International Journal of Engineering Education Vol. 32, No. 2(B), pp. 869–872, 2016
Printed in Great Britain

0949-149X/91 $3.00+0.00

Guest Editorial

Trends in Software Engineering for Engineering Education

MIGUEL ÁNGEL CONDE-GONZÁLEZ
Department of Mechanical, Computer Science and Aerospace Engineering, GRIAL Research group, University of Leon, Spain.
E-mail: miguel.conde@unileon.es

ANTONIO SARASA-CABEZUELO
Faculty of Computer Sciences, ILSA Research group, Complutense University of Madrid, Spain. E-mail asarasa@ucm.es

Software engineering is a key discipline in the field of Computer Science because it provides tools and techniques to manage the complexity of a software project development. Software Engineering involves issues such as requirements specification, project planning, effort estimation, economic estimation, etc.

The application of Computer Sciences in the context of Education as a way to support teaching and learning processes has, as a final consequence, the development of specific software systems such as [1] Learning Management Systems (LMSs), Virtual Simulators, Remote Labs, Expert Systems, etc. Usually, these systems are very complex both from a technological and from a design and development point of view. They implement a high number of functionalities or services, integrate heterogeneous tools, should take into account groups of interdisciplinary experts, and their final users have very different and heterogeneous profiles.

In order to manage the complexity of developing these educational software systems, it is necessary to adapt the generic techniques of Software Engineering. Therefore, these adaptations and new development processes have generated a new discipline that can be denominated as Software Engineering for Engineering Education [2, 3].

The goal of this Special Issue is to explore some of the new trends in Software Engineering for Engineering Education. To that end, 17 papers were selected from 34 submissions. The selection process was done according to a rigorous peer-review process, with three external reviewers who are recognized experts on the research field. This process comprises three stages:

- Preliminary evaluation. In this stage, reviewers read the paper in order to know if the submission fits within the scope of the Special Issue. As a result, some submissions were rejected.
- First review. In this stage the reviewers' team carried out a full review of the submissions that passed the first stage. During this review, reviewers' team provided feedback to the authors so that they could perform changes.
- Second Review. In the third stage, the reviewers checked if their suggestions were taken into account and if the quality of the final version of the paper was the expected one. In case more changes were necessary, new review iterations were carried out.

The result of this process is a Special Issue that includes high quality research papers which cover five topics related to Software Engineering for Engineering Education: Gamification, Simulation, LMS tools, Didactics and Innovation, and Software Engineering Techniques application. Below, the papers are presented grouped according to these topics.

Gamification [4] is a popular topic in a lot of learning experiences because it facilitates the learning process [5] and increases students' motivation [6, 7]. In this Special Issue, there are two papers related to this topic. The first paper by Pascual-Miguel et al. entitled "Design and implementation of a business simulation game tool for services and digital economy courses in engineering degrees" presents a business game simulation tool that can be used to learn concepts related to the digital and services economy, and fulfils the new training needs of engineers in business and management subjects. The second one is entitled "Developing Computer Science Learning System with Hybrid Instructional Method" by Yu-Hsin Hung et al. This paper describes an online gamified learning environment that is used to learn three subjects: Database Management, Programming, and Data Structure. Each subject has problem-solving mini games for students to accomplish tasks using relevant concepts. Results show that the system is very useful for studying Computer Science concepts.

Another topic in the Special Issue is Simulation [8]. In many cases, this topic is associated with gamification. Virtual simulators are very useful tools to learn concepts in engineering contexts because students can carry out experiments in a simple and economic way in a controlled environment. Sometimes, there are experiments that can only be done with a simulator. There are two papers

selected about this topic in this Special Issue. Pozo-Ruz et al. wrote the first one. It is entitled ''Multidisciplinary Power Electronics Courses with Online Simulation Tools'' and it proposes the use of a simulation tool based on the free-software Easy Java Simulation (EJS). This tool aims to make easier the study of Power Electronics at the undergraduate level in engineering courses where the students used to be unfamiliar with Electronic Engineering. The simulator was didactically designed to explain the Power Electronics converter operation step-by-step with a high level of interactivity. Results show that the tool can effectively motivate students to understand the principle of operation of power converters. The second paper included in this topic is entitled ''Design Patterns Combination for Agile Development of Teaching/Learning Haptic Simulators'' by C. Fernández et al. It describes the development of a haptic simulator engine which goal is to be used as a teaching/learning tool using a framework named SHULE. The paper presents a cataract surgery simulator used as a teaching/learning tool. The development of the simulator constitutes an example of the use of the Scrum methodology and how different design patterns can be used.

The development of tools integrated into LMSs [9–11] is also very popular in learning contexts. The reason for this is that LMS do not always include the functionalities that institutions require or they should be adapted to work in a specific way. Some of these tools are integrated and others work in an independent way. This Special Issue includes two papers about this topic. The first contribution is entitled ''OpenIRS-UCM: An Integral Solution for Interactive Response Systems'' by C. Garcia et al. It describes an open source tool called OpenIRS-UCM that implements an interactive response system. It supports different clickers, can be run in different platforms (Windows, Linux and MacOS) and is totally integrated with the Moodle LMS. In addition, the authors of this paper evaluated the application in terms of several software quality metrics (program complexity and usability), the level of acceptance currently achieved, and its potential adoption in real life environments. Moreover, they compare OpenIRIS-UCM with other commercial tools. The second paper was written by F. Jurado and M. Redondo, and it is entitled ''IMS-LTI and Web-Services for Integrating Moodle to an Eclipse-Based Distributed Environment for Learning to Program''. The paper presents an experience about the integration of specific tools to support particular learning activities in a LMS. The authors defined a centralized access point constituted by an LMS, an architecture based on Tuple Spaces, and the use of eLearning IMS-LTI specification to allow communication and information exchange among different services and components.

Other interesting topic in the context of Software Engineering for Engineering Education consists of the adaptation of generic techniques or tools from Software Engineering [12, 13] to educational contexts, or the development of new ones. Usually, the generic techniques that are used in traditional software projects cannot be directly applied in educational environments because these are very different to other business scenarios. Within this topic, the special issue has 4 contributions. The first one entitled ''Tailoring ISO/IEC 12207 for Usability Engineering'' by Cristina Manresa and Esperanza Amengual is focused on the usability issue. Authors have compiled the main usability practices and techniques and have done a mapping between these practices and the software life cycle processes defined in ISO/IEC 12207:2008. The paper aims to facilitate the definition and planning of usability-related courses, with special attention to the integration of usability practices and techniques into software development processes. The second contribution, written by Azeddine Chikh and Jawad Berri, is entitled ''A Software Engineering Framework to Assist Instructors in Eliciting Course Requirements''. Authors propose a software engineering framework that aims to assist instructors in expressing their courses' requirements. In this sense, the framework allows representing the know-what and know-how knowledge for a course teaching. To do so, they define a set of learning activities that would be the building blocks of e-courses. In order to do so, the system has a graphical user interface that provides the necessary tools for producing a course description, that later could be transformed automatically into a course specification that could be implemented as an e-course. The third paper entitled ''A Radial Basis Function Neural Network for Predicting the Effort of Software Projects Individually Developed in Laboratory Learning Environments'' by Cuauhtémoc López et al., poses the use of a mathematical model called Radial Basis function Neural Network (RBFNN) for predicting the effort that graduate students need to develop software projects. Authors have tested that the effort prediction accuracy of a RBFNN is statistically better than that obtained from a Multiple Linear Regression (MLR). In order to test it, both methods were trained from a data set of 328 projects developed by 82 students between the years 2005 and 2010, then, the models were tested using a data set of 116 projects developed by 29 students between the years 2011 and first semester of 2012. Results show that a RBFNN having as independent variables new and changed code, reused code and

programming language experience of students can be used at the 95.0% confidence level for predicting the development effort of individual projects.

The last topic is didactics and innovative methods in learning experiences [14]. It is the largest group of papers with 8 contributions. The first paper entitled ''Learning Project Management Skills in Engineering Through a Transversal Coordination Model'' by Ana Gonzalez-Marco et al. presents an innovative project management teaching approach to complement technical concepts with competences in the field of human skills. The model combines project-based learning and collaborative learning with a transversal coordination between the same subjects in different engineering degree programs, considering the different curricula of each degree involved. With this model, the students can develop group competences like teamwork and communication, and gets professional skills such as leadership, negotiation and team management. The second paper, entitled ''A Teaching-Learning Model for Software Engineering Courses through Sensor-Based Cognitive Approach'', by Indra Gandhi discusses the importance of student's preference in initiating a cognitive based learning environment for engineering education. The authors pose a model to evaluate the cognitive behavior of the students and to analyze the requirements of the students towards Software Engineering related courses. The need for cognitive based learning is assessed through a ''Need For Cognition'' scale with the focus on software system analysis and development. The response is evaluated through wearable sensor technology. The results obtained revealed the fact that there should be a renaissance in engineering education by continuously reviving the teaching methodology for engineering students.

The third paper entitled ''Characterizing Software Engineering Students' Discussions during Peer Instruction: Opportunities for Learning and Implications for Teaching'' by Tom Adawi et al. describes a qualitative research study that aims to explore in what type of discussions are engineering students participating during peer instruction sessions. Results show that students participate in three qualitatively different types of discussions: affirmative discussions, motivating discussions and argumentative discussion, and that students do not always engage in discussions that support their learning in the best way. Taking this into account, the paper discusses the implications of using peer instruction as a teaching method. The next paper was written by Olaf Radant et al. and is entitled ''Assessment of continuing educational measures in software engineering: A view from the industry''. This paper deals with the importance of continuing educational measures to support organizations in their competence management programs. The authors present an interview study regarding continuing education measures in the context of the BearingPoint consulting company. The study involved some of the most experienced software engineering professionals. Results show an increasing relevance of continuing educational measures, provide relevant educational measures, improvement areas in employees' training and instruments for control and evaluation. Pamela Flores et al wrote the paper entitled ''Persistent Ideas in Software Design Course: A Qualitative Case Study'' which presents a research about the persistent ideas that students have when designing software, and analyses possible relationships between them. Authors have carried out a qualitative study with Master degree students in the Software Design course. Results help to identify and address problems related to Software Design course, and show the main problems that students find despite of the instruction. Luis Miguel Serrano-Camara et al. wrote the paper ''MoCAS: A Mobile Collaborative Tool for Learning Scope of Identifiers in Programming Courses''. This paper presents an instructional framework for collaborative learning, called CIF, which is used to analyze the level of Bloom's taxonomy. It also presents a mobile collaborative tool called MoCAS that supports CIF and that is aimed at the domain of scope of identifiers in programming learning. These systems were evaluated in an educational context with respect to students' performance and motivation. Results show that students using CIF and MoCAS obtained statistically significant higher grades and were more motivated than students that do not use MoCAS. The next paper entitled ''Are Learning Software Systems Well-Prepared to Support Self-Regulated Learning Strategies?'' has been written by Mario Manso et al. Authors have evaluated how prepared are the tools that teachers and students use to learn to support self-regulated learning strategies. To do so, they have defined several criteria for designing and implementing software tools with proper support for these self-regulated learning processes and strategies. In addition, the authors have defined a questionnaire to evaluate the level of support offered by a software tool for those strategies. Results from this questionnaire allow improving these tools and provide useful information to facilitate tools selection. The last paper entitled ''Students' Knowledge Sharing to Improve Learning in Academic Engineering Courses'' was written by María Luisa Seis-Echaluce et al. Authors present an experience of the creation of content by students in the context of an engineering course. The paper aims to show the impact in teamwork competence of contents created by students as learning resources.

In order to do so, an experience was carried out. Students created more than 500 learning resources that can be managed by using a knowledge management system, called BRACO. This tool stores information about resources and allow conducting searches according to each student's profile and needs. Moreover, authors did evaluations comparing the students of the group that used the system and resources stored in it (experimental group), with the students that did not experience this intervention (control). From this comparison it was possible to see that the students in experimental group obtained better results in relation to indicators of positive learning results, such as student-student interaction, teamwork development and final grades during the teamwork process.

Finally, we would like to thank Professor Ahmad Ibrahim, Editor in chief of International Journal of Engineering Education, for the opportunity to edit this special issue. In addition, we are grateful to all reviewers for their excellent work for providing feedback to the authors.

References

1. A. Sarasa-Cabezuelo and J. L. Sierra-Rodríguez, Ingeniería del Software en eLearning, *IEEE-RITA*, **6**(4), 2011, pp. 164–166.
2. F. J. García-Peñalvo, A. S. Cabezuelo and J. L. S. Rodríguez, *Software Educativo: Casos de Estudio y Métodos de Desarrollo. Versión Abierta Español–Portugués*, 2013.
3. F. Garcia-Penalvo, A. S. Cabezuelo, and J. L. S. Rodriguez, Innovating in the Engineering Processes: Engineering as a Means of Innovation, *IEEE Revista Iberoamericana de Tecnologias del Aprendizaje*, **9**(4), 2014, pp. 131–132.
4. J. Hamari, J. Koivisto, and H. Sarsa, *Does Gamification Work?—A Literature Review of Empirical Studies on Gamification*, presented at 2014 47th Hawaii International Conference on System Sciences (HICSS), 2014.
5. S. Deterding, D. Dixon, R. Khaled and L. Nacke, *From game design elements to gamefulness: defining "gamification"*, presented at Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments, Tampere, Finland, 2011, pp. 9–15.
6. B. Amir and P. Ralph, *Proposing a theory of gamification effectiveness*, presented at Companion Proceedings of the 36th International Conference on Software Engineering, Hyderabad, India, 2014, pp. 626–627.
7. C. I. Muntean, *Raising engagement in e-learning through gamification*, presented at 6th International Conference on Virtual Learning ICVL, 2011, pp. 323–329.
8. C. Aldrich, *Simulations and the Future of Learning: An Innovative (and Perhaps Revolutionary) Approach to e-Learning*: Wiley, 2003.
9. H. Coates, R. James and G. Baldwin, A Critical Examination Of The Effects Of Learning Management Systems On University Teaching And Learning, *Tertiary Education and Management*, **11**(1), 2005, pp. 19–36.
10. M. Á. Conde, F. J. García-Peñalvo, M. Alier, E. Mayol and C. Fernández-Llamas, Implementation and design of a service-based framework to integrate personal and institutional learning environments, *Science of Computer Programming*, **88**, 2014, pp. 41–53.
11. F. J. García-Peñalvo, M. Á. Conde, M. Alier and M. J. Casany, Opening Learning Management Systems to Personal Learning Environments, *Journal of Universal Computer Science*, **17**(9), 2011, pp. 1222–1240.
12. M. A. Conde, F. J. Garcia-Penalvo, D. A. Gomez-Aguilar and R. Theron, Exploring Software Engineering Subjects by Using Visual Learning Analytics Techniques, *IEEE Revista Iberoamericana de Tecnologias del Aprendizaje*, 10(4), 2015, pp. 242–252.
13. J. L. Sierra-Rodríguez and A. Sarasa-Cabezuelo, Special collection on software engineering for e-learning-part I, *Journal of Research and Practice in Information Technology*, **44**(2), 2012, pp. 107–109.
14. A. Sarasa-Cabezuelo and J. L. Sierra-Rodriguez, *Software engineering for eLearning*, presented at Proceedings of the First International Conference on Technological Ecosystem for Enhancing Multiculturality, Salamanca, Spain, 2013, pp. 81–86.

Miguel Ángel Conde-González holds a PhD in Computer Science (2012, University of Salamanca). He has worked for GPM and Clay Formación Internacional R&D department in different eLearning projects. From 2010 to 2012 he was researching at the University of Salamanca. Now he is an associate lecturer at the University of León, member of GRIAL research group of the University of Salamanca and the Robotics Research Group of the University of León. His PhD thesis is focused on the merging of informal, non-formal and formal learning environments.

Antonio Sarasa-Cabezuelo, is an associate professor in the Computer Science School at the Complutense University of Madrid, Spain (UCM). His research is focused on the language-oriented development of XML-processing applications, and on the development of applications in the fields of digital humanities and e-learning. He was one of the developers of the Agrega project on digital repositories (a pioneer project in this field in Spain). He is a member of the research group ILSA. He has participated in several research projects in the fields of software language engineering, digital humanities and e-learning, and he has published over 50 research papers in national and international conferences and journals.