

***Hotel Management system***

***Requirements Specification***

**Group members:**

*Jurgen Halili (Group Leader)*

*Endi Shima*

*Arlind Hoxha*

*Soni Deliallisi*

*Kosta Belba*

*Vinsent Hasula*

**Accepted by:**

*Ari Gjerazi, Julia Marku*

Table of Contents

1.Executive Summary ..................................................................................................................................3

1.1 Project overview......................................................................................................................................3

1.2 Purpose and scope of this specification ................................................................................................3

2.Product/ Service Description ....................................................................................................................4

2.1 Product context .....................................................................................................................................4

2.2 User Characteristics............................................................................................................................... 4

2.3 Assumptions...........................................................................................................................................5

2.4 Constraints ............................................................................................................................................5

2.5 Dependencies........................................................................................................................................6

3.Requirements...........................................................................................................................................7

3.1 Functional Requirements ..................................................................................................................8-12

3.2 Non-Functional Requirements..............................................................................................................13

3.2.1 Product Requirements.......................................................................................................................13

3.2.1.1 User Interface Requirements..........................................................................................................13

3.2.1.2 Usability...........................................................................................................................................14

3.2.1.3 Efficiency.........................................................................................................................................14

3.2.1.3.1 Performance Requirements.........................................................................................................14

3.1.2.3.2 Responsiveness..........................................................................................................................15

3.2.1.4 Dependability .................................................................................................................................16

3.2.1.5 Security...........................................................................................................................................17

3.2.2 Organizational Requirements............................................................................................................18

3.2.1.1 Environmental Requirements.........................................................................................................18

3.2.1.2 Operational Requirements.........................................................................................................18-19

3.2.1.3 Development Requirements...........................................................................................................19

3.2.3 External Requirements......................................................................................................................21

3.2.3.1 Regulatory Requirements..............................................................................................................21

3.2.3.2 Ethical Requirements.....................................................................................................................22

3.2.3.3 Legislative Requirements...............................................................................................................23

3.2.3.3.1 Accounting Requirements...........................................................................................................23

3.2.3.3.2 Security Requirements................................................................................................................24

3.3 Domain Requirements.........................................................................................................................25

4.Software Design......................................................................................................................................27

4.1 User Scenarios................................................................................................................................27-34

4.2 Use Cases ......................................................................................................................................34-37

4.3 BPMN..............................................................................................................................................37-38

4.4 Data Flow Diagrams........................................................................................................................38-41

4.5 Entity-Relationship Diagram.................................................................................................................42

4.6 Activity Diagrams.............................................................................................................................43-48

4.7 Timing Diagram.....................................................................................................................................49

4.8 State Diagram..................................................................................................................................50-55

4.9 Sequence Diagrams........................................................................................................................56-59

4.10 Collaboration Diagrams.....................................................................................................................60

4.11 Class Diagram....................................................................................................................................61

5. Design Patterns ................................................................................................................................61-62

6.Appendix.............................................................................................................................................63-70

1. **Executive Summary**

**1.1 Project overview:** Our project is a comprehensive hotel management system designed to streamline and automate various aspects of hotel operations. The system caters to a diverse range of users, including clients, managers, administrators, and workers, each with their specific roles and responsibilities.

The primary goal of the system is to provide an efficient and user-friendly platform that enhances the overall guest experience, optimizes operational efficiency, and ensures seamless communication and coordination among different stakeholders within the hotel environment. By leveraging advanced technology and incorporating various security measures, the system aims to meet the evolving needs and demands of the hospitality industry.

The functionalities of the system are designed to cover a wide range of areas, including room reservations, pricing management, inventory control, staff management, guest services, and administrative tasks. These functionalities are implemented at different levels of security to ensure that the right access and privileges are granted to the appropriate user roles.

**1.2 Purpose and scope of this specification:** The purpose of this specification document is to provide a comprehensive overview of the hotel management system and its functionalities. It serves as a reference guide for the development team, stakeholders, and other relevant parties involved in the project.

The scope of this specification encompasses the entire system, from its conceptualization and design to its implementation and deployment. It outlines the key features and capabilities of the system, along with the specific user roles and their corresponding access rights. The document also highlights the non-functional requirements, such as usability, performance, security, and compliance.

By clearly defining the purpose and scope of the project, this specification serves as a roadmap for the development team, guiding them in the implementation of the system's features and functionalities. It ensures that all stakeholders have a clear understanding of the project objectives, enabling effective collaboration and communication throughout the development process.

Additionally, this specification document sets the foundation for future enhancements and updates to the system. It provides a baseline for evaluating the system's performance, identifying areas for improvement, and accommodating future changes or additions to meet evolving business needs and industry trends.

1. **Product/Service Description**

**2.1 Product Context**

The Hotel Management System is designed with the specific context of the tourism industry in mind. It recognizes the unique needs and challenges faced by hotels in delivering exceptional guest experiences and optimizing their operations. By tailoring the system to the hotel it serves, it becomes a valuable tool in automating processes, improving efficiency, and enhancing the overall guest satisfaction.

Considering the proximity of the hotel to a beach, the system incorporates features that cater to beach-related activities. For instance, clients can easily find information about nearby beach events, book beach umbrellas, and access weather updates to plan their activities accordingly. This integration adds value to the overall guest experience, making the system a valuable asset for the hotel in attracting beach-loving travelers.

**2.2 User Characteristics**

The Hotel Management System places great emphasis on usability, recognizing that users of the system will come from diverse backgrounds and possess varying levels of technical expertise. The goal is to create an intuitive and user-friendly interface that enables all users to interact with the system effortlessly.

To achieve this, the system undergoes rigorous testing and incorporates feedback from users throughout the development process. User experience (UX) designers conduct extensive research to understand the needs, preferences, and pain points of different user groups. This includes conducting user interviews, surveys, and usability testing sessions to gather valuable insights.

Based on the findings, the system's user interface is carefully designed to facilitate easy navigation and seamless interactions. The layout is structured logically, ensuring that users can locate and access the desired features without confusion. Visual cues, such as clear icons, intuitive buttons, and consistent color schemes, are employed to enhance usability and provide a visually appealing experience.

Moreover, the system takes into consideration accessibility requirements for users with special needs. Accessibility guidelines, such as those outlined by the Web Content Accessibility Guidelines (WCAG), are followed to ensure that individuals with visual impairments, hearing impairments, or mobility limitations can effectively utilize the system. This includes providing alternative text for images, incorporating keyboard navigation support, and ensuring proper color contrast for readability.

In addition to the user interface design, the system also provides ample support and assistance to users. Clear and concise user documentation, tutorials, and tooltips are available to guide users through the system's functionalities. Help desk or customer support services are readily accessible to address any user inquiries or issues promptly.

By prioritizing overall easability and reachability, the Hotel Management System aims to empower all users, regardless of their technological background or familiarity with similar systems. The dedication to user-centered design principles ensures that the system offers a seamless and enjoyable experience for all individuals engaging with it, contributing to high user satisfaction and efficient system utilization.

**2.3 Assumptions**

In developing the Hotel Management System, several considerations and assumptions are made to ensure a focused and relevant solution that caters to the specific needs of the hotel and its guests. These assumptions are based on careful analysis of the hotel's location, target market, and industry best practices. By taking these factors into account, the system can deliver a tailored and effective solution.

One significant assumption is that the hotel is situated near a beach. This assumption serves as a starting point for incorporating features and functionalities that capitalize on the hotel's proximity to a popular tourist attraction. As a result, the system includes specific functionalities related to beach events and activities. This may involve providing information on local beach events, such as concerts, festivals, or sports tournaments, allowing guests to stay updated and make informed decisions about their leisure plans. Additionally, the system may offer a feature that allows guests to check the availability of beach umbrellas, enabling them to reserve them in advance or receive real-time updates on their availability. These beach-related features enhance the overall guest experience and contribute to the hotel's unique selling proposition.

While incorporating location-specific features, the system avoids making assumptions regarding user demographics such as age, gender, or personal backgrounds. Instead, the focus is on creating an inclusive and user-friendly experience that caters to a broad range of users. The system aims to provide an intuitive interface and seamless functionality that can be easily accessed and utilized by individuals from various backgrounds and age groups. By adopting a user-centered design approach, the system prioritizes usability, accessibility, and ease of navigation, ensuring that all users, regardless of their demographic characteristics, can comfortably interact with the system and avail themselves of its features.

By making these assumptions and considering the specific context of the hotel, the Hotel Management System can provide a tailored and effective solution that aligns with the hotel's objectives and enhances the guest experience. The system's focus on location-specific features and its commitment to inclusivity and usability contribute to its overall value proposition and potential for success in the competitive hotel industry.

**2.4 Constraints**

In addition to the assumptions, the Hotel Management System also operates within certain constraints. One significant constraint is the need to consider the median wealth around the location of the hotel. Understanding the economic context allows the hotel management to determine appropriate pricing strategies, room rates, and overall financial considerations. The system should be designed to strike a balance between profitability and affordability, taking into account the target customer base and the surrounding economic conditions.

Another constraint to consider is the availability and reliability of internet connectivity. As the system is web-based, it relies on stable internet access for users to access its features and functionalities. Adequate measures should be in place to handle intermittent or limited internet connectivity to ensure uninterrupted service and minimize disruptions for both clients and hotel staff.

**2.5 Dependencies**

The Hotel Management System relies on various dependencies that contribute to its functionality and success. One key dependency is geolocation services. Integrating geolocation data enables the system to provide location-specific information and services, such as nearby attractions, restaurants, and transportation options. It enhances the overall user experience by offering personalized recommendations and tailored services based on the user's current location.

Weather data is another important dependency for the system. Integrating real-time weather information enables clients to make informed decisions regarding their activities and plans during their stay. For instance, clients can check weather forecasts to decide whether to engage in outdoor activities or opt for indoor amenities offered by the hotel.

Considering the median wealth in the surrounding area also plays a crucial role in the system's design and functionality. This dependency helps determine the pricing strategies, room rates, and promotional offers that align with the economic profile of the target market. It allows the hotel management to tailor their services to meet the expectations and budget constraints of potential guests.

Furthermore, the Hotel Management System may have dependencies on third-party services and APIs for additional features and integrations. For instance, it may rely on payment gateways to process online transactions securely or utilize external services for data backup and storage, ensuring data integrity and disaster recovery capabilities.

These dependencies, along with the identified constraints, must be carefully considered during the development and implementation of the Hotel Management System to create a robust, efficient, and user-friendly solution that aligns with the hotel's specific context and requirements.

1. **Requirements**

* System requirements in enough detail for designers to design a system satisfying the requirements and testers to verify that the system satisfies requirements.
* Organize these requirements in a way that works best for your project. See Appendix DAppendix D, Organizing the Requirements for different ways to organize these requirements.
* Describe every input into the system, every output from the system, and every function performed by the system in response to an input or in support of an output. (Specify what functions are to be performed on what data to produce what results at what location for whom.)
* Each requirement should be numbered (or uniquely identifiable) and prioritized.

See the sample requirements in Functional Requirements, and System Interface/Integration, as well as these example priority definitions:

**Priority Definitions**

The following definitions are intended as a guideline to prioritize requirements.

* Priority 1 – The requirement is a “must have” as outlined by policy/law
* Priority 2 – The requirement is needed for improved processing, and the fulfillment of the requirement will create immediate benefits
* Priority 3 – The requirement is a “nice to have” which may include new functionality

It may be helpful to phrase the requirement in terms of its priority, e.g., "The value of the employee status sent to DIS **must be** either A or I" or "It **would be nice** if the application warned the user that the expiration date was 3 business days away". Another approach would be to group requirements by priority category.

* A good requirement is:
* Correct
* Unambiguous (all statements have exactly one interpretation)
* Complete (where TBDs are absolutely necessary, document why the information is unknown, who is responsible for resolution, and the deadline)
* Consistent
* Ranked for importance and/or stability
* Verifiable (avoid soft descriptions like “works well”, “is user friendly”; use concrete terms and specify measurable quantities)
* Modifiable (evolve the Requirements Specification only via a formal change process, preserving a complete audit trail of changes)
* Does not specify any particular design
* Traceable (cross-reference with source documents and spawned documents).

## ***Functional Requirements***

In the example below, the requirement numbering has a scheme - BR\_LR\_0## (BR for Business Requirement, LR for Labor Relations). For small projects simply BR-## would suffice. Keep in mind that if no prefix is used, the traceability matrix may be difficult to create (e.g., no differentiation between '02' as a business requirement vs. a test case)

The following table is an example format for requirements. Choose whatever format works best for your project.

For Example:

| **Req#** | **Requirement** | **Comments** | **Priority** | **Date Rvwd** | **SME Reviewed / Approved** |
| --- | --- | --- | --- | --- | --- |
| LMS\_BR\_01 | Allow users (clients, workers, managers, administrators) to register and create accounts with appropriate roles and access levels. | Business Process = “User Registration” | 1 | 12/05/22 | Jurgen Halili |
| LMS\_BR\_02 | Implement a secure login mechanism to authenticate users and control access to various system functionalities. | Business Process = “User Authentication” | 1 | 12/05/22 | Jurgen Halili |
| LMS\_BR\_03 | Provide functionality for clients to check the availability of rooms, view real-time prices, and make reservations. | Business Process = “Room Availability and Booking” | 1 | 12/05/22 | Jurgen Halili |
| LMS\_BR\_04 | Allow clients to request room service directly through the app, which notifies the workers responsible for the requested task. | Business Process = “Room Service” | 3 | 12/05/22 | Jurgen Halili |
| LMS\_BR\_05 | Enable clients to provide their thoughts and reviews on the hotel, workers, and managers. | Business Process = “Reviews and Feedback” | 2 | 12/05/22 | Jurgen Halili |
| LMS\_BR\_06 | Implement a notification system to alert workers about rooms requiring service/cleaning. | Business Process = “Worker Notifications” | 3 | 12/05/22 | Jurgen Halili |
| LMS\_BR\_07 | Provide managers with a dashboard to view room availability, prices, worker performance, and manage orders. | Business Process = “Manager Dashboard” | 1 | 12/05/22 | Jurgen Halili |
| LMS\_BR\_08 | Allow managers to create and assign orders/tasks to workers, specifying the room number and required tasks. | Business Process = “Order Management” | 2 | 12/05/22 | Jurgen Halili |
| LMS\_BR\_09 | Enable managers to update room prices and availability in real-time. | Business Process = “Price and Availability Management” | 2 | 12/05/22 | Jurgen Halili |
| LMS\_BR\_10 | Provide administrators with extensive control over the website and app, allowing them to update content, manage orders, and rearrange prices and accommodations. | Business Process = “Administrator Privileges” | 1 | 12/05/22 | Jurgen Halili |
| LMS\_BR\_11 | Allow guests to check-in and check-out smoothly, providing necessary details and generating invoices.  Reservation Management | Business Process = “Guest Check-in and Check-out” | 1 | 19/05/22 | Jurgen Halili |
| LMS\_BR\_12 | Enable the hotel staff to manage and modify guest reservations, including room upgrades, date changes, and cancellations. | Business Process = “Reservation Management” | 1 | 19/05/22 | Jurgen Halili |
| LMS\_BR\_13 | Provide a system for managing housekeeping tasks, such as assigning and tracking room cleaning schedules. | Business Process =” Housekeeping Management” | 1 | 19/05/22 | Jurgen Halili |
| LMS\_BR\_14 | Maintain detailed guest profiles, including personal information, preferences, and past stay history for personalized service. | Business Process = “Guest Profile Management” | 1 | 19/05/22 | Jurgen Halili |
| LMS\_BR\_15 | Generate accurate bills and invoices for guest services, including room charges, additional amenities, and taxes. | Business Process = “Billing and Invoicing” | 1 | 19/05/22 | Jurgen Halili |
| LMS\_BR\_16 | Streamline front desk operations, including guest registration, key management, and handling guest inquiries. | Business Process = “Front Desk Operations” | 1 | 19/05/22 | Jurgen Halili |
| LMS\_BR\_17 | Track and manage room maintenance tasks, ensuring timely repairs and maintenance to maintain guest satisfaction. | Business Process = “Room Maintenance” | 1 | 19/05/22 | Jurgen Halili |
| LMS\_BR\_18 | Provide tools for managing events and conferences held at the hotel, including room booking, catering, and audio-visual requirements. | Business Process = Event and Conference Management” | 2 | 19/05/22 | Jurgen Halili |
| LMS\_BR\_19 | Integrate with a loyalty program to track and reward guest loyalty, offering exclusive benefits and discounts. | Business Process = “Loyalty Program Integration” | 2 | 19/05/22 | Jurgen Halili |
| LMS\_BR\_20 | Monitor and manage inventory levels of hotel supplies, including linens, toiletries, and other amenities, to ensure availability. | Business Process = “Inventory Management” | 2 | 19/05/22 | Jurgen Halili |
| LMS\_BR\_21 | Generate reports and analytics on various aspects of the hotel's performance, including occupancy rates, revenue, and guest feedback. | Business Process = “Analytics and Reporting” | 2 | 19/05/22 | Jurgen Halili |
| LMS\_BR\_22 | Provide tools for managing hotel staff, including shift scheduling, task assignment, and performance tracking. | Business Process = “Staff Management” | 2 | 19/05/22 | Jurgen Halili |
| LMS\_BR\_23 | Enable online booking functionality through the hotel's website and integrate with online travel agencies to manage room availability and rates. | Business Process = “Online Booking and Channel Management” | 2 | 19/05/22 | Endi Shima |
| LMS\_BR\_24 | Support the booking and management of group reservations, including room allocation, group billing, and reporting. | Business Process = “Group Reservations ” | 2 | 19/05/22 | Endi Shima |
| LMS\_BR\_25 | Allow guests to interact with the system in their preferred language and display rates and invoices in different currencies. | Business Process = “Multi-language and Multi-currency Support” | 3 | 19/05/22 | Endi Shima |
| LMS\_BR\_26 | Develop a mobile application for guests to easily access hotel services, make reservations, and receive notifications. | Business Process = “Mobile Application” | 3 | 19/05/22 | Endi Shima |
| LMS\_BR\_27 | Integrate the hotel management system with the POS system to track and manage payments for additional services like restaurants and spas. | Business Process = “Integration with Point of Sale (POS) System” | 3 | 19/05/22 | Endi Shima |
| LMS\_BR\_28 | Implement robust security measures and access control to safeguard guest information and ensure system integrity. | Business Process = “Security and Access Control ” | 3 | 19/05/22 | Endi Shima |
| LMS\_BR\_29 | Integrate with third-party services such as payment gateways, online review platforms, and online travel agencies for seamless operations. | Business Process = “Third-party API Integration” | 3 | 19/05/22 | Endi Shima |
| LMS\_BR\_30 | Provide a platform for guests to submit feedback and complete surveys, helping the hotel management understand and improve guest satisfaction. | Business Process = “Online Guest Feedback and Surveys” | 3 | 19/05/22 | Endi Shima |
| LMS\_BR\_31 | Enable the management and tracking of room amenities, such as minibar items, in-room entertainment, and special requests, ensuring prompt restocking and fulfillment of guest preferences. | Business Process = “Room Amenities Management ” | 2 | 19/05/22 | Endi Shima |

## 

**3.2 Non-Functional Requirements**

**3.2.1 Product Requirements**

**3.2.1.1 User Interface Requirements**

The user interface of the Hotel Management System should be designed to be easy to use and web-based. It should provide a seamless and intuitive experience for all users. Here are the specific requirements for the user interface:

* Layout: The layout should be well-structured, with a clear and consistent design throughout the system. The information should be organized logically, making it easy for users to find what they are looking for. The layout should be responsive, adapting to different screen sizes and devices.
* Navigation: The navigation should be user-friendly and intuitive, allowing users to move through the system effortlessly. A clear and accessible menu or navigation bar should be provided, enabling users to access different functionalities and sections of the system. The navigation should follow standard conventions, ensuring familiarity and ease of use for users.
* Visual Design: The visual design should be visually appealing, reflecting the branding and image of the hotel. The color scheme should be well-chosen, using colors that are visually pleasing and consistent with the hotel's branding guidelines. The typography should be legible, with appropriate font styles and sizes for easy reading. The use of icons and visual elements should enhance the user experience, providing visual cues and aiding in understanding the system's functionalities.
* Ease of Use: The user interface should be designed with simplicity and clarity in mind, minimizing complexity and cognitive load for users. User interactions should be intuitive, requiring minimal effort and providing clear feedback. The system should utilize appropriate input controls, such as dropdowns, checkboxes, and radio buttons, to facilitate user input and selection.
* Accessibility: The user interface should be accessible to users with special needs, complying with accessibility guidelines and standards (e.g., Web Content Accessibility Guidelines - WCAG). Considerations should be given to provide alternative text for images, support for screen readers, and keyboard navigation options.

**3.2.1.2 Usability**

Usability is a crucial aspect of the Hotel Management System to ensure that the system is intuitive and user-friendly. Consider the following guidelines and requirements to enhance usability:

* Intuitive System: The system should be designed with a user-centric approach, considering the needs and expectations of the users. The user interface should be self-explanatory, requiring minimal training for users to navigate and perform tasks.

Common user actions should be easily discoverable, reducing the learning curve for new users. Consistent terminology, labels, and icons should be used throughout the system, promoting familiarity and reducing confusion.

* Clear and Concise User Instructions: Provide clear instructions and guidelines within the system to assist users in performing various tasks. Use concise and easy-to-understand language in instructions, avoiding jargon or technical terms whenever possible.

Incorporate tooltips or contextual help to provide additional information or explanations for specific functionalities.

* Error Prevention and Handling: Implement measures to prevent errors and minimize the impact of user mistakes. Use validation techniques to ensure data integrity and accuracy, providing informative error messages when input errors occur.

Allow users to undo actions and provide a clear pathway to correct mistakes without losing any unsaved data.

* Feedback and Response: Provide immediate feedback to users for their actions, such as displaying loading indicators of progress bars during time-consuming processes.

Use meaningful messages to inform users about the outcome of their actions, ensuring they understand the system's response.

Consider using visual cues, such as success or error messages, to communicate the result of user interactions.

* Accessibility Requirements: Ensure that the system meets accessibility standards to accommodate users with special needs.

Comply with the Web Content Accessibility Guidelines (WCAG) to provide accessible features for individuals with disabilities.

Support keyboard navigation and provide alternative input methods to accommodate users who cannot use a mouse.

Use appropriate color contrast ratios to ensure readability for users with visual impairments.

Provide text alternatives for images and multimedia content to aid users relying on screen readers.

**3.2.1.3 Efficiency**

**3.2.1.3.1 Performance Requirements**

* Performance requirements focus on the response time, throughput, and resource utilization expectations of the Hotel Management System. Consider the following aspects to ensure optimal performance:
* Response Time: The system should aim to provide fast response times to user actions. The response time is the time taken for the system to process a user request and provide a result. The response time should be kept as low as possible to ensure a seamless user experience.
* Throughput: The system should be capable of handling a significant number of concurrent users and processing their requests efficiently. Throughput refers to the number of user requests that the system can handle in a given time period. It should be designed to accommodate high traffic loads without compromising performance.
* Resource Utilization: The system should be optimized to make efficient use of system resources, such as CPU, memory, and disk space. It should aim to minimize resource consumption and avoid unnecessary bottlenecks or performance degradation due to resource limitations.
* Performance Benchmarks and Targets: Performance benchmarks should be established to measure and evaluate the system's performance. These benchmarks can include metrics such as response time, throughput, and resource utilization. Targets should be set to define the desired performance levels. The system should be designed and optimized to achieve these targets, aiming for the fastest possible response times and optimal resource utilization.
* Load Testing: Load testing should be performed to simulate real-world usage scenarios and evaluate the system's performance under different load conditions. This helps identify any performance bottlenecks, scalability issues, or areas for optimization.

**3.2.1.3.2 Responsiveness**

Responsiveness refers to the system's ability to quickly and efficiently respond to user interactions. The following requirements ensure that the Hotel Management System provides a responsive user experience:

* Minimum Acceptable Response Times: The system should strive to have response times that are as fast as possible. A specific requirement is to ensure that the system maintains response times below a certain threshold. In this case, the minimum acceptable response time is set to be less than 1.5 seconds for all user interactions. This ensures that users receive prompt feedback and can perform tasks without experiencing significant delays.
* Feedback Indicators: The system should provide immediate feedback to users after their actions, indicating that their request has been received and is being processed. This can include visual cues such as loading spinners, progress bars, or success/error messages. These indicators help users understand that the system is actively working on their request, even if it takes more time to complete.
* Asynchronous Processing: To enhance responsiveness, the system should utilize asynchronous processing whenever possible. Long-running tasks or operations that require significant processing time should be handled asynchronously, allowing users to continue interacting with the system without waiting for the task to complete. This approach ensures that the user interface remains responsive and doesn't become unresponsive or freeze during resource-intensive operations.
* Optimized Network Communication: The system should aim to minimize network latency and optimize data transfer between the client and server. Techniques such as caching, compression, and efficient data exchange formats can be employed to reduce the time required for data transmission, resulting in improved responsiveness.

**3.2.1.4 Dependability**

Dependability requirements focus on ensuring the reliability and availability of the Hotel Management System. The following requirement addresses backup and recovery mechanisms using a non-local database:

* Backup and Recovery Mechanisms: The system should implement robust backup and recovery mechanisms to ensure the integrity and availability of data. Instead of relying solely on local backups, the system should utilize a non-local database for storing data backups. This approach provides an additional layer of protection against data loss in case of hardware failures, natural disasters, or other unforeseen events.
* Regular Data Backups: The system should perform regular backups of critical data stored in the non-local database. The backup frequency should be determined based on the importance and volatility of the data. Scheduled backups should be performed at appropriate intervals to minimize the risk of data loss.
* Data Recovery: In the event of data loss or corruption, the system should have mechanisms in place to facilitate data recovery. This includes procedures for restoring data from backups stored in the non-local database. The recovery process should be well-documented and tested to ensure its effectiveness and efficiency.
* Redundancy and Fault Tolerance: To enhance system dependability, the non-local database should be designed with redundancy and fault tolerance in mind. This can involve replication of data across multiple servers or using a distributed database system to ensure high availability and data durability.
* Disaster Recovery Planning: The system should have a comprehensive disaster recovery plan in place to address potential disruptions and ensure business continuity. This plan should outline the steps and procedures to be followed in the event of a major system failure, including the recovery of data from the non-local database.

**3.2.1.5 Security**

Security requirements are crucial for protecting sensitive data and ensuring the integrity of the Hotel Management System. The following requirements address data protection measures and compliance with industry regulations and standards:

* Data Protection Measures: The system should implement robust data protection measures to safeguard sensitive information. This includes the use of encryption techniques to secure data both in transit and at rest. Encryption should be applied to sensitive data such as user credentials, financial transactions, and personally identifiable information (PII). Additionally, secure storage mechanisms should be employed to prevent unauthorized access or data breaches.
* Compliance with Industry Regulations and Standards: The system should adhere to relevant industry regulations and standards related to data security, privacy, and confidentiality. This may include compliance with regulations such as the General Data Protection Regulation (GDPR) or Payment Card Industry Data Security Standard (PCI DSS). The system should implement necessary controls, policies, and procedures to ensure compliance and protect user data.
* User Authentication and Authorization: The system should employ robust user authentication mechanisms to verify the identity of users accessing the system. This can include the use of strong passwords, multi-factor authentication (MFA), or biometric authentication where applicable. Authorization mechanisms should be implemented to control access to different system functionalities and data based on user roles and privileges.
* Secure Communication: The system should utilize secure communication protocols, such as HTTPS, to encrypt data transmission over the network. This ensures that sensitive information exchanged between the client and server remains protected from unauthorized interception or tampering.
* Security Auditing and Monitoring: The system should incorporate logging, auditing, and monitoring mechanisms to track and analyze system activities. This helps in detecting and responding to security incidents, unauthorized access attempts, or suspicious behavior. Regular security audits should be conducted to assess the system's security posture and identify potential vulnerabilities.

**3.2.2 Organizational Requirements**

**3.2.2.1 Environmental Requirements**

Environmental requirements focus on the system deployment, hardware/environment specifications, and compatibility with different hardware devices. Consider the following aspects to ensure proper functioning in various environments:

* System Deployment: The Hotel Management System should be designed for easy deployment in different environments, including on-premises or cloud-based hosting. Clear guidelines and documentation should be provided to assist system administrators in the installation, configuration, and deployment process.
* Hardware/Environment Specifications: The system should have defined hardware and environment specifications to ensure optimal performance. This includes specifying the minimum hardware requirements, such as processor, memory, and storage capacity, necessary for running the system efficiently. Additionally, any specific software dependencies or compatibility requirements should be documented.
* Compatibility with Different Hardware Devices: The system should be compatible with a wide range of hardware devices commonly used by users. This includes desktop computers, laptops, tablets, and smartphones. The user interface and functionality should be responsive and adaptable to different screen sizes, resolutions, and input methods to provide a consistent user experience across devices.
* Operating System Compatibility: The Hotel Management System should be compatible with various operating systems, such as Windows, macOS, Linux, iOS, and Android. The system should be thoroughly tested on different operating system versions to ensure compatibility and functionality.
* Browser Compatibility: The system's web-based interface should be compatible with popular web browsers, including but not limited to Google Chrome, Mozilla Firefox, Microsoft Edge, and Safari. It should be tested and optimized to work seamlessly across different browsers and versions.
* Network Connectivity: The system should be designed to operate effectively in different network environments, such as wired and wireless networks. It should consider potential network limitations, bandwidth constraints, and intermittent connectivity scenarios to ensure reliable operation and data synchronization.

**3.2.2.2 Operational Requirements**

Operational requirements focus on staff training and support, as well as system maintenance and upgrade procedures. Consider the following aspects to ensure smooth operation and maintenance of the Hotel Management System:

* Staff Training and Support Requirements: Adequate training should be provided to staff members who will be using the system, such as administrators, managers, workers, and support personnel. Training sessions should cover the system's functionalities, user interface, data entry procedures, and troubleshooting techniques. Clear and comprehensive documentation, including user manuals and training materials, should be created to assist staff in understanding and using the system effectively. Ongoing support, such as a helpdesk or technical support team, should be available to address any questions or issues that may arise during system operation.
* System Maintenance: The system should have defined maintenance procedures to ensure its smooth operation. This includes regular monitoring of system performance, database optimization, and data backup procedures. System administrators should schedule routine maintenance activities to ensure the system's stability and reliability. Maintenance activities should be documented and adhered to consistently.
* Upgrade Procedures: System upgrades and updates should be handled by administrators. They should be responsible for evaluating and implementing system upgrades, including updates to the software and hardware components. Procedures for testing upgrades in a separate environment, ensuring compatibility, and minimizing downtime should be established. Clear documentation should be provided to guide administrators through the upgrade process, including any necessary backup and rollback procedures.
* Change Management: A change management process should be in place to handle any modifications or enhancements to the system. This includes assessing the impact of proposed changes, conducting testing and validation, and obtaining necessary approvals before implementing changes. Change logs and documentation should be maintained to track system modifications and ensure accountability.
* Incident and Problem Management: Processes for incident and problem management should be established to address any system failures, errors, or user-reported issues. These processes should include procedures for logging incidents, conducting root cause analysis, and implementing corrective actions. Incident resolution times and problem resolution targets should be defined to ensure timely response and minimize system disruptions.

**3.2.2.3 Development Requirements**

Development requirements focus on the programming languages used, as well as coding standards and guidelines for the Hotel Management System. Consider the following aspects to ensure efficient and standardized development practices:

* Programming Languages: The Hotel Management System will be developed using a combination of programming languages to achieve its functionality. The identified programming languages for the system development include:
  + JavaScript: Used for client-side scripting and enhancing the user interface interactivity.
  + HTML: Used for structuring the web-based user interface and presenting content.
  + CSS: Used for styling and visually formatting the web-based user interface.
  + PHP: Used for server-side scripting and implementing server logic.
  + AJAX: Used for asynchronous communication between the client and server to enhance system responsiveness.
  + XML: Used for data exchange and storage of structured information.
  + Java: Used for server-side logic and back-end processing.
* Coding Standards and Guidelines: It is essential to establish coding standards and guidelines to ensure consistency, readability, and maintainability of the system's codebase. The following coding standards and guidelines should be followed during development:
* Naming Conventions: Consistent naming conventions for variables, functions, classes, and other code elements should be adopted. This promotes clarity and makes the code easier to understand.
* Indentation and Formatting: Clear indentation and formatting rules should be followed to enhance code readability. Consistent use of spaces, tabs, line breaks, and braces helps in maintaining a clean and organized code structure.
* Commenting: Adequate and meaningful comments should be included to explain the code's purpose, logic, and any complex algorithms or business rules. This facilitates better understanding, debugging, and future maintenance of the codebase.
* Modularity and Reusability: Code should be organized into modules or components with well-defined responsibilities. This promotes code reusability, modularity, and easier maintenance. Encouraging the use of functions, classes, and libraries helps in achieving this goal.
* Error Handling and Exception Handling: Appropriate error handling and exception handling mechanisms should be implemented to ensure robustness and prevent unexpected system failures. Error messages and logging should provide meaningful information for debugging and troubleshooting purposes.
* Security Best Practices: The development should follow security best practices to mitigate vulnerabilities and protect against common web application attacks, such as cross-site scripting (XSS) and SQL injection. This includes input validation, secure data handling, and appropriate access control mechanisms.

**3.2.3 External Requirements**

**3.2.3.1 Regulatory Requirements**

Regulatory requirements are crucial for ensuring that the Hotel Management System complies with industry-specific regulations and standards. Consider the following aspects to meet the regulatory requirements:

* Identify Industry-Specific Regulations: Research and identify the industry-specific regulations and standards applicable to the hotel management sector. This may include regulations related to data privacy, security, financial transactions, customer rights, and any other relevant legal requirements.
* Compliance Assessment: Evaluate the Hotel Management System against the identified regulations and standards to ensure compliance. This involves reviewing the system's functionalities, data handling processes, security measures, and privacy practices. Identify any gaps or areas where enhancements or modifications are needed to align with the regulations.
* Data Privacy and Protection: Ensure that the system adheres to data privacy regulations and protects personal and sensitive information of users. Implement measures such as consent management, data anonymization, secure storage, and access controls to safeguard user data. Comply with regulations such as the General Data Protection Regulation (GDPR) or any other applicable data protection laws in the respective jurisdiction.
* Financial Compliance: If the system handles financial transactions or sensitive financial information, ensure compliance with relevant regulations such as Payment Card Industry Data Security Standard (PCI DSS) or any other applicable financial regulations. Implement secure payment processing, encryption of financial data, and adherence to financial transaction auditing requirements.
* Customer Rights and Consumer Protection: Incorporate features and functionalities that protect customer rights and provide transparency. This may include the ability for customers to manage their personal data, control privacy preferences, and access their booking history. Comply with regulations related to cancellation policies, refund processes, and customer support.
* Legal and Contractual Compliance: Ensure that the system complies with all applicable laws, contracts, and agreements. This includes intellectual property rights, licensing agreements, terms of service, and any other legal obligations. Review and update the system's terms and conditions to align with legal requirements.
* Documentation and Record-Keeping: Maintain proper documentation and records to demonstrate compliance with regulatory requirements. This includes documenting the system's security measures, data handling processes, and any actions taken to address regulatory compliance. Keep records of audits, assessments, and any certifications obtained to showcase compliance efforts.

**3.2.3.2 Ethical Requirements**

Ethical requirements are essential to guide the proper and responsible use of customer data within the Hotel Management System. Consider the following guidelines for the ethical use of customer data:

* Data Privacy and Consent: Respect and protect the privacy of customer data by implementing appropriate measures. Obtain informed consent from customers regarding the collection, storage, and use of their personal information. Clearly communicate the purposes for which the data will be used and ensure that customer data is only accessed and processed in accordance with their consent.
* Data Security: Implement robust security measures to safeguard customer data from unauthorized access, disclosure, or misuse. This includes encryption of sensitive data, secure storage practices, regular security audits, and access controls. Minimize the retention of customer data to only what is necessary and securely dispose of data that is no longer needed.
* Transparency and Communication: Maintain transparency in how customer data is collected, used, and shared. Provide clear and easily understandable privacy policies and terms of service that outline the system's data practices. Regularly communicate any updates or changes to these policies and allow customers to exercise control over their data preferences.
* Anonymization and Aggregation: Anonymize and aggregate customer data whenever possible to protect individual privacy. Ensure that customer data is not personally identifiable when performing analysis or sharing data in aggregated form. Implement measures to prevent re-identification of anonymized data.
* Data Access and Control: Enable customers to access, review, and update their personal information within the system. Provide mechanisms for customers to easily modify their consent preferences, manage their data sharing options, and request the deletion of their data. Respect and honor customer data access and control rights.
* Fair and Non-discriminatory Use: Ensure that customer data is used in a fair and non-discriminatory manner. Avoid any unethical practices that may result in discrimination based on factors such as race, gender, ethnicity, or any other protected characteristics. Use customer data solely for the purposes disclosed and agreed upon with customers.
* Compliance with Applicable Laws and Regulations: Adhere to all relevant laws and regulations pertaining to data protection, privacy, and consumer rights. Stay up to date with changing legal requirements and ensure ongoing compliance with evolving ethical standards in data management and usage.

**3.2.3.3 Legislative Requirements**

**3.2.3.3.1 Accounting Requirements**

Accounting requirements involve the support for financial reporting and tracking within the Hotel Management System, as well as integration with accounting systems. Consider the following aspects to meet the accounting requirements:

* Financial Reporting and Tracking: The Hotel Management System should provide robust capabilities for financial reporting and tracking. This includes generating reports such as revenue summaries, occupancy rates, room rates, and other financial metrics. The system should accurately record and track financial transactions, such as room bookings, payments, refunds, and expenses.
* Integration with Accounting Systems: To ensure seamless financial management, the Hotel Management System should support integration with accounting systems. This integration enables the automatic transfer of financial data from the hotel management system to the accounting software used by the organization. This ensures accuracy and eliminates the need for manual data entry, reducing the chances of errors and improving efficiency.
* Chart of Accounts: The system should support the implementation of a chart of accounts specific to the hotel's financial structure. This allows for accurate categorization and classification of financial transactions according to the organization's accounting practices. The chart of accounts should align with standard accounting principles and facilitate financial reporting and analysis.
* General Ledger Management: The system should have capabilities for managing the general ledger, which is the central repository of financial transactions. This includes the ability to record journal entries, track accounts payable and receivable, reconcile accounts, and generate financial statements. The system should provide adequate controls and audit trails to ensure the integrity and accuracy of financial data.
* Audit Trail: An audit trail feature is essential to maintain a record of all financial activities within the system. This includes capturing details such as the user responsible for a transaction, the date and time of the transaction, and any modifications made. The audit trail provides transparency and accountability, facilitating financial audits and ensuring compliance with financial regulations.
* Financial Controls: The system should incorporate financial controls to prevent fraud and ensure proper financial management. This may include user access controls, segregation of duties, approval workflows for financial transactions, and other measures to mitigate the risk of unauthorized or fraudulent activities.
* Compliance with Financial Standards: Ensure that the system complies with applicable financial standards, such as Generally Accepted Accounting Principles (GAAP) or International Financial Reporting Standards (IFRS). This ensures that financial data and reports generated by the system adhere to recognized accounting practices and can be easily understood by stakeholders.

**3.2.3.3.2 Security Requirements**

Security requirements are crucial for protecting customer data and preventing data breaches within the Hotel Management System. Consider the following measures to ensure the security of customer data:

* Data Encryption: Implement robust encryption mechanisms to protect customer data both in transit and at rest. Use strong encryption algorithms to safeguard sensitive information, such as personal details, payment data, and passwords. Encryption ensures that even if data is intercepted or accessed without authorization, it remains unreadable and unusable.
* Access Control: Enforce strict access controls to prevent unauthorized access to customer data. Implement user authentication mechanisms, such as usernames and passwords, and consider adding additional layers of security, such as multi-factor authentication. Limit access privileges to only authorized personnel based on their roles and responsibilities.
* Secure Storage: Store customer data in secure and protected storage systems. Ensure that the storage infrastructure has appropriate security measures in place, including access controls, firewalls, intrusion detection systems, and regular security audits. Implement mechanisms to monitor and log any unauthorized access attempts.
* Regular Security Audits: Conduct regular security audits and assessments of the Hotel Management System to identify vulnerabilities and weaknesses. Perform penetration testing to simulate potential attacks and evaluate the system's resilience. Address any identified security issues promptly and implement necessary patches and updates to mitigate risks.
* Security Incident Response: Establish a comprehensive security incident response plan to handle potential security breaches or incidents. Define roles and responsibilities of personnel involved in the response process, establish communication protocols, and outline steps to contain and mitigate the impact of security incidents. Regularly test and refine the incident response plan to ensure its effectiveness.
* Secure Communication: Ensure that all communication channels within the system, such as APIs, web services, and data exchanges, are secured using protocols like HTTPS (HTTP Secure) to encrypt data in transit. Implement measures to prevent man-in-the-middle attacks and eavesdropping.
* Regular Data Backups: Implement a robust backup strategy to ensure the availability and integrity of customer data. Regularly back up the data to a secure off-site location or cloud-based storage. Test the backup and recovery processes periodically to verify their effectiveness.
* Security Training and Awareness: Provide regular security training and awareness programs to all personnel involved in the management and operation of the Hotel Management System. Educate employees about best practices for data security, password hygiene, phishing awareness, and social engineering prevention. Foster a security-conscious culture within the organization.
* Compliance with Data Protection Laws: Ensure compliance with relevant data protection laws, such as the General Data Protection Regulation (GDPR) or other applicable regulations based on the operating jurisdiction. Understand and adhere to legal requirements regarding the collection, storage, processing, and transfer of customer data.

**3.3 Domain Requirements**

Domain requirements refer to the specific requirements related to the hotel industry and the need to comply with industry best practices and standards. Consider the following aspects when addressing the domain requirements of the Hotel Management System:

* Room Management: The system should provide comprehensive features for efficient room management, including room availability tracking, booking and reservation management, room categorization, and inventory control. Ensure that the system supports different room types (e.g., single, double, suite) and can handle various pricing models (e.g., seasonal rates, discounts, packages).
* Guest Services: Incorporate functionalities that enhance guest services and overall customer experience. This may include features such as express check-in/check-out, personalized greetings and preferences, room service requests, special requests handling, and loyalty program integration. The system should facilitate seamless communication between guests and hotel staff.
* Billing and Invoicing: Implement robust billing and invoicing capabilities to accurately calculate and process guest charges. The system should handle different payment methods (e.g., cash, credit card, online payments) and provide clear and itemized invoices. Consider integrating with external payment gateways or accounting systems to streamline financial transactions.
* Reporting and Analytics: Provide comprehensive reporting and analytics capabilities to assist in decision-making and performance evaluation. The system should generate reports on occupancy rates, revenue, guest feedback, room service usage, and other key metrics. Consider offering customizable reports and data visualization tools to cater to specific reporting needs.
* Integration with External Systems: Ensure compatibility and seamless integration with external systems commonly used in the hotel industry. This may include property management systems (PMS), online travel agencies (OTA), channel managers, point-of-sale (POS) systems, and customer relationship management (CRM) software. Integration enables data synchronization, streamlines operations, and enhances overall efficiency.
* Compliance with Industry Standards: Adhere to industry best practices and standards relevant to the hotel industry. Stay updated with industry trends, technological advancements, and emerging standards. Consider certifications or compliance programs, such as ISO 27001 for information security management, to demonstrate adherence to internationally recognized standards.
* Scalability and Flexibility: Design the system to be scalable and adaptable to accommodate the changing needs of a hotel business. As hotels may vary in size and complexity, the system should support the growth and expansion of the hotel operations. Consider modular architecture and cloud-based infrastructure to facilitate scalability and flexibility.
* User Training and Support: Provide comprehensive user training and support materials to assist hotel staff in effectively utilizing the system. Offer user-friendly documentation, training sessions, and a dedicated support channel to address any queries or issues that may arise during system usage.

**4. Software Design**

**4.1 User Scenarios**

**Client User Scenarios**

1. Scenario: Searching for Available Rooms:
   * The client opens the hotel management system and navigates to the room reservation section.
   * They specify their desired check-in and check-out dates, number of guests, and any specific room preferences.
   * The system displays a list of available rooms that match the client's criteria, along with detailed information such as room type, amenities, and pricing.
   * The client compares the options, selects a room, and proceeds to book it securely through the system.
2. Scenario: Exploring new Offers and Real-Time prices:
   * The client logs into the hotel management system and visits the offers section.
   * They browse through the latest promotional offers, packages, and discounts available.
   * The system presents real-time prices and detailed information about each offer, including any restrictions or additional benefits.
   * The client selects a preferred offer, reviews the terms and conditions, and proceeds to book it or make an inquiry if needed.
3. Scenario : Purchasing bundles and making future reservations:
   * The client accesses their account and selects the bundles section in the hotel management system.
   * They explore different bundled services or amenities offered by the hotel, such as spa packages or dining experiences.
   * The system provides comprehensive details and pricing information for each bundle, including any customization options.
   * The client selects a desired bundle, reviews the inclusions, and proceeds to purchase it or makes a reservation for a future date.
4. Scenario: Requesting Room Service
   * The client logs into the hotel management system and accesses the room service section.
   * They browse through the available room service options, such as food and beverages or additional amenities.
   * The system allows the client to select desired items from a menu, specify any special instructions or dietary requirements, and place an order.
   * The order is received by the hotel staff, who fulfill the request and ensure timely delivery to the client's room.
5. Scenario: Providing Feedback and Reviews
   * After checking out, the client receives an email with a link to provide feedback and reviews on their hotel experience.
   * They click on the link, which directs them to the hotel management system's feedback section.
   * The system presents a structured form where the client rates different aspects of their stay, such as room cleanliness, staff service, and facilities.
   * The client also has the option to provide specific comments or suggestions to further improve the hotel's services.
6. Scenario: Accessing Concierge Services and Local Recommendations
   * The client accesses the hotel management system's concierge services section or contacts the concierge through the system.
   * They seek assistance in arranging transportation, booking local tours or activities, or getting recommendations for nearby attractions.
   * The system provides personalized suggestions, based on the client's preferences and interests, and assists in making necessary reservations or arrangements.
7. Scenario: Checking Loyalty Program Benefits and Redeeming Points
   * The client logs into their loyalty program account through the hotel management system.
   * They navigate to the loyalty program section, which displays their current points balance, membership tier, and available benefits.
   * The system allows the client to redeem points for room upgrades, complimentary services, or other exclusive perks directly through the system.

**Worker User Scenarios:**

1. Scenario: Checking Rooms Requiring Service
   * The worker logs into the hotel management system and accesses the worker dashboard.
   * They navigate to the room service section, where they can view a list of rooms that require cleaning, maintenance, or other services.
   * The system displays room numbers, specific service requests, and any additional notes provided by the managers or clients.
   * The worker selects a room from the list, acknowledges the task, and updates the status once the service is completed.
2. Scenario: Receiving Direct Notifications from Managers
   * The worker receives a notification on their mobile device through the hotel management system.
   * The notification contains a brief description of the task or request assigned to them by a manager.
   * The worker opens the notification, which directs them to the relevant section of the system for more details.
   * The system provides comprehensive information about the task, including priority, room number, and specific instructions from the manager.
   * The worker acknowledges the notification, proceeds to complete the task, and updates the status in real-time.
3. Scenario: Updating Room Status and Availability
   * The worker accesses the room management section in the hotel management system.
   * They view the current status and availability of rooms, including occupied, vacant, and reserved rooms.
   * The system allows the worker to update the status of rooms based on cleaning, maintenance, or readiness for check-in.
   * The worker marks rooms as clean and available for guests, ensuring accurate and up-to-date information in the system.
4. Scenario: Reporting Issues or Deficiencies
   * The worker encounters a problem or identifies a deficiency in a room or facility during their routine duties.
   * They log into the hotel management system and navigate to the issue reporting section.
   * The system provides a form where the worker can describe the issue in detail, attach relevant photos if necessary, and specify the location.
   * The worker submits the report, which is then received by the appropriate department or supervisor for resolution.
5. Scenario: Collaborating with Other Workers
   * The worker accesses the collaboration or communication feature within the hotel management system.
   * They communicate with other workers or departments to coordinate tasks, share information, or seek assistance.
   * The system provides a platform for messaging, sharing documents or photos, and maintaining a collaborative work environment.
   * The worker can engage in real-time discussions, exchange updates, and ensure effective teamwork and coordination.
6. Scenario: Requesting Support or Additional Resources
   * The worker encounters a situation that requires additional support or resources to complete a task.
   * They access the support or resource request feature in the hotel management system.
   * The system allows the worker to submit a request, specifying the nature of assistance required and the urgency.
   * The request is received by the relevant department or supervisor, who evaluates and provides the necessary support or resources.
7. Scenario: Providing Updates and Feedback to Managers
   * The worker accesses the feedback or performance review section within the hotel management system.
   * They provide regular updates on completed tasks, ongoing projects, or any challenges faced during their duties.
   * The system allows the worker to provide feedback or suggestions to improve work processes, facilities, or team collaboration.
   * The updates and feedback are received by managers, who can review, respond, and take necessary actions based on the information provided.

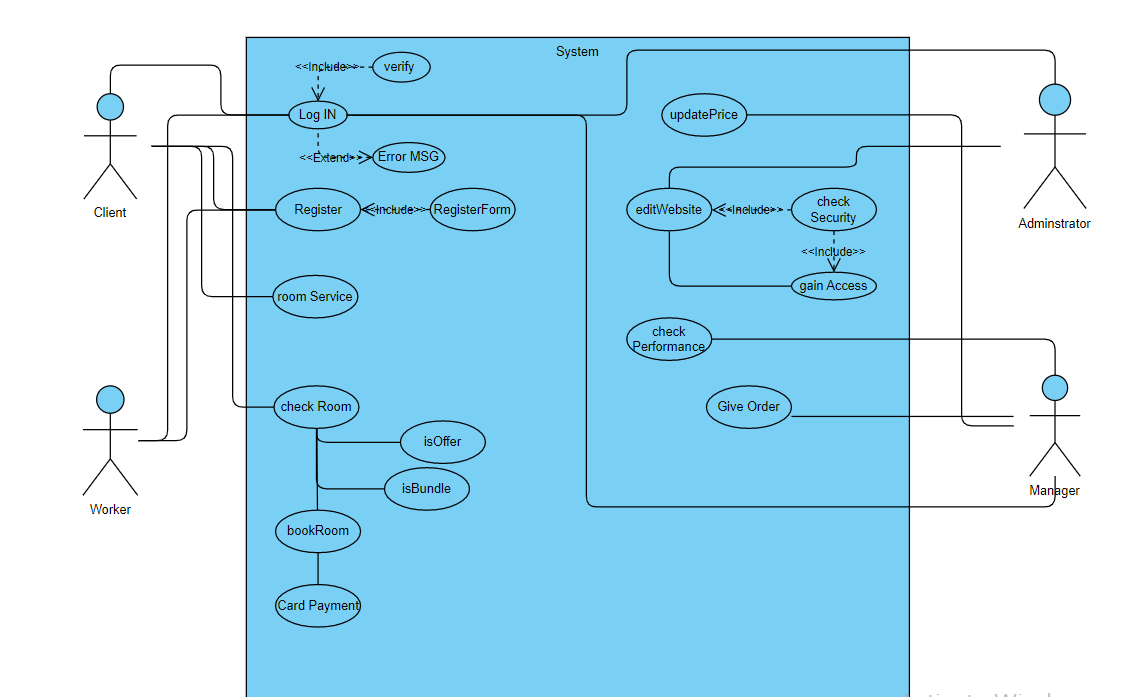
**Manager User Scenarios:**

1. Scenario: Viewing Room Availability and Pricing
   * The manager logs into the hotel management system and accesses the manager dashboard.
   * They navigate to the room management section, where they can view the availability and pricing of each room.
   * The system displays a comprehensive list of rooms, along with their current status (occupied, vacant, or reserved) and corresponding prices.
   * The manager can easily update the room availability and pricing based on demand, seasons, or other factors.
2. Scenario: Monitoring Worker Performances and Reviews
   * The manager accesses the performance management section in the hotel management system.
   * They can view detailed performance metrics and reviews for each worker, based on feedback from clients and other staff members.
   * The system provides a summary of key performance indicators, such as cleanliness ratings, responsiveness, and customer satisfaction scores.
   * The manager can analyze the performance data, identify areas for improvement, and provide constructive feedback or recognition to workers.
3. Scenario: Assigning Tasks and Providing Instructions to Workers
   * The manager accesses the task management section in the hotel management system.
   * They can create and assign specific tasks to individual workers or teams, based on operational requirements.
   * The system allows the manager to define task details, priorities, deadlines, and provide necessary instructions or attachments.
   * The assigned tasks are automatically sent to the respective workers, who can acknowledge, track progress, and update task status within the system.
4. Scenario: Updating Room Prices and Direct Availability
   * The manager navigates to the pricing and availability section within the hotel management system.
   * They can update the prices of rooms based on demand, seasons, promotions, or other factors.
   * The system provides a user-friendly interface to modify prices, set discounts, and update pricing rules.
   * The manager can also directly update the availability status of rooms, ensuring accurate and real-time information for clients and workers.
5. Scenario: Reviewing and Responding to Guest Feedback
   * The manager accesses the guest feedback section in the hotel management system.
   * They can view feedback and reviews submitted by clients regarding their stay, services, or other aspects of the hotel.
   * The system provides a consolidated view of feedback, allowing the manager to analyze trends, identify areas for improvement, and respond to guest concerns.
   * The manager can take appropriate actions, such as resolving issues, offering compensation, or implementing changes based on the feedback received.
6. Scenario: Generating Reports and Analytics
   * The manager utilizes the reporting and analytics features within the hotel management system.
   * They can generate various reports, such as occupancy rates, revenue summaries, staff performance, and guest satisfaction.
   * The system provides customizable reporting templates and advanced analytics tools to extract valuable insights from the data.
   * The manager can use these reports to make informed decisions, identify opportunities for growth, and track the overall performance of the hotel.
7. Scenario: Collaborating with Administrators and Other Managers
   * The manager engages in collaborative activities through the hotel management system.
   * They can communicate with administrators and other managers to discuss strategies, share information, and seek guidance.
   * The system facilitates real-time messaging, document sharing, and virtual meetings to foster effective collaboration and decision-making.

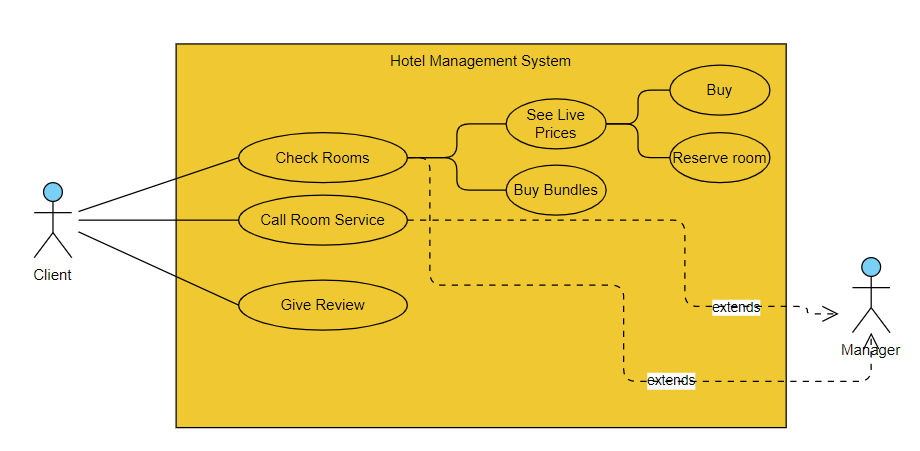
**Administrator User Scenarios**

1. Scenario: Managing System Settings and Configuration
   * The administrator logs into the hotel management system with their privileged access.
   * They navigate to the system settings section, where they can configure various aspects of the system.
   * The administrator can customize general settings, such as hotel information, contact details, and branding elements.
   * They can also manage user roles and permissions, granting or revoking access rights to different functionalities of the system.
2. Scenario: Updating Website Content and Layout
   * The administrator accesses the content management system (CMS) integrated within the hotel management system.
   * They can easily update and modify website content, including text, images, promotional banners, and special offers.
   * The CMS provides a user-friendly interface that allows the administrator to arrange and format content, ensuring a visually appealing and engaging website.
   * The administrator can preview the changes before publishing them to the live website.
3. Scenario: Managing Database and Backup/Recovery
   * The administrator ensures the integrity and security of the hotel management system's database.
   * They perform routine database maintenance tasks, such as optimizing performance, managing backups, and ensuring data consistency.
   * The administrator sets up regular backup schedules to safeguard critical data in case of system failures or data loss incidents.
   * They also establish recovery mechanisms to restore the system and data to a previous state if necessary.
4. Scenario: System Upgrades and Software Updates
   * The administrator oversees system upgrades and software updates for the hotel management system.
   * They evaluate the compatibility and impact of new releases, considering the system's stability and user experience.
   * The administrator plans and coordinates the deployment of upgrades, ensuring minimal disruption to the system's availability and functionality.
   * They may conduct testing and quality assurance processes before implementing major upgrades to ensure a smooth transition.
5. Scenario: Monitoring System Performance and Security
   * The administrator actively monitors the performance and security of the hotel management system.
   * They utilize monitoring tools and dashboards to assess system health, including resource utilization, response times, and error logs.
   * The administrator promptly investigates and addresses any performance issues or security vulnerabilities identified.
   * They may implement security measures such as firewall configurations, encryption protocols, and access controls to protect sensitive data.
6. Scenario: Providing Technical Support and Training
   * The administrator acts as a point of contact for technical support and training related to the hotel management system.
   * They assist users, including managers, workers, and clients, in troubleshooting system-related issues, answering queries, and providing guidance.
   * The administrator conducts training sessions to onboard new users, familiarizing them with system functionalities, workflows, and best practices.
   * They also create and maintain user documentation, tutorials, and FAQs to facilitate self-help resources for users.
7. Scenario: Compliance with Industry Regulations and Standards
   * The administrator ensures that the hotel management system complies with industry-specific regulations and standards.
   * They stay updated with legal requirements, such as data privacy regulations, accounting standards, and security protocols.
   * The administrator implements measures to safeguard customer data, secure financial transactions, and maintain regulatory compliance.
   * They may conduct periodic audits and assessments to verify adherence to industry regulations and address any compliance gaps.

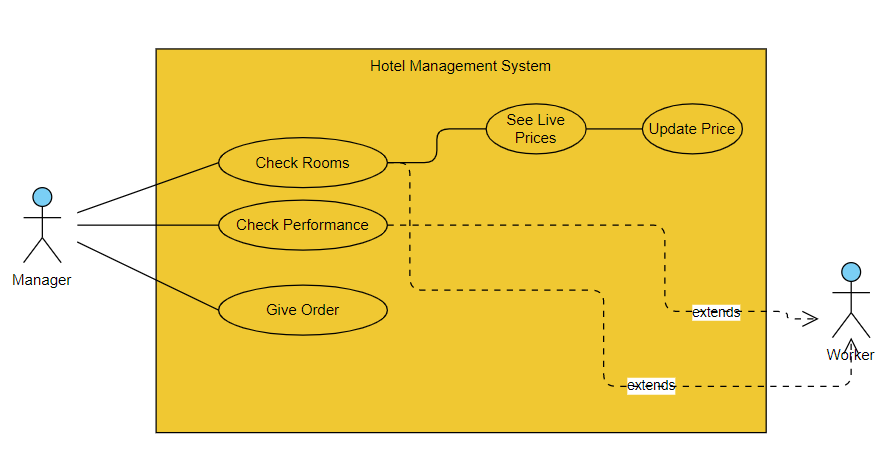
**4.2 Use Cases**

General Use Case:

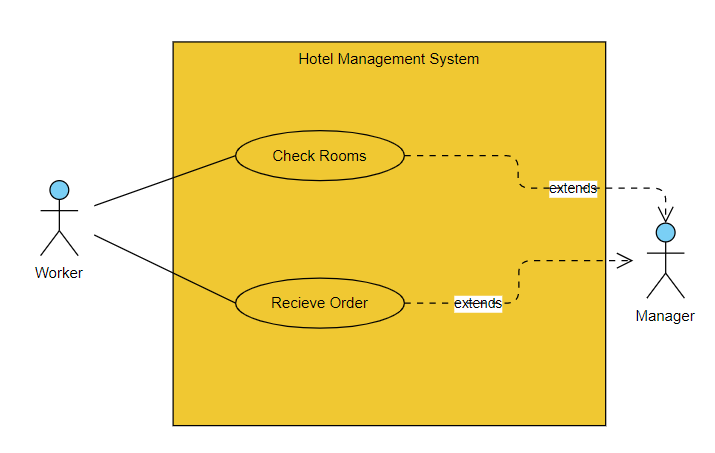
Client Use Case:



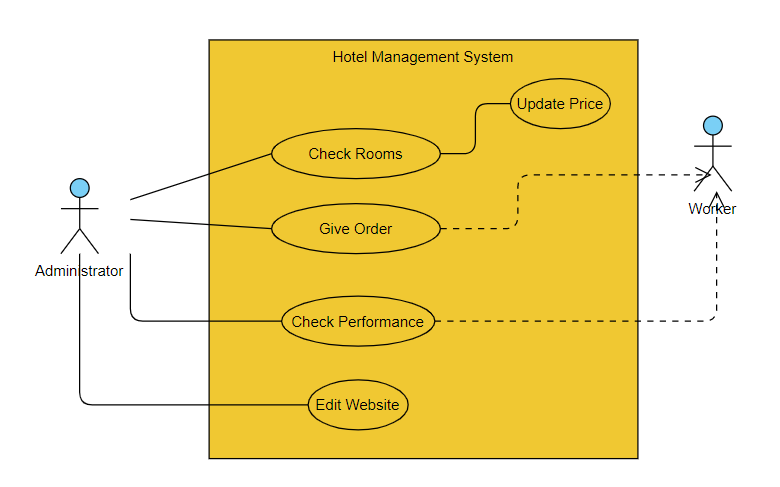
Manager Use Case:



Worker Use Case:

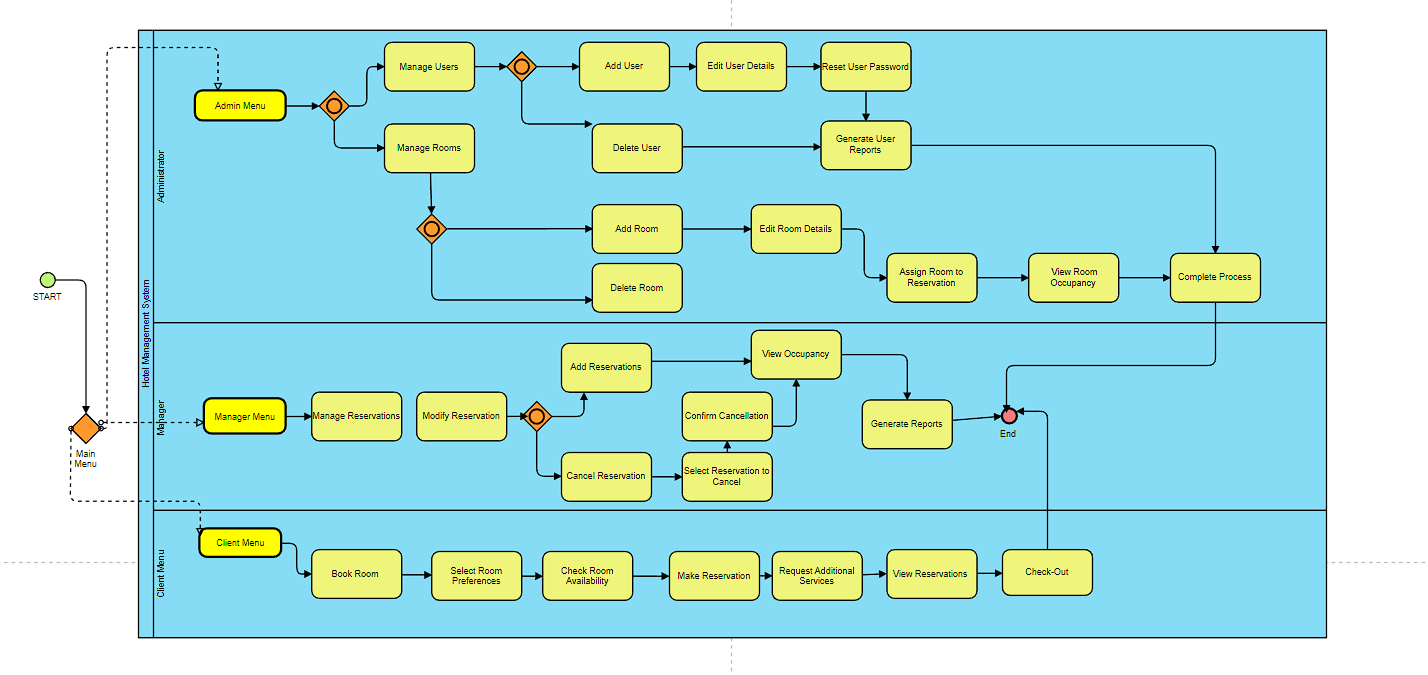


Administrator Use Case:

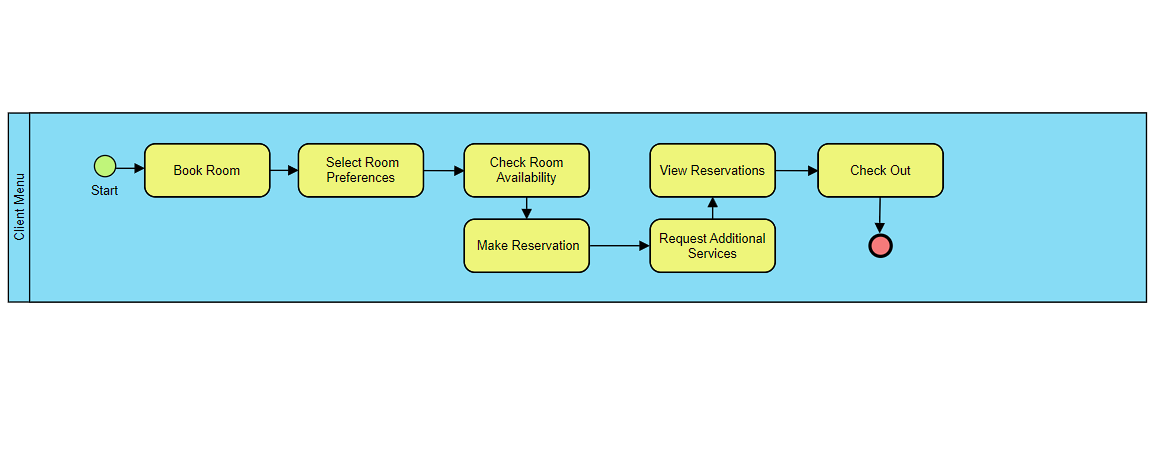


**4.3 BPMN**

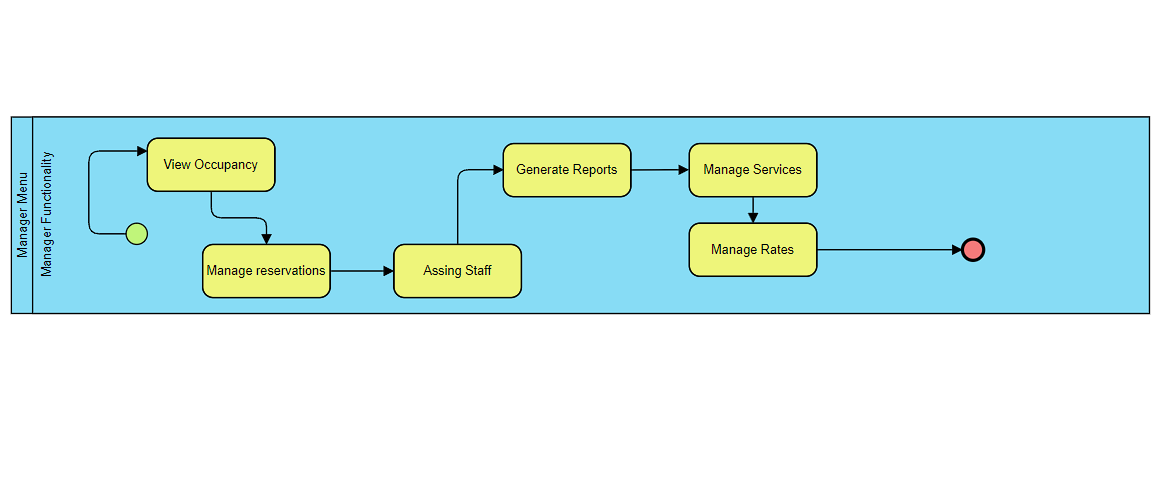
Hotel Management System BPMN:



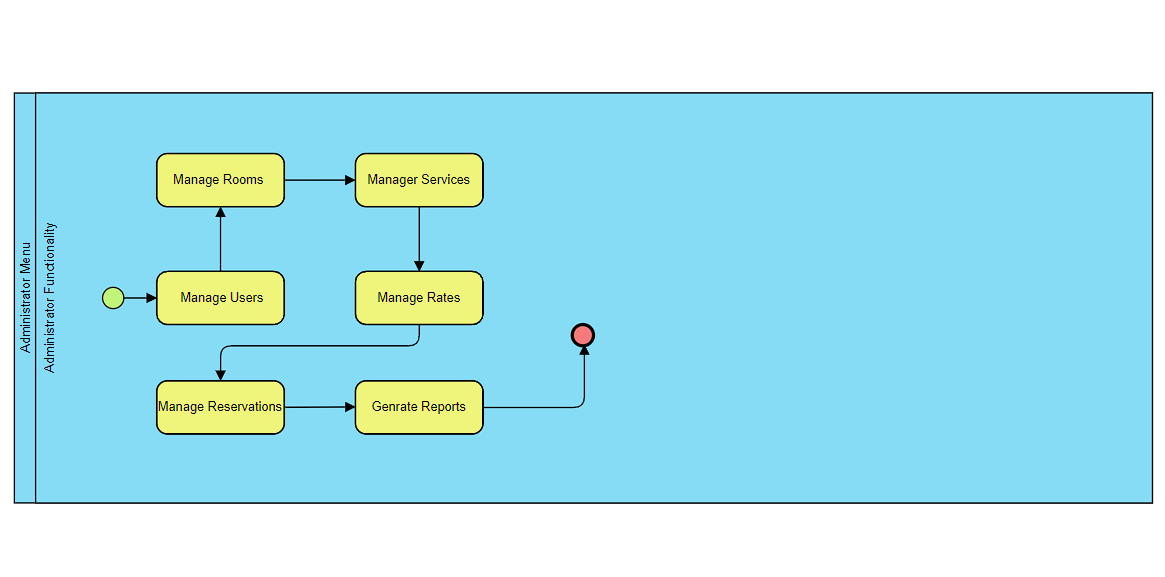
Client BPMN:



Manager BPMN:

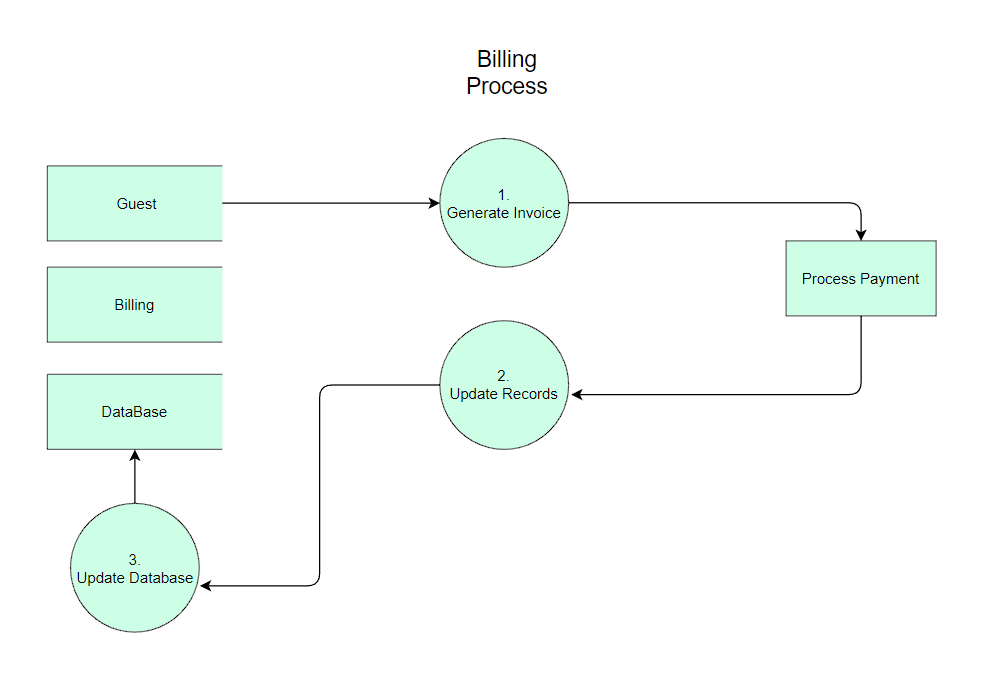


Administrator BPMN:

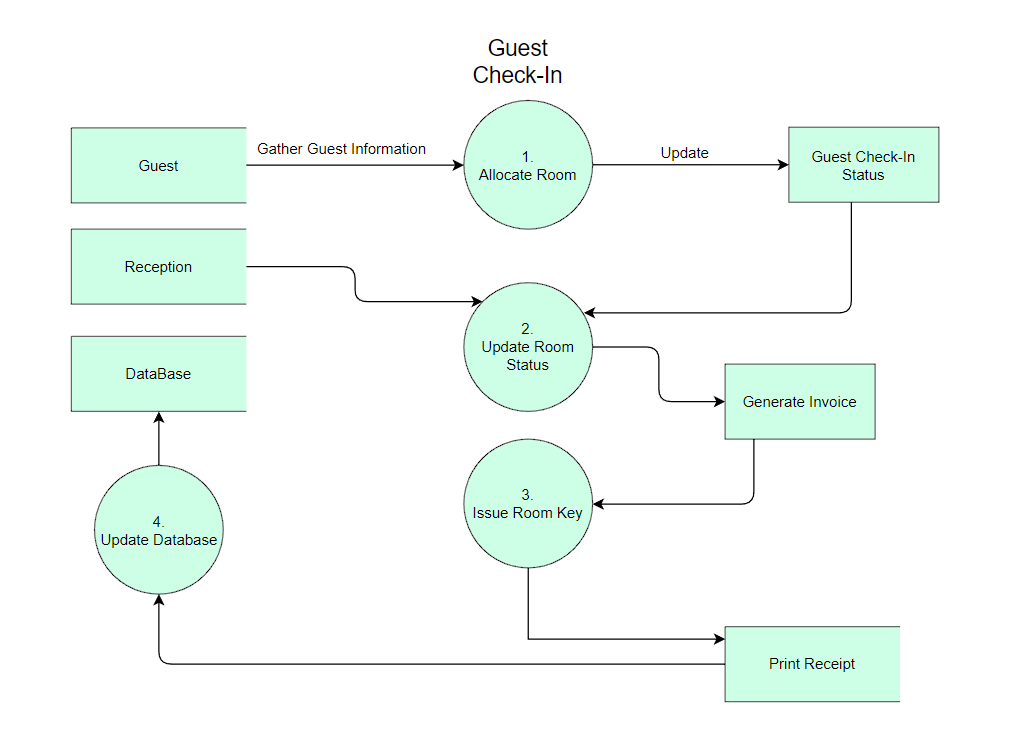


**4.4 Data Flow Diagrams**

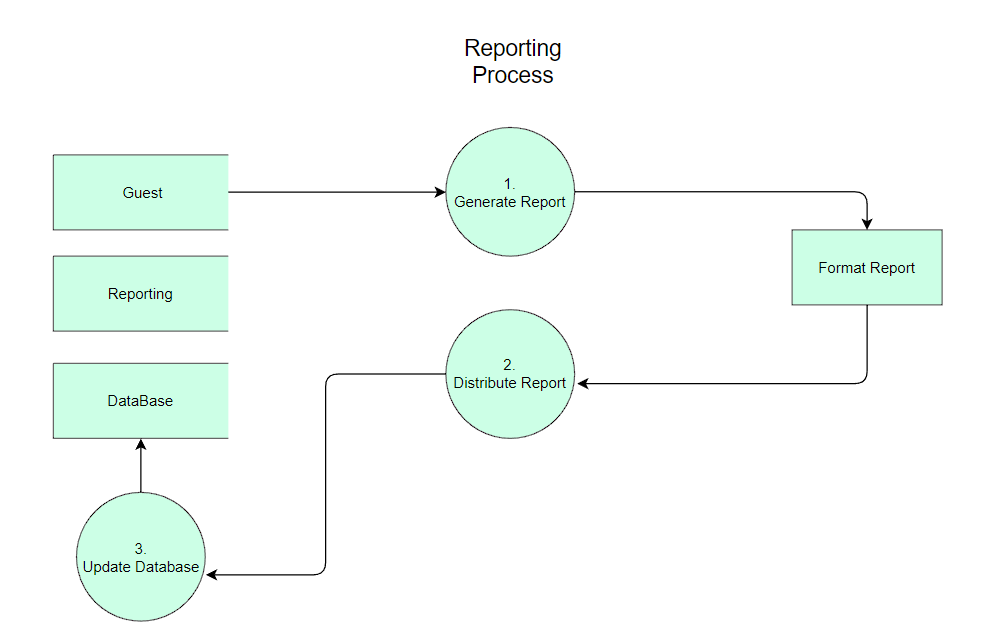
DFD Billing Process:



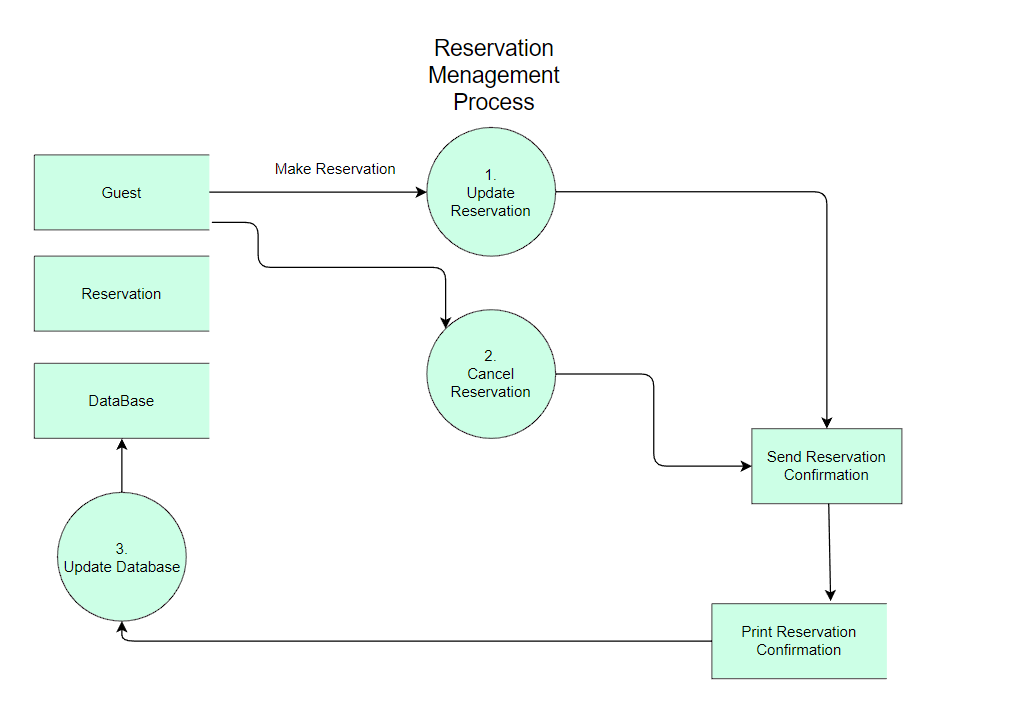
DFD Guest Check-In:



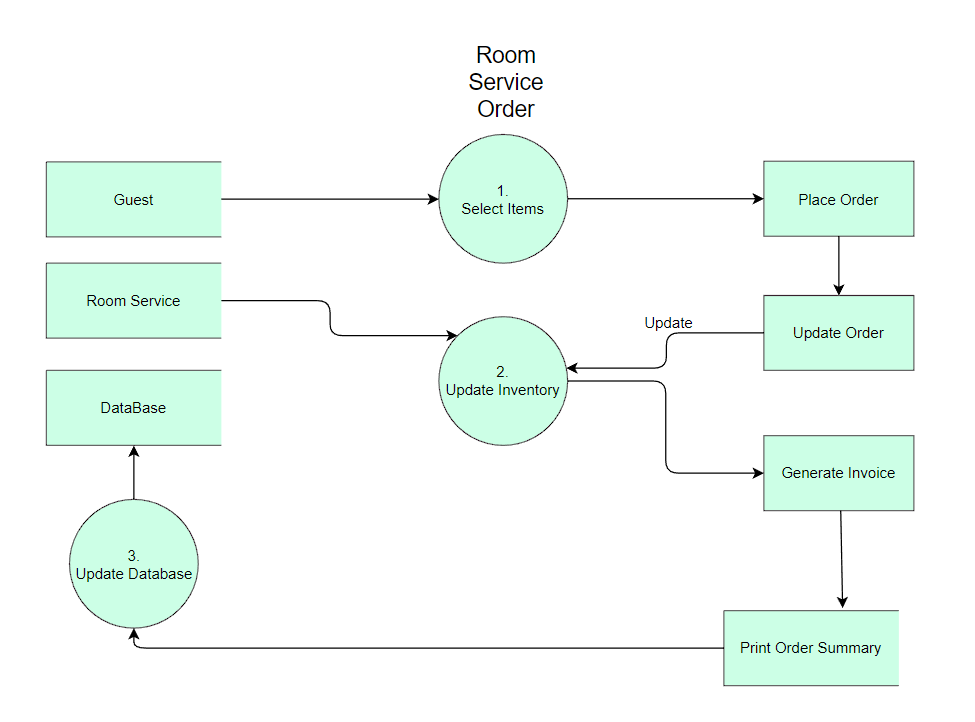
DFD Reporting Process:



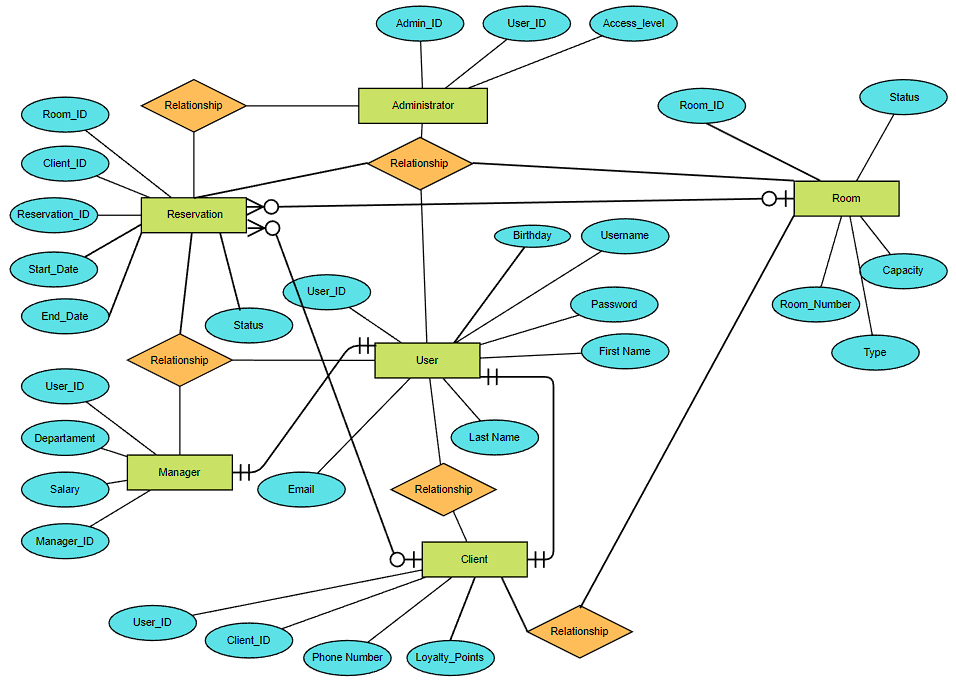
DFD Reservation Management Process:



DFD Room Service Order:

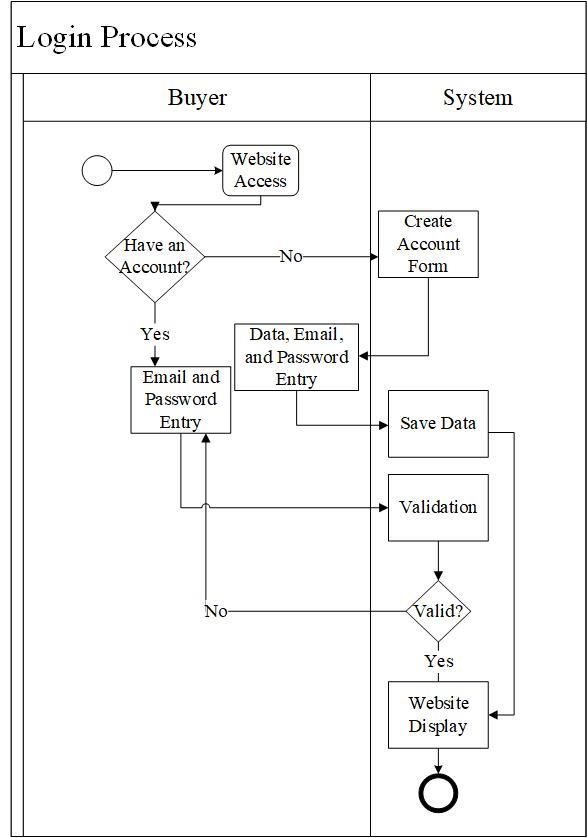


**4.5 Entity-Relationship Diagrams**

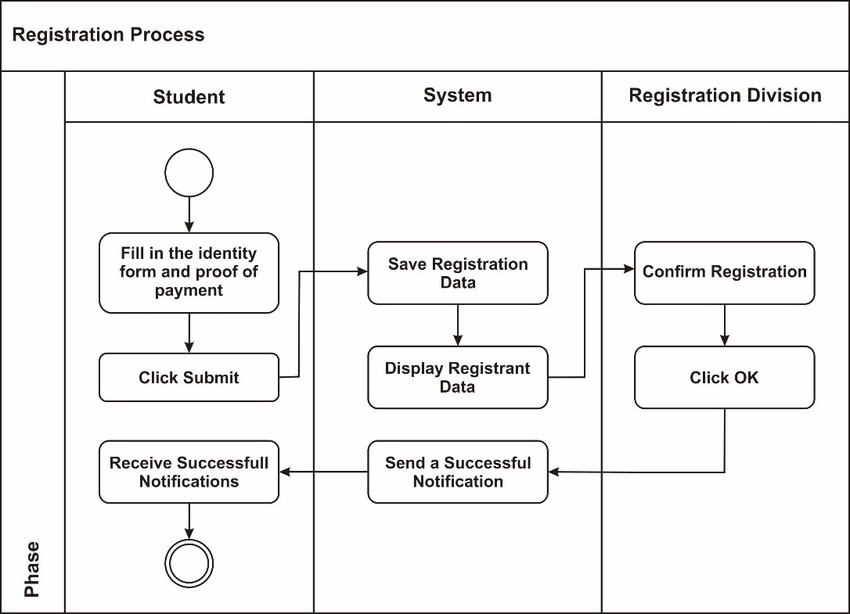
****

**4.6 Activity Diagrams**

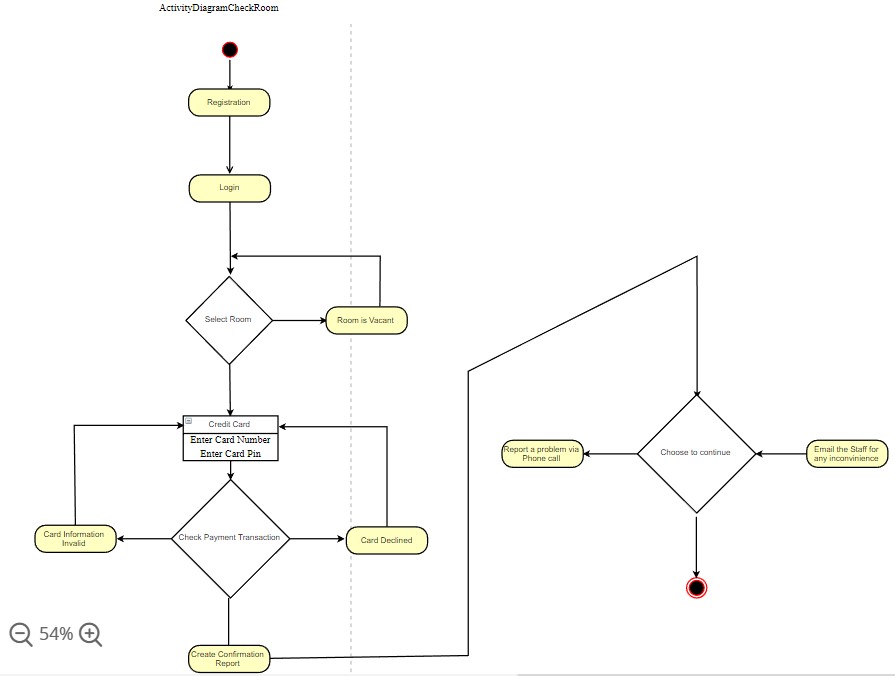
Log In Activity Diagram:



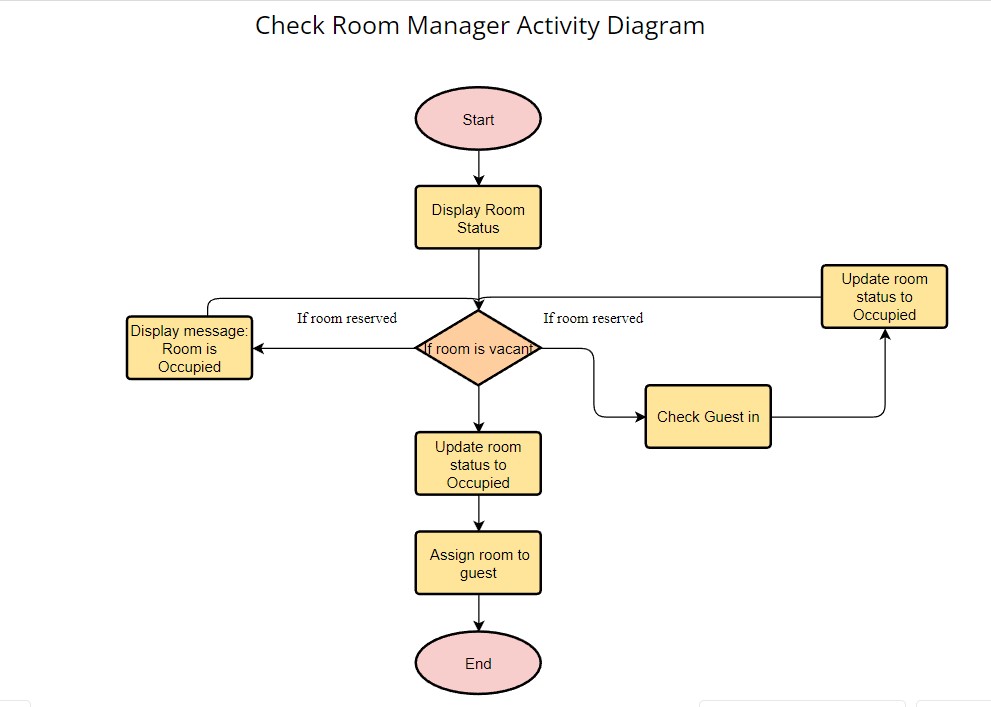
Sign up/Register Activity Diagram:



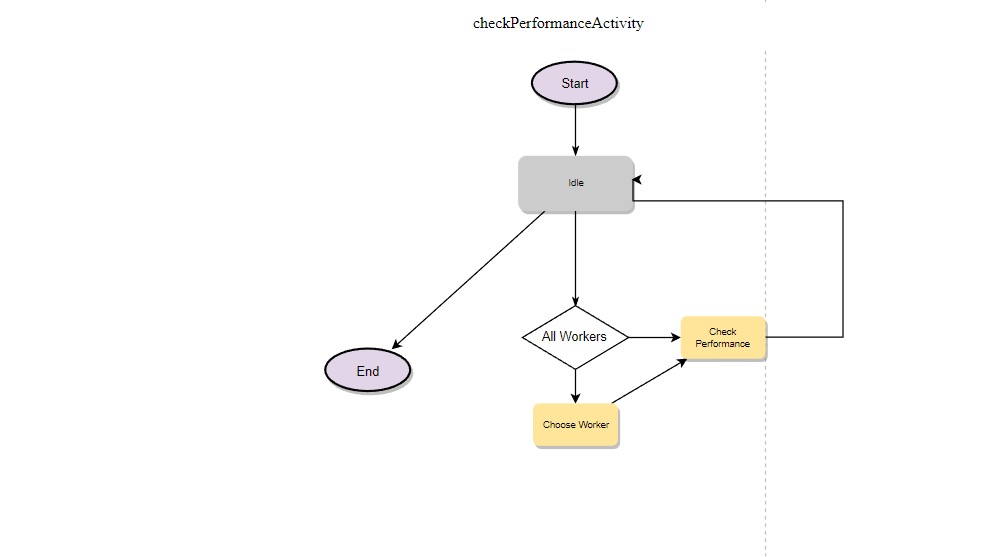
Check Room Activity Diagram:



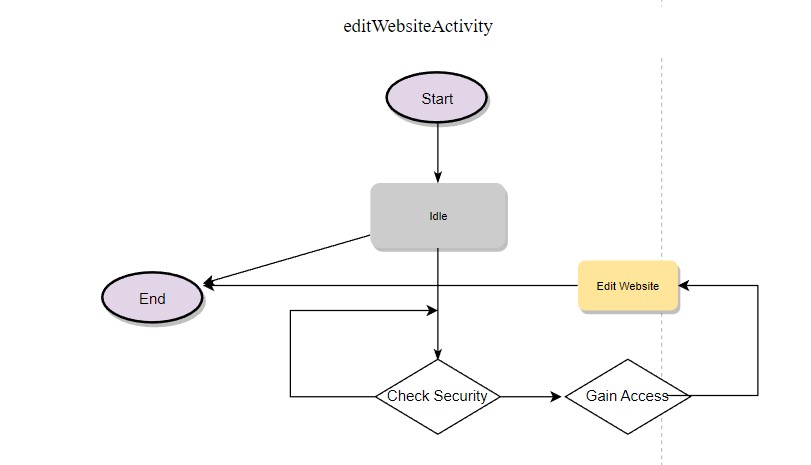
Check Room Activity Diagram (Manager):



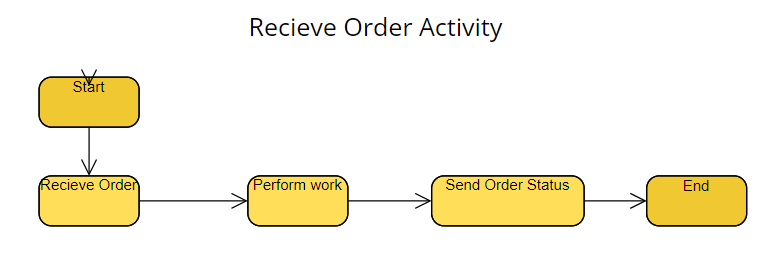
Check Performance Activity Diagram:



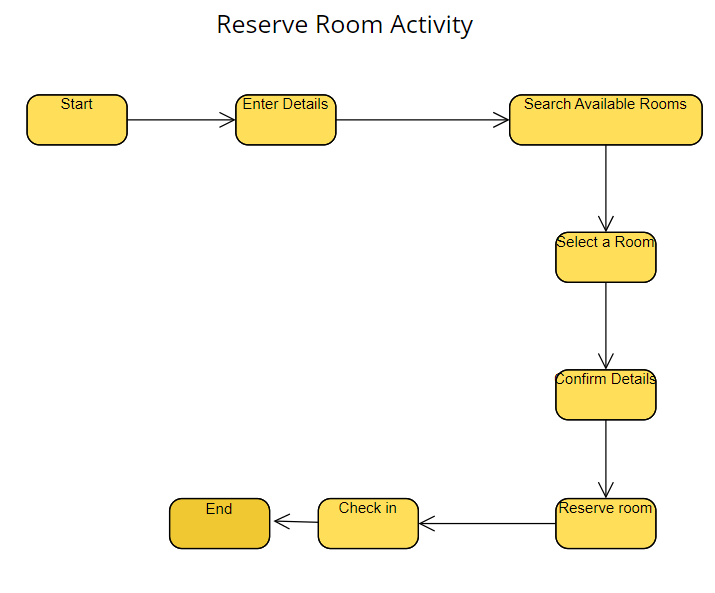
Edit Website Activity Diagram:



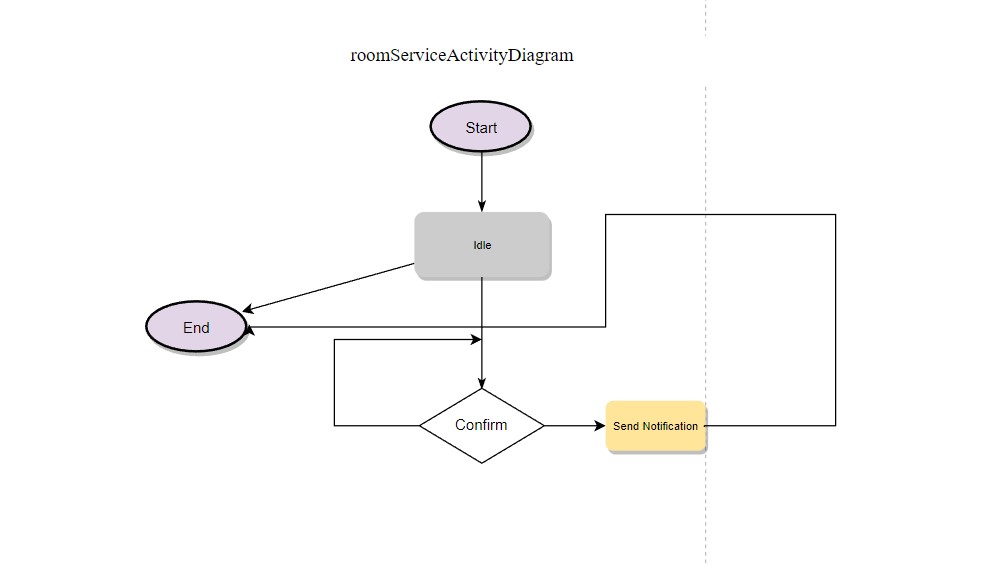
Receive Order Activity Diagram:



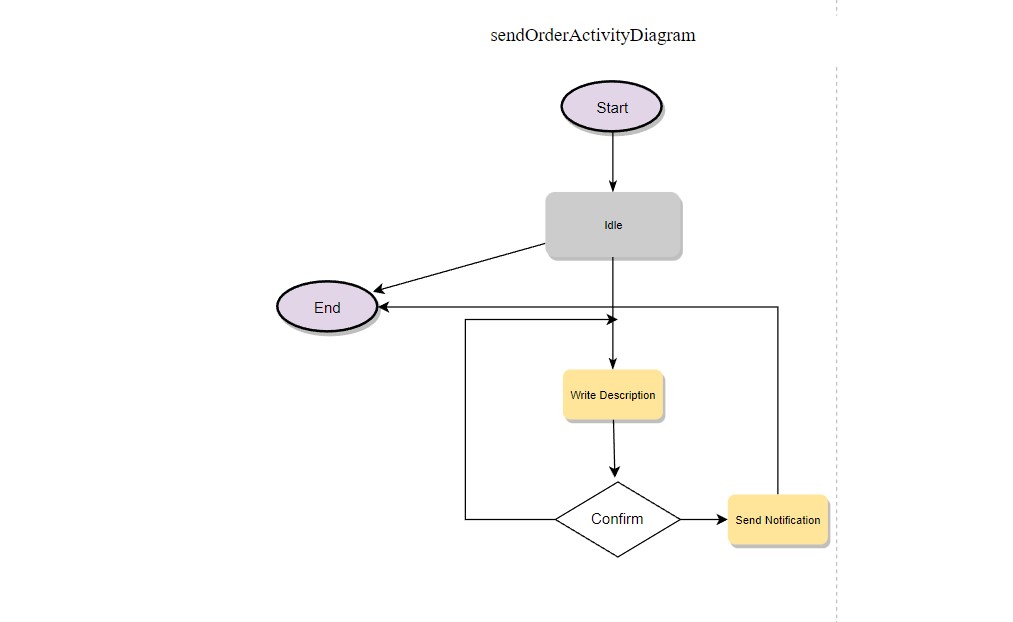
Reserve Room Activity Diagram:



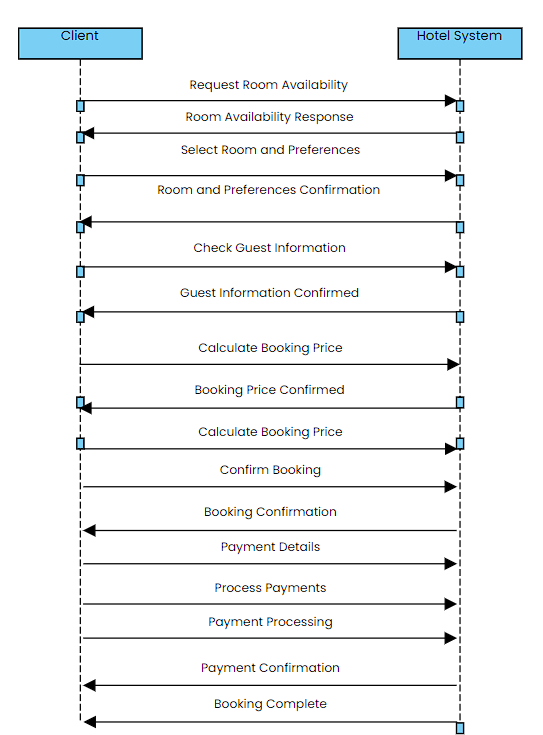
Room Service Activity Diagram:



Send Order Activity Diagram:

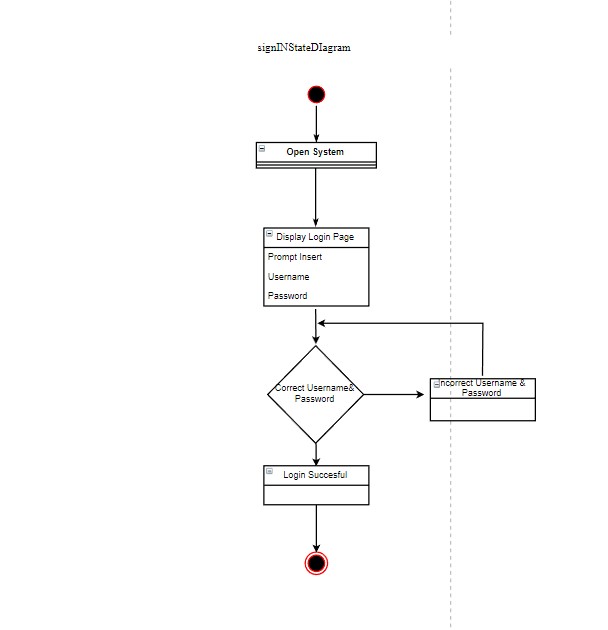


**4.7 Timing Diagram**

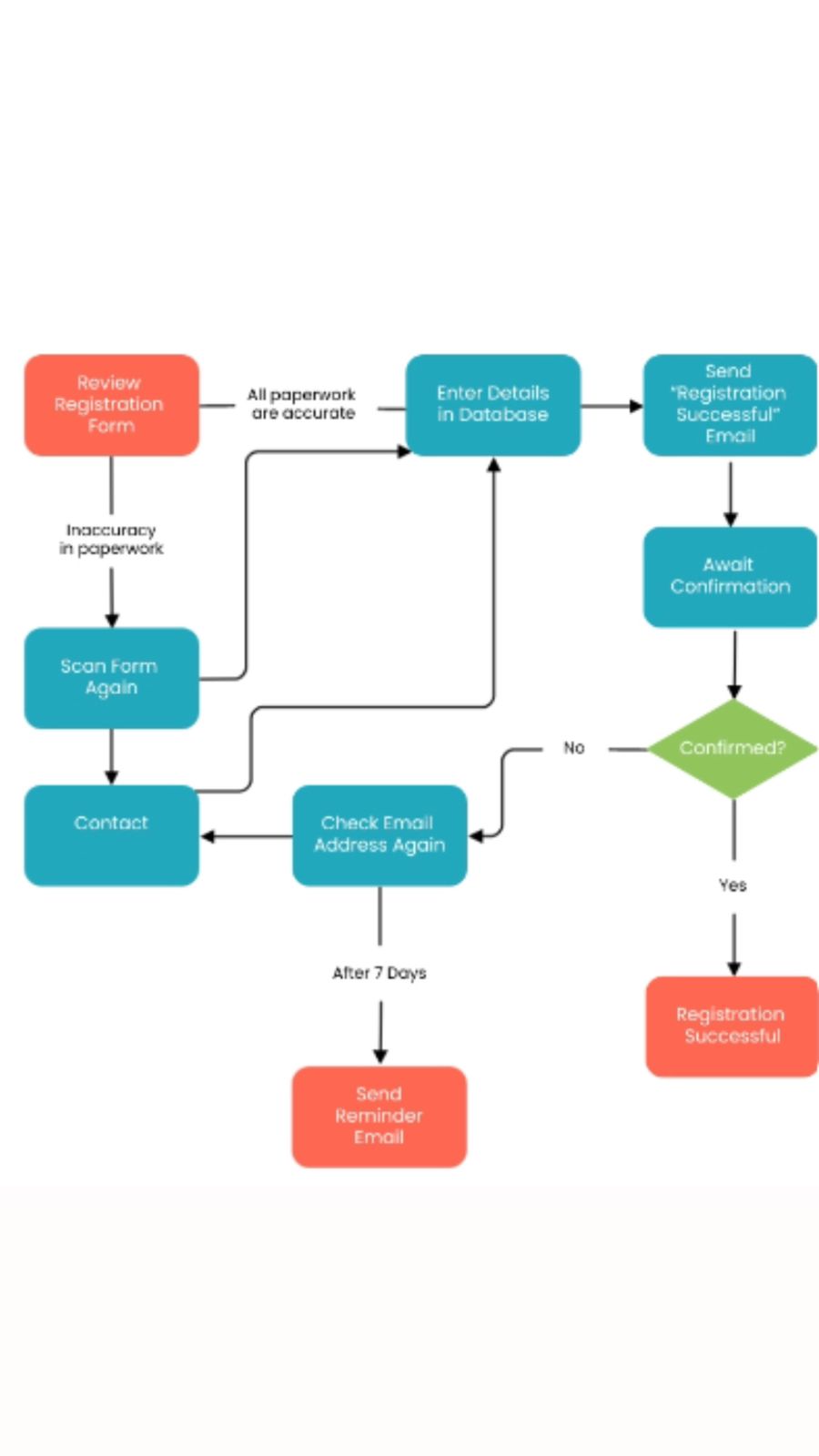
****

**4.8 State Diagram**

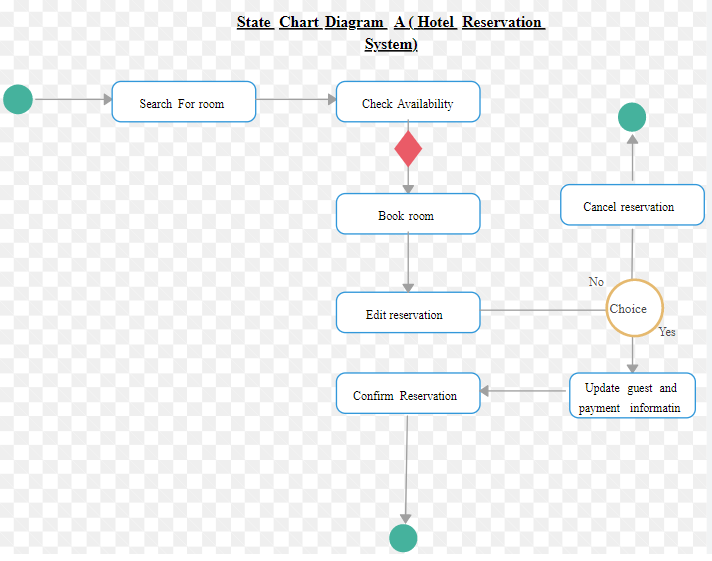
Log In State Diagram:



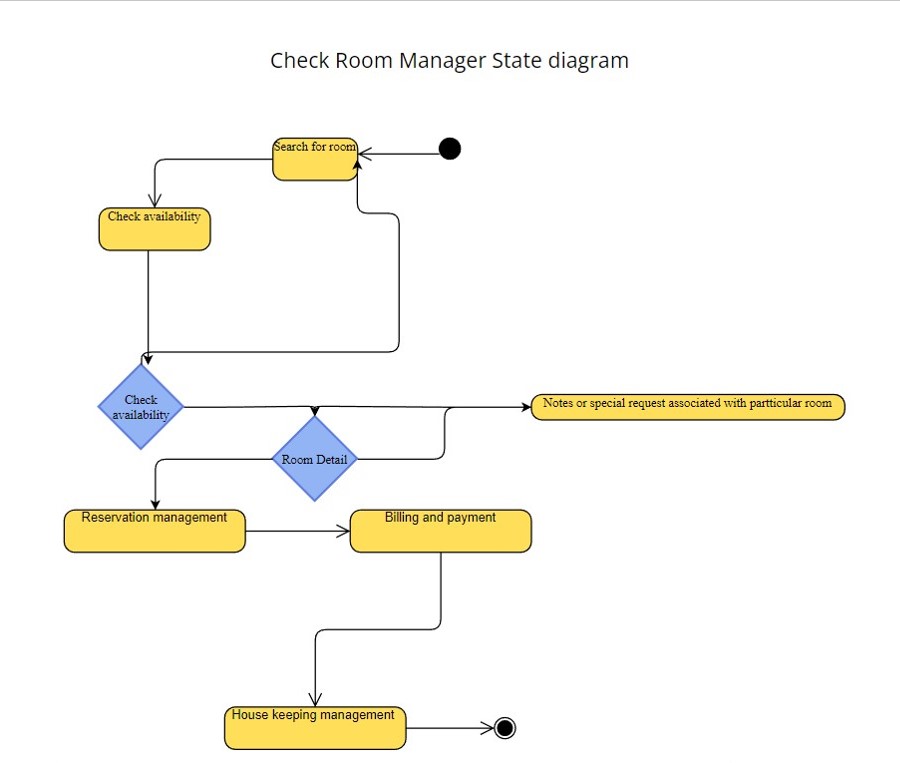
Register/Sign Up Diagram:



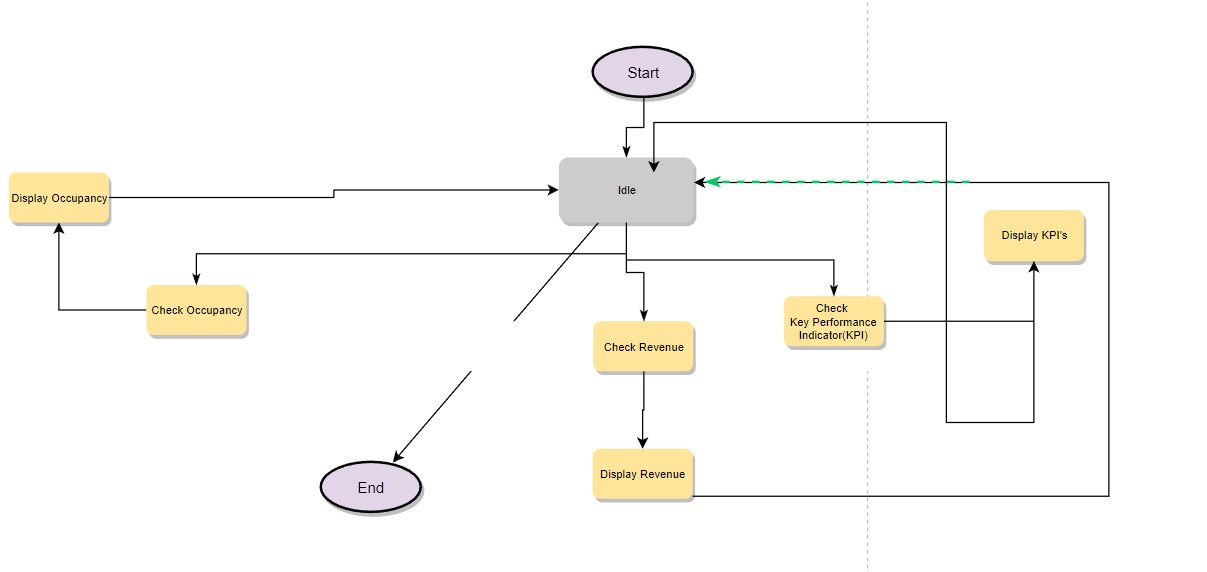
Check Room State Diagram:



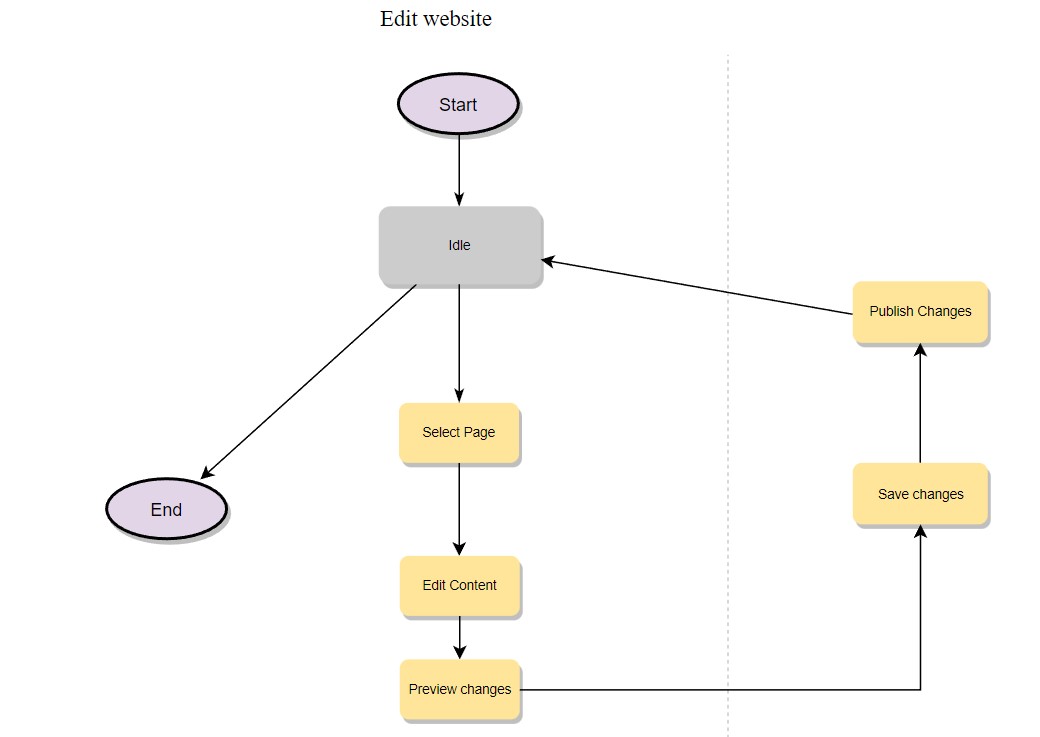
Check Room State Diagram (Manager):



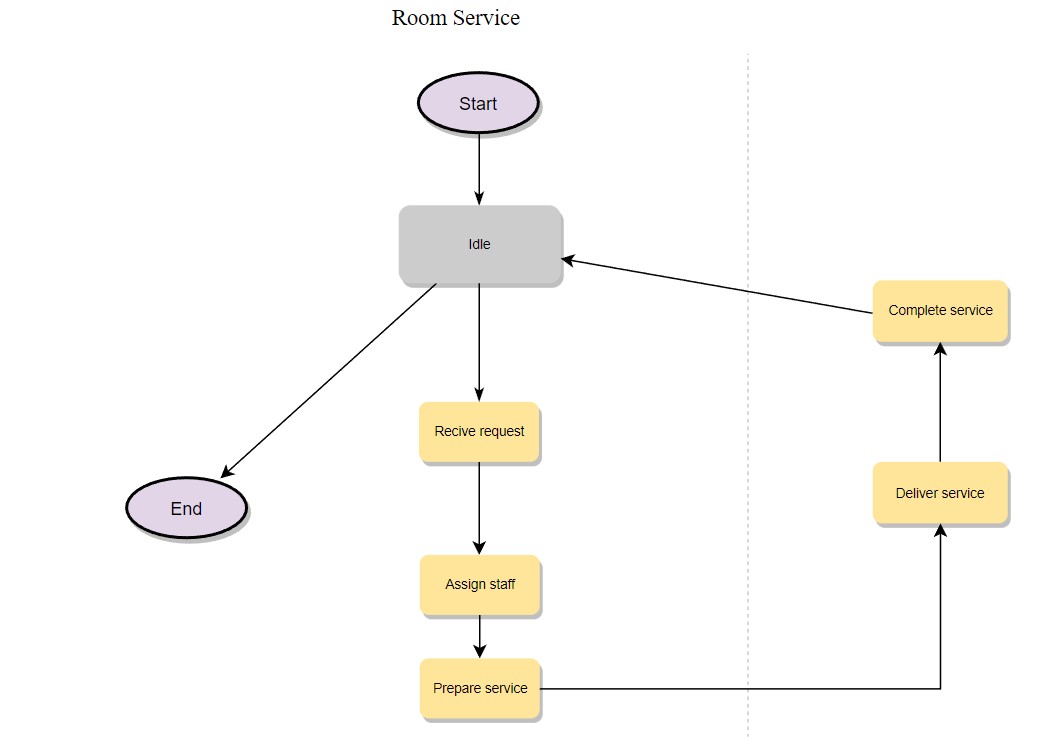
Check Performance State Diagram:



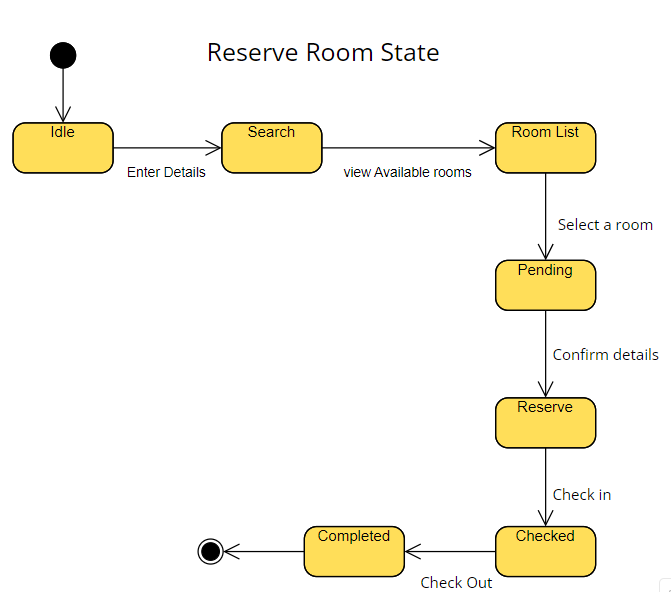
Edit Website State Diagram:



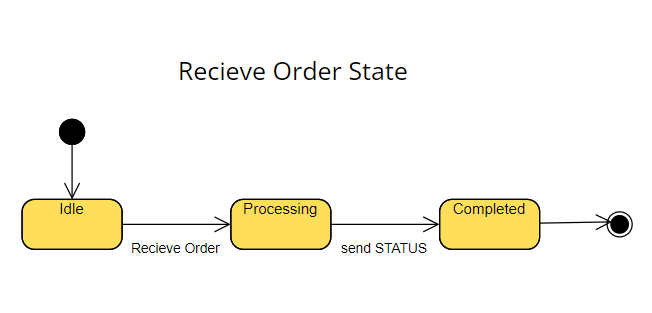
Room Service State Diagram:



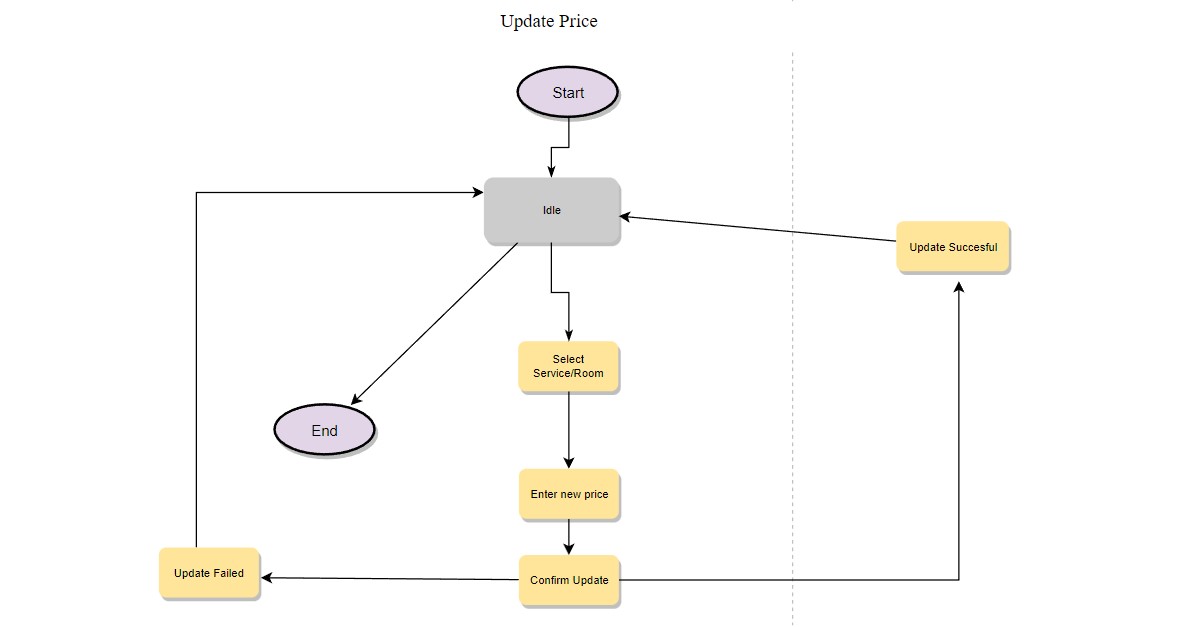
Reserve Room State Diagram:



Receive Order State Diagram:

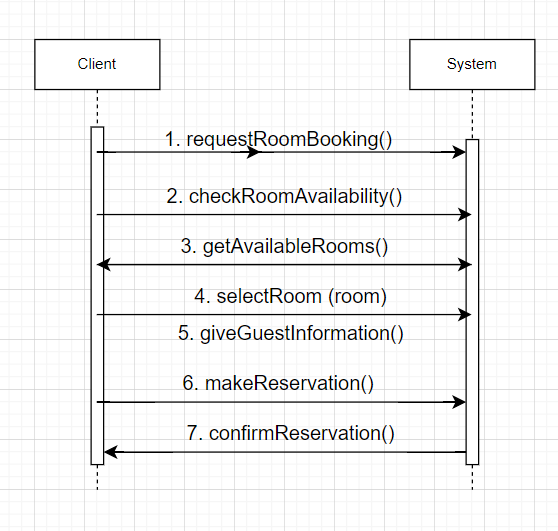


Update Price State Diagram:

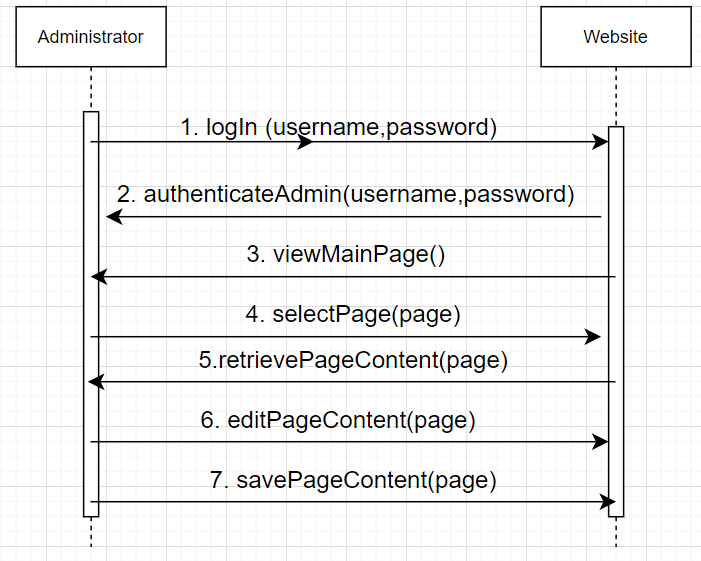


**4.9 Sequence Diagrams:**

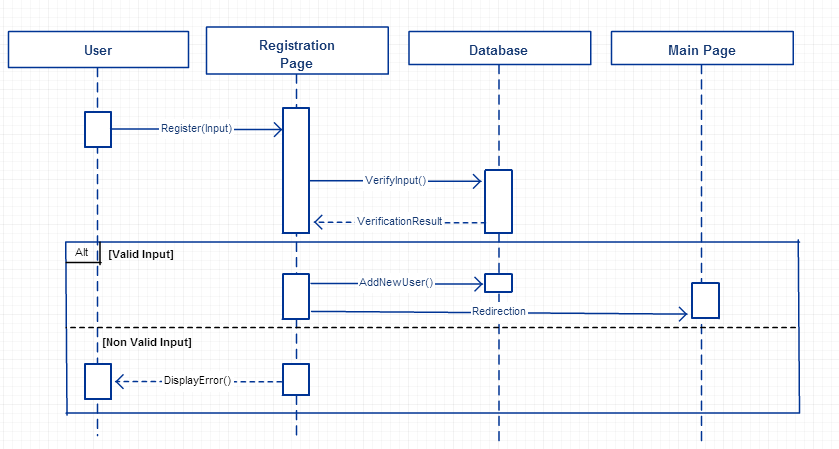
Check Room Sequence Diagram:



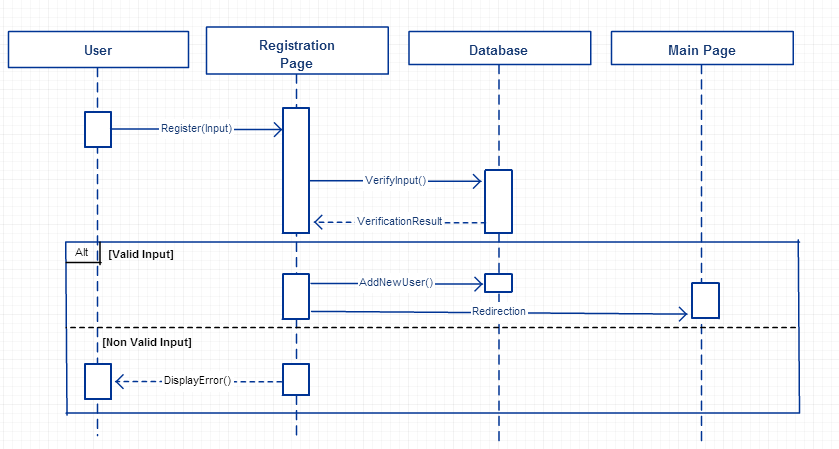
Edit Website Sequence Diagram:



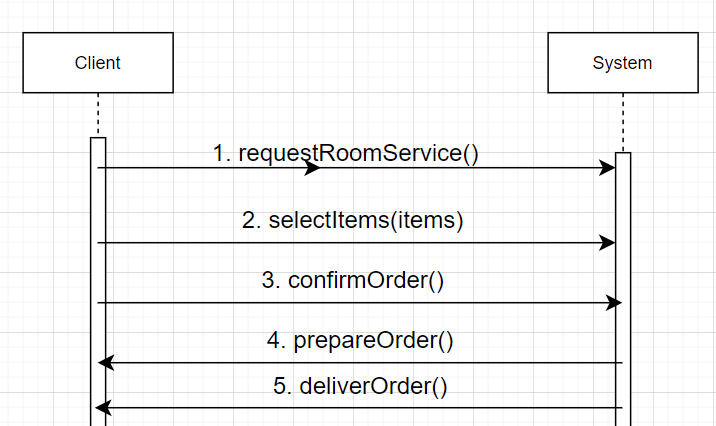
Log In Sequence Diagram:



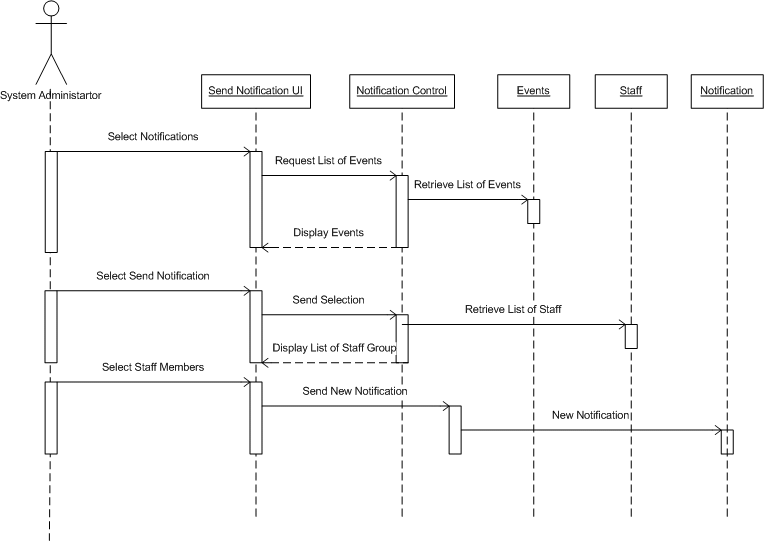
Register/Sign Up Sequence Diagram:



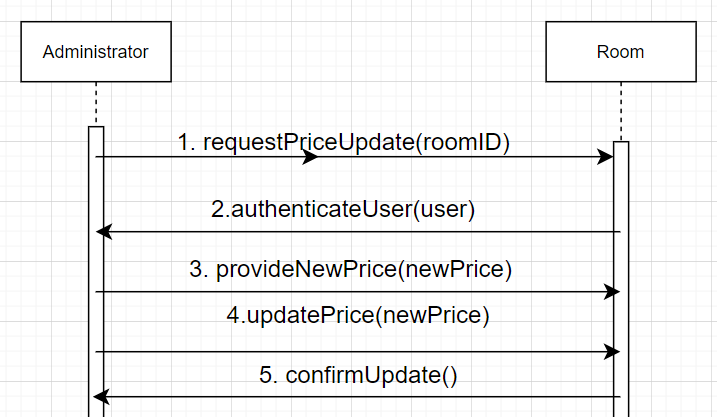
Room Service Sequence Diagram:



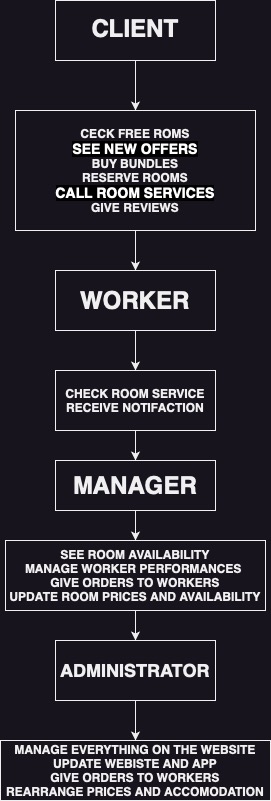
Send Order Sequence Diagram:



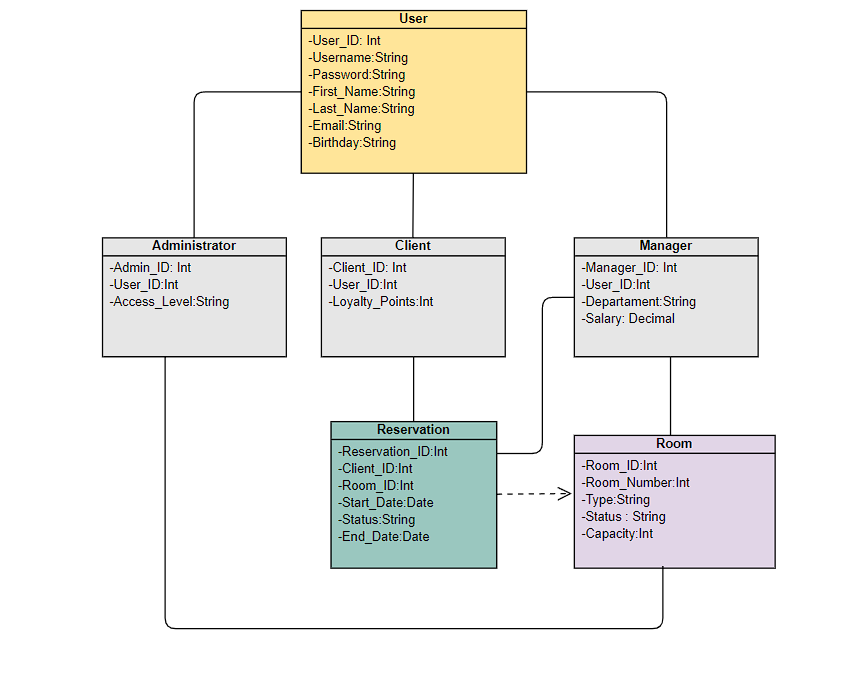
Update Price Sequence Diagram:



**4.10 Collaboration Diagram**

****

**4.11 Class Diagram**

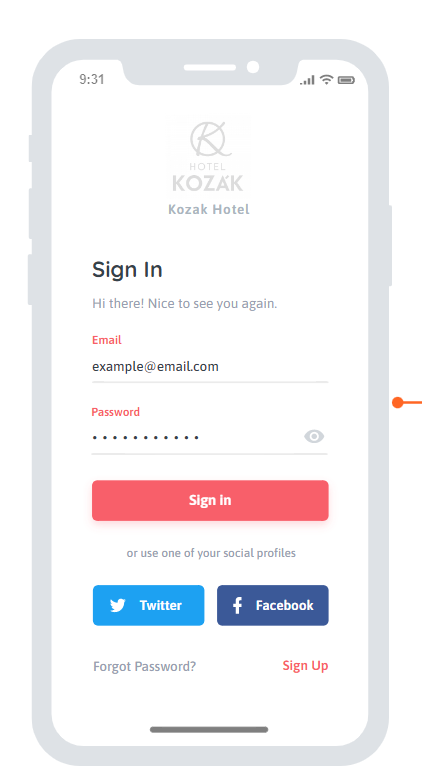
****

**5. Design Patterns**

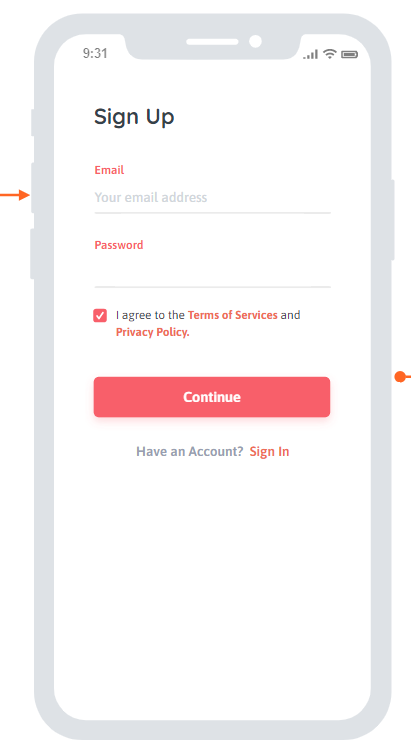
1. Factory Method Pattern:
   * The Factory Method pattern can be used to create different types of users in the system (Clients, Managers, Administrators, Workers) by encapsulating the object creation process.
   * By using a factory method, you can replace direct object construction calls with calls to a special factory method, providing flexibility in creating user objects based on their roles and functionalities.
2. Abstract Factory Pattern:
   * The Abstract Factory pattern can be applied to create different types of reservations in the system, such as Room reservations, Event reservations, or Facility reservations.
   * The abstract factory can define the common interface for creating reservation objects, while concrete factory classes can implement the creation of specific types of reservations.
   * This pattern allows you to decouple the client code from the concrete reservation classes, providing flexibility to switch between different types of reservations.
3. Prototype Pattern:
   * The Prototype pattern can be used to create a prototype of a reservation object that can be cloned to create new instances with similar attributes.
   * This pattern can be beneficial when creating multiple reservations with similar details, such as copying the attributes of an existing reservation and modifying specific details for the new reservation.
4. Singleton Pattern:
   * The Singleton pattern can be applied to ensure that certain classes have only one instance throughout the system.
   * For example, a HotelSettings class that stores and manages global settings for the hotel management system can be implemented as a singleton, allowing access to the settings from various components.
5. Builder Pattern:
   * The Builder pattern can be used to construct complex objects step by step, providing flexibility in creating orders or booking requests.
   * For example, when a client wants to make a booking, the Builder pattern can help guide the user through the necessary steps, such as selecting room types, specifying preferences, adding additional services, and confirming the booking.
6. Observer Pattern:
   * The Observer pattern can be employed to establish a subscription mechanism for notifying different components or users about events and updates in the system.
   * For instance, when a new reservation is made or modified, the Observer pattern can notify relevant entities, such as managers or administrators, ensuring real-time updates and synchronization.
7. State Pattern:
   * The State pattern can be utilized to represent different behaviors and functionalities based on the state of a user or a reservation.
   * For example, a user can transition between different states, such as a normal user, a registered guest, or a checked-in guest, each having different behaviors and access rights within the system.

**6. Appendix**

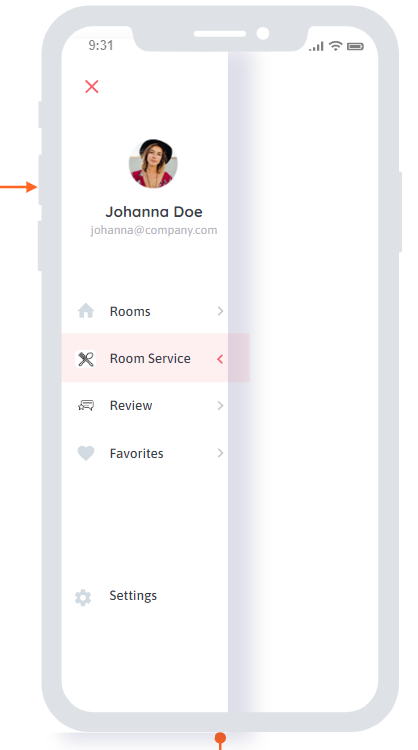
**6.1 Log In Page:**

****

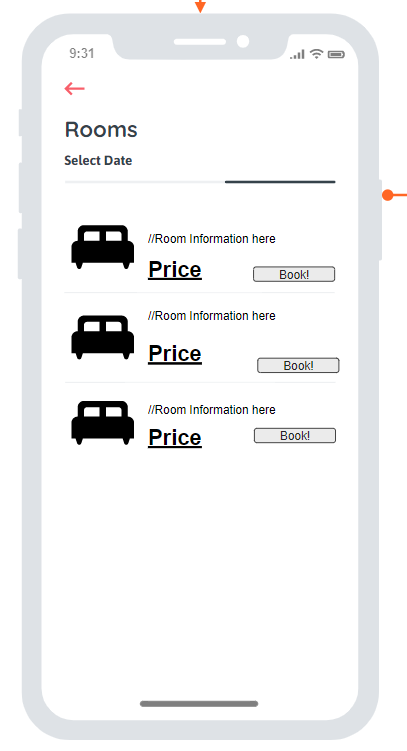
**6.2 Sign Up Page:**

****

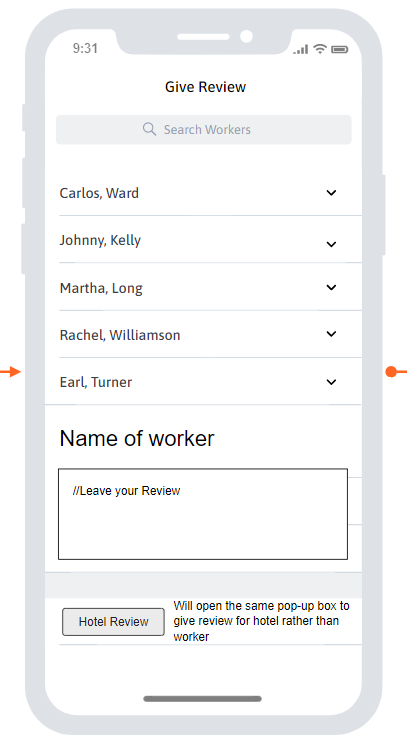
**6.3 Client Menu**

****

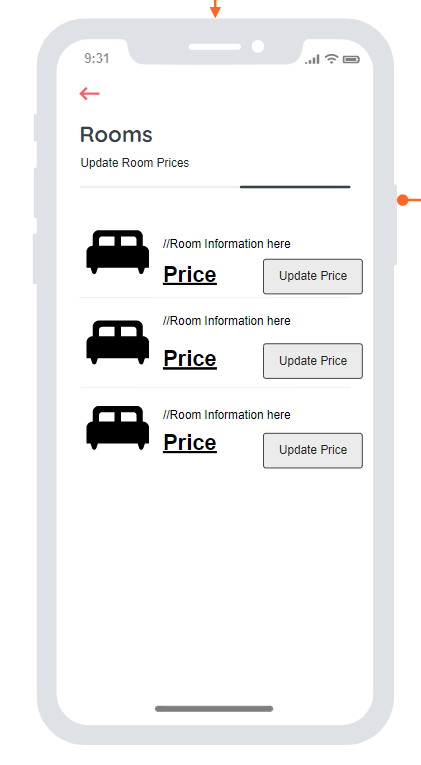
**6.4 Book Room**

****

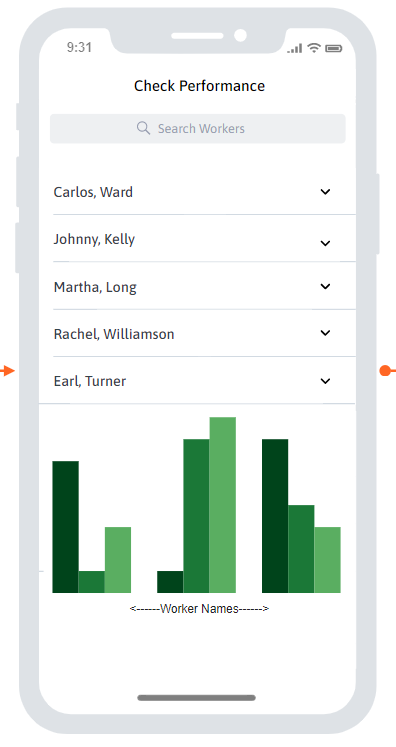
**6.5 Give Review**

****

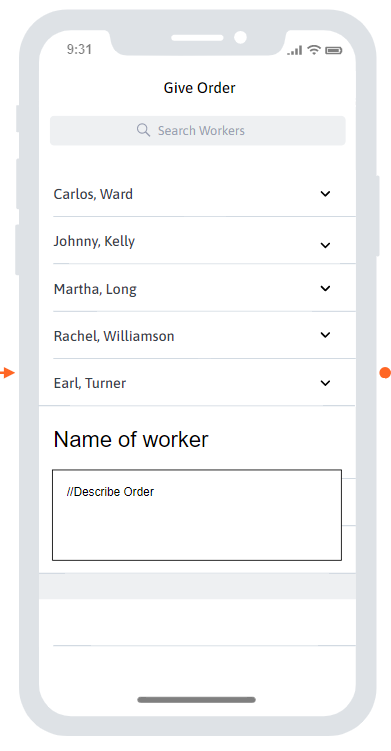
**6.6 Update Room Price**

****

**6.7 Check Performance**

****

**6.8 Give Order**

****