Perceived usefulness and motivation students towards the use of a cloud-based tool to support the learning process in a Java MOOC

Miguel Morales Chan[1], Mónica de la Roca[1], Carlos Alario-Hoyos[2], Roberto Barchino Plata[3], José Amelio Medina[3], Rocael Hernández Rizzardini[1]

1 GES Department, Galileo University, Guatemala
{amorales, monica_dlr, roc}@galileo.edu
2 Department of Telematic Engineering, Universidad Carlos III de Madrid, Spain
calario@it.uc3m.es
3 Computer Science Department, University of Alcalá, Madrid
{roberto.barchino, josea.medina}@uah.es

Abstract

This study aims to investigate the student perception, motivation and utility towards the use of a cloud-based tool, called Codeboard in a MOOC on programming with Java. Codeboard is a web-based IDE (Integrated Development Environment) to learn programming and it was used to enhance the learning activities of our MOOC "Java Fundamentals for Android Development" which is part of the Professional Android Developer MicroMasters Program. This MOOC was deployed in edX and ran from January to June 2017. More than 34,000 learners registered for this course of 6 weeks in "Self-paced mode" (at students pace). The results show the usefulness of including Codeboard to develop formative activities, to check the learning progress and the impact of this tool on students learning process reflected in aspects such as motivation, learning curve and perceived benefits.

Keywords.
MOOC, Java, Codeboard, Cloud-based tools, Learning activities.

1. Introduction

Nowadays, cloud-based tools (CBTs), also known as Web 2.0 tools, offer new opportunities in the educational domain. CBTs allow the exchange of ideas, comments, links to resources, and reuse study content in learning environments that are managed by the professors and students themselves [1]. These tools have the potential to be used in a wide range of learning activities, students are able interact with one another, solve case studies, produce reports, and create conceptual designs [2]. Usluel & Mazman suggest that some CBTs encourage collaborative learning and facilitate active learning [3]. Bates states that these tools provide an excellent opportunity for generating new learning scenarios that promotes interaction and knowledge construction [4].

Another important aspect that is enhanced by CBTs is the fact that the metacognitive self-regulation strategies, such as, planning activities to learn, monitoring self-learning and sharing knowledge through the interaction of learning resources, make the learning process more efficient; the arguments in favor of this facts focus on the idea that the students are able to become producers and reviewers of their own work.

Based on the conception that these type of tools have the potential to strengthen and improve the construction and knowledge management process, many of these benefits could be integrated to new scenarios, such as massive open online courses (MOOCs).

This research is focused on investigating student perception, utility, and motivation towards the use of a cloud-based tool, named Codeboard[1]. Codeboard is a web-based IDE (Integrated Development Environment) to learn programming. It was used to enhance the learning activities of our MOOC "Java Fundamentals for Android Development", which is part of the Professional Android Developer MicroMasters Program. The purpose of this MOOC, which was deployed in edX, is to collaborate, interact, and learn Java language in a MOOC environment.

In this context, the central research questions of this work are presented as follows: (RQ1) What are the main advantages perceived by students when they use Codeboard as a learning resource in the MOOC? and (RQ2) How does Codeboard improve student´s motivation into the MOOC?

This paper is organized as follows: Section 2 presents the pedagogical approach and MOOC structure, outlines the MOOC learning environment and the tools used to support the learning process. Section 3 describes the integration of the web-based development environment (Codeboard) in the edX platform. Section 4 discusses and reports on the learning experience, and is followed by some conclusions in Section 5.

2. Pedagogical approach and MOOC structure

Considering the differences between xMOOCs and cMOOCs [5,6], we chose to follow an xMOOC approach because it promotes the teaching model "cognitivist-behaviorist", which resembles more to traditional higher education. The MOOC contents rely primarily on videos, discussion forums, multiple-choice quizzes and other types of assignments [7]. In an xMOOC, the role of the teacher also changes to a tutor who encourages students learning process. [8]

The MOOC "Java Fundamentals for Android Development" was conceived for developers familiar with object-oriented programming languages, and interested in building Android applications. This MOOC is not only about Java; it is about how you use Java on the development of Android applications, and about the basic knowledge learners need to begin programming with Android. Previous experience with the Java programming language is expected.

[1] https://codeboard.io/

These contents all together provide the scaffolding the learner needs to understand and expand his knowledge and practice of Java. The activities have the purpose to make the student practice and get immediate feedback of his progress on the content. These activities were developed using edX built-in tools and Codeboard, as the only external tool.

2.1 MOOC structure

The structure and sequencing of the MOOC was conceived to support the learning objectives of each topic that is covered in the course syllabus. The MOOC has 5 lessons, and each lesson combines several video lectures, instructional activities, a questionnaire at the end of the lesson, and academic support through different means, such as tutoring sessions, forums, and email. The alignment of these main lesson components ensures an internally consistent structure to help learners accomplish the learning goals. In general, the course content builds towards greater complexity, starting with basic topics and moving towards complex ones. As Fink [9] states: "The goal is to sequence the topics so that they build on one another in a way that allows students to integrate each new idea, topic, or theme with the preceding ones as the course proceeds".

2.2 MOOCs Environment and Learning Tool Support

This MOOC exploits the multimodal potential of online communications. It emphasizes the collaborative process between tutors and learners through the use of ***open discussion forums***, virtual ***tutoring sessions*** using ***Skype, Google Hangouts, or Slack*** as means to exchange knowledge, solve problems, solve exercises, express doubts, or simply develop a collaborative project. A forum component is associated to each lesson. Two tutors monitor this forum answering learners' questions, clarifying concepts, and providing technical assistance. **Tutoring sessions** are interactive events hosted by specialists who have extensive teaching experience. They build upon the lessons course and allow learners to participate live in collaborative discussions. Each session lasts up to an hour in total, including time for questions and answers towards the end.

Another key component of this learning environment is "***Interactivity***". In a MOOC that teaches students how to program with Java, it is very important to have a web-based development environment to practice and share knowledge, besides of simply watching videos. The faculty of this course chose Codeboard due to its versatility, and easy integration with the edX platform. This CBT gives the students a basic scenario where they can compile, run and test Java code in a safe and friendly environment.

3. Integration of a web-based development environment

Codeboard is a web-based development environment that support the IMS LTI standard. It consists of a source code editor, a compiler, built-in automation tools, and

a debugger. Codeboard exercises can be seamlessly integrated within the edX platform because edX also support the IMS LTI standard; the edX platform is the LTI tool consumer, and the external tool (Codeboard) is the LTI tool provider [10].

Adding an LTI component to a course on edX requires several steps: (1) add the LTI passport to the course configuration (LTI ID, client key, and client secret); (2) add the LTI component to a course unit, selecting the "advanced" option from the "add new component" section; and (3) configure the LTI with a name and the URL of the corresponding LTI activity in edX. The most important advantage of an LTI component on edX is that the integration of this external tool is seamless for the students. They find Codeboard activities embedded on their content; they interact with them and can share their results with peer or tutors all inside the edX platform.

3.1 Activities that enable students to practice, to integrate concepts, and to learn new ones.

In each lesson, there are activities that involve the use of Codeboard to solve Java exercises with the aim to improve learners programming skills and understanding. Codeboard is easy, straightforward, secure and highly productive. A learner can understand how a programming exercise works. Simple changes can be implemented and deployed immediately without affecting the original program, or other learners.

The learner can compile and run the new code with the changes, and verify if the code is having the expected behavior. With this type of activities, it is possible to practice the concepts in an interactive way. The student can even make a complete new programming exercise (using the learned concepts), and see if it works. An example of these type of activity is the "Salary Calculation" it asks the students to modify the application that calculates the yearly salaries of employees. The students have to be sure the class employee implements the interface Taxes, define the method getGrossSalary() within employee class and consider other conditions that are given in the instructions of the activity. Figure 1 shows an example of Codeboard activity.

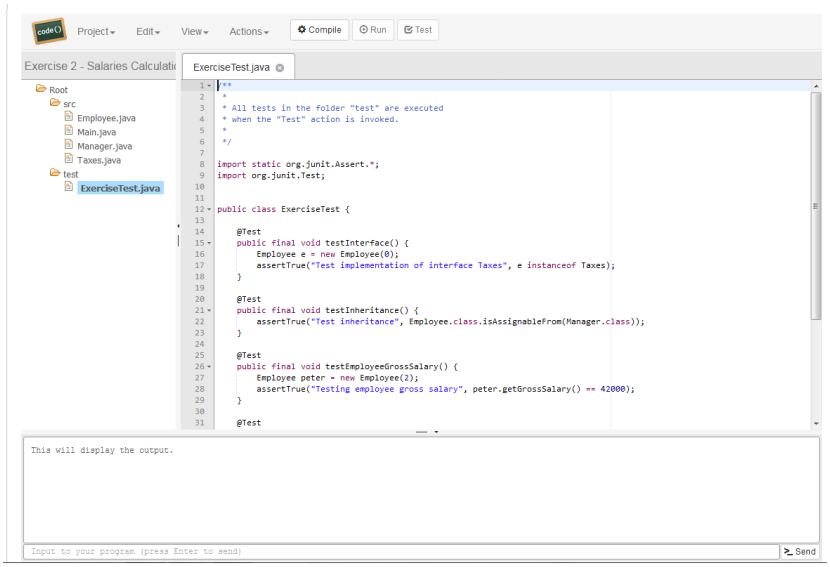

Fig1. Example of Codeboard activity (image Source: https://codeboard.io)

3.2 Special activities to share and learn from peers

Throughout the MOOC content, there are special activities that were designed to lead students in the process of collaborating with one another. The approach followed in this type of activities involves examining the role of that students may play in their learning process, attitudes, engagement and the responsibility they have on shaping their own learning experience.

To share and learn from each other is one of the great advantages of Codeboard. Students were asked to share their solutions with their peers by posting the link at a special forum. This way, anyone could review a solution and learn from it; even better, students could give each other advices of better programming practices (See Figure 2).

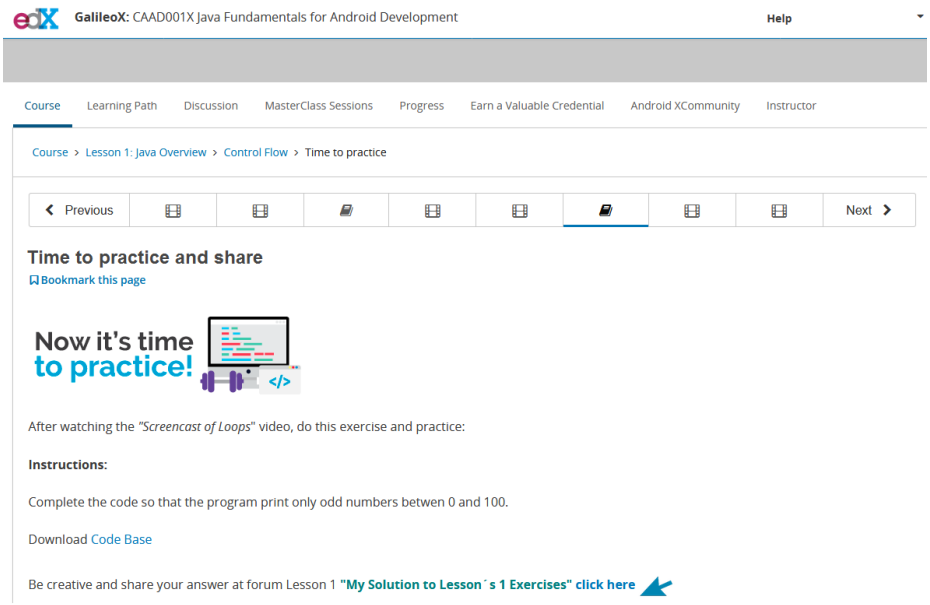

Fig2. Example of share and learn with Codeboard activity

3.3 More efficient and effective feedback

It is important to realize that in something as complex and ever changing as programming, there are always many ways to do something correctly. One of the main problems that a tutor has to face is how to review and grade an assignment; students' submissions are just lines of code. With Codeboard the submission process of an exercise to be reviewed by a tutor or a peer becomes easier and efficient. The student only needs to share a link, and the tutor or peer just needs to compile and run the program to test that it works. Finding errors in case the program does not work correctly

is also simple, and the tutor give a better feedback to the student´s work. The figure 3 shows an example learn activity delivered without using the Codeboard.

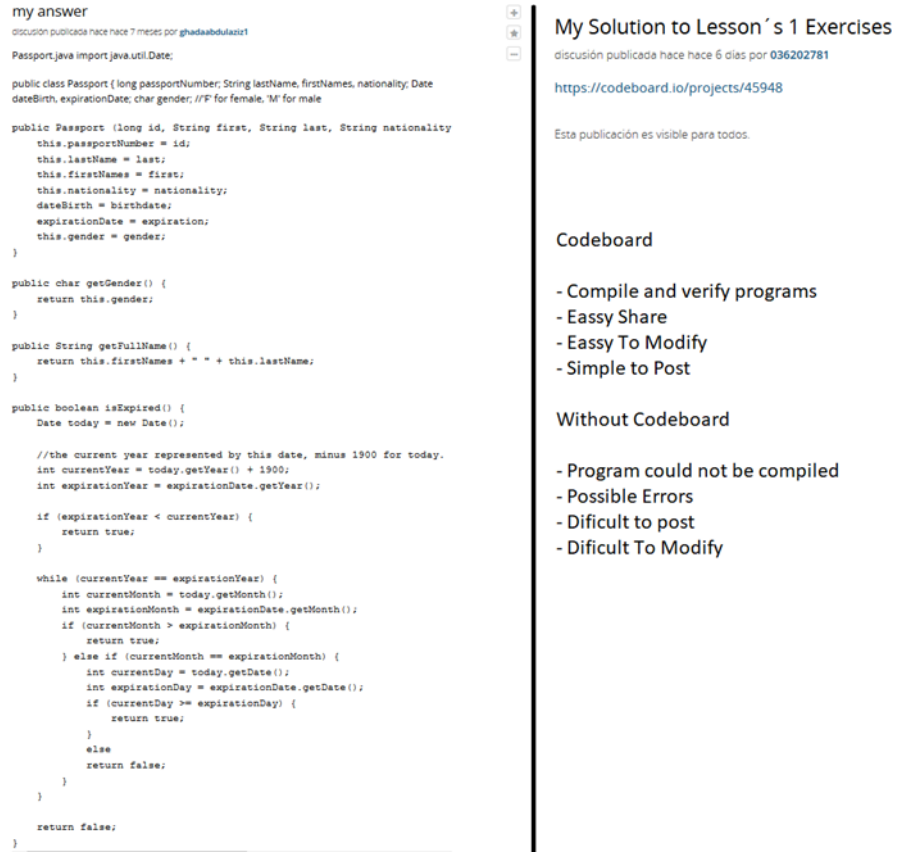

Fig3. Example of learning activity without using Codeboard.

4. Results

The experience presented corresponds to the “Java Fundamentals for Android Development” MOOC, implemented during January 2017 with 34,967 learners from 193 countries registered in the course. Most of the participants were located in India (15.8%), followed by United States (15.1%), Canada (4.3%), Pakistan (2.7%) and México (2.5%). 32.3% were pre-university students, 45.3% had a bachelor's degree and 19.4% had a postgraduate degree. In the first lesson a total of 6,330 different people had watched at least one lecture.

4.1 Online Survey

The overall goal of this study was to determine the usefulness of including Codeboard to develop formative activities, to check the learning progress and the impact of this tool on student learning processes reflected in aspects such as motivation, learning curve and perceived benefits. For this purpose, we conducted an online survey of 20 questions classified into nine main categories. Table 1 shows the survey structure and question types.

Table 1. Survey structure and question types

No.	Section	Number of questions	Survey question types
1	Demographic Data	5 questions	Closed-ended question (Multiple Choice)
2	Perceived Usefulness	9 questions	Closed-ended question (Multiple Choice) & Set of questions using a 5 point Likert scale (from strongly disagree to strongly agree)
3	Motivation	5 questions	Set of questions using a 4 point Likert scale (from absolutely unmotivated to absolutely motivated)
4	Attitude Toward Usage	1 question	Set of questions using a 5 point Likert scale (from strongly disagree to strongly agree)

4.2 Demographic Data

The findings reported in this paper are based on data collected from the 93 learners who had completed this online survey.

The typical learner's age was between 30 and 39 years (29%). Further, 19.4% of the participants were between 40 and 49 years old, and 29.1% were between 20 and 29 years old. Only less than 7.5% were younger than 20, while 15.1% were older than 50 years. In summary we can determine that nearly half of the interviewees were between 30 and 49 years old. Participants have a high level of educational attainment: 31.2% of participants have postgraduate degree and 44.1% Bachelor´s degree. Only less than 20.4% are pre-university students. They also have an experience with technology in general; nearly, 93% reported that they are considered intermediate or advanced user with technology.

4.3 Perceived Usefulness

One relevant part of the study was to assess if participants knew about Codeboard and its main features (functions) before starting the course. 89.2% of the participants indicated that they did not know Codeboard before starting the course.

A set of questions using a 5 point Likert scale (from strongly disagree to strongly agree) was used to determine the overall perception of Codeboard.io. Table 2 shows some of the results.

Table 2. Codeboard.io learning activities student's perception (usefulness)

	M	σ
Did using the Codeboard.io allow you to perform the exercises easier and faster?	3,89	0,76
Did the use of the Codeboard.io facilitate the delivery of the assigned exercises?	4,05	0,68
Did using the Codeboard.io allow you to evaluate your learning more effectively?	3,80	0,90
Did the Codeboard.io allow you to share your knowledge with other course participants?	3,71	0,92
Did the use of the Codeboard.io allow you to store the acquired knowledge?	3,75	0,89
Do you think using Codeboard.io to send your exercises is useful for the teacher to give you feedback in a more effective way?	3,90	0,86
In general, do you consider that using the Codeboard.io within the course was useful for your learning?	4,05	0,86

According to these results, learners considered that using Codeboard in our MOOC, was useful to learn programming with the Java language.

With respect to the main benefits that learners see in using Codeboard, Figure 4 shows the result.

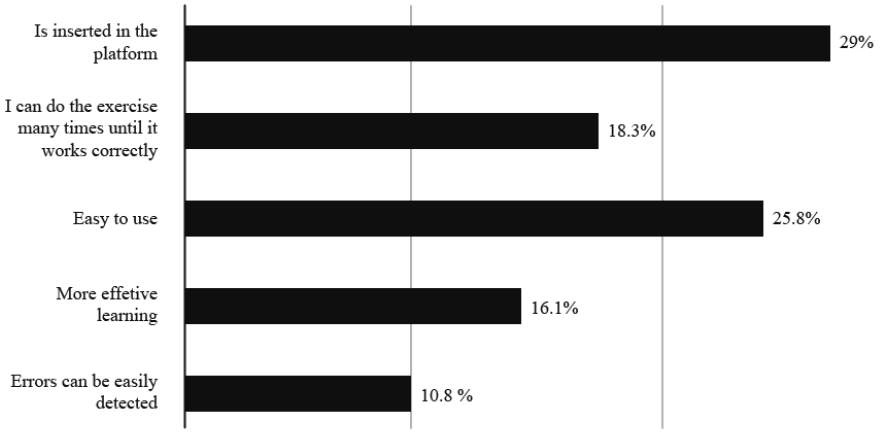

Fig4. student benefits of using Codeboard

54.8% learners identified that ease of use and the seamless in the edX platform were the main benefits of using this CBT.

4.3 Motivation

This section summarizes the results of learners' motivation for using Codeboard as a learning resource in the MOOC. Focusing on the motivational aspects, we applied the intrinsic motivation measures according to Tseng & Tsai [11] to assess learners' perception in the use of this CBT. Table 3 shows the motivational attitude towards learning to use a new tool (Codeboard), towards performing the learning activities proposed in this MOOC using the Codeboard, and towards reflecting the knowledge gained from completing the learning activity using this tool (Codeboard).

Table 3. Intrinsic motivation regarding aspects of Codeboard

Intrinsic Motivation	Learning to use Codeboard that you had not used before	Performing the learning activities proposed in this MOOC using the Codeboard	Reflecting the knowledge gained using Codeboard
Absolutely Unmotivated	8.6%	5.4%	7.5%
Unmotivated	12.9%	10.8%	15.1%
Motivated	59.1%	59.1%	54.8%
Very Motivated	19.4%	24.8%	22.6%

The participants' attitudes of motivational aspects were highly ranked, and participants also indicated positive learning outcomes.

5. Conclusions

This paper presents the results obtained from the integration of a cloud-based tool, called Codeboard, into one MOOC on "Java fundamentals for android development", deployed in the edX platform. The overall aim of this integration was to know the student perceptions, motivation and utility towards the use of this tool, and its impact on students learning process.

The integration of Codeboard into learning activities was simple and seamless for the students, because both Codeboard and edX support the IMS LTI standard, allowing Codeboard activities were shown within the lesson content.

According to the results, learners considered that using Codeboard was useful to learn programming with the Java language, because using Codeboard to solve programming exercises was easy, secure, and highly productive, it made possible to practice the learned concepts in an interactive way. Codeboard integration into learning activities increased student's engagement with the course.

The student's motivation findings were connected to the IDE and its ability to meet the student's needs, Codeboard can compile, run and test Java code in a safe and

friendly environment. To share and learn from each other was another motivation factor, and one of the advantages of Codeboard.

All learning activities that integrated Codeboard received positively evaluations by the learners, Codeboard gave us the chance to enrich the activities and it represented a significant change in the way of teaching and learning Java. It was a successful innovation, and one to build on in next runs of this MOOC.

Acknowledgements: This study has been co-funded by the Erasmus+ Programme of the European Union, project MOOC-Maker (561533-EPP-1-2015-1-ES-EPPKA2-CBHE-JP).

6. Reference

1. Geser, G. (2007). Open Educational Practices and Resources. OLCOS Roadmap, 2012.
2. Hernández, R. (2015). Flexible Educational Environments: Orchestration of Learning Activities Through. Graz University of Technology.
3. Usluel, Y. K., & Mazman, S. G. (2009). Adoption of Web 2.0 tools in distance education. *Procedia-Social and Behavioral Sciences*, *1*(1), 818-823.
4. Bates, T. (2011). Understanding web 2.0 and its implications for education. In Lee, M. and McCoughlin, C. (Eds.): Web 2.0-Based E-Learning: Applying Social Informatics for Tertiary Teaching, pp. 22-41, Information Science Reference, Hershey PA.
5. Hew, K. F. (2016). Promoting engagement in online courses: What strategies can we learn from three highly rated MOOCS. *British Journal of Educational Technology*, *47*(2), 320-341.
6. Morales Chan, M., Hernandez Rizzardini, R., Barchino Plata, R., & Amelio Medina, J. (2015). MOOC Using Cloud-based Tools: A Study of Motivation and Learning Strategies in Latin America. *INTERNATIONAL JOURNAL OF ENGINEERING EDUCATION*, *31*(3), 901-911.
7. Gráinne Conole, G. (2015). MOOCs as disruptive technologies: strategies for enhancing the learner experience and quality of MOOCs. *Revista de Educación a Distancia*, *0*(39).
8. Rodriguez, C. O. (2012). MOOCs and the AI-Stanford like courses: Two successful and distinct course formats for massive open online courses. *European Journal of Open, Distance and E-Learning*.
9. Fink, L. D. (2003). Creating significant learning experiences: An integrated approach to designing college courses.
10. IMS Global Learning Consortium (2014), Learning Tools Interoperability 2.0. Retrived from: http://www.imsglobal.org/lti/
11. Tseng, S.C., & Tsai, C.C. (2010). Taiwan college students' self-efficacy and motivation of learning in online peer-assessment environments. Internet and Higher Education, 13, 164-169.