



SOFTWARE APPLICATION DEVELOPMENT

Software solution Development with a Database
Management System (DBMS)

UNIT ID: H7E1 04

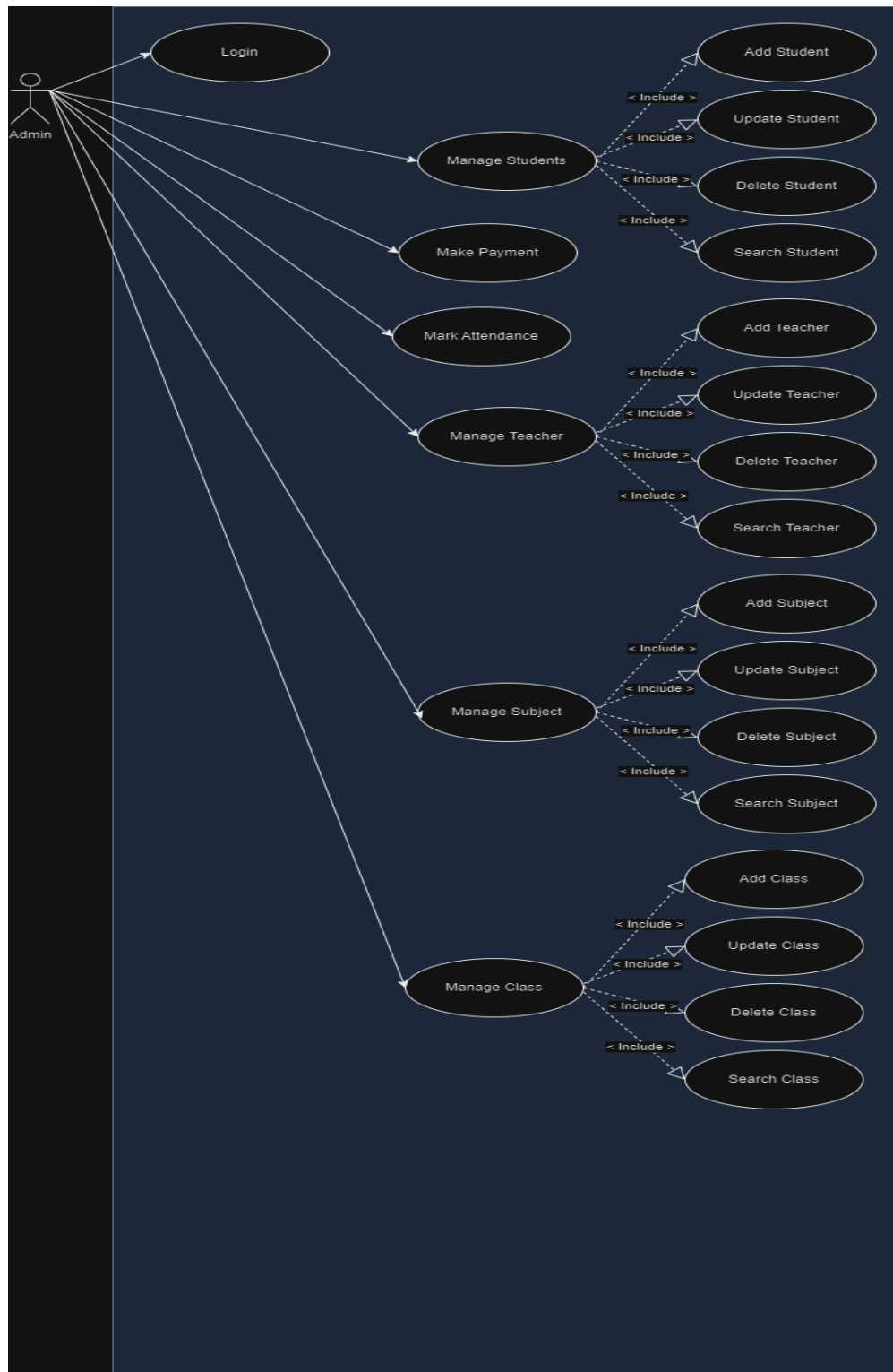
ASSIGNMENT ID: H7E1 04/AS/02

Batch: Moose-Matara

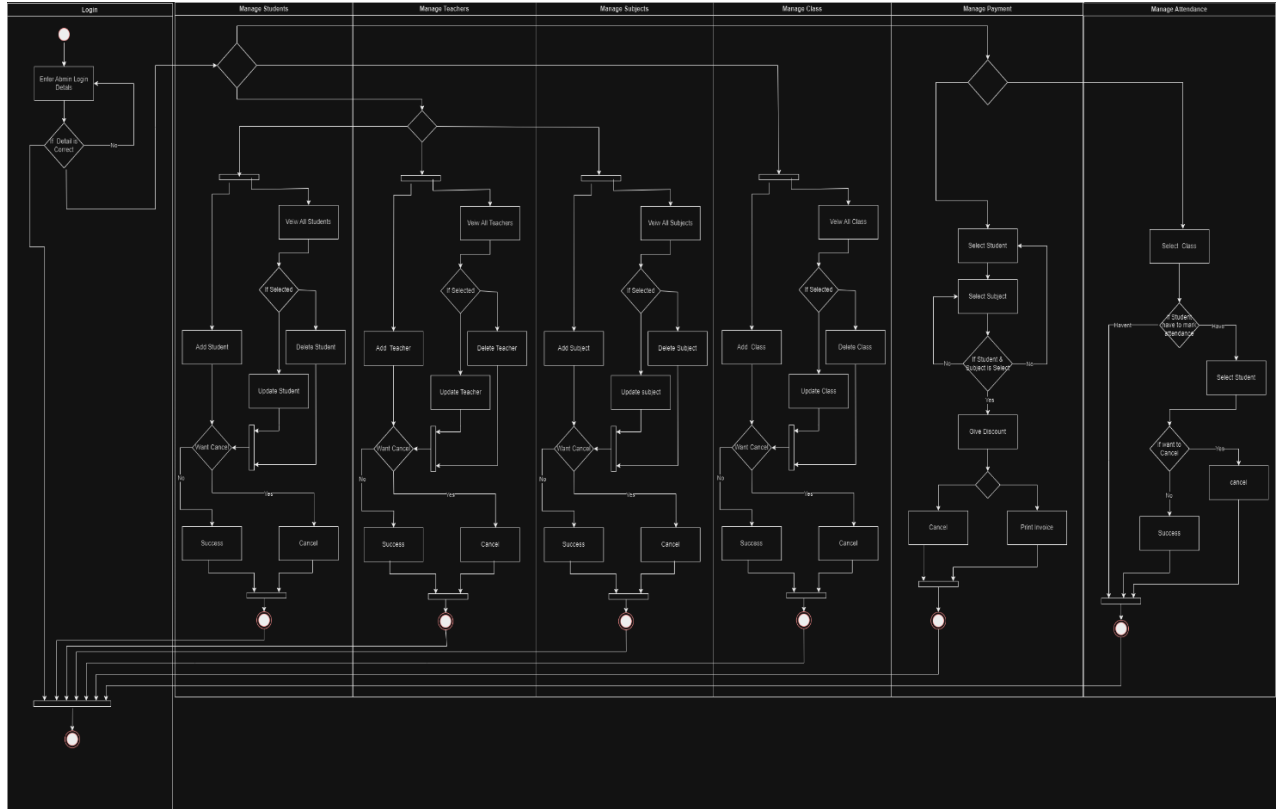
Sanchitha Heshan

(200312713429)

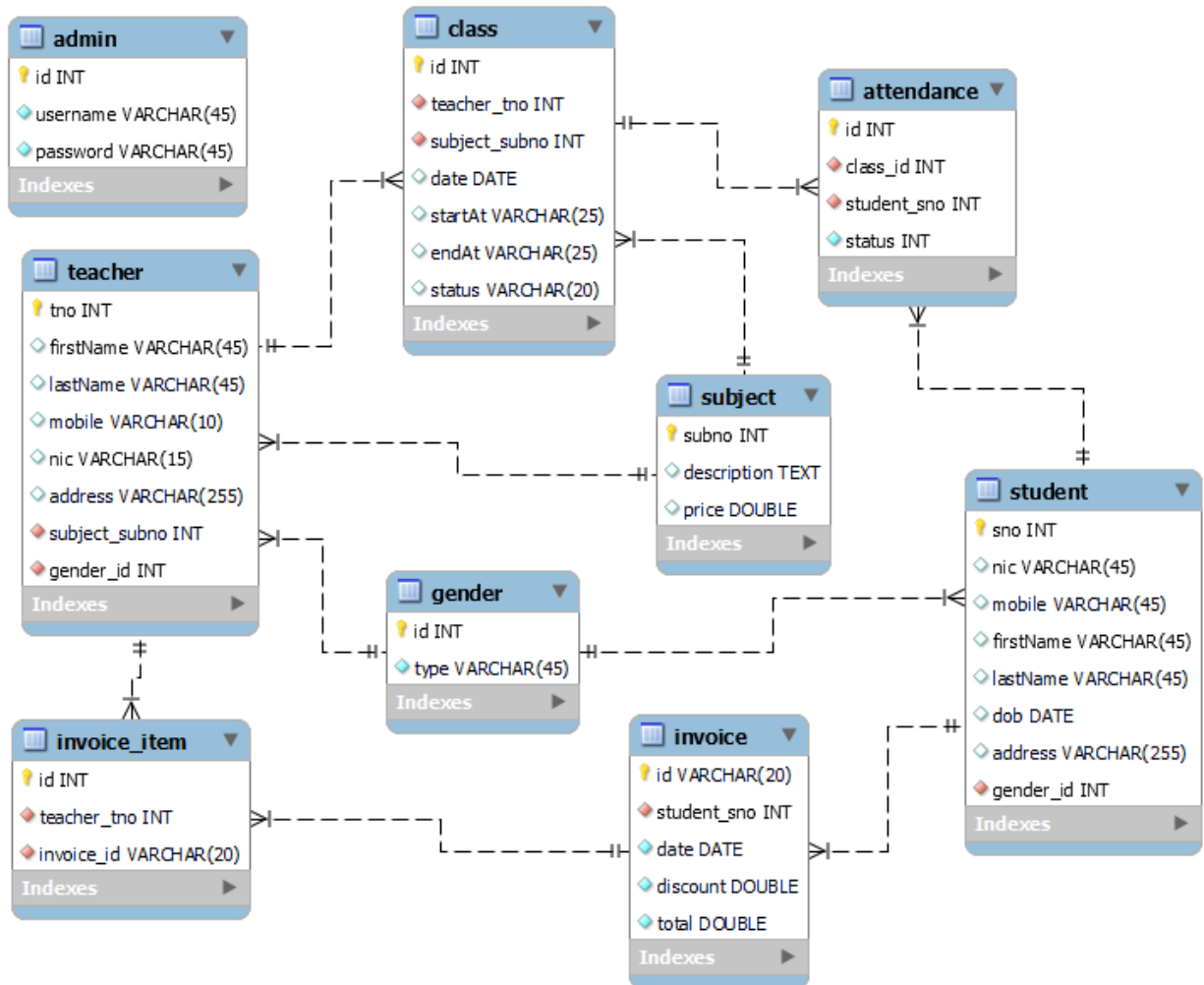
Use Case Diagram



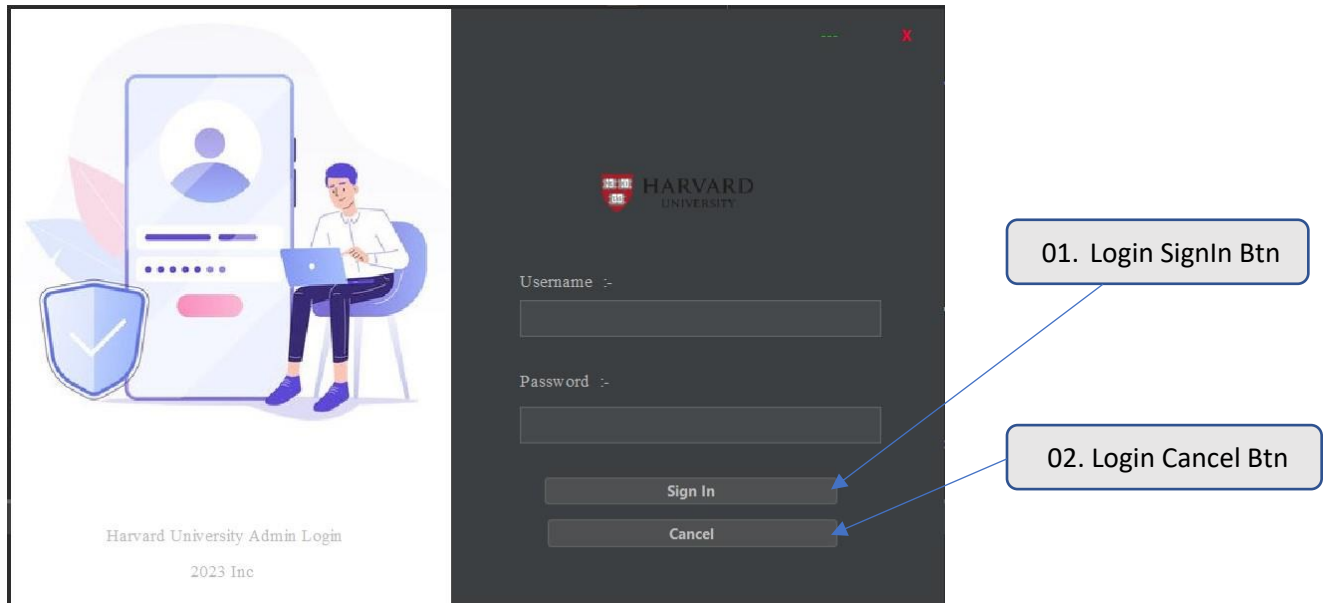
Activity Diagram



ER Diagram



Admin Login Page



01. Login SignIn Btn

```
private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {  
  
    if (!jTextField3.getText().isEmpty() || !String.valueOf(jPasswordField1.getPassword()).isEmpty()) {  
        try {  
            ResultSet resultSet = MySQL.Search("SELECT * FROM 'admin' WHERE 'username'='" + jTextField3.getText() + "' "  
                                                + "AND 'password'='" + String.valueOf(jPasswordField1.getPassword()) + "' ");  
  
            if (resultSet.next()) {  
                Home home = new Home();  
                home.setVisible(true);  
                this.dispose();  
            } else {  
                jTextField3.setText("");  
                jPasswordField1.setText("");  
                JOptionPane.showMessageDialog(this, "Invalid Email Or Password", "Error", JOptionPane.ERROR_MESSAGE);  
            }  
        } catch (Exception e) {  
            e.printStackTrace();  
        }  
    } else {  
        JOptionPane.showMessageDialog(this, "Invalid Email Or Password", "Error", JOptionPane.ERROR_MESSAGE);  
    }  
}
```

02. Login Cancel Btn

```
private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {  
    dispose();  
}
```

Home Page

Harvard University

Home

Manage Student

Manage Teachers

Manage Subjects

Manage Payments

Mark Attendance

Summary

All Students
4

All Teachers
4

All Subjects
6

Toady Class Shedule

No	Subject Name	Teacher NIC	Teacher Name	Date	Start At	End At	Status
1	Sinhala	199934162876	kusal Mendis	2024-05-13	02:44 PM	08:44 PM	Held
2	Sinhala	199934162876	kusal Mendis	2024-05-13	02:39 PM	08:39 PM	Canceled
5	Sinhala	199934162876	kusal Mendis	2024-05-13	07:19 PM	12:00 AM	Held
4	Science	199924381627	Pansilu Perera	2024-05-13	11:02 AM	12:02 PM	Held
6	Science	199924381627	Pansilu Perera	2024-05-13	09:17 AM	10:30 AM	Held
3	History	199428462317	Kumari Mudalige	2024-05-13	11:02 AM	12:02 PM	Held

2023 || Harvard University Admin Mode

01. Home Page Load

01. Home Page Load

```
private void jPanel5MouseClicked(java.awt.event.MouseEvent evt) {  
    AddStudentPanel.setVisible(false);  
    AddTeacherPanel.setVisible(false);  
    AddSubjectPanel.setVisible(false);  
    AddSubjectPanel.setVisible(false);  
    AddPaymentPanel.setVisible(false);  
    AttendancePanel.setVisible(false);  
  
    Content  
    HomeObjectCount();  
    loadHomeClassSheduleTable();  
    Content  
    HomePanel.setVisible(true);  
}
```

```

private void HomeObjectCount() {
    int StudentCount = 0;
    int TeacherCount = 0;
    int SubjectCount = 0;

    try {
        ResultSet result1 = MySQL.Search("SELECT COUNT(`sno`) AS `studentCount` FROM `student`");
        ResultSet result2 = MySQL.Search("SELECT COUNT(`tno`) AS `teacherCount` FROM `teacher`");
        ResultSet result3 = MySQL.Search("SELECT COUNT(`subno`) AS `subjectCount` FROM `subject`");

        while (result1.next()) {
            StudentCount += result1.getInt("studentCount");
        }

        while (result2.next()) {
            TeacherCount += result2.getInt("teacherCount");
        }

        while (result3.next()) {
            SubjectCount += result3.getInt("subjectCount");
        }

        jLabel73.setText(String.valueOf(StudentCount));
        jLabel74.setText(String.valueOf(TeacherCount));
        jLabel75.setText(String.valueOf(SubjectCount));

    } catch (Exception e) {
        e.printStackTrace();
    }
}

```

```

private void loadHomeClassSheduleTable() {
    try {
        ResultSet resultSet = MySQL.Search("SELECT * FROM `class` "
            + "INNER JOIN `subject` ON `subject`.`subno`=`class`.`subject_subno` "
            + "INNER JOIN `teacher` ON `teacher`.`tno`=`class`.`teacher_tno` ORDER BY `date` DESC");
        DefaultTableModel tModel = (DefaultTableModel) jTable10.getModel();
        tModel.setRowCount(0);

        Date date = new Date();
        SimpleDateFormat format = new SimpleDateFormat("yyyy-MM-dd");
        String stringToday = format.format(date);

        while (resultSet.next()) {
            if (stringToday.equals(resultSet.getString("date"))) {
                Vector<String> vector = new Vector<>();
                vector.add(resultSet.getString("id"));
                vector.add(resultSet.getString("description"));
                vector.add(resultSet.getString("nic"));
                vector.add(resultSet.getString("firstName") + " " + resultSet.getString("lastName"));
                vector.add(resultSet.getString("date"));
                vector.add(resultSet.getString("startAt"));
                vector.add(resultSet.getString("endAt"));
                vector.add(resultSet.getInt("status") == 1 ? "Held" : "Cancelled");
                tModel.addRow(vector);
            }
        }

    } catch (Exception e) {
        e.printStackTrace();
    }
}

```

Manage Students Page

Harvard University

Home

Manage Student

Manage Teachers

Manage Subjects

Manage Payments

Mark Attendance

Manage Students

First Name :-
Last Name :-
Date Of Birth :-
Address :-

Mobile :-
Nic No :-
Gender :-

Id
Select

Search Students By NIC No.....

Id	Nic	Mobile No	First Name	Last Name	Gender	Date Of Birth	Address
1	200834162719	0761634288	Sanuth	Nenuka	Male	2008-05-07	12/5, Matara, Sri La...
2	200316342876	0762849319	Pasan	Mihisara	Male	2003-07-30	12/4, Hakmana, Sri ...
3	200346284976	0762846179	Pasindu	Muthumina	Male	2003-08-09	12/8, Matara, Sri La...
5	200349821679	0769824167	Sangeetha	Wijeman	Female	2005-05-24	12/5, Matara, Sri La...

Save

Update

Delete

Cancel

01. Manage Student Btn

03. Select Student

02. Student Save Btn

04. Student Update Btn

05. Student Delete Btn

06. Student Cancel Btn

07. Student Search

01. Manage Student Btn

```
private void jPanel6MouseClicked(java.awt.event.MouseEvent evt) {  
    HomePanel.setVisible(false);  
    AddTeacherPanel.setVisible(false);  
    AddSubjectPanel.setVisible(false);  
    AddSubjectPanel.setVisible(false);  
    AddPaymentPanel.setVisible(false);  
    AttendancePanel.setVisible(false);  
  
    container  
    loadStudentsTable();  
    container  
    AddStudentPanel.setVisible(true);  
}
```

```
private void loadStudentsTable() {  
    DefaultTableModel model = (DefaultTableModel) jTable2.getModel();  
    model.setRowCount(0);  
  
    try {  
        ResultSet resultset = MySQL.Search("SELECT * FROM `student`");  
  
        while (resultset.next()) {  
            Vector<String> studentTableRow = new Vector<>();  
            studentTableRow.add(resultset.getString("sno"));  
            studentTableRow.add(resultset.getString("nic"));  
            studentTableRow.add(resultset.getString("mobile"));  
            studentTableRow.add(resultset.getString("firstName"));  
            studentTableRow.add(resultset.getString("lastName"));  
            studentTableRow.add((resultset.getString("gender_id").equals("1")) ? "Male" : "Female");  
            studentTableRow.add(resultset.getString("dob"));  
            studentTableRow.add(resultset.getString("address"));  
            model.addRow(studentTableRow);  
        }  
  
    } catch (Exception ex) {  
        ex.printStackTrace();  
    }  
}
```

02. Student Save Btn

```
private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {  
    String fieldStatus = validateStudentsFields();  
    if (fieldStatus.equals("Success")) {  
        String firstName = jTextField1.getText();  
        String lastName = jTextField2.getText();  
        String dob = jTextField33.getText();  
        String address = jTextField6.getText();  
        String mobile = jTextField7.getText();  
        String nic = jTextField11.getText();  
        String gender = String.valueOf(jComboBox5.getSelectedItem());  
        String result = (gender.equals("Male")) ? "1" : "2";  
  
        try {  
            ResultSet resultSet = MySQL.Search("SELECT * FROM `student` WHERE `nic`='" + nic + "'");  
  
            if (resultSet.next()) {  
                Alert("This Student Already Exist", false);  
            } else {  
                MySQL.Ins("INSERT INTO "  
                    + "`student` (`firstName`,`lastName`,`dob`,`address`,`mobile`,`nic`,`gender_id`) "  
                    + "VALUES('" + firstName + "','" + lastName + "','" + dob + "','" + address + "','"  
                    + "'" + mobile + "','" + nic + "','" + result + "')");  
  
                loadStudentsTable();  
                ManageStudentFieldReset();  
            }  
        } catch (Exception e) {  
            e.printStackTrace();  
        }  
    } else {  
        Alert(fieldStatus, true);  
    }  
}
```

```
private String validateStudentsFields() {  
    if (jTextField1.getText().isEmpty()) {  
        return "Please Enter First Name ....";  
    } else if (jTextField2.getText().isEmpty()) {  
        return "Please Enter Last Name ....";  
    } else if (jTextField6.getText().isEmpty()) {  
        return "Please Enter Living Address ....";  
    } else if (jTextField7.getText().isEmpty()) {  
        return "Please Enter Mobile Number ....";  
    } else if (jTextField7.getText().length() != 10) {  
        return "Invalid Mobile Number ....";  
    } else if (jTextField11.getText().isEmpty()) {  
        return "Please Enter National ID Number ....";  
    } else if (jComboBox5.getSelectedIndex() == 0) {  
        return "Please Select Gender ....";  
    } else {  
        return "Success";  
    }  
}
```

03. Select Student

```
private void jTable2MouseClicked(java.awt.event.MouseEvent evt) {  
    if (evt.getClickCount() == 2) {  
        int row = jTable2.getSelectedRow();  
  
        if (row != -1) {  
            jTextField8.setText(String.valueOf(jTable2.getValueAt(row, 0)));  
            jTextField11.setText(String.valueOf(jTable2.getValueAt(row, 1)));  
            jTextField7.setText(String.valueOf(jTable2.getValueAt(row, 2)));  
            jTextField1.setText(String.valueOf(jTable2.getValueAt(row, 3)));  
            jTextField2.setText(String.valueOf(jTable2.getValueAt(row, 4)));  
            jComboBox5.setSelectedItem(jTable2.getValueAt(row, 5));  
            jTextField33.setText(String.valueOf(jTable2.getValueAt(row, 6)));  
            jTextField6.setText(String.valueOf(jTable2.getValueAt(row, 7)));  
  
            jTextField33.setEnabled(false);  
            jTextField11.setEnabled(false);  
            jTable2.setEnabled(false);  
            jButton2.setEnabled(false);  
            jComboBox5.setEnabled(false);  
            jButton15.setEnabled(true);  
            jButton6.setEnabled(true);  
        } else {  
            jTable2.setEnabled(true);  
            jButton2.setEnabled(true);  
        }  
    }  
}
```

04. Student Update Btn

```
private void jButton6ActionPerformed(java.awt.event.ActionEvent evt) {  
  
    String status = valdateStudentsFields();  
    if (status.equals("Success")) {  
        MySQL.Iud(" UPDATE `student`"  
            + " SET `firstName`='" + jTextField1.getText() + "',"  
            + "`lastName`='" + jTextField2.getText() + "',"  
            + "`address`='" + jTextField6.getText() + "',"  
            + "`mobile`='" + jTextField7.getText() + "'"  
            + " WHERE `sno`='" + Integer.valueOf(jTextField8.getText()) + "'");  
  
        ManageStudentFieldReset();  
    } else {  
        Alert(status, true);  
    }  
}
```

05. Student Delete Btn

```
private void jButton15ActionPerformed(java.awt.event.ActionEvent evt) {  
    MySQL.Iuc("DELETE FROM `student` WHERE `sno`='" + Integer.valueOf(jTextField8.getText()) + "' ");  
    ManageStudentFieldReset();  
}
```

06. Student Cancel Btn

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {  
    ManageStudentFieldReset();  
}
```

```
private void ManageStudentFieldReset() {  
    jTextField1.setText("");  
    jTextField2.setText("");  
    jTextField6.setText("");  
    jTextField7.setText("");  
    jTextField11.setText("");  
    jTextField8.setText("");  
    Date date = new Date();  
    SimpleDateFormat format = new SimpleDateFormat("yyyy-MM-dd");  
    String stringDate = format.format(date);  
    jTextField33.setText(stringDate);  
    jTextField10.setText("Search Students By NIC No.....");  
    jButton6.setEnabled(false);  
    jButton2.setEnabled(true);  
    jButton15.setEnabled(false);  
    jTextField11.setEnabled(true);  
    jTable2.setEnabled(true);  
    jComboBox5.setEnabled(true);  
    jComboBox5.setSelectedIndex(0);  
  
    jTextField1.grabFocus();  
    loadStudentsTable();  
}
```

07. Student Search

```
private void jTextField10KeyReleased(java.awt.event.KeyEvent evt) {  
    try {  
  
        ResultSet resultset = MySQL.Search("SELECT * FROM `student` WHERE `nic` LIKE '%" + jTextField10.getText() + "%'");  
        DefaultTableModel model = (DefaultTableModel) jTable2.getModel();  
        model.setRowCount(0);  
  
        while (resultset.next()) {  
            Vector<String> studentTableRow = new Vector<>();  
            studentTableRow.add(resultset.getString("sno"));  
            studentTableRow.add(resultset.getString("nic"));  
            studentTableRow.add(resultset.getString("mobile"));  
            studentTableRow.add(resultset.getString("firstName"));  
            studentTableRow.add(resultset.getString("lastName"));  
            studentTableRow.add(resultset.getString("dob"));  
            studentTableRow.add(resultset.getString("address"));  
            model.addRow(studentTableRow);  
        }  
  
    } catch (Exception e) {  
        e.printStackTrace();  
    }  
}
```

Manage Teachers Page

Harvard University

Home

Manage Student

Manage Teachers

Manage Subjects

Manage Payments

Mark Attendance

Manage Teachers

Teacher First Name :-
Teacher Last Name :-
Teacher Address :-
Teacher Gender :- ☒ Male ☐ Female
Teacher NIC No :-
Teacher Mobile No :-
Subject :-
Id :-

Search Teachers By NIC No

Id	Mobile No	Nic No	Subject	First Name	Last Name	Gender	Address
1	0762819349	199934162876	Sinhala	kusal	Mendis	Male	12/A, Colombo, Sri ...
3	0762849316	199924381627	Science	Pansilu	Perera	Male	12/4, Colombo, Sri L...
4	0741685234	199428462317	History	Kumari	Mudalige	Female	12/9, Colombo, Sri L...
5	0769768249	199958316749	Civil	Pramitha	Pahan	Male	12/6, Katharagama, ...

SaveDeleteUpdateCancel

01. Manage Teacher Btn

03. Select Teacher

02. Teacher Save Btn

04. Teacher Delete Btn

05. Teacher Update Btn

06. Teacher Cancel Btn

07. Teacher Search

01. Manage Teacher Btn

```
private void jPanel2MouseClicked(java.awt.event.MouseEvent evt) {  
    AddStudentPanel.setVisible(false);  
    HomePanel.setVisible(false);  
    AddSubjectPanel.setVisible(false);  
    AddPaymentPanel.setVisible(false);  
    AttendancePanel.setVisible(false);  
  
    //Content  
    //LoadSubjectComboBox  
    try {  
        ResultSet resultSet = MySQL.Search("SELECT * FROM `subject` ");  
  
        Vector<String> vector = new Vector<String>();  
        vector.add("Select");  
  
        while (resultSet.next()) {  
            vector.add(resultSet.getString("description"));  
        }  
  
        DefaultComboBoxModel Combomodel = new DefaultComboBoxModel(vector);  
        jComboBox6.setModel(Combomodel);  
  
    } catch (Exception e) {  
        e.printStackTrace();  
    }  
    //LoadSubjectComboBox  
    loadTeachersTable();  
    //Content  
    AddTeacherPanel.setVisible(true);  
}
```

```
private void loadTeachersTable() {  
    DefaultTableModel model = (DefaultTableModel) jTable1.getModel();  
    model.setRowCount(0);  
  
    try {  
        ResultSet resultset = MySQL.Search("SELECT * FROM `teacher` "  
            + "INNER JOIN `subject` ON `teacher`.`subject_subno`=`subject`.`subno`");  
  
        while (resultset.next()) {  
            Vector<String> row = new Vector<>();  
            row.add(resultset.getString("tno"));  
            row.add(resultset.getString("mobile"));  
            row.add(resultset.getString("nic"));  
            row.add(resultset.getString("description"));  
            row.add(resultset.getString("firstName"));  
            row.add(resultset.getString("lastName"));  
            row.add(resultset.getString("gender_id").equals("1") ? "Male" : "Female");  
            row.add(resultset.getString("address"));  
            model.addRow(row);  
        }  
    } catch (Exception e) {  
        e.printStackTrace();  
    }  
}
```

02. Teacher Save Btn

```
private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {  
    String status = validateTeacherField();  
    if (status.equals("Success")) {  
        String subjectName = String.valueOf(jComboBox6.getSelectedItem());  
  
        try {  
            ResultSet resultSet = MySQL.Search("SELECT * FROM `teacher` WHERE `nic`='" + jTextField38.getText() + "'");  
  
            if (resultSet.next()) {  
                Alert("This Teacher Already Exist", false);  
            } else {  
                ResultSet result = MySQL.Search("SELECT * FROM `subject` WHERE `description`='" + subjectName + "'");  
  
                if (result.next()) {  
                    MySQL.Iud("INSERT INTO `teacher`"  
                        + "(`firstName`,`lastName`,`mobile`,`nic`,`address`,`gender_id`,`subject_subno`)"  
                        + "VALUES"  
                        + "(" + jTextField35.getText() + "','" + jTextField27.getText() + "','"  
                        + "'" + jTextField37.getText() + "','"  
                        + "'" + jTextField38.getText() + "','" + jTextField36.getText() + "','"  
                        + "'" + Integer.valueOf(buttonGroup1.getSelection().getActionCommand()) + "','"  
                        + "'" + result.getInt("subno") + "'"");  
  
                    ManageTeacherFieldReset();  
                    loadTeachersTable();  
                }  
            }  
        } catch (Exception e) {  
            e.printStackTrace();  
        }  
    } else {  
        Alert(status, true);  
    }  
}
```

```
private String validateTeacherField() {  
    if (jTextField35.getText().isEmpty()) {  
        return "Please Enter First Name ....";  
    } else if (jTextField27.getText().isEmpty()) {  
        return "Please Enter Last Name ....";  
    } else if (jTextField36.getText().isBlank()) {  
        return "Please Enter Living Address ....";  
    } else if (jTextField38.getText().isBlank()) {  
        return "Please Enter National ID Number ....";  
    } else if (jTextField37.getText().isBlank()) {  
        return "Please Enter Mobile Number ....";  
    } else if (jTextField37.getText().length() != 10) {  
        return "Invalid Mobile Number ....";  
    } else if (buttonGroup1.getSelection() == null) {  
        return "Please Select Gender ....";  
    } else if (jComboBox6.getSelectedIndex() == 0) {  
        return "Please Select Subject ....";  
    } else {  
        return "Success";  
    }  
}
```


03. Select Teacher

```
private void jTable1MouseClicked(java.awt.event.MouseEvent evt) {  
    if (evt.getClickCount() == 2) {  
        int teacherRow = jTable1.getSelectedRow();  
  
        if (teacherRow != -1) {  
            jTextField24.setText(String.valueOf(jTable1.getValueAt(teacherRow, 0)));  
            jComboBox6.setSelectedItem(String.valueOf(jTable1.getValueAt(teacherRow, 3)));  
            jTextField35.setText(String.valueOf(jTable1.getValueAt(teacherRow, 4)));  
            jTextField27.setText(String.valueOf(jTable1.getValueAt(teacherRow, 5)));  
            jTextField38.setText(String.valueOf(jTable1.getValueAt(teacherRow, 2)));  
            jTextField37.setText(String.valueOf(jTable1.getValueAt(teacherRow, 1)));  
            boolean condition = String.valueOf(jTable1.getValueAt(teacherRow, 6)).equalsIgnoreCase("Male") ? true : false;  
            if (condition) {  
                jRadioButton1.setSelected(true);  
            } else {  
                jRadioButton2.setSelected(true);  
            }  
            jTextField36.setText(String.valueOf(jTable1.getValueAt(teacherRow, 7)));  
            jRadioButton1.setEnabled(false);  
            jRadioButton2.setEnabled(false);  
            jTextField38.setEnabled(false);  
            jTable1.setEnabled(false);  
            jButton3.setEnabled(false);  
            jButton16.setEnabled(true);  
            jButton5.setEnabled(true);  
        }  
    }  
}
```

04. Teacher Delete Btn

```
private void jButton16ActionPerformed(java.awt.event.ActionEvent evt) {  
    MySQL.Iuc("DELETE FROM `teacher` WHERE `tno`='" + jTextField24.getText() + "'");  
    ManageTeacherFieldReset();  
    loadTeachersTable();  
}
```

05. Teacher Update Btn

```
private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {  
    String status = validateTeacherField();  
    if (status.equals("Success")) {  
        String tno = jTextField24.getText();  
        String subjectName = String.valueOf(jComboBox6.getSelectedItem());  
  
        try {  
            ResultSet resultSet = MySQL.Search("SELECT * FROM `subject` WHERE `description`='" + subjectName + "' ");  
  
            if (resultSet.next()) {  
                MySQL.Iud("UPDATE `teacher` "  
                    + "SET"  
                    + " `firstName`='" + jTextField35.getText() + "',"  
                    + " `lastName`='" + jTextField27.getText() + "',"  
                    + " `mobile`='" + jTextField37.getText() + "',"  
                    + " `address`='" + jTextField36.getText() + "',"  
                    + " `subject_subno`='" + resultSet.getInt("subno") + "' "  
                    + "WHERE"  
                    + " `tno`='" + tno + "' ");  
            }  
  
        } catch (Exception e) {  
            e.printStackTrace();  
        }  
  
        loadTeachersTable();  
        ManageTeacherFieldReset();  
    } else {  
        Alert(status, true);  
    }  
}
```

06. Teacher Cancel Btn

```
private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {  
    ManageTeacherFieldReset();  
}
```

```
private void ManageTeacherFieldReset() {  
    jTextField35.setText("");  
    jTextField27.setText("");  
    jTextField36.setText("");  
    jTextField38.setText("");  
    jTextField37.setText("");  
    jTextField24.setText("");  
    jTextField12.setText("Search Teachers By NIC No .....");  
    jComboBox6.setSelectedIndex(0);  
    jRadioButton1.setSelected(true);  
    jTextField35.grabFocus();  
  
    jButton16.setEnabled(false);  
    jButton5.setEnabled(false);  
    jButton3.setEnabled(true);  
    jRadioButton1.setEnabled(true);  
    jRadioButton2.setEnabled(true);  
    jTextField38.setEnabled(true);  
    jTable1.setEnabled(true);  
}
```

07. Teacher Search

```
private void jTextField12KeyReleased(java.awt.event.KeyEvent evt) {  
    if (jTextField12.getText().length() != 0) {  
        try {  
            ResultSet resultset = MySQL.Search("SELECT * FROM `teacher` WHERE `nic` LIKE '%" + jTextField12.getText() + "%'");  
            DefaultTableModel model = (DefaultTableModel) jTable1.getModel();  
            model.setRowCount(0);  
  
            while (resultset.next()) {  
                Vector<String> row = new Vector<>();  
                row.add(resultset.getString("tno"));  
                row.add(resultset.getString("mobile"));  
                row.add(resultset.getString("nic"));  
                row.add(resultset.getString("firstName"));  
                row.add(resultset.getString("lastName"));  
                row.add((resultset.getString("gender_id")).equals("1") ? "Male" : "Female");  
                row.add(resultset.getString("address"));  
                model.addRow(row);  
            }  
        } catch (Exception e) {  
            e.printStackTrace();  
        }  
    } else {  
        loadTeachersTable();  
    }  
}
```

Manage Teachers Page

01. Manage Subject Btn

03. Delete Subject Btn

05. Cancel Subject Btn

07. Select Subject

02. Save Subject Btn

04. Update Subject Btn

06. Search Subject

Harvard University

Manage Subject

Search Subject By Name ...

Id :-

Subject Name :-

Subject Price :-

id	Description	Price
1	Sinhala	25000.0
2	Maths	50000.0
3	Science	75000.0
4	English	45000.0
5	History	100000.0
6	Civil	50000.0

Add New Class

Id :-

Select Subject :- Select Teacher :- Teacher Nic :-

Start At :- End At :- Select Date :- Status :-

Search Class By Subject Name ...

ID	Subject	Teacher Nic No	Teacher Name	Date	Start At	End At	Status
1	Sinhala	199934162876	kusal Mendis	2024-05-13	02:44 PM	08:44 PM	Held
2	Sinhala	199934162876	kusal Mendis	2024-05-13	02:39 PM	08:39 PM	Canceled
5	Sinhala	199934162876	kusal Mendis	2024-05-13	07:19 PM	12:00 AM	Held
4	Science	199924381627	Pansilu Perera	2024-05-13	11:02 AM	12:02 PM	Held
6	Science	199924381627	Pansilu Perera	2024-05-13	09:17 AM	10:30 AM	Held

01. Add Class To Select Subject

02. Select Class Start Btn

03. Select Class End Time Btn

04. Select Class Held Date Btn

05. Class Save Btn

06. Class Select

07. Class Update Btn

08. Class Delete Btn

09. Class Cancel Btn

09. Class Search

01. Manage Subject Btn

```
private void jPanel7MouseClicked(java.awt.event.MouseEvent evt) {  
    AddStudentPanel.setVisible(false);  
    AddTeacherPanel.setVisible(false);  
    HomePanel.setVisible(false);  
    AddSubjectPanel.setVisible(false);  
    AddPaymentPanel.setVisible(false);  
    AttendancePanel.setVisible(false);  
  
    Content  
    loadSubjectsTable();  
    loadClassSheduleTable();  
  
    try {  
        ResultSet resultSet = MySQL.Search("SELECT * FROM `subject`");  
  
        Vector<String> vector = new Vector<String>();  
        vector.add("Select");  
  
        while (resultSet.next()) {  
            vector.add(resultSet.getString("description"));  
        }  
  
        DefaultComboBoxModel Smodel = new DefaultComboBoxModel(vector);  
        jComboBox1.setModel(Smodel);  
  
    } catch (Exception e) {  
        e.printStackTrace();  
    }  
  
    Content  
    AddSubjectPanel.setVisible(true);  
}
```

```

private void loadSubjectsTable() {

    DefaultTableModel tModel = (DefaultTableModel) jTable3.getModel();
    tModel.setRowCount(0);

    try {
        ResultSet resultSet = MySQL.Search("SELECT * FROM `subject`");

        while (resultSet.next()) {
            Vector<String> row = new Vector<>();
            row.add(resultSet.getString("subno"));
            row.add(resultSet.getString("description"));
            row.add(resultSet.getString("price"));
            tModel.addRow(row);
        }

    } catch (Exception e) {
        e.printStackTrace();
    }

}

```

```

private void loadClassSheduleTable() {
    DefaultTableModel tableModel = (DefaultTableModel) jTable4.getModel();
    tableModel.setRowCount(0);

    try {
        ResultSet resultSet = MySQL.Search("SELECT * FROM `class` "
            + "INNER JOIN `teacher` ON `class`.`teacher_tno`=`teacher`.`tno` "
            + "INNER JOIN `subject` ON `class`.`subject_subno`=`subject`.`subno` ORDER BY `date` DESC");

        while (resultSet.next()) {
            Vector<String> vector = new Vector<String>();
            vector.add(resultSet.getString("id"));
            vector.add(resultSet.getString("description"));
            vector.add(resultSet.getString("nic"));
            vector.add(resultSet.getString("firstName") + " " + resultSet.getString("lastName"));
            vector.add(resultSet.getString("date"));
            vector.add(resultSet.getString("startAt"));
            vector.add(resultSet.getString("endAt"));
            vector.add(resultSet.getInt("status") == 1 ? "Held" : "Cancelled");
            tableModel.addRow(vector);
        }

    } catch (Exception e) {
        e.printStackTrace();
    }

}

```


02. Save Subject Btn

```
private void jButton9ActionPerformed(java.awt.event.ActionEvent evt) {  
    String status = ValidateManageSubjectsField();  
    if (status == "Success") {  
        MySQL.Iud("INSERT INTO `subject`(`description`,`price`)"  
            + " VALUES ('" + jTextField25.getText() + "','" + jTextField4.getText() + "','" );  
        subjectFieldReset();  
        loadSubjectsTable();  
    } else {  
        Alert(status, true);  
    }  
}
```

03. Delete Subject Btn

```
private void jButton19ActionPerformed(java.awt.event.ActionEvent evt) {  
    MySQL.Iud("DELETE FROM `subject` WHERE `subno`='" + jTextField26.getText() + "'");  
  
    subjectFieldReset();  
    loadSubjectsTable();  
}
```

04. Update Subject Btn

```
private void jButton7ActionPerformed(java.awt.event.ActionEvent evt) {  
  
    String status = ValidateManageSubjectsField();  
    String subId = jTextField26.getText();  
    String description = jTextField25.getText();  
    String price = jTextField4.getText();  
  
    if (status == "Success") {  
        MySQL.Iud("UPDATE `subject` SET `description`='" + description + "', `price`='" + price + "' WHERE `subno`='" + subId + "'");  
        subjectFieldReset();  
        loadSubjectsTable();  
    }  
}
```

05. Cancel Subject Btn

```
private void jButton8ActionPerformed(java.awt.event.ActionEvent evt) {  
    subjectFieldReset();  
    loadSubjectsTable();  
}
```

```

private void subjectFieldReset() {
    jTextField9.setText("Search Subject By Name ...");
    jTextField25.setText("");
    jTextField4.setText("");
    jTextField26.setText("");
    jButton19.setEnabled(false);
    jButton7.setEnabled(false);
    jButton9.setEnabled(true);
    jTable3.setEnabled(true);
    jTextField25.grabFocus();
}

```

06. Search Subject

```

private void jTextField9KeyReleased(java.awt.event.KeyEvent evt) {
    if (jTextField9.getText().length() != 0) {
        try {
            ResultSet resultSet = MySQL.Search("SELECT * FROM `subject` WHERE `description` LIKE '%" + jTextField9.getText() + "%' ");
            DefaultTableModel tModel = (DefaultTableModel) jTable3.getModel();
            tModel.setRowCount(0);
            while (resultSet.next()) {
                Vector<String> row = new Vector<>();
                row.add(resultSet.getString("subno"));
                row.add(resultSet.getString("description"));
                row.add(resultSet.getString("price"));
                tModel.addRow(row);
            }
        } catch (Exception e) {
            e.printStackTrace();
        }
    } else {
        loadSubjectsTable();
    }
}

```

07. Select Subject

```

private void jTable3MouseClicked(java.awt.event.MouseEvent evt) {
    if (evt.getClickCount() == 2) {
        int tableRow = jTable3.getSelectedRow();

        if (tableRow != -1) {
            jTextField26.setText(String.valueOf(jTable3.getValueAt(tableRow, 0)));
            jTextField25.setText(String.valueOf(jTable3.getValueAt(tableRow, 1)));
            jTextField4.setText(String.valueOf(jTable3.getValueAt(tableRow, 2)));

            SubjectName = String.valueOf(jTable3.getValueAt(tableRow, 1));

            jButton9.setEnabled(false);
            jTable3.setEnabled(false);
            jButton19.setEnabled(true);
            jButton7.setEnabled(true);
        }
    }
}

```


01. Add Class To Select Subject

```
private void jComboBox1ItemStateChanged(java.awt.event.ItemEvent evt) {
    String Subject = String.valueOf(jComboBox1.getSelectedItem());

    try {
        ResultSet resultSet = MySQL.Search("SELECT `subno` FROM `subject` WHERE `description`='" + Subject + "' ");

        if (resultSet.next()) {
            int subNo = resultSet.getInt("subno");

            ResultSet result = MySQL.Search("SELECT * FROM `teacher` WHERE `subject_subno`='" + subNo + "' ");

            Vector<String> vector = new Vector<>();
            vector.add("Select");

            while (result.next()) {
                vector.add(result.getString("firstName"));
            }

            DefaultComboBoxModel tModel = new DefaultComboBoxModel(vector);
            tcjComboBox2.setModel(tModel);
        }

    } catch (Exception e) {
        e.printStackTrace();
    }
}
```

02. Select Class Start Btn Time

```
private void jButton10ActionPerformed(java.awt.event.ActionEvent evt) {
    timePicker1.showPopup(this, 50, 50);
}
```

03. Select Class End Time Btn

```
private void jButton20ActionPerformed(java.awt.event.ActionEvent evt) {
    timePicker2.showPopup(this, 50, 50);
}
```

04. Select Class Held Date Btn

```
private void jButton21ActionPerformed(java.awt.event.ActionEvent evt) {
    dateChooser1.showPopup();
}
```

05. Class Save Btn

```
private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {  
    String status = validateAddClassFields();  
  
    if (status == "Success") {  
  
        try {  
            ResultSet result = MySQL.Search("SELECT * FROM `subject` "  
                + "INNER JOIN `teacher` ON `teacher`.`subject_subno`=`subject`.`subno` "  
                + "WHERE `description`='" + String.valueOf(jComboBox1.getSelectedItemAt()) + "' "  
                + "AND `nic`='" + jTextField31.getText() + "' ");  
  
            if (result.next()) {  
                String statusId = String.valueOf(jComboBox3.getSelectedItemAt()).equals("Held") ? "1" : "2";  
                MySQL.Iud("INSERT INTO `class`(`teacher_tno`,`subject_subno`,`date`,`startAt`,`endAt`,`status`) "  
                    + "VALUES('" + result.getString("tno") + "','" + result.getString("subno") + "','"  
                    + "'" + jTextField29.getText() + "','" + jTextField5.getText() + "','"  
                    + "'" + jTextField28.getText() + "','" + statusId + "')");  
  
                resetAllAddClassFields();  
                loadClassSheduleTable();  
            }  
        } catch (Exception e) {  
            e.printStackTrace();  
        }  
    } else {  
        Alert(status, true);  
    }  
}
```

```
private String validateAddClassFields() {  
  
    if (jComboBox1.getSelectedIndex() == 0) {  
        return "Please Select Subject";  
    } else if (tcjComboBox2.getSelectedIndex() == 0) {  
        return "Please Select Teacher";  
    } else {  
        return "Success";  
    }  
}
```

07. Class Update Btn

```
private void jButton23ActionPerformed(java.awt.event.ActionEvent evt) {  
    String status = validateAddClassFields();  
  
    if (status.equals("Success")) {  
        String statusId = String.valueOf(jComboBox3.getSelectedItem()).equals("Held") ? "1" : "2";  
        MySQL.Iud("UPDATE `class` SET `date`='" + jTextField29.getText() + "',"  
            + " `startAt`='" + jTextField5.getText() + "', `endAt`='" + jTextField28.getText() + "', "  
            + "`status`='" + statusId + "' WHERE `id`='" + jTextField32.getText() + "' ");  
  
        loadClassSheduleTable();  
        resetAllAddClassFields();  
        jButton23.setEnabled(false);  
        jButton24.setEnabled(false);  
    } else {  
        Alert(status, true);  
    }  
}
```

08. Class Delete Btn

```
private void jButton24ActionPerformed(java.awt.event.ActionEvent evt) {  
    MySQL.Iud("DELETE FROM `class` WHERE `id`='" + jTextField32.getText() + "' ");  
    loadClassSheduleTable();  
    resetAllAddClassFields();  
}
```

09. Class Cancel Btn

```
private void jButton25ActionPerformed(java.awt.event.ActionEvent evt) {  
    resetAllAddClassFields();  
    jComboBox1.setEnabled(true);  
    tcjComboBox2.setEnabled(true);  
    jTable4.setEnabled(true);  
    jComboBox3.setEnabled(false);  
    loadClassSheduleTable();  
}
```

09. Class Search

```
private void jTextField30KeyReleased(java.awt.event.KeyEvent evt) {
    try {

        Date date = new Date();
        SimpleDateFormat format = new SimpleDateFormat("yyyy-MM-dd");
        String stringdate = format.format(date);

        ResultSet resultSet = MySQL.Search("SELECT * FROM `class` "
            + "INNER JOIN `subject` ON `subject`.`subno`=`class`.`subject_subno` "
            + "INNER JOIN `teacher` ON `teacher`.`tno`=`class`.`teacher_tno` "
            + "WHERE `description` LIKE '%" + jTextField30.getText() + "%' "
            + "AND `date`='" + stringdate + "' ORDER BY `date` DESC ");
        DefaultTableModel tmodel = (DefaultTableModel) jTable4.getModel();
        tmodel.setRowCount(0);

        while (resultSet.next()) {
            Vector<String> vector = new Vector<>();
            vector.add(resultSet.getString("class.id"));
            vector.add(resultSet.getString("subject.description"));
            vector.add(resultSet.getString("nic"));
            vector.add(resultSet.getString("firstName") + " " + resultSet.getString("lastName"));
            vector.add(resultSet.getString("date"));
            vector.add(resultSet.getString("startAt"));
            vector.add(resultSet.getString("endAt"));
            vector.add(resultSet.getString("status"));
            tmodel.addRow(vector);
        }

    } catch (Exception e) {
        e.printStackTrace();
    }
}
```

Manage Payment Page

Harvard University

Home

Manage Student

Manage Teachers

Manage Subjects

Manage Payments

Mark Attendance

Add Payment

Student NIC :- Name :- Mobile :- Student ID :-

Search

Cancel

No	Subject ID	Subject Name	Teacher ID	Teacher Name	Price	Payment Status
----	------------	--------------	------------	--------------	-------	----------------

Invoice Items

Invoice No

Teacher ID	Subject ID	Teacher Name	Subject Name	Price
------------	------------	--------------	--------------	-------

Item Count :-

Total :-

Discount :- %

Grand Price :-

Make Payment & Print Invoice

Cancel

01. Manage Payment Btn

02. Search Student

03. Search Student Have To Pay Subjects Btn

04. Cancel Student Btn

05. Select Subject

07. Make Payment Btn

06. Add Discount

08. Cancel All Btn

01. Manage Payment Btn

```
private void jPanel10MouseClicked(java.awt.event.MouseEvent evt) {  
    HomePanel.setVisible(false);  
    AddTeacherPanel.setVisible(false);  
    AddSubjectPanel.setVisible(false);  
    AddStudentPanel.setVisible(false);  
    AttendancePanel.setVisible(false);  
  
    jTextField45.setText(generateUniqueId());  
    AddPaymentPanel.setVisible(true);  
}
```

02. Search Student

```
private void jTextField13KeyReleased(java.awt.event.KeyEvent evt) {  
    try {  
  
        ResultSet resultSet = MySQL.Search("SELECT * FROM `student` WHERE `nic` LIKE '" + jTextField13.getText() + "' ");  
  
        while (resultSet.next()) {  
            jTextField48.setText(resultSet.getString("firstName") + " " + resultSet.getString("lastName"));  
            jTextField50.setText(resultSet.getString("mobile"));  
            jTextField51.setText(resultSet.getString("sno"));  
        }  
  
    } catch (Exception e) {  
        e.printStackTrace();  
    }  
}
```

03. Search Student Have To Pay Subjects Btn

```
private void jButton11ActionPerformed(java.awt.event.ActionEvent evt) {  
    try {  
        DefaultTableModel model = (DefaultTableModel) jTable14.getModel();  
        model.setRowCount(0);  
        int no = 0;  
  
        ResultSet resultSet = MySQL.Search("SELECT * FROM `Subject` INNER JOIN `teacher` ON `teacher`.`subject_subno`='subject`.`subno` ");  
  
        while (resultSet.next()) {  
            no += 1;  
            Vector<String> vector = new Vector<String>();  
            vector.add(String.valueOf(no));  
            vector.add(resultSet.getString("subno"));  
            vector.add(resultSet.getString("description"));  
            vector.add(resultSet.getString("tno"));  
            vector.add(resultSet.getString("firstName") + " " + resultSet.getString("lastName"));  
            vector.add(resultSet.getString("price"));  
            ResultSet result = MySQL.Search("SELECT * FROM `invoice` " +  
                "INNER JOIN `invoice_item` ON `invoice`.`id`='invoice_item`.`invoice_id` " +  
                "WHERE `student_sno`='" + jTextField51.getText() + "' AND `teacher_tno`='" + resultSet.getInt("tno") + "' ");  
            if (result.next()) {  
                vector.add("Paid");  
            } else {  
                vector.add("Not Paid");  
            }  
            model.addRow(vector);  
        }  
    } catch (Exception e) {  
        e.printStackTrace();  
    }  
}
```


04. Cancel Student Btn

```
private void jButton12ActionPerformed(java.awt.event.ActionEvent evt) {  
    jTextField13.setText("");  
    jTextField48.setText("");  
    jTextField50.setText("");  
    jTextField51.setText("");  
  
    DefaultTableModel model = (DefaultTableModel) jTable14.getModel();  
    model.setRowCount(0);  
}
```

05. Select Subject

```
private void jTable14MouseClicked(java.awt.event.MouseEvent evt) {  
    int total = 0;  
    if (evt.getClickCount() == 2) {  
        DefaultTableModel model = (DefaultTableModel) jTable9.getModel();  
        int row = jTable14.getSelectedRow();  
  
        if (row != -1) {  
            if (String.valueOf(jTable14.getValueAt(row, 6)).equals("Paid")) {  
                Alert("This Subject Already Paid", true);  
            } else {  
                Vector<String> vector = new Vector<String>();  
                vector.add(String.valueOf(jTable14.getValueAt(row, 3)));  
                vector.add(String.valueOf(jTable14.getValueAt(row, 1)));  
                vector.add(String.valueOf(jTable14.getValueAt(row, 4)));  
                vector.add(String.valueOf(jTable14.getValueAt(row, 2)));  
                vector.add(String.valueOf(jTable14.getValueAt(row, 5)));  
  
                model.addRow(vector);  
                ((DefaultTableModel) jTable14.getModel()).removeRow(row);  
  
                for (int i = 0; i < jTable9.getRowCount(); i++) {  
                    double price = Double.parseDouble(jTable9.getValueAt(i, 4).toString());  
                    total += price;  
                }  
            }  
        }  
  
        jLabel68.setText(String.valueOf(total));  
        jLabel67.setText(String.valueOf(jTable9.getRowCount()));  
  
        double percentage = 0;  
        jLabel66.setText(String.valueOf(percentage));  
        if (jTextField46.getText().isEmpty()) {  
            percentage = 0;  
        } else {  
            percentage = Double.parseDouble(jTextField46.getText());  
        }  
        double discount = (total * percentage) / 100;  
        jLabel66.setText(String.valueOf(total - discount));  
    }  
}
```

06. Add Discount

```
private void jTextField46KeyReleased(java.awt.event.KeyEvent evt) {  
    double total = Double.parseDouble(jLabel68.getText());  
    double precentage;  
    if (jTextField46.getText().isEmpty()) {  
        precentage = 0;  
    } else {  
        precentage = Double.parseDouble(jTextField46.getText());  
    }  
  
    double discount = (total * precentage) / 100;  
    jLabel66.setText(String.valueOf(total - discount));  
}
```

07. Make Payment Btn

```
private void jButton13ActionPerformed(java.awt.event.ActionEvent evt) {  
    Date date = new Date();  
    SimpleDateFormat format = new SimpleDateFormat("yyyy-MM-dd");  
    String stringDate = format.format(date);  
    String invoiceNo = jTextField45.getText();  
    String sno = jTextField51.getText();  
  
    MySQL.Iud("INSERT INTO 'invoice' VALUES('" + invoiceNo + "','" + sno + "','" + stringDate + "','" + jTextField46.getText() + "','" + jLabel68.getText() + "','" );  
    for (int i = 0; i < jTable9.getRowCount(); i++) {  
        String teacherId = String.valueOf(jTable9.getValueAt(i, 0));  
        MySQL.Iud("INSERT INTO 'invoice_item'('teacher_tno','invoice_id') VALUES('" + teacherId + "','" + invoiceNo + "')");  
    }  
  
    HashMap<String, Object> parameters = new HashMap<>();  
    parameters.put("Parameter1", stringDate);  
    parameters.put("Parameter2", jTextField51.getText());  
    parameters.put("Parameter3", jTextField48.getText());  
    parameters.put("Parameter4", jTextField13.getText());  
    parameters.put("Parameter5", jTextField50.getText());  
    parameters.put("Parameter6", invoiceNo);  
    parameters.put("Parameter7", jLabel67.getText());  
    parameters.put("Parameter8", jLabel68.getText());  
    parameters.put("Parameter9", jTextField46.getText() + "%");  
    parameters.put("Parameter10", jLabel66.getText());  
  
    try {  
        JTableModelDataSource dataSource = new JTableModelDataSource(jTable9.getModel());  
        JasperPrint report = JasperFillManager.fillReport("src/reports/java2Assignment.jasper", parameters, dataSource);  
        JasperViewer.viewReport(report, false);  
  
        jTextField13.setText("");  
        jTextField48.setText("");  
        jTextField50.setText("");  
        jTextField51.setText("");  
  
        DefaultTableModel model = (DefaultTableModel) jTable14.getModel();  
        model.setRowCount(0);  
        resetAllPaymentFieldsTables();  
        jTextField45.setText(generateUniqueId());  
    } catch (Exception e) {  
        e.printStackTrace();  
    }  
}
```


08. Cancel All Btn

```
private void jButton14ActionPerformed(java.awt.event.ActionEvent evt) {  
    jTextField13.setText("");  
    jTextField48.setText("");  
    jTextField50.setText("");  
    jTextField51.setText("");  
  
    DefaultTableModel model = (DefaultTableModel) jTable14.getModel();  
    model.setRowCount(0);  
    resetAllPaymentFieldsTables();  
    jTextField45.setText(generateUniqueId());  
}
```

```
private void resetAllPaymentFieldsTables() {  
    jTextField13.setText("");  
    jTextField45.setText("");  
    jLabel67.setText("");  
    jLabel68.setText("");  
    jTextField46.setText("");  
    jLabel66.setText("");  
  
    DefaultTableModel invoiceItemTModel = (DefaultTableModel) jTable9.getModel();  
    invoiceItemTModel.setRowCount(0);  
}
```

Manage Attendance Page

Harvard University

Attendance

Select Today Class

ID	Teacher NIC	Teacher	Subject ID	Subject	Start At	End At
1	19993416287...	kusal Mendis	1	Sinhala	02:44 PM	08:44 PM
2	19993416287...	kusal Mendis	1	Sinhala	02:39 PM	08:39 PM

Select Student

Id	Nic No	Student Name
----	--------	--------------

Status :- Present

Add Cancel

Attendance Mark Students

ID	Subject ID	Subject	Teacher ID	Teacher	Student Nic	Student Na...	Date	Status	StartAt	EndAt
----	------------	---------	------------	---------	-------------	---------------	------	--------	---------	-------

01. Manage Attendance Btn

02. Select Class

03. Select Student

04. Mark Attendance Btn

05. Cancel Btn

01. Manage Attendance Btn

```
private void jPanell13MouseClicked(java.awt.event.MouseEvent evt) {  
    AddStudentPanel.setVisible(false);  
    AddTeacherPanel.setVisible(false);  
    AddSubjectPanel.setVisible(false);  
    AddSubjectPanel.setVisible(false);  
    AddPaymentPanel.setVisible(false);  
  
    Date date = new Date();  
    SimpleDateFormat format = new SimpleDateFormat("yyyy-MM-dd");  
    String stringToday = format.format(date);  
  
    loadAttendanceTeacherTable("SELECT * FROM `subject` \n"  
        + "INNER JOIN `class` ON `subject`.`subno`=`class`.`subject_subno` \n"  
        + "INNER JOIN `teacher` ON `class`.`teacher_tno`=`teacher`.`tno` \n"  
        + "WHERE `status`!='Cancelled' AND `date`='" + stringToday + "'");  
  
    HomePanel.setVisible(false);  
    AttendancePanel.setVisible(true);  
}
```

```
private void loadAttendanceTeacherTable(String query) {  
    try {  
        ResultSet resultSet = MySQL.Search(query);  
        DefaultTableModel model = (DefaultTableModel) jTable11.getModel();  
        model.setRowCount(0);  
  
        while (resultSet.next()) {  
            String time = resultSet.getString("class.endAt");  
            if (!checkTimeIsPassedOrNot(time)) {  
                Vector v = new Vector();  
                v.add(resultSet.getString("class.id"));  
                v.add(resultSet.getString("teacher.nic"));  
                v.add(resultSet.getString("teacher.firstName") + " " + resultSet.getString("teacher.lastName"));  
                v.add(resultSet.getString("subject.subno"));  
                v.add(resultSet.getString("subject.description"));  
                v.add(resultSet.getString("class.startAt"));  
                v.add(resultSet.getString("class.endAt"));  
                model.addRow(v);  
            } else {  
                System.out.println("pass");  
            }  
        }  
    } catch (Exception e) {  
        e.printStackTrace();  
    }  
}
```

02. Select Class

```
private void jTable1MouseClicked(java.awt.event.MouseEvent evt) {  
    if (evt.getClickCount() == 1) {  
  
        int selectedIndex = jTable1.getSelectedRow();  
  
        if (selectedIndex != -1) {  
            loadAttendanceStudentTable(jTable1.getValueAt(selectedRowIndex, 3).toString());  
            loadAttendanceMarkedTable();  
        }  
    }  
}
```

```
private void loadAttendanceStudentTable(String subno) {  
    try {  
        ResultSet resultSet = MySQL.Search("SELECT * FROM invoice_item "  
            + "INNER JOIN teacher ON teacher.tno = invoice_item.teacher_tno "  
            + "INNER JOIN invoice ON invoice.id = invoice_item.invoice_id "  
            + "INNER JOIN student ON student.sno = invoice.student_sno "  
            + "WHERE teacher.subject_subno = '" + subno + "' ");  
  
        DefaultTableModel model = (DefaultTableModel) jTable2.getModel();  
        model.setRowCount(0);  
  
        while (resultSet.next()) {  
            ResultSet result = MySQL.Search("SELECT * FROM 'attendance' "  
                + "WHERE 'class_id'='" + jTable1.getValueAt(jTable1.getSelectedRow(), 0) + "' "  
                + "AND 'student_sno'='" + resultSet.getString("student.sno") + "' ");  
  
            if (!result.next()) {  
                Vector v = new Vector();  
                v.add(resultSet.getString("student.sno"));  
                v.add(resultSet.getString("student.nic"));  
                v.add(resultSet.getString("student.firstName") + " " + resultSet.getString("student.lastName"));  
                model.addRow(v);  
            }  
        }  
    } catch (Exception e) {  
        e.printStackTrace();  
    }  
}
```

```
private void loadAttendanceMarkedTable() {  
    DefaultTableModel model = (DefaultTableModel) jTable3.getModel();  
    model.setRowCount(0);  
    try {  
        ResultSet result = MySQL.Search("SELECT * FROM 'attendance'\n"  
            + "INNER JOIN 'class' ON 'class'.id='attendance'.class_id\n"  
            + "INNER JOIN 'subject' ON 'subject'.subno='class'.subject_subno\n"  
            + "INNER JOIN 'teacher' ON 'teacher'.tno='class'.teacher_tno\n"  
            + "INNER JOIN 'student' ON 'student'.sno='attendance'.student_sno\n"  
            + "WHERE 'class_id'='" + jTable1.getValueAt(jTable1.getSelectedRow(), 0) + "'");  
  
        while (result.next()) {  
            Vector v = new Vector();  
            v.add(result.getString("attendance.id"));  
            v.add(result.getString("subject.subno"));  
            v.add(result.getString("subject.description"));  
            v.add(result.getString("teacher.tno"));  
            v.add(result.getString("teacher.firstName") + " " + result.getString("teacher.lastName"));  
            v.add(result.getString("student.nic"));  
            v.add(result.getString("student.firstName") + " " + result.getString("student.lastName"));  
            v.add(result.getString("class.date"));  
            v.add(result.getInt("attendance.status") == 1 ? "Present" : "absent");  
            v.add(result.getString("class.startAt"));  
            v.add(result.getString("class.endAt"));  
            model.addRow(v);  
        }  
    } catch (Exception e) {  
        e.printStackTrace();  
    }  
}
```

03. Select Student

```
private void jTable12MouseClicked(java.awt.event.MouseEvent evt) {  
    int selectedRow = jTable12.getSelectedRow();  
  
    if (selectedRow != -1) {  
        jTable12.setEnabled(false);  
        System.out.println("selected");  
        jComboBox2.setEnabled(true);  
        jButton17.setEnabled(true);  
    } else {  
        jComboBox2.setSelectedIndex(0);  
        jComboBox2.setEnabled(false);  
        jButton17.setEnabled(false);  
        jTable12.setEnabled(false);  
    }  
}
```

04. Mark Attendance Btn

```
private void jButton17ActionPerformed(java.awt.event.ActionEvent evt) {  
  
    if (jTable12.getSelectedRow() != -1) {  
        String classId = jTable11.getValueAt(jTable11.getSelectedRow(), 0).toString();  
        String sno = jTable12.getValueAt(jTable12.getSelectedRow(), 0).toString();  
        String status = "Present".equals(jComboBox2.getSelectedItem().toString()) ? "1" : "2";  
  
        MySQL.Iud("INSERT INTO `attendance` (`class_id`,`student_sno`,`status`) "  
            + "VALUES ('" + classId + "','" + sno + "','" + status + "')");  
        int selectedRowIndex = jTable11.getSelectedRow();  
  
        if (selectedRowIndex != -1) {  
            loadAttendanceStudentTable(jTable11.getValueAt(selectedRowIndex, 3).toString());  
            loadAttendanceMarkedTable();  
        }  
        resetAllAttendancePage();  
    } else {  
        JOptionPane.showMessageDialog(this, "Please Select Class", "Error Message", JOptionPane.ERROR_MESSAGE);  
    }  
}
```

05. Cancel Btn

```
private void jButton18ActionPerformed(java.awt.event.ActionEvent evt) {  
    resetAllAttendancePage();  
    DefaultTableModel model2 = (DefaultTableModel) jTable13.getModel();  
    model2.setRowCount(0);  
    DefaultTableModel model1 = (DefaultTableModel) jTable12.getModel();  
    model1.setRowCount(0);  
}
```

THANK YOU