



# CMPT 363 - PART 3\_TEAM 18

04.08.2022

---

**Manav Patel** – [mjpatel@sfu.ca](mailto:mjpatel@sfu.ca) – 301368020  
**Dhruv Patel** – [djpatel@sfu.ca](mailto:djpatel@sfu.ca) – 301385233  
**Vraj Patel** – [vrajp@sfu.ca](mailto:vrajp@sfu.ca) – 301384968  
**Jordan Lembong** – [lembongj@sfu.ca](mailto:lembongj@sfu.ca) – 301354072  
**Young Been Lee** – [ybl3@sfu.ca](mailto:ybl3@sfu.ca) – 301244854

## Component 1: Cognitive Walkthrough

### Context and scenario

Every university student needs an application to find up-to-date information on a regular basis such as course or campus related information. There are different reasons to use SFU snap, for example, either to search for your classrooms on the first day of a semester, to find assignment deadlines or exam schedules without having to open canvas all the time. Additionally, it helps search for other campus related information such as latest weather conditions, dining places with timings and exact location, library services that allows booking a room or library timings, contact information based on each department, and transit information that shows bus timings from the campus. The application may be used by other general users as well that do not require to login to find campus information. There are some rules they should comply with. Users typically want a faster and reliable way to get information in a timely manner. They may use the app at any time of the day from any location with their smartphones, either with an IOS or Android operating system.

### User identification

The online application is designed mainly for students who are currently studying at SFU. Their courses schedules are integrated into the app when logged in and has various other activities for students to find hangout places. The student who evaluated our design is Parth Patel, who took this course two semesters ago.

<b>Action Sequence:</b>  Our Task for component 1 is: Accept any new message request and chat with that user.	<b>Does the user know what to do given the action?</b>	<b>Can the user find the right interface component to perform this action?</b>	<b>Can the user associate the feedback from the interface to the correct action they perform?</b>	<b>Does the user understand the feedback so that they know where they are in the task after performing the correct action?</b>
<b>Action 1:</b> Go to the socials page and view	Yes, the user knew you had to click on the	Yes, the user was able to navigate to the	Yes, as the display changed	Yes, as the user clicks on the “social”




current chats	socials button from the dashboard to proceed to the socials page.	social page by using the “social” button on the dashboard and clicking on the chat icon on the top right corner of the homepage of the social to go to the “messages” page.	because of their action i.e clicking on the social button & proceeding to the socials page.	button and there is a transition to the “social” page, the toolbar reads “Social”, which helped the user know they are on the Social's page.
<b>Action 2:</b> View and accept any new chat request	No, the user couldn't identify the new chat requests because there was no indication of which chats are new.	No, the user wasn't able to identify the “new chat request” button because it was confused with being a heading instead of a clickable button. Therefore the user was confused if the messages shown on the “Messages” page were new requests or older messages.	Once the user was made aware of the fact that the ribbon was a button, he was able to associate the transition to a new screen as a feedback after performing the correct input. Before being made aware of this, the user clicked on one of the names and it took the user to a chat screen which confused the user. He thought he had accepted a message request and was directed to	No, the user initially clicked on the older message and was taken to a chat window that was for an older friend, which the user believed to be the request but there was no option to accept or reject which helped the user know they had made a mistake.  After helping the user locate the “new chat request” button, the user saw “requests” written on the toolbar of the screen which

			the chat page. But in actual he clicked on a friend with older messages and continued the chat.	helped him understand he was doing the right corrective action.
<b>Action 3:</b> Proceed to message the newly accepted chat request User	Yes, the user knew he had to exit the message request page by clicking the back arrow and once he was in the "messages" page, he had to click on the user to go to the chat page for that specific user he added.	Yes, the user was able to click the back arrow button to return to the "messages" page and he clicked on the newly accepted message request to start a chat with that user.	Yes, the user could associate the transition(change of display) to the blank "new message request" page by clicking the accept button on the request.	Yes, once the user clicked the accept button on the message request. The display transitioned to an empty request page, which led the user to believe the request had been accepted and he could now go to the messages page to start a chat with the newly accepted message request of the user.


### General summary of the results

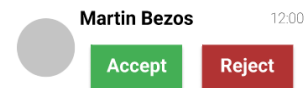
Having received feedback from the user at each step of the walkthrough and having watched the user as he performed each action, we categorized the steps the user took into two categories: The user performed it quickly and the user took time figuring out the step. This helped us draw out the strengths and weaknesses in our design as well as think of future modifications to the current design.

### Strengths:



1. The user was able to find the “chat” button  easily even though it doesn't say that it's for messaging, just through visual display it was easily identified as a messaging section. This is a huge strength because one of our functional requirements was to have icons that the users can easily associate to their everyday life. View [Figure 1](#). For a full screenshot of the social page emphasizing on the “chat” icon.
2. The search bar under the message section  is a key feature, as it can help the user find the person they want to message faster, by typing that person's name in the search button and then clicking on this  icon. View [Figure 2](#), for a full screenshot.

### Weakness:

1. The “New Message Requests” button seems more like a heading,  and so the user was having difficulty finding the new message requests. View [Figure 3](#). For a full screenshot of the “messages” page showing the “new message request” button.
2. The user has no option to view the profile of another user who sent a message request, the only options are either “Accept” or “Reject”. View [Figure 4](#). for a full screenshot of the new message request page.



### Improvement suggestions:

1. Putting a right arrow button on the “New Message Requests” ribbon can help the user easily identify it as a button to a new page showing message requests. View [Figure 5](#). for a full screenshot of the modified design for the “new message request” page. 
2. By clicking on the profile icon of the user , one can view the profile of the person who sent the message request, and this will direct the user to that person's profile page.

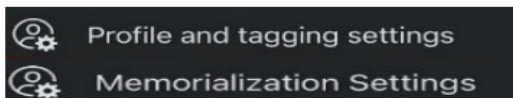
## Component 2: Reflection

As we progress through the design, we learned the different methods we need to keep a focus on while making improvements by empathizing on the purpose, defining our ideas, researching on our user groups, concentrating on usability testing, and then analysing our user personas, and finally designing the interface while keeping a focus on testing by taking feedback.

### Methods used to create a better design

#### Heuristic Evaluation:

An effective heuristic evaluation helped us to measure the usability of user interfaces by independent walkthroughs and reporting issues for applications such as Facebook. We learnt how to evaluate user interfaces by forming different categories to help us enhance our product usability from the early stages in development. From our evaluation, we learnt that giant social media companies like facebook have flaws in their designs that they need to work on to provide a better user experience. We noticed that Facebook has a poor aesthetic and minimalist design from our evaluation. The facebook application is filled with too many functionalities on a single page which loses its main purpose of allowing a user to post. Additionally, some of the icons make the functionalities unclear to understand. Based on these evaluations, we tried to avoid the issues and adapted a good usability in our design to improve the quality of our product.




For example, these icons on facebook have no clear relation to the function, and the functions use the same icon.

Learning from these, we decided to keep the final design of our product as minimalistic as possible. The heuristic evaluation helped us identify and focus on specific issues without having to get feedback from other users. It showed us how it may impact overall user experience and how this may provide quick and beneficial feedback to designers.

#### Requirements Gathering & Specification:

Gathering the requirements and specifications was essential when the usability for our product is designed. To have a clear view of how the final product would give a great user experience, specifying these requirements in a document helped us make a clear overview to specify the users for our interface and find their needs.




After having a clear context identification, user identification, functional and non-functional requirements, we were able figure out how exactly the app needs to be designed based on the information collected. We figured out who the users are, where and when the application will be used, how important of a role would the social aspect of the application play in a user's experience and what part of the application has the most essential aspect. We learnt how to distinguish between functional and non-functional requirements before we would list them down for our interface. We came up with a list of clear and concise requirements for each section and worked on it to better understand the purpose. We learnt how the functional requirements help the system work and identify the missing requirements that our users would need to connect socially with each other. On the other hand, non-functional requirements helped us understand the details on how the product will perform. We also came up with sketches for our context and user identification to better understand the exact audience of our application.

### Prototyping:

Learning from our functional and non-requirements, we were able to create and design our prototypes. Our prototypes helped us develop a sample version in different stages which helped us test different features before we would display the final product. These helped our work flow with ease resulting in an increased speed and value for our design.

Initially we started by creating several low-fidelity prototypes on paper that satisfied the requirements we came up with. We were then able to short-list and pick a few prototypes that would meet a user's needs. We specified how the design met the requirements to get a clear view on how to move forward. We were then able to learn how to use Balsamiq to sketch these low-fidelity prototypes in an effort to get closer to the final product. However, these prototypes only considered the main features of the app. We further had to work in detail to specify each requirement such as what would clicking a specific icon do.

After designing the LFPs, we were able to learn on how to create both horizontal and vertical medium-fidelity prototypes. With our H-MFP, we tried to make it interactive by focusing on the main components such as the main page where all the pages can be found such as the social and events page. Additionally, we worked on to show the main features of the events and social pages. The H-MFP provided us with the essential product concept to create a better design. Things such as what would happen after an



icon is clicked were included in the V-MFP. The V-MFP implemented the features with a meaningful design and enlightened the tasks. Our V-MFP focuses on features such as notifications, favorites, messages and message requests.

With our prototypes, we were able to create a robust design with interactive components that also helped us to handle errors.

### **Cognitive Walkthrough:**

Before moving onto the final product, we need to figure out how the features of the application can be interesting and beneficial for the user. Through a cognitive walkthrough, we were able to understand the context and in what scenario the application would be used, understand who the target users are and finally a task with actions. To better evaluate our prototypes, we were able to get feedback from a student whose experience in UX design helps us to improve our design by improving the feedback given. These helped us walk through each of the steps of the task flow by getting answers to our directed questions. With this goal, we got an incentive for the aspects of our interface that could be challenging to new users. Finally, we learnt what components of the interface need improvements such as making changes to a few of the icons to provide a better and simpler understanding to the user. This helps get closer to the final design of the application.

### **Tools used**

We learnt how to use Balsamiq to design our LFPs to help us understand how to improve our interface by adding different components. Balsamiq helped us by making the design flow easier by letting us draw, add essential components, and add additional comments to the design for better understanding. It simplified our pathway to creating our MFP design.

Figma on the other hand is a powerful tool for prototyping. However, we did face a few difficulties while designing our MFPs. Looking at the complexity of our design, we had to learn within a limited time to improve our experience to use Figma. We did have difficulty finding icons and images since Figma does not provide in-built images apart from the simple shapes. We had to manually work through to find icons and import files to create the prototypes. However, this helped us to create an interactive and simple design.



# Appendix



Figure 1.

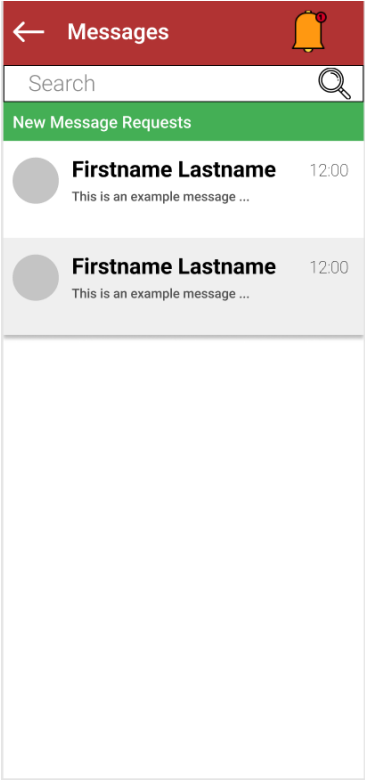


Figure 2.

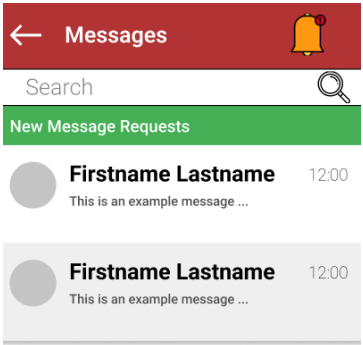


Figure 3.

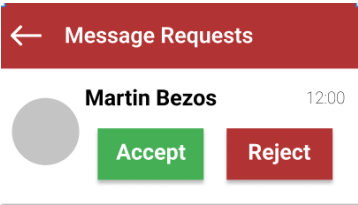
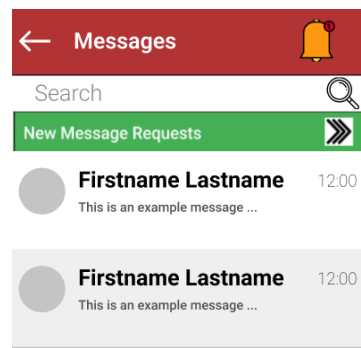


Figure 4.



**Figure 5.**