1. **Introduction:**

**1.1: Purpose:**

The goal of this document is to explicitly describe the functional and non-functional requirements of the Hotel Management System being designed as part of the ITSE course project. This Software Requirements Specification (SRS) is a formal contract between the development team, the instructor (as the client), and any future stakeholders who might review or extend the system.

This SRS assists in ensuring that the same requirements and functionalities are known to all the team members. It gives a foundation to the developers from where they can begin their work with clarity, ensures that the testers are aware of what should be tested, and keeps the project on track with its goals. The document will also prove useful in the future if any upgrades or modifications are needed because it has all the basic details.

This system is meant to automate and simplify major hotel functions It allows them to check room availability, book rooms, order food, request room service, and check out all in one place. For the admin or hotel staff, it provides tools to manage rooms, handle food menus, and view customer requests efficiently. The idea is to cut down on manual work, reduce errors, and create a smoother, more convenient experience for everyone involved in running or staying at the hotel. The document contains extensive use cases, diagrams, assumptions, and user interface descriptions. Through the creation of this SRS, we seek to avoid misunderstandings, handle expectations, and provide a seamless development process from beginning to end.

As our system is for human resources, the SRS ensures we keep our priorities in line with their speed, reliability, simplicity, and structured recordkeeping. Any feature we design will be directly based on what's outlined here, so this document is essentially the basement of the entire project.

* 1. **: Scope:**

Hotel Management System (HMS), created for the purpose of this project is a desktop application designed to assist hotel staff in performing daily activities in a systematic and effective manner. The overall aim is to transition regular hotel operations from paper-based or manual systems to a computerized system that is more efficient, reliable, and manageable.

This is a single-branch-only system intended to be used in a single hotel branch and is not targeted for chain hotels or multiple locations. The design is aimed at producing a simplified but working simulation of an actual hotel front desk system. It is being designed as a course project under time and academic limitations, so the scope is narrowly defined to encompass all significant operations without exceeding what can be reasonably developed and tested within the semester.

The primary features included are:

* + 1. **Customer Management:**

Staff can check in or out and edit or insert new customers. All relevant information, including ID, contact information, and room assignment, will be stored in the system.

* + 1. **Room Booking and Availability:**

Users will be able to display real-time room status (available, booked, under maintenance), assign rooms to customers, and cancel bookings if needed.

**1.2.3 Staff Records Management:**

Administers also have access to a part where they can add new employees, set roles (e.g., receptionist, housekeeping), and view active workers.

* + 1. **Billing and Payments:**

Whenever a client checks out, the system will automatically compute their total billings based on their length of stay and services consumed. Payment status (e.g., Paid, Pending) will be manually input by employees.

* + 1. **Login System:**

Admin and Receptionist will have independent login access. Admin-level users can add rooms, maintain staff, and modify some settings.

* + 1. **Report Generation:**

The system will automatically prepare daily reports comprising number of customers served, rooms booked, and earnings. Admins can see the reports for improved decision-making.

* + 1. **View Menu and Food Ordering:**

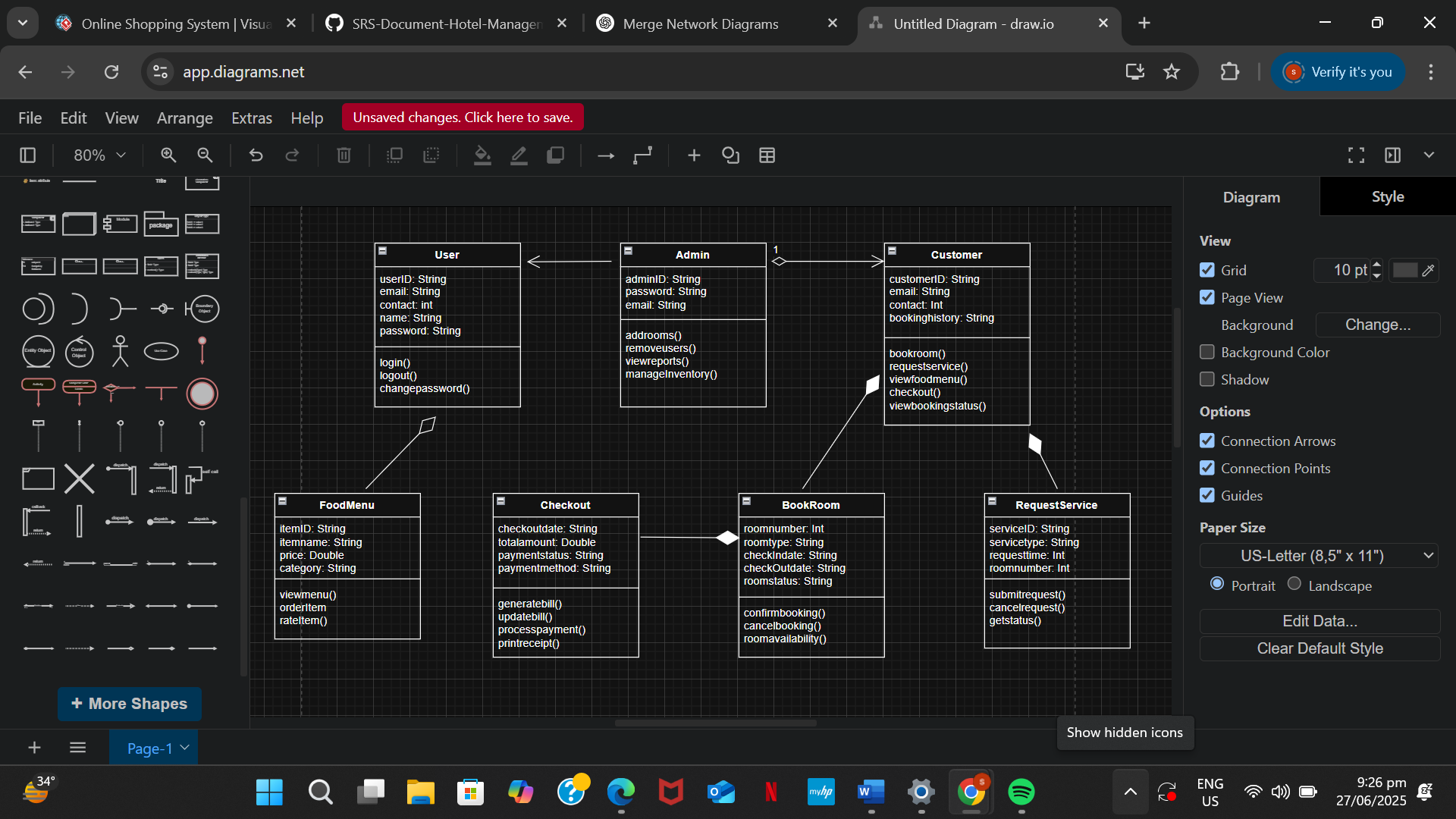
Guests can browse the hotel food menu via the system interface. Receptionists can place orders for guests, and the charges will be billed to their room for payment at check-out time.

* + 1. **Service Requests:**

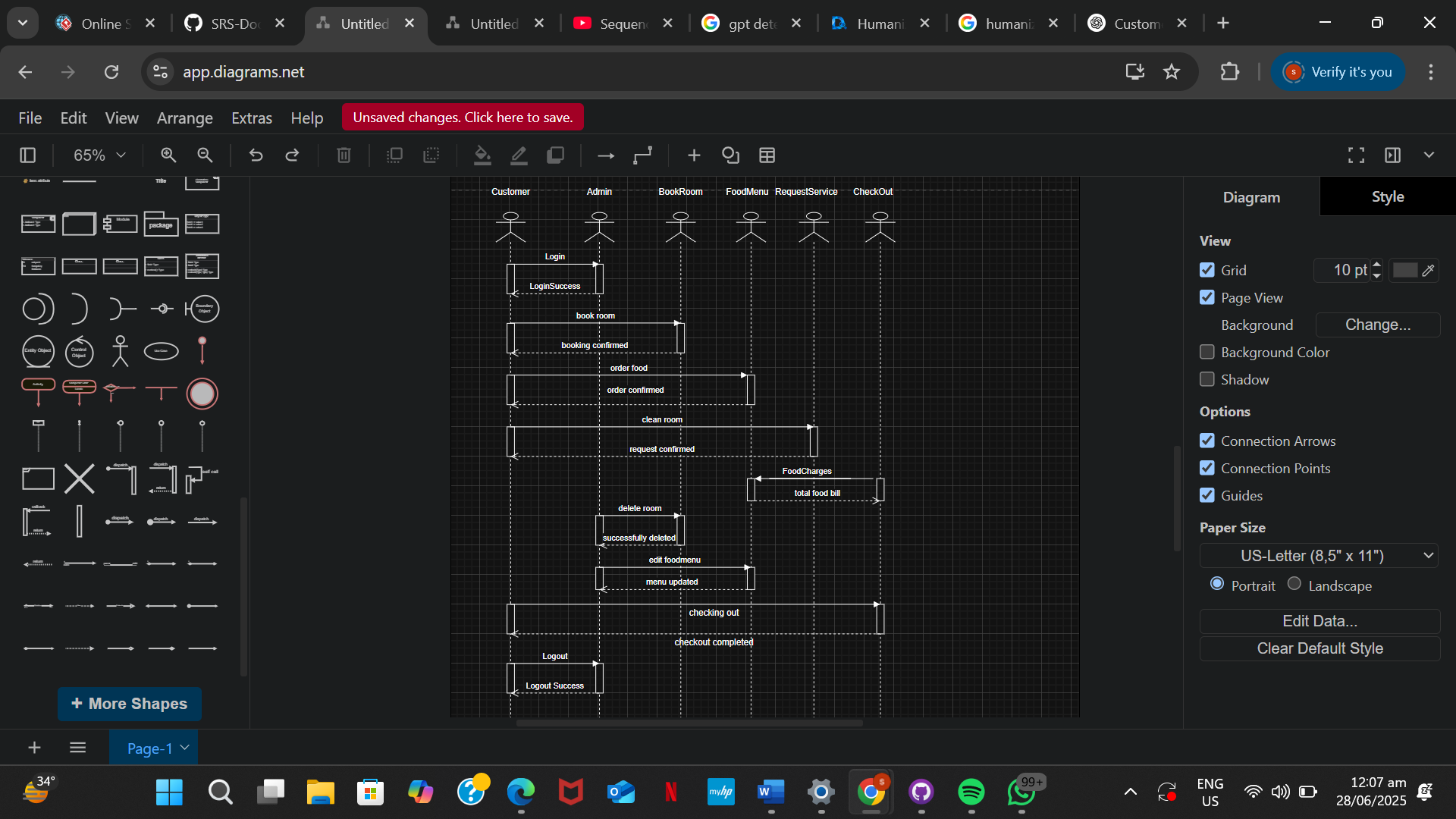
Guests may request extra services like laundry, housekeeping, or maintenance. These requests will be entered against their room record and forwarded to the staff concerned for follow-up.

The aim is to design a system that is practical and realistic, with emphasis on the primary processes that take place at a hotel's reception and admin desk. The features have been selected according to feasibility, complexity, and relevance from an academic perspective, in consultation with the team and instructor.

**1.3: Class Diagram:**

****

**1.4: Sequence Diagram:**

****

**1.5: Definitions, Acronyms, and Abbreviations:**

**1.5.1 SRS (Software Requirements Specification):**

This is a detailed document in which client demands and system functionality is clearly defined. Its purpose is to provide clear guidance for software development.

**1.5.2 Admin:**

It’s a user of the system who has full access to everything. Admin can edit/delete/add rooms, it manages customers data and control system settings.

**1.5.3 Customer:**

An individual who occupies hotel services. Customers check-in check-out and billing records are maintained in system.

**1.5.4 Check-in:**

It’s a process when customer start staying in a hotel. In this room is assigned and customer details are entered in the system.

**1.5.5 Check-out:**

A process when customer leaves the hotel, at this stage bill is generated and the room is again shows in Available Status.

**1.5.6 UI:**

It’s apart of the system that shows up infront of the user and they interact with it. UI includes menus, buttons, forms and screen through which user can use different features.

**1.6: References:**

**1.6.1 GeeksforGeeks - Software Requirements Specification (SRS):**

This website provides format, structures and examples of Software Requirements Specification (SRS) on a basic and advanced level. A conceptual guidance was taken from here in creating this document.

**1.6.2 SRS Sample Document:**

Example SRS document shared by the instructor was used to create the base of the structure of this report. Formatting, section headings, aur content flow is followed by the help of this.

**1.6.3 TutorialsPoint – Software Requirements Specification Guide:**

From this online source additional clarity and explanation was taken, especially about functional vs non-functional requirements and use-cases.

**1.7: Overview:**

This Software Requirements Specification (SRS) document is detailed guide for developing the Hotel Management System (HMS). It is meant to present a clear and structured description and understanding of both functional and non-functional requirements of the software. This process helps for smooth communication among clients, developers, testers, and project stakeholders. The main goal of this SRS document is to minimize confusion, ensure that user expectations and needs align with system behavior, and highlight the development process. The proposed system aims to control the follow of daily hotel operations, which include room booking, customer check-in and check-out, food menu, ordering food, room request service and payment handling, and administrative tasks. The SRS details all expected user interactions, data flows, system limitations, and performance standards that the final product must meet. This document is divided into several organized sections, each section is focusing on a specific task or aspect of the system. The Introduction section outlines the project goals, intended audience, and the system's scope. It sets clear expectations for what the system should achieve. The Overall Description provides an overview of the system environment, user roles such as Admin and Customer

* Introduction section gives the clear starting point of the document. It briefly describes that what Hotel Management System actually is, who will use it and what topics this document will cover. The purpose of this section is to make the reader understand the background of system and base of next upcoming sections is strongly built.
* Purpose section explains the real reason of making the system. Goals are highlighted in this section for example making hotel operations efficient, managing room bookings, and simplifying check-in and check-out process. This part relates real life problems with system
* Scope section describes that what functions system will perform and what not, It includes key features like room management, checkout and roles of user. This section gives a clear direction to the development team so that they remain focused on goals and doesn’t get distracted.
* Definitions, Acronyms, and Abbreviations section gives the short forms of all the technical terms which are used in the document. Its purpose its to understand the content without any confusion even if the person is technical or not.
* References section explain all the sources from which help is taken in the making of

Software Requirements Specification (SRS) document, In this online resources like GeeksforGeeks, SRS sample and other learning websites are included which were very useful in preparing the content.

In Conclusion, This SRS Document provide strong foundation for the successful development of Hotel Management System. In this system’s purpose, scope, technical terms and references are clearly defined so that from starting every team member have a shared understanding about the project. This initial clarity avoids misunderstanding and confusion between different development stages like design, coding and testing.

A well structured SRS document not just save time and hard work but also improves the quality of software and future maintainability.

**External Interface Requirements:**

**User Interfaces (Screens, Menus, Forms):**

Interface of this Hotel management system is based on simple and and user friendly environment. For two roles different interfaces are made: customer and admin. Both interfaces are console based but are easy to understand.

**Customer interface:**

**Login Screen:**

Customer enters email and password to login into the system if credentials are not matched error message is shown.

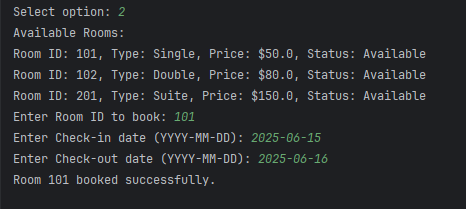
**Customer Menu:**

After login a menu appears for the user where customer enter option of own desire.

1. Room availability
2. Book room
3. Order food
4. Request room service
5. Checking out

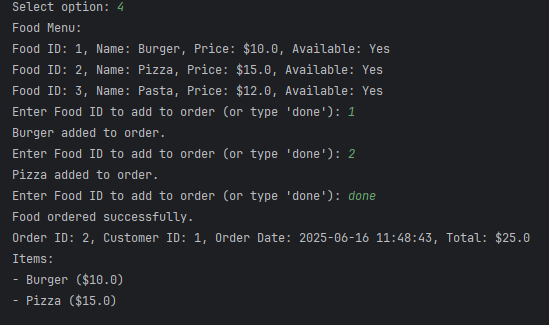
**Booking Form:**

When customer books room a form type menu shows up where customer have to enter room type, check in and check out date.



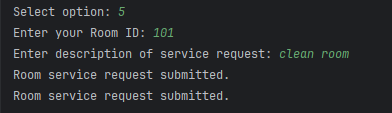
**Food Menu:**

When customer orders food, system shows available food items and prices, and customer places its order.



**Room Service Form:**

For room service request simple input is taken from the customer like “cleaning required” or “need extra towels”.



**Confirmation Screens:**

After every action a message is shown on the screen that the work has been done successfully.

**Admin interface:**

**Login Screen:**

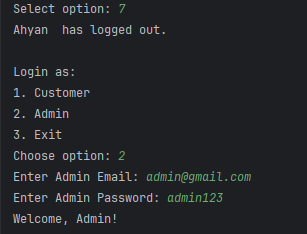
Admin also enters email and password to login and if wrong credentials are entered system denies the access.

**Admin Menu:**

After login following options are shown to admin:

1. Add rooms
2. Delete/Update existing rooms
3. Add food items
4. Delete/Update food items
5. View room bookings
6. View room service record

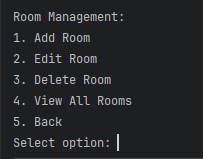
Login as Admin:



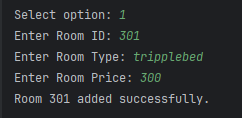
Menu:



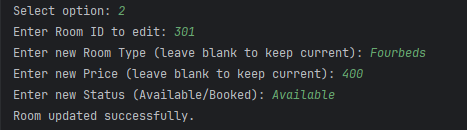
Manage room:



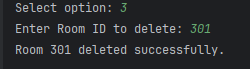
Add Room:



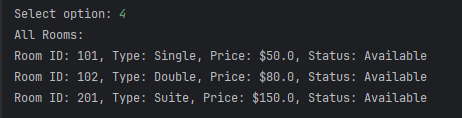
Edit Room:



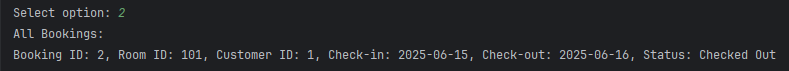
Delete Room:



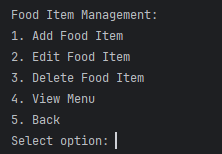
View Rooms:



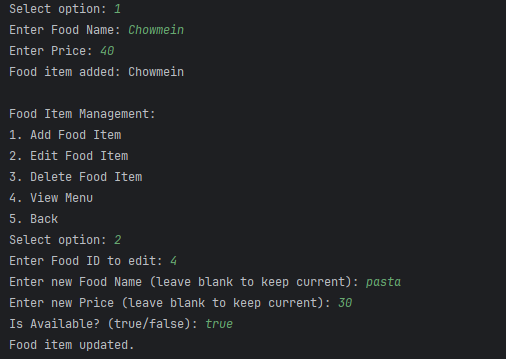
Manage Bookings:



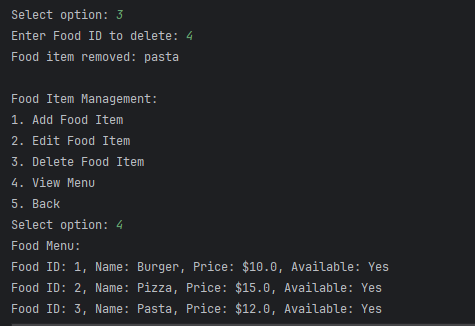
Manage FoodItems:



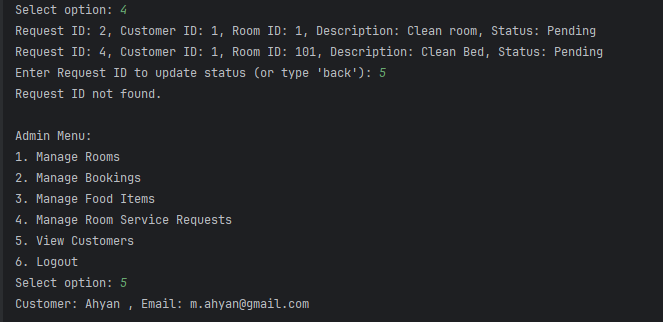
Add And Edit :



Delete and View Fooditems:



Manage room service and View customers:



**Hardware Interfaces:**

Specialized hardware is not needed to run this system. Only basic hardware components are used in this system.

**Computer/Laptop:**

System is made on java so it can be run on any Windows, Mac or Linux PC.

**Keyboard and mouse:**

For user input and navigation keyboard and mouse is used.

**Display Screen:**

System output(menus, forms, confirmations) is shown on screen.

This system does not directly interact with physical servers, biometric devices or hardware sensors.

**Software Interfaces:**

Hotel management system interact with some software components.This interaction is done on basic level.

**Java runtime environment(JRE):**

To run the system JRE must be install in pc or laptop.

**Java development:**

During development and testing JDK is used.

**Operating system:**

System can be run on any modern OS like Windows, Linux, MacOS.

**Optional:File I/O Libraries:**

If you want to store data in future java file handling can be used to store data in local files.

Present system does not use any external software or third party APIs, but in future database can be connected to it like (MySQL, SQLite).

**Communication Interfaces:**

Current version of the system is standalone and is not linked with any network or web-based communication features, may be added in future.

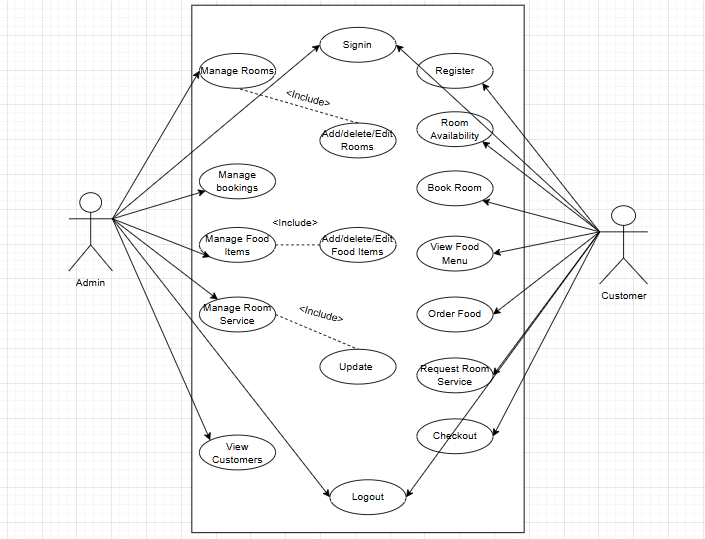
**Email Notification (Future Plan):When customer book any room or checksout system can send email automatically.(By using SMTP protocol)**

**Web-Based Version (Future Plan):If this project is made web based then HTTP/HTTPS protocols will be used.**

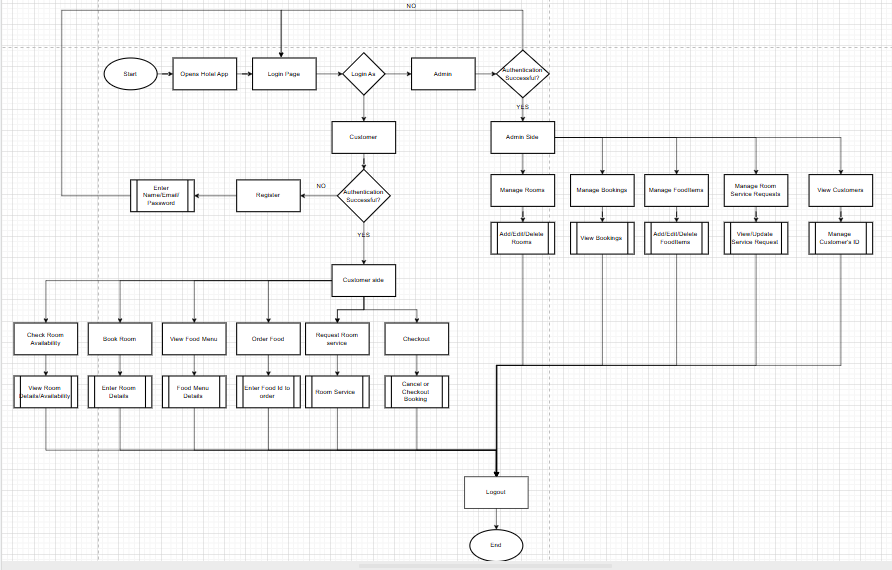
**Multi-user Networking:TCP/IP protocols can be used to connect multiple users in a networked environment.**

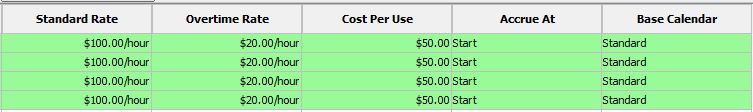
**Currently communication is internal, means its between user and system through input output commands. External API or web services are not used in this version.**

**Use Case Diagram:**



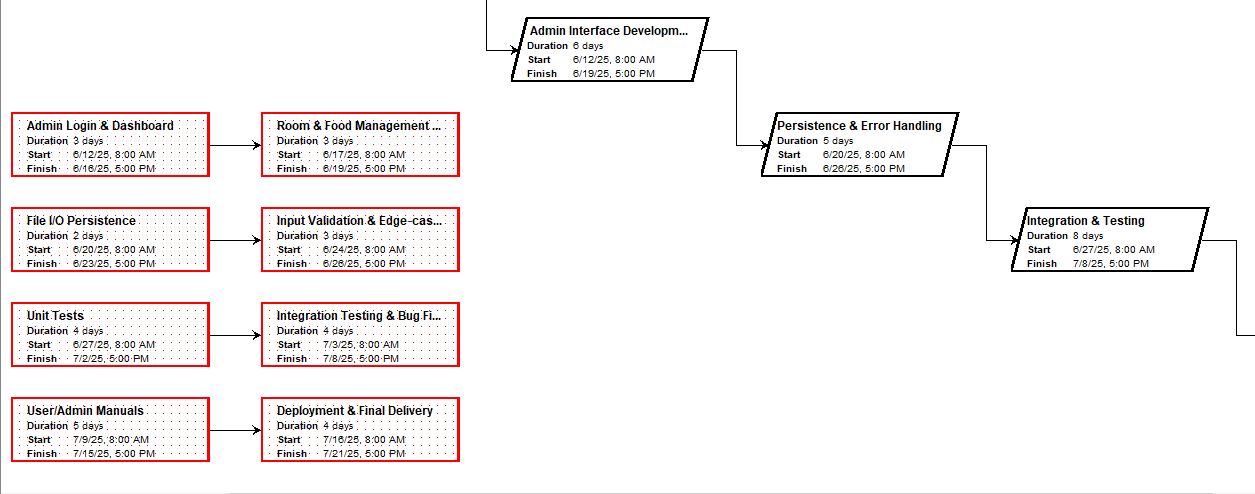
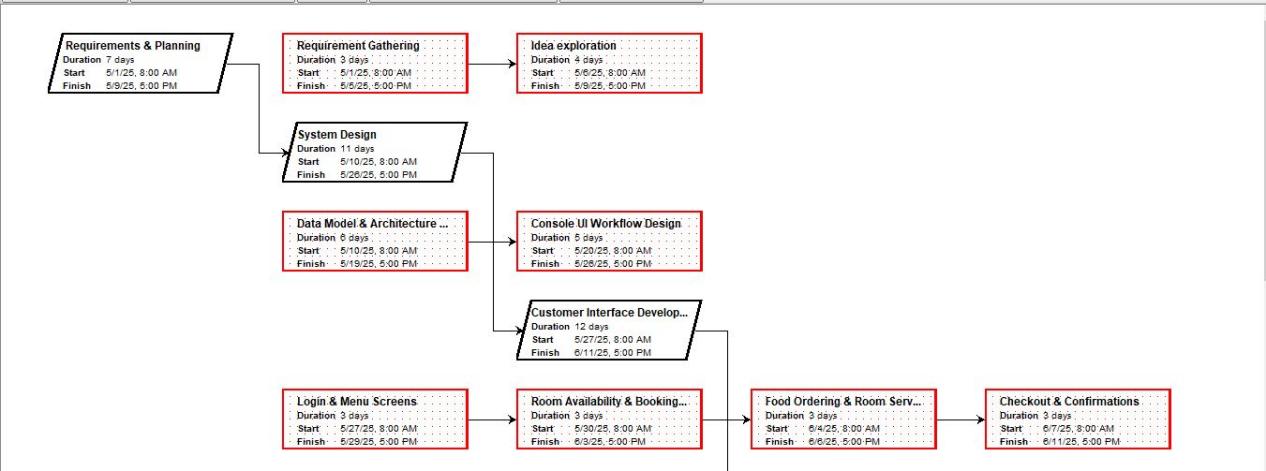
**System Flow Diagram:**



**Resource Sheet:** 



**Pert Chart:**



**Gantt Chart:**

