

# OOP Project Report – Group 41

Aleksandra Orlinova Savova, David Dobin, Ionut Liviu Moanta,  
Junwon Yoon, Luis Cabo Villagomez, Marina Madaras

## 1 INTRODUCTION

The overall goal of the Heuristic Usability Evaluation is to get a different perspective on our current application user interface. We are taking into account present design problems that we might have overlooked as a team, having gotten feedback from a diverse group of evaluators. Our goal is to improve our application design to ensure that our product is user-friendly and accessible.

For our prototype, we have prepared a document containing screenshots of the current state of our user interface. Moreover, we added indications explaining the possible interactions with the interface in order to help the other team evaluate in a more organized manner and provide them with a sense of orientation.

The evaluators noted down parts of our front-end overview that were counter-intuitive or inefficient, according to the heuristics we have learned during lectures. In this report, we will discuss what can be taken away from the experts' observations to further improve the quality of our application's ease of use and, in consequence, end up with a more polished end product.

## 2 METHODS

In this section of the report, we will provide an overview of the method for our heuristic usability evaluation. There will be five sections - how we instructed our experts, what prototype we provided them, the evaluation procedure guidelines, factual steps taken in it, and what heuristics the experts referred to.

### 2.1 Experts

For the evaluation, we have recruited six experts who are peers from our course. As a Computer Science student, each expert has a strong understanding of usability principles and based on their knowledge and experience in the field of software development and user interface design can offer a unique perspective and objective assessment.

Furthermore, as fellow students, the recruited evaluators are well aware of the overall functionality of the application. For this reason, it should be noted that their knowledge and, as a result, user experience might differ from the one we expect from the average app user. Hence, it is expected that their preconception of what the application should feature might prevent them from fully discovering all interface problematic points that a new user might encounter. Nonetheless, based on the expertise of the evaluators, we also consider fixing the discovered problems in the prototype of greatest priority, as they might cause difficulty even to advanced users.

The prototype of the interface shown to the evaluators contained all of the basic requirements for the application, with the only exception being the "Connect to server" scene.

A few examples of slides from our prototype are given above. Only a few slides are included, but the other features were presented

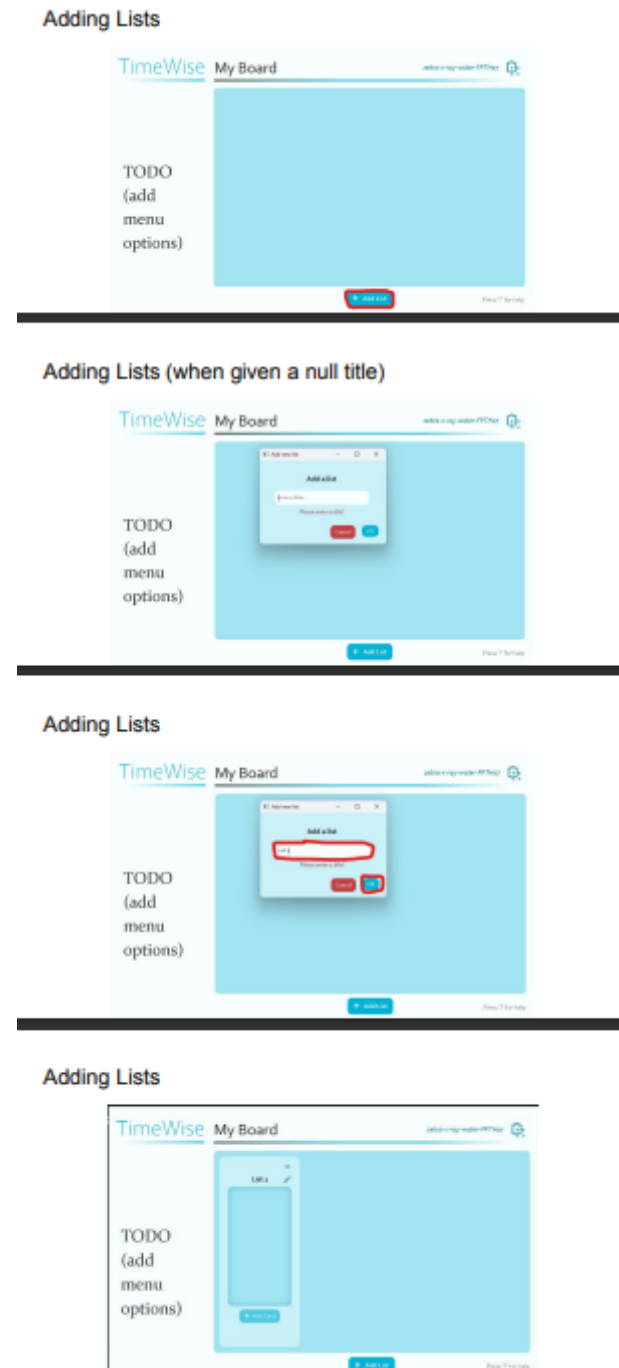


Figure 1: Prototype PDF, 'Adding Lists' Section

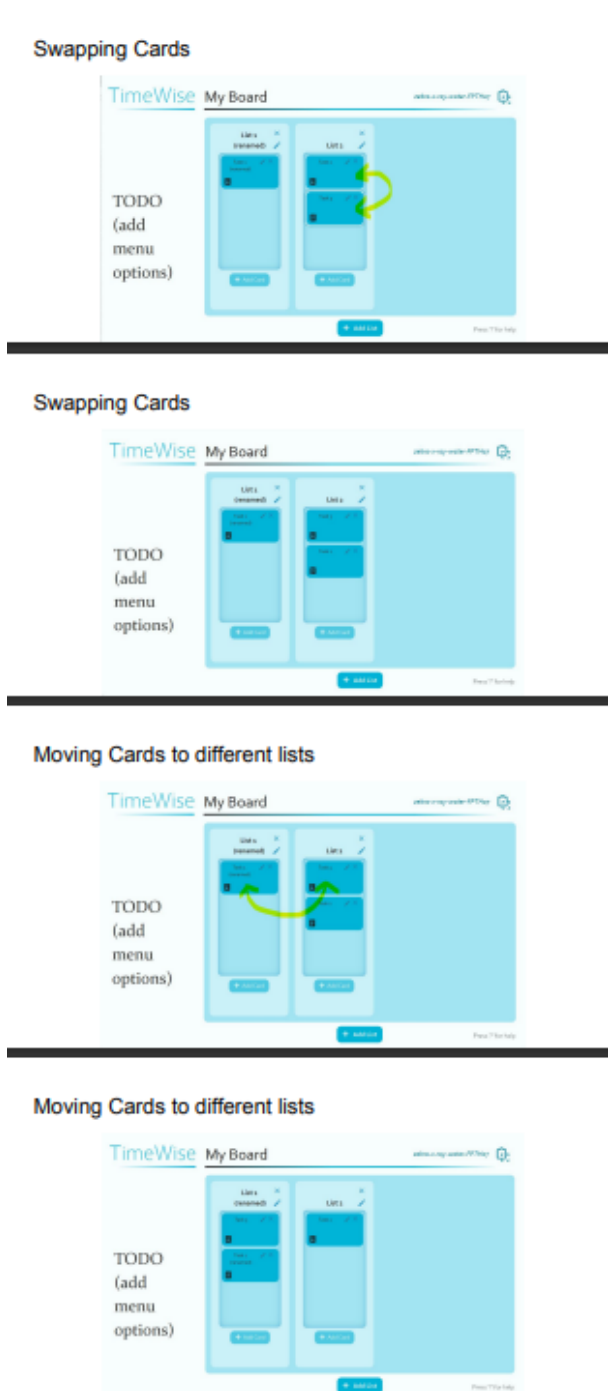


Figure 2: Prototype PDF, 'Swapping Cards' Section

similarly. We also ensured that the features are presented in order (e.g. adding lists are presented before deleting lists), to minimize confusion.

## 2.2 Procedure

**2.2.1 Instructing the Experts.** We have provided the expert group with a prototype of our application, and 10 usability heuristics[1] for user interface design. They could refer to the heuristics given, keeping their evaluations consistent.

**2.2.2 Prototype Overview.** The experts were shown a prototype of our application, in the form of a pdf file with screenshots. The screenshots show the details of all the fully-functional features implemented in our application. The screenshots were accompanied by additional explanations and visual indications to define which feature is being presented so that it resembles the transitions that are happening in the actual application.

**2.2.3 Experts' Procedure.** The experts would:

- First look at the prototype to understand the logic behind the application.
- Examine the behaviour of the functionalities in typical usage scenarios.
- Individually review the features of the application by going through the prototype a third time, now more in detail, and contemplate on what problems might arise.
- Take notes on all observations and evaluate the application based on the usability heuristics we have learnt about in lectures.
- Each expert would proceed to record their own observations and feedback in the survey we have provided.

**2.2.4 Factual steps taken in the Procedure.** One of the team members, Marina, was the observer responsible for supervising the experts. She ensured that:

- all experts would analyze the prototype individually;
- all experts would go through the application prototype multiple times to find usability problems;
- during the evaluation, any given hints were also taken into account, as to factor out biased input when conducting the review.

**2.2.5 Heuristics used by the Experts.** The experts were provided with 10 usability heuristics.

The 10 heuristics are:

1. Visibility of system status
2. Match between system and the real world
3. User control and freedom
4. Consistency and standards
5. Error prevention
6. Recognition rather than recall
7. Flexibility and efficiency of use
8. Aesthetic and minimalist design
9. Help users recognize, diagnose, and recover from errors
10. Help and documentation

## 2.3 Measures

We researched usability problems that future users might encounter, through studying the root causes for them and their specific contexts. We collected the data we will be analyzing in this report through an online, anonymous, survey. Our evaluators were prompted to fill in a form created by us for each usability problem they have found. The fields they were asked to complete were:

- Problem description;
- Likely/ actual difficulties;
- Specific context;
- Assumed causes;
- Choose the heuristic which it concerns;
- Other remarks.

The cumulative answers were gathered, by the survey in a table. This way, we could easily see overlapping usability problems but also encompass the current accessibility level of our application.

## 3 RESULTS

### 3.1 Results

After having our program design analyzed by a fellow team of evaluators, we have come to several main conclusions, which will be used as improvement guidelines for the upcoming design of our application.

1. The lists inside of a board have a limited display capacity of only three cards at a time, requiring the user to scroll down to view additional information. The "add card" button is positioned too high, and the cards themselves appear longer than necessary, potentially leading to wasted screen space and inefficient use of the available display area. This may cause inconvenience and frustration for users who frequently work with larger lists.

Heuristic: Flexibility and efficiency of use

2. The app does not provide clear and concise instructions to users to help them navigate the app and avoid confusion and frustration. It was noted by evaluators that not all functionality is documented properly, which implies that the user might remain completely unaware of existing features and limit their experience with the app. In the case specified, dragging cards is an important feature for the user to know, therefore it should be included in the help screen or other instructional materials so the user does not miss out on it.

Heuristic: Help and documentation

3. The current design lacks a feature that allows users to cancel the creation of a new card, which could lead to cluttered overviews and unnecessary information. The user may also feel constrained if they start creating a new card but change their mind and are unable to undo the creation.

Heuristic: User control and freedom

4. The application lacks a proper description, leaving users unsure about its purpose and how best to utilize it. The absence of a clear overview or explanation may lead to confusion or reduced user interaction, as users may not understand what the application is meant to do or how it can benefit them.

Heuristic: Help and documentation

5. The current design of the application features small text fields and error messages that can easily remain unnoticed by users, particularly those who are new to the application or have less digital experience. For example, not being aware of the requirement to add a title when creating a new card may cause frustration when users attempt to perform the action and fail to notice the current

error prevention mechanisms.

Heuristic: Help users recognize, diagnose, and recover from errors

6. The text field for adding cards to a list is small and difficult to read, which can be a problem for users with poor eyesight. Thus, any mistakes made while typing the title are not immediately apparent, requiring the user to rename the card.

Heuristic: Flexibility and efficiency of use

7. The current design of the application lacks a safe mechanism for deleting cards or lists. This means that if a user accidentally presses the delete button instead of the rename button, for example, no confirmation or warning message appears to ensure that the user wants to delete the card or list. As a result, users may unintentionally delete important information, causing frustration and potentially even loss of data.

Heuristic: Help users recognize, diagnose, and recover from errors

8. The edit and delete buttons on cards and lists are in very close proximity to each other in the current design of the application. Users may accidentally delete cards or lists when they intend to edit them. Since there is no confirmation screen to ensure that they want to permanently delete the content, this can result in the unintended loss of important information without a way to recover it.

Heuristic: Consistency and standards

### 3.2 Adjustments

To settle on what adjustments to make to our application, we analyzed the underlying causes the evaluators provided. By looking at the cause-effect relationship of our usability problems, as a team we brain-stormed and we were able to pinpoint how to materialize ideas for improvements.

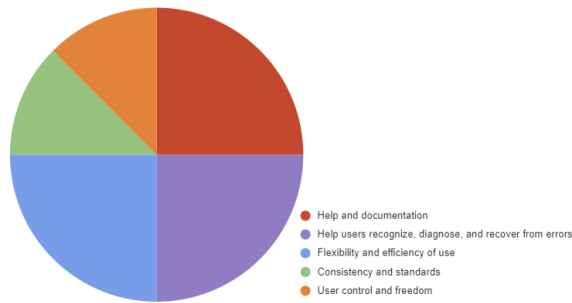
1. Trying to solve the problem of user confusion and frustration when using our application, we have decided to make some changes in the help screen. Now the help screen will not only contain the information about the keyboard shortcuts but also the fact that the user can double-click to view the card details.

2. To minimize the screen space wasted, we can make each card view smaller and place the 'Add Card' button lower. We hope this change will make the application more convenient, especially for users who add many cards to one list.

3. To provide a concise, organized view of the interface, we have decided to add an 'X' button on the interface, next to the 'OK' button. Now the users can cancel the addition of cards to lists. This option can help users maintain a cleaner and more organized overview and avoid potential data redundancy and unnecessary deletion of cards that were not intended to be created in the first place.

4. Provide a short description on the start screen of the app and tooltips when on mouse hover on buttons can help users better understand the value and functionality of the application and improve their overall experience with it.

Heuristics matched to their frequency

**Figure 3: The frequency of the heuristics**

5. To avoid potential confusion, we can improve the visibility and the prominence of the error message that is displayed when a user attempts to create objects with an empty title field. The user will then know they have to type at least one character to proceed with the creating or renaming action.

6. To improve the user experience, we have spaced out the edit and delete buttons, as their close vicinity could make the user click one by mistake.

7. To have our application provide maximal usability to the user, it may be helpful to incorporate a feature that allows users to add or edit descriptions for cards, either through a dedicated edit screen or through a tooltip or context menu option. This can help users better organize and contextualize their content, leading to a more efficient and effective workflow.

8. To address the issue with self-deletion, it may be helpful to add a confirmation message or dialog box that appears when a user tries to delete a card or list, allowing them to reconsider before making any irreversible changes, which would help prevent this issue.

### 3.3 Prioritization of problems

When analyzing the raised problems, we took a look at the main heuristics concerning them, because we believe heuristics make a good starting point for overall efficient categorization. We have found that the main three troublesome ones were:

1. Flexibility and efficiency of use;
2. Help users recognize, diagnose and recover from errors;
3. Help and documentation
4. Consistency and standards;
5. User control and freedom.

From here we derived connections within our problems, and were able to prioritize them:

1. Limited display capacity for cards on lists;
2. Small text and button size throughout the application;
3. Lack of cancellation feature for new card creation;
4. Edit and delete buttons in close proximity;
5. Small error messages;

6. Small text and button size throughout the application;
7. No fail-safe for deletions;
8. Absence of clear application description.

Based on the feedback we received, there is a clear pattern of features of advanced functionality our further implementation should focus on, namely the detailed overview of a card and a start screen, containing the overview of all recently accessed boards.

## 4 CONCLUSIONS AND IMPROVEMENTS

After the experts assessed our prototype, based on the heuristic problems they highlighted, we prioritized the issues in order of importance and settled upon changes that improved them through a thorough group discussion.

### 4.1 Main Conclusions

Based on the received feedback, there seem to be several usability concerns with our application, which were overlooked in the initial stage of development. These include:

1. Inconsistent placement of important functions, such as the "delete" and "rename" buttons, which can lead to accidental removals.
2. No option for the cancellation of the addition of a card.
3. Inconsistent font and size of texts across the application, which could cause unpleasant experiences for the users.
4. Too large cards compared to lists, which limits the number of cards within a list.
5. Small error messages when a user displays 'unaccepted' behaviour. To improve the usability of the application, it was necessary to address these points and redesign certain aspects of the interface so we can deliver a more convenient way for users to utilize our app. Additionally, incorporating consistent feedback into the design process helped us identify and address any remaining issues with our application.

### 4.2 Improvements

We proceeded to make the following improvements:

1. The 'rename' and 'remove' buttons are placed in a more appropriate layout.
2. The user is given the option to cancel the addition of a card.
3. The text size and fonts are consistent throughout the application.
4. The card size is proportional to the list size, i.e. smaller.
5. The error messages have larger font size and brighter coloring so that they appear more prominent.

After thorough consideration, we decided to make these improvements to maintain uniformity throughout the app. We have decided against implementing the fail-safe feature for the deletion of fields, as the most common scenario of accidental removal was caused by the proximity of the 'rename' and 'remove' buttons; hence with their new layout this has a smaller chance of occurring.

We, instead, focused on making our application responsive, as we strongly believe that it's a key factor for a pleasant user experience. We have provided snippets of our final GUI design which show that our application implemented all of the above-mentioned changes.

### 4.3 Final GUI design

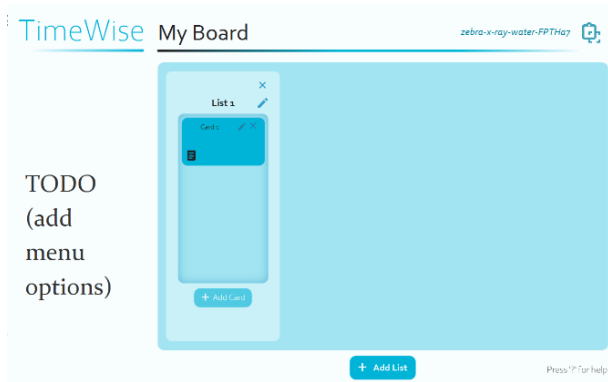


Figure 4: Initial Main Screen

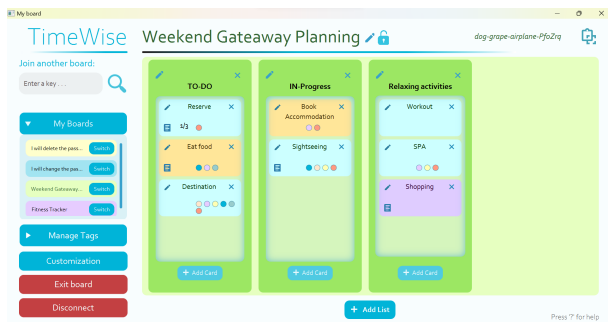


Figure 5: Improved Main Screen

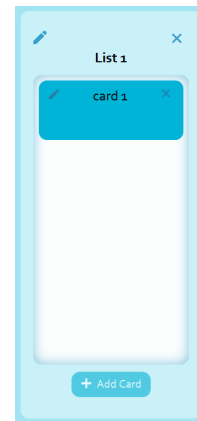


Figure 7: Improved List

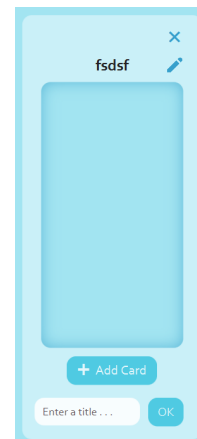


Figure 8: Initial Card Addition

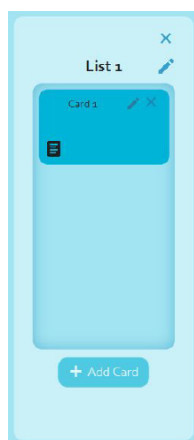


Figure 6: Initial List

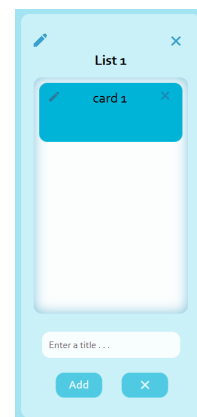
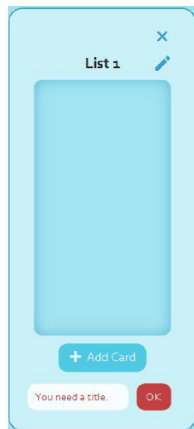
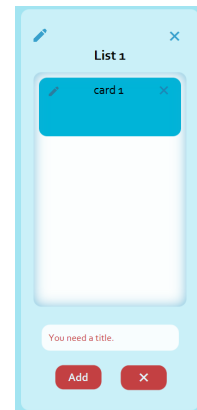


Figure 9: Improved Card Addition



**Figure 10: Initial Null Title Error**



**Figure 11: Improved Null Title Error**

## REFERENCES

- [1] [n.d.]. <https://www.nngroup.com/articles/ten-usability-heuristics/>.