Increasing Interactivity of Paper Prototyping with Smart Pen

Seyong Ha
Seoul National University
1 Gwanak-ro, Gwanak-gu,
Seoul 151-742, Korea
alcarin@snu.ac.kr

Jiwan Park
NeoLab Convergence Inc.
Teheran-ro 32-gil, Gangnam-gu,
Seoul 135-921, Korea
jiwanpark@neolab.net

Joonhwan Lee
Seoul National University
1 Gwanak-ro, Gwanak-gu,
Seoul 151-742, Korea
joonhwan@snu.ac.kr

ABSTRACT

Wireframes have been widely used in the early phase of user interface (UI) design processes. However, because they lack interactivity in usage, it is not possible for users to have a realistic experience. Through our preliminary study, we obtained findings that provide implications for developing a wireframe prototyping system. We designed a prototyping tool for ad hoc mobile prototyping, which allows designers build a prototyping system quickly and make a prototype interactive. The system supports a designer's natural behaviors and builds an interactive prototype from the designer's sketches. Our evaluation showed both positive and negative feedback. From users' feedback, we discovered that our tool was useful for interactive low-fidelity prototypes and that participants were interested in using it. Negative feedback provided design implications for a pen interaction technique to further develop prototyping tools.

Author Keywords

Prototyping; Pen-and-paper interaction; Low-fidelity; Interactive prototype.

ACM Classification Keywords

H.5.2. Information interfaces and presentation (e.g., HCI): User Interfaces: Prototyping.

INTRODUCTION

Since it is not easy to build a perfect user interface from the beginning [4], designers tend to follow a rapid and iterative design process rather than a sequential one. Throughout the process, designers ideate, implement, and test their ideas repeatedly. By doing so, they can get feedback from users and then refine the user interface. To obtain valuable feedback from users during this iterative process, designers build various forms of early design mock-ups such as paper prototypes and wireframes. Since the time and effort designers put into prototyping vary depending on the prototyping system's fidelity, designers should consider when to use and build each prototype in the process.

The term "fidelity" indicates how a prototype appears to users. It can be considered either in terms of interactivity or the quality of the UI design. That is, the higher the prototype's fidelity is, the closer the prototype's appearance, functionality, or both are to the final product. For example, visual artifacts made by the Photoshop program have high fidelity in terms of design quality such as colored buttons or a nicely designed layout; however, because Photoshop does not respond to user interactions, it can be considered a low-fidelity program in terms of interactivity. On the other hand, the prototypes made by computer programming languages, such as Visual Basic or HTML, have high fidelity in interactivity, but somewhat lower fidelity in design detail. Although user interface designers usually prefer to use low-fidelity prototypes in the early stage [5, 6, 9], they also want to use prototypes with high fidelity in interactivity because highly interactive prototypes can allow designers to verify how users receive the information flow on the user interface and to test out the interface in a real-life context. However, the cost of building high-fidelity prototypes is higher than low-fidelity ones at the early stage, it is not easy for designers to make the high-fidelity prototypes from the beginning.

For this reason, the aim of our research is to enhance interactivity of a paper prototype in the early stage of design process. Specifically, we focus on a wireframe prototype in a mobile situation. Wireframe prototypes are widely used in the early stage of design processes and are visual representations of user interfaces, including information architecture, an arrangement of user interface elements, and navigations of user interfaces. Since a wireframe is drawn with a pen on paper or on a whiteboard, it has low design quality and lacks interactivity for those who want to test user interface interactions [9]. We assume that increasing interactivity in the early stage of design processes can help designers to obtain valuable feedback from users, and finally, improve the quality of a user interface. We present a prototyping tool that builds an interactive prototype from rough sketches on paper without any external behaviors such as photographing and programming.

RELATED WORK

Pen-and-Paper Technology

In this research, we used a digital pen technology called *nCode.* A *nCode* is an almost invisible point that represents 48bit information within a 2mm square. Since the codes can be printed on paper-like material, a user can assign certain digital information on a paper by printing the codes and read the information from the paper with a *nCode* pen. The *nCode* pen looks like a normal ballpoint pen, but the camera built into the pen reads *nCode* and enables a user to interact with the paper. In this case, each *nCode* represents the location on the paper, and thus the pen tracks the user's pen movements. In terms of similar technology, Anoto technology has been widely used. Another type of pen-and-paper technology is the ultrasonic technology. Unlike the *nCode* technology requiring paper with special patterns of *nCode* on its surface, the ultrasonic technology is applicable to any kind of paper. However, in spite of no restriction in choosing the paper, this technology is less flexible to use because it requires calibration before drawing, which forces people to avoid moving the paper after the calibration.

Existing Prototyping Systems

A growing number of articles have addressed the concept of interactive prototyping. SILK has shown that how to support designers in the early stage of the design process, when they prefer to sketch early ideas on paper [2]. SILK uses an electronic pad and a stylus that preserves the feeling of pen-and-paper interaction, and with these familiar items, designers can quickly draw and modify the user interface sketches. Furthermore, they can illustrate how the user interface screen will change and transform the screen sketch into interactive prototypes. These help designers test the user interface in a running mode on a computer with real users. DENIM takes many concepts from SILK and extends them to website design [3]. However, while SILK focuses on a user interface sketch in detail, DENIM covers the entire design process by supporting various levels of refinement, from design details to website navigation.

There are several systems that build a prototype while preserving designers' sketches on paper. For instance, Bolchini *et al.* proposed a 'Paper in Screen' prototyping method. They photograph each user interface sketch, send it to a mobile device, and crop it to match the device's screen size [1]. This prototyping method does not provide a user interaction method, but it enables users to complete a prepared task within the actual context of usage by ordering pictures of a sketch in a device's photo album. The POP added more functionality to the 'Paper in Screen'-like method. With POP, designers can assign tap interaction areas on the sketch photo and link the area to other photo This method allows test users to interact with paper prototypes on actual devices. MobiDev, originally built to support the development of an application in emerging countries, utilizes vision technology when building interactive prototypes [8]. It also captures mobile UI sketches and makes links to each photo using computer vision technology. With the MobiDev, users can convert a paper prototype into a high-fidelity prototype.

UISKEI is an on-screen sketch-based prototyping tool that supports early stages of the design process. Like other computer-based prototyping tool, UISKEI allows designers to draw user interface. With the computing power, UISKEI convert the rough user interface sketches in to refined user interface element. Moreover, UISKEI supports more complex user interface navigation by an event-condition-action structure, which let designers to define user behavior upon prototype [7].

As presented, previous researchers have proposed a way of building prototypes using photos or digitized sketches. These approaches require extra works for photographing or digitizing of the interface sketches. Moreover, in order to make the drawings interactive, designers have to work on prototyping software.

PRELIMINARY STUDY

To understand how to support designers' work on prototyping, we first conducted a preliminary user study to figure out how designers use sketches when building wireframe prototypes and to determine the primary factors in prototyping.

Method

Six designers from four different companies participated in this research (F=4, M=2). Participants were 27–30 years of age and had at least one year of experience in either the user interface design or user experience design field. Participants were asked to complete two tasks. First, they drew 5 to 7 mobile application user interface sketches that they had designed before. Then, they made a paper prototype on the basis of their first task output. After completing the tasks, follow-up interviews were carried out. All interviews were semi-structured and included questions about the prototyping. For example, *"Can you tell me when to use paper prototyping?" "What kinds of tools have been used instead of paper prototyping?"* and *"What is your purpose for paper prototyping?"* All the observations and interview data were transcribed to describe designers' actions and verbal utterances in detail.

Findings

When drawing user interface elements, designers expressed the user interface abstractly using a rectangular-shaped

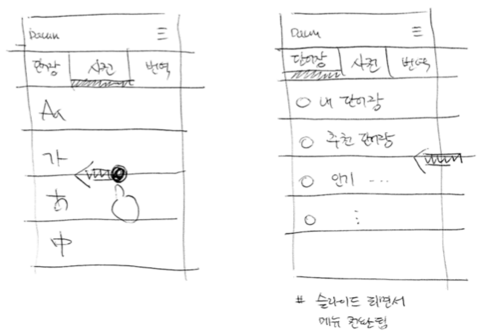

Figure 1. A sample of wireframes that present swipe user interaction and a user annotation..

element. Since the object of prototyping in the early phase was not inspecting design details, they did not represent the user interface layout in detail. Rather, designers emphasized the user interface flow. They focused on the validity of the information architecture or ensuring that the transition interaction is acceptable to users. For these reasons, as shown in Figure 1, designers tended to represent the user interaction with the tap interaction. Although they considered other interactions at that moment such as long press or swiping, they did not illustrate such interactions. Instead, they represented other interactions by sketching button elements or annotating them in spare parts of the template page (Figure 1). For example, they illustrated the swipe gesture by drawing two brackets on both sides of the UI.

In the sketching activity, designers represented the user interface flow by marking arrows between each user interface screen. Designers represented the relationships between user interface elements that should be touched and the user interface screen that would show when the corresponding user interface element was touched. By designing user interface flows, they were able to draw an outline of the user interface design (Figure 2).

In the user interviews, most of the designers said that they built a low-fidelity prototype to check the user interaction flow before moving on to the next phase in the early design stage. This finding forced us to provide a function that is similar to the zooming function in DENIM [3] to review an overall user interaction flow.

Moreover, designers said that they did not use professional prototyping tools often. Although all of them agreed that they draw user interface sketches on paper at first, they use tools like PowerPoint or Keynote for prototyping in the end. One of them said:

"We know that the prototype made by the PowerPoint software was not good enough to accurately represent users' interactions. The PowerPoint documents, however, are much more comfortable to share among team members, even if the team members are far from being designers." – Subjects 1 & 5

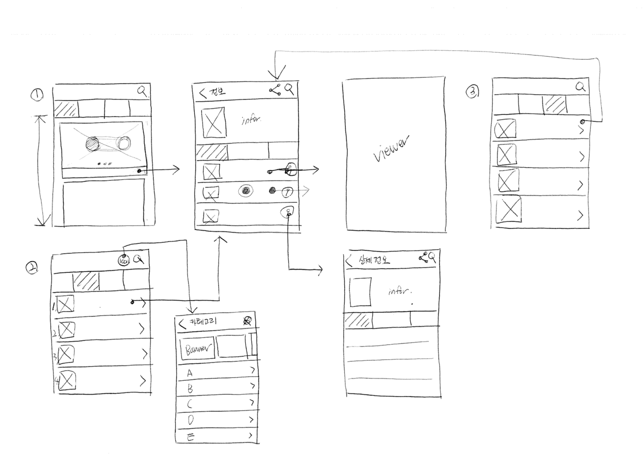

Figure 2. A wireframe shot with interaction navigation.

Because most designers build prototypes when they work with team members or clients, sharing and distribution are decisive. Thus, if a certain prototype were difficult to share or deliver, they would not use it. The designers interviewed said that the paper prototype is not suitable to accurately represent users' interactions. With PowerPoint, designers could present the interactions using the animation effect embedded in the PowerPoint tool; however, it takes too long to present the animation effects. The following quote from one designer underlines this finding:

"Although I worked with PowerPoint, both PowerPoint-based prototypes and paper-based prototypes were not enough to give me a true user interaction experience; thus, I did not do a user test without an interactive prototype that engineer built." – Subject 6

Based on the observations and interview results, we developed two major design guidelines for the prototyping tools. First, the tools should support the interactivity of sketching behaviors. Second, the prototyping output needs to be shared easily among team members.

SYSTEM DESIGN

The prototyping tool consists of the *nCode* pen called Neo1, a template paper, and an Android-based mobile application for mobile prototyping. Once the designer sketches user interfaces and draws screen transition interactions, every stroke in the paper is transmitted to a connected mobile application, and when the designer pushes the convert button, all the strokes and interactions are presented on his mobile device. Then, the designer is able to test his sketches in a real context.

Interaction with Paper

To assist the designers to build an interactive prototype using their natural skills in drawing with pen, we need to define a way to interact with paper first. Since the Neo1 pen

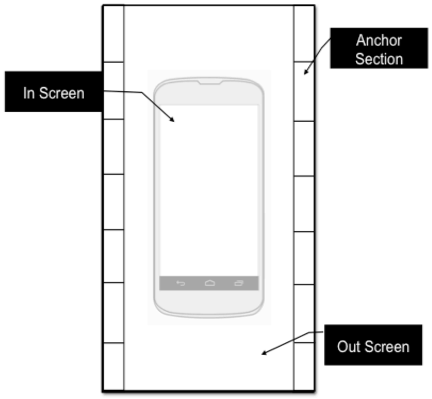

Figure 3. A template paper design.

can interact with the *nCode* printed paper, we decided to design a template paper, which is divided into three areas: a design area; In-screen, an extra area; Out-screen, and an anchor area. Each area supports a specific purpose for designers' pen behaviors. For instance, the design area, which has a similar size to a real device, supports designers' user interface sketching behaviors and the anchor area supports their linking behaviors (Figure 3).

However, even if the paper area has certain attributes, to clarify what a designer's stroke is intended to be, we need to classify the designer's sketching intent. Thus, we implemented two pen-mode concepts: a drawing mode and a linking mode. The designer simply changes the mode by tapping the pen tip in the extra area of the template page. Depending on the pen mode, designers' pen behaviors are processed differently even if they draw on the same area of the paper. If they draw strokes in the drawing mode, the application recognizes that the strokes are related to the sketching actions. On the other hand, with the linking mode, the mobile application classifies input strokes as related to linking actions. By combining the pen mode with the template paper, we defined pen-and-paper interactions.

Pen Interaction

From the observation above, we simply define a user's interaction with paper into two types of action: sketching and linking. A sketching action is a normal action in which the designer draws his/her user interface on paper or annotates it on a redundant space. A linking action is a connecting action that makes a link between user interface elements on one paper with another paper.

Sketching Actions

Sketching user interfaces is the same as the natural pen behavior that designers practice every day. Using the drawing mode on the pen, designers draw whatever they want to express on the design area. Sketching outside the design area or across the design area and other areas does not affect the digital output; however, it remains on the physical paper, so the designer can utilize the extra area as an annotation section.

Linking Actions

Linking actions are more complicated than sketching actions; thus, we separate them into three types of sub-actions: boxing, anchoring, and connecting. Only the exact sequence of these three can make a link between an element and other screen sketches. Basically, when drawing user interfaces, it is impossible to distinguish a clickable user interface from others because we do not support a gesture recognition technique. Therefore, we make a chain of actions to support a linking action:

1. First, designers perform a boxing action on the design area. Since boxing is the behavior of selecting a specific area, we defined boxing as when designers draw rectangular or mark brackets at the top-left and bottom-right corners of the design area (Figure 4-b).

2. After designers set the box for clickable area, they perform an anchoring action. By drawing a line between the box and the anchor sections, a touchable area is anchored to the anchors. Currently, there are only seven anchors on the template paper, so the designer makes seven links per one

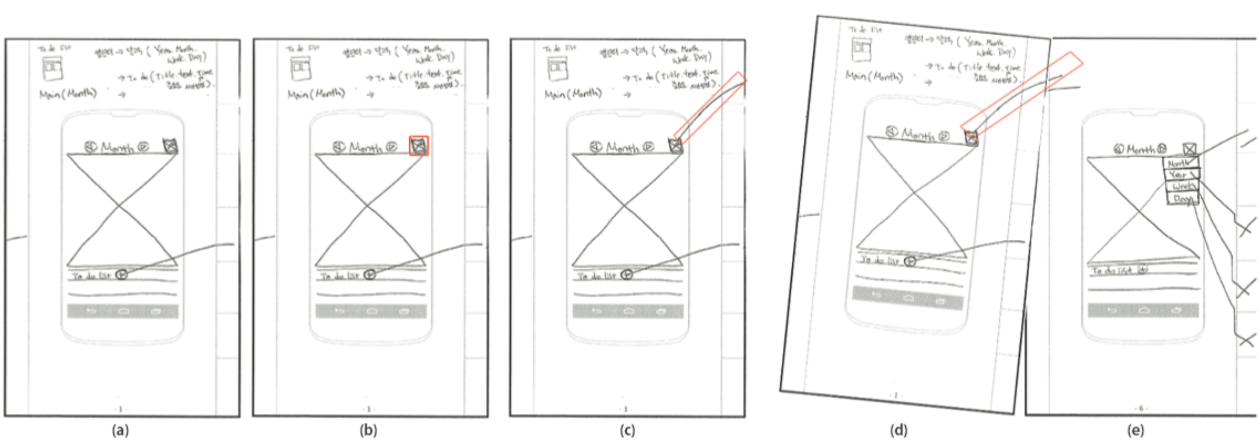

Figure 4. The sequence of linking user element and other sketch. (a) drawing user interface, (b) draw a box, (c) linking touchable area and one anchor, (d-e) draw a line to link the anchor with other sketch

paper only (Figure 4-c).

3. Finally, designers overlap the other template paper, which will be the next screen when clicked, and draw a line across the anchor area and the extra or anchor area on the other paper (Figure 4-d,e).

If designers fail to complete the above interaction sequence, the linking action will be cancelled and they will have to restart the whole sequence from the first part. From this perspective, learning this sequence might be the major challenge in using our prototyping tool.

USER EVALUATION

The main goal of the user evaluation was to assess the ease of using the prototyping tool and the interaction technique for paper linking. User satisfaction was measured using questionnaires to examine whether the prototyping tool is acceptable by designers.

Process

Eight participants (F=7, M=1) were invited to the evaluation study. All of them have a minimum of one year of user interface design experience in their profession (Avg. = 2.5). Five of the eight were considered as having user interface design expertise and three had user experience design expertise.

Each participant performed one training task and two main tasks in a controlled environment. One video camera and one voice recorder had recorded participants' behavior, and no time limit on the experiment was enforced. Each participant read the basic information about the evaluation first. Then, the study began by having each participant sign the consent form. The evaluation session started with a training session. In the training session, participants were asked to complete several tasks to become accustomed to using the prototyping tool. Following the training session, the two main tasks were conducted, consisting of the following: The first main task was to make an existing mobile application user interface prototype with our tool. Seven pictures of the existing application, which have screen transition marks, were provided to each participant. The second task was to design and prototype a user interface of a schedule application. There was no restriction to the design details, but participants were asked to limit their number of sketches to 5 to 7.

Finally, we administered questionnaires to obtain the feedbacks on using the prototyping tool. The questionnaires were designed referring to the post-study system usability questionnaires to evaluate the prototyping tool. The questionnaires contained three items to gather demographic information and 18 items to measure the usability of the application across the following factors: ease of learning, satisfaction, usefulness, ease of use, and level of fidelity.

RESULTS

The study results are classified in terms of three main objectives:

- Sketching user interfaces: To gather data on the drawing of user interfaces or drawing behaviors.

- Pen mode change: To gather data on pen usage.

- Navigating user interfaces (Screen transition): To gather data on the interaction technique or behaviors during the building screen transition interaction.

Sketching User Interfaces

Since participants used a Neo1 pen, which acts like a normal ballpoint pen, they had no difficulty in drawing a wireframe. However, the participants indicated some weaknesses of the current prototyping tool. Most of them said that when they sketched the tap-like user interface, it was tedious to sketch the same user interface element repeatedly. They said that it would be preferable if there were a way to use a predefined user interface element, but this comes from pen-and-paper interaction. To overcome this, a computer-based tool or hardware assistance is required. In addition, they pointed out that the current system did not support revising the wireframe.

Many designers mentioned that they were frustrated about whether their sketches were transmitted properly into the mobile device. Some participants pushed the convert button very often to check their user interface sketches. We conclude that this result came from the fact that there was no feedback in our system and they were pressured to avoid failure because they could not revise their sketches.

One interesting result is that participants looked like they were using a digital tablet device. They checked the mobile device often and paid attention to their sketching like designing visual details not like rough sketch. We found that the difference between the participant who appeared to be using a tablet device and the other who seemed to be only paying attention to the paper was explained by the distance between the participants and the mobile device. In latter case, one participant placed the mobile device farther away from himself. In the follow-up interview, he said the reason he put the mobile device farther way was so that he could focus more on sketching on the paper.

Pen Mode Change

Since it is natural for designers to draw with a pen, there was no feedback about the pen interaction during the drawing. Therefore, the only feedback regarding the use of a pen pertained to the pen mode change.

To navigate user interfaces, participants had to change the pen mode from drawing to linking mode. The changing motion, tapping the paper, is quite intuitive, but the participants felt that changing the pen mode was not convenient at first. However, they were able to master this interaction only after several attempts. Once they learned

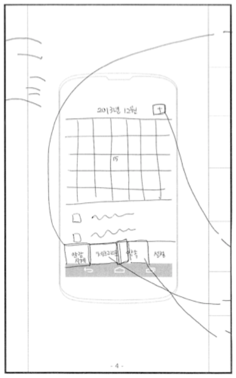

Figure 5. User interfaces wireframe with multiple link area.

Figure 6. User interfaces with many exit points. User can move to other screen either by clicking 'OK' or 'Cancel'

how, they had no difficulty in changing the pen mode, but recognizing and remembering the pen mode consistently distracted them. Sometimes participants forgot what the pen mode was, especially when they were focused on sketching. In the interviews, they said recognizing the pen mode continuously was challenging. They also had difficulty in annotating in the extra area. Because we set a one tap-motion to change the pen mode, annotating in the extra area led to many changes in the pen mode. For example, when they made an annotation, participants may not know what the pen mode is now.

Navigating User Interfaces

The steps of navigating user interfaces were very difficult to learn. Among the interactions related to navigation, anchoring and connecting were the most confusing. One of the participants said that unlike drawing, anchoring and connecting were not motions he/she had done before when drawing a wireframe.

The output lines of navigating activities became a drawback in the system. Most participants were annoyed with the lines. After they made link-lines on sketches, the lines made the sketches messy, as shown in Figure 5. The anchor sections are located on the right edge of the template paper, so users should draw a line across the sketch when linking the right-positioned user interface element. Although they can make links by drawing link-lines around the device picture in the template (Figure 5), these lines also make the sketch messy overall.

In prototyping a schedule application, many participants were faced with the reusable problem. A dialog user interface or tabbed navigation user interface can be presented many times, however since the paper cannot be duplicated, designers should redraw the similar user interface as many as they needed. For instance, when drawing 'add task' or 'add schedule' user interface, either 'OK' or 'cancel' button were linked to the same screen, however, in the current system, the anchors can have only one link, thus, participants have to draw the same user interface to navigate user interfaces exactly (Figure 6).

Discussion & Limitations

Although the entry barrier of our prototyping tool is somewhat higher, the test users were satisfied at the end of evaluation. However, the tool still has a number of problems in terms of usage.

We supported natural pen behaviors; thus, the designers did not have any trouble in sketching user interfaces. They could build their prototypes by drawing as usual; however, this is also a disadvantage of the current prototyping tool. Since the paper cannot be easily duplicated, the designers had trouble with repeatedly drawing the same user interface element, whereas the other computer-based tools, such as Balsamiq or AxureRP, supports this copy-and-paste functionality. This may lead to difficulties in an iterative design cycle.

During the tasks, we expected that the users would feel free to draw and build an interactive prototype, but some designers expressed anxiety in prototyping. This finding does not fit with our expectation of low-fidelity prototyping. Since the final output is not only the paper itself, but the running prototype on a mobile device, participants wanted to get feedback from the mobile device and check whether they failed when sketching or not. This may limit designers' creativity. However, some did not behave like this; instead, they drew a wireframe on the paper. The difference between the two groups was the distance between the mobile device and the participant. We conclude that the distance may affect them, so it needs to be tested in future research.

Although the participants felt comfortable at the end, they mentioned the initial inconvenience of our prototyping tool. The linking actions were the most uncomfortable features for them. Although they used the pen and linked the user interface sketches by drawing, they do not engage in this action every day and was thus unfamiliar. Although, they learned how to do this eventually, it still confused them. Therefore, the interaction techniques, particularly regarding linking, have to be reconsidered; it makes learning difficult and even makes the output paper messy. During the follow-up interviews, participants suggested different types of linking actions, such as sticker paper method or adopting pen gesture recognition.

CONCLUSIONS & FUTURE WORKS

User studies showed that designers were unwilling to make screen transition between sketches by draw lines across papers because the lines for linking makes the sketches messy. In order to avoid these problems, designers have to draw a linking line around the sketch area. We have explored the methods for solving this inconvenient. One way of solving the problem is using stickers instead of drawing a linking line. Stickers are split into the two that are attached both sides of the sketches. Designers overlap stickers with each sketch and draw a line.

One comment made on the post-interview is that participants were not sure that this tool is useful in a collaborative work. They agreed upon the fact that they can quickly convert the sketches into mobile and test the design on the device, however, since revising and reusing the design are difficult, it may be not that useful. Therefore, follow-up research needs to be conducted in co-designing situations, where designers communicate with co-workers and stakeholders. We believe that this future research will provide additional insight into how interactive prototyping tool can be used during the design process and what features the team members needs.

This paper presents a pen-based prototyping tool that builds an interactive mobile prototype from paper sketches. The tool is designed on the basis of the observations of a wireframe prototyping and provides many advantages of low-fidelity prototyping while offering fundamental level of touch interaction.

The prototyping tool is implemented on top of the .Code system that can track the movements of the designers' pen strokes on paper. Based on the preliminary study, we design tool that support designer's natural behaviors like as if they build a paper prototype by sketching.

We performed user studies with eight professional designers, showing that learning curve in using the prototyping tool should be declined and that there is considerable merit in the tool that supports medium level of fidelity. The designers were satisfied about quick converting from the rough sketches and achieving screen transitions through touch interaction on mobile device.

REFERENCES

1. Bolchini, D., Pulido, D., and Faiola, A. FEATURE "Paper in screen" prototyping. Interactions 16, 4 (2009), 29.
2. Landay, J.A. SILK: sketching interfaces like krazy. (1996), 398–399.
3. Newman, M.W., Hong, J., and Landay, J.A. DENIM: finding a tighter fit between tools and practice for Web site design. CHI '00: Proceedings of the SIGCHI conference on Human Factors in Computing Systems, (2000).
4. Nielsen, J. Iterative user-interface design. Computer 26, 11 (1993), 32–41.
5. Rettig, M. Prototyping for tiny fingers. Communications of the ACM 37, 4 (1994), 21–27.
6. Rudd, J., Stern, K., and Isensee, S. Low vs. high-fidelity prototyping debate. Interactions 3, 1 (1996).
7. Segura, V.I.C.C.V.B., Barbosa, S.D.J., and Simões, F.P. UISKEI: a sketch-based prototyping tool for defining and evaluating user interface behavior. ACM (2012), 18–25.
8. Seifert, J., Pfleging, B., del Carmen Valderrama-Baham ondez, E., Hermes, M., Rukzio, E., and Schmidt, A. Mobidev: a tool for creating apps on mobile phones. ACM (2011), 109–112.
9. Synder, C. Paper prototyping: The fast and easy way to design and refine user interfaces. Newnes (2003).