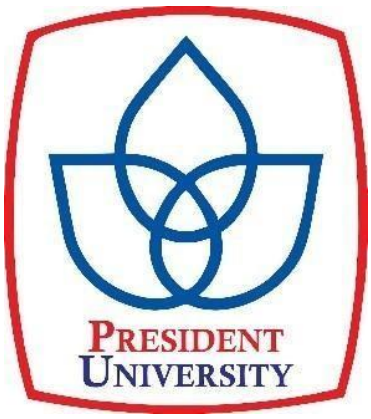


OBJECT ORIENTED VISUAL PROGRAMMING

Final Project Report

ONLINE EXAM SYSTEM

By:

Fajri Fathur Rahman (001202200064)

Ruspan Kurnadi Manurung (001202200166)

Ricky Setiawan (001202200068)

PRESIDENT UNIVERSITY

2023

Course Learning Outcome
CLO2: Able to create well-structured solutions by implementing appropriate programming methods in a variety of information systems cases that are related to Object Oriented Programming.
CLO3: Able to visualize information in order to deliver ideas in IT field

Grading Summary

CLO	Indicators	Grades
2	Student is able to create well-structured solutions using OOP approach	
	Student is able to implement OOP methods in a variety of information systems	
3	Students are able to visualize information in order to deliver project idea by implementing required GUI components in their project.	
Total		

Problem Statement

The educational landscape, characterized by online learning and remote work, faces a significant challenge regarding assessments. Students and educators alike encounter issues with traditional assessment methods, such as paper-based quizzes, which are logistically cumbersome, environmentally unsustainable, time-consuming to grade, and lacking in immediate feedback. Additionally, these methods fail to cater to the preferences of digitally inclined learners. The COVID-19 pandemic has further exacerbated these problems, as the swift transition to online platforms has revealed the limitations of existing tools designed for assessments, resulting in subpar user experiences, technical glitches, and security concerns. Students feel disconnected and unmotivated, while educators struggle to manage and grade online quizzes efficiently.

To address these challenges, we propose the development of a dedicated Online-Quiz Program, tailored specifically for assessments, to offer an enhanced experience for both students and educators. This solution comprises two primary components: an Admin Page for educators and a Student Page for learners. The Admin Page will empower educators to effortlessly create, manage, and grade quizzes. It will include features such as a versatile quiz builder with various question types, an automatic grading system for objective questions, and plagiarism detection tools. Real-time analytic will also be provided to track student progress and identify areas requiring additional support.

On the other hand, the Student Page will provide students with a seamless quiz-taking experience. It will feature a user-friendly interface, a timer to maintain time discipline, and a review option before final submission. Immediate automated feedback on objective questions will be provided, enriching the learning process for students.

The proposed Online-Quiz Program serves as a comprehensive solution, effectively addressing the identified problems. By providing a dedicated platform for online quizzes, it enhances student engagement through its user-friendly interface, immediate feedback, and robust anti-plagiarism measures. For educators, it streamlines quiz management and grading, freeing up time for more pedagogical tasks.

To implement this solution, a team comprising software developers, education technology experts, and UX/UI designers will be assembled. They will collaborate to design and develop the Online-Quiz Program, ensuring its usability, security, and effectiveness. Throughout the development process, iterative testing and feedback collection from potential users will be conducted to refine the system before its full-scale roll-out.

With the Online-Quiz Program, we envision an elevated teaching and learning experience, where assessments not only measure understanding but also serve as a tool for learning. As the future of education unfolds, it becomes crucial to harness technology to overcome challenges and enhance student engagement and learning outcomes.

1. The Problem

In today's era of online learning and remote work, both students and educators are facing a significant challenge regarding assessments. The traditional methods of conducting assessments, such as paper-based quizzes, have several drawbacks that hinder effective learning. These methods are logistically cumbersome, environmentally unsustainable, time-consuming to grade, and lack immediate feedback. Furthermore, they fail to cater to the needs and preferences of digitally inclined learners. The COVID-19 pandemic has further exacerbated these issues, as the sudden transition to online platforms revealed the limitations of existing assessment tools, resulting in subpar user experiences, technical glitches, and security concerns. Students feel disconnected and unmotivated, while educators struggle to efficiently manage and grade online quizzes.

Other than that, the fact that it is an easier, more effective and efficient way to assign quiz for lecturers or teachers with an online quiz management shows that there's more to it in technology to develop. By working on the earliest development for technologies shows improvement in human understanding for future development as well. Humans tend to seek comfort in this era for easy things, so the development of work to be more efficient and effective is a requirement for this era. The use of technologies that we have must be developed to be able to develop more.

2. Proposed Solution

To address these challenges, we propose the development of a dedicated Online-Quiz Program that is specifically designed to enhance the assessment experience for both students and educators. This program will consist of two main components: an Admin Page for educators and a Student Page for learners.

The Admin Page will provide educators with a user-friendly interface to effortlessly create, manage, and grade quizzes. It will incorporate various features such as a versatile quiz builder with different question types (multiple-choice, fill-in-the-blanks, essay, etc.), an automatic grading system for objective questions, and plagiarism detection tools. Additionally, the Admin Page will offer real-time analytic to track student progress and identify areas that require additional support.

On the other hand, the Student Page will ensure a seamless quiz-taking experience for learners. It will feature a user-friendly interface that is easy to navigate, a timer to

maintain time discipline during quizzes, and a review option before final submission. Immediate automated feedback on objective questions will be provided to enhance the learning process for students.

Online quiz system will show an amazing engagement for future development of our grading system. It could show a lot of improvement if we would always make new things from what we have now. This online quiz system shows a lot of result by managing lots of things for lecturers to grade students.

3. Why the Solution Fixes the Problem and How it Will be Implemented:

The proposed Online-Quiz Program serves as a comprehensive solution to address the identified problems. By providing a dedicated platform for online quizzes, it will enhance student engagement through its user-friendly interface, immediate feedback, and robust anti-plagiarism measures. This solution will also streamline quiz management and grading for educators, freeing up their time to focus on more pedagogical tasks.

To implement this solution, a team of software developers, education technology experts, and UX/UI designers will be assembled. They will collaborate to design and develop the Online-Quiz Program, ensuring its usability, security, and effectiveness. Throughout the development process, iterative testing and feedback collection from potential users will be conducted to refine the system before its full-scale roll-out.

With the implementation of the Online-Quiz Program, we envision an elevated teaching and learning experience, where assessments not only measure understanding but also serve as a tool for learning. It is imperative to leverage technology to overcome the challenges faced in the education landscape, enhance student engagement, and improve learning outcomes.

Encouraging and developing the use of technology could boost knowledge and human capability to its max reach so we could develop ourselves to a better tribute in future technologies.

Application Description

Online-Quiz Master is a dynamic application designed to revolutionize the way educators create, manage, and grade quizzes, and how students take them. For educators, it's a powerful tool that simplifies the process of quiz creation and management. It offers a range of question types and an automatic grading system for objective questions, making the assessment process more efficient.

For learners, Online-Quiz Master provides an engaging and interactive platform to take quizzes. It offers an intuitive interface, a review option before submission, and instant feedback on answers, enhancing the learning experience.

Online-Quiz Master stands out from other online assessment tools by offering a dedicated platform for both educators and students. It's not just an assessment tool; it's a comprehensive solution designed to improve the teaching and learning experience. It features robust anti-plagiarism measures and real-time analytic, ensuring a fair and effective assessment process.

If you're seeking an efficient, user-friendly, and secure solution for conducting online quizzes, Online-Quiz Master is the perfect choice. It's more than just an app; it's a platform that aims to transform the landscape of online assessments and contribute to the future of education. Experience the difference with Online-Quiz Master.

1. What does your app do?

With Online-Quiz Master, educators get the power to create, manage, and assess quizzes at their fingertips. The admin page provides a comprehensive suite of tools for teachers to add, update, delete, and view all questions with ease. It comes with a variety of question types and automatic grading functionality for objective questions.

For students, Online-Quiz Master offers a seamless and engaging quiz-taking experience. They can work on the quizzes created by their teachers, review their answers before submission, and receive immediate automated feedback, making learning interactive and fun.

2. What problem do you solve?

Online-Quiz Master addresses the pain points associated with traditional and existing online assessment methods. It eliminates the logistical hassles and environmental impact of paper-based quizzes, provides a dedicated and secure platform for online assessments, and enhances student engagement and learning outcomes. For teachers, it simplifies quiz management and grading, saving them valuable time.

3. What is unique about your app?

Unlike many other online assessment tools, Online-Quiz Master is designed with a dual focus on teachers and students. It offers a unique combination of a powerful admin page for educators and an intuitive student page for learners. Furthermore, our app incorporates robust anti-plagiarism measures, real-time analytic, and a user-friendly interface, making it a comprehensive solution for online quizzes.

4. Why should someone download your app rather than something else?

If you're looking for a seamless, efficient, and engaging way to conduct quizzes, Online-Quiz Master is the app for you. It's more than just an assessment tool; it's a platform designed to enhance the learning and teaching experience. With Online-Quiz Master, you get a dedicated, secure, and user-friendly platform for online quizzes. Join us on this journey to redefine assessments and shape the future of education.

Implementation of OOP

Encapsulation is like a protective wrapper that prevents the data from being accessed by the code outside this wrapper. Encapsulation means hiding the internal details or mechanics of how an object does something for security reasons. There's different types of access modifiers to operate the concept of inheritance and what is mainly used is the private and public method or variables. The public modifiers allow you to access methods or variables from different classes even though you didn't call the class identifier first and it kinda makes it hard to do some things. So that's why there's private modifiers to make it more efficient to use the existing variables or method so it would be called or set the value by mentioning modify the method or variables to unidentified variables or methods first and set it on other method on other class with "Set.." or "Get.." to call it.

Inheritance is like a child inheriting traits from their parents.While working on the main class of the java file, inheritance apply working with a new class and inhereting the same principles that the main class has to the child class. The parent/superclass could implement their functions or attributes to the child/subclass' method by implementing a specific syntax to the child class for it to inherit the parent class. Inheritance isn't as simple as that because we need to make a parent or the general class for the class to contain and refer to all the child class to work effective and efficient.

Polymorphism allows an object to take on many forms. It determines what kinda function to run while it's running. It works by running a pointer to the object (functions or properties) while overriding the main method. It allows objects of different classes to be treated as objects of a common superclass and enables code to be written that can work with objects of different types without needing to know their specific classes.

Abstraction is the concept of object-oriented programming that "shows" only essential attributes and "hides" unnecessary information to show to the user. While you create the class of the program, the object of each methods are written on the main class. So what is important is to call the method of the program to work as it should for the user.

1. In our code, encapsulation is implemented through:
    a) The use of private instance variables (`jButton1`, `jButton4`, `jButton5`, `jLabel1`) in index.java

```
132
133         // Variables declaration - do not modify
134         private javax.swing.JButton jButton1;
135         private javax.swing.JButton jButton4;
136         private javax.swing.JButton jButton5;
137         private javax.swing.JLabel jLabel1;
138         // End of variables declaration
139     }
```

b) The use of Public methods (`jButton1ActionPerformed`, `jButton4ActionPerformed`, `jButton5ActionPerformed`) in index.java.

These instance variables represent JButton objects, and the encapsulation is represented by the fact that these buttons' behaviors are defined within this class and not accessible outside it.

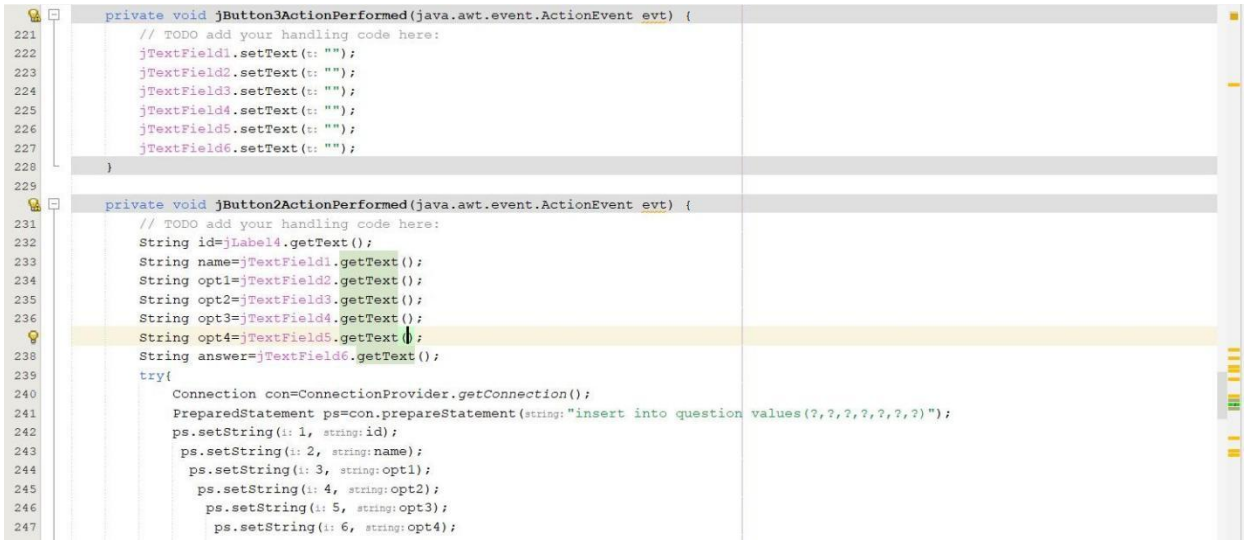
```
77      private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
78          // TODO add your handling code here:
79          setVisible(b: false);
80          new loginAdmin().setVisible(b: true);
81      }
82
83      private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {
84          // TODO add your handling code here:
85
86          int a=JOptionPane.showConfirmDialog(parentComponent: null,message: "Do you really want to Exit Apllication",title: "Select",optionType: JOptic
87          if(a==0)
88          {
89              System.exit(status: 0);
90          }
91      }
92
93      private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {
94          // TODO add your handling code here:
95          setVisible(b: false);
96          new studentDetails().setVisible(b: true);
97      }
```

c) The addNewQuestion class in addNewQuestion.java demonstrates the encapsulation principle of OOP through the use of public methods such as jButton1ActionPerformed, jButton2ActionPerformed, and jButton3ActionPerformed. These methods represent the behaviors associated with three JButton objects in the class. The encapsulation is represented by the fact that these buttons' behaviors are defined within the addNewQuestion class and are not directly accessible or modifiable outside it.

```
    private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
        adminHome.open=0;
        setVisible(b: false);
```

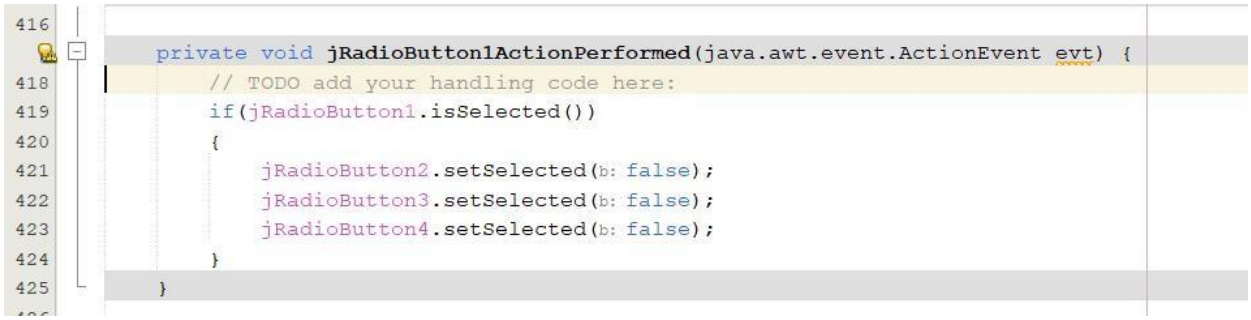
d) The jButton1ActionPerformed method, for example, is responsible for handling the action when the first button is clicked. In this scenario, it makes the current window disappear by setting its visibility to false. On the other hand, the jButton2ActionPerformed method is tasked with the process of adding a new question

to the database. This involves collecting the user's input from the text fields, establishing a database connection, and running an SQL command to insert the new question. Finally, the jButton3ActionPerformed method is designed to respond to the third button's click event by resetting all the text fields, effectively clearing the form for new input.



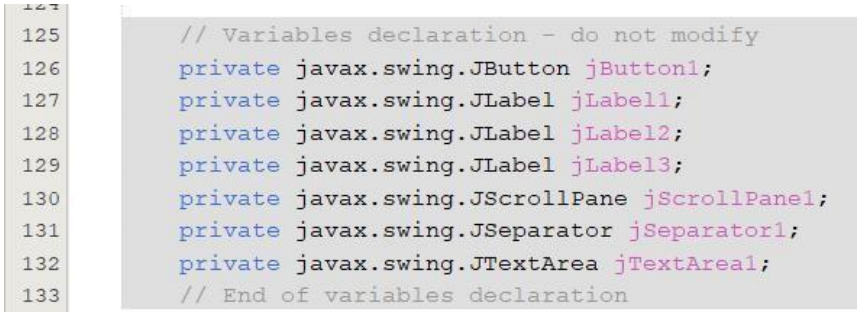
```
221     private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
222         // TODO add your handling code here:
223         jTextField1.setText(t: "");
224         jTextField2.setText(t: "");
225         jTextField3.setText(t: "");
226         jTextField4.setText(t: "");
227         jTextField5.setText(t: "");
228         jTextField6.setText(t: "");
229     }
230
231     private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
232         // TODO add your handling code here:
233         String id=jLabel4.getText();
234         String name=jTextField1.getText();
235         String opt1=jTextField2.getText();
236         String opt2=jTextField3.getText();
237         String opt3=jTextField4.getText();
238         String opt4=jTextField5.getText();
239         String answer=jTextField6.getText();
240         try{
241             Connection con=ConnectionProvider.getConnection();
242             PreparedStatement ps=con.prepareStatement(string:"insert into question values(?,?,?,?,?,?,?)");
243             ps.setString(i: 1, string:id);
244              ps.setString(i: 2, string:name);
245               ps.setString(i: 3, string:opt1);
246                ps.setString(i: 4, string:opt2);
247                 ps.setString(i: 5, string:opt3);
                     ps.setString(i: 6, string:opt4);
```

e) The use of private methods in quizExamStudent.java where its function would be modified in the main method to work in all 4 functions of the "if …setSelected" jRadioButton.



```
416
418     private void jRadioButton1ActionPerformed(java.awt.event.ActionEvent evt) {
419         // TODO add your handling code here:
420         if(jRadioButton1.isSelected())
421         {
422             jRadioButton2.setSelected(b: false);
423             jRadioButton3.setSelected(b: false);
424             jRadioButton4.setSelected(b: false);
425         }
    }
```

f) The use of private instance variables (`jButton1`, `jButton11`, `jButton12`, `jLabel13`, `jScrollPane1`, `jSeparator1`, `jTextArea1`) in instuctionStudent.java.

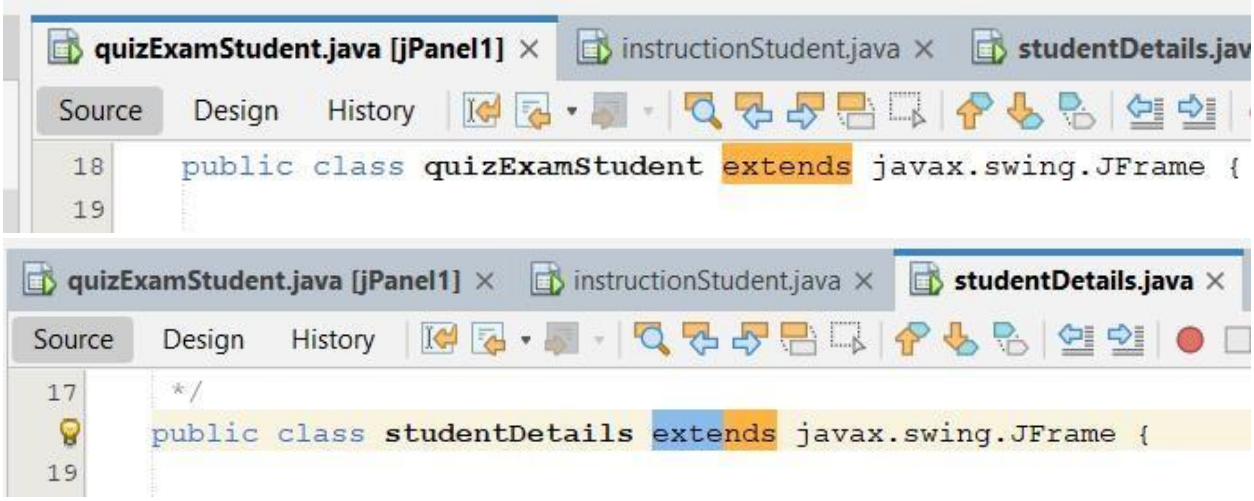


```
125     // Variables declaration - do not modify
126     private javax.swing.JButton jButton1;
127     private javax.swing.JLabel jLabel1;
128     private javax.swing.JLabel jLabel2;
129     private javax.swing.JLabel jLabel3;
130     private javax.swing.JScrollPane jScrollPane1;
131     private javax.swing.JSeparator jSeparator1;
132     private javax.swing.JTextArea jTextArea1;
133     // End of variables declaration
```

2. In our code, inheritance is implemented through:
   a) The `index` class in this code extends the `JFrame` class, meaning it inherits all the fields and methods from `JFrame` in index.java.

```
12      */
13     public class index extends javax.swing.JFrame {
14
```

Other than the `index` class, there are other classes in other files, like `quizExamStudent` class and `studentDetails` class.



   b) This getContentPane in index.java is a method from JFrame that returns the contentPane object, which is the container for the JFrame's components.

```
38      getContentPane().setLayout(new org.netbeans.lib.awtextra.AbsoluteLayout());
39
40      jButton4.setFont(new java.awt.Font("Segoe UI", 1, 14)); // NOI18N
41      jButton4.setIcon(new javax.swing.ImageIcon(getClass().getResource("/index student.png"))); // NOI18N
42      jButton4.setText("Student");
43      jButton4.addActionListener(new java.awt.event.ActionListener() {
45          public void actionPerformed(java.awt.event.ActionEvent evt) {
46              jButton4ActionPerformed(evt);
47          }
48      });
49      getContentPane().add(jButton4, new org.netbeans.lib.awtextra.AbsoluteConstraints(863, 21, -1, -1));
```

3. In our code, polymorphism is implemented through:
   a) The method `jButton1ActionPerformed` in index.java is an example of polymorphism. When the "Admin" button (`jButton1`) is clicked, a `loginAdmin` object is created and becomes visible.

```
77      private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
78          // TODO add your handling code here:
79          setVisible(false);
80          new loginAdmin().setVisible(true);
        }
```

If the button in index.java pressed was the "Student" button (`jButton4`), a `studentDetails` object would be created instead.

```
93      private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {
94          // TODO add your handling code here:
95          setVisible(b: false);
96          new studentDetails().setVisible(b: true);
        }
```

Both subclasses (loginAdmin and studentDetails) are treated as instances of their superclass, JFrame.

Other than loginAdmin and studentDetails' button function, other overriding in other java files. It all mostly functions on the exit button and back button.

b) In adminHome.java, the MouseEvent evt represents any mouse event (like clicking, dragging, etc.).

```
129     private void jMenu6MouseClicked(java.awt.event.MouseEvent evt) {
130         // TODO add your handling code here:
131         JFrame jf=new JFrame();
132         jf.setAlwaysOnTop(alwaysOnTop: true);
133         int a=JOptionPane.showConfirmDialog(parentComponent: jf, message: "Do you really want to Logout",title: "Select",optionType: JOptionPane.YES_
134         if(a==0)
135         {
136             setVisible(b: false);
137             new loginAdmin().setVisible(b: true);
138         }
139 
        }
```

4. In our code, abstraction is implemented through:

a) In index.java, the `index` class provides an abstraction for the GUI form. It provides a high-level interface (buttons for Admin, Student, Exit) for interacting with the system, without requiring the user to understand how the GUI form is created and managed. The ActionListener attached to the button is one of the abstraction examples in our code. The user only needs to know that when a button is clicked, the associated method will be called.

```
43          jButton4.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
45              jButton4ActionPerformed(evt);
46          }
47      });
```

b) In adminHome.java, the method performs opening a new question input form.

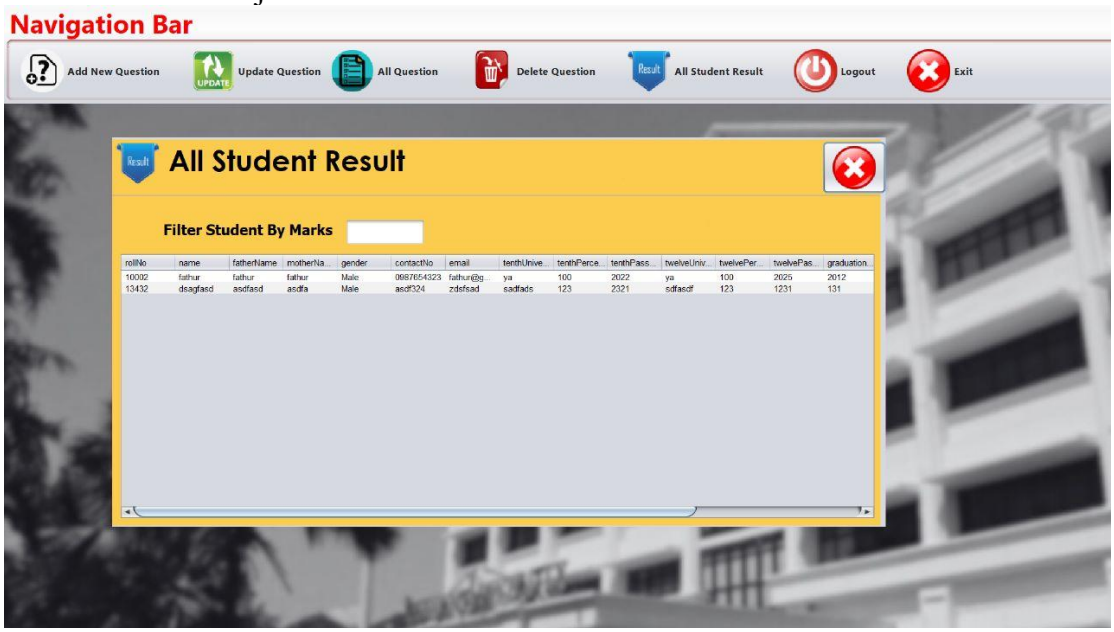
```
153     private void jMenu1MouseClicked(java.awt.event.MouseEvent evt) {
154         // TODO add your handling code here:
155         if(open==0)
156         {
157             new addNewQuestion().setVisible(b: true);
158             open=1;
159         }
160         else
161         {
162             JFrame jf=new JFrame();
163             jf.setAlwaysOnTop(alwaysOnTop: true);
164             JOptionPane.showMessageDialog(parentComponent: jf,message: "One Form is already open");
165         }
        }
```

c) In instructionStuent.java, the method performs opening the next Frame of another java file. In this context, it opens after overriding the `start` button in jButton in studentDetails.java.

```
1
2  import javax.swing.JOptionPane;
3
4
5
6  /*
7   * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
8   * Click nbfs://nbhost/SystemFileSystem/Templates/GUIForms/JFrame.java to edit this template
9   */
10
11 /**
12  *
13  * @author hp
14  */
15 public class instructionStudent extends javax.swing.JFrame {
16 public String rollNo;
17     /**
18      * Creates new form instructionStudent
19      */
20     public instructionStudent() {
21         initComponents();
22         jTextArea1.setEditable(b: false);
23
24     }
25
26     public instructionStudent(String rollNo1) {
27         initComponents();
28         JOptionPane.showMessageDialog(parentComponent: null,message: rollNo1);
29         rollNo=rollNo1;
30     }
```

User Interface Design

1. allStudentResult.java

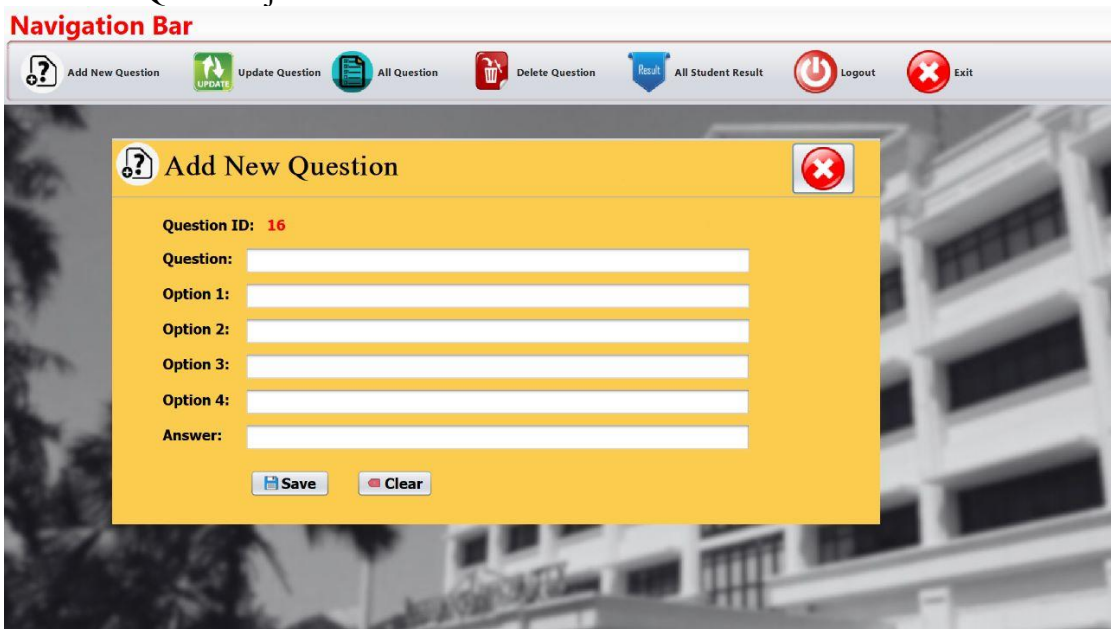


This UI form of java frame shows you the results of students' test results with assistance search bar with java textfield to filter students' results by their marks. The exit logo works to exit the page you are in ("All Student Result") back to default page.

The navigation bar panel works just as it is with other frame exist. The seven button works as it is marked.

- Click "Add New Question and it would take you to the all students java frame (to "addNewQuestion.java")
- Click "Update Question" and it would take you to the update questions java frame (to "updateQuestion.java")
- Click "All Question" and it would take you to the all questions java frame (to "allQuestion.java")
- Click "Delete Question" and it would take you to the delete question java frame (to "deleteQuestion.java")
- Click "All Student Results" and it would take you to the all student result java frame (to "allStudentResults")
- Click "Logout" and it would log you out from the page you are in take you back to login page for admin (to "index.java")
- Click "Exit" and exit you from the GUI and it would ask for confirmation before you completely exit the page.

2. addNewQuestion.java



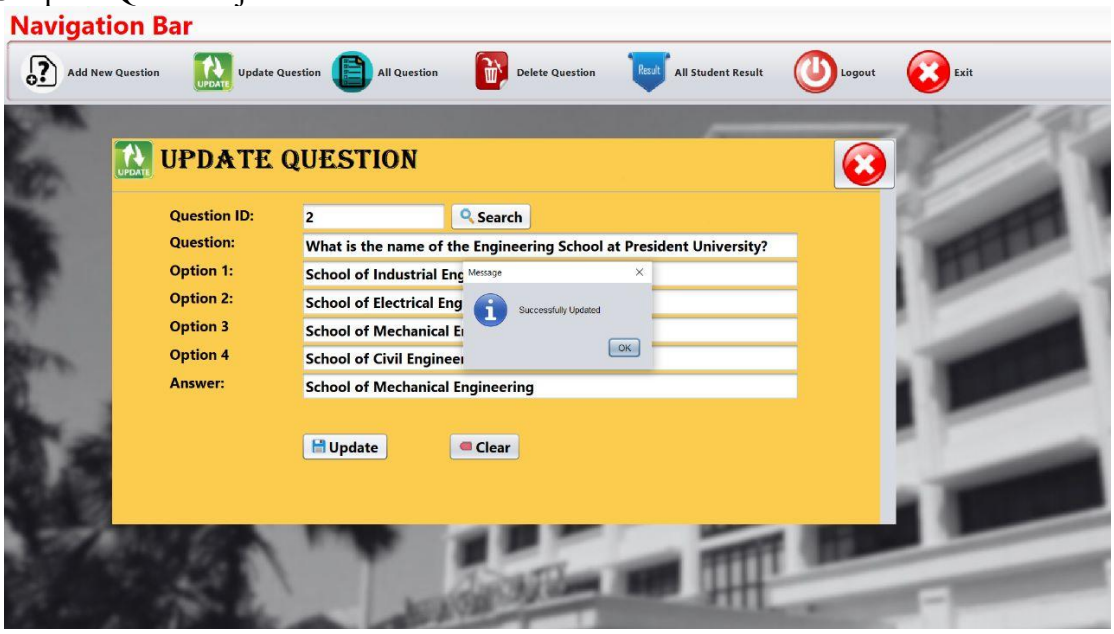
This UI form of java frame gives you a form to fill as in adding question(s) to the quiz. The "Question ID" works as it shows for representing the question number thoroughly. The form works as it shows to write the question and option for the answer. The option must be 4 and the answer should be written exactly as it is in the option as it is in the answer column. The save button require clicking for it to be inserted to the question table and automatically shift to new form with a new and arrange Question ID (question number) while before that would appear a pop up with "Successfully updated" notification and "OK" button in it. The clear button works for the admin if admin changes their mind to delete the question entirely.

The exit logo works to exit the page you are in ("Add New Question") back to default page.

The navigation bar panel works just as it is with other frame exist. The seven button works as it is marked.

- Click "Add New Question and it would take you to the all students java frame (to "addNewQuestion.java")
- Click "Update Question" and it would take you to the update questions java frame (to "updateQuestion.java")
- Click "All Question" and it would take you to the all questions java frame (to "allQuestion.java")
- Click "Delete Question" and it would take you to the delete question java frame (to "deleteQuestion.java")
- Click "All Student Results" and it would take you to the all student result java frame (to "allStudentResults")
- Click "Logout" and it would log you out from the page you are in take you back to login page for admin (to "index.java")
- Click "Exit" and exit you from the GUI and it would ask for confirmation before you completely exit the page.

3. updateQuestion.java



This UI form of java frame shows you the form to fill as in updating the question if admin made a mistake when admin write the question in add new question form. It gives you an assistance in finding the question by using a search tool with java textfield and search button with Question ID as its indicator to find the question admin are indicating. After clicking the update button it would update the new question followed by notification pop up message of "Successfully updated" and an "OK" button. The clear button works for the admin if admin changes their mind to delete the question entirely.

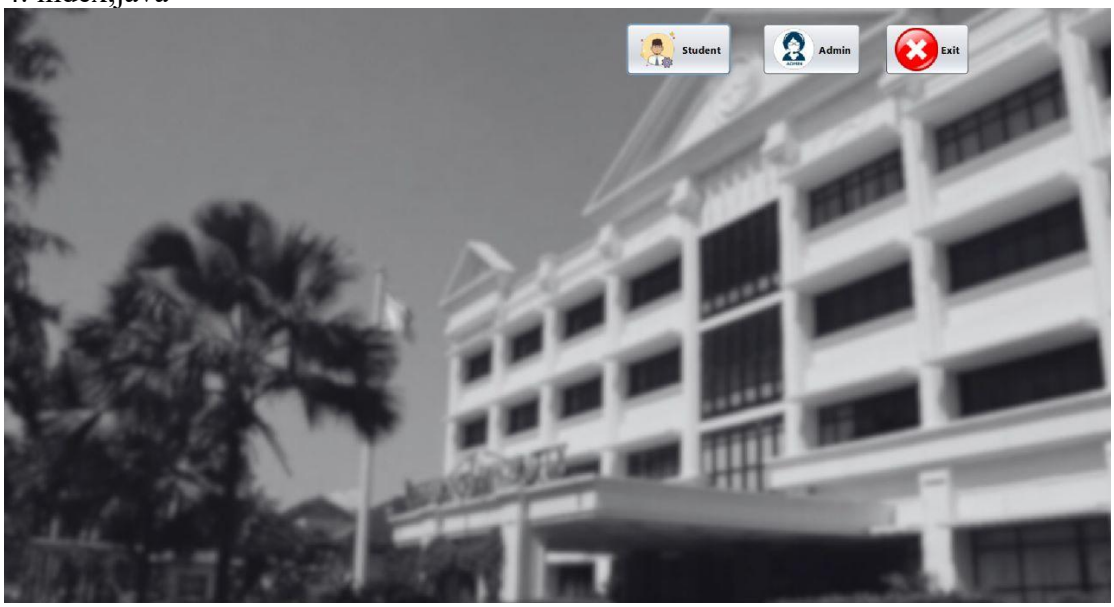
The exit logo works to exit the page you are in ("All Student Result") back to default page.

The navigation bar panel works just as it is with other frame exist. The seven button works as it is marked.

- Click "Add New Question and it would take you to the all students java frame (to "addNewQuestion.java")
- Click "Update Question" and it would take you to the update questions java frame (to "updateQuestion.java")
- Click "All Question" and it would take you to the all questions java frame (to "allQuestion.java")
- Click "Delete Question" and it would take you to the delete question java frame (to "deleteQuestion.java")
- Click "All Student Results" and it would take you to the all student result java frame (to "allStudentResults")
- Click "Logout" and it would log you out from the page you are in take you back to login page for admin (to "index.java")
- Click "Exit" and exit you from the GUI and it would ask for confirmation before you completely exit the page.

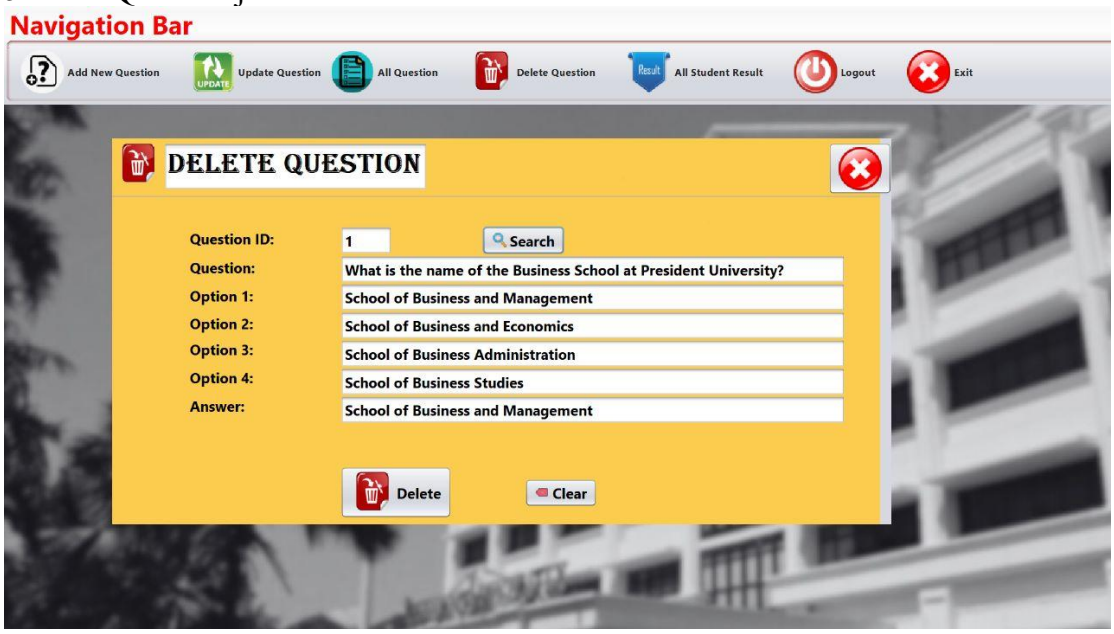
Click "Exit" and exit you from the GUI and it would ask for confirmation before you completely exit the page.

4. index,java



This UI form of java frame shows you the default or homepage of first appearance of the java program. This should pop first for all user. This page only consist of 3 java button. Student button which would send you to studentDetails.java page, Admin button would send you to loginAdmin.java page, and the exit logo works to exit the page you are in (“All Student Result”) back to default page.

5. deleteQuestion.java



This UI form of java frame shows you the option to delete question(s) by indicating the question first with initialize the Question ID with the intended question number in the textfield of the column and initiate it with the search button. After that, you could initiate the "Delete" button and delete the intended question that appear. The clear button works for the admin if admin changes their mind to delete the question entirely.

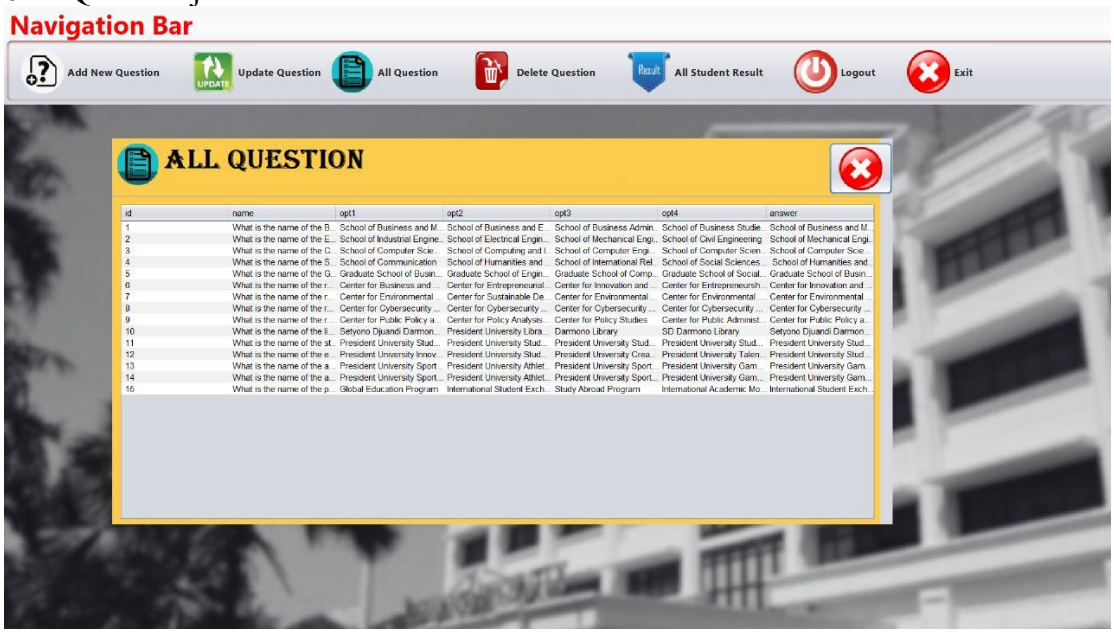
The exit logo works to exit the page you are in ("All Student Result") back to default page.

The navigation bar panel works just as it is with other frame exist. The seven button works as it is marked.

- Click "Add New Question and it would take you to the all students java frame (to "addNewQuestion.java")
- Click "Update Question" and it would take you to the update questions java frame (to "updateQuestion.java")
- Click "All Question" and it would take you to the all questions java frame (to "allQuestion.java")
- Click "Delete Question" and it would take you to the delete question java frame (to "deleteQuestion.java")
- Click "All Student Results" and it would take you to the all student result java frame (to "allStudentResults")
- Click "Logout" and it would log you out from the page you are in take you back to login page for admin (to "index.java")
- Click "Exit" and exit you from the GUI and it would ask for confirmation before you completely exit the page.

Click "Exit" and exit you from the GUI and it would ask for confirmation before you completely exit the page.

6. allQuestion.java

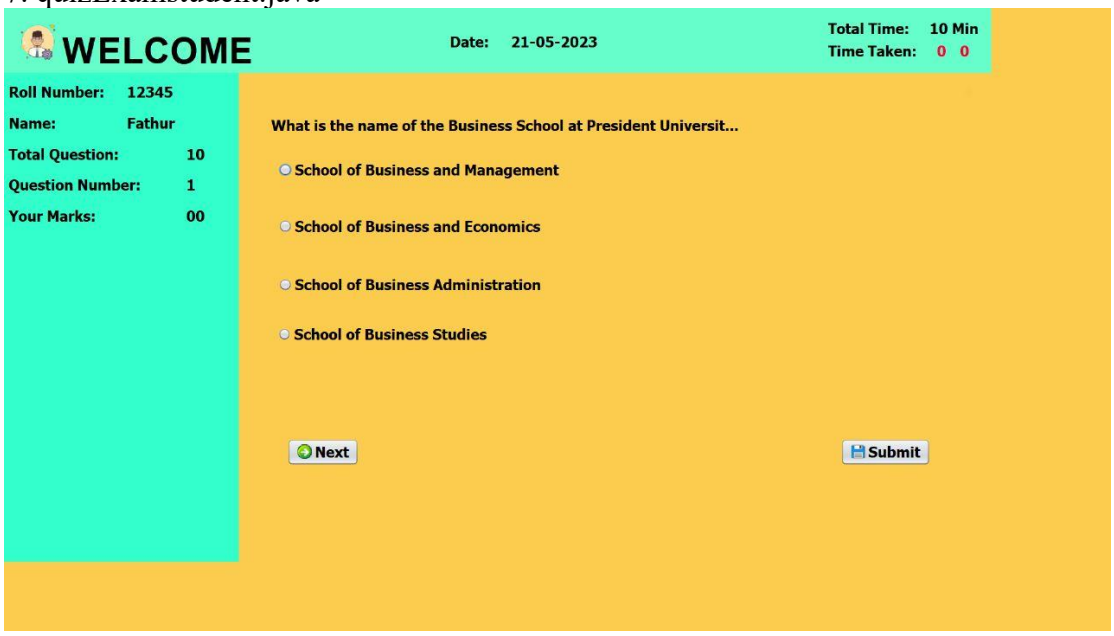


This UI form of java frame show you the list of the table existing in the localhost/phpmyadmin database that consist of the question that we have listed.
The exit logo works to exit the page you are in ("All Student Result") back to default page.
The navigation bar panel works just as it is with other frame exist. The seven button works as it is marked.

- Click "Add New Question and it would take you to the all students java frame (to "addNewQuestion.java")
- Click "Update Question" and it would take you to the update questions java frame (to "updateQuestion.java")
- Click "All Question" and it would take you to the all questions java frame (to "allQuestion.java")
- Click "Delete Question" and it would take you to the delete question java frame (to "deleteQuestion.java")
- Click "All Student Results" and it would take you to the all student result java frame (to "allStudentResults")
- Click "Logout" and it would log you out from the page you are in take you back to login page for admin (to "index.java")
- Click "Exit" and exit you from the GUI and it would ask for confirmation before you completely exit the page.

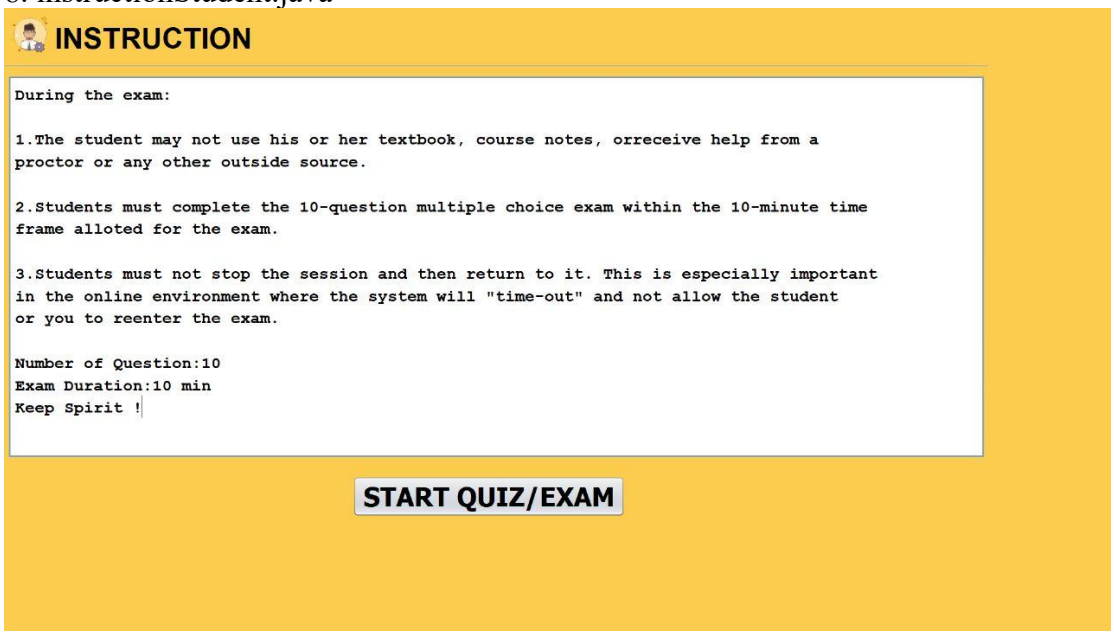
Click "Exit" and exit you from the GUI and it would ask for confirmation before you completely exit the page.

7. quizExamstudent.java



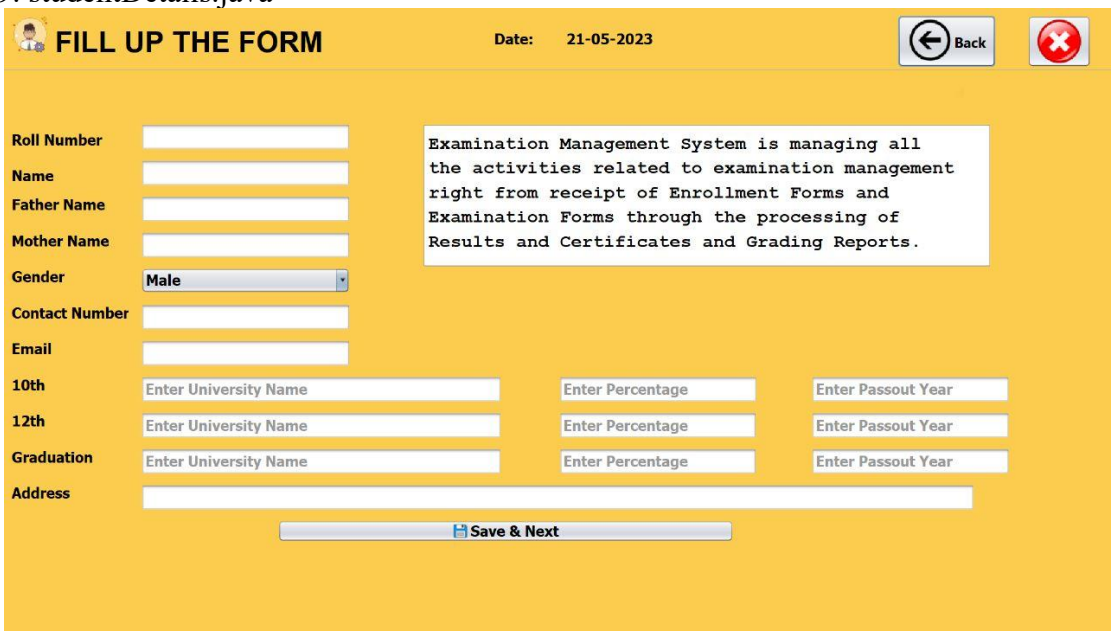
This UI form of java frame shows you the “main event” of the program which is the exam page where it shows questions and options based on the database that we created and it shows thoroughly all the information that we need based on how it’s programmed. The time constructed based on the minute and second taken. The date is constructed with the library of simple.date.format. The roll number become the indicator of the student who take the test and it shows based on the database as well as name and marks. The next and submit button works and the next button indicates the next constructed form of tables data from 2 tables of question and option where it would change in basic arrangement of next Question ID. The submit button would automatically end the quizExamStudent page to the next page of the marks and total correct answered question.

8. instructionStudent.java

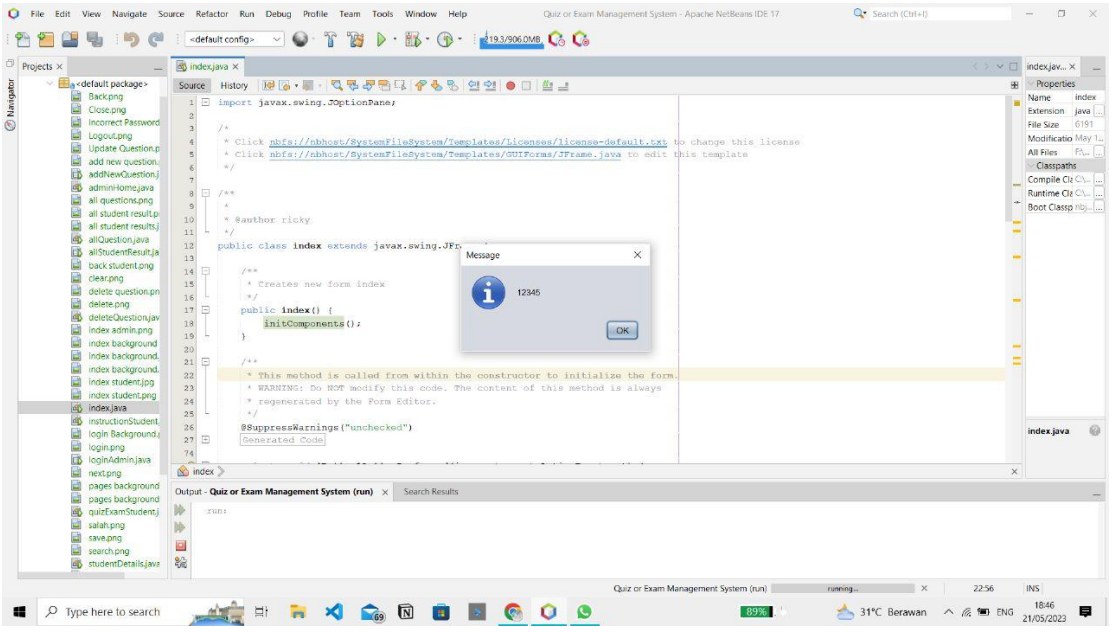


This UI form of java frame shows you a basic look of the instruction inside a textarea and the start quiz button would lead you to the quizExamStudent page or the quiz page immediately after showing you an individual pop up of the roll number you inputted in studentDetails.java. The exit logo works to exit the page you are in (“All Student Result”) back to default page.

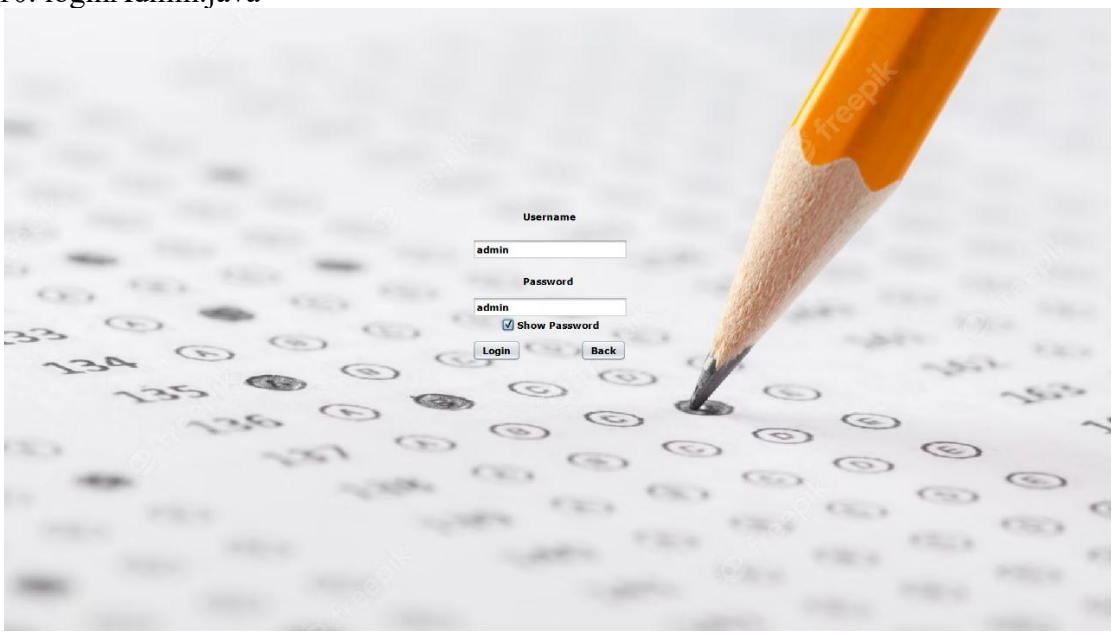
9. studentDetails.java



This UI form of java frame gives you a form to fill as a student to continue to the exam by clicking the save & next button. It would work after you complete the form of each that written with textfield and java function to make it look as such and click save & next) button. The exit logo works to exit the page you are in ("All Student Result") back to default page.



10. loginAdmin.java



This UI form of java frame shows you a page where you log in as an admin and it only works if you are registered in the database. The textfield are connected to the tables and the checkbox are connected to the textfield. The button of Login would initiate it to the next page and back would get you to previous page.

REFERENCES

Sam R. 2020. Object Oriented Programming - The Four Pillars of OOP. (https://youtu.be/1ONhXmQuWP8, accessed on May 13th 2023).

Javatpoint. 2021. Java Layout Manager. (https://www.javatpoint.com/java-layout-manager, accessed on May 13th 2023).

Minh. H. 2020. Java Connect to MySQL Database Step by Step. (https://youtu.be/duEkh8ZsFGs, accessed on May 13th 2023).

Javatpoint. 2021. Java Database Connectivity with 5 Steps. (https://www.javatpoint.com/steps-to-connect-to-the-database-in-java, accessed on May 13th 2023).

Apache. 2019. Quick Start Tutorial. (https://netbeans.apache.org/kb/docs/java/quickstart.html, accessed on May 13th 2023).

ProgrammingKnowledge2. 2023. How to Create First Java Program in NetBeans IDE for Java Programming Windows 11. (https://www.youtube.com/watch?v=SfSR4E7PZ-U, accessed on May 13th 2023).

Ravikiran A. 2023. What is Encapsulation in Java and How to Implement It. (https://www.simplilearn.com/tutorials/java-tutorial/java-encapsulation, accessed on May 13th 2023).

Huang. S. 2019. How do I call a GUI form from another Java class. (https://stackoverflow.com/questions/58957183/how-do-i-call-a-gui-form-from-another-java-class, accessed on May 13th 2023).

REQUIREMENT

- **Min pages is 10 pages**
- **Provide as much graphics as you can—not only diagrams, but any kind of diagrams and illustrations that will make it easier for us to understand and evaluate your effort. Graphics are always helpful, you have probably heard the saying "a picture is worth a thousand words"!**
- **Consistency, clarity, and correctness of the report are critical. The report must be complete, in the sense that it is *self-contained* and the reader doesn't need other materials to understand it. Anticipate what could be confusing to the graders and clarify every possible source of ambiguity or inconsistency.**
- **The report should have professional appearance; make sure that it is neat, easy to read and understand, with clearly labeled section headings, figure captions, pagination, and without grammatical and typographical errors. Also, check that diagrams and images are *readable when printed* (i.e., letters or symbols are not too small and illegible). If you are using colors in images and diagrams, check that they are discernable when printed in black-and-white/grayscale. Every figure/table must be referenced in the text and properly described.**
- ***Do not write your report as a collection of hints for which only you know the actual meaning.* Write your report from a third person's point of view. You know everything about your project so you do not need much information to understand what is in. A third person has only general knowledge of OOP and needs help in understanding how general principles are applied in your specific context.**
- **The report should be submitted to the lecturer before the final examination week ended in pdf and docx format**
- **The application should be submitted in the Gdrive**
- **Late reports will be levied a late penalty of 10% per day, up to 3 days late. After that, no credit will be given, unless you provide a written excuse from a physician. Since the deadlines are known well ahead, there will be no extensions for any of the deadlines. Please do not bother asking.**

MYSQL SYNTAX

```
create database qems;
use qems;

create table question(id varchar(10), name varchar(500), opt1
varchar(500), opt2 varchar(500), opt3 varchar(500), opt4
varchar(500), answer varchar(500));

create table student(rollNo varchar(10),
name varchar(100),
fatherName varchar(100),
motherName varchar(100),
gender varchar(50),
contactNo varchar(10),
email varchar(100),
tenthUniversityName varchar(200),
tenthPercentage varchar(10),
tenthPassoutYear varchar(5),
twelveUniversityName varchar(200),
twelvePercentage varchar(10),
twelvePassoutYear varchar(5),
graduationUniversityName varchar(200),
graduationPercentage varchar(10),
graduationPassoutYear varchar(5),
address varchar(500),
marks int(5));
```