**Examination System**

**[Ahmed Ahsan, 20K-0343 | BCS-2B]**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**0. Acknowledgment**

Most of the project has been covered under the programming concepts taught in the classes. Although, to increase the attraction, and efficiency at some points in the program, few built-in/ helper functions were used from the internet. No specific website has been favored in our searches- the one with the best idea has been adopted. These websites include stackoverflow, geeksforgeeks, and tutorialspoint. The ideas taken from internet are: Loading Bar and Typing Animations at the beginning of the program.

**1. Introduction**

As the name of the project briefs the function of the project, this project is about automating the Examination System .Primarily this program starts with a beautiful loading screen. Furthermore, a selection menu from which user could select if user’s role as an admin, Examiner, Student or Parent.

Admin can approve or decline a student and examiner being registered. This happens to be using temporary files which stores the details of the examiners and students who are to be accepted by admin. Furthermore it can see the details of examiner and student by accessing login files of examiner and student. Thirdly, admin can delete or update the questions of existing question paper. This is possible through reading the accessing test paper files created by examiner. Admin can also delete the existing student or examiner. Admin can access the login files of examiners (non temporary) from which details of the examiner and student can be deleted.

Using Examiner panel, examiner can login to its home page and perform all their activities which has been assigned to them like adding questions, set examination pattern. Moreover, examiner’s ids are automatically generated to prevent duplication of examiners. Examiners are been asked about their details including; qualification, name, age and CGPA which are later approved by the admin. Examiner can also choose their subject of interest.

Using Student panel, students could register and then login to perform their all activities, such as giving test. Students are prompted with a choice which test they want to attempt. Students are also assigned automated ids to prevent duplication. After the students has given the test, their result is automatically generated. Students can view their marks at the end of the test. Full result can be viewed from student’s panel too.

Parent’s panel is used to view the result and other details of a particular student.

**2. Tools and technologies used**

VS code editor and Dev c++ ide has been used.

Libraries used are:

<iostream> **=> for basic input output functions**

<fstream> => **for reading,appending and writing files**

<windows.h> => **system(“cls”);for clearing up the console screen, Sleep();for pausing the screen**

<string> => **for using string class**

<stdio.h> => **getch(); to get character from the user**

<vector> => **vector was used to store strings of subjects name string array was not used in this case because it is easy to delete items from vectors and vectors provide functions such as vector.erase(), vector.size() and vector.begin()**

**3. Programming Concepts Used**

**Filing**: for storing the details of examiners and students and the exam papers

**Abstract class** : Panel class is used as abstract class because its has the login and registering functionalities which is used by both Examiner and Students.

**Inheritance**: Panel class is inherited by Examiner and student.

**Loops**: used to write data in the files,used for input validation.

**Switch Cases**: used to provide a menu driven program

**Friend** **functions**: Admin function a friend function of both examiner and Student class to access private members of these classes.

**Structures** : examiner structure was made to make ease the use of variables of examiners

**Conditional Statements** : to swtich between the conditions program encounter

**Polymorphism**: Constructor overloading was performed and examiner’s copy constructor was made private to prevent any accidental copy.Futhermore two functions have been overridden; login and register.

**Abstraction** : Examiner and student classes have been made,which hides the implementation of the functions contained by the classes from the global program.

**Encapsulation**: Structures and Classes are made and variables are defined within them.

**4. Link to source**

https://drive.google.com/drive/folders/1Lpz5TmgUI9o2RCKPNLOnXZu62Si9Qj0L?usp=sharing

**5. Future work**

We could have implemented this project in gui.In addition, we could have made an online application for the program which could realistically present our goal. Although have tried the to add every other functionality which we could.

We had also had an idea of implementing the camera and AI system,which could automatically detect the unwanted movements of the person giving exams, to prevent cheating. Furthermore, databases can also be use instead of files to store data. Due to our limited learning we couldn’t go further.

Besides, we could’ve also created a fully-fledged website, which would take the exams and generate the results and upload results on the databases on their portal.