Rapid and rich prototyping: proof of concepts for experience

Elbert-Jan Hennipman[1], Evert-Jan R.G. Oppelaar[1], Gerrit C. van der Veer[1], Bert Bongers[2]

[1]Open Universiteit Nederland
Valkenburgerweg 177
6419 AT Heerlen
The Netherlands
ehn@ou.nl, oppelaar@gmail.com, gerrit@acm.org

[2]University of Technology Sydney
PO Box 123
Broadway
NSW 2007 Australia
bertbon@xs4all.nl

ABSTRACT

In this paper we explore a suitable prototyping technique and approach for an experience-oriented design process without the need for rich equipped labs and resources. The key solution lies in 'mixed-fidelity' prototypes with interaction-enabled 'front-ends' and simple 'back-ends'. We illustrate and validate this approach mainly with a student project done by the authors dealing with environmental aware mobile information.

KEYWORDS

Experience design, prototyping techniques, environmental awareness

INTRODUCTION

An iterative design process, where the design goes through several stages before it is finalized, is well accepted to be an effective contribution to the development of interactive systems (Nielsen, 1993). In order to keep this process within time and money constraints, prototypes are used for these stages. This is also the case for Activity-centered (i.e. Yang & James, 2008) or task-centered design processes (i.e.Van der Veer & Van Welie, 2000)

In this paper we aim for a rapid and rich prototyping method which fits in an experience-oriented design process to make it doable and affordable for small companies and institutions with little resources and specialized knowledge available. We explore this method through a case study of a new mobile concept which is 'situation aware'.

DEFINITIONS

A dictionary such as the American Heritage Dictionary defines 'prototype' as an original type, form or instance that serves as a model on which later stages are based or judged' (Eckersley, 2007). Other dictionaries, such as Oxford (2008) and Cambridge (2008) have similar definitions. These definitions imply that 'prototype' is an ambiguous term: many things of very different quality can be called a prototype, varying from rough sketches to very precise clay models to (partly) running systems. What all these forms have in common, is that they serve as materials to reflect and evaluate design ideas for future artifacts.

Our design focus is *experience,* which is also an ambiguous term. We approach experience as a continuously evolving story of the user who first has expectations about an artifact, then gets confronted with it, uses it, and evaluates it afterwards. This takes place in a specific context, at a certain time and the user brings previous findings, mental models and experiences to it (i.e. Vyas & van der Veer, 2006; Wright, McCarthy, & Meekison, 2003; Yamazaki & Furuta, 2007).

Prototyping in an experience design context means that the prototype gets confronted with 'the wild': the intended context and target groups the designer (or design team) aims to design for. In many cases, it might additionally be useful to test the prototype in unexpected contexts with different target groups because interactive systems become more widely used by different groups of users and in various new situations.

TOWARDS EXPERIENCE PROTOTYPING

Prototyping can be done with different goals in mind. Floyd classically separated the three 'E'-goals: Explorative, Experimental and Evolutionary (Bäumer, Bischofberger, Lichter, & Züllighoven, 1996; Interaction-Design.org, 2004). Explorative prototyping is often done very rapidly, *in-situ*, and with few resources in order to gain insight in how people react on new concepts and ideas. It offers information about what use cases and tasks can be fulfilled with the concept, what contexts are applicable in which it makes sense to use the concept and what users are interested.

ECCE'08, September 16-19, 2008, Madeira, Portugal.
ACM 978-1-60558-399-0/08/09

Experimental prototyping is done in order to carefully evaluate if the concept (often in a more mature state) actually *does* meet the expectations. Experimental prototypes are often tested against predefined use-cases and scenarios with end-users.

In Evolutionary prototyping, experimental and explorative go hand in hand. Over time, the prototype is evaluated several times and is becoming more and more mature, to end in a final prototype that is ready for production.

A typical design process of an interactive system starts with idea gathering and requirements engineering (using personas, scenarios, and use cases). Already in this stage, it is useful both for future users and for the design team to visualize the ideas with sketching and other simple modelling techniques (named *low fidelity* prototypes). Making ideas visible, together with observation and interviewing, helps to identify market requirements, and to evaluate multiple design concepts (Rudd, Stern, & Isensee, 1996). In other words: sketching helps to externalize the thinking process (making tangible what is intangible), to make ideas visible and to engage discussion about it (Baskinger, 2008; Buxton, 2007). This is to be compared with an explorative prototyping setting.

When the design and requirements of the future artifact becomes more mature, it makes sense to continue prototyping, now with more advanced prototypes which offer a detailed overview of the artifact and, in the end, full functionality. This is useful for the technical engineering team (programmers) and for detailed end-user evaluation (Rudd et al., 1996).The prototyping settings can still be explorative, but experimental or evolutionary is more often the case.

There are many ways to construct prototypes. It is not our intention to discuss all these ways, but no matter what way is chosen, it is important that the chosen way can be evaluated in relevant contexts (environments) in order to gain insight in the actual experiences that people get when they are confronted with the artifact. Many prototyping methods structure the *interface* quite well (with labels or tables or…), but this is not the complete story.

Research labs at Apple (i.e. Laurel & Mountford, 1990) and Philips (i.e. Aarts & Marzano, 2003) advocate 'experience prototyping', where requirements engineering is not only done by development of scenarios and translation of use cases, but also by developing feasible prototypes that provide proof of concept (Aarts & Diederiks, 2006). With this kind of prototyping, it is possible to go beyond functionality and usability in order to ask questions as 'What role will the artifact play in a user's life?' and 'How should it look and feel?'. To answer these questions properly in consult with end users, 'high resolution' prototypes are needed in order to let people concretely visualize the design (Houde & Hill, 1997).

When it comes to experience prototyping, in our view all three prototyping settings can be the case. On one hand, it is possible to evaluate how predefined use cases, tasks and activities are functioning with the interactive artefact, which is active in the right context and with the right users (experimental or evolutionary setting). On the other hand, it is possible to gain knowledge about what activities and use cases are surfacing when the (concept) artifact is used within possible contexts and with possible users (explorative setting).

Simple experience prototyping: mixed fidelity

Experience prototyping is done by Philips and Apple in well equipped research labs with much technology and resources available. The resulting prototypes are of 'high fidelity': they look like fully developed products with high technology and functionality. However, the practice of experience prototyping is not restricted to companies with a high R&D budget. Buchenau and Suri (2000) define the essential difference between classical prototyping and experience prototyping as the turn from passive towards active audience participation. An experience prototype can be anything made from any material as long as it allows interacting with it. Experience prototyping can be done with very low-tech methods and improvisation with basic materials (Buchenau & Suri, 2000). However, low-fidelity prototypes made out of primitive materials or using simple techniques have their disadvantages: it is harder to claim that the evaluation findings are originated by the actual concept of the artifact or by the innate characteristics of the prototype (Lim, Pangam, Periyasami, & Aneja, 2006).

In our view, the ultimate solution is the use of 'mixed fidelity prototypes'. This term was introduced by McCurdy et al. (2006), who categorize the amount of 'fidelity' in prototypes through five dimensions: 1) level of visual refinement, 2) breadth of functionality, 3) depth of functionality, 4) richness of interactivity and 5) richness of data model. For an experience prototype, it is important that the 'front-end', the part of the prototype visible by the end users, should be as real as possible (affecting dimensions 1, 2 and 4). The back-end of the prototype (dimensions 3 and 5) is not seen by end-users and can be developed with very simple techniques (including wizard of Oz). This approach makes it possible to perform experience prototyping without sophisticated research labs and resources. The following case shows this approach.

INFOZPHERE

The concept, design and prototype of the InfoZphere was the result of a bachelor-project done by Elbert-Jan Hennipman, Evert-Jan Oppelaar and Wouter Broekhof under supervision of Bert Bongers in 2005 at the Vrije Universiteit, Amsterdam (Bongers, 2006).

The InfoZphere is a concept and framework that enables entities (objects, buildings) in the world to 'radiate' information, and so create a rich environment in which location related information is readily available. The practical use thereof varies from grocery stores radiating weekly advertisements to light bulbs announcing their lumens, watts and remaining lifespan. A wide variety of uses can be thought of. Our focus was the InfoZphere to become an integral part of the tangible world around us. The radiated information can be accessed by a (mobile) device, and delivered to the end-user, allowing detailed exploration.

A more sophisticated use could be to offer personalized information, though this would require the entities in the world not only to radiate, but also to receive and process incoming information.

To prove the concept of the InfoZphere, we developed a working prototype. The development process had four phases: (1) creating a syntax that enables information radiation, (2) developing a model for the interaction, (3) implementing both of these in enough detail for a prototype, and (4) proving our concept through a demonstration with a working prototype.

In order to 'prove our concept', the prototype had to give the end-user the *experience* that the information was really radiated by an entity in their environment.

The student project, with virtually no budget, was on a tight schedule. These conditions make this project, and especially the development process, an interesting case study on rapid prototyping for experience.

Phase 1, creating an 'information radiation' syntax

The initial phase was to define the intended use and capabilities of our information radiating environment. We used well-known requirements engineering techniques (personas, scenarios, interviews, and questionnaires) to elicit some of the possible uses of the technology, and to analyze the capabilities our syntax should have. Rough sketches were used (figure 1), both within the team, and to illustrate our ideas to potential users.

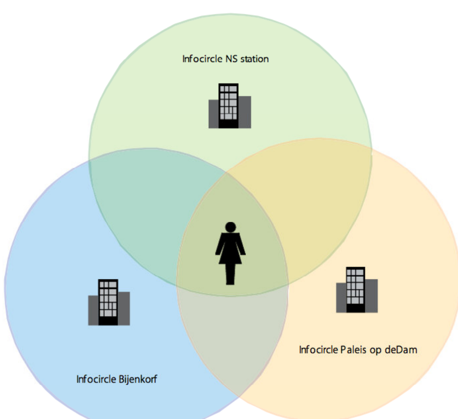

Figure 1: example sketch of InfoZphere environmental awareness. Unpublished work.

Developing the syntax was mainly done through literature study. The syntax was designed in a layered way to support several existing wireless technologies and to make sure that new emerging technologies could be used as well.

Phase 2, the new interaction model

This information-augmented environment needed a new interaction model. Several issues needed to be addressed in this interaction model, such as using push or pull, protecting privacy, the type of information being radiated, the representation of that information, personalization, system availability, device capabilities, relevant environmental information, and more.

We observed that it might sometimes be useful to place a 'virtual self' somewhere else to get the personalized and location-based information from that spot (figure 2).

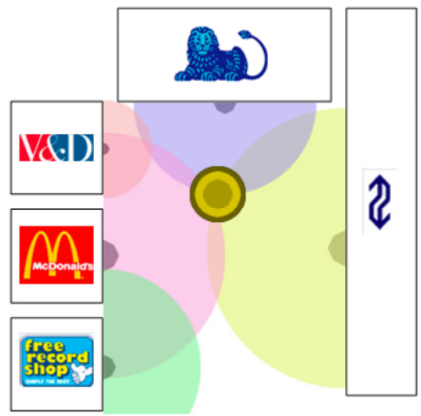

Figure 2: environmental awareness: individual information sources. Unpublished work.

In this phase we explored different interaction models, using low fidelity prototypes, such as sketches and mock-ups. We found that different situations ask for different interaction models. In some situations personalized information was preferred (e.g. train departures), while in other situations personalization had no use at all (e.g. the lamp announcing its probable lifespan).

In this phase low fidelity prototypes were strong enough, because we were not exploring the overall experience, but only aspects related to functionality and usability.

Phase 3, implement syntax and interaction model

To be able to demonstrate our concept, we needed to realize some functionality in both syntax and interaction model. We chose to implement only the most essential parts of the syntax. In our case, we decided to limit our prototype to Wireless Access Points as information radiators, because they are readily available and because one of their functionalities is to radiate information. We chose to implement the interaction model in HTML, because most (mobile) devices already include some kind of web browser.

Though our syntax was designed to accommodate existing and future wireless technology, we implemented it only for WLAN, which saved us quite some time. Implementing the interaction model in HTML saved time as well, since no real software-development was needed for that.

This approach reduced our initial design problem (creating a working prototype) to 'enabling WLAN detection and connectivity through the user interface' and 'creating a visual representation of a possible device'.

Phase 4, proving the concept through a prototype

We used a tool called Zinc (2008) in combination with Adobe Flash technology to create the prototype.

Through Zinc and a command-line tool we could detect all available Wireless Network SSIDs. To make a distinction between 'normal' wireless networks and wireless networks being part of the InfoZphere, we decided to use a naming convention (only SSIDs starting with 'IZ_' were part of our InfoZphere).

The prototype then matched the SSID to preconfigured items (warehouse, train station, the tube), which simply appeared or disappeared if the associated network was in or out of range. We used preconfigured items to make sure there was no lag in operation of the prototype, and to be able to present the user a graphically appealing interface in stead of basic textual navigation. If a user chose one of the available 'locations', the prototype would register itself to the WLAN router via DHCP, and automatically load a webpage that was accessible through that router. That webpage then followed the interaction model (figure 3).

Using this approach, the prototype was already designed in a way that was very scalable (adding more correctly named Wireless Access Points automatically resulted in more 'locations' being found).

Figure 3: 'real' environmental-aware information. Unpublished work.

Despite our efforts to reduce lag in the detection and listing of all 'locations within reach', our solution was found to be a bit sluggish. Still, users of the prototype experienced the InfoZphere as if it actually existed.

Using such a device in a real-world context, in stead of a simulated context would have improved the intended experience even more.

Phase 5, evaluation

Creating a prototype that is robust enough to be used 'in the wild' offers great opportunities for a rich evaluation. Because of the realistic context, users will not only analyze the prototype itself, but inevitably raise question about its use in that context. This enables research of 'matching the intended experience', as well as research of new and unthought-of features for the prototype.

From our experience we claim that the choice of prototyping technique should depend on what aspect of the system has to be analyzed. When the target is to do usability evaluation, it might be sufficient to work with paper prototyping, wireframes, or scenarios. Other prototyping techniques should be considered when the intention is to do explorative experience evaluation, because the lived experience is very dependant on context (the situation, task at hand, the environment).

When a prototype is being used in a real-world setting, users will form another 'User Virtual Machine' of the prototype. Realizing what it can do already, they might get inspired by the context to propose new features and other functionality. It is this creative user-feedback that is very hard to get in a lab setting (where most of the user-feedback would probably be corrective in nature). By enabling this explorative prototyping, relatively cheap evaluation techniques such as think aloud protocol and observation can be valuable assets to improve the prototype, without having to do extensive market research. At the same time, the users are working with the prototype, and will identify issues in 'intended use' as well.

Our project was done in 2005, in the Netherlands. One of the InfoZphere nodes was London Underground station Westminster. Consequently, our project was never evaluated in the wild, but in a simulated lab setting. This did yield feedback on the experience and workings of the prototype itself, but there was no creative process.

In a similar student project, students developed a location aware street game. A prototype of that game was tested and evaluated in a real-life context, resulting in people actually running around to locate each other. The users of that prototype envisioned other applications and games to be implemented in the prototype. Here, the feedback was both corrective and creative.

We found that using a mixed fidelity prototype it was possible to create an experience in which the user got involved, becoming an active thinker rather than a passive observer or passive user of a tool.

The experience evaluation of such projects is difficult. Vyas & Van der Veer (2006) developed a methodology to do qualitative experience evaluation for a specific prototype (set-top box). This qualitative evaluation method deals carefully with the idea of experience as a holistic concept and concentrates on four different entrance dimensions of experience (emotional, cognitive, practical, sensual). For the evaluation of future projects, we would like to use and adapt this methodology.

Back in 2005 there were few tools to do rapid mixed fidelity prototyping, especially when prototyping location based information devices. Nowadays better tools have been developed for prototyping, such as ActivityDesigner (Yang & James, 2008), and Roomware (shown on the Dutch CHI conference 2008, (Alchemyst, 2008)) for example for network connectivity. When designing for experience it is irrelevant what tools are being used, as long as the user is not confronted with implications of the tools that could be disruptive in the intended (and/or lived) experience.

Other solutions, other characteristics

In a later student project that was loosely inspired on the InfoZphere, the sluggishness experienced by the users of our prototype was solved using the wizard-of-Oz technique. The consequence was that someone was continuously monitoring the locations of the users, and enabled or disabled a small game when the user reached a certain spot. Though this worked very well for two users with two 'hot-spots', there was a scalability problem. More users or more 'hot spots' would make the task of the 'wizard of Oz' too complex to maintain and to manage.

Other approaches can be thought of, e.g. combining Google maps functionality with techniques as Skyhook Wireless's WPS (used on the Apple iPhone) or Googles own 'My Location' functionality. These technologies however were not readily available during our InfoZphere project in 2005.

DISCUSSION

When designing for experience, the use of a mixed fidelity prototype can be of great value. It combines the benefits of a high fidelity prototype at the front-end with the affordability and time-efficiency of a low fidelity prototype at the back-end.

Essential in the process is to find out what kind of prototype can be used at a certain stage in the design process.

From our observations we conclude that an essential aspect is the creativity of the developer. Out-of-the-box thinking is necessary to find possibilities and opportunities for saving time and money in the development of the back-end. Use of readily available technologies, masked by a hiding interface or manipulated by a 'wizard of Oz' can in many situations trigger the desired user experience.

We found that different techniques can achieve the same result, but each will have its own side-effects (such as scalability, manageability, and controllability).

Aarts, A., & Diederiks, E. (2006). *Ambient Lifestyle - From Concept to Experience*: BIS Publishers.

Alchemyst. (2008). Roomware. Retrieved June 20, 2008, from http://roomwareproject.org/

Baskinger, M. (2008). Pencils Before Pixels: A Primer in Hand-Generated Sketching. *Interactions, 15*(2), 28-36.

Bäumer, D., Bischofberger, W. R., Lichter, H., & Züllighoven, H. (1996). *User interface prototyping\—concepts, tools, and experience*. Paper presented at the Proceedings of the 18th international conference on Software engineering.

Buchenau, M., & Suri, J. F. (2000). *Experience prototyping*. Paper presented at the DIS '00: Proceedings of the 3rd conference on Designing interactive systems, New York, NY, USA.

Buxton, B. (2007). *Sketching User Experiences*. San Francisco, USA: Morgan Kaufmann.

Eckersley, M. (2007, april 4, 2008). Experience Prototyping Techniques. *Interaction Design: Principles and Practices*, from http://www.humancentered.net/blog/IxDWk13-ExpProto.pdf

Houde, S., & Hill, C. (1997). What Do Prototypes Prototype? In M. Helander, L. T. & P. Prabhu (Eds.), *Handbook of Human-Computer Interaction (2nd ed.)*. Amsterdam: Elsevier Science.

Interaction-Design.org. (2004). Prorotyping. Retrieved June 20, 2008, from http://www.interaction-design.org/encyclopedia/prototyping.html

Lim, Y.-K., Pangam, A., Periyasami, S., & Aneja, S. (2006). *Comparative analysis of high- and low-fidelity prototypes for more valid usability evaluations of mobile devices*. Paper presented at the NordiCHI '06: Proceedings of the 4th Nordic conference on Human-computer interaction: changing roles.

McCurdy, M., Connors, C., Pyrzak, G., Kanefsky, B., & Vera, A. (2006). *Breaking the fidelity barrier: an examination of our current characterization of prototypes and an example of a mixed-fidelity success*. Paper presented at the CHI 2006: Proceedings of the SIGCHI conference on Human Factors in computing systems.

Nielsen, J. (1993). Iterative User-Interface Design. *Computer, 26*(11), 32-41.

Rudd, J., Stern, K., & Isensee, S. (1996). Low vs. High Fidelity: prototyping debate. *interactions, 3*(1), 76-85.

Van der Veer, G. C., & Van Welie, M. (2000). *Task Based Groupware Design: putting theory into practice*. Paper presented at the DIS '00: Proceedings of the 3rd conference on Designing interactive systems.

Vyas, D., & van der Veer, G. C. (2006). *Rich evaluations of entertainment experience: bridging the interpretational gap*. Paper presented at the ECCE '06: Proceedings of the 13th Eurpoean conference on Cognitive ergonomics. from http://doi.acm.org/10.1145/1274892.1274921

Wright, P., McCarthy, J., & Meekison, L. (2003). Making sense of experience. In M. A. Blyhe, A. F. Monk, K. Overbeeke & P. C. Wright (Eds.), *Funology: From Usability to Enjoyment* (pp. 43-53): Kluwer Academic Publishers.

Yamazaki, K., & Furuta, K. (2007). Design Tools for User Experience Design. *Lecture Notes in Compter Science, 4550*, 298-307.

Yang, L., & James, A. L. (2008). *Activity-based prototyping of ubicomp applications for long-lived, everyday human activities*. Paper presented at the CHI 2008: Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems.