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 product, "Unruly Guitar", is, as of the time of writing, a Spring boot application prototype which uses Gradle and has a JavaFX GUI. It is being developed for academic purposes, by students at the Technical University of Delft. The product seeks to provide aid for the collaboration of scale-able teams, using join-able boards with multiple TODO lists, which include outlined tasks.

The central question of the evaluation was how usable the product was for users of any level of technical knowledge, as those users constitute the main target group of the product. The evaluators were low-level experts in the field, as they themselves had also been developing their own JavaFX application with the same purpose as "Unruly Guitar". Furthermore, they were also well-versed with the process of Heuristic Usability Evaluation, having studied it prior to the evaluation.

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 of 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".

Before the evaluation, Group 22 had dedicated a certain amount of effort towards achieving these goals. This effort was realised as a collection of user stories and epics that steered the team during development. These user stories would describe the intended user experience, so as to develop the product from the perspective of a user, keeping in mind the low level of entry regarding technical knowledge. However, specialised expertise regarding user usability was absent, and the product was not well polished yet, so flaws were expected to an extent. In addition, Group 22 had not reviewed the usability principles referenced earlier in this essay during development, only being made aware of them in anticipation of the evaluation.

The main goal of the evaluation was to amend these flaws, by analysing the report, and translating them to concrete goals that would be added to the backlog of the developers. Group 22 sees usability as integral to the product, as the target group of the product would be largely lost if were it not up to par.

1.2 Prototype

The product in question consists mainly of a board overview screen, where users can organise tasks, represented as "cards", into lists. These lists could be used as "To Do", "Doing" , "Done" or any other category the user found fitting, in the style of Trello [1], but to allow users flexibility, this is not necessary, and users can add, remove, and name their lists in any way they see fit. Users can also add tags, sub-tasks, descriptions, and other potential addenda not yet implemented at the time of writing to the cards. The boards can be accessed in parallel by all members of the board. Editing cards and organising them into lists constitutes the core functionality of the application.

On a more slightly higher level, the boards have a unique ID that users can use to join them, so whole teams can have access to the same board - in this sense, a board simulates a real-life whiteboard in a meeting room. A board can be renamed, and several customisation options exist to alter the appearance of a board. Furthermore, it can be optionally password protected, which bars any users that do not have access to the password from joining. Users can also choose to leave boards if they so wish.

Boards are hosted on servers, which can have admins, authenticated with admin passwords. An admin has the power to see all the boards hosted on a server, as opposed to a user, who can only see the boards they previously joined. An admin can also delete boards.

The UI design of the board seeks to emulate the idea of a blackboard, with a black background and a high-contrast and bright 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 "go back"-buttons provide the user with instant visual feedback on their actions.

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. [2]. In anticipation to the evaluation, Group 22, the creators of "Unruly Guitar", compiled a list of recognised usability principles [3], 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 by members of Group 22.

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 most of the inner workings of the system and had a general idea of how the system works.

2.2 Procedure

Two side-by-side instances of the client facing part of our system were prepared on one machine and presented to each expert. The experts were notified that there are three servers running, two of which were servers running our application whilst one of them was a server running an incompatible application, they were given the corresponding three IP addresses.

The experts were not instructed in any way shape or form as to how the application works or what they should expect, the only information they were provided was the aforementioned IP addresses.

The experts all followed a similar pattern of actions in order:

- Tried to connect to a server.
- Joined a board.
- Added a list.
- Tried to add tasks to that list.
- Tried to connect to the other server.
- Tried to connect with the other client to witness client synchronization.

The experts were using the heuristics mentioned at the beginning of section 2.

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 where the following is implemented or violated, how they have come upon this discovery, and/or how they would see a solution to improve on this in any way. This completed form, as sent in by 4 evaluators was our raw data for this 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 – the fastest needing fix being the top one.

3.1 System

- Multiple instances of error pop-ups, that the user might not know the origin and/or meaning of
  - 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:
  - Editing Board name
  - Board settings page not functional
  - 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
- Usability:
  - 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 return button from Board Settings page
  - Not many keyboard shortcuts, only TAB and SHIFT + TAB
  - Login screen fields do not get cleared
  - A user should not see the full Board ID in the 'previous boards' list, but their names
  - Tags, location, and time aren't visible when 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

- Overall not satisfied with 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

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

3.4 Solutions

1. Create more Issues that fix smaller problems
2. Finish mandatory issues to have all functionalities working
3. Fix visuals

4 CONCLUSIONS

Our main conclusion from these results is that the system largely fails on two fronts - Visibility of System Status and Error Recovery (principles #1 and #9, as detailed by Nielsen), with less pressing matters regarding the visual aspect. A number of interface elements that suggested certain functionalities that were at the time not yet implemented was also remarked.

Taking these points to heart, the main improvements whose necessity arise as evident are the following:

- Replace standard error messages with more user-friendly popups, or forgo popups entirely in favour of highlighting the error within the same screen.
- Clearly show the status of the user (admin or not).
- Add cancel buttons to the Add Card and Settings scenes.
- Use restrictions to prevent errors, such as a character limit on certain text fields, date-time and location input validation.
- Polish the visual aspect of the app, making sure the full-screen behaviour is not distracting, and ensuring visual consistency across the entire application.

Not all the issues found during the evaluation will be translated into planned Improvements. For instance, some of the issues were caused by bugs or features that were already planned to be implemented, so this does not change the course of development. Other issues were a matter of personal preference, such as the colour scheme - this could be changed, or the option to let a user customise the appearance of the application could be added, but the team does not see it as a priority.

After the improvements detailed in this section have been made, the product is expected to be much more easy to use for users that, for instance, might not know what an IP address or a hash code (the board key) is. The app is also expected to be significantly more forgiving to errors the user might cause. This is significantly more in line with the aforementioned Usability Heuristics.

REFERENCES

[1] [n.d.]. Trello - a well-known task organiser for collaborative work. here.
[2] J. Norman et al. 1994. How to Conduct a Heuristic Evaluation. Nielsen Norman Group.
[3] J. Nielsen. 2020. 10 Usability Heuristics for User Interface Design. Nielsen Norman Group.