Hotel Management System [HMS] Requirements Specification

Version 3.0

March 8th, 2025

Use this Requirements Specification template to document the requirements for your product or service, including priority and approval. Tailor the specification to suit your project, organizing the applicable sections in a way that works best, and use the checklist to record the decisions about what is applicable and what isn't.

The format of the requirements depends on what works best for your project.

This document contains instructions and examples which are for the benefit of the person writing the document and should be removed before the document is finalized.

To regenerate the TOC, select all (CTL-A) and press F9.

Table of Contents

1. Executive Summary

1.1 Project Overview

The Hotel Management System [HMS] is a management system designed to streamline the operations of a luxury hotel. This luxury hotel contains all of the normal services that every hotel is supposed to have but extending its capabilities by adding premium features such as casino, gym, restaurant and bar facilities. HMS also includes a mobile guest portal

2. Product/Service Description

2.1 Product Context

HMS is a centralized system (meaning that the system acts as a single unified platform for managing all hotel operations and data).

Characteristics of the Centralized HMS Core Operations:

- All **data** is stored in a centralized database, meaning that the data is consistent throughout the major components.
- End-Users can access the system through a single dashboard without the need of multiple systems or tools. The users would only be able to see features and data **relevant** to their roles.
- **Real-Time Updates** meaning changes made in one component/module are immediately reflected across all other modules (e.g guest checking-in activating the update of the room status for housekeeping and billing).
- Maintaining one system is proven to be more cost-effective than maintaining multiple.

HMS, while centralized, is **not** fully independent, it relies on third-party integrations for many of its core functionalities. Without these related systems, it would be difficult to make tasks like updating availability and processing payments possible and, in the worst case, those tasks might take manual intervention. This hybrid approach has the benefits of centralization combined with the external tools that would best require dedicated systems.

User Interface Overview:

The HMS is designed with multiple layers of user interfaces to cater to different user groups and functions. The interfaces are structured to provide both an intuitive experience for guests and a robust, role-specific interface for staff. The following outlines the key interface components:

1. Guest-Facing Interface: Mobile Guest Portal

- **Purpose:**

  Provides guests with a self-service platform to manage their stay, including booking rooms, submitting service requests, accessing digital keys, and viewing billing details.

- **Key Features:**

  - **Guest Profile Management:**

    Allows guests to create and update their personal information.

  - **Reservation and Booking:**

    Enables room booking, modification of reservations, and real-time availability checks.

  - **Service Requests:**

    Offers a streamlined process to request housekeeping, room service, or other amenities.

  - **Digital Key Access:**

    Facilitates room entry using a digital key, integrated via the Flexipass SDK.

  - **Feedback Submission:**

    Automatically collects post-stay feedback to enhance service quality.

2. Staff Interface: Secure Staff Login and Dashboards

Purpose:

Provides internal users (receptionists, housekeeping, department managers, facility staff, and administrators) with secure, role-based access to the various operational modules of the HMS.

Login Process:

Authentication:

Staff members log in using a secure login interface where they provide credentials (username and password). These credentials are validated against a centralized user database.

Role-Based Access Control (RBAC):

Post-authentication, the system identifies the user's role and restricts access to only those modules relevant to their responsibilities.

Session Management and Security:

Upon successful login, a secure session is initiated. All user activities are tracked through audit logs maintained by the Security and Administration Module.

Dashboard and Module Access:

Customized Dashboards:

After logging in, staff are directed to a role-specific dashboard. For example:

- **Receptionists:** Access the Front Desk Module for check-in/check-out operations and payment processing.
- **Housekeeping Staff:** Use a simplified interface for updating room statuses.
- **Department Managers:** View department-specific performance metrics and manage staff schedules.
- **Administrators:** Access advanced tools for system monitoring, user management, and security settings.

Third-Party Integrations:

Online Travel Agencies (OTAs) for real-time room availability sync (e.g., Booking.com) and searching and booking.

Payment Gateways (e.g., Stripe, PayPal) to process guest payments.

Digital keys generated by **Flexipass SDK**

Components of the HMS

1. Reservation and Booking Module

Purpose: Manages room reservations and bookings.

Key Features:

- Online booking interface (web and mobile).
- Real-time availability updates.
- Integration with OTAs (e.g., Booking.com).
- Cancellation and modification management.

Role: Ensures that room bookings are handled efficiently and accurately.

2. Front Desk Module

Purpose: Handles guest check-in, check-out, and front desk operations.

Key Features:

- Digital check-in/check-out.
- Room assignment.
- Invoice generation and payment processing.

Role: Provides a smooth and efficient experience for guests at the front desk.

3. Housekeeping Module

Purpose: Tracks room cleaning status and manages cleaning tasks.

Key Features:

- Real-time room cleaning status updates (e.g., clean, dirty, occupied).
- Task assignment to housekeeping staff.
- Notifications for urgent cleaning requests.

Role: Ensures that rooms are cleaned and maintained promptly.

4. Billing and Payment Module

Purpose: Manages guest bills and payment processing.

Key Features:

- Automated billing for room charges and additional services.
- Integration with payment gateways (e.g., Stripe, PayPal).
- Receipt generation and refund handling.

Role: Ensures accurate and timely billing for guests.

5. Facility Management Module

Purpose: Manages access and usage of premium facilities (e.g., gym, casino, restaurant, bar).

Key Features:

- Booking ID verification for facility access.
- Integration with digital key for billing.

6. Reporting and Analytics Module

Purpose: Provides insights into hotel performance and operations.

Key Features:

- Customizable dashboards for key metrics (e.g., occupancy, revenue).
- Financial reports (e.g., profit and loss, balance sheets).
- Analyze occupancy rates.
- Display review data (e.g. recent feedback, average ratings).

Role: Helps management make data-driven decisions.

7. Staff Management Module

Purpose: Manages employee schedules, attendance, and permissions.

Key Features:

- Shift scheduling for staff.
- Attendance tracking.
- Performance monitoring.
- Role-based access control.
- Department Specific Dashboard for managers.

Role: Ensures efficient management of hotel staff.

8. Security and Administration Module

Purpose: Ensures system security and manages user access.

Key Features:

- User account management.
- Role-based permissions.
- Audit logs for tracking system activity.
- System backups and recovery.

Role: Protects the system and ensures data integrity.

9. Finance Module

Purpose: Handles financial management and reporting.

Key Features:

- Payroll management.
- Tax calculations and compliance.
- Financial reporting (e.g., profit and loss, cash flow statements).

The HMS is designed to cater to a diverse set of users (customers and staff), each with unique needs and technical expertise. Below are the general customer profiles:

User Characteristics

- *Guests:*

  Profile: Primary users of the system, interacting with it to manage their stay, access services, and handle payments..
  Experience: Varies from tech-savvy to novice.
  Technical Expertise: Basic.
  Other Characteristics: Use the Mobile Guest Portal for booking rooms, making service requests, and viewing bills.

- *Receptionists:*

  Profile: Front-line staff managing guest check-ins, check-outs, and payments.
  Experience: Familiar with hotel operations and customer service.
  Technical Expertise: Moderate; comfortable using software for reservations, billing, and room management.
  Other Characteristics: Verify guest IDs, assign rooms, process payments, and generate invoices.

- *Housekeeping Staff:*

Profile: Housekeeping staff use the system to update the status of rooms. They need a straightforward and minimalistic interface to quickly log room status without requiring advanced technical skills.
Experience: Skilled in cleaning operations.
Technical Expertise: Low (basic UI interaction).
Other Characteristics: Update room status (clean/occupied) via a simplified interface.

- *Department Manager (Housekeeping, Food & Beverage, Casino):*
  Profile: Managers overseeing specific departments (areas of the hotel) and their staff.
  Experience: Experienced in department-specific operations, staff management, and decision-making.
  Technical Expertise: Moderate (comfortable using analytics, reports, and staff management tools).
  Other Characteristics:
  - View department-specific metrics (e.g., occupancy, staff performance).
  - Manage staff schedules, attendance, and performance within their department.

- *General Manager:*
  Profile: Oversees all hotel operations, including financial performance, staff management, and accounting.
  Experience: Experienced in hotel management, decision-making, leadership, and financial oversight.
  Technical Expertise: Moderate to High (comfortable using analytics, reports, and financial tools).
  Other Characteristics:
  - Monitor overall hotel performance (e.g., occupancy, revenue).
  - Oversee payroll, tax compliance, and financial reporting through the Finance Module.
  - Full access to every metric and employee data.
  - Ensure accurate billing and financial transactions across all modules.

- *Facility Staff (Casino, Gym, Restaurant, Bar):*

  Profile: Staff managing access and usage of premium facilities.
  Experience: Operate gyms, casinos, or restaurants.
  Technical Expertise: Low (scan/verify Booking IDs).
  Other Characteristics:
  - Verify guest access to facilities using Booking IDs.
  - Log service usage and charges.

- *Administrators:*

  Profile: Administrators are responsible for maintaining the system, ensuring security, and managing user access. They require advanced tools for technical configuration, monitoring, and troubleshooting to keep the system running smoothly.
  Experience: High (technical configuration).
  Technical Expertise: High (technical configuration).
  Other Characteristics: Manage user accounts, permissions, audit logs, and system backups.
  .

2.2 Assumptions

- **Internet Connectivity:** Staff and guests have reliable internet access for real-time system use.
- **Device Availability:** Staff have access to devices (PCs/tablets) to interact with the system.
- **Third-Party Systems**: OTAs, POS, and payment gateways are operational and accessible.
- **Staff Training:** Users receive basic training to navigate their assigned modules.
- **Guest Compliance:** Guests provide valid IDs and adhere to hotel policies during check-in.

2.3 Constraints and Dependencies

It is important to consider constraints for the Hotel Management System's design and implementation, the constraints focus on resource, technical and security constraints. The criticality of the system should also be considered as the system is designed to be used by a business (in this case a hotel), considerations are made for the system as it impacts the operations, revenue and guest satisfaction. Audit trails are also important for ensuring transparency and compliance, key considerations will be made regarding Audit Logging and Trailing.

Constraints

Resource Constraints (Hardware, Software, Human):

- The HMS must handle a number of concurrent users.
- Ensure sufficient server resources to avoid bottlenecks.
- To support the **digital key system,** compatible hardware is needed (e.g. IoT enabled door locks).
- The Mobile Guest Portal must be compatible with a wide range of devices (iOS, Android) and screen sizes.
- Integrate with external third-party systems (e.g. OTAs, Payment Gateways) using APIs.
- Skilled developers are needed in the development team.
- Staff must be trained to use the HMS effectively.

Security Constraints (Data Protection, Audit Trails, Vulnerability Management):

- Sensitive data (e.g. guest information, payment details) should be encrypted both in transit and at rest
- Role-based Access Control (RBAC) to restrict access to sensitive modules
- Ensure digital keys are encrypted and can be revoked instantly if compromised.
- Conduct regular security assessments to identify and fix vulnerabilities.
- Keep all software components (e.g., operating systems, libraries) up to date with security patches.

Criticality Constraints (Impact of the Business, Downtime Costs, Disaster Recovery)

- Handling of booking, payments, billing, crucial to the revenue of the business.
- Poor guest experiences caused by malfunctioning of the system.
- Downtime of the system can result in bookings being lost, leading to revenues also being lost.
- Implement regular backups of the database and system configurations.
- Ensure quick restoration in case of data loss or system failure.

Audit Trails

- **Audit Logging:** Log all actions performed by users (e.g. check-ins, billing adjustments and log all system events (e.g. every API call, error messages).
- Ensure all logs are stored in a location that is secure and tamper-proof.
- Define a log retention policy of 7 years (year is specified for this project).
- Monitor logs in real-time.
- Maintain logs in a format that can be easily reviewed during credits.

Dependencies

Internal Dependencies

Centralized Database:

- **Dependency:** A unified data repository that serves all modules (reservations, billing, housekeeping, etc.).
- **Impact:** Ensures consistency and real-time updates across different operations. Database performance and uptime are critical for overall system functionality.

Mobile Guest Portal:

- **Dependency:** Interface for guests to manage bookings, check-in/out, service requests, and payments.
- **Impact:** Must be compatible with various devices and screen sizes, as well as reliably communicate with the centralized database.

Inter-Module Communication:

- **Dependency:** Seamless integration among internal modules (Reservation, Front Desk, Housekeeping, Billing, Reporting, etc.).
- **Impact:** Efficient inter-module communication is necessary to maintain data consistency and to trigger real-time updates across different areas of the hotel.

Third-Party Integrations

Online Travel Agencies (OTAs):

- **Dependency:** Real-time room availability synchronization and reservation updates.
- **Impact:** Without active OTA integration, booking data may not be up-to-date, affecting occupancy and revenue.

Payment Gateways (e.g., Stripe, PayPal):

- **Dependency:** Processing guest payments and handling refunds.
- **Impact:** Reliance on these services for secure payment processing is critical; any downtime or security issue can directly affect financial transactions.

Flexipass SDK:

- **Dependency:** Generation and management of digital keys for guest room access.
- **Impact:** This integration is crucial for ensuring secure and convenient digital key issuance. Disruptions could impact the guest check-in experience.

3. Requirements

3.1 Functional Requirements

Req#	Requirement	Comments	Priority	Date Rvwd	SWE Reviewed / Approved
FR_GST_01	The system shall allow the guests to search for available rooms by room type, check-in and check-out dates and price range through the OTAs and seamlessly redirect them to the Mobile Guest Portal to complete the booking.		1		
FR_GST_02	The system shall allow the guests to complete their booking on the Mobile Guest Portal by including pre-filled search details from OTAs, guest account creation or login, room selection and entry of personal and payment details. After payment is confirmed, the system shall then send the request for approval to the receptionist through the Mobile Guest Portal.		1		
FR_GST_03	The system shall allow the guest to cancel a booking before check-in and shall refund the payment.		1		
FR_GST_04	The system shall allow the guest to request a booking modification before check-in (e.g. room type, check-in date). The system shall then send the request to the receptionist. The receptionist can check if the modification is possible (e.g. new specified room type is available) and confirm it.		1		

Req#	Requirement	Comments	Priority	Date Rvwd	SWE Reviewed / Approved
FR_REC_01	The system shall provide to the receptionist a real-time inventory of room status (available, booked, under maintenance). The real-time availability list shall be displayed on the receptionist's dashboard.		1		
FR_REC_02	The system shall allow the receptionist to modify a booking in special cases (e.g. room is under maintenance) and shall send a notification to a guest.		1		
FR_GST_05	The system shall allow guests to check-in through the mobile app or the receptionist can complete the check-in for them. After the check-in the system shall generate the digital key for the room (valid for the duration of stay) which a customer can store in their phone.		1		Hazis Voda
FR_GST_06	The system shall use Flexipass to connect the digital key provided to the guest with the lock in their room.		1		Hazis Voda
FR_GST_07	A guest can access facilities and use services of the hotel after a facility staff member has scanned their digital key in the mobile guest portal.		1		Hazis Voda
FR_GST_08	The system shall register each expense made by the guest to their final invoice.		1		Jurgen Hila
FR_GST_09	The guest can request room service through the mobile guest portal.		2		Jurgen Hila

Req#	Requirement	Comments	Priority	Date Rvwd	SWE Reviewed / Approved
FR_GST_10	The system shall allow guests to view their expenses at any time through the mobile guest portal.		1		Jurgen Hila
FR_GST_11	When a guest is checking out, the system shall generate an itemized invoice, which guests can pay online through payment gateways or at the front desk and shall revoke the digital key.		1		Xhois Cano
FR_GST_12	The system should automatically send a survey through the mobile guest portal to collect feedback after check-out.		2		Xhois Cano
FR_GM_01	The system should generate reports for the general manager summarizing post check-out guest survey feedback.		2		Xhois Cano
FR_GST_13	The system can allow guests to request urgent cleaning by specifying the reason of the request. The housekeeping staff on duty shall be alerted for the request.		3		Orgest Baçova
FR_HK_01	The system shall allow housekeeping staff to notify of a maintenance issue (e.g. plumbing or electrical) to the housekeeping department manager.		2		Orgest Baçova
FR_GST_14	The system should track the cleaning schedule for each room based on guest preferences. The guests might indicate whether they want daily cleaning or a different schedule. A room is automatically scheduled for cleaning when the guest has checked-out or when the guest hasn't specified a custom schedule (in that case daily cleaning).		1		Orgest Baçova

Req#	Requirement	Comments	Priority	Date Rvwd	SWE Reviewed / Approved
FR_HKM_01	The system should be able to provide housekeeping department managers with a real-time view of all room cleaning statuses. Room status must be either Dirty or Clean.		3		Daron Delvina
FR_HKM_02	The system should allow Housekeeping Department managers to assign cleaning tasks to housekeeping staff.		2		Daron Delvina
FR_HK_02	The housekeeping staff shall be able to view a list with the rooms assigned to them for cleaning.		1		Daron Delvina
FR_HK_03	The housekeeping staff shall be able to update the status of the room after cleaning.		1		
FR_DM_01	All Department managers shall schedule shifts for all staff in their department.		1		
FR_DM_02	Department managers shall be able to view details about staff department, like staff list, performance and attendance for members of staff of their department(e.g. task completion time).		1		
FR_GM_02	The general manager shall be able to view staff performance, attendance for the entire hotel.		1		
FR_GM_03	The general manager shall have access to hotel metrics like occupancy rates, revenue and expenses.		1		
FR_GM_04	The system shall generate customizable reports according to filters specified by the general manager (for stakeholder meetings).		2		
FR_GM_05	The system shall allow the general manager to view audit logs for transparency and compliance.		1		
FR_GM_06	The general manager shall handle payroll processing and tax compliance.		1		

Req#	Requirement	Comments	Priority	Date Rvwd	SWE Reviewed / Approved
FR_DM_03	The department manager shall be able to register new hirees in the staff list of their department.		1		
FR_ADM_01	The administrator shall be able to manage user access, and grant and revoke permissions for each employee.		1		
FR_ADM_02	The administrator shall be able to back up the system and restore it in case of system failure.		1		
FR_FS_01	Facility staff members could use the system to view their work hours.		3		
FR_GST_15	The Mobile Guest Portal shall provide a simple form for guests to submit complaints to the relevant department manager regarding the complaint type and should therefore alert that manager.		2		

3.1.1 Product Requirements

Requirements which specify that the delivered product must behave in a particular way e.g. execution speed, reliability, etc.

3.2.1.1 Usability Requirements

The system needs to be easy and intuitive for users to interact with the system, these are the usability requirements:

- The system's interface must be intuitive and easy to use, requiring no more than 1 hour of training for new employees to perform basic operations.
- The user interface should incorporate best practices in UX design, ensuring users can efficiently navigate through the system without extensive guidance.
- The system must be designed for accessibility, with settings like high contrast, text-to-speech, and alternative text for images.
- The system should have a complete documentation showcasing the detailed roles, privileges, and permissions for each user type, ensuring that users using the system are guided on their respective functionalities.

3.2.1.2 Performance Requirements

Scalability:

- The system shall be designed to handle up to 500 simultaneous users (guests and staff) without any degradation in performance.
- The system shall be scalable to accommodate future growth, supporting up to 1000 users without requiring major architectural changes.

Response Time:

- The system must process room booking requests, in less than 3 seconds per transaction under normal load conditions.
- All user interactions, including guest check-in/check-out, transaction processing, and room service requests, must be completed with minimal delays (within 2 seconds) to enhance the user experience.

System Performance:

- The system must be capable of handling up to 1000 database transactions per minute without slowdowns or system crashes.
- The database should be optimized to ensure fast query execution, especially during high-traffic periods (e.g., check-in/check-out times).
- Load testing must be conducted regularly to simulate peak usage and ensure the system meets performance benchmarks.

3.2.1.3 Availability

- The system shall maintain an uptime of 99.9% during business hours (6 AM–12 AM local time).
- Outside business hours, the system shall maintain an uptime of 99.5%.
- Critical modules (e.g., booking, check-in/out, payment processing) shall have 99.99% availability during peak hours (9 AM–11 AM and 6 PM–8 PM).

- **Scheduled Maintenance:** Maintenance windows shall not exceed 2 hours per month and shall occur during off-peak hours (2 AM–4 AM local time).
- **Scheduled Maintenance:** Users shall be notified of scheduled maintenance 72 hours in advance via email and in-system notifications.
- **Unscheduled Maintenance:** The system shall notify users immediately upon detecting an issue and provide regular updates until resolution.
- **Unscheduled Maintenance:** Unscheduled maintenance shall not exceed 1 hour per quarter**.**
- The system shall have a *Mean Time Between Failures* of ≥10,000 hours.
- The system shall not exceed 1 system-wide failure per quarter.
- For individual modules (e.g., booking, billing), the maximum permitted failures shall be ≤2 per month.

3.2.1.4 Security

The system shall follow the following requirement procedures:

Encryption:

- All sensitive data (e.g., guest details, payment information, keycard data) shall be encrypted using AES-256 encryption.

Activity:

- The system shall log 100% of user actions (e.g., logins, bookings, payments) with timestamps, user IDs, and IP addresses.
- Audit logs shall be retained for 7 years for compliance and auditing purposes.
- Audit logs shall be accessible only to authorized administrators and protected from tampering.

Role-Based Access Control:

- Modules shall communicate only with authorized modules based on predefined roles and permissions (e.g., housekeeping cannot access billing data).
- User sessions shall expire after 15 minutes of inactivity and require re-authentication.

Data Integrity Checks:

- The system shall perform daily checksum validation on critical data sets (e.g., guest profiles, booking records) to detect tampering.

Malware and Intrusion Prevention:

- The system shall run real-time antivirus and anti-malware scans on all servers and endpoints.
- The system shall include an IDS to detect and block unauthorized access attempts.

3.1.2 Organizational Requirements

The system needs to follow these following organizational policies and procedures:

Environmental Requirements:

- These requirements focus on sustainability, energy efficiency, and ethical considerations:
- The system shall minimize resource usage (e.g., CPU, memory) to reduce energy consumption.

Operational Requirements:

Security and Access Control

- All users (staff and administrators) shall use strong passwords (minimum 12 characters, including uppercase, lowercase, numbers, and special characters) that expire every 90 days.

- All security incidents (e.g., data breaches, unauthorized access) shall be reported to the IT security team within 15 minutes of detection.

Backup and Recovery

- The system shall perform daily automated backups of all critical data (e.g., bookings, guest details, transactions).
- A disaster recovery plan shall be in place, with a maximum recovery time objective (RTO) of 1 hour for critical modules.

Vendor and Third-Party Management

- All third-party vendors (e.g., payment gateways, OTAs) shall comply with the organization's security and privacy policies.
- Third-party vendors shall undergo annual security audits to ensure compliance with organizational standards.

Development Requirements:

- The system shall be implemented in phases, starting with core modules (e.g., booking, check-in/out) and gradually adding more advanced features.
- A pilot implementation shall be conducted at one hotel location for 3 months to identify and resolve issues before full rollout.
- The system shall be developed using Agile methodologies, with bi-weekly sprints, daily stand-ups, and sprint reviews.

3.1.3 External Requirements

Interoperability Requirements

Third-Party Integrations:

- The system shall integrate with Online Travel Agencies (OTAs) (e.g., Booking.com, Expedia) for real-time room availability synchronization.
- The system shall integrate with Point-of-Sale (POS) systems (e.g., Square, Toast) for restaurants and bars to log orders and link charges to guest bills.
- The system shall integrate with payment gateways (e.g., Stripe, PayPal) to process guest payments securely.

Legislative Requirements

GDPR/CCPA Compliance:

- The system shall provide tools for guest data consent, deletion requests, and access logs to comply with GDPR (General Data Protection Regulation) and CCPA (California Consumer Privacy Act).

PCI DSS Compliance:

- Payment processing modules shall comply with PCI DSS v4.0 standards to ensure secure handling of credit card information.
- The system shall automatically apply region-specific taxes (e.g., VAT, tourism taxes) to invoices to comply with local tax laws.

Globalization Requirements

- The system shall support 5 languages (e.g., English, Spanish, French, Mandarin, Arabic) for guest and staff interfaces.
- The system shall support 10 currencies for payments and billing, with real-time exchange rate updates.

4. User Scenarios/Use Cases

Provide a summary of the major functions that the product will perform. Organize the functions to be understandable to the customer or a first time reader. Include use cases and business scenarios, or provide a link to a separate document (or documents). A business scenario:

- Describes a significant business need
- Identifies, documents, and ranks the problem that is driving the scenario
- Describes the business and technical environment that will resolve the problem
- States the desired objectives
- Shows the “Actors” and where they fit in the business model
- Is specific, and measurable, and uses clear metrics for success

Use cases are associated with a particular Functional Requirement. Assuming you have the first functional requirement named BR_01, you will map it into the Use Case called UC_01 and user scenario US_01. Please keep this naming convention throughout all your use cases and diagrams.

5. Diagrams

In this section you are going to place all of the diagrams that you build throughout to the course, in following with the slides presented throughout the weeks.

5.1 ER Diagram

Standard ERD for your project. Not much but the skills gained in the DBMS course are required.

5.2 Use Case Diagram (general)

Use Case Diagram (only one, with all the use cases).

5.3 Activity Diagram

Each Activity Diagram should be associated with an use case, associated with a particular requirement which is further associated with a particular use-case. E.g BR_01 which becomes UC_01 which becomes AC_01.

5.4. Class diagram.

One class diagram (general) for all the classes. Edit it afterwards with the design pattern implemented in it.

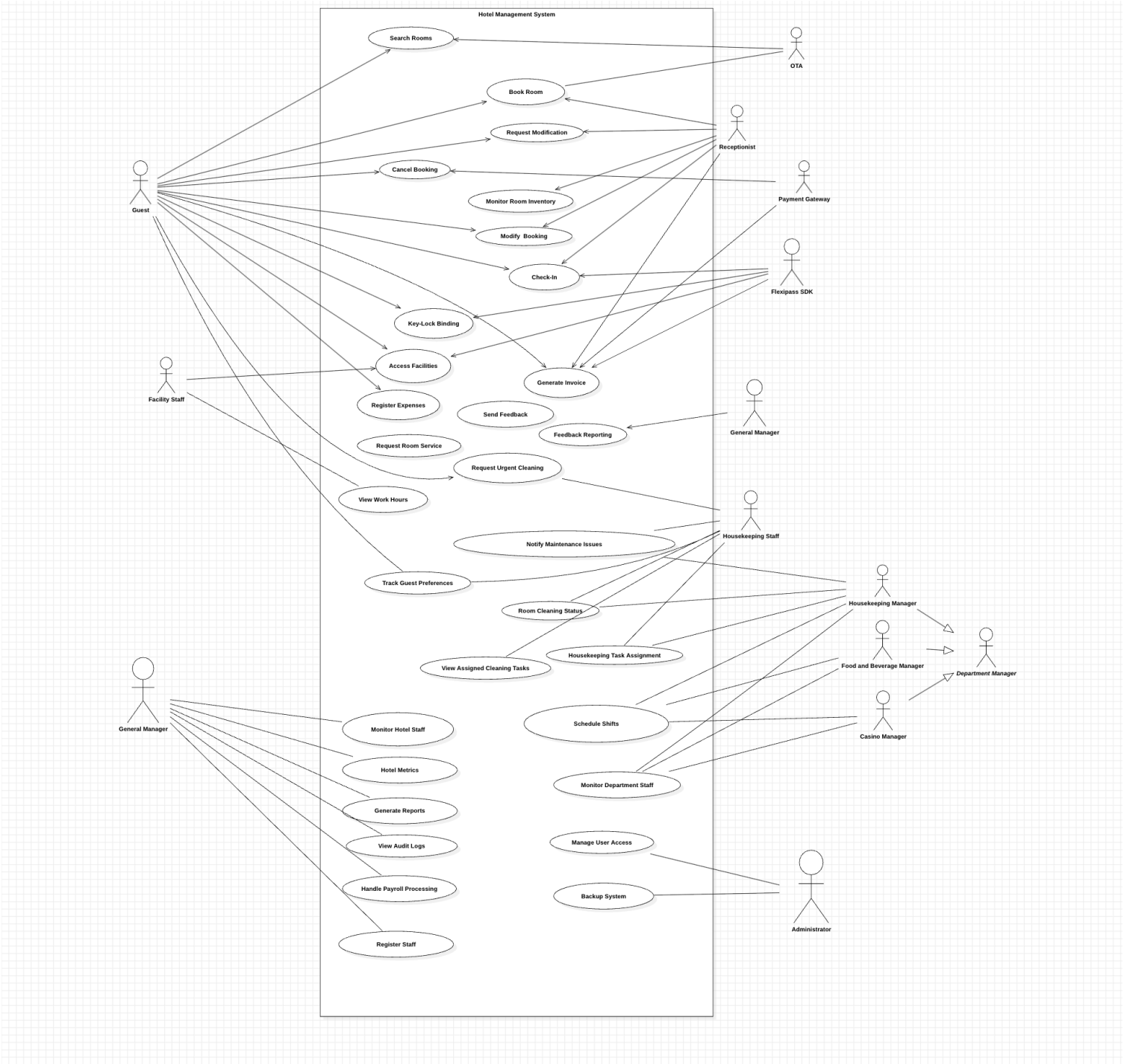
5.5 State diagram

Place all the relevant state diagrams here.

5.6 Sequence diagram.

All sequence diagrams are associated with an Activity Diagram. A Sequence Diagram is built based on an activity diagram. If the activity diagram is named AC_07, the Sequence Diagram will be named SC_07.

5.7. Collaboration diagram



All collaboration diagrams directly relate to a sequence diagram. If a sequence diagram is named SC_07, then the collaboration diagram is named CC_07

6. Design Patterns

Choose the relevant design patterns for your project. For each, give a reasoning and the associated class and sequence diagram. These are NOT part of the above diagrams, and need not carry the following naming scheme.

7. Appendix.

Organizing the Requirements

This section is for information only as an aid in preparing the requirements document.

Detailed requirements tend to be extensive. Give careful consideration to your organization scheme. Some examples of organization schemes are described below:

By System Mode
Some systems behave quite differently depending on the mode of operation. For example, a control system may have different sets of functions depending on its mode: training, normal, or emergency.

By User Class
Some systems provide different sets of functions to different classes of users. For example, an elevator control system presents different capabilities to passengers, maintenance workers, and fire fighters.

By Objects
Objects are real-world entities that have a counterpart within the system. For example, in a patient monitoring system, objects include patients, sensors, nurses, rooms, physicians, medicines, etc. Associated with each object is a set of attributes (of that object) and functions (performed by that object). These functions are also called services, methods, or processes. Note that sets of objects may share attributes and services. These are grouped together as classes.

By Feature
A feature is an externally desired service by the system that may require a sequence of inputs to affect the desired result. For example, in a telephone system, features include local call, call forwarding, and conference call. Each feature is generally described in a sequence of stimulus-response pairs, and may include validity checks on inputs, exact sequencing of operations, responses to abnormal situations, including error handling and recovery, effects of parameters, relationships of inputs to outputs, including input/output sequences and formulas for input to output.

By Stimulus
Some systems can be best organized by describing their functions in terms of stimuli. For example, the functions of an automatic aircraft landing system may be organized into sections for loss of power, wind shear, sudden change in roll, vertical velocity excessive, etc.

By Response
Some systems can be best organized by describing all the functions in support of the generation of a response. For example, the functions of a personnel system may be organized into sections corresponding to all functions associated with generating paychecks, all functions associated with generating a current list of employees, etc.

By Functional Hierarchy
When none of the above organizational schemes prove helpful, the overall functionality can be organized into a hierarchy of functions organized by common inputs, common outputs, or common internal data access. Data flow diagrams and data dictionaries can be used to show the relationships between and among the functions and data.

Additional Comments
Whenever a new Requirements Specification is contemplated, more than one of the organizational techniques given above may be appropriate. In such cases, organize the specific requirements for multiple hierarchies tailored to the specific needs of the system under specification.

There are many notations, methods, and automated support tools available to aid in the documentation of requirements. For the most part, their usefulness is a function of organization. For example, when organizing by mode, finite state machines or state charts may prove helpful; when organizing by object, object-oriented analysis may prove helpful; when organizing by feature, stimulus-response sequences may prove helpful; and when organizing by functional hierarchy, data flow diagrams and data dictionaries may prove helpful.