

# Developing Open Educational Resources through Learning Design and Agile Practices

Maurício M. Arimoto<sup>1, 2</sup>, Leonor Barroca<sup>3</sup>, Ellen F. Barbosa<sup>1</sup>

<sup>1</sup>Institute of Mathematics and Computer Sciences, University of São Paulo (ICMC/USP)  
São Carlos – São Paulo – Brazil

<sup>2</sup>Centre of Technological Sciences, State University of Northern Paraná (UENP)  
Bandeirantes, Paraná – Brazil

<sup>3</sup>Faculty of Mathematics, Computing and Technology, The Open University (UK)  
Milton Keynes – United Kingdom

mauricio@uenp.edu.br, l.barroca.@open.ac.uk, francine@icmc.usp.br

**Abstract.** *One of the current challenges related to Open Educational Resources (OERs) is how to produce quality and relevant materials to be reused and adapted to different contexts and learning situations. In this paper we present a flexible and systematic method for OERs, called AM-OER, that allows the development of OERs to evolve incrementally, and be modified and improved as needed. Practices of Learning Design are incorporated into the OERs development, making the design more understandable and shareable, and facilitating the reuse and adaptation. We applied AM-OER through a case study involving the development of a course in the software engineering domain. The results obtained suggest the applicability and usefulness of the AM-OER in the development of OERs.*

## 1. Introduction

Open Educational Resources (OERs) have emerged together with the changes and advancements in computer and communication technologies, which facilitate access to information and knowledge and encourage cooperation and collaboration. OERs enable open and flexible process of learning, providing suitable and tailored opportunities to individual needs. They can be characterized as “educational materials that are openly available for use by educators and students, without an accompanying need to pay royalties or licence fees” [Butcher 2015].

OERs can play an important role in the transition to an educational system in which innovation, customization, high quality, cost reduction and efficiency go together. However, one of the challenges faced by developers and practitioners of OERs has been how to produce quality and relevant materials that can be reused and adapted in different context and learning situations. Research in the literature also reports the difficulty to reuse and adapt third-part OERs to their own context [Amiel et al. 2011, OPAL 2011].

Recently, research on Learning Design (LD) has increased due to its potential to support in the definition and creation of a more effective pedagogical design of learning materials [Conole 2013]. An example is the Open University Learning Design Initiative (OULDI) project proposed by the UK Open University to derive a more “practice-focused approach for Learning Design” [Brasher et al. 2012]. It includes a set of practices and artifacts to represent design: (1) Course Map View: shows the main components for the course, enabling educators to think about the design of a course; (2) Learning Outcomes View: maps learning activities and assessment with the learning outcomes of the course; (3) Pedagogy Profile: articulates type of activities in which learners undertaken during the course; (4) Course Dimensions: provides more information on the nature of the course; and (5) Task Swimlane View: maps tasks that learners undertake to the contents and tools they use during the course.

In most initiatives related to the delivery of OERs, the production activities are considered the most costly, highlighting the need for an effective production process for a sustainable delivery of OERs [Schuwer et al. 2010]. However, there is a lack of initiatives to foster the design and creation of quality OERs with reduced time and costs. In [Masson and Udas 2009] it is discussed the use of principles and practices of agile methods for enabling adoption and management of OERs. The authors proposed an agile approach only focused on management of OERs. In [Patricia et al. 2010] a life cycle for OERs with a set of well-defined phases was defined, based on ADDIE model [Molenda 2003]. This life cycle is plan-driven, with little flexibility and lack of involvement of users (especially educators and learners) throughout the development. Approaches for the development of Learning Objects (LOs) can be found [Kemczinski et al. 2012, Braga et al. 2013, Lapolli et al. 2010]. Such approaches are focused only on the development LOs, and may not be suitable for the context of OERs.

In this context, we propose an Agile Method for the Development of OERs (AM-OER) based on practices from agile methods [Ambler 2001, Schwaber and Beedle 2002], which provide flexible and suitable approaches for elaboration of products in constant evolution, such as OERs. We also incorporate into the AM-OER method, practices of LD from the OULDI project [Conole 2013]. The idea is to facilitate the reuse and adaptation of OERs and to contribute to their quality by embedding pedagogical design practices. A case study was performed in order to validate AM-OER by applying it in the design and creation of an OER on Free/Libre and Open Source Software (FLOSS). The obtained results suggest that AM-OER is effective in the design and creation of OERs.

This paper is organized as follows. In Section 2 we describe the AM-OER; In Section 3 we report the results of a case study conducted to validate AM-OER by designing and creating an OER on FLOSS; concluding remarks and future work are presented in Section 4.

## 2. AM-OER Overview

In Figure 1 we provide an overview of the AM-OER, that proposes a flexible life cycle for the development of OERs, allowing the team to make changes, if necessary, throughout the process. The life cycle has three phases: (1) Kickoff; (2) Development; and (3) Sharing.

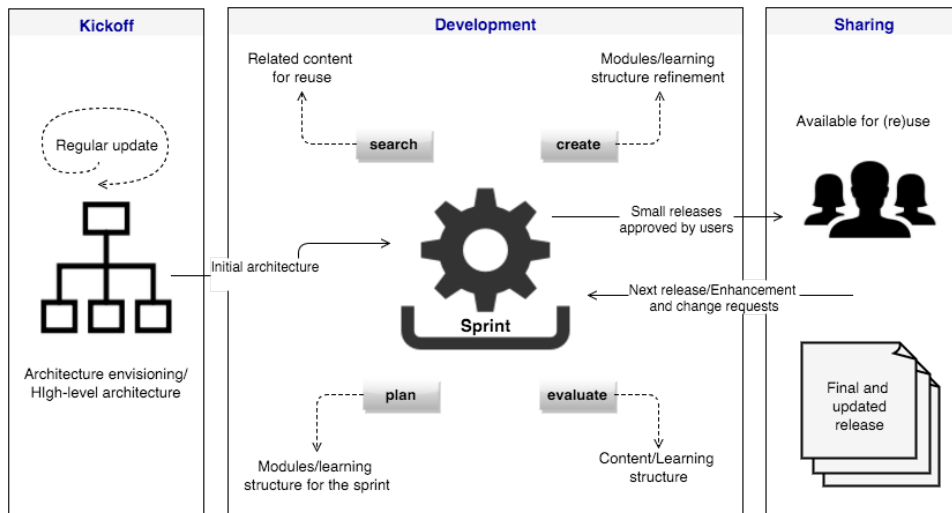


Figure 1. AM-OER life cycle for OERs.

The OER development project starts from the learning needs and/or problems identified by the educators. A high-level architecture with the main components needed to the OER is sketched. The team gets together to plan the OER releases (small deliveries of OER) and the sprints (predefined period of time) to develop them. They define which parts (modules) of

OER will be developed firstly, in the current sprint. Modules are developed iteratively and incrementally by short sprints, with a sprint lasting from days to few weeks. During the sprint, team members communicate and constantly interact with each other to discuss the activities, monitor their progress, and to identify the main obstacles that hinder the progress of work.

At the end of the sprint, the team gets together to evaluate the produced release and the progress of the sprint as a whole, identifying changes, improvements and needs for new components. Small releases of the OER approved by the educators are delivered in a relatively short period of time to be used by learners, obtaining important feedback about their learning experiences on the OER. The process is repeated until all planned releases are developed and a final version of the OER is delivered (final and updated release). Next, we provide an overview of the main phases of AM-OER.

## 2.1. Kickoff

At the beginning of the OER project (sprint #0), users (educators and learners) come together with the development team to identify and discuss on the learning needs and/or problems to be addressed by the intended OER and its overall structure.

Educators start by establishing the general goals of the intended OER, i.e., what they intend to cover/present in the OER to be developed. For instance, the general goal of an OER could be: (1) to explain the fundamentals of FLOSS development; or (2) to provide students a general introduction to FLOSS licenses.

After that, educators sketch the learning objectives of the OER, consisting of specific statement of teaching/learning intention. Many OERs do not have basic elements of LD such as learning objectives. This makes it hard to assess the OER in terms of its overall purpose and intended learning outcomes, and of the pedagogical alignment of learning materials, activities and assessments [DeVries 2013]. Examples of learning objectives statements may be: (1) students would understand the main ideas behind FLOSS licenses; or (2) students will be able to select an appropriate license for using in a product.

The context or domain in which the OER will initially intended to be applied should also be specified. The context should also address cultural and languages issues. For instance, learning materials in English are very common worldwide, but cultural context of learning usually differs from one country to another. Then, these issues should be considered during the design and creation of OERs.

Finally, the main components of the OER are sketched, but without too much detail, following the concept of **architecture envisioning** from agile methods [Ambler 2001], as the design should be constantly evolving throughout the sprints. LD practice from OULDI [Brasher et al. 2012] are used together to identify and map the main components of the OER:

- Learning contents: such as lessons, lab activities, study guides/guidelines, readings, examples, support materials, case studies, pilot projects, among others.
- Learning assessments: such as in-text questions, self-assessment questions, oral or written presentations, reports, essays, among others.
- Dialogical and collaboration aspects related to the usage of OER: such as instant messaging system, chat, social networking, mailing list, brainstorming sessions, peer-to-peer works, work in groups, among others.

As a result, an *Initial Architecture View* is created, being used as input for the next phase. At the end of this phase, the whole team should agree on the OER to be developed and delivered at the end of the project.

## 2.2. Development

The whole team comes together to conduct the **sprint planning** [Schwaber and Beedle 2002], wherein small modules of the OER should be prioritised by the educators and selected to be

developed during the current sprint, based on the `Initial Architecture View`.

The learning structure for the OER is sketched iteratively and incrementally, following the concept of **iterative modelling** from agile methods [Ambler 2001]. LD practices from OULDI [Brasher et al. 2012] are used together to map the main components of the learning structure, including the learning activities, and their connections to the intended learning outcomes, the contents, the tools and assessments. The result produces the `Learning Structure View` for the current sprints. This is and mandatory artifact that is used, updated and integrated throughout the development.

The primary metadata for the OER should be identified and gathered. Metadata describes relevant characteristics of the OER, facilitating its reuse and recovery by search engines. When an OER has integrated metadata, any user can easily find it [Madden 2010].

Following the concept of **design storming** from agile methods [Ambler 2001], the team comes together with the **active participation of users** [Beck and Andres 2004] to create a JIT design to sketch specific aspects of the OER activities flow. By incorporating practices of LD from OULDI [Brasher et al. 2012], the team members refine and decompose the OER activities in simpler activities and atomic tasks, helping them to reflect upon one aspect of the design solution and how they can transform it for a more effective OER with embedded pedagogical design practices. The result culminates with the `Learning Task View`.

The OER activities and contents should be created, including any kind of content and activities such as open textbooks, html pages, lecture notes, audios/videos, images, animations, tests, among others. The team should check for related contents to be reused to compose the intended OER. It may be easier or more efficient to reuse and adapt existing contents than creating new ones [Leverly 2012]. Regardless of reuse third-party material or not, an OER is developed by the team through several and repetitive sprints, culminating with the delivery of **small releases** [Beck and Andres 2004] at the end of a sprint or small set of sprints.

The licensing policies to share the OER with others should be established. OER implies the use of open licenses with little or no restriction. Open license enables sharing an OER without payment of a royalty or license fees.

At the end of the sprint, the team conducts the **sprint review** [Schwaber and Beedle 2002] to discuss, evaluate and approve/disapprove the OER module produced within the sprint. In order to assist in the OERs evaluation we proposed a set of assessment criteria classified into three categories:

1. Didactic-pedagogical, such as: (1) accuracy of the contents for the learning objectives; (2) relevance of the contents/fits for purpose; and (3) pedagogical design of materials.
2. Technical, such as: (1) interoperability; (2) accessibility; (3) usability; (4) discoverability; and (4) localisation/globalisation capabilities.
3. Legal, such as: (1) intellectual property and rights; and (2) open licenses policies for the contents.

Finally, the team conducts the **sprint retrospective** [Schwaber and Beedle 2002] to evaluate the overall sprint execution. They should identify “what worked well”, “what did not work well” and “what needs to change and improve”. Lessons learned and feedback from the evaluation should be analyzed, gathered and used for changing and improving the following sprints, contributing to the continuous improvement process.

### 2.3. Sharing

The OER release is delivered in a learning environment for use by learners. This is critical to identify weaknesses and propose improvements. Educators should provide support to learners in their activities and monitor their progress and achievements. Data about the learner’s experience should be collected and analyzed to improve the quality of the OER release.

Effective access to the OER should be allowed through platforms, repositories, and institutional or stand-alone websites together with associated metadata and license with few or no restriction on its (re)use. Web 2.0 technologies and social software can be used as support such as, microblogging and social networking. Slideshare and Flickr are easy mechanisms for sharing OERs. Other tools such as, RSS Feeds, help to link related contents, as well to disseminate and facilitate access to the OER. The OER release should also be available in CD/DVD formats, especially for users with Internet access limitations. The widely availability of OER allows other users to reuse and adapt it to their own context and needs.

Educators should gather feedback from learner's experience on the use of OER release in a learning situation, as well as the feedback provided by others users. Changes and improvements should be analyzed and reviewed to be added in the further release.

The AM-OER development cycle should be restarted whenever changes and adjustments are required to keep the OER up-to-date and improve its quality.

### **3. Applying AM-OER in the FLOSS domain: A Case Study**

Our case study has exploratory characteristics because we intended to analyze the applicability and usefulness of the AM-OER to effectively support the development of OERs. The case study is characterised as a holistic case study where the case under study is investigated as a whole. The "case" is the same as the unit of analysis, i.e., the AM-OER method.

#### **3.1. Planning**

In the scope of this study we investigated whether the AM-OER is applicable and useful for the development of OERs. We started from the proposition that the development of OERs can benefit from the usage of a process based on agile methods and that incorporates LD practices.

The study was applied in an academic environment in the Institute of Mathematics and Computer Sciences (ICMC) at University of São Paulo (USP). This study addressed a real problem within a context of Software Engineering. It consisted of the development of an introductory course on FLOSS.

Two participants (and the researcher responsible for the study) took part of the study. They have basic background on the development of learning materials and on the subject and topics related to development of this course.

The data were collected through documentation, annotation and artifacts generated during the study, followed by a brief interview with participants.

#### **3.2. General Overview of the OER on FLOSS**

The course on FLOSS was designed and created based on the book chapter: *Software Livre e Propriedade Intelectual: Aspectos Jurídicos, Licenças e Modelos de Negócio* [Kon et al. 2011].

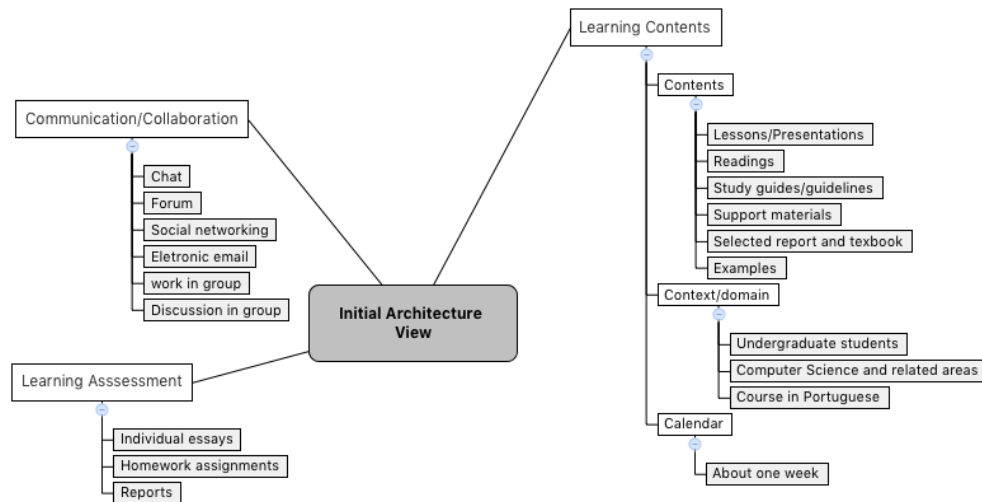
The researcher and the educator were responsible to identify and establish the general goals and learning objectives of the course, followed by the definition of its overall structure.

The general goal of the OER on FLOSS course is to introduce the FLOSS movement and clarify how specific characteristics of FLOSS may affect their use by individuals, companies, and governments [Kon et al. 2011]. At the end of the course, it is expected that the students will achieve the following learning objectives/outcomes:

- Understand the motivation behind the emergence of FLOSS movement.
- Have essential background on the concepts and definitions of FLOSS.
- Have the ability to select an appropriate license that fits their purpose, and to understand the implications and consequences of using it in a product.

- Be aware of the benefits related to FLOSS.
- Have background on the characteristics and development culture of FLOSS.
- Be able to select appropriate methodologies and tools for FLOSS development.

In Figure 2 we show the Initial Architecture View, summarizing the overall structure for the FLOSS course. It maps the main components of the course including the contents covered by the course, assessment criteria to assess learners, and the dialogic and collaboration activities during the learning intervention. We highlight that the Initial Architecture View is a specific outcome of the application of the AM-OER.



**Figure 2. OER on FLOSS: Overall structure.**

The FLOSS course was in Portuguese and the target audience was undergraduate students in Computer Science or related areas, such as Information Systems and Computer Engineering. The course lasts approximately one week. The students participate in activities to reinforce the theoretical concepts on FLOSS. No prerequisite is required.

The main contents of the FLOSS course include: presentations/lessons, readings, study guides/guidelines, supporting materials, selected report and textbook, and examples of methodologies, frameworks and supporting tools.

Students are evaluated during the course by individual essays, homework assignments and reports. They communicate and collaborate with colleagues and educator/tutor through chat, forum, social networking, electronic email, work and discussion in group.

The FLOSS course was divided into three modules:

1. Module I – FLOSS Movement: addresses general aspects about the problems and issues related to closed or proprietary software, and how FLOSS may become an alternative. It briefly discusses the emergence of the FLOSS movement, perspectives, motivations, implications and consequences of adopting FLOSS.
2. Module II – FLOSS licenses: addresses general aspects of licensing options for FLOSS. It emphasises the characteristics and types of FLOSS licenses. It also describes the main licenses for FLOSS.
3. Module III – Development and Business Model: addresses general aspects and peculiarities associated to the development and to the business models for FLOSS.

Both the educator and the student played the role of: (1) designer, who was responsible for the creation and refinement of the overall structure for the course; and (2) media creator, who was responsible for the creation of contents/media, and activities related to the course.

In Figure 3 we summarize the overview of the learning structure for the course, represented by the Learning Structure View, which is a specific outcome of the application of the AM-OER. It describes and maps, within the learning context, how learning activities are linked with the learning outcomes, contents, tools, and assessment. For instance, a student enrolled in the course can take as activity *summarize FLOSS characteristics*. At the end, a learning outcome could be that the learner has acquired critical understanding on the subject (*background on FLOSS development*) according to the assessment activity conducted.

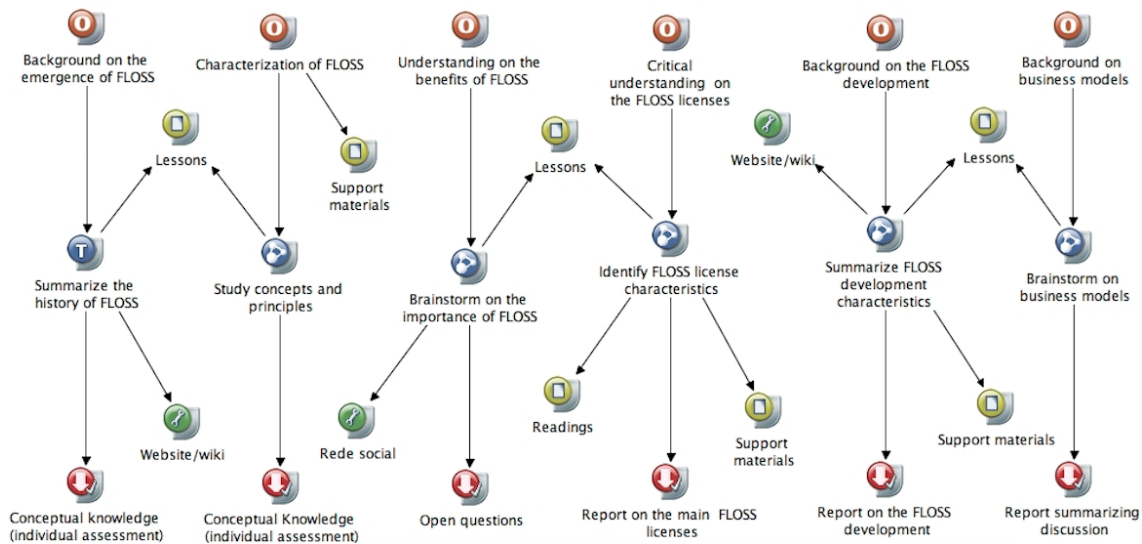


Figure 3. OER on FLOSS: Learning structure overview.

The learning structure was refined to visualize the flow of course activities and to determine the tasks that students need to perform to achieve the learning outcomes. In Figure 4 we show a sketch of the Learning Tasks View, representing a simple learning activity (*summarize FLOSS development characteristics*) from the overall Learning Structure View, broken down into simple and atomic tasks with contents, tools and assessments. For instance, a student enrolled in such activity needs to perform different tasks such as *identify development culture*, to identify the main characteristics related to the FLOSS development. The Learning Tasks View is a specific outcome of the application of the AM-OER.

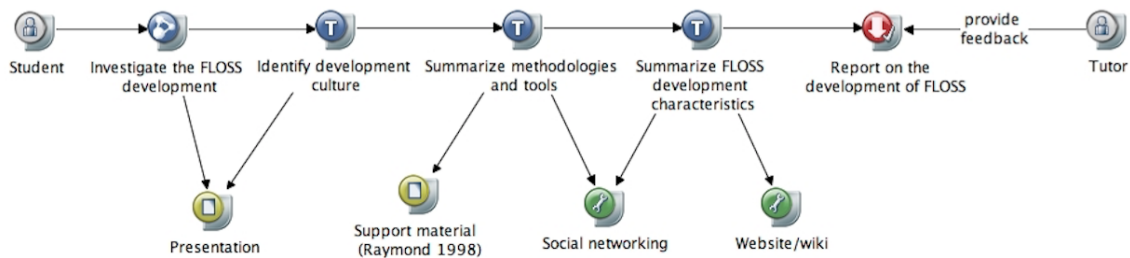


Figure 4. OER on FLOSS: Learning structure refinement.

Associated learning contents were created/adapted and integrated into the FLOSS course activities. They essentially include html pages, lecture notes, readings, text documents, open textbooks, images and videos. In Figure 5 we exemplify a topic of the Module I, briefly explaining essential concepts and definitions associated to FLOSS.

Throughout the development, the educator was responsible for evaluating the FLOSS course. He/she takes into consideration the criteria adopted by the AM-OER, including: (1) accuracy and relevance of the contents; (2) adequacy and alignment of learning objectives to the contents/media, tools, and assessment activities; (3) usability and readability of contents

and activities, among others. All suggestions for corrections, modifications and improvements were gathered and considered to improve the course.

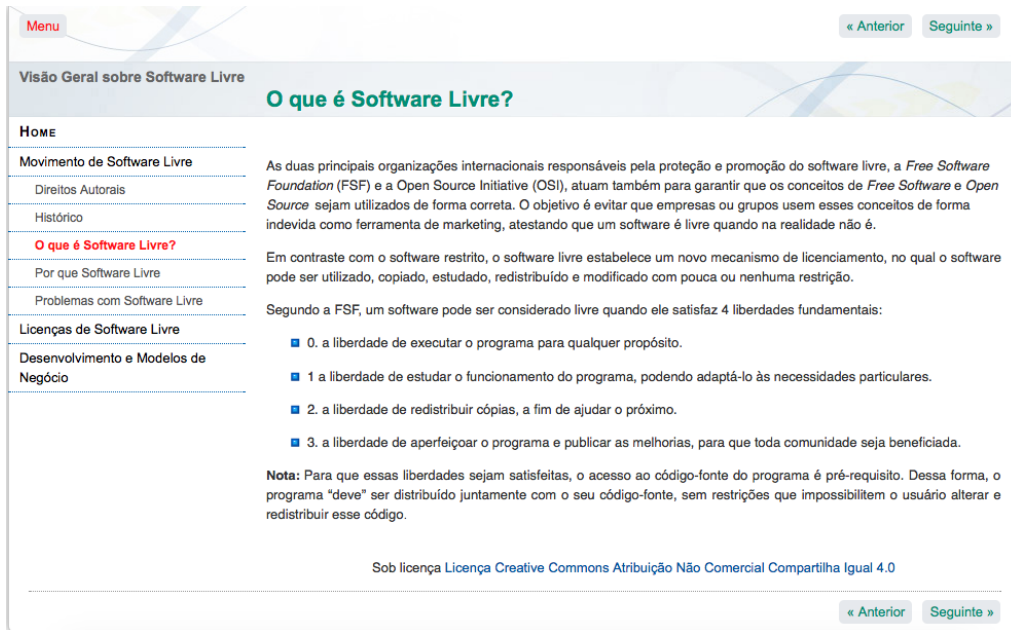


Figure 5. OER on FLOSS: Learning structure refinement (in Portuguese).

Asynchronous (electronic mail) and synchronous communications tools, such as instant messaging system and videoconference, were used to support the communication among participants throughout the course development.

The educator was responsible for licensing the course contents. The FLOSS course is under the Creative Commons License – Attribution Non-Commercial Share Alike 4.0 International<sup>1</sup>. It follows the same lines of the license used in the reused material that served as the basis for the development of this course.

### 3.3. Summary of the Results

The results show that AM-OER was useful to plan, design and build the course. According to the participants, the “method helped to shape the course as a whole and to determine the contents and activities that should be part of the course”. They pointed out that the activities of the method allow a focus on the learning process; besides thinking in the learning content itself, the educator also thinks about the activities that the students must perform to achieve the learning outcomes at the end of the course, helping the systematic structuring of learning.

Participants reported shortcomings and weaknesses of the AM-OER. According to them, some activities of the AM-OER were described superficially or were disconnected from each other. The main problems reported by the participants were related to: (1) identify and describe relevant metadata associated to the course contents; (2) search for related contents that could be reused to compose the OER; (3) properly evaluate and select the contents found; and (4) understand and establish the proper licensing policies for the course contents.

The results provided significant feedback for the development and improvement of the AM-OER method. All suggestions and changes identified in the case study were considered to improve it. For instance, participants had difficulty to seek for related contents to be reused to compose the course. We provided some guidelines on issues to be considered when searching for related contents and gathered useful search supporting mechanism.

<sup>1</sup>creativecommons.org/licenses/by-nc-sa/4.0



The main limitations of this case study were: (1) the number of participants in the study was small. New studies with a larger number of participants should be considered further; (2) the study was applied in an academic environment and in a specific domain of knowledge; (3) the study involved a single case study, then it can not be replicated by another researcher; and (4) the results of the case study do not allow generalization. These limitations will be considered in the development of new assessment studies, which are being planned to be carried out further.

Although preliminary, the results show that the proposed agile method with embedded learning design practices is feasible and provides support to the design and creation of OERs [Arimoto and Barbosa 2013, Arimoto et al. 2015].

It is important to highlight that we have also conducted an experiment to validate AM-OER, by applying it in the development of an OER on software testing. The experiment aimed at evaluating the effectiveness, efficiency and quality of the results produced by the usage of AM-OER [Arimoto et al. 2016]. The results showed evidences on the effectiveness and efficiency of AM-OER in the development of OERs.

#### **4. Conclusions and Future Work**

In this paper we presented the AM-OER for the design and creation of OERs, that is a combination of agile and LD practices. AM-OER provides a flexible approach for accommodating changes and improvements throughout the development of OERs. LD practices are embedded in the development of OERs, making the design more understandable, explicit and shareable, improving quality and facilitating further reuse and adaptation.

We have investigated the applicability of the AM-OER through an exploratory case study within the software engineering domain. It consisted of the development of an introductory course on FLOSS. The preliminary results indicate that AM-OER is applicable and useful in the design and creation of OERs. Furthermore, the experiment involving the development of an OER on software testing have shown the effectiveness and efficiency of AM-OER in the development of OERs.

Currently, we are working to plan and conduct new empirical studies with the application of AM-OER in the development of different types of OERs. The idea is to develop and provide OERs for the teaching and training in a variety of areas and domains of knowledge. AM-OER will be evolved according to the results obtained from the studies. We also intend to define a theoretical framework for the quality assurance of OERs. We expect the framework will serve as a basis for evaluation and quality assurance of any type of OERs.

#### **Acknowledgment**

We are grateful to the Brazilian funding agencies FAPESP (grants 2011/06204-1 and 2013/16759-6), CNPq and CAPES for their financial support.

#### **References**

- Ambler, S. W. (2001). Agile design practices. [agilemodeling.com/essays/agileDesign.htm](http://agilemodeling.com/essays/agileDesign.htm). [02/2016].
- Amiel, T., Orey, M., and West, R. (2011). Recursos educacionais abertos (REA): modelos para localização e adaptação. 12(1):112–125.
- Arimoto, M. M. and Barbosa, E. F. (2013). Towards the establishment of an agile method for OERs development and delivery. In *IEEE Frontiers in Education Conference (FIE)*, pages 541–547, Oklahoma.

- Arimoto, M. M., Barroca, L., and Barbosa, E. F. (2015). A learning design method for open educational resources. In *IEEE Frontiers in Education Conference (FIE)*, pages 1–9, El Paso, Texas.
- Arimoto, M. M., Barroca, L., and Barbosa, E. F. (2016). AM-OER: an agile method for the development of open educational resources. 15(2):1–26.
- Beck, K. and Andres, M. (2004). *Extreme Programming explained: embrace change*. Addison-Wesley.
- Braga, J. C., Pimentel, E., and Dotta, S. (2013). Metodologia INTERA para o Desenvolvimento de Objetos de Aprendizagem. In *XXIII Simpósio Brasileiro De Informática na Educação (SBIE)*, pages 306–315.
- Brasher, R., Cross, A., Galley, S., and Weller, M. (2012). Final project report of the OULDI-JISC project: challenge and change in curriculum design process. Technical report, The Open University: United Kingdom.
- Butcher, N. (2015). A basic guide to open educational resources (OER). A. Kanwar & S. Uvalic-Trumbic (Eds.). Vancouver and Paris: COL and UNESCO.
- Conole, G. (2013). *Designing for learning in an open world (explanation in the learning sciences, instructional systems and performance technologies Series)*. Springer.
- DeVries, I. (2013). Evaluating open educational resources: lessons learned. volume 83, pages 56–60. *Procedia - Social and Behavioral Sciences*.
- Kemczinski, A., Costa, I. A., Wehrmeister, M. A., Hounsell, M. S., and Vahldick, A. (2012). Metodologia para Construção de Objetos de Aprendizagem Interativos. In *XXIII Simpósio Brasileiro De Informática na Educação (SBIE)*, page 10.
- Kon, F., Lago, N., Meirelles, P., and Sabino, V. (2011). *Software livre e propriedade intelectual: aspectos jurídicos, licenças e modelos de negócio*, pages 59–107. XXX Jornada de Atualização em Informática (JAI'2011), Sociedade Brasileira de Computação.
- Lapolli, F., Motta, C. L. R., Cruz, C. M., and Oliveira, C. E. T. (2010). Modelo de Desenvolvimento de Objetos de Aprendizagem Baseado em Metodologias Ágeis e Scaffoldings. *Revista Brasileira de Informática na Educação*, 18(2):41–53.
- Leverly, L. (2012). Finding relevant OER in higher education: A personal account. *Glennie, J., Harley, K., Butcher, N. and Wyk, T. van (Eds), Open Educational Resources and Change in Higher Education: Reflections from practice*.
- Madden, T. (2010). Turning a resource into an open educational resource (OER). UK Physical Sciences Centre – Briefing Paper.
- Masson, P. and Udas, K. (2009). An agile approach to managing open educational resources. *On the horizon*, 17(3):256–266.
- Molenda, M. (2003). In Search of the Elusive ADDIE Model. *Performance Improvement*, 45(5):34–36.
- OPAL (2011). Beyond OER: shifting focus to open educational practices. OPAL Report 2011. Essen, Germany: Open Education Quality Initiative.
- Patricia, C. C. S., Rocio, R. M. G. d., and Elizabeth, R. P. A. (2010). OER's Production Cycle with Social Authorship and Semantic Tools. In *IEEE Education Engineering Conference, EDUCON 2010*, pages 121–128.
- Schuwert, R., Wilson, T., Valkenberg, W. V., and Lane, A. (2010). Production of OER: a quest for efficiency. In *7th Annual Open Education Conference*, pages 2–4, Barcelona, Spain.
- Schwaber, K. and Beedle, M. (2002). *Agile software development with Scrum*. Prentice Hall, 1 edition.