# FINAL UCD REPORT

CPSC 481 Fall 2020

Team P T04  Philmo Gu


| | |
|---|---|
| Eric Wu | 30055876 |
| Jack Yang | 30062393 |
| Sophia Ngo | 30063935 |
| Soumya Kumaria | 10124004 |
| Sydney Kwok | 30073206 |


Portfolio: https://sophiango-uofc.github.io/Team-P-CPSC-481/

Repository: https://github.com/sophiango-uofc/Team-P-CPSC-481

# EXECUTIVE SUMMARY

Our effort re-evaluates the idea of bringing people's clothes, accessories, and other personal belongings onto the social network, while addressing existing problems with outfit creation and planning services. The closet app holds an array of features that allow a multitude of diverse users to be comfortable in adding, creating, and sharing user outfits and clothing items to our system. Some tools we utilized include: competitive product survey, surveys and questionnaires, and scenarios methods from the IDEO method cards. These tools aided in our discovery of better UX conscious design choices that eventually lead to our prototypes. The lo-fi prototype, designed in Balsamiq, set the basic groundwork to plan the most important tasks needed in the design, while the hi-fi prototype, designed in Figma, gave us the appropriate preliminaries to conduct a heuristic evaluation to revise and iterate our final prototype. Every step we took ensured the creation of a well-produced and well-designed app.

# INTRODUCTION

This report summarizes the research and design processes that have gone into the development of the team project we have been working on this semester.  We will begin by describing the problem, our solution to it, as well as the expected users and stakeholders. Then we will review the method, process and findings from our user research in stage 2. Next, we will highlight important design choices we made, before going into the design of our Low-Fi and Hi-fi prototypes. We will then discuss the process and findings from the heuristic evaluation on our prototype, and the design changes made based on them. Finally, we will conclude the report by stating other changes that should or could have been made, and by forming conclusions about our research and design journey with this project.

## DESIGN PROBLEM

Everyone has struggled with the question "what should I wear today?" Some people spend too much of their precious time standing in front of the closet, trying to pick their outfit for the day. Additionally, people often forget about certain clothing items that have been placed in the deep, dark corners at the back of their closet. Also, when shopping for new clothes, one might wonder if the clothing item they are picking out will match with anything in their closet.

## DESIGN SOLUTION

Our solution is to help users save time with our virtual closet mobile application! By using the app to create outfits and save them to a calendar, the user can know exactly which clothes to pick out upon arriving at their closet. The app allows the user to upload clothing items, then search, sort, and categorize them, so they will always know what clothes they own. Also, by having one's entire closet at their fingertips, they can quickly check to see if a new clothing item will be a good addition to their closet. There is also a feed where users can share and follow others to gain inspiration for their next outfit.

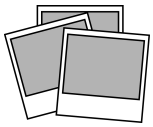
# END-USERS & STAKEHOLDERS

---

**Average Joe People | End-user**

- Average Joe, anyone who is looking for recommendations of clothing to wear based on the weather, or who does not know what to wear, can use this app to help them decide which clothes to select. They can also utilize it to see their friend's or social media influencers' wardrobe (that is, the items that they decide to share), and share their own wardrobe selection.

**Social Media Influencers | End-user**



- Social media influencer's followers can interact with them more by seeing what is in their closet. This will be focussed more towards influencers who typically share their outfits on social media. Followers can access their closet and see what outfits they have shared with their loyal followers.

**Clothing Retailers | Stakeholder and End-user**

- Clothing retailers could create profiles on the app and display their entire catalogue or new collections of items to users. Once the retailer has categorized their clothing items, regular users can browse those items or have them suggested to them by the app. Clothing retailers would also be stakeholders because if the app does well, it could increase their sales.

**Clothing Designers | Stakeholder and End-user**

- Similar to the clothing retailers, clothing designers can add their designs to their profile to showcase them to the public, or make potential clothing designs and or offers to clothing retailers to promote themselves.

**Personal stylists | Stakeholder and End-user**

- With this app, people can browse and look for style ideas without the help of a stylist. But, the app can also be used to showcase and promote stylists' work. People can browse the different outfits they come up with to understand how they style outfits before committing to their service. Personal stylists would also be stakeholders since if the app does well, they could promote their services to more people through our platform and potentially get more clients.

# USER RESEARCH METHODS

---

To help us conduct our user research, we enlisted the help of the IDEO method cards. We chose to use the Competitive Product Survey (Learn), Surveys and Questionnaires (Ask), and Scenarios (Try) methods.

## PROCESS & FINDINGS

Using the Competitive Product Survey method, we read a plethora of user reviews of existing systems that offer similar services to our own (see appendix A for competitive product survey findings). Through this, we were able to get an idea of the basic functional requirements and identify points of weakness that we could improve upon in our own design. We found that a major point of convenience for users was the ability to plan outfits ahead of time. For this reason, that became a major task that we wanted to make sure we designed well.

Using the Surveys and Questionnaires method, we made a Google Forms survey to help understand what features potential users would want (see appendix B for survey questions and final survey). After dispatching the survey and analyzing the results, we were given a better idea of which features to place our focus on. One of our concerns was regarding the involvement of social media influencers on our platform. We were surprised to see relatively

positive results, so this encouraged us to further consider the idea (see figure 1 in appendix B for survey question result).

Finally, we applied our findings from the previous methods to construct scenarios that would allow us to connect users with specific features we planned to implement (see appendix C for constructed scenarios). For example, a lot of our scenarios involved saving, creating, or viewing outfits. This indicated to us that the outfit feature would be important to a wide spectrum of users. These scenarios helped us obtain a deeper understanding of our users' motivations so that we had a better idea of how to prioritize features.

# DESIGN CHOICES

---

In the transition from the research phase to the design phase, important decisions about the design of our system had to be made. One of these choices regards the main navigation menu for the app. As a user, this would be important since this is a big part of usability and flow in the app. We looked at other apps to gain understanding of how a typical app might look like so that we could follow standards and provide a sense of consistency and familiarity to users of our app. We thought that seamless navigation was really important to our system, so we wanted the navigation buttons to be clearly visible, while still being relatively compact and tucked away. We were able to achieve this with the navigation bar.

Another important choice was choosing colours for the application. We tried to choose more neutral colours such that the app would appeal to users of any gender. First impressions affect the decision that one might make to use the app. So, it was important to find a colour scheme

that would be attractive yet not overlooked by someone because they feel it does not represent them well.

## LOW-FI DESIGN LESSONS LEARNED


When drawing sketches for the brainstorming portion, we should have drawn sketches collaboratively instead of drawing them individually. Due to individual sketching, we were not able to come up with multiple versions of the affinity diagram. We combined all of our ideas into one affinity diagram which was used to design the low fidelity prototype in Balsamiq (see appendix D for the affinity diagram). We learned that it is better to have multiple ideas for designing an application rather than just choosing one and sticking to it.

## HI-FI DESIGN LESSONS LEARNED

Since we were the creators and evaluators of our own app, it was really hard to critique our own app. It was hard to view things from the perspective of the user. As reviewers, we found it hard to interpret and rate the comments of the evaluators. There are multiple interpretations possible for the same thing. When creating our high fidelity prototype using Figma, it took a lot of time as there was a learning curve involved in working with the program. If we could have finished our prototype earlier, we might have resolved more usability related issues.

# HEURISTIC EVALUATION & FINDINGS

---

The purpose of the heuristic evaluation was to test our hi-fi prototype and take a closer look at how certain rules are included or are missing. We split up the team into three evaluators and two reviewers. The evaluators were in charge of looking through the prototype and following a template to record whether certain rules have been applied or violated. Then they gave reasoning of the importance of each rule (see appendix E for heuristic evaluations). This was done by trying to think from a user perspective and comparing other mobile apps to ours. After the evaluators finished filling up the template, the reviewers were to extract tasks and changes to implement in a future iteration. They also rated each task on a scale of how severe the problem is (see appendix F for reviewer's findings).

We found that there were clarification issues within our app, such as confusing icons, some buttons that appear to be clickable but are not, and some screens not being clear enough. For some rules, we found it harder to judge since it is our own prototype. This includes the efficiency of use and aesthetics, as we have a bias for our app. We also discovered that we limited the user's freedom by forgetting to implement some back buttons and deletion of items. Our app also did not include any help and documentation.

# CHANGES MADE

---

After reflecting on the results of our heuristic evaluation, we revisited our hi-fi prototype to fix the issues that were addressed in the evaluation report. Our first few issues had to do with the match between system and real world. Originally, almost all of the text on the app was in a light blue font, which can be difficult for users to read. We made changes so that filled text fields would be black, making them easier to read and indicating to the user that they have filled in the text field (see figure 8 in appendix G). Another issue had to do with the calendar. Originally we indicated that a date had an outfit saved to it by making the date in the calendar orange. We thought this may be too subtle for the user, so we changed it to an orange shirt icon on that date (see figure 9 in appendix G). This also matches the app and makes more sense.

Our next issue involved user freedom and control. We had found that in our previous iteration of the prototype, there was no way for users to delete or edit an outfit or item from their closet. We identified this as a severe issue, so we made sure to allow users to take these actions (see figure 10 in appendix G). We also added back buttons to some screens, allowing the user to escape from unwanted or accidental states.

We also had an issue with consistency and standards in our outfit creation screen. Previously, greyed out check marks were clickable. We thought this might create a lot of confusion for the user, so we changed the checkmarks to black, indicating to the user that they can click on them. Another point of confusion was that plus icons were left on clothing items, even after they were selected. Since a user is not able to add multiple tops, bottoms or shoes to an outfit, we removed the plus icons after one was added (see figure 11 in appendix G).

Next, we tried to remedy issues with recognition rather than recall. This included adding text like “choose an outfit” or messages like “outfit added” to certain screens to indicate the purpose of the screen or the actions that the user has just completed (see figure 12 in appendix G).

Our final issue was with help and documentation. To provide documentation and help new users get started with the app, we added a help icon to the profile page. This opens up a popup that provides the user with guidelines on what to do in the app and how to get started (see figure 13 in appendix G).

# FINAL & FUTURE CHANGES

---

Moving into stage five, we added a settings screen and made changes to allow users to comment on posts and create their own posts. We also allow users to save more than one outfit to one calendar date.

In the future, changes should be made to smooth out any consistency issues. These issues would be found after rigorous use of the app, and are relatively easy to fix. We could also further improve the design of the app to make it more visually appealing and exciting to use. This could mean adding more design elements such as shapes and colours, or making smooth transitions between screens. Also, when adding an item, the user is given a choice to associate the item to a weather condition. The app currently does not do much with the weather condition. In the future we should add a way for the user to sort by that parameter.

Additional changes could also be made to add more features to interact with everything in the app, such as clicking on an image in the feed will show the attached outfit if there is any. This can be extended to sharing a clothing item and allowing the user to click on it to see any notes and how the user categorized it. We can also add the feature to allow users to check multiple categories at once, rather than one at a time. The categorization could also be made more intuitive by changing how the menu works and possibly adding a search to account for users that will add many more categories.

In conclusion, creating the app made for a successful experience. We learned a lot about the process of design from start to finish. This includes learning about the design thinking process and using prototypes to bring our idea to life. Through the development of the project, we also learned how to work in order to meet strict deadlines. Overall, this project was a great way for us to implement concepts we learned about in class and gain hands on experience with user centered design methodologies. It has also given us a greater appreciation for all of the work and thought that goes into the design of systems that we as users too often take for granted.

# APPENDIX A: COMPETITIVE SURVEY FINDINGS

**Apps Reviewed:**

- My Dressing - Fashion Closet App
- Stylebook App
- ClosetSpace
- Stylicious

**General Findings**:

- Seems pretty standard to offer users multiple ways to upload clothing items. Users want to be able to take a photo of the clothing piece and edit out the background, but uploading and editing their own photos can be time consuming and have poor results if the lighting is not good. For this reason, they also want to be able to upload images saved on their device, upload images by web URL, and to be able to search for items from specific manufacturers/stores and add them in that way.
- When uploading items, users want to be able to add details about a clothing piece such as its cost, colour, material, season and brand so that they can later search their closet for items by these parameters.
- Once a user has made an outfit, they would like to be able to save the outfit, share it with other users, assign it a calendar date for them to wear it on, and to code the outfit according to the event.

# APPENDIX B: SURVEY QUESTIONS

- On a scale from 1-10, how important to you are the clothes you wear?
- Why don't/do users care?
- How often do you have issues choosing what to wear?
- How long does it take for users to decide on what to wear
- Do you use an application to keep track of all the items in your closet?
- If yes, provide the name of the app/service you use.
- Do you plan your outfits?
- How often do you have difficulty buying clothes?
- What's your general opinion on social media "influencers"? Do you like/dislike them? Would you choose to use or not use a system because of the prevalence of influencers on the platform?
- Do you follow someone because you like their fashion?
- Do you ever wish you can share an outfit?
- Do you ever wish you can find out what someone on the internet is wearing?

Survey link: https://forms.gle/UbVoiAhMzXzHYNWKA

Edit survey / view results: https://docs.google.com/forms/d/1pY5zzfmaWYEC8yRvKsf1FpyYEZvpZEi2drW7L6kTxrM/edit?usp=sharing

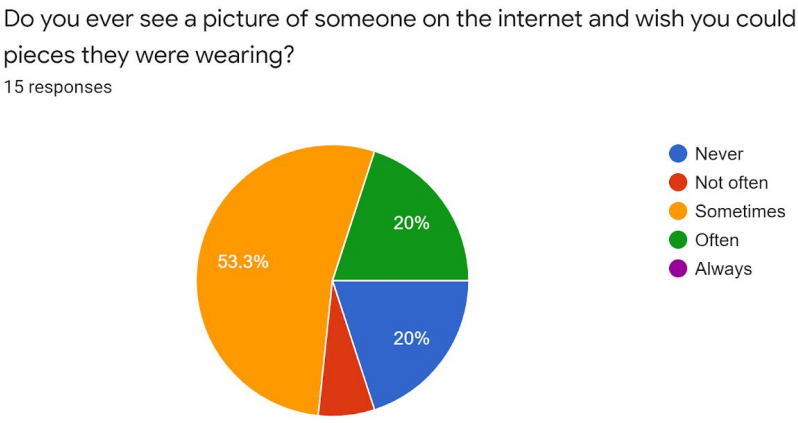



Figure 1: Result of survey question regarding social media

# APPENDIX C: SCENARIOS

**Scenario 1: User wants to create an outfit based on a specific occasion**
The user wants to find a suitable outfit for a research conference. The user will open our application and look for the list of previously created outfits and choose the category of professional outfits. The user will either pick a previously created outfit or they can mix and match items from previously created outfits to create a new outfit.

**Scenario 2: User wants to view the current/previous outfits of a celebrity/influencer they like/follow**
The user is on social media and going through their feed, and sees a celebrity they follow have an outfit that appeals to their style. From there the user wants to go through and look at all of that celebrity's previous outfits that they had shared publicly. The user opens up our application and searches up or visits the profile of the celebrity. Then the users can look at the previous outfits of the celebrity and a photo of them in it (if made public), and can see more details on the outfit upon pressing it.

**Scenario 3: User sees an online photo of a celebrity wearing an outfit that they really like and wants to find out where they got the clothing items from**
The user is scrolling on social media and sees a photo of a celebrity wearing an outfit that they really like. The user wants to find out where these articles of clothing are from so they go on our app and visit the profile of the celebrity. They can then look through the celebrity's outfits on their profile and after selecting the outfit, they can view a breakdown of each piece in the outfit and details about them (e.g., brand, cost).

**Scenario 4: User wants to dress for unexpected weather**
The user wakes up one morning and opens their blinds to find out that winter has come in overnight. Seeing as how it's been almost a year since last winter, the user has forgotten how to dress for the winter. Luckily, they can open our application and view the outfits they've labelled as "winter" outfits. Now they can recall what kind of clothing they wore in this weather last year and can get dressed in no time.

# APPENDIX D: AFFINITY DIAGRAM

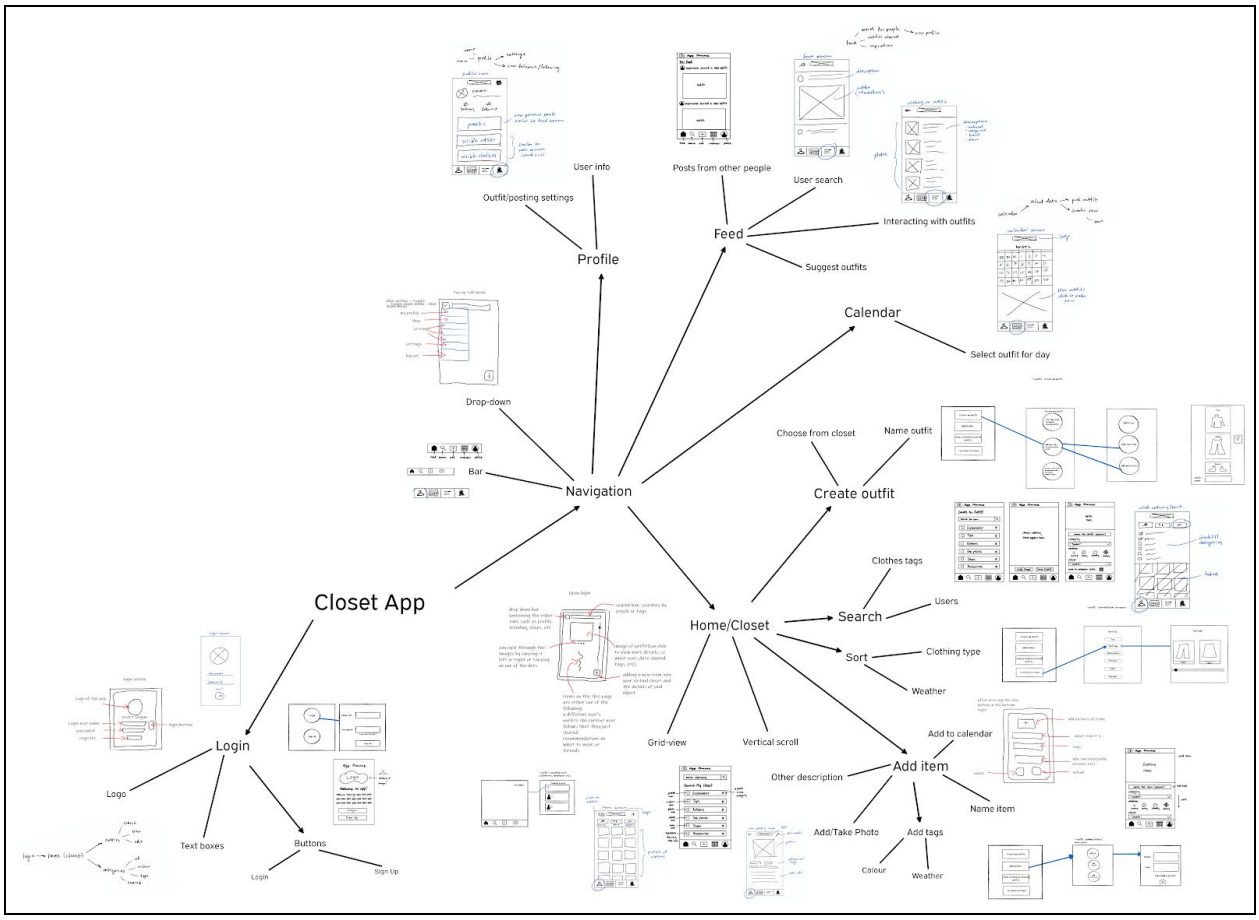



Figure 2: Affinity diagram for organising ideas presented in sketches

# APPENDIX E: HEURISTIC EVALUATIONS

**Heuristic Evaluation, Evaluated by Eric Wu**

| Rule of thumb | Is the rule applied? | Is this rule violated? | How can this rule improve usability? |
|---|---|---|---|
| **Visibility of system status** | This rule is implemented when the user switches to a different screen/page on the app, in which a transition will occur to load the upcoming page. | No, since all the loading/animations happen instantaneously (less than 1 second). | It helps users understand what will occur next, and to show that the application has not crashed by showing a animation. |
| **Match between user and the real world** | When the user wants to check their outfits, in which they will click on the clothes hanger, calendar icon, or click the bottom right icon to view their profile. | No, as all the icons are informative on what will occur once the user clicks on it, such as the newspaper icon will result in the social media feed popping up. | It improves the visual appeal of the app and conveys information visually without confusing the user. |
| **User control and freedom** | When the user wants to delete an outfit from the calendar, or add a whole new outfit to the app. | This rule is violated when the user is not able to edit their existing outfits with new items, and are not able to delete existing outfits. | Helps users in having more freedom with their interactions of the app, allowing for a more enjoyable experience. |
| **Consistency and standards** | When the user has to choose between the buttons, add a new outfit, or an existing outfit to a certain calendar date. And the back button being a left arrow and confirmations are check mark icons. | No, since all the visual elements are consistent. | Helps users have a clearer understanding on how the app works, and decreases the amount of time they need to spend to learn it. |
| **Error prevention** | This rule is applied when the user wants to delete an outfit from a certain date, asking whether they want to delete it or not. | This rule is violated when the user wants to add an outfit or add an existing outfit to a calendar date, where it doesn't ask for confirmation. | Makes sure that the user does not accidentally do something they didn't want to occur. |
| **Recognition rather than recall** | This rule is applied through the usage of icons, and brief/short text on certain pages such as the add outfit screen. | This rule is violated, as when the user goes and presses on the pick existing outfit button, no title is displayed telling a user to select an outfit. | Reduces the amount of information the user will have to remember, and makes sure the user will know what to do. |
| **Flexibility and efficiency of use** | This rule is applied through the usage of 2 add outfit buttons. (one in the existing outfits saved on the app page, and another on the calendar page after clicking on a empty date) | No, as there are not other features to implement multiple buttons for. | Decreases the learning curve for the functionalities of the app, and decreases the time required for certain actions. |
| **Aesthetic and minimalist design** | This rule is applied with the minimalist navigation bar at the bottom of the app, and the overall design is aimed towards it. | No, as the entire app's designs are minimalist, users will not be overwhelmed. | Decreases the confusion the users will have when navigating the app, by preventing irrelevant information from being shown. |
| **Help users recognize, diagnose and recover from errors** | This rule is not applied, as no error messages are shown to the user when the system detects it. (Such as trying to add a new outfit without any items, it does not display any kind of message). | This rule is violated, as when the system detects an error, it does not display what has gone wrong to the user. | Helps users identify, and correct the issue that is causing the error. Thus avoiding confusion on why a particular part does not seem to work. |
| **Help and documentation** | This rule is applied through the usage of labels and visuals (such as pictures), to guide the user, but there is no official documentation. | This rule is violated, as there is no official documentation for the user to refer to for help on what a particular part does or is asking for. | Gives the user the opportunity to find help, by easily searching up information regarding the section that they do not understand. |

Figure 3: Eric Wu's Heuristic Evaluation

Heuristic Evaluation of Closet App

| Rule of Thumb | Is this rule being applied? How so? | Is this rule violated? How so? | How can this rule further improve usability, utility and desirability? |
|---|---|---|---|
| 1. Visibility of system status | Transitions are made when interacting with certain buttons and buttons are highlighted when selected | No, there is no loading or indication of any status since it is all instant | When system is loading show some animation to show it is working |
| 2. Match between system and the real world | Icons in the app match what it's supposed to do, such as calendar, news feed, and profile | Icon can be vague, like the show outfits screen | Makes it more clear to the user what they are looking at |
| 3. User control and freedom | Allowed to delete outfits on the calendar, can add their own categories | Cannot delete or edit an item or outfit once added and sometimes cannot return to previous a page | Gives user more freedom to customize and explore the app seamlessly |
| 4. Consistency and standards | All back buttons are shown as left arrow icon, confirms are checkmark icons | Checkmark being greyed out even though user can click on it, or vise versa, users brought to profile screen | Making it similar to other apps will allow a smaller learning curve when using it for the first time |
| 5. Error prevention | Does not allow item to be added when certain parts are not filled | No, all actions are in a linear flow that forces the user through a certain path | Allow users to confirm certain actions before committing |
| 6. Recognition rather than recall | Most are represented by icons, or explained in short text | No, all icons are simple and most things have a title or some text explanation | Makes sure the user knows what they are clicking before even clicking on it |
| 7. Flexibility and efficiency of use | Can select previously created outfits to put on the calendar | No, not many shortcuts needed since it is mostly viewing and straight forward buttons/text boxes | Speeds up the process of creating an outfit |
| 8. Aesthetic and minimalist design | Simple navigation bar, few things shown on the screen at a time | No, app looks simple and contains the necessary information on each page | Allows for a clean and simple look for the app which appeals to users |
| 9. Help users recognize, diagnose and recover from errors | No, no errors are shown when completing any task | No indication when adding item or outfit parts of which parts are required | Lets the user know what they need to do to accomplish the task |
| 10. Help and documentation | Some examples and suggestions shown in search bars | Not enough examples or suggestions are shown | Helps revolve difficulties one might have when using the app |

Evaluation by Sophia Ngo

Figure 4: Sophia Ngo's Heuristic Evaluation

Evaluation of: Team P Closet App Hi-Fi Prototype
Evaluator: Sydney Kwok

## 10 Steps to Improve Usability, Utility, and Desirability by Implementing Nielsen and Molich's UI Design Guidelines

- Choose the website or app for which you want to critique and improve usability, utility and desirability.
- Then work through the list to see whether or not the website or app follows Nielsen and Molich's 10 rules of thumb.
- Finally, improve the website or app by further applying the 10 guidelines.

| Rule of Thumb | Is this rule being applied? How so? | Is this rule violated? How so? | How can this rule further improve usability, utility and desirability? |
|---|---|---|---|
| 1. Visibility of system status | The rule is not being applied other than through "dissolve" or "move in" sort of transitions between screens. | It's not applied it but I don't think it's been violated either. This lets the user know that that their request is being processed when there is latency in completing the request, but since it's just a prototype with no backend, it's not all that necessary. | Visibility of system status/loading during login/register and view closet/outfits can help keep the user informed about what's going on while their requests are being processed by the system. |
| 2. Match between system and the real world | Yes. The system uses standard icons and familiar language to make interaction as natural and logical for the user as possible. | Maybe with the icon for viewing outfits? It isn't all that intuitive, but it's difficult to find an icon that can be that small and still communicate "click me to view your outfits" | Following conventions minimizes ambiguity for users so that interaction with the system meets their expectations and is as seamless as possible |
| 3. User control and freedom | Yes. The system allows users to delete outfits that were saved to a calendar date. This gives users the freedom to undo their past actions. | Perhaps in other aspects. It would be nice for users to be able to delete items and outfits from their closet. | This rule gives the user the freedom to leave unwanted states without having to go through a lengthy or troublesome process. |
| 4. Consistency and standards | Yes. Different functionalities are pretty well differentiated and the system is pretty consistent in having similar things look and act similarly. The UI is not only consistent with itself but also with similar types of apps. | No, the rule is not violated. The system is consistent and for the most part follows universal standards. | This rule makes it easier for the user to familiarize with the system. It also allows for the user to apply their experience with other systems they've used in the past, effectively flattening the learning curve faced when just getting started with this system. |
| 5. Error prevention | Yes. When taking a picture to upload a clothing item or when deleting an outfit from a calendar date, the user is asked to confirm before commiting to the action. | No. The system uses alerts to notify the user when it thinks they may be making an error. | This rule prevents errors in the interaction from occuring. |
| 6. Recognition rather than recall | Yes. Guiding instructions for many tasks are discreetly communicated through icons or labels in the UI. | Although not applied, I don't think it was violated because tasks are pretty simple and intuitive in this system, so lengthy instruction is unnecessary. Perhaps we could add small "click me to do this" prompts or tutorial-like notes though? | The UX is easier and more enjoyable for the user if instructions for use of the system are easily visible/retrievable. |
| 7. Flexibility and efficiency of use | No, the system does not really provide more than one way for users to complete a given task. | Not applied but not quite violated either. There's only so many ways to complete the tasks for this system and we tried to keep things as simple as possible, so I don't know that there's a way to make things much more efficient. | This allows for more than one way for users to complete a task so users can complete tasks more efficiently. |
| 8. Aesthetic and minimalist design | Yes. The system design is quite minimalist and does not contain irrelevant info, so as not to clutter the user's view and to maximize the limited interface space of a small mobile device. | No. The app design is very minimalistic and avoids overwhelming the user. | An aesthetic and minimalist design is not only nice to look at but also prevents from distracting or overwhelming the user with unnecessary or irrelevant information. |
| 9. Help users recognize, diagnose and recover from errors | No, the system does not communicate to the user when they have made an error. | Yes. The system provides no aid in recognizing, diagnosing or recovering from errors | Clear error messages in plain language help users identify problems and make it easy for the user to recover from them. |
| 10. Help and documentation | There are labels that intend to help or guide the user, but there is no documentation for the user to refer to. | There is no way for the user to search for help or documentation on how to complete tasks on the system. However as mentioned before, not sure that this system requires too much documentation. | This rule makes it very easy for the user to be able to find the help they need when they run into issues while completing tasks on the system. |

Figure 5: Sydney Kwok's Heuristic Evaluation

# APPENDIX F: REVIEWER'S FINDINGS

You are required to classify the prototype problems severity as it was discussed in the heuristic evaluation lecture. Evaluators should meet together to share their findings and cross check the identified problems and make decisions about the problems that need to be fixed.

The severity of a usability problem is a combination of these factors:

- The frequency with which a problem occurs. Is it common or rare?
- The impact of the problem if it occurs. Will it be easy or difficult for the users to overcome?
- The persistence of the problem. Is it a one-time problem that users can overcome once they know about it or will users repeatedly be bothered by the problem?

| Rate | 0 | 1 | 2 | 3 | 4 |
|---|---|---|---|---|---|
| | Doesn't seem to be a usability problem; not a problem | Cosmetic problem; need not be fixed unless extra time is available on project | Minor usability problem; fixing this should be given low priority | Major usability problem; important to fix, should be given high priority | Usability catastrophe; must fix |
| 1. Visibility of system status | | E | So, Sy | | |
| 2. Match between system and the real world | E | So, Sy | | | |
| 3. User control and freedom | | | Sy | So, E | |
| 4. Consistency and standards | So, Sy, E | | | | |
| 5. Error prevention | So, Sy | | E | | |
| 6. Recognition rather than recall | So | Sy | E | | |
| 7. Flexibility and efficiency of use | So, E | Sy | | | |
| 8. Aesthetic and minimalist design | So, Sy, E | | | | |
| 9. Help users recognize, diagnose and recover from errors | | | | | So, Sy, E |
| 10. Help and documentation | | | So | | Sy, E |

Figure 6: Jack Yang's Evaluation Findings

Heuristic Review by : Soumya Kumaria

| Rule of Thumb | Findings | Ratings | | |
|---|---|---|---|---|
| | | Sydney | Eric | Sophia |
| 1) Visibility of System Status | The transition from one window to another is smooth (less than 1 second). Adding a loading screen between transitions may help. | 1 | 1 | 1 |
| 2) Match between system and the real world | There should be some changes made to view outfit screen. The icons seem to be confusing/vague. | 1 | 0 | 1 |
| 3) User control and freedom | Need to add the option for deleting items and outfits. Outfits and items can't be deleted from the closet presently. | 2 | 3 | 3 |
| 4) Consistency and standards | Some of the checkmarks are greyed out, even though they can be clicked. | 0 | 0 | 1 |
| 5) Error prevention | A confirmation message needs to be added to the calendar date – add outfit screen. | 0 | 1 | 0 |
| 6) Recognition rather than recall | A title needs to be added for choosing a previously created outfit screen. | 1 | 2 | 0 |
| 7) Flexibility and efficiency of use | No changes to be made. | 0 | 0 | 0 |
| 8) Aesthetic and minimalist design | No changes to be made. | 0 | 0 | 0 |
| 9) Help users recognize, diagnose and recover from errors | There are no error messages being displayed when an error is caused by a user. Greying out of options and check marks aren't good indicators of errors. | 4 | 4 | 4 |
| 10) Help and documentation | Some documentation to explain how to use the features of this app are required. Some troubleshooting instructions should be available. | 2 | 3 | 3 |

The Ratings are based on the following table (taken from Heuristic Evaluation Lecture) –

| Rating | Description |
|---|---|
| 0 | Doesn't seem to be a usability problem |
| 1 | Cosmetic problem, some visuals need to be fixed |
| 2 | Minor usability problem |
| 3 | Major usability problem, important to fix |
| 4 | Usability Catastrophe, must fix |

Figure 7: Soumya Kumaria's Evaluation Findings

# APPENDIX G: CHANGES MADE EXAMPLES

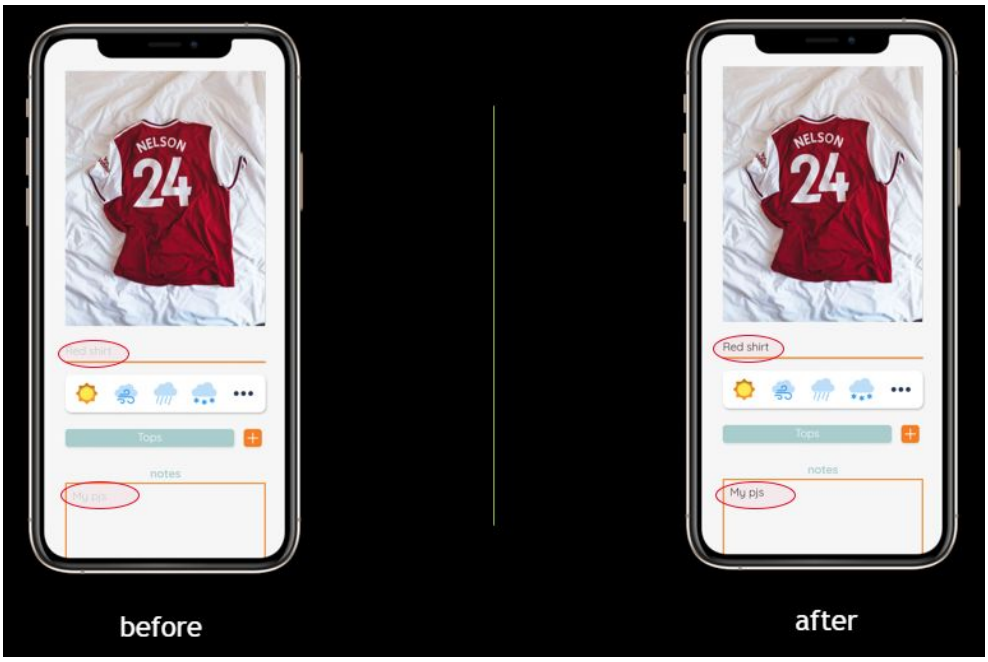



Figure 8: Changing text colour

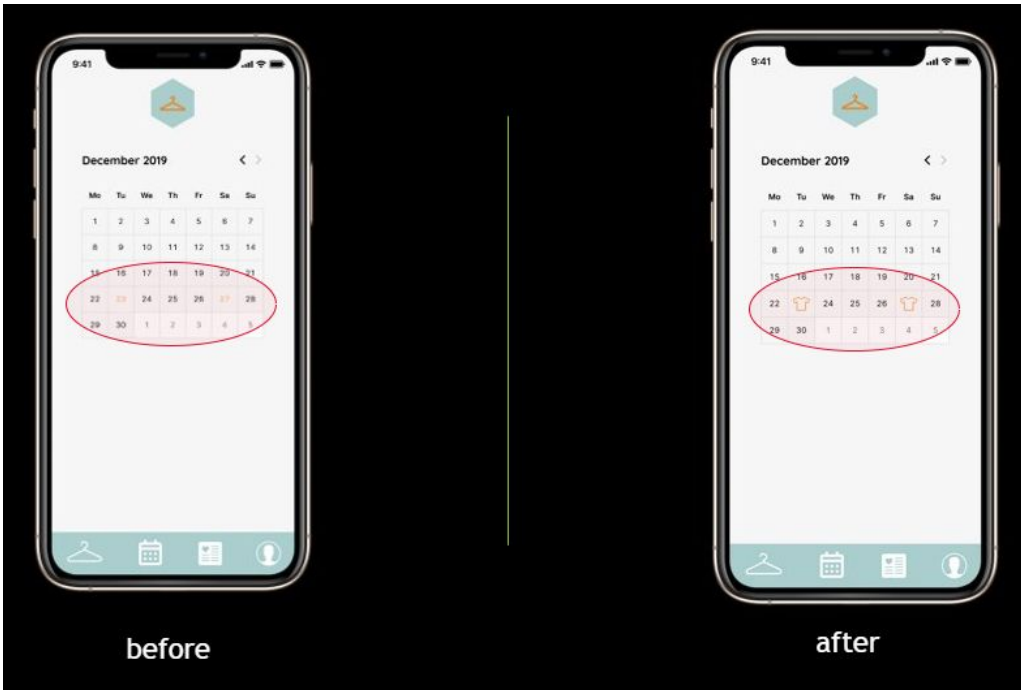



Figure 9: Changing orange text to orange shirt icons

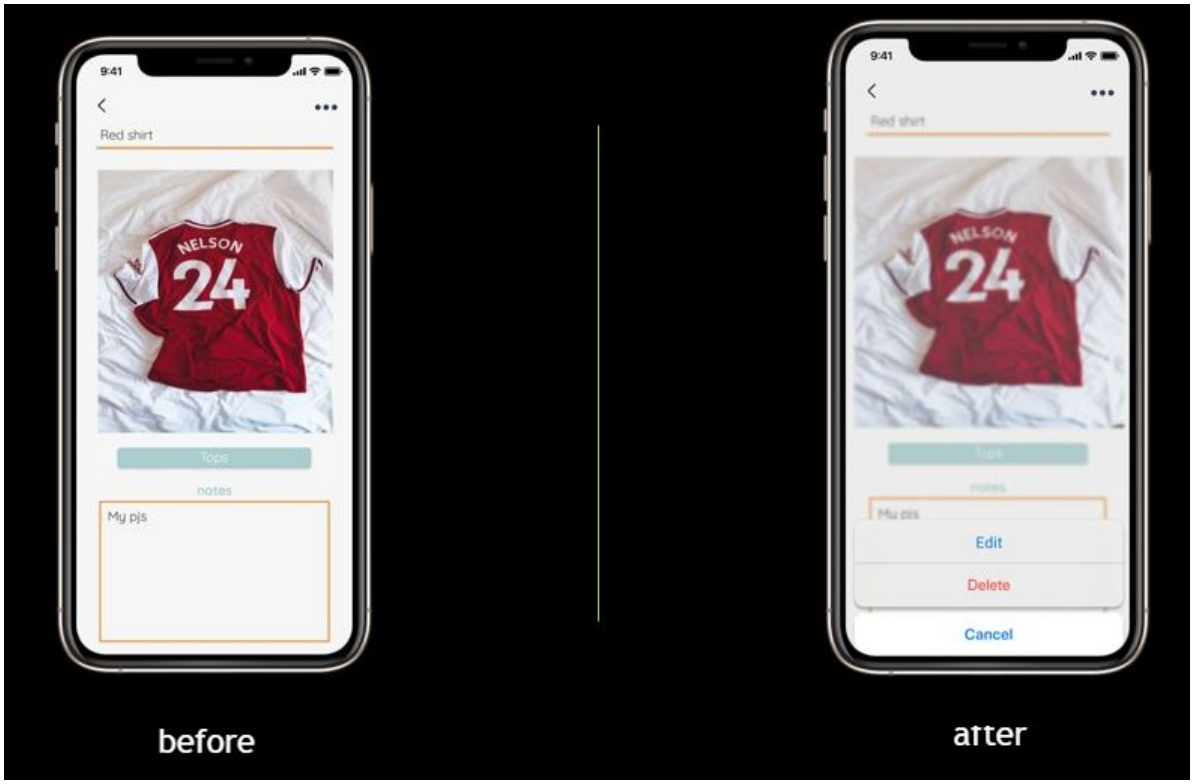



Figure 10: Option to delete

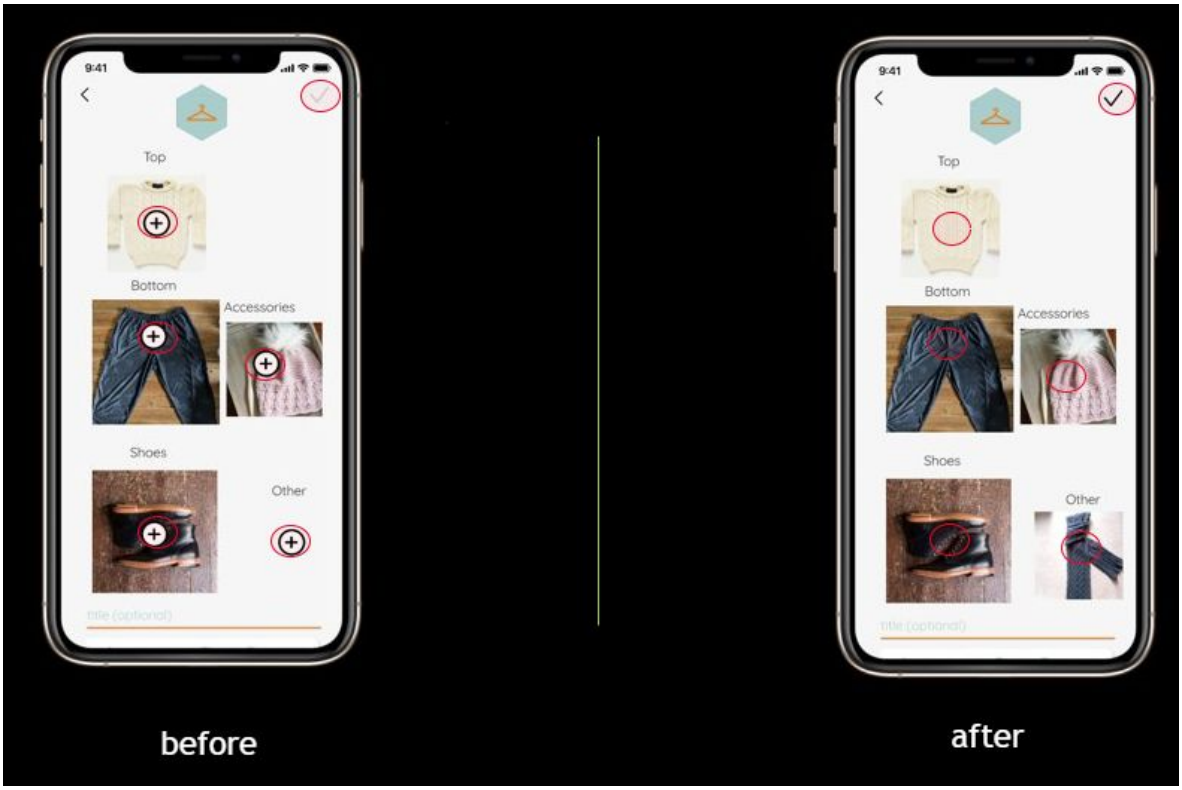



Figure 11:  Greyed out checkmark and plus icons

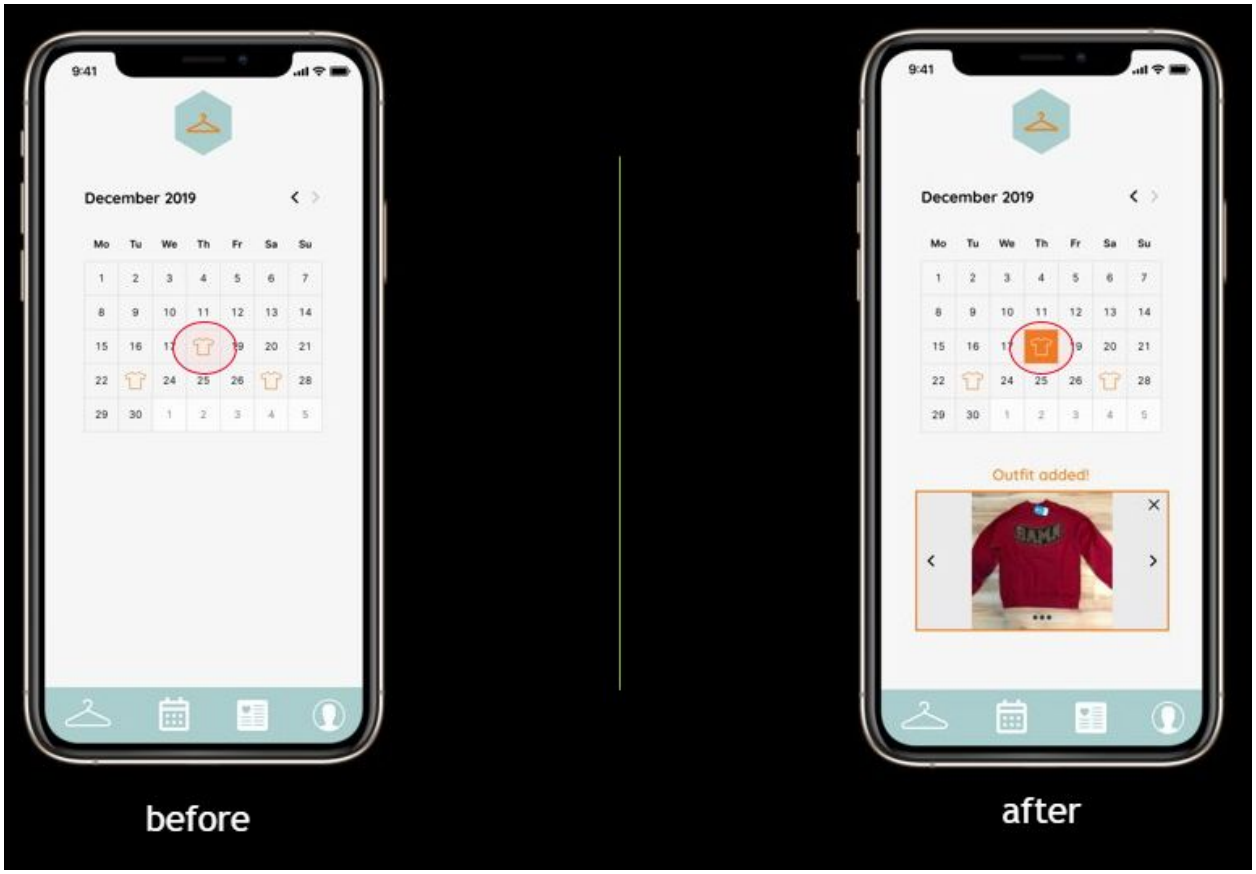



Figure 12: Added “outfit added!” text

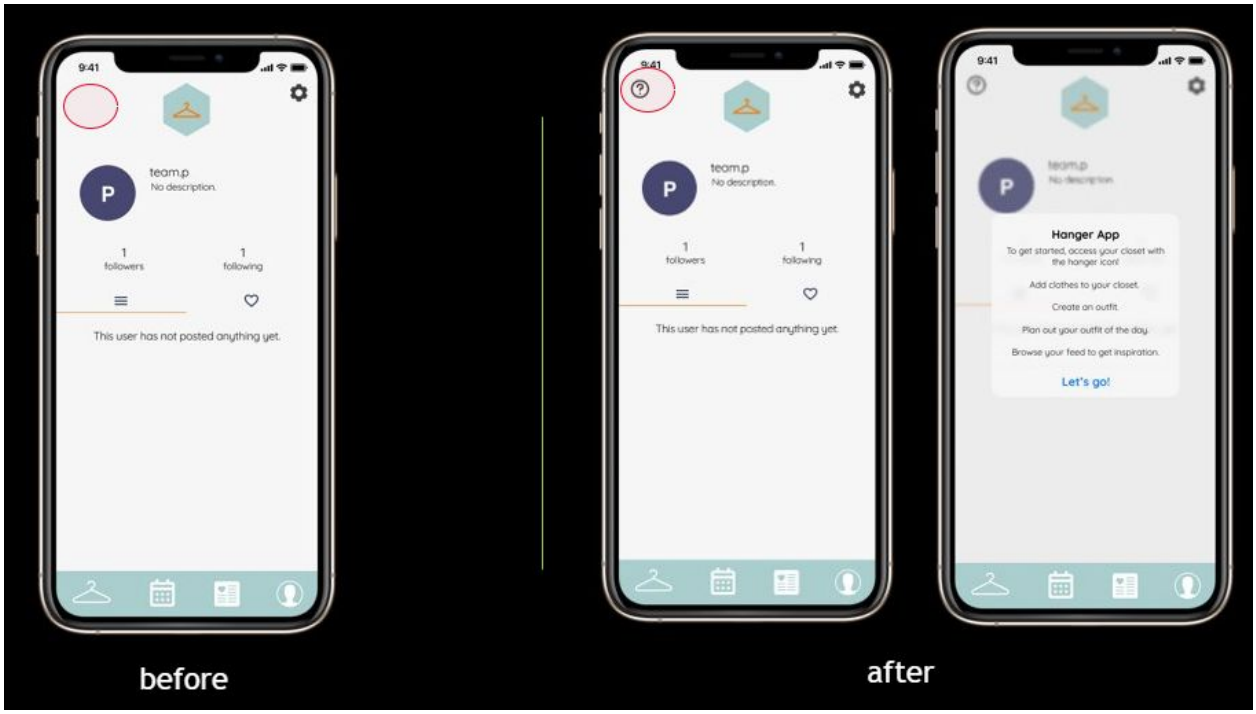



Figure 13: Addition of help and documentation