

Decoded: Exploring user involvement in the early stages of software development

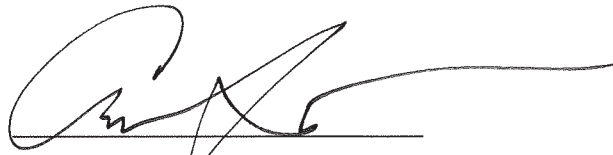
Edward Siefert

Submitted to the faculty of the Herron School of Art and Design
in partial fulfillment of the requirements for the degree
Master of Fine Arts in Visual Communication Design
Herron School of Art and Design
Indiana University
May 2017

Decoded: Exploring user involvement in the early stages of software development

By
Edward Siefert
Master of Fine Arts

Herron School of Art and Design



Aaron Ganci, Assistant Professor
Advisor



Chad Eby, Assistant Professor
Committee Member



Terri Wada, Adjunct Instructor
Committee Member

Accepted May 2017

Professor Valerie Eickmeier
Dean of Herron School of Art and Design

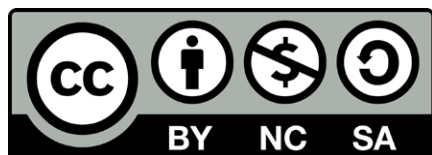
Date

8/5/2017

DECODED

Exploring user involvement in the early stages of software development.

By Ed Siefert



This document is protected under the following terms:

Attribution — You must give appropriate credit, provide a link to the license, and indicate if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.

NonCommercial — You may not use the material for commercial purposes.

ShareAlike — If you remix, transform, or build upon the material, you must distribute your contributions under the same license as the original.

No additional restrictions — You may not apply legal terms or technological measures that legally restrict others from doing anything the license permits.

Abstract

This research aims to explore how user involvement in software development can contribute to innovation in interfaces and system functionality as well as create supporting literature for human-centered design in the software development process. To achieve this, a mixed-methodology approach is used to validate users as co-creators in the early stages of development. This is done through direct engagement with users, the adaptation of the GOMS framework to develop human-centered methods for engagement, and the use of evaluative surveys. By combining GOMS and human-centered design, researchers could frame engagement methods for the elicitation of system functionality and interface design requirements. Researchers then synthesized requirements from user generated data, developed a prototype, and compared it to a prototype developed without user involvement. Early results show that user-generated prototyping provides key insights into the development of software features, user flow, and information architecture.

INDEX

EX

Introduction	4
Objectives	5
Justification	6
Background	12
Research Questions	31
HM Statement	32
Hypothesis	38
Approach	42
Methodology	48
Findings	76
Discussion	78
Limitations	85
Next Steps	87
References	88
Appendix	91

Objectives

This research aims to accomplish three objectives. One, measure the differences between the same product with different levels of user involvement through user's hedonic and pragmatic perceptions. Two, explore whether users can become co-creators of software through their involvement in the creation of low-fidelity prototyping during the early stages of software development. Three, add to literature of user-generated prototyping, by providing a case study of its implementation and its use within the development process.

Introduction

User involvement is a growing field of research in many areas of design. However, there is little research of users as co-creators of software products. This thesis aims to explore the user's ability to contribute to the development of software by involving users in a creative role within the development process. This is done through direct engagement of end users and allowing them to design interfaces and user flows.

JUSTI-

FICA-

TION



User Involvement In Software

User involvement in software interface design is an understudied topic in academic literature and professional conversations. As of 2008 conversations revolving around the subject still focused on whether users should be involved or not.⁽²⁾ As concepts of participatory design have become more accepted, the conversation focused on users as evaluators of products.⁽⁷⁾ This was a good step toward a true implementation of a participatory design methodology. But, it also shows the limited extent to which the democratizing principals of participatory design are applied in the field of software. The shift in conversation from “should users be involved” to “how users should be involved,” presents an opportunity to explore the topic further and see if higher levels of user involvement equate to improved products.

2 Jun He, William King, The Role of User Participation in Information Systems Development: Implications from a Meta-Analysis, Journal of Management Information Systems / Summer 2008, Vol. 25, No. 1, pp. 301–331.

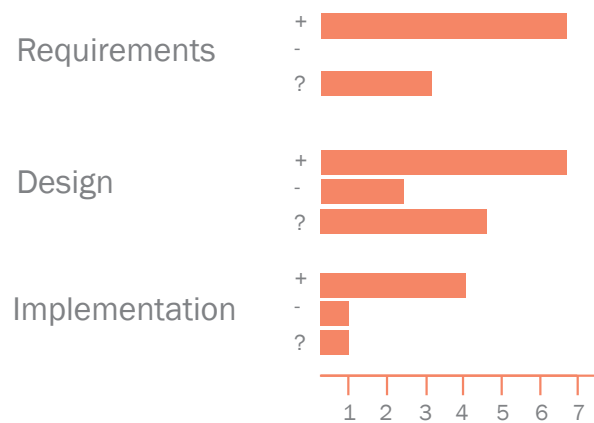
7 Fischer, G, Beyond “Couch Potatoes” From Consumers to Designers and Active Contributors, First Monday, 2002

The connection between user input and project success is well-documented within software design. A study by Butt et al, conducted in 2012 reviewing over 80 projects showed those with user involvement had more long-term success than those without user involvement.⁽¹⁰⁾ The long-term success of software projects is also confirmed by many authors in the field of software design including Cooper and Norman who state that user involvement in software design is critical to knowing the users' goals and understanding the mental models of users.^(19,22)

10 Bano, Muneera, and Didar Zowghi. "A systematic review on the relationship between user involvement and system success." *Information and Software Technology* 58 (2015): 148-169.

19 Cooper, Alan, Robert Reimann, and David Cronin. *About face 3: the essentials of interaction design*. John Wiley & Sons, 2007.

22 Hutchins, E. L., and J. D. Hollan. "D. Norman (1986). *Direct Manipulation Interfaces*." *User centered system design* (81): 87-124.



**AN ANALYSIS OF PROJECTS INVOLVING
 USERS AND THEIR OUTCOMES.
 THE GRAPH MARKS WHERE USER
 INVOLVEMENT OCCURRED IN THE
 DEVELOPMENT AND HOW MANY
 PROJECTS WERE ANALYZED**

Current literature supports the role of users as evaluators and using their input in the software design process. What is limited in software literature is the documentation and evaluation of projects involving users as co-creators. From the literature that does exist, there is a strong opposition to users taking an active role in the creation of interfaces. Cooper even states that users should never become designers.^(1,9) Some practitioners even refer to user co-creation as unethical, as it may allow designers to avoid their responsibilities. This may be indicative of the segmentation of software design that Cooper speaks of. In the book *About Face*, Cooper mentions that there is often little cross collaboration in the development teams.^(1,9) Based on the literature, this stark segmentation of roles has quite possibly become dogma in regards to users and their role in the development process. However, most of the rationale for this is based on anecdotal evidence with few documented studies to show as evidence.

There is opposition to users as co-creators and literature on the user co-creation in software design is limited. However, we see evidence from other fields of design that support the user co-creation process.⁽⁴⁾ There have been several papers that show prototyping with users can produce insights.^(1,10) This process has even been implemented in many projects.^(1,4)

1Kujala, Sari, *user involvement: a review of the benefits and challenges*, *Behavior and Information Technology* vol. 22 no. 1, 1-16, 2003

3Nicolas Ind, Nick Coates, *The meanings of co-creation*, *European Business Review*, Vol. 25 Iss 1 pp. 86 - 95

4Sanders E., Stappers P., *Co-creation and the new landscape, co-design*, 2008



**SOFTWARE DEVELOPMENT OFTEN SEES
CLEAR DISTINCTION OF ROLES, WITH LITTLE
COLLABORATION AMONG THE SEGMENTATIONS.**

BACK.

GROW

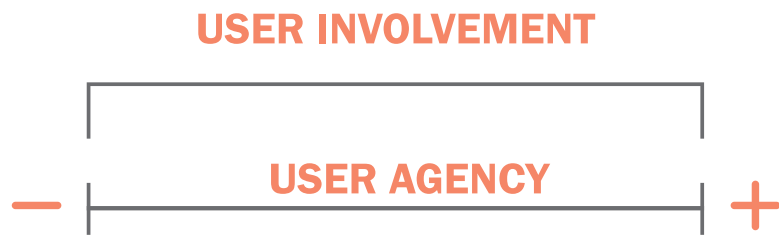


IND

User Involvement as a Broad Term

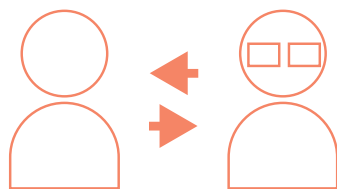
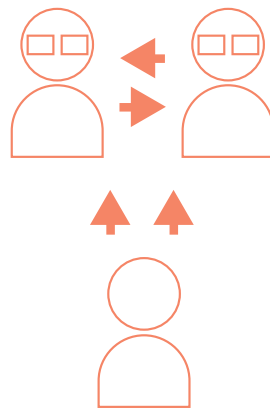
In a research study of co-creation projects, Sari Kujala analyzed the landscape and benefits of user involvement.⁽¹⁾ Though Kujala, states that there are difficulties that are involved with user involvement including the expenditure of time and finances. However, she states that difficulties are offset by the benefits.⁽¹⁾ These benefits are shown in the higher sales rates of products that are developed utilizing participatory methods.⁽¹⁾ In her research, she found that utilizing participatory methods lead to many technical improvements and better meeting of user needs. She shows that users' needs, when engaged in the design process, are better met through requirement elicitation. Though these elicitations can take many forms, actively having users express their needs and wants help designers to form an understanding of users.⁽¹⁾

¹Kujala, Sari, *user involvement: a review of the benefits and challenges*, *Behavior and Information Technology* vol. 22 no. 1, 1-16, 2003



**USER INVOLVEMENT IS A SPECTRUM.
THE MORE INVOLVEMENT USERS HAVE
THE MORE AGENCY THEY HAVE IN THE
DEVELOPMENT PROCESS.**

**IN MANY PARTICIPATORY
DESIGN PRACTICES,
USERS ACT AS SOURCES
OF INFORMATION AND
TAKE A NO ROLE AT
ALL IN DECISIONS OR
KNOWLEDGE CREATION.**



**COLLABORATIVE
DESIGN FOCUSES ON
A DEMOCRATIZED
MEANS OF KNOWLEDGE
CREATION INVOLVING
USER AS OFTEN AS
POSSIBLE**

Kujala notes that user involvement is a broad term that incorporates many methodological perspectives, functions as an umbrella term for engaging participants, and refers to many types of engagement.⁽¹⁾ Kujala sums participatory design as the most basic form of user involvement and only shows a limited concern for users' preferences.⁽¹⁾ However, she states that any user involvement will lead to improved development practices and that it should be adapted.⁽¹⁾

The purpose of user involvement and participatory design was further discussed by Nicholas Ind. His discussion of participatory design focused on how participatory design is a means of building value through shared meaning making.⁽³⁾ Ind introduces

**COLLABORATIVE DESIGN,
IS “KNOWLEDGE AGREED
UPON BY A COMMUNITY,”
AS OPPOSED TO
TRADITIONAL PRACTICE
WHERE “KNOWLEDGE IS
DETERMINED BY ELITES.”³**

Nicolas Ind

participatory design under a broader term of co-creation. However, for the purposes of this research, we refer to co-creation as a form of user involvement. He proposes that user involvement is where companies engage stakeholders to create products and services.⁽³⁾ Ind finds that most users are seen as sources of information as opposed to collaborators, and this limits the value that user involvement can create.⁽³⁾ In Ind's view co-creation should actively involve the user.⁽³⁾ Ind states the trend of user involvement is still developing but is not used to its fullest.⁽³⁾ This limited

involvement from external individuals is what he calls the managerial perspective of co-creation.⁽³⁾ Within the managerial perspective, only people within the firm have an active voice. This is a closed system that focuses internally with limited involvement from users.⁽³⁾ In general, co-creation (or collaborative design) is “knowledge agreed upon by a community,” as opposed to traditional practice where “knowledge is determined by elites.”⁽³⁾

*1*Kujala, Sari, *user involvement: a review of the benefits and challenges*, *Behavior and Information Technology* vol. 22 no. 1, 1-16, 2003

3 Nicolas Ind, Nick Coates, *The meanings of co-creation*, *European Business Review*, Vol. 25 Iss 1 pp. 86 - 95

Participatory Design, Collaborative Design and Democratized Innovation

That top-down means of engagement is standard in software design practices. In most user involvement practices, designers utilize users as sources of information.⁽¹⁾ In design literature, this is the standard form of user involvement, i.e. a passive actor, and is defined as Participatory Design.^(1,7,4) This is a legacy trait from the classical relationship between users and designers.⁽⁷⁾ The perception of seeing little need for involving users as collaborators is relatively accepted, thus users see limited involvement in the design process.^(7,3) Kujala states this is the most common level of user involvement.⁽¹⁾ However, there is a growing demand to increase the role users and laymen play in the design process.^(3,4)

1Kujala, Sari, user involvement: a review of the benefits and challenges, Behavior and Information Technology vol. 22 no. 1, 1-16, 2003

3 Nicolas Ind, Nick Coates, The meanings of co-creation, European Business Review, Vol. 25 Iss 1 pp. 86 - 95

4Sanders E., Stappers P., Co-creation and the new landscape, co-design, 2008

7 Fischer, G, Beyond "Couch Potatoes " From Consumers to Designers and Active Contributors, First Monday, 2002

“USERS CAN PLAY A CO-CREATING ROLE THROUGHOUT THE DESIGN PROCESS, I.E. AS CO-DESIGNERS.”⁴

Sanders and Stappers

Two of the leading voices in the movement of increased user involvement are Sanders and Stappers. In their work, they propose a form of user involvement where users actively participate in the creation process.⁽⁴⁾ In their approach, designers and researchers work to facilitate user creativity.⁽⁴⁾ This level of involvement, as defined by Sanders & Stappers, focuses on active user involvement throughout the process. Sanders and Stappers define this as co-design, or collaborative design, where “designers and people not trained in design work together in the design process.”⁽⁴⁾ This perspective places the user and designer on a more equal level throughout the design process. The authors propose this level of involvement is a stark divergence from traditional

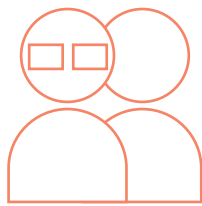
design research.⁽⁴⁾ In their proposition of collaborative design, users actively engage and create with designers and researchers. They state that in co-design “users can play a co-creating role throughout the design process, i.e. co-designers.”⁽⁴⁾

Another voice that is prominent in emphasizing user involvement is Eric Von Hippel. Von Hippel emphasizes the ability of users to innovate independently from external sources and the benefits user innovation can provide a greater community.⁽⁶⁾ Often when speaking of democratized innovation, the term lead-user is used to describe the makers of

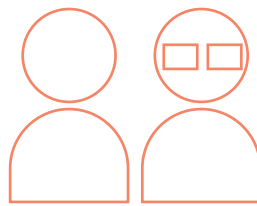
products.⁽⁶⁾ These are individuals who have some technical skill and who create systems for their own needs or in some cases as a hobby.⁽⁶⁾ These users have been enabled through more assessable technology.⁽⁶⁾ Von Hippel states that users are now more empowered than ever in their ability to create products with tools that require less skill to use and have a smaller cost.⁽⁶⁾ This is seen most in Open Source Software where the line between creator and user is often blurred, and lead-users have strong control over the product.⁽⁶⁾

⁴Sanders E., Stappers P., *Co-creation and the new landscape, co-design, 2008*

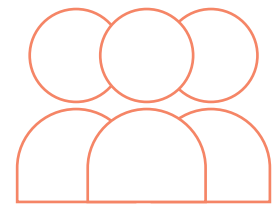
⁶Von Hippel, Eric, *Democratized innovation February 2006*



**Participatory
Design**



**Collaborative
Design**



**Democratized
Innovation**



THE METHODOLOGIES THAT ENCOMPASS USER INVOLVEMENT RANGE IN HOW USERS ARE INVOLVED. PARTICIPATORY DESIGN FOCUSES ON A MORE TOP DOWN APPROACH, COLLABORATIVE DESIGN INVOLVES USER AS COLLABORATORS, AND DEMOCRATIZED INNOVATION IS A USER DRIVEN PROCESS.

Prototyping as a Bridge to Allow Users to Design

Gerhard Fischer proposes that designers must create opportunities for users to actively create.⁽⁷⁾ He states that much of the software created sees people as consumers and this must change through making users co-creators.⁽⁷⁾ Fischer proposes that users want to actively create and be involved in the development process in a meaningful way but lack the tools available to them. For Fischer, designers face the challenge of not

creating programs but providing users the ability and resources to become creators.

⁽⁷⁾ Users innately want to change their environment to suit their needs. This is seen in power users and domain designers, those that possess the skills and ability to actively engage software and alter it to their needs.⁽⁷⁾ Fischer states that software must change to empower users within the development process to facilitate their involvement by containing mechanisms that allow users to gain more agency.⁽⁷⁾

Fischer states that this does not mean that

design is left to users. Instead users should be treated as experts in their domain and be given the agency to change systems to best meet their needs.⁽⁷⁾

USERS SHOULD BE TREATED AS EXPERTS IN THEIR DOMAIN AND BE GIVEN THE AGENCY TO CHANGE SYSTEMS.⁷

⁷ Fischer, G, *Beyond "Couch Potatoes" From Consumers to Designers and Active Contributors*, First Monday, 2002

This involvement of users as creators, has seen many services successfully implemented.⁽⁸⁾ With the collaborative design approach, designers and non-designers work at the front end of design to express their ideas and needs through prototyping.⁽⁹⁾ This form of prototyping allows non-designers to be able to create and develop a means of communication to express their needs⁽⁹⁾ As Sanders and Stappers state the tools provided to users must be able to facilitate their ability to design.⁽⁹⁾ Therefore, the designer's role is not to create but to facilitate the creativity of users. This is echoed by Fischer who says that the role of the designer is to provide users the means to become designers of their own domain.⁽⁷⁾ Despite this challenge we see little effort in the software community to incorporate users in prototyping. This hesitance is shown in the study by Bano and Zowghi, who analyzed 80 projects involving users.⁽¹⁰⁾

TOOLS PROVIDED TO USERS MUST BE ABLE TO FACILITATE THEIR ABILITY.⁴

The study shows that there are various levels of involvement throughout the development process and specifically in the design of the interface.⁽¹⁰⁾ In the study, they provide a list of studies where users were involved in the design of the product interfaces and features.⁽¹⁰⁾ One instance of lack of collaboration in interface development is in a case study by Chamberlain et al.⁽¹¹⁾ Their study typifies the involvement of users in the design of software. Even though they note users have input in the early stages of the planning process, this involvement drops off during the design and development stages.⁽¹¹⁾ In this particular study, once the program started to be developed, users were removed as part of the design process and assumed an advisory role in evaluating the prototypes.⁽¹¹⁾

7 Fischer, G, *Beyond "Couch Potatoes" From Consumers to Designers and Active Contributors*, First Monday, 2002

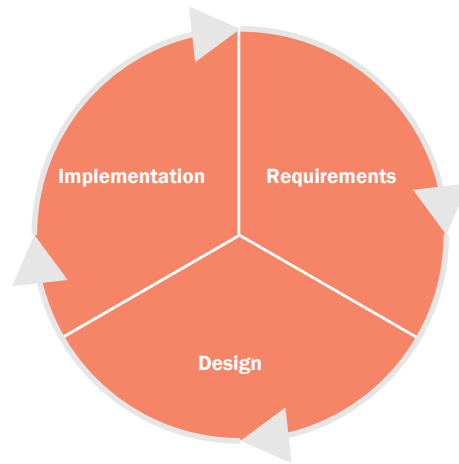
8 Marc Steen, Menno Manschot, and Nicole De Koning, *Benefits of Co-design in Service Design Projects*

9 Sanders E., Stappers P., *Probes, Toolkits and Prototypes: three Approaches to Making in Codesigning, co-design*, 2014

10 Bano, Muneera, and Didar Zowghi. "A systematic review on the relationship between user involvement and system success." *Information and Software Technology* 58 (2015): 148-169.

11 Chamberlain, Stephanie, Helen Sharp, and Neil Maiden. "Towards a framework for integrating agile development and user-centred design." In *International Conference on Extreme Programming and Agile Processes in Software Engineering*, pp. 143-153. Springer Berlin Heidelberg, 2006.

THERE ARE THREE GENERAL PHASES OF DEVELOPMENT: REQUIREMENTS, DESIGN, IMPLEMENTATION. USER INVOLVEMENT CAN TAKE PLACE IN ANY OF THESE AREAS



Requirements Design Implementation

10	14	6
-----------	-----------	----------

BANO AND ZOWGHI, SEGMENTED SEVERAL STUDIES INTO THE DEVELOPMENT PROCESS

User Involvement in Software Design And The Levels of Participation

In their study of user involvement Bano and Zowghi demonstrate how many projects have used user involvement in software development. The authors demonstrate that there are three stages in development where users can be involved. These sections are requirement analysis, design, and implementation.⁽¹⁰⁾ Though the study shows the great breadth of user involvement, it also shows how limited user involvement is in particular stages. This is most clear when looking at user involvement in the design stages of software development.

The lack of user involvement in the design of software was also demonstrated in several other studies identified by Bano and Zowghi.⁽¹⁰⁾ In these studies, user involvement focused on providing feedback of the designs and testing of prototypes as opposed to more collaborative means of involvement.^(11, 12, 13, 14) In the identified studies, the process of involvement was the same; designers and developers provided users prototypes to engage and critique. In one particular study, users were highly involved in the requirement section and early stages of ideation.⁽¹¹⁾ However, when the actual testing of the designs took place users were supplemented with in-house staff, who acted as proxies for users.⁽¹¹⁾ This limitation shows the tentative acceptance of user involvement in the software development process.

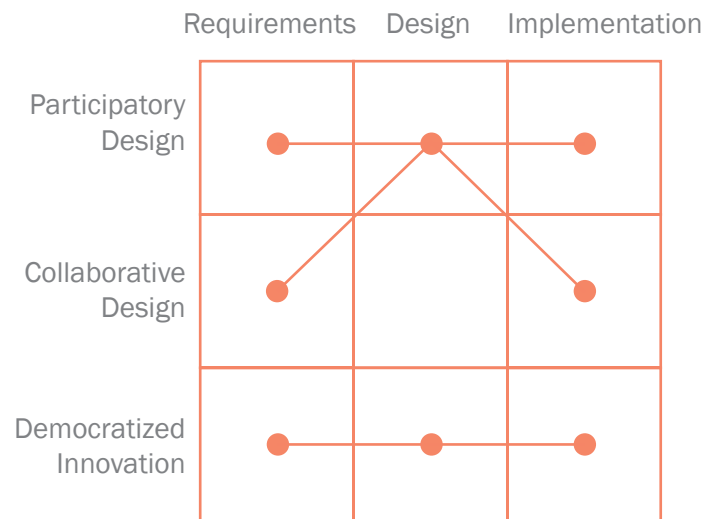
10 Bano, Muneera, and Didar Zowghi. "A systematic review on the relationship between user involvement and system success." *Information and Software Technology* 58 (2015): 148-169.

11 Chamberlain, Stephanie, Helen Sharp, and Neil Maiden. "Towards a framework for integrating agile development and user-centred design." In *International Conference on Extreme Programming and Agile Processes in Software Engineering*, pp. 143-153. Springer Berlin Heidelberg, 2006.

12 S. Pekkola, N. Kaarilahti, and P. Pohjola, "g," In *Proceedings of the ninth conference on Participatory design: Expanding boundaries in design*-Volume 1, pp 21-30. ACM, 2006

13 Wu, J.B., and Marakas, G.M. *The impact of operational user participation on perceived system implementation success: An empirical investigation.* *Journal of Computer Information Systems*, 46, 5 (2006), 127-140

14 K. Kautz, "Participatory Design Activities and Agile Software Development," *Human Benefit through the Diffusion of Information Systems Design Science Research*, pp. 303-316, 2010



WHEN BREAKING DOWN LITERATURE INTO CATEGORIES OF PROCESSES AND DEVELOPMENT PHASES WE SEE A GAP.

User Involvement and Development Models

That perception of user involvement could be related to the lack of a specific process to involve users in development. In many development models, there is no place built-in for user involvement, and it is often added to pre-existing stages. Traditionally, user involvement takes place at two points in a development process during the requirements stage and during implementation.⁽¹⁶⁾

This leads to the main critique of software development models. In Butt and Amad's paper on software development, they state all software development models lack focus on user involvement and user interfaces.⁽¹⁶⁾ They state that these are more afterthoughts within the process and are often left until later. Butt and Amad propose that, for a successful product users must be involved throughout the process and must be put at the forefront of the development process.⁽¹⁶⁾

Inevitably, one limitation to user involvement maybe the process and how viability is measured. There are many ways to measure a product well before the actual coding of the software.⁽²²⁾ Outside of measuring the amount of sales, most product measurement comes well before the product is released using surveys. These early tests help designers and developers to benchmark how users perceive the software. These normally take place in the early stages of development after the design phase. These are sometimes referred to as pre-alpha tests.

¹⁶ Butt, Saad Masood, Wan Fatimah Wan Ahmad, and Wan Fatimah. "An overview of software models with regard to the users involvement." *International Journal of Computer Science Issues (IJCSI)* 9, no. 3 (2012): 1.

²² Hanington, Bruce, and Bella Martin. *Universal methods of design: 100 ways to research complex problems, develop innovative ideas, and design effective solutions*. Rockport Publishers, 2012.

Product Validity

Those pre-alpha tests provide prospective users with low-fidelity mock-ups of the software and uses surveys to measure user perspectives of the of the product. These tests provide developers with quick feedback on the application and allow them to measure viability and hedonic perceptions. Through the user feedback, modification of the prototype can be made and developers can understand how the product may fair in the market. One of the most common surveys to achieve this is the Standard Usability Survey (SUS).⁽¹⁷⁾ Developed by Brooke, this survey is often called quick and dirty as it provides simple questions to participants that provide insight into how a product is perceived by users.⁽¹⁷⁾ The SUS is known for its high validity even at low sample numbers. This validity means it is a highly effective and accurate at measuring the usability of a system.⁽¹⁷⁾ Another important aspect of software is understanding the user's hedonic perception of a product.⁽¹⁵⁾ Through the measurement of these perceptions, developers and designers can gain insight into the software's success. Hassenzahl developed the AttrakDiff to measure four types of product perceptions. Focusing on four categories: how an individual using the product is preserved; how the experience of the using is perceived; how usable the system is; and how attractive the system is.⁽¹⁵⁾ Though it has low internal reliability, its ability to measure four distinct categories of perceptions is helpful when trying to measure a product holistically.⁽¹⁵⁾ The SUS and AttrakDiff are useful because of their designed purpose. Using these tools, a basis for viability and acceptance of a product can be established.

Utilizing these surveys a product can be measured and through a controlled study key questions about user involvement can be answered. The results of these surveys can provide knowledge of the outcomes and help to validate the product against industry benchmarks.

¹⁵ Hassenzahl, Marc. "The effect of perceived hedonic quality on product appealingness." *International Journal of Human-Computer Interaction* 13, no. 4 (2001): 481-499.

¹⁷ Brooke, John. "SUS-A quick and dirty usability scale." *Usability evaluation in industry* 189.194 (1996): 4-7.

TO KNOW IF USER INVOLVEMENT CAN IMPROVE PRODUCTS THESE SURVEYS ARE NEEDED TO ANSWER KEY QUESTIONS. THE RESULTS OF THESE SURVEYS HELP TO VALIDATE A PRODUCT INVOLVING USERS AS CREATORS BY MATCHING IT AGAINST INDUSTRY BENCHMARKS.

RE-

SEARCH

QUES-

CH

TIONS

HOW MIGHT INTERACTIONS WITH DIFFERENT USER INVOLVEMENT TO EACH OTHER ON **HEDONIC** AND PERCEPTIONS?

- Hedonics refers to the product's perceived ability to support the achievement of feelings and status.

Hassenzahl 2008

INTERFACES LEVELS OF COMPARISON BASED PRAGMATIC

Pragmatics refers to the product's
perceived ability to support the
achievement of task based goals.

Hassenzahl 2008

RQ1

DO USERS PERCEIVE PRODUCTS CREATED WITH GENERATIVE USER PROTOTYPING TO BE MORE VIABLE COMPARED TO THOSE WITH NO USER INVOLVEMENT?

RQ2

DO USERS PERCEIVE PRODUCTS CREATED WITH GENERATIVE USER PROTOTYPING TO HAVE GREATER HEDONIC VALUES COMPARED TO THOSE WITH NO USER INVOLVEMENT?

PROCESS QUESTIONS

Is there a noticeable benefit to users having a generative role when designing interfaces?

Does prototyping with users generate unique information for use in interface design and systems functionality?

What differences exist between prototypes made with different levels of user involvement?

HYPOTHESIS

S

H2

Users perceive products created with generative user prototyping to have higher hedonic values compared to those with no user involvement.

H1

Users perceive products created with generative user prototyping to have higher hedonic values compared to those with no user involvement.

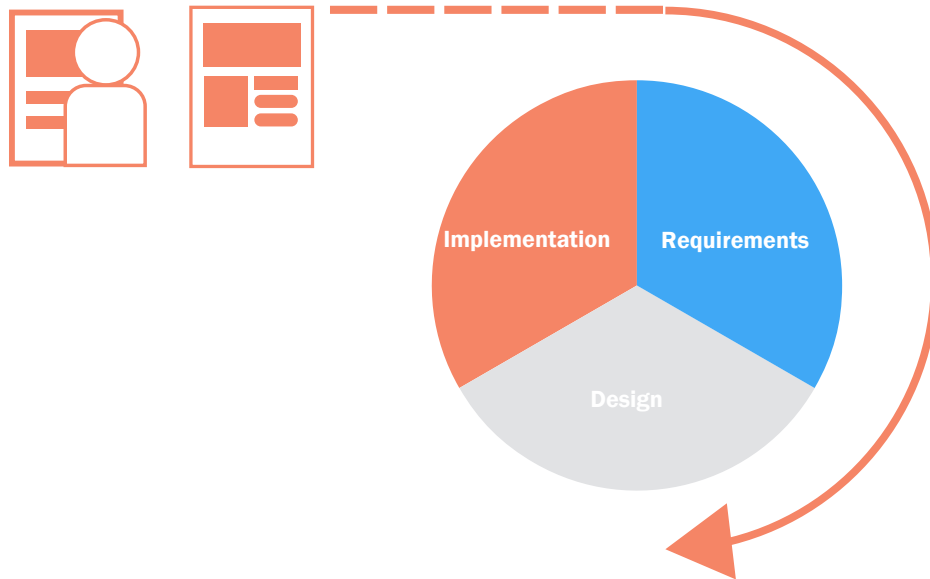
AP.

PROA

АСН

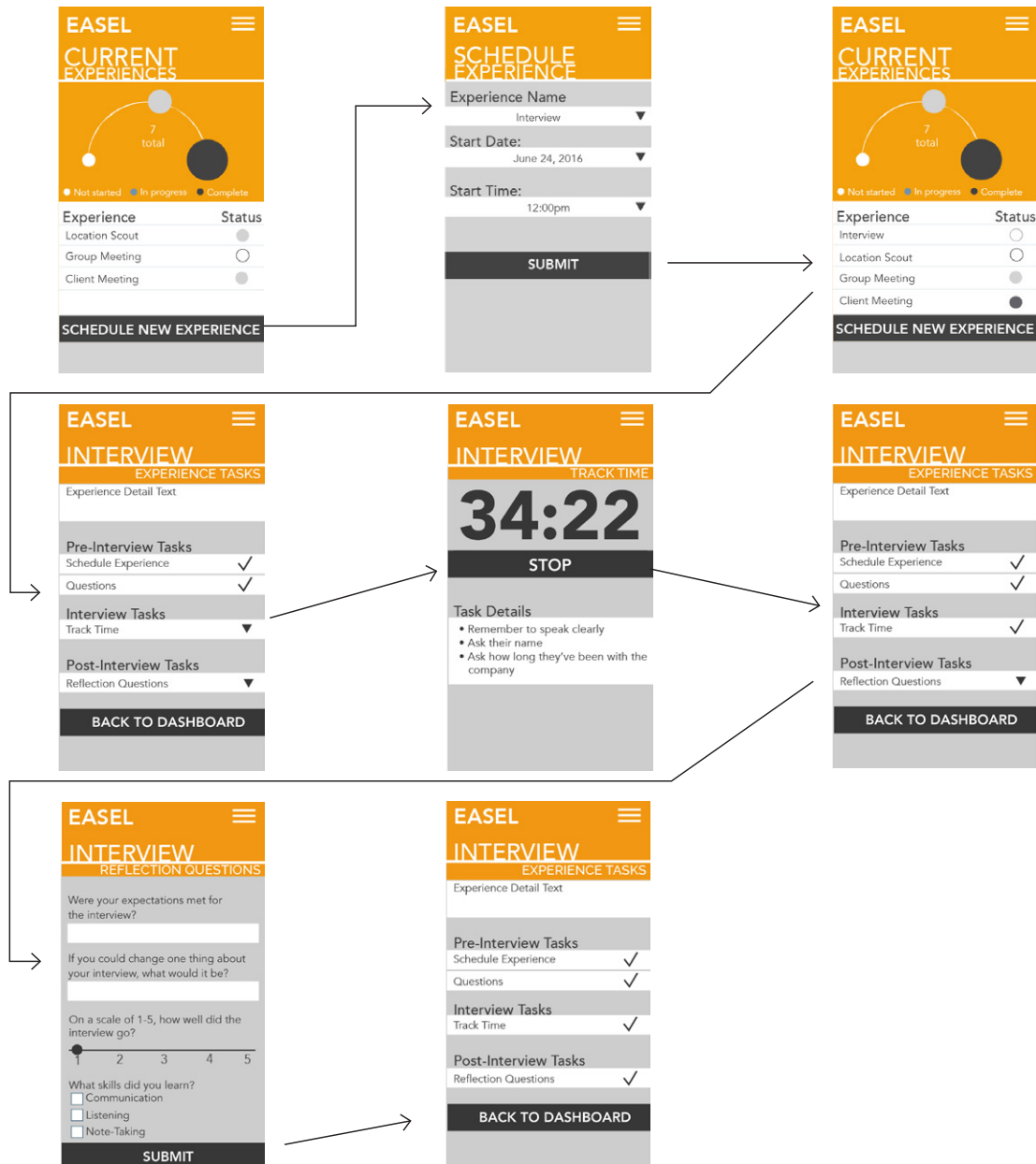
EASEL

To answer these questions and know if collaborative design can affect the outcome of a product, there must be a direct comparison between the same product utilizing different methods of development. Therefore, any changes in perception of usability can be measured. To achieve this EASEL was selected as a candidate for the study. EASEL is a reflective learning application currently in alpha at IUPUI. It serves to assist students' metacognitive abilities by giving prompted reflection before, during, and after experiential learning exercises. This is done through targeted prompts provided to students by educators. This application was chosen because of its stage in the development process. With the application being in the alpha stage of development an alternative interface and user flow could be developed within a short time frame and assist the future development of the application.



THERE MUST BE A DIRECT COMPARISON BETWEEN THE SAME PRODUCT UTILIZING DIFFERENT METHODS OF DEVELOPMENT.

GRAPH OF EASEL USER JOURNEY MAP



EASEL was an ideal candidate for this study, as the development of the application was completed without user involvement. This limitation would allow for measurement of variables in the study. The application user flow and UI were already designed by the developers. The user flow consisted of three main goals: Scheduling, Documentation, and Reflection. In the application, students are given a series of tasks to complete. Students are asked to create a schedule of experiences, selecting from a list of experiences provided by educators and completing a series of reflective prompts before, during, and after the experience. Within these experiences are a series of tasks. Each task must be completed to move to the next task. By completing the sequential tasks within the experience, students can fulfill the requirements set by educators.

There are two main user groups for EASEL. This is because EASEL functions as an intermediary between students and educators. The transaction between the two user groups was defined initially as two-way, with students being given direction from educators to assist in experiential learning.

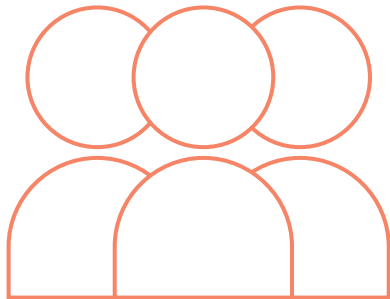
This study focused on engaging student user group. This was for practicality as students were the most assessable user group and would have the least amount of experiences with developing interfaces. The student side of the application was the only interface and user flow which was created. The teacher side of the application was not yet developed.

METHA ODOLE OGY



Overview

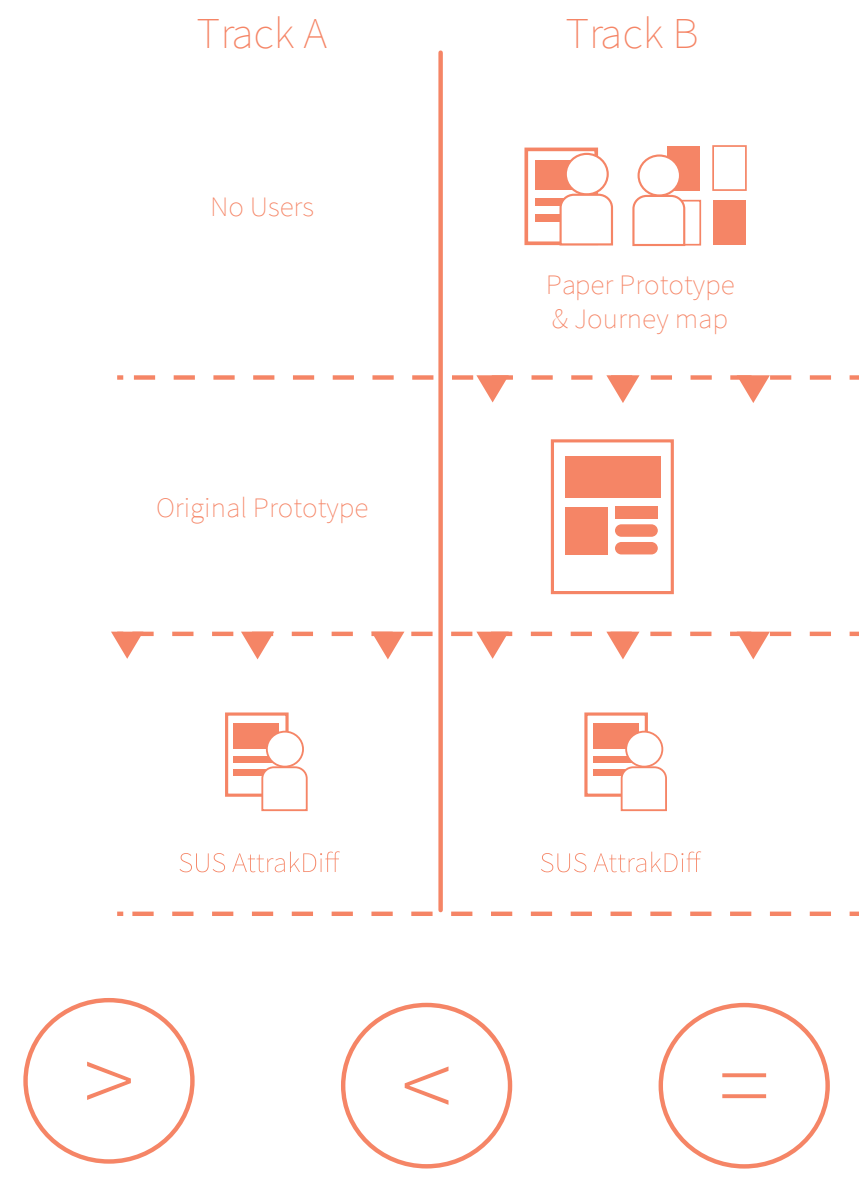
This study focused on evaluating the use of user generated paper prototyping tools within the context of interface design. This was done by adapting several frameworks and methods that already exist in the HCI realm and melding them with a human centered methodology. Specifically, the researcher used a paper prototyping method widely employed by interface designers, Brooke's SUS, and an adapted version of the AttrackDiff scale developed by Hassenzahl. The result being a mixed method study incorporating generative frameworks for the use of user prototyping and a mixed method evaluation of the final interface product.



**38 PARTICIPANTS WERE
RANDOMLY SELECTED
FOR THIS STUDY**

STUDENTS

A FOUR PHASE PROCESS WAS USED TO FRAME THE STUDY



Phase 1 | User engagement

Phase 2 | (Re)design prototype

Phase 3 | User Evaluation

Phase 4 | Results

GOMS

To frame the engagement with users and to better discern what methods to develop for the engagement, frameworks already in use within the realm of HCI were implemented. After evaluating several frameworks, the GOMS framework was selected as it provided well-defined categories for the segmentation of systems and actions within those systems. GOMS was selected over the competing framework used by Cooper due to its ability to scale. Though Cooper's user engagement framework is well-tested, the methodology is highly structured. Cooper's model provides a step-by-step model for user engagement.⁽¹⁹⁾ This structure is both a benefit and a detriment to Cooper's engagement model. This decision was made because of the ability of GOMS to scale well at multiple levels of abstraction, whereas Cooper's framework focused on high level.

THE BENEFIT OF GOMS IS ITS ABILITY TO SCALE.

The benefit of GOMS is its ability to scale.

⁽²⁰⁾ GOMS focuses on four categories (Goals, Operators, Methods, and Selection Rules) that are applicable to a system action and activities outside of the application.⁽²⁰⁾ This scalability of GOMS helps to frame and ladder methods to explore those interactions. This use of GOMS is a divergence from its intended purpose of evaluating existing interfaces. However, the ability of GOMS to classify aspects of users' interactions with systems provides a strong basis for creating interactions with systems from a high level of abstraction. The researchers aimed to utilize this framework to develop methods of engagement with users.

In short, GOMS is a method for evaluating tasks within a system and the user's knowledge of how to perform the task in terms of goals, operators, methods, and selection rules.⁽²⁰⁾ The definitions of these categories are provided below.

¹⁶ Butt, Saad Masood, Wan Fatimah Wan Ahmad, and Wan Fatimah. "An overview of software models with regard to the users involvement." *International Journal of Computer Science Issues (IJCSI)* 9, no. 3 (2012): 1.

²³ Hanington, Bruce, and Bella Martin. *Universal methods of design: 100 ways to research complex problems, develop innovative ideas, and design effective solutions*. Rockport Publishers, 2012.

GOALS

What the user wants, or intends, to accomplish. ⁽²⁰⁾

OPERATORS

The actions users can make in the software. These are most often described as physical actions such as tapping, clicking, spoken commands, or even eye movement. There are many levels of operators and the definition can vary based on the level of abstraction. ⁽²⁰⁾

METHODS

A sequence of operators and sub-goals that assist in accomplishing primary goals. If multiple methods exist, then selection rules are applied to decide what methods should be used. ⁽²⁰⁾

SELECTION RULES

are the personal rules that users follow in selecting methods to accomplish goals. Application of selector rules depends on what the user considers the most effective solution based on their perception. ⁽²⁰⁾

From these definitions, the actions completed by users and systems could be categorized. These were segmented into user actions and system actions. System actions were given the more specific name of functions. Interfaces fell outside of the definition of actions and were classified as features, again drawing from the definitions by Hutchins et al. User actions were then categorized as Operators as defined by the GOMS framework.⁽²⁰⁾ Based on GOMS and how it applies to software, it can be stated that functions of a software and user actions may be considered operators, operators being how a person reaches a goal.⁽²⁰⁾ On the other hand, interfaces can be considered features as they are what allow the user to access the software functions, thereby acting as an intermediary.⁽²¹⁾ This is true when drawing a comparison to other products. Take for instance, an analog telephone. A telephone has three main functions: sending voice, receiving voice, and initiating/receiving a call. These functions are provided to the user through features such as having a number pad to initiate a call, having a microphone to send voice, a ringer to know when a call is incoming, and having a speaker to hear voice. At its very basic application, interface features are there to access functions. Thus, goals are what a person wants to accomplish, functions and user actions act as operators for people to accomplish their goals, and features allow people to access operators needed to accomplish goals.

FUNCTIONS ACT AS OPERATORS FOR PEOPLE TO ACCOMPLISH THEIR GOALS.

FEATURES ALLOW PEOPLE TO ACCESS OPERATORS NEEDED TO ACCOMPLISH GOALS.

²⁰ John, Bonnie. "Why GOMS?." *interactions* 2, no. 4 (1995): 80-89.

²¹ Hutchins, E. L., and J. D. Hollan. "D. Norman (1986). *Direct Manipulation Interfaces.*" *User centered system design* (81): 87-124.

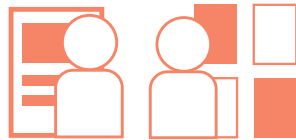
User Engagement Process

Based on that understanding of GOMS, users' interactions with software were segmented into three categories: actions, functions, and features. Actions are physical activities that people take while interacting with the software. Functions are software actions people can use to accomplish their goals. Since the combination of Actions and Functions allow users to accomplish goals they can broadly be considered operators within the GOMS model. Interface elements were then defined as features, categorized outside of the GOMS model, and left separate for the first section of the intervention. This was done to distinguish interface features from software functions and user actions, since the interface only facilitates accomplishing goals by providing a medium between user actions and system functions.⁽²¹⁾ Through this segmentation, the operators (System Functions / User Actions) and features could be looked at independently from each other and laddered later in the sessions. The researcher could then apply the interface and its features to Selector Rules, the fourth category of GOMS. This choice was made as interface design provided the most variation with infinite possibilities of visual outcomes. By sub-setting the intervention into operators (user actions/system functions) and features, the researcher could look at the two categories independently and allow users to decide what features best facilitate their methods and goals. This was decided to separate the participants' correlation between functions and features.

21 Hutchins, E. L., and J. D. Hollan. "D. Norman (1986). *Direct Manipulation Interfaces*." *User centered system design* (81): 87-124.

User Engagement Session

The study involved 10 users to gather requirements and elicit design concepts. These participants were from a Computer Science class from a midwestern university and part of the target user group of EASEL. These users ranged in age and academic standing, but were primarily liberal arts majors. These participants were then asked to complete tasks relating to how they would accomplish the goals of scheduling, documenting, and reflecting.



Paper Prototype Journey map

**10 participants were
engaged to complete a
two stage engagement.**

Journey Mapping

For the first part of the intervention, participants were prompted to provide a combination of operators based on a main goal and series of sub-goals. Participants then selected the methods (operators and sub goals) they would like to use in completing the primary goals. This was done through journey mapping. (22) In common literature regarding user experience design, journey maps show people's preferred or current process through a series of touch points through the use forward chaining. In this study, the researcher asked participants to write, draw, or speak their preferred journey to the three main steps of scheduling, documenting, and reflecting. This data was used to help frame the user's ideal journey through the scheduling, documenting, and reflecting stages of the application.

22 Hanington, Bruce, and Bella Martin. Universal methods of design: 100 ways to research complex problems, develop innovative ideas, and design effective solutions. Rockport Publishers, 2012.

Paper Prototyping

Once participants had defined what methods they would use to accomplish their primary goals, the participants were then prompted to utilize features in the form of paper prototypes to illustrate their ideal interface. This method is not defined in literature as it was used in this study. However, paper prototyping or throw-away prototyping is a well-documented method used by designers and developers to show the general user-flow of software.⁽²²⁾ For this section of the intervention, participants were instructed to utilize the toolkit to design a rough interface. This was done by presenting the participants with the journey maps developed during the previous section of the intervention. Participants were then asked to demonstrate how they would envision an interface to facilitate those methods. The participants were then given a series of paper tools with various interface features and asked to place those features on a mock-up of a device. Participants could use the tools provided or draw their own if they did not see a feature they needed to complete their task. Once participants were finished, they were asked to share their prototypes with researcher and explain the decisions made in their designs. In total, four prototypes were created.

²² Hanington, Bruce, and Bella Martin. *Universal methods of design: 100 ways to research complex problems, develop innovative ideas, and design effective solutions*. Rockport Publishers, 2012.

Data Analysis of Prototyping Session

The combination of functions, actions, sub-goals and methods were compiled into one data set. This was done through transcribing the data onto post-it notes and affinity diagramming.⁽²²⁾

This medium would allow for categorization and assembling of the raw data from the prototyping session into user requirements. The process started with transcribing key concepts and actions used in the journey maps. Each detail given during the prototype session was meticulously written down and organized. These were then sorted based on key concepts of the transcribed post-it notes. The first data set was operators as these presented the most similarities to each other. Operators tended to have specific concepts such as use of word processors. These were then grouped and given a categorical title representing the operators found. Goals were the next category to categorize. These were broader and tended to have less specificity than other data sets. From the analysis, these formed assistive reasoning to the use of operators and tended to focus on concepts of user needs and use context. Once the goals were categorized and grouped by similarity, the researchers focused on unifying the goals and operators into conjoined categories. Through several iterations, the researchers assembled the data into logical groupings. These were then further categorized into subordinate and dominate arrangements by laddering operators and goals defined from the journey maps.

22 Hanington, Bruce, and Bella Martin. *Universal methods of design: 100 ways to research complex problems, develop innovative ideas, and design effective solutions*. Rockport Publishers, 2012.

That process was continued with the paper prototypes. Different color post-it notes were used for each prototype created to help distinguish data sets. This part of the analysis process helped to identify system side operators and helped to confirm task-based goals within the application. Though the prototypes were low fidelity, there were relevant findings for work flow requirements and navigational preferences. These were recorded on post-it notes and then sorted based on similarities. These transcriptions were then categorized. After categorizing the prototype data, the information was merged with the data from the journey maps to look at overlaps and differences of content. After analysis, five distinct categories of requirements were derived from the user prototypes defining functional and experiential qualities. The five categories were search-able contacts, dashboard navigation, integration of current contacts, links to system calendar, and adaptable communication.

Outcomes From Analysis

The findings of this analysis helped to inform the development of two outcomes. One was a new user flow. The second was definition of two heuristic categories (Functional and Experiential Qualities). These would serve to guide the development of a new prototype.

New User Flow

After analysis, a pattern to the journey maps designed by users was discovered. Unlike the original journey defined in the prototype of Schedule> Document> Reflect. Users defined Communication as the first step in their process. This was noted not just in scheduling but also in the other segments of Documentation and Reflection. To reflect the reality of user actions, a Communication goal was added in the user flow and integrated within the new application framework. Therefore, the new user journey was defined as Communicate> Schedule> Document> Reflect. The Communication function was thus a requirement from student users, and Reflection was a requirement from educators.

Schedule ➔ **Document** ➔ **Reflect**

The original prototype had three phases in the user flow.

Communicate ➔ **Schedule** ➔ **Document** ➔ **Reflect**

After analysis from affinity mapping a four phase

Heuristics

Another key outcome of the analysis process was the definition of design requirements and heuristics. These were defined in two overarching categories of Experiential and Functional Qualities. These terms were borrowed from Cooper.⁽¹⁹⁾ The functional qualities focused on three aspects, Media/Communication, Memory System, and Assistive Functions. Experiential qualities focused on use needs, containing two sub-categories of Personalization and Ease of use. These heuristics provided general benchmarks when designing functions in the system and would serve as a road map in the development of an alternative interface.

¹⁹ Cooper, Alan, Robert Reimann, and David Cronin. *About face 3: the essentials of interaction design*. John Wiley & Sons, 2007.

Functional Qualities

Memory System

Informational Assistance

Participant List, Location, Reminders, Notifications, Day of Confirmation, Reflective Prompts, Project Management, Integrated Contact List

Centralized Schedule

Collaborative Scheduling, Link to System Calendar, Shared Calendar

Assistive Functions`

File Management

File Organization, Collaborative Documentation, File Sharing, Archivability, Uploading Files

Accessibility

Auto Summarization of Text, Auto Transcription of Audio, Transcription of Paper Notes,

Media/Communication

Multimedia

Photo collages, Audio, Video Documentation, Photo Documentation, Stylus Function

Text

In-text Notation, External Word Processor, Annotation of Documents

Multilateral Communication

Topical Chat Function, Social Media, Email

Experiential Qualities

Personalization

Adaptability

Customization, Social and Business use

Task Facilitation

Multitasking, Parallel-tasking

Ease of use

Fluidity

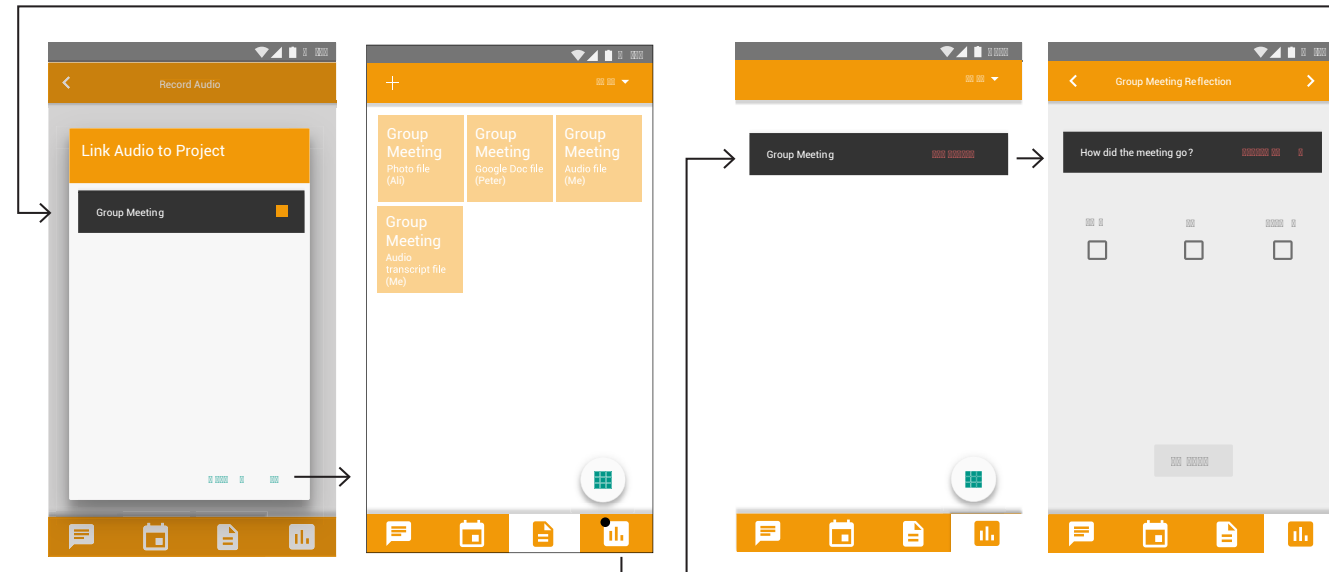
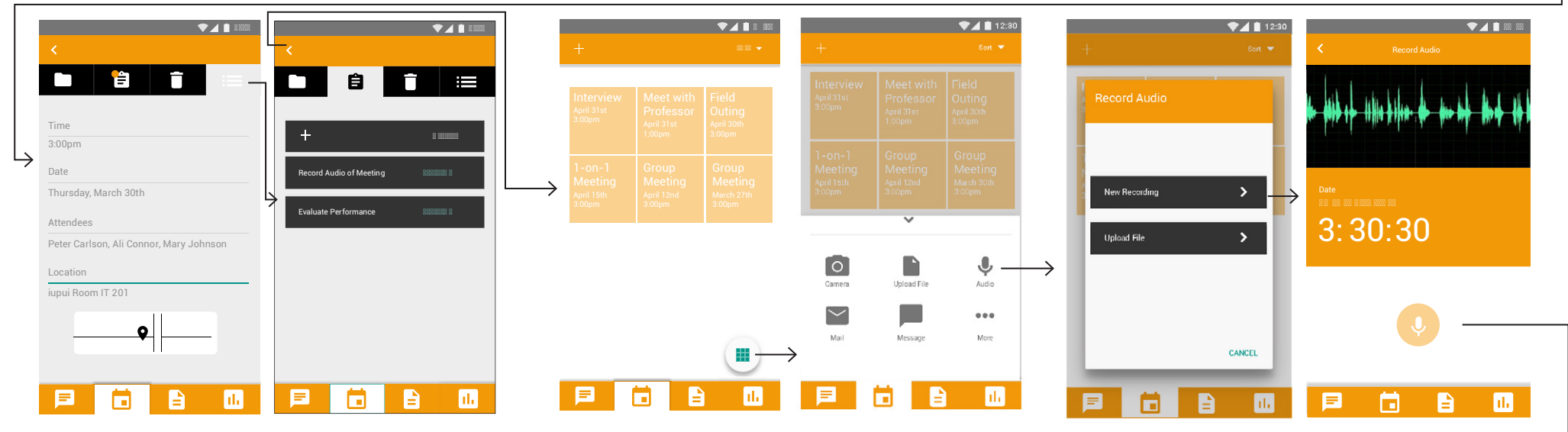
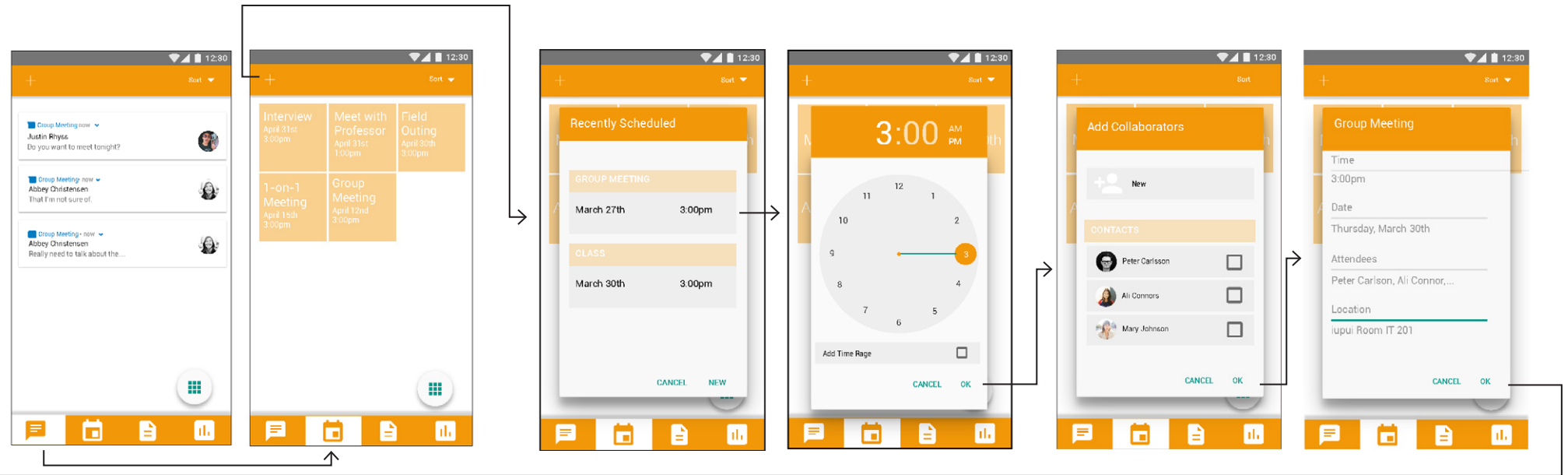
Succinct actions, Understandable Architecture, Easy Navigation

Discover-ability

Discoverable Content, Function Discover-ability

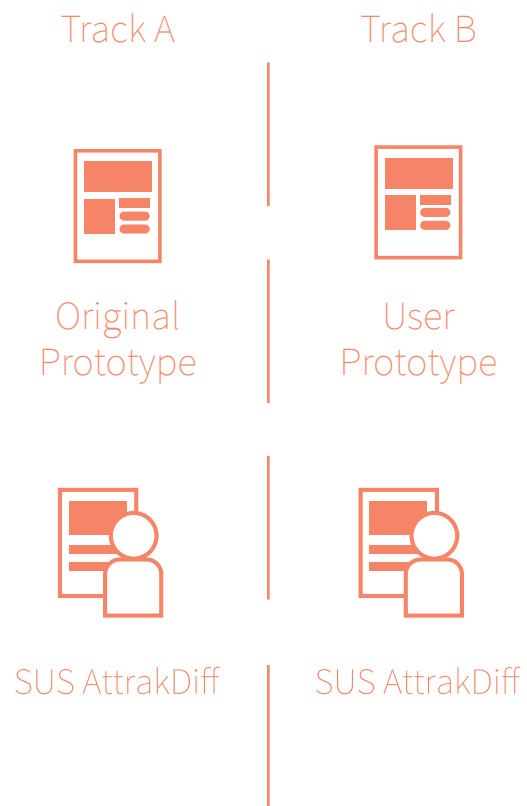
Alternative Interface Design

To understand whether there is a difference in perception, the two prototypes were compared to each other in an A/B test. Though there were differences in the user flow and the interfaces, it was important to know how these would affect user perceptions. Therefore, a double-blind study was set up with one group receiving the original prototype and another receiving the user-generated prototype. After completion of the test, users were asked to fill out a SUS and AttrakDiff survey to measure the perceptions of users.



AB Test

To understand whether there is a difference in perception, the two prototypes were compared to each other in an A/B test. Though there were differences in the user flow and the interfaces, it was important to know how these would affect user perceptions. Therefore, a double-blind study was set up with one group receiving the original prototype and another receiving the user-generated prototype. After completion of the test, users were asked to fill out a SUS and AttrakDiff survey to measure the perceptions of users.



Experimental Set Up

After the completion of the prototype, researchers conducted two evaluations with students from a CS class at IUPUI. There were 34 total participants in the study. Information regarding demographics was not recorded. Participants were divided evenly into two groups. Both groups were given one low fidelity prototype with limited click-through capability and an iPad to complete the interaction. The use of iPads was chosen to best replicate the experience of mobile use of EASEL. Group one was given the original prototype which was developed to test the concept of EASEL. The second group was given the prototype developed from the user engagement methods defined above. These groups were then asked to complete a series of tasks on the prototypes and provide feedback in the form of a SUS and AttrakDiff survey.

Measurement

The use of these surveys provided two well-tested means of evaluation. The SUS and the AttrakDiff. These tools help to measure a product against industry benchmarks and segment the user's interaction with a product into different categories. For the SUS, these are learn-ability, usability, and viability. For the AttrakDiff, these are perceptions of individuals, perceptions of the system, usability, and Attractiveness.

SUS

The SUS survey was developed by Brooke to measure a system. The survey looks at two categories of the system: learnability and usability. These are measured through 10 questions. The questions are scaffolded into positive and negative inquiries. Each juxtaposed set looks at a specific aspect of the system and how users perceive it. It uses a 5-point scale to measure each question. After completing a walkthrough of the software, users are asked to complete the survey which measures the overall viability of the product through a series of scoring mechanisms. From the overall score the product can be measured against industry benchmarks.

In this survey you will be shown for sections of word choices. Each section has a scale with opposing words at each end of the scale. Mark on the scales where you feel the application would be. Be sure to only fill in one box in each row, so each word set only has one box filled in.

HQ1
On the scales below, what words do you believe would describe a person using the application?
Be sure to mark only one box on each row.

Amateurish Professional

Gaudy Classy

Disposable Valuable

Unintelligent Intelligent

Unapproachable Approachable

Disheveled Presentable

Cautious Bold

HQS
On the scales below, what words do you believe best describe your experience with the program?
Be sure to mark only one box on each row.

Trite Innovative

Boring Exciting

Challenging Easy

Commonplace New

Displeasing Pleasing

Undesirable Desirable

Typical Original

PQ
On the scales below, what words do you believe describe using the program and its functions?
Be sure to mark only one box on each row.

Technical Human

Complicated Simple

Impractical Practical

Cumbersome Direct

Unpredictable Understandable

Confusing Clear

Unruly Manageable

Unusable Usable

Cold Warm

ATT
On the scales below, what words do you believe describe how the program looks?
Be sure to mark only one box on each row.

Bad Good

Ugly Beautiful

Unwelcoming Welcoming

Drab Colorful

Standard Creative

AttrakDiff

The AttrakDiff is a means of measuring people's perceptions of products. Developed by Hassenzahl It is segmented into four categories: HQI, HQS, PQ, and ATT. HQI measures how a person perceives an individual using the system. HQI is how individuals perceive the systems. HQS measures how users perceive the experience using the system. PQ measures usability. ATT measures attractiveness. The benefit of this survey is its ability to target four distinct aspects of product interaction.

	Strongly disagree						Strongly agree			
1. I think that I would like to use the prototype frequently	<input type="checkbox"/>	1	<input type="checkbox"/>	2	<input type="checkbox"/>	3	<input type="checkbox"/>	4	<input type="checkbox"/>	5
2. I found the prototype unnecessarily complex	<input type="checkbox"/>	1	<input type="checkbox"/>	2	<input type="checkbox"/>	3	<input type="checkbox"/>	4	<input type="checkbox"/>	5
3. I thought the prototype was easy to use	<input type="checkbox"/>	1	<input type="checkbox"/>	2	<input type="checkbox"/>	3	<input type="checkbox"/>	4	<input type="checkbox"/>	5
4. I think that I would need the support of a technical person to be able to use the prototype	<input type="checkbox"/>	1	<input type="checkbox"/>	2	<input type="checkbox"/>	3	<input type="checkbox"/>	4	<input type="checkbox"/>	5
5. I found the various functions in the prototype were well integrated	<input type="checkbox"/>	1	<input type="checkbox"/>	2	<input type="checkbox"/>	3	<input type="checkbox"/>	4	<input type="checkbox"/>	5
6. I thought there was too much inconsistency in the prototype	<input type="checkbox"/>	1	<input type="checkbox"/>	2	<input type="checkbox"/>	3	<input type="checkbox"/>	4	<input type="checkbox"/>	5
7. I would imagine that most people would learn to use the prototype very quickly	<input type="checkbox"/>	1	<input type="checkbox"/>	2	<input type="checkbox"/>	3	<input type="checkbox"/>	4	<input type="checkbox"/>	5
8. I found the prototype very cumbersome to use	<input type="checkbox"/>	1	<input type="checkbox"/>	2	<input type="checkbox"/>	3	<input type="checkbox"/>	4	<input type="checkbox"/>	5
9. I felt very confident using the prototype	<input type="checkbox"/>	1	<input type="checkbox"/>	2	<input type="checkbox"/>	3	<input type="checkbox"/>	4	<input type="checkbox"/>	5
10. I needed to learn a lot of things before I could get going with the prototype	<input type="checkbox"/>	1	<input type="checkbox"/>	2	<input type="checkbox"/>	3	<input type="checkbox"/>	4	<input type="checkbox"/>	5

Findings

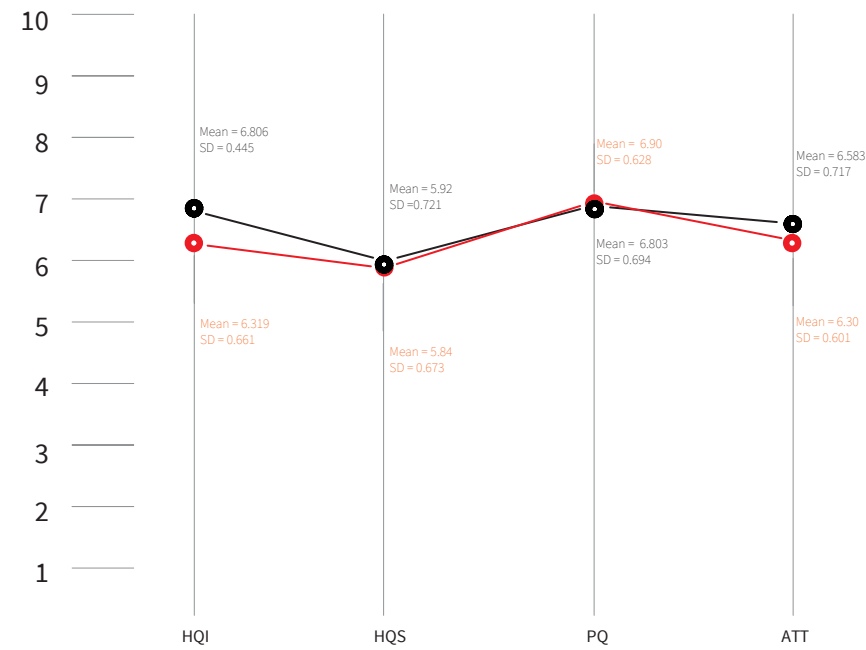
To know if hedonic perceptions and viability varied between the user-generated and original prototypes, the SUS and modified AttrakDiff surveys were compared to each other. Utilizing paired T-tests, the average scores were calculated to discover if there was statistical significance to the responses. In total, there were 34 responses to the surveys, 17 for each group totaling a response rate of 100%.

In the analysis of the surveys, the modified AttrakDiff was looked at first. This survey measures four aspects of a product. HQI measures the perceptions people have of users of the product. HQS measures the perceptions of the product experience. PQ measures perceptions of usability. ATT measures the aesthetics of the product. These categories help to provide an overview of strong and weak points in a product. The researcher compared the two prototypes based on these categories through a comparison of the means to each question.

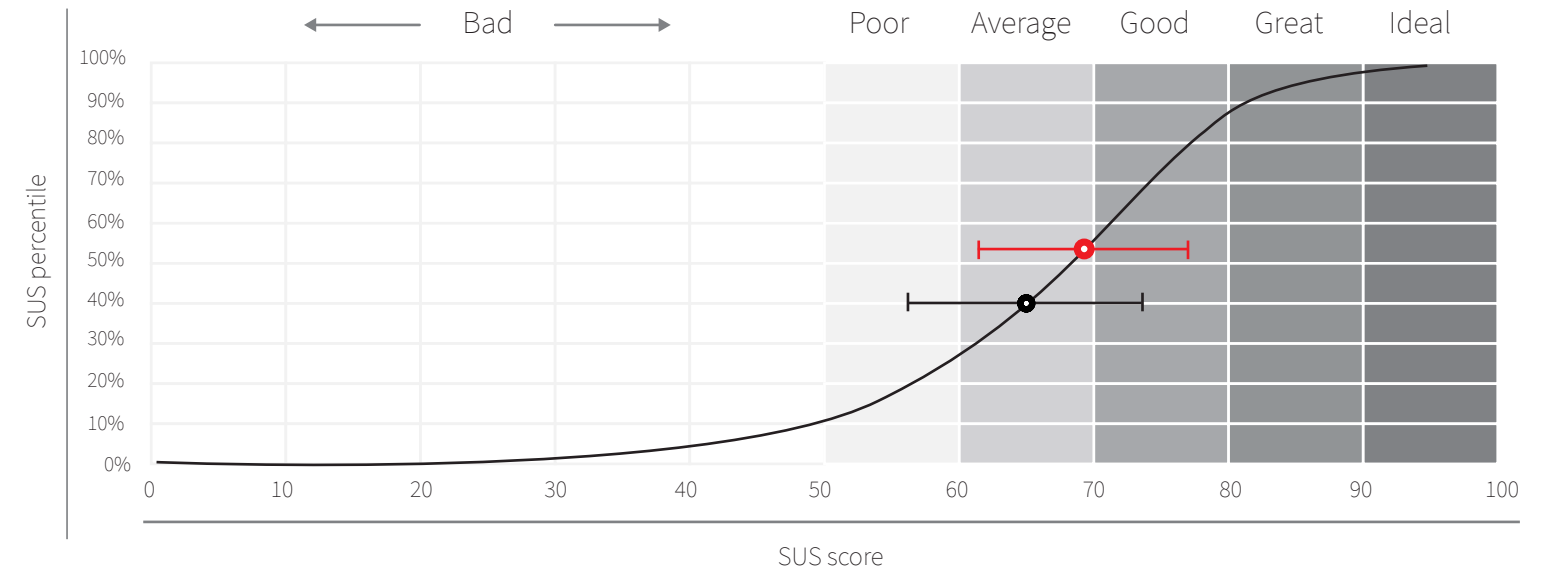
Starting with HQI, noted a statistically significant difference between the user-generated prototype and the original prototype. The user-generated prototype, was given an overall score of 6.806 and a standard deviation of .445 compared to the original prototype which had a 6.319 and a standard deviation of .661. To know if this was statistically significant a T-test was conducted. The results of the test returned a 95% confidence interval indicating that the results were significant.

The same tests were conducted on HQS, PQ, and ATT: the results of the T-test on these categories showed no statistically significant results. The mean of HQS for the user-generated prototype was 5.92 with a standard deviation of .721. The original prototype had an HQS score of 5.84 with a standard deviation of .673. PQ scores were also very similar. The original prototype had a score of 6.90 with a standard deviation of .628. The PQ score of the user-generated prototype was 6.803 with a standard deviation of .694. The ATT scores of the prototypes were not significant, though the user-generated prototype had a higher score 6.583 compared to 6.30.

AttrakDiff scores by question group

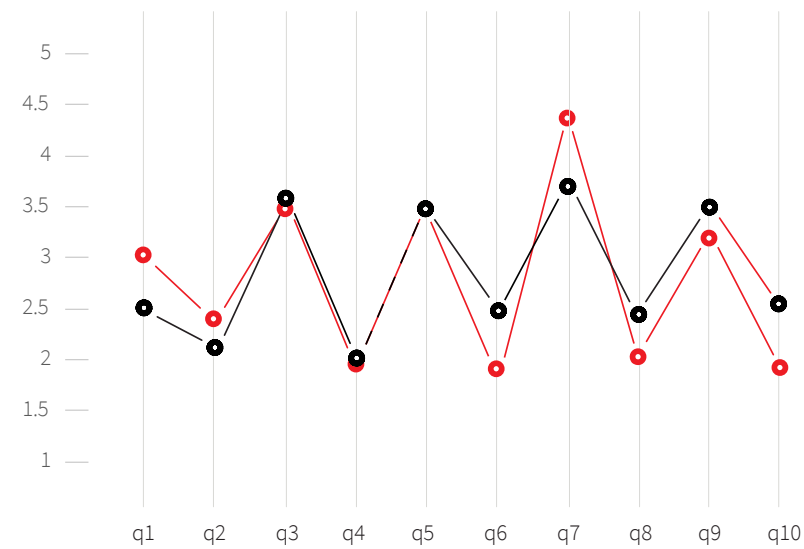


SUS scores by percentile



● Original prototype
● User generated prototype

SUS scores by question



USERS THEN WERE ASKED TO EVALUATE THE PROTOTYPES. THESE EVALUATIONS WERE ANALYZED AND GRAPHED TO SEE WHICH PROTOTYPE USERS PREFERRED.

RESULTS WERE MIXED. BUT INDICATED THAT THE TWO PROTOTYPES WERE PERCEIVED TO BE THE SAME.

DI-

scus-

SION



Based on the findings from the surveys, it is inconclusive as to whether or not user engagement in the early stages of design has a measurable impact on users' perceptions of a product. This can be confirmed by the lack of significant results from the surveys conducted.

There are a few conclusions that can be drawn from the data collected. The increased perception of individuals using the application (PQ) was statistically higher than that of the original prototype. This higher rating could have resulted from two variables. The first was the additional functionality. Compared to the original prototype, there were several functions that were added to the user involvement prototype. These included communication functions, a collaborative learning aspect, and a file management system. These added functions were noted in the basic data totals of PQ. The PQ of the user involvement prototype was higher compared to the original prototype, although it was not significant. The second aspect that may have led to the higher HQI rating could have been the new UI. For the user involvement prototype, Google material design was used to make the user interface. The UI was measured through attractiveness (ATT). Users perceived the user involvement prototype to be more attractive. However, this was most likely the result of the new interface. This decision made the general interface similar to applications users were familiar with. Although these individual qualities were not significantly more than the original prototype, the combination of both the functionality and attractiveness may have contributed to the higher HQI score. This relationship could be explained through a cumulative effect. Though PQ nor ATT were not statistically significant, the added functionality and the new interface could have affected users' perception of individuals who would use the application.

With the lack of statistically significant results from the SUS, there is no indication that the original prototype was more viable despite the higher average score of the original prototype. There are several possibilities for this. It is probable to assume that low fidelity prototypes are not capable of providing users with enough information to give realistic usability information.

There is also, an indication that user-involvement in the early stages does provide some value to the development process. This is from the requirements gathered during the prototyping sessions. When compared to the original prototype the requirements gathered matched those proposed by the EASEL creators.

The user-generated requirements showed gaps in the original EASLE requirements. This was most noted in the accessibility requirement. These outcomes do show usefulness to user engagement methods for the requirement elicitation. This also alleviated a great deal of effort in the design of the interface as much of the design requirements for functions and features were already defined after the data analysis.

What is seen in this study is the limits to the role of users, in the design phase of development. When looking at the requirements it is apparent that users are very capable of seeing contextual needs. Users knew what they used for their tasks and when they would use them. These are very basic aspects and focus on contextual uses. From the comparison of user generated requirements to the outline from the easel prototype, users could find many basic requirements aligning to those defined by the easel development team as well as new ones. What users were not able to discover was media and channels of use. Even when looking at the data, mediums are not often mentioned by users. What this proposes is something not specifically defined in literature, a role for users. Sanders and tappers mention engaging users as experts.⁽⁴⁾ But what are they experts in? It is possible to say that they are not experts in technology. The end users engaged did not identify any new or experimental technology, nor is it reasonable to expect them to. Not every person can be expected to know about technology trends, not even all experts are familiar with technology trends. What users are capable of defining are contextual uses. These are the situations in which users engage technology and how the technology is used.⁽²³⁾

Inevitably the findings from this study relate back to a statement made by Cooper. In *About Face*, he states that the software design process is too segmented and siloed.⁽¹⁹⁾ Though he mentions this when talking about the role of researchers, designers, and developers; this statement can well be applied to the involvement of users. Even though user may not have much knowledge of technology or systems, they are able to identify how they complete tasks and what tools they would need to complete those tasks. This can provide a great deal of insight into the basic uses of software.

¹⁹ Cooper, Alan, Robert Reimann, and David Cronin. *About face 3: the essentials of interaction design*. John Wiley & Sons, 2007.

²³ Coutaz, Joëlle, James L. Crowley, Simon Dobson, and David Garlan. "Context is key." *Communications of the ACM* 48, no. 3 (2005): 49-53.

LIMIT.

ATION



Though this research aims to add to literature of the co-creation process in software interface design, it is limited in three ways. One is the stage at which the study takes place in the development process. With limited features, available in the prototypes, it may not be possible to accurately measure the impact of users' input. Secondly, Due to the time limit of this study and the limited interaction that users have with the prototypes, it may not be able to conclusively address the ability of co-creation in interface design to affect users' perceptions of software. The third limitation is the UI design, this being different form the original prototype could have affected perceptions of the product.

NEXT

STEPS



For next steps and future research, users should be engaged within the final interface design, by providing input as the interface and product flows are being designed. This type of engagement will allow for the evaluation of user involvement in further iterations of software design. By extending user involvement deeper in the development process the extent of user's abilities and their contributions can be measured.

References

- 1 Kujala, Sari, user involvement: a review of the benefits and challenges, Behavior and Information Technology vol. 22 no. 1, 1-16, 2003
- 2 Jun He, William King, The Role of User Participation in Information Systems Development: Implications from a Meta-Analysis, Journal of Management Information Systems / Summer 2008, Vol. 25, No. 1, pp. 301–331.
- 3 Nicolas Ind, Nick Coates, The meanings of co-creation, European Business Review, Vol. 25 Iss 1 pp. 86 - 95
- 4 Sanders E., Stappers P., Co-creation and the new landscape, co-design, 2008
- 5 Richard M. Ryan and Edward L. Deci, Self-Determination Theory and the Facilitation of Intrinsic Motivation, Social Development, and Well-Being, American Psychologist January 2000
- 6 Von Hippel, Eric, Democratized innovation February 2006
- 7 Fischer, G, Beyond “Couch Potatoes ” From Consumers to Designers and Active Contributors, First Monday, 2002
- 8 Marc Steen, Menno Manschot, and Nicole De Koning, Benefits of Co-design in Service Design Projects
- 9 Sanders E., Stappers P., Probes, Toolkits and Prototypes: three Approaches to Making in Codesigning, co-design, 2014
- 10 Bano, Muneera, and Didar Zowghi. “A systematic review on the relationship between user involvement and system success.” Information and Software Technology 58 (2015): 148-169.
- 11 Chamberlain, Stephanie, Helen Sharp, and Neil Maiden. “Towards a framework for integrating agile development and user-centred design.” In International Conference on Extreme Programming and Agile Processes in Software Engineering, pp. 143-153. Springer Berlin Heidelberg, 2006.
- 12 S. Pekkola, N. Kaarilahti, and P. Pohjola, “g,” In Proceedings of the ninth conference on Participatory design: Expanding boundaries in design-Volume 1, pp 21–30. ACM, 2006
- 13 Wu, J.B., and Marakas, G.M. The impact of operational user participation on perceived system implementation success: An empirical investigation. Journal of Computer Information Systems, 46, 5 (2006), 127–140
- 14 K. Kautz, “Participatory Design Activities and Agile Software Development,” Human Benefit through the Diffusion of Information Systems Design Science Research, pp. 303–316, 2010

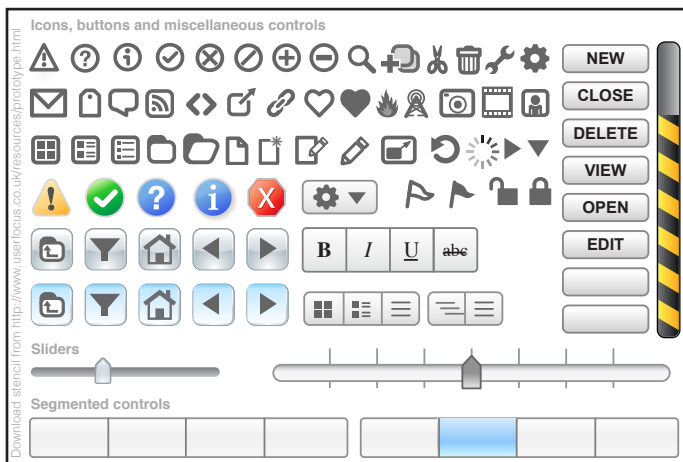
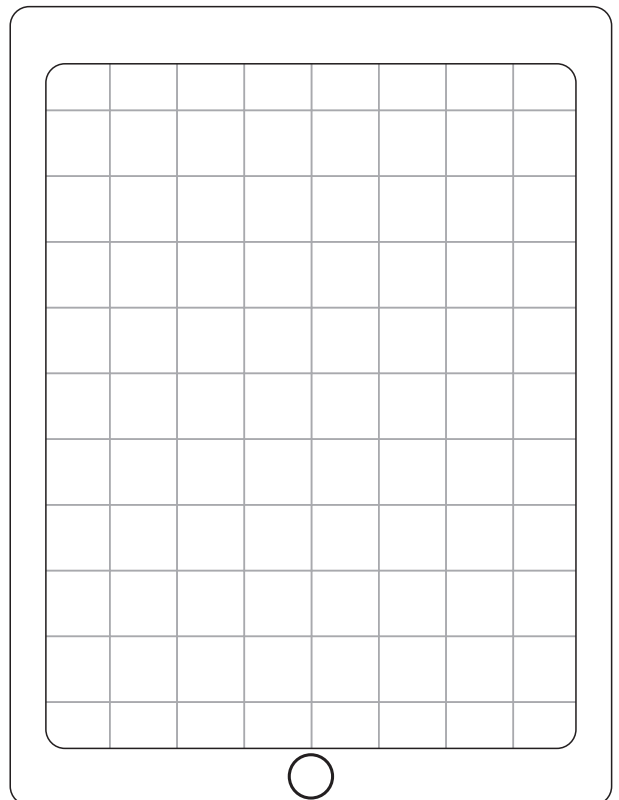
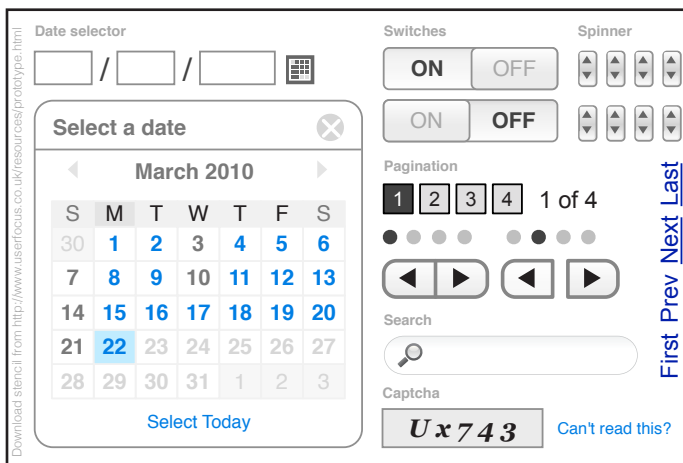
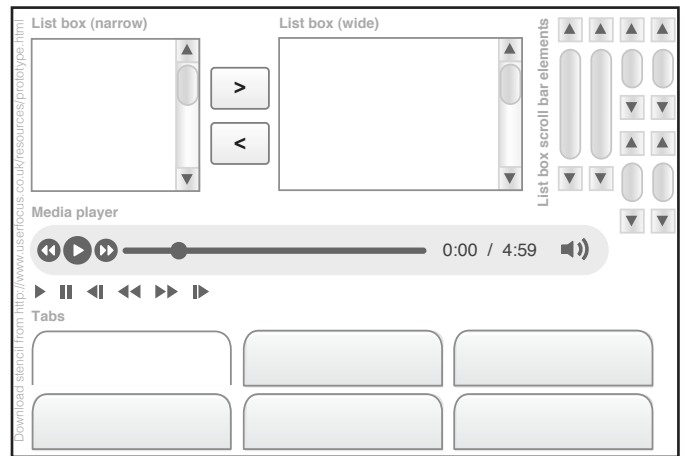
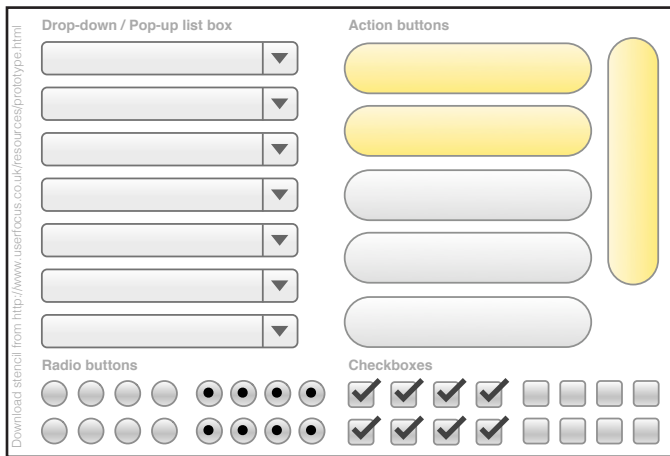
- 15 Hassenzahl, Marc. "The effect of perceived hedonic quality on product appealingness." *International Journal of Human-Computer Interaction* 13, no. 4 (2001): 481-499.
- 16 Butt, Saad Masood, Wan Fatimah Wan Ahmad, and Wan Fatimah. "An overview of software models with regard to the users involvement." *International Journal of Computer Science Issues (IJCSI)* 9, no. 3 (2012): 1.
- 17 Brooke, John. "SUS-A quick and dirty usability scale." *Usability evaluation in industry* 189.194 (1996): 4-7.
- 18 Isleifsdottir, Jonheidur, and Marta Larusdottir. "Measuring the user experience of a task oriented software." In *Proceedings of the International Workshop on Meaningful Measures: Valid Useful User Experience Measurement*, vol. 8, pp. 97-101. 2008.
- 19 Cooper, Alan, Robert Reimann, and David Cronin. *About face 3: the essentials of interaction design*. John Wiley & Sons, 2007.
- 20 John, Bonnie. "Why GOMS?." *interactions* 2, no. 4 (1995): 80-89.
- 21 Hutchins, E. L., and J. D. Hollan. "D. Norman (1986). *Direct Manipulation Interfaces*." *User centered system design* (81): 87-124.
- 22 Hanington, Bruce, and Bella Martin. *Universal methods of design: 100 ways to research complex problems, develop innovative ideas, and design effective solutions*. Rockport Publishers, 2012.
- 23 Coutaz, Joëlle, James L. Crowley, Simon Dobson, and David Garlan. "Context is key."

APP-

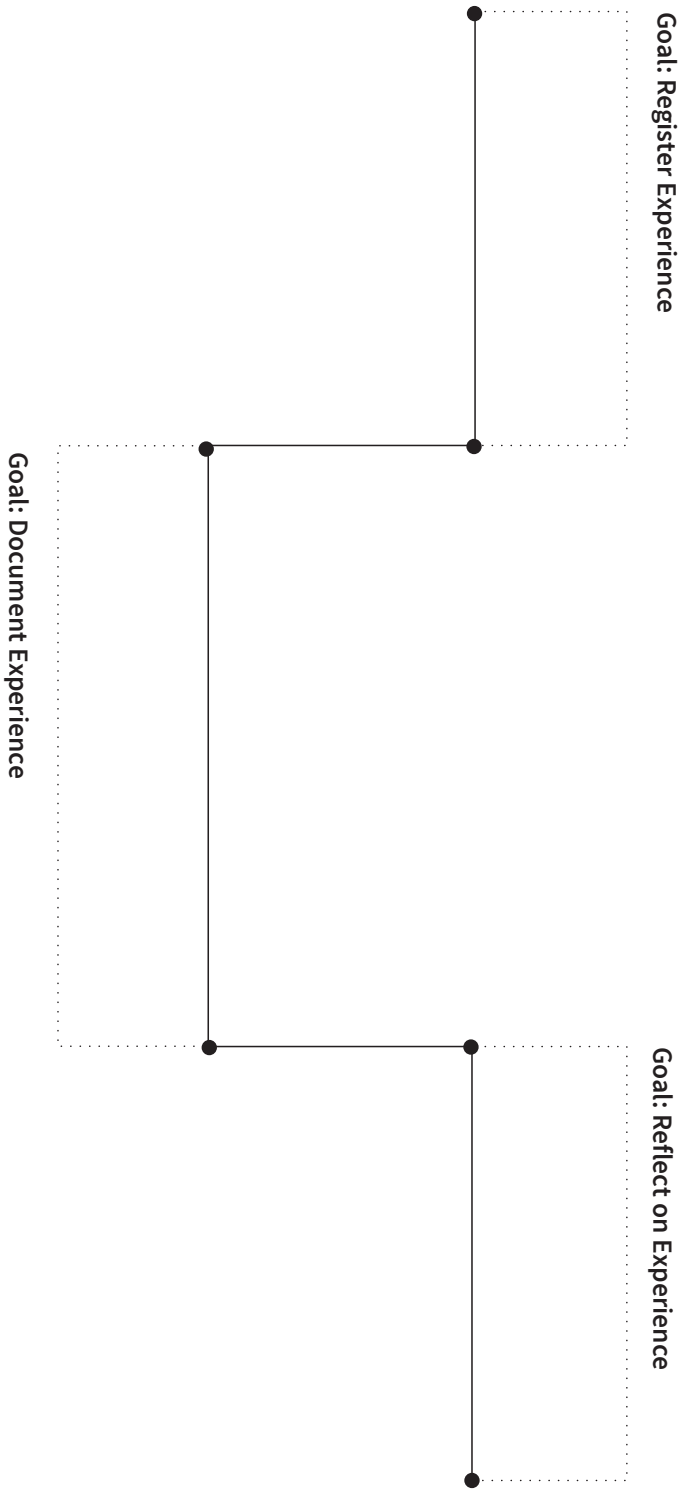
PEN-

DIX

PAPER PROTOTYPING KIT



JOURNEY MAP



In this survey you will be shown for sections of word choices. Each section has a scale with opposing words at each end of the scale. Mark on the scales where you feel the application would be. Be sure to only fill in one box in each row, so each word set only has one box filled in.

HQI

On the scales below, what words do you believe would describe a person using the application?
Be sure to mark only one box on each row.

Amateurish	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Professional
Gaudy	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Classy
Disposable	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Valuable
Unintelligent	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Intelligent
Unapproachable	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Approachable
Disheveled	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Presentable
Cautious	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Bold

HQS

On the scales below, what words do you believe best describe your experience with the program?
Be sure to mark only one box on each row.

Trite	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Innovative
Boring	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Exciting
Challenging	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Easy
Commonplace	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	New
Displeasing	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Pleasing
Undesirable	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Desirable
Typical	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Original

PQ

On the scales below, what words do you believe describe using the program and its functions?
Be sure to mark only one box on each row.

Technical	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Human
Complicated	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Simple
Impractical	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Practical
Cumbersome	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Direct
Unpredictable	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Understandable
Confusing	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Clear
Unruly	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Manageable
Unusable	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Usable
Cold	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Warm

ATT

On the scales below, what words do you believe describe how the program looks?
Be sure to mark only one box on each row.

Bad	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Good
Ugly	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Beautiful
Unwelcoming	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Welcoming
Drab	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Colorful
Standard	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Creative

	Strongly disagree				Strongly agree
1. I think that I would like to use the prototype frequently	1	2	3	4	5
2. I found the prototype unnecessarily complex	1	2	3	4	5
3. I thought the prototype was easy to use	1	2	3	4	5
4. I think that I would need the support of a technical person to be able to use the prototype	1	2	3	4	5
5. I found the various functions in the prototype were well integrated	1	2	3	4	5
6. I thought there was too much inconsistency in the prototype	1	2	3	4	5
7. I would imagine that most people would learn to use the prototype very quickly	1	2	3	4	5
8. I found the prototype very cumbersome to use	1	2	3	4	5
9. I felt very confident using the prototype	1	2	3	4	5
10. I needed to learn a lot of things before I could get going with the prototype	1	2	3	4	5

