OOP Heuristic Usability Evaluation Report – Group 22

Mikołaj Gazeel, Vlad Drăgutoiu
Matiss Berzins, Bogdan Micu
Madeline Sebastian, Michael da Rocha Frota Fokkens

ABSTRACT

This essay was written as a report concerning the Heuristic Usability Evaluation of the product "Unruly Guitar", conducted in independent runs by heuristic usability experts. The central question of the evaluation was how usable the product was for users of any level of technical knowledge. The evaluators were supervised by M. Gazeel and V. Drăgutoiu, as to ensure their reports were in accordance with several usability heuristics that they outlined. Individual findings were analysed and aggregated into a collective report by M. Berzins and B. Micu.

1 INTRODUCTION

1.1 Evaluation objective

The target group of the product consists of teams, from any field, that want to coordinate a collaborative effort. These people might not necessarily be tech-savvy and should preferably be able to use the product and all its core features without having to sift through a lengthy documentation or tutorial. As such, ease-of-use and an intuitive interface were highly prioritised by Group 22, the developers of "Unruly Guitar". The main goal of the evaluation was to amend any flaws that would compromise this by analysing the report and translating it to concrete goals that would be added to the backlog of the developers.

1.2 Prototype

The product in question consists mainly of a board overview screen, where users can organise tasks, represented as "cards", into lists.

Figure 1: The board overview screen.

To access a board, a user first has to log on. The user can enter a username and the server IP on this screen. They can also claim admin access, and provide the admin password. Admins have access to all boards currently on the server, as opposed to users, who can only access boards if they have the invite key.

The boards can be accessed in parallel by all members of the board, using the board key. Users can also create new boards.

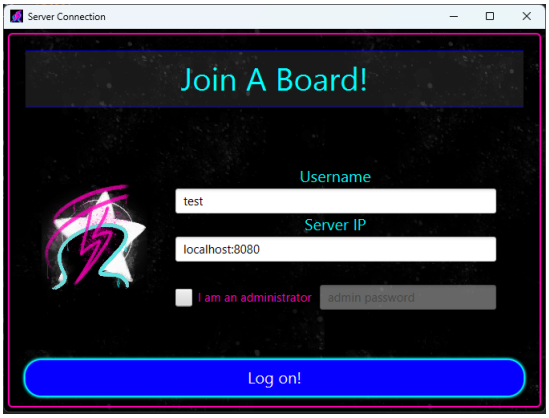

Figure 2: The logon screen.

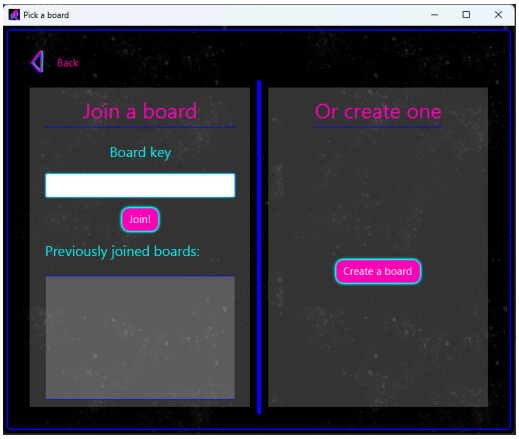

Figure 3: The boards screen.

In the board settings, a board can be renamed, and several customisation options exist to alter the appearance of a board. The board can also have tags, which then become accessible to all cards of the board. Users can also choose to leave or delete boards if they so wish.

When double-clicking on a specific card, a user can edit its title, tags, description, and subtasks.

The GUI design of the board seeks to emulate a blackboard, with a black background and a high-contrast, bright colour palette. The UI can also be interacted with without use of the mouse, as all elements are traverse-able using the TAB key. Generous highlighting and error pop-ups provide the user with instant visual feedback on their actions.

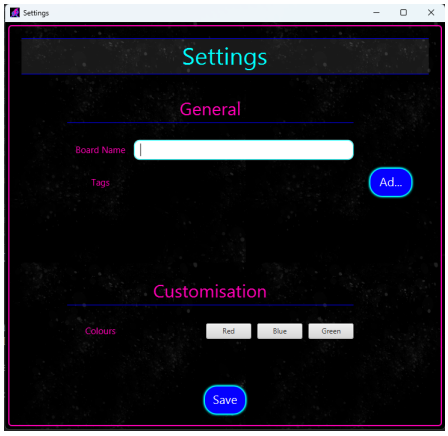

Figure 4: The settings screen.

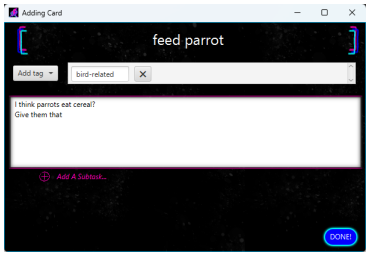

Figure 5: The card details screen.

2 METHODS

The Heuristic Usability Report itself was conducted in alignment with the method detailed in the article *How to Conduct A Heuristic Evaluation*, Nielsen et. al. [1]. In anticipation to the evaluation, Group 22, the creators of "Unruly Guitar", compiled a list of recognised usability principles [2], which served as the heuristics in question. These are:

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

Thereafter, three experts were asked to volunteer to evaluate the product, again in accordance with the article. The subject of the evaluation was the product itself, as opposed to a mock-up or a paper-only prototype, as the product at the time of the evaluation was deemed sufficiently functional to be evaluated. The output of the evaluation consisted of three lists, one per evaluator, which provided succinct explanations of how the product related to each separate heuristic, with special care given to violations of the heuristics. Then, they were aggregated and analysed.

2.1 Experts

The experts evaluating our application were:

- Dorian Herbiet
- Teun Bosch
- Alex Tabacaru
- Rares Iordan

The experts were members of a different group from our course, four of them evaluated our project, and provided us with feedback. These experts, due to also working on an application with a similar goal and utilizing the same frameworks, were well acquainted with the technologies the system relied on.

2.2 Procedure

In preparation for the evaluation, some setup was required. These consisted of the following:

1. Set up two servers running the application. Note down the IP address.
2. On one of the servers, add a board. Note down the board key.
3. Set up two client applications. Run them side-by-side on the same machine.
4. Set up a third server running an incompatible application. Note down the IP address.

Three evaluators were summoned shortly afterwards. They were provided with the machine running the client applications, as well as the IP addresses of all three servers and the board keys. Then, they were given a set of instructions.

The instructors of Group 22 provided the evaluators with the following instructions:

1. Connect to a server with a given IP address.
2. Try to join a board with a given board key.
3. Try to add a list.
4. Try to add tasks to the list.
5. Connect to a different server, with a different, provided IP address.
6. Join the same board on both clients and observe client synchronisation.

The three evaluators conducted the evaluations individually, but they were allowed to consult each other. They were also allowed to consult the instructors in accordance with *How to Conduct a Heuristics Evaluation*(1994). When prompted, the instructors provided the evaluators assistance in the form of verbal documentation on the GUI - no back-end implementation was exposed.

The evaluators were instructed to conduct three passes over the application. Each time, they were provided a different instruction set, mimicking three different common usage scenarios. The other two instruction sets consisted of the following:

Second pass-through:

1. Create a total of 5 boards. Give them all a distinct name.
2. Rejoin all of those boards sequentially. No board keys were provided by the instructors.
3. Close one of the client applications.
4. Join one of the newly created boards with the other client.
5. Close the other client application.
6. Run a new client application.

7. Join one of the newly created boards on the new client.

Third pass-through:

1. Join a board. It does not matter which one.
2. Add 3 lists to the board. Give each a distinct name.
3. Add 10 cards to the first list. The titles of the cards are provided. These are:
   3a. "Locate a parrot store."
   3b. "Travel to the parrot store."
   3c. "Buy a parrot."
   3d. "Buy bird feed."
   3e. "Put the parrot in a suitable birdcage."
   3f. "Research how large a birdcage should be."
   3g. "Care for the parrot."
   3h. "Love the parrot."
   3i. "Hold the parrot."
   3j. "Forfeit all material possessions to the parrot."
4. Organise the cards into lists. Change lists titles to suit if needed. Add or delete lists if needed.
5. Organise the cards within a list by reordering them. Changing card titles is not allowed.

During the evaluations, any problems or comments were diligently noted down by the instructors. No interpretation or editing was made to these.

2.3 Measures

The experts where each provided a form filled with a rephrased version of our heuristics. For each of these they were asked:

- Whether the heuristic was followed or violated, and to what degree.
- How they came to that conclusion.
- In case of a violation, a proposed solution.

These questions were formalised in a form. The completed forms, together with notes from the instructors, were used as the raw data for the report.

3 RESULTS

Following the meeting with the other group for the Heuristic Usability Evaluation, we have asked each of their members to fill out a form based of questions on different heuristics regarding our application prototype. All the filled-out forms have been considered while making the following list. The different problems have been categorized and ordered/prioritized in terms of severity – from urgently needed to "can wait".

3.1 System

- Multiple instances of error pop-ups, that the user might not know the origin and/or meaning of, would affect the "Match between system and the real world" heuristic.
  - 400 HTTP error during list creation with an Empty title field
  - DeploymentException error when entering an invalid hostname
  - Sometimes the app shows system status messages
- Connection to the server
  - The 8080 port shouldn't be entered by the user after the hostname
  - Connects with 'localhost' but not with 127.0.0.1

3.2 Client

- A couple of functionalities not working as intended, affecting "Error prevention":
  - Editing the Board name
  - Board settings page
  - Editing and deleting Lists
  - Editing and deleting Cards
  - Creating Cards
    - Can't add Sub-Tasks
    - Non-time values can be entered
    - Non-existent locations can be entered
- The following issues affect the "User control and Freedom" heuristic:
  - No cancel button when adding a card
  - It is not clear whether I'm logged in as an admin or not
  - Can log in with the same user on multiple instances on the same computer
  - No back button from Board Settings page
  - Not many keyboard shortcuts, only TAB and SHIFT + TAB
  - Login screen fields do not get cleared when leaving the page and coming back
  - A user should not see the full Board ID in the 'previous boards' list, but their names
  - Tags, location, and time aren't visible after adding a card
  - No character limit to the names and descriptions of Board, CardList and Card
  - Some documentation should be provided as it is hard for users to get around some features like, for ex., adding a card.

3.3 Visual

Visual issues affect the "Aesthetic and minimalist design" heuristic.

- Overall not satisfied with the visual aspect of the application:
  - Very specific theme and bright colors that give out a blurry feeling
  - Some buttons, like the back button on the Board View screen, are hard to spot
  - List of previous boards has a different theme than the rest of the app
  - Add List feature has a different theme than the rest of the app
- Resizing:
  - App switches between Fullscreen and windowed mode automatically often and unnecessarily
  - Sometimes when automatically full screened, the 'Press EXIT to get out of Fullscreen mode' persists and doesn't remove itself
- Recommended color coding:
  - Color-code adding and removing cards for minimal confusion
  - Color-code the color choices in Board Settings page

With these issues, we can create the following prioritization matrix based on their impact on the application and the frequency they appear in.

Impact	Frequency 1	2	3	4	5
1	Ability to add wrong values (time/place)		Theme discrepencies		Specific theme
2		Color Coding	Only board ID is visible from log in	Buttons that are hard to spot	
3	Admin visibility	Board Settings page	Error handling	App resizing	
4	Server connections	Cancelling card creation		Shortcuts	Board name editing
5			Subtasks unavailable		Editing cards and lists

Figure 6: Prioritization matrix

After thorough analysis, testing in a timely manner and, use of the previously mentioned matrix, we have concluded to opt for the following changes within our application:

4 CONCLUSIONS

Our main conclusions from the results were that the issues found could be summarised as a mix of lacking implementations, technical bugs, and exposing technical details to the user that could use a layer of indirection or translation from technical jargon to layman English. Lastly, some navigation elements were hard to spot or not traversable with shortcuts

4.1 Improvements

"Missing features and bugs were straightforward enough - they simply needed to be coded. Having the UI suggest features that did not exist and having bugs was extremely confusing and was not acceptable. Secondly, having error messages at all would be minimized in favour of using GUI to, for instance, highlight fields with invalid input in red or show tool-tips. Nielsen's 5th heuristic detailed this. In some places, error messages were necessary, for instance, when the error was network-related. In that case, the error messages would appear in custom windows with a style that was in line with the rest of the app and in plain English. Any unexpected errors simply showed as "Something went wrong. That's all we knew."" Secondly, all functionality and navigation was made to be accessible with keyboard shortcuts. A help screen with documentation was added, that could also be accessed with keyboard shortcuts. The last thing was a GUI overhaul - the design needed to be more aesthetically pleasant, and most crucially, clear, in line with Nielsen's 8th heuristic. "Back" buttons needed to be ubiquitous, large, and visible. Lastly, the team decided to also satisfy the second heuristic, "Match between the system and the real world." This will show in the final GUI design.

4.2 Final Design

Firstly, the navigation of the system is laid out in this illustration.

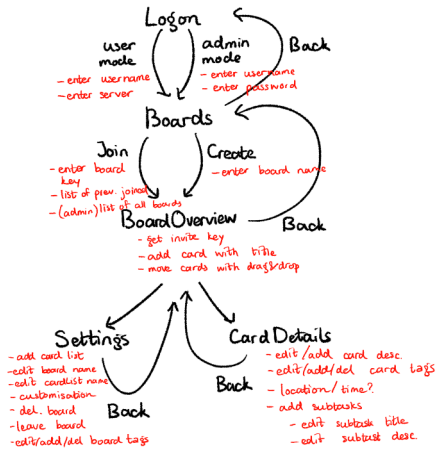

Figure 6: A "sitemap" of the application.

The new GUI is light mode, and has a lower contrast, less aggressive colour palette. It has one signature hue, which is green. This helps it keep a cohesive theme, while contrasting heavily with the original neon-coloured navigation buttons.

A series of sketches that show that illustrate the idea of the design will follow. This design might not be implemented in the final product.

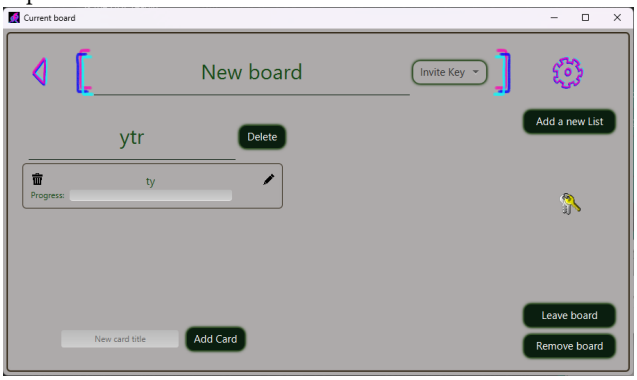

Figure 7: The board overview screen.

This is the main screen. It gives the user access to a lot of functionality, such as being able to copy the board key, setting a password for the board, leaving it, deleting it, and editing the board contents. However, due to the minimalist design, as recommended by Usability Heuristic #8, they are still not overwhelming, while still being easily visible. Using trashcan, pencil, gear and key icons satisfies the second heuristic with skeuomorphism.

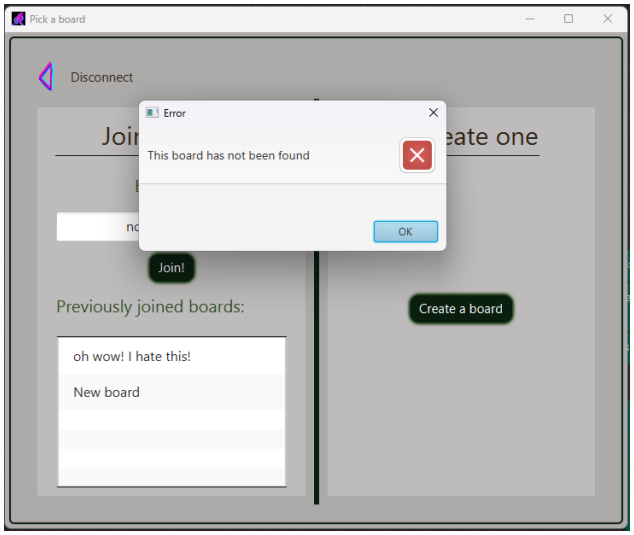

Figure 9: The boards screen.

This screen shows that the technical details of the application have been hidden from the user in favour of more easily understandable error messages. The board selection menu has also been made more user friendly by showing the board names instead of their keys.

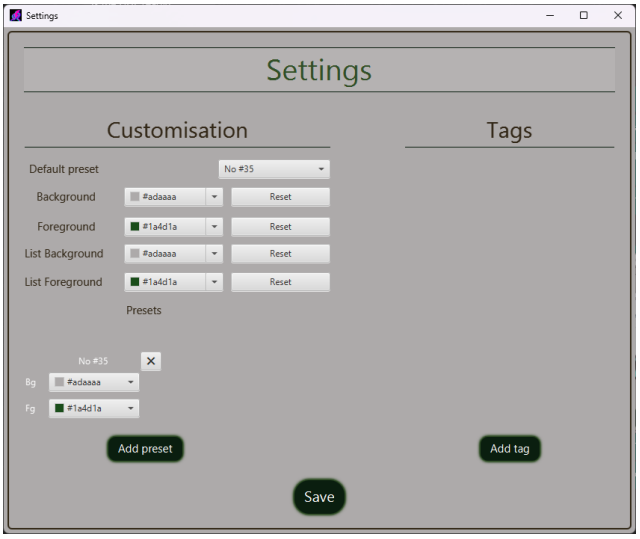

Figure 10: The card details screen.

In the settings screen, the user can specify their preferred appearance of the app. Instead of having the user enter hex-codes, the app shows a colour picker on click, and clear labels are used to communicate clearly what colour corresponds to which GUI element - background, foreground or otherwise. Furthermore, the user can enter tags here, which are also heavily customisable.

REFERENCES

[1] J. Norman et al. 1994. How to Conduct a Heuristic Evaluation. Nielsen Norman Group.
[2] J. Nielsen. 2020. 10 Usability Heuristics for User Interface Design. Nielsen Norman Group.