Get Realistic! - UCD Course Design and Evaluation

Marta Larusdottir, Virpi Roto, Jan Stage, Andres Lucero

Get Realistic! - UCD Course Design and Evaluation

Marta Larusdottir[1], Virpi Roto[2], Jan Stage[3] and Andres Lucero[2]

[1] Reykjavik University, Menntavegur 1, 109 Reykjavik, Iceland
[2] Alto University, P. O. Box 11000, FI-00076 Aalto, Finland
[3] Ålborg University,P. O. Box 159, DK – 9100, Ålborg, Denmark

marta@ru.is; virpi.roto@aalto.fi;
jans@cs.aau.dk; andres.lucero@aalto.fi

Abstract. There is an increasing demand for software, suitable for large segments of users with different needs and competences. User-Centred Design (UCD) methods have been used in the software industry and taught to software developers to meet the various needs of users. The field of UCD covers a broad set of topics that can be covered in a range of courses with various content. In this paper we describe the design of a two-week course focusing on teaching UCD methods to students with various backgrounds that are useful for the students in the future. The course schedule included lectures and workshop activities where the lecturers taught UCD topics and coached the students in developing skills for using the selected UCD methods during the course to design and evaluate an interactive system. Additionally, we describe two types of course evaluations that we conducted: qualitative weekly evaluations and a post-course survey.

The results show that students were in general positive about the course content and the combination of lectures and workshop activities. Hi-fi prototyping was the UCD method that the students rated as being most useful for the course and their future. They particularly liked how realistic these were for the users. The least useful method in the course and in the future was "Walking the Wall", where students read an affinity diagram and make design suggestions. Finally, we suggest changes for a prospective course, based on the results of the evaluations.

Keywords: User Centred Design course, User Centred Design methods, Computer Science curricula, Course design, Course evaluation

1   Introduction

User-Centred Design (UCD) is a rich and varied discipline. The basic aim is to combine design and evaluation in the development of a software system and focus these activities on the prospective users of the system that is being developed. The literature includes extensive research on UCD concepts, principles and methods. One of the classical references provides an overview of the discipline [21]. Other references focus on the principles behind UCD [9], or try to identify how software practitioners define and work with UCD [8].

Teaching UCD is of key importance in order to increase its influence in software development. Software development will not change towards a more user-centred

approach unless there are practitioners available with UCD skills. Nevertheless, the literature on teaching of UCD is very limited. An early workshop aimed to produce a list of skills that are necessary and important for UCD practitioners. They see UCD as a process that should yield a high level of utility and usability by developing good task flows and user interfaces. Therefore, UCD practitioners should have the knowledge, skills, and other characteristics needed for considering and involving users [6].

Only a few authors discuss or present the design of courses on UCD. Seffah and Andreevskaia [23] present the approach behind and the content of a course on human-centred design for university students in computer science, who will be future practitioners in software development. They describe a list with 17 different skills on design and evaluation that should be developed in a UCD course, but do not mention UCD methods or report which ones they taught, if any. They neither outline the contents of a specific course nor any experiences from teaching it.

A stronger focus on teaching of UCD has been present in elementary and secondary school level. In England, for example, teaching design and technology was introduced at that level, and there is considerable documentation of content and experiences from this teaching. Nicholl et al. [27] found that contrary to official directives, there was a clear lack of opportunities for pupils to experience user-centred approaches when undertaking tasks in classes on this topic. Thus there are studies that provide insight into the teachers' teaching practices on user-centred design. In relation to this, there is significant literature on teaching of design and creativity in elementary and secondary school, e.g. Hill [10].

The introduction of UCD in software organizations is the focus of some research literature. This literature typically describes how developers in a software organization was introduced to and trained in UCD methods, e.g.[3,12].

A different stream of research focuses on training of people with mental disabilities to participate in UCD processes. Waller [26] reports from training workshops where users and people who use augmentative and alternative communication were introduced to the UCD process and the related methods. Feedback from participants indicates that they felt more empowered to evaluate systems and to engage in the design of new systems. Prior [22] have adapted UCD methods for a similar purpose.

One of the challenges when teaching a topic like UCD is to assess the quality of the course and its impact on the participants. Some of the aforementioned literature on introducing UCD in software organizations includes such assessment activities. Contrastingly, the limited literature on university level teaching of UCD that is mentioned above, has much less focus on assessment.

This paper reports from an empirical study of a course in UCD for university-level students. We describe and discuss the specific topics within UCD we decided to teach, how we assessed the impact of the course on the participants, and outline a redesign of the course based on the experiences gained. In the following section, we provide a more detailed overview of selected literature on teaching of UCD methods and evaluation of UCD methods used in industry. In section 3, we describe our case which was a two-week university course in UCD. Section 4 presents the method used in our study of the course. In section 5, we provide the results from the study of the course. In section 6

we explain the lessons learned and the reaction to those. In section 7 we discuss the study and provide the conclusion in section 8.

2 Related Work

In this section we will describe some of the literature on how to teach UCD methods, on the evaluation of UCD methods in industry and on the google design sprint process.

2.1 Teaching UCD methods

In this section we study which UCD methods are seen as important to teach, and focus especially on the methods used for design. Our literature review on UCD course design, reported in the previous chapter, showed that UCD courses for university-level students have not reported the set of methods taught on those classes. Therefore, we broadened our scope to cover publications reporting more general courses on HCI rather than just UCD, since UCD methods are under the HCI methods umbrella. In this section we report two works supported by The Special Interest Group on Computer-Human Interaction by the Association for Computing Machinery (ACM SIGCHI). These works have been influenced by a broad range of international experts in the field and are therefore worth a closer analysis.

First, an annual 'Introduction to HCI' course at the CHI conference[1], ACM SIGCHI's premium conference provides an overview to HCI for newcomers in the field, including content on theory, cognition, design, evaluation, and user diversity [17]. The design content on this 4-hour course focuses on user-centered design methods such as surveys, interviews, focus groups, ethnography, and participatory design. Due to the short duration of the course, only the principles of each method are covered, and the course participants are not supposed to gain skills on using these methods.

Second, ACM SIGCHI has conducted an international project 2011-2014 in order to document HCI educators', practitioners', and students' perspectives on the most important topics in HCI [5]. While the project was not focusing on UCD but the broader field of HCI, the authors see human-centeredness in the core of HCI: "HCI focuses on people and technology to drive human-centered technology innovation" (ibid. p. 70). The outcome of this SIGCHI Education Project is a list of HCI topics that were seen as very important or important by the study respondents, and a recommendation to "offer a flexible, global, and frequently refreshed curriculum" (ibid., p. 72). The living curriculum is important because technological advancements constantly bring new topics to be studied.

Since our focus in this paper is on UCD methods, we investigate the methods that were prioritized by all respondents in surveys across the time frame of the SIGCHI Education Project. The important methods risen from the survey are divided into design methods and empirical methods. The design research methodologies considered important or very important are interaction design, interviews, observation, paper/low-

---

[1] https://sigchi.org/conferences/conference-history/chi

fidelity prototyping, prototyping (general), and usability testing. The empirical methods listed were for academic researchers (e.g., problem formation and research design) and not UCD methods to be taught for practitioners.

Churchill et al. [5] report the results of an English-language survey (n=616) in more detail. Interaction design was mentioned as the most important subject (discipline) in HCI, and experience design as the most important topic. Agile/iterative design, experience design, interaction design, and participatory design were rated as the most important design paradigms in HCI. In design methods, for example, field study, interviews, prototyping, usability testing, and wire-framing were rated as very important. Churchill et al. [5] suggest that these results provide a valuable starting point for a unified vision of HCI education. However, we have not found scientific publications reporting students' perspectives on the usefulness of different UCD methods as part of an interaction design project.

2.2   Evaluation of UCD methods used in Industry

In this section we give an overview of some of the current literature on how usability techniques have been integrated in software development in industry.

Bygstad, Ghinea, and Brevik [4] surveyed professionals working at Norwegian IT companies to investigate the relationship between software development methodologies and usability activities. In their findings, there was a gap between intention and reality. The IT professionals expressed interests and concerns about the usability of their products, but they were less willing to spend resources on it in industrial projects with time and cost constraints. The results of their survey also revealed that the IT professionals perceived usability activities and software development methods to be integrated, which the authors believed is a positive sign.

Bark et al. [1] conducted a survey on the usage and usefulness of HCI methods during different development phases. They examined whether the type of the software projects had any effects on HCI practitioners' perception of the usefulness of the methods. The results show that there was fairly little correlation between the frequency of using a particular technique and how useful it was perceived by the HCI practitioners. One conclusion in the study is that HCI practitioners tend to have a personal and overall evaluation of the different techniques rather than evaluating the actual usefulness of the methods in their daily work when developing particular software.

An international web-based survey by Monahan et al. [19] reported the state of using several field study techniques and how effective they were considered to be by usability practitioners in education and industry. The results show that more than half of the respondents rated observations as an extremely effective method and about 40% of the respondents rated user testing as extremely effective. The most influential factor for choosing a method for participants working in the software industry was time constraints.

Venturi, Troost and Jokela [25] investigated the adoption of user centred design (UCD) in software industry. The results of the study show that the most frequently used method was user interviews. Additionally, hi-fi and low-fi prototyping methods were frequently used. Overall, the most frequently used evaluation methods are qualitative,

allowing rapid feedback to the design activities using expert and heuristic evaluation or "quick and dirty" usability test methods. The results also show that UCD methods are typically used during the early phases of the product life cycle.

A survey study on the usage of 25 usability techniques was conducted in Sweden by Gulliksen et al. in 2004 [8]. The results show that the usability techniques that received the highest rating by the usability professionals were those that were informal, involved users and were concerned with design issues. Techniques such as expert-based evaluations and benchmarking that do not involve users, received the lowest ratings by the usability professionals. There was a general agreement among the participants that it is important to integrate usability techniques into the software development process they were using. Some participants mentioned difficulties during the integration, especially those that were using RUP (Rational Unified Process) as their development process.

Another survey study was conducted in Sweden in 2012, where the usage of 13 user centred design methods in agile software projects was studied [13]. The methods used by more than 50% of the participants were: workshops, low-fi prototyping, interviews and meetings with users. The most frequently used methods were: low-fi prototyping, informal evaluation with users and scenarios. These were used at least twice a month. The participants also rated the usefulness of using the methods. More than 90% of the participants rated formal usability methods as "very good" or "fairly good", but that method was typically used twice to six times a year.

2.3   Google Design Sprints

Created as a means to better balance his time on the job and with his family, Jake Knapp optimized the different activities of a design process by improving team processes. Knapp noticed that despite the large piles of sticky notes and the collective excitement generated during team brainstorming workshops, the best ideas were often generated by individuals who had a big challenge and not too much time to work on them. Another key ingredient was to have people involved in a project all working together in a room solving their own part of the problem and ready to answer questions. Combining a focus on individual work, time to prototype, and an inescapable deadline Knapp called these focused design efforts "sprints".

A big important challenge is defined, small teams of about seven people with diverse skills are recruited, and then the right room and materials are found. These teams clear their schedules and move through a focused design process by spending one day at each of its five stages (i.e., map, sketch, decide, prototype, test). On Monday, a map of the problem is made by defining key questions, a long-term goal, and a target, thus building a foundation for the sprint week. On Tuesday, individuals follow a four-ste Created as a means to better balance his time on the job and with his family, Jake Knapp optimized the different activities of a design process by improving team processes. Knapp noticed that despite the large piles of sticky notes and the collective excitement generated during team brainstorming workshops, the best ideas were often generated by individuals who had a big challenge and not too much time to work on them. Another key ingredient was to have people involved in a project all working together in a room solving their own part of the problem and ready to answer questions. Combining a focus on

individual work, time to prototype, and an inescapable deadline Knapp called these focused design efforts "sprints".

The Design Sprint [16] is a process to solve problems and test new ideas by building and testing a prototype in five days. The main premise for Google design sprints is seeing how customers react before committing to building a real product. It is a "smarter, more respectful, and more effective way of solving problems", one that brings the best contributions of everyone on the team by helping them spend their time on what really matters. A series of support materials such as checklists, slide decks, and tools can be found on a dedicated website (https://www.thesprintbook.com).

3 The Case – Experimental Design Course

This paper reports an international course that was planned during early 2017 and executed in July-August 2017 as an intensive course. The Experimental Design Course at Tallinn University, Estonia, lasted for two weeks, Monday to Friday. The main learning objective on the course was ability to apply common user-centred design methods and interaction design tools in practice during a two-week interaction design sprint. A total of 18 international students worked on designing and evaluating a software system, and used altogether 15 UCD methods along the way. The students brainstormed ideas for the systems themselves, so they worked on five different systems, but all used the same methods for analysing, designing and evaluating the systems. The students worked in five groups, with three or four members in each group, which were formed by the lecturers. The strategy while forming the groups was to have varying backgrounds, genders, and nationalities in each group. The course schedule is illustrated in Table 1.

Table 1. The schedule of the course for the two weeks.

Week 1	Monday	Tuesday	Wednesday	Thursday	Friday
Morning *Hands-on work in italics*	Introduction	Contextual interviews, Introduction to UX	Introduction to Interaction design, *Affinity diagram*	*Scenario, LoFi prototyping*	*UX goals, SUS, Preparing presentations*
Afternoon *Hands-on work in italics*	*Visioning workshop*	*Contextual interviews, Affinity diagram*	*Personas, Scenarios. Walking the wall*	*LoFi prototyping, Heuristic evaluation*	Interim presentations
Week 2	Monday	Tuesday	Wednesday	Thursday	Friday
Morning lecture	Usability evaluation	*HiFi prototyping*	*HiFi prototyping*	UX evaluation, *HiFi prototyping*	*UX evaluation, Preparing presentations*
Afternoon workshop	*Usability evaluation*	*Usability evaluation, Redesigning*	*Usability evaluation, Redesigning*	*HiFi prototyping*	Final presentations

The students generally had lectures and group work sessions from nine in the morning until around four in the afternoon, see Table 1 for the schedule. During the first three days the students were introduced to the following user centred design methods: Visioning, contextual interviews [11], affinity diagram [18] or KJ method [15], walking the wall, personas and scenarios [11]. After an introduction of the method the students used each of the methods with supervision from the lecturer. The next two days the students were introduced to user experience (UX) goals [14], they made low fidelity prototypes on paper of the user interface and evaluated those through heuristic evaluations [20]. They also used the System Usability Scale (SUS) questionnaire for evaluation [2] and additionally evaluated the interface according to their UX goals. During the second week the students were introduced to formal usability evaluations. They prototyped the interface using the Just-in mind prototyping tool (https://www.justinmind.com/) and did an informal think aloud evaluation on that prototype. After redesigning the prototype, the students stated measurable usability and UX goals and made a summative user evaluation to check the measurable goals. At the end of the course all the students presented their work to each other with a 15 minutes presentation to the class. In Table 1 the course schedule can be seen. We chose the methods introduced to the students partly on results on what methods IT professionals rate as good methods for UCD [13].

The students were asked to develop some software for international students in a foreign country, but otherwise they could choose the domain for their application. Two groups wanted to assist students find courses, one focused on choosing courses at a particular university but the other group focused on courses within a subject of interest between universities. Two groups chose to assist international students in food related issues. One group wanted to assist in buying food in a foreign country and the other in hearing about and learning recipes from local and international students.

Figure 1 shows illustrations of the paper based prototype of the first four screens the users meet from the student group assisting in buying food at the grocery store.
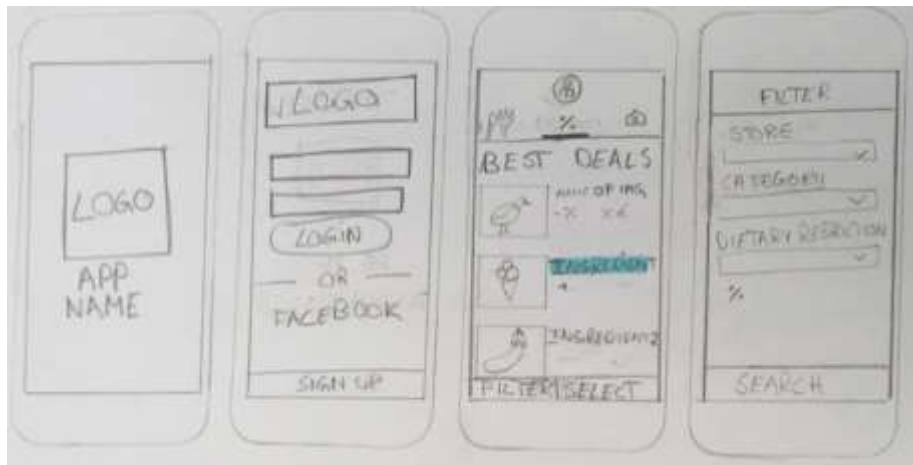

Figure 1: Illustrations of the low-fi prototypes from one of the groups.

The focus in the student projects was chosen because it was easy for the students to imagine how the user groups are, since they were themselves in the same postion. Additionally, we chose this because they students were asked to collaborate between groups during user testing, acting as users for another group and getting students as users to their user testing. With this approach we ensured that the students would find participants for their user testing that were representative for the user groups.

On figure 2, there are illustrations of hi-fi prototypes of the best deals screen before and after the formal user testing session.
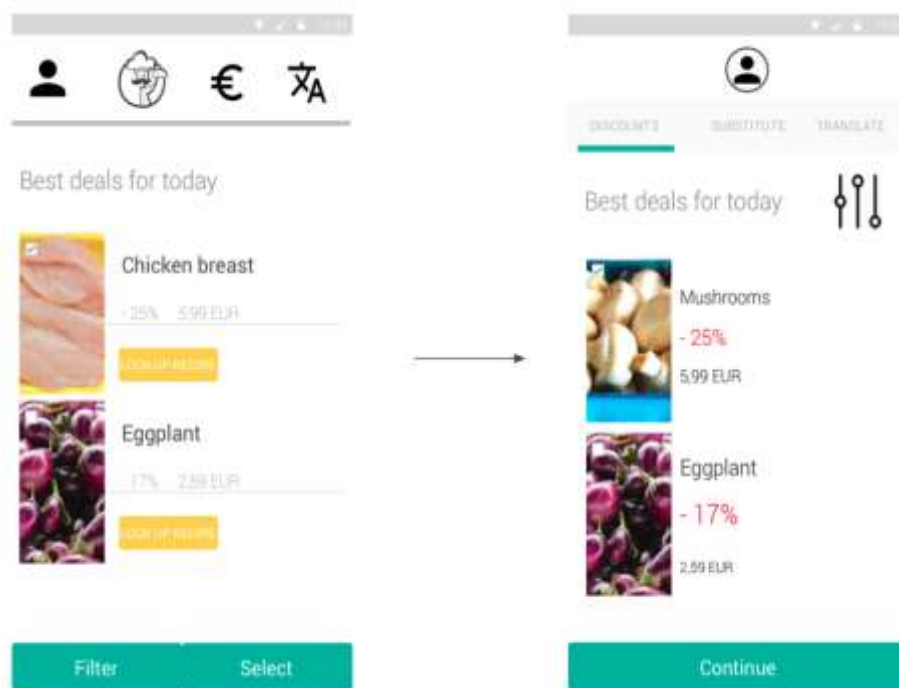

Figure 2: Hi-fi prototypes of the „Best deals" screen before and after user testing.

The hi-fi prototypes were clickable and the students had made one path through the prototype for the user testing. There was not database designed and no data inserted, so the user in the user testing could only chose the data that was „hardwired" in the prototype.

4 Method

This section explains the background of the students and the data gathering methods.

4.1 Background of the Students

The students had various backgrounds, both concerning nationality, gender, age and education. The 18 students were from 11 countries: 8 students from the Nordic countries, 7 from other European countries, 1 from Canada, and 2 from Asia. We had 10 female students and 8 male. The age range was from 22 to 42 years. We had five students with a high school degree, studying for the Bachelor degree, 10 students having a Bachelor degree, out of which seven were studying for their Master degree, and three with a Master degree, where one was studying at a PhD level. The fields of their study was: Computer Science (four students), Collaborative and industrial design (three students), Interaction design (two students), and one student in each of the fields: Informatics, human centred technology, international design business management and software engineering.

Seven students had been working at software companies and the time span was from three months up to more than five years. The jobs roles included: Advisor/mentor, analyst, consultant, lead UI/visual designer, service design intern, test developer, UX designer, UX manager and UX researcher. Some students mentioned more than one job role they had had. We did not ask particularly about, if they had taken similar courses previously, but in the informal discussions with students we found that some of them had quite good knowledge and skills in the subjects of the course, while others had not seen similar material before.

Some examples of the types of software the students had been developing in industry include: Online systems for music learning; survey tool based on a map, a software system to support the diagnosis protocol of children with ADHD (Attention Deficit Hyperactivity Disorder), websites where one can order food from shop with same day delivery, software for supporting weather measurements and software for issuing credit cards and handling money transfer.

4.2 Data Gathering Methods

Two methods were used to gather feedback from students on their opinions on the course: a weekly evaluation form for collecting open-ended feedback, and a questionnaire on the UCD methods taught on the course. Both were distributed on paper at the end of the week. The methods will be described in more detail below.

The Weekly Evaluation. Students were asked to draw their right hand on an empty A4 sheet of paper as the last thing in the afternoon during both Fridays, so data was gathered twice with this method during the course. In the space of the thumb, they were asked to write what they thought was good during the current week, in the space for the index finger, things they wanted to point out, in the third finger space what was not good, in the space for the fourth finger what they will take home with them and the fifth

finger what they wanted more of. The students wrote sentences, so this was a qualitative method. The students handed in their evaluation by putting it in a box that was placed at the back of the room, so the lecturers did not watch who was delivering in the evaluation forms, to keep the anonymity. When all the students had handed in their evaluations, we asked if there was something that they wanted to share with the group. There were open discussions for about 15 minutes of improvements that could be made to the course.

When we had gathered the evaluations after the first week, one of the lecturers inserted all the answers from the students to a Google spreadsheet to be able to share those with the other lecturers. As this was meant to be a formative evaluation method, the results were discussed and some changes were made to the schedule according to the feedback. The results were analyzed with thematic analysis [7].

The Method Questionnaire. The questionnaire was on paper and contained 3 pages. On the first page there were: 4 questions on the student's background, 3 questions on their currently highest achieved degree, one question on whether they were studying currently or not, and 4 on their current education (if applicable). Also, on the first page they were asked if they had worked in the industry. If so, they were asked to fill in 5 more questions about the work role and company.

On the second page of the questionnaire the students were asked to rate their opinion of the UCD methods used on the course. For each method they were asked to rate:
a) If the method was thought provoking;
b) if the method was useful for the course; and
c) if they thought that the method would be useful for their future job/education.

For each item we provided a 7-point scale from 1 = not at all to 7 = extremely so. The 15 methods they evaluated were: Visioning, Contextual interviews, Affinity diagram, Walking the wall, Personas, Scenarios, UX goals, Low fidelity prototypes (paper prototypes), Heuristic evaluation, Evaluation of lo-fi prototypes with user using the SUS questions, Evaluation of lo-fi prototypes with user using the UX goals, Hi-fi prototyping (digital), Think-aloud evaluations of hi-fi prototypes, Measurable usability goals setting and Summative evaluation (usability and user experience).

On the third page, there was just one open question for any other comments that they would like to share with us. They had a full A4 page to freely share their comments.

The questionnaire was filled in right after the retrospective hand evaluation during the last session of the class. The students typically used 20 minutes to fill in the questionnaire. When all the students had filled in this questionnaire, discussion was facilitated on the overall evaluation on the course.

5   Results

In this section the results both from the weekly evaluations and the Methods questionnaire will be described.

5.1 Results from the Weekly Evaluation

The students wrote one or two sentences of each of the five categories: What was good; what they wanted to point out; what was not as good what they would take home with them and what they wanted more off. When all the students had handed in their evaluations, we asked if there was something that they wanted to share with the group. There were open discussions for about 15 minutes of improvements that could be made to the course. Some of the comments were quite detailed, e.g., 2 students commenting on one particular lecture by the Chilean guy, and a particular subject (the list of the emotions for setting UX goals) and some comments are more general, e.g., one student saying that the lectures were interesting. Four students commented on the group work projects in some way and four on meeting new people and their openness. Six commented on the teachers in some way and three on the lectures and their subject. We got four comments on having different teachers during the course and one particularly commented that it was good to have both group work and lectures the same day.

5.2 Results from the Methods Questionnaire

Results from the methods questionnaire were calculated and analyzed and are presented in table 2. We calculated the average grade for the 18 students. The scale used was 1 to 7, where 1 = not at all to 7 = extremely so. We have marked the methods getting the highest grading and the lowest grading with colors.

Table 2. Results from the Methods questionnaire

UCD methods	Thought provoking	Useful in the course	Useful in the future
Visioning	4,6	5,4	5,1
Contextual inquiry	5,1	5,7	5,7
Affinity diagram	4,5	4,9	4,8
Walk the Wall	4,0	3,5	4,0
Personas	4,2	4,7	5,1
Scenarios	4,1	4,8	5,1
UX goal setting	5,7	6,2	6,3
Lo-fi prototyping	4,9	5,9	5,5
Heuristic Evaluation	4,4	4,6	5,1
Evaluation of lo-fi prototypes - SUS	4,6	5,3	5,3
Evaluation of lo-fi prototypes - UX	5,0	5,4	5,1
Hi-fi prototyping (digital)	5,8	6,8	6,8
Think-aloud evaluation	5,6	6,2	6,2
Measureable usability goal setting	5,4	6,1	6,2
Summative evaluation	5,7	6,3	6,4

The method getting the highest score in all phases (the immediate, the course period and for the future use) was Hi-fi prototyping, where the students made prototypes using the Just-in mind tool. This is when the ideas really came to life and we could see that they students loved this activity. They even stayed up until after midnight to make the prototypes work and look good. The second highest for the course was Summative evaluation, where the students measured effectiveness, efficiency, satisfaction and UX factors. UX goals setting and Think aloud evaluation had the same score as the third and fourth best method for the course.

The students gave the method: Walking the wall the lowest grading. During walking of the wall, the students got visitors from other groups that should suggest design ideas for the group while reading an affinity diagram showing results from interviews with users. A paired-samples t-test was conducted to compare the difference of the ratings of the walking the wall method and the hi-fi prototyping. There is a statistical difference in the grading of walking the wall (M=3.53, SD=1.94) and hi-fi prototyping (M=6.78, SD=0.55) with (p=0.01). One student mentioned that this activity was rather useless since the students did not use the results of this activity in the next one. The second lowest as useful for the course was the heuristic evaluation. This could be because, the students did two evaluations one after the other without redesigning their designs between the evaluations. The third lowest in the grading for usefulness in the course was the method Personas. This could also be because the Personas where not really used again during other activities.

The open comments students gave overall about the course were quite positive. The background of the students varied, from being studying on BS level to a PhD level, and some students had been and were working in industry. The negative comments show clearly that the course was too basic for some students. Most of the comments could be met by giving more clear description about the course, like timings and objectives of the course should be clear in the course description.

6   Lessons Learned and Reactions

In this section we will describe the lessons learned based on the results from the evaluations from students and the evaluations from the lecturers present at the course. We will first give a summary of the lessons learned and then present the changes made to the course schedule as a reaction to the lessons learned.

6.1   Lessons Learned

In summary, we have described above how we successfully taught a course on UCD for university-level students. We have also presented the specific topics within UCD that were included in the course. In addition, we have described how we assessed the impact of the course on the participants and outlined a redesign of the course based on the experiences gained.

Two methods were used to evaluate the course by gathering feedback from students on their opinions about the course and the methods taught: an open-ended evaluation

form for collecting qualitative feedback, and a questionnaire. Our results demonstrate the general usefulness of these techniques for course evaluation. It was valuable to get feedback from the students after the first week and discuss with them what adjustments could be made for the second week. Some comments were on the dissemination of the course so some students thought that the course description, so we could explain why this happened. Additionally, we chose sometimes to give the students more time during the workshops, if they did not manage to finish using a method within the timeframe scheduled. We thought they would appreciate not strictly following the plan, but some students commented that we did not follow the announced course schedule in details and found it stressful. Also we had not clearly stated that students should be there until 16, so some students were upset that this was not made clear.

The students were generally positive about the course content, and they particularly liked the combination of lectures and workshop activities every day, and the involvement of several teachers. Altogether, they used 15 UCD methods during the course. The first observation from the results is that four UCD methods were rated highest: hi-fi prototyping, because the prototypes were very realistic for the users; summative evaluation, where they measured effectiveness, efficiency, satisfaction and UX factors; setting UX goals and think-aloud evaluation. It was also obvious observing the students that when they got to do the running prototypes they stayed until late at night. That was the only day during the two weeks that they got so focused in using the method we had introduced to them. The UCD methods that the students rated lowest were "Walking the Wall", where students should read an affinity diagram and suggest design ideas. They commented it took long time to use the method and the output was not that valuable. The students also gave heuristic evaluation and personas low rating. We had indications that the reasons for these low ratings were that the results were either not used in subsequent activities or not used at all.

The students used the UCD method one after the other and the previously used should feed information to the next method. The students found it hard to see the relevance in walking the wall and making personas and found these methods a bit disconnected to the remaining methods. On the other hand, they saw the relevance in conducting user testing and iterating the prototypes accordingly. They did three evaluations and iterated the prototypes afterwards, first an informal evaluation of the paper prototypes, then evaluating the UX goals and the third was user testing. This was mostly scheduled during the second week. Some students commented that they would have liked this to be earlier in the course. These comments inspired us to change the course schedule accordingly for the next round of the course.

The latter observation demonstrates that teaching of a method without relevant application of it is very difficult. Rather than being a problem with the method itself, this is a reminder for us educators of the importance of planning teaching activities in a way that students can hopefully understand their logic and make direct connections with their ongoing learning process. Many students commented that they would have wanted to be in more direct contact with users and not only meet the students taking part in the course. They really stressed that they wanted the user testing to be as realistic as possible.

6.2   Reaction to Lessons Learned

Since the students rated realistic prototypes highly and user evaluations, but commented that these methods could have been introduced earlier in the course, we have decided to teach these methods during the first week of the course during the summer 2018. This schedule fits very well to the Google design sprint schedule, where the fourth day focuses on making realistic prototypes and the fifth day focuses on conducting user evaluations with 5 users. We will base the course schedule on the Google design sprint schedule.

During the first week we will follow the Google design sprint schedule completely, like explained in section 2.3. During the second week we will cover user experience in more detail. The students set UX goals and evaluate against those goals. They redesign the hi-fi prototypes and evaluate once more to gather feedback on the user experience. Overall, the focus during the first week is on designing the right product, and during the second one the focus is on designing the product right.

We will choose methods from the Google design sprint process to get the overall idea about the product, setting the stage, setting a long term goal, mapping, sketching, speed critique and storyboarding. These will be the new methods that we will cover. Some are similar to the once given in the course 2017, like sketching is similar to low-fi prototyping, but is now done in context with other methods that are used in advance in the course. We will not include the contextual interviews, the scenarios, the affinity diagram and the walking the wall methods. Also heuristic evaluation will not be covered. These methods were not highly rated by the students and we believe the methods of Google design sprints will work better for the course.

7   Discussion

In relation to hi-fi prototyping, students seemed to particularly like this method, especially how realistic the resulting prototypes were for the users. This finding is in line with the ideas behind the Google Design Sprint [16] with its main premise of seeing how customers react before committing to building a real product. This finding also seems to suggest that for similar one-to-two week student projects, something like Google Design Sprints can help students focus by providing a big challenge, time restraints, and bringing the best individual contributions of everyone on the team.

While the context, respondents, and the questionnaire used in this study were quite different from those used by Churchill et al. [5], we briefly discuss one difference between the results of these two studies. In our study, hi-fi prototyping was rated highly, while in [5], hi-fi prototyping was not even mentioned, although paper/low fidelity prototyping and prototyping (general) were seen highly important. The reason might be that the students enjoy hi-fi prototyping with the modern prototyping tools and see paper prototyping too cumbersome or old-fashioned. Future work is needed to properly study the usefulness of teaching lo- vs. hi-fi prototyping for students.

In relation to software development practice, it is interesting to compare the UCD method ratings made by the students to the frequency of use of UCD methods reported by practitioners. Venturi et al. [25] found that hi-fi prototyping and heuristic evaluation

was used frequently, setting quantitative usability goals less frequently, and quantitative usability evaluation (summative) and Personas least frequently. So except for heuristic evaluation, there is good correspondence. This shows that it is possible to effectively teach several of the methods that are most relevant in practice.

8   Conclusion

This paper reports from an empirical study of a two week intense course in UCD for 18 international university-level students. The schedule was switching between lectures and workshops using 15 UCD methods. We used two methods to evaluate the course content and schedule, a weekly questionnaire and a method questionnaire measuring the usefulness of the methods. Both course evaluation methods gave good insights into how to adjust the course for the next occasion. Since the students rated hi-fi prototypes and formal user evaluations highly and commented that these methods were used quite late in the course schedule, we have decided to follow the Google design sprint process during the next occasion of the course. Then hi-fi prototypes will be conducted during the fourth day of the course and formal user evaluations on the fifth day. This will give the students the possibility to see some realistic output earlier in the course, which seems to be appreciated by the students.

References

1. Bark, I., Følstad, A., Gulliksen, J., McEwan, T. and Benyon, D.: Use and usefulness of HCI methods: Results from an exploratory study among nordic HCI practitioners, In: People and Computers XIX - The bigger picture. London: Springer, 201-217 (2006)
2. Brooke, J., SUS: A quick and dirty usability scale. In P.W. Jordan, B. Thomas, B.A. Weerdmeester & I.L. McClelland (Eds.), Usability Evaluation in Industry (pp. 189-194). London, UK: Taylor & Francis, (1996).
3. Bruun, A. and Stage, J. (2012) Training software development practitioners in usability testing: an assessment acceptance and prioritization. In Proceedings of the 24th Australian Computer-Human Interaction Conference (OzCHI '12). ACM, New York, 52-60.
4. Bygstad, B., Ghinea, G. & Brevik, E.: Software development methods and usability: Perspectives from a survey in the software industry in Norway. Interacting with Computers, 20 (3), 375-385 (2008)
5. Churchill, E. F., Bowser, A., & Preece, J. (2016). The future of HCI education: a flexible, global, living curriculum. interactions, 23(2), 70-73.
6. Dayton, T. (1993) Skills needed by user-centered design practitioners in real software development environments: Report on the CHI92 workshop. SIGCHI Bulletin 25(3): 16-31.
7. Ezzy, D., Qualitative analysis: Practice and innovation. Psychology Press (2002).
8. Gulliksen, J., Boivie, I., Persson, J., Hektor, A. & Herulf, L.: Making a Difference: A Survey of The Usability Profession in Sweden. In: Proceedings of NordiCHI 2004, Tampere, Finland 207-215 (2004)
9. Gulliksen, J., Göransson, B., Boivie, I., Blomkvist, S., Persson, J. and Cajander, Å. (2003) Key principles for user-centred systems design. Behaviour and Information Technology 22(6):397-409.

16

10. Hill, A. (1998). Problem solving in real-life contexts: An alternative for design in technology education. International Journal of Technology and Design Education, 8, 203–220.

11. Holtzblatt, K., Wendell, J. B., & Wood, S.. Rapid contextual design: a how-to guide to key techniques for user-centered design. Elsevier, (2004).

12. Häkli, A. (2005) Introducing User-Centred Design in a Small-Size Software Development Organization. Master thesis, Helsinki University of Technology, Department of Computer Science and Engineering.

13. Jia, Y., Larusdottir, M.K., and Cajander, A., The usage of usability techniques in Scrum projects. In: M. Winckler, P. Forbrig, R. Bernhaupt, eds. Proceedings of the HCSE 2012. LNCS, vol. 7623. Heidelberg: Springer, 331–341, (2012).

14. Kaasinen, E., Roto, V., Hakulinen, J., Heimonen, T., Jokinen, J. P., Karvonen, H., ... & Tokkonen, H., Defining user experience goals to guide the design of industrial systems. Behaviour & Information Technology, 34(10), 976-991 (2015).

15. Kawakita, Jiro. The original KJ method. Tokyo: Kawakita Research Institute (1991).

16. Knapp, J., Zeratsky, J., & Kowitz, B. Sprint: How to solve big problems and test new ideas in just five days. Simon and Schuster (2016).

17. Lazar, J. K., & Barbosa, S. D. (2018). Introduction to human-computer interaction. In Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems (p. C03). ACM.

18. Lucero, A. Using affinity diagrams to evaluate interactive prototypes. In Human-Computer Interaction, 231-248. Springer, Cham (2015).

19. Monahan, K., Lahteenmaki, M., Mcdonald, S. & Cockton, G.: An Investigation into the use of Field Methods in the Design and Evaluation of Interactive Systems. In: Proceedings of the 22nd British HCI Group Annual Conference, People and Computers, 99-108 (2008)

20. Nielsen, J., Molich, R. Heuristic evaluation of user interfaces. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (pp. 249-256). ACM. (2016)

21. Norman, D. A. and Draper, S. W., Eds. (1986) User Centered System Design: New Perspectives on Human-Computer Interaction. Hillsdale, NJ: Erlbaum.

22. Prior, S. (2010) HCI methods for including adults with disabilities in the design of CHAMPION. CHI '10 Extended Abstracts on Human Factors in Computing Systems (CHI EA '10). ACM, New York, 2891-2894.

23. Schwaber, K.: Scrum development process. OOPSLA'95 Workshop on Business Object Design and Implementation (1995)

24. Seffah, A., Andreevskaia, A. Empowering software engineers in human-centered design. In *Proceedings of the 25th International conference on Software engineering* (pp. 653-658). IEEE Computer Society (2003).

25. Venturi, T., Troost, J., & Jokela, T.: People, organizations, and processes: An Inquiry into The Adoption of User-Centered Design in Industry. International Journal of Human-Computer Interaction, 21(2), 219-238. (2006)

26. Waller, A., Balandin, S. A., O'Mara, D. A. and Judson, A. D. (2005) Training AAC users in user-centred design. In Proceedings of the 2005 international conference on Accessible Design in the Digital World (Accessible Design'05). BCS Learning & Development.

27. Nicholl, B., Hosking, I. M., Elton, E. M., Lee, Y., Bell, J., & Clarkson, P. J. Inclusive design in the Key Stage 3 classroom: An investigation of teachers' understanding and implementation of user-centred design principles in design and technology. *International Journal of Technology and Design Education*, *23*(4), 921-938 (2013).