

# OOP Project Report – Group 42

Pepijn van Egmond, Sarah van de Noort, Daniel de Dios Allegue, Yusuf Barış Köse,  
Costin Teodorescu, Iurii Poteluev

## 1 Introduction

The report's methods follow a different approach to traditional user testing. In the latter, the team carrying out the evaluation is usually reluctant to answer the questions of the evaluators, given that the problems with the interface must be found by using the interface itself. However, in this case, it is useful to answer the experimenter's questions, as we also want to evaluate the usability from users which don't necessarily lie within the domain of the application being evaluated.

The application being evaluated is a task manager application. Every user has the option to create or join a board, to be able to manage cards which contain several tasks. There are also further functionalities that allow customization of each board, and assigning tags to certain cards to allow the users to categorize cards within the same domain.

## 2 Prototype

Given the current state of the application development, the prototype being evaluated does not involve a fully functional model. However, the development process is in an iterative state which allows the team of evaluators to assess a set of designs and transitions which are very similar to the expected final state of the product.

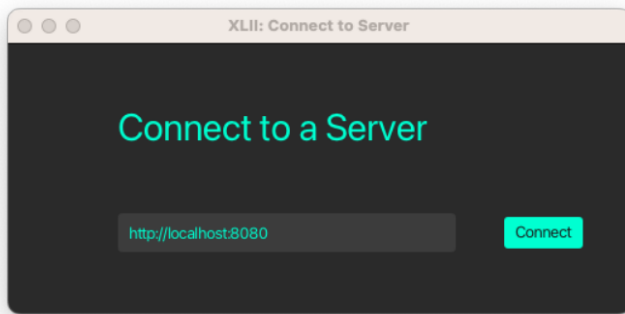


Figure 1

The prototype evaluated is a static model of our application. It represents a series of pages which allow the user to perform the required operations. The first page the user will encounter is the one represented in figure 1, which allows him to connect to the server. Once the connection is established with the server, the user will be connected to the primary page in the application, which shows all the lists which are accessible in that particular board (figure 2).

From this page, all the board related operations are easily accessible, such as adding a card or creating a new list inside a specific board (figure 3). It is important to note that the prototype described in the process is not fully functional, and does not include some features which are part of the final product. However, the prototype

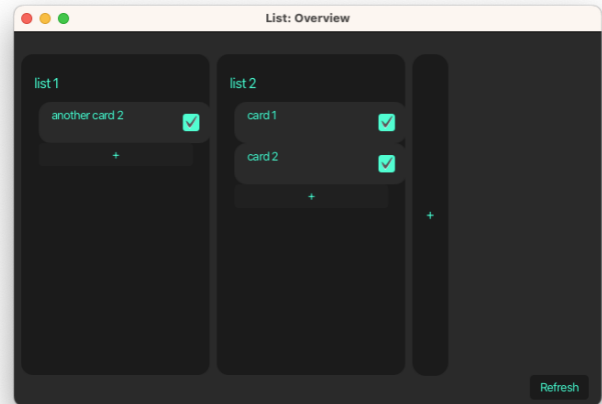


Figure 2

is sufficient to evaluate the core operations which can be accessed by the user in the final solution.

## 3 The Team of Evaluators

In most cases, an heuristic usability evaluation can be carried out by a single expert. However it was found by Nielsen that the most effective number of evaluators falls between 4 to 8 [1]. It was concluded that fewer evaluators would usually fail to find all the problems with the prototype, and that the results found by a larger number of evaluators was similar to the results found by 3 to 6 evaluators in most cases. This number selection also takes into account the cost-benefit analysis. In this case, the report accounts for the results found by a team of 5 evaluators.

During this evaluation, our team of experts consists of 5 computer science students attending TU Delft. They are accomplished in object-orientated programming, algorithms and data structures, which makes them an appropriate set of evaluators for our application. Furthermore, they have interests in software and application development, so they can also rely on the context of the developers. Finally, their experience in collaborative software projects demonstrates the knowledge required to develop a user-friendly application, and therefore evaluate our heuristic usability to a high standard.

## 4 Method of Evaluation

The team of evaluators was instructed to assess the prototype at issue of the set of heuristics mentioned in the next section. Once they were given the instructions, they were given a defined period of time of 1 hour to inspect the prototype individually. During this period, the evaluator experimented with the elements and transitioned through the interface several times using the set of

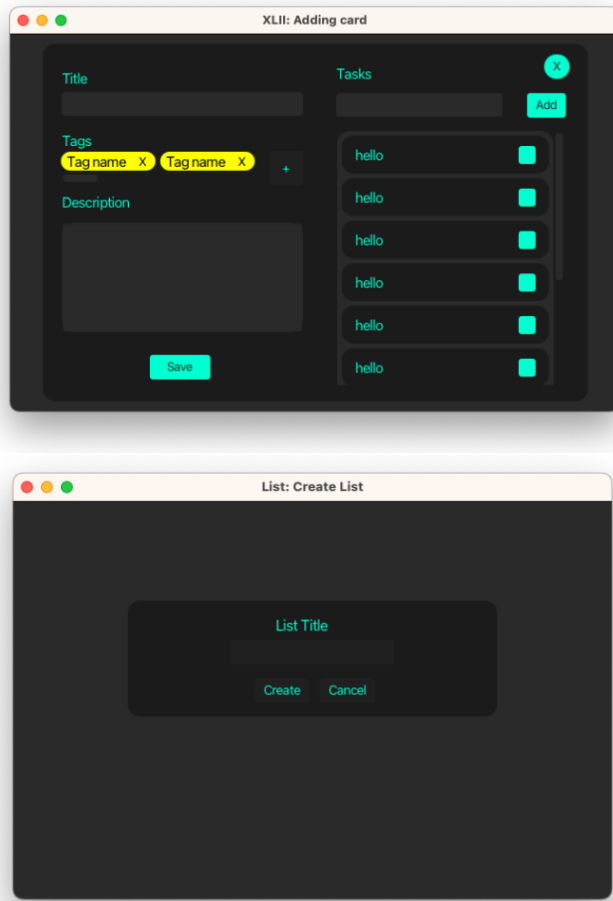


Figure 3

designs and descriptions within the prototype. Given the nature of the report, the experimenters were aided with a domain-specific assistant in case they faced a problem which blocked the state of the application during the evaluation.

In addition, the evaluators were instructed to perform a typical usage scenario relevant to the application context. In this case, they were given a set of actions to aid them to add a new card to a list, and then update the name of the newly created card. By performing this operation, the experimenter was forced to explicitly use the interface at the expected performance of the final application.

While the evaluators worked through the requirements, they faced a series of issues which deteriorated their experience. When one of these problems arised during the interface analysis, they were instructed to note them done if they thought they were relevant to the usability heuristics given. Furthermore, they were asked to use the format used in the table in figure 4, being as specific and descriptive as possible.

Description of problem found	Consequences for the user facing this problem	In which state of the application did the problem arise? (For example, in the board overview page)	What might be the cause of the problem?
Problem #1	...	...	...
Problem #2	...	...	...

Figure 4

## 5 Heuristics being evaluated

These statements corresponds to the set of heuristics described by Jakob Nielsen [2] article published by the Nielsen Norman Group in 2020.

### 1. Visibility of System Status

This includes providing information on the navigated route, indicating the user's current location, and ensuring they are aware of the server connection status.

### 2. Match with the real world

This design principle addresses the possibility that the user might be unfamiliar with the context of the application. In this case, the designers of the application should never assume prior-knowledge from the user, and always use familiar terms for the communication with the user.

### 3. User control and freedom

When this heuristic is covered, the user is always able to undo or redo his actions in a clear and concise way. The navigational awareness the application provides should not be broken by unexpected errors or dead ends, as this can lead to user frustration and confusion.

### 4. Consistency and standards

The whole application should have a similar look, and use the same terminology for similar operations. It should also follow external consistency by using industry-conventions. So that button placements and design decisions will not confuse the users.

### 5. Aesthetic and minimal design

The application should only include the necessary information for the user to navigate through it, and use it efficiently. The user should never be distracted by secondary elements.

### 6. Help and documentation

The documentation available to the user should be clear, concise and action-based. The application should ensure the documentation is easily reachable by the user at each state.

## 7. Error prevention, recognition and recovery

Including descriptive error messages is very important, as they allow the user to receive context of their actions. Furthermore, it should be simple to recover from these errors, and the user should never be blocked in a certain state after receiving an error message.

## 8. Recognition rather than recall

The user should not rely on the use of their memory to be able to navigate and operate the application.

## 9. Flexibility and ease of use

The application should be designed with efficiency and speed in mind, allowing the user to complete tasks quickly and with minimal effort.

## 6 Identification of problems

During the evaluation of the prototype, the experimenters were able to recognize a set of 6 concurrent problems. These issues will be discussed in this section:

**1. There is no clear way to search for a board, or to join a board with someone else's invite key:** This issue corresponds to the visibility of system status usability principle. In the current iteration of the product design, after the user connects to a server, the home page is loaded, where all the boards available are displayed.

**2. Unconventional use of layout and in-app terminology:** This issue corresponds to the consistency and standards principle. The placement of the cancel and add buttons within the layout feel unconventional. Furthermore the use of non-conventional terminology such as "Board" and "Card" can confuse users who are not familiar with the application.

**3. All board options are hidden to the edge, within the drop-down menu:** This issue relates to the principle "flexibility and ease of use". This design choice may require users to search for the options and put in more effort to accomplish basic tasks. This issue persists in the list overview where the option to delete a board is very subtly hinted with a drop-down icon.

**4. The password-protected term might not be descriptive enough within the population:** This issue corresponds to the usability principle "match with the real world". As the application's target user is practically the whole population, the designers need to make the assumption that the user might not be familiar with the term 'password-protected board', which might cause confusion within the non-technical users.

**5. The add list button is located between the list of boards and existing lists:** This issue concerns the "minimalistic and aesthetic design" heuristic. The design of this particular view might often seem too loaded, as there are several buttons together with different functionalities. As there is no clear distinction between both elements in the interface, it can often be confusing for users who may have trouble distinguishing this button as the one that adds lists.

**6. There is no clear access from a specific board, to another existing board:** This problem assimilates with the "recognition rather than recall" heuristic. In the current iteration of the design, the user needs to remember all his previous steps to access a new board in the home page. For some users this might lead them to a blocked state of the application, which severely deteriorates their experience.

Overall, the application needs to address these usability issues to improve the user experience and make it more user-friendly. This can be done by improving consistency and standards in the layout and terminology, providing clear feedback, enhancing flexibility and ease of use, and improving clear affordances.

## 7 Analysis and Improvements

The current design of the user interface is consistent with some of the usability principles described. However, in most areas there are necessary changes to improve the user's experience while using the interface. The most concurrent problem the results show is that there is a missing continuous feedback loop; as the user should be communicated the state after each action. This faces a set of situations in which the user might be in a particular state, and without any communication he/she needs to rely too much on recall, rather than being able to easily recognise the next step. Other results show that sometimes the interface has too many elements, and in consequence, some of these elements need to be partially hidden at first sight for the user. By adhering to the "minimalistic and aesthetic design" principle, the user should be offered with a clean design in which all the available options are easily reachable within each state.

To solve these issues, we have a designated plan to update the user interface which adheres better to the heuristic usability principles, in order of priority:

**1. Adding a quick access to the home page in every state of the application:** This improvement expands the freedom of the user within the application. Some scenes are reached through a series of states, which without this quick access element, might need to be remembered by the user to get back to the home page. Therefore, this change is the most crucial in terms of usability. Due to its weight, every scene in our application which requires it will have an additional quick access element to the home page. By creating a quick access element, the user does not have to remember his steps to access another board anymore. Therefore, this change improves the similarity of the interface with several heuristics such as "recognition rather than recall", "visibility of status" and "user freedom".

**2. Showing all the available operations/transitions at first glance:** The experimenters had issues finding the operations for a given board, given they were hidden in a drop down menu. Therefore, all of these operations should be visible in the design at all moments. This improvement enhances the visibility of status of the user, as well as progresses the design towards a more aesthetic and minimalistic product. This update can be considered one of the most important, as exposing all the operation possibilities at once (without having the need for the user to look for them), will also have a decisive impact in terms of usability. Due to its major

influence, all drop-down menus will be removed and all operations will be exposed clearly.

**3. Clearing up the interface view and adding more descriptive elements:** It was pointed out by the evaluators that some of the sections seemed too complex, such as the board view. In this case, the user might feel overwhelmed with the options, and confused by the similarity with some of those elements. In our new designs, we have taken into account the possibility of non-technical users, therefore, it is pivotal to include descriptive and clear concepts, to avoid confusion within this user base. In the next iteration step of our product, all elements which contain a description will now include a explicit interpretation of their function. Therefore, some sections of the UI (Board view and Edit Card view) need to be changed to reduce the number of elements present, as well as including more descriptive labels for the buttons. This will improve the user experience by focusing on the “match with the real world” and “aesthetic and minimalistic design” heuristics.

**4. Offering help in every state of the application:** This involves focusing on the “help and documentation principle”. Although no problem specifically about help and documentation was pointed out by the evaluators, adding a quick access to the documentation or help in every view reduces the possibility of user confusion. In this case, the term “password-protected” could be defined and consistency within the “Boards” and “Cards” will be maximized. Although not as key as others, given some users might not lie within the technical context of the application, logging the state in every scene of the application to the user can maximize efficiency of use. Therefore, some of the base scenes of the application will now include a help icon, which loads a concise help guide.

During the next weeks of development, the team will work towards a product iteration which is more comprehensive with the described heuristic usability principles by applying the improvements exposed in this section.

## 8 Conclusion

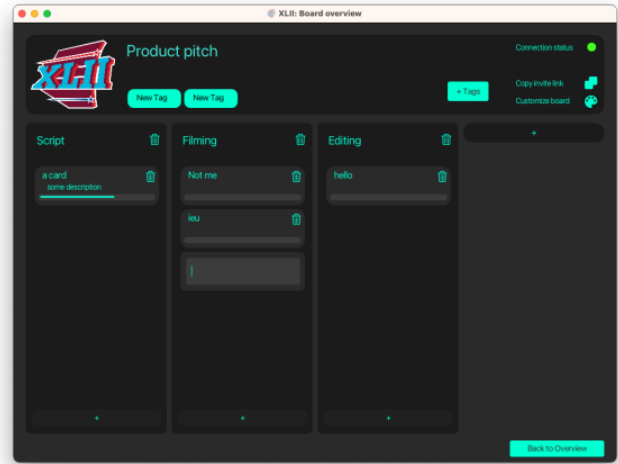


Figure 6

According to the analysis described in the "Analysis and Improvements" section, there are some changes which could be applied to the interface in order to improve usability and user experience. Therefore, we have implemented the updates as mentioned in that section, starting with the ones showed in figure 5 and 6.

This new interface now solves the 2 most important problems the experimenters exposed. The first image represents the home page, where each board can be accessed. Now, all the operations are exposed at first glance. When the user opens a board, he will be redirected to the second image (figure 6). This image represents a board. Being one of the most important scenes, it now includes a quick access button back to the home page.

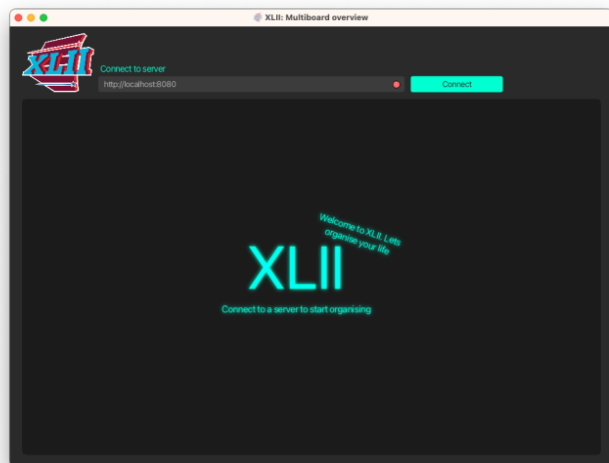


Figure 5

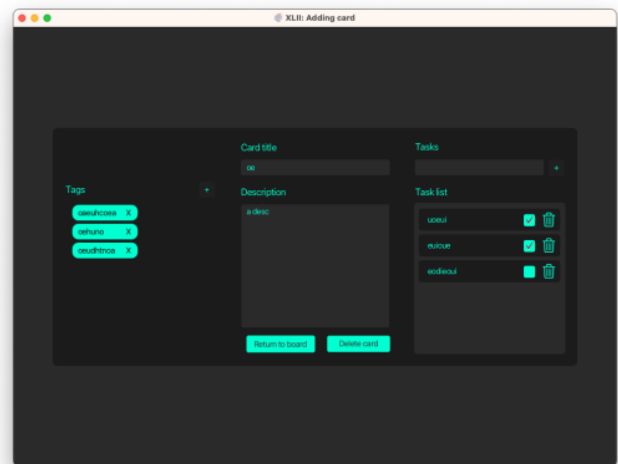


Figure 7

In this new scene of the updated interface, the user now has very descriptive elements in the UI. By making this change, non-technical users are now able to use the application as consistent and familiar language is used in every scene.

To conclude, the final user interface should have the properties described. Therefore, it is important to show an access to a documentation page in every state of the application, and in the Board Overview page there should be a quick access element to the home page, allowing the user to quickly return to the starting point. Furthermore, the terms used in the application such as “Board”,

“Card” and “Password-Protected” should be briefly described in the documentation, to minimize the confusion in users. Finally, all options to edit/delete or create within boards or lists should be displayed and not hidden below a drop-down menu, and buttons which carry out operations should be as descriptive as possible; while maintaining brevity.

## 9 References

- [1] Nielsen, How to Conduct a Heuristic Evaluation (1994) [2]  
Nielsen, 10 Usability Heuristics for User Interface Design (2020)