

CSE 1105 – Object Oriented **Programming Project**

Heuristics Usability Evaluation **Report**

by Group 23 – Serban Radu, Neel Lodha, Bogdan Paramon, Vasil
Chirov, Suruo Zhang and Filip Cajagi

INTRODUCTION

Graphical User Interface (GUI) is a crucial aspect of Computer Science, as it forms the bridge between the user and the system. It represents the way in which the user interacts with the program, and it encompasses all aspects of design, including layout, graphics, text, fonts, styles and other visible elements that exist with the objective of creating a portal to the application's functionality and main use.

The purpose of our evaluation report is gaining a deeper insight into the user experience, usability and overall design of our application, in hope of understanding the criteria that has to be met in order to make a step in the right direction in our development going forward.

METHODS

- **EXPERTS**

We recruited a team of 6 fellow students that are currently enrolled in this course, so take into the account that this report may be affected by their somewhat biased perspective, having worked on the same project and having knowledge regarding not only the requirements of the application, but on the heuristics evaluation process as well.

They have the experience of an above-average user, considering the fact that the probability of them having used task-managing applications before is very high.

- **PROCEDURE**

We had a brief discussion with the expert team, about the instructions of the evaluation process, as it has been showcased to us during the lecture and in the resources present in lecture material (e.g. the Nielsen-Norman Group - [Heuristic Evaluation Articles, Videos, Reports, Training Courses, and Online Seminars by NN/g \(nngroup.com\)](https://nngroup.com))

As advertised also in the aforementioned documentation, the experts have been asked to focus on the following heuristics:

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

- **MEASURES (Data Collection)**

The measurement is for the overall user experience with the application and especially the GUI, as we strive for the ease of use in our development. Experts have been asked to try out different screens, functions and maneuver

around our application, and then report issues found along the way, as well as the corresponding heuristic.

● RESULTS

The following problems have been pointed out:

1. Purpose of the search bar unclear - Heuristic 7 - **2**

Suggestion : Placeholder text can be updated to offer the user a clearer idea of what the input should look like, while still keeping the UI clean and minimal.

2. Scrolling necessary for 5+ boards - Heuristic 8 - **2**

Suggestion : The board list can be slightly increased in size, or a search box can be implemented, to allow the user to browse through boards flawlessly.

3. Cannot easily see where drag and drop lands - Heuristic 7 - **2**

Suggestion : A placeholder of the Task can be placed between the existing Tasks of a TaskList while hovering the TaskList to offer the user an idea of where the Task will land when dropped.

4. Unclear if the n-th list has been added - Heuristic 1 - **1**

Suggestion : Add auto-scroll functionality to the list of TaskLists, to give visual feedback to the user when adding a new list.

5. When a board is created, the board is not opened instantly, but instead the board's name is added to a list of boards, which makes the user search for the board after they have created it - Heuristic 1 - **1**

Suggestion: When a board is created, it must open automatically.

6. The height of cards is unnecessarily big, since they only hold a title. That makes it difficult for the user to look at more than 4 cards at a time, because they have to scroll - Heuristic 7 - **1**

Suggestion: Make cards shorter.

7. The login screen is too small and it is hard to distinguish the small text on it - Heuristic 9 - **1**

Suggestion: Make the login screen, as well as other screens bigger and ideally scalable and responsive.

9. The error message shown when failing to connect to a server is not specific enough - Heuristic 8 - **2**

Suggestion: Have a different error message appear depending on whether the server is closed or the address is invalid.

● **CONCLUSIONS & IMPROVEMENTS**

While it was already obvious that the application would be far from satisfactory as it is only at prototype status, this evaluation has pointed out many problems that have to be dealt with as soon as possible in order to make the user experience better, the design more enjoyable and the text-displaying more legible.

Our application should provide more help regarding the result of any action done by the user, as pointed out in problems 1, 3 and 4. The intuition of the user should not be a thing that the developers rely on consistently, so we will try switching to an approach that provides visible updates on actions more often.

Also, error handling is a very important aspect in the development of any application, and that is something that requires more work as pointed out also in problems 9. This can be achieved by:

- proper testing of all methods

- more trial and error sessions, testing our application in unexpected situations
- more clear error messages - interpreting and dissecting the error for the user etc.
- diagnosing network / connection errors

Problems in readability and sizing issues are also apparently common, and are (as expected) linked to each other, as showcased by problems 2, 6 and 7. Sizing can be addressed via implementing full screen options or even making the window customizable while maintaining proportions automatically.

Combined, all of those improvements and reinforcements should make the application more enjoyable, more consistent and more useful overall, by avoiding the reliance on user intuition and paying more attention to each possible scenario that can happen during the use of the program.