**Title; -** Online Quiz Application

**About the Project: -**

This project is about a Quiz Application created using Java Programming Language. In this Project, the user will have 10 Questions each having 4 options and 1 correct answer. This project contains an interactive GUI design for the user.

**Requirements: -**

PC with Eclipse IDE installed.

**How to use: -**

* The user starts from the Login Screen where they will be asked to enter their name for the Quiz.
* After entering their name, the user should click on the Play button to proceed or click on the Exit button to close the application.
* After clicking on Play, another screen will pop up showing the rules of the Quiz. Once the user has read and accepted the rules, they should click on the checkbox so that the Start button will be enabled.
* When the user is ready, they can click on the Start button to start the Quiz.
* After the Quiz starts, the user will see a Question with 4 options. The user will also see a timer of 15 seconds on the right side along with 3 buttons: Next, 50-50 and Submit.
* If the user answers the question correctly, they will receive one point for that question. If the question gives incorrect answer or the question is left unanswered, then the user will not get a point for that question.
* The next button will take the user to the next question. But keep in mind that after clicking on next, the user won’t be able to attempt the previous question again. If the user has answered the question, then it will be checked whether the answer is correct or not. Otherwise, the question will be marked as unanswered, and user will not receive any point.
* The 50-50 button gives user a lifeline. Once the user clicks the button, the 2 incorrect options among the 4 options will get disabled and the user can then choose from the remaining 2 options.
* The Submit button won’t be accessible until the user reaches the last question. Once the user reaches the last question, the next button will be disabled, and the user can submit their test. If the timer runs out here, then the test gets auto submitted and the final score will be calculated and displayed to the user.
* After checking the score, the user can click on Exit button to close the application.

**Components used for the Project: -**

For creating this project, I used the following Java components in my program:

1. javax.swing package: This package is the core package of this project since I used it to create the GUI for the whole project.
2. java.awt package: This package is important for styling of all the components in order to make GUI more interesting.
3. java.awt.event package: This package is important to control the usability of the JComponents like JButton, JRadioButton for this specific application.
4. JComponents: JComponents like JLabel, JButton, JTextField etc. were used to create the main GUI for the project.
5. ImageIcon class: The ImageIcon class plays an important role in adding images to the GUI.
6. Timer class: The Timer class is used in the Quiz GUI for displaying the timer information of the Quiz (In this case, to show that the user has 15 seconds to answer each question).

**Challenges: -**

During the creation of this application, I faced several challenges. It took me a lot of time to figure out a solution to all these challenges since this is my first time creating such applications.

* The first challenge that I faced was adding images to the GUI. I explored some solutions and tried using them to solve the problem, but to no avail. Then I tried one solution where I had to add the images folder to my project source (src) folder, then create a JLabel and set its icon using the setIcon() function and the ImageIcon class along with ClassLoader.getSystemResource() method to load the image in the JLabel.
* The next challenge that I faced was creating the timer of the Quiz. I tried using the Graphics class for creating the timer, but unfortunately, this put a lot of load on my application and I wasn’t able to use the application properly. To tackle this challenge, I came across a solution which uses the Timer class for creating the timer. After using this solution, my application started working smoothly.
* The next challenge was to develop the logical program behind all the buttons in the GUI. For some of them, it was easy. But for the ones in the Quiz, it was a little tricky and difficult. Eventually, I was able to successfully develop all the logical programming behind all the buttons and my application was finally ready.

I decided to use java swing for the main GUI instead of java AWT since it is less bulkier and has more components and features as compared to AWT. Furthermore, swing gave me the flexibility to add images to all my components whereas AWT is limited to a few components only.

I understand that the main objective of this project is to create an Online Quiz Application, but I didn’t have any prior experience in creating online applications using java. Hence, I decided to create a normal quiz application, and once I gain enough knowledge and experience in creating online applications using java, I will be able to use it for more such projects in the future.

Thank You for reading this document and I hope that you liked my project on Quiz Application.