STUDENT DATABASE MANAGEMENT SYSTEM

A MINI PROJECT REPORT

Submitted by

DHARANEEISH B K **230701072**

JAYASUDHAN V **230701131**

In partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE

RAJALAKSHMI ENGINEERING COLLEGE (AUTONOMOUS)

THANDALAM

CHENNAI-602105

2024 - 25

BONAFIDE CERTIFICATE

Certified that this project report "**STUDENT DATABASE MANAGEMENT SYSTEM**" is the bonafide work of **"DHARANEEISH B K (230701072), JAYASUDHAN (230701131)"**

who carried out the project work under my supervision.

Submitted for the Practical Examination held on _____

SIGNATURE

Mrs.Divya.M
Assistant Professor,
Computer Science and Engineering,
Rajalakshmi Engineering College ,Thandalam,
Chennai -602105

SIGNATURE

Mr.Ragu G
Assistant Professor ,
Computer Science and Engineering,
Rajalakshmi Engineering College,Thandalam,
Chennai-602105

INTERNAL EXAMINER

EXTERNAL EXAMINER

ABSTRACT:

A **Student Database Management System (SDMS)** is a comprehensive software solution designed to streamline and automate administrative and academic tasks within educational institutions. This system enables efficient management of student-related data, including enrollment, attendance, grades, and personal information. The SDMS provides functionalities for students, teachers, and administrators, offering a user- friendly interface for data entry, retrieval, and reporting.

Key features include online registration, real-time attendance tracking, grade and performance management, and communication tools for notifications and updates. The system ensures data security and accessibility through role-based permissions, while its integration capabilities support seamless connection with other institutional software. By reducing manual processes, the SDMS minimizes errors, enhances decision-making, and improves overall efficiency.

This system is particularly beneficial for schools, colleges, and universities aiming to adopt digital transformation, fostering better collaboration among stakeholders and improving the overall academic experience.

TABLE OF CONTENTS

Chapter 1 INTRODUCTION

1.1 INTRODUCTION

The **Student Database Management System (SDMS)** developed using **NetBeans**, **Java**, and **SQL** is a robust and efficient application designed to streamline the management of student-related data for educational institutions. This system leverages the power of Java for building a secure, interactive, and user-friendly interface while utilizing SQL for reliable and scalable database management. NetBeans serves as the integrated development environment (IDE) to facilitate seamless development, debugging, and deployment of the application.

The SDMS offers a centralized platform for managing key student information, including personal details, academic records, attendance, and course enrollment. The use of SQL ensures that the data is stored securely and can be accessed or modified with precision and speed. Java's object-oriented capabilities enable modular development, making the system highly maintainable and adaptable to future enhancements.

Key functionalities of this system include efficient data entry, real-time updates, automated report generation, and secure user authentication. The integration of Java and SQL ensures cross-platform compatibility and high performance, making the system suitable for schools, colleges, and universities of varying scales.

This project demonstrates how modern programming tools and technologies can be combined to address real-world challenges in education management, providing a practical and scalable solution for enhancing institutional operations.

1.2 OBJECTIVES

2 **Efficient Data Management**
To provide a centralized platform for storing, organizing, and retrieving student information such as personal details, academic records, attendance, and course enrollments.

3 **Automation of Administrative Tasks**
To automate repetitive and time-consuming tasks like data entry, report generation, and record updates, thereby reducing manual effort and minimizing errors.

4 **User-Friendly Interface**
To offer a simple and intuitive interface for administrators, teachers, and students, ensuring ease of use and accessibility for all stakeholders.

5 **Data Accuracy and Integrity**
To ensure that all student records are accurate, consistent, and up-to-date by utilizing SQL for secure and reliable database management.

6 **Real-Time Access and Updates**
To enable real-time access to student data and facilitate quick updates, ensuring the system reflects the most recent information.

1.3 MODULES

- **User Management Module**

The User Management Module is responsible for handling user roles and access levels within the system, such as administrators, teachers, and students. It includes functionalities for user registration, login, and authentication to ensure secure access to the system. Role-based permissions are implemented to restrict access to certain features and data, ensuring that users interact with the system according to their designated roles.

2. **Student Information Management Module**

This module serves as the core repository for all student-related information. It maintains detailed records of each student, including personal details, academic history, and contact information. Administrators and authorized users can add, update, or delete student records, ensuring the database remains accurate and up-to-date

3. Admin Dashboard Module

The Admin Dashboard Module provides administrators with an overview of key metrics, such as the total number of students, attendance rates, and fee collection statuses. It offers a centralized control panel to manage system settings, monitor operations, and oversee the performance of all modules, ensuring the system runs smoothly.
Each module integrates seamlessly, creating a comprehensive and efficient solution for managing student data and institutional operations.

4. User Login Module

The User Login Module is a critical component of the Student Database Management System, designed to ensure secure access to the platform. This module manages the authentication and authorization of users, including administrators, teachers, and students. It provides a login interface where users enter their credentials, such as usernames and passwords, to access the system.

5 Database Module

The Database Module is the backbone of the Student Database Management System, responsible for storing and managing all data related to donors, recipients, blood inventory, and transactions. This module ensures data integrity, security, and efficient retrieval through structured queries and indexing. It uses SQL for database management, supporting complex queries and reporting needs.

Chapter 2 SURVEY OF TECHNOLOGIES

2.1 SOFTWARE DESCRIPTION

The **Student Database Management System (SDMS)** is a desktop-based application developed to manage and streamline student-related data and operations within educational institutions. The system is built using **Java** as the primary programming language, with **NetBeans** as the Integrated Development Environment (IDE) for development and debugging. **SQL** serves as the database backend, ensuring reliable, scalable, and efficient data storage and retrieval.

This software is designed to handle a wide range of administrative tasks, including student enrollment, attendance tracking, grade management, and fee tracking. Its modular design ensures that each feature operates as an independent yet interconnected component, enhancing system flexibility and maintainability.

2.2 LANGUAGES

The system is primarily developed using java as the primary language for server-side scripting, providing dynamic content generation and database connectivity and sql for managing and manipulatingthe relational database that stores all data**.**

2.2.1 Java

Purpose: The primary programming language used for developing the application logic and creating a user-friendly graphical user interface (GUI).

Role: Java's object-oriented nature ensures modular development, making the system maintainable and scalable. Tools like Java Swing or JavaFX are used to design the interface.

Advatages:

Platform Independence
Java is a platform-independent language, meaning the Student Database Management System (SDMS) can run on any operating system (Windows, macOS, Linux) without modification. As long as the system has a Java Virtual Machine (JVM) installed, the SDMS can function seamlessly, providing flexibility in deployment and use.

2.2.2 SQL

Role: SQL (Structured Query Language) is used for managing and manipulating the relational database that stores all data**.**

Usage: All information related to student data and transactions in thes ystem are stored and implemented as a relational schema using SQL.

Advantages:

- **Efficient Data Management**: SQL allows for efficient querying, updating, and management of large datas

Tools and Technologies Supporting Java

- **NetBeans IDE**: Used for developing, debugging, and testing the SDMS.
- **Apache Maven** (optional): To manage dependencies and build automation.
- **Java 8+**: Recommended version to leverage modern language features like streams and lambda expressions for cleaner code.

Chapter 3 REQUIREMENTS AND ANALYSIS

3.1 REQUIREMENT SPECIFICATION

3.1.1 Functional Requirements

User Authentication and Authorization

- User Registration and Login: Allow users to register, create accounts, and log in securely.
- Role-Based Access Control: Assign specific permissions based on user roles (student, teacher).

Student Information Management

- The system must allow administrators and authorized users to add, update, delete, and view student records.
- Student records should include personal information, academic history, contact details, and enrollment status.
- The system must provide search and filter capabilities to quickly retrieve student information by name, ID, or other attributes.

Backup and Data Recovery

- The system must regularly back up all data to prevent loss due to system failures.
- Users should be able to restore data from backups when necessary.
- The system must maintain data integrity, ensuring no information is corrupted or lost.

User Interface (UI)

- The system must provide a user-friendly interface for all stakeholders (students, teachers, administrators).
- The interface should be intuitive, responsive, and easy to navigate.

- The system should include forms for data entry (e.g., student registration, course enrollment), and interfaces for viewing and updating records.

Admin Dashboard Module

- The Admin Dashboard Module is a central feature of the Student Database Management System (SDMS) designed to provide administrators with a comprehensive overview of the system's operations and key performance metrics.
- Management Tools: The module serves as the main control panel where administrators can monitor, manage, and access various system features. It plays a crucial role in streamlining administrative tasks and ensuring smooth system operations.

3.1.2 Non-Functional Requirements

1. Performance

- **Response Time**: The system must provide quick responses for all user interactions, such as loading student records, processing transactions, and generating reports. Average response time should be under 3 seconds for most operations.
- **Scalability**: The system should be capable of handling an increasing number of users and data (e.g., students, courses, and transactions) without a significant decrease in performance. It must be able to scale both vertically (upgrading the existing hardware) and horizontally (adding additional servers if needed).
- **Concurrency**: The system must support concurrent access by multiple users, including administrators, teachers, and students, without performance degradation or data conflicts.

2. Availability

- The system must be available for use 24/7, with minimal downtime. Scheduled maintenance should be done during off-peak hours and should not exceed more than a few hours per month.
- The system should ensure high availability by utilizing redundant infrastructure, load balancing, and failover mechanisms.

3. Reliability

- The SDMS must be reliable and ensure that data is consistently available without corruption or loss. The system should ensure accuracy in all operations, such as attendance tracking, grade management, and fee processing.
- **Error Handling**: The system should detect and handle errors gracefully, providing users with meaningful error messages when something goes wrong, rather than causing a system crash.

4. Security

- **Data Encryption**: Sensitive data such as student records, grades, and payment information must be encrypted during transmission (e.g., using HTTPS) and while at rest (using database-level encryption).
- **Authentication and Authorization**: The system must ensure only authorized users can access specific features based on their roles (admin, teacher, student). This includes strong password policies, multi-factor authentication (optional), and secure session management.
- **Audit Trails**: The system should log critical actions, such as changes to student records or grades, along with the user responsible, to provide accountability and traceability.

5. Maintainability

- The system should be designed for easy maintenance and upgrades. The codebase should be well-documented and follow industry best practices for structure and organization, allowing new features or bug fixes to be implemented without significant effort.
- **Modular Architecture**: The system should have a modular design, making it easy to replace or upgrade individual components without affecting the entire system.
- **Error Logs and Monitoring**: The system should log errors, warnings, and other system events, allowing administrators to quickly identify and resolve issues.

6. Usability

- The system should provide an intuitive, user-friendly interface for all users, including administrators, teachers, and students. User interfaces should be simple and easy to navigate with clear instructions and visual feedback.
- **Cross-Browser and Cross-Platform Compatibility**: If the SDMS has a web interface, it must work across popular browsers (Chrome, Firefox, Safari) and on various devices (desktops, tablets, smartphones) without issues.
- **Localization**: The system should be capable of supporting multiple languages or regional settings to accommodate users from different locations (if needed).

7. Interoperability

- The SDMS should be able to integrate with other existing systems within the institution, such as email systems, SMS services, and external payment gateways. The system should also support standard data exchange formats (e.g., CSV, XML, JSON) for data import/export.
- The system should be designed to easily integrate with other third-party applications such as Learning Management Systems (LMS) or library management tools if required in the future.

8. Backup and Data Recovery

- The system must provide automatic backup of all critical data at regular intervals (e.g., daily or weekly).
- Backup data should be stored in secure, remote locations and be easily retrievable in case of data loss or system failure.
- In the event of a disaster or failure, the system must ensure that data can be recovered within a predefined period (e.g., within 24 hours).

9. Compliance

- The system must comply with relevant data protection laws and regulations, such as the **General Data Protection Regulation (GDPR)**, **Family Educational Rights and Privacy Act (FERPA)**, or any local educational privacy laws, to ensure that students' personal data is handled securely and ethically.
- The system should provide users with the ability to request their data or request its deletion, in compliance with legal requirements.

10. Portability

- If required, the SDMS should be portable, meaning it can be deployed on different types of systems or platforms with minimal modification. This can include switching between cloud-based environments, local servers, or hybrid cloud solutions.

3.2 HARDWARE AND SOFTWARE REQUIREMENTS

Hardware Requirements:

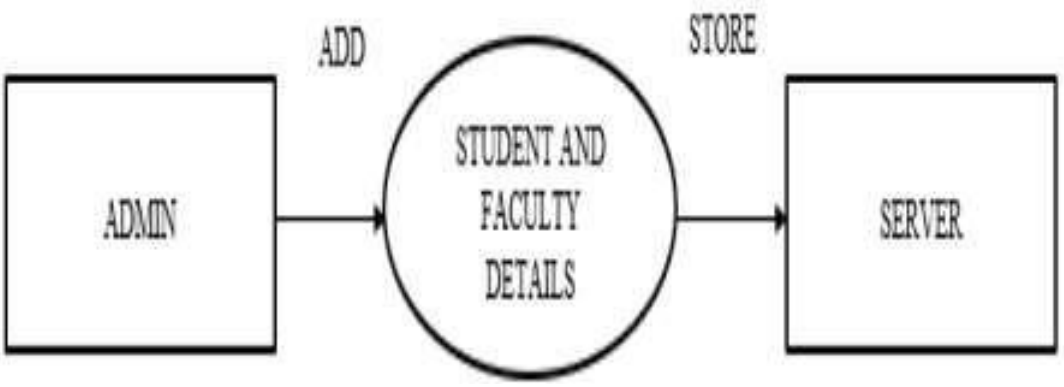
- Desktop PC or Laptop: A reliable desktop PC or laptop to host the Blood Donation Bank Management System.
- Processor: Intel® Core TM i3-6006U CPU @ 2.00GHz or equivalent for efficient processing.
- RAM: 4.00 GB RAM to handle concurrent user requests and database operations.
- System Architecture: 64-bit operating system, x64 based processor for optimal performance.
- Monitor Resolution: 1024 x 768 monitor resolution for clear display of the system interface.
- Input Devices: Keyboard and Mouse for user interaction.
- Server with high processing power and ample storage capacity
- Reliable network infrastructure

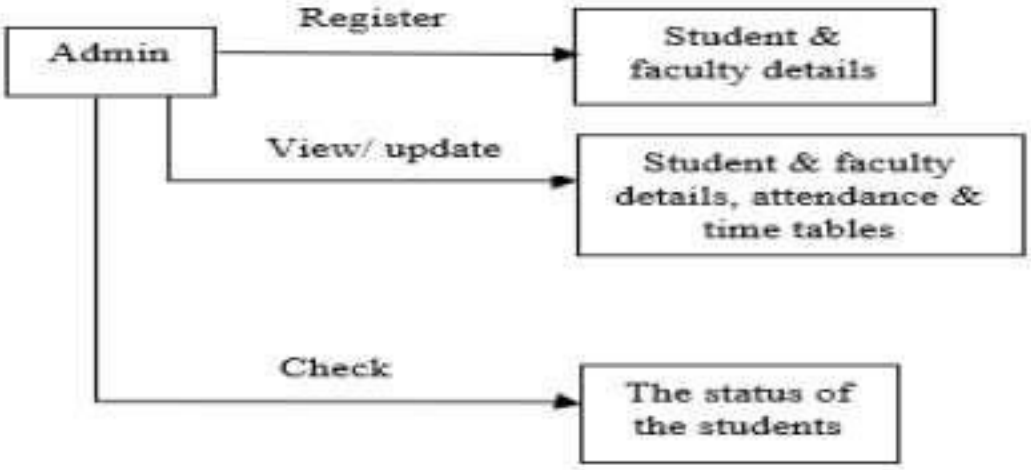
Software Requirements:

- Operating System: Windows 10
- Code editor :NetBeans
- Front End: JAVA
- Back End: MySQL
- Middleware: JDBC , JDK 8+
- Version Control: Git

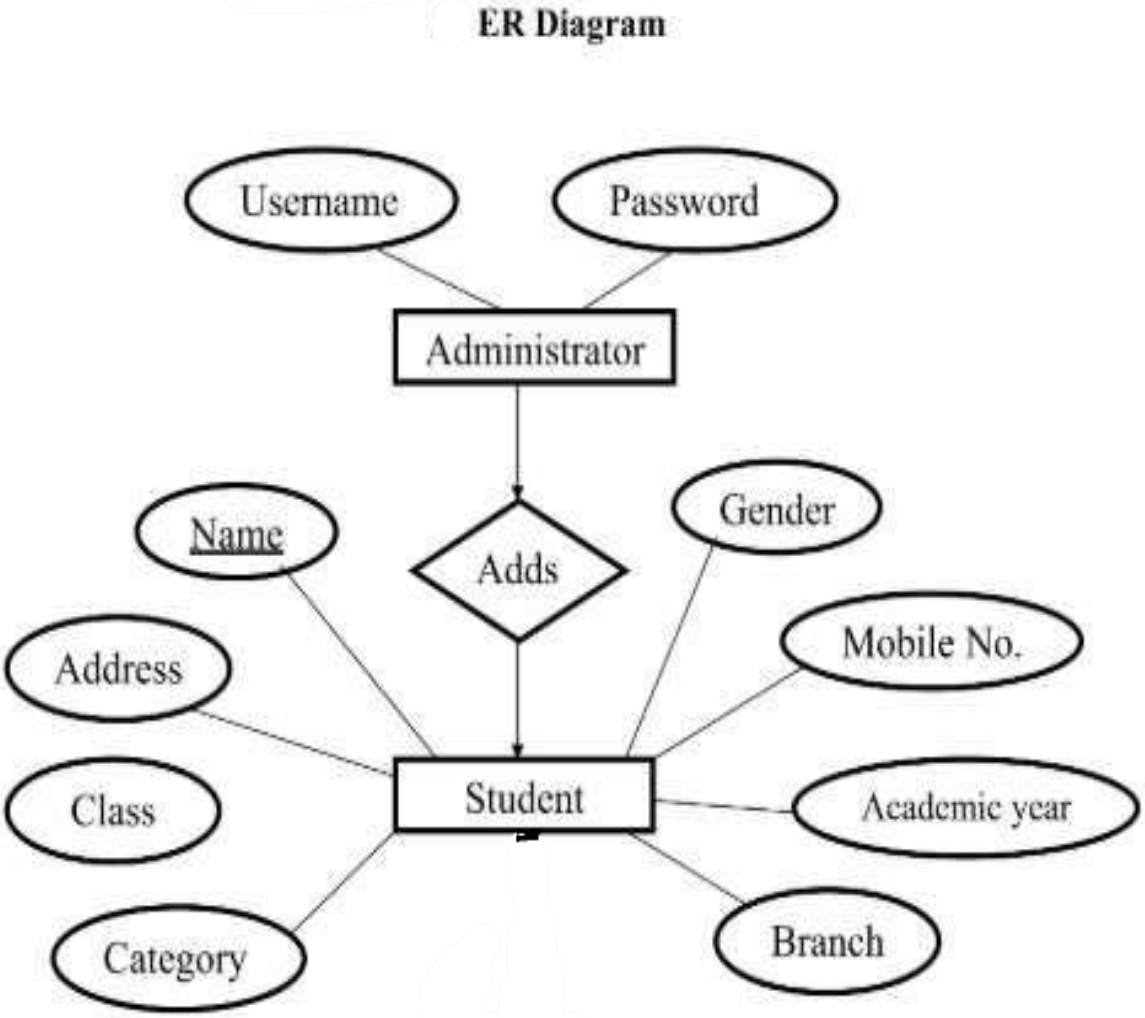
3.3 ARCHITECTURE DIAGRAM

A visual diagram that provides an overall view of the student database management system, identifying the external entities that interact with the system and the majordata flows between these entities and the system.





3.4 ER DIAGRAM



3.5 NORMALISATION

Normalization is the process of organizing data in a database to reduce redundancy and improve data integrity. It involves dividing a database into two or more tables and defining relationships between the tables. The steps to normalize a database table for Blood Bank Management System are as follows.

Raw Database

Attribute	Datatype	Example Value
student_id	INT	101
student_name	VARCHAR(50)	John Doe
student_email	VARCHAR(100)	john.doe@example.com
student_phone_no	VARCHAR(15)	9876543210
student_address	VARCHAR(100)	123 Elm Street, NY
course_id	INT	201
course_name	VARCHAR(50)	Data Structures
course_credits	INT	4
faculty_id	INT	301
faculty_name	VARCHAR(50)	Dr. Jane Smith
department_id	INT	401
department_name	VARCHAR(50)	Computer Science
enrollment_id	INT	501
enrollment_date	DATE	2024-11-20
semester	VARCHAR(20)	Fall 2024
grade	CHAR(2)	A
classroom_id	INT	601
classroom_name	VARCHAR(50)	Room A101

First Normal Form (1NF)

First Normal Form (1NF) establishes the foundational structure for organizing data in a relational database. It ensures that each cell within a table contains a single, indivisible value, commonly referred to as atomic values. This rule eliminates the presence of multiple values or lists within a single cell, thereby simplifying data manipulation and retrieval. Additionally, 1NF mandates that each record in the table is unique and can be distinctly identified by a primary key. The primary key serves as a unique identifier for each row, preventing duplicate records and ensuring data integrity within the table.

Raw Table Before 1NF:

StudentID	StudentName	StudentPhoneNo	CourseName	FacultyName	Grade
101	John Doe	9876543210	Data Structures, OOP	Dr. Jane Smith	A, B

Table After 1NF:

StudentID	StudentName	StudentPhoneNo	CourseName	FacultyName	Grade
101	John Doe	9876543210	Data Structures	Dr. Jane Smith	A
101	John Doe	9876543210	OOP	Dr. Jane Smith	B

Second Normal Form (2NF)

Second Normal Form (2NF) ensures that a table is in First Normal Form (1NF) and eliminates partial dependencies. All non-prime attributes must depend on the entire primary key, not just a part of it. This is achieved by decomposing tables with composite primary keys, ensuring every attribute relates fully to the key, reducing redundancy and improving data consistency.

1. Student Table:

StudentID	StudentName	StudentPhoneNo
101	John Doe	9876543210

2. Course Table:

CourseName	FacultyName
Data Structures	Dr. Jane Smith
OOP	Dr. Jane Smith

3,Enrollment Table:

StudentID	CourseName	Grade
101	Data Structures	A
101	OOP	B

Chapter 4 PROGRAM CODE

1. LOGIN PAGE

```
package student;

import db.MyConnection;
import java.util.logging.Level;
import javax.swing.JOptionPane;

/**
 *
 * @author jayas
 */
public class login extends javax.swing.JFrame {

    /**
     * Creates new form login
     */
    public login() {
        initComponents();
    }

    /**
     * This method is called from within the constructor to initialize the
form.
     * WARNING: Do NOT modify this code. The content of this method is always
     * regenerated by the Form Editor.
     */
    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
    private void initComponents() {

        jPanel1 = new javax.swing.JPanel();
        jLabel1 = new javax.swing.JLabel();
        jLabel2 = new javax.swing.JLabel();
        jLabel3 = new javax.swing.JLabel();
        jTextField1 = new javax.swing.JTextField();
        jPasswordField1 = new javax.swing.JPasswordField();
        jButton1 = new javax.swing.JButton();
        jButton2 = new javax.swing.JButton();
```

```
        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
        setUndecorated(true);
        addWindowListener(new java.awt.event.WindowAdapter() {
            public void windowOpened(java.awt.event.WindowEvent evt) {
                formWindowOpened(evt);
            }
        });

        jPanel1.setBackground(new java.awt.Color(255, 0, 255));

        jLabel1.setFont(new java.awt.Font("Segoe UI", 1, 48)); // NOI18N
        jLabel1.setText("Login");

        jLabel2.setFont(new java.awt.Font("Segoe UI", 1, 24)); // NOI18N
        jLabel2.setText("Username:");

        jLabel3.setFont(new java.awt.Font("Segoe UI", 1, 24)); // NOI18N
        jLabel3.setText("Password:");

        jTextField1.setFont(new java.awt.Font("Segoe UI", 1, 14)); // NOI18N

        jPasswordField1.setFont(new java.awt.Font("Segoe UI", 0, 24)); //
NOI18N

        jButton1.setText("Login");
        jButton1.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton1ActionPerformed(evt);
            }
        });

        jButton2.setText("Cancel");
        jButton2.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton2ActionPerformed(evt);
            }
        });

        javax.swing.GroupLayout jPanel1Layout = new
javax.swing.GroupLayout(jPanel1);
        jPanel1.setLayout(jPanel1Layout);
        jPanel1Layout.setHorizontalGroup(
```

```
jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel1Layout.createSequentialGroup()
                .addContainerGap(258, Short.MAX_VALUE)
                .addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED_SIZE,
201, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(257, 257, 257))
            .addGroup(jPanel1Layout.createSequentialGroup()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
                    .addGroup(jPanel1Layout.createSequentialGroup()
                        .addGap(140, 140, 140)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING, false)
                            .addComponent(jLabel2,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
                            .addComponent(jLabel3,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
                        .addGap(30, 30, 30)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING, false)
                            .addComponent(jTextField1)
                            .addComponent(jPasswordField1,
javax.swing.GroupLayout.DEFAULT_SIZE, 195, Short.MAX_VALUE)))
                    .addGroup(jPanel1Layout.createSequentialGroup()
                        .addGap(212, 212, 212)
                        .addComponent(jButton1)
                        .addGap(50, 50, 50)
                        .addComponent(jButton2)))
                .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
        );
        jPanel1Layout.setVerticalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addGap(62, 62, 62)
                .addComponent(jLabel1)
```

```
                .addGap(22, 22, 22)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
BASELINE)
                    .addComponent(jLabel2,
javax.swing.GroupLayout.PREFERRED_SIZE, 31,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(jTextField1,
javax.swing.GroupLayout.PREFERRED_SIZE, 31,
javax.swing.GroupLayout.PREFERRED_SIZE))
                .addGap(34, 34, 34)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
BASELINE)
                    .addComponent(jLabel3)
                    .addComponent(jPasswordField1,
javax.swing.GroupLayout.PREFERRED_SIZE, 32,
javax.swing.GroupLayout.PREFERRED_SIZE))
                .addGap(63, 63, 63)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
BASELINE)
                    .addComponent(jButton1)
                    .addComponent(jButton2))
                .addContainerGap(65, Short.MAX_VALUE))
        );

        javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
        getContentPane().setLayout(layout);
        layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()
                .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addContainerGap())
        );
        layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jPanel1, javax.swing.GroupLayout.Alignment.TRAILING,
```

```
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
        );

        pack();
        setLocationRelativeTo(null);
    }// </editor-fold>

    private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
        System.exit(0);
    }

    private void formWindowOpened(java.awt.event.WindowEvent evt) {

    }

    private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
        if(!isEmpty()){
            java.sql.Connection con = MyConnection.getConnection();
            PreparedStatement ps;

            String username = jTextField1.getText();
            String password = String.valueOf(jPasswordField1.getPassword());

            try {
                ps = (PreparedStatement) con.prepareStatement("select * from
admin where username = ? and password = ?");
                ps.setString(1, username);
                ps.setString(2, password);
                ResultSet rs = ps.executeQuery();
                if (rs.next()) {
                    JOptionPane.showMessageDialog(this, "Login success");
                    Home home = new Home();
                    home.setVisible(true);
                    home.pack();
                    this.dispose();
                } else {
                    JOptionPane.showMessageDialog(this, "Invalid username or
password", "Login Failed",2);
                            }


            } catch (java.sql.SQLException ex) {
```

```
java.util.logging.Logger.getLogger(login.class.getName()).log(Level.SEVERE,
null, ex);
            }
               //SwingUtilities.invokeLater(() ->new Home().setVisible(true));
        }
    }
  public  boolean isEmpty(){
    if(jTextField1.getText().isEmpty()){
        JOptionPane.showMessageDialog(this,"User is missing");
             return false;
    }if(String.valueOf(jPasswordField1.getPassword()).isEmpty()){
        JOptionPane.showMessageDialog(this,"password missing");
             return false;
    }
    return true;

  }
    /**
     * @param args the command line arguments
     */
    public static void main(String args[]) {
        /* Set the Nimbus look and feel */
        //<editor-fold defaultstate="collapsed" desc=" Look and feel setting
code (optional) ">
        /* If Nimbus (introduced in Java SE 6) is not available, stay with the
default look and feel.
         * For details see
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
         */
        try {
            for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
                if ("Nimbus".equals(info.getName())) {
                    javax.swing.UIManager.setLookAndFeel(info.getClassName());
                    break;
                }
            }
        } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(login.class.getName()).log(java.util.loggin
g.Level.SEVERE, null, ex);
        } catch (InstantiationException ex) {
```

```
java.util.logging.Logger.getLogger(login.class.getName()).log(java.util.loggin
g.Level.SEVERE, null, ex);
        } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(login.class.getName()).log(java.util.loggin
g.Level.SEVERE, null, ex);
        } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(login.class.getName()).log(java.util.loggin
g.Level.SEVERE, null, ex);
        }
        //</editor-fold>

        /* Create and display the form */
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new login().setVisible(true);
            }
        });
    }

    // Variables declaration - do not modify
    private javax.swing.JButton jButton1;
    private javax.swing.JButton jButton2;
    private javax.swing.JLabel jLabel1;
    private javax.swing.JLabel jLabel2;
    private javax.swing.JLabel jLabel3;
    private javax.swing.JPanel jPanel1;
    private javax.swing.JPasswordField jPasswordField1;
    private javax.swing.JTextField jTextField1;
    // End of variables declaration
}
```

2 . LOADING PAGE

```
package  student;
import java.util.Timer;
import  java.util.TimerTask;
import javax.swing.SwingUtilities;

/**
 *
 * @author jayas
 */
public class loading extends javax.swing.JFrame {

    /**
     * Creates new form loading
     */
    public loading() {
        initComponents();
        initialize();
    }

    /**
     * This method is called from within the constructor to initialize the
form.
     * WARNING: Do NOT modify this code. The content of this method is always
     * regenerated by the Form Editor.
     */
    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
    private void initComponents() {

        jPanel1 = new javax.swing.JPanel();
        jLabel1 = new javax.swing.JLabel();
        jProgressBar1 = new javax.swing.JProgressBar();
        jLabel2 = new javax.swing.JLabel();

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

        jPanel1.setBackground(new java.awt.Color(255, 0, 255));

        jLabel1.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/student/student-management-
system.png"))); // NOI18N
```

```
        jLabel1.setMaximumSize(new java.awt.Dimension(650, 398));

        jLabel2.setFont(new java.awt.Font("Times New Roman", 1, 36)); //
NOI18N
        jLabel2.setText("Loading...");

        javax.swing.GroupLayout jPanel1Layout = new
javax.swing.GroupLayout(jPanel1);
        jPanel1.setLayout(jPanel1Layout);
        jPanel1Layout.setHorizontalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addGap(402, 402, 402)
                .addComponent(jLabel1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addGap(372, 372, 372))
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addGap(414, 414, 414)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.
LEADING)
                    .addComponent(jLabel2,
javax.swing.GroupLayout.PREFERRED_SIZE, 186,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(jProgressBar1,
javax.swing.GroupLayout.PREFERRED_SIZE, 629,
javax.swing.GroupLayout.PREFERRED_SIZE))
                .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
        );
        jPanel1Layout.setVerticalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addGap(72, 72, 72)
                .addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED_SIZE,
496, javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                .addComponent(jLabel2, javax.swing.GroupLayout.PREFERRED_SIZE,
43, javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                .addComponent(jProgressBar1,
javax.swing.GroupLayout.PREFERRED_SIZE, 23,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addContainerGap(92, Short.MAX_VALUE))
        );

        javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
        getContentPane().setLayout(layout);
        layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        );
        layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup()
                .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addContainerGap())
        );

        pack();
        setLocationRelativeTo(null);
    }// </editor-fold>
    private void initialize(){
        Timer timer = new Timer();
        timer.schedule(new TimerTask(){

            public void run(){
                dispose();
                SwingUtilities.invokeLater(() ->new Home().setVisible(true));
            }
        },3000);

    }
    /**
     * @param args the command line arguments
     */
    public static void main(String args[]) {
```

```
        /* Set the Nimbus look and feel */
        //<editor-fold defaultstate="collapsed" desc=" Look and feel setting
code (optional) ">
        /* If Nimbus (introduced in Java SE 6) is not available, stay with the
default look and feel.
         * For details see
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
         */
        try {
            for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
                if ("Nimbus".equals(info.getName())) {
                    javax.swing.UIManager.setLookAndFeel(info.getClassName());
                    break;
                }
            }
        } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(loading.class.getName()).log(java.util.logg
ing.Level.SEVERE, null, ex);
        } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(loading.class.getName()).log(java.util.logg
ing.Level.SEVERE, null, ex);
        } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(loading.class.getName()).log(java.util.logg
ing.Level.SEVERE, null, ex);
        } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(loading.class.getName()).log(java.util.logg
ing.Level.SEVERE, null, ex);
        }
        //</editor-fold>

        /* Create and display the form */
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new loading().setVisible(true);
            }
        });
    }
// Variables declaration - do not modify
```

```
    private javax.swing.JLabel jLabel1;
    private javax.swing.JLabel jLabel2;
    private javax.swing.JPanel jPanel1;
    private javax.swing.JProgressBar jProgressBar1;
    // End of variables declaration
}
```

3. HOME PAGE

```
package student;

import java.awt.Color;
import javax.swing.JOptionPane;
import javax.swing.table.DefaultTableModel;
import java.io.File;
import javax.swing.filechooser.FileNameExtensionFilter;
import javax.swing.ImageIcon;
import java.awt.Image;
import java.sql.SQLException;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.swing.JFileChooser;
import java.lang.NumberFormatException;


/**
 *
 * @author jayas
 */
public class Home extends javax.swing.JFrame {
    Student student=new Student();
    int xx,xy;
    private DefaultTableModel model;
    private int rowIndex;
    private String imagePath;
    public Home() {
        initComponents();
    }

    /**
     * This method is called from within the constructor to initialize
the form.
     * WARNING: Do NOT modify this code. The content of this method is
```

```
always
     * regenerated by the Form Editor.
     */
    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
    private void initComponents() {

        jPanel1 = new javax.swing.JPanel();
        jPanel2 = new javax.swing.JPanel();
        jLabel1 = new javax.swing.JLabel();
        jTabbedPane1 = new javax.swing.JTabbedPane();
        jPanel3 = new javax.swing.JPanel();
        jPanel4 = new javax.swing.JPanel();
        jTextField1 = new javax.swing.JTextField();
        jTextField2 = new javax.swing.JTextField();
        jComboBox1 = new javax.swing.JComboBox<>();
        jTextField3 = new javax.swing.JTextField();
        jTextField4 = new javax.swing.JTextField();
        jTextField5 = new javax.swing.JTextField();
        jTextField6 = new javax.swing.JTextField();
        jTextField7 = new javax.swing.JTextField();
        jTextField8 = new javax.swing.JTextField();
        jTextField9 = new javax.swing.JTextField();
        jLabel2 = new javax.swing.JLabel();
        jLabel3 = new javax.swing.JLabel();
        jLabel4 = new javax.swing.JLabel();
        jLabel5 = new javax.swing.JLabel();
        jLabel6 = new javax.swing.JLabel();
        jLabel7 = new javax.swing.JLabel();
        jLabel8 = new javax.swing.JLabel();
        jLabel9 = new javax.swing.JLabel();
        jLabel10 = new javax.swing.JLabel();
        jLabel11 = new javax.swing.JLabel();
        jPanel6 = new javax.swing.JPanel();
        jButton1 = new javax.swing.JButton();
        jLabel12 = new javax.swing.JLabel();
        jPanel7 = new javax.swing.JPanel();
        jLabelImage = new javax.swing.JLabel();
        jPanel5 = new javax.swing.JPanel();
        jPanel8 = new javax.swing.JPanel();
        jPanel9 = new javax.swing.JPanel();
        jScrollPane1 = new javax.swing.JScrollPane();
        jTable1 = new javax.swing.JTable();
```

```
        jButton4 = new javax.swing.JButton();
        jButton5 = new javax.swing.JButton();
        jButton8 = new javax.swing.JButton();
        jButton9 = new javax.swing.JButton();


setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

        jPanel1.setBackground(new java.awt.Color(255, 0, 255));

        jPanel2.setBackground(new java.awt.Color(255, 255, 255));
        jPanel2.addMouseMotionListener(new
java.awt.event.MouseMotionAdapter() {
            public void mouseDragged(java.awt.event.MouseEvent evt) {
                jPanel2MouseDragged(evt);
            }
        });

        jLabel1.setFont(new java.awt.Font("Times New Roman", 1, 36)); //
NOI18N
        jLabel1.setText("STUDENT MANAGEMENT SYSTEM");

        javax.swing.GroupLayout jPanel2Layout = new
javax.swing.GroupLayout(jPanel2);
        jPanel2.setLayout(jPanel2Layout);
        jPanel2Layout.setHorizontalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEAD
ING)
            .addGroup(jPanel2Layout.createSequentialGroup()
                .addGap(22, 22, 22)
                .addComponent(jLabel1,
javax.swing.GroupLayout.PREFERRED_SIZE, 770,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
        );
        jPanel2Layout.setVerticalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEAD
ING)
             .addGroup(jPanel2Layout.createSequentialGroup()
                .addGap(17, 17, 17)
```

```
                .addComponent(jLabel1,
javax.swing.GroupLayout.PREFERRED_SIZE, 66,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addContainerGap(17, Short.MAX_VALUE))
        );

        jPanel3.setBackground(new java.awt.Color(255, 255, 255));

        jPanel4.setBackground(new java.awt.Color(255, 0, 255));
        jPanel4.setBorder(javax.swing.BorderFactory.createLineBorder(new
java.awt.Color(204, 0, 204), 4));

        jTextField1.setEditable(false);
        jTextField1.setBackground(new java.awt.Color(204, 204, 204));
        jTextField1.addActionListener(new
java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt)
{
                jTextField1ActionPerformed(evt);
            }
        });


        jComboBox1.setFont(new java.awt.Font("Times New Roman", 0, 16));
// NOI18N
        jComboBox1.setModel(new javax.swing.DefaultComboBoxModel<>(new
String[] { "Male", "Female" }));

        jTextField5.addActionListener(new
java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt)
{
                jTextField5ActionPerformed(evt);
            }
        });

        jTextField5.addKeyListener(new java.awt.event.KeyAdapter() {
            public void keyTyped(java.awt.event.KeyEvent evt) {
                jTextField5KeyTyped(evt);
            }
        });

        jLabel2.setFont(new java.awt.Font("Segoe UI", 0, 14)); // NOI18N
        jLabel2.setText("Student ID");
```

```
        jLabel3.setFont(new java.awt.Font("Segoe UI", 0, 14)); // NOI18N
        jLabel3.setText("Student Name");

        jLabel4.setFont(new java.awt.Font("Segoe UI", 0, 14)); // NOI18N
        jLabel4.setText("Age");

        jLabel5.setFont(new java.awt.Font("Segoe UI", 0, 14)); // NOI18N
        jLabel5.setText("Gender");

        jLabel6.setFont(new java.awt.Font("Segoe UI", 0, 14)); // NOI18N
        jLabel6.setText("Email");

        jLabel7.setFont(new java.awt.Font("Segoe UI", 0, 14)); // NOI18N
        jLabel7.setText("Phone No");

        jLabel8.setFont(new java.awt.Font("Segoe UI", 0, 14)); // NOI18N
        jLabel8.setText("Father Name");

        jLabel9.setFont(new java.awt.Font("Segoe UI", 0, 14)); // NOI18N
        jLabel9.setText("Mother Name");

        jLabel10.setFont(new java.awt.Font("Segoe UI", 0, 14)); //
NOI18N
        jLabel10.setText("Address Line 1");

        jLabel11.setFont(new java.awt.Font("Segoe UI", 0, 14)); //
NOI18N
        jLabel11.setText("Addres Line 2");

        jPanel6.setBackground(new java.awt.Color(255, 51, 255));
        jPanel6.setBorder(javax.swing.BorderFactory.createLineBorder(new
java.awt.Color(204, 0, 204), 4));

        jButton1.setFont(new java.awt.Font("Segoe UI", 0, 18)); //
NOI18N
        jButton1.setText("Browse");
        jButton1.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt)
{
                jButton1ActionPerformed(evt);
            }
        });
```

```
        jLabel12.setFont(new java.awt.Font("Segoe UI", 0, 18)); //
NOI18N
        jLabel12.setText("IMAGE");

        jPanel7.setBackground(new java.awt.Color(204, 204, 204));

        javax.swing.GroupLayout jPanel7Layout = new
javax.swing.GroupLayout(jPanel7);
        jPanel7.setLayout(jPanel7Layout);
        jPanel7Layout.setHorizontalGroup(

jPanel7Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEAD
ING)
            .addComponent(jLabelImage,
javax.swing.GroupLayout.DEFAULT_SIZE, 134, Short.MAX_VALUE)
        );
        jPanel7Layout.setVerticalGroup(

jPanel7Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEAD
ING)
            .addComponent(jLabelImage,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        );

        javax.swing.GroupLayout jPanel6Layout = new
javax.swing.GroupLayout(jPanel6);
        jPanel6.setLayout(jPanel6Layout);
        jPanel6Layout.setHorizontalGroup(

jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEAD
ING)
            .addGroup(jPanel6Layout.createSequentialGroup()
                .addGap(21, 21, 21)

.addGroup(jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.Alig
nment.LEADING)
                    .addComponent(jLabel12,
javax.swing.GroupLayout.PREFERRED_SIZE, 67,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(jButton1))
                .addGap(105, 105, 105)
```

```
                .addComponent(jPanel7,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addContainerGap(107, Short.MAX_VALUE))
        );
        jPanel6Layout.setVerticalGroup(

jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEAD
ING)
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel6Layout.createSequentialGroup()
                .addGap(23, 23, 23)
                .addComponent(jLabel12)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addComponent(jButton1)
                .addGap(38, 38, 38))
            .addGroup(jPanel6Layout.createSequentialGroup()
                .addContainerGap()
                .addComponent(jPanel7,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addContainerGap())
        );

        javax.swing.GroupLayout jPanel4Layout = new
javax.swing.GroupLayout(jPanel4);
        jPanel4.setLayout(jPanel4Layout);
        jPanel4Layout.setHorizontalGroup(

jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEAD
ING)
            .addGroup(jPanel4Layout.createSequentialGroup()
                .addContainerGap()

.addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alig
nment.LEADING)
                    .addComponent(jPanel6,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                    .addGroup(jPanel4Layout.createSequentialGroup()
```

```
.addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alig
nment.LEADING, false)
                            .addComponent(jLabel4,
javax.swing.GroupLayout.PREFERRED_SIZE, 37,
javax.swing.GroupLayout.PREFERRED_SIZE)
                            .addComponent(jLabel6,
javax.swing.GroupLayout.PREFERRED_SIZE, 37,
javax.swing.GroupLayout.PREFERRED_SIZE)
                            .addComponent(jLabel2,
javax.swing.GroupLayout.PREFERRED_SIZE, 71,
javax.swing.GroupLayout.PREFERRED_SIZE)
                            .addComponent(jLabel3,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                            .addComponent(jLabel7,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                            .addComponent(jLabel5,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                            .addComponent(jLabel9)
                            .addComponent(jLabel10,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                            .addComponent(jLabel11)
                            .addComponent(jLabel8,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
                        .addGap(64, 64, 64)

.addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alig
nment.LEADING, false)
                            .addComponent(jTextField1)
                            .addComponent(jTextField2)
                            .addComponent(jTextField3)
                            .addComponent(jComboBox1,
javax.swing.GroupLayout.PREFERRED_SIZE, 100,
javax.swing.GroupLayout.PREFERRED_SIZE)
                            .addComponent(jTextField4)
                            .addComponent(jTextField5)
                            .addComponent(jTextField6)
                            .addComponent(jTextField7)
```

```
                            .addComponent(jTextField8)
                            .addComponent(jTextField9,
javax.swing.GroupLayout.DEFAULT_SIZE, 291, Short.MAX_VALUE))
                        .addGap(0, 0, Short.MAX_VALUE)))
                .addContainerGap())
        );
        jPanel4Layout.setVerticalGroup(

jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEAD
ING)
            .addGroup(jPanel4Layout.createSequentialGroup()
                .addGap(31, 31, 31)

.addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alig
nment.BASELINE)
                    .addComponent(jTextField1,
javax.swing.GroupLayout.PREFERRED_SIZE, 28,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(jLabel2))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alig
nment.BASELINE)
                    .addComponent(jTextField2,
javax.swing.GroupLayout.PREFERRED_SIZE, 28,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(jLabel3))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alig
nment.BASELINE)
                    .addComponent(jTextField3,
javax.swing.GroupLayout.PREFERRED_SIZE, 28,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(jLabel4))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alig
nment.BASELINE)
                    .addComponent(jComboBox1,
```

```
javax.swing.GroupLayout.PREFERRED_SIZE, 28,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(jLabel5))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alig
nment.BASELINE)
                    .addComponent(jTextField4,
javax.swing.GroupLayout.PREFERRED_SIZE, 28,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(jLabel6))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alig
nment.BASELINE)
                    .addComponent(jTextField5,
javax.swing.GroupLayout.PREFERRED_SIZE, 28,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(jLabel7))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alig
nment.BASELINE)
                    .addComponent(jTextField6,
javax.swing.GroupLayout.PREFERRED_SIZE, 28,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(jLabel8))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alig
nment.BASELINE)
                    .addComponent(jTextField7,
javax.swing.GroupLayout.PREFERRED_SIZE, 28,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(jLabel9))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alig
```

```
nment.BASELINE)
                    .addComponent(jTextField8,
javax.swing.GroupLayout.PREFERRED_SIZE, 28,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(jLabel10))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alig
nment.BASELINE)
                    .addComponent(jTextField9,
javax.swing.GroupLayout.PREFERRED_SIZE, 28,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(jLabel11))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                .addComponent(jPanel6,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addContainerGap())
        );

        jPanel5.setBackground(new java.awt.Color(255, 0, 255));
        jPanel5.setBorder(javax.swing.BorderFactory.createLineBorder(new
java.awt.Color(204, 0, 204), 4));

        javax.swing.GroupLayout jPanel8Layout = new
javax.swing.GroupLayout(jPanel8);
        jPanel8.setLayout(jPanel8Layout);
        jPanel8Layout.setHorizontalGroup(

jPanel8Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEAD
ING)
            .addGap(0, 0, Short.MAX_VALUE)
        );
        jPanel8Layout.setVerticalGroup(

jPanel8Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEAD
ING)
            .addGap(0, 44, Short.MAX_VALUE)
        );

        jPanel9.setBackground(new java.awt.Color(255, 0, 255));
```

```
        jPanel9.setBorder(javax.swing.BorderFactory.createLineBorder(new
java.awt.Color(153, 153, 153), 4));

        jTable1.setModel(new javax.swing.table.DefaultTableModel(
            new Object [][] {

            },
            new String [] {
                "Student ID", "Student Name", "Age", "Gender", "Email",
"Phone No", "Father Name", "Mother Name", "Address Line 1", "Address
Line 2", "imagePath"
            }
        ) {
            boolean[] canEdit = new boolean [] {
                false, false, false, false, false, false, false, false,
false, false, false
            };

            public boolean isCellEditable(int rowIndex, int columnIndex)
{
                return canEdit [columnIndex];
            }
        });

        jTable1.setRowHeight(30);
        jTable1.addMouseListener(new java.awt.event.MouseAdapter() {
            public void mouseClicked(java.awt.event.MouseEvent evt) {
                jTable1MouseClicked(evt);
            }
        });
        jScrollPane1.setViewportView(jTable1);

        javax.swing.GroupLayout jPanel9Layout = new
javax.swing.GroupLayout(jPanel9);
        jPanel9.setLayout(jPanel9Layout);
        jPanel9Layout.setHorizontalGroup(

jPanel9Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEAD
ING)
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel9Layout.createSequentialGroup()
                .addGap(0, 0, Short.MAX_VALUE)
                .addComponent(jScrollPane1,
javax.swing.GroupLayout.PREFERRED_SIZE, 832,
```

```
javax.swing.GroupLayout.PREFERRED_SIZE))
        );
        jPanel9Layout.setVerticalGroup(

jPanel9Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEAD
ING)
            .addComponent(jScrollPane1,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.DEFAULT_SIZE, 429, Short.MAX_VALUE)
        );

        jButton4.setFont(new java.awt.Font("Segoe UI", 0, 18)); //
NOI18N
        jButton4.setText("Add New");
        jButton4.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt)
{
                jButton4ActionPerformed(evt);
            }
        });

        jButton5.setFont(new java.awt.Font("Segoe UI", 0, 18)); //
NOI18N
        jButton5.setText("Update");
        jButton5.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt)
{
                jButton5ActionPerformed(evt);
            }
        });

        jButton8.setFont(new java.awt.Font("Segoe UI", 0, 18)); //
NOI18N
        jButton8.setText("delete");
        jButton8.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt)
{
                jButton8ActionPerformed(evt);
            }
        });

        jButton9.setFont(new java.awt.Font("Segoe UI", 0, 24)); //
NOI18N
```

```
        jButton9.setText("Logout");
        jButton9.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt)
{
                jButton9ActionPerformed(evt);
            }
        });

        javax.swing.GroupLayout jPanel5Layout = new
javax.swing.GroupLayout(jPanel5);
        jPanel5.setLayout(jPanel5Layout);
        jPanel5Layout.setHorizontalGroup(

jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEAD
ING)
             .addGroup(jPanel5Layout.createSequentialGroup()
                 .addContainerGap()

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alig
nment.LEADING)
                     .addComponent(jPanel8,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                     .addComponent(jPanel9,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
                 .addContainerGap())
             .addGroup(jPanel5Layout.createSequentialGroup()
                 .addGap(23, 23, 23)
                 .addComponent(jButton4)
                 .addGap(54, 54, 54)
                 .addComponent(jButton5)
                 .addGap(61, 61, 61)
                 .addComponent(jButton8)
                 .addGap(325, 325, 325)
                 .addComponent(jButton9)
                 .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
        );
        jPanel5Layout.setVerticalGroup(

jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEAD
ING)
```

```
            .addGroup(jPanel5Layout.createSequentialGroup()
                .addContainerGap()
                .addComponent(jPanel8,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                .addComponent(jPanel9,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 33,
Short.MAX_VALUE)

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alig
nment.BASELINE)
                    .addComponent(jButton4)
                    .addComponent(jButton5)
                    .addComponent(jButton8)
                    .addComponent(jButton9))
                .addGap(19, 19, 19))
        );

        javax.swing.GroupLayout jPanel3Layout = new
javax.swing.GroupLayout(jPanel3);
        jPanel3.setLayout(jPanel3Layout);
        jPanel3Layout.setHorizontalGroup(

jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEAD
ING)
            .addGroup(jPanel3Layout.createSequentialGroup()
                .addContainerGap()
                .addComponent(jPanel4,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(18, 18, 18)
                .addComponent(jPanel5,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addContainerGap())
```

```
        );
        jPanel3Layout.setVerticalGroup(

jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEAD
ING)
            .addGroup(jPanel3Layout.createSequentialGroup()
                .addContainerGap()

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alig
nment.LEADING)
                    .addComponent(jPanel4,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                    .addComponent(jPanel5,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
                .addContainerGap())
        );

        jTabbedPane1.addTab("Śṭūḍēṇṭ", jPanel3);

        javax.swing.GroupLayout jPanel1Layout = new
javax.swing.GroupLayout(jPanel1);
        jPanel1.setLayout(jPanel1Layout);
        jPanel1Layout.setHorizontalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEAD
ING)
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addContainerGap()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alig
nment.LEADING)
                    .addComponent(jPanel2,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                    .addComponent(jTabbedPane1,
javax.swing.GroupLayout.Alignment.TRAILING))
                .addContainerGap())
        );
        jPanel1Layout.setVerticalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEAD
```

```
ING)
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addContainerGap()
                .addComponent(jPanel2,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                .addComponent(jTabbedPane1,
javax.swing.GroupLayout.PREFERRED_SIZE, 639,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addContainerGap(44, Short.MAX_VALUE))
        );

        javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
        getContentPane().setLayout(layout);
        layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup()
                .addComponent(jPanel1,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addContainerGap())
        );
        layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        );

        pack();
    }// </editor-fold>

    public void init(){
        tableViewStudent();
        jTextField1.setText(String.valueOf(student.getMax()));

    }
```

```
    private void tableViewStudent(){
        student.getStudentValue(jTable1, "");
        model = (DefaultTableModel)jTable1.getModel();
        jTable1.setRowHeight(30);
        jTable1.setShowGrid(true);
        jTable1.setGridColor(Color.black);
        jTable1.setBackground(Color.white);

    }
     private void clearStudent(){

        jTextField1.setText(String.valueOf(student.getMax()));
        jTextField1.setText(null);
        jTextField2.setText(null);
        jTextField3.setText(null);
        jTextField4.setText(null);
        jTextField5.setText(null);
        jTextField6.setText(null);
        jTextField7.setText(null);
        jTextField8.setText(null);
        jTextField9.setText(null);
        jComboBox1.setSelectedIndex(0);
        jLabelImage.setIcon(null);
        jTable1.clearSelection();
        imagePath= null;

    }
    public boolean isEmptyStudent(){
        if(jTextField2.getText().isEmpty()){
            JOptionPane.showMessageDialog(this,"student name is
missing");
            return false;
        }

         if(jTextField3.getText().isEmpty()){
            JOptionPane.showMessageDialog(this,"Age  is missing");
            return false;
        }
         if(jTextField4.getText().isEmpty()){
            JOptionPane.showMessageDialog(this,"email  is missing");
            return false;
        }
         if(!jTextField4.getText().matches("^.+@.+\\..+$")){
            JOptionPane.showMessageDialog(this,"invalid mail id");
```

```
            return false;
        }
         if(jTextField5.getText().isEmpty()){
            JOptionPane.showMessageDialog(this,"phone NO  is missing");
            return false;
        }
         if(jTextField5.getText().length()>=15){
            JOptionPane.showMessageDialog(this,"not a vaild No");
            return false;
        }
         if(jTextField6.getText().isEmpty()){
            JOptionPane.showMessageDialog(this,"father's name is
missing");
            return false;
        }

         if(jTextField7.getText().isEmpty()){
            JOptionPane.showMessageDialog(this,"Mother's name is
missing");
            return false;
        }

         if(jTextField8.getText().isEmpty()){
            JOptionPane.showMessageDialog(this,"address line 1 is
missing");
            return false;
        }

         if(jTextField9.getText().isEmpty()){
            JOptionPane.showMessageDialog(this,"address line 2 is
missing");
            return false;
        }

         if(imagePath==null){
             JOptionPane.showMessageDialog(this,"please add your
image");
             return false;
         }
         return true;
    }
    private void jButton8ActionPerformed(java.awt.event.ActionEvent evt)
{
        clearStudent();
```

```
    }

    private void jButton9ActionPerformed(java.awt.event.ActionEvent evt)
```

```
{
        int a = JOptionPane.showConfirmDialog(this, "do you want to
logout now?","select",JOptionPane.YES_NO_OPTION);
       if(a==0){
           this.dispose();
       }
    }

    private void jPanel2MouseDragged(java.awt.event.MouseEvent evt) {
        int x=evt.getXOnScreen();
    int y=evt.getYOnScreen();
    this.setLocation(x-xx,y-xy );
    }

    private void jButton4ActionPerformed(java.awt.event.ActionEvent evt)
{
        if(isEmptyStudent()){
            if(!student.isEmailExist(jTextField4.getText())){
                if(!student.isPhoneExist(jTextField5.getText())){
                     int id = student.getMax();
            String name =jTextField2.getText();
            String age =jTextField3.getText();
            Object gender =jComboBox1.getSelectedItem();
            String email= jTextField4.getText();
            String phone  =jTextField5.getText();
            String father =jTextField6.getText();
            String mother =jTextField7.getText();
            String address1 =jTextField8.getText();
            String address2 =jTextField9.getText();
            student.insert(id, name, age, gender, email, phone, father,
mother, address1, address2, imagePath);
            jTable1.setModel(new DefaultTableModel(null,new
Object[]{"Student ID","Student Name","Age","Gender","Email","Phone
No","Father Name","Mother Name","Address Line 1","Address Line 2","Image
Path"}));
            student.getStudentValue(jTable1,"");
            clearStudent();

                }
                else{
                    JOptionPane.showMessageDialog(this,"This phone No
alredy exists");

                }
```

```
        }
            else{
                 JOptionPane.showMessageDialog(this,"This email alredy
exists");
            }


        }
    }

    private void jTextField5ActionPerformed(java.awt.event.ActionEvent
evt) {

    }

    private void jTextField5KeyTyped(java.awt.event.KeyEvent evt) {
         if(!Character.isDigit(evt.getKeyChar())){
            evt.consume();
        }
    }

    private void jButton1ActionPerformed(java.awt.event.ActionEvent evt)
{
        JFileChooser file = new JFileChooser();
        file.setCurrentDirectory(new
File(System.getProperty("user.home")));
        FileNameExtensionFilter filter = new
FileNameExtensionFilter("*.image","jpg","gif","png");
        file.addChoosableFileFilter(filter);
        int output =file.showSaveDialog(file);
        if(output==JFileChooser.APPROVE_OPTION){
            File selectFile =file.getSelectedFile();
            String path = selectFile.getAbsolutePath();
            jLabelImage.setIcon(imageAdjust(path,null));
            imagePath = path;
        }
            else{
             JOptionPane.showMessageDialog(this,"add image");
        }
    }
```

```
    private void jTextField1ActionPerformed(java.awt.event.ActionEvent
evt) {
        // TODO add your handling code here:
    }

    private void jButton5ActionPerformed(java.awt.event.ActionEvent evt)
{
        if(isEmptyStudent()){
            try {
                int id = Integer.parseInt(jTextField1.getText());
                String name =jTextField2.getText();
                String age =jTextField3.getText();
                Object gender =jComboBox1.getSelectedItem();
                String email= jTextField4.getText();
                String phone  =jTextField5.getText();
                String father =jTextField6.getText();
                String mother =jTextField7.getText();
                String address1 =jTextField8.getText();
                String address2 =jTextField9.getText();
                student.update(id, name, age, gender, email, phone,
father, mother, address1, address2, imagePath);
                jTable1.setModel(new DefaultTableModel(null,new
Object[]{"Student ID","Student Name","Age","Gender","Email","Phone
No","Father Name","Mother Name","Address Line 1","Address Line 2","Image
Path"}));
                student.getStudentValue(jTable1,"");
                clearStudent();
            } catch (SQLException ex) {
                Logger.getLogger(Home.class.getName()).log(Level.SEVERE,
null, ex);

            }
        }
    }

    private void jTable1MouseClicked(java.awt.event.MouseEvent evt) {
        model = (DefaultTableModel) jTable1.getModel();
        rowIndex = jTable1.getSelectedRow();
        jTextField1.setText(model.getValueAt(rowIndex,0).toString());
        jTextField2.setText(model.getValueAt(rowIndex,1).toString());
        jTextField3.setText(model.getValueAt(rowIndex,2).toString());
        Object gender =model.getValueAt(rowIndex,3).toString ();
        if(gender.equals("Male")){
```

```
            jComboBox1.setSelectedIndex(0);
        }else{
             jComboBox1.setSelectedIndex(1);
        }
        jTextField4.setText(model.getValueAt(rowIndex,4).toString());
        jTextField5.setText(model.getValueAt(rowIndex,5).toString());
        jTextField6.setText(model.getValueAt(rowIndex,6).toString());
        jTextField7.setText(model.getValueAt(rowIndex,7).toString());
        jTextField8.setText(model.getValueAt(rowIndex,8).toString());
        jTextField9.setText(model.getValueAt(rowIndex,9).toString());
        String path = model.getValueAt(rowIndex,10).toString();
        imagePath = path;
        jLabelImage.setIcon(imageAdjust(path,null));


    }


       private ImageIcon imageAdjust(String path, byte[] pic){
        ImageIcon myImage = null;
        if(path != null){
            myImage = new ImageIcon(path);
        }
        else{
            myImage = new ImageIcon(pic);
        }
        Image img = myImage.getImage();
        Image newImage =
img.getScaledInstance(jLabelImage.getWidth(),jLabelImage.getHeight(),Ima
ge.SCALE_SMOOTH);
        ImageIcon icon = new ImageIcon(newImage);
        return icon;
         }
    /**
     * @param args the command line arguments
     */
    public static void main(String args[]) {
        /* Set the Nimbus look and feel */
        //<editor-fold defaultstate="collapsed" desc=" Look and feel
setting code (optional) ">
        /* If Nimbus (introduced in Java SE 6) is not available, stay
with the default look and feel.
         * For details see
```

```
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
         */
        try {
            for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
                if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());
                    break;
                }
            }
        } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(Home.class.getName()).log(java.util.l
ogging.Level.SEVERE, null, ex);
        } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(Home.class.getName()).log(java.util.l
ogging.Level.SEVERE, null, ex);
        } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(Home.class.getName()).log(java.util.l
ogging.Level.SEVERE, null, ex);
        } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(Home.class.getName()).log(java.util.l
ogging.Level.SEVERE, null, ex);
        }
        //</editor-fold>

        /* Create and display the form */
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new Home().setVisible(true);
            }
        });
    }

    // Variables declaration - do not modify
    private javax.swing.JButton jButton1;
    private javax.swing.JButton jButton4;
    private javax.swing.JButton jButton5;
    private javax.swing.JButton jButton8;
```

```
    private javax.swing.JButton jButton9;
    private javax.swing.JComboBox<String> jComboBox1;
    private javax.swing.JLabel jLabel1;
    private javax.swing.JLabel jLabel10;
    private javax.swing.JLabel jLabel11;
    private javax.swing.JLabel jLabel12;
    private javax.swing.JLabel jLabel2;
    private javax.swing.JLabel jLabel3;
    private javax.swing.JLabel jLabel4;
    private javax.swing.JLabel jLabel5;
    private javax.swing.JLabel jLabel6;
    private javax.swing.JLabel jLabel7;
    private javax.swing.JLabel jLabel8;
    private javax.swing.JLabel jLabel9;
    private javax.swing.JLabel jLabelImage;
    private javax.swing.JPanel jPanel1;
    private javax.swing.JPanel jPanel2;
    private javax.swing.JPanel jPanel3;
    private javax.swing.JPanel jPanel4;
    private javax.swing.JPanel jPanel5;
    private javax.swing.JPanel jPanel6;
    private javax.swing.JPanel jPanel7;
    private javax.swing.JPanel jPanel8;
    private javax.swing.JPanel jPanel9;
    private javax.swing.JScrollPane jScrollPane1;
    private javax.swing.JTabbedPane jTabbedPane1;
    private javax.swing.JTable jTable1;
    private javax.swing.JTextField jTextField1;
    private javax.swing.JTextField jTextField2;
    private javax.swing.JTextField jTextField3;
    private javax.swing.JTextField jTextField4;
    private javax.swing.JTextField jTextField5;
    private javax.swing.JTextField jTextField6;
    private javax.swing.JTextField jTextField7;
    private javax.swing.JTextField jTextField8;
    private javax.swing.JTextField jTextField9;
    // End of variables declaration
}
```

4. CONNECTION CODE

```
package db;

import java.sql.Connection;
import java.util.logging.Logger;
import java.util.logging.Level;
import java.sql.DriverManager;
public class MyConnection {
    private  static final String username="root";
    private static final String passward="STDPRO1";
    private static final String
dataConn="jdbc:mysql://localhost:3306/students_management";
    private static Connecption con =null;

    public static Connection getConnection(){
        try{
        Class.forName("com.mysql.cj.jdbc.Driver");
        con=DriverManager.getConnection(dataConn,username,passward);
    }catch(Exception ex){
       System.out.println(ex.getMessage());
    }
        return con;
 }
}
```

5. DATABASE CODE

```
create database students_management;

use students_management;

CREATE TABLE admin (
  id int NOT NULL AUTO_INCREMENT,
  username varchar(45) NOT NULL,
  password varchar(45) NOT NULL,
  PRIMARY KEY (id)
);

INSERT INTO admin VALUES (1,'admin','123');

CREATE TABLE student (
  id int NOT NULL AUTO_INCREMENT,
  name varchar(150) NOT NULL,
  date_of_birth date NOT NULL,
  gender varchar(10) NOT NULL,
  email varchar(100) NOT NULL,
  phone varchar(15) NOT NULL,
  father_name varchar(150) NOT NULL,
  mother_name varchar(150) NOT NULL,
  address1 text NOT NULL,
  address2 text NOT NULL,
  image_path varchar(200) NOT NULL,
  PRIMARY KEY (id)
);


CREATE TABLE course (
  id int NOT NULL,
  student_id int DEFAULT NULL,
  semester int DEFAULT NULL,
  course1 varchar(200) DEFAULT NULL,
  course2 varchar(200) DEFAULT NULL,
  course3 varchar(200) DEFAULT NULL,
  course4 varchar(200) DEFAULT NULL,
  course5 varchar(200) DEFAULT NULL,
  PRIMARY KEY (id),
  KEY fk_student_id (student_id),
  CONSTRAINT fk_student_id FOREIGN KEY (student_id) REFERENCES student
```

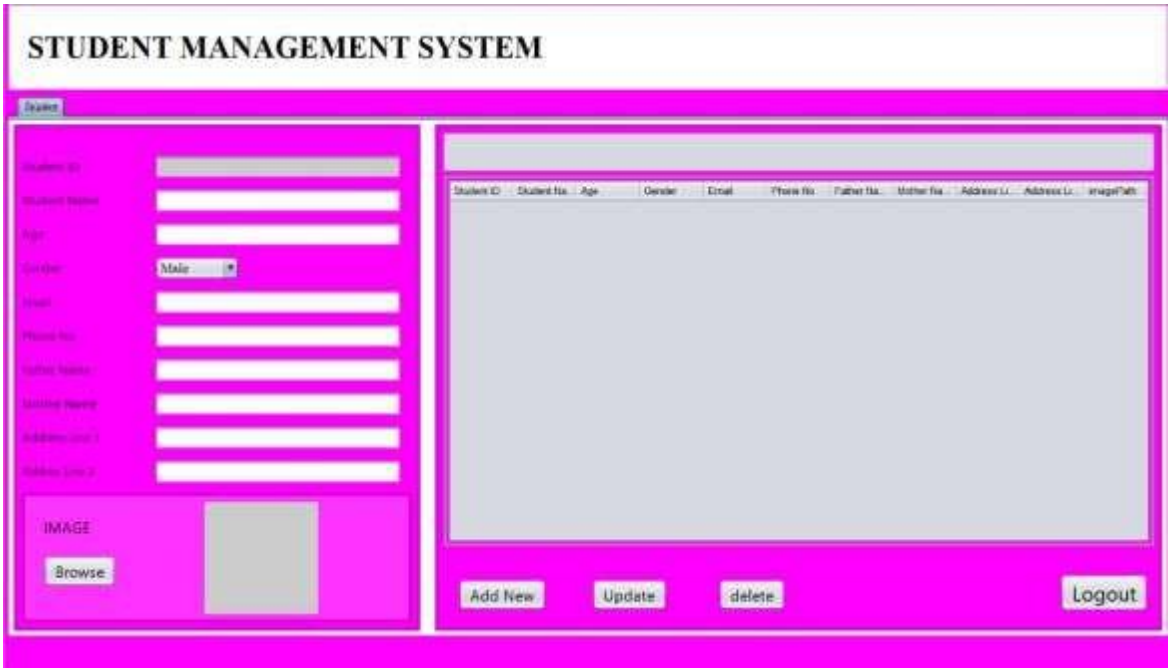
```
(id) ON DELETE CASCADE ON UPDATE CASCADE
);

CREATE TABLE score (
  id int NOT NULL AUTO_INCREMENT,
  student_id int NOT NULL,
  semester int NOT NULL,
  course1 varchar(200) NOT NULL,
  score1 double NOT NULL,
  course2 varchar(200) NOT NULL,
  score2 double NOT NULL,
  course3 varchar(200) NOT NULL,
  score3 double NOT NULL,
  course4 varchar(200) NOT NULL,
  score4 double NOT NULL,
  course5 varchar(200) NOT NULL,
  score5 double NOT NULL,
  average double NOT NULL,
  PRIMARY KEY (id),
  KEY fk_stu_id (student_id),
  CONSTRAINT fk_stu_id FOREIGN KEY (student_id) REFERENCES student (id)
ON DELETE CASCADE ON UPDATE CASCADE
);
```

Chapter 5 RESULTS AND DISCUSSION

LOADING PAGE

LOGIN PAGE

HOME PAGE


STUDENT MANAGEMENT SYSTEM
Student
Male
IMAGE
Browse
Add New
Update
delete
Logout


STUDENT MANAGEMENT SYSTEM
Student
Male
IMAGE
Browse
Add New
Update
delete
Logout

Chapter 6 CONCLUSION

6.1 Conclusion

The Student Database Management System is a robust and efficient solution tailored to address the complexities of managing academic data in educational institutions. It combines advanced features such as user authentication, admin dashboards, and streamlined middleware to provide a seamless experience for users. The system ensures the secure and efficient storage, retrieval, and processing of student records, course details, faculty assignments, and enrollment information. Developed using Java and SQL in the NetBeans environment, it leverages a modular and scalable architecture to accommodate future enhancements and increasing data demands.

Through the application of normalization principles, the system minimizes redundancy and ensures data integrity, fostering a reliable database structure. It also supports key functionalities like grade tracking, student performance analysis, and course management, which are integral to academic operations. This project not only demonstrates technical expertise but also highlights the potential of database-driven solutions in improving administrative efficiency and decision-making in educational settings. By addressing current challenges and offering scalable growth opportunities, this system represents a significant step towards the modernization of academic management processes.

Chapter 7 REFERENCES

7.1 REFERENCES

- W3Schools SQL Tutorial: https://www.w3schools.com/sql/
- JavaTpoint Java and SQL Integration Guide: https://www.javatpoint.com/
- GeeksforGeeks Database Management System (DBMS) Tutorials: https://www.geeksforgeeks.org/dbms/

Article

“Efficient Data Management Using SQL and Java in Educational Systems,” International Journal of Advanced Research in Computer Science.