Innovative eLearning Technologies in the Open Education Era

Pavel Boytchev
boytchev@fmi.uni-sofia.bg
Faculty of Mathematics and Informatics - KIT,
Sofia University "St. Kliment Ohridski"
Sofia

Svetla Boytcheva
svetla.boytcheva@gmail.com
Institute of Information and Communication Technologies,
Bulgarian Academy of Sciences
Sofia

ABSTRACT

This paper discusses the role of open educational resources (OER) in Science, Technology, Engineering and Mathematics (STEM) learning. Usually the media used in OER are video lessons, presentations, some open textbooks, etc. But how does this paradigm apply to interactive learning environments? Can users have permissions to engage in the 5R activities (Retain, Reuse, Revise, Remix and Redistribute) both to the course content and to the platform for the course? The authors present a model for creation of educational virtual environments. The designed methodology is applied in development of various Computer Graphics-related university courses – Educational Languages and Environments, Geometry of Motion, and Fundamentals of Computer Graphics. There are presented three use cases for open educational virtual environments as illustration of the proposed methodology – Suica, Mecho and Meiro. For each of these environments are presented directions for 5R activities for both the course content and the educational virtual environment. The approbation of this innovative eLearning tools at bachelor's degree level shows improvement in students' comprehension of the course material and performance during the courses. The OER have a significant impact on the co-creative activities of the students.

CCS CONCEPTS

• **Computing methodologies → Computer graphics**; • **Applied computing → Interactive learning environments**; *E-learning*; Distance learning.

KEYWORDS

eLearning, Computer Graphics, Open Education, Interactive learning environment

ACM Reference Format:
Pavel Boytchev and Svetla Boytcheva. 2019. Innovative eLearning Technologies in the Open Education Era. In *Proceedings of the 20th International Conference on Computer Systems and Technologies (CompSysTech '19), June 21–22, 2019, Ruse, Bulgaria.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3345252.3345300

CompSysTech '19, June 21–22, 2019, Ruse, Bulgaria

ACM ISBN 978-1-4503-7149-0/19/06…$15.00
https://doi.org/10.1145/3345252.3345300

1 MOTIVATION

The "Open educational resources" (OER) term is officially introduced at UNESCO's 2002 Forum on Open Courseware [2]. There are several definitions of OER, but the last updated definition provided by The Hewlett Foundation[1] is:

> "Open Educational Resources are teaching, learning and research materials in any medium – digital or otherwise – that reside in the public domain or have been released under an open license that permits no-cost access, use, adaptation and redistribution by others with no or limited restrictions."

The term OER is closely related to the term "open content"[2] – copyrightable work, which is either in the public domain or is licensed in some of the Creative Commons licenses[3] CC-0, CC-BY, CC-BY-SA. This allows users to engage in the 5R activities:

- Retain - make and own copies of the content;
- Reuse - use the content in a wide range of different ways.
- Revise - modify the content, including translation into other languages;
- Remix - combine the content with other resources;
- Redistribute - share the content and its modifications with other users.

The initial version of 5Rs, so called 4Rs of openness (Reuse, Redistribute, Revise, Remix) is proposed by David Wiley in 2009 [4] and later revised by him.

There are different types of OER – that are created with primary educational purpose and openly available; that are created with another purpose, but are repurposed for educational needs; that are adopted from other OER using translation and localization of some existing educational resources. Additionally, there are four major groups of OER - open courses, open textbooks, open courseware and open course frameworks.

The common media used in OER are video lessons, presentations, open textbooks, etc. The localization to the domestic language for such OER is mainly done manually.

The emerging technologies and innovative learning ICT-based technologies used in STEM learning cause the need of more advanced OER that converge both the learning environment and the course content.

Usually a software is not considered as an open content, because it is described as an open source under GNU General Public license[4] and Free and open-source software (FOSS)[5]. But in eLearning both

[1] https://hewlett.org/strategy/open-educational-resources/
[2] http://opencontent.org/definition/
[3] https://creativecommons.org/
[4] https://www.gnu.org/licenses/gpl-3.0.html
[5] http://freeopensourcesoftware.org/index.php/Main_Page

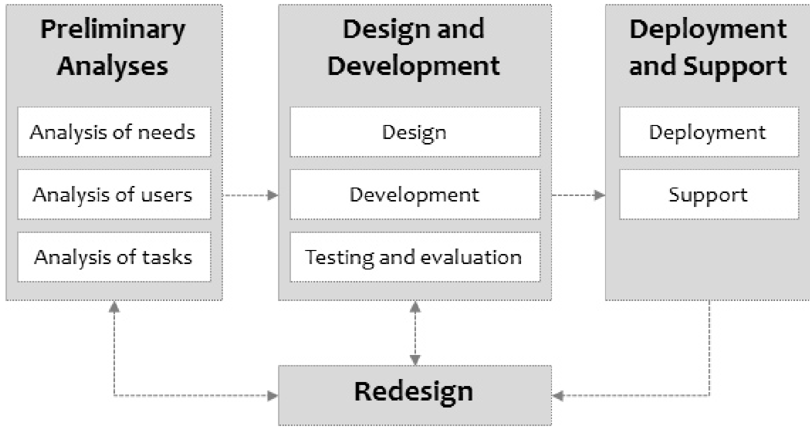

Figure 1: A model for creation of educational software

the educational virtual environment and the course content can be considered as OER.

Providing an open access to the course content and the educational platform allows educators more freedom and flexibility in both course material adoption and different methodologies implementation in the eLearning platform.

One more benefit needs to be mentioned for Computer Science students – they can be in two different roles – using the course content as educational source and exploring the eLearning virtual environment as an open source software, that they can study, adapt and modify.

The main objective of this study is to present the main steps in the design and the development of educational virtual environments in the context of OER.

This paper is structured as follows: Section 1 presents the motivation and objectives of the research; Section 2 briefly overviews some open educational resources; Section 3 describes the main steps in design and development of educational virtual environments; Section 4 overviews three use cases for open educational virtual environments – Suica, Mecho and Meiro; and Section 5 concludes and briefly sketches directions for further work.

2 RELATED WORK

One of the pioneers among the leading universities that started providing its courses as OER is the Massachusetts Institute of Technology (MIT) – MIT OpenCourseWare initiative[6]. Nowadays the trend for openness emerged and the majority of leading universities provide OER. Members of the Open Education Consortium (OEC) [7] are 243 institutions from 44 countries worldwide and they provide courses and OER in 29 languages.

Currently there are many portals that provide opportunities to create, explore and collaborate with educators around the world in order to improve the curriculum. One of these portals is Create OER with Open Author[8] that provides options for building resources, lessons and modules, both for secondary school and higher education. The resources are organized in collections that can be searched and reused. OER Commons contains 400+ Full university courses, 400+ Interactive Mini-Lessons and Simulations, 400+ Adaptations of existing open work, 160 Open textbooks, and 2,400+ K-12 Lesson Plans, Worksheets, and Activities.

The term "open pedagogy" is proposed by Weller [9] in 2013 to emphasize the role of OER not only as a source in education, but also as a networking factor and a main force in co-creative activities. As a further development of OER initiative in 2018 Wiley and Hilton III [10] propose a new term "OER-enabled pedagogy", which makes a connection between 5Rs in OER and Papert's ideas [8] for notion of constructionism in education.

There are many OER, but their adequate application in the education context is still a challenging task. Andy Lane and Patrick McAndrew discuss the quality of OER [5] and the problems that arise by the sparsity of different OER used in a single course, rather than one integrated course material.

Lane [5] discusses two aspects of OER – properties of OER and properties of people, who use these resources. Lane presents a comparison of education resources used in traditional universities and open education universities. OER allows more flexibility and modularity in the open university courses and the list of course prerequisites is also reduced, because for the majority of modules no prior qualification is needed to register for the courses.

In his OER Handbook [3] Gurrell presents a guideline for the "OER Lifecycle:" Find, Compose, Adapt, Use and Share. Gurrell also discusses different levels of openness of the OER. The problem is caused by combination of OER with different licenses and by the fact that resources with not explicitly stated license are considered as copyrighted.

3 DESIGN AND DEVELOPMENT METHODOLOGY

The development of educational virtual environments requires specific phases and steps due to their characteristics. Being software, the creation of such environment passes through phases of design, development, testing and deployment. Its application in educational

[6]https://ocw.mit.edu/index.htm
[7]https://www.oeconsortium.org/
[8]https://www.oercommons.org/

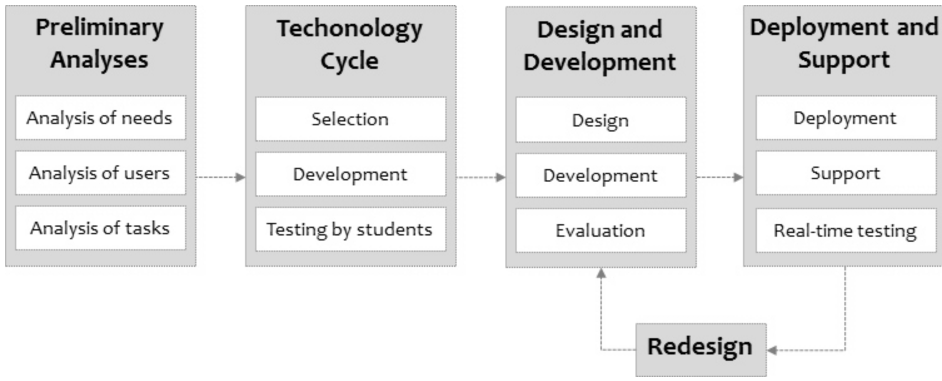

Figure 2: A modified model for creation of educational virtual environments

context requires additional phases. A theoretical model of development of educational software for education of computer scientists is developed by Mihnev and Zafirova-Malcheva [7]. A summary of the model is shown in Figure 1.

Each of the three phases "Preliminary Analysis", "Design and Development" and "Deployment and Support" consists of steps with various elements (not included in the figure). For example, the analysis of tasks in the first phase contains the following elements:

- General educational goals
- Analysis of information processes
- Preliminary conditions
- Specific educational goals
- Software requirements
- Educational content
- Learning activities

This model provides a flexible framework for adaptation in different practical situations. The creation of the software virtual environments described in this paper uses a modification of this model – Figure 2. The first change of the model is the inclusion of a technological phase. The virtual environments use hardware accelerated 3D graphics in a browser. The educational openness of the software requires that it is accessible via a broad spectrum of devices, both desktop and mobile. If the used technology is too novel, it might be not well supported on common devices, and this will render the virtual environments useless to many students.

It is expected that the educational activities use a blended learning approach by being available both for classrooms-based teaching and for online distant teaching. This expectation brings additional requirements for the technology as the blended learning sharpens the differences between personal and remote interaction between students and teachers [6].

These considerations are the ground for adding a standalone technological phase, which includes:

- Selection of technologies that support the requirements, defined in the analyses phase.
- Development of test application, which uses the basic functionality of the selected technologies.
- Testing the application by students with their own devices.

The result of this phase is a definite answer whether the selected technologies are available and affordable to students. Some of the students try the application on several personal devices. If most of the students report that the used technology is working well on their devices, then it makes sense to go to the design and development of the actual virtual environments.

Another change of the model is the testing of the virtual environments. The alpha testing and preliminary evaluation are performed in the design and development phase, while the beta testing is moved to the deployment phase. This change is imposed by the timeframe of the academic calendar and it is the main reason to conduct beta testing in real-time. This provides an additional advantage of using the software in real situation. The feedback received by students is much more detailed and accurate.

4 EDUCATIONAL VIRTUAL ENVIRONMENTS

4.1 Suica and Educational Languages and Environments

The Faculty of Mathematics and Informatics at Sofia University runs the programme "Effectiveness of Students Education", which goals include the design and development of new educational materials, software and methodologies. In 2014 the programme supported a project for recreating the discipline "Educational Languages and Environments" (ELE) by creating and using the virtual environment, called Suica.

The ELE discipline has a predominant educational direction. Its main goal is to develop practical skill in future teachers to create interactive graphical online educational applications, used to support their teaching activities. The applications can be run on different platforms – smartphones, tablets, laptops and desktops.

Suica is a set of teaching materials and a dedicated JavaScript library that wraps WebGL functionalities, defines various graphical assets, manages their properties and controls a programmable browser-based mobile-friendly 3D animation. The software is compatible with learning management systems like Moodle. Learning modules based on Suica can be included in Moodle courses as external modules.

The main purpose of Suica is to provide programming tools for creation of educational animations via simplified programming interface. A typical Suica program is much shorter and easier to comprehend than a corresponding program in WebGL or Three.js.

The educational content of ELE is split in five modules. The first one introduces the base technologies HTML5, CSS, JavaScript and Document Object Model. The second module presents Suica and its graphical objects. There are numerous live examples and students are encouraged to explore them including exploration via code modifications. The third module focuses on implementing programmable animation and mouse-based interactivity. The fourth module is about building user interfaces and mobile applications. The last module shows the process of development of three educational applications, their internal code and reasoning as well as pedagogical impact and educational applicability.

Figure 3 represents snapshots from these applications – Mathematics (exploring right triangle), Physics (exploring ballistics) and Biology (exploring DNA and amino acids).

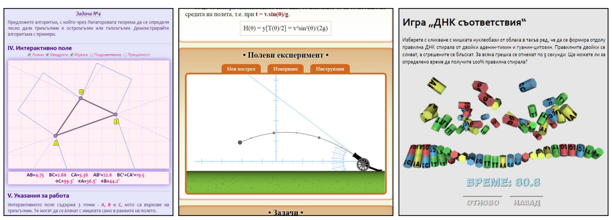

Figure 3: Suica projects in Mathematics (left), Physics (middle) and Biology (right)

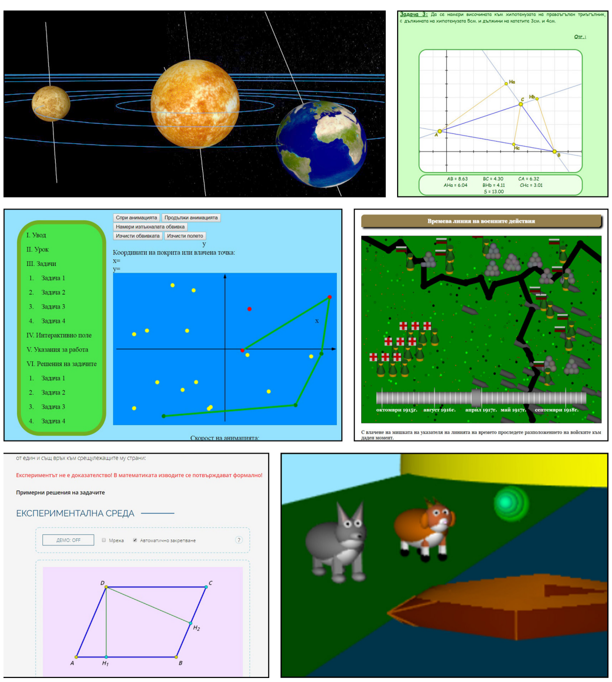

Figure 4: Students' project in Suica

Currently the ELE discipline is taught as several courses – as mandatory undergraduate course for pedagogical specialties, as an undergraduate course for distant learning, as elective course for other non-pedagogical specialties and as graduate course for pedagogical and eLearning master's programmes.

The students' evaluation is based on four practical tests, where they solve programming tasks by developing small applications, and a final project, which must be a complete mobile educational application with interactive graphics. Figure 4 shows snapshots of six students' project in different domains – Astronomy, Geometry, Recreational Mathematics, Computer Graphics and History.

4.2 Mecho and Geometry of Motion

The programming virtual environment Mecho (MECHanical Objects) is developed specifically for the undergraduate course Geometry of Motion. Initially Mecho is written in Elica, in 2012 it is rewritten in C++ and OpenGL, and in 2015 it is rewritten in JavaScript and WebGL.

Mecho represents virtual mechanical structures as hierarchy, some examples are shown in Figure 5. The first level is for individual elements like rods, beams, gears. This is the basic building blocks, that students modify and arrange in more complex structures.

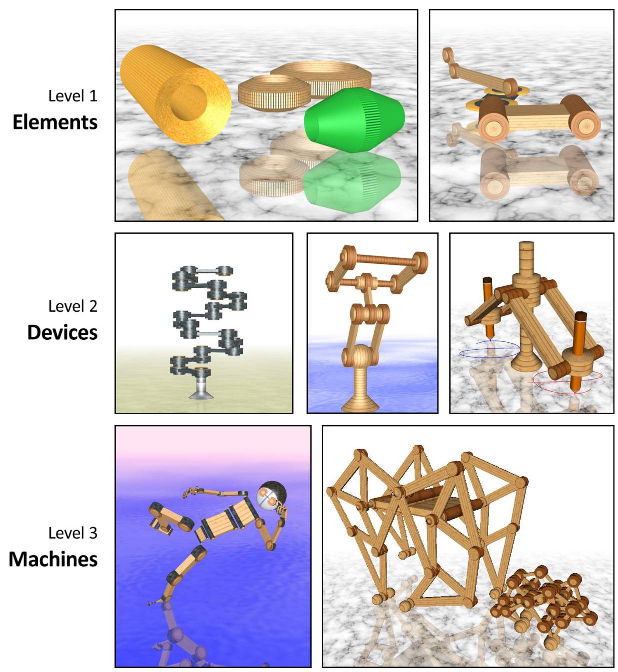

Figure 5: Hierarchy of virtual elements, devices and machines in Mecho

The second level is for simple devices composed of first-level elements. The construction of such devices require knowledge of Computer Science (Programming), Mathematics (Geometry) and Physics (Mechanics).

The highest level in the hierarchy of constructible models is for advanced machines, which may include both first-level elements and simple devices. Such virtual machines can represent complex motion, relation or linkage, like a human skeleton, that can be animated programmatically; or a model of a walking kinetic sculpture initially designed by the Dutch artist Theodorus Jansen.

One of the most notable features of Mecho is the possibility to model different approaches to a problem. For example, the making of a cycloidal motion along a circle can be implemented as:

- a virtual device constructing the cycloid – Figure 6;
- an abstract motion-only device with no invisible elements;
- as a trajectory, defined mathematically by the equation:

$$(x, y, z) = \left(8\cos\left(t + \frac{\sin(8t)}{8}\right), 8\sin\left(t + \frac{\sin(8t)}{8}\right), 1 + \cos(8t)\right)$$

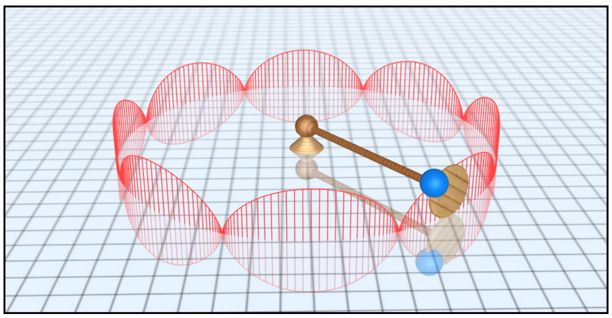

Figure 6: Virtual device for circular cycloid

The educational virtual environment Mecho is used in the undergraduate elective course "Geometry of Motion". The software is used to help the student experiencing creativity by implementing their mathematical and programming skills.

4.3 Meiro and Fundamentals of Computer Graphics

A project started in 2013 lead to the creation of a software library for constructing interactive models. They were used in the undergraduate compulsory course "Fundamentals of Computer Graphics" (FCG). Hundreds models were created and incorporated in the course materials; snapshots of 280 of them are included in Figure 7.

These models are built in C++ using OpenGL as a graphical engine. Although this provides performance, there are issues making such models inconvenient to use. Namely, it is difficult to make them online or to have them running on mobile devices. They needed compilation into executable files before running, while distribution of compiled code is considered risky. The distribution of sources leads to problems for students with less popular operating systems or hardware platforms.

To resolve these problems all models are recreated in a newly developed virtual environment called Meiro. The name comes from the Japanese word for maze. The purpose of Meiro is to serve as a platform for all interactive activities in FCG – demonstrations during lectures, tasks from exercises and online assessments.

Meiro is a virtual maze, created dynamically by following some guiding parameters for its size, density and complexity. Topologically the generated maze is a 3D building with floors and stairs between them. The distribution of stairs, walls and passages are

Figure 7: Interactive models in Fundamentals of Computer Graphics

customizable. An isolated snapshot of navigationally complex floor from the 3D maze is presented in Figure 8.

The virtual models that existed as C++ software (see Figure 7) are rewritten in JavaScript with Three.js and Meiro. These models are used in several ways in the course: as standalone demonstrations models; as gamified models; and as external learning models.

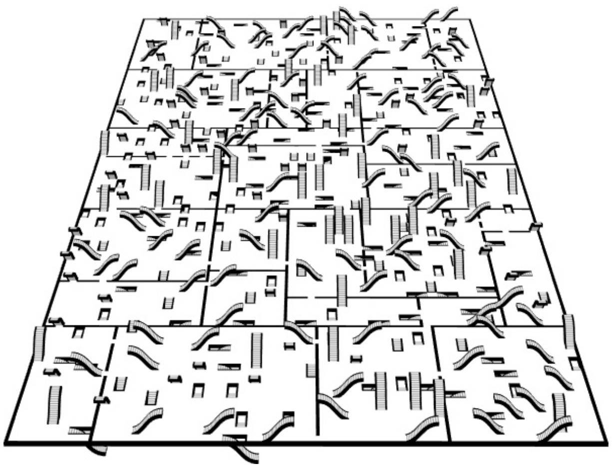

Figure 8: A single floor with upstairs and downstairs

4.3.1 Demonstration models. The several hundred Meiro-based models are used as demonstrational educational content. The models are integrated within the lecture notes and can be run directly from the presentations. Their interactivity is useful for lecture-time explorations, especially when students are curious of what would happen if some condition is changed.

Initially, the visual content in the lecture notes was only as static images. Few years later the notes were enriched by hundreds of short animations (usually 30 seconds each). This smooths the learning curve for the students. The final step was to introduce interactive models, which replace most of the static images and the video clips in the lecture notes.

Snapshots of Meiro models used as standalone demonstration are shown in Figure 9 – the left four snapshots. They are from the topics of orthogonal shadows of complex objects, constructing scenes of custom 3D primitives, central projection implemented via matrices, and flexible skins via a surface subdivision method.

4.3.2 Embedded gamified models. The second way of using Meiro models is as assets in the Meiro environment. After a maze is generated as a 3D scene, it is possible to populate the halls with preselected models. They keep all their initial functionality, but are now incorporated in a larger virtual world. When a Meiro maze is being generated, it is a "nude" structure of floors, walls, doors and stairs. Then the maze configurator allocates the 3D models to the largest halls in the maze – Figure 9 (right four snapshots). When Meiro models are used in a Meiro maze students consider this as a game – they can navigate in the maze, looking for models. When they find one, they can interactively play with it.

4.3.3 External learning models. The Meiro-based models can be used as learning educational content. As long as all models are JavaScript programs in HTML5 pages, they are distributed as source code and can be executed directly in a browser. Thus, when students access the lecture notes, they also access the code of the interactive models.

The students are encouraged to study the code by modifying it, to test different hypotheses, to investigate how the topics from the lectures are implemented in software, and so on.

The graphical engine used for Meiro and for all models is Three.js. The exercises are also in Three.js, so studying the models helps the exercises and vice versa.

Meiro, as a software environment is not explicitly studied in the course, because it is tuned for the purpose of FCG, rather than to be a general 3D programming tool. This is the reason for the exercises to use custom non-Meiro models. The left half of figure 10 shows some of these models, which students are asked to build by themselves in the computer lab.

One of the evaluation components in FCG is a project. Currently Meiro is not used for all programming activities in the course, so students' projects may or may not use Meiro as a virtual platform. The right half of figure 10 shows snapshots from students' projects submitted during the latest course, which was conducted in January 2019. They all represent some scenes (indoor and outdoor) with animated characters.

The code of these characters is reused from the code of Meiro. The original Meiro character can be see in the top left snapshot in figure 9 – it is a blue humanoid shape. The characters in the students' project change the colours, add clothing and other attributes, some projects even modify the structure of the character.

Using gamified models leads to the next logical step – Meiro is currently being improved to cover the evaluation and assessment of students. The next version of Meiro models will be enriched with evaluation and assessment models. In this way the full educational cycle from lectures, through exercises up to evaluation can be performed within a single environment. The basic features of such environment are:

- Time – No restrictions, Meiro could be played at any time and any number of times.
- Location – No restrictions, Meiro could be played at any location, provided there is internet connection.
- Hardware – Meiro could be played on any hardware utilizing the acceleration of a graphical processor (all modern smartphones and computers provide such support).
- Software – Meiro could be played on any operating system with browser supporting WebGL and HTML5 (all modern browsers provide such support).

Using Meiro and Meiro models for students' assessment and evaluation requires some change in the current implementation of Meiro. One possible architecture is shown in Figure 11.

The system will reuse the students' profiles in Moodle and will send evaluation scores to the corresponding table. Inside Meiro there will be new modules – for automatic generation of 3D interactive models for evaluation, for personalization and configuration, and for collecting results (educational, performance, etc).

5 CONCLUSION AND FURTHER WORK

The presented methodology is used in the implementation of three Computer Graphics courses that have different objectives and target audiences. The approbation of this innovative eLearning tools at bachelor's degree level shows improvement in students' comprehension of the course material and performance during the courses.

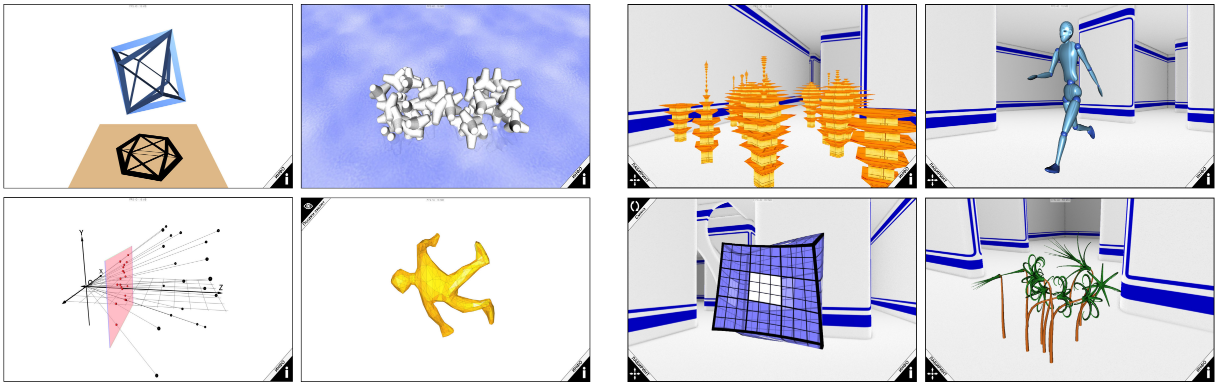

Figure 9: Meiro models as standalone demonstrations (left) and gamified in-maze models (right)

Figure 10: Models built in lab classes (left) and as projects (right)

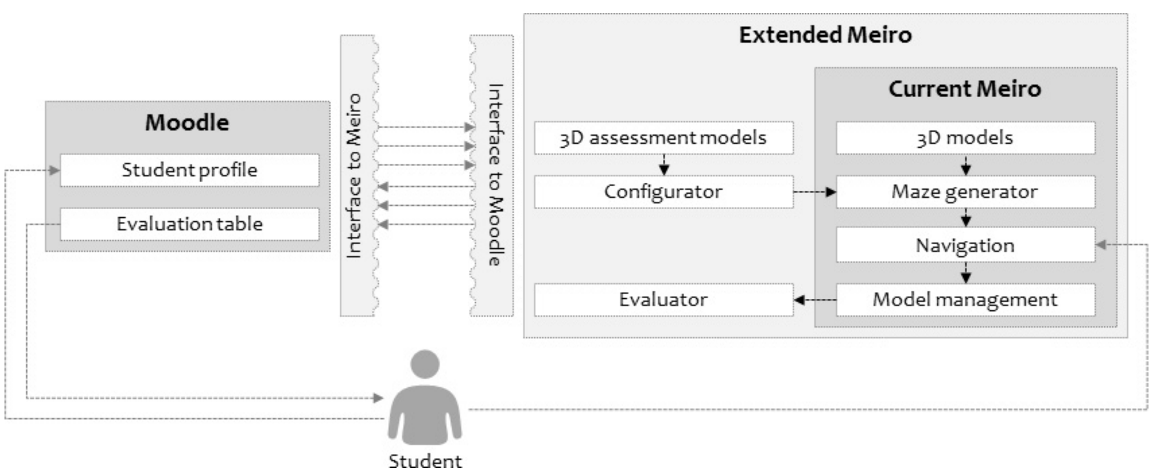

Figure 11: Extending Meiro for assessment and evaluation

The results, achieved by students, as well as the methodology of their evaluation and assessment are discussed in details in [1].

The OER have a significant impact in co-creative activities of the students. OER gives opportunity to students to access several innovative courses from various educational institutions. But sometimes it is difficult to find the right course for the desired audience. Thus, rich metadata needs to be added as a supplementary description to the course content, following some common standards, like Dublin Core Standard[9] [10], or some specific standards related to education. This will allow presented educational environment to be organized in course collections.

ACKNOWLEDGMENTS

The research is partially supported by Sofia University "St. Kliment Ohridski" Research Science Fund project N80-10-75/12.04.2019 "Use of high performance computing technological tools for competence development for applying the inquiry-based approach in STEM education at secondary school" and by the National Scientific Program "Information and Communication Technologies in Science, Education and Security" (ICTinSES) financed by the Ministry of Education and Science.

REFERENCES

[1] Pavel Boytchev and Svetla Boytcheva. 2018. Evaluation and assessment in TEL courses. In *AIP Conference Proceedings*, Vol. 2048. AIP Publishing, 020035.

[2] Mei-Hung Chiu. 2016. *Science Education Research and Practice in Asia: Challenges and Opportunities*. Springer.

[3] S Gurrell. 2008. *Open Educational Resources Handbook for Educators Version 1.0*. The Center for Open and Sustainable Learning. 284 pages.

[4] John Hilton III and David A Wiley. 2009. The creation and use of open educational resources in Christian higher education. *Christian Higher Education* 9, 1 (2009), 49–59.

[5] Andy Lane and Patrick McAndrew. 2010. Are open educational resources systematic or systemic change agents for teaching practice? *British Journal of Educational Technology* 41, 6 (2010), 952–962.

[6] Pencho Mihnev and Temenuzhka Zafirova-Malcheva. [n. d.]. E-learning activities in a Blended Learning mode: experience in teaching Computer Science students. *iCERi2016 Proceedings*, 7126–7134.

[7] Pencho Mihnev and Temenuzhka Zafirova-Malcheva. [n. d.]. A general model for educational software design and development. 4079–4089.

[8] Seymour Papert and Idit Harel. 1991. Situating constructionism. *Constructionism* 36, 2 (1991), 1–11.

[9] Martin Weller. 2013. The Battle for Open–A Perspective. *Journal of Interactive Media in Education* (2013).

[10] David Wiley and John Levi Hilton III. 2018. Defining OER-enabled pedagogy. *The International Review of Research in Open and Distributed Learning* 19, 4 (2018).

[9] http://dublincore.org/documents/dcmi-terms/

[10] http://www.dublincore.org/documents/dcmi-type-vocabulary/