






Module Code: ITS66704 (April 2024)

Module Name: Advanced Programming

Assignment No./Title	Assignment Task 2 & Task 3 (Group Project) 20% (PART A - ANALYSIS AND DESIGN) 30% (PART B - DEVELOPMENT) 10% (PRESENTATION)
Course Tutor/Lecturer	Mr. Subit Timalisina
Submission Date	Week 09: TBA (PART A - ANALYSIS AND DESIGN) Week 12: TBA (PART B - DEVELOPMENT) Week 13: TBA (PRESENTATION)

Student Name	Student ID	Student Signature
Prabin Joshi	0358667	
Sushil Kumar Thakur	0358230	
Aayushma Shrestha	0358270	
Lina Maharjan	0358308	
Ravi Prasad Kanu	0358238	

ProjectPartBReport_GroupNo.3

Marks/Grade:	Evaluated By:
Evaluator's Comments:	

Table of Content

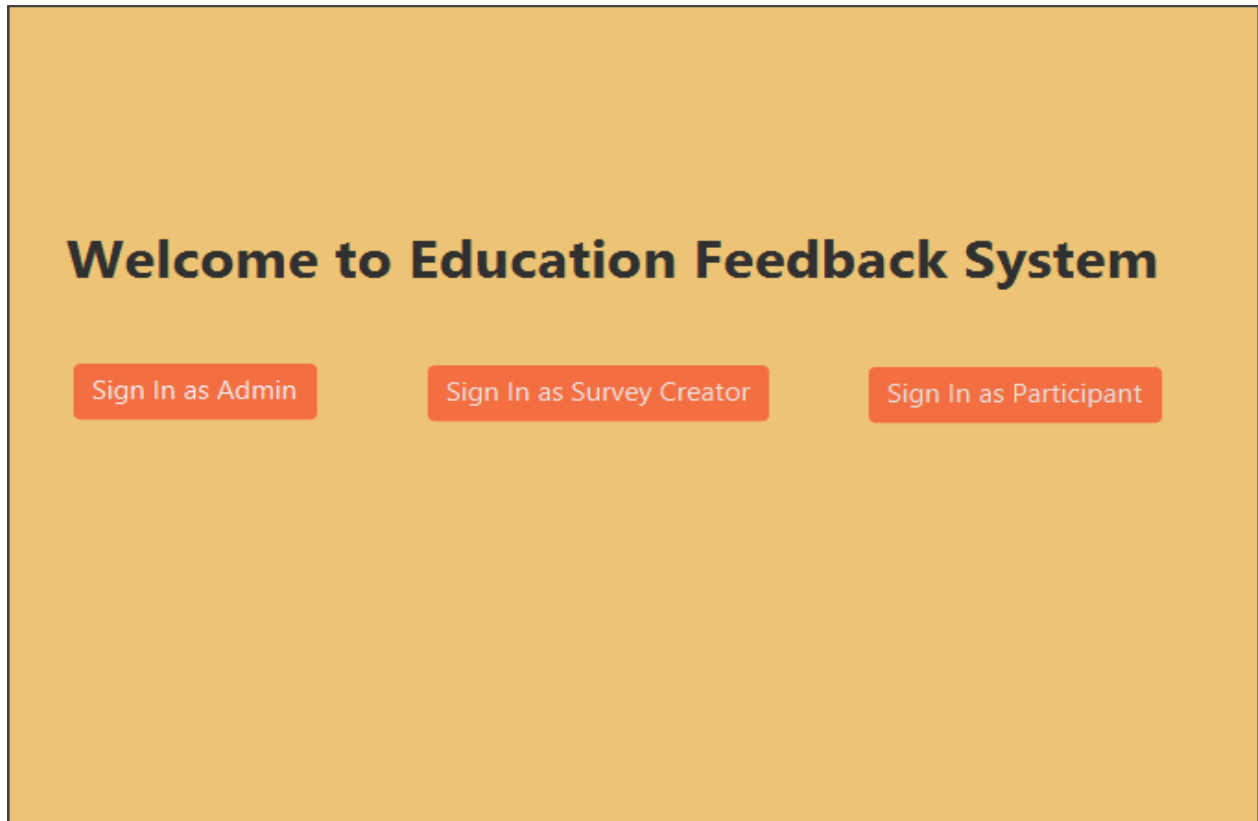
Table of Content	2
Introduction	3
Testing:	4
Initial Page	4
Register Page.....	5
Login Page	7
As Admin	7
As Survey Creator:	8
As Participant:	8
Begin the Survey Page	9
Questions Page.....	9
CSV File Screenshot	10
- AdminData.csv	10
- Creator.csv	10
- Participant.csv.....	11
- SurveyQuestion.....	11
OBJECT-ORIENTED CONCEPTS	12
• Encapsulation:	12
Conclusion	13
Source Code	14
Appendix:.....	21

Introduction

The Education Feedback System is essential in the rapidly evolving terrain of Nepal nowadays. The goal of the proposed project is to create a survey management system that is especially suited to the educational environment in Nepal. Using object-oriented design concepts, this system will be centered on the Education Feedback System and will guarantee reliability, security, and flexibility. Through the utilization of both real-time data collecting and thorough data analysis, the system will be able to adapt to the specific requirements of Nepal's varied educational environment. The objective is to provide a safe, adaptable, and user-friendly platform that closes the current gaps in the educational process and promotes ongoing development.

Testing:

Initial Page

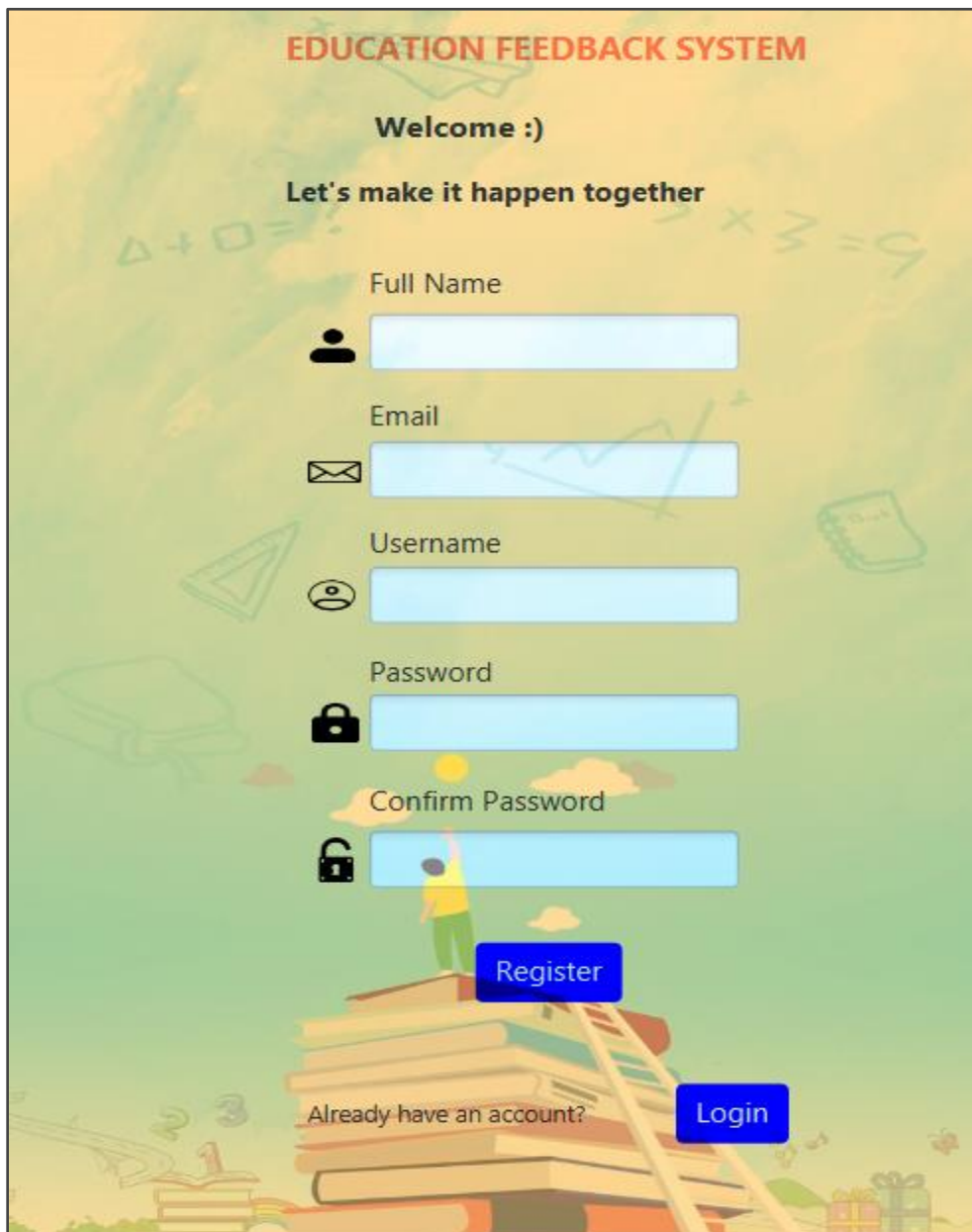


In this page, you can select 3 types of ***“Sign In”*** option for users with different roles as Admin, Survey Creator and as a Participant.

Register Page

As a Admin Page

Here, a new user can “**Register**” by giving their full name, username, email address and password in the form. Then it is registered in the *csv* file. If you already have an account, you can also Click on the Login button.



The registration form is titled "EDUCATION FEEDBACK SYSTEM" in red. Below the title, it says "Welcome :)" and "Let's make it happen together". The form includes five input fields: "Full Name" (with a person icon), "Email" (with an envelope icon), "Username" (with a person icon), "Password" (with a lock icon), and "Confirm Password" (with an open lock icon). At the bottom, there is a "Register" button, a "Login" button, and a link that says "Already have an account?". The background of the form is a colorful illustration featuring a stack of books, a ladder, a sun, clouds, and various educational symbols like a ruler, compass, and mathematical equations.

EDUCATION FEEDBACK SYSTEM

Welcome :)

Let's make it happen together

Full Name

Email

Username

Password

Confirm Password

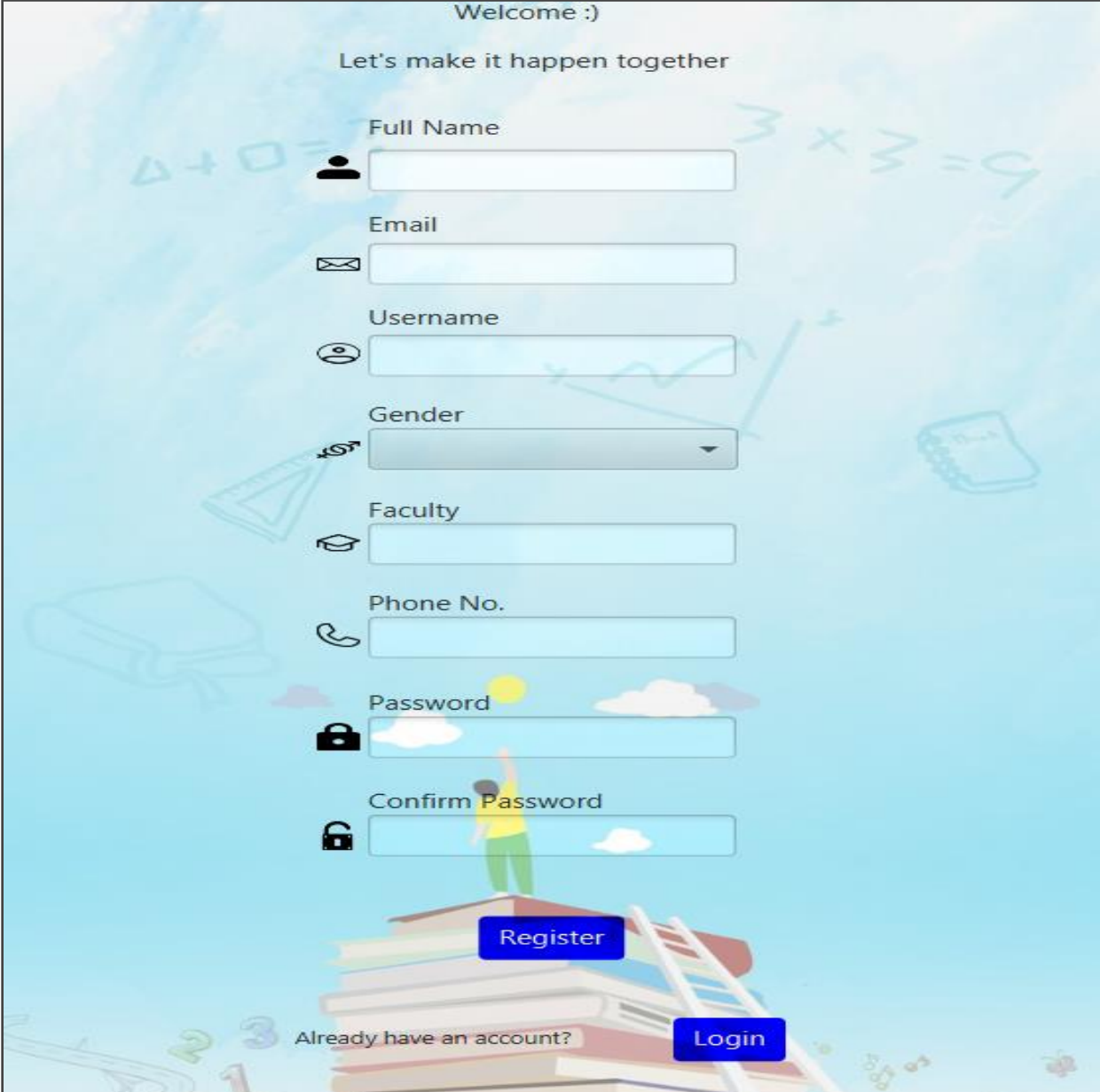
Register

Already have an account?

Login

As a Participant

As new participants you can **“Register”** by providing yours Full name, username, email address, faculty, gender, Phone number and password in the form. Then it is registered in *csv* file. And also, If you already have an account you can also Click on the Login button.



Welcome :)

Let's make it happen together

Full Name

Email

Username

Gender

Faculty

Phone No.

Password

Confirm Password

Already have an account?

The form is set against a light blue background with faint, artistic sketches of mathematical symbols (like $\Delta + 0 =$, $3 \times 3 = 9$), a ruler, a notebook, and a stack of books. A small figure of a person is standing on the stack of books, reaching up towards the password fields.

Login Page

As Admin

In this page if you are an admin and already have an account you can login and start creating or view survey questions that you have previously created. You can “**Login**” through your Email/Username and password and if you don’t have an account this page also guides you to sign Up as an admin page.


Education Feedback System

Welcome Back :)


To keep connected with us please login by using email or username and password

Sign In as Admin

Email/Username



Password




[Forget Password?](#)

Login

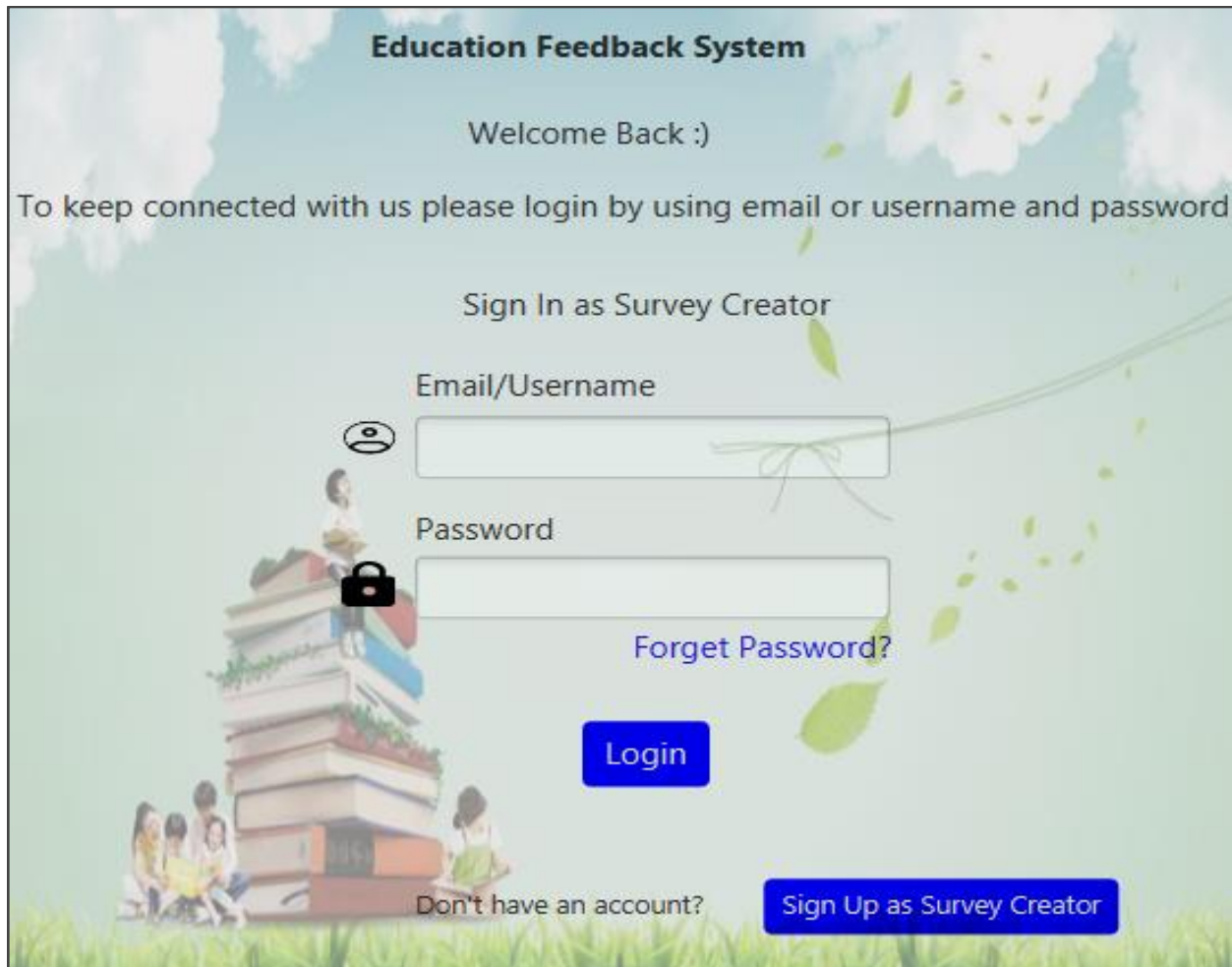
Don't have an account?

Sign Up as Admin



As Survey Creator:

In this page if you are a Survey Creator and already have an account you can login and start creating or view survey questions that you have previously created. You can “**Login**” through your Email/Username and password and if you don’t have an account this page also guides you to sign Up as an admin page.



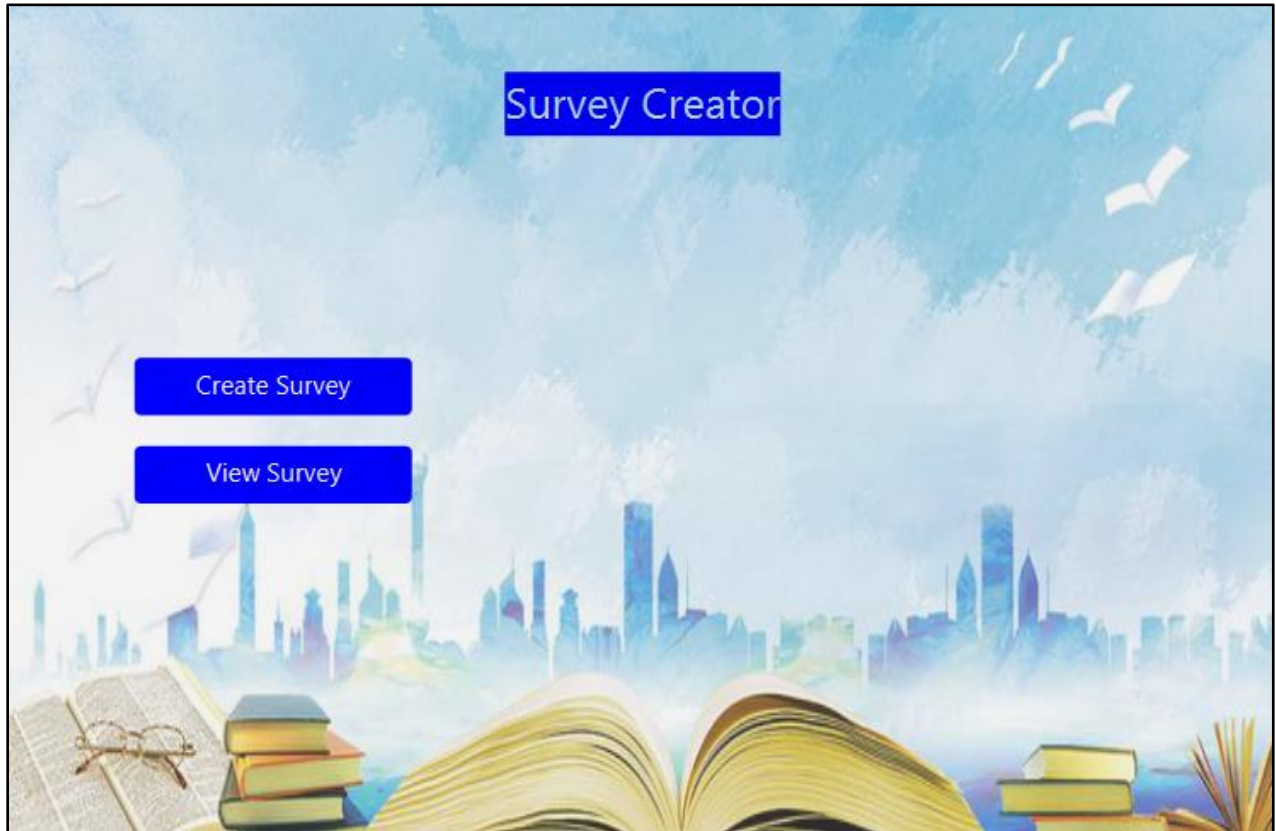
The screenshot shows the login interface for the 'Education Feedback System'. The background features a stack of books with children sitting around them, set against a sky with clouds and falling leaves. The text on the page reads: 'Education Feedback System', 'Welcome Back :)', 'To keep connected with us please login by using email or username and password', 'Sign In as Survey Creator'. There are two input fields: 'Email/Username' with a user icon and 'Password' with a lock icon. A blue 'Login' button is positioned below the password field. A link 'Forget Password?' is located to the right of the password field. At the bottom, there is a link 'Don't have an account?' and a blue button labeled 'Sign Up as Survey Creator'.

As Participant:

In this page if you are a participant and already have an account you can “**Login**” and start doing survey questions. You can login through your Email/Username and password and if you don’t have an account this page also guides you to sign Up as a participant page.

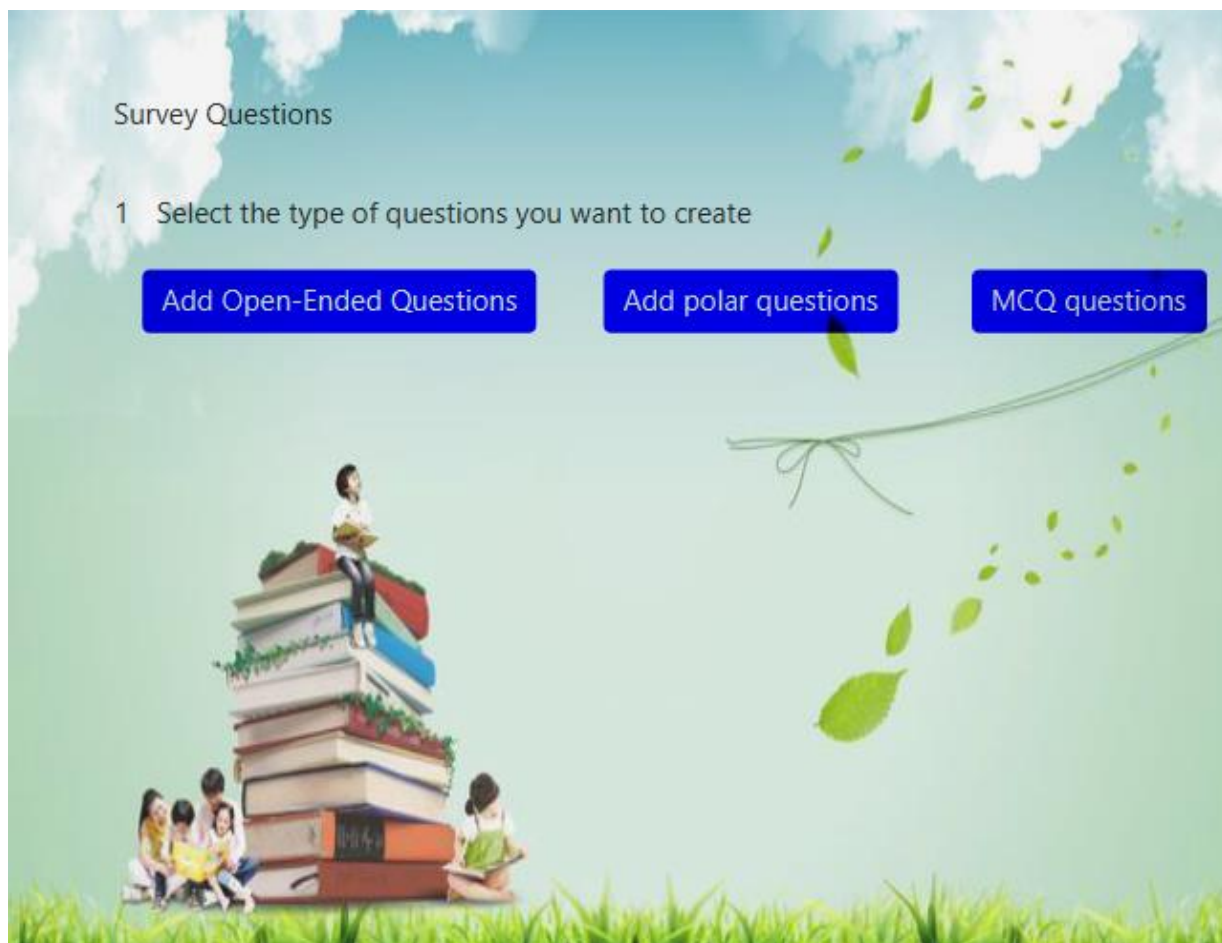
Begin the Survey Page

This page guides the survey creators if they want to create a new survey or view the completed previous survey results.



Questions Page

This page guides the survey creators to choose what type of questions they want to add to their survey. Here in this figure below you can see three types of survey questions i.e; Open end questions, polar questions and MCQ questions. Survey creators can choose any of these types of questions.



CSV File Screenshot

- AdminData.csv

```
AdminData.csv x
1 "Prabin","prabin17@gmail.com","prabin11","123","123"
2 "yangma","yangma17@gmail.com","yangma11","yangma","yangma"
3 "aayush","aayush@gmail.com","aayush17","1234","1234"
```

- Creator.csv

```
Creator.csv x
1 "sushil","sushil1@gmail.com","sushil1","234","234","male","bcs","123"
2 "Rakesh","rakesh17@gmail.com","rakesh11","male","BHM","999999999","rakesh","rakesh"
```

- Participant.csv

```

≡ Participant.csv ×
1  "prabin","prabi","prabin11","male","bcs","2222","joshi","joshi"
2  "John Doe","john@example.com","j1","male","maths","33333","jphn","john"
3  "aayush","aayush@","a2","male","bhm","17","1717","1717"
4  "yangma","yangma@","yangma11","female","bcs","222","222","222"

```

- SurveyQuestion

```

✓ SurveyQuestions.1SurveyQuestion
  ≡ mcq.csv
  ≡ polar.csv
  ≡ tquestions.csv

```

```

≡ mcq.csv ×  ≡ polar.csv  ≡ tquestions.csv
1  Overall, how do you rate the quality of teaching in schools, colleges and universities in Nepal?
2  Excellent
3  Good
4  Average
5  Poor

```

```

≡ polar.csv ×
1  Do you think the quality of teaching overall is acceptable for schools, colleges and universities? Yes No

```

```

≡ tquestions.csv ×
1  What obstacles must academic staff members overcome in order to uphold the institution's high standards
2
3

```

Users' registration data is stored and arranged in the corresponding CSV files (AdminData.csv for administrators and Creator.csv for creators). Every file records necessary details about the user, guaranteeing that the data is organised and saved for later use or consultation. This data storage technique makes it easier to manage and access user information by keeping an organised record of all registered users. Likewise, we manage participant data that is saved in participant csv.

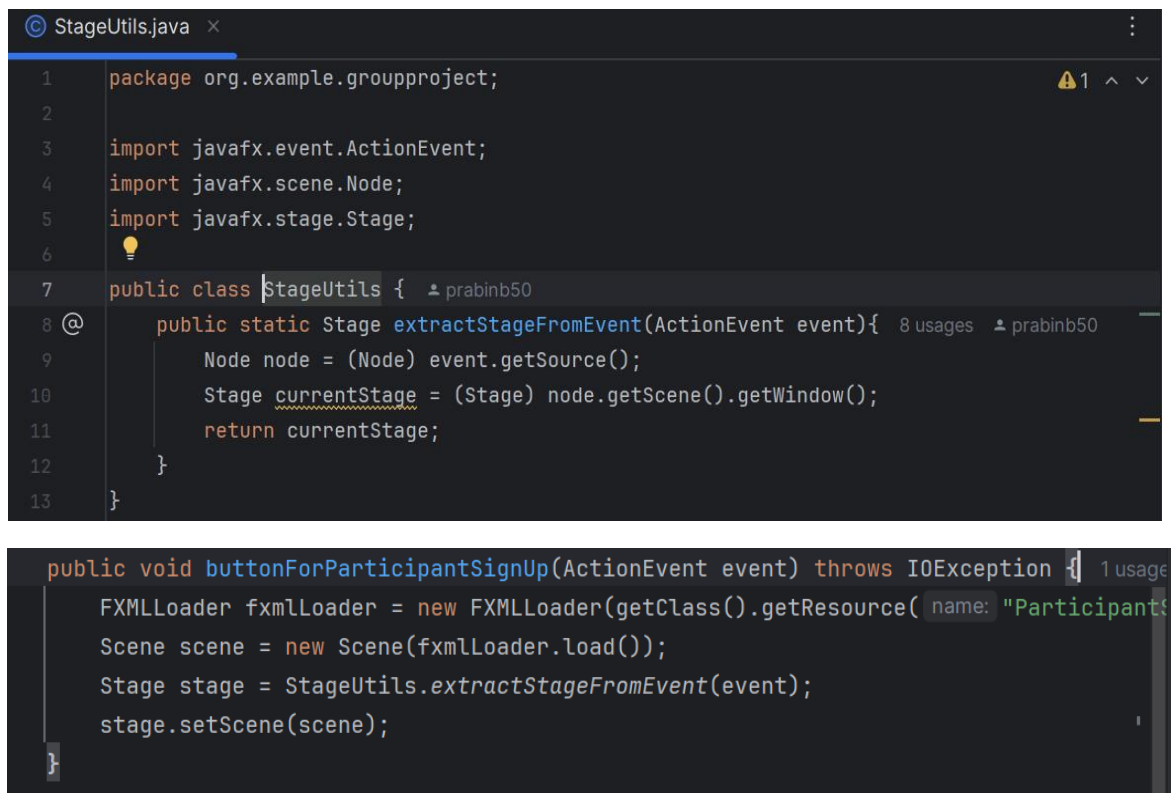
Survey Creator maintains an organised approach to managing survey data by grouping survey questions into distinct CSV files and enabling a survey author to add new questions. The fact that

every kind of question is kept in a separate file makes it simple to manage and modify the survey questions as needed.

OBJECT-ORIENTED CONCEPTS

The Object-Oriented Concepts module is essential for the system's architecture and code structure. It leverages key object-oriented principles including instantiation, encapsulation, inheritance, and polymorphism. These concepts enhance the codebase's measured quality, adaptability, and maintainability, ensuring the system's robustness and adaptability to future enhancements.

- **Encapsulation:**



```
StageUtils.java x
1 package org.example.groupproject;
2
3 import javafx.event.ActionEvent;
4 import javafx.scene.Node;
5 import javafx.stage.Stage;
6
7 public class StageUtils {  prabinb50
8     @ public static Stage extractStageFromEvent(ActionEvent event){ 8 usages  prabinb50
9         Node node = (Node) event.getSource();
10        Stage currentStage = (Stage) node.getScene().getWindow();
11        return currentStage;
12    }
13 }

public void buttonForParticipantSignUp(ActionEvent event) throws IOException { 1 usage
    FXMLLoader fxmlLoader = new FXMLLoader(getClass().getResource("Participant.fxml"));
    Scene scene = new Scene(fxmlLoader.load());
    Stage stage = StageUtils.extractStageFromEvent(event);
    stage.setScene(scene);
}
```

The OOP idea of encapsulation is used to good use in the code that is supplied. Through its static function `extractStageFromEvent`, the `StageUtils` class encapsulates the procedure for extracting a `Stage` from an `ActionEvent`. By abstracting away the specifics of the `Stage`'s retrieval, this function gives other classes a clear, reusable interface. This encapsulated feature is used by the `buttonForParticipantSignUp` method in the `ParticipantLoginController` class to call `StageUtils` to acquire the current `Stage`. `stageFromEvent` is extracted (event). This illustrates encapsulation since the `ParticipantLoginController` only utilises the supplied method and doesn't need to

understand the ins and outs of extracting the stage from the event. The division of responsibilities results in code that is more legible, modular, and manageable and upholds the concepts of OOP by encouraging reuse and minimizing connection between classes.

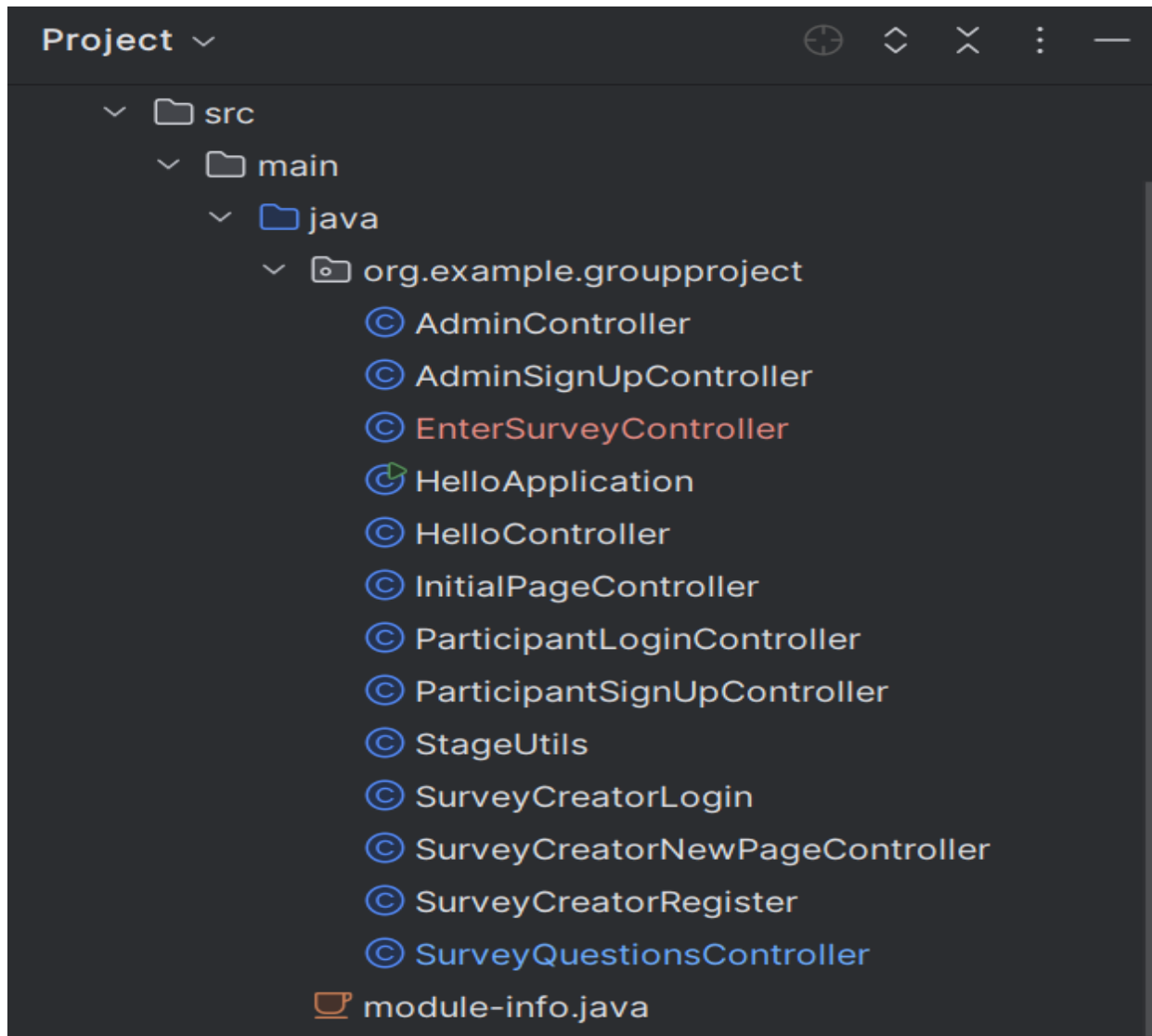
Conclusion

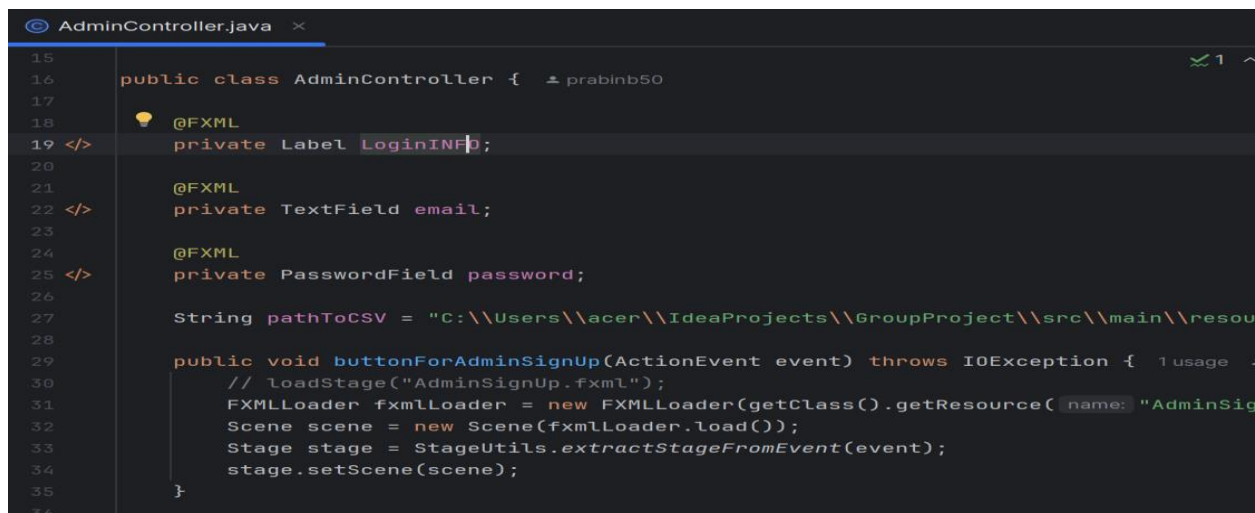
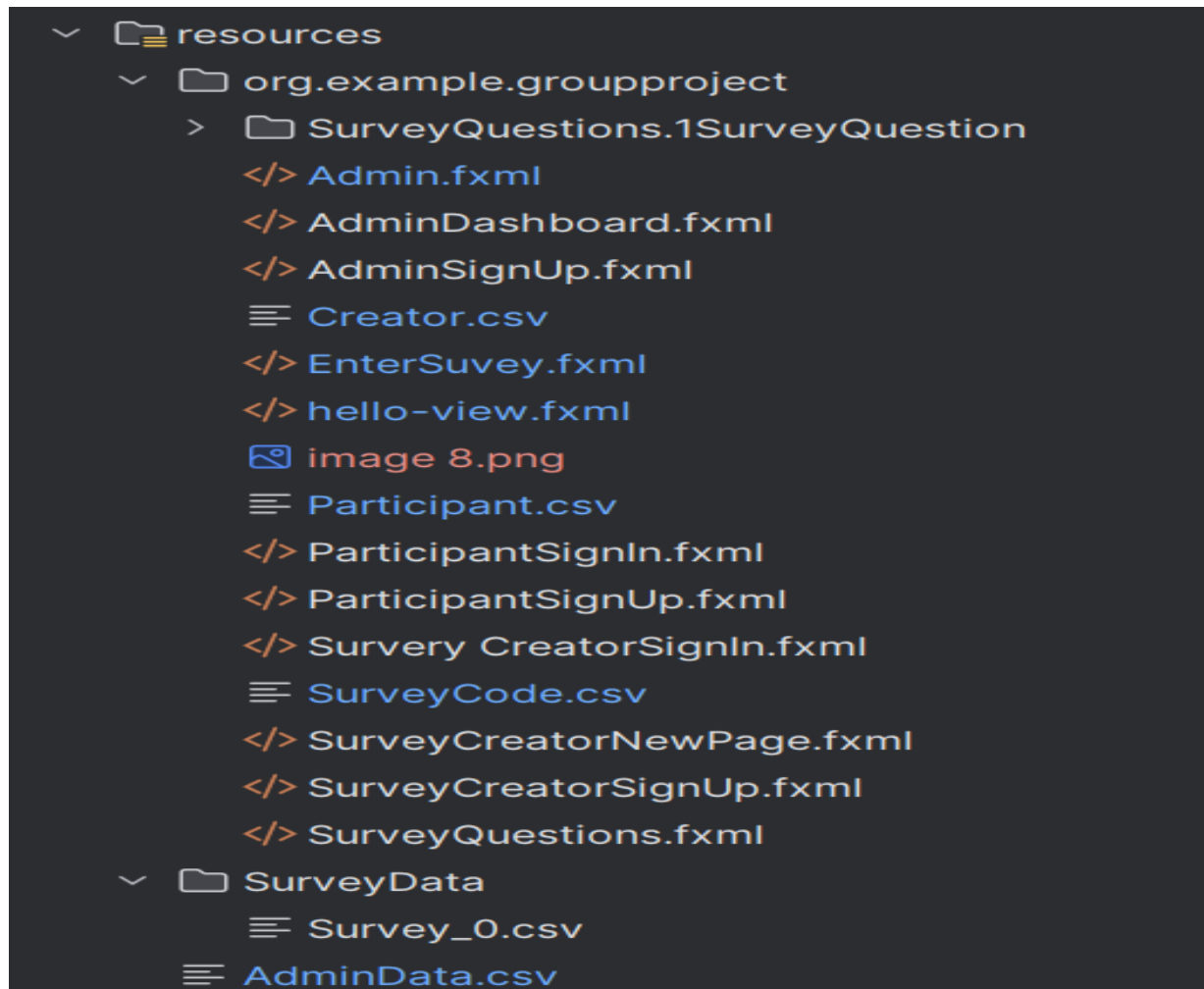
In summary, the creation of the Survey Management System for Nepal's educational sector is an encouraging step in the use of advanced programming to address important issues. The system maintains resilience and flexibility by utilizing encapsulation, principles of object-oriented design. This modular framework facilitates secure data collecting, survey development, and thorough analysis by integrating smoothly with database operations and user interface components.

The system's functionality and focus on user design are enhanced by the integration of secure data processing, configurable survey formats, and real-time data gathering. Enhancing decision-making processes in education satisfies practical goals in addition to meeting technical criteria. The Survey Management System is an inventive solution that is specifically designed for Nepal's educational environment, showcasing its capacity to fix current gaps and foster ongoing development.

Additionally, the project emphasizes the use of state-of-the-art programming approaches to give educational stakeholders a dependable and effective tool. Its extensive feature set and easy-to-use interface demonstrate its value to the advancement of education and its preparedness to adapt to the changing needs of the Nepalese educational system. All things considered, the Survey Management System is a monument to the successful fusion of cutting-edge programming and useful solutions, tackling important issues in education with a finely tuned and user-friendly application.

Source Code






```

AdminSignUpController.java x
15
16 public class AdminSignUpController {  prabinb50
17
18     @FXML
19 </> private Label userINFO;
20
21     @FXML
22 </> private TextField adminName;
23
24     @FXML
25 </> private TextField adminEmail;
26
27     @FXML
28 </> private TextField adminUsername;
29
30     @FXML
31 </> private PasswordField adminPassword;
32
33     @FXML
34 </> private PasswordField adminConfirmPassword;
35
36     String pathToCSV = "C:\\Users\\acer\\IdeaProjects\\GroupProject\\src\\main\\resource
37
38     public void buttonForAdminRegister() { 1 usage  prabinb50
39         String AdminName = adminName.getText();
40         String AdminEmail = adminEmail.getText();
41         String AdminUsername = adminUsername.getText();
42         String AdminPassword = adminPassword.getText();
43         String AdminConformPassword = adminConfirmPassword.getText();
44         try {

```

```

EnterSurveyController.java x
17 import java.util.stream.Stream; 12 3 8
18
19 public class EnterSurveyController {
20
21     @FXML 2 usages
22 </> public TextField SCode;
23
24     @FXML
25 </> private Label validationLabel;
26
27     private String baseDir = "C:\\Users\\acer\\IdeaProjects\\GroupProject\\src\\ma
28
29     @FXML 1 usage
30     public void EnterSurvey() {
31         String inputCode = SCode.getText();
32         boolean isCodeVerified = false;
33
34         try (Stream<Path> paths = Files.list(Paths.get(baseDir))) {
35             for (Path path : (Iterable<Path>) paths::iterator) {
36                 if (Files.isDirectory(path) && path.getFileName().toString().matches
37                     File randomCSVFile = new File(path.toString(), child: "0Survey_0.
38
39                     if (randomCSVFile.exists()) {
40                         try (BufferedReader br = new BufferedReader(new FileReader(
41                             String line;
42                             while ((line = br.readLine()) != null) {
43                                 if (line.trim().equals(inputCode)) {
44                                     isCodeVerified = true;
45                                     validationLabel.setText("Code verified. Reading

```

ProjectPartBReport_GroupNo.3

```
InitialPageController.java x
4  import javafx.fxml.FXMLLoader;
5  import javafx.scene.Parent;
6  import javafx.scene.Scene;
7  import javafx.scene.control.Label;
8  import javafx.stage.Stage;
9
10 import java.io.IOException;
11
12 public class InitialPageController {  prabinb50
13
14     @FXML
15     private Label userINFO;
16
17     public void buttonForAdminSignIn() throws IOException { 1 usage  prabinb50
18         loadStage( sceneName: "/org/example/groupproject/Admin.fxml");
19     }
20
21     public void buttonForCreatorSignIn() throws IOException{ 1 usage  prabinb50
22         loadStage( sceneName: "/org/example/groupproject/Survery CreatorSignIn.fxml");
23     }
24
25     public void buttonForParticipantSignIn() throws IOException{ 1 usage  prabinb50
26         loadStage( sceneName: "/org/example/groupproject/ParticipantSignIn.fxml");
27     }
28
29     @FXML 3 usages  prabinb50
30     public void loadStage(String sceneName) throws IOException{
31         try {
```

```
ParticipantLoginController.java x
16 public class ParticipantLoginController {  prabinb50
21 </> private TextField ParticipantEmail;
22
23     @FXML
24     private PasswordField ParticipantPassword;
25
26     String pathOfCSVParticipant = "C:\\Users\\acer\\IdeaProjects\\GroupProject\\src\\
27
28     public void buttonForParticipantSignUp(ActionEvent event) throws IOException { 1
29         // loadStage("AdminSignUp.fxml");
30         FXMLLoader fxmlLoader = new FXMLLoader(getClass().getResource( name: "Particip
31         Scene scene = new Scene(fxmlLoader.load());
32         Stage stage = StageUtils.extractStageFromEvent(event);
33         stage.setScene(scene);
34     }
35
36     public void buttonForParticipantLogin(ActionEvent event){ 1 usage  prabinb50
37         String emailText = ParticipantEmail.getText();
38         String passwordText = ParticipantPassword.getText();
39         try {
40             FileReader fileReader = new FileReader(pathOfCSVParticipant);
41             CSVReader csvReader = new CSVReader(fileReader);
42             String[] rows;
43             boolean found = false;
44             while ((rows = csvReader.readNext()) != null){
45                 if (rows.length >= 7 && emailText.equals(rows[1]) && passwordText.equals
```

```

ParticipantSignUpController.java x
17
18 public class ParticipantSignUpController {  prabinb50
19     @FXML
20     private Label userINFO;
21
22     @FXML no usages
23     private Label loginINFO;
24
25     @FXML
26     private ChoiceBox<String> creatorGender;
27
28     @FXML prabinb50
29     public void initialize(){
30         if(creatorGender != null){
31             creatorGender.getItems().addAll("male", "female", "others");
32         } else{
33             System.err.println("error");
34         }
35     }
36
37     @FXML
38     private TextField participantSignUpName;
39
40     @FXML
41     private TextField participantSignUpEmail;
42
43     @FXML
44     private TextField participantSignUpUsername;

```

```

StageUtils.java x
1 package org.example.groupproject;
2
3 import javafx.event.ActionEvent;
4 import javafx.scene.Node;
5 import javafx.stage.Stage;
6
7 public class StageUtils { prabinb50
8     @
9     public static Stage extractStageFromEvent(ActionEvent event){ 8 usages prabinb50
10         Node node = (Node) event.getSource();
11         Stage currentStage = (Stage) node.getScene().getWindow();
12         return currentStage;
13     }

```

ProjectPartBReport_GroupNo.3

```
SurveyCreatorLogin.java x
16 public class SurveyCreatorLogin {  prabinb50
17
18
19
20
21 @FXML
22 </> private TextField SurveyCreatorEmail;
23
24 @FXML
25 </> private PasswordField SurveyCreatorPassword;
26
27 String pathToCSVCreator = "C:\\Users\\acer\\IdeaProjects\\GroupProject\\src\\ma
28
29 public void buttonForSurveyCreatorSignUp(ActionEvent event) throws IOException
30 // loadStage("AdminSignUp.fxml");
31 FXMLLoader fxmlLoader = new FXMLLoader(getClass().getResource("name: "Survey
32 Scene scene = new Scene(fxmlLoader.load()));
33 Stage stage = StageUtils.extractStageFromEvent(event);
34 stage.setScene(scene);
35 }
36
37 public void buttonForSurveyCreatorLogin(ActionEvent event){ 1 usage  prabinb50
38 String emailText = SurveyCreatorEmail.getText();
39 String passwordText = SurveyCreatorPassword.getText();
40 try {
41     FileReader fileReader = new FileReader(pathToCSVCreator);
42     CSVReader csvReader = new CSVReader(fileReader);
43     String[] rows;
44     boolean found = false;
45     while ((rows = csvReader.readNext()) != null){
46         if (emailText.equals(rows[1]) && passwordText.equals(rows[6])){
47             LoginINF0.setText("Login Successful");
48         }
49     }
50 } catch (IOException e) {
51     LoginINF0.setText("Login Failed");
52 }
53 }
```

```
SurveyCreatorNewPageController.java x
1 package org.example.groupproject;
2
3 import com.opencsv.CSVWriter;
4 import javafx.event.ActionEvent;
5 import javafx.fxml.FXML;
6 import javafx.fxml.FXMLLoader;
7 import javafx.scene.Scene;
8 import javafx.scene.control.Label;
9 import javafx.stage.Stage;
10
11 import java.io.File;
12 import java.io.FileWriter;
13 import java.io.IOException;
14 import java.util.Random;
15
16 public class SurveyCreatorNewPageController {  prabinb50
17 @FXML
18 </> private Label userINF0;
19
20 public void buttonForCreateSurvey(ActionEvent event) throws IOException { 1 usage
21     userINF0.setText("Survey Created click on view survey");
22     createCSVFile();
23     FileWriter fileWriter1 = new FileWriter("fileName: "C:\\Users\\acer\\IdeaProjec
24     CSVWriter csvWriter1 = new CSVWriter(fileWriter1);
25     String data1 = generateRandomWord();
26     String[] data2 = {data1};
27     csvWriter1.writeNext(data2);
28     csvWriter1.close();
29     userINF0.setText("data1");
30 }
```

```

© SurveyQuestionsController.java x
23 public class SurveyQuestionsController {  prabinb50 *
158 private File createNewDirectory(String baseDir) { 1 usage  prabinb50
165     try (Stream<Path> paths = Files.list(Paths.get(baseDir))) {
166         nextNumber = paths.filter(Files::isDirectory) Stream<Path>
167         .map(path -> path.getFileName().toString()) Stream<String>
168         .filter(name -> name.matches(regex: "\\d+SurveyQuestion"))
169         .map(name -> Integer.parseInt(name.replace(target: "SurveyQuestion",
170         .max(Comparator.naturalOrder()) Optional<Integer>
171         .orElse(other: 0) + 1;
172     } catch (IOException e) {
173         e.printStackTrace();
174     }
175
176     File newDir = new File(baseDir, child: nextNumber + "SurveyQuestion");
177     if (!newDir.exists()) {
178         newDir.mkdirs();
179     }
180     return newDir;
181 }
182
183 @FXML 1 usage  prabinb50
184 > public void saveTQuestions(String dirPath) { saveQuestionsToCSV( questionPrefix: "tquest
187
188 @FXML 1 usage  prabinb50
189 > public void saveMCQs(String dirPath) { saveQuestionsToCSV( questionPrefix: "mcq" fileNa

```

```

© SurveyCreatorRegister.java x
17
18 public class SurveyCreatorRegister {  prabinb50
19
20 @FXML
21 </> private Label userINFO;
22
23 @FXML
24 </> private TextField creatorName;
25
26 @FXML
27 </> private TextField creatorEmail;
28
29 @FXML
30 </> private TextField creatorUsername;
31
32 @FXML
33 </> private ChoiceBox<String> creatorGender;
34
35 @FXML  prabinb50
36 public void initialize(){
37     if(creatorGender != null){
38         creatorGender.getItems().addAll(...es: "male", "female", "others");
39     } else{
40         System.err.println("error");
41     }
42 }
43
44 @FXML
45 </> private TextField creatorFaculty;

```

```
HelloApplication.java x
1 package org.example.groupproject;
2
3 import javafx.application.Application;
4 import javafx.fxml.FXMLLoader;
5 import javafx.scene.Scene;
6 import javafx.stage.Stage;
7
8 import java.io.IOException;
9
10 public class HelloApplication extends Application {
11     @Override
12     public void start(Stage stage) throws IOException {
13         FXMLLoader fxmlLoader = new FXMLLoader(HelloApplication.class.getResource(
14             "scene.fxml"));
15         Scene scene = new Scene(fxmlLoader.load(), 320, 240);
16         stage.setTitle("EDUCATION FEEDBACK SYSTEM");
17         stage.setScene(scene);
18         stage.show();
19     }
20
21     public static void main(String[] args) { launch(); }
22 }
23
```

Appendix:

<https://github.com/prabinb50/javafx>