Usability testing

Who – the client, co-worker(s) of client

Due to the client being the intended user of the solution it makes sense for them to test the usability of the solution. Being in the same field of the client their co-worker(s) would do similar work, if not the same providing more data and opinions.

When & duration – 13/7, roughly 1 hour per participant

The usability testing will be done after the functionality testing has been completed. This also follows the timeline of the project development. The large 1 hour timeframe is due to there only being one usability test and it needing to be broad. The large timeframe also allows for further statements and notes to be created.

Where – laptop

This closely recreates the same enviroment that the client would be using the solution on meaning that the program will essentially run identical. This means that there will not be any unexpectedly long loading times or processing times in the final solution when used by the client.

|  |  |  |  |
| --- | --- | --- | --- |
| Test item | Procedure | Time taken/score given | Comments  Red – user 1  Blue – user 2 Green – additional comments |
| Ease of logging into the program | Provide the user their username and password and instruct them to login | 5 seconds  5 seconds | Easy to understand  Quick and efficient  Users generally have no difficulty logging in however the simplicity provides an issue when security is brought up |
| Aesthetics of login page | Load login page and ask user for their thoughts on the aesthetics of the page on a scale of 1 to 5 | 3/5  1/5 | Does not need to be as big  Dull and boring |
| Aesthetics of main page | Load main page and ask user for their thoughts on the aesthetics of the page on a scale of 1 to 5 | 4/5  3/5 | Use the extra space I have by making things bigger  Very use to understand what things would do |
| Aesthetics of all other sub pages | Load all other pages and ask user for their thoughts on the aesthetics of the pages on a scale of 1 to 5 | 3/5  4/5 | Some have a lot empty space namely search  Mainly need to fill up empty space by removing it or using it |
| Readability of all UI | Ask user to provide their thoughts on the readability of the solutions UI on a scale of 1 to 5 | 5/5  5/5 | Easy to understand  Everything is clear and concise |
| Simplicity of navigation to different pages | Instruct user to perform navigation tasks such as switching classes or accessing different areas of the software. Ask user if they thought the system was satisfactory | Yes  Yes |  |
| Simplicity of exiting the program from login and main page | Load main page and instruct user to exit | Yes  Yes |  |
| Simplicity of closing any sub page | Load new page and instruct user to exit | Yes  Yes |  |
| Ease of creating a new class | Instruct user to create a new class with new parameters | Less than 45 seconds  Less than 40 seconds | Cursor is hard to read |
| Efficiency of the sort feature | Instruct the user to input the following dummy data:  Name: b,f,a,r Age: 14,21,51,22 Score: 32,55,100,75 Prompt user to sort each inputted column | 2 minutes  1 minute |  |
| Efficiency of the search feature | Instruct the user to isolate 22 from the age column using the search function | 20 seconds  15 seconds | Remove INT keyword from results  Would be nice if it could automatically search a highlighted item within the class |
| Attractiveness of the different themes | Instruct the user to switch themes to various themes and rate each theme’s attractiveness on a scale from 1 to 5 | 4/5  5/5 | All themes are great |
| Efficiency of the themes feature | List the time it takes to switch and apply themes | Less than 1 second  Less than 1 second | Due to the hardcoded method the themes are being applied by it might cost more CPU during long sessions |
| Efficiency of the stats feature | Instruct the user to fetch the stats of the score column and list the time it takes for the program to display them | Less than 30 seconds  Less than 30 seconds |  |
| Relevance of the stats displayed | Ask the user on their thoughts on the stats displayed and if they’re useful | Yes  Yes | Quite helpful  very relevant to my class if it would be used |
| Intuitiveness of adding data to a class | Instruct user to add the following data to their column Name: f,u,I,o and ask them to rate their ease of doing so on a scale from 1 to 5 | 5/5  5/5 |  |
| Intuitiveness of editing data in a class | Instruct user to edit the data inputted previously from f,u,I,o to w,q,e,r and ask them to rate their ease of doing so on a scale from 1 to 5 | 5/5  5/5 |  |
| Intuitiveness of deleting data from a class | Instruct user to delete the data previously inputted and ask them to rate their ease of doing so on a scale from 1 to 5 | 3/5  4/5 | Data can’t be deleted before the cell is double clicked which is quite time consuming  It would be cool if one could delete all highlighted items at once |
| Simplicity of editing a column name | Instruct user to edit their first column from Name to First Name | Less than 30 seconds  Less than 30 seconds | Make the label more clear |
| Interconnectedness seeing if user will save a class | Prompt user to save their created class | 10 seconds  10 seconds | Add hotkeys to menu strip items |
| Interconnectedness seeing if user will import a class | Prompt user to import their previously saved class and ask for their rating on the save/open system on a scale from 1 to 5 | 10 seconds  10 seconds | Quick and efficient |
| Tolerance of inputting wrong data in a class | Instruct user to input characters into an integer column | Yes  Yes | Difficult to break the program |
| Simplicity of deleting a class | Instruct user to delete a previously created class | 5 seconds  5 seconds |  |

List of modifications to be made:

Simple encryption system – due to the ease that a malicious user can find the correct username/password simply by digging through the system files it makes sense to implement a very basic encryption system. It will not a full encryption system using keys to encrypt and decrypt data but instead an easier method that reformats data within the code in an attempt to at least make it harder for a malicious user to gain full access to the solution

Hotkeys for menustrip items – the addition of hotkeys would increase the usability of the program, with the user no longer needing to click multiple times on the screen to perform an action, instead being able to use their keyboard which is much faster.

Optimize the themes feature – currently it is rather heavy CPU consuming which may reduce the performance of the solution over a long period of usage.

Automatically fill in the search input box with the selected item within a class – this wouldn’t do a lot; maybe a slight usability increase

Allow mass removal from a class by selecting multiple items – improve convenience

Allow user to remove data within a cell by backspacing it without needing to highlight the data within – this would make it a lot easier to remove items in a cells as currently the user has to double click a cell and then backspace it – a total of 3 clicks instead of 2 – this is unfortunately impossible within the vb limitations

UI changes:

* Change label on main screen from ‘select column’ to ‘select column to rename/delete’ for higher readability
* Change the background colour of textboxes so it is easier to see the cursor
* If hotkeys are added make it clear that they exist
* Reduce the size of the login screen as most of it is blank space
* Reduce the size of all other subforms due to all the unused area
* Change size and position of side UI on the left side of main screen to fill in unused area and make it look nicer overall
* Change text size of header on main form and move empty class area south
* Filter out INT keyword from search results