Relevance of UCD Education to Software Development – Recommendation for Curriculum Design

Amir Dirin[1] and Marko Nieminen[2]
[1]*Business Information Technology, Haaga-Helia UAS, Helsinki, Finland*
[2]*Computer Science Dept., Aalto University, Espoo, Finland*

Keywords: Project based Learning, User-Centred Design, Mobile Application Development.

Abstract: This paper describes the implementation and impact of specific focus on user-centred design (UCD) methods and practices to a mobile application concept development course. The course has been held for two consecutive years at a University of Applied Sciences. This project-based course educates students in teamwork, user study techniques, data interpretation, and scenario-based design. Moreover, by applying usability assessment methods such as heuristic evaluations and usability tests, students learn the basics of usability engineering methods and principles. In addition to outcomes on students' learning, the course has resulted in several innovative mobile application concepts some of which are in the process of being commercialized. This paper presents student feedback about the applied methods months after the completion of the course. The questionnaire-originating results indicate that students gain knowledge and skills that are very valuable for their future careers in software and service development. Knowledge on UCD is relevant in contemporary mobile application development and an important asset for students to become competitive in job markets. Almost all students have applied the educated methods in their work, and they are willing to promote the UCD methods and framework to their professional peers. The results are further used to discuss and emphasize the importance of UCD education.

1 INTRODUCTION

Users' role in product concept development has become increasingly important (Kujala et al. 2005) (Constantine 2006). User-Centred Design (UCD) in the development of interactive systems and devices has an increasing importance in product development organizations (Nieminen 2004), (Gould and Lewis 1985), and (Gould et al. 1997). Contemporary software development methodologies and approaches, such as UCD, lean (Stone 2012), and agile (Szalvay 2004) promote customers' and users' involvement in application development. This trend and market demand has also had an impact on the arrangements and approaches in software development companies (Cusumano 2008) (Bosch 2009). This progress introduces new demands, skill requirements and mindset changes in teaching and learning of software development.

The trend on customer and user focus has changed the demands in software industry in hiring new employees. This was specially confirmed in Finland based on the interview study conducted with IT companies by Haaga-Helia University of Applied Sciences (Soitinaho and Palviainen 2015). Software development skills are not anymore limited to programming knowledge (Ruparelia 2010). Programmers are expected to have basic user study and negotiation skills in addition to technological proficiencies. Software companies are searching for employees, such as consultants and programmers, who have competencies in engaging with end users. The employees must be able to conduct end user studies, analyze the collected data, and derive requirements and specifications based on such findings. As a result of such activities, the resulting software is more likely to meet users' needs and expectations (Kristensson et al. 2002) which is important for software business.

Contrasting with the past, a software development company does not anymore receive full requirements directly from customers. The traditional role as an implementation partner that delivered software based on the pre-defined requirements has changed. Nowadays, software developers are expected to communicate and interact closely with customers when creating and completing the requirements and implementation for

Dirin, A. and Nieminen, M.
Relevance of UCD Education to Software Development – Recommendation for Curriculum Design.
In *Proceedings of the 8th International Conference on Computer Supported Education (CSEDU 2016) - Volume 2*, pages 112-120
ISBN: 978-989-758-179-3

software and digital services.

Industry and education always impact each other (Richardson and Hynes 2008) despite occasional conflicts on the overall values (Tasker and Packham 1993). The demand from industry drives the educational content and, vice versa, novel development methods and practices traverse to companies as students with new skills are hired as new employees. For a sustainable software development business, end-users' involvement in development gaining increasing importance in ensuring the acceptance and perceived quality of the resulting system. In addition to ascertaining that the resulting software is error-free, developers must make sure that it is usable and fulfills users' essential needs. Developer skills, such as user studies, data analysis, and usability assessment have become increasingly important.

Undergraduate students gain their basic technological competence in information technology (IT) related courses at the University of Applied Sciences. Traditionally, IT faculties of these educational institutes have provided various courses to teach the latest technologies in different semesters. These courses are both basic and advanced courses in programming and software engineering. After the completion of their degree program, students are hired by software companies to various positions as programmers, consultants, and product managers, to mention a few.

In order to anticipate the changing demands of software companies as well as to improve students' competitiveness at the job market, we have included a user-centered design course to the course curriculum of the Business Information Technology (BIT) faculty. At this course, we have been teaching human factors in software development processes. The course is a six credit compulsory course for second semester students. A typical number of students who participate in the course varies between 20-30 students per semester. The course contains basic team working requirements, methods and techniques for user studies such as diary, questionnaire, interview etc. and various qualitative data analysis methods, such as transcript coding (Weston et al. 2001), task and environment analysis (Hackos and Redish 1998a) and affinity diagramming (Holtzblatt et al. 2005). Additionally, students learn to do concept design with scenario-based design (Rosson and Carroll 2002) and low-fidelity prototyping, as well as various usability assessments methods. Students are given a project topic that they have to work on as a team of 4-5 fellow students. The topics are either recommended by partner companies, given by course instructors, or they can be selected and defined by the students themselves. In addition to these, students are given a presentation topic that they have to present to the class. The presentation topics, cover the theoretical parts of the course e.g. user study methods, usability assessments, benefit of the usability etc. At the end of their presentation students are asked to raise two questions from their presentations that the class should answer with their own words and submit their result by the due date. After the presentation, the groups start to work on the project based on given UCD framework phases. At each phase, groups share their findings with other classmates.

2 UCD METHODS AND PROCESS AT THE COURSE

As mentioned in the widely-accepted principles of user-centered design in ISO 9241-210, we need to involve users in the software development process. Based on users' feedback the design will be modified. User requirements should have the focus in all stages of product development cycle (ISO 2010). ISO 9241-210 defines three different levels for UCD: I. Cooperative design; designers and the user involved in all stages; II. Participatory design; users' occasionally participate in the design process; III. Contextual design; design based on the actual context.

Based on widely known UCD principles, Preece et al. (2002) recommends user involvement in various stages of development. Additionally, Preece provides appropriate methods at various stages of the artifacts / product development. These methods are also applied during project development by students.

2.1 UCD Framework for Student Projects at the Course

In the fall of 2013, we run the course with a UCD framework. Students were asked to apply the UCD framework for m-learning application development (Dirin and Nieminen 2014a). The applied UCD framework consisted of three main parts: identification and definition of the stakeholders/role-players (Dirin and Nieminen 2014b), exploration and description of the context-of-use, and processes (Dirin and Alamäki 2015).

The UCD framework instructs that the application stakeholders are identified at the elicitation phase. This ensures users' involvement in defining the requirements and impacting the design of

the application concept as early as possible. The stakeholders of the potential application often directly or indirectly affect the application development at the various phases of UCD framework. Figure 1 presents the UCD framework for m-learning application development.

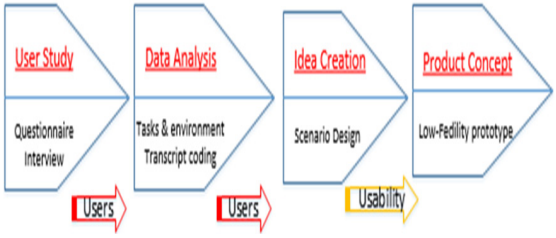

Figure 1: The applied UCD development framework.

The framework requires users' involvement in all stages of the application concept development. Accordingly, the project group is asked to carry out the processes to attain the required concept design of the given project topics. The phases of the framework are as follows:

User Study – This is done by applying methods such as questionnaires and semi-structured interviews. In this phase, the designer aims to learn about the users' existing means to handle their work related tasks.

Data Analysis – Analysis of the collected data in user study phase. The analysis consists of transcript coding of user interviews (Saldana 2009) as well as the analysis and description of users' tasks and environments (Hackos and Redish 1998b). The overall requirements for the target application are identified in this phase.

Idea Creation – By using affinity diagrams (Holtzblatt et al. 2005), actions and requirements created at the previous phases are categorized. Use cases and scenarios are applied as design methods to create a communicable description of the application concept to the target users.

Product Concept – Scenarios are shared with 3-5 users for their feedback. A scenario reflects the potential overall concept of the application. After users' review of the scenario, the designers conduct a short, semi-structured interview to learn about users' viewpoints about the application concept and the intended functionalities. User feedback is analyzed to validate the feasibility of the concept, to ensure that the users and designers share the same understanding of the potential application, and that the functionality of the application fulfils users' needs. When necessary, designers may return back to the previous phase to modify the scenarios. After validating the concept through scenarios, students are asked to design a low-fidelity or high-fidelity prototype which is based on the proposed scenarios. Finally, students are requested to perform usability evaluation with heuristic evaluation guidelines (Nielsen 1995) on the prototype.

Most project topics that are given to students are real projects originating from industrial companies. As a result, students are advised to conduct the user study with actual stakeholders and potential application users.

Finally, the project teams have to share the final concept of their application with their classmates and submit the final report for their course assessment. In addition to being just an assignment for studies, those groups of students who come up with a novel mobile application prototype are advised to contact startup school for possible commercializing support.

3 IMPLEMENTATION OF THE COURSE

Students are requested to conduct the user study after they have studied and learned the phases and methods of the framework. Students develop competence on user study techniques and research ethics.

After students have successfully conducted the user study, they have to provide the interview transcript and prepare a presentation about the interview experiences. This is shared with other classmates. This helps the other groups to learn about their fellow students' achievements and help them to identify what was done properly in their project.

For the next step, the teacher describes the data analysis methods in face-to-face sessions through examples and class based activities. These include transcript coding, task and environment analysis, and the creation of affinity diagrams. Students are given time for two weeks to create the needs and requirements list based on the data analysis sessions. Similarly, as in the previous phases, the project groups have to report their findings and present their findings to other classmates in a face-to-face session. This phase improves students' skills in teamwork as they have to work as a team to come up with a list of requirements.

Similarly, students are requested to report, and categorize the requirements list by applying affinity diagrams and finally share the requirements list and categorized list of requirements with fellow classmates in a face-to-face session. Following that, students must write a scenario or scenarios which contain all the categorized requirements that represent the potential application functionalities. The scenario

method makes students learn that they can present an application concept even through a simple story.

As this course is taught in the second semester of the Business Information Technology (BIT) degree program, students learn the importance of the user study and concept development through scenario. This is another mind-set change that the initial software application concept is not necessarily presented only by graphical presentations or coding. However, they eventually have to proceed and present the concept graphically as a low-fidelity prototype. The scenario then is shared and reviewed with potential users. Often, 3 to 5 users are needed to review, 3 of whom are new users and the rest are users who have participated in the requirements elicitation phase. Those in the groups who already have experience in coding have the option to design a non-functional prototype through coding or using the commercial prototype design applications such as Marvel (Marvel 2014). The low-fidelity prototype will then be evaluated in a usability lab study with potential users. This is typically the first usability lab experience for the students. For this part, they need to prepare a set of tasks that users must carry out during the test sessions. In a normal way, the test sessions are recorded to enable further analysis.

Table 1: Summary of prototypes and designs from the course.

Application	Prototype Design	
	Context	*Users*
M-learning app (Java)	Higher education	Students and Teachers
Driving Licenses	Private driving school	Driving licenses' candidate Instructors
Business Game	Higher education	Students and Teachers new visitors
mHealth	Hospitals	Nurses Doctors Administrator
Tourism Guidelines	Small business	Marketing Sales and Managers

4 OUTCOMES: CONCEPTS AND APPLICATIONS FROM THE COURSE

What kind of innovative concepts have resulted from the course, then? The course has been organized consecutively every semester (spring, autumn) since fall 2013. Some of the students at the course have successfully completed the course with innovative mobile application concepts including proof-of-concept prototypes. Based on the resulting application concepts, the overall innovativeness and outcome of the student projects has been promising. Some groups (5-10%) fail to reach a proper application concept for the given project in each implementation. The dropout rate is relatively low in this course despite it being very time consuming, especially for second semester students. In the fall 2014 semester, the dropout rate was 7% (Dirin and Alamäki 2015).

The outcomes from the project have been intriguing for the students to such extent that there has been at least one start-up company from each course implementation by students (project team members).

Many innovative application concepts that have been dropped off from further development were mainly due to the lack of motivation or support; these include financial resources. Table 1 and following list briefly describe the most recent concepts developed at the course.

- A Context-Aware Nurse Assistance at Elderly Houses (Dirin et al. 2015): A context-aware mobile web services that provide the basic location based service to assist nurses in their work related activities.
- Adaptive m-learning application for driving license candidates: An adaptive mobile learning application that helps driving license candidates to record and pass the independent driving part with their smartphones. The application delivers the reports to the instructors in driving schools about the route that users drive as well as possible failures etc. during driving.
- Customer Guidance Game (Dirin and Vainio 2015): A mobile game application that helps new visitors to learn the location and offices in the target office premises.
- A Value-Added Mobile Guide Service for Small Tourism Companies (Alamäki and Dirin 2014): A cloud-based mobile service that help kayakers and bikers to navigate.

5 RESULTS: IMPACT OF THE COURSE

In order to evaluate the impact and importance of the content and the course to the students, we conducted an online questionnaire with the class in December

2014, two months after the completion of the course. In addition to this, the online questionnaires were also delivered to the participants of an earlier course implementation that had been implemented in 2013.

Data. Altogether 19 students that had taken the course provided their answers. For the courses completed in 2014, the majority of the students (18 from total of 22 participants) answered the questionnaire and provided their feedback. One student had taken the course already in spring 2013.

Method. The results were gathered using a Google Forms online questionnaire consisting of 19 questions. A link to the questionnaire was sent to students' email. The questions addressed the various phases of the applied UCD framework, methods applied, and overall satisfaction with the framework.

There were two sections in the questionnaire. In the first part we asked students to describe in a qualitative way some the overall feedback about the methodology, what they remember from the methodology, what they considered the drawback of the methodology, and what they would like to improve. The second part of the questionnaire focused on quantitative questions. The low number of responses, however, constrains the analysis of thorough statistical testing. Therefore, the emphasis in this paper is in the qualitative results that are supported with answers to the quantitative part.

Results. Altogether 17 respondents provided their answer to question "What do you remember from the UCD framework methods". Most answers were rather brief, just listing/mentioning the methods by name. Two respondents elaborated the theme slightly more aligning the UCD methods with the stages of the UCD process (requirements gathering with user studies, ideation of features / specification of functionality, prototyping and concept/technical development, testing) including the iterative nature of user-centred development work.

In the free-form answers, most respondents mentioned interviews (N=8/17), prototyping (7), questionnaires (6), scenarios (6), and testing (6) in various ways (as a process stage as well as prototype testing). An educationally much-presented method heuristic evaluation was mentioned a few times (3) as well as transcript coding (3).

However, when asked directly (in the "quantitative questions part", N=19) "Which method you have applied in user study phase", and presented with choices *interview*, *questionnaire*, *diary*, and *shadowing*, almost all students (N=18/19) pointed out interviews and questionnaires as the way to get basic information about their users (see fig. 2). Moreover, slightly over half (N=10/19) of the respondents had additionally used observations to get further knowledge about users. Shadowing was not that commonly applied among respondents (1).

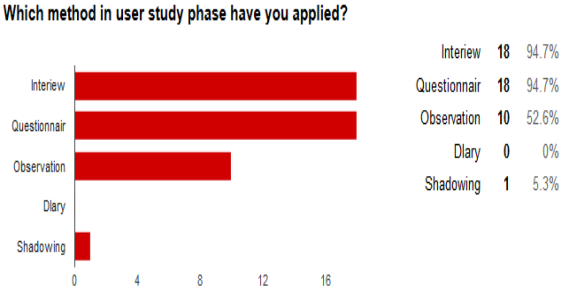

Figure 2: The applied methods by students.

In the free form answers about the methods, transcript coding (i.e. transcribing interviews into text and analysing/coding that data) was mentioned by 3 of 17 respondents. Students in the classroom often present objections for writing the interview transcript as a laborious and time consuming activity. This objection has been heard many times during the course. Therefore, it is interesting to find out the utility and practical applicability of transcript coding in real-life development settings.

Based on respondents' experience in industry, the attitude toward the transcript coding appears to have changed (fig. 3). The majority of the students (N=15/19) find transcript coding as a very useful or useful method for interview data analysis. None of the participants considered transcript coding as "not at all useful" method for their work.

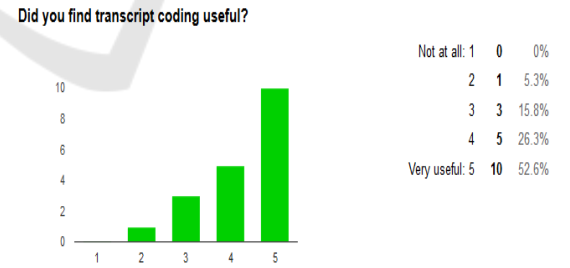

Figure 3: Answers on the usefulness of transcript coding.

Scenarios were the method which students learned to present their mobile application concept to the potential users. Scenarios were among the most mentioned methods in the qualitative answers (6/17) indicating its importance and good applicability in industrial/commercial application and service development. In the quantitative part, majority of the students (11/19) considered scenarios as an excellent design approach (fig. 4).

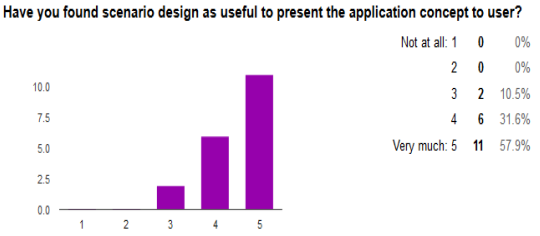

Figure 4: Usefulness of scenarios in design.

After preparing the scenarios, students have been requested to assess their proposed scenario with the potential users. According to the answers, most respondents (15/19, fig. 5) experience that scenario reviews provide much valuable feedback from users. Additionally, course participants have considered that scenario reviews are a good approach to receive most of the users' feedback on the initial application concept.

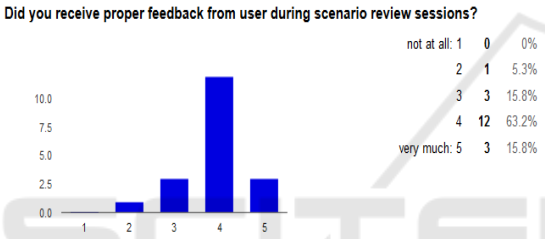

Figure 5: Respondents' answers on user feedback from scenario reviews.

As a final question summarising the industrial and commercial experience that the graduated students had about the UCD framework, a question about the efficiency of the method was presented (fig. 6).

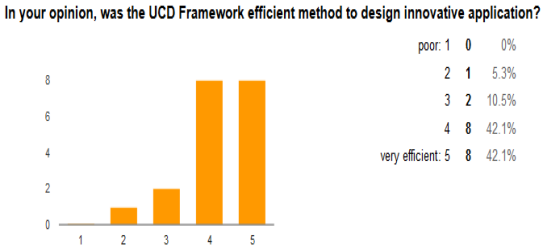

Figure 6: Answers on UCD framework efficiency.

Slightly less than half of the respondents (N=8/19) considered the UCD framework as very efficient (5). A similar amount found the UCD framework as an efficient methodology for designing applications and services. One respondent considered the framework as an inefficient way for doing development work.

Would these students act as advocates of the UCD framework? The Net Promoter Score (NPS) (Kristensen and Eskildsen 2011) can be used to assess how the students value the UCD methods and process after completing the course. The questions in our survey addressing this were "On a scale from 1 to 10 would you apply the same method in other projects" (fig. 7) and "On a scale from 1 to 10 would you recommend the method to other developers" (fig. 9).

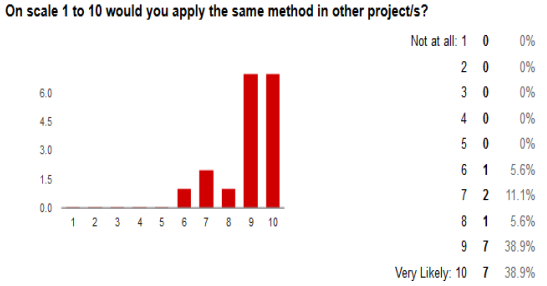

Figure 7: Applying the UCD framework in other projects.

Most students (N=14/19) are expected to apply the UCD methods in their forthcoming projects.

The last numeric question was focused on NPS to confirm whether the students would like to recommend the methodology to others. Majority of students would recommend the method to their colleagues and other developers (fig. 8).

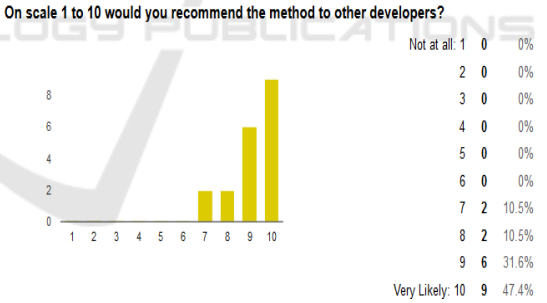

Figure 8: Recommending the UCD framework to other developers.

Students' qualitative feedback on the contents and utility of the framework and the course was mainly positive. The course was considered as "*A great course*" and "*interesting topic*". The open-ended question "What were the most interesting things in the UCD framework" prompted for answers such as: "*Engage users at all UCD phases*", "*Scenario creation and prototype design*", "*Great to conduct a user study, this was my first experience*", and "*The coding of transcripts into quantitative data was surprisingly robust*".

Following issues were raised as challenges, though: "*not very easy to make prototype*", "*choosing the topic and brainstorming"*, "*to get clear meaning from interviewee*", and "*time*". One respondent pointed out that "*I would like to come to watch this course once again after I take requirements analysis course*". Additionally, the contents of the course were considered to require more practical acquaintance: "*UCD Framework was pretty hard to follow as first timer and all the principles and methodologies require practising.*"

6 DISCUSSION

The collected data through our questionnaire study indicates that students are in general satisfied with the outcomes and the competence that they have developed during the User-Centered Design (UCD) course - despite the fact that they considered the development phases as a time consuming efforts. The contents of the course did not receive much comments on question "What was missing in the methodology" ("*I don't know*", "*nothing comes to mind*") even though answers included "*more interviewees*" and "*user experience guidelines*".

Out of the UCD methods, interviews, questionnaires, scenarios, prototyping, and testing were considered of great importance for real-life development activities. These methods do constitute a solid body in the user-centred development process making the contents of the course/framework nicely aligned with the working-life demands. An interesting finding in the results was the changed appreciation of transcript coding, which appears as an "inconvenient" method during the course – but somewhat more appreciated in real-life settings.

The overall positive experience towards the course is well-reflected in both qualitative answers as in numeric responses. Most students appear to act as advocates of the methodology even in their professional surroundings. They consider the framework efficient and worth recommending to their professional peers indicating that there appears a good fit between the contents of the course and the contemporary software and service development.

The experience that we have gained with the implementation of this course is valuable. We expect that if the project topic appeals to students, it motivates them more towards deep learning and appropriate results. Since the results in this paper do not directly enable us to conclude that, we propose this as a future research topic. Possibilities for outreach and spin-offs may provide increased motivation for at least some of the students. We suggest that the influence of motivating characteristics of the final innovative concepts and applications should be taken more carefully into account when evaluating the learning experience.

Despite our attempts toward quantitative results, the number of respondents limits our analysis and conclusions to be based on qualitative results that are supported with the "numerical answers". Statistically significant results are not possible to achieve based on these results. Nevertheless, the qualitative analysis paves the way forward, and as we continue getting complementing answers to our survey, also such analyses are enabled. However, in addition to just getting survey data, interviews would provide even deeper insights about the uses and utility of the user-centred methods and processes for industrial application.

This course had originally been offered to those students in their sixth semester. By sixth semester, students have already developed competence on software engineering related courses such as programming, project management, etc. However, the changes in course curriculum at the Business Information Technology department, resulted that this course is offered to second semester students. Offering the course already at the second semester has introduced some unavoidable challenges including students' lack of knowledge on software development process, software engineering, etc. At this stage of their studies, students often require more explanations as to why this type of course needed in their curriculum. At the same time, commitment to the deadlines to return the assignments, reporting, and the presentation itself has become more challenging to some students. However, when offered at this "early stage" of their studies, the course con be considered as a mindset changer for freshmen.

7 CONCLUSIONS

The main goal of the User Centered Design course is to develop and increase students' capabilities on consulting and user study research. Moreover, the course aims to teach students the development of new application concept. Additionally, the course highlights the importance of users' involvement in software development process at beginning of their studies.

Nowadays, the software development companies involve more often potential users in their requirement elicitation and design phases.

Therefore, the user-centred design course is considered as an appropriate way to educate and prepare students for these demands at the job markets. With the current contents of the course, this goal is well reached. Additionally, the course enables students to select different carrier paths as usability expert or a user study expert. Those students who have had job interviews for developer positions appreciate their competence on usability and user study techniques. Therefore, as the collected data in the questionnaire indicates, students are willing to promote the UCD principles to their friends and colleagues. As an additional motivating factor, the course helps some teams to pursuit commercializing their concepts through a start-up school. So far, the course has been implemented by only one instructor. The instructor is mainly responsible for group coordination and lecturing. The number of students who are currently enrolled are more than 50 students in each semester. For future development and implementation the aim is to recruit previous students' of the course as tutors. Tutors are then responsible to help their dedicated group to achieve better results.

This paper demonstrates that students who successfully pass the course do acknowledge the importance of UCD and user experience related courses. Therefore, the Business Information Technology (BIT) department at the target University of Applied Sciences aims at extending their course curriculum with user-centred design and user experience related topics. These offerings are the response to the software companies' demands on students' increased competence requirements. Education on UCD increases students' competitiveness in the current and future job markets.

Finally, to conclude, our experience in educating students to use user-centred methods in software and service development is perhaps best illustrated with the following quotation from the responses: "*It's real and very important for future career.*"

REFERENCES

Alamäki, A. & Dirin, A., 2014. A. Alamäki and A. Dirin, "Designing Mobile Guide Service for Small Tourism Companies Using User Centered Design Principle,." In *International Conference on Computer Science, Computer Engineering, and Social Media, Thessaloniki, Greece*. pp. 47–58. Available at: http://sdiwc.net/conferences/cscesm2014/.

Bosch, J., 2009. From software product lines to software ecosystems. *Proceedings of the 13th International Software Product Line Conference*, pp.111–119. Available at: http://dl.acm.org/citation.cfm?id=1753235.1753251.

Constantine, L., 2006. Users, Roles, and Personas. In *The Persona Lifecycle*. pp. 498–519.

Cusumano, M.A., 2008. The changing software business: Moving from products to services. *Computer*, 41, pp.20–27.

Dirin, A. & Alamäki, A., 2015. ANALYSING THE IMPLEMENTATION OF TWO DISTINCT PEDAGOGICAL APPROACHES IN HIGHER EDUCATION INSTITUTION. , (inted2015, Madrid, Spain).

Dirin, A. & Nieminen, M., 2014a. Framework for addressing usability and user experience in m-learning. *Journal of Computers*, in press.

Dirin, A. & Nieminen, M., 2014b. Managing m-learning application development : Roles and Responsibilities. In A. Dirin & M. Nieminen, eds. *International Conference on Advanced ICT (ICAICTE-2014)*.

Dirin, A. & Vainio, V., 2015. Case Study : Adapting UCD Framework for Mobile Learning in Customer Guided Game . In *2nd International Conference on Communication and Technology ICCT 2015, 17-18 Jan.* DUBAI, UAE. Available at: http://isro.eu.com/dubai_comp2014/Default.aspx#.

Dirin, M., Dirin, A. & Laine, T.H., 2015. User-Centered Design of a Context-Aware Nurse Assistant (CANA) at Finnish Elderly Houses. In *The 9th International Conference on Ubiquitous Information Management and Communication*. The Mulia, Bali, Indonesia.

Gould, J.D., Boies, S.J. & Ukelson, J., 1997. How to Design Usable Systems. In M. G. Helander, T. K. Landauer, & P. V. Prabhu, eds. *Handbook of Human Computer Interaction*. Elsevier Science, pp. 231–254.

Gould, J.D. & Lewis, C., 1985. Designing for usability: key principles and what designers think. *Communications of the ACM*, 28(3), pp.300–311.

Hackos, J.T. & Redish, J.C., 1998a. *User and Task Analysis for Interface Design*, Available at: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=784573.

Hackos, J.T. & Redish, J.C., 1998b. *User and Task Analysis for Interface Design*, John Wiley & Sons.

Holtzblatt, K., Wendell, J.B. & Wood, S., 2005. Chapter 8 - Building an Affinity Diagram. In *Interactive Technologies*. pp. 159–179. Available at: http://www.sciencedirect.com/science/article/pii/B9780123540515500094.

ISO, 2010. Ergonomics of human-system interaction - Part 210: Human-centred design for interactive systems (ISO 9241-210:2010(E)). *ISO*, pp.1–32.

Kristensen, K. & Eskildsen, J., 2011. Is the Net Promoter Score a reliable performance measure? In *2011 IEEE International Conference on Quality and Reliability, ICQR 2011*. pp. 249–253.

Kristensson, P., Magnusson, P.R. & Matthing, J., 2002. Users as a Hidden Resource for Creativity: Findings from an Experimental Study on User Involvement. *Creativity and Innovation Management*, 11, pp.55–61.

Available at: http://doi.wiley.com/10.1111/1467-8691.00236.

Kujala, S. et al., 2005. The role of user involvement in requirements quality and project success. *13th IEEE International Conference on Requirements Engineering (RE'05)*.

Marvel, 2014. Free mobile & web prototyping for everyone. No coding required. , p.https://marvelapp.com/features/. Available at: https://marvelapp.com/features/.

Nielson, J., 1995. Heuristic Evaluation: How-To: Article by Jakob Nielsen. *Nielson Norman GroupNorman*. Available at: http://www.nngroup.com/articles/how-to-conduct-a-heuristic-evaluation/.

Nieminen, M., 2004. Information Support for User-Oriented Development Organisation. Considerations based on the Construction and Evaluation of Knowledge Storage. *Tesis Doctoral*, p.228.

Preece, J., Rogers, Y. & Sharp, H., 2002. *Interaction Design: Beyond Human-Computer Interaction*, Available at: http://linkinghub.elsevier.com/retrieve/pii/S0010448586800215.

Richardson, I. & Hynes, B., 2008. Entrepreneurship education: towards an industry sector approach. *Education + Training*, 50, pp.188–198.

Rosson, M.B. & Carroll, J.M., 2002. Scenario-Based Design. In *The human-computer interaction handbook: fundamentals, evolving technologies and emerging applications*. pp. 1032–1050.

Ruparelia, N.B., 2010. Software Development Lifecycle Models. *SIGSOFT Softw. Eng. Notes*, 35, pp.8–13. Available at: http://doi.acm.org/10.1145/1764810.1764814.

Saldana, J., 2009. An Introduction to Codes and Coding. In *The coding manual for qualitative researchers*. Los Angeles: Sage Publications, pp. 1–31.

Soitinaho, J. & Palviainen, H., 2015. THE IMPACT OF DIGITAL REVOLUTION ON THE COMPETENCES OF BUSINESS INFORMATION TECHNOLOGY STUDENTS IN. In *9th International Technology, Education and Development Conference*. Madird, 2015, pp. 1–10.

Stone, K.B., 2012. Four decades of lean: a systematic literature review. *International Journal of Lean Six Sigma*, 3, pp.112–132.

Szalvay, V., 2004. An introduction to agile software development. *Danube Technologies*. Available at: http://www.danube.com/docs/Intro_to_Agile.pdf.

Tasker, M. & Packham, D., 1993. Industry and higher education: A question of values. *Studies in Higher Education*, 18, pp.127–136.

Weston, C. et al., 2001. Analyzing Interview Data : The Development and Evolution of a Coding System. *Qualitative Sociology*, 24, pp.381–400.