SPEEDING UP THE PROTOTYPING OF LOW-FIDELITY USER INTERFACE WIREFRAMES

An Undergraduate Research Scholars Thesis

by

MELISSA ZHANG

Submitted to the LAUNCH: Undergraduate Research office at
Texas A&M University
in partial fulfillment of the requirements for the designation as an

UNDERGRADUATE RESEARCH SCHOLAR

Approved by
Faculty Research Advisors: Dr. Tracy Anne Hammond
Dr. Paul Taele

May 2022

Major: Computer Science

RESEARCH COMPLIANCE CERTIFICATION

Research activities involving the use of human subjects, vertebrate animals, and/or biohazards must be reviewed and approved by the appropriate Texas A&M University regulatory research committee (i.e., IRB, IACUC, IBC) before the activity can commence. This requirement applies to activities conducted at Texas A&M and to activities conducted at non-Texas A&M facilities or institutions. In both cases, students are responsible for working with the relevant Texas A&M research compliance program to ensure and document that all Texas A&M compliance obligations are met before the study begins.

I, Melissa Zhang, certify that all research compliance requirements related to this Undergraduate Research Scholars thesis have been addressed with my Research Faculty Advisors prior to the collection of any data used in this final thesis submission.

This project required approval from the Texas A&M University Research Compliance & Biosafety office.

TAMU IRB #: 2021-1501M Approval Date: 01/31/2022 Expiration Date: 01/30/2025

TABLE OF CONTENTS

ABSTRACT

Speeding Up the Prototyping of Low-Fidelity User Interface Wireframes

Melissa Zhang
Department of Computer Science and Engineering
Texas A&M University

Research Faculty Advisor: Dr. Tracy Anne Hammond
Department of Computer Science and Engineering
Texas A&M University

Research Faculty Advisor: Dr. Paul Taele
Department of Computer Science and Engineering
Texas A&M University

Prototyping for user interfaces (UIs) is essential when it comes to giving an idea for how the final result will look and function. This allows both designers and stakeholders to test if all the project requirements are being fulfilled and the functionality of the UI, saving time and money further down the road when the final product is being made. However, current established prototyping tools have an emphasis on visual aesthetics without support for quickly producing low-fidelity prototyping. This leads to the initial prototyping process to be time-consuming and labor-intensive as designers need to start from scratch. In this research, we propose a solution that hopefully speeds up the low-fidelity UI prototyping process by streamlining the workflow. Based on previous research of scenario-based UI design and utilizing UI design patterns and sketching, we created a Figma plugin tool for guiding the creation of low-fidelity wireframe prototypes. We then conducted a user study which revealed that the plugin was easy to learn and use with a lower subjective workload for physical demand, temporal demand, and frustration. The contribution of our work is providing an alternative approach for designers and stakeholders to collaborate together

and facilitate easy prototyping of low-fidelity UI wireframes.

DEDICATION

To our families, instructors, and peers who supported us throughout the research process.

ACKNOWLEDGMENTS

Contributors

I would like to thank my faculty advisors, Dr. Tracy Hammond and Dr. Paul Taele, and my mentor Matthew Runyon for their guidance and support throughout the course of this research.

Thanks also go to my friends and colleagues and the department faculty and staff for making my time at Texas A&M University a great experience.

Funding Sources

Undergraduate research did not receive any funding.

NOMENCLATURE

UI	User Interface
UML	Unified Modeling Language
NASA-TLX	NASA Task Load Index
RTLX	Raw NASA Task Load Index

1. INTRODUCTION

During the creation of user interfaces (UI), UI prototypes are necessary to visualize and test functionality before a final design is approved [1]. These UI prototypes are not only able to help the designer figure out flaws in their design, but UI prototypes also allow non-technical stakeholders and target users point out potential issues early on [1, 2, 3, 4, 5]. Additionally, stakeholders are able to quickly see their feedback being incorporated into the design, demonstrating how UI prototypes can encourage effective group communication [2, 4, 6]. Furthermore, designers and stakeholders are not the only ones benefiting from UI prototypes. Completed UI prototypes are able to help new developers quickly orient themselves and begin contributing to an ongoing project [2]. The importance of UI prototyping can then be seen with the advantages of being able to test functionality before investing in costly implementations and facilitating effective communication.

1.1 UI Design Process

Before further discussions on UI prototyping, it is important to understand what role UI prototyping plays in the overall UI design process. Depending on the UI's development stage, the UI design process will vary. However, the general UI design process is as follows.

1.1.1 Understanding the Problem

Every UI has a purpose to fulfill. As such, the start of the UI design process is defining the problem the UI should solve. This problem definition should include the background, goals, success criteria, target audience, and scope. Additionally, the team structure and project deadline would also be considered during this stage [1].

1.1.2 Implement Competitor Benchmarking

Designing a new UI will often use inspiration and references from existing UIs. This includes UIs designed by competitors in the industry. From there, design benchmarks can be created to serve as a helpful guideline to give a direction for future UI development [1].

1.1.3 Define Screens and User Behavior

During the UI design, it is helpful to think about what screens are needed during a user's journey through the UI. From when the user first starts using the UI to when the user achieves their goal, every step of the interaction needs to be designed. This is the user flow which helps demonstrates the connection between different interactions in a simple to understand flow chart. This user flow will then dictate how each screen would interact with each other [1].

1.1.4 Develop Sketches and Wireframes

With an understanding of what screens are needed to accommodate the appropriate user behavior, the next step is create sketches that focus on the layout and functionality of elements such as buttons and content in order to achieve the desired interface functionality. This can illustrate a high-level idea of how the interfaces can meet the user's needs. Sketches can then be converted into more detailed wireframes which provides a low-fidelity prototype guideline for the final screen. At this stage, an iterative design process will be undertaken as input from relevant stakeholders are taken into consideration [1].

1.1.5 Create and Maintain a Design System

Before moving on to high-fidelity prototypes, it is important to create a design system. A design system will group UI elements together and provide clear and consistent guidelines on how elements should look. These include attributes such as color, typography, and imagery. As such, the design system provides a library of reusable elements that give both developers and stakeholders a clear idea on what the high-level prototype will look like [1].

1.1.6 Develop High-fidelity Prototypes

With a clear layout and design in mind, a high-fidelity prototype can be created. The high-fidelity prototype is an interactive prototype which is visually similar to the final product. This high-fidelity prototype can then be presented to both stakeholders and users to test usability, providing an opportunity to receive feedback and improve the design before any resources are spent on code implementation or marketing [1].

1.1.7 Hand-off

At this stage, the final high-fidelity prototype is presented to stakeholders with a discussion of what changes have been made. Stakeholders could then give feedback that may bring the design process back to any earlier stage [1]. However, if stakeholders are satisfied with the final process, then the UI design process has been completed.

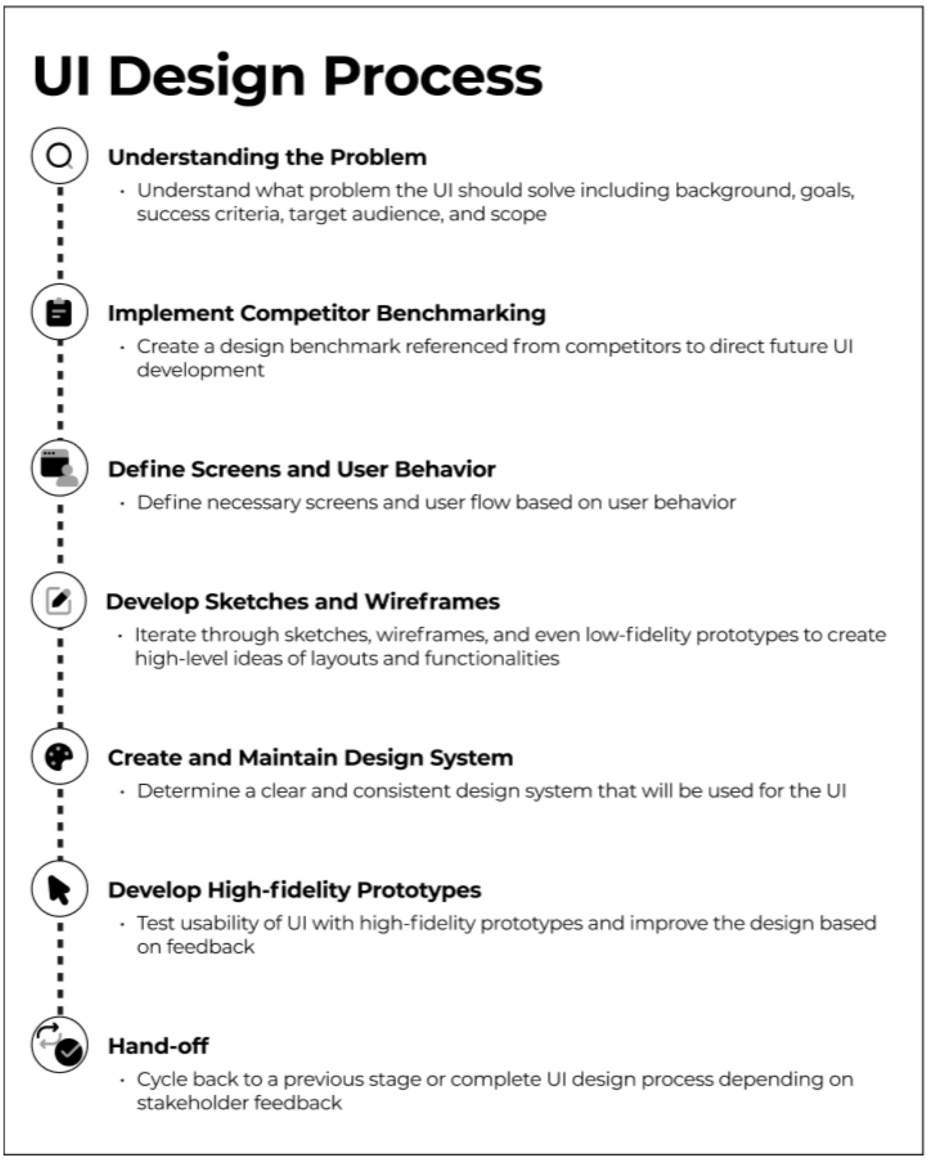


Figure 1.1: Summary of UI design process

From this UI design process summarized in Figure 1.1, it is apparent to see how UI prototyping (both low-fidelity and high-fidelity) is an iterative process that takes a lot of time in order to improve and finesse designs and user interactions. As such, speeding up the UI prototyping process can potentially speed up the completion of the overall UI. This increase in speed would then prove to be desirable to designers and stakeholders who wish to release their product as soon as possible.

1.2 Understanding Fidelity

Fidelity as it relates to a prototype refers to how well the prototype conveys the final look and feel of the final product. This can be in reference to the visual design, contact, and interactivity of the prototype [4]. So far, fidelity has been presented as either low or high fidelity. However, in reality, these are very broad categories and the fidelity of the prototype can change over the course of many iterations [4]. Even so, the fidelity of a prototype is often grouped to be either low-fidelity or high-fidelity.

1.2.1 Low-fidelity Prototyping

Low-fidelity prototyping focuses on presenting an abstract and high-level look at the developing UI. Its role is to check and test functionality rather than the visual appearance of the product, determining the fundamental screens and important user flows the product will have [4, 5]. At this fidelity, prototyping can be done cheaply, quickly, and in collaboration with both technical and non-technical members. It is excellent for brainstorming and all team members and stakeholders are able to get a better expectation of what the final product will look like [4, 5, 6]. User testing can be done early and low-fidelity prototypes can perform usability testing just as well as high-fidelity prototypes [5, 7]. Low-fidelity prototyping techniques include using pen and paper to create a simple physical prototype and taking advantage of specialized software to create a clickable wireframe which provides a more interactive prototype as shown in Figure 1.2 [4].

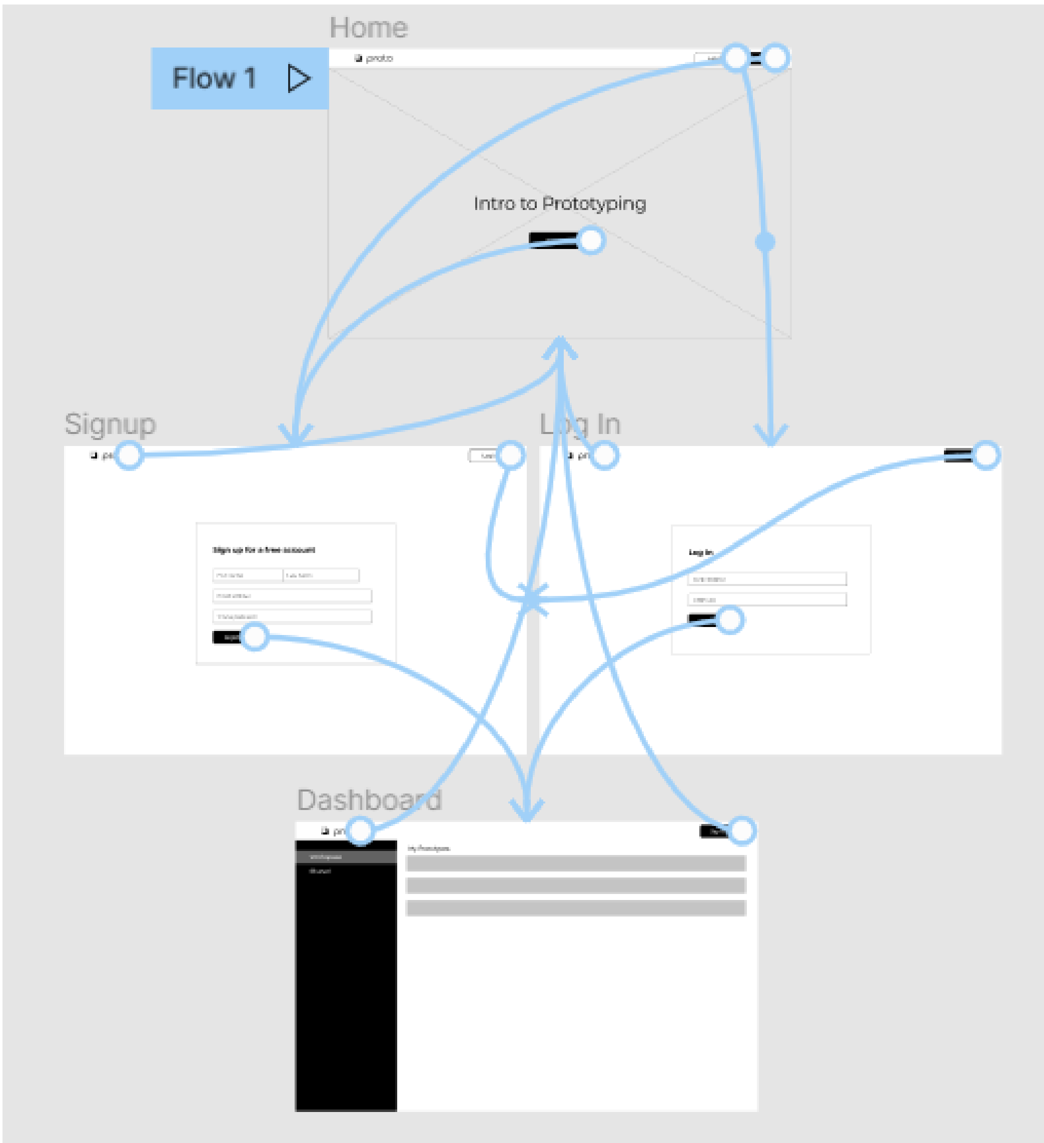


Figure 1.2: Example of low-fidelity prototyping

1.2.2 High-fidelity Prototyping

High-fidelity prototyping, on the other hand, presents visuals and functionality that is similar or even an exact replica of the final product [4]. As described previously, at this fidelity the focus is to conduct user tests and get any design feedback. Due to how polished this prototype is, feedback from user testing is more relevant to the final product and how developers will implement the UI [4, 5]. However, this polish does come at a higher cost compared to low-fidelity prototyping as a lot more time and money is needed to create a functioning and visually appealing prototype [4]. Techniques used to develop high-fidelity prototyping includes digital prototypes developed by specialized software (such as in Figure 1.3) and coded prototypes [4].

Unless a product already exists with a similar UI to the desired final product, it is hard to speed up high-fidelity prototyping. In order for a unique UI to be created, time and money need to be spent on the high-fidelity prototype. Therefore, the focus of our research is on speeding up low-fidelity prototyping where a lot of time is spent on exploring user flows and functionality. The research will specifically focus on clickable wireframes which can be reused in the high-fidelity prototype and thus indirectly speed up the entire prototyping process.

1.3 UI Prototyping Tools

Previously, UI prototyping was often created using physical mediums like pen and paper [3]. As a study in 2008 shows, 72% of UI developers were using art supplies due to how widely available, quick, and easy to use they were [3]. In contrast, the UI prototyping landscape has changed immensely in the current digital age. As a survey taken in 2021 shows, around 90% of participants use software for UI prototyping [8]. The UI prototyping software ranked in order of usage were Figma [9], Adobe XD [10], InVision [11], ProtoPie [12], and Sketch [13]. Furthermore, Figma was shown to be used by a vast majority of participants with more than five times the amount of users as compared to Adobe XD in second place [8]. However, all of these tools are meant to be used mainly in tandem to high-fidelity prototype development where an abundance of visual customization options are provided. As such, there is little to no built-in support when it comes to needing to quickly produce low-fidelity prototypes which do not have a high requirement for

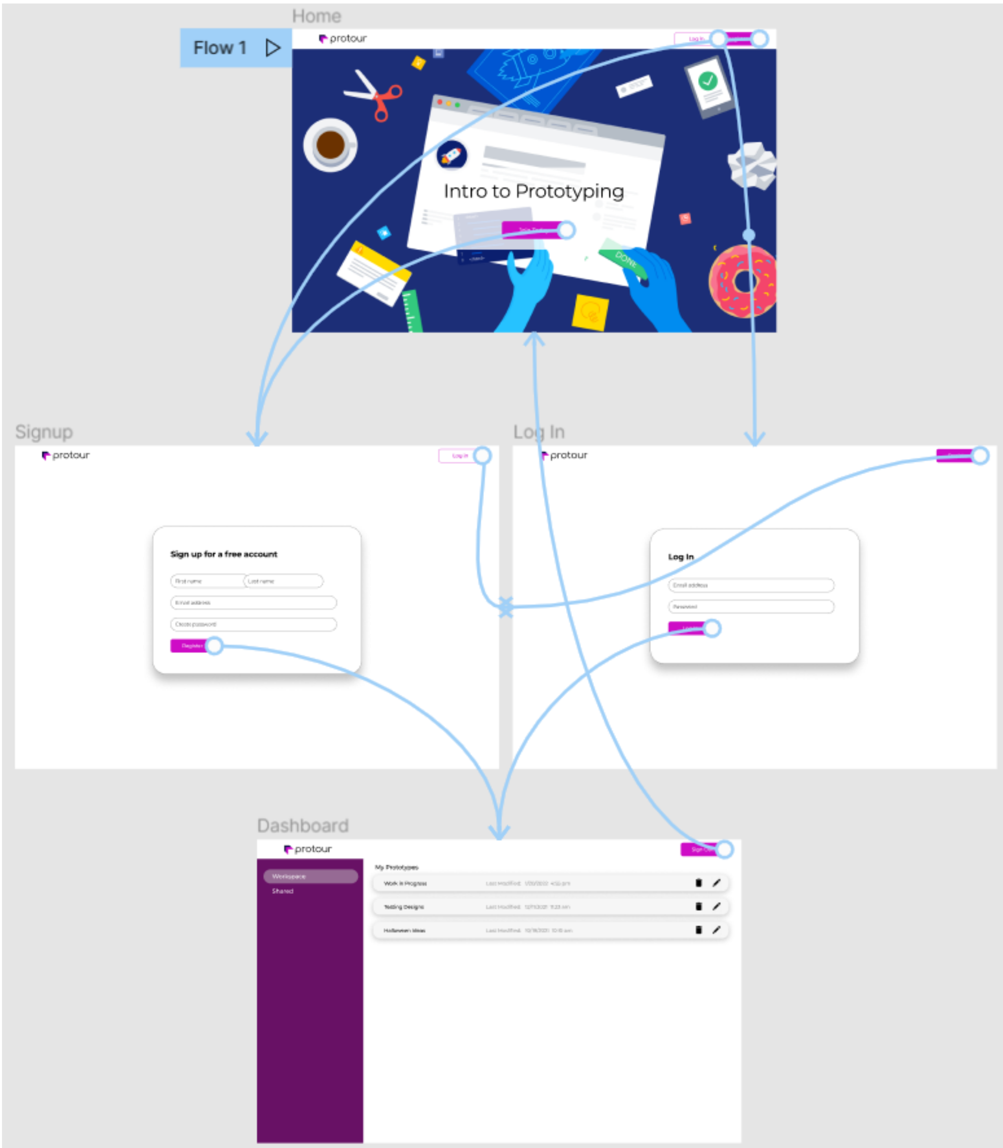


Figure 1.3: Example of high-fidelity prototyping

visual aesthetics. On the other hand, low-fidelity software like Balsamiq [14] is not as widely used for prototyping.

This research will focus on speeding up low-fidelity prototyping by introducing a tool which integrates previously researched methods of scenario-based design, UI design patterns, and sketching to create a more streamlined workflow, combining the user behavior and the generation of low-fidelity wireframe prototyping into one step. By doing so, designers and stakeholders will be able to speed up the testing and refinement of UI functionalities, preventing the need for costly changes later on when developing for high-fidelity UI.

2. RELATED WORK

Previous research explored methods to speed up the UI prototyping process. The methods focused on for this research are based on scenario-based design, UI design pattern, and sketching.

2.1 Scenario-Based Design

Scenario-based design is a family of techniques in which the future use of a system is already described in the early development process [15]. Due to scenarios being focused on the user of the final product, designers are better able to empathize with the user and analyze key usage scenarios in greater detail [15]. This scenario-based design can be seen in the user behavior stage of the UI design process. At that stage, the user flow graphs relationships between UI screens, demonstrating how each screen interacts with each other through the user's interaction. The user flow can then provide a means to help quickly generate prototype connections within a low-fidelity wireframe. This generation of prototypes via a graphical model can potentially speed up the UI prototyping time as well as help designers visually understand the scenarios that they are dealing with as previous scenario based UI prototyping has found. The two models which will be explored is the use of unified modeling language (UML) and relational graphs.

2.1.1 Unified Modeling Language

Unified modeling language (UML) is defined as a "general-purpose visual modeling language that is used to specify, visualize, construct, and document the artifacts of a software system" [16]. As such, they are often used by software engineers to develop software, application, and framework systems [17]. Research has also been done in regards to using UML models as a means to generate executable code [18]. This includes research relating to UI prototype generation using UML [19].

In 2006, even with the use of UI interface builders and UI management systems, the development of UIs was still time-consuming since each UI object had to be laid out explicitly [19]. Elkoutbi et al. conducted research which took scenarios based on project requirements and used them to automatically derive UI prototypes. UMLs were chosen to model these scenarios since the flexibility of UMLs meant that a wide range of issues could be covered such as use cases, scenarios, state behaviors, and operation declarations [19]. Their approach involved using scenario based UMLs to extract UI and object interaction information which will be analyzed and used for automatic UI prototyping [19]. The implemented approach was then tested on a class of graduate students where it was found that on average, students took half an hour to convert a single scenario into a UI prototype. In comparison, it was estimated that the same task by hand would have taken three times the effort at least [19].

2.1.2 Relational Graphs

Prior work demonstrated success in graphs for data that have an inherent relation among data elements [20]. These graphs are thus a suitable way to represent links between content [21]. For example, sitemaps are graphs used to represent a website's structure which demonstrates connections between web pages and content [22]. In the context of scenario-based design, scenarios can be abstracted and visualized in a graph to help designers in the UI design phase [23]. In the research of Kusano et al., the proposed hierarchical scenario tool allowed user's to tag written scenarios and visualize these tags as nodes on a graph [23]. This graph can help designers consider groups of requirements based on how nodes are connected as well as possible conflicts such as if a scenario tag has been used too broadly which can cause users to be potentially confused [23]. This graph visualization can help effectively understand relationships between scenarios, making the management of multiple scenarios easier [23]. Additionally, graphs offer a great benefit of supporting ideation which is important for low-fidelity UI prototyping where quick brainstorming and early user testing can occur [5, 21, 24].

2.2 UI Design Pattern

Patterns have the characteristics of being practical, existing solutions to reoccurring problems [25]. As such, UI design patterns serve as a means to help designers create UIs better and faster [26]. Not only do UI design patterns help the UI design process, it also helps improve the prototyping process. As Suleri et al. found, one problem designers encountered during rapid prototyping was insufficient time to create quality UI designs [27]. Their research solution was to create a UI design pattern and guideline library which helped designers find, view, and download customizable sample patterns [27]. Through usability evaluations with user experience designers, they found that overall their UI design pattern and guideline library was perceived as a useful approach to rapid prototyping with good usability and learnability [27]. Building upon this research, a later study also found that the subjective workload, such as physical demand and effort, experienced by designers using the UI design pattern approach was significantly less in comparison to traditional approaches of rapid prototyping [28].

2.3 Sketching

Sketching has been widely used as a means to create prototypes [3]. The ability to quickly iterate and communicate high-level ideas has made sketching a common tool for the early UI design process [29, 30]. As such, prototyping tools for sketch-based designs have been created to take advantage of sketching while also allowing the interactivity of prototyping [31, 32, 33]. Designers are able to use these tools to rapidly brainstorm, develop, and iterate through ideas for UI designs [31]. Very little training is needed for these tools which makes them easy to use even for beginners [32]. Finally, these sketching tools are able to refine the UI with input from designers, developers, and other stakeholders [33].

2.4 Impacts on Research

As previously mentioned, the proposed streamlined workflow solution to speed up low-fidelity UI wireframe prototyping involves combining the user behavior and low-fidelity prototyping stage into one. This will combine the use of a graph which maps the user's behavior into the wireframe prototyping process. This mapping of user behavior which determines how the final wireframe prototype interacts can be considered a scenario-based design. As such, the use of a graph holds the potential of speeding up the generation of a low-fidelity UI prototypes as previous research has shown [19]. Furthermore, users could benefit from being able to better manage multiple scenarios with the graph [23].

Before any prototyping can begin, the generation of UI components need to occur first. As such, the implementation of UI design patterns can help provide a quick way to generate quality UI while also not increasing the subjective workload of users unnecessarily in the process [27, 28].

Additionally, the low-fidelity prototyping process is an iterative process [1]. As such, it is necessary to be able to brainstorm and quickly iterate through designs with all team members and stakeholders [4, 5, 6]. As previously explored, sketching can provide a great method to brainstorm and communicate [29, 30]. Integrating sketching into the prototyping process could also make the prototyping process more accessible to even non-technical members of the team due to the low learning curve [32].

Overall, previous research in scenario-based design, UI design patterns, and sketching have provided a direction for what methods could help speed up the prototyping of low-fidelity UI wriframes.

3. DESIGN

During the UI design process, a user flow is often represented as a flow chart which can demonstrate the connection between screens and user interactions as seen in Figure 3.1 [1]. These user flows can then be used to create wireframes that better illustrates a high-level idea on how the final application will be laid out and interacted with by the user [1]. From there, the wireframes can then be made into clickable wireframe prototypes to test out preliminary functionality [4]. Through the iterative design process, the user flow and wireframe design process is interconnected with any updates to the user flow propagating into the wireframe and vice versa [1, 34]. As such, we propose a custom workflow tool which can take advantage of the user flow to help generate the low-fidelity wireframe prototype. This combination of steps within the UI design process as well as the use of graphs, UI design patterns, and sketching will hopefully speed up the overall prototyping speed of low-fidelity UI wireframes.

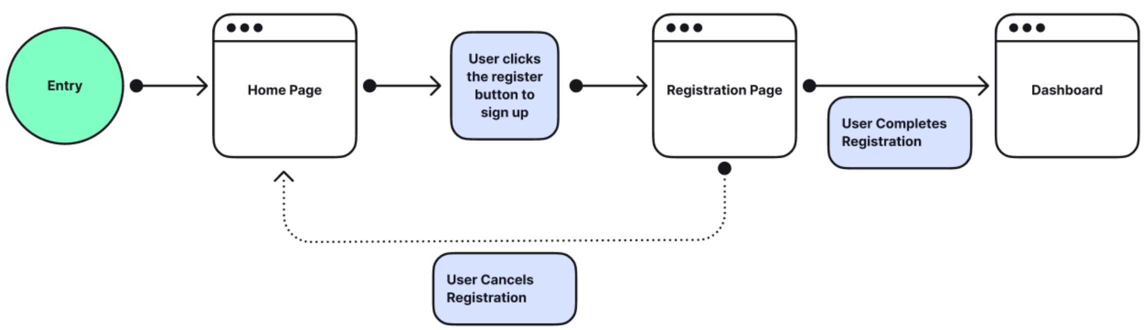


Figure 3.1: Example of a simple user flow

UI prototyping applications already exist which offer a variety of functionalities to create well developed prototypes, both low-fidelity and high-fidelity. Our research expands on prior applications by building off of these applications to introduce our custom workflow. As stated in the introduction, Figma [9] was ranked as the top UI prototyping software [8]. We therefore chose Figma as the prototyping application we'll build off with the development of a plugin.

3.1 Limitations in Figma

Even though Figma is the top UI prototyping software, it does not offer a lot of built-in tools for low-fidelity prototyping. For one, there is not a built-in component library. If a user wants to use pre-made components, then they would need to find one to duplicate from the community resources or make their own. Additionally, when prototyping in Figma, the prototyping arrows would overlap the actual UI screens as seen in Figure 1.2. Due to this overlap, elements could be hidden by the arrow and selection of these elements would be difficult. If a prototype needs a large amount of screens, the user potentially needs to fumble between zooming out and checking specific elements in order to see exactly where a prototype connection goes to.

3.2 Solution Objectives and Requirements

With our solution, we are aiming to achieve the following objectives:

- The plugin should be able to create and show the user flow
- The plugin needs to be able to generate prototype connections
- The plugin should be able to help in the creation of UI
- The plugin should facilitate easy prototype iteration and brainstorming

Since prototype connections between screens and the user flow can be represented by a graph, the plugin will need to be able to show a graph where each node represents a screen and the edge represents a prototype connection from an interactive element to a different screen. Additionally, the user flow incorporates more than just screens and prototype connections. It also needs to know what scenario the screen is being used in. As such, it is important that self-explanatory names and descriptions are able to be added to nodes. However, nodes should only include items important for UI prototyping to avoid unnecessarily cluttering the view.

Not only does the plugin need to help in the creation of UI, it needs to help create UI quickly. This is because prototyping can only occur after the necessary UI has been created. As such, the quicker the UI is complete, the earlier the prototyping process can begin. As previously discussed, UI design patterns and sketching are a great way to achieve this fast UI creation. As such, the plugin will need to implement a UI design pattern library as well as sketching capabilities. The inclusion of sketching would also help with any quick, iterative brainstorming needed. The plugin would then need to only make sure that the creation and deletion of screens, prototype connections, and UI components are easy. By doing so, low-fidelity prototypes can easily be created and destroyed to facilitate the iterative process.

3.3 Breakdown of Figma Plugin

Taking into consideration the solution objectives and requirements, the purpose of the plugin is to create a graph which can be used as a user flow in addition to providing prototyping functionality by controlling prototype paths, generating layouts and elements from UI design patterns, and providing shortcuts to the sketching tool as shown in Figure 3.2.

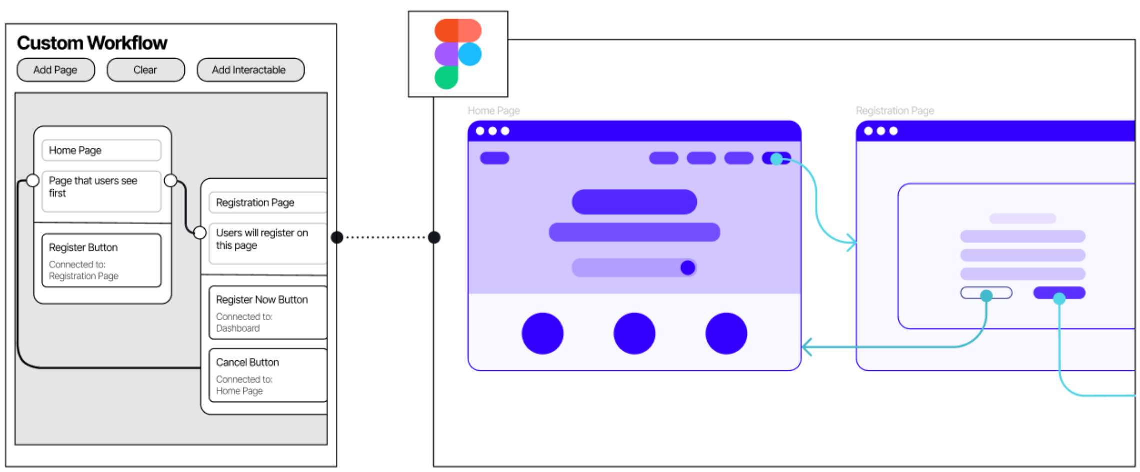


Figure 3.2: Interaction between the plugin and Figma

3.3.1 Graphs

Graphs were used in order to easily manage the prototyping paths between different screens. Each node of the graph consists of the information related to the screen, including the name for the screen, a description about the screen, and the interactive elements within the screen as seen in Figure 3.3. These nodes can be added using the "Add Screen" button or removed by right clicking on a node and pressing the X button. Additionally, each node has one input and a varying amount of output connectors. The output connectors correspond to the order in which the interactive elements are shown within the node. For example, in Figure 3.3, the second output connector from the top corresponds to the About interactive element.

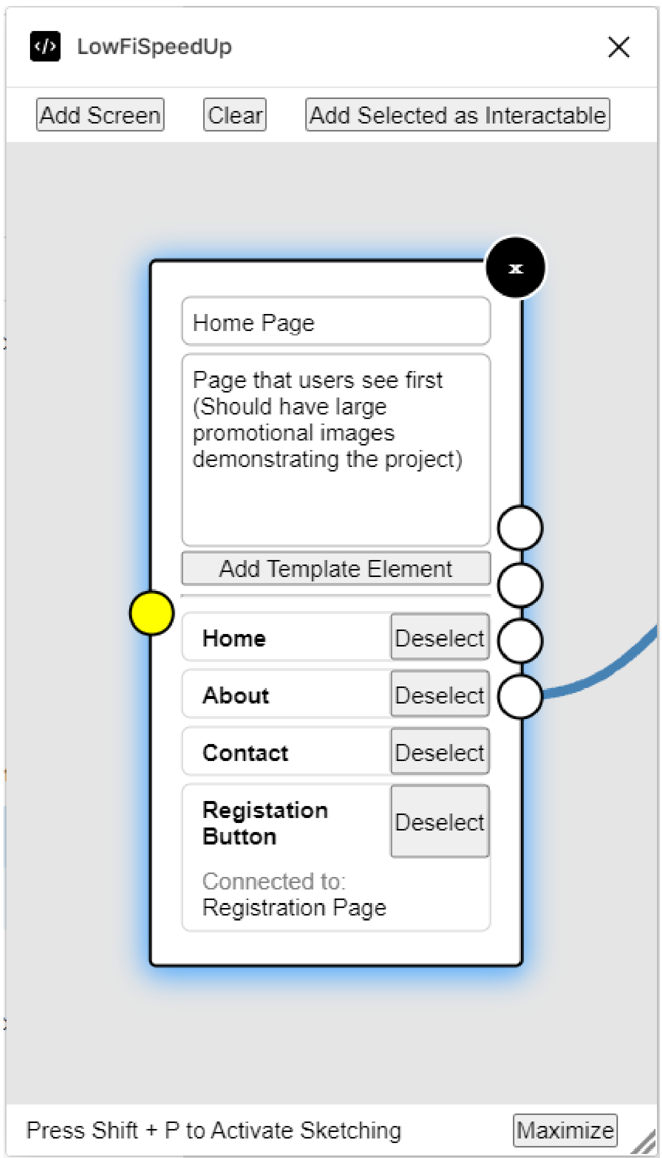


Figure 3.3: Example of a node in the plugin

3.3.1.1 Prototyping Between Screens

As previously mentioned, each node will have input and output connectors. These connectors can be linked together in order to create prototyping paths in Figma. For example, in Figure 3.4, the Registration Button interactive element on the Home Page is linked to the Registration Page screen and has established a prototype connection in Figma. This prototyping connection can be easily removed with the removal of the connection using a right click, deletion of the target screen, or deletion/deselection of the interactive element.

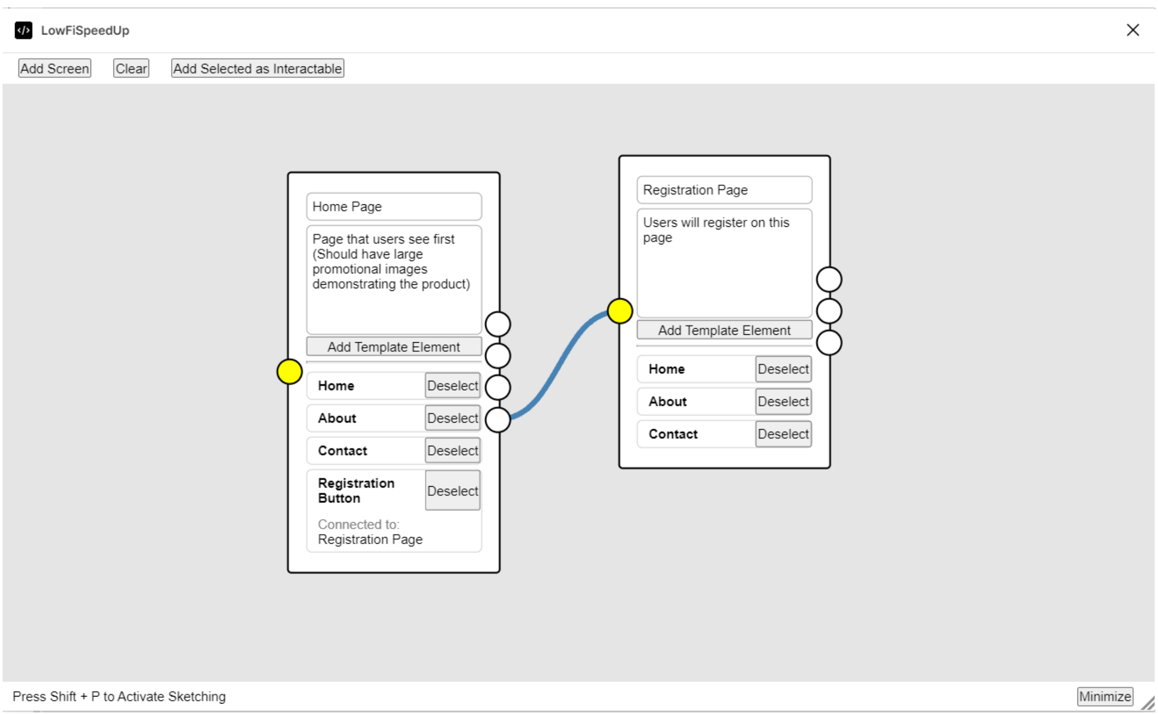


Figure 3.4: Example of prototyping in the plugin

3.3.2 UI Design Patterns

In order to speed up the UI creation process, the creation of a new screen in the plugin will prompt the user to choose from a selection of UI design layouts as shown in Figure 3.5. Similarly, if the user wishes to quickly create elements, they are able to choose from UI design components as seen in Figure 3.6. With the generation of base layouts and UI components, users should be able to quickly create their desired UI. The generation of UI layouts and components will automatically populate the node with default interactive elements allowing for quick prototyping.

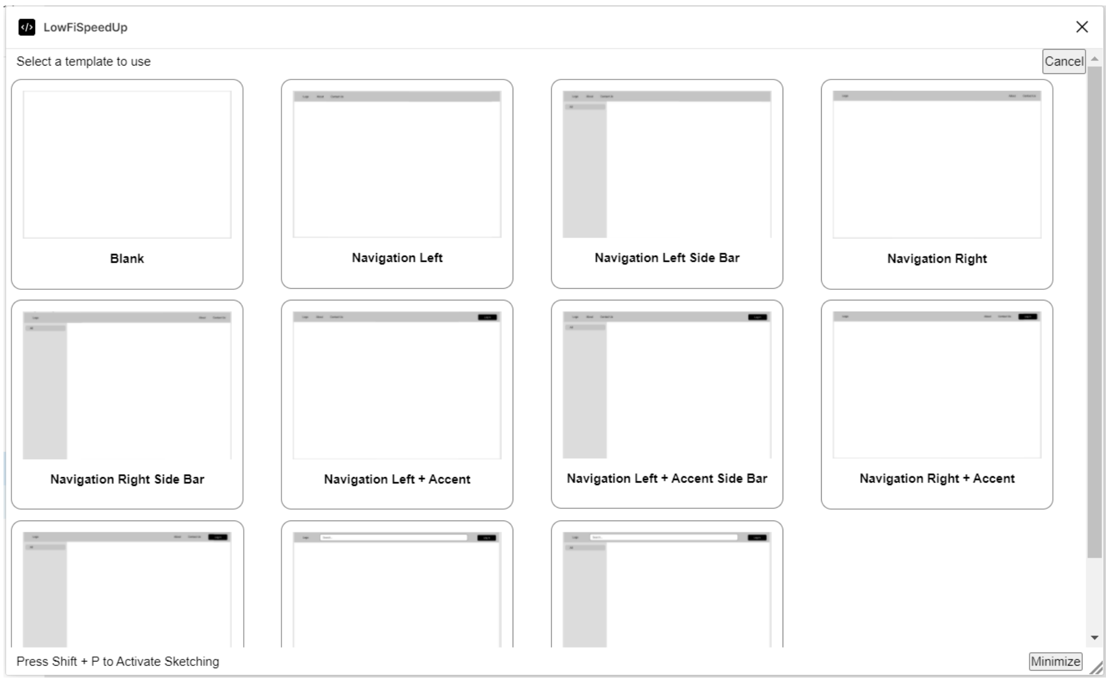


Figure 3.5: Selection of UI design layouts offered by plugin for new screens

3.3.3 Sketching and Built-in Tools

Sketching can be utilized through Figma's built-in pencil tool. Users can also choose to use the other built-in tools (such as shapes, text, etc.) to create elements that the UI design patterns chosen do not provide. Sketches or custom elements can also be used as interactive elements by selecting them and adding the element to a node through the "Add Selected as Interactable" button as shown in Figure 3.7.

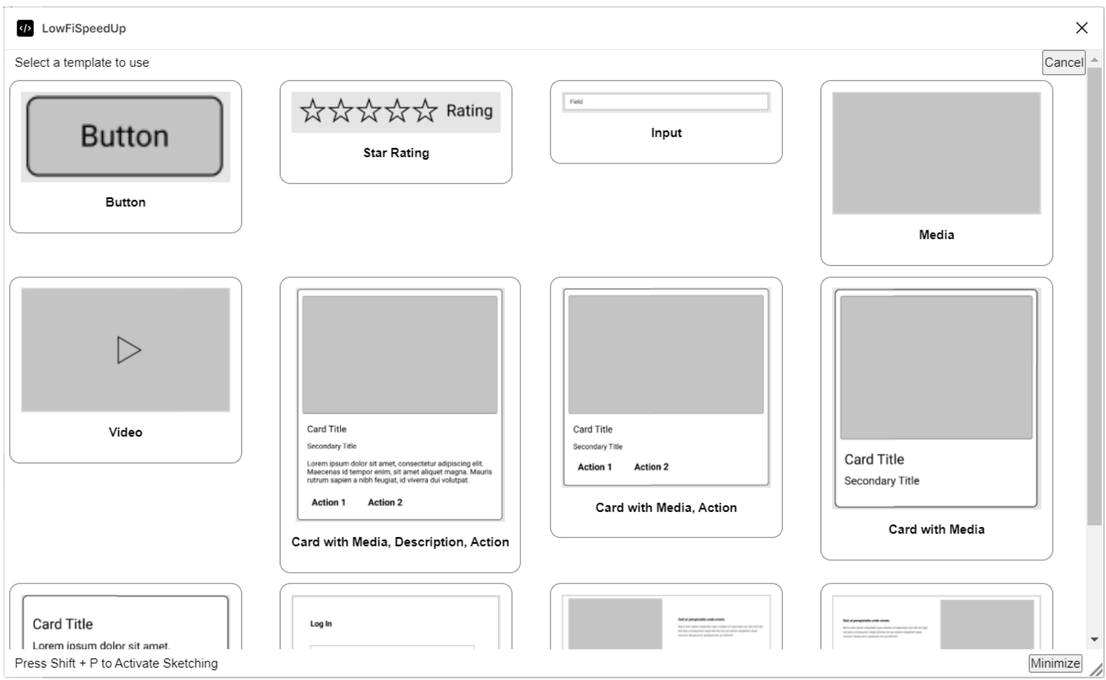


Figure 3.6: Selection of UI design components offered by plugin

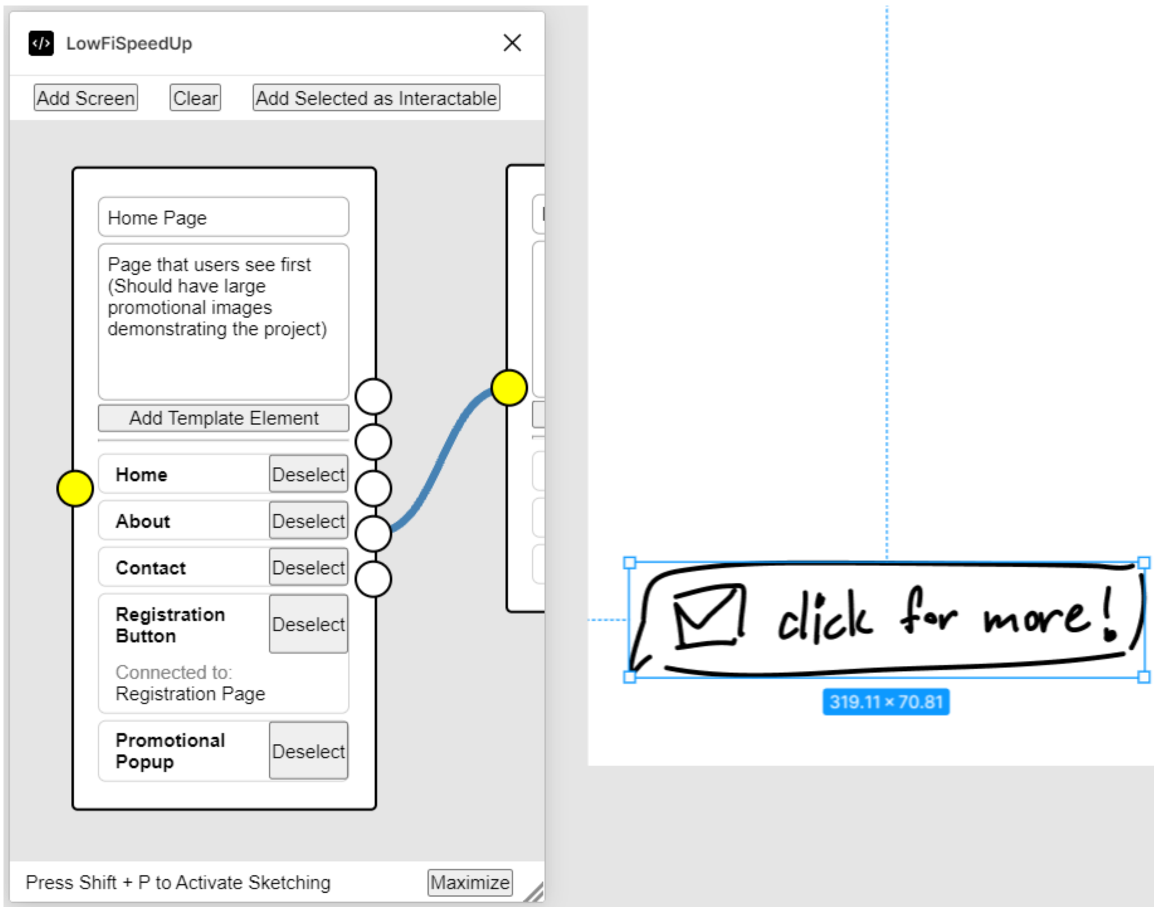


Figure 3.7: Example of adding custom interactive elements in the plugin

4. RESULTS

4.1 Study Goals

The main goal of the user study is to determine if the custom workflow plugin can speed up the prototyping speed compared to using the base tool of Figma. Given that the user study is open to those without any prior knowledge of prototyping UI, the study will also look into the ease of learning and use for each tool.

4.2 Study Participants

Before beginning the study, participants were first asked to complete a pre-study survey. This survey asked them about their age range, gender, and self-reported understanding of UI design, UI prototyping, and Figma on a seven point scale. From the pre-study survey, the following information was found:

- All 28 participants fell under the 18 to 24 years of age bracket
- Participants consisted of 18 females, 9 males, and 1 non-binary person
- The UI design experience among participants were very mixed as seen in Figure 4.1
- Most participants had little to no UI prototyping experience as seen in Figure 4.2
- Most participants had little to no experience using Figma as seen in Figure 4.3

4.3 Study Procedures

During the study, participants were asked to completed two UI creation and prototyping tasks for both Figma and the custom workflow plugin. In order to reduce order bias, the first tool introduced to each participant alternated between Figma and the custom workflow plugin. A tutorial is given about the simple usage of the tool before participants are introduced to their task. For each task, participants are asked to create and prototype a series of UI screens for an imaginary client. The first task involves prototyping a simple product website in order to familiarize the

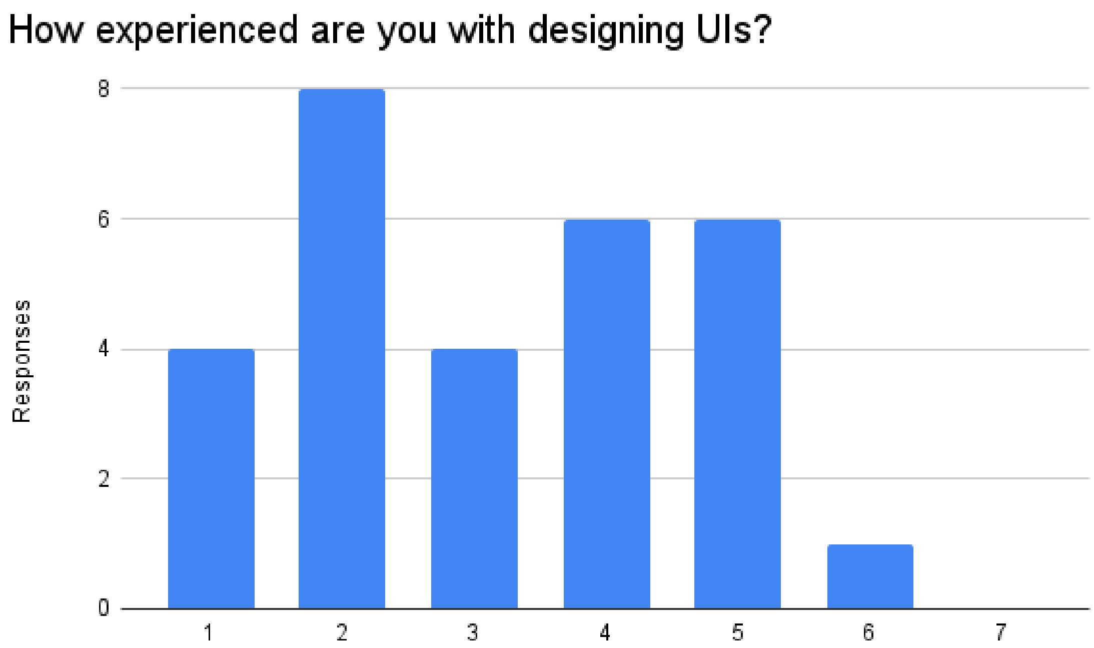


Figure 4.1: Participant UI design experience responses

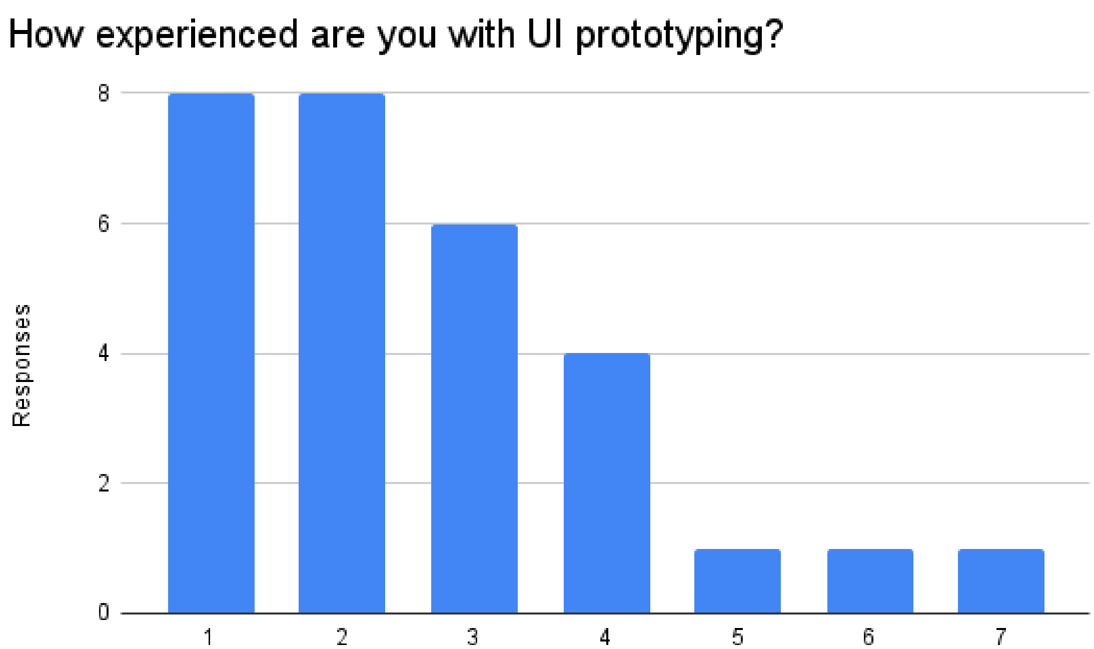


Figure 4.2: Participant UI prototyping experience responses

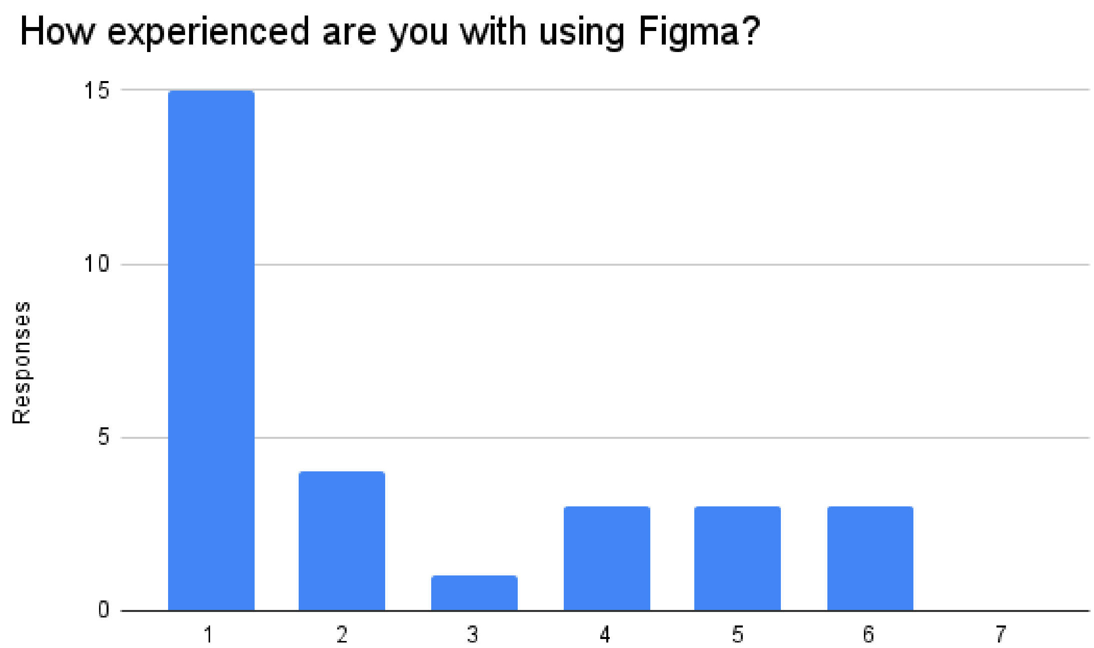


Figure 4.3: Participant Figma experience responses

participant with the tool they are using. The second task involves prototyping a simple online store in order to see how well the participants do after getting to know the tool. Due to the focus of the study being about UI prototyping and not UI design, UI design tasks such as the user flow describing how the screens interact with each other as well as a design layout of how the final screens look were given to participants. Additionally, the same UI design pattern library was given to both Figma and the custom workflow plugin in order to reduce bias caused by having to start from scratch. Participants were also given the opportunity to ask questions throughout the tasks in order to help them out. The time taken to do each task was then documented. In total, participants completed four tasks. After each task, participants were asked to complete a NASA-TLX to measure the perceived workload of each task.

We chose to use the NASA-TLX due to its widespread use and its focus on individual performance for a task [35, 36]. The NASA-TLX uses six sub-scales relating to the workload factors of mental demand, physical demand, temporal demand, performance, effort, and frustration level as seen in Table 4.1 [37]. For our study, the six factors are measured on a seven point scale to derive an estimate of the workload. Usually the values from the six sub-scales will be weighted in the NASA-TLX, but for this study we will be using the Raw TLX (RTLX) which eliminates the weighting process in the NASA-TLX. We used the Raw TLX due to its simplicity and its comparability to the original version of NASA-TLX [38].

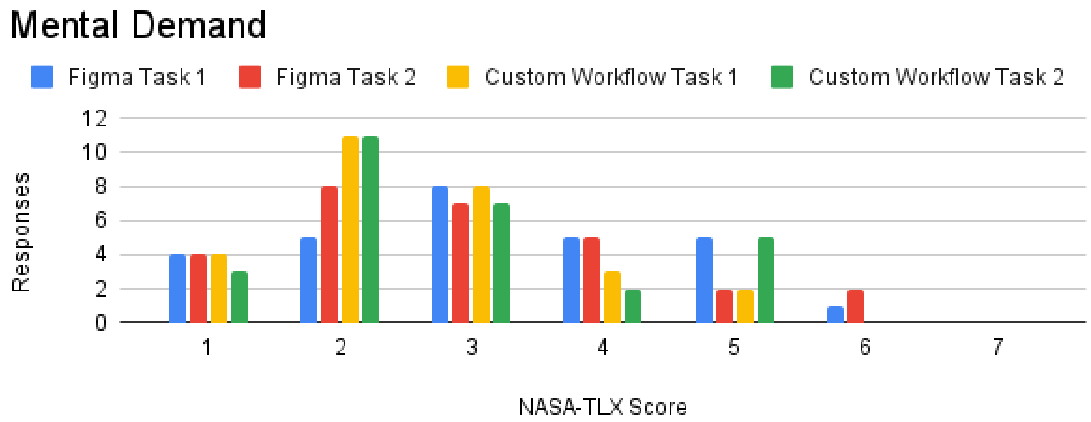
Table 4.1: NASA-TLX six sub-scales endpoint values and description

Title	Endpoints	Description
Mental Demand	Very Low / Very High	How mentally demanding was the task?
Physical Demand	Very Low / Very High	How physically demanding was the task?
Temporal Demand	Very Low / Very High	How hurried or rushed was the pace of the task?
Performance	Perfect / Failure	How successful were you in accomplishing what you were asked to do?
Effort	Very Low / Very High	How hard did you have to work to accomplish your level of performance?
Frustration	Very Low / Very High	How insecure, discouraged, irritated, stressed and annoyed were you?

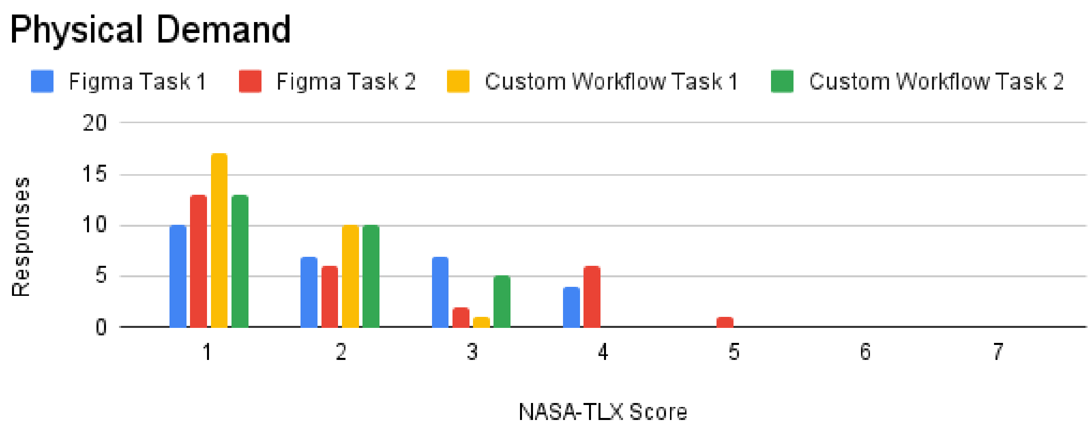
At the end, a post-study survey was given which asks a seven point likert scale questions about the learning curve of each tool. Long answer form questions were then asked about any previous knowledge that may have helped with the use of the tools and any comments regarding the usability of the custom workflow plugin compared to Figma.

4.4 Study Results

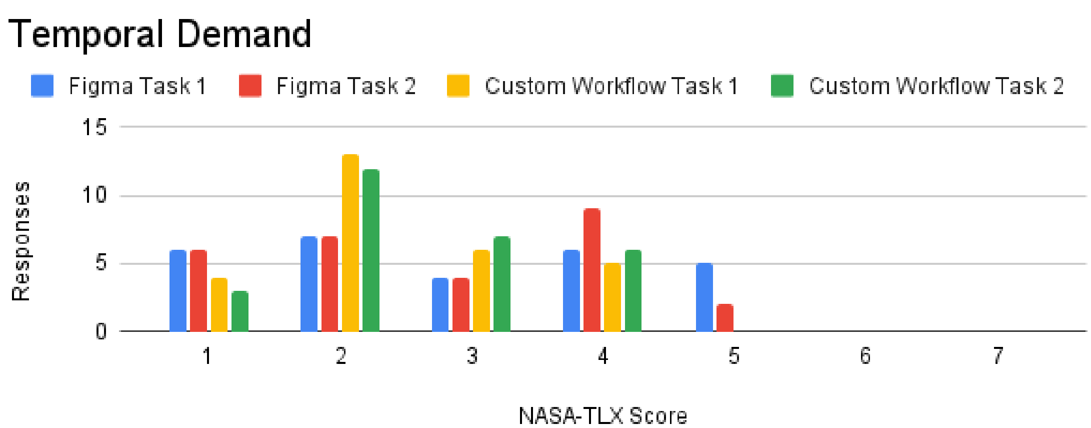
The results of the study can be found in Figure 4.4, Figure 4.5, Table 4.2, Table 4.3, and Table 4.4.



(a)

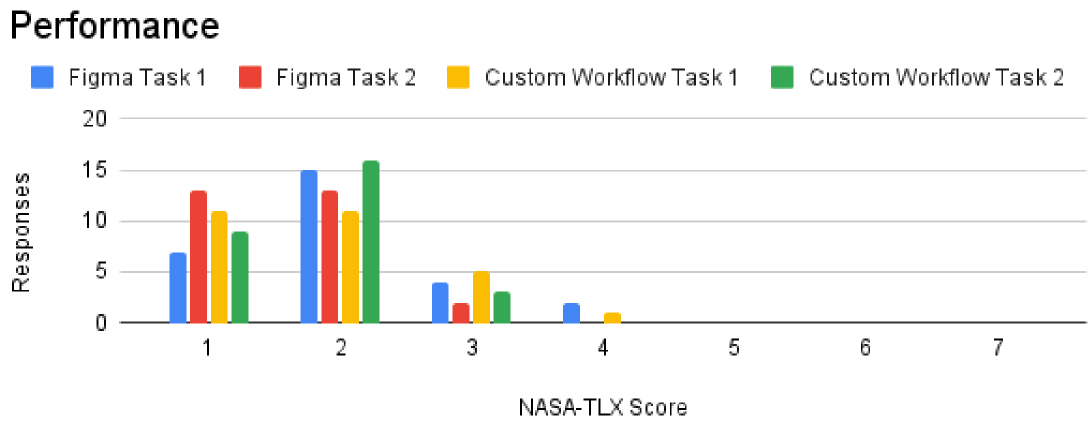


(b)

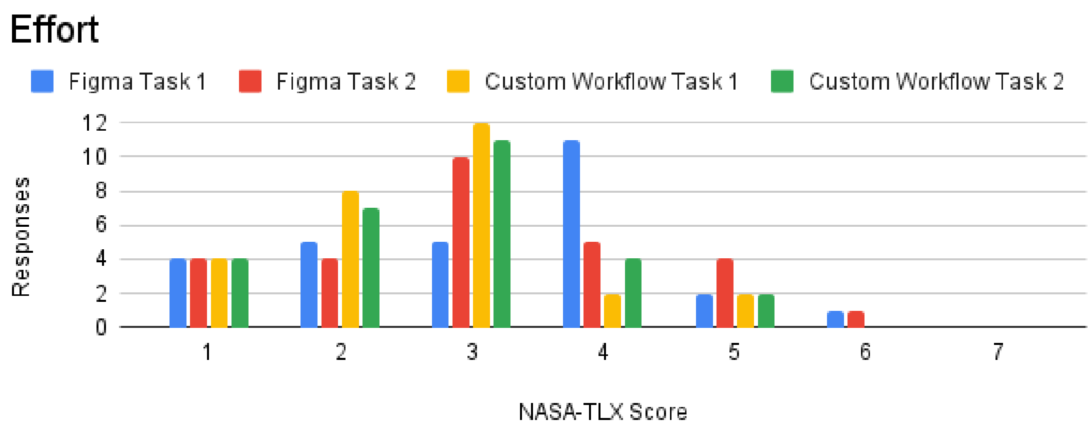


(c)

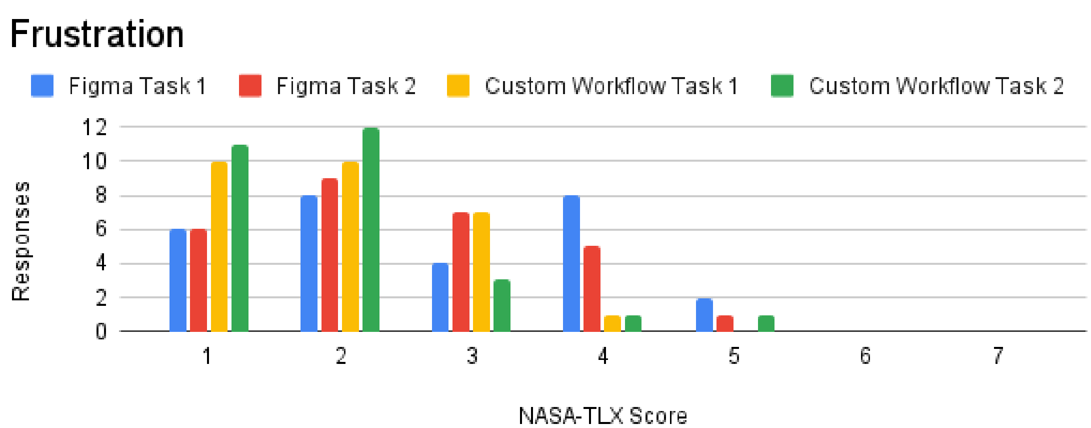
Figure 4.4: Raw NASA-TLX score distribution for the sub-scales of (a) mental demand, (b) physical demand, and (c) temporal demand



(d)



(e)



(f)

Figure 4.4: Raw NASA-TLX score distribution for the sub-scales of (d) performance, (e) effort, and (f) frustration

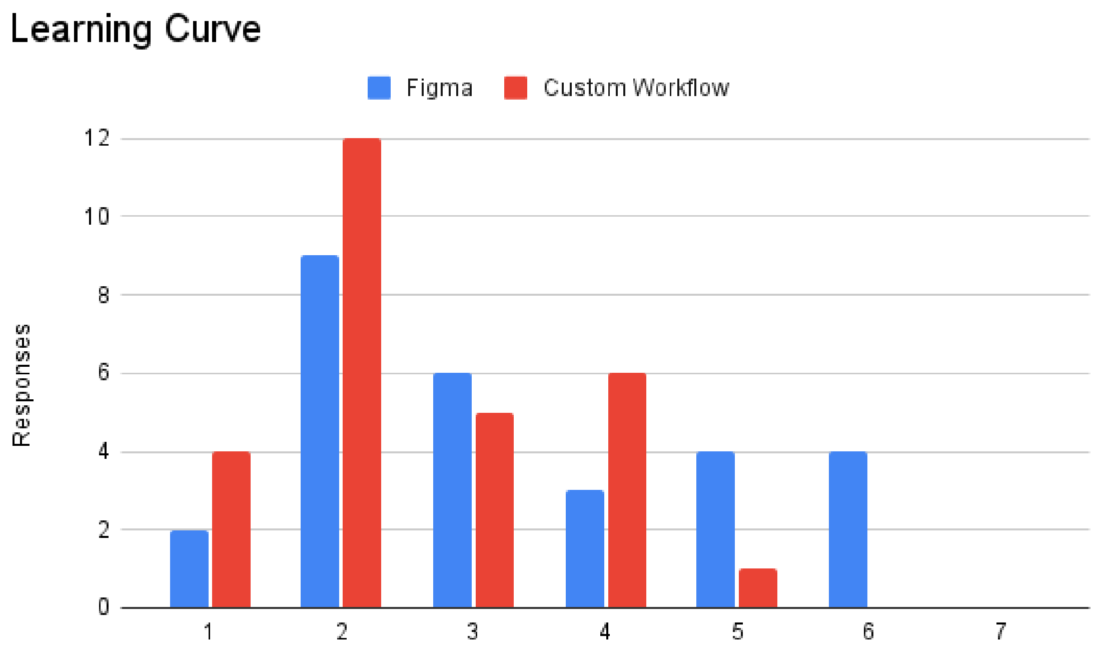


Figure 4.5: Learning curve scores of Figma and Custom Workflow

Table 4.2: Analysis of previous knowledge that helped with the usage of (a) Figma and (b) the custom workflow

What helped	Number of times mentioned
Other applications	10
Figma knowledge	9
Shortcuts knowledge	6
Prototype experience	3
Web usage knowledge	2
Block programming	1
UI components	1

(a) Figma

What helped	Number of times mentioned
Other applications	9
Figma knowledge	4
Diagram experience	3
Shortcuts knowledge	2
Block programming	2
UI components	2
Intuitive interface	2
Prototype experience	1
Visual programming	1

(b) Custom Workflow

Table 4.3: Applications that were mentioned that helped with usage of (a) Figma and (b) the custom workflow

Application Name	Number of times mentioned
Canva	2
Google Slides	1
Adobe Photoshop	1
Unity	1
Adobe Illustrator	1
Wix	1
Miro	1
Lucid Chart	1

(a) Figma

Application Name	Number of times mentioned
Unreal blueprint	2
Canva	1
LabView	1
Unity	1
Blender	1
Scratch	1
Wix	1
Node RED	1
Miro	1
Lucid Chart	1

(b) Custom Workflow

Table 4.4: Common comments regarding custom workflow compared to Figma

Comment	Number of times mentioned
Easy to add UI design patterns	12
Easy connections	8
Liked modular design	7
Annoying to resize	6
Connector not aligned	5
User friendly	4
Ease of navigation	3
Felt efficient	2

5. DISCUSSION

As stated before, the goal of the study is to determine whether the custom workflow plugin can speed up the prototyping speed compared to the base Figma tool as well as to assess ease of learning and use.

5.1 The Impact on Prototyping Speed

Looking at the documented times, we can not conclude if the custom workflow can speed up the UI prototyping process of low-fidelity wireframes compared to Figma. As seen in Table 5.1, the p-value is too high for us to confidently say that the custom workflow significantly changes the time taken to prototype UI wireframes. However, this minimal impact on prototyping speed could be due to the tasks being too simple as the low NASA-TLX scores in Figure 4.4 suggests.

Table 5.1: Two-tailed t-test p-values derived from documented times taken to do each task

	p-value
Task 1	0.822
Task 2	0.235
Both	0.626

5.2 Ease of Learning

Both Figma and the custom workflow plugin seem to have a low learning curve as shown in Figure 4.5. Previous knowledge of Figma and relevant concepts for UI, application use, and abstract thinking helped participants use Figma and the custom workflow plugin as seen in Table 4.2. In fact, it seems that the most useful knowledge for quickly learning Figma and the custom workflow plugin is through the experience of using other similar applications. From the list seen in Table 4.3, participants benefit from knowing other design applications like Canva [39], Adobe Photoshop [40], and Adobe Illustrator [41] when it comes to using Figma. We believe that even though these are all different applications, applications geared toward design may have similar structures and keyboard shortcuts which are then able to be translated to the usage of Figma. On the other hand, participants who knew node-based applications like Unreal blueprints [42], LabView [43], and Unity [44] were able to quickly adapt to the usage of the custom workflow plugin which also uses nodes.

5.3 Ease of Use

According to the NASA-TLX, there does seem to be a difference in the subjective workload between Figma and the custom workflow. As seen in Table 5.2, $p < 0.05$ for physical demand, temporal demand, and frustration, showing that the use of the custom workflow plugin had a statistically significant effect on these scores. Looking at the distributions in Figure 4.4, we can see that the custom workflow scores skew towards a lower value, suggesting that the custom workflow plugin helps lower the subjective workload in terms of the physical demand, temporal demand, and frustration.

Table 5.2: Chi-squared p-values for NASA-TLX sub-scale scores

Sub-scale	p-value
Mental Demand	0.2171
Physical Demand	0.0073
Temporal Demand	0.0102
Performance	0.8879
Effort	0.0716
Frustration	0.0159

As seen in Table 4.4, in terms of usability the top three features participants commented about was the ease of adding UI design patterns, prototype connections, and the modular design of the custom workflow which extracts the essential prototyping information. However, participant comments also reveal that the usability could still be improved as the size of the custom workflow plugin can be annoying to adjust. The location of output connectors in the current design can also lead to confusion as they are not adjacent to their respective interactive element.

6. FUTURE WORK

6.1 Usability Improvements

Before any future studies can be undertaken, the custom workflow plugin needs to be improved. As seen in Table 4.4, many flaws were pointed out about the custom workflow plugin's usability. None of these flaws hindered the participants abilities to create and prototype UI wireframes, but it did cause inconveniences that affected the participants experience of the proposed custom workflow plugin. Therefore, improvements will need to be made before any future studies take place

6.1.1 Expanding Functionality

Due to limited time, some functionalities of the custom workflow plugin were not fully developed. One of the functionalities was the implementation of UI design patterns. For the user study, the UI design patterns presented were only a select few. This was necessary for the user study since implementing more patterns would unnecessarily burden participants with learning all the various UI design patterns rather than quickly creating and prototyping UI wireframes. However, in order for the tool to be useful for the prototyping of a variety of low-fidelity UI wireframes and not just the ones presented in the user study, more UI design patterns need to be included. Ideally, these UI design patterns would form a UI design pattern language which can be used to generate any UI necessary in a given domain [26]. However, an immense amount of time is needed to create a good pattern language which may not even be adopted by users [45]. As such, we recommend including a UI design pattern library within the custom workflow plugin instead. These UI design pattern libraries, such as Material Design [46] and Pttrns [47], are already being used in industry and are able to help designers speed up their work [48].

6.2 Future Study

The focus of the current study was to see how the proposed custom workflow would impact prototyping speed. However, in order to determine how well the proposed custom workflow solution performs in industry, it would be beneficial to see how the custom workflow plugin fairs in an actual UI design process from the starting ideation phase to the final low-fidelity prototype. In this environment, all the functionalities of the tool can be tested and a closer observation of how designers, developers, and stakeholders interact with the custom workflow plugin can be documented.

7. CONCLUSION

Our research proposed a custom workflow Figma plugin which takes advantage of previously researched methods of scenario-based design, UI design patterns, and sketching in order to combine the user behavior and low-fidelity wireframe prototyping stages into one. We then conducted a user study which explored whether our proposed prototyping workflow would speed up the prototyping process while also offering an easy to learn and use tool. From the study, we could not conclude that the custom workflow plugin could speed up the prototyping of low-fidelity UI wireframes. However, we were able determine that the custom workflow plugin had a low learning curve and was able to lighten the workload for physical demand, temporal demand, and frustration. Additionally, the custom workflow plugin was able to demonstrate great usability when it came to adding UI design patterns and creating prototype connections.

From these results, the custom workflow plugin has showen that it is easy to learn and use even for users with no experience in UI design or prototyping. As such, it has the potential of being able to support both designers and stakeholders through the entire low-fidelity wireframe prototyping phase.

REFERENCES

[1] I. Kreimer and B. Hodgson, "What is ui design?: A 2022 guide to user interface design." https://maze.co/collections/ux-ui-design/ui-design, May 2021. Accessed on January 17, 2022.

[2] G. Tai, "A communication architecture from rapid prototyping," in *Proceedings of the 2005 Workshop on Human and Social Factors of Software Engineering*, HSSE '05, (New York, NY, USA), p. 1–3, Association for Computing Machinery, 2005.

[3] A. S. Carter and C. D. Hundhausen, "How is user interface prototyping really done in practice? a survey of user interface designers," in *2010 IEEE Symposium on Visual Languages and Human-Centric Computing*, pp. 207–211, 2010.

[4] N. Babich, "Prototyping 101: The difference between low-fidelity and high-fidelity prototypes and when to use each." https://blog.adobe.com/en/publish/2017/11/29/prototyping-difference-low-fidelity-high-fidelity-prototypes-use, Nov 2017. Accessed on January 17, 2022.

[5] J. Downs, "Low vs high fidelity prototypes: A complete breakdown." https://www.justinmind.com/blog/low-fidelity-vs-high-fidelity-prototypes/, Feb 2021. Accessed on March 28, 2022.

[6] E. Gerber and M. Carroll, "The psychological experience of prototyping," *Design Studies*, vol. 33, no. 1, pp. 64–84, 2012.

[7] M. Walker, L. Takayama, J. Landay, and Leila, "High-fidelity or low-fidelity, paper or computer choosing attributes when testing web prototypes," *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 46, 09 2002.

[8] T. Palmer and J. Bowman, "2021 design tools survey." https://uxtools.co/survey-2021, 2021. Accessed on January 17, 2022.

[9] "Figma." https://www.figma.com/.

[10] "Adobe XD." https://www.adobe.com/products/xd.html.

[11] "InVision." https://www.invisionapp.com/.

[12] "ProtoPie." https://www.protopie.io/.

[13] "Sketch." https://www.sketch.com/.

[14] "Balsamiq." https://balsamiq.com/.

[15] M. B. Rosson and J. M. Carroll, "Scenario-based design," in *Human-computer interaction*, pp. 161–180, CRC Press, 2009.

[16] J. Rumbaugh, I. Jacobson, and G. Booch, *Unified Modeling Language Reference Manual, The (2nd Edition)*. Pearson Higher Education, 2004.

[17] H. Koc, A. Erdoğan, Y. Barjakly, and S. Peker, "Uml diagrams in software engineering research: A systematic literature review," *Proceedings*, vol. 74, p. 13, 03 2021.

[18] F. Ciccozzi, I. Malavolta, and B. Selic, "Execution of uml models: a systematic review of research and practice," *Software & Systems Modeling*, vol. 18, no. 3, pp. 2313–2360, 2019.

[19] M. Elkoutbi, I. Khriss, and R. K. Keller, "Automated prototyping of user interfaces based on uml scenarios," *Automated Software Engineering*, vol. 13, no. 1, pp. 5–40, 2006.

[20] I. Herman, G. Melancon, and M. Marshall, "Graph visualization and navigation in information visualization: A survey," *IEEE Transactions on Visualization and Computer Graphics*, vol. 6, no. 1, pp. 24–43, 2000.

[21] H. Dammak, A. Dkhil, and M. Gardoni, "Graph-based tools for ecm search result analysis to support the ideation step," in *Product Lifecycle Management in the Digital Twin Era* (C. Fortin, L. Rivest, A. Bernard, and A. Bouras, eds.), (Cham), pp. 170–180, Springer International Publishing, 2019.

[22] "What is a sitemap for a website? (& what they do for seo)." https://slickplan.com/sitemap/what-is-a-site-map, Nov 2021. Accessed on April 2, 2022.

[23] K. Kusano, M. Nakatani, and T. Ohno, "Scenario-based interactive ui design," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, (New York, NY, USA), p. 391–394, Association for Computing Machinery, 2013.

[24] A. Dkhil, H. Dammak, M. Mouatassim, and M. Gardoni, "A new approach to ideation based on linkography combined with graph theory," *International Journal on Interactive Design and Manufacturing (IJIDeM)*, pp. 1–16, 2022.

[25] A. Seffah, "The evolution of design patterns in hci: From pattern languages to pattern-oriented design," in *Proceedings of the 1st International Workshop on Pattern-Driven Engineering of Interactive Computing Systems*, PEICS '10, (New York, NY, USA), p. 4–9, Association for Computing Machinery, 2010.

[26] C. Kruschitz and M. Hitz, "Analyzing the hci design pattern variety," in *Proceedings of the 1st Asian Conference on Pattern Languages of Programs*, AsianPLoP '10, (New York, NY, USA), Association for Computing Machinery, 2010.

[27] S. Suleri, N. Kipi, L. C. Tran, and M. Jarke, "Ui design pattern-driven rapid prototyping for agile development of mobile applications," in *Proceedings of the 21st International Conference on Human-Computer Interaction with Mobile Devices and Services*, MobileHCI '19, (New York, NY, USA), Association for Computing Machinery, 2019.

[28] S. Suleri, Y. Hajimiri, and M. Jarke, "Impact of using ui design patterns on the workload of rapid prototyping of smartphone applications: An experimental study," in *22nd International Conference on Human-Computer Interaction with Mobile Devices and Services*, MobileHCI '20, (New York, NY, USA), Association for Computing Machinery, 2020.

[29] J. A. Landay and B. A. Myers, "Interactive sketching for the early stages of user interface design," in *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 43–50, 1995.

[30] M. W. Newman and J. A. Landay, "Sitemaps, storyboards, and specifications: A sketch of web site design practice," in *Proceedings of the 3rd Conference on Designing Interactive Systems: Processes, Practices, Methods, and Techniques*, DIS '00, (New York, NY, USA), p. 263–274, Association for Computing Machinery, 2000.

[31] J. Lin, "A visual language for a sketch-based ui prototyping tool," in *CHI '99 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '99, (New York, NY, USA), p. 298–299, Association for Computing Machinery, 1999.

[32] J. Lin, M. Thomsen, and J. A. Landay, "A visual language for sketching large and complex interactive designs," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '02, (New York, NY, USA), p. 307–314, Association for Computing Machinery, 2002.

[33] A. Coyette, J. Vanderdonckt, and Q. Limbourg, "Sketchixml: A design tool for informal user interface rapid prototyping," in *Proceedings of the 3rd International Conference on Rapid Integration of Software Engineering Techniques*, vol. 4401, pp. 160–176, 09 2006.

[34] V. Odysseos, "Prototypes vs user flows." https://blog.overflow.io/prototypes-vs-user-flows-cc52b6361306, Aug 2018. Accessed on April 1, 2022.

[35] R. A. Grier, "How high is high? a meta-analysis of nasa-tlx global workload scores," in *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 59, pp. 1727–1731, SAGE Publications Sage CA: Los Angeles, CA, 2015.

[36] S. Rubio, E. Díaz, J. Martín, and J. M. Puente, "Evaluation of subjective mental workload: A comparison of swat, nasa-tlx, and workload profile methods," *Applied psychology*, vol. 53, no. 1, pp. 61–86, 2004.

[37] S. G. Hart and L. E. Staveland, "Development of nasa-tlx (task load index): Results of empirical and theoretical research," in *Advances in psychology*, vol. 52, pp. 139–183, Elsevier, 1988.

[38] S. G. Hart, "Nasa-task load index (nasa-tlx); 20 years later," *Proceedings of the human factors and ergonomics society annual meeting*, vol. 50, no. 9, pp. 904–908, 2006.

[39] "Canva." https://www.canva.com/.

[40] "Photoshop." https://www.adobe.com/products/photoshop.html.

[41] "Illustrator." https://www.adobe.com/products/illustrator.html.

[42] "Unreal engine blueprint." https://docs.unrealengine.com/4.27/en-US/ProgrammingAndScripting/Blueprints/.

[43] "Labview." https://www.ni.com/en-us/shop/labview.html.

[44] "Unity." https://unity.com/.

[45] Y. Pan and E. Stolterman, "Pattern language and hci: Expectations and experiences," in *CHI '13 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '13, (New York, NY, USA), p. 1989–1998, Association for Computing Machinery, 2013.

[46] "Material design." https://material.io/design.

[47] "Pttrns." https://www.pttrns.com/.

[48] L. Colusso, C. L. Bennett, G. Hsieh, and S. A. Munson, "Translational resources: Reducing the gap between academic research and hci practice," in *Proceedings of the 2017 Conference on Designing Interactive Systems*, DIS '17, (New York, NY, USA), p. 957–968, Association for Computing Machinery, 2017.

APPENDIX A: USER STUDY TASK 1

Task 1

Background
You've been hired to create a product website that showcases a smart coffee machine that promises to boost employee productivity. The client is looking to have the following screens for potential customers to interact with:

- Landing Page
- About Page
- Contact Page

User Flow
The company wants to make sure the user can access the page in the following ways:

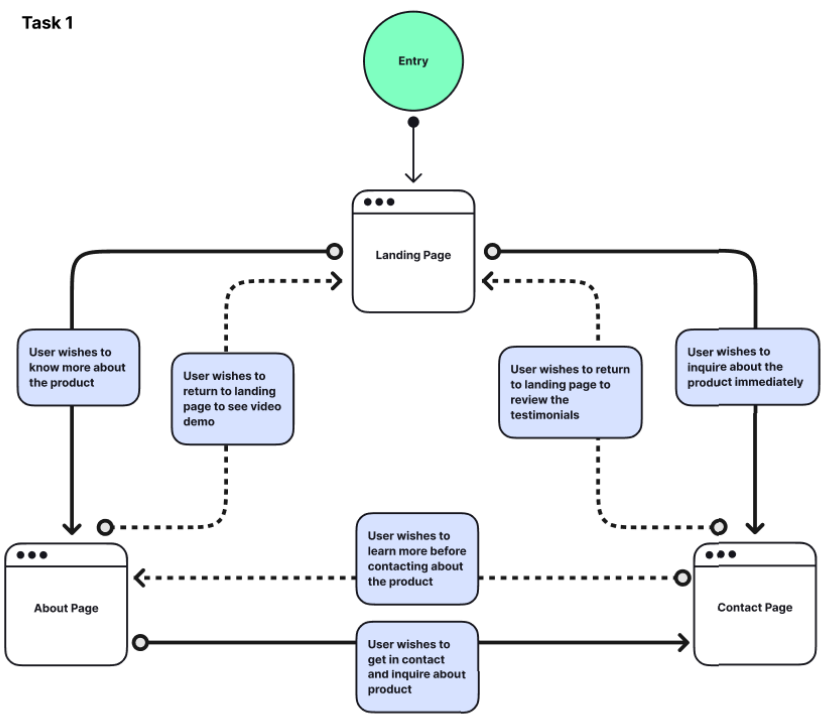


Figure A.1: Page 1 of Task 1 instructions

Design Layout

Landing Page

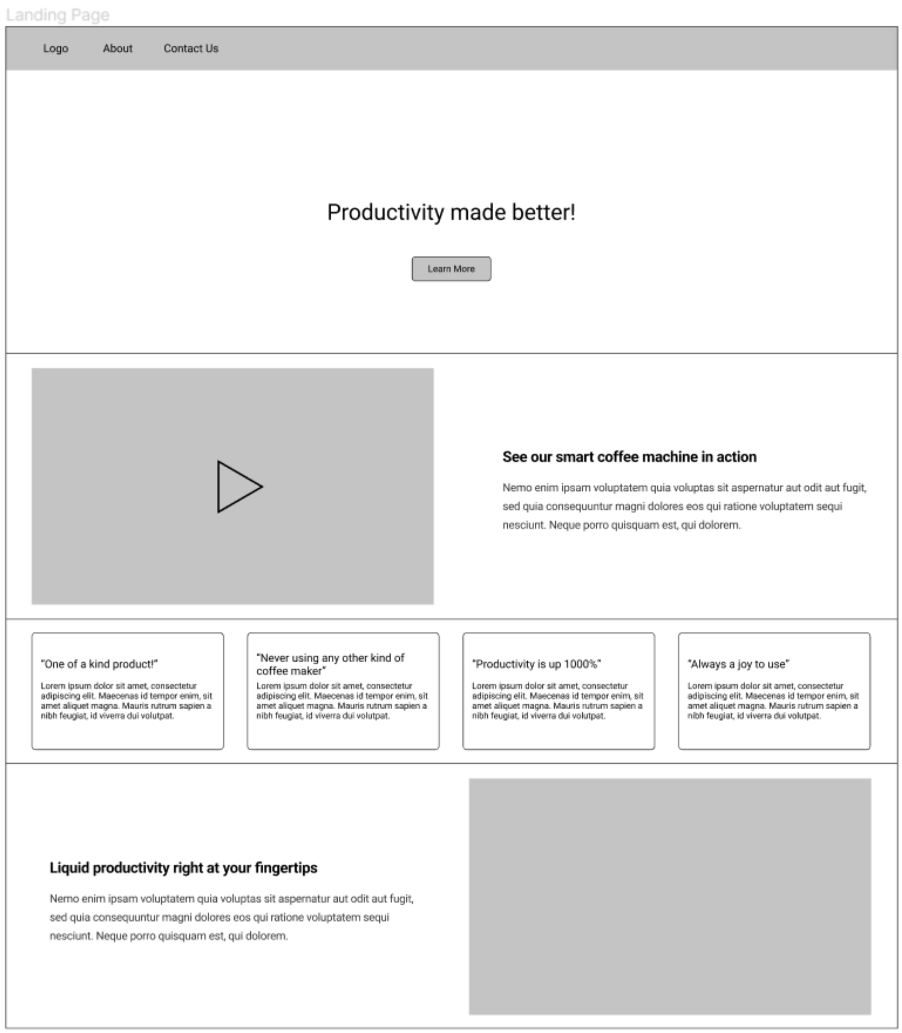


Figure A.2: Page 2 of Task 1 instructions

About Page

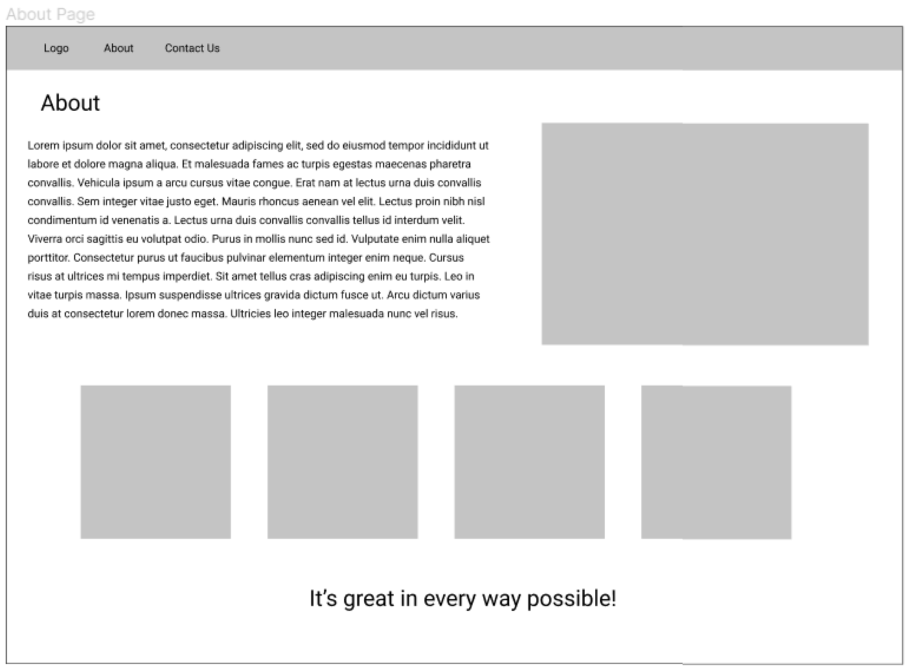
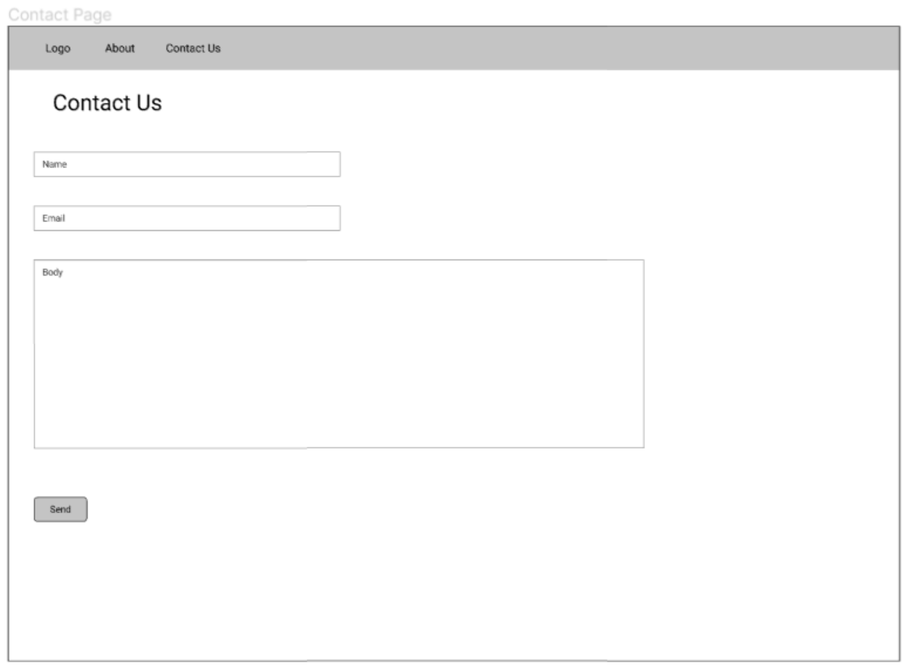


Figure A.3: Page 3 of Task 1 instructions

Contact Page



<u>Goal</u>

Given the above information, create the desired UI (including prototyping the interactions requested).

Figure A.4: Page 4 of Task 1 instructions

APPENDIX B: USER STUDY TASK 2

Task 2

<u>Background</u>
The company found that the website was a great success. With this new success, they want to open an online store. They want to the following screens to be included:

- Catalog Page
- Detail Page for a singular item
- Login Page
- Checkout Page

<u>User Flow</u>
The company wants to make sure the user can access the page in the following ways:

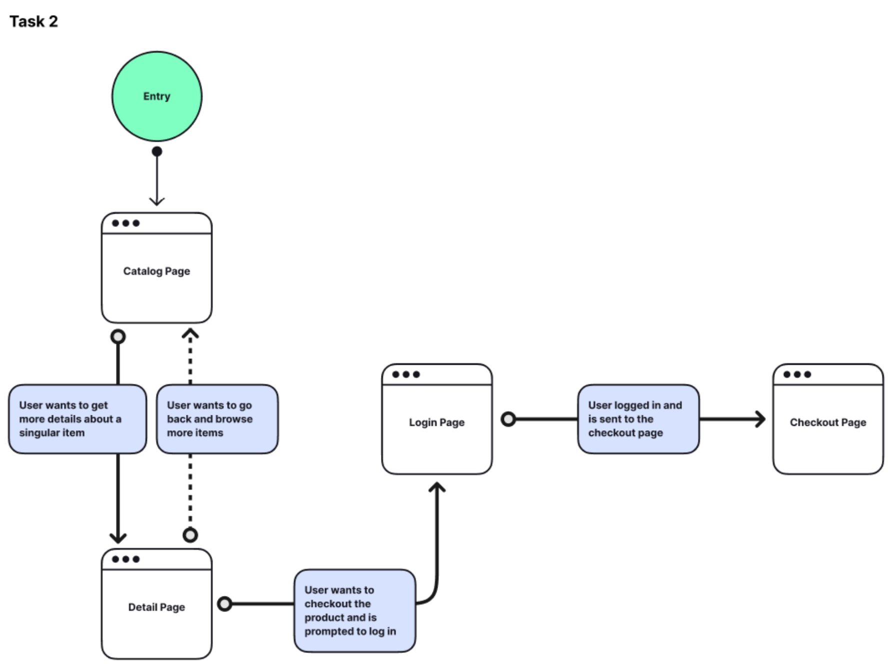


Figure B.1: Page 1 of Task 2 instructions

Design Layout

Catalog Page

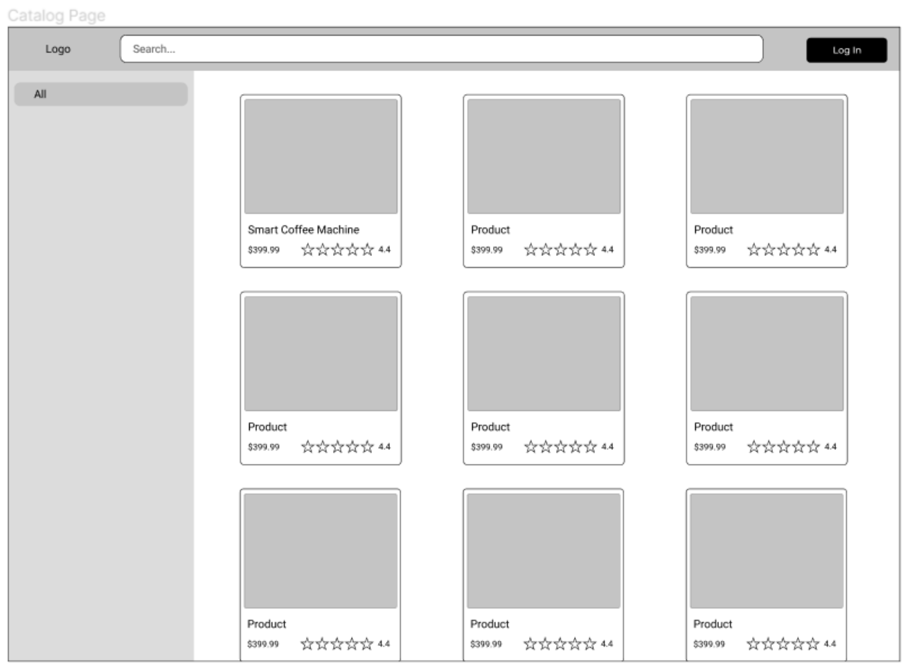


Figure B.2: Page 2 of Task 2 instructions

Detail Page

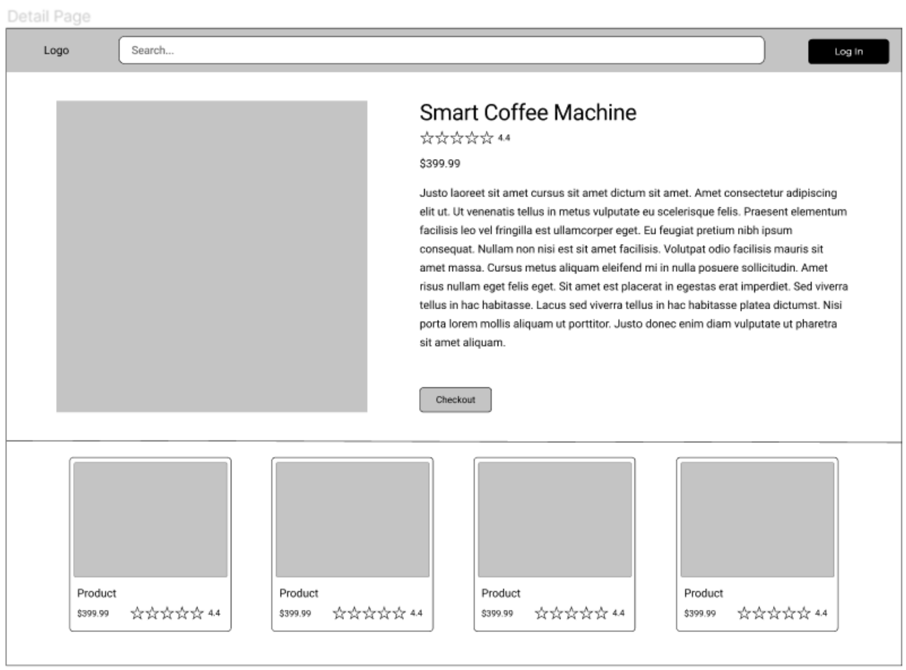


Figure B.3: Page 3 of Task 2 instructions

Login Page

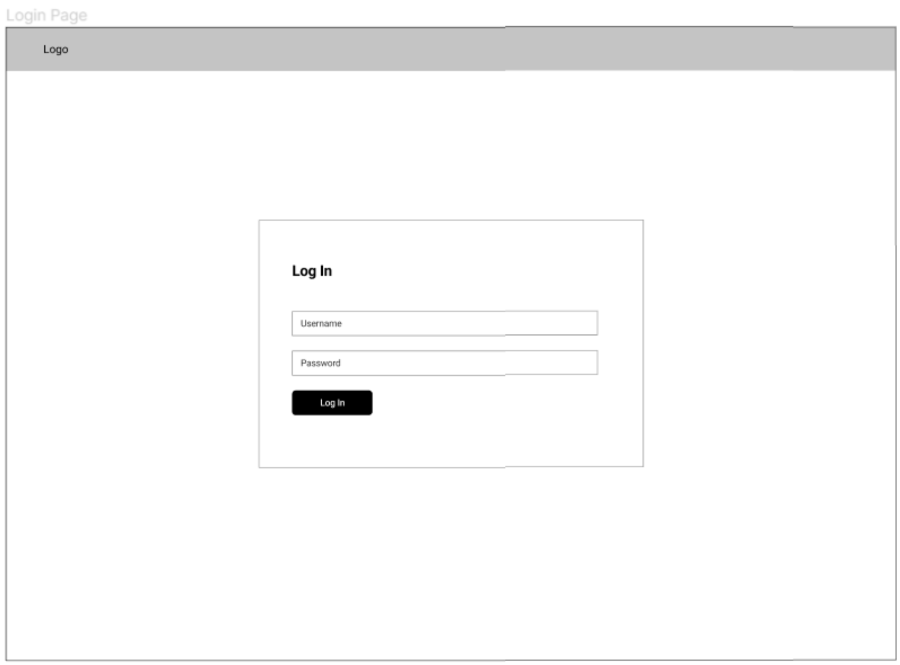
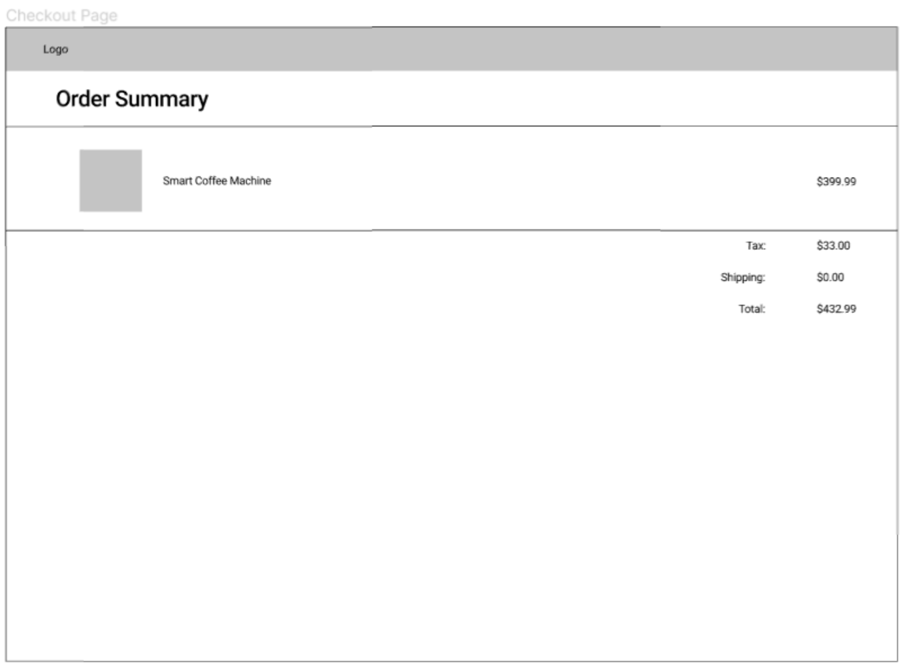


Figure B.4: Page 4 of Task 2 instructions

Checkout Page



Goal

Given the above information, create the desired UI (including prototyping the interactions requested).

Figure B.5: Page 5 of Task 2 instructions

APPENDIX C: USER STUDY DOCUMENTED TIMES

Table C.1: Documented time in minutes for participants to complete each task

Figma Task 1	Figma Task 2	Custom Workflow Task 1	Custom Workflow Task 2
8:58	13:02	14:33	11:41
20:53	17:58	13:25	12:04
22:27	17:13	13:51	14:53
14:04	16:10	16:39	22:36
21:04	17:56	10:58	12:40
12:05	12:20	17:35	18:31
16:13	14:21	12:01	9:54
9:11	11:15	13:22	14:56
20:21	13:07	12:42	11:42
10:50	9:01	14:11	14:40
15:22	12:11	14:57	14:35
11:18	14:15	18:39	18:41
21:49	18:10	15:17	14:09
9:39	8:02	12:34	13:03
25:48	20:47	15:23	16:23
12:52	15:55	21:07	22:44
23:00	17:02	14:22	16:04
13:43	17:05	19:08	22:02
19:11	22:29	14:05	14:42
8:01	11:10	14:28	11:31
24:11	18:28	14:04	17:15
10:22	13:39	17:15	21:58
20:46	24:11	17:54	22:40
13:35	10:35	19:48	17:48
18:39	17:39	16:49	15:31
15:04	12:50	23:14	20:20
15:23	13:47	12:19	13:14
12:26	12:23	19:24	21:52