A Course Design for Individualized Learning Leveraging Didactic Functions of Mobile Learning Devices

https://doi.org/10.3991/ijac.v10i2.7352

Sigrid Schefer-Wenzl(✉) and Igor Miladinovic
University of Applied Sciences Campus Vienna, Austria
sigrid.schefer-wenzl@fh-campuswien.ac.at

Abstract—Mobile computing and wireless technologies are transforming traditional educational patterns. The use of mobile devices for educational purposes led to what is known as *mobile learning*. In recent years, mobile learning strategies have been increasingly adopted in higher education. They support personalized, informal, situated, contextualized and ubiquitous learning. In practice, we have witnessed that courses in software engineering education often do not lead to the expected learning outcomes. In this paper, we will therefore present a motivating and learner-centered approach for mobile e-learning in application prototyping. We use mobile devices as key enabler serving different didactic functions in the context of learning, prototyping and collaboration.

Keywords—application prototyping, e-learning, mobile learning, software engineering education.

1 Introduction

Over the last few years, we can see a strong shift from hardware towards software in many different areas (see, e.g., [1-4]). A result of this process is an increased need for highly skilled software engineers. Our University is strongly committed to help the industry to satisfy this demand by ensuring that our graduate students are very familiar with modern software engineering and developments tools.

In scope of the Bachelor-level course “Mobile Application Development” we have introduced a mobile e-learning approach. The main goal was to create a motivating learning environment, where the students would be able to individually identify their development potential and to continuously improve it by targeted learning.

We designed and optimized all the learning material of the course for the presentation on smartphones and tablets. These devices are also used for application development, enabling interactive testing and making successful progress immediately visible. By using appropriate tools students are also able to communicate and collaborate virtually by their devices. This triple function of smartphones and tablets as mobile e-learning devices, together with a selected mix of multiple modern didactic methods make this course to a best practice template for teaching advanced application prototyping. In this paper we will present the detailed concept for our applied mobile e-learning approach as well as major learning outcomes.

The remainder of this paper is structured as follows. Section 2 discusses related work on mobile learning in higher education as well as in software engineering education. Subsequently, Section 3 introduces our best-practice mobile e-learning approach for application prototyping including learning outcomes and the didactic method mix. Section 4 concludes the paper.

2 Related Work

2.1 Mobile Learning in Higher Education

In the area of teaching and learning in higher education, mobile computing technologies have become a very important research topic over the past few years (see, e.g., [5-8]). The use of mobile devices for educational purposes led to what is known as *mobile learning*, providing new opportunities for active learning by utilizing the advantages of mobility and wireless technology [7, 8]. The advancements in communication technology enable learners to use mobile technology anytime and anywhere to access up-to-date educational resources and individualize learning [9]. In addition, it facilitates the interaction with other students to share information or work collaboratively on a task or project. Thus, mobile learning is both formal and informal. Situated and contextualized learning, i.e. the application of knowledge and skills to specific contexts, is also supported, since learners can apply what they learn immediately [5, 10].

Some students also experience frustrations and challenges in mobile learning scenarios (see, e.g., [10]). Frustrations may result, for example, from the reluctance of instructors to provide learning material that is optimized for mobile devices. Moreover, other applications on the device may distract the students from learning. Some studies also argue that mobile learning leads to constant connectivity and thereby fosters digital addiction of students as their personal identity merges with their mobile devices [10].

2.2 Mobile Learning in Software Engineering Education

Fox et al. list several challenges instructors face when teaching software engineering topics [11]. Amongst others, these challenges include the vast amount of material that needs to be taught in short time in order to enable students to start their own software projects, the limited quality of most software engineering textbooks, as well as the dissatisfying learning outcomes of such courses. It is also important that students receive proper feedback on their practical programming tasks in a timely manner which is often not realized in software engineering courses [12].

With the advent of mobile devices and especially smartphones, several authors started to argue that the teaching of programming should be done directly on the mobile devices itself (see, e.g., [13-16]). Many authors also advocate for combining traditional, e-learning, and mobile learning strategies in order to successfully teach programming contents (see, e.g., [17]). One positive effect of integrating mobile learning in software engineering education is that it allows implementing and including new technologies to make students more prepared for their future work [18]. Thus, besides students' interests, potential employment opportunities were important motivations for our focus on mobile app development for teaching software engineering principles.

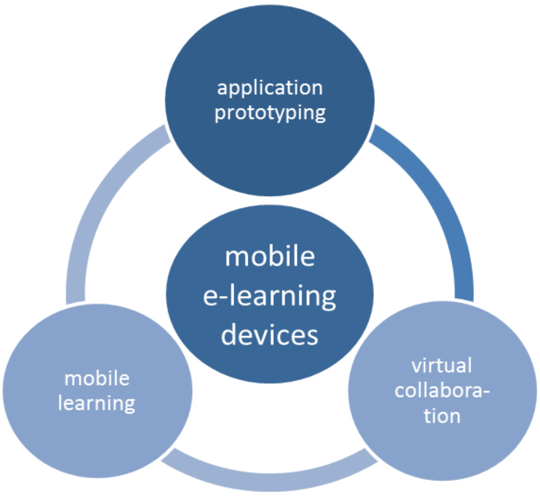

Fig. 1. Triple didactic function of mobile devices in our course design

3 Best-Practice Approach

3.1 Problem statement

The primary goal of a Bachelor program in Computer Science is to provide students the foundations for future work and careers in computation-based problem solving. Our study program at the University of Applied Sciences Campus Vienna emphasizes the development of analytical skills, acquisition of knowledge and understanding of systems, programming languages, and tools required for effective computation-based problem solving. Key skills gained with our degree program include a solid understanding of software development as well as comprehension of modern software engineering principles.

However, as experience has shown, students' capability of practical application of programming skills and software engineering principles leaves room for improvement. We therefore currently undergo a substantial update and refurbishment of our study program in order to strengthen these skills. One key element is to design and implement more motivating and learner-centered course concepts. In this paper, we present one concept which we applied in the area of mobile application development. The course design includes a selected mix of several best-practice didactic methods tailored to cover the identified shortcoming.

3.2 Mobile Learning Concept

The main goal of our Bachelor-level course "Mobile Application Development" is the design, implementation, and testing of mobile applications for smartphones and tablets. Before designing our course concept, we formulated a number of guidelines (similar to, e.g., [19]):

- Our focus is on teaching fundamental principles in software engineering and application prototyping instead of skills that turn out to be obsolete quickly. Therefore, if any vendor-specific technology is used throughout the course it is merely for knowledge delivery.
- In order to provide a realistic setting and thereby prepare students for their future jobs, the course challenges students with all aspects of application development: from designing a graphical user interface, over using sensors, to leveraging backend online services.
- The course provides solid theoretical background information to enable students to apply their knowledge also on future technologies in the area of mobile application development.
- Within the course, students get familiar with the basic concepts and will be guided through their first steps of application development. Later on, students are more and more encouraged to gain knowledge via their own research. However, the instructors provide extensive coaching facilities throughout the whole course.
- We foster the collaboration between students by encouraging group works and providing tools that support virtual collaboration. The collaborative element should enable a more motivating development experience and eventually lead to better results.
- Besides using mobile devices as development tools, they are also applied as primary learning enabler (see Figure 1). All course material as well as the didactic concept are optimized for mobile devices, such as smartphones and tablets. Thus, mobile devices serve as the primary learning object for students to enhance programming skills by developing mobile applications. In addition, mobile devices serve as the primary learning medium as all learning material is designed for mobile learning. All collaboration tools are also selected for use on mobile devices. This triple didactical function of mobile devices aims at ensuring a sustainable learning success by fostering situational and contextual targeted learning.

3.3 Learning Outcomes

We defined the following learning outcomes for our "Mobile Application Development" course:

- Understand and apply an effective software engineering process, based on knowledge of widely used software process models.
- Employ team working skills including organized planning, time management and inter-group negotiation.
- Capture, document and analyze requirements.
- Translate a requirements specification into an implementable design, following a structured process.
- Make effective use of UML and apply software design strategies.
- Design a testing strategy for a software system, employing techniques such as unit testing, test driven development and functional testing.
- Evaluate the final projects by checking compliance with the requirements, and analyze the design and implementation.

3.4 Didactic Method Mix

Our "Mobile Application Development" course is divided into two parallel parts, a lecture and a tutorial part (see Figure 2). In order to achieve the learning outcomes, we apply a set of well-known best-practice didactic methods. In the lecture, we implemented the didactic concept of an inverted classroom (see, e.g., [20, 21]). In an inverted classroom, lecture content is provided over some asynchronous medium. Students access this content outside of class. During the class hours students are involved in learning activities such as in-class assignments, laboratories, and discussions.

The students reflect each self-study phase in a learning diary, where they answer questions like: "What did I learn? What was new to me? Was there something that changed my views and why? Focus on and analyze the themes important to you." and "What did I not understand? What went against my own ideas and why? What was less comprehensible and why? Focus on and analyze questions that left you puzzled." Based on the learning diary entries, the in-class hours start with clarification of open questions and then focus on further deepening the contents. This inverted classroom concept enables a strong learner-orientation as well as individualization of course contents during class hours. The mobile learning aspect further supports location- and time-independent learning as well as immediate interaction between students about learning contents.

The tutorial part of the course is intertwined with the lecture part. It aims at applying the concepts from the lecture within a comprehensive student project, i.e. the design and development of a mobile application for smartphones and tablets. The student groups work on these projects on their own. In addition, they are supported by regular meetings with their instructors as well as a range of software tools for project management, code version management and tools supporting the virtual collaboration of students during the software development process. The main goal of this tutorial course part is to introduce students to the nature and complexity of real software projects.

We officially start the tutorial part with a *Hackathon* event. A hackathon (hacking marathon) can be described as a problem-focused computer programming event, where people come together to develop prototype applications by collaborating intensively over a short period of time [22]. Organizations use hackathons to test their new products and services as well as to generate new ideas [23]. They either focus on attracting external developers or requesting own employees to focus on a specific problem. Hackathons are also seen as a tool to broaden participation of women and minorities [24]. In accordance with [25], the main goals of our hackathon event where to support team formation, provide an opportunity to creatively develop and assess app ideas via face-to-face interactions and thereby increase motivation of the students to put a lot of effort in realizing their ideas.

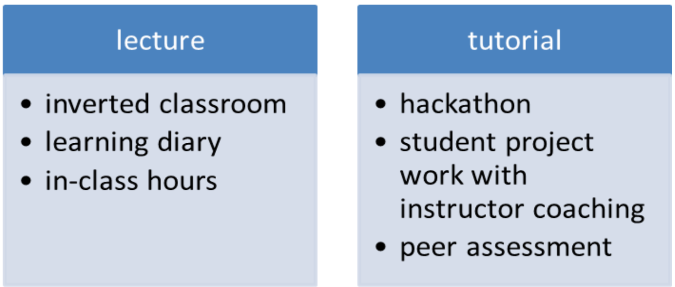

Fig. 2. Didactic method mix in the Mobile Application Development course

Within the hackathon event, students first collect in a brainstorming manner as many app ideas as possible, then form teams and subsequently reduce the number of app ideas, until each team has decided on one particular app idea, which they are going to develop. Following [26], each team defines the requirements for their app in four prioritization categories using the *MoSCoW* system - *M*ust have, *S*hould have, *C*ould have, and *W*on't have but would like [26]. After the hackathon event, students start working on their projects with regular coaching meetings with their instructors.

After the projects are finished, the final apps are evaluated via peer assessment. The projects are distributed between the different student groups, which then check the conformance between the original MoSCoW requirements and the behavior of the final implementation.

3.5 Advantages and Obstacle

The main advantages of our approach are: (1) the support of individual learning patterns concerning time, location, and speed of learning; (2) students can access learning material situational and contextualized while developing their mobile applications; (3) students anchor knowledge by deepening and discussing contents in the lecture which are first captured by the students on their own; (4) students apply the contents in a complex software development project.

The main obstacle in our learning setting is that students enter our course with very different levels of programming skills. This is mainly because we also offer a part-time study allowing students to pursue a study while working full time. Thus, our student groups include professional software developers next to students with very limited software development experience. Our mobile learning concept addresses this issue by enabling individualized learning speed and intensity.

4 Conclusion

Teaching software engineering topics is a challenging task. In order to meet the demands of the mobile age, research suggests teaching programming directly on mobile devices. This may be combined with another educational trend called mobile learning, where mobile devices are used as the primary learning medium. One positive effect of integrating mobile learning in software engineering education is that it allows implementing and including new technologies to make students more prepared for their future work. We therefore designed a Bachelor-level course concept for Mobile Application Development using smart phones and tablets as the key tool for learning, prototyping, and collaboration. In future research, we will apply and evaluate this course concept in different student settings.

5 References

[1] K. Joshi and T. Benson, "Network Function Virtualization", In: *IEEE Internet Computing*, vol. 20, nr. 6, pp. 7-9, 2016. https://doi.org/10.1109/MIC.2016.112

[2] D. Kreutz et al, "Software-Defined Networking: A Comprehensive Survey", In: *Proceedings of the IEEE*, vol. 103, nr 1, pp. 14-76, 2015. https://doi.org/10.1109/JPROC.2014.2371999

[3] R. Nane, "A Survey and Evaluation of FPGA High-Level Synthesis Tools", In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol 35., nr. 10, pp. 1591-1604, 2015.

[4] A. A. Abidi, "The Path to the Software-Defined Radio Receiver", In: *IEEE Journal of Solid-State Circuits*, vol. 42, nr. 5, pp. 954-966, 2007. https://doi.org/10.1109/JSSC.2007.894307

[5] M. Ally, J. Prieto-Blazquez, "What is the future of mobile learning in higher education?", In: *Revista de Unicersidad y Sociedad del Conocimiento (RUSC)*, vol.11, nr. 1, pp. 142-151, 2014.

[6] J. B. Roberts, "Accessibility in M-Learning: Ensuring equal access", In: Z. L. Berge and L. Y. Muilenburg (Eds.), *Handbook of mobile learning*, New York, NY, Routledge, 2013.

[7] P. R. H. Ramos, F. J. G. Penalvo, M. A. C. Gonzalez, "Towards Mobile Personal Learning Environments (MPLE) in Higher Education", In: *Proc. of the 2nd International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM),* Salamanca, Spain, October 2014. https://doi.org/10.1145/2669711.2669973

[8] M. Sarrab, M. Elbasir, S. Alnaeli, "Towards a Quality Model of Technical Aspects for Mobile Learning Services: An Empirical Investigation", In: *Computers in Human Behavior*, vol. 55, part A, pp. 100-112, 2016. https://doi.org/10.1016/j.chb.2015.09.003

[9] L.F. Mottiwalla, "Mobile Learning: A Framework and Evaluation", In: *Computers in Education*, vol. 49, nr. 3, pp. 581-596, 2007. https://doi.org/10.1016/j.compedu.2005.10.011

[10] J. Gokas, M. M. Grant, "Mobile Computing Devices in Higher Education: Student perspectives on learning with Cellphones, Smartphines & Social Media", In: *Internet and Higher Education*, vol 19., pp. 18-26, 2013. https://doi.org/10.1016/j.iheduc.2013.06.002

[11] A. Fox et al., "Software Engineering Curriculum Technology Transfer: Lessons learned from MOOCs and SPOCs", Technical report, EECS Department, University of California, Berkeley, 2014.

[12] T. Staubitz et al., "Towards Practical Programming Exercises and Automated Assessment in Massive Open Online Courses", In: *Proc. of the 4th Annual IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE)*, Zhuhai, China, 2015. https://doi.org/10.1109/TALE.2015.7386010

[13] M. Pasamontes et al. Easy mobile device programming for educational purposes. *In Proc. of the 44th IEEE Conference on Decision and Control, and the European Control Conference 2005*, Seville, Spain, December 2005. https://doi.org/10.1109/CDC.2005.1582691

[14] N. Tillmann et al., "The Future of Teaching Programming is on Mobile Devices", In: *Proceedings of the 17th ACM annual conference on Innovation and technology in computer science education (ITiCSE)*, Haifa, Israel, July 2012. https://doi.org/10.1145/2325296.2325336

[15] Y. C. Hsu and Y. H. Ching, "Mobile App Design for Teaching and Learning: Educators' Experiences in an Online Graduate Course", In: *The International Review of Research in Open and Distance Learning,* vol. 14, no. 4, pp. 117-139, 2013. https://doi.org/10.19173/irrodl.v14i4.1542

[16] A. Esakia, D. S. McCrickard, "An adaptable model for teaching mobile app development", In: *IEEE Frontiers in Education Conference (FIE)*, Erie, PA, USA, December 2016. https://doi.org/10.1109/FIE.2016.7757478

[17] G. Venugopal-Wairagade, "Study of a Pedagogy Adopted to Generate Interest in Students Taking a Programming Course", In: *Proc. of the International Conference on Learning and Teaching in Computing and Engineering (LaTICE),* Mumbai, India, April 2016. https://doi.org/10.1109/LaTiCE.2016.27

[18] L. Briz-Ponce, "Mobile Devices and Apps, Characteristics and Current Potential on Learning", In*: Journal of Information Technology Research (JITR)*, vol. 8, nr. 4, pp. 26-37, 2015. https://doi.org/10.4018/JITR.2015100102

[19] K. Sung and A. Samuel: "Mobile Application Development Classes for the Mobile Era", In: *Proc. of the 2014 conference on Innovation & technology in computer science education (ITiCSE),* Uppsala, Sweden, June 2014. https://doi.org/10.1145/2591708.2591710

[20] M. J. Lage, G. J. Platt and M. Treglia, "Inverting the Classroom: A Gateway to Creating an Inclusive Learning Environment", In: *The Journal of Economic Education,* vol. 31, no. 1, pp. 30-43, 2000. https://doi.org/10.1080/00220480009596759

[21] G.C. Gannod, J.E. Burge and M.T. Helmick, "Using the Inverted Classroom to Teach Software Engineering", In: *Proc. of the 30th International Conference on Software Engineering (ICSE),* Leipzig, Germany, May 2008. https://doi.org/10.1145/1368088.1368198

[22] H. Topi and A. Tucker, "*Computing Handbook. Information Systems and Information Technology*" 3rd edition, CRCPress, 2014.

[23] B. Rosell, S. Kumar, J. Shepherd, "Unleashing innovation through internal hackathons", In: *Proc. of the IEEE Innovations in Technology Conference (InnoTek)*, Warwick, RI, USA, May 2014. https://doi.org/10.1109/InnoTek.2014.6877369

[24] G. Richard et al., "StitchFest: Diversifying a College Hackathon to Broaden Participation and Perceptions in Computing", In: *Proc. of the 46th ACM Technical Symposium on Computer Science Education (SIGCSE)*, Kansas City, MO, USA, March 2015. https://doi.org/10.1145/2676723.2677310

[25] E. Trainer et al., "How to Hackathon: Socio-technical Tradeoffs in Brief, Intensive Collocation", In: *Proc. of the International Conference on Computer-Supported Collaborative Work (CSCW)*, San Francisco, CA, USA, February 2016. https://doi.org/10.1145/2818048.2819946

[26] D. Clegg and R. Barker, "*Case Method Fast-Track: A RAD Approach*", Addison-Wesley, Boston, MA, USA, 1994.

AUTHORS

Sigrid Schefer-Wenzl is a senior researcher and lecturer at the University of Applied Sciences Campus Vienna, Austria (sigrid.schefer-wenzl@fh-campuswien.ac.at).

Igor Miladinovic is head of the degree program Information Technologies and Telecommunication at the University of Applied Science Campus Vienna, Austria (igor.miladinovic@fh-campuswien.ac.at).

This article is a revised version of a paper presented at the International Conference on E-learning in the Workplace 2017 (ICELW 2017), held in June 2017, at Columbia University in New York, NY, USA. Article submitted 15 May 2017. Published as submitted by the authors 24 June 2017.