

Movie Ticket Booking System

Submitted By:

A22126511163 – PASUPULETI BHAGYA

A22126511167 – PIDUGU DINESH

A22126511175 – SANGARU SIDDU

A22126511181 – VADDI GAYATHRI

A22126511209 – DHARMAVARAPU KALYAN

ABSTRACT

This project aims to develop a movie ticket booking system using Java, featuring an admin interface to manage theaters and showtimes. The system allows users to browse movies, select showtimes, and book tickets. The methodology involved designing a user-friendly interface, implementing backend functionalities for managing bookings and theaters, and ensuring secure admin access. Key findings indicated that the system efficiently handles multiple bookings and provides a smooth user experience, with robust admin features for theater management.

Table of contents

- 1. Introduction
 - 1.1 Background
 - 1.2 Objective
 - 1.3 Scope
- 2. Literature Review
- 3. Methodology
 - 3.1 Approach
 - 3.2 Procedures
- 4. Implementation
 - 4.1 Design
 - 4.2 Development
 - 4.3 Admin Features
 - 4.4 Challenges
- 5. Results
 - 5.1 Data
 - 5.2 Analysis
- 6. Discussion
 - 6.1 Interpretation
 - 6.2 Comparison
 - 6.3 Implications
- 7. Conclusion
 - 7.1 Summary
 - 7.2 Future Work
- 8. Appendices

1. Introduction

1.1 Background

The digital age has transformed how people purchase movie tickets, moving from physical box offices to online systems. This project aims to create a Java-based system that simplifies the ticket booking process while enabling admin control over theater management.

1.2 Objective

The objective is to develop a movie ticket booking system that allows users to browse movies, select showtimes, and book tickets efficiently, with an admin interface for managing theaters and showtimes.

1.3 Scope

The project includes developing a user interface for browsing and booking tickets, an admin interface for managing theaters and showtimes, a backend for managing bookings and payments, and a database for storing all relevant data. It does not cover the deployment of the system or integration with external movie databases.

2. Literature Review

Existing movie ticket booking systems, such as Fandango and BookMyShow, offer extensive features but can be complex and costly to develop. This project seeks to provide a simpler, cost-effective solution using Java.

3. Methodology

3.1 Approach

The project uses Java for both frontend (Swing for GUI) and backend (JDBC for database connectivity). The database used is SQL PLUS.

3.2 Procedures

1. Design the system architecture.
2. Develop the database schema.
3. Implement the user interface using Java Swing.

4. Implement backend functionalities using Java and JDBC.
5. Implement the admin interface for managing theaters, movies, and showtimes.
6. Test the system with sample data.

4. Implementation

4.1 Design

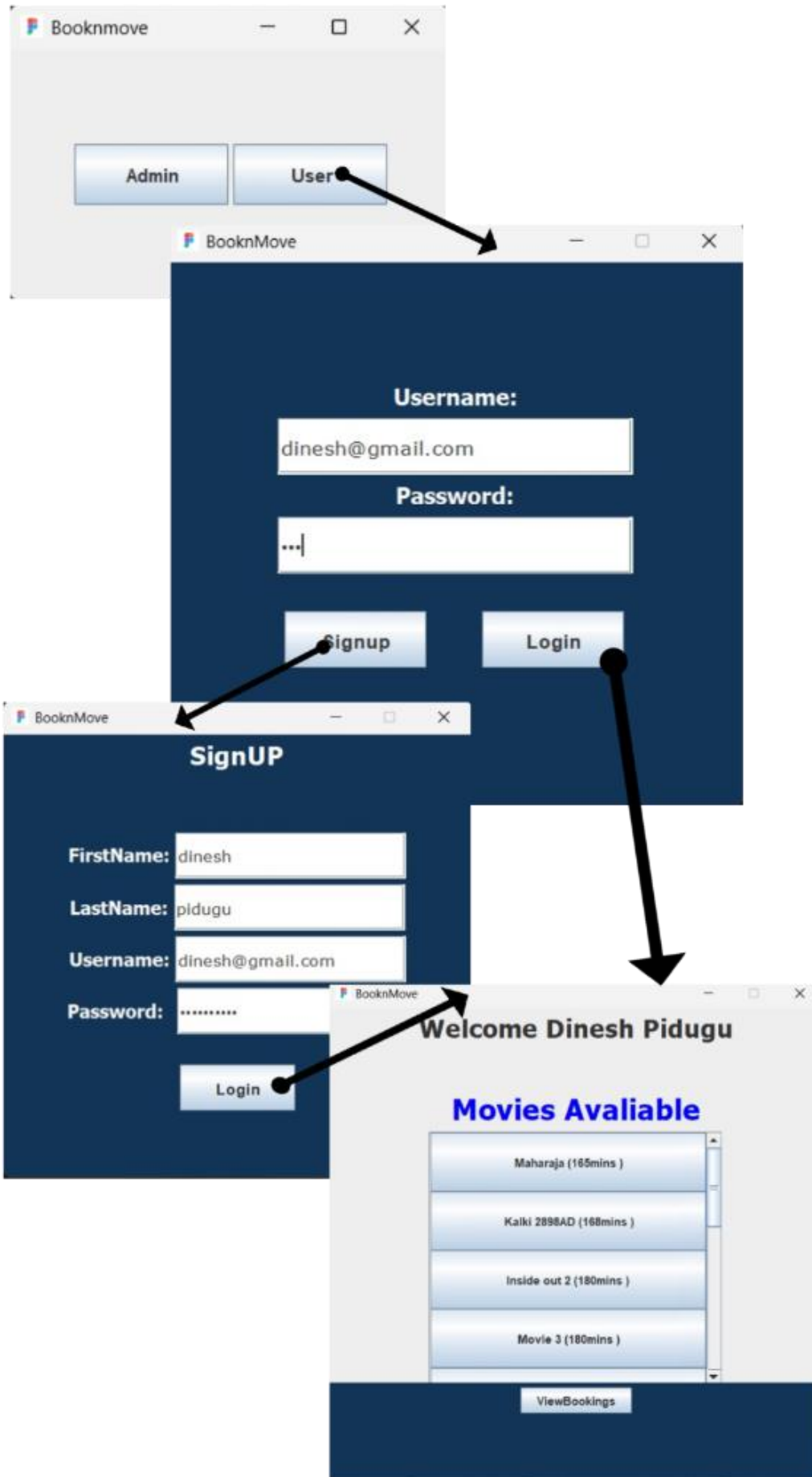
The system is designed with a modular approach: a user interface for interactions, a database for storing data, backend services for processing requests, and an admin interface for management.

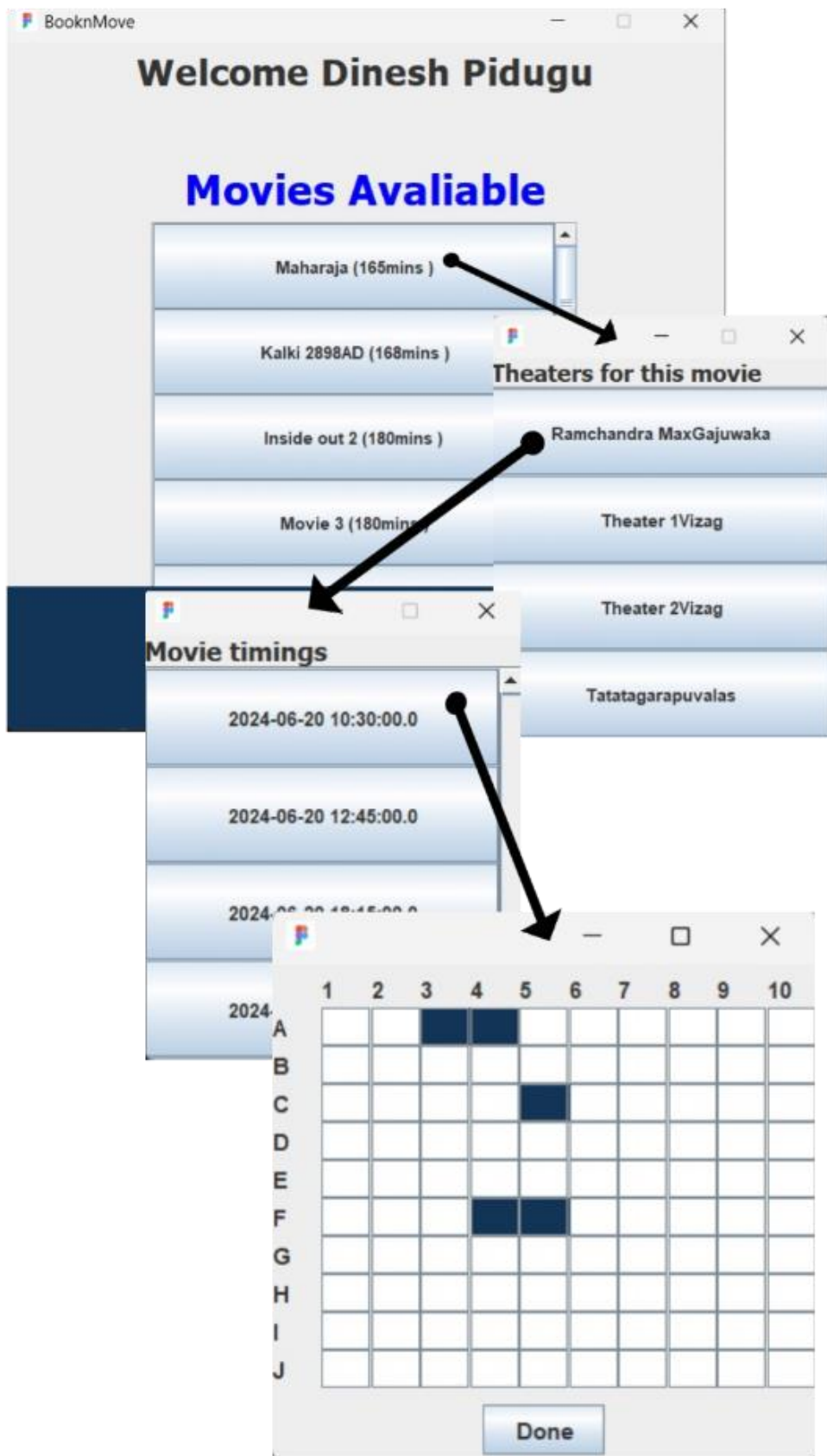
4.2 Development

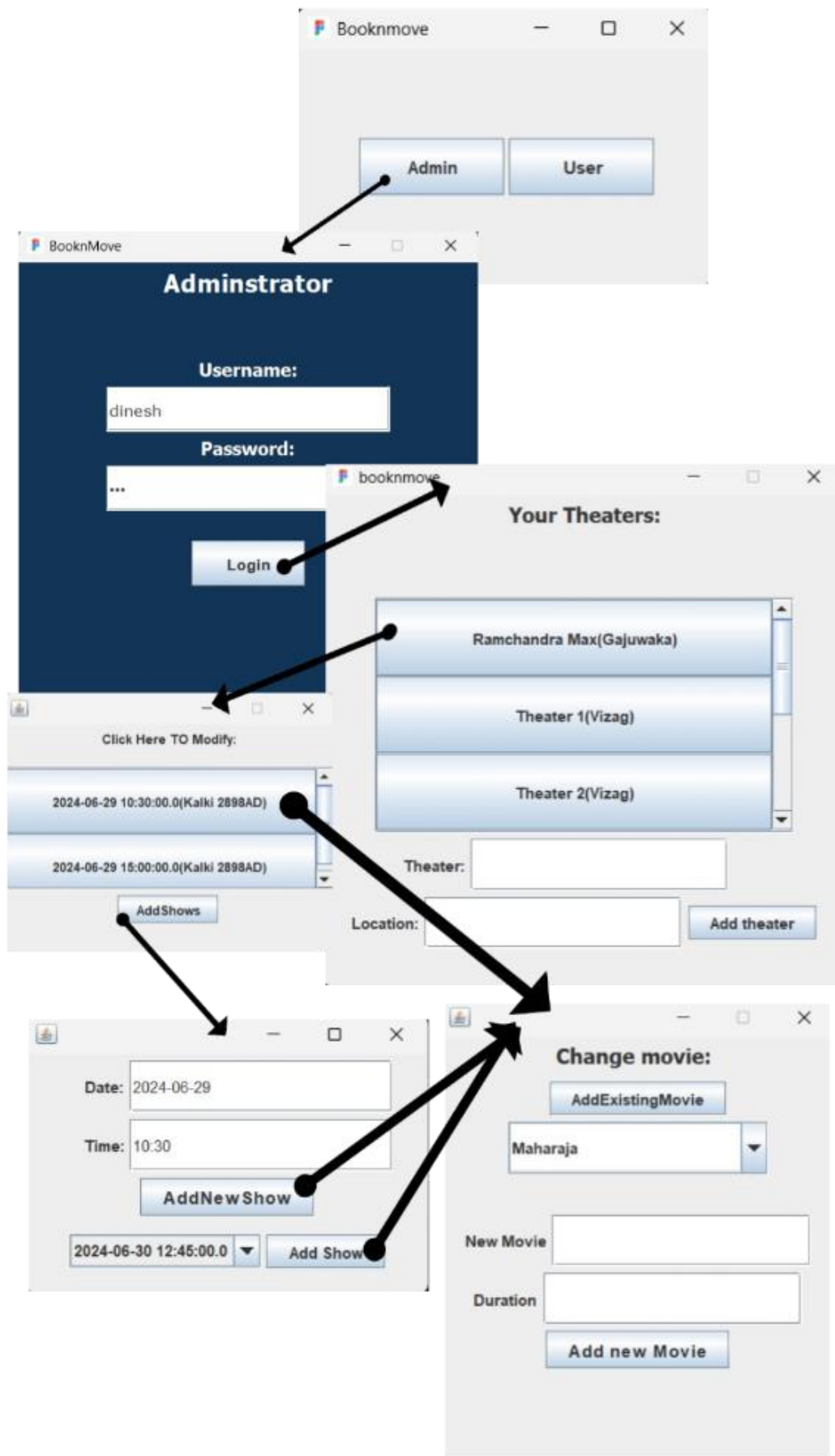
- **User Interface**: Developed using Java Swing, providing screens for browsing movies, selecting showtimes, and booking tickets.
- **Backend**: Implemented using Java and JDBC to handle data operations and business logic.
- **Database**: SQL database designed to store movie information, showtimes, user details, bookings, and theater information.
- **Admin Interface**: Developed using Java Swing, allowing admins to add, update, and delete theaters, movies, and showtimes.

4.3 Challenges

Challenges included ensuring database connectivity, handling concurrent bookings, creating an intuitive user interface, and implementing robust admin functionalities. These were addressed by optimizing SQL queries, using synchronized methods, and refining the GUI design.







5. Admin Functionalities

5.1 Theater Management

The admin interface allows administrators to add, update, and delete theater information. This includes theater names, locations, and available screens.

5.2 Movie Managemet

Admins can manage movie information, including adding new movies, updating existing movie details, and removing movies from the database

5.3 Showtimes Management

Admins can set and modify showtimes for different movies in various theaters. This includes scheduling new showtimes, updating existing ones, and canceling showtimes.

6. Results

6.1 Data

The system was tested with sample data including multiple movies, showtimes, and theater information. Results showed efficient booking processes and accurate data storage.

6.2 Analysis

The system handled multiple concurrent users without performance degradation. Admin functionalities were tested to ensure accurate management of theaters, movies, and showtimes. User feedback indicated that the interface was easy to use and the booking process was smooth.

7. Discussion

7.1 Interpretation

The results demonstrate that the Java-based movie ticket booking system is capable of managing bookings efficiently. The use of Java Swing and JDBC proved effective for developing the system, and the admin functionalities provided robust management options.

7.2 Comparison

Compared to existing systems, this project offers a simpler alternative with a focus on core functionalities, making it suitable for small to medium-sized cinemas.

7.3 Implications

This project shows the potential for using Java in developing robust ticket booking systems. It provides a foundation that can be expanded with additional features like online payment integration and mobile app support.

8. Conclusion

8.1 Summary

This project successfully developed a movie ticket booking system using Java. The system allows users to browse movies, select showtimes, and book tickets efficiently. Additionally, the admin interface enables effective management of theaters, movies, and showtimes.

8.2 Future Work

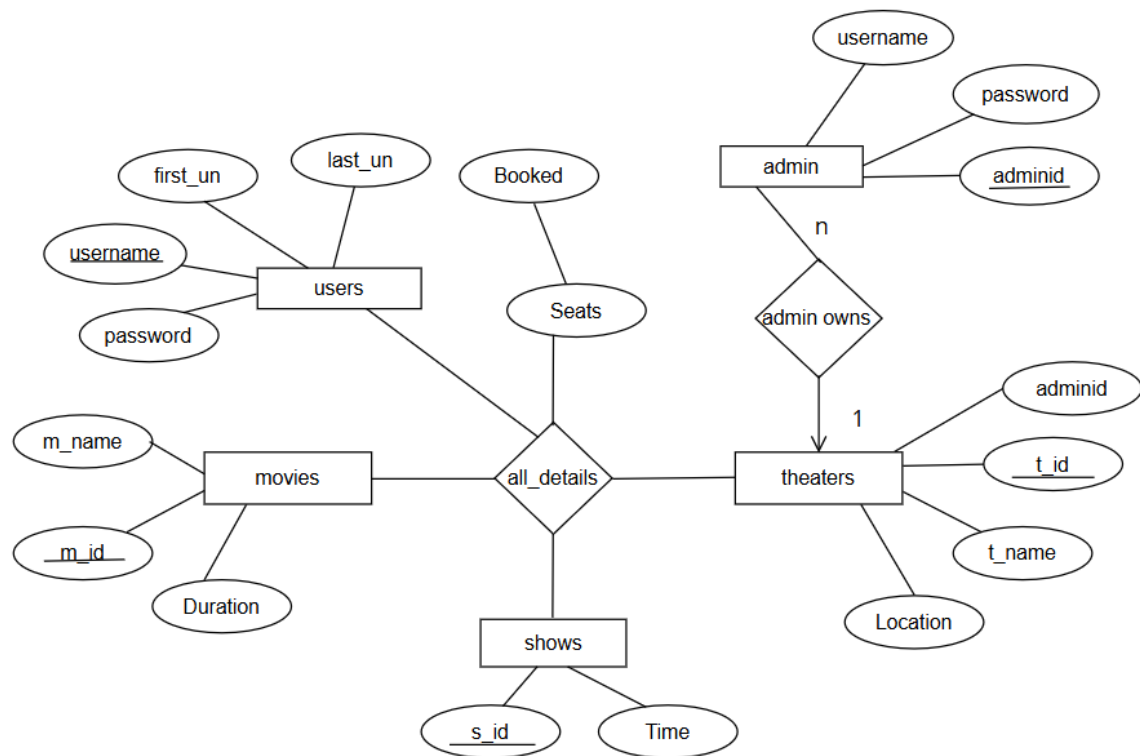
Future improvements could include integrating online payment gateways, adding more advanced search functionalities, and developing a mobile application version of the system.

9. Appendices

Appendix A: ER Diagram

```
SQL> CREATE OR REPLACE PROCEDURE INSERT_SEATS(mid_ IN NUMBER, tid_ IN NUMBER, SHOWID IN NUMBER)
  2  IS
  3      c CHAR(1);
  4  BEGIN
  5      FOR j IN 0..9 LOOP
  6          c := CHR(65 + j);
  7          FOR k IN 1..10 LOOP
  8              INSERT INTO all_details(mid, tid, sid, bool, show_id)
  9                  VALUES (mid_, tid_, c || TO_CHAR(k), 0, SHOWID);
 10          END LOOP;
 11      END LOOP;
 12  END;
 13  /

Procedure created.
```



Appendix B: Source Code

```
import javax.swing.DefaultComboBoxModel;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JComboBox;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.JPasswordField;
import javax.swing.JScrollPane;
import javax.swing.JTextField;
import java.awt.*;
import java.sql.*;
```

```

import java.util.ArrayList;
import java.util.stream.Collectors;

public class loginform {
    static Color back = new Color(0x123456);
    static ImageIcon logo=new ImageIcon("figma_logo.png");
    public static JFrame page1 = new JFrame();
    static Font labelFont = new Font("Tahoma", Font.BOLD, 16);
    static Font textFieldFont = new Font("Verdana", Font.PLAIN, 14);
    static Font buttonFont = new Font("SansSerif", Font.BOLD, 14);
    static Cursor curs=new Cursor(Cursor.HAND_CURSOR);
    public static void main(String [] args) throws ClassNotFoundException, SQLException {
        JFrame homepage = new JFrame("Booknmove");
        homepage.setIconImage(logo.getImage());
        homepage.setSize(300,200);
        homepage.setLocationRelativeTo(null);
        homepage.setLayout(new GridBagLayout());
        homepage.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        JButton admin = new JButton("Admin");
        admin.setPreferredSize(new Dimension(100, 40));
        admin.setFocusable(false);
        admin.setCursor(curs);
        admin.addActionListener(e->{
            try {
                homepage.dispose();
                admin();
            } catch (ClassNotFoundException e1) {
                e1.printStackTrace();
            } catch (SQLException e1) {
                e1.printStackTrace();
            }
        });
    }
}

```

```

    }
});
JButton user =new JButton("User");
user.setFocusable(false);
user.setPreferredSize(new Dimension(100, 40));
user.setCursor(curs);
user.addActionListener(e->{
    try {
        homepage.dispose();
        user();
    } catch (ClassNotFoundException e1) {
        e1.printStackTrace();
    } catch (SQLException e1) {
        e1.printStackTrace();
    }
});
homepage.add(new JLabel(" "));
homepage.add(admin);
homepage.add(new JLabel(" "));
homepage.add(user);
homepage.setVisible(true);
}

public static void user() throws ClassNotFoundException, SQLException{
    Class.forName("oracle.jdbc.driver.OracleDriver");
    Connection connection = DriverManager.getConnection(
        "jdbc:oracle:thin:@localhost:1521:xe",
        "C##booknmove",
        "dinesh"
    );
    Statement stmt=connection.createStatement();

```

```
page1.setTitle("BooknMove");
page1.setSize(420, 420);
page1.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
page1.setLocationRelativeTo(null);
page1.getContentPane().setBackground(back);
page1.setIconImage(logo.getImage());
page1.setResizable(false);
page1.setLayout(new BorderLayout());
JPanel northpanel = new JPanel();
northpanel.setBackground(back);
page1.add(northpanel, BorderLayout.NORTH);
northpanel.setPreferredSize(new Dimension(100, 80));
JLabel namelabel = new JLabel();
namelabel.setForeground(Color.WHITE);
northpanel.add(namelabel);
namelabel.setFont(new Font("Tahoma", Font.BOLD, 22));
JPanel westpanel = new JPanel();
westpanel.setPreferredSize(new Dimension(50, 0));
westpanel.setBackground(back);
page1.add(westpanel, BorderLayout.WEST);
JPanel centerpanel = new JPanel();
page1.add(centerpanel, BorderLayout.CENTER);
centerpanel.setBackground(back);
JLabel username = new JLabel("Username:");
username.setForeground(Color.white);
username.setFont(labelFont);
username.setVerticalAlignment(JLabel.TOP);
username.setHorizontalTextPosition(JLabel.LEFT);
centerpanel.add(username);
JTextField email = new JTextField("example@gmail.com");
```

```

email.setPreferredSize(new Dimension(250, 40));
centerpanel.add(email);
email.setFont(textFieldFont);
JLabel pass = new JLabel("Password:");
pass.setForeground(Color.white);
centerpanel.add(pass);
pass.setFont(labelFont);
JPasswordField password = new JPasswordField();
password.setPreferredSize(new Dimension(250, 40));
centerpanel.add(password);
JLabel gap = new JLabel(" ");
gap.setSize(250, 40);
centerpanel.add(gap);
JButton signup = new JButton("Signup");
signup.setPreferredSize(new Dimension(100, 40));
signup.setFocusable(false);
centerpanel.add(signup);
signup.setFont(buttonFont);
signup.setCursor(new Cursor(Cursor.HAND_CURSOR));
signup.addActionListener(e->{
    page1.dispose();
    try {
        signup();
    } catch (ClassNotFoundException | SQLException e1) {
        e1.printStackTrace();
    }
});
JLabel gap1 = new JLabel(" ");
centerpanel.add(gap1);
JButton signin = new JButton("Login");

```

```

signin.setPreferredSize(new Dimension(100, 40));
signin.setFocusable(false);
centerpanel.add(signin);
signin.setFont(buttonFont);
signin.setCursor(new Cursor(Cursor.HAND_CURSOR));
JPanel eastpanel = new JPanel();
eastpanel.setPreferredSize(new Dimension(50, 0));
eastpanel.setBackground(back);
page1.add(eastpanel, BorderLayout.EAST);
JPanel southpanel = new JPanel();
southpanel.setBackground(back);
page1.add(southpanel, BorderLayout.SOUTH);
southpanel.setPreferredSize(new Dimension(100, 40));
JLabel errormsg = new JLabel();
errormsg.setForeground(new Color(0xFF0000));
errormsg.setFont(new Font("Roboto", Font.PLAIN, 20));
southpanel.add(errormsg);
signin.addActionListener(sign -> {
    String u_name = email.getText();
    char[] passwordChars = password.getPassword();
    String u_pass = new String(passwordChars);
    if (u_name == null || u_name.trim().isEmpty() || u_pass == null ||
u_pass.trim().isEmpty()) {
        errormsg.setText("Enter Valid Details");
    } else {
        try {
            ResultSet rs=stmt.executeQuery("select * from users where
username='"+u_name+"' and pass='"+u_pass+"'");
            if(rs.next()){
                page1.dispose();
                home(u_name, u_pass, rs.getString(1), rs.getString(2));
            }
        } catch (SQLException e) {
            errormsg.setText(e.getMessage());
        }
    }
});

```



```

        }
        else{
            errmsg.setText("INCORRECT username/password**");
        }
        connection.close();
    } catch (SQLException ex) {
        System.out.print(ex);
        errmsg.setText("Database error");
    } catch (ClassNotFoundException e1) {
        e1.printStackTrace();
    }
}

});
page1.setVisible(true);
}

public static void admin () throws SQLException, ClassNotFoundException{
    Class.forName("oracle.jdbc.driver.OracleDriver");
    Color back = new Color(0x123456);
    page1.setTitle("BooknMove");
    page1.setSize(420, 420);
    page1.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    page1.setLocationRelativeTo(null);
    page1.getContentPane().setBackground(back);
    page1.setIconImage(logo.getImage());
    page1.setResizable(false);
    page1.setLayout(new BorderLayout());
    JPanel northpanel = new JPanel();
    northpanel.setBackground(back);
    page1.add(northpanel, BorderLayout.NORTH);
    northpanel.setPreferredSize(new Dimension(100, 80));

```

```
JLabel namelabel = new JLabel("Adminstrator");
namelabel.setForeground(Color.WHITE);
northpanel.add(namelabel);
namelabel.setFont(new Font("Tahoma", Font.BOLD, 22));
JPanel westpanel = new JPanel();
westpanel.setPreferredSize(new Dimension(50, 0));
westpanel.setBackground(back);
page1.add(westpanel, BorderLayout.WEST);
JPanel centerpanel = new JPanel();
page1.add(centerpanel, BorderLayout.CENTER);
centerpanel.setBackground(back);
JLabel username = new JLabel("Username:");
username.setForeground(Color.white);
username.setFont(labelFont);
username.setVerticalAlignment(JLabel.TOP);
username.setHorizontalTextPosition(JLabel.LEFT);
centerpanel.add(username);
JTextField email = new JTextField("dinesh");
email.setPreferredSize(new Dimension(250, 40));
centerpanel.add(email);
email.setFont(textFieldFont);
JLabel pass = new JLabel("Password:");
pass.setForeground(Color.white);
centerpanel.add(pass);
pass.setFont(labelFont);
JPasswordField password = new JPasswordField("167");
password.setPreferredSize(new Dimension(250, 40));
centerpanel.add(password);
JLabel gap = new JLabel(" ");
gap.setSize(250, 40);
```

```

centerpanel.add(gap);
JLabel gap1 = new JLabel(" ");
centerpanel.add(gap1);
JButton signin = new JButton("Login");
signin.setPreferredSize(new Dimension(100, 40));
signin.setFocusable(false);
centerpanel.add(signin);
signin.setFont(buttonFont);
signin.setCursor(new Cursor(Cursor.HAND_CURSOR));
JPanel eastpanel = new JPanel();
eastpanel.setPreferredSize(new Dimension(50, 0));
eastpanel.setBackground(back);
page1.add(eastpanel, BorderLayout.EAST);
JPanel southpanel = new JPanel();
southpanel.setBackground(back);
page1.add(southpanel, BorderLayout.SOUTH);
southpanel.setPreferredSize(new Dimension(100, 40));
JLabel errormsg = new JLabel();
errormsg.setForeground(new Color(0xFF0000));
errormsg.setFont(new Font("Roboto", Font.PLAIN, 20));
southpanel.add(errormsg);
signin.addActionListener(e->{
    try (Connection connection = DriverManager.getConnection(
        "jdbc:oracle:thin:@localhost:1521:xe",
        "C##booknmove",
        "dinesh"
    )) {
        Statement stmt=connection.createStatement();
        String u_name = email.getText();
        char[] passwordChars = password.getPassword();

```

```

        String u_pass = new String(passwordChars);

        if (u_name == null || u_name.trim().isEmpty() || u_pass == null ||
u_pass.trim().isEmpty()) {
            errmsg.setText("Enter Valid Details");
        } else {
            try {
                ResultSet rs=stmt.executeQuery("select * from admin where
username='"+u_name+"' and password='"+u_pass+"'");
                if(rs.next()){
                    page1.dispose();
                    adminhome(rs.getInt(3));
                }
                else{
                    errmsg.setText("INCORRECT username/password*");
                }
                connection.close();
            } catch (SQLException ex) {
                System.out.print(ex);
                errmsg.setText("Database error");
            } catch (ClassNotFoundException e1) {
                e1.printStackTrace();
            }
        }
    } catch (SQLException e1) {
        e1.printStackTrace();
    }
});
page1.setVisible(true);
}

public static void home(String u_name,String password,String firstname,String lastname)
throws SQLException,ClassNotFoundException{

```

```
Connection connection = DriverManager.getConnection(
    "jdbc:oracle:thin:@localhost:1521:xe",
    "C##booknmove",
    "dinesh"
);
Statement stmt=connection.createStatement();

JFrame page2 = new JFrame();
page2.setTitle("BooknMove");
page2.setSize(520, 520);
page2.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
page2.setLocationRelativeTo(null);
page2.setIconImage(new ImageIcon("figma_logo.png").getImage());
page2.setResizable(false);
page2.setLayout(new BorderLayout());

JPanel northpanel1=new JPanel();
page2.add(northpanel1,BorderLayout.NORTH);
northpanel1.setPreferredSize(new Dimension(100,80));
JLabel welcomLabel=new JLabel("Welcome "+firstname+" "+lastname);
northpanel1.add(welcomLabel);
welcomLabel.setFont(new Font("Tahoma", Font.BOLD, 26));

JPanel sorthpanel1=new JPanel();
sorthpanel1.setBackground(back);
page2.add(sorthpanel1,BorderLayout.SOUTH);
sorthpanel1.setPreferredSize(new Dimension(150,100));
    JButton viewbookings=new JButton("ViewBookings");
    viewbookings.setFocusable(false);
    viewbookings.setCursor(curs);
    sorthpanel1.add(viewbookings);

JPanel eastpanel1=new JPanel();
page2.add(eastpanel1,BorderLayout.EAST);
```

```

eastpanel1.setPreferredSize(new Dimension(100,100));
JPanel westpanel1=new JPanel();
page2.add(westpanel1,BorderLayout.WEST);
westpanel1.setPreferredSize(new Dimension(100,100));
JPanel centerpanel1 =new JPanel();
page2.add(centerpanel1,BorderLayout.CENTER);
JLabel mviea=new JLabel("Movies Aavailable");
centerpanel1.add(mviea);
mviea.setForeground(Color.blue);
mviea.setFont(new Font("Tahoma", Font.BOLD, 30));
ResultSet rs1=stmt.executeQuery("Select count(m_name) as moviec from movies");
rs1.next();
int moviecount=rs1.getInt(1);
JPanel buttonPanel = new JPanel();
buttonPanel.setLayout(new GridLayout(moviecount, 1));
rs1=stmt.executeQuery("select * from movies");
while(rs1.next()) {
    String moviename=rs1.getString(1);
    int duration=rs1.getInt(2);
    JButton x =new JButton(moviename+" (" +duration+"mins )");
    x.setPreferredSize(new Dimension(30,60));
    x.setFocusable(false);
    x.setCursor(new Cursor(Cursor.HAND_CURSOR));
    int movieid=rs1.getInt(3);
    x.addActionListener(f->{
        try{
            Connection connection1 = DriverManager.getConnection(
                "jdbc:oracle:thin:@localhost:1521:xe",
                "C##booknmove",
                "dinesh"
            );
        }
    });
}
}

```

```

    );

    Statement stmt1=connection1.createStatement();

    ResultSet restherater1=stmt1.executeQuery("select count(DISTINCT (tid))
from all_details where mid="+movieid);

    restherater1.next();

    JFrame thpage=new JFrame();
    thpage.setResizable(false);
    thpage.setLocationRelativeTo(null);
    thpage.setIconImage(logo.getImage());
    thpage.setSize(new Dimension(250,300));
    thpage.setLayout(new BorderLayout());

    Statement stmt2=connection1.createStatement();

    ResultSet restherater2=stmt2.executeQuery("select DISTINCT (tid) from
all_details where mid="+movieid);

    JLabel movie_the=new JLabel("Theaters for this movie");
    movie_the.setSize(new Dimension(250,30));
    movie_the.setFont(labelFont);
    thpage.add(movie_the,BorderLayout.NORTH);

    JPanel timpanel =new JPanel();

    System.out.println(restherater1.getInt(1));

    timpanel.setLayout(new GridLayout(restherater1.getInt(1),1));
    while(restherater2.next()){

        Statement stmt3=connection1.createStatement();

        ResultSet restherater3=stmt3.executeQuery("select *from Theaters where
t_id="+restherater2.getInt(1));

        restherater3.next();

        String theater=restherater3.getString(2);

        JButton y =new JButton(theater+restherater3.getString(3));

        y.setPreferredSize(new Dimension(30,60));

        y.setFocusable(false);

        y.setCursor(new Cursor(Cursor.HAND_CURSOR));

```

```

int theaterid=restherater2.getInt(1);
y.addActionListener(g->{
    try {Connection connection2 = DriverManager.getConnection(
        "jdbc:oracle:thin:@localhost:1521:xe",
        "C##booknmove",
        "dinesh"
    );
    Statement stmt4=connection2.createStatement();
    ResultSet resdate=stmt4.executeQuery("select
Count(distinct(show_id)) from all_details where mid="+movieid+" and tid="+theaterid);
    resdate.next();
    JFrame timpage=new JFrame();
    timpage.setResizable(false);
    timpage.setIconImage(logo.getImage());
    timpage.setSize(new Dimension(250,300));
    timpage.setLayout(new BorderLayout());
    timpage.setLocationRelativeTo(null);
    JLabel movie_tim=new JLabel("Movie timings");
    movie_tim.setSize(new Dimension(250,30));
    movie_tim.setFont(labelFont);
    timpage.add(movie_tim,BorderLayout.NORTH);
    Statement stmt5=connection2.createStatement();
    ResultSet resdate1=stmt5.executeQuery("select distinct(show_id)
from all_details where mid="+movieid+" and tid="+theaterid);
    JPanel timeanddate = new JPanel();
    timeanddate.setLayout(new GridLayout(resdate.getInt(1),1));
    while(resdate1.next()){
        int showid=resdate1.getInt(1);
        Statement stmt6=connection2.createStatement();
        ResultSet resdate2=stmt6.executeQuery("select time from shows
where sid="+showid);

```



```

resdate2.next();

Timestamp time=resdate2.getTimestamp(1);

JButton z =new JButton(time.toString());

z.setPreferredSize(new Dimension(30,60));

z.setFocusable(false);

z.setCursor(new Cursor(Cursor.HAND_CURSOR));

z.addActionListener(tickets->{

    try{

        Connection connection3 = DriverManager.getConnection(

            "jdbc:oracle:thin:@localhost:1521:xe",

            "C##booknmove",

            "dinesh"

        );

        Statement stmt7=connection3.createStatement();

        ResultSet reseats=stmt7.executeQuery("select sid,bool from

all_details where mid="+movieid+" and tid="+theaterid+" and show_id="+showid);

        JFrame ticketpage=new JFrame();

        ticketpage.setIconImage(logo.getImage());

        ticketpage.setSize(new Dimension(300,300));

        ticketpage.setLayout(new BorderLayout());

        ticketpage.setLocationRelativeTo(null);

        JPanel ticks=new JPanel();

        ticketpage.add(ticks,BorderLayout.CENTER);

        ticks.setLayout(new GridLayout(11,11));

        JLabel emp = new JLabel(" ");

        ticks.add(emp);

        ArrayList<String> seats=new ArrayList<String>();

        for(int eaha=1;eaha<=10;eaha++){

            JLabel emp1 =new JLabel(eaha+"");

            ticks.add(emp1);

        }

```

```

for(int row=1;row<=10;row++){
    char xie= (char)(row+64);
    JLabel emp1= new JLabel(xie+"");
    ticks.add(emp1);
    for(int col=1;col<=10;col++){
        reseats.next();
        String temp=reseats.getString(1);
        JButton tick = new JButton();
        if(reseats.getInt(2)==1){
            tick.setBackground(new Color(0x123456));
            tick.setEnabled(false);
        }
        else{
            tick.setCursor(new
Cursor(Cursor.HAND_CURSOR));
            tick.setFocusable(false);
            tick.setBackground(Color.WHITE);
            tick.addActionListener(ee-> {
                if (seats.contains(temp)) {
                    tick.setBackground(Color.WHITE);
                    seats.remove(temp);
                } else {
                    tick.setBackground(new Color(0x34eb83));
                    seats.add(temp);
                }
            });
        }
        ticks.add(tick);
    }
}
ticketpage.add(ticks);

```

```

JPanel done =new JPanel();

JButton Donebtn = new JButton("Done");

Donebtn.setFocusable(false);

done.add(Donebtn);

Donebtn.addActionListener(recipt->{

    try{

        Connection                connection4                =

DriverManager.getConnection(

        "jdbc:oracle:thin:@localhost:1521:xe",

        "C##booknmove",

        "dinesh"

    );

    for(String seat : seats){

        Statement stmt8=connection4.createStatement();

        System.out.println(seat+"booked");

        String                Query="insert                into

recipts(uname,theater,movie,time,duration,s_no)

values(""+u_name+"",""+theater+"",""+moviename+"", TO_TIMESTAMP(""+time+"", 'YYYY-

MM-DD HH24:MI:SS.FF'),"+duration+"",""+seat+"")";

        stmt8.executeQuery(Query);

        System.out.println(Query);

        Statement stmt9=connection4.createStatement();

        Query="update  all_details  set  bool=1  where

mid="+movieid+" and tid="+theaterid+" and show_id="+showid+" and sid="+seat+"";

        stmt9.executeQuery(Query);

        System.out.println(Query);

    }

    seats.clear();

    Statement stmt10=connection4.createStatement();

    stmt10.executeQuery("commit");

    connection4.close();

    ticketpage.dispose();

```

```

        JOptionPane.showMessageDialog(null, "Ticket Booked
Successfully");

    }
    catch(Exception hell){
        System.out.println(hell);
    }
});
ticketpage.add(done, BorderLayout.SOUTH);
ticketpage.setVisible(true);
connection3.close();
}
catch(SQLException last){
    System.out.println(last.getMessage());
}
});
timeanddate.add(z);
}
JScrollPane timesc=new JScrollPane(timeanddate);
timpage.add(timesc, BorderLayout.CENTER);

timesc.setVerticalScrollBarPolicy(JScrollPane.VERTICAL_SCROLLBAR_ALWAYS);

timpage.setVisible(true);
connection2.close();
}
catch(SQLException fi){
    fi.printStackTrace();
}
});
timpanel.add(y);
}
JScrollPane timscroll =new JScrollPane(timpanel);

```

```

        thpage.add(timscroll, BorderLayout.CENTER);
        thpage.setVisible(true);
        connection1.close();
    }
    catch(Exception xie){
        System.out.println(xie);
    }
});
buttonPanel.add(x);
}
JScrollPane scrollPane = new JScrollPane(buttonPanel);
scrollPane.setPreferredSize(new Dimension(300, 260));

scrollPane.setVerticalScrollBarPolicy(JScrollPane.VERTICAL_SCROLLBAR_ALWAYS);
centerpanel1.add(scrollPane);
page2.setVisible(true);
}

public static void signup() throws SQLException, ClassNotFoundException {
    JFrame page11 = new JFrame();
        page11.setTitle("BooknMove");
        page11.setSize(420, 420);
        page11.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        page11.setLocationRelativeTo(null);
        page11.getContentPane().setBackground(back);
        page11.setIconImage(logo.getImage());
        page11.setResizable(false);
        page11.setLayout(new BorderLayout());
        JPanel northpanel1 = new JPanel();
        northpanel1.setBackground(back);
        page11.add(northpanel1, BorderLayout.NORTH);
        northpanel1.setPreferredSize(new Dimension(100, 80));

```

```
JLabel namelabel1 = new JLabel("SignUP");
namelabel1.setForeground(Color.WHITE);
northpanel1.add(namelabel1);
namelabel1.setFont(new Font("Tahoma", Font.BOLD, 22));
JPanel westpanel1 = new JPanel();
westpanel1.setPreferredSize(new Dimension(40, 0));
westpanel1.setBackground(back);
page11.add(westpanel1, BorderLayout.WEST);
JPanel centerpanel1 = new JPanel();
page11.add(centerpanel1, BorderLayout.CENTER);
centerpanel1.setBackground(back);
JLabel fname=new JLabel("FirstName:");
fname.setForeground(Color.white);
fname.setFont(labelFont);
fname.setVerticalAlignment(JLabel.TOP);
fname.setHorizontalTextPosition(JLabel.LEFT);
centerpanel1.add(fname);
JTextField f_name= new JTextField();
f_name.setPreferredSize(new Dimension(200, 40));
centerpanel1.add(f_name);
f_name.setFont(textFieldFont);
JLabel lname=new JLabel("LastName:");
lname.setForeground(Color.white);
lname.setFont(labelFont);
lname.setVerticalAlignment(JLabel.TOP);
lname.setHorizontalTextPosition(JLabel.LEFT);
centerpanel1.add(lname);
JTextField l_name= new JTextField();
l_name.setPreferredSize(new Dimension(200, 40));
centerpanel1.add(l_name);
```

```
l_name.setFont(textFieldFont);
JLabel username1 = new JLabel("Username:");
username1.setForeground(Color.white);
username1.setFont(labelFont);
username1.setVerticalAlignment(JLabel.TOP);
username1.setHorizontalTextPosition(JLabel.LEFT);
centerpanel1.add(username1);
JTextField email1 = new JTextField("example@gmail.com");
email1.setPreferredSize(new Dimension(200, 40));
centerpanel1.add(email1);
email1.setFont(textFieldFont);
JLabel pass1 = new JLabel("Password:");
pass1.setForeground(Color.white);
centerpanel1.add(pass1);
pass1.setFont(labelFont);
centerpanel1.add(new JLabel(" "));
JPasswordField password1 = new JPasswordField();
password1.setPreferredSize(new Dimension(200, 40));
centerpanel1.add(password1);
JLabel gap1 = new JLabel(" ");
gap1.setSize(250, 40);
centerpanel1.add(gap1);
JButton signin1 = new JButton("Login");
signin1.setPreferredSize(new Dimension(100, 40));
signin1.setFocusable(false);
centerpanel1.add(signin1);
signin1.setFont(buttonFont);
signin1.setCursor(new Cursor(Cursor.HAND_CURSOR));
JPanel eastpanel1 = new JPanel();
eastpanel1.setPreferredSize(new Dimension(40, 0));
```

```

        eastpanel1.setBackground(back);
        page11.add(eastpanel1, BorderLayout.EAST);
        JPanel southpanel1 = new JPanel();
        southpanel1.setBackground(back);
        page11.add(southpanel1, BorderLayout.SOUTH);
        southpanel1.setPreferredSize(new Dimension(100, 40));
        JLabel errormsg1 = new JLabel();
        errormsg1.setForeground(new Color(0xFF0000));
        errormsg1.setFont(new Font("Roboto", Font.PLAIN, 20));
        southpanel1.add(errormsg1);
        signin1.addActionListener(sign->{
            String
firstname=f_name.getText(),lastname=l_name.getText(),user_name=email1.getText();
            char[] passwordChars = password1.getPassword();
            String pass_word = new String(passwordChars);
            if((firstname==null||firstname.trim().isEmpty())){
                errormsg1.setText("Enter Your FirstName");
            }
            else if(lastname==null||lastname.trim().isEmpty()){
                errormsg1.setText("Enter Your LastName");
            }
            else if(user_name==null||user_name.trim().isEmpty()){
                errormsg1.setText("Enter your Email");
            }
            else if(pass_word==null||pass_word.trim().isEmpty()){
                errormsg1.setText("Create a Password");
            }
            else if(pass_word.length()<8){
                errormsg1.setText("Password must be 8 charatcer");
            }
            else{

```



```

try {
    Connection connection11 = DriverManager.getConnection(
        "jdbc:oracle:thin:@localhost:1521:xe",
        "C##booknmove",
        "dinesh"
    );
    Statement stmt11=connection11.createStatement();
    ResultSet res1= stmt11.executeQuery("select * from users where
username='"+user_name+"'");
    if(res1.next()){
        errorMsg1.setText("User Already exists");
        JButton loginhereButton=new JButton("sign in");
        southpanel1.add(loginhereButton);
        loginhereButton.addActionListener(action1->{
            page11.dispose();
            try {
                user();
            } catch (ClassNotFoundException | SQLException e1) {
                e1.printStackTrace();
            }
        });
    }
    else{
        Statement stmt12=connection11.createStatement();
        stmt12.executeQuery("insert into users(first_un,last_un,username,pass)
values '"+firstname+"','"+lastname+"','"+user_name+"','"+pass_word+"'");
        errorMsg1.setText("Login sucessful");
        errorMsg1.setForeground(new Color(0x00FF00));
        Statement stmt13=connection11.createStatement();
        stmt13.executeQuery("Commit");
        Thread.sleep(1000);
    }
}

```

```

        page11.dispose();
        home(user_name, pass_word, firstname, lastname);
    }
}
catch(SQLException ex1){
    System.out.println(ex1);
} catch (ClassNotFoundException e1) {
    e1.printStackTrace();
} catch (InterruptedException e1) {
    e1.printStackTrace();
}
}
});
page11.setVisible(true);
}

public static void adminhome(int adminid) throws
SQLException,ClassNotFoundException{
    Connection connection2 = DriverManager.getConnection(
        "jdbc:oracle:thin:@localhost:1521:xe",
        "C##booknmove",
        "dinesh"
    );
    JFrame homepage =new JFrame("booknmove");
    homepage.setSize(new Dimension(420,420));
    homepage.setResizable(false);
    homepage.setLocationRelativeTo(null);
    homepage.setLayout(new BorderLayout());
    homepage.setIconImage(logo.getImage());
    homepage.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    try{
        Statement stmt=connection2.createStatement();

```

```

        ResultSet res2=stmt.executeQuery("select count(t_id) from theaters where
adid="+adminid);
        res2.next();
        int nooftheaters=res2.getInt(1);
        ResultSet res1= stmt.executeQuery("select * from theaters where adid="+adminid);
        JPanel theaterlist =new JPanel();
        theaterlist.setLayout(new GridLayout(nooftheaters,1));
        while(res1.next()){
            JButton y =new JButton(res1.getString(2)+"("+res1.getString(3)+")");
            y.setPreferredSize(new Dimension(30,60));
            y.setFocusable(false);
            y.setCursor(new Cursor(Cursor.HAND_CURSOR));
            int theaterid=res1.getInt(1);
            theaterlist.add(y);
            y.addActionListener(action2->{
                JFrame admintime=new JFrame();
                admintime.setSize(new Dimension(320,250));
                admintime.setResizable(false);
                admintime.setLocationRelativeTo(null);
                admintime.setLayout(new BorderLayout());
                JPanel timePanel=new JPanel();
                ArrayList<Integer> showsnotin= new ArrayList<Integer>();
                try{
                    Connection connection3 = DriverManager.getConnection(
                        "jdbc:oracle:thin:@localhost:1521:xe",
                        "C##booknmove",
                        "dinesh"
                    );
                    Statement stmt2=connection3.createStatement();
                    ResultSet res3=stmt2.executeQuery("select count(distinct(show_id)) from
all_details where tid="+theaterid);

```

```

        res3.next();

        int showcount=res3.getInt(1);

        timePanel.setLayout(new GridLayout(showcount,1));

        // String filtershows;

        ResultSet res4=stmt2.executeQuery("select distinct(show_id) from all_details
where tid = "+theaterid);

        while(res4.next()){

            Statement stmt3=connection3.createStatement();

            ResultSet res5=stmt3.executeQuery("select * from shows where
sid="+res4.getInt(1));

            showsnotin.add(res4.getInt(1));

            res5.next();

            Statement stmt4=connection3.createStatement();

            int showid =res5.getInt(1);

            ResultSet res6=stmt4.executeQuery("select distinct(mid) from all_details
where show_id="+showid+" and tid="+theaterid);

            res6.next();

            Statement stmt5=connection3.createStatement();

            int movieid= res6.getInt(1);

            ResultSet res7=stmt5.executeQuery("select m_name from movies where
m_id="+movieid);

            res7.next();

            Timestamp time=res5.getTimestamp(2);

            JButton z =new JButton(time.toString()+"("+res7.getString(1)+")");

            z.setPreferredSize(new Dimension(30,60));

            z.setFocusable(false);

            z.setCursor(new Cursor(Cursor.HAND_CURSOR));

            timePanel.add(z);

            z.addActionListener(action3->{

                JFrame showspage=new JFrame();

                showspage.setLocationRelativeTo(null);

                showspage.setLayout(new BorderLayout());

```

```

showspage.setSize(320,370);
showspage.setResizable(false);
JPanel top=new JPanel();
showspage.add(top, BorderLayout.NORTH);
JLabel change = new JLabel("Change movie: ");
change.setFont(labelFont);
top.add(change);
JPanel centerpanel=new JPanel();
showspage.add(centerpanel, BorderLayout.CENTER);
JComboBox<String> selectMovie=new JComboBox<>();
DefaultComboBoxModel<String> model =new
DefaultComboBoxModel<String>();
selectMovie.setModel(model);
try {
    Connection connection4 = DriverManager.getConnection(
        "jdbc:oracle:thin:@localhost:1521:xe",
        "C##booknmove",
        "dinesh"
    );
    Statement stmt6=connection4.createStatement();
    ResultSet res8=stmt6.executeQuery("select m_name from movies
where not m_id="+movieid);
    while(res8.next()){
        model.addElement(res8.getString(1));
    }
    centerpanel.add(selectMovie);
    JButton changemovie=new JButton("Change Movie");
    changemovie.setFocusable(false);
    changemovie.setCursor(curs);
    centerpanel.add(changemovie);
    changemovie.addActionListener(e->{

```

```

        try {
            String selectedMovie=(String)selectMovie.getSelectedItemAt();
            Statement stmt8= connection4.createStatement();
            ResultSet res9=stmt8.executeQuery("select m_id from movies
where m_name='"+selectedMovie+"'");
            res9.next();
            int selectedMovieid=res9.getInt(1);
            Statement stmt7 =connection4.createStatement();
            stmt7.executeQuery("update          all_details          set
mid='"+selectedMovieid+"',bool=0 where show_id='"+showid+" and tid='"+theaterid+" ");
            stmt7.executeQuery("commit");
            showspage.dispose();
        } catch (SQLException e1) {
            e1.printStackTrace();
        }
    });
}

catch(Exception e){
    System.out.println(e.getMessage());
}

selectMovie.setEditable(true);
centerpanel.add(selectMovie);
selectMovie.setPreferredSize(new Dimension(200,40));
JPanel Southpanel=new JPanel();
showspage.add(Southpanel,BorderLayout.SOUTH);
Southpanel.setPreferredSize(new Dimension(0,200));
JLabel newmovielabel=new JLabel("New Movie");
JLabel newmoviesdurationlabel= new JLabel("Duration");
Southpanel.add(newmovielabel);
JTextField moviebar= new JTextField();
moviebar.setFont(textFieldFont);

```

```

moviebar.setPreferredSize(new Dimension(200,40));
Southpanel.add(moviebar);
Southpanel.add(newmoviesdurationlabel);
JTextField durationbar= new JTextField();
durationbar.setFont(textFieldFont);
durationbar.setPreferredSize(new Dimension(200,40));
Southpanel.add(durationbar);
JButton aadnewmovie=new JButton("Add new Movie");
aadnewmovie.setFocusable(false);
aadnewmovie.setFont(buttonFont);
aadnewmovie.setCursor(curs);
Southpanel.add(aadnewmovie);
aadnewmovie.addActionListener(action5->{
    try{
        Connection connection4 = DriverManager.getConnection(
            "jdbc:oracle:thin:@localhost:1521:xe",
            "C##booknmove",
            "dinesh"
        );
        String newmovie=moviebar.getText();
        int duration=Integer.parseInt(durationbar.getText());
        Statement stmt8= connection4.createStatement();
        ResultSet res9=stmt8.executeQuery("select  max(m_id)  from
movies");
        res9.next();
        int newMovieid=res9.getInt(1)+1;
        Statement stmt10=connection4.createStatement();
        stmt10.executeQuery("insert into movies(m_name,duration,m_id)
values('"+newmovie+"','"+duration+"','"+newMovieid+"')");
        Statement stmt7 =connection4.createStatement();

```

```

        stmt7.executeQuery("update all _details set mid="+newMovieid+"
where show_id="+showid+" and tid="+theaterid+" ");

        stmt7.executeQuery("commit");

        showspage.dispose();

    }

    catch(Exception e){

        System.out.println(e);

    }

});

showspage.setVisible(true);

});

}

String delimiter = ", ";

String                                notin                                =

showsnotin.stream().map(String::valueOf).collect(Collectors.joining(delimiter));

JScrollPane timeScrollPane=new JScrollPane(timePanel);
admintime.add(timeScrollPane,BorderLayout.CENTER);

JPanel southJPanel=new JPanel();
southJPanel.setPreferredSize(new Dimension(100,60));
admintime.add(southJPanel,BorderLayout.SOUTH);

JButton addshowButton=new JButton("AddShows");
addshowButton.setFocusable(false);
addshowButton.setCursor(curs);
addshowButton.addActionListener(action4->{

    JFrame showspage1=new JFrame();

    showspage1.setLocationRelativeTo(null);

    showspage1.setLayout(new BorderLayout());

    showspage1.setSize(new Dimension(320,220));

    JPanel southbar=new JPanel();

    southbar.setPreferredSize(new Dimension(0,50));

    showspage1.add(southbar,BorderLayout.SOUTH);

```



```

JComboBox<String> selectshow=new JComboBox<>();

DefaultComboBoxModel<String>          modell          =new
DefaultComboBoxModel<String>();

selectshow.setModel(modell);

try{

    Connection connection5 = DriverManager.getConnection(
        "jdbc:oracle:thin:@localhost:1521:xe",
        "C##booknmmove",
        "dinesh"
    );

    Statement stmt12=connection5.createStatement();

    System.out.println(notin);

    notin.trim();

    if (!notin.isEmpty()){ResultSet res16=stmt12.executeQuery("select time
from shows where not sid in("+notin+"");

    while(res16.next()){

        Timestamp time=res16.getTimestamp(1);

        modell.addElement(time.toString());

    }}

    else{

        ResultSet res16=stmt12.executeQuery("select time from shows");

        while(res16.next()){

            Timestamp time=res16.getTimestamp(1);

            modell.addElement(time.toString());

        }

    }

}

catch(Exception e4){

    System.out.println(e4);

}

southbar.add(selectshow);

```

```

JButton addshow=new JButton("Add Show");
southbar.add(addshow);
addshow.addActionListener(action5->{
    try{Connection connection5 = DriverManager.getConnection(
        "jdbc:oracle:thin:@localhost:1521:xe",
        "C##booknmove",
        "dinesh"
    );
    Statement stmt12=connection5.createStatement();
    ResultSet result =stmt12.executeQuery("select sid from shows where
time=to_timestamp '"+selectshow.getSelectedItemAt()+"',          'YYYY-MM-DD
HH24:MI:SS.FF')");
    result.next();
    showspage1.dispose();
    addseats(result.getInt(1), theaterid);
    }
    catch(Exception e5){}
});
JPanel centerbar=new JPanel();
showspage1.add(centerbar, BorderLayout.CENTER);
centerbar.add(new JLabel("  "));
JLabel date=new JLabel("Date:");
centerbar.add(date);
JTextField jdata= new JTextField("2024-06-29");
jdata.setPreferredSize(new Dimension(200,40));
centerbar.add(jdata);
JLabel time=new JLabel("  Time:");
centerbar.add(time);
JTextField jtime= new JTextField("10:30");
jtime.setPreferredSize(new Dimension(200,40));
centerbar.add(jtime);

```

```

JButton addnewshow=new JButton("AddNewShow");
addnewshow.setFocusable(false);
addnewshow.setCursor(curs);
addnewshow.setFont(buttonFont);
addnewshow.addActionListener(action5->{
    try{
        String sdate=jdata.getText();
        String stime=jtime.getText();
        Connection connection6 = DriverManager.getConnection(
            "jdbc:oracle:thin:@localhost:1521:xe",
            "C##booknmove",
            "dinesh"
        );
        Statement stmt13=connection6.createStatement();
        ResultSet ress=stmt13.executeQuery("select * from shows where
time=to_timestamp '"+sdate+" '"+stime+"', 'YYYY-MM-DD HH24:MI:SS.FF')");
        if(ress.next()){
            System.out.println("Already Exists");
        }
        else{
            Statement stmt14 =connection6.createStatement();
            ResultSet ress1=stmt14.executeQuery("select max(sid) from
shows");
            ress1.next();
            int sid=ress1.getInt(1)+1;
            String sql="insert into shows
values '"+sid+"',to_timestamp '"+sdate+" '"+stime+"', 'YYYY-MM-DD HH24:MI:SS.FF')";
            stmt14.executeUpdate(sql);
            System.out.println("Added");
            showspage1.dispose();
            addseats(sid, theaterid);
        }
    }
});

```

```

        JOptionPane.showMessageDialog(null, "Show has been added
        Succesfully");
    }
}
catch(Exception exe){
    System.out.println(exe);
}
});
centerbar.add(addnewshow);
showspage1.setVisible(true);
});
southJPanel.add(addshowButton);
JPanel northJPanel=new JPanel();
JLabel click=new JLabel("Click Here TO Modify:");
northJPanel.add(click);
northJPanel.setPreferredSize(new Dimension(100,40));
admintime.add(northJPanel,BorderLayout.NORTH);
admintime.setVisible(true);
}
catch(Exception ex3){
    System.out.println(ex3);
}
});
}
JScrollPane theaters=new JScrollPane(theaterlist);
homepage.add(theaters,BorderLayout.CENTER);
JPanel south=new JPanel();
south.setPreferredSize(new Dimension(0,120));
homepage.add(south,BorderLayout.SOUTH);
JLabel gapy=new JLabel("
");
south.add(gapy);

```

```

JLabel thea=new JLabel("Theater:");
south.add(thea);

JTextField theatername=new JTextField();
theatername.setPreferredSize(new Dimension(200,40));
south.add(theatername);

JLabel gapx=new JLabel("          ");
south.add(gapx);

JLabel loc=new JLabel("Location:");
south.add(loc);

JTextField location=new JTextField();
location.setPreferredSize(new Dimension(200,40));
south.add(location);

JButton theateradd=new JButton("Add theater");
theateradd.setFocusable(false);
theateradd.setCursor(curs);
theateradd.addActionListener(e->{

if(theatername.getText().trim().isEmpty()||location.getText().trim().isEmpty()||location.getTex
t()==null||theatername.getText()==null){

        System.out.println("Enter Values");
    }
    else{
        try{
            Connection connection3 = DriverManager.getConnection(
                "jdbc:oracle:thin:@localhost:1521:xe",
                "C##booknmove",
                "dinesh"
            );
            Statement stmt3=connection3.createStatement();
            ResultSet res4=stmt3.executeQuery("select max(t_id) from theaters");
            res4.next();

```

```

        int newid =res4.getInt(1)+1;

        Statement stmt4=connection3.createStatement();

        stmt4.executeQuery("insert      into      theaters(t_id,t_name,location,adid)
values("+newid+", '"+theatername.getText()+"', '"+location.getText()+"', '"+adminid+"");

        stmt4.executeQuery("commit");

        homepage.dispose();

        adminhome(adminid);
    }

    catch(SQLException ex2){
        System.out.println(ex2);
    } catch (ClassNotFoundException e1) {
        e1.printStackTrace();
    }
});

south.add(theateradd);

JPanel north=new JPanel();
north.setPreferredSize(new Dimension(0,80));
JLabel head=new JLabel("Your Theaters:");
head.setFont(labelFont);
north.add(head);

homepage.add(north, BorderLayout.NORTH);

JPanel east=new JPanel();
east.setPreferredSize(new Dimension(40,0));
homepage.add(east, BorderLayout.EAST);

JPanel west=new JPanel();
west.setPreferredSize(new Dimension(40,0));
homepage.add(west, BorderLayout.WEST);

homepage.setVisible(true);
}

catch(SQLException ex1){
    System.out.println(ex1);
}

```

```

    }
}

public static void addseats(int showid ,int theaterid){
    JFrame showspage=new JFrame();
    showspage.setLocationRelativeTo(null);
    showspage.setLayout(new BorderLayout());
    showspage.setSize(320,370);
    showspage.setResizable(false);
    JPanel top=new JPanel();
    showspage.add(top,BorderLayout.NORTH);
    JLabel change = new JLabel("Change movie: ");
    change.setFont(labelFont);
    top.add(change);
    JPanel centerpanel=new JPanel();
    showspage.add(centerpanel,BorderLayout.CENTER);
    JComboBox<String> selectMovie=new JComboBox<>();
    DefaultComboBoxModel<String> model =new DefaultComboBoxModel<String>();
    selectMovie.setModel(model);
    try {
        Connection connection4 = DriverManager.getConnection(
            "jdbc:oracle:thin:@localhost:1521:xe",
            "C###booknmove",
            "dinesh"
        );
        Statement stmt6=connection4.createStatement();
        ResultSet res8=stmt6.executeQuery("select m_name from movies");
        while(res8.next()){
            model.addElement(res8.getString(1));
        }
        centerpanel.add(selectMovie);
    }
}

```

```

JButton changemovie=new JButton("AddExistingMovie");
changemovie.setFocusable(false);
changemovie.setCursor(curs);
centerpanel.add(changemovie);
changemovie.addActionListener(e->{
    try {
        String selectedMovie=(String)selectMovie.getSelectedItemAt();
        Statement stmt8= connection4.createStatement();
        ResultSet res9=stmt8.executeQuery("select m_id from movies where
m_name='"+selectedMovie+"'");
        res9.next();
        int selectedMovieid=res9.getInt(1);
        Statement stmt7 =connection4.createStatement();
        CallableStatement call=connection4.prepareCall("{call
insert_seats( '"+selectedMovieid+"','"+theaterid+"','"+showid+"')}");
        call.execute();
        stmt7.executeQuery("commit");
        showspage.dispose();
    } catch (SQLException e1) {
        System.out.println(e1+ "exc command");
    }
});
}
catch(Exception e){
    System.out.println(e.getMessage());
}
selectMovie.setEditable(true);
centerpanel.add(selectMovie);
selectMovie.setPreferredSize(new Dimension(200,40));
JPanel Southpanel=new JPanel();
showspage.add(Southpanel,BorderLayout.SOUTH);

```



```
Southpanel.setPreferredSize(new Dimension(0,200));
JLabel newmovielabel=new JLabel("New Movie");
JLabel newmoviesdurationlabel= new JLabel("Duration");
Southpanel.add(newmovielabel);
JTextField moviebar= new JTextField();
moviebar.setFont(textFieldFont);
moviebar.setPreferredSize(new Dimension(200,40));
Southpanel.add(moviebar);
Southpanel.add(newmoviesdurationlabel);
JTextField durationbar= new JTextField();
durationbar.setFont(textFieldFont);
durationbar.setPreferredSize(new Dimension(200,40));
Southpanel.add(durationbar);
JButton aadnewmovie=new JButton("Add new Movie");
aadnewmovie.setFocusable(false);
aadnewmovie.setFont(buttonFont);
aadnewmovie.setCursor(curs);
Southpanel.add(aadnewmovie);
aadnewmovie.addActionListener(action5->{
    try{
        Connection connection4 = DriverManager.getConnection(
            "jdbc:oracle:thin:@localhost:1521:xe",
            "C###booknmove",
            "dinesh"
        );
        String newmovie=moviebar.getText();
        int duration=Integer.parseInt(durationbar.getText());
        Statement stmt8= connection4.createStatement();
        ResultSet res9=stmt8.executeQuery("select max(m_id) from movies");
        res9.next();
```

```

        int newMovieid=res9.getInt(1)+1;

        Statement stmt10=connection4.createStatement();

        stmt10.executeQuery("insert      into      movies(m_name,duration,m_id)
values('"+newmovie+"','"+duration+"','"+newMovieid+"')");

        Statement stmt7 =connection4.createStatement();

        stmt7.executeQuery("EXEC
insert_seats('"+newMovieid+"','"+theaterid+"','"+showid+"')");

        stmt7.executeQuery("commit");

        showspage.dispose();
    }

    catch(Exception e){
        System.out.println(e);
    }

});

showspage.setVisible(true);
}
}

```

End of the source code;