Massive Open Online Courses on Web Development Education: A Case Study

Aracele Garcia de Oliveira Fassbinder
Federal Institute of Education, Science and Technology of South of Minas Gerais (IFSULDEMINAS)
Muzambinho – Minas Gerais – Brazil
aracele.garcia@ifsuldeminas.edu.br

Ellen Francine Barbosa
Institute of Mathematics and Computer Sciences
University of São Paulo (ICMC-USP)
São Carlos – São Paulo – Brazil
francine@icmc.usp.br

Marcelo Fassbinder
Institute of Mathematics and Computer Sciences
University of São Paulo (ICMC-USP)
São Carlos – São Paulo – Brazil
marcello.fassbinder@usp.br

***Abstract*— (Research to Practice Full Paper) Even though web programming has become a dominant programming model, it still seems underrepresented as a curriculum topic within many computing programs. Such issue in web development education is not recent and was evidenced in several studies. In a massive, open, and online context, in turn, another issue is the difficulty faced by instructors when designing competency-based activities to support the development of skills demanded by the industry, such as the use of agile practices to encourage simple, more lightweight, and faster software development. Such practices include pair-programming; sharing, discussing, and giving feedback; iterative and incremental development, among others. Believing that web development education needs further investigation, we developed a MOOC about web development that combines Bootstrap and CodeIgniter together with Agile Practices, in order to support lifelong learning in such context. Project-Based Learning was the main active learning strategy used to guide activities in the platform. The course design was based on the Learning Design Framework for MOOCs (LDF4MOOCs), an approach to design for learning that brings together important features and principles in order to provide a deeper understanding of MOOCs in the Software Engineering area and related subjects. In this paper, besides to describe a case study about a web development-based MOOC, we also discuss our findings by evidencing that there are still several technological and pedagogical challenges that need to be addressed so as to enhance the MOOCs learning experience for web development education.**

Keywords—MOOCs; Software Engineering; Web Development Education; Videos; Flipped Learning; Patterns.

I. INTRODUCTION

Web programming has become a dominant programming model. However, according to Miller and Connolly [1], it still seems underrepresented as a curriculum topic within many computing programs. Such gap in web development education is not recent and was evidenced, for instance by Lim [2], who highlighted that "Computer Science and Information Systems departments needed to re-examine their curricula in order to prepare students to face the challenge of not only being able to function in a traditional data processing environment, but also be productive in a computing world that is now swamped with web technologies".

The challenges for teaching web development to the masses, through a Massive Open Online Course (MOOC) for example, are even larger, given the difference in characteristics between learners, as evidenced by Gay and Djafarova [3]. In general, MOOCs are considered online courses that require no prior qualifications for entry, can be accessed by anyone, and attract a diverse audience from a variety of learning and professional backgrounds. In such massive, open and online context, another issue explored by Fassbinder et al. [4] is the difficulty faced by instructors when designing competency-based activities to support the development of skills demanded by the industry, such as the use of agile practices to encourage simple, more lightweight and faster software development [5]. Such practices include pair-programming; sharing, discussing, and giving feedback; and iterative and incremental development, among others.

Believing that web development education needs further investigation, we developed a MOOC about web development that combines Bootstrap and CodeIgniter together with Agile Practices in order to support lifelong learning in such context. The design of the course was based on the Learning Design Framework for MOOCs (LDF4MOOCs) [6, 7] an approach to design for learning that brings together important features and principles in order to provide a deeper understanding of MOOCs in the Software Engineering area and related subjects.

LDF4MOOCs also provides a systematic support for educators and learning designers when developing learning maps for open and virtual courses. The framework can be used to support research in this context, being based on theoretical principles from (i) Massive Open Online Courses; (ii) Flipped Learning and active teaching-learning methodologies as a pedagogical strategy; and (iii) Educational Design Patterns and

978-1-5386-1174-6/18/$31.00 ©2018 IEEE

Pattern Language as a strategy to record and share tacit design knowledge

This paper is organized as follows: Section II presents an overview of teaching web programming in massive, open and online contexts; Section III describes the methodology used to design a web development-based MOOC; Section IV presents a case study about the related MOOC; Section V discusses the results and presents the implications for practice, i.e., design of MOOCs on Web Development Education; and Section VI presents our conclusions and perspectives for future works.

II. WEB DEVELOPMENT EDUCATION-BASED MOOCS

In a previous study about MOOCs in Software Engineering Education, we identified that there is a concentration of courses about "Computing Essentials", which includes Computer Science foundations to support software product design, among others [4]. Studies specifically focused in support lifelong learning in Web Development Education through open and online courses are still under investigation. Thus, next we highlight works that have investigated such content but in a virtual distance education context, in general, as well as the main barriers instructors and students may face.

Due to the technical content, teaching web development at a virtual distance has several challenges. First, according to Wang [9], instructors need to demonstrate explicitly what to do and how to do it in a conversational and interactive teaching environment. Second, in this context, students learn best by doing, being engaged in hands-on exercises and solving relevant problems through active learning strategies [9]. Third, there is a concern about specialized software that needs to be installed by students and technical problems that can arise.

Wiedenbeck [10] reports an investigation of the challenges that students without programming experience encounter in an online introductory web development course, such as (i) technological issues when configuring a variety of software; (ii) because their websites were live on the web, students also expressed concern regarding matters of privacy and intellectual property; (iii) particularly for students without prior experience, an extra module or even a new course on basic web fundamentals may be necessary.

As described in Section I, the main contribution of this paper is the design of a Web Development-based MOOC. Section III presents the learning design strategy used to create the MOOC. The MOOC itself is detailed in Section IV, and implications for practice, i.e., lessons learned, are presented in Section V.

III. A LEARNING DESIGN FRAMEWORK FOR MOOCS

Considering the proliferation of MOOC initiatives, such as different types of MOOCs as well as MOOC providers and open platforms to deliver them, there is a need of a common strategy that can be used to manage and engineer MOOCs. This strategy could also be used to reduce barriers to access to information and good practices related to the construction of MOOCs as well as foster the next generation of research and development in this field. Besides, it is also an attempt to promote international trade in the development and use of MOOCs and related services or tools.

In our previous studies [6, 7], we developed and validated a learning design framework that can be used by practitioners and technologists when studying, designing or developing MOOCs as well as mechanisms related to them (e.g., virtual learning environments, learning design tools, and artifacts).

The ideas and solutions embodied in the proposed Learning Design Framework for MOOCs, also named LDF4MOOCs, are mainly grounded on Software Engineering fundamentals due to our area of expertise and because we believe MOOCs, as well as other forms of educational modules (e.g. open educational resources), can be treated as a software product. Having in mind this analogy (i.e., software as a product; MOOC as a product), mechanisms such as life cycle process, design strategy, artifacts, and tools could be defined to support the development process of MOOCs.

The LDF4MOOCs framework consists of: (i) a MOOCs Life Cycle process, which describes fundamental steps to plan, offer, and evaluate a MOOC; and (ii) an Educational Design Pattern Language for MOOCs, which is based on problems and recurring solutions to solve the main activities described in the MOOCs Life Cycle. LDF4MOOCs is briefly described next.

A. A MOOCs Life Cycle Process

The MOOCs Life Cycle Process is divided into five phases, as illustrated in Figure 1.

FIGURE 1 - A LIFE CYCLE PROCESS FOR MOOCS.

During the Exploration phase, the MOOC team needs to clarify the main goal of the MOOC project. It is also important to identify the institution needs and how will the MOOC contribute to the institution's mission. Funding matters are also approached in this phase. The platform that will be used to deliver the MOOC is defined. Furthermore, it is also important to define the teaching/learning challenge by diving into the context and the course audience. Personas may be identified; they are fictional characters that represent a target public. Designers use personas to obtain a detailed description (stories) of the typical user in a context, such as their needs, details, preferences, motivations, and other characteristics that should influence design [12].

The Planning phase, in turn, serves to decide which stories have to be implemented and which can be left for later. In this case, the stories will help the MOOC team to define the syllabus and design the MOOC learning model, i.e., the main pedagogic structure used to identify interdisciplinary opportunities, define instructional goals and activities, and

develop the MOOC Learning Map. According to Filatro [13], a Learning Map, also named Activities Map, is a resource created during the Planning phase of an educational design process and contains the organization of classes as well as the learning outcomes to be achieved by the students considering the activities proposed in the map.

Considering the MOOCs Life Cycle Process (Figure 1), the pedagogical design of the course must be defined in the Planning phase and, in the LDF4MOOCs framework, it is driven by the Flipped Learning (FL) [16] ideas and fundamentals. Flipped Learning is an option to guide the design for learning in MOOCs in order to enhance the learning experiences, increase student's engagement in the course, and emphasize self-directed learning. FL is used to support the task "Defining the course structure (pedagogy strategy)" in the Planning phase, which will help instructors and MOOC teams to create the MOOC Learning Map. The issue of Flipped Learning as a pedagogical strategy for MOOCs is elaborated further in [14, 15].

During the development phase, the MOOC Learning Map created during the planning phase is implemented using some MOOC provider (e.g. Coursera, Veduca, Future Learn, among others) or a MOOC open platform (i.e. Tim Tec, Google Course Builder, and open edX). It requires the search or the development of learning contents, such as Open Educational Resources (e.g., videos, figures, games, open textbooks).

Lastly, the Delivery phase consists in opening the course for enrollment and running the MOOC. Once the MOOC landing page is published, so learners can find the course and enroll.

Evaluation activities can be performed during all phases of the MOOCs Life Cycle Process. It is also important to collect and analyze feedback from learners in several ways in order to promote improvement in the new versions of the course. It is also important to define the status of the MOOC after each launching has been completed, which may depend of the used platform.

B. An Educational Design Pattern Language for MOOCs

There are different mechanisms to capture and encapsulate knowledge related to learning design [17]. For example, theories consisting of a set of design principles; identification of best practices in teaching and learning; and the formalization of knowledge and best practices through patterns and pattern languages.

According to Pressman and Maxim [18], patterns constitute a mechanism for capturing domain experience and knowledge to allow it to be reapplied when a new problem is encountered. In general terms, a pattern describes a solution to a problem in a recurring manner. A pattern language, in turn, is a way of subdividing a general problem and its complex solution into a number of related problems and their respective solutions.

Thus, considering the activities described in the MOOCs Life Cycle Process and the recurring solutions to implement them, an Educational Design Pattern Language for MOOCs (EDPL) was developed and validated [6], and is presented in Figure 2.

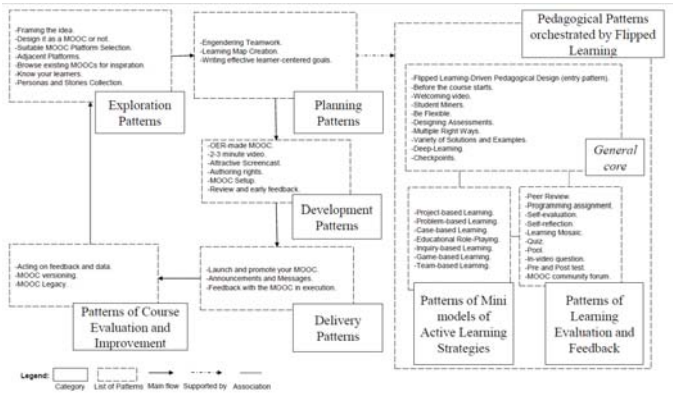

FIGURE 2 – EDUCATIONAL DESIGN PATTERN LANGUAGE GRAPH.

A brief description of the main patterns, in the form of patlets (problem-solution pair), is presented next. The complete set of patterns has been omitted due to lack of space, but they are available at http://caed.icmc.usp.br/mooc. The issue of development of the EDPL is elaborated further in [6, 7].

TABLE 1 – SOME PEDAGOGICAL PATTERNS BASED ON FLIPPED LEARNING.

Pattern Name	Brief Description (problem-solution pair)
BEFORE THE COURSE STARTS	You want to motivate active learning since the beginning of your MOOC. You already are using PATTERNS OF MINI MODELS OF ACTIVE LEARNING STRATEGIES to support the course itself. **Therefore**, you should define one or more activities that students must take before the course starts, in order to activate prior knowledge, prepare them to take a more active role, and create a connection with the rest of the course. This will also encourage the development of competencies related to self-management or self-regulation of learning. Solutions include open-ended questions, discussion forum, readings, the creation of an initial project without using the concepts which will be covered in the course later. Student Miners is also an option. In addition, PRE TEST can also be used (PATTERNS OF LEARNING EVALUATION AND FEEDBACK).
WELCOMING VIDEO	When launching a new course, or meeting students for the first time, one of the most important activities to plan is how to make a good start. A welcoming presentation has the potential to skew the students' opinions of the content being presented. As popular culture tells us, "the first impression is the last impression", and in the MOOC context this situation is amplified because of the diversity and openness. **Therefore**, to head this off, consider adding a welcoming video to the MOOC as well as the first thing a student should see when enrolling. Make it as compelling as you can. A welcoming video can also explain key aspects of the syllabus, class expectations, and instructors' personal biographical information. If the course is based on PATTERNS OF MINI MODELS OF ACTIVE LEARNING STRATEGIES, you should describe the main strategy used.
BE FLEXIBLE	Instructors want to deliver MOOCs and students want to learn through a MOOC because of many factors, but flexibility is a

	striking one. As an instructor, it is difficult to decide which parts of your MOOC can be flexible or fixed, or how to turn a MOOC more flexible without destroying the main properties (massivity, virtual, openness, and course). The delivery mode, platform functionalities, the MOOC subject area, for example, can interfere. **Therefore**, try to consider the core levels of flexibility, such as access, interaction with resources, allow different languages, flexible assessment tracks, self-paced learning, among others.
DEEP-LEARNING	Your MOOC's target audience is still passive, very focused on achieving a certificate at the end of the course, or they are novices in the MOOC context, but you want to encourage them to engage in authentic learning tasks for deep-learning, creation and knowledge sharing. **Therefore**, during your LEARNING MAP CREATION, try to contemplate activities that stimulate the development of high-level abilities, such as draw, build, program, criticize, experiment, moderate, compare, organize, among others. It is also related to WRITING EFFECTIVE LEARNER-CENTERED GOALS. Moreover, you should include activities that encourage students to use digital technologies to create and share their own content (blog, videos, texts, etc.), and actions that motivate learners to discuss and learn more, even after the end of the course, so that they can act as agents of transformation in their contexts. You can also choose PATTERNS OF MINI MODELS OF ACTIVE LEARNING STRATEGIES.
CHECKPOINTS	In flexible and non-linear MOOCs, the interaction between participants is essential to the MOOC success. However, participants work on the learning activities at a different pace, and sometimes even the order of undertaking activities differs, making it hard to synchronize the learning experiences between participants. Some participants diverge into independent explorations branching out of the MOOC activities. Sharing these could enhance the social learning experience, but at the same time, it makes synchronization even harder. **Therefore**, you should create regular "checkpoints", which allow participants to synchronize their own learning with the course flow and pace, catch up on the social vibe and follow the recent highlights. Such checkpoints could be synchronous events, recorded for those who cannot attend at the time. They can also be asynchronous events, such as forum posts or emails.

TABLE 2 – EXAMPLE OF PATTERN RELATED TO ACTIVE LEARNING STRATEGIES ADAPTED TO MOOC CONTEXT.

Pattern Name	Brief Description (problem-solution pair)
PROJECT-BASED LEARNING	You want to develop an authentic, flexible, and active MOOC using active learning strategies. You may know how to apply those strategies in a face-to-face context or even a formal virtual course, but you are unaware of any methods for adapting and applying these strategies in a massive and virtual context. **Therefore**, considering that such strategies are not new and there are existing patterns related to them, mainly from face-to-face environments, the presented adapted solutions describe steps to apply each of them considering their specificities. Specific solutions are omitted due to lack of space, but the patterns describe steps to apply the related active learning strategy but adapted to the MOOC context.

TABLE 3 - SET OF PATTERNS THAT SUMMARIZE EVALUATION STRATEGIES ADAPTED TO THE MOOC CONTEXT.

Pattern Name	Brief Description (problem-solution pair)
PEER REVIEW PRE AND POST-TEST QUIZ SELF-EVALUATION SELF-REFLECTION	You want to use different strategies to support learning evaluation and feedback in the context of online and massive course, but you are unaware of techniques for doing that. What adaptations are needed for this context? **Therefore**, in general, you should try to make use of self-assessments based on clear objectives and criteria through rubrics or scales, which can be used by the students to evaluate their performance and knowledge development. Self-assessments can also support further reflection and help students to verify their understating of activities or content. Assessment activities should be authentic, based on real contexts. Moreover, the questions should be more divergent than convergent. That is, most of the questions can be answered in various ways, considering several options. If the MOOC platform allows, you should consider using videos with quizzes or simple internal questions that need to be answered before continuing with the course. Specific solutions were omitted due to lack of space, but the patterns describe steps to apply the related strategies but adapted to the MOOC context.

TABLE 4 - SET OF PATTERNS TO SUPPORT MOOC DEVELOPMENT PHASE.

Pattern Name	Brief Description (problem-solution pair)
2-3 MINUTE VIDEO	You are aware that creating a full-length video of your lecture for online delivery is too demanding, from both a teaching and a student learning perspective. You cannot retain students' enthusiasm and passion for a full hour video, and you know that their attention will drop dramatically at various points across a long single recording. **Therefore**, you need to create smaller discrete episodes that you feel confident learners should be able to watch in their entirety. Current studies about MOOC videos indicate frequent changes in the size of videos for MOOCs. The smaller the better. In addition to learning and engagement aspects, technical issues (i.e. mobile devices, networking) also contribute to the adoption of even shorter videos. Thus, you should organize your video lecture material into sensible two or three-minute chunks. You can then be confident that students will stay focused throughout the duration of the video. This could be checked through the use of analytics within your MOOC platform.
ATTRACTIVE SCREENCAST	You have decided that you are going to do a screencast on some topic. You have developed some ideas about what you would

	like to show. How do you transfer that into a screencast without missing the important topics or making the screencast confusing? **Therefore**, you should begin by making a written draft of your screencast script. The script should be detailed enough to know what it should include without missing the important topics. A good script has clear beginning and end. The beginning should tell the viewer what the screencast is all about. The end should have a strong summary of what the screencast has taught or accomplished. Another point is to set a time limit on how long the screencast should take. Make this a hard limit for yourself as the screencaster. This allows you to focus on the content at hand without spending too much time on the inconsequential parts of the screencast. As a rule of thumb, a screencast should not be longer than 5 minutes. Also, you should make a deliberate effort to take about twice as long as you normally would complete the task. This prevents you from moving your mouse too quickly. Another tip is to prepare the text snippets and other resources that you would use beforehand and store them into a file or auxiliary folder to avoid distractions, prevent recording typos and save time. Also, have in mind that silent screencasts tend to be less engaging. In addition, a good screencast should not contain any user interface elements that could distract the viewer from the main task at hand, for instance, e-mail alerts, blinking icons, the time on the menu bar or even a provocative desktop wallpaper. Don't forget to momentarily emphasize your mouse cursor to attract viewer's attention.

Next, we describe the design of our MOOC on Web Development, organized along the main phases defined in the Life Cycle (Figure 1) and using the main patterns previously described.

IV. A Case About Web Development Education

In mid-2017, IFSULDEMINAS signed a partnership with the company that manages the MiríadaX platform. The institution became the first Brazilian Federal Institute of Education, Science and Technology partner of MiríadaX and joined a partnership of other 105 institutions around the world. Since that, the institute's faculty has been boosted to deliver MOOCs using such web environment. Next, we describe the MOOC named Web Development with Bootstrap, CodeIgniter and Agile Practices, which is the IFSULDEMINAS' initial course launched on such platform.

A. Exploration

The patterns Know your Learners and Personas and Stories Collection (Figure 2) were used to understand the audience' knowledge on web development and agile practices and to collect user stories about the skills they would like to develop by attending a MOOC on such context. The next step was to use a mind map (Figure 3) to resume and organize the main needs, curiosities, suggestions as well as characteristics and content appointed by the target audience and collected from a virtual questionnaire.

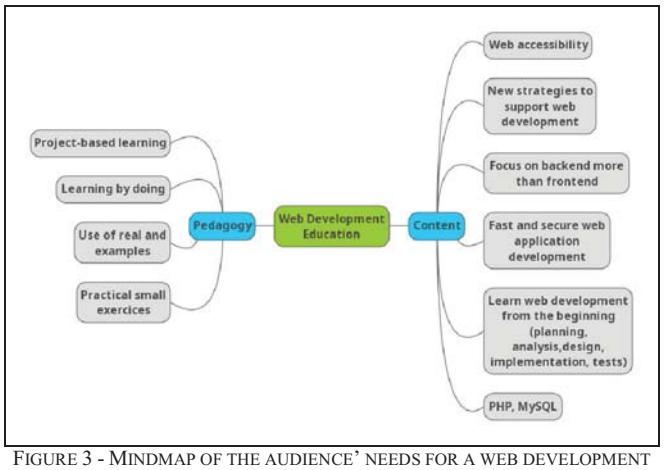

Figure 3 - Mindmap of the audience' needs for a web development MOOC.

As we previously mentioned, the MOOC was delivered through MiríadaX Platform. The courses on the platform consist of compulsory and optional activities. Course content is organized in modules. Each module may contain one or more external resources (e.g., video, figure, text from external servers), poll (e.g., a test to know students' opinion about a topic), P2P (i.e., pair evaluation), and multiple-choice questions (i.e., questions where the student should choose one or more correct answers). The platform also provides a discussion forum, tools to support instructors-students' communication in an automatic way, general reports on the course statistics (e.g., demographic data, students' progress in the course). Data can also be exported to be analyzed by external approaches and tools.

B. Planning

Considering the main syllabus and MiríadaX available resources, we used the LDF4MOOCs framework to develop the activities map related to the course. We specifically used ideas from Flipped Learning and Project-based learning.

The main topics of the map, learning outcomes (objectives), activities as well as the main patterns in the MOOC design are described in Table 5.

Table 5 - Main features of the MOOC named Web Development with Bootstrap, CodeIgniter and Agile Practices.

Objectives	Main Activities	Main patterns considered
Topic: Before starting the Course		
Examine related knowledge prior to the course; Identify the course organization; Define your own learning goals.	Welcome video; Introduce yourself using a video or text; Learners' characterization survey; Pre-test; Self-evaluation; Initial question to active prior knowledge.	Before the course starts; Welcoming video; Be flexible; Self-evaluation; Pre and Post-test.
Topic: Web Development Fundamentals		
Draw connections between prior knowledge on web development and the content to be approached by the course.	Use of a model system and hands-on activities to explore the concepts related to web development fundamentals; Quizzes.	Deep-learning; Quiz.
Topic: Towards the Practical Project		
Identify the concepts related to the use of agile	Lecture on agile practices.	2-3 minute video; Deep-learning; Peer review;

practices in the web development; Organize a Requirements Analysis Document for your practical project by using agile practices; Define a viable minimum version of a software to be implemented during the course.	Hands-on activities related to the requirements analysis document creation by using agile practices. Pair-evaluation of the document. Video-interviews with agile development practitioners.	PROJECT-BASED LEARNING.
Topic: Bootstrap - User Interface prototyping		
Identify the concepts related to the use of Bootstrap; Prototype and implement the main user interfaces related to the viable minimum version of your project.	Demonstration video about Bootstrap: guide of installation, configuration, among other features; Hands-on activities related to paper prototypes and implementation of user interfaces by using Bootstrap.	ATTRACTIVE SCREENCAST; DEEP-LEARNING; CHECKPOINT; SELF-REFLECTION.
Topic: CodeIgniter - Backend		
Identify the concepts related to the use of CodeIgniter; Prototype and implement the backend related to the viable minimum version of your project; Implement the integration between Bootstrap and CodeIgniter considering the code developed for your project.	Demonstration video about CodeIgniter: guide of installation, configuration, among other features; Hands-on activities related to the integration of backend (CodeIgniter) and frontend (Bootstrap); Testing activities: lecture and hands-on activities.	ATTRACTIVE SCREENCAST; DEEP-LEARNING; CHECKPOINT; SELF-REFLECTION.
Topic: Additional Resources		
Identify additional code resources, new functionalities or new viable minimum versions to be implemented in your project.	Demonstration videos and texts to support the students in the selection and development of new functionalities for their projects.	ATTRACTIVE SCREENCAST; CHECKPOINT.
Topic: Delivery of the Practical Project and Pair-Evaluation		
Test the design and deliver the viable minimum version of the project constructed during the course; Judge the version shared by other colleagues using a specific rubric.	Demonstration report about hosting applications on the web using free resources; Pair-evaluation of the practical projects delivered by the end of the course.	PEER REVIEW.
Topic: Consolidation and Experiences		
Compare your software version to your colleagues' versions; Value and critique your performance in the course.	Post-test; Final self-evaluation; Course evaluation.	PRE AND POST-TEST; SELF-EVALUATION;

C. Development

Specific videos were recorded and edited using the IFSULDEMINAS media service. Content in each week consisted of approximately five or six small videos, from one up to five minutes of duration. All the learning materials used in the course, such as videos, images, and texts, were also based on common Open Educational Resources (OERs) guidelines. Finally, the Learning Map (Table 5) was implemented in MiríadaX. Figure 4 shows the course editing view used by the instructors.

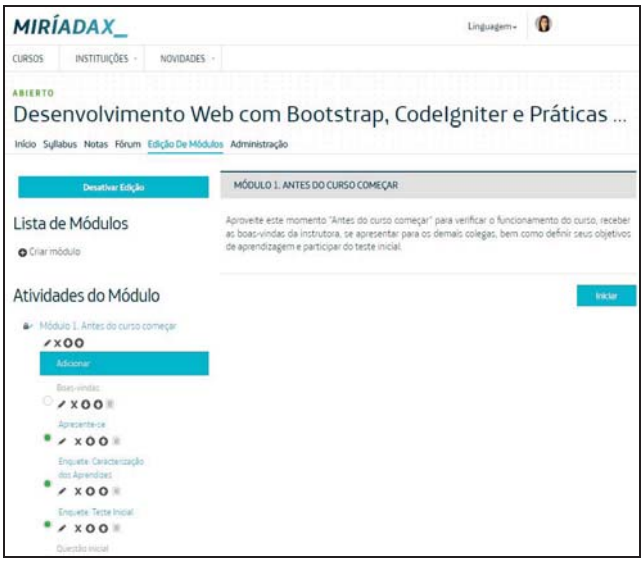

FIGURE 4 - MODULE 1 TOPIC HEADINGS AS DISPLAYED IN MIRÍADAX.

Before launching the MOOC, the MiríadaX Teaching and Learning team analyzed it and sent us a report with recommendations. This allowed us to improve the course before it reaches hundreds of users.

D. Delivering

The course was launched in November 2017 and stayed online for ten weeks. When the course opened, only the first module, named "Before starting the Course", was released. It was reserved to introduce some initial questions to motivate registered users to reflect on concepts to be worked during the course, to conduct a pretest, and to support self-regulated learning.

It was mandatory to conclude the first module to advance the course and access the remaining modules. Despite such requirement, the students were free to complete the course at their own pace.

E. Evaluation and Improvement

The first edition of the course had 665 enrolled students. From those, 465 (69.92%) effectively began the course. 44 of them (9.5%) reached the last week. From those, 15 (34%) achieved all the requirements to receive a certificate, i.e., a completion rate of 3.22%. The requirements were: watching the required videos, performing compulsory tasks, delivering the proposed software, and attending a pair-evaluation, according to the MOOC Learning Map (Table 5). The main reasons for dropout: lack of time for hands-on activities; other urgent concerns (face-to-face course requirements), technical difficulty to install software, among others.

Regarding the geographical distribution, 661 of the 665 registrants identified their locations. They were drawn from different countries of the world, according to Figure 5. The countries with the highest enrolments are shown in Table 6.

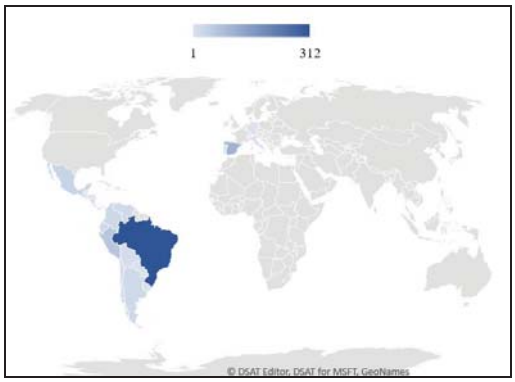

FIGURE 5 - GEOGRAPHICAL REPRESENTATION.

TABLE 6 – GEOGRAPHICAL DISTRIBUTION.

Country	Participants
Brazil	312 (47.2%)
Spain	106 (16%)
Peru	61 (9.2%)
Mexico	33 (5%)
Colombia	25 (3.8%)
Equador	24 (3.6%)
Argentina	23 (3.5%)
Venezuela	18 (2.7%)

In addition, 11 were from Chile (1.7%), 11 from Bolivia (1.7%), 7 from Portugal (1.1%), 5 from Panama (0.8%), and 4 from Paraguay (0.6%). Uruguai, Guatemala, El Salvador, Nicaragua had 3 registrants each one (0.5%). Dominican Republic and Costa Rica had 2 each one (0.3%). Finally, Italy, Germany, Singapore, Cuba, and Cape Verde Island had 1 each one (0.2%).

Regarding the personal characteristics, of the 186 registrants who identified their genre, 82.2% were male and 17.8% were female.

Of the 174 participants who identified their age, 39.7% were between 25 and 34; 23.6% were between 18 and 24; 21.3% were between 35 and 44; 11.5% were between 45 and 54, and 4% were between 55 and 64 years old.

Of the 171 participants who identified their educational or professional qualifications, 47.4% were undergraduate students; 31.6% had an undergraduate education, 14.6% were teachers or researchers in universities; 4.1% were university staff (administrative work), and 2.3% had a high school education.

TABLE 7 – DIMENSIONS OF DESIRED CHARACTERISTICS FOR MOOC LEARNING MAPS [10].

Dimensions
(1) Competency-based design
(2) User-centered learning
(3) Self-Regulation Learning
(4) Social Networking and Collaborative Learning
(5) Deep Learning, knowledge creation and sharing
(6) Assessment and Understanding
(7) Accommodate differences
(8) Feedback

In addition to the social and demographic data analysis, we verified the students' opinion about interventions and design decisions made for the related MOOC. The questions were grouped into eight dimensions by considering the desired characteristics for MOOC Learning Maps, i.e., MOOC projects, as presented in Fassbinder et al. [4]. Table 7 summarizes the dimensions. Some questions are briefly described next.

- I liked the MOOC used Project-Based Learning as the main teaching strategy (Competency-based design) (Figure 6).

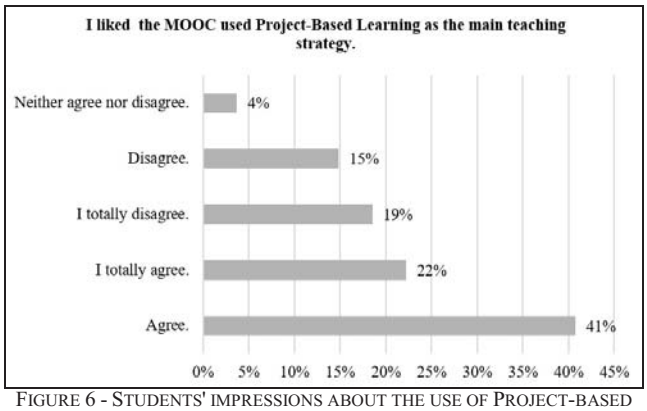

FIGURE 6 - STUDENTS' IMPRESSIONS ABOUT THE USE OF PROJECT-BASED LEARNING AS A TEACHING STRATEGY.

- I liked the course contemplated a teaching process more oriented to the achievement of practical activities than a theoretical exposition of content (User-centered learning).
- I liked the course used short videos (1 to 5 minutes) to guide the students in the activities that should be developed (User-centered learning).
- I felt engaged, motivated and able to manage my own learning (Self-regulation learning).

V. IMPLICATIONS FOR PRACTICE

This section details some lessons learned from the design and delivery of the *Web Development with Bootstrap, CodeIgniter and Agile Practices MOOC* previously described. They are related to the desired characteristics for MOOC Learning Maps, as summarized in Table 7. Some technological and pedagogical issues of designing MOOCs to support Web Development Education are described.

A. Competency-based design

Aligning the MOOC syllabus with students and job market' needs is essential but challenging. Interventions need to be performed in order to keep the course open to everyone. In our MOOC, there is a module related to web fundamentals, e.g., HTML, CSS, among others. We designed this module to be mandatory for novice students in the subject. Advanced students should use the module to remember important points related to web development. Students were also free to choose different services or tools to host their projects in the Web. We used videos recorded by professionals to highlight important skills required for a web developer. Furthermore, we used Project-Based Learning as the main teaching strategy to guide and support students in carrying out activities that promote the development of web development skills, competencies, and personal factors.

B. User-centered learning

The MOOC syllabus was identified through a survey of potential users and professional software engineers. At the beginning of the course, we used activities that activated student's past experiences with web development, such as self-evaluation and forum. In addition, students had the opportunity to take our MOOC at their own pace.

C. Self-Regulation Learning

In our MOOC, we used different strategies to encourage self-regulated attitudes towards learning. For instance, the students were asked about *"Have you specified one or more learning goals you want to achieve by the end of the course? No? So, try to complete the sentence: "By the end of the course, I hope to learn or be able to do X, with the purpose of Y (e.g., apply in my work, develop a school task, improve my knowledge in order to get a job). For this, I will use the following Z strategies (e.g., study 2 hours a week, improve my time management, set priorities)*.

However, when students were asked about the effective definition of learning goals to be achieved by the end of the course, we identified an irregular distribution in the related answers (Figure 7). Students had difficulty to define their learning goals, manage time, and control the own learning process. This gap deserves special attention by instructors and demands further investigation. MOOC platforms also need to consider new features to support self-regulation learning.

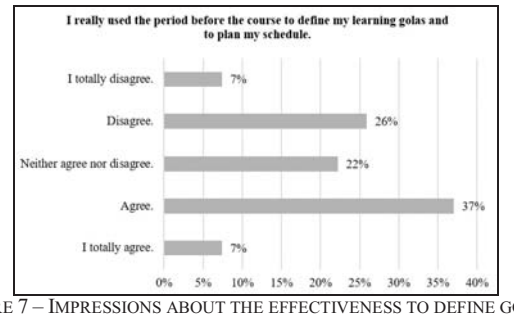

FIGURE 7 – IMPRESSIONS ABOUT THE EFFECTIVENESS TO DEFINE GOALS.

D. Social Networking and Collaborative Learning

Our course attempts to motivate collaborative and communication skills, which are essential in web development, through forum and pair review. Development in pairs is also a skill required, but the construction of the proposed software was carried out individually. In previous studies [4], we identified that more than 70% of self-paced distance learners seem to be open to peer collaboration but prefer working independently. MOOC platforms also need to consider new features to support collaborative activities that involve programming topics.

E. Deep Learning, knowledge creation and sharing

The proposed MOOC uses activities that motivate top-level thinking skills development [8], such as: (i) introduce yourself using a video; (ii) use of a model system and hands-on activities to explore the concepts related to web development fundamentals; hands-on activities related to the creation of a requirements analysis document by using agile practices; hands-on activities related to prototypes and implementation of user interfaces by using Bootstrap; hands-on activities related to the integration of backend (CodeIgniter) and frontend (Bootstrap). We also encouraged students to use digital technologies (e.g., Github) to share their own software; and to discuss and learn more, even after the end of the course, so they can act as agents of transformation in their contexts.

F. Assessment and Understanding

We prioritize formative rather than summative assessments. Case studies were used to support assessments activities and were based on real contexts. However, finding authentic case studies on web development is time-consuming because there are no educational repositories to support such task.

G. Accommodate differences

The proposed MOOC is open for students with different levels of background in web development. They had the opportunity to choose the subject of the software to be developed during the course. Other personalized adjustments were also considered to accommodate several students' backgrounds, needs, and learning styles.

H. Feedback

Two tutors were available to answer the questions posted by students in the discussion forum. Weekly e-mails were also automatically sent by the instructor. The course was available in Portuguese; students could use Portuguese, Spanish or English to communicate in the forum. The interaction between Portuguese and Spanish speakers was not a problem, as exemplified next, but the social presence in the forum was still low.

Student A (Question in Spanish): *"Al importar la base de datos me mustre este error. Alguien me puede sugerir que debo hacer. Muchas gracias."*
Student B (Answer in Portuguese): *"Consegui importar o SQL fazendo algumas alterações. Retirei alguns comentários e linhas de código que pouco interferem na criação das tabelas. Segue em anexo."*
Student A (Thanks in Spanish): *"Muchas gracias."*

VI. CONCLUSIONS AND FUTURE WORKS

This paper has presented an overview of MOOCs in the context of Web Development Education and analyzed them from the perspective of the challenges instructors have faced when designing for learning. A case study about a web development-based MOOC is detailed. The learning design strategy used to design the course is also summarized. The Learning Design Framework for MOOCs (LDF4MOOCs) is based on theoretical principles from (i) Massive Open Online Courses; (ii) Flipped Learning and active teaching-learning methodologies as a pedagogical strategy; and (iii) Educational Design Patterns and Pattern Language as a strategy to record and share tacit design knowledge.

We discuss our findings evidencing that there are still several technological and pedagogical challenges that need to be addressed so as to enhance the MOOCs learning experience for web development education. Instructors can use the lessons learned to improve web development course design in the massive, open and online context.

Our work in this area is ongoing. One MOOC in the Software Testing context is under development. In addition, we plan to keep investigating new methods and strategies to support lifelong learning in Web Development Education.

ACKNOWLEDGMENT

The authors are grateful to the Brazilian funding agencies (CNPq, Fapemig, Fapesp), CAPES (Procad 071/2013) and the IFSULDEMINAS for their financial support.

REFERENCES

[1] C. S. Miller and Y. R. Connolly, “Introduction to the Special Issue on Web Development,” ACM Transactions on Computing Education (TOCE) - Special Issue on Web Development, v. 15, n. 1, 2015.

[2] B. B. L Lim, “Teaching Web development technologies in CS/IS curricula,” Proceedings of the twenty-ninth SIGCSE technical symposium on Computer science education. Atlanta, Georgia, USA: 1998.

[3] G. Gay, N. Djafarova and L. Zefi, “Teaching Accessibility to the Masses,” Proceedings of the 14th Web for All Conference on The Future of Accessible Work - W4A ’17.

[4] A. G. O. Fassbinder et al., “Massive open online courses in software engineering education,” 2017 IEEE Frontiers in Education Conference (FIE), Indianapolis.

[5] L. Barroca and K. Kear, “Agile practices taught online at a distance,” The Online, Open and Flexible Higher Education Conference. 2016.

[6] A. G. O. Fassbinder, G. Magoulas and E. F. Barbosa, “Towards an Educational Design Pattern Language for Massive Open Online Courses (MOOCs),” In: 24th Conference on Pattern Languages of Programs (PLoP). Vancouver, Canada, 2017a.

[7] A. G. O. Fassbinder, G. Magoulas and E. F. Barbosa, “Developing an Educational Design Pattern Language for MOOCs,” In: XXVIII Brazilian Symposium on Computers in Education (SBIE). Recife, Pernambuco, Brazil, 2017b.

[8] D. R. Krathwohl, “A revision of Bloom's taxonomy: An overview. Theory into practice,” v. 41, pp. 212-218, 2002.

[9] Y. D. Wang, "Teaching web development at a distance," Proceedings of the 2011 conference on Information technology education. ACM, 2011.

[10] T. H. P. S. Wiedenbeck, "Learning Web Development: Challenges at an Earlier Stage of Computing Education," 2011.

[11] M. Arimoto, L. Barroca and E. F. Barbosa, “AM-OER: An agile method for the development of open educational resources,” Informatics in Education, v. 15, n. 2, p. 205–233, 2016.

[12] M. Cohn, “User stories applied: For agile software development,” Addison-Wesley Professional, 2004.

[13] A. Filatro, “Design Instrucional na Prática,” Pearson, 2008.

[14] A. G. O. Fassbinder, M. Fassbinder, E. F. Barbosa and G. Magoulas, "Towards a MOOC Design Model based on Flipped Learning and Patterns: A Case on Introductory Courses," XXI Congresso Internacional de Tecnologia Educativa (TISE), Santiago, Chile. 2016.

[15] A. G. O. Fassbinder, M. Fassbinder and E. F. Barbosa, "From flipped classroom theory to the personalized design of learning experiences in MOOCs," Frontiers in Education Conference (FIE), IEEE, 2015.

[16] J. Bergmann and A. Sams, “Flip your classroom: Reach every student in every class every day,” International Society for Technology in Education, 2012.

[17] R. Koper and C. Tattersall, "Preface to learning design: A handbook on modelling and delivering networked education and training," Journal of Interactive Media in Education, 2005.

[18] R. Pressman and B. Maxim, “Software Engineering,” 8ª Edition. McGraw Hill, 2014.