Hotel Management System Requirements Documentation

By : Lukas Hamm, Mustafa Abrams, Kaleo Chase

Table of Contents

Customer Problems Statements & System Requirements

Problem Statement:

A hotel is almost like a living breathing organism. You have the front desk staff and managers who work 24/7 to help guests, the guests who are the hotel's customers, and other staff that help the hotel run to perfection. At every hotel there are a certain number of rooms available, each room pertaining to a different room type. The room types vary in range, but the number of rooms does not. Customers place reservations for an allotted amount of time with whatever specified room type they'd like, if that room type happens to be available at the time. Once the customer has stayed for their reservation the customer pays for their stay and leaves the hotel. This opens the room again making it available for another patron to use.

Glossary of Terms:

- **Guest** – a person who stays at the hotel and uses its services; can have a booking or not
- **Reservation** – an arrangement made in advanced or day of by a guest to secure a room for a certain period of time; includes different room types and services
- **Check in** – the process of registering a guest into the hotel system when they

  arrive
- **Check out** - the process of ending a guest's stay; normally includes settling payment and return of room keys
- **Room type** - category of rooms offered by the hotel
- **Front desk staff** – employees responsible for assisting guests with check in, check out, and guest inquiries

- **Room availability** – status of the room either being booked or vacant; used to determine if a guest can be checked into a room
- **Payment methods** – the option for guests to either pay with a debit/credit card for their stay
- **Guest profile** – stored record of a guest's personal information, preferences, history, and payment information
- **Cancellation** – process of voiding a stay before a guest arrives; maybe subject to a cancellation fee

User Interface Requirements:

No.	Priority Weight	Description
GuestTracker	High	Dashboard design for an overview, Guest list with details like name, room #, etc.
RoomType	High	Dropdown menu to let employees select from our 4 options
ReserverPro	High	Buttons next to guest name that allow for easy check-in/check-out from room
PaymentGateway	High	Ability to add payment to guest profile/for reservations

GuestProfile	Medium	Allows employees to update guests profiles at any time; Basic text box changes and saves it
EmployeeAdd	Medium	Allows managers to add/delete employees; Thinking large buttons for this
AvailabilityMatrix	High	Shows a monthly or weekly view with room availability (red = booked, green = available)
ServiceReq	Low	Allows guests let employees to know what they want during booking or at check-in; Will be added onto the current reservation or guest profile
GuestMax	High	Display a message if the guest limit has been reached.

Functional Requirements:

No.	**Priority Weight**	**Description**

GuestTracker	High	The management system should be able to have up to 50 guests checked into rooms
RoomType	High	The four different types of rooms are single queen, single king, double queen, double king
ReserverPro	High	The system should allow for booking, check-in, and check-out functionalities
PaymentGateway	High	The system should be able to process payments, only credit card/debit card allowed (NO CASH!)
GuestProfile	Medium	The system should store guest details (i.e. name, credit card details) and preferences for future bookings
EmployeeAdd	Medium	The system should allow managers to add/delete employees
AvailabilityMatrix	High	The system should provide a real-time view of room availability

ServiceReq	Low	The system should have the ability to add information about what the guest might ask for to their reservation
GuestMax	High	The system should not allow more than 50 guests at a time to be checked in.

Nonfunctional Requirements:

Usability: needed to ensure staff can keep reservations up to date and manage reservation data easily. It is also important to be able to access what is reserved to prevent double bookings.

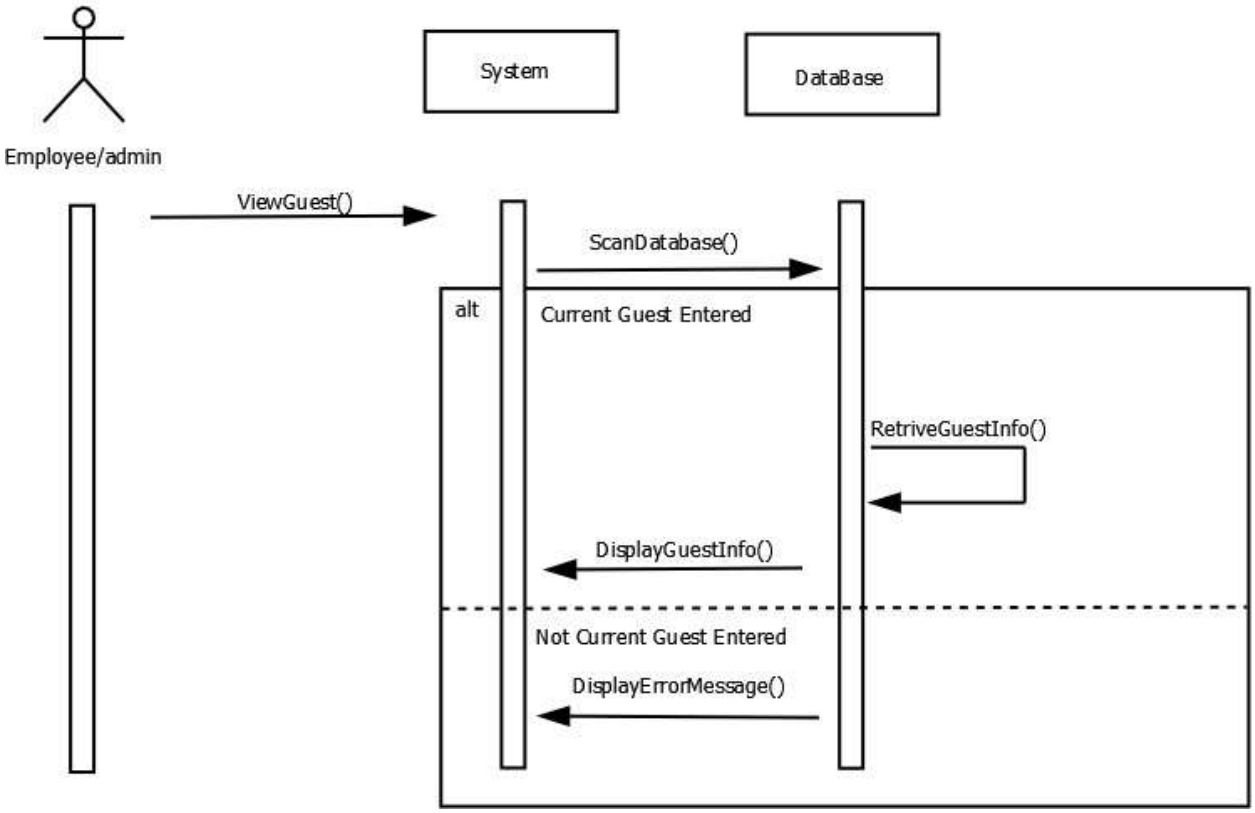
Reliability: The system needs to be stable in order to obtain booking information consistently. Also needed so that reservation information is input correctly, server or human error can cause major issues pertaining to peoples' reservations.

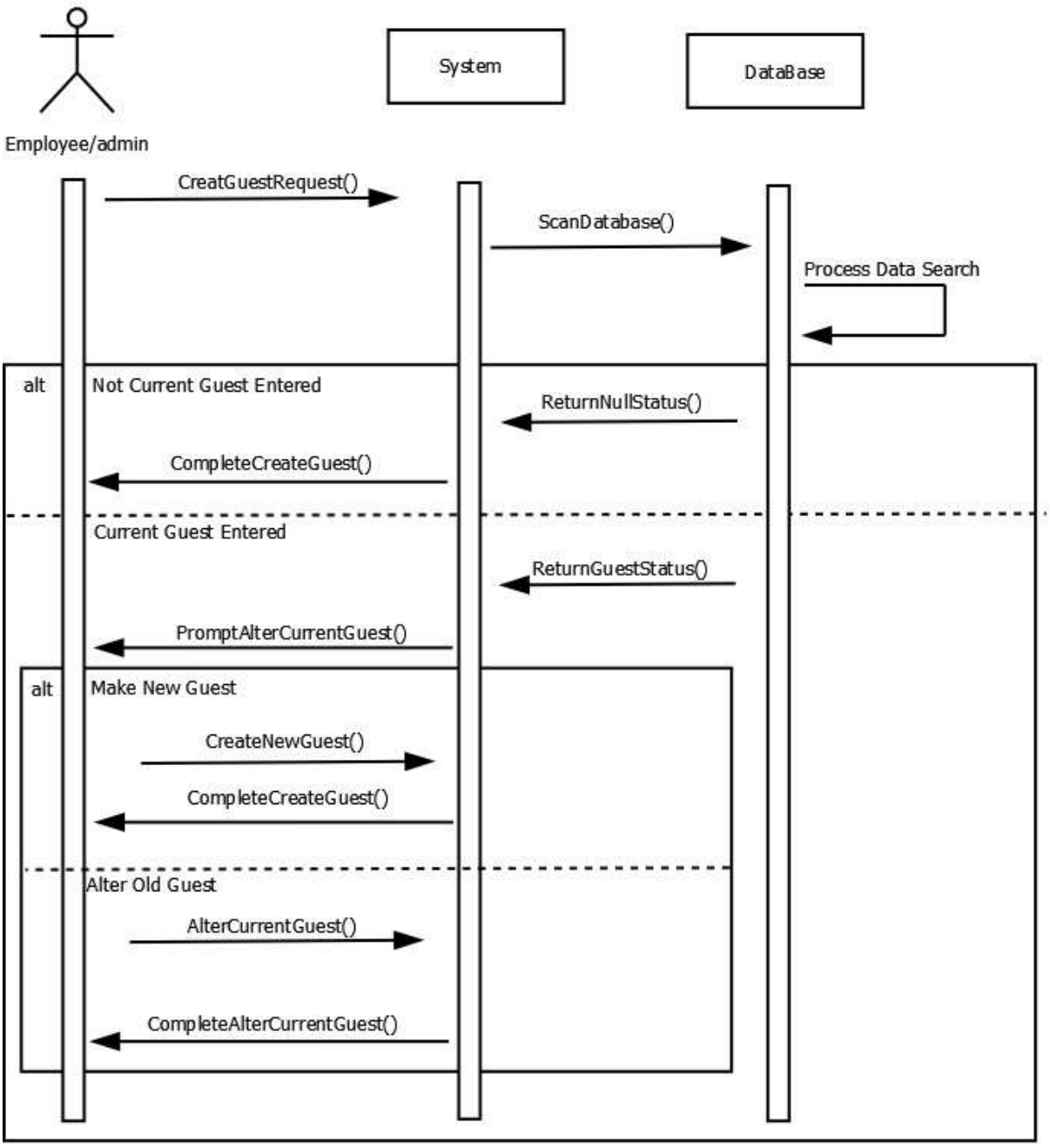
Performance: needed for the same reasons as usability, however the availability/accuracy of the booking information is more important compared to the speed/efficiency employees can update booking information.

Functionality: Capability reusability and security are not major issues pertaining to reservation info. However, still has some merit needing to protect sensitive customer data such as card information.

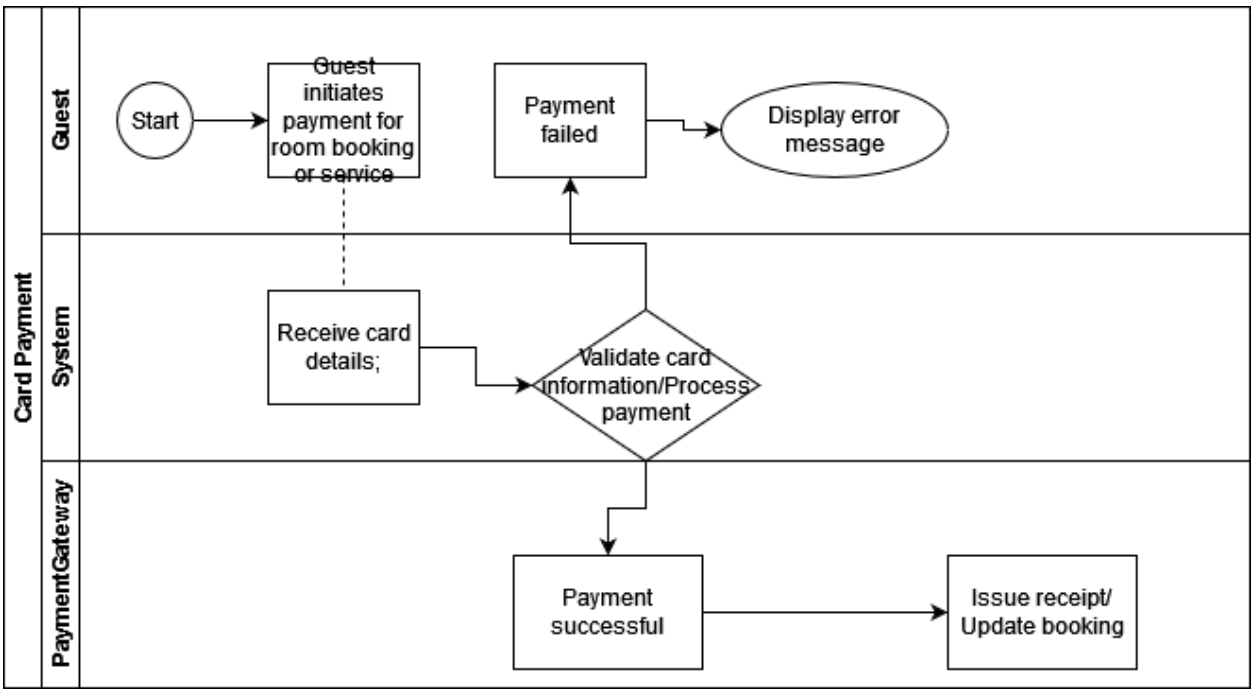
Supportability: Needed if the software ever needs slight updates. Whether it is to determine pricing, adding or removing room types, or altering the availability of requested services.

System Sequence Diagram :





Activity Diagram:


Card Payment
Guest
System
PaymentGateway
Start
Guest initiates payment for room booking or service
Payment failed
Display error message
Receive card details;
Validate card information/Process payment
Payment successful
Issue receipt/ Update booking

User Interface Specification:

Usage Scenario	Navigation	Clicks	Keystrokes
Add/delete employee	Homepage, add/delete employee, confirmation	<=5	<75
Searching guests	Homepage, search bar, search button, search results	<=3	<25
Viewing Guest Info	Homepage, search bar, search button, search results, guest name	<=5	0
Update Room Types	Homepage, update room type, adjust current room # and type, update room information	<=5	<25
Add/Delete Reservation	Homepage, guest search, reservation confirmation, add/delete reservation	<=10	<75

User Effort Estimations:

Mustafa Abrams

FA24-IN-INFO-C450-26590

Module 7 - User Interface Specification

User Interface Specification

- **Select Use Cases:**
    1. **Add/Delete Employee (Manager)**

Add/Delete Employee (Manager):

- Form fields for entering the employee name, ID, and role.
- Buttons for adding and deleting employees.
- Confirmation message after actions.

2. Search Guests (Front Desk Staff)

search Guests (Front Desk Staff):

- **Search bar to enter guest name or reservation number.**
- **Displays search results in a table format with guest details.**
- **When a result is selected, guest info is shown in a read-only format.**

3. **View Guest Info (Manager and Front Desk Staff)**:

- After selecting a guest from the search, their details are displayed for viewing only.

4. Update Room Types (Manager):

- A form for updating room types, sizes, services available, and price.
- Confirmation after saving.

5.Add/Delete Reservation (Manager and Front Desk Staff):

- A form to add or delete a reservation with guest details and dates.
- Buttons for both actions, with alerts after submission.

1. Add/Delete Employee (Manager)

- A form with the following input fields:
  - **Employee Name** (text box)
  - **Employee ID** (text box)
  - **Role** (dropdown or text box)
  - Buttons:
    - **Add Employee**
    - **Delete Employee**

The layout can be simple, with fields aligned vertically and buttons below the input fields. After submitting, the system should display a confirmation message like "Employee successfully added" or "Employee deleted."

2. Search Guests (Front Desk Staff)

- A search bar at the top where the user can enter the guest's name or reservation number.
- A **Search** button next to the search bar.
- Results are displayed below in a table with columns like:
  - **Guest Name**
  - **Reservation Number**
  - **Check-in/Check-out Dates**
  - **Room Number**

3. View Guest Info (Manager and Front Desk Staff)

- Once a guest is selected from the search results, a detailed view appears with fields like:
  - **Guest Name**
  - **Check-in Date**
  - **Check-out Date**
  - **Room Number**
  - **Special Requests** (if any)

Information would be displayed in read-only format for viewing purposes.

4. Update Room Types (Manager)

- A form with the following input fields:
  - **Room Type** (e.g., Single, Double, Suite)
  - **Room Size** (text field for square footage)
  - **Services Available** (e.g., Wi-Fi, breakfast included)
  - **Price** (numeric input)
  - A **Save** button to update the room type information.

5. Add/Delete Reservation (Manager and Front Desk Staff)

- A form with fields to add or delete a reservation:
  - **Guest Name** (text field)
  - **Room Selection** (dropdown)
  - **Check-in Date** (calendar input)
  - **Check-out Date** (calendar input)
  - Buttons:
    - **Add Reservation**
    - **Delete Reservation**

The form will validate the data and show confirmation after successful actions.

No.	Priority Weight (1 - 5: 1: lowest, 5: highest)	Description
PaymentGateway	5	Ability to add payment info to guest profiles/for reservations

ViewGuestInfo	2	Ability to view guest information from the dashboard
AddRez	5	Adding/deleting reservations from the database/dashboard
SearchGuests	4	Ability to search through guest profiles in the database
RoomType	2	Ability to select from the 4 different options of room types
AvailabilityMatrix	1	Shows the weekly view with room availability
GuestProfile	2	Adding a guest profile
ServiceReq	1	Ability to add information to a reservation about certain service requests from the guest
GuestMax	1	No more than 50 guest can be checked in

Req't	**PW**	**UC1**	**UC2**	**UC3**	**UC4**	**UC5**	**UC6**	**UC7**	**UC8**
REQ1	5	X							

REQ2	2		X						
REQ3	5			X					X
REQ4	4				X				
REQ5	2					X			
REQ6	1						X		
REQ7	2							X	
REQ8	1								X
REQ9	1							X	
Max PW	**5**	**5**	**2**	**5**	**4**	**2**	**1**	**2**	**5**
Total PW	**15**	**5**	**2**	**5**	**4**	**2**	**1**	**3**	**6**

No.	Description
UC1	allows the system to automate the price of rooms for the guests to pay
UC2	Allows managers and staff to see data associated with the chosen guest
UC3	allows reservations to be added and deleted by managers as well as staff
UC4	allows guests to be searched to find relevant data efficiently
UC5	allows information about room types to be altered (size or services associated with the given room

UC6	shows the number of rooms currently available to create a reservation for
UC7	allows data to be stored for a particular guest
UC8	Makes data associated with guests and/or reservations adjustable if it happens to change

Hotel Management System

System Architecture and System Design

Architectural Styles

Our hotel management system has a client server style.

Identifying Subsystems

- Employee Management
- guest Catalog
- room/service Processing

- Payment Gateway

Mapping Subsystems to Hardware

Our system is going to run on two separate machines. The client side which allows whoever is accessing the database to see all of its corresponding details through a convenient UI. As well as the server side which will be a compiled database that is able to be updated and modified by users with high enough authorization to do so. Otherwise, it will be a sperate entity to hold all of the proper data.

Persistent Data Storage

The hotel management system requires to save data outliving a single execution of the system. In our system, the persistent objects are the guests, employees/managers (usernames), room types, and payment gateways. The storage strategy is to store them in a relational database, in our case we are using MySQL.

Global Control Flow

Execution Orders

The hotel management system we have created is an event-driven system. The system has different pages with buttons where users can interact with clicks from their mouse. These events, when clicked, trigger different actions such as running a query or starting a new function.

Time Dependency

We do not have any types of time dependency or timers in our system. As stated above, since our system is event-driven, there is no concern for any type of real-time actions or a real-time system. Yes, guests are supposed to

be checked in and our at certain times, but there is no time constraint on this as it is an action that checks a user out of a room at a certain time.

Concurrency

Hardware Requirements

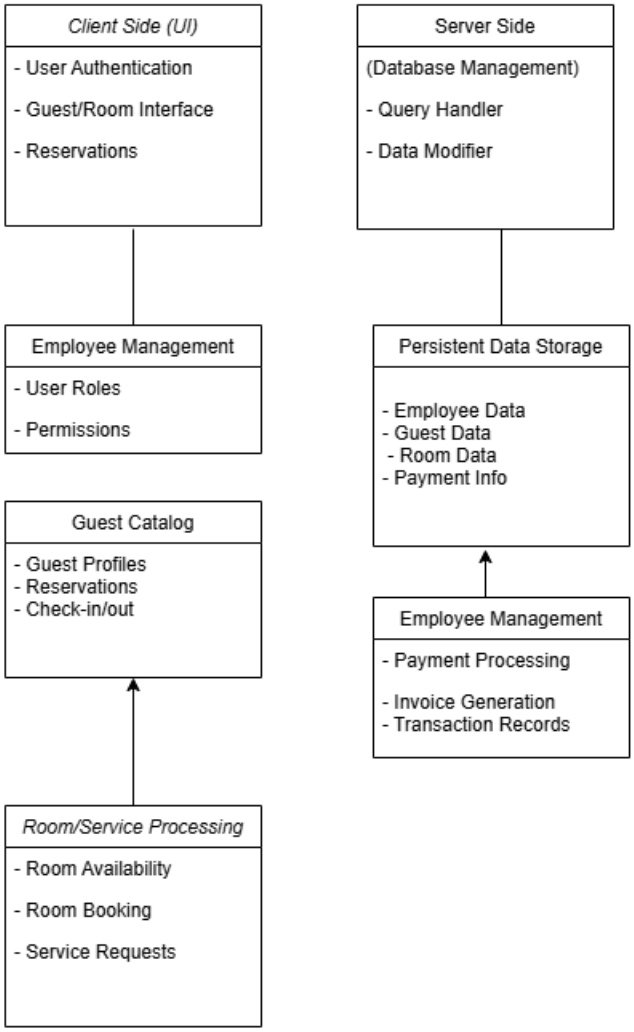
Color display

Minimum resolution: 1280x720

Storage needed: At least 1-2 GB of space

Minimum network speed/bandwith: 10 mb/s

Hotel Management System



Plan of Work:

Week 1: Set up what my front-end, back-end, and database will be used ✓

Week 2: Set up environment, connect all software together. Figure out the framework for the whole system. ✓

Week 3: Creation of a log-in page for managers and front desk staff.

Week 4: Creation of registration for new employees to add to the system .

Week 5: Creation of features such as add, modify, remove reservations from the system.

Week 6: Implementation of features into the system

Week 7: Same as Week 6.

Week 8: Verify that all implemented features work in unison.

Week 9: Test user (manager and front desk staff) authorization and registration

Week 10: Creation of features that allow more information from guests within the system (i.e. address, credit card information, phone number, email address).

Week 11: Implementation of said features from Week 10.

Week 12: Creation of features for billing, kind of adding onto the features from prior implementations.

Week 13: Same as week 12.

Week 14: Verifying that all implemented features in the second sprint work well. Debugging and performance testing as well.

Week 15: Record demonstration and progress made for final presentation.