OOP Project Report – Group 79

Leonardo Marcuzzi, Aistė Macijauskaitė, Aldas Lenkšas
Jan Maris, Rebecca Andrei

1 INTRODUCTION

The purpose of TALIO is to provide an efficient and effective way for individuals or teams to manage their tasks and projects. Therefore, the usability of the prototype is of utmost importance as it directly impacts its usefulness and adoption.

A heuristic evaluation will involve examining the interface design, information architecture, and the interaction patterns that users have with the prototype. It will also take into account the established usability heuristics such as visibility of system status, consistency and standards, and error prevention and recovery, among others. The evaluation will identify areas where the application meets or falls short of these usability principles.

2 PROTOTYPE DESCRIPTION

The initial interface of the prototype presents a Login Screen *(fig. 1)* that features the Talio logo and text fields prompting the user to input their username, password, and Server URL. The Login Screen also contains two buttons, namely the Login Button and the Sign Up Button, which perform their corresponding functions when clicked. The latter opens the User Registration Screen *(fig. 2)*, which includes multiple Text Fields requiring the user's personal information such as their first and last name, desired username, password, and Server URL they intend to access. The User Registration Screen also includes two additional buttons, the Register Button and the Close Button. The Close Button leads the user back to the Login Screen, while the Register Button directs them to the same scene as the Login Button.

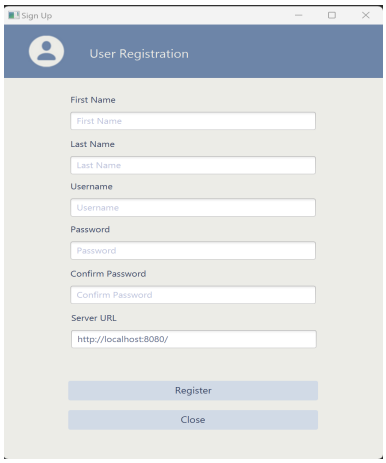

Figure 2: Sign Up Screen

Upon pressing the Login Button from the Login Screen, the user gains access to the Card List Overview Scene. Initially, the Card List Overview Scene is empty *(fig. 3)*, except for a "Disconnect From Server" Button, which returns the user to the Login Screen, and a "+" Button, which enables them to add a new list *(fig. 4)*. Each list includes an editable title, denoted as "New list" initially, but which can be modified by double-clicking on the Title Text Field and inputting a new name *(fig.5)*. Each list also includes a trash can icon that deletes it when pressed *(fig. 6)*, and another "+" Button, which allows the user to add a card to it by opening a new Add Card Scene *(fig. 7)*. The Add Card Scene presents a Text Field that prompts the user to give a name to the new card and an "OK" Button, which adds the card to the specified list and directs the user back to the Card List Overview Scene *(fig. 8)*.

Figure 1: Login Screen

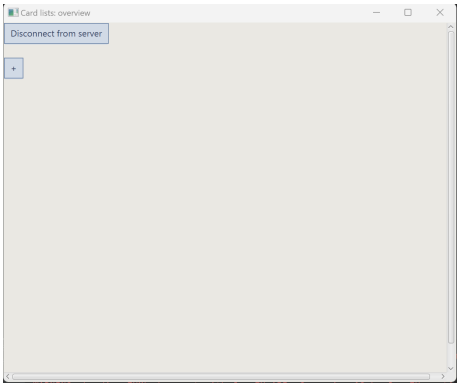

Figure 3: Empty Card List Overview Scene

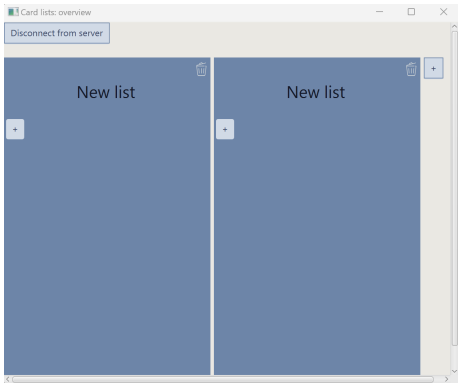

Figure 4: Empty List Template

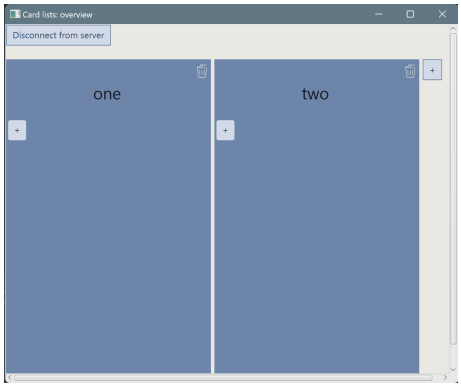

Figure 5: Update List Title

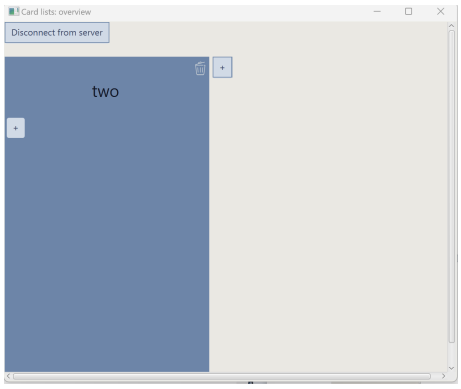

Figure 6: Delete List

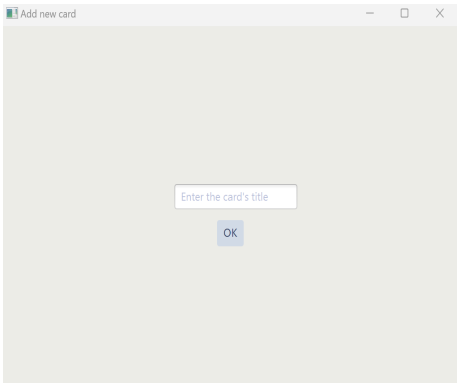

Figure 7: Add Card Scene

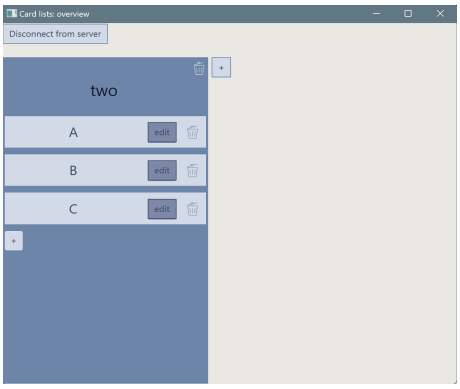

Figure 8: List with Cards

Similarly, the card in the Card List Overview displays its title and two buttons, namely the "Edit" Button and the trash can icon, which allows the user to modify the card's name by reopening the Add Card Scene *(fig. 9)* or delete it permanently *(fig. 10)*, respectively. Additionally, the user can drag and drop a card from one list to another *(fig. 11, 12, 13)*.

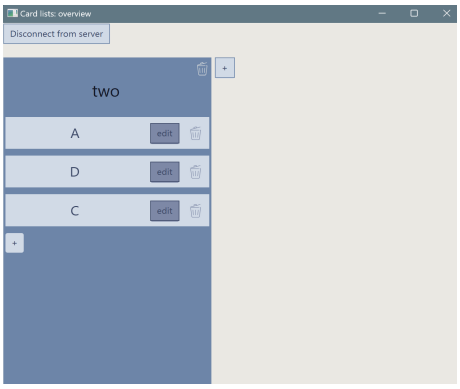

Figure 9: Update Card Title

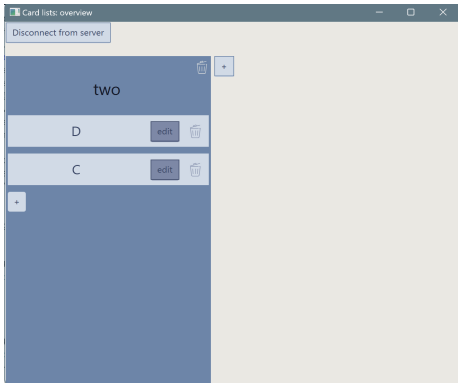

Figure 10: Delete Card

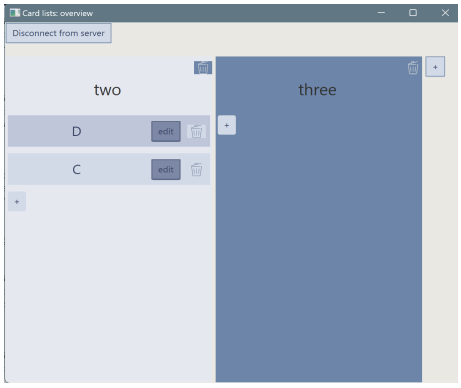

Figure 11: Selecting a Card

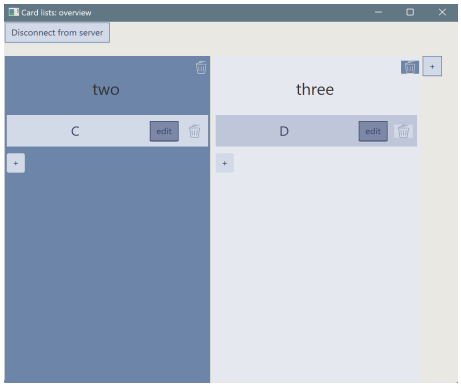

Figure 12: Dragging the Card

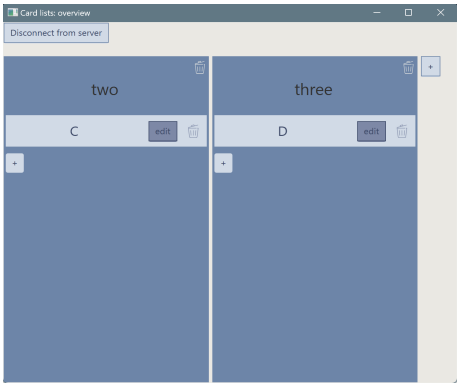

Figure 13: Dropping the Card

3 METHODS

3.1 Experts

The experts we have chosen to aid us in completing the heuristic usability evaluation were five other students in the Computer Science and Engineering bachelor program at TU Delft from another Group. Taking into account the fact that they have also been working on their own version of TALIO for some time, it would be fair to conclude that they have full knowledge of the scope of the application, its libraries, and APIs.

3.2 Procedure

We used a cognitive walkthrough model to test the usability of our project. The experts received a working prototype of our application and were asked to go through several scenarios, provided below, whilst writing down their thoughts. This evaluation would examine the steps a new user would take to complete these tasks and identify any potential issues that may hinder their progress.

The evaluation process involved each expert examining the application individually with complete control over the interface, having been encouraged to explore unique cases, such as adding a card with no title, or an extremely long title, so that they could test and review the prototype's full potential. To ensure a controlled evaluation outcome, several observers were present during this process, providing guidance and hints to the experts as needed while keeping the evaluation focused on the predetermined objectives. Additionally, we requested that our experts navigate through every part of the application multiple times, in order to further increase the rigor of the evaluation, but also to recreate a typical usage scenario, where the users are already familiar with the prototype.

This evaluation also considers the user's mental models and expectations, i.e., how the user expects the website to behave based on their prior experience with similar systems. This involves examining the consistency of the interface design and the navigation system with common user expectations. The cognitive walkthrough would also consider potential errors that a user may make during the task completion process and assess the prototype's ability to recover from such errors. The experts would identify any error messages that may be confusing or unhelpful to the user, and suggest

ways to improve the recovery process.

Below are the five main tasks that a user may use the application to complete:

(1) Sign up / Log in.
(2) Enter a board with a server URL.
(3) Add, remove, and edit lists within this board.
(4) Add, remove, move, and edit cards within a list.
(5) Work on the same board with multiple clients.

Evaluators assessed these five key tasks that users typically perform on the TALIO application. The tasks were then evaluated by using Jakob Nielsen's ten heuristics for user interface design, with a focus on assessing the processes involved from a user's perspective:

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.

3.3 Measures (Data Collection)

Our primary goal throughout the evaluation was to measure how client-friendly the prototype was. Additionally, since we allowed our evaluators full freedom over the application, we were interested to see if they could identify design bugs that had previously escaped our notice. Thus, we were looking to our evaluators to report both issues with the application itself, as well as suggestions to aid in making the interface more user-friendly, and overall improving the GUI.

Our methods of collecting the data were simple, but effective - whilst using the application, the experts remarked upon everything that they felt could be even slightly improved. As such, the data from the evaluation was presented in the form of a text file consisting of notes which highlighted specific issues that the experts observed throughout their exploration of the app. For instance, when the evaluators attempted to enlarge the application to full screen, they noticed that it did not scale properly. As a result, they noted that "the application has a non-responsive design on full-screen". All of the data we collected follows the same format as this example, both for describing issues that have to do with usability and also for various UI design changes.

4 RESULTS

To further examine each of the problems, we estimated its severity in terms of usability principles. We looked at two factors to determine the severity - frequency and impact. The impact is considered high when it hinders users from utilizing the website and, therefore, is difficult to overcome. The severity of frequency is high if the problem occurs frequently within the application, causing inconvenience to the user. In a four-quadrant priority matrix, the problem could fall into one of the four categories:

- High impact and high frequency
- High impact and low frequency
- Low impact and high frequency
- Low impact and low frequency

By using a four-quadrant priority matrix, we can prioritize the problems based on their severity and determine which ones require immediate attention. This approach helps to focus our efforts on the most critical issues and make the most efficient use of the resources.

Here is the list of problems, each problem is linked with one or more heuristics that it violates, and also has a prioritization level assigned to it.

- When opening multiple clients, making changes on one board do not appear automatically on the other.
  - 1: Visibility of system status
  - High impact and high frequency
- Non-responsive design on full screen.
  - 4: Consistency and standards
  - High impact and high frequency
- Non-consistent UI (e.g. adding a card list and a card looks very different).
  - 4: Consistency and standards
  - High impact and high frequency
- User cannot drag lists.
  - 3: User control and freedom
  - High impact and high frequency
- Delete icon is almost not visible.
  - 1: Visibility of system status
  - Low impact and high frequency
- When entering an invalid URL address and then entering a valid one, the error message stays.
  - 1: Visibility of system status
  - High impact and low frequency
- User cannot add a card with an empty name (but it doesn't say that anywhere).
  - 5: Error prevention
  - High impact and low frequency
- The buttons do not feel like buttons(there is no visual indication).
  - 1: Visibility of system status, 8: Aesthetic and minimalist design
  - Low impact and high frequency
- Application doesn't ask the user for confirmation before deletion.
  - 3: User control and freedom
  - Low impact and high frequency
- Card list name letters are too big.
  - 8: Aesthetic and minimalist design
  - Low impact and high frequency
- If a very long name is entered for a card list, it doesn't show the full name. It's confusing.
  - 2: Match between system and the real world
  - Low impact and low frequency
- When a user wants to move a card by dragging a list to the end, it would only go to the end of the frame of the window.

- 2: Match between system and the real world
- Low impact and low frequency

5 CONCLUSION AND IMPROVEMENTS

5.1 Improving Buttons

Throughout the assessment process, the experts provided numerous observations pertaining to the visibility and general design of the buttons embedded within our program. Their feedback indicated that the delete button lacked visibility and that the buttons incorporated throughout the application did not conform to their mental models of what a button is supposed to look like based on their previous experience.

To redress this issue, we have implemented a hover behavior on the buttons such that when a user hovers their mouse over one, a slight visual effect will be produced, making it more evident that the button is clickable *(fig. 14)*. Additionally, we have modified the cursor's appearance and enhanced their visibility by labeling them appropriately, thereby making their function clear to the user.

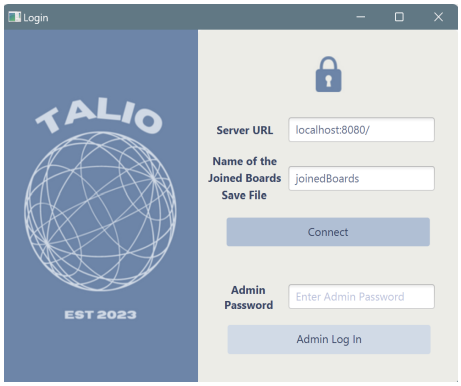

Figure 14: Mouse Hover over Connect Button

Moreover, after implementing the Multi-Board Milestone, in our Board Overview scene, we have added a menu that features clear indications for the Disconnect, Join, and Add New Buttons *(fig. 15)*. Not only are these buttons clearly labeled, but they also have associated images that correlate with their function. Furthermore, they incorporate a hover effect when the user's mouse is on them.

Additionally, we have employed a simple and intuitive design for the template of our Boards. The "Leave" and "View" Buttons are clearly labeled, and the visibility of the Delete button has been vastly improved by increasing the contrast between the background color and the icon color.

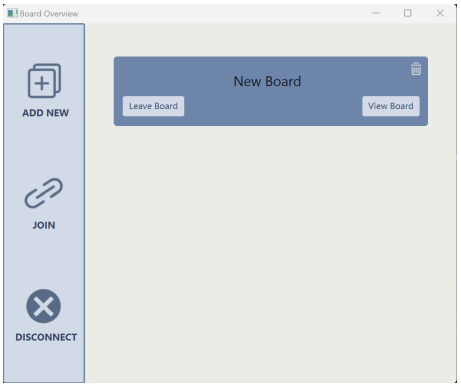

Figure 15: Board Overview

5.2 Implementing Pop-Ups

During their exploration of our application, our experts noted a concern regarding the absence of pop-up windows accompanying critical action buttons, such as the delete button. As a result, the user was unable to take precautions to prevent deletion after pressing the button.

To remedy this issue, we have introduced confirmation pop-ups to various buttons. Specifically, we have attached a pop-up window to every deletion button, prompting the user to confirm or cancel the action *(fig. 16)*. Additionally, when joining a board, a pop-up window appears requesting a board invite key *(fig. 17)*.

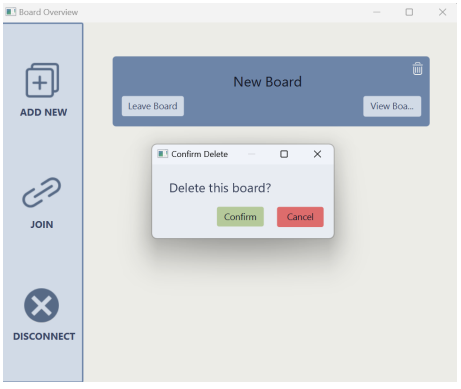

Figure 16: Delete Board Pop Up

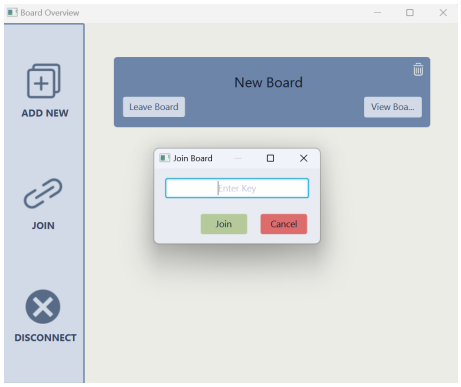

Figure 17: Join a Board Pop Up

Figures 16 and 17 are a demonstration of some of the pop-ups we have integrated throughout our program. They have a standardized design, featuring buttons that are both labeled with text and color-coded to indicate their respective functions. Furthermore, the user has the ability to reposition the pop-up to view a specific section of the overview, if needed.

5.3 Flexible Display

One of the primary observations made by our experts early on in their evaluation pertained to the absence of full-screen support in our application. When attempting to maximize the window, the user interface would appear distorted and confined to a corner of the screen, rendering the design non-responsive.

It was crucial to rectify this issue, as full-screen support is considered a standard feature in almost all software. Consequently, we decided to render the entire application adaptable to any window size, ensuring that the UI adjusts dynamically to suit the display used by the user to access the application.

Figure 18: Example of Full Screen Application

Illustrated in Figure 18 is an exemplar of our Card List Overview Scene in full-screen mode. With the recent modifications, the lists now span vertically to occupy the entire height of the screen. Furthermore, the menu and list template maintain their structure and effectively segregate the various buttons, resulting in a clear and organized display.

5.4 Design Consistency

Our experts provided feedback regarding the overall design of our application, highlighting a need for greater consistency. While the aforementioned changes have already contributed towards achieving a more uniform design, we recognize the need for further improvements.

To this end, we have made several adjustments to our overviews and templates to ensure that the design is consistent and intuitive throughout the entirety of the application.

Figure 19: Example List Pop Up

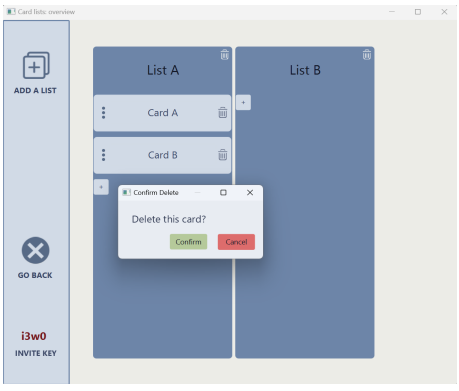

Figure 20: Delete Card Pop Up

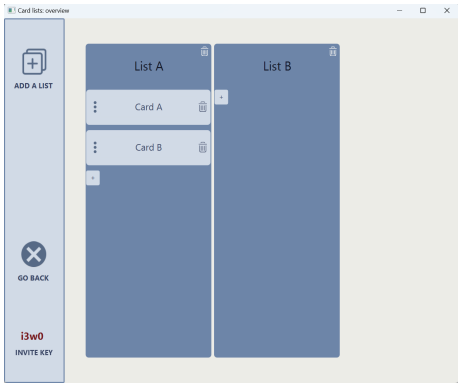

Figure 21: Example of a List Overview for a specific Board

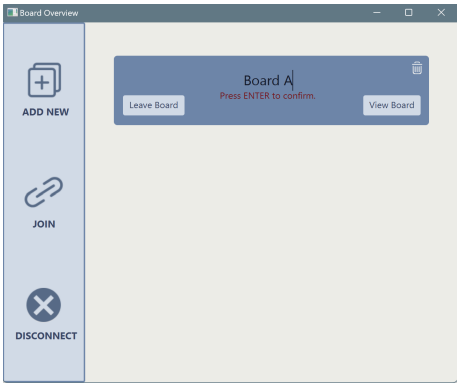

Figure 22: Board Title Updating

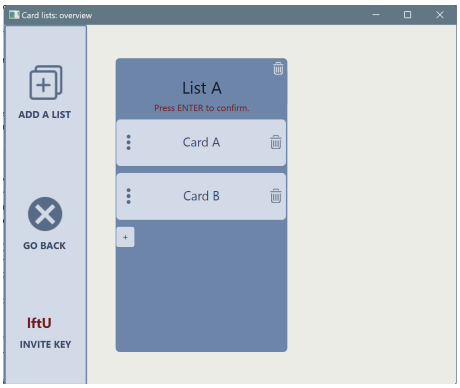

Figure 23: List Title Updating

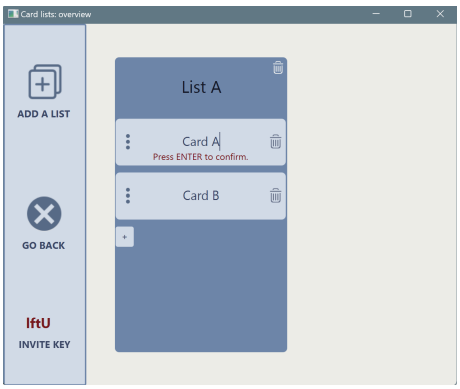

Figure 24: Card Title Updating

As previously stated, our pop-ups have an almost identical design, featuring a prompt that helps to clarify the action being confirmed or canceled by the user. Similarly, both the Board Menu and the List Menu adhere to a consistent design and layout, with buttons that have comparable functions bearing similar labels and images.

Moreover, we have employed a cohesive color palette throughout our application, which greatly facilitates user navigation. In lieu of an Add Card Scene, creating a new card is accomplished by pressing the "+" button, generating a card with the placeholder "New Card", much like how adding a Board or List is executed by interacting with the "Add New" button. These measures work together to promote a consistent and streamlined user experience.

5.5 Intuitive Title Updating

In earlier versions of our prototype, when a user attempted to modify the title of a Board, List, or Card, the new name was updated with every keystroke. However, when multiple clients are running, excessive communication between them caused delays in updating the changes. To circumvent this issue, our latest prototype version implements a modification that saves the changes only when the user presses the ENTER key.

Consequently, every time a user begins to type in a title, a warning label appears, reminding them to press the ENTER key in order to ensure that their modifications are saved. This change is intended to optimize the user experience by reducing delays and streamlining the communication process between multiple clients.

5.6 Removing Login/Sign Up Interface

In response to the feedback provided by our experts, we have made a significant change to the user authentication process within our application. Specifically, we have removed the Login / Sign Up interface which was deemed unnecessary. In its place, we have implemented a new menu that enables users to connect to the server

as either an Administrator (with a randomly generated password required) or a normal user.

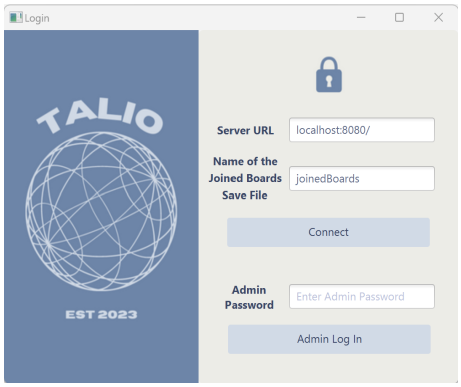

Figure 25: New Login Interface

5.7 Conclusion

From this particular evaluation, it can be inferred that the product is still in its nascent stages and requires further development. The identified deficiencies in the user experience and interface design suggest that the application is lacking in its usability and aesthetic appeal. These findings highlight the importance of having a user-centered approach in software development, where user feedback is taken into account during the development process. Consequently, the improved version of the application sought to address the issues highlighted during the evaluation by providing users with an intuitive and aesthetically pleasing interface.