MANIPAL INSTITUTE OF TECHNOLOGY

MANIPAL

(A constituent unit of MAHE, Manipal)

OBJECT ORIENTED PROGRAMMING (CSE -2143) MINI PROJECT REPORT ON

HOTEL MANAGEMENT SYSTEM

SUBMITTED TO

Department of Computer Science & Engineering

by

Keshav Agrawal – 220962436 – 73

Ishan – 220962270– 41

Ankur Monga – 220962137– 81

Kirtan Desai – 220962428 – 72

Name & Signature of Evaluator 1

Name & Signature of Evaluator 2

(July 2023 – Nov 2023)

TABLE OF CONTENT

INTRODUCTION

The smooth and effective administration of a hotel's operations is critical to its success in the fast-paced hospitality sector. An innovative approach to address the increasing demands of hotel personnel and customers are a reliable Hotel Management Application. With the help of this software
platform, hoteliers will be able to provide satisfactory guest experiences and optimize them operations.
A complete and integrated system, the Hotel Management Application covers every aspect of hotel management, including guest services, bookings, reservations, and billing. It acts as the hotel's digital nerve centre, promoting a personalised and responsive environment that meets the many demands and preferences of visitors while making daily duties for hotel employees easier. Hoteliers can track and optimise income streams, manage their room inventory with ease, increase the effectiveness of check-in and check-out procedures, and offer a variety of services that improve the entire guest experience using this application. This programme is designed to satisfy the particular needs of properties of all sizes, whether you operate a large resort or a tiny boutique hotel.
The Hotel Management Application makes sure that managers, employees, and hotel owners have access to real-time data, analytics, and guest feedback by utilising technology. This promotes data-driven decision-making and ongoing improvement. Additionally, it backs eco-friendly policies and sustainability efforts, which is in line with the rising demand for ecologically conscious hotel services.

BACKGROUND THEORY

1.1 The Need

- **Efficiency:** Hotel management software automates various tasks, such as reservations, check-ins, check-outs, billing, and inventory management, making operations more efficient and reducing the risk of human errors.
- **Guest Experience:** It enhances the guest experience by offering online reservations, personalized services, and faster check-in/check-out processes.
- **Inventory and Room Management:** Hotel software helps optimize room availability and pricing, ensuring that rooms are filled and priced appropriately. This maximizes revenue.

1.2 HOW HOTEL MANAGEMENT SOLVES THIS

- **Object-Oriented Programming (OOP):**Object-oriented programming is essential for designing the structure of your hotel management system.

Organize your code into classes and objects to model real-world entities in the hotel industry, such as rooms, reservations, guests, and staff members. Use OOP concepts like inheritance, encapsulation, and polymorphism to create a maintainable and modular codebase.

- **Java and JavaFX:**
  Java is a versatile and widely used programming language for desktop application development. JavaFX is a framework for building rich and interactive GUI applications.
  JavaFX provides tools for creating a user-friendly and aesthetically pleasing UI. It includes FXML for defining UI layouts, controllers to manage user interactions, and CSS for styling.

- **Event Handling:**
  JavaFX relies heavily on event-driven programming. Implement event handling to respond to user interactions, such as button clicks, mouse events, and text input.
  Create event listeners and handlers to execute appropriate actions when users interact with the UI.

METHODOLOGY

3.1 Functions Used

Many Functions have been used within the Code for the application. Most have not been set as methods but rather under button handlings codes. This was done to simplify the method definitions which would other be rather confusing considering the number of labels and other GUI materials required for the proper implementation. Following are some functions that have been used.

1) Function to add and Manage Room Details

This function is to initially add the all the details of all the rooms like Room No, Room Type, Special Facilities etc. This has been included within the start method under button action events.

2) Function to update Occupancy and Vacancy of Room under Room Class

This function sets the occupancy and vacancy of the room and updates the details of the customer who has booked the room. This is invoked when a new Booking is set to update the Main Hotel Occupancy along with updating Room Details.

3) Function to show Room Service Requests

This function shows all the current pending service requests for various rooms of the Hotel. We have use Array List along with a List Box to store Requests and Bookings.

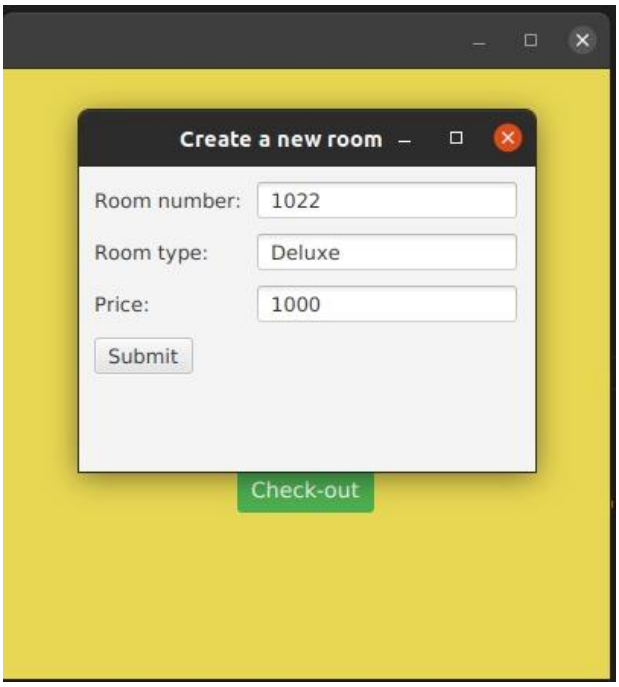
4) Functions to show Occupied and Unoccupied Rooms

This function shows the total number of occupied and unoccupied rooms in the hotel at a given time.

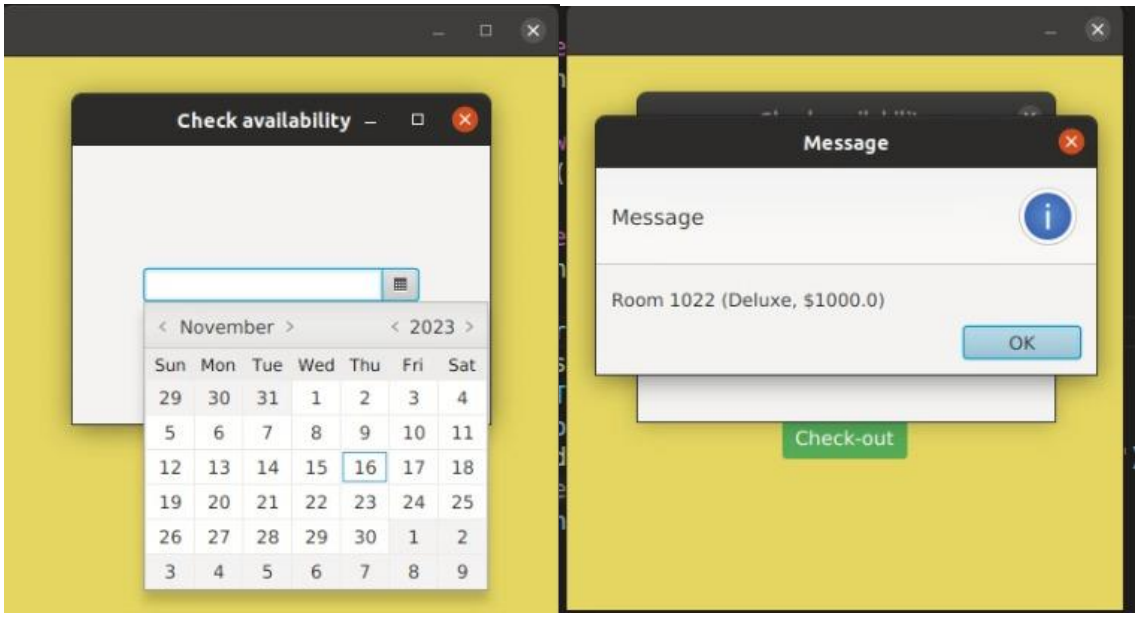
These are a few of the main important functions that have been used in our application. Other than this, many more functions have been used, but they aren't necessarily methods and most of the functions have been set with in Event Handling Scopes of Buttons.

RESULTS AND DISCUSSIONS

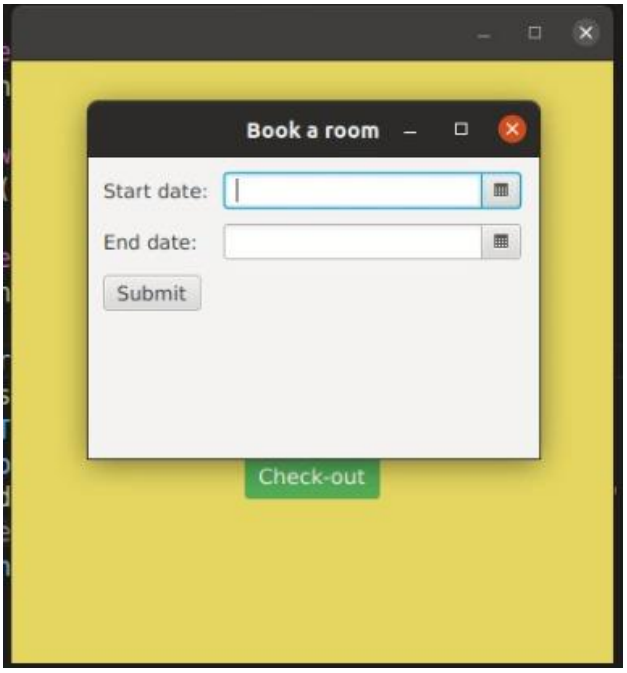
The application starts with the home page with the name of the hotel and buttons linking to other pages.

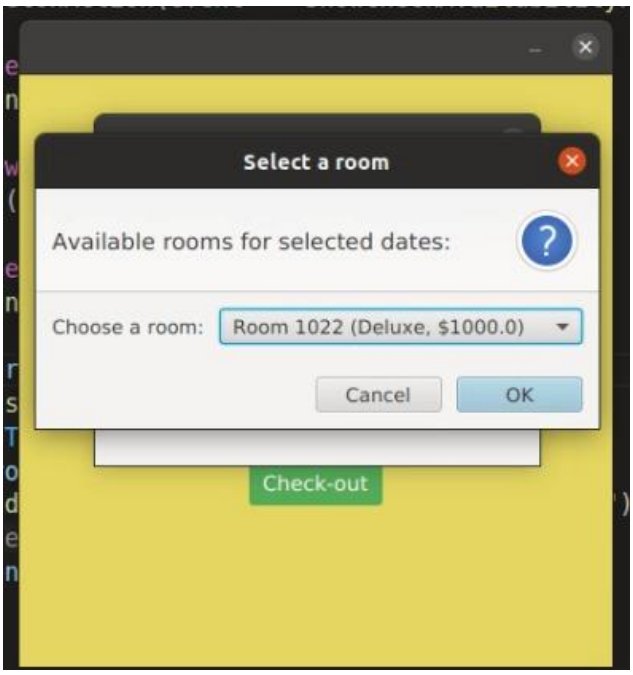


The user can create a new room using the first button and set the room number, type and price for the room.

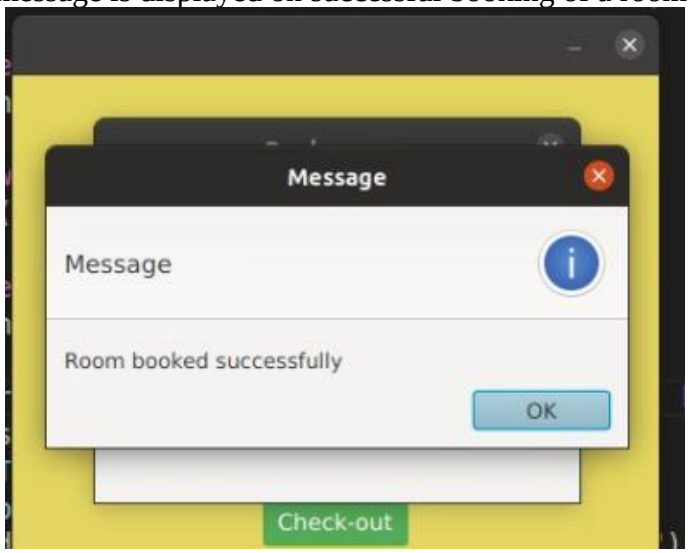


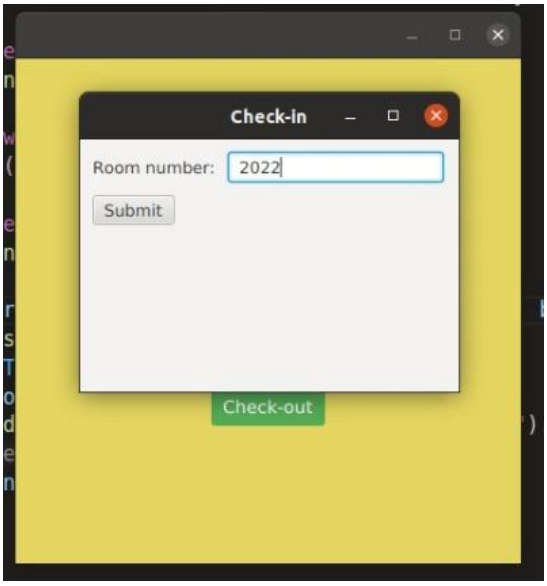
The user can check if a room is available on a certain date from the calendar and get the details of the available room through a popup message. An error message will be displayed if the room is unavailable on the given date.

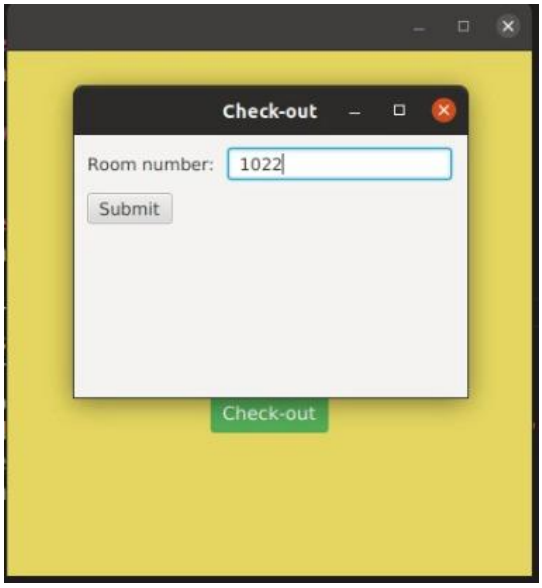


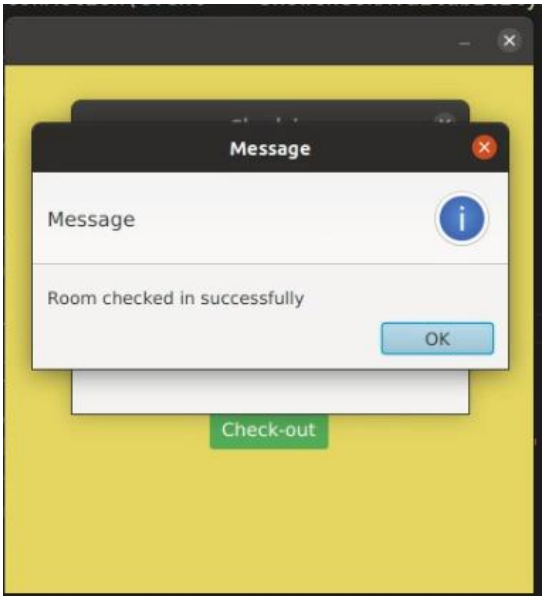


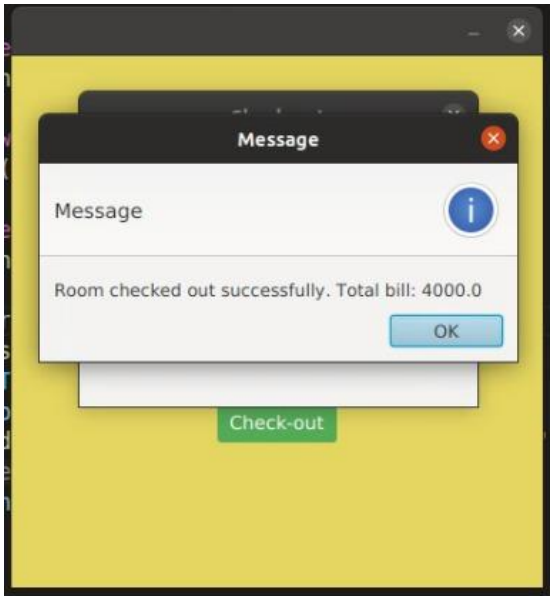
The user can also book a room by specifying the duration of their stay using a calendar and select a room according to their preferences and availability given in a drop down menu. A confirmation message is displayed on successful booking of a room.



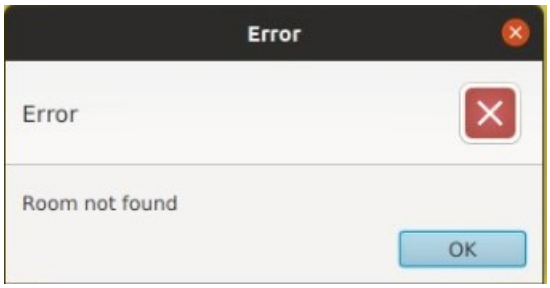








The check-in and check-out options ask for the room number to check-in or check-out and a confirmation message is displayed on the screen. In case a wrong room number is entered an error message will be displayed.



FUTURE ENHANCEMENTS

Given the present knowledge and time constraints, only certain features have been implemented. Further development can be done on the project to increase user satisfaction, usability and wider application.

Some of the features include:

1. User Authentication and Authorization: Implement user accounts with different roles and permissions. This allows you to control who can access, modify, and delete data within the system. It's crucial for ensuring data security and privacy.

2. Real-Time Payment System: Integrate a payment gateways to implement a real time transaction system to make the process seamless.

3. Database Integration: Integrating a robust database will enable the system to store and manage guest information, reservations, and transactions efficiently, ensuring data accuracy and accessibility.

4. Mobile Accessibility: Develop a mobile application or make your JavaFX application responsive to different screen sizes. This ensures that users can access and manage assets on their smartphones and tablets, enhancing convenience and flexibility.

5. Alerts and Notifications: Implement a notification system that alerts users about important events, such as low inventory levels, upcoming maintenance tasks, or asset check-out due dates. Notifications can be sent via email, SMS, or in-app alerts.

6. User-Friendly UI/UX: Continuously improve the user interface and user experience based on feedback and emerging design trends. A user-friendly interface enhances user satisfaction and adoption.

REFERENCES

[1]. https://openjfx.io/openjfx-docs/

[2]. Java The Complete Reference 11th edition, Herbert Schildt, McGraw Hill Publication

[3]. Ford, G.L. et al. (2022). Tactical Asset Management. In: Ancell, G., et al. Power System Assets. CIGRE Green Books. Springer