**Criteria 7 – Kian Aristidou**

**Security**

**PIN security measure**

One of the functionalities implemented into the program to increase security was a PIN function, requiring the user to input the correct PIN before gaining access to the rest of the program, and consequently the questions they added which the user may wish to keep private. Upon using the program for the first time, the user will be prompted to create a PIN to secure the program. The PIN created is stored in a text file which is not encrypted, and it is read by the program when the user inputs a PIN to check to see if it matches. Ideally, the file storing the PIN would be encrypted to ensure further security however due to economic constraints as well as the lack of developer experience this was unable to be implemented.

**Project backups**

Over the course of development, the solution (and other files that were used to submit the criteria) was backed up onto Git Hub using the Visual Studio integration. After each day of working on the solution, a commit was made and pushed to Git Hub, thus acting as a backup for the entire project. Additionally, Git Hub allows one to view previous commits, so if one accidentally made an unwanted change in their latest commit, they would be able to look at the commit history and restore the old files. The backups were a form of cloud backup, and thus it was immune to device related problems such as the storage device breaking. Furthermore, the type of backup used allowed it to be used by any device, accounting for in case a device breaks down and another device needs to be used.

**Physical storage on two separate devices**

As the project was worked on with two devices using the Git integration on Visual Studio, it hence acted as another form of backup, instead of the cloud backup on Git Hub. As a form of physical backup, it is impervious to problems exclusive to cloud backups, such as the loss of internet or server issues. Storing on two different devices also increases the chances of successful backups, as in the case of one device breaking the data will be able to be retrieved from the other device.

**Data structures**

**CSV**  
To store the data used by the solution CSV files were used, for both the short answer and multiple-choice questions. Writing a CSV file can be done with ease, as all one needs to do when writing the data is use the entered data and add commas to it as a delimiter. Using a CSV allows for simple separation of the data entered as well as easy manipulation of the data, as all one needs to do within the code is split it by the comma values. Hence, CSV work well with arrays, which were used throughout the solution.

**One dimensional array**

These were used to store the data from the CSV files to be manipulated throughout the solution. The data from the CSV files were read and then added to the arrays in its entirety. Using an array allowed for retrieval of the data from the CSV files and then allowing for editing, reading, and other miscellaneous functions. Arrays were also optimal for storing a large quantity of data, which the solution has the potential of requiring. A one-dimensional array was sufficient enough to complete all of the required of the functionalities so it was decided that a two-dimensional array was not necessary.

**Lists**

This data structure was used to edit existing data in the arrays and then put it back into the array so the files could be rewritten. From the array that was created to store the CSV values it was then added to a list containing the values that need to be edited or manipulated. A list was also used for the search feature as it allowed for only adding the necessary data needed to perform a search, increasing efficiency and effectiveness. Lists were more effective than arrays here as it allowed for creating a new list with only the values needed for the functionality and all of the functionalities were capable of being carried out using this data structure.

**Global Functions**

The functions used throughout the program had some stored in the frmMain visual basic under a public module that enabled for those stored within this form to be used all throughout the solution even in other forms. These were functions that were integral to functions that are used across different forms, such as creating the arrays that are used to store the questions and the variables that store the values that will auto-fill the textboxes when editing a question. Housing these in a public module increased the efficiency of the code as rewriting lines of code that were used in multiple sections of the program did not need to be re-implemented.