

# OOP Project Report – Group 59

Borislav Marinov, Edward Oh Jang Hun, Wiktor Cupiał, Jaouad Hidayat, Eduard Faraon, Eduardo Hernández

## 1 INTRODUCTION

The objective of this report is to present the results of a heuristic evaluation conducted on a prototype of a GUI created on JavaFX for the Talio application. The Talio application is a project management tool that allows users to create and manage boards, lists, and tasks. The application has four main pages: the homepage, the overview page, the board details page, and the card details page.



Figure 1: Home Page

The homepage is where users establish a connection with the server. After clicking the 'Connect' button, the user is directed to the overview page.

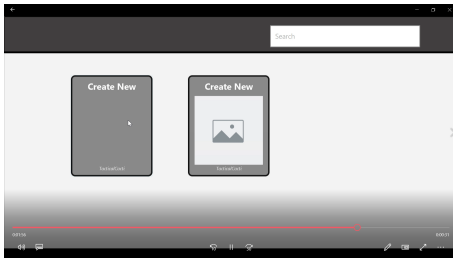


Figure 2: Boards Overview page

On this page, all existing boards are present, and there is a search bar that is still in the process of implementation. Users can create a new board by clicking the 'New Board' button, which takes them to the 'Board Details' page.

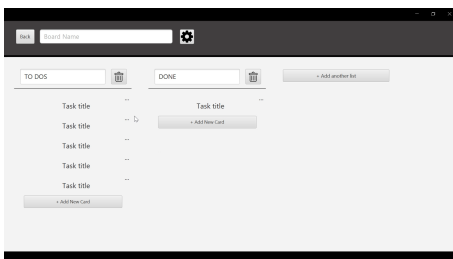


Figure 3: Board Details page

On this page, users can see the specific details of a board, which includes a collection of lists with different tasks inside of them.

Users have the option to go back to the overview page, customize and leave the board via the 'Settings' button that opens a smaller screen where options for background and font colour are presented.

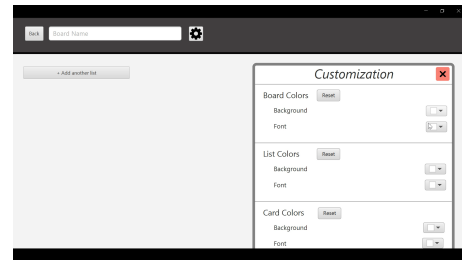


Figure 4: Customization page

They can enter a new list, delete an existing list, or add a new task. Tasks are added in a stack-like manner inside of a list. On each task, there is a 'Details' button that allows the user to enter the task and see its title, description, and list of sub-tasks.

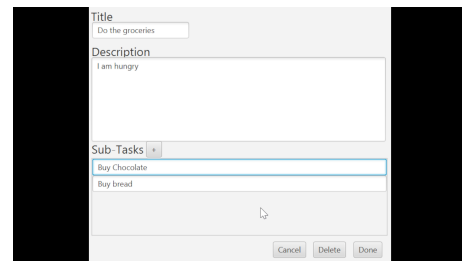


Figure 5: Card details page

On the task details page, users can add a new sub task via the 'Plus' button. This page also has three additional buttons that allow for saving an edit to the card, canceling an edit, and deleting the whole card from the list. The presented video went through all of the implemented functionality, allowing a full view of the features.

To evaluate the design, a group of 6 experts were recruited. They were provided with a video of a working version of the interface that presented a normal use case for the application.

## 2 METHODS

### 2.1 Experts

To do our evaluation we recruited 6 experts. They are experts that work in the field and they are familiar with the Heuristic Usability Evaluation and they have done it before. They have also researched

different designs for an application with a similar purpose, meaning they already have a perspective and idea, as to what such an application requires.

In this section of the report, we will describe in detail the procedure of conducting a heuristic usability evaluation, including the instructions for the experts, what they will be seeing, what they need to do step by step, and the heuristics they will be using.

## 2.2 Procedure

Beforehand, the group of experts received a document containing the guidelines for this evaluation. Inside this document we have placed clear instructions on how the review should be conducted. This includes using the 10 heuristics rules, following a certain feedback format for each issue found and grading this issue on a severity from 1-5 with 5 being the top priority for our team to address.

Alongside the instruction document, a video presentation of a typical usage scenario was presented to the experts. In the video, all implemented features of our application, the navigation between different sections and the design of each scene were demonstrated in a structured manner, so they can familiarize themselves easily with the state of the user interface at that time.

For this evaluation, each expert followed a simple but effective step-by-step list of actions:

- (1) Watch the video normally in order to familiarize themselves with the overall design and organization of the application.
- (2) The second step requires them to go through the video once more, but this time, they pause and examine each different scene that is shown. They focus on the design and accessibility of each feature and how intuitive the structure of each page is.
- (3) The third step, which is related to the second, is writing down the problems that the expert discovers during their more in detail examination of the video. They follow a format of presenting the location, essence and a possible fix for each issue.
- (4) The final step is to watch the overall video for the third time in a normal fashion in order to determine if there is an issue with the transitions and the whole interaction flow of the application after they have become more familiar with each separate scene.

The experts evaluate the user interface against a set of heuristics or guidelines. Every problem that the experts present is connected to an certain heuristic. After all experts have completed their individual, objective and unbiased review of the user interface, they come together and combine their lists of problems, presenting a final list of the frequency of each issue and its overall final severity rating.

## 2.3 Heuristics used

The heuristics used in a heuristic usability evaluation can vary depending on the project or design being evaluated. However, for the goals of this heuristic evaluation we have chosen 10 rules for the experts to organize found problems. These include:

- **Visibility of system status:** This means that the system should always provide feedback to users within a reasonable time frame, so they are informed about what is going on.
- **Match between system and the real world:** A system should use language, concepts, and phrases that are familiar to users, rather than system-oriented terms. It should also follow real-world conventions, presenting information in a logical and natural order.
- **User control and freedom:** Users may choose system functions by mistake, so it's essential to provide an emergency exit that allows them to leave an unwanted state without going through a lengthy dialogue.
- **Consistency and standards:** Users should not have to wonder whether different words, situations, or actions mean the same thing. The system should follow platform conventions and use consistent language and visuals.
- **Error prevention:** If errors do occur, the system should provide clear messages that indicate the problem and offer solutions.
- **Recognition rather than recall:** Users shouldn't have to remember information from one part of the dialogue to another. The system should make objects, actions, and options visible and provide instructions when appropriate.
- **Flexibility and efficiency of use:** The system should cater to both inexperienced and experienced users, with features that can speed up the interaction for users.
- **Aesthetic and minimalist design:** The dialogues should not contain information which is irrelevant or rarely needed. This allows for all relevant units of information to be clearly distinguishable.
- **Help users recognize, diagnose, and recover from errors:** Error messages should be expressed in plain language, precisely indicate the problem, and constructively suggest a solution.
- **Help and documentation:** If necessary, documentation should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.

## 2.4 Measures

The experts each give individual reviews and their own interpretation of the severity of each found issue on a scale of 1 - 5. They provide us with a thorough problem description and place their rating next to that description. The issues are separated per heuristic. Then, they come together and determine which problems from all the found ones are the biggest concern based on how crucial it is to be implemented in the application for user satisfaction and ease, which is the final raw format of our results.

## 3 RESULTS

For simplicity sake and display of results, we have calculated the average severity rating of each heuristic's problems. We have decided to show the final results in a table (Table 1) containing reduced data from the feedback document, the average and peak severity for each problem in the heuristics so we can clearly see what heuristic is in the most severe state overall. We also decided to create a column for the highest severity as this will help to distinguish

heuristics that have a high deviation in the problems found, which allows us to determine outliers and more serious problems in each category. This approach of portraying results also allows for quick demonstration into which heuristics have the most troublesome issues for the current state of the application and where the most improvements would be needed. For example, a severity rating of 1 and 2 suggests that it is not crucial to address this issue immediately and can be left out of this summary section, while a severity of more than 3 is definitely something to be quickly addressed and resolved.

Usability Heuristic	Average	Peak
Visibility of system status	3	4
Match between system and the real world	3	3
User control and freedom	3	5
Consistency and standards	3	4
Error prevention	3	3
Recognition rather than recall	2	2
Flexibility and efficiency of use	4	5
Aesthetic and minimalist design	3	3
Help users diagnose and recover from errors	2	2
Help and documentation	1	1

**Table 1: Representation of the upwards rounded average severity for problems and the highest severity problem per heuristic.**

The complete feedback document has underlined a range of problems with the heuristics of our prototype. Prioritizing which of these problems we tackle is a very important part of the post-evaluation process. As we do not have unlimited time, we need to rank them against each other by evaluation which ones are most fundamental for our application. The most important problems we have determined with help of the experts are the following:

- There is no button that can be used to disconnect from the joined server. This is the highest priority problem, as it has a maximum severity rating and is highest on the minimum requirements we have determined for our application. It is connected to one of the most crucial heuristics - user control and freedom, which is related to a quality user experience.
- The second problem is connected to the visibility of system status and error prevention. The user is not presented with a dialog for confirmation whenever he tries to perform destructive and non-reversible actions such as deleting tasks and lists, which could result in accidental loss of information.
- Another problem for the user control heuristic is the option to exit a board. The application does not allow the user to press a button to leave a board after he has entered it in the single board overview.
- The next issue is related to flexibility and efficiency of use, as there is no button to delete a sub-task from the task details, which limits the organization of each task.
- The final problem we have selected to explicitly address is a design consistency issue, as the search box for inputting a

board key to join a board on board overview page changes its location when switching between scenes a multitude of times.

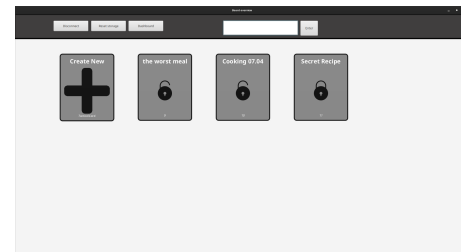
There were more highlighted issues that concerned themselves with minor changes in the interface that are not critical to an adequate user experience, which is why they were left out of the above list. The exact way we were able to separate between these issues and the more important ones is that we compared them to the minimum requirements that we have determined this application and the severity that experts allocated to each problem. The more they were of importance to basic functionalities that our research found and the higher the severity, the bigger the possibility for a problem to be categorized as critical and included in this results section.

## 4 CONCLUSIONS AND IMPROVEMENTS

Based on the results of the evaluation, we have identified several usability issues that need to be addressed to improve the user experience. These issues are related to the most problematic heuristics from our results, namely consistency, user notifications, error prevention, and user freedom. The main conclusion we can draw from the results section is that the current state of the application at time of evaluation did not allow the user full control over each component and decision on the interface.

In order to deal with this troublesome quality of Talio, we have introduced several improvements to its design and user interactions with the UI.

Firstly, we added a disconnect button to the board overview page to allow users to switch between different servers. In figure 2, which is the old design for this page, there is no disconnect button, while on the image below, you can see the newly implemented feature in the left section of the utility bar.



**Figure 6: Disconnect button on board overview page**

The second big improvement that we want to highlight is that right now every action that is destructive (irreversible) asks for confirmation from the user prior to any execution in contrast to before when pressing the delete button for a list would instantly delete it from interface and database.

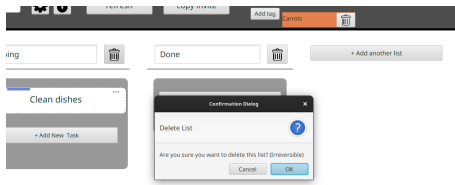


Figure 7: Warning displayed when deleting list

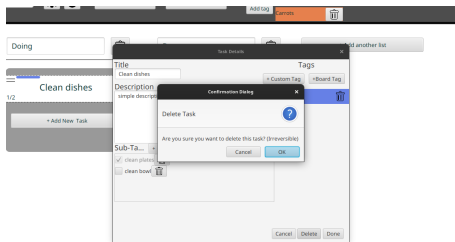


Figure 8: Warning displayed when deleting task

We designed these alerts in a way that gives the user explicit information that the action he is going to perform is irreversible. We hope that this change will greatly improve visibility of system status and error prevention in our application.

The third major improvement is the ability of a user to leave a joined board. This is an essential addition, as it allows the user to clean up his workspace and achieve a more satisfactory experience. On the previous customization page, there was no evident leave board button, while on the updated version, there is an informative option to remove board from the joined boards.

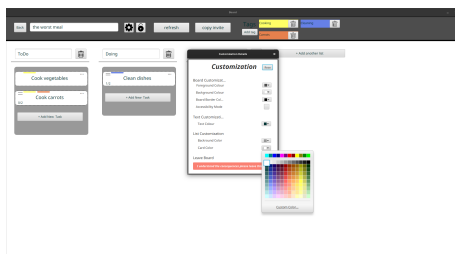


Figure 9: Customization page, leave board button

Addressing the fourth issue from the prioritized problems list, we have added the capability for a user to delete a sub task from a task to allow for easier management of the smaller task details page. This gives the user the ability to remove unnecessary components from the interface, making it less polluted. Before the update, a sub task was a plain text field, but now it is more interactive, allowing for selection and deletion (see figure 5 for old version).

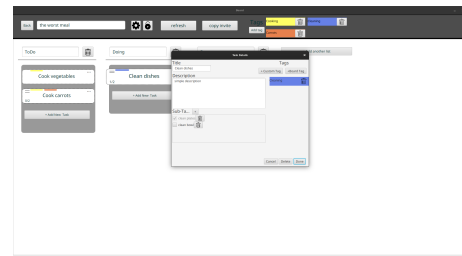


Figure 10: Sub tasks update

The final improvement from the problems list is the new search box, which allows the user to join a board via a key. Before, this text field would not be visible when loading the boards page, but now it is clearly present at all times. This is a major improvement, as the user can now join a board at any moment he desires to, giving him more freedom of use (Please refer to figure 6 to see the updated version of the search box).

If we were to examine the severity averages and peaks from Table 1, it is evident that the flexibility and efficiency of use heuristic was the one our previous design had the most faults with. For this reason, we have implemented some features and interface changes to improve upon it. We added 2 main changes for this heuristic:

- shortcuts for editing and easier navigation in a single board

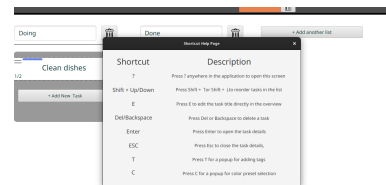


Figure 11: Shortcuts documentation

- a default connection button, so you do not have to write a server address for your server

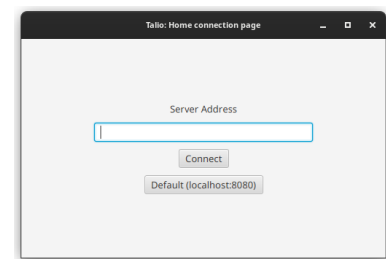


Figure 12: Default connect button

Overall, we believe that these changes lead to a significant improvement in the user experience of our Talio application.