International Conference
FROM DELC TO VELSPACE
Plovdiv, 26–28 March 2014

DISTRIBUTED CLOUD BASED SYSTEM FOR ACTIVE/E-LEARNING

Stoyan Cheresharov, Hristo Krushkov

Abstract. *This paper describes a study of building a cloud based distributed E-Learning system for remote active learning. We use for education the same tools and methods, used for software development and project management. The students have to be put in the real life environment right from the beginning of their education. The teacher takes the role of a project manager, technical leader and trainer. As a result of this approach the students receive a lot of practical knowledge, learn to make decisions, take responsibilities, communicate remotely and participate in building real life software projects.*

Keywords: Distributed system, E-Learning, Cloud, Remote Learning, Active Learning

1. INTRODUCTION

The Information Technologies are a moving target. They literally change every day. The Web Technologies are one very good example. HTML, CSS, and Java Script standards, frameworks and ways of doing things change very fast. We have to deal with huge amount of information. How to transfer knowledge which becomes obsolete before we can even absorb it? We have to use the cutting edge technologies and methods in teaching. Learning by doing and authoring, other words Active Learning [1], becomes a necessity. There is clear evidence for the benefits of active learning [3]. The described techniques and tools are augmenting the concepts of the Active Learning by adding Technology Enhancements and approaches used in the Project Management. We will start by making an observation and a claim.

An observation can be made, that the knowledge in IT is just way too much for a mere mortal. It is enough, for example, just to look the number of the CSS properties and their possible values. And this is only one of the technologies necessary to learn in order to get the job done. The solution to this problem is instead of learning we have to learn how to learn. There are two possible directions we can go:

- improving the capacity of the memory and abilities to learn fast. We have to learn techniques for increasing the memory capacity, fast reading, creating

visual representations of the knowledge for easier understanding and remembering. There are such techniques [6]-[8]. But they are not scientifically proven. The human brain has limits. Our assumption is that we can improve the capacity and the speed of the brain only to a certain degree. We still don't know how to unleash the full potential of the human brain.

- engaging the technologies in the learning process. Technology Enhanced Learning [2]. The technologies can help to avoid the limits of the human brain to store and process information.

A claim can be made that, the software development, education and learning cannot be separated.

If it is not possible for an ordinary human brain to remember so much information the software developers are put in the situation to learn and recall what they know all the time. Software development process is in fact an endless learning process. The new technologies emerge on the horizon every day and the accumulated knowledge becomes obsolete very rapidly. That means the professional software developers and the students face the same problems of learning things as fast as possible. If the professional software developer and a beginner are in an equal situation when they face a new technology, what gives the advantage of the experienced developer? The secret is in the tools and approaches of disposal. The experienced developer already knows how to learn, where to find the information and he/she has a lot of tools. The good developer is not the one that knows everything, but the one who knows where to find everything he/she needs. There are no novice and experienced when it comes to technologies. We are all novice. All we have to do is just instead of teaching the students concrete technology, to teach them how to learn, introduce communication and other tools for software development. But if the experienced developer and a students are equal that means the students can produce the very same results as the developers. Other words they can learn by doing and they can produce high quality software as well. All they need is just a proper guidance. So the process of education in fact can be seen as a software development process. A student can start producing quality software on a very early stage of his education. As you can see later in this paper this has been proven. The role of the teacher in the IT converts to a role of a Project Manager, technical leader and trainer. In fact the problems we face in the education are the same like the problems we have in the software development process. This is the reason we cannot separate the education, learning and software development. The Software Development Process includes technical training for constant improvement of the knowledge of the staff. The quality of a product depends on the quality of the developers.

2. USING THE PROJECT MANAGEMENT TOOLS AND PRACTICES IN THE EDUCATION

As we see from the introduction, the software development, education and learning cannot be separated and they face the same common problems. In this case we can

combine and exchange the tools and approaches from one area to another. We can apply the Project Management tools and approaches in the education. At the same time we can use and transfer educational tools and practices in the software development process. The conclusion that can be made from the points above is that we have to use Active Technology Enhanced Learning [2]. This combination introduces the best from the educational and technology worlds. The students have to be put in a suitable environment as early as possible. Actual coding in the software development practice is very little. With other words knowing a concrete technology takes very little place. It is a proven fact that switching from one technology to another can be done overnight. Instead of teaching concrete technologies the teacher should introduce common concepts. And use the concrete technology only as an illustration. Concepts like MVC, Design Patterns, ORM etc. can be told first [5].

The tools used by the software teams are actually fun and easy to learn. They increase the productivity and help to avoid the limitations of a human brain. The students nowadays don't need to be told what a social network is and how to use it. It is natural to them. This is the reason we can jump into this opportunity and take advantage of it. Using the modern cloud technologies a distributed platform for remote education and project management can be build. We have done it already in this study. The project management tools are remote and distributed by nature. They have been built to be used by geographically distributed software teams in different time zones. In fact the distributed remote learning has been invented already by the software engineers. All developers are working in such environment. In fact building software is very difficult to impossible without them. The tools described in this document cannot be separated from the software development process. Very little time is spent actually writing code. Most of the time is spent learning, communicating and improving the existing code. So the accent should be on the tools for communication, learning and collaboration. The tools change the paradigm of creating software. Every developer chooses the environment and the time to work. This leads to a productivity increase. Students put in such environment can choose the best time and environment to learn and work. The Project Management usually requires a distributed communication system. In fact not even all Software Development Teams are using such environment. Below in the document we are describing tools that can be used to build a Distributed Cloud Based Active/E-Learning System. It can be used in education or Project Management.

3. THE TOOLS FOR BUILDING DISTRIBUTED CLOUD BASED SYSTEM FOR ACTIVE/E-LEARNING

The concrete technologies and tools described below are used only as an illustration and to give a real life example. They can be replaced with similar

technologies and tools which offer the same functionality. What is important is to build similar network of services.

1) Video broadcasting and recording.
   Video tutorials (screencasts) are very important part of such distributed E-Learning system. The ability of the students to participate in a conversation increases the value of such media. The communication can be done verbally or by instant messages. The virtual studio or class room should allow the students to attend the course at the time of actual broadcast or watch it later at their convenience.
   In our setup we were using Google+ Hangout on Air. It allows up to 9 attendees. Questions can be asked verbally or in a form of a message during the session. The technology is web based. Other words all you need are just a web browser and Internet in order to use it. There are many more such services. Some of them are web/cloud based, others require installation of a native application. Such system allows the teacher to communicate ideas very easy. Everyone of the students can watch it in a convenient time. This increases the level of acceptance of the knowledge. The videos can have searchable transcriptions. We were using https://www.youtube.com/coolcsn
2) Blog posts allow the students to read instead of just watch and follow the instructions. They can be used for quick reference. They are searchable. It is easier to find a paragraph in a written document instead of a long video. The code snippets should be easier to find copy and paste. The code snippets itself should be organized in a different service like “pastebin” or “gist” for example. Our blog posts can be found at http://coolcsn.blogspot.com/
3) Mind Maps are modern way to represent the information. It is based on the Tony’s Busan “Radial thinking” [8]. It is referred by the author as the 21-th century education. The human brain remembers with pictures instead of abstract terms. In the process of remembering we have to use all our senses. The process should be turn into a pleasant joyful experience. This goal can be achieved only by using pictures as much as possible. This is what Mind Maps are encouraging. There are software products and cloud based services for building, collaborating and sharing such Mind Maps. We were using two Cloud Based Services for the purpose of this study http://mind42.com/ and http://meindmeister.com/.
4) The key point in the system/platform plays the Source Control Management System or also known as Version Control System. Every student has his own repository. It can turn later in a precious resource for his development activities. The student can use it as a portfolio on what he/she has done. There should be one central repository where the work of all individuals should be collected. This is the repository of the team. The students must be involved in the process and send as PR (Pull Requests) everything they have done during the exercises. Everybody can use it. Since these repositories are usually public the students become a source of spreading knowledge and solutions. There are private and public repositories. We were using as a

public repository https://github.com/wingman007/fmi. An example of a public repository for collaboration, where the students were building an Open Source Software project is https://github.com/coolcsn. For private repositories we have used services such as https://bitbucket.org/.

5) WiKi systems are another important channel for collaboration and communication. Each of the previously mentioned SCM systems in point 4, has an optional WiKi. The WiKi can be used for project management and sharing ideas. The students can join forces and work on different ideas together. They turn to be the creators not only of the content. They become the generators of the ideas.
6) Issue tracking systems are used in the software development teams for reporting bugs, requesting features, discussing and solving all kind of issues etc. Issue trackers are a perfect project management tool. These tools can be used to assign tasks to the student and track the progress. Each of the mentioned in point 4 SCM systems offers an optional Issue Tracker.
7) Very often we need to share code snippets. Having all code snippets well organized in a separate location allows for reuse and easy search. There are syntax highlighting features included in the mentioned services. In our study we were using code snippet systems for sharing code http://pastebin.com/ and https://gist.github.com/.
8) Instant messaging and voice communication plays a crucial role in any distributed remote E-Learning system. Communication is the key to any activity. The ability of the students to communicate not only with the teacher, but between each other opens the door for collaboration. The knowledge comes not only from one source, but each node can emit knowledge. In our study we were using mainly three instant messaging systems:
   8.1) Skype – voice communication. Allows conference calls.
   8.2) Google+ – It is perfectly integrated with other Google services.
   8.3) IRC channel #coolcsn on http://freenode.net/. IRC doesn't allow voice communication.

   There are many other technologies which can be used to replace the mentioned here, e.g. gotomeeting, CISCO etc.
9) Discussion thread, forum/maillist are a perfect way for asynchronous communication and sharing knowledge. The advantage is that the conversation continues in time. Other people can review, follow the thread and find answers. Each student can look for an answer, ask a question or give an answer. We were using Google Group (forum/maillist) system https://groups.google.com. Other hosted web or cloud services are available as well, for example http://www.nabble.com/.
10) A project management system usually tries to unite some of the services described already in the previous points. Some systems add automation to the day to day tasks. But what such system offers is an issue/bug tracker and WiKi. These are the most important features. So if we don't want to use the

build in issue/bug tracking and WiKi of a given SCM we can replace them with a project management system. There are very complicated and comprehensive Project Management systems. We were using http://www.hostedredmine.com/ for assigning tasks and following the progress.

11) Google Drive https://drive.google.com/ opens a whole new world in the way we communicate and collaborate. It is not anymore necessary to send documents by e-mail. It is enough to share a document with all students or ask them to share what they have created. With Google Drive feedback and survey forms have been created and distributed. Excel sheets with lists of the students and tasks have been shared. There should be a way to work together and share different kind of documents not only HTML documents as it is in the WiKi. Google Drive offers collaboration on Word Documents, Excel Sheets, Powerpoint Presentations, Collecting feedback, quizzes, voting systems etc.
12) To glue together all resources we can use a Learning Management System (LMS) or systems similar to Distributed eLearning Center (DeLC) http://delc.fmi.uni-plovdiv.net/. Such systems can be used as a central point of gravitation for all other services. The LMS Moodle http://moodle.com/ offers thread discussion in a form of a forum. And many other activities related to Active E-Learning. Courses can be created in such systems allowing feedback, control and grade books.
13) The Cloud based IDEs allow for collaboration and shorten the time from starting a project to a deployment. They reshape the way we are developing software. We were using Codenvy and Cloud 9 as Cloud Based IDEs for this study.
14) Different Cloud based services can be used if necessary such as IaaS, PaaS and SaaS. We were using "AppFog" as a deployment server to display the final result of the joint effort of all students.

Having an e-mail address is not even discussed above. The assumption is that every participant in such system has an e-mail address.

As a bare minimum we have to have Source Control Management System, Issue Tracker, WiKI and instant messaging communication or at least e-mail communication.

The system doesn't have a central point. So if one of the services fails others still can continue working. A role of a central point can play one of the services, where links to other services can be created.

4. THE RESULTS

Such platform as described above has been used for a group of 36 students to learn a concrete technology from scratch for 10 weeks. We were working as a team every week for 3.5 astronomy hours. We were using Source Control Management Systems, Video Tutorials and the communication tools. As a result an Open Source

Software was born. The participation of each student is clear and easy to be seen. The level of participation can be used as assessment. Everyone of the students not only learns, but builds his own repository of reusable code that can be used by others as well.

The same was done with a bigger group of 60 students under the same conditions. They not only have been able to continue what was done from the previous group but added an extra value.

The same experiment was conducted with a group of 15 students spending 240 hours during 3 months starting from scratch and working on an Open Source Software. The students have been able to build professional modules of reusable software used by the community.

5. THE CONCLUSION

A Distributed Cloud Based E-Learning system can be build very easy. It can be used for Active Learning. It helps to overcome the limitations of the human brain to learn and process the information. The teacher becomes a Project Manager and Technical Trainer.

If we don't separate the process of education and real work the effort materializes in something useful and good that can change the world and make it a better place.

REFERENCES

[1] D'Silva, I., Active learning, *Journal of Education Administration and Policy Studies*, Vol. 2, (6), 2010, 77–82.

[2] Agre, G. and D. Dochev, An Approach to Technology Enhanced Learning by Application of Semantic Web Services, *Bulgarian Academy of Sciences, Cybernetics and Information Technologies*, Vol. 8, No. 3, 2008.

[3] Woldeamanuel, M., H. Atagana and T. Engida, Ethiopian university science and technology instructors' attitudes toward active learning, *International Journal of Science and Technology Educational Research*, Vol. 4, (3), 2013, 47–56.

[4] Coldwell, J., A. Craig and A. Goold, Using eTechnologies for active learning, School of Information Technology, *Technical Reports Computing Series*, 2011.

[5] Hristov, H., Difficulties and Solutions when Changing the Paradigm. Teaching of Object-Oriented Analysis, Design and Programming, *Information Technology Conference "Interaction theory – practice. Key problems Solutions"*, 2011.

[6] Buzan, T., The Speed Reading Book, Harlow, BBC, 2010.

[7] Buzan, T., The Memory Book, Harlow, BBC, 2010.

[8] Buzan, T., The Mind Map Book, Harlow, BBC, 2010.

Stoyan Nikolov Cheresharov
Hristo Dimitrov Krushkov
Plovdiv University "Paisii Hilendarski",
Faculty of Mathematics and Informatics,
Department of Software Engineering,
236 Bulgaria Blvd ., 4003 Plovdiv , Bulgaria
stoyancheresharov@gmail.com, hdk@uni-plovdiv.bg

РАЗПРЕДЕЛЕНА „ОБЛАЧНО" БАЗИРАНА СИСТЕМА ЗА АКТИВНО ЕЛЕКТРОННО ОБУЧЕНИЕ

Стоян Черешаров, Христо Крушков

Резюме. *В тази статия ние представяме разпределена „облачно" базирана система за електронно, дистанционно и активно обучение. Използваме в образованието същите инструменти и методи, които се използват за развитие и управление на софтуерни проекти. Студентите трябва да бъдат поставени в реална работна атмосфера още в началото тяхното обучение. Преподавателят играе ролята на ръководител на проект, технически треньор и лидер. Като резултат от този подход студентите придобиват много практически знания, учат се да взимат решения, носят отговорности, комуникират дистанционно и участват в създаването на реални софтуерни проекти.*