagement of students in the learning process of basic concepts of algorithms, data structures and pointers. We conducted a deep study about Moodle and how to implement gamification plugins into the environment. We used and configured HotPotatoes, Games, LevelUp and Badges plugins. We defined the lessons about the specific concepts of algorithms and we created them into Moodle environment. Then we implemented several games, like hangman, crosswords, cryptex, snakes and ladders, hidden picture, book with questions and others. We used LevelUp with the objective to gamify the students' learning experience by allowing them to earn experience points to level up in their courses. Badges were used in order to motivate students in their achievements and to show their progress.

In order to assess our environment, we conducted an experience with two groups of students, the first one used the gamified learning environment and the second one was taught the same concepts by professors using the traditional methods. The analysis of our results showed that the group that learned the algorithms and programming concepts using our gamified learning environment had expressive higher grades than the second group, what was considered very positive.

Also, for our qualitative assessment, students reported important results, as they liked the usability of the environment and most of them answered that they prefer to learn by a traditional approach integrated with motivational tools using games.

Our gamified learning environment engaged students through tools that easily facilitate social learning and knowledge sharing through forums, chat, blogs and games, increasing students' content retention and improving their motivation for learning, resulting in higher final learning results, comparing with students taught with traditional methods.

The experience of working with free and open-source software distributed under the GNU license was very positive because we concluded that Moodle is more than just an application for distance learning solutions, it is a widely used tool that meets several needs of gamification concepts and any new implementation through the use of plugins for new solutions.

The use of LevelUp was crucial because it caused a competition with students, creating a social pressure for increasing student's level of engagement. The use of badges illustrated the visual representations of merits and achievements, providing feedback to students on their performance as they earned points within the environment, bringing feelings of competence.

The use of Badges was important because it motivated students in their achievements and created a comparison of their progress in the courses.

The results gathered on this project are very important and positive. They can serve as a basis for the academic commu-

nity to start the development of more gamification environments for teaching computer programming, making the learning process for first year's students less difficult.

Based on our study, we conclude that the use of gamification has an important role in the teaching of algorithms and programming concepts to first-year students of Engineering and Computer Science courses, as it is more attractive to youth students and provides more intense interactions and experiences than traditional methodologies. Our gamified teaching environment increased students' interest and enhanced pleasure while students needed to perform a task, increasing the content retention and improving motivation for learning.

#### REFERENCES

- [1] Begosso, L. R., Begosso, L. C., Begosso, R. H. "An approach for the use of Learning Objects in teaching Computer Programming concepts". In 2016 IEEE Frontiers in Education Conference, Erie, PA, pp. 1-8, 2016. <https://doi.org/10.1109/FIE.2016.7757619>.
- [2] Werth, E. P., Werth, L. "Effective Training for Millennial Students". *Adult Learning*, 22(3), pp. 12-19, 2011. <https://doi.org/10.1177/104515951102200302>.
- [3] Uskov, A., Sekar, B. "Serious games, gamification and game engines to support framework activities in engineering: Case studies, analysis, classifications and outcomes". In IEEE International Conference on Electro/Information Technology, Milwaukee, WI, pp. 618-623, 2014. <https://doi.org/10.1109/EIT.2014.6871836>.
- [4] Nah, F. FH., Zeng, Q. , Telaprolu, V. R., Ayyappa, A. P., Eschenbrenner, B. "Gamification of Education: A Review of Literature". In *HCI in Business*, vol 8527, Nah F.FH. (eds), Lecture Notes in Computer Science: Springer, pp.401-409, 2014. [https://doi.org/10.1007/978-3-319-07293-7\\_39](https://doi.org/10.1007/978-3-319-07293-7_39).
- [5] Freitas, S. A. A., Lacerda, A. R. T., Calado, P. M. R. O. , Lima, T. S., Canedo, E. D. "Gamification in Education: A methodology to identify student's profile". In 2017 IEEE Frontiers in Education Conference, Indianapolis, IN, pp. 1-8, 2017. <https://doi.org/10.1109/FIE.2017.8190499>.
- [6] Surendeleg, G., Murwa, V. Yun, H., Kim, Y. S. "The role of gamification in education – a literature review". In *Contemporary Engineering Sciences*, Vol. 7 (29), 1609-1616, 2014. <https://doi.org/10.12988/ces.2014.411217>.
- [7] Barata, G., Gama, S., Jorge, J., Goncalves, D. "Engaging engineering students with gamification". In *Games and virtual worlds for serious applications (vs-games)*, 2013 5th international conference, p. 1-8, 2013. <https://doi.org/10.1109/VIS-GAMES.2013.6624228>.
- [8] Moodle. Documentation about Moodle Platform. Retrieved from <https://moodle.org>. 2018.