Collaborative Learning Tools For Formal And Informal Engineering Education

Bogdan Paul Fazakas, Ovidiu Claudiu Iuonas, Cosmin Porumb, Bogdan Iancu
Faculty of Automation and Computer Science, Computer Science Department
Technical University of Cluj-Napoca
Cluj-Napoca, Romania
Bogdan.Fazakas@student.utcluj.ro, Ovidiu.Iuonas@student.utcluj.ro, Cosmin@hpm.ro, Bogdan.Iancu@cs.utcluj.ro

Abstract - **Microservices deployment of Rich Internet Applications is not a new research topic; many articles have been published and related innovative technologies released in the last years. Thus, the research work presented in this article is focused on surveying the current collaborative learning concepts and technologies and proposing a new virtual classware approach, implemented and deployed as a set of independent components responsible for specific collaborative learning services. While engineering education methodologies are complex, flexibility or openness should be considered as high priorities. Thus, an actual implementation, according to blended training scenarios, has been presented.**

Keywords — WebRTC; Kurento; e-Learning; microservices; content creation; computer supported collaborative learning; Real-time communications;

I. INTRODUCTION

In recent years, with the advances of the Internet and e-learning technologies, a blended mode of teaching and learning, which effectively combines the traditional face-to-face learning and e-learning, has evolved. The related methodology is not widely adopted in higher and postgraduate engineering education due to vary reasons. One of them is regarding the courses design under the blended learning environment. Moreover, a big number of teachers do not consider the e-learning methodologies as stable and functional enough for engineering, especially for hand-on laboratory and project task completion. [1]

According to Hopartean [7], in the classic blended learning model, the teachers assign tasks, conduct regular lectures, or train students' skills while the students attend the online autonomous learning act and cooperative learning sessions, or accomplish teachers' assignments. Furthermore, the teachers make assessments over students' learning effect and solve their problems. So, teachers set objectives and tasks of different levels, they put forward requirements and suggestions according to the teaching contents and make assessment to students' learning effects through task-based activities. They also answer students' questions and offer essential teaching to major and difficult situations. In addition, different multimedia elements are used for supplementing the educational contents. Nevertheless, the teachers create flexible and diversified theoretical and practical scenarios and teaching contents, using authentic materials to let students come upon more technical information related to real problems/projects. The students work out their own learning plans, determining learning methods autonomously. They conduct on-line autonomous learning when going through each course unit and complete the related quiz while the teachers encourage students to cooperate with each other for finishing simple learning tasks or complex group-based projects. Through cooperative learning, students do not only acquire knowledge, their team spirit and coordination will also be fostered, skills in dealing with people will be improved and abilities to express themselves will be enhanced [1].

The current trends in engineering education market are related to effective virtual classroom solutions able to cover synchronous and asynchronous teaching and learning needs. They should be easy to customise and deploy according to different scenarios while each university or college has the own set of rules and constraints. This paper proposes a new virtual classware approach, implemented and deployed as a set of independent components responsible for specific collaborative learning services. Due to a set of non-functional requirements, the virtual classroom system has been designed for complying with the Microservices architecture (MSA).

The article is structured as follows: Section 2 presents a list of relevant collaborative learning tools for knowledge building, underlining their main pros and cons. Section 3 illustrates a novel virtual classware system that implements a different way of blended learning, and enables the instructors and learners to effectively collaborate in two different ways: synchronous or asynchronous. Section 4 presents a series of experimental results, while Section 5 concludes the paper.

II. Related Work

A lot of research has been done in learning area. Ding and Cao [6] presented a RECT (Remote Collaboration Tutor) learning model and its implementation in the Master Program of Software Engineering. This RECT model prototype is based on the method of adaptive learning, as well as face-to-face communication. Nevertheless, a set of asynchronous collaborative tools has been designed, which adopt the cloud platform capabilities in order to promote sharing of curriculum-related resources with the remote collaborative system aiming to enhance the interaction between distant team members.

Ouya et all [8] designed a platform proposition as a support for universities' educational system in a limited Internet connection context. Students follow lectures live via video conferencing but they attend laboratory sessions taught by on-site faculty staff. The proposed system offers a web interface that allows authenticated users to access different services (Management of users, classes and courses, Courses planning, real-time File Transfer, etc.), according to their predefined profile. Their conclusion was that a signaling server is needed to establish real time connections. The server is considered the basis of their system. A feature appreciated by teachers who use their system was screen sharing, because it allows them to show their students real practical work.

Chandran et all [4] research has been focused on defining an actual e-learning architecture model starting from issues in current e-learning applications, as well as presenting the Hybrid Instructional Model as the blend of the traditional classroom and online education and its customization for e-learning applications running on the cloud computing infrastructure. This paper highlights the e-learning (non)functional requirements such as openness, scalability, and development/customization costs, while it identified the limitations of existing e-learning systems hard to dynamically scale and extend with new features. In addition, the integration with existing e-learning systems or modules could be expensive.

Moldovan et all [2] describe the eRAF framework and the innovative multimedia annotation methods it implements, based on advanced human-computer interaction and synchronous and asynchronous interpersonal collaboration concepts. This research has been continued in Hopartean et all [7], which aims at improving the instructors' and learners' experience by customizing eRAF tools for education and training sector.

A blended learning environment, that implements the face-to-face teaching and e-learning capabilities in Advanced Software Engineering, was proposed by Qiu et all [5]. A set of integrated projects was selected as stimulus to learning. Both inter and intra-group collaborative learning are encouraged. The survey conducted at the end of the course showed that students accept problem-based learning very well, and their academic achievements were also better than expected. The methodology consisted of grouping students in teams, dividing the semester in project phases and developing the project using iterations.

Other commercial tools offered by WizIQ, Blackboard or Dokeos have been considered. Most of them are related to the sync or async learning activities even if the development of skills and competences is a very complex and laborious process. These state-of-the-art concepts have been taken into account by the authors when designing an innovative computer supported collaborative learning framework and implemented a virtual classroom software system that provides with blended learning capabilities.

III. Technical Solution

Collaborative learning concepts are widely adopted in informal education programmes, especially career development. The current work started from this aspect and highlights the importance of adapting such collaborative tools for formal engineering education requirements. This way, two types of interactive learning sub-systems - synchronous and asynchronous collaborative ones - have been designed. They consist of independent modules and components able to interoperate each other according to the concept of microservices architecture (MSA).

The block diagram illustrated in Figure 1 presents the microservices-based system architecture and its main modules, which implement both synchronous and asynchronous collaborative learning approaches. MSA deployment points out a response-actor programming style while today's end-users expect dynamic yet consistent experiences across a wide range of devices. This constrained the authors to design and implement a scalable, adaptable, modular, and easy to adapt virtual classroom.

From the technical point of view the system component are implemented using the following programing languages and technologies: JavaScript and related frameworks for both client (Angular 1) and server-side (NodeJS), HTML5, CSS and SASS extension for CSS 3, Socket.io, PHP and CodeIgniter framework. The preferred protocol used for data communication is HTTP(S) while the multimedia data encoding/decoding and compression standards are VP8 (video) and OPUS (audio). In addition, the educational resources created by trainers when using asynchronous collaborative instruments comply with SCORM standards

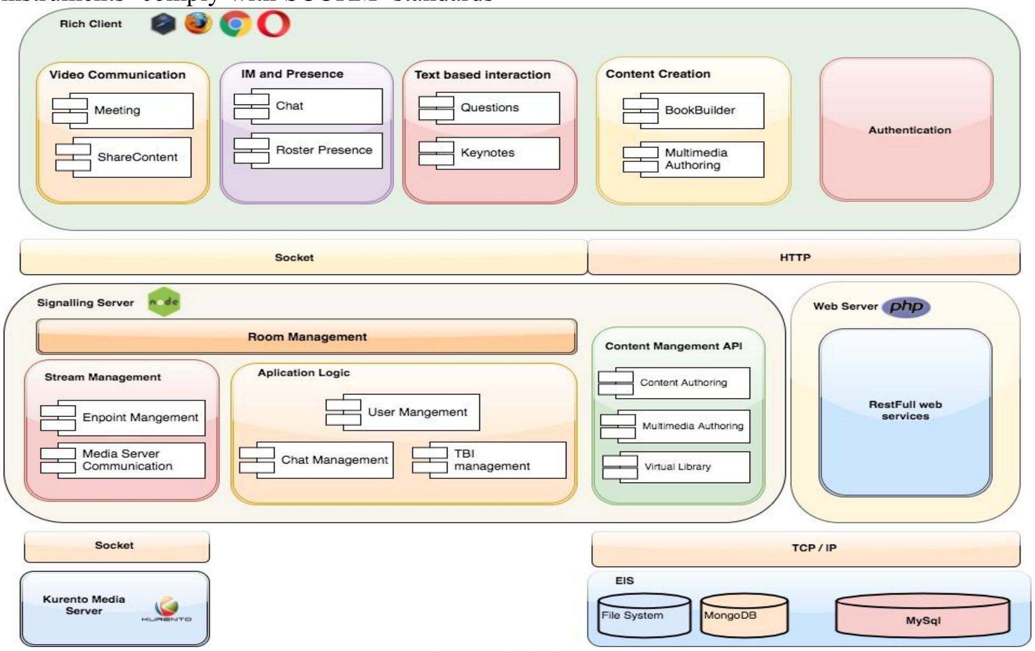

Fig. 1 – Block diagram

and aggregate different types of multimedia resources.

As follows, both the synchronous and asynchronous collaborative learning solutions are described, together with their interconnection and interaction.

A. *Synchronous Collaborative Learning*

The following modules compose the synchronous collaborative learning sub-system: *Video Communication*, *IM and presence*, and *Text-based user interaction*. They provide with advanced services accessible in a dedicated rich user interface, while rich client components typically communicate to a NodeJs-based signalling server, via websocket. Same communication concept is used for implementing the message exchange mechanism between signalling server and Kurento media server.

Video Communication module accomplishes real-time audio and video lecturing and personalised content sharing, including applications and multimedia resources. The dedicated rich user interface extends open source WebRTC capabilities and invokes Kurento media library (kurento-utils lib) that transparently facilitates the WebRTC video collaboration mechanism. NodeJS signaling server manages the virtual channels between the rich client and Kurento media server by handling the user's endpoint management and enabling the *many to many video communication* approach. In order to create the media pipeline (virtual room and streams) the signaling server should extend the open source client API offered by Kurento Media Server (kurento-client lib). Kurento Media Server (KMS) is a WebRTC-compliant media server that allows transcoding, recording, broadcasting and routing of audio-visual data packages.

The current research contains networking-based improvements in terms video collaboration mechanism supported by WebRTC client components and KMS instances. A special case has been identified when deploying the collaborative learning applications over academic LAN. In this context, two new microservices (STUN and TURN instances) have been used in order to support NAT transversal mechanism.

IM and presence microservice is responsible for handling the course attendance and virtual interaction between teachers and students. In such a case, teachers are considered as moderators within the course virtual space while students as attendees in the live course roster. The teacher has full control of the sync session by conducting the flow of lecturing or activating students interested in sharing ideas, answering teacher's questions or asking the own ones. Each student has the possibility to 'raise hand' when trying to lively interact with the virtual class. Moreover, the sync interaction is also supported by instant messaging component.

Text-based user interaction module supports the async interaction between teachers/trainers and students. It is typically used during the lectures while the audience can ask questions without interrupting the speakers or when moderators can send keynotes without disturbing the lecture flow. This interaction is bidirectional, meaning that both teacher and student can initiate the async flow. Students can use the question component for asking different questions, while teachers has the possibility to accept/decline the question(s) and to change its status (waiting for answer, ongoing, answered, declined). The keynote component gives the teachers the opportunity to send hints, links or messages to the attendance. Technically speaking, the *text-based user interaction* and *IM and presence* module is based on socket.io APIs while the communication protocol is HTTP(S).

B. Asynchronous Collaborative Learning

According to the trainers' and teachers' requirements, an innovative set of authoring tools have been designed and implemented in terms of content creation. Such instruments allow the instructors being teachers or trainers to develop interactive content and deliver it in a compact manner to students enrolled in the educational programme. Analogue, each student has the possibility to annotate the content as public or private contribution. Private means not visible in study group or class while public contribution should be visible and annotatable by anybody enrolled in the same course. Nevertheless, private contributions can be shared among the synchronous sessions when the student becomes active and can share whole resources stored into the own section of the virtual library.

The content creation and delivery modules compose the asynchronous collaborative learning sub-system. Such modules provide with a dedicated rich user interface that allows the training staff to easily import basic materials, convert them in special content, annotate this content with different multimedia elements and aggregate the resources according to SCORM standards.

The following modules compose the asynchronous collaborative learning sub-system: *BookBuilder*, and *MultimediaAuthoring*. They provide with advanced services accessible in a dedicated rich user interface, while rich client components asynchronously invoke the virtual library microservices, via HTTP. These microservices mainly hosted by NodeJS deal with uploading basic materials such as MS Office or video formats, as well as PDF files and converting them to internal resources easier to handle, then aggregating such resources into a SCORM pack to be delivered in a compact format. In addition, BookBuilder enables the trainers/teachers to annotate the basic materials imported as internal resources with images, video sequences or HTML pages. The annotation elements are automatically transformed into book resources to be aggregated into the final form at will.

The MultimediaAuthoring components allow the teachers/trainers to create and publish value-added educational resources being able to handle different kind of multimedia elements, including video sequences imported from the local hard drive or a public repository, as well as lively recorded by end-user. Such recording functionality is accomplished by the rich user interface based on WebRTC capabilities that facilitates both audio/video and desktop recording.

C. Basic functionality

Common services module implements the basic functionality of the collaborative learning system. Its main components - authentication, room management, EIS access mechanism - are exposed for both types collaborative learning tools.

Authentication component is responsible for subscription management, login/logout, role assignment or profile handling. A successful login guarantees the temporary presence subscription with a specific role received. Room management component enables the students to switch between ongoing courses, workshops, laboratories, or projects occurring at a moment. Another component task is to ensure that the content shared among a room/course is available only for its roster with temporary presence subscription for that specific room/course.

Nevertheless, whole content shared among online courses such as questions, keynotes, course recordings, video content, presentations or documents, should be stored into the enterprise information system. It should be easily retrieved when needed and aggregated as public content to be delivered to students in a compact format.

IV. EXPERIMENTAL RESULTS

This section presents the experimental results achieved during the implementation of the proposed collaborative learning tools within a pilot project delivered for a software development company based in Cluj-Napoca, Romania. It was used for supporting the internship and training programmes whereas such activities should involve personnel outside the business entity. Both internship and training programmes are focused on a special topic and include the documentation created and delivered by asynchronous collaborative tools (authoring instruments), lecturing activities supported by sync collaborative learning components and tuition/coaching stuff delivered by the blended learning instruments.

In order to increase the training services, both parties decided to organize different courses available for beneficiary employees as well as its national and international partners at will. This way, the training resources and services should be delivered in the beneficiary's network and the affiliates' ones. This required a virtual classroom component to be implemented and integrated within a blended learning environment. Its main objective was to support live lecturing while more than 50% of the face-to-face sessions have been remotely conducted.

In addition, the trainees employed by beneficiary's partner's always attend the lectures from their own offices, thus a special form of blended learning solution is a must.

Figure 2 illustrates the lecturing act conducted by the trainer that already involved a student for sharing ideas. The rich user interface provides audiovisual capabilities (main video slot, as well as the ones in the bottom), session control (left-hand side buttons) and roster management (right hand side).

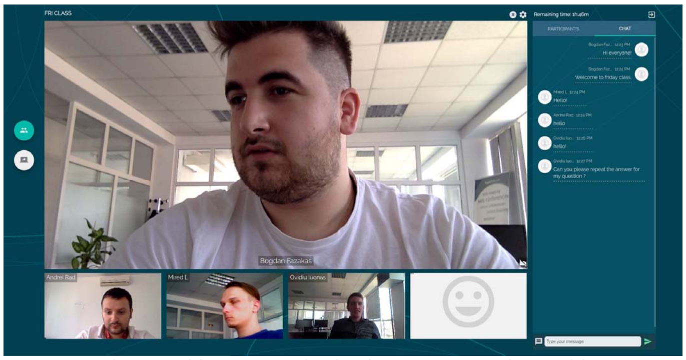

Fig. 2 – Video communication instruments. Trainer's side

Content sharing tools illustrated in Figure 3 (instructor actually sharing his own IDE), enable the participants to proactively co-operate during the educational/training act by facilitating teamwork competences and developing communication and group skills. Technically speaking, once the trainer starts sharing his desktop, application or even any type of content, the student is notified about the sharing event. Thus, the students can automatically visualize and listen the instructor's explanations regarding the current content.

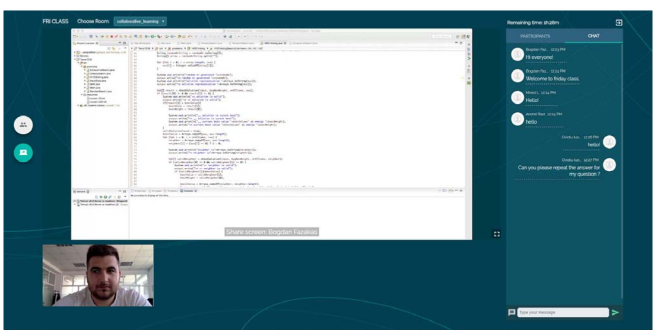

Fig. 3 – Content sharing tools. Trainer's section

Real-time interaction and delivery increases the efficiency of the training process and usually enables the trainees to clarify all the aspects before completing the lectures. The audience can ask questions during the training act without interrupting the trainer's lecture, as presented in Figure 4. The students use the question component for asking different questions, while teachers have the possibility to accept/decline the question(s) and to change the status. As already mentioned, the text-based interaction is bidirectional, meaning that both teacher and student can initiate an asynchronous communication flow.

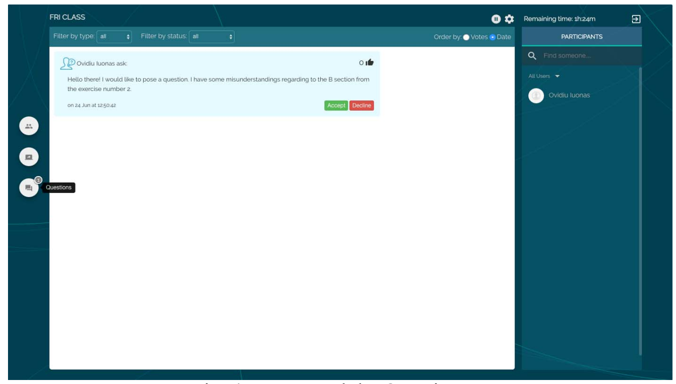

Fig. 4 – TBI module. Questions component

The beneficiary also asked instructors for developing interactive training resources (video tutorials, complex assessment content, serious games) and deliver such content in a compact manner to students enrolled in the training programme. Figure 5 illustrate the rich user interface of BookBuilder and how it can be used when creating value-added educational content.

Fig. 5 – Content creation tools. Trainer's side

While the content is stored into the virtual library, it is automatically aggregated then delivered to the trainees enrolled in the training programme. The students can visualize the material, each view being logged in the database, so, the study progress can be monitored and measured by trainers and beneficiary. Figure 6 illustrates the rich user interface on student's end during the individual study process.

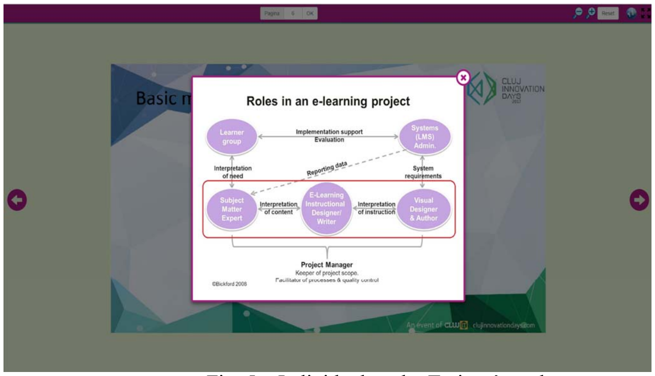

Fig. 5 – Individual study. Trainee's end

Each student has the possibility to annotate the content as public or private contribution. Private means not visible in the study group or class while public contribution should be visible and annotatable by anybody enrolled in the same course. Nevertheless, private contributions can be shared among the synchronous sessions when the student becomes active and can share whole resources stored into the own section of the virtual library.

The *video communication module* was also successfully tested at "Cluj Innovation Days", an event of ClujIT hold in Cluj-Napoca, Romania, on 30-31 March 2017.

V. CONCLUSIONS

The paper presents a novel virtual classware technology consisting of independent open-source components, each one being implemented as a microservice responsible for specific collaborative learning capabilities. While engineering education methodologies are complex and flexibility or openness should be considered as high priorities, a real implementation according to blended training scenarios has been illustrated.

As already mentioned in this article, the research objectives were to make the training and teaching activities much simpler and more effective, especially when creating and delivering value-added educational content and services. Analogue, vary types of learners, which use a dedicated rich user interface that supports a high degree of interactivity, should assimilate such content and services.

These goals have been achieved due to the of microservices architecture, which enables the implementation of a set of loosely-coupled software components that provide with synchronous (*Video Communication*, *IM and presence*, and *Text-based user interaction*) and asynchronous (*BookBuilder*, and *MultimediaAuthoring*) collaborative learning functionalities.

In the next phases of the development process, the focus will be on extending the video communication and content sharing tools for supporting the integration of special equipment within video collaboration sessions by capturing the data retrieved from its output, processing such data and, rendering it in a dedicated video component. Its main scope is to expand the range of training services in other domains of interests.

ACKNOWLEDGMENT

This work was supported by HyperMedia Company, Cluj-Napoca, Romania, under an internal research grant. The results presented in this paper were obtained with the support of the Technical University of Cluj-Napoca through the research Contract no. 1996/12.07.2017, Internal Competition CICDI-2017.

REFERENCES

[1] Porumb C., Porumb S., Orza B., Aurel V., Blended Learning Concept and its Applications to Engineering Education, Advanced Engineering Forum Vols. 8-9, 2013

[2] Moldovan, R., Orza, B., Vlaicu, A. and Porumb, C., Advanced human-computer interaction in external resource annotation, In Automation, Quality and Testing, Robotics, 2014 IEEE International Conference on (pp. 1-6) , IEEE

[3] Xiao L., Wang Z., *Cloud Computing: a New Business Paradigm for E-learning*. Third International Conference on Measuring Technology and Mechatronics Automation ,2011.

[4] D. Chandran and S. Kempegowda, Hybrid e-learning platform based on cloud architecture model: a proposal, Proceedings of the International Conference on Signal and Image Processing,2010 534-537.

[5] M. Qiu, and L. Chen, A Problem-based Learning Approach to Teaching an Advanced Software Engineering Course, Proceedings of the 2nd International Workshop on Education Technology and Computer Science, pp 252-255,2011

[6] Ding Q., Cao S., RECT: A Cloud-Based Learning Tool for Graduate Software Engineering Practice Courses With Remote Tutor Support, IEEE Access, March 2017

[7] Hopartean, R., Orza, B., Porumb, C., Mihon, D., Meza, S., External Resource Annotation Framework and its applications in E-learning, Interaction Design and Architecture(s) Journal - IxD&A, N.31, 2016, pp. 86-97

[8] S Ouya, G Mendy, C Seyed, AB Mbacke,I Niang WebRTC platform proposition as a support to the educational system of universities in a limited Internet connection context, 14-16 Dec. 2015

[9] Y Liao, Z Wang, Y Luo, The Design and Implementation of a WebRTC based Online Video Teaching System, Oct 3-5,2016

[10] B Garcia, LL Fernandez, M Gallego, F Gortázar, Analysis of video quality and end-to-end latency in WebRTC,Dec 4-8,2016

[11] LL Fernández,, MP Diaz, RB Mejías, Kurento: a media server technology for convergent WWW/mobile real-time multimedia communications supporting WebRTC.June 4-7,2013

[12] CC Spoiala, A Calinciuc, CO Turcu, C Filote, Performance comparison of a WebRTC server on Docker versus Virtual Machine, 13th International Conference on DEVELOPMENT AND APPLICATION SYSTEMS, Suceava, Romania, May 19-21, 2016

[13] A. Bouabdallah, S. El Jaouhari Certified multimedia statement with WebRTC and Microservices, The 2016 Asia Pacific Conference on Multimedia and Broadcasting (APMediaCast),Nov 17-19,2016