**Criteria 9 Evaluation Report**

**Introduction**

**Purpose of the software**

The solution aims to assist individuals with their studies by allowing users to save questions of their choice to the program and then later study them as they wish.

**Evaluation criteria**

Please refer to criteria 5’s final design to see the original evaluation criteria.

**Evaluation table**

|  |  |  |
| --- | --- | --- |
| Evaluation criteria | Strategy | Results |
| (FR01) User can add their own questions. | To evaluate this functional requirement the user will be observed whilst attempting to complete the functional requirement as well as writing down how the program behaves. The user is prompted to add a question to the solution and then any complications they come across are noted.  Question to measure efficiency:   * Can a question be added in less than 3 minutes?   Questions to measure effectiveness:   * Can a question be added without any difficulties? * Is the process intuitive and does not require extra time to learn the process of adding a question? | * For the short answer questions, the testers did not encounter any complications with creating them and were able to save them for future use. * However, complications arose when making multiple choice questions, ranging from not saving with the right data and to not being in the right format. * The solution met the criteria for efficiency as both users were able to create questions in less than 3 minutes. * The solution partially met the effectiveness criteria, as the process was intuitive and did not require further explanation, however some difficulties were experienced when creating the questions in the form of bugs. |
| (FR02) Multiple question formats (multiple choice, short answer) | The strategy employed was requesting the testers to attempt to create both a multiple choice and short answer question.  Question to measure effectiveness: - Can questions of all formats be added without any errors?  Question to measure efficiency: - Can the type of question be changed in less than 30 seconds? | * Results are similar to the previous criteria; the user was able to create short answer questions but came across bugs when making a multiple-choice question. * Thus, the effectiveness of this criteria was not achieved due to the inability to create multiple choice questions without errors. * However, the users were able to change the type of questions in less than 30 seconds, fulfilling the efficiency criteria. |
| (FR03) Adding tags to the questions | The feature of adding tags was tested with the users creating tags of their own choice and observing whether they were saved and able to be used later.  Questions to measure effectiveness: - Can the user add any tag they wish? - Can the user add any tag to any question?  Questions to measure efficiency: - Can the user make a tag in less than 2 minutes? - Can the user attach a tag to a question in less than a minute? | * Both testers experienced no difficulty in creating tags, and they were able to be used to be added to a question of their choice. * As such, the effectiveness criteria was met as the testers had no issues with creating a tag and adding it to a question. * Furthermore, the testers were able to complete both tasks referenced in the efficiency criteria within an amount of time below than the one outlined. |
| (FR04) Being able to sort the questions by tags | Although part of the original design, this feature was ultimately not implemented due to economical constraints preventing the time needed to implement such a feature. Instead, the program is able to sort the added questions alphabetically. | * Not applicable as the functionality that was to be tested was not implemented. |
| (FR05) A ‘score’ system to see which questions were struggled with the most | Testing this criterion would involve the user attempting to study the questions they have added and then stating that they had gotten their answer incorrect.  Question to measure effectiveness: - Does the application correctly keep track of the score on each question?  Question to measure efficiency: - Does the program make note of the score changes within a short amount of time? | * Both testers were not able to find success with this feature due to the error of the file being used by another process always occurring whenever they stated they got a question incorrect. * Hence, the criteria for both effectiveness and efficiency were not met. |
| (FR06) Buttons for saving and exporting | Both the saving questions and exporting the questions as a test were tested for this criterion. Any errors with this process was noted.  Questions to measure effectiveness: - Does the program save the question and retain it for future use? - Does the program export the questions in the correct format?  Questions to measure efficiency: - Can the program save and export questions in less than a minute? | * Test results showed that both functionalities were working, however some errors arose when attempting to save a multiple-choice question. These questions were not always saved properly – having incorrectly formatted data. * The criteria for effectiveness was partially met, as the program does not properly save multiple-choice questions, other than that the solution meets the criteria. * The solution does meet the efficiency criterion, being able to complete both functionalities in less than a minute. |
| (FR07) Customisation options for the feature of opening up on boot | The auto-boot feature was implemented within the solution, however there was insufficient time to implement customisation options for it. | * Not met due to the functionality not being implemented. |
| (FR08) A PIN as a security measure, the user can set up one and it will be used to gain access to the program on boot | The testers were asked to create a PIN of their choice and then attempt to log in with it.  Questions to measure effectiveness: - Does the program accept the PIN the user set up beforehand? - Does inputting the correct PIN gain access to the rest of the program?  Questions to measure efficiency: - Can the PIN be set up within 5 minutes? - Can logging in be done in a timely manner? | * The testers did not encounter any issues with creating a PIN and logging into the program with the PIN they had created. * The criterion for effectiveness was met as the program accepted the PIN the user had created beforehand as well as successfully gaining access to the rest of the program after doing so. * Both of the criteria for efficiency were also met with both the action of creating a PIN and logging in taking less than a minute. |
| (NR01) Easy to use | Questions were asked to the testers asking for how simple it was to use the program as well as observing how proficiently they were able to use the solution.  Question to measure effectiveness: - Can the program be used without further explanation?  Question to measure efficiency: - Can the user intuitively use the program within 30 minutes of first using it? | * The testers stated how the program was intuitive and further guidance was rarely needed, if not at all. * Hence, the solution was both effective in this field as well as efficient, as the testers were able to complete the usability test in less than 30 minutes. |
| (NR02) Efficient; the program works fast | This criterion concerns how fast the solution can perform each of its functionalities and aims to do so all within a timely manner. It was observed how long it took for each of the testers to complete their set tasks, as well as running one of the tests on a low-end device to ensure that it could still perform to a sufficient standard even when used on cheaper devices.  Question to measure effectiveness: - Do all the features work in the program without slow down?  Question to measure efficiency: - Can *all* the functionalities of the program be completed within 10 minutes? | * Even on the low-end device all of the functionalities were able to be completed within a timely fashion. Neither tester voiced any complaints in this regard. * To this end, both the criteria for effectiveness and efficiency were met as the solution prevailed with completing the tasks quickly without any major slowdowns. |
| (NR03) Robustness | The test for this criterion was carried throughout the usability test with any errors the testers encountered noted down.  Question to measure effectiveness: - Are there no errors encountered during use of the program?  Question to measure efficiency: - If an error occurs, can the program resume functioning after at least a minute? | * Throughout the usability test a number of errors were encountered for some of the functionalities, such as the score system or saving multiple choice questions. This means that the solution cannot be called robust. * The solution does not meet the criterion for effectiveness as errors were encountered, however the program was able to resume functioning afterwards hence fulfilling the criterion for efficiency. |
| (NR04) Consumes little resources | To measure whether the solution met this criterion the program was run on a low-end device to measure whether it consumes little resources so that even cheap devices are capable of using the application.  Question to measure effectiveness: - Does the program work in its entirety on a low-end device?  Question to measure efficiency: - Can a low-end device complete each of the functions in around 5 minutes? | * The test that was conducted on a low-end device faced no issues with conducting each of the tasks. The tester did not state any issues in this regard. * Hence, the solution was effective and efficient with each of the functionalities capable of being completed and each functionality being completed in less than 5 minutes. |
| (NR05) Readable UI | During the usability testing the testers were asked if they had any issues with interpreting the UI. They were also observed when asked to complete a certain task and seeing whether they were able to understand the UI to complete their task.  Question to measure effectiveness: - Does the UI explain everything sufficiently enough so that further explanation is not required?  Question to measure efficiency: - Can the user understand the program within 15 minutes of first using it? | * Even when the testers were asked to complete a task without much guidance, they were able to interpret the UI and complete the task without any challenges. * Hence, the criteria for effectiveness was met in that further explanation of the UI was not needed for the testers to complete their tasks. * They were also able to understand the program within 15 minutes of using it, thus being efficient. |
| (NR06) Attractive UI | During the usability tests the testers were asked how attractive they found the UI as well as some suggestions they had on how to improve the UI. | * Both responses outlined the lack of colour within the program’s UI, hence dampening its attractiveness – meaning that the criterion was not fully met. |

**Evaluation of Development Model Implemented**

The development model used for this project was a waterfall model. Given the small scope of the project, time constraints, and only one developer working on the solution it was decided that the simple and linear nature of the waterfall development model would best suit the project at hand. This development model allowed for the creation of an easy-to-understand Gantt chart that gave a guide as to where in development the solution should be at various stages of time. Due to only one developer working on the project, not much consideration was needed in terms of making the Gantt chart with multiple collaborators in mind. The strict nature of the development model ensured that the project remained on time and was not severely hampered by falling behind schedule, which the Gantt chart assisted in ensuring the project was completed. Despite the drawbacks of the development model, the waterfall model proved to be sufficient for the project due to its small scale, allowing for a prototype to be completed by the end of the development.

That said, elements from the other two development models may have proved beneficial for the project and it may have been a positive idea to implement a hybrid model. One of the drawbacks of the waterfall model is that client feedback is not weaved through the entire development cycle, only leaving room for it at the end. If a hybrid model including the Agile structure was used, then client feedback could have been implemented within the development cycle allowing for some points of feedback outlined in the evaluation to be dealt with earlier in development rather at the end where there is little time to implement the improvements. Although a Spiral model does have its benefits, it is best suited for large scale projects and would not be appropriate for this kind of project. Its complicated nature would only hamper the development of the solution.

**Conclusion of Evaluation**

Through the results in the evaluation criteria table, it can be concluded that the final solution fails to fully fulfill all of the specifications laid out in previous stages of development. Although the solution is able to meet at least most of the criterion to some degree, glaringly it does not meet certain criterion at all, such as FR07 which the feature outlined in this criterion was not implemented in the final solution. Hence, the prototype can only be labelled as such, an unfinished product that requires more development time to enable it to be a complete solution.

The conclusion above would suggest there may have been complications in either analysis, design, or development stage during development. Some of the requirements outlined during the analysis stage were unable to be implemented in the prototype of the solution, which was also seen during the creation of the evaluation criteria within the design stage of the problem-solving methodology. Hence, it could be concluded that the requirements created during the analysis stage were out of scope given the constraints of the project, and thus lies the problem within the analysis stage which inevitably affected the rest of the development. Although the evaluation criteria did have some entries that were unable to be fulfilled, a majority of them were met. Further, the solution design created during the design phase was successfully implemented – albeit with some alterations – and served to be a helpful guide for the development stage hinting that the design stage was mostly a success. However, the complications that arose during the development stage cannot be ignored. Heavily suffering from the economic and technical constraints of the project, the prototype ended up being unable to meet all of the evaluation criteria, pointing to some issues with being present with the development stage. Hence, it can be concluded that whilst the three stages helped in creating a prototype for the project, complications within the analysis and development stage led to the final creation not being able to meet standards previously established.

From the results obtained in the evaluation table, the following improvements required for the solution are listed below:

* Fix the bugs encountered with creating multiple-choice questions (including bugs surrounding data being formatted and saved incorrectly and not being able to edit them) so that the effectiveness portion of the criteria can be fulfilled as well.
* Implement the sorting by tags functionality as it was one of the established criteria that did not have its feature in the prototype of the solution.
* Fix the bugs surrounding the score feature as it currently is unable to fulfill the criteria as it faces a bug that prevents it from working from intended.
* Implement customisation capabilities of the auto-boot feature as it currently the prototype does not meet the criterion for it.
* Add some colour to the UI to improve its attractiveness.