Hotel Management System – Specification Document

Authors: ESSETTI Yasmine, MASMOUDI Nour
Date: 2024/2025

1. Introduction

1.1 Project Overview

The Hotel Management System (HMS) is a Java-based application designed to streamline hotel operations, including room reservations, user management, and administrative tasks. It features:

Two user roles: Administrators (manage rooms/users/bookings) and Guests (book rooms, view reservations).
In-memory data storage: Uses Java Arraylists (no database)
GUI: Built with NetBeans (Swing) for usability.

1.2 Motivations

- Address inefficiencies in manual hotel management.
- Provide a scalable, user-friendly solution for small to mid-sized hotels.
- Academic exercise in object-oriented design and Java development.

2. Specification

2.1 Key Concepts & Constraints

- **Actors**:
  - **Admin**: Manages rooms, bookings, and user accounts.
  - **Guest**: Books rooms, cancels reservations, updates profile.
- **Communication model (client-server)**:
  - **Client:** The user interacts with the system through the GUI, sending requests such as booking a room or logging in.
  - **Server:** The server processes the requests via the Business Logic Layer, which in turn communicates with the Data Layer to store or retrieve information.
  - **Response:** The server sends a response back to the client, which is then displayed in the user interface.
- **Constraints**:
  - No persistent storage (data lost on app closure).
  - Simplified payment simulation (no real gateway).
  - Lack of Advanced Reporting features (financial reports, pattern analysis)

- **Excluded features**:
  - Mobile application.
  - Multi-language support.
  - Cloud integration.

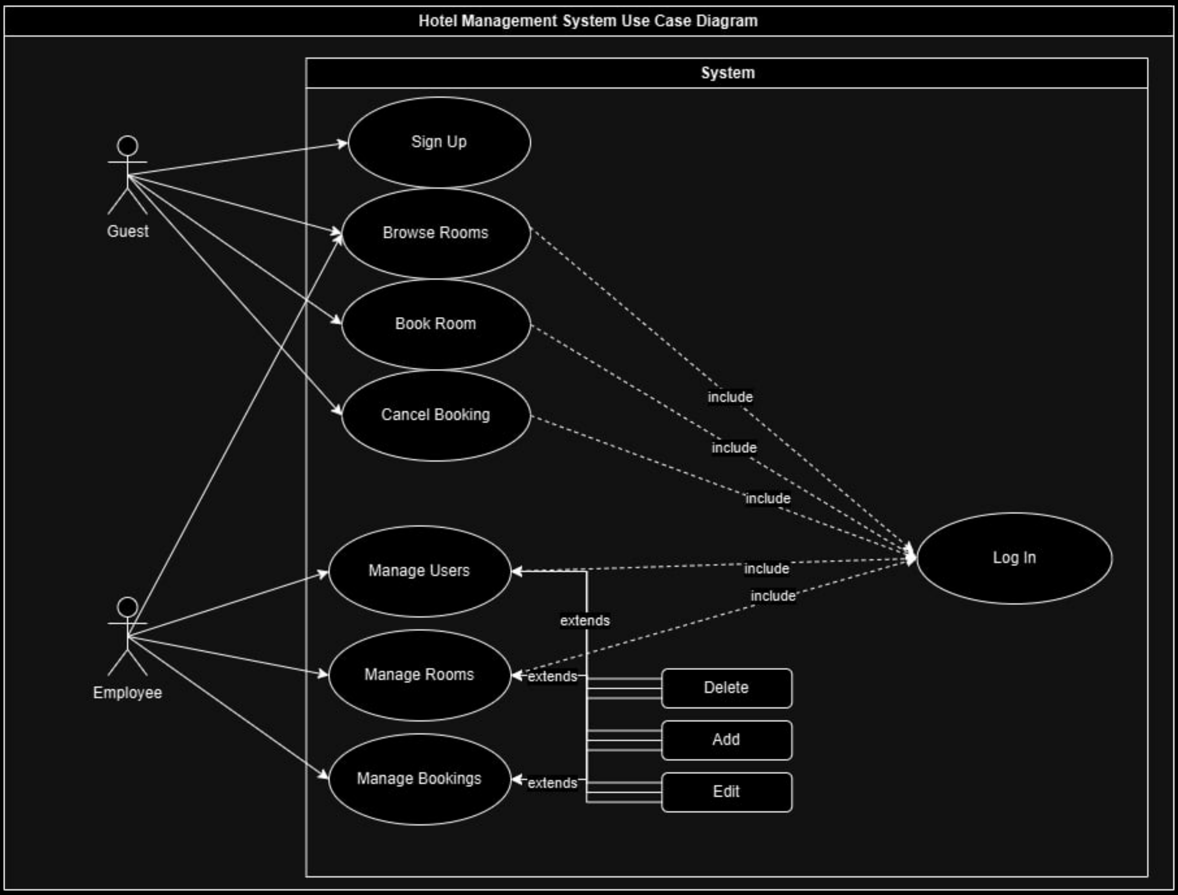
2.2 Functionalities (User Perspective)

Functionality	Description	Precondition
Register	Create an account with email, name, password.	
Book Room	Select room, dates, and confirm booking.	User must be logged in, and room must be available
Cancel Booking	Cancel via booking ID.	User must be logged in, and room must be available
Browse Rooms	Browse through the list of available rooms for booking.	User must be logged in

Non-Functional Features

1. **Usability**: An intuitive and easy-to-navigate interface for guests to interact with, especially for booking rooms and browsing.
2. **Performance**: The system responds quickly to guest queries with minimal delays.
3. **Scalability**: Ability to handle increased data as the number of guests and reservations grows.

3. Use Case Diagram



4. Use Case Priority list (User Perspective)

Use Case	Priority	Reason
Book Room	High	Core functionality for revenue generation.
Register	High	Essential for transactions and operations.
Cancel Booking	Medium	Secondary to booking.