ESSAY

Various Aspects to Consider in Choosing User-Centered Methods

Viivi Uhari

CS-E4900 - User-Centered Methods for Product and Service Design in Aalto University

ARTICLE HISTORY
Compiled March 5, 2026

ABSTRACT
This essay looks at the benefits and deficits of different user-centered methods, and their applicability. This is done by first introducing user-centered activities and discussing the importance of using multiple methods in a flexible manner. After that, different methods are introduced and discussed by walking through some of the user-centered activities, each having a section of their own. Finally, the contents of this essay are summarized.

1. Introduction

Human-centered design (HCD) provides a human-centered perspective to design and development processes (International Organization for Standardization [ISO], 2019). The ISO standard (2019) also defines the principals a human-centered approach should follow and its four main activities. The standard uses the term human-centered design, but its information still applies to user-centered design (UCD) which is the term used in this essay. The first activity in UCD is to specify and understand the context-of-use of the designed system which includes the users, their goals and tasks and the system's environment. The next activity is to specify the user requirements which are gathered together into a coherent set that will provide the basis for the design and evaluation of the system. The third activity defined by the standard is to produce design solutions, and the last step is to evaluate them. Since the UCD process is a multistage process, multiple methods have been designed to execute its activities, and as ISO (2019) defines, a step in the planning of a UCD process is to identify appropriate methods for these activities.

Because of this large number of methods, there is a broad selection to choose from. Using multiple methods to execute an activity of the design process is even encouraged. Wilson (2006) explains some of the benefits of triangulation which refers to the use of multiple methods in data collection and analysis. Multiple methods in this case "can literally mean different methods or it can mean using different variations of the same method" (Wilson, 2006, p. 46). The benefits include finding core problems with a system, producing more convincing data and reducing inappropriate uncertainty. For example, I have used user stories, use case templates and preliminary personas in defining possible features of a software, which have made the design document of the software more convincing. I have also conducted usability tests with various users to identify core problems with a prototype.

Since there are numerous methods to choose from, a designer should be flexible

in choosing the appropriate ones for their purpose. A designer should not fan over only one method (Wilson, 2006). Roto, Lee, Law and Zimmerman (2021) also report that the practitioners of Service Design (SD) and User Experience Design (UXD) are flexible in choosing their methods. Roto et al. (2021) analyzed the perceptions of the differences of SD and UXD, by asking about the characteristics of these fields from their researchers and practitioners. The respondents frequently commented that "methods [are] not boxed to a practice" and that the scope of the designers' work is in reality defined by their contextual needs (Roto et al., 2021, p. 1921). Most of the respondents also viewed that the practices' methods could also complement each other. As an example, Ismirle (2018) considers user experience mapping as a method of UXD but as an extension of customer journey modeling which Roto et al. (2021) note to be a method of SD. Hence, a designer should be openminded in deciding on what methods to use.

In addition to thinking how well a method could address the problem at hand, a designer should consider their required resources (time, money, manpower). Especially "due to increasingly short product realization cycles" there is a need for fast designing processes (Millen, 2000, p. 280). Not all organizations will have enough experience or additional resources either to execute some of the more advanced methods. A designer could also consider whether the methods are engaging of the user or designer as according to ISO (2019), a UCD process should involve the user when possible. One final aspect to consider could be that whether the methods can be adapted into an online environment since the popularity of remote work has been on the rise.

2. User-Centered Methods in Gathering Information about the Context-of-Use

One of the first activities in a UCD process is gathering information about the context-of-use. Since ISO (2019, p. 6) states that the design should be "based upon an explicit understanding of users, tasks, and environments", it is important to get valuable information during this step. First, a simple method would be to use questionnaires. They can involve users in the design process for example by asking about their problems with the current systems, but getting a clear picture of the users' tasks or goals with just predefined questions can be challenging. It might be difficult to construct correct questions and with questionnaires, the designer cannot ask follow-up questions either. Boynton and Greenhalgh (2004) provide information about using questionnaires as a research method, and one deficit of questionnaires that they point out, is that different participants may understand questions differently if constructed poorly. Open-ended questions should also be used instead of close-ended ones since with closed ones the "richness of potential responses is lower" (Boynton & Greenhalgh, 2004, p. 1314). With open-ended questions users can explain their views, which is important as the problems in UCD might not be well defined.

Another more interactive method for understanding the context-of-use, is qualitative interviewing. DiCicco-Bloom and Crabtree (2006) talk about unstructured and semi-structured interviews, which can also be used for user-centered research. The authors explain, that in unstructured interviews, the interviewer observes the actions of participants (in our case users) taking notes at the same time and asking questions from their key informants. DiCicco-Bloom and Crabtree (2006) also refer to this as participant observation, which Millen (2000) explains to be one way to maximize the learnings from the field as observation riches the understandings through personal

experience.

Semi-structured interviews on the other hand are usually scheduled, there is no observation, and some preliminary questions are written down for the interview (DiCicco-Bloom & Crabtree, 2006). The interviewer may also ask follow-up questions that arise during the interview (DiCicco-Bloom & Crabtree, 2006). These methods allow the interviewer to ask more relevant questions, which can help gather more valuable information about the context-of-use. The involvement of a user increases even more in individual in-depth interviews as the phases of the interview deepen allowing a space for the interviewee to ask questions (DiCicco-Bloom & Crabtree, 2006). Another benefit of qualitative interviews is to "explore meaning and perceptions to gain a better understanding" (DiCicco-Bloom & Crabtree, 2006, p. 314). This appeared true to me as well during one of my projects in user-centered product design when we gathered data about our possible users with interviews. This was a good approach to our problem since we did not have sufficient background information to construct questionnaires.

If a designer does not want to dive deeply into the goals and task of an individual, they can also conduct semi-structured group interviews (DiCicco-Bloom & Crabtree, 2006). This can help better understand the requirements of multiple users and user groups. DiCicco-Bloom and Crabtree (2006) also say that group interviews usually take the form of focus groups, and that the groups represent entities within a set of groups. The groups could for example represent different user groups in UCD.

Another method combining observation and questioning, is contextual inquiry. Holtzblatt and Beyer (1993) use this as one of their methods in their detailed process on customer-centered design. The authors explain, that with contextual inquiry, a designer meets a user in the user's work context, observes their work, the user narrates what is being done and the designer may ask questions from the user. This reminds me of unstructured interviews. As the user can answer questions about the work at hand, the designer might get more valuable insights than with just questionnaires. A user can even forget about some underlying problems of the current system which contextual inquiry can help bring to light (Holtzblatt & Beyer, 1993).

One final method of gathering information worth mentioning here is the use of cultural probes. Gaver, Dunne and Pacenti (1999) used these informal personalized probes to gather information about their users in various different environments. The idea of Gaver et al. (1999) was to gather information in a novel and interactive manner by distributing different interactive objects to their users. They had the users interact with the probes and send them back. With this method, follow-up questions are not possible, but the scope of possible answers is broad. This method might be suitable for UCD as the goal can be to get answers to undefined questions. Gaver et al. (1999) reported that the probes provided valuable insights and detailed information about the sites which allowed the authors to shape their designs to fit them.

Required Resources These methods require different amounts of recourses. Creating and delivering questionnaires might not require a lot of effort. Questionnaires would require even less effort if ready-made ones would be used (Boynton & Greenhalgh, 2004). However, this might not work so well in the field of UCD as most of the problems are project-bound. Interviewing takes up time from the interviewer and interviewee, but luckily group interviews can reduce both amounts. Contextual inquiries can be more efficient than interviews since Holtzblatt and Beyer (1993, p. 94) explain that the method has been used to "optimize both development time and the customers' own time". During contextual inquiries, the users can carry on with their work with some interrupts from the researcher, which the user might find less annoy-

ing than just answering questions. The overall time that goes into gathering data via interviews maybe even shorter than what goas into questionnaires because another set of questions might be required to ask follow-up questions. Regarding cultural probes, time goes into gathering background information about the users and creating the probes. If use cultural probes were to be used in the same way as Gaver et al. (1999) did, the time this method takes would be dependent on the users on how long it takes for them to provide answers. Time can also go into refining the probes.

Rapid ethnography provides some solutions for reducing the amount of time needed for field research which Millen (2000) introduces as a collection of field research methods intended for gathering and analyzing information in a limited time frame. He introduces three ways of speeding up traditional ethnography: making the most of the time in the field, narrowing down the focus and using collaborative and computerized data analysis. Millen (2000) says that using a more interactive research approach in the field can help make the most of it, and as it so happens, structured interviews, participant observation and contextual inquiries are already some examples of it. According to Millen (2000), a designer should narrow their focus sufficiently before starting field work. I used interviewing techniques in defining the scope of a software project I was working on and as the project proceeded, we were able to focus our interviews with users on specific features which helped to get a clearer understanding of their needs.

Require Experience Questionnaires do not require a lot of experience to develop but the researcher still needs to be careful in constructing them, so that they are valid and reliable (Boynton & Greenhalgh, 2004). Interviewing does not require a lot of experience either, but as DiCicco-Bloom and Crabtree (2006) point out, the interviewer should be ready to depart from the planned structure which might require some skills. I conducted interviews in pairs for a project on user-centered product design, which helped with our inexperience with interviewing since both interviewers could chime in resulting in various follow-up questions. Finally, the use of cultural probes might require quite the most experience as Gaver et al. (1999) had artis-designer backgrounds. The designer needs to think about what and how many probes to use, what should they look like, and how should the users interact with them.

Reach of Users Another aspect to consider with these methods, is their reach. Questionnaires can be used to reach a lot of users around the world in a small amount of time, but interviewers might reach only a few people at a time. As a solution for this, Millen (2000) suggest having multiple designers in the field at once, so that the designers can focus on different things simultaneously. During my project on user-centered product design, we interviewed our users which appeared to be the right approach since we did not have enough subjects in mind so that the larger reach of questionnaires would have been beneficial. In addition, having multiple people conducting interviews in parallel helped us reduce time. According to Millen (2000), getting multiple views from the field can also help better understand the situation, which was true for us as well. So, even though the resources saved by reducing the time of one designer are invested in multiple designers, I would argue that having a team of multiple users is overall a greater benefit. ISO (2019, p. 6) also states that as a UCD principle "the design team includes multidisciplinary skills and perspectives". Finally, cultural probes could be used to reach users in various locations at the same time similar to questionnaires. This would however require more resources to the creation of cultural probes as their amount would increase.

Participation Questionnaires can be delivered and answered online but they are not very engaging, and a user might leave a questionnaire unanswered because it might not be directly addressed to them. Interviews are more personal. They can

Table 1. A summary of information gathering methods

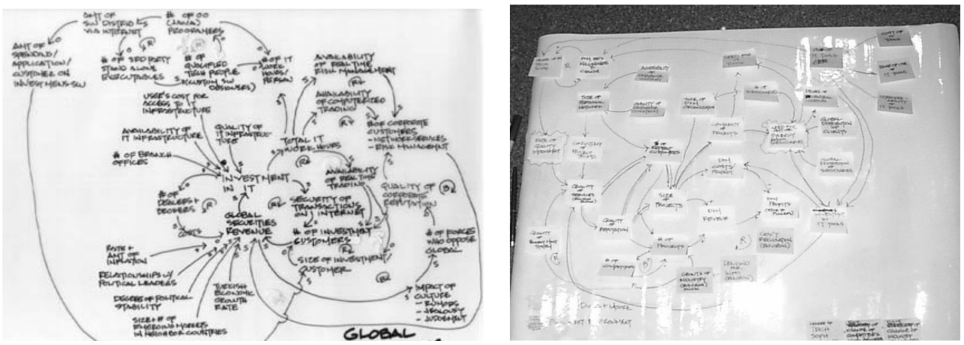
	Benefits	Deficits	Needed Resources	Reach	Experience	Participation
Question-naires	Easy and simple.	No follow-up questions.	Not much time.	Multip-le users.	Not much	Not very engaging. Can be online.
Contextual Inquiry	Observe actions. Questions about current activity.	No in-depth questions.	Observing time. No reserved time from user.	Obser-ved user(s).	Not much. Spontaneous questioning.	Engaging. Cannot be online.
Individual Interviews	In-depth, follow-up questions. User can query.	Concentra-tes on indi-viduals.	Reserved time from designer and user.	Inter-viewed user.	Same as above.	Engaging. Can be online. Better in person.
Group Interviews	Follow-up questions. Multiple answers.	No personal in-depth views.	Same as above.	Inter-viewed users.	Same as above.	Not as engaging. Can be online. Better in person.
Cultural Probes	Broad answer scope.	No follow-up questions.	Creation of Probes. Time-dependent. on users.	Multip-le users.	Some for creation of probes.	Engaging. Online/ physical artifacts. In person delivery.

be done online but I would argue that they are better face-to-face. The distance between the interviewer and interviewee then decreases and the interviewee can feel more comfortable in answering and asking questions. For example, Gaver et al. (1999) decided to visit their users face-to-face to explain their project during which the probes also sparked lively dialogs between the designers and users. Regarding the probes, Gaver et al. (1999, p. 29) highlighted that if their method would be repeated, the probes should not be "machine-addressed letters". I think that they could however be electronical artifacts, but their delivery should be accompanied with an on-site visit. The use of cultural probes appears quite engaging, and a designer could possibly get a lot of answers with them even though some of the groups of Gaver et al. (1999) did not mail back all the probes.

To summarize (See Table 1), questionnaires might appear like an easy and quick way to gather data, but they can end up costing more if multiple surveys are needed. If the focus is not clear and the goal is not to reach a lot of users, qualitative interviews and contextual inquiries might be the right approach. These may take up more time, but it can be reduced by interviewing multiple users at once or by multiple researchers. Even though the focus does not need to be as narrow as with questionnaires, limiting it will reduce the time needed in the field. Cultural probes could be used to engage the users in research even more and they are applicable with an even wider focus. The use of probes might however require some more experience and its required time is unpredictable. Remote measures can be applied to all of these methods, but with questionnaires the effects will be the less negative.

3. User-Centered Methods in Constructing a Common Understanding

As user-centered design is usually done in teams, it is important that the gathered data is analyzed in a manner that produces a clear and common understanding of the context-of-use and the system's requirements for the whole team. The data from field work can come in many forms. DiCicco-Bloom and Crabtree (2006) talk about notes and audio tapes where as Holtzblatt and Beyer (1993) talk about drawing pictures during contextual inquiry. DiCicco-Bloom and Crabtree (2006) described some analysis strategies such as crystallization which rely on the researcher's expertise which are not however collaborative or rapid in comparison to others that I will discuss, which is why



(a) Example of a preliminary causal model (b) Causal model from a case study

Figure 1. Examples of causal models portraying relations (Millen, 2000)

I am not focusing on them. DiCicco-Bloom and Crabtree (2006) also mention using software programs to assist with data analysis, which Millen (2000) promotes as a way to speed up data analysis. However, if data is just analyzed by software programs, the connections in the data might be left unclear for the design team as they do not construct those relations themselves.

Millen (2000) highlights collaborative data analysis as a tool for rapid ethnography which for example different modeling processes, personas and scenarios are examples of. With these methods, a team uses cognitive processes to develop an understanding of the data. Holtzblatt and Beyer (1993) use affinity diagrams to bring data together from contextual inquiries. They are constructed by witting down the gathered insights on post-it notes which are grouped together into clusters defined by the team. This approach brings together data from different subjects and designers creating relations between them. An affinity diagram can also show where more data is needed (Holtzblatt & Beyer, 1993). Millen (2000) also explains that they were able to identify gaps in their learning as they constructed an informal causal model of their visited site (See Figure 1b). I have done similar clustering exercises during my Sprint Retrospective meetings, which are meetings to discuss the work methods of the team. Creating clusters have helped us to identify related problems and focus on core problems. Holtzblatt and Beyer (1993) also construct work model diagrams, that display the contexts users work in, their physical environments and roles, and sequences of actions. These diagrams help focus conversations and they help make disagreements explicit (Holtzblatt & Beyer, 1993). I would say that these types of models, for example the ones in Figure 1, could have helped our team during our software project as well as it was difficult coming up with a shared understanding without a visual model to refer to. Finally, Holtzblatt and Beyer (1993) construct a redesigned work model that brings together their models.

Pruitt and Grudin (2003) talk about personas in creating an understanding about the context-of-use and as a tool for conversation for the team. The authors explain personas as fictional characters that represent the user groups of a system which can create a strong focus on users and work contexts. In addition, because they are widely known, the team has a common understanding of their audience, and the service or product is easier to discuss (Pruitt & Grudin, 2003). Overall, this method has a lot of the same benefits as modeling languages. Pruitt and Grudin (2003) also mention scenario construction when comparing it to personas. Scenario construction can be simpler, but according to the authors, scenarios can be created to promote any feature. I have written down user stories in previous projects, which remind of scenario

construction, during which I have also realized that they are easy to write to promote any idea. Having personas to constrain the stories could help explain them better and make them more convincing.

Finally, Ismirle (2018) uses experience mapping to gather the thoughts of her team's interviewed participants to display commonalities and differences in their contextual experiences. The goal of her use of experience maps is to depict a variety of holistic user experiences and not to just focus on the average user but instead incorporate possible edge cases as well. In my experience, these types of edge cases are also important to account for in software development to ensure that the software will work at all times. Similarly, a designer decides whose experiences they will design for. Personas can also take into consideration a wide range of users when creating multi-dimensional and embodied personas (Ismirle, 2018). Ismirle (2018) created an experience map for each of her analyzed participants, which helped to see clear differences. An opposite result is created with the redesigns by Holtzblatt and Beyer (1993) as customer-specific data is left out. Ismirle (2018) also reported that experience mapping helped her to identify gaps where more data was needed. However, similar to personas and scenarios, this can be a risky tool to use. Experience maps can narrow the focus, since designers make statements of experiences based to their own worldview (Ismirle, 2018). Ismirle (2018) says that having a team to create the maps could have widened her scope.

Required Resources Analyzing data with software programs can be fast and it does not require a lot of effort from the designer either. The construction of affinity diagrams or work models might not take very long as the whole team works together, but it does require human resources. The construction of personas can take a long time and require additional resources depending on the method's thoroughness. For example, in the second project of Pruitt and Grudin (2003), posters, tools of use, and promotional items were created which can however advance the use of personas by bringing them closer to the designers. The resources can also pay themselves back since according to Pruitt and Grudin (2003), the designers are able to produce generative data once affiliated with the personas. Scenario construction does not require as many resources, but constructing experience maps of individuals requires more resources that constructing coherent models of the users. However, Ismirle (2018) does not suggest that an experience map should be created of every user. This approach could actually maybe be used to consider a few edge cases in addition to the main user groups depicted by larger models.

Required Experience The construction of different diagrams and scenarios appear to require the least amount of experience. The construction of affinity diagrams is not complicated as Holtzblatt and Beyer (1993) explains it, but the work model diagrams on the other hand can require knowledge of the modeling languages. Fortunately, Holtzblatt and Beyer (1993) explain that their teams pick up the languages quickly. Analyzing data with a software program requires knowledge about the program, and the use of personas appears to require the most experience. Since it is a powerful tool, the designers need to know how to use them, so that their risks can be limited as personas "can take down the wrong path" if not used appropriately (Pruitt & Grudin, 2003, p. 14). Pruitt and Grudin (2003) also explain that their second personas-effort turned out better because of their gained experience (and additional resources). Ismirle (2018) used a template to construct experience maps to help with her inexperience, but it also somewhat restricted her.

Participation The analysis of data via software programs does not require live presence of the designer, but most of the collaborative data analysis methods would best to conduct face-to-face with other designers. Holtzblatt and Beyer (1993) instruct

Table 2. A summary of methods for constructing understanding

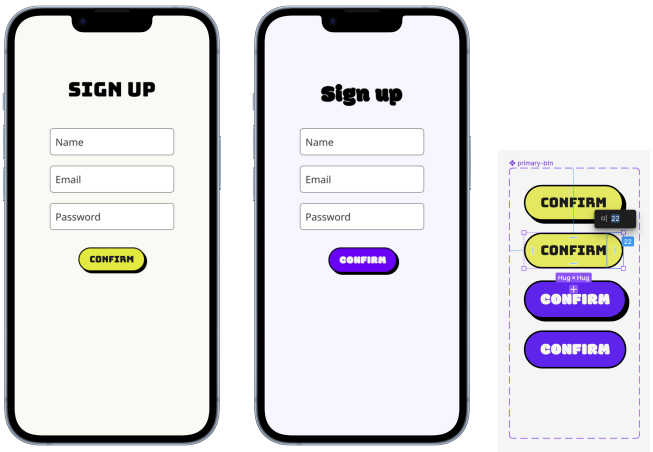
	Benefits	Deficits	Needed Resources	Experience	Participation
Software programs	Easy and simple. Displays patterns.	Does not require cognitive effort.	Not much time.	Knowledge of the program.	Not Very engaging. Can be online.
Visual Models	Creates relations + focus. Shows data gaps. Conversation tool.	Can disregard individual experiences.	Not much time.	Not much for simple diagrams. Knowledge of models.	Engaging. Can be online. Better in person.
Scenarios	Creates focus. Conversation tool.	Risky. Grants power to the designer. Can favor ideas.	Not much time.	Not much	Same as above.
Personas	Creates relations + focus. Generative. Conversation tool. Wide range of users.	Risky. Grants power to the designer.	Time depends on throughness. Other resources.	The most of these methods.	Very engagin.g Can be used remotely after defining.
User Experinece Mapping	Creates focus. Shows data gaps. Conversation tool. Wide range of users.	Risky. Grants power to the designer. Can favor ideas.	Same as above.	Some. Templates can help.	Engaging. Can be online. Teams better in person.

to do all of their methods in person with physical artifacts. They even suggest dedicating one entire room for the design process. One of the main benefits of the physical presence of others and artifacts is that the supporting data right in front of a designer and the rest of the team (Holtzblatt & Beyer, 1993). This is difficult to achieve with online. Diagrams can be shared via screen sharing but it is impossible to make sure that everyone is paying attention. Additionally, there is a higher threshold to start speaking. In my Sprint Retrospectives, we have gotten the most done when the whole team has been together physically as people have been more willing to express their ideas. The other sources do not contribute to the fact that whether the methods should be done online or face-to-face. I would imagine however that after a persona has come familiar, it could be used as a conversational tool online without a problem. Finally, collaborative data analysis tools are more engaging compared to computer-based ones, and personas especially engage team members effectively (Pruitt & Grudin, 2003).

As a summary, software-based analysis tools are a fast way to help with the analysis of data, but collaborative methods could be used especially when working in teams as they help to create a more coherent understanding. Models and personas can be used as conversational tools and to back up ideas. Models and experience maps also help realize where more understanding is needed. The use of personas is the most engaging method of these, but it also requires the most experience and resources. I would suggest using other less risky methods such as scenarios and models before diving into extreme persona-efforts. Personas could then be tried out as a complimenting method to the others. Analysis of data with software programs can be performed remotely, but collaborative data analysis methods can produce more value when performed in in the same room with others. Table 2 gathers these ideas.

4. User-Centered Methods in Producing, Evaluating and Iterating a Design Solution

After the data about the context-of-use has been understood and the system requirements specified, a designer can start to develop design solutions which should also be evaluated. A design solution that is detailed can give valuable insights about the final product, but it is not wise to produce almost complete products at first as resources



(a) Interface version 1 (b) Interface version 2 (c) Creating button versions

Figure 2. Example of interfaces and interface elements created with Figma.

might go into producing unusable features. ISO (2019) also states that the UCD process should be iterative. One of the most common ways to produce design solutions is prototyping. Prototypes can either be of low-fidelity or high-fidelity depending on how well you want them to match the final product. Holtzblatt and Beyer (1993) use an approach where they first develop a user environment design based on their redesigned work model. The environment design helps a team to figure out the basic structure and functions of the system and it is constructed of paper artifacts which can be moved around. This model is tested only by the team with walking scenarios. Afterwards, the design of the user interface (UI) can start. Prototypes are first constructed of paper moving toward more careful ones. One of the main ideas of Holtzblatt and Beyer (1993) is to focus on the underlying structure of the system at first so that level decisions can be put off. Another one of their principles is to create a prototype of the detail that you want the users to respond to.

Low-fidelity prototypes focus on different aspects of the system than high-fidelity ones, as they address larger functionalities. Holtzblatt and Beyer (1993) explain that low-fidelity prototypes can steer the user into testing more general functions than looking into details. However, low-fidelity prototypes can be more difficult to use in testing since their testing is not very explicit as the user needs to imagine more and more instructions might be needed. Holtzblatt and Beyer (1993) test their rough UI paper prototypes by asking users to perform tasks with them. However, I am not sure how a user should conduct their work with a paper prototype and give feedback about it since it does not support many actions. I would say that if paper is used for prototyping a UI, it should just give a feel of what the UI could look like.

The approach of Holtzblatt and Beyer (1993) appears to diminish the effect of a good UI as it focuses on it so late in their process and it is constrained by the user environment design. In the world today, trendy and aesthetic UIs are becoming more popular, and in my opinion, it is not wise to constrain them too much. Gaver et al. (1999) also think that aesthetics are an integral part of functionality. Millen (2000) mentions rapid design tools for developing screen-based UIs. The design tool that comes to my mind is popularity gaining Figma. With Figma, a designer is able to develop more realistic looking software interfaces (See Figure 2a and Figure 2b) than with paper and prototypes that also provide functionality. As these design tools are

relatively new, Holtzblatt and Beyer (1993) might have a different view about using paper as a medium for prototyping in our current time. Multiple sources also mentioned participatory co-creation for producing design solutions which does not however appear applicable in my field (software development) as Pruitt and Grudin (2003, p. 13-14) explain it to make sense "when designing for a relatively small, accessible group".

These prototypes or other design solutions should be evaluated and refined based on user-centered evaluation which is categorized into user-based testing and inspection-based evaluation (ISO, 2019). User-based testing is done by users in either monitored laboratory-type environments or in the field which will give more accurate information about how the system fits into its context. User-based testing is beneficial as it involves the user which is one of the principals of UCD according to ISO (2019). I would say that user-based testing would be best to conduct with new users as well as the ones that were used to define the system's requirements. The already identified users could test the aspects of the systems that were designed based on their knowledge and the new users could reveal unexpected deficits. For example, Holtzblatt and Beyer (1993) use new customers to evaluate their redesigned work models.

Inspection-based evaluation on the other hand is executed by professionals. Professionals can discover other problems that the actual users would not. They evaluate systems based on previous experience, guidelines and standards, and they can consider a wider range of users and tasks than would be tested by a user. This type of testing does however tend to focus on obvious problems and might not be appropriate when considering complex and innovative systems. (ISO, 2019).

Other aspects and artifacts of the UCD process should also be evaluated in addition to the final design solutions and feedback can be for example gathered with similar methods that were used for user research. Holtzblatt and Beyer (1993) refine some of their models using contextual inquiries, some personas by Pruitt and Grudin (2003) were refined by consulting a user group panel, and Ismirel (2018) evaluated her experience maps with her original participants.

Required Resources Usually, low-fidelity paper prototypes are quicker to develop than high-fidelity ones. The approach of Holtzblatt and Beyer (1993) uses multiple prototypes of different levels which requires even more time and effort. However, this might pay itself back in the future as the underlying structure is well thought out before the implementation of the UI, which leads to an interface that will address the user's needs well. For example, at the start of a software project of mine, we started developing the UI almost straight away without a clear idea of the underlying data structure, which lead to having to modify the UI multiple times. In another project we first made a low-fidelity prototype of a hardware product which we refined according to the results of user-based tests. This process was wise as our retrieved feedback helped modify the design in the right direction before spending too much time on the first prototype. Constructing prototypes with design tools is slower and more time consuming than drawing on paper but changing small aspects of the design can be faster as there is no need to start from scratch. I have actually been able to try out slightly different designs with Figma in a fairly short amount of time. Figure 2 demonstrates these simple changes with colors and fonts, and Figure 2c shows how small elements such as buttons can be modified. The way the user environment design is constructed by Holtzblatt and Beyer (1993) also supports quick modifying. Regarding testing, inspection-based evaluation is quicker and simpler to carry out than user-based testing (ISO, 2019). It does however require finding and paying suitable inspectors.

Required Experience Constructing prototypes with paper does not require a lot

Table 3. A summary of methods for producing and evaluating designs

	Benefits	Deficits	Needed Resources	Experience	Participation
Low-fidelity Prototypes	Easy and simple. Test the structure and main functions.	Confusing tests. Only tests a few aspects.	Not much.	Not much.	Not Very engaging. Teams better in person.
High-fidelity Prototypes	Feedback related to the actual product. Tests detailed and main aspects.	User might focus only on details.	Inital one takes more time than above. Modifying is fast.	More than above. Knowledge of tools.	Engaging. Software-based tools can be used remotely.
User Environment Design	Creates a structure without concentrating on details.	Can disregard the importance of details.	Not much. Modifying is fast.	Not much.	Engaging. No users involved. Teams in person.
User-Based Testing	Involves users. Data of the fit into the context-of-use. Discovers odd problems.	Can leave some problems undiscovered.	Time depends on throughness. Time-dependent on the testers.	Not much from designer nor user.	Engaging. Can be online. Better in person.
Inspection-Based Testing	Easy and simple. Tests a larger set of actions. Discovers obvious problems.	Can leave some problems undiscovered.	Same as above. Requires less testers and time. Recruiting professionals.	Not much from designer. Tester needs to be experienced.	Engaging. Can be online. Better in person.

of experience but constructing prototypes with design tools requires knowledge of the tools. I have limited experience in prototyping hardware products, but I would imagine that there is a higher leap in expertise of what a low-fidelity prototype requires to a high-fidelity one than with software programs. The overall prototyping approach that Holtzblatt and Beyer (1993) use does not appear complicated. User-based testing does not require a lot of experience from the tester if the test is well outlined, and the tester knows what to inspect. I for example was able to perform simple user-based testing with hardware prototypes without previous experience but with some guidance on how the practicalities of user-based testing work. Inspection-based evaluation requires an experienced professional.

Participation Creating a user environment design engages the whole team, but it requires physical contact to work. Paper prototypes and prototypes created with design tools can be constructed individually. Using software design tools does not even require physical contact as the designers can construct prototypes on a remote platform. However, creating prototypes online might not be as engaging as creating them face-to-face. Regarding user engagement during the testing of prototypes, high-fidelity prototypes work better since there is usually more things to inspect and different actions to take. For example, most of the time, users have been eager to try out my prototypes created on Figma. User-based testing engages the designer and user quite well. User-centred testing should be executed in person as some actions cannot be seen remotely especially with hardware products. I have demonstrated a software program to users online, but having them present with me would have been more helpful. Testing of a software program remotely would require that the user has the needed equipment to run the software and testing hardware would require its presence.

A summary of the methods described here is displayed in Table 3. Overall, I support the idea of Holtzblatt and Beyer (1993) that a designer should first produce low-fidelity prototypes before moving onto complicated ones so feedback could be retrieved quickly. ISO (2019) also explains that a benefit of producing design solutions is to incorporate users' feedback into the design early in the process. As an exception to this, I highlight the use of rapid design tools when designing software programs. This is because they can produce high-fidelity prototypes quickly on a platform where the designs are easy to alter with the support of online interactions. Regarding testing, user-based testing is a better method if there is time for it as it involves the user. User-centered evaluation

of software programs is possible to execute online but for the convenience of the user, I would recommend doing it in person.

5. Summary

As a conclusion, there are many methods to choose from for various UCD activities and as Wilson (2006) explains, there are benefits in using multiple ones. Using multiple different methods for one activity can help improve the results of that activity as they become more evident. If there is not enough resources to use multiple different methods, using different variations of one method can be enough. For example, information could be gathered by multiple designers observing different activities and usability tests could be executed by different users. There is no right method for an activity since all of them have different strengths. There is no right way to combine these methods either. For example, there is no definition on which information gathering techniques produce the most valuable data for the construction of personas. In the end, a designer should choose a method that fits their project best and not be afraid to explore methods from other fields as well as some of the practitioners surveyed by Roto et al. (2021) do. There are a lot more methods developed in addition to the ones I have described here for each activity. Generally, a designer needs to find a balance between the exhaustiveness of a method and the value of its results.

References

Boynton, P. M., & Greenhalgh, T. (2004). Selecting, designing, and developing your questionnaire. *BMJ?: British Medical Journal*, *328*(7451), 1312–1315. doi:10.1136/bmj.328.7451.1312

DiCicco-Bloom, B., & Crabtree, B. F. (2006). The qualitative research interview. *Medical Education*, *40*(4), 314–321. doi:10.1111/j.1365-2929.2006.02418.x

Gaver, B., Dunne, T., & Pacenti, E. (1999). Cultural Probes. *Interactions*, *6*(1), 21–29. doi:10.1145/291224.291235

Holtzblatt, K., & Beyer, H. (1993). Making Customer-centered Design Work for Teams. *Commun. ACM*, *36*(10), 92–103. doi:10.1145/163430.164050

International Organization for Standardization. (2019). *Ergonomics of human-system interaction–Part 210: Human-centred design for interactive systems* (ISO Standard No. 9241-210)

Ismirle, J. (2018). Using Experience Maps to Consider Individual Stories. *Proceedings of SIGDOC 2018 Conference on the Design of Communication*. New York, NY: ACM. doi:10.1145/3233756.3233954

Millen, D. R. (2000). Rapid Ethnography: Time Deepening Strategies for HCI Field Research. *Proceedings of DIS 2000 Conference on Designing Interactive Systems: Processes, Practices, Methods, and Techniques*. New York, NY: ACM. doi:10.1145/347642.347763

Pruitt, J., & Grudin, J. (2003). Personas: Practice and Theory. *Proceedings of DUX 2003 Conference on Designing for User Experiences*. New York, NY: ACM. doi:10.1145/997078.997089

Roto, V., Lee, J. -J., Law, E. L. -C., & Zimmerman, J. (2021). The Overlaps and Boundaries Between Service Design and User Experience Design. *Proceedings of DIS 2021 Conference on Designing Interactive Systems*. New York, NY: ACM. doi:10.1145/3461778.3462058

Wilson, C. E. (2006). Triangulation: The Explicit Use of Multiple Methods, Measures, and Approaches for Determining Core Issues in Product Development. *Interactions*, *13*(6), 46–ff. doi:10.1145/1167948.1167980