Learning Environment for Problem-based Learning in Teaching Software Components and Service-oriented Architecture

Muthu Ramachandran and Rezan Sedeeq
School of Computing and Creative Technologies, and Engineering, Faculty of Arts, Environment and Technology, Leeds Beckett University, Headingly Campus, Leeds, U.K.

Keywords: Problem based Learning (PBL), Neural Pathway based Learning (NPL), Learning Environment for CBSE (LECBSE), Learning Environment for SOA (LESOA), SOA, Cloud Computing, e-Learning.

Abstract: CBSE (Component Based Software Engineering) has been with us nearly two decades since the popularity of object-oriented programming in some form or other. The concept of component abstraction is to increase the abstraction level to provide large chunk of software solution (building blocks), therefore, it results in increased productivity. Sadly, CBSE has been taught in the way we have been teaching programming with smaller abstraction known as functions and procedures. Similarly, OO programming languages have also been taught in a similar manner. Therefore, we have adopted a combination of Problem-Based Learning (PBL) and Neural Pathway Based Learning (NPL) techniques to teach high level CBSE modelling and concepts, large-scale design, and design for reuse with UML components. One of our main objectives of this paper is to develop a learning technique based on PBL and NPL. The research methodology is based delivery of the developed learning content into existing VLE and to study how well the learning has improved on these two modules delivered on the MSc Software Engineering course. The evaluation by students shows high satisfaction for this semester batch of post-graduate students. This paper presents the learning environment for PBL based learning on our VLE which provides an excellent learning experience for students to learn advanced topics such as software components and service computing.

1 INTRODUCTION

CBSE (Component Based Software Engineering) has been with us nearly two decades since the popularity of object-oriented programming in some form or other. The concept of component abstraction is to increase the abstraction level to provide large chunk of software solution (building blocks), therefore, it results in increased productivity. Sadly, CBSE has been taught in the way we have been teaching programming with smaller abstraction known as functions and procedures. Similarly, OO programming languages have also been taught in a similar manner. In addition, CBSE has also been proposed as a container of classes in the UML paradigm (a de-facto standard): composing objects/packages into components in the deployment view. Components do support OO design concepts; however, our ultimate aim is to decompose requirements into components: no doubt we will use OO models as a kind of one of modelling view to capture requirements. In addition, there is a shortage of skilled software engineers who are trained in component-based software development, secure software development, cloud service development (service computing) and domain specific languages. Therefore, we have developed a taught MSc Course in Software Engineering focussing on these skills and delivering individual modules.

To teach these subjects at the Post Graduate Level, we needed a unique pedagogical research, one of which is problem-based learning and the other is neural pathway learning. The definition of a Problem-Based learning (PBL) is learning by solving a large, real-world problem (Barg, M et al. 2000). Fee and Holland-Minkley (2010) states that Problem-Based Learning (PBL) is a pedagogy that centres student learning around open-ended, student driven problems facilitated by an instructor in order to achieve the learning outcomes of a course. Our definition of PBL has been tailored for computing courses and defined as “Mapping real-world conceptual models (a set of software solution to a real-world problem) directly to high level component abstraction”. PBL is suitable for all levels and courses as it helps to develop

Ramachandran, M. and Sedeeq, R.
Learning Environment for Problem-based Learning in Teaching Software Components and Service-oriented Architecture.
DOI: 10.5220/0006257702490255
In *Proceedings of the 9th International Conference on Computer Supported Education (CSEDU 2017) - Volume 1*, pages 249-255
ISBN: 978-989-758-239-4

understanding and critical thinking. PBL (2016 a & b) provides a more detailed bibliography on PBL practices.

A neural pathway, neural tract, or neural face, connects one part of the nervous system to another via a bundle of axons, the long fibbers of neurons. A neural pathway connects a part of the nervous system to another using bundles of axons. The optic nerve is an example of a neural pathway because it connects the eye back to the brain (Wikipedia 2016). Neural Pathway Based Learning (NPL) is a pedagogical technique to use a combination of teaching strategies to create deep understanding, knowledge, and critical evaluation by using techniques such as a multiple choice quiz (MCQ, Synap 2016), animations, games and puzzle based learning, and repeated learning, etc.

Component based SE is now a standard large-scale design strategy in most of industrial systems, including Agents-oriented SE and in Aspects-Oriented SE. Therefore, the author wrote a book on CBSE (Ramachandran 2008) to tackle this drawback in education. Since 2009, we have been successfully teaching CBSE concepts, large-scale design, and design for reuse with use case models (UML) components using Problem-Based Learning (PBL) pedagogical approach and Neural Pathway Based Learning (NPL) techniques. Our approach to teaching CBSE takes you from requirements to component abstraction with specific techniques on how to arrive this step. Each week after an hour lesion, they were provided with practical and hand-on problem based learning lab sessions which consist of practical problem and they are asked to find our multiple solutions often solutions for software problems are multiple and therefore you will see a variety of solutions discovered by students and this has been closely discussed with them to identify what is right and could be wrong. This is also followed by an NPL based activity using MCQs and Puzzles which are available online and they can complete in their own time. The instructor can monitor their success and number of attempts, etc. on blackboard. The feedback and marks are provided instantly to students.

Therefore, students have been enjoying the course with full power of designing CBSE based system development lifecycle, as they are given the "freedom" to develop their own knowledge in their own way with the guidance of a tutor. Our CBSE teaching methods also demonstrated how to use best practice guidelines on component design.

With our unique approach to teaching Software Engineering students include directly mapping UML and conceptual models into component models helps students to critically think putting learning into the context of real-world experience. In addition, most of hands-on exercises made them to think about careful design of component interfaces with design rational discussion in the class amongst their peers, have improved their learning on designing large-scale abstraction and its benefits. This has made a tremendous impact and enthusiasms in students' learning and have been able to publish research papers (Ramachandran and Jamnal 2014) while on this module. Carefully designed labs coupled with implementation labs that are spread equally during a semester have improved students' understanding of software components and productivity with the concept of reuse and system integration. They also learned how useful to have interfaces for system integration as part of their learning in the lab sessions. During implementation, keep it strictly to implement their component design as exactly as possible to UML component model that they learned. In addition, our teaching method includes component-based design for a range of problems**.** This includes both small and large-scale applications, to encourage critical thinking and critical evaluation of the solutions found. PBL experience has been great from students' feedback as they have chosen as the best module of the year from 2014-16.

With a strong focus on CBSE learning in Semester 1 students then moved on to learning service computing, where they implement component-based Service-Oriented Architecture (SOA). In essence, it is time for us to re-visit our wealth of knowledge and experiences in requirements engineering, modelling, design, metrics, and testing strategies that can support components more explicitly. For example, how we can use the idea of system decomposition into subsystems as components and they can be mapped onto fine grain reusable components.

2 PROBLEM BASED LEARNING FRAMEWORK

Problem-based learning (PBL) is one of the key pedagogical approaches to teaching practices based on providing several different types of problems. However, there is a lack of dedicated e-learning environments to support the subject of component-

based design which is a foundation to other modules in our MSc computing cohorts. In addition, existing PBL approaches lack dedicated and directed learning strategies with real-world problems that are commonly used and popular real-world applications ranging from small to large. We have written a dedicated book on the subject knowledge, now we need a web-enabled learning environment supporting our PBL based teaching strategies. Therefore, this project aims to fill that gap and develop such a learning system, which would orient students' toward meaning-making over fact-collecting" (Rhem, 1998). This would have huge potential for enterprise in both academia and industry. We aim to achieve this by providing technology and subject content with different problem scenarios, with different difficulty levels of the problem scenarios and a number of solutions to choose. Visual instructions and exercise will be provided to challenge learners and act as a facilitator (giving guidance) is also part of this project's aim. This project aims to use the existing e-learning environment (blackboard to launch our web-based PBL) for emphasising foundation and practice which is easy and enjoyable. We aim to develop a learning environment which will provide a rich set of visual, games, multi-media, digital artefacts and use of cloud technologies.

This project has currently been funded as part of Leeds Beckett Curriculum Innovation and Digital Learning Projects 2016-17. One of our main objectives of this project is to develop different types of problem-based learning patterns with populated design assets, code assets, examples, and real-world case studies for our students. This will give opportunity to explore current existing E-Learning environment already provided by the university in areas of exploring Problem Based Learning (PBL) for assessment. One of the main outcomes of this project is an intuitive prototype knowledge based learning environment for software components based design and implementation techniques.

Software component based development is the natural foundation for developing web services, software as a service in a cloud computing, etc. The associated technologies in practice grow rapidly. Therefore, it is hard to offer new courses without having a sound foundation for learning these new concepts and technologies that students can appreciate while learning in the e-learning environments.

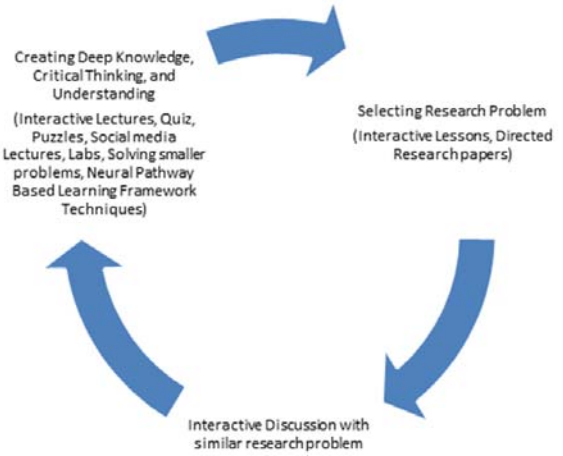

Figure 1: Problem-Based Learning Framework.

We have now witnessed the rapid change in technologies from product based to service based with the emergence of SOA and Cloud computing technologies. Teaching these evolving software concepts and technologies poses many challenges in the HE sector (Mircea 2012 and Ramachandran et. al 2014). Therefore, this project aims to make e-learning intuitive and fun, and allow students to take ownership of their own learning through facing real problem solving experience. For further background our teaching practices in this subject area, please also refer to, The Missing Principles in CBSE Education and in Practice (Ramachandran 2015).

We believe having this project, which will place us at the forefront of virtual and interactive learning our modules on Software components and architecture, service-oriented computing, cloud computing, Big Data and Business Intelligence with Analytics. According to Kolb (2016) and Marzano, Pickering, and Pollock (2001) research has demonstrated that increases can be achieved if factors such as note taking; similarities and differences in students; NPL approach and assigning homework etc. have in the past improved learning. Therefore, our NPL based approaches enhances learning and are designed quizzes, puzzles by taking this into perspective by reflecting.

Software components (Ramachandran 2008; Ramachandran 2012) and software as a service are part of the emerging service computing paradigm. Service-Oriented Architecture (SOA) and service computing is a newly emerging concept for all our software intensive and web-enabled systems. Software component based development is the natural foundation for developing web services, software as a service in a cloud computing, etc.

Furthermore, the associated technologies are growing rapidly. Therefore, it is hard to offer new courses without having a sound foundation for learning these new concepts and technologies that students can experience and appreciate while learning. This project aims to develop a learning environment for emphasising foundation and practice which is practical and enjoyable. This environment will include interactive materials, animations, games-based approach to learning, etc. using a web based learning environment. The outcome of this project will include a learning environment and associated materials for digital learning. This project has also had a world leading industrial partnership with SOA School (www.soaschool.com). Developing a virtual learning environment on SOA and cloud computing will also enhance our e-learning portfolio as well as changing our own IT system in education to help reduce our costs. There are no existing learning environments for this subject expertise as far as we know and this will be integrated with our blackboard environment.

PBL is a learning environment in which the problem drives the learning (UBC, 2015). Problem-based learning is one of the key pedagogical approaches to teach practices based on providing several different types of problems. We aim to achieve this by providing technology and subject content with different problem scenarios **and a number of solutions** to choose. Visual instructions and exercise will be provided to challenge learners. Different types of problem-based learning patterns will be developed with populated design assets, code assets, examples, and real-world case studies. This type of experience for MSc students will also improve overall student experience in learning, giving them flexibility in problem solving and learning.

Our course is one of the few in the country that teaches large scale software development and what is more it is very popular with the students in terms of subject interest and offered at Franchise College and their feedback (Ramachandran and Jamnal 2014 and Ramachandran SFHEA Presentation 2015). However, there is a lack of dedicated e-learning environments to support this subject. This project aims to fill that gap and develop such a system. This would have huge potential for enterprise in both academia and industry. There are two major benefits to students namely enhanced student learning and enhanced recruitment to our courses. This will be a unique technology for promoting learning and teaching for a technology based courses. Students will benefit from the content and achieve learning through some of the interactive content prepared for this project. We believe this will also make us one of the first Learning Environment for Component-Based Software Engineering (LECBSE) provider in the world.

How to introduce this approach in your class room?
Spend first few weeks of introduction to basic design concepts as a refresher course as some of them coming from industry with lots of programming skills. Simultaneously, spend a week or two with labs session on use case modelling. Then introduce CBSE with clear understanding of object-oriented design concepts. At least spend three weeks of lab session on mapping those use cases to components and interface design. We have been using one single case study on banking and ATM components as they are easy to understand as an application. Finally, at least three to four weeks on implementing components as exactly to the UML component standard in Java or C#. We use our approach to blended learning with two learning strategies: Problem-Based Learning (PBL) and Neural-Pathway Based Learning (NPL). The PBL emphasis on problem solving strategies by providing a number of software design challenges to find an ideal solution using CBSE principles and design notations which is checked instantly and provided feedback during the lab sessions. This further refined to analyse their understanding using NPL strategies by providing them with MCQ, Puzzle, and Repeated Interactions. Each topic on CBSE learning has been enhanced with two major learning strategies.

3 LEARNING ENVIRONMENT FOR CBSE (LECBSE)

Technology enhanced learning is based on specific learning strategies which should be combined with current developments in emerging technologies as it is the current for today's learners. For example, the popular use of social media and cloud based technologies. Therefore, we have identified a pillars of technology enhanced learning in the context of emerging technologies such as social media, cloud, and virtual reality. Figure 2 shows the pillars of learning technologies.

Figure 2 Pillars of Technology Enhanced Learning Framework & the Future of E-Learning.

3.1 SOA based Design of LBCSE

We are currently working on the CLT Leeds Beckett funded project on developing a cloud based virtual learning environment to teach Component Based Software Engineering (CBSE) and Service-Oriented Architecture (SOA) and the SOA Based design of LBCSE is shown in Figure 3. Service computing has flexibility to exchange and make changes quickly. In addition, integrating various systems. Chang and Hisao (2011) has proposed SOA based e-learning systems that is characterised as a reusable and interchangeable resources and can reduce the cost of educational services. However, we have proposed an integrated cloud based e-learning environment in Figure 3 (a high level architectural design using the SOA paradigm) and a more detailed architectural design with e-learning services that are integrated with machine learning and data analytics techniques that system can ubiquitously learn about the learners.

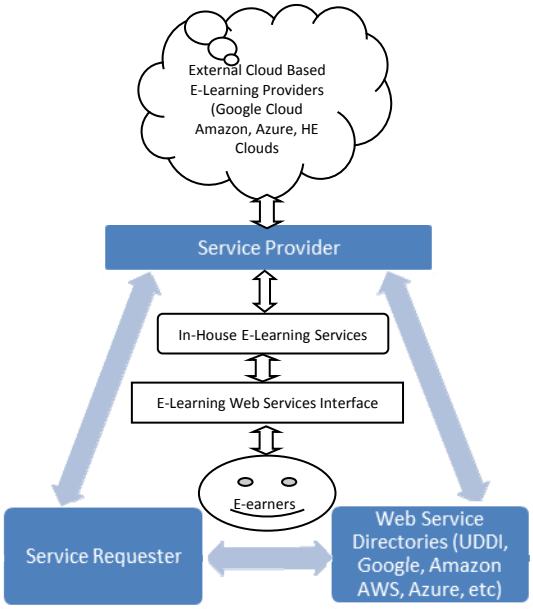

Figure 3 The High Level SOA Architecture Paradigm for E-Learning System

SOA based implementation provides flexibility to change and plugin to emerging web and cloud based technologies. The SOA paradigm is based on a three dimensional view: service provider, service requester and a directory. We have customised the paradigm for E-Learning as shown in Figure 3. Figure 4 shows our interface design for PBL based learning in our VLE. Students can access their learning content from anywhere and on any device.

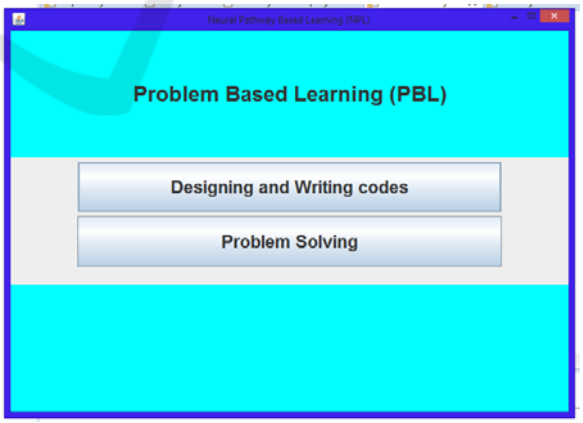

Figure 4 Problem Based Learning (PBL) Techniques.

We have categorised PBL into following problems and solutions categories:

1. Design problems requiring outline solutions (solutions could be more than one type)

2. Given the design solutions requiring relevant code skeleton which conforms to component standard taught in the lectures
3. Given the Code requiring to draw relevant design solutions (it could be more than one)

4 EVALUATION, RESULTS AND ANALYSIS

E-learning has become one of major aspect of our University portfolio in line with emerging technologies worldwide in higher education. In our university this has been shown through the implementation of Centre For Learning And Teaching (CLT) and adoption of UKPSF and Teaching Excellency Framework (TEF) (Ramachandran 2008;2014;2015, TEL 2016). All these they underpin the teaching pedagogies required for our subject area.

4.1 Evaluation of the LCBSE

The research project supports in getting a vision on various principles and different reflections additionally, associates to comprehend how individuals think in unexpected ways and differently. The qualitative process method has chosen for this research project, as it supports to come with deeper concepts of E-Learning, in this project dissertation by using qualitative methods, several questions are prepared in term of usability, Course Content and Activities, Assessments and designing the quizzes. Two types of qualitative methods have used:

- Likert scale.
- Open-ended question.

Likert Scales "is a psychometric response scale primarily used in questionnaires to obtain participant's preferences or degree of agreement with a statement or set of statements. Likert scales are a non-comparative scaling technique and are one-dimensional (only measure a single trait) in nature. Respondents are asked to indicate their level of agreement with a given statement by way of an ordinal scale" (Bertram, 2007). The Open-end question is the unstructured question that the potential answer not suggested by the project researcher and the response of the respondents' not expected, it is her/ his own words and provides the participants a scope to grant the information that respondents think to be suitable for the questions (Tran et al, 2016).

- 13 participants have been involved for the questionnaires.
- 17 questions have been asked based on 7 different aspects.
- Likert scale and one open question have used for evaluation project.
- Participants have been asked to test full system.

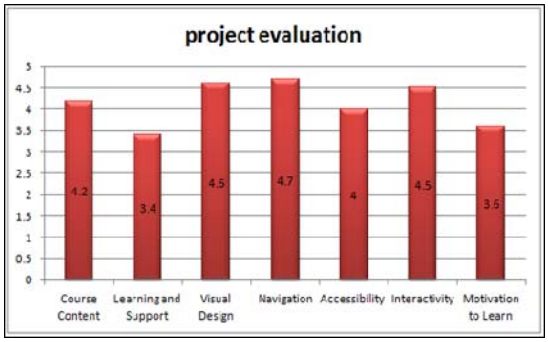

Figure 5: Overall Results of the Project Evaluation by Questionnaire and interviews.

As shown in Figure 5 PBL based learning environment was evaluated by a group learners who has expressed more than 95% satisfactory of their learning experiences.

ACKNOWLEDGEMENTS

We would like to thank CLT (Centre for learning and Technology) at Leeds Beckett University for funding this project.

5 CONCLUSION

It is difficult for leaners in advanced computing courses to catch with deeper understanding of the high level concepts underpins current and emerging technologies. We have developed two advanced modules to teach the foundations of high level concepts of components based development and service computing. However, it also requires different teaching and learning strategies. We have employed successfully a problem based learning in these tow advanced courses. Therefore, we have adopted a combination of Problem-Based Learning (PBL) and Neural Pathway Based Learning (NPL) techniques to teach high level CBSE modelling and concepts, large-scale design, and design for reuse

with UML components. This paper presented the learning environment for PBL based learning on our VLE which provides excellent learning experience for students to learn advanced topics such as software components and service computing.

REFERENCES

BCS (2009) http://www.bcs.org/server.php?show=conWebDoc.32532

Barg, M et al. (2000) Problem-Based Learning for Foundation Computer Science Courses, Computer Science Education, 2000, Vol. 10, No. 2, pp. 109–128

Chang, V. (2016). Review and discussion: e-learning for academia and industry. International Journal of Information Management.

Chang, C. C., and Hsiao, K-C (2011) A SOA-Based e-Learning System for Teaching Fundamental Information Management Courses, Journal of Convergence Information Technology, Volume 6, Number 4. April 2011

CPCHUK (2008) https://cphcuk.files.wordpress.com/2014/01/briefingnote_final.pdf

Fee, S. B., and Holland-Minkley, A.M (2010) Teaching computer science through problems, not solutions. Computer Science Education 20(2): 129-144 (2010)

Kolb (2016) Kolb' s theory learning is a cognitive development, http://www.learning-theories.com/experiential-learning-kolb.html

Marzano, J. R., Pickering, J. D., Pollock, E.J (2001) Classroom Instructions that Works: Research based strategies FOR INCREASING STUDENT ACHIEVEMENT, ASCD, USA

Neural Pathway (2015) https://www.truevitality.com.au/articles/creating-new-neural-pathways-2/, accessed 19th October.

PBL (2016a) A starter bibliography about problem-based learning, http://www.gp-training.net/training/educational_theory/pbl/bibliography.htm

PBL (2016b) Teaching Methods Problem based learning, http://www.nottingham.ac.uk/pesl/resources/problembased/

Ramachandran, M (2008) Software components: guidelines and applications, Nova Publishers, NY, https://www.novapublishers.com/catalog/product_info.php?products_id=7577

Ramachandran, M (2015) Quiz based learning https://app.box.com/s/vt0c9bj3p2qrrnfp3os4qyeey8y6k1vm

Ramachandran, M (2015) Senior HEA Fellowship presentation, Leeds Beckett University, May 2015

Ramachandran, M and Jamnal, G (2014) Developing Reusable .NET Software Components, Science and Information Conference (SAI), IEEE CS, August 27-29, 2014, London, UK

Synap (2015) Multiple choice questions wiki app www.synap.ac

TEL (2016), Teaching and learning Framework, https://www.gov.uk/government/publications/a-framework-for-technology-enhanced-learning

Tran, V.T., Porcher, R., Falissard, B. and Ravaud, P., (2016). Point of data saturation was assessed using resampling methods in a survey with open-ended questions. Journal of Clinical Epidemiology.