OOP Project Heuristics Usability evaluation - Group 60

Lucas Paul, Daria Bucur, Mateo Nasse, Maxim Istomin, Damjan Kosutic

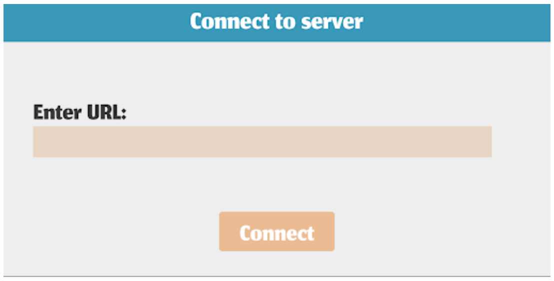
1 INTRODUCTION

The objective of this report is to discover and evaluate as many problems/inconveniences in our user interface design from multiple perspectives so that we can solve and/or get rid of these before we go further with implementing our design. We will be reviewing usability issues in different areas that cover the whole structure of the project. There will be a priority for those that rank the highest in the scales of frequency and impact. This will help improve the usability of our application and the experience of the average user. We do not want any type of user to have problems with the application.

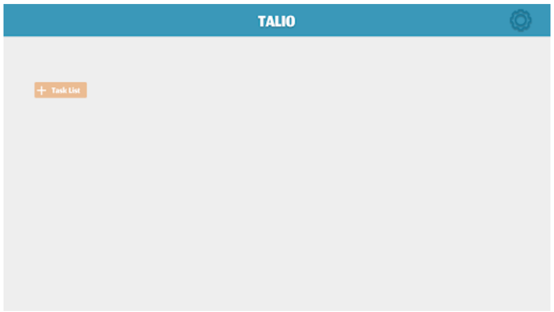
We chose to design our prototype in Figma because of the versatility of its features and the ease of sharing it provides. Most importantly, Figma provides an easy and intuitive way of showing the flow of the application by linking buttons and nodes to take the user to different screens depending on which options they choose. This allows us to replicate the feel of the real application effortlessly, and convey the ideas that are being implemented. It is also useful as it gives us a lot of freedom in what shapes, boxes, buttons, and colors to use to replicate our idea as precisely as possible. For our prototype we chose a lighter color palette that is easy on the eyes but in the future we would want the user to be able to choose between different color palettes so they can have one that suits them best.

2 PROTOTYPE SHOWCASE

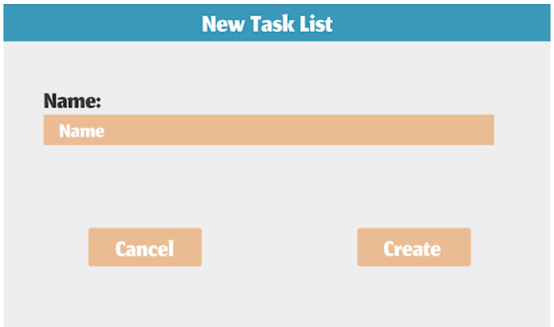
The prototype[1] we sent the experts starts with this screen.



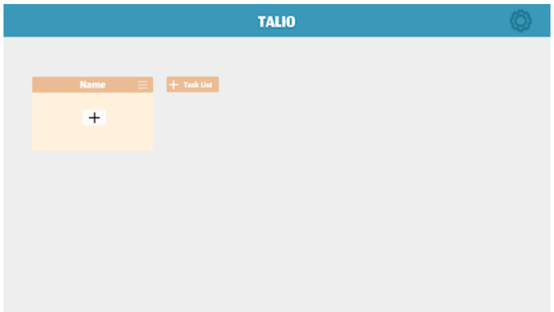
The prototype offers multiple choices by linking the buttons on screen to other screens. For instance, clicking on the connect button (assuming a correct URL is entered) takes us to this screen:



This is the main screen of the application. It clearly states the name of the app as well as two interactive buttons. One of those buttons is the "Add Task List" button, which will add a task list to the board after one is created. Clicking on it takes us to the following screen:

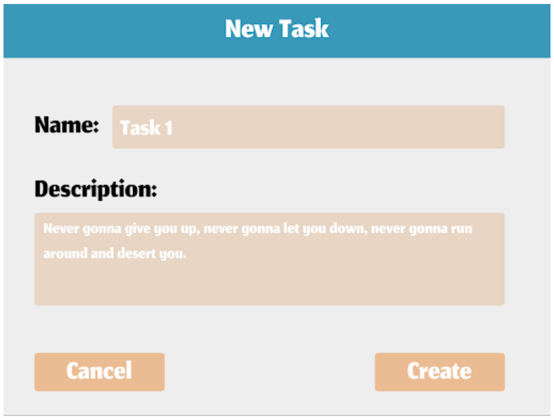


As you can see, there are two options, creating the task list or canceling the action, which effectively takes us back to the previous screen. When a task list is created, it gets added to the board. That looks like this:



[1] https://www.figma.com/file/NqFYnYvu45aCSZNgWsXR8c/Pre-Feedback-Prototype?node-id=0-1&t=uvzYoCkJpBTrY041-0

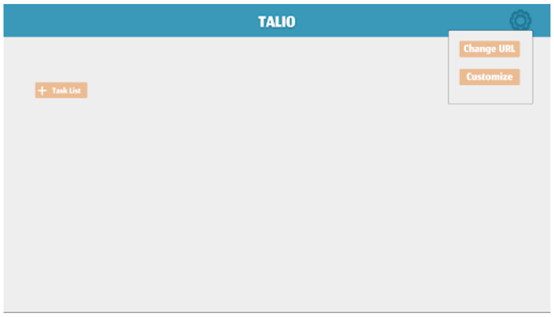
At this point there are a couple more options that show the flow of the application. For instance, since there is a new task list on screen, the button for adding a task also shows up. Clicking on it will take us to a similar screen to the one we saw when clicking on the "Add Task List" button.


New Task

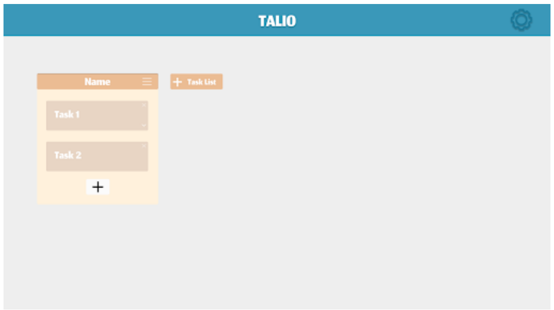
Name: Task 1

Description: Never gonna give you up, never gonna let you down, never gonna run around and desert you.

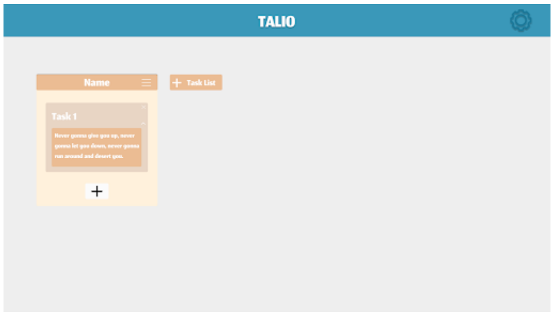
Of course we are limited to how many screens we can make as increasing the possibilities will eventually lead to an exponential increase in screens that we would have to design, but we have included all screens necessary to understand the whole idea behind the application. All buttons are able to be clicked more than once. Here are some other screens showing the flow of the application: When the options button is clicked, this is what it shows:



Two options, to change the URL of the server, to reconnect, and the customize button, which gives options of themes to choose for the coloring scheme of the application.



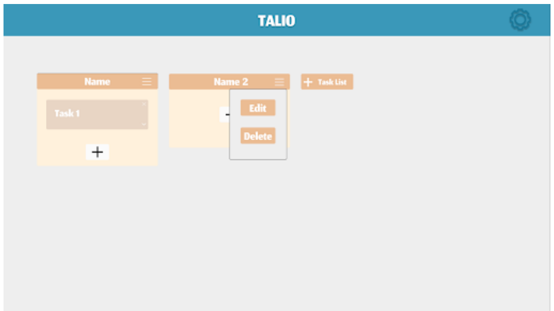
This is what the board looks like with multiple tasks in a task list. As you can see, the tasks have two buttons. An "X" for deleting the task, and a caret button for expanding the task. Clicking on the "X" deletes the task and clicking on the caret button will open the task with its description. Like this:



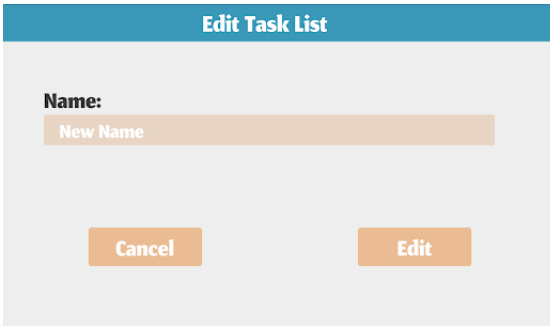
Task 1

Never gonna give you up, never gonna let you down, never gonna run around and desert you.

The task list also has a triple bar button on the top right, which props a pop up showing the options to edit or delete the task list.



Similar to the task, clicking on it will delete the task list. Clicking on the edit button will open a window to edit the name of the task list.


Edit Task List

Name: New Name

And thus the prototype of our application. As stated before, this is what the experts went through multiple times to look for improvements.

3 METHODS

We recruited one team of six experts. They were another group in CSE 1105 (Computer Science & Engineering, Object Oriented Programming Project) at the TU Delft and thus they had no previous experience with making heuristic evaluations. This selection was done because of a lack of accessibility of professionals in that field for our team. The procedure the experts followed is the following. Firstly, they were instructed to go, on their own in isolation from the other experts, through the whole prototype at least 3 times, writing down the usability issues they could find. The prototype they went through is the prototype that is described in the Introduction. They would start from the opening page/pop-up and try to simulate a usual usage of the application and go through all the available options. Each usability issue was noted down with the format described in the Measures section. The usability heuristics they were instructed to use are listed below and are the ones found by research by Nielsen, J. (1994) [2]. Secondly, they would share their findings with the rest of the experts and discuss if they agree with the issues noted, their respective descriptions and use cases, the impact and frequency, and the heuristics they are categorized by. Thirdly, the experts would go through the design one final time as a team to try to spot anything that they have not seen yet and write these down alongside the other problems already found. Lastly, the experts would individually send their lists of issues to one of the members of our group, in a pdf format.
These are the heuristics which we asked the experts to use:

- Visibility of system status
- Match between system and the real world
- User control and freedom
- Consistency and standards
- Error prevention
- Recognition rather than recall
- Flexibility and efficiency of use
- Aesthetic and minimalist design
- Help users recognize, diagnose and recover from errors
- Help and documentation

This format of the procedure was chosen as having the experts work individually increases the chance of specific and realistic usages of the application and hence decreases the omission of usability issues. Then, by making them work as a group, they would refine the problems to make them more comprehensive and accurate, as well as find some problems they might have not spotted individually. Also, we recruited the whole team so as to increase the chance of finding as many of the problems as possible, consistent with the research done by Nielsen, J. & Molich, R. (1990, March) [3].

3.1 Measures

The experts needed to report the usability problems they found in the prototype. The format they were instructed to use is listed and explained below. This formatting of the problems was chosen so as to provide a concise, distinct, and clear form, making it easier to avoid duplicates. The format and descriptions were taken from research done by Cockton et al. (2004) [1].

- Problem description: a brief description of the problem
- Likely/actual difficulties: the anticipated difficulties that the user will encounter as a consequence of the problem
- Specific contexts: the specific context in which the problem may occur
- Assumed causes: description of the cause(s) of the problem

4 RESULTS

The following is the list of the usability issues our prototype received from the experts. The issues have been shortened to a brief description and are ordered by severity.

(1) Visibility of the X button for deleting tasks - high severity (8 impact, 10 frequency), heuristic: aesthetics and minimalist design.
(2) No option to edit or to mark a task as completed, only to delete it - high severity (impact 7, frequency: 9), heuristic: error prevention.
(3) Missing edit task functionality - high severity (6 impact, 8 frequency), heuristic: user control and freedom, flexibility and efficiency of use.
(4) Missing delete confirm dialog for tasks and task lists - high severity (10 impact, 3 frequency), heuristic: error prevention.
(5) No ability to move tasks between lists (or within lists) - high severity (5 impact, 8 frequency), heuristic: user control and freedom, flexibility and efficiency of use.
(6) Visibility of the settings button - medium severity (6 impact, 6 frequency), heuristic: aesthetic and minimalist design, flexibility and efficiency of use.
(7) Missing exit button for the application - medium severity (8 impact, 4 frequency), heuristic: user control and freedom.
(8) Lacking ability to move the lists among each other - medium severity (7 impact, 4 frequency), heuristic: user control and freedom, flexibility and efficiency of use.
(9) Missing board title and server display on the board - medium severity (6 impact, 5 frequency), heuristic: visibility of system status.
(10) White text with a beige color have a bad contrast ratio - medium severity (6 impact, 4 frequency), heuristic: aesthetic and minimalist design.
(11) The style popup does not show the active style - medium severity (5 impact, 4 frequency), heuristic: flexibility and efficiency of use.
(12) Ambiguity of “Change URL” button - medium severity (4 impact, 4 frequency), heuristic: flexibility and efficiency of use.
(13) Misleading icon used for the settings button - low severity (3 impact, 2 frequency), heuristic: flexibility and efficiency of use.
(14) Lack of recommended action flow with the color of dialog buttons - low severity (1 impact, 1 frequency), heuristic: flexibility and efficiency of use.

5 ANALYSIS

From the feedback received, we have noticed a trend in some of the problems. One of the main suggestions we got was that there should be a middle screen between deleting things. This makes most sense to us when deleting a board which contains multiple task lists, and when deleting a task list, which can contain multiple tasks. But we decided that deleting a task was better to be a direct action since it can become cumbersome for the user to delete multiple tasks if each time they have to confirm the action. Another of the main suggestions had to do with the contrast of the buttons in terms of the color scheme. For example, the “X” button in the tasks was not easy to see. Similar buttons had the same issue. We were also missing some functionality, such as editing tasks. We decided to prioritize these issues as their severity was much higher compared to the others.

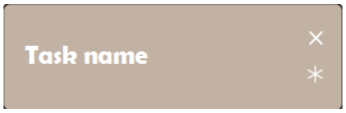
6 CONCLUSION

To fix the above presented problems, we decided to add multiple screens and popups to the application. The functionality for multi-board was also added, so we kept in mind the suggestions we had in the past to also implement it correctly. The colors and shapes of the different buttons were also edited. The final prototype of our design can be found here[2].

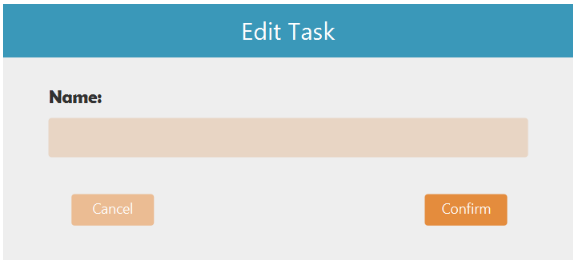
6.1 Improvements

List of the issues that were solved and how they were solved:

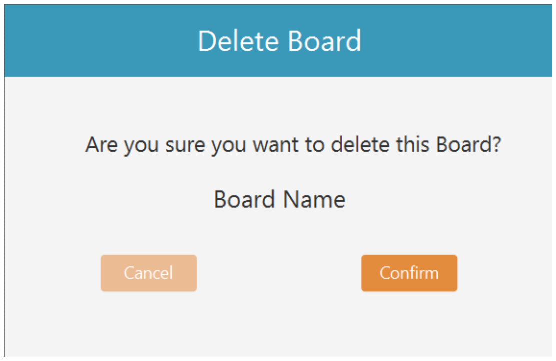
- 1: Visibility of the X button for deleting tasks - high severity (8 impact, 10 frequency), heuristic: aesthetics and minimalist design, flexibility and efficiency of use.
  - This issue was solved by making the X button a bit thicker so that it would be more visible, thus increasing the dominance of the X on the screen (helping the aesthetics) as well as it being more apparent, improving the efficiency of use.



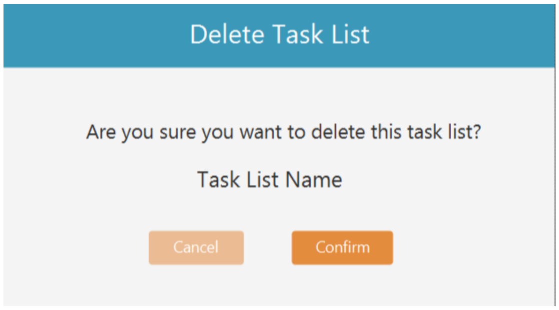
- 3: Missing edit task functionality - high severity (6 impact, 8 frequency), heuristic: user control and freedom, flexibility and efficiency of use.
  - This issue was solved by adding a star button to the tasks (as seen in the first issue) that opens a popup asking you to edit the task name, and you could cancel or confirm the edit of the name. When the confirm button is clicked it will update the name in the database and the actual screen.



- 4: Missing delete confirm dialog for tasks and task lists - high severity (10 impact, 3 frequency), heuristic: error prevention.
  - This issue was solved by creating a popup dialog when clicking on the delete button for task lists. This popup dialog was not created for tasks as the severity and impact of accidentally deleting a task is a lot lower than that of a task list. This dialog was therefore also created when deleting a board in the final design as this has even more impact than the deletion of a task list.



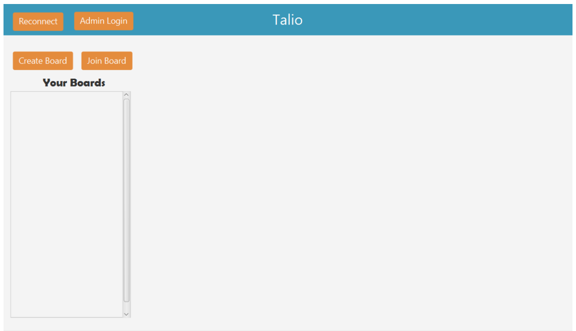
Are you sure you want to delete this Board?



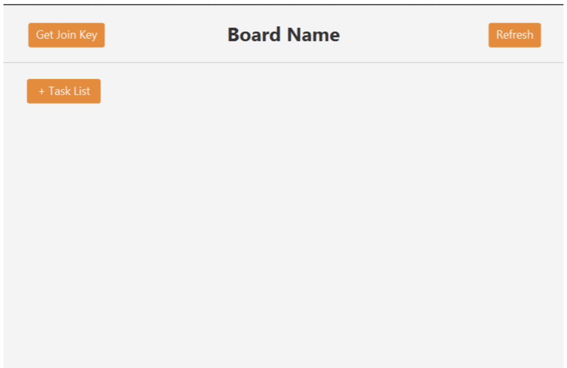
Are you sure you want to delete this task list?

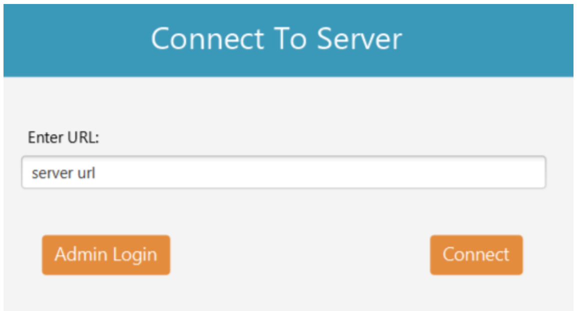
- 5: No ability to move tasks between lists (or within lists) - high severity (5 impact, 8 frequency), heuristic: user control and freedom, flexibility and efficiency of use.
  - This issue was solved by adding drag & drop functionality to the application. This increases user control and freedom and flexibility as they can order the tasks as they wish.
- 6: Visibility of the settings button - medium severity (6 impact, 6 frequency), heuristic: aesthetic and minimalist design, flexibility and efficiency of use.
  - This issue was not so much solved as worked around. What was decided is that the setting button would not be in the application and instead of it the application has multiple buttons that clearly state their purpose and are easier to see, increasing the efficiency and simplicity of the GUI.

[2] https://www.figma.com/file/PPKuepseSPsuFj7pd6QqnW/Feedback-Prototype?node-id=0-1&t=nULWAxOWUUZ1Nqvn-0



- 7: Missing exit button for the application - medium severity (8 impact, 4 frequency), heuristic: user control and freedom.
  - This issue was solved by JavaFX itself, as the scenes don't have a button for exiting the application, but when running the client it automatically creates those. This is also why they can't be seen in the above pictures.
- 9: Missing board title and server display on the board - medium severity (6 impact, 5 frequency), heuristic: visibility of system status.
  - This issue was partially solved. When a board gets added to the list in the above picture, the name gets shown and you can click on the board to go into it resulting in the below picture. In this picture the board name automatically gets set to the correct name. The server display was not added but there is a way to see which server you are connected to by clicking on the reconnect button. This will take you to the 'connect to server' screen which holds the server URL you are connected to.





REFERENCES

[1] G. Cockton, A. Woolrych, L. Hall, and M. Hindmarch. 2004. Heuristic evaluation of user interfaces.. In *People and Computers XVII — Designing for Society*. Springer London. https://doi.org/10.1007/978-1-4471-3754-2_9

[2] J. Nielsen. 1994. Heuristic Evaluation.. In *Usability Inspection Methods*. John Wiley & Sons, 25–62.

[3] J. Nielsen and R. Molich. 1990. Heuristic Evaluation.. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. Association for Computing Machinery, 249–256.