COLLEGE HOSTEL BOOKING SYSTEM

A PROJECT REPORT

Submitted by

PRAVINA M (2303811724322084)

in partial fulfillment of requirements for the award of the course

CGB1201 – JAVA PROGRAMMING

in

ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

K. RAMAKRISHNAN COLLEGE OF TECHNOLOGY

(An Autonomous Institution, affiliated to Anna University Chennai and Approved by AICTE, New Delhi)

SAMAYAPURAM – 621 112

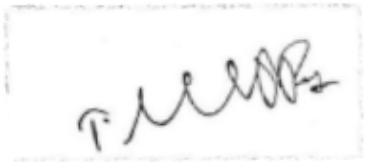
DECEMBER, 2024

K. RAMAKRISHNAN COLLEGE OF TECHNOLOGY (AUTONOMOUS)

SAMAYAPURAM – 621 112

BONAFIDE CERTIFICATE

Certified that this project report on **" COLLEGE HOSTEL BOOKING SYSTEM"** is the bonafide work of **PRAVINA M(2303811724322084)** who carried out the project work during the academic year 2024 - 2025 under my supervision.



Signature

Dr. T. AVUDAIAPPAN M.E.,Ph.D.,

HEAD OF THE DEPARTMENT,

Department of Artificial Intelligence,

K. Ramakrishnan College of Technology,

Samayapuram, Trichy -621 112.

Signature

Mrs.S.GEETHA M.E.,

SUPERVISOR,

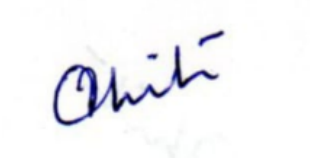
Department of Artificial Intelligence,

K. Ramakrishnan College of Technology,

Samayapuram, Trichy -621 112.

Submitted for the viva-voce examination held on 3.12.24

INTERNAL EXAMINER

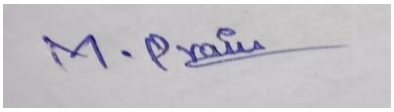


EXTERNAL EXAMINER

DECLARATION

I declare that the project report on "**COLLEGE HOSTEL BOOKING SYSTEM** " is the result of original work done by me and best of my knowledge, similar work has not been submitted to "**ANNA UNIVERSITY CHENNAI**" for the requirement of Degree of **BACHELOR OF TECHNOLOGY**. This project report is submitted on the partial fulfillment of the requirement of the award of the **CGB1201 – JAVA PROGRAMMING.**

Signature



PRAVINA M

Place: Samayapuram

Date: 3/12/2024

ACKNOWLEDGEMENT

It is with great pride that I express our gratitude and indebtedness to our institution, **"K. Ramakrishnan College of Technology (Autonomous)",** for providing us with the opportunity to do this project.

I extend our sincere acknowledgement and appreciation to the esteemed and honourable Chairman, **Dr. K. RAMAKRISHNAN, B.E.,** for having provided the facilities during the course of our study in college.

I would like to express our sincere thanks to our beloved Executive Director, **Dr. S. KUPPUSAMY, MBA, Ph.D.,** for forwarding our project and offering an adequate duration to complete it.

I would like to thank **Dr. N. VASUDEVAN, M.TECH., Ph.D.,** Principal,who gave the opportunity to frame the project to full satisfaction.

I thank **Dr.T.AVUDAIAPPAN, M.E.,Ph.D**., Head of the Department of **ARTIFICIAL INTELLIGENCE AND DATA SCIENCE**, for providing his encouragement in pursuing this project.

I wish to convey our profound and heartfelt gratitude to our esteemed project guide **Mrs.S.GEETHA M.E**., Department of **ARTIFICIAL INTELLIGENCE AND DATA SCIENCE**, for her incalculable suggestions, creativity, assistance and patience, which motivated us to carry out this project.

I render our sincere thanks to the Course Coordinator and other staff members for providing valuable information during the course.

I wish to express our special thanks to the officials and Lab Technicians of our departments who rendered their help during the period of the work progress.

VISION OF THE INSTITUTION

To serve the society by offering top-notch technical education on par with global standards.

MISSION OF THE INSTITUTION

- Be a centre of excellence for technical education in emerging technologies by exceeding the needs of industry and society.
- Be an institute with world class research facilities.
- Be an institute nurturing talent and enhancing competency of students to transform them as all- round personalities respecting moral and ethical values.

VISION AND MISSION OF THE DEPARTMENT

To excel in education, innovation and research in Artificial Intelligence and Data Science to fulfill industrial demands and societal expectations.

Mission 1: To educate future engineers with solid fundamentals, continually improving teaching methods using modern tools.

Mission 2: To collaborate with industry and offer top-notch facilities in a conductive learning environment.

Mission 3: To foster skilled engineers and ethical innovation in AI and Data Science for global recognition and impactful research.

Mission 4: To tackle the societal challenge of producing capable professionals by instilling employability skills and human values.

PROGRAM EDUCATIONAL OBJECTIVES (PEOS)

PEO 1: Compete on a global scale for a professional career in Artificial Intelligence and Data Science.

PEO 2: Provide industry-specific solutions for the society with effective communication and ethics.

PEO 3: Hone their professional skills through research and lifelong learning initiatives.

PROGRAM OUTCOMES

Engineering students will be able to:

1. **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
4. **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5. **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
6. **The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
7. **Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
8. **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

9. **Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
10. **Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
11. **Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
12. **Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

PROGRAM SPECIFIC OUTCOMES (PSOs)

- **PSO 1:** Capable of working on data-related methodologies and providing industry-focussed solutions.
- **PSO2:** Capable of analysing and providing a solution to a given real-world problem by designing an effective program.

ABSTRACT

The **College Hostel Room Booking System** aims to efficiently manage room bookings while prioritizing students based on application date or special needs. The system allows students to view available rooms in real-time, preventing overbooking by ensuring that once a room is booked, it cannot be reserved by another student until it is released. Rooms are prioritized for booking based on the student's application date, with students who have special needs being given higher priority. The system consists of a Room class to manage room status, a Student class to store student details (such as name, application date, and special needs), and a Room Booking Manager class to handle booking and releasing rooms. The user interface is built using Swing, providing buttons for booking and releasing rooms, with real-time updates displayed in a text area, ensuring smooth interaction between students and the booking system.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

" The **College Hostel Room Booking System** is designed to streamline the process of managing room reservations for students while ensuring fair and efficient room allocation. The system enables students to view real-time availability of hostel rooms, submit booking requests, and release rooms when no longer needed. It also prevents overbooking by ensuring that once a room is reserved, it cannot be double-booked until it is released. Furthermore, the system incorporates a prioritization mechanism, where room assignments are based on factors like the application date or special needs of students, ensuring that those with urgent requirements or special conditions are given precedence. This system is built using Java's Swing framework, providing an intuitive interface for students to interact with and efficiently manage their hostel bookings.

1.2 OBJECTIVE

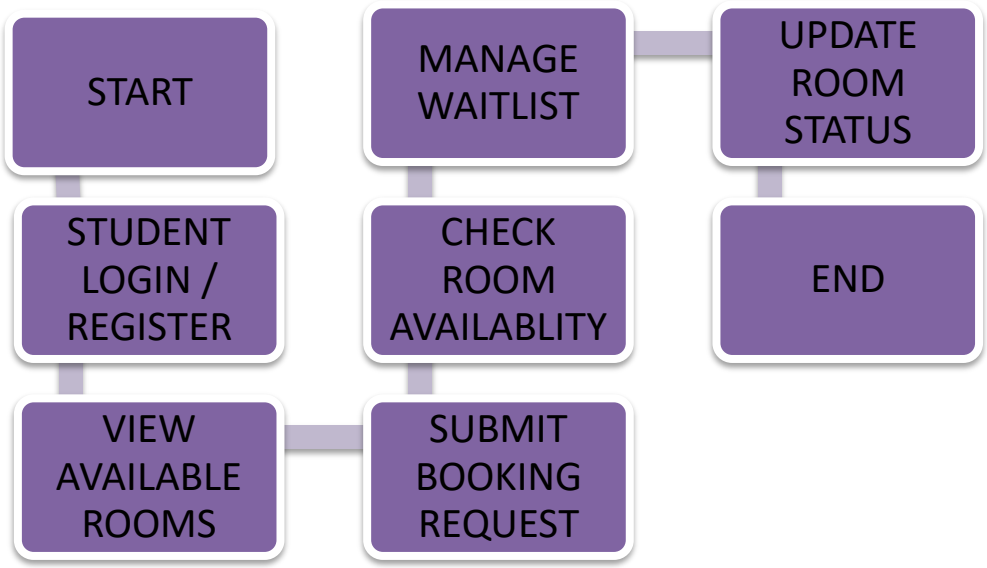
The objective of the **College Hostel Room Booking System** is to provide an efficient, fair, and user-friendly platform for managing hostel room reservations. The system aims to allow students to view real-time availability of rooms, submit booking requests, and release rooms when no longer needed. It ensures that rooms are not overbooked by enforcing availability checks and allows for prioritization of room assignments based on application dates or special needs. The system seeks to automate and simplify the booking process while promoting fairness and accessibility for all students, ensuring an equitable allocation of rooms.

CHAPTER 2
PROJECT METHODOLOGY

2.1 PROPOSED WORK

The proposed work involves developing a **College Hostel Room Booking System** that automates and streamlines the room reservation process for students. The system will feature real-time room availability tracking, prevent overbooking, and enable students to submit booking requests based on priority criteria such as application date or special needs. The work will include designing an intuitive user interface using Java Swing, implementing efficient room management algorithms, and ensuring that the system is capable of handling multiple bookings and releases while maintaining fairness and accessibility for all students.

2.1 **BLOCK DIAGRAM**



CHAPTER 3
JAVA PROGRAMMING CONCEPTS

1. Swing for GUI Development

JFrame, JPanel, JLabel, JTextField, JTextArea, JButton: These components create the user interface, such as the main window, labels for room info, text areas for displaying room status, and buttons for booking or releasing rooms.

JScrollPane: Makes the JTextArea scrollable for viewing long lists of rooms.

Pop-ups (JDialog, JOptionPane): Used to show notifications or error messages, like room booking confirmations or errors.

2. Event Handling

ActionListener Interface: Handles user actions like clicking buttons for booking or releasing rooms.

Anonymous Classes and Lambda Expressions: Simplify event handling code for buttons, making it more concise.

3. Control Structures

If-Else Statements: Handle logic for checking room availability during booking.

Switch Statements: Could manage different actions based on room numbers or booking states.

Loops: Iterate over rooms to display availability or check status.e Statements:

CHAPTER 4
MODULE DESCRIPTION

4.1 ROOM MANAGEMENT MODULE:

Manages room details like room numbers and availability status. It updates the room status to "available" or "booked" as needed. Ensures real-time synchronization of room data across the system**.**

4.2BOOKING REQUEST MODULE:

Handles student room booking requests by capturing details such as name and application date. It checks room availability and processes bookings. Notifies students of successful or failed bookings.

4.3 BOOKING PRIORITIZATION MODULE:

Prioritizes room assignments based on application dates or special needs. Ensures that students with urgent requirements are given preference. Helps in fair allocation of rooms based on predefined criteria..

4.4 BOOKING MANAGEMENT MODULE:

Oversees the booking process by confirming, canceling, or updating bookings. Tracks room availability and status changes in real-time. Ensures consistency of room data during the booking process..

4.5 CONCURRENCY AND SYNCHRONIZATION MODULE:

Handles multiple booking requests to prevent room double-booking. Uses synchronization techniques like locks to ensure data integrity. Ensures the booking process remains accurate in concurrent user scenarios.

4.6 REAL-TIME AVAILABILITY UPDATE MODULE:

Updates room availability in real-time based on bookings and cancellations. Provides students with accurate and up-to-date room status. Ensures smooth and immediate feedback for room availability.

4.7 ROOM STATUS DISPLAY MODULE:

Displays real-time room availability status to students. Updates dynamically as rooms are booked or released. Provides an intuitive and user-friendly interface for viewing and selecting rooms.

CHAPTER 5

CONCLUSION

The College Hostel Room Booking System offers an efficient and organized approach to managing hostel room reservations. Through key modules such as room management, booking requests, prioritization, and real-time availability updates, the system ensures smooth booking processes, fair room allocation, and real-time synchronization. The use of concurrency control and effective user interface design enhances the system's reliability and user experience, making it a practical solution for managing hostel accommodations while preventing overbooking and ensuring fair access for all students.

REFERENCES:

Oracle Corporation (n.d.). "Java Platform, Standard Edition Documentation".
https://docs.oracle.com/en/java/
The official documentation from Oracle on Java, covering libraries like Swing and concurrency mechanisms.

APPENDICES

APPENDIX A – SOURCE CODE

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.util.ArrayList;
import java.util.List;


public class RoomBookingApp {
    // Room class
    static class Room {
        private String roomNumber;
        private boolean available;

        public Room(String roomNumber) {
            this.roomNumber = roomNumber;
            this.available = true;
        }

        public String getRoomNumber() {
            return roomNumber;
        }
```

```
    public boolean isAvailable() {
        return available;
    }

    public void book() {
        this.available = false;
    }

    public void release() {
        this.available = true;
    }
}

// RoomBookingManager class
static class RoomBookingManager {
    private List<Room> rooms;

    public RoomBookingManager(List<Room> rooms) {
        this.rooms = rooms;
    }
```

```
    public List<Room> getRooms() {
        return rooms;
    }

    public boolean bookRoom(String roomNumber) {
        for (Room room : rooms) {
            if (room.getRoomNumber().equals(roomNumber) &&
room.isAvailable()) {
                room.book();
                return true;
            }
        }
        return false; // Room not available or not found
    }

    public void releaseRoom(String roomNumber) {
        for (Room room : rooms) {
            if (room.getRoomNumber().equals(roomNumber) &&
!room.isAvailable()) {
                room.release();
                return;
            }
        }
```

```
    }
}

private JFrame frame;
private RoomBookingManager bookingManager;
private JTextArea roomDisplayArea;

public RoomBookingApp() {
    // Initialize rooms
    List<Room> rooms = new ArrayList<>();
    rooms.add(new Room("101"));
    rooms.add(new Room("102"));
    rooms.add(new Room("103"));
    rooms.add(new Room("104"));

    // Initialize booking manager
    bookingManager = new RoomBookingManager(rooms);

    // Initialize UI
    frame = new JFrame("Room Booking System");
    frame.setSize(400, 300);
    frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
```

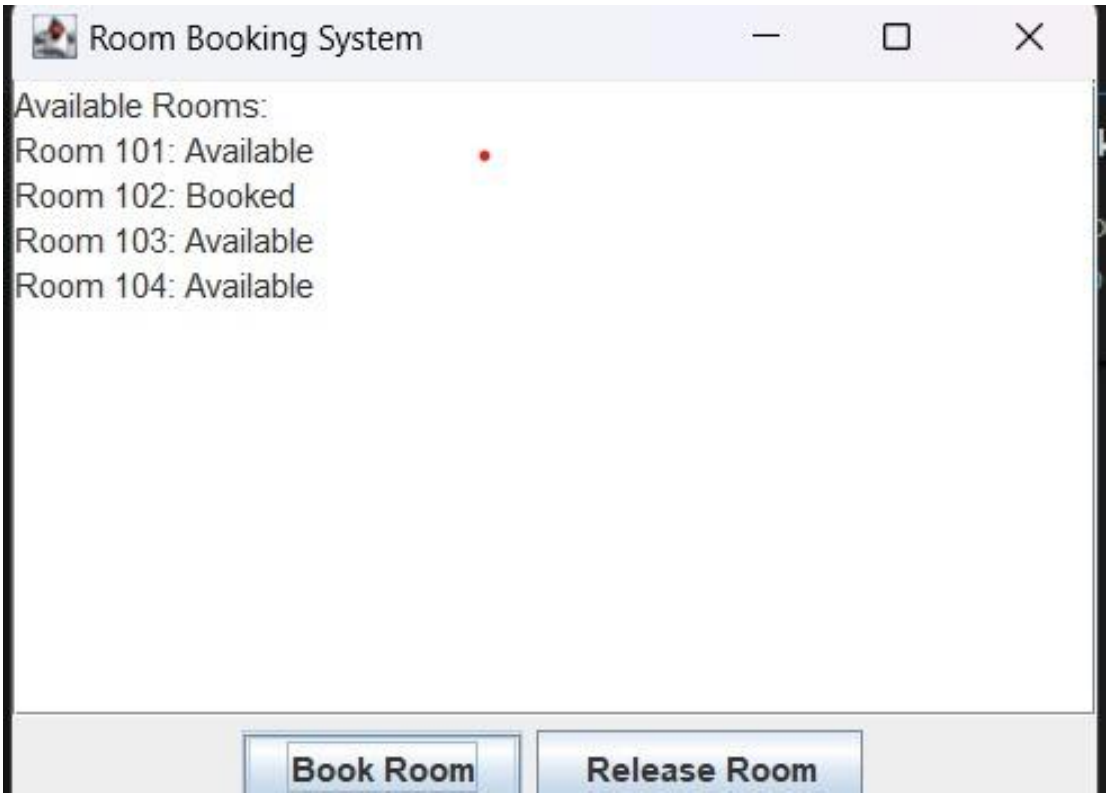
```
        roomDisplayArea = new JTextArea();
        roomDisplayArea.setEditable(false);
        updateRoomDisplay();

        // Book Room Button
        JButton bookRoomButton = new JButton("Book Room");
        bookRoomButton.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e) {
                String roomNumber =
JOptionPane.showInputDialog(frame, "Enter room number to
book:");
                if (roomNumber != null &&
!roomNumber.trim().isEmpty()) {
                    boolean success =
bookingManager.bookRoom(roomNumber);
                    if (success) {
                        JOptionPane.showMessageDialog(frame, "Room " +
roomNumber + " successfully booked.");
                    } else {
                        JOptionPane.showMessageDialog(frame, "Room " +
roomNumber + " is not available.");
                    }
```

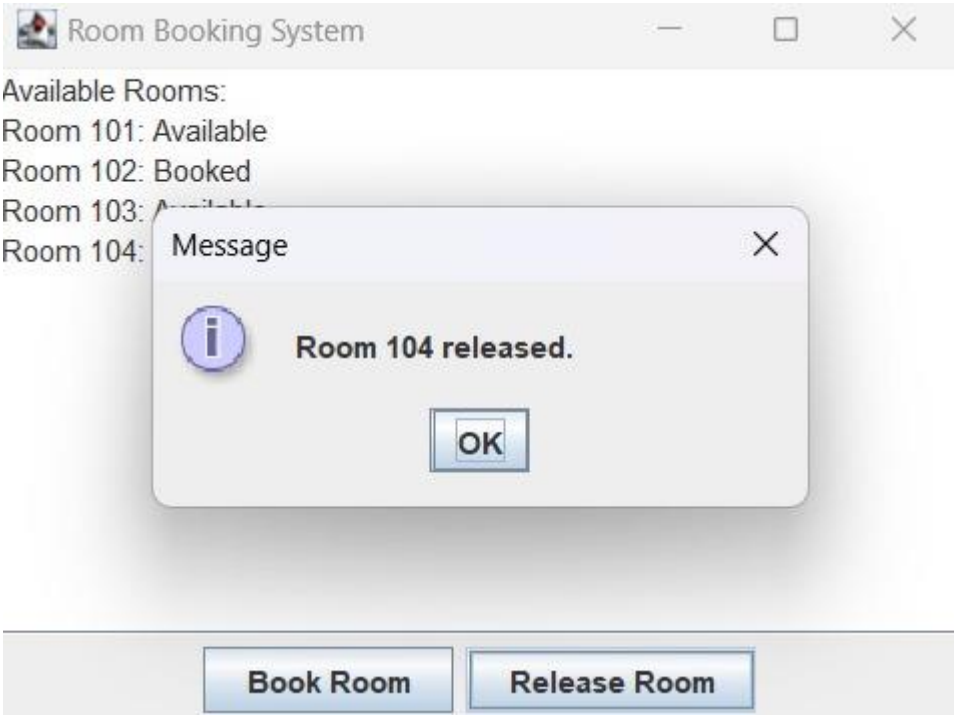
```
                updateRoomDisplay();  // Update room display after
booking
            }
        }
    });

    // Release Room Button
    JButton releaseRoomButton = new JButton("Release Room");
    releaseRoomButton.addActionListener(new ActionListener() {
        @Override
        public void actionPerformed(ActionEvent e) {
            String roomNumber =
JOptionPane.showInputDialog(frame, "Enter room number to
release:");
            if (roomNumber != null &&
!roomNumber.trim().isEmpty()) {
                bookingManager.releaseRoom(roomNumber);
                JOptionPane.showMessageDialog(frame, "Room " +
roomNumber + " released.");
                updateRoomDisplay();  // Update room display after
releasing
            }
        }
    });
```

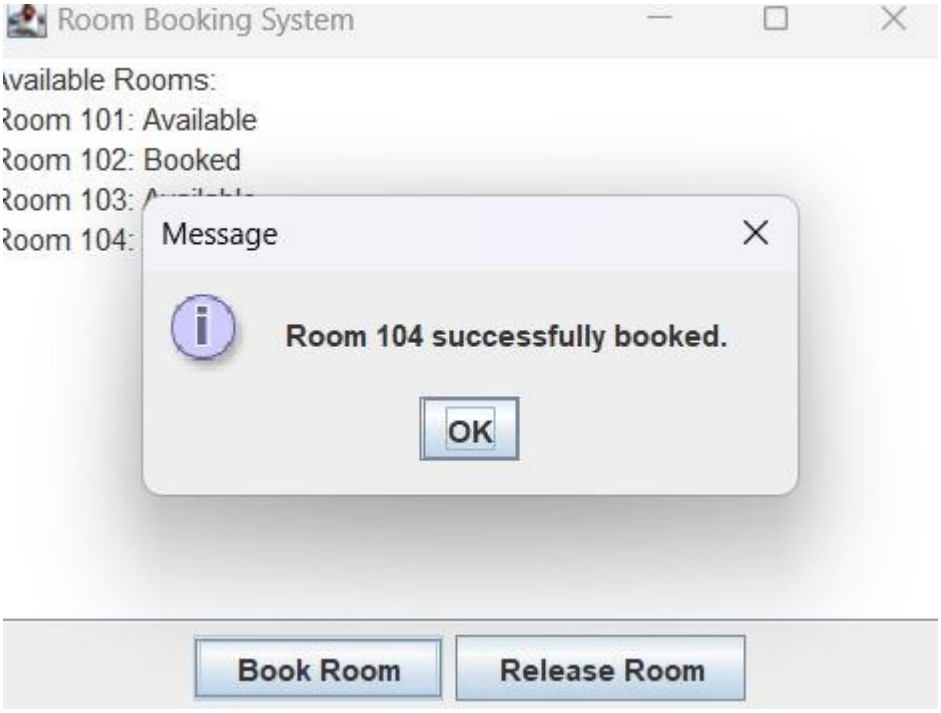
```
        // Panel for buttons
        JPanel buttonPanel = new JPanel();
        buttonPanel.add(bookRoomButton);
        buttonPanel.add(releaseRoomButton);

        // Layout setup
        frame.setLayout(new BorderLayout());
        frame.add(new JScrollPane(roomDisplayArea),
BorderLayout.CENTER);
        frame.add(buttonPanel, BorderLayout.SOUTH);

        frame.setVisible(true);
    }

    // Method to update room display area
    private void updateRoomDisplay() {
        StringBuilder displayText = new StringBuilder("Available
Rooms:\n");
        for (Room room : bookingManager.getRooms()) {
            displayText.append("Room " + room.getRoomNumber() +
": " + (room.isAvailable() ? "Available" : "Booked") + "\n");
        }
```

```
        roomDisplayArea.setText(displayText.toString());
    }

    public static void main(String[] args) {
        SwingUtilities.invokeLater(new Runnable() {
            @Override
            public void run() {
                new RoomBookingApp();
            }
        });
    }
}
```

APPENDIX B – SCREENSHOTS






Room Booking System
vailable Rooms:
Room 101: Available
Room 102: Booked
Room 103:
Room 104:
Message
Room 104 successfully booked.
OK
Book Room
Release Room