UTM
UNIVERSITI TEKNOLOGI MALAYSIA

SECD 2523 DATABASE
SECTION 08
20242025/1

LECTURER'S NAME: DR. SEAH CHOON SEN

GROUP MEMBERS :

NAME	MATRIC NO.
CHANG WEN XUEN	A23CS5012
LIM CHEN XI	A23CS0103
TAN ZHEN LI	A23CS5025
TEOW ZI XIAN	A23CS0279
BENJAMIN CHEW JUN JIE	A23CS0210

Table of Contents

1.0 Introduction

The hotel booking system is designed to streamline hotel operations, including room reservations, guest management for check-ins and check-outs, housekeeping coordination, luggage handling, and room service management. It provides real-time updates on room availability, automated task allocation, and a centralized interface to enhance communication across different departments. This system will reduce booking conflicts, improve operational efficiency, and enhance the guest experience through self-service portals and personalized services. Hotel staff, such as housekeepers and room service staff, can efficiently coordinate tasks through the system. With automation and real-time tracking, the system ensures that staff can focus more on delivering high-quality service, leading to improved guest satisfaction and operational performance. This project applies relational database design principles to ensure data integrity, enabling the hotel to manage rooms and service requests effectively while providing insightful reports on performance and revenue.

2.0 Background Study

Hotels often face challenges with managing room availability, guest preferences, and internal operations, such as housekeeping and room service. The hospitality industry relies heavily on efficient coordination between multiple services, such as front-desk management, housekeeping, room service, and luggage handling, to ensure smooth operations. Many hotels still use outdated systems, manual tracking, or isolated tools for managing bookings and staff tasks, which often results in errors, delays, and poor guest experiences. As guest expectations increase, especially with the availability of self-service technology, hotels must adopt advanced systems that coordinate booking, housekeeping, and customer service functions. A database-based hotel booking system provides a solution by automating operations, improving data accuracy, and enhancing communication between departments, helping hotels remain competitive and meet modern demands.

3.0 Problem Statement

Managing hotel operations involves several interconnected challenges that can disrupt efficiency and impact customer satisfaction.

1. Manual Errors and Booking Conflicts

Hotels often encounter double bookings and incorrect room assignments due to delayed or inaccurate room availability updates. Without an automated system, errors may occur during reservations or cancellations, leading to customer dissatisfaction. Managing these issues manually becomes especially challenging during peak seasons, resulting in operational inefficiencies and a loss of business.

2. Lack of Real-Time Information and Task Monitoring Between Departments

Housekeepers, room service staff, and receptionists often struggle to stay aligned, as they rely on fragmented communication methods like phone calls or paper logs. This lack of coordination leads to delays in room preparation, duplicated tasks, and incomplete services, impacting the guest experience. Staff need real-time access to room statuses and guest requests to ensure timely service delivery. Smooth interdepartmental collaboration is crucial to maintaining hotel efficiency, but without a centralized system, miscommunication is frequent.

3. Inefficient Luggage Handling and Room Navigation

Bellboys often face difficulties in identifying the correct rooms for luggage delivery, especially in larger hotels. Without proper tracking mechanisms, luggage can be misplaced, or delivery may be delayed, leaving guests frustrated. Additionally, bellboys waste valuable time searching for rooms without effective navigation tools, further slowing down operations.

4. Poor Guest Experience

Guests increasingly expect faster check-in, efficient service delivery, and personalized care. Delayed responses to service requests, long wait times at check-in/out counters, and a lack of personalized amenities diminish the overall guest experience. Poor service quality can affect guest retention, reducing the hotel’s reputation and profitability.

4.0 Proposed Solutions

4.1 Solutions

1. Automate Room Booking and Availability Management

Implement a real-time booking system that updates room availability instantly, ensuring that reservations, cancellations, and check-in are reflected immediately. This system will prevent double bookings and eliminate human errors by validating bookings automatically. Guests will receive confirmation notifications for their reservations, enhancing transparency through an internal dashboard of the hotel booking system application.

2. Real-Time Task Monitoring and Notifications Between Departments

Equip staff with an internal dashboard of the hotel booking system application that displays real-time room statuses and task updates. Each department will have real-time visibility of their assigned duties through the dashboard, reducing the risk of miscommunication and duplicated tasks. Timely notifications and task prioritization will ensure deadlines are met, reducing downtime between check-ins. Receptionists can view room availability and update reservations without delay. In addition, housekeeper and room service staff will receive automated alerts when rooms become available or need immediate attention. The dashboard enables status updates for tasks. For example, housekeepers can update room status as“Room Cleaned” or “Room Cleaning” and room service staff can update status “Room Service Delivered” or “Room Service Preparing”. This seamless coordination will ensure smooth operations, even during high-demand periods.

3. QR Code-Based Luggage Tracking and Navigation

Attach QR codes to guests’ luggage to help bellboys quickly identify the corresponding room. By scanning the QR code, the system will display the guest’s room number, eliminating guesswork and minimizing delivery delays. Additionally, the system will provide optimized navigation routes within the hotel to help bellboys find rooms efficiently. Bellboys can mark deliveries as completed in the system, ensuring transparent tracking.

4. Enhance the Guest Experience with Self-Service and Personalization

Introduce a mobile application that allows guests to handle check-in, check-out, and service

requests effortlessly. Guests can use the application to directly request housekeeping, room service, or other hotel services without needing to contact the reception. Additionally, the application can provide information on hotel amenities and services, allowing guests to make informed choices during their stay. Real-time feedback features within the application will also allow guests to report issues instantly, enabling the hotel to address concerns promptly and maintain high service standards. Connect the system to the centralized database to reduce delay and error between customer's requests and responses. Thus enhance customer satisfaction by ensuring the efficiency of customer services.

4.2 Feasibility Study

4.2.1 Technical feasibility

The proposed hotel booking system is technically feasible, as it leverages widely available technologies and infrastructure. It can be developed using relational databases along with a mobile app for reservations and service requests. The app will also include QR code scanning features to help bellboys manage luggage efficiently. A reliable internet connection is essential to allow real-time updates, notifications, and access for both guests and staff. Staff members will need mobile devices, such as smartphones or tablets, to access task notifications and scan luggage codes on the go. The project requires the involvement of IT professionals to design, develop, and maintain the system, with sufficient training provided to hotel employees to ensure seamless use of the new tools. Additionally, risk mitigation plans will be in place to handle technical failures through regular backups and disaster recovery solutions. The system will also be designed to scale with future hotel expansion, making it sustainable in the long term.

4.2.2 Operational feasibility

The hotel booking system is operationally feasible, as it aligns with the hotel's current needs and can be smoothly integrated into daily operations. The new system will streamline several core activities, such as room bookings, guest check-ins and check-outs, housekeeping tasks, and luggage tracking. By providing real-time updates, the system ensures that tasks are coordinated efficiently across departments, minimizing delays and human errors. QR code scanning for luggage tracking will enhance accuracy and reduce the likelihood of misplaced items, while the mobile app will allow guests to make service requests directly, eliminating the need for front desk involvement. To ensure smooth adoption, staff will undergo training on the use of the internal dashboard of the mobile apps, and QR code systems. Feedback from both staff and guests will be collected during the initial stages to address any issues promptly. With improved efficiency, faster service delivery, and better coordination among departments, the new system will enhance overall operations without disrupting the guest experience.

4.3 Cost-Benefit Analysis (CBA)

Assumptions	
Discount Rate	10%
Sensitivity Factor (Cost)	1.1
Sensitivity Factor (Benefits)	0.9
Annual Change in Production Costs	7%
Annual Change in Benefits	5%

Estimated Cost	
Hardware	RM 30000
Software	RM 10000
Training	RM 5000
Maintenance	RM 3000 per year
Salary (IT staff)	RM 20000 per year

Estimated Benefits	
Increase in Bookings	RM 40000 per year
Operational Efficiency	RM 15000 per year
Customer Retention	RM 10000 per year

Costs	Year 0	Year 1	Year 2	Year 3	Year 4
Development Costs • Hardware • Software • Training	 33000 11000 5500				
Total	49500				
Production Costs • Maintenance • Salary		 3300 22000	 3531 23540	 3778 25188	 4042 26951
Annual Prod.Costs (Present Value)		25300 23000	27071 22373	28966 21763	30993 21169
Accumulated Costs		72500	94873	116636	137805

Benefits	Year 0	Year 1	Year 2	Year 3	Year 4
• Increase in Bookings • Operational Efficiency • Customer Retention		36000 13500 9000	37800 14175 9450	39690 14884 9923	41675 15628 10419
Annual Benefits (Present value)		58500 53182	61425 50764	64497 48458	68022 46460
Accumulated benefits		53182	103946	152404	198864
Gain or Loss		-19318	9073	35768	61059
Profitability Index	1.23				

Profitability index = 1.23, showing that it is a good investment because its index is more than one.

5.0 Objectives

1. Automate Room Booking and Availability Management

Ensure real-time updates to prevent booking conflicts and streamline reservations.

2. Improve Coordination Between Departments

Provide an internal dashboard of system applications for assigning and tracking tasks across receptionist, housekeeping, room service, and bellboy.

3. Enhance Luggage Handling and Navigation

Use QR codes to help bellboys track and deliver luggage to the correct rooms efficiently.

4. Ensure Timely Task Completion

Notify housekeepers and room service staff of their tasks in real-time with prioritization for urgent requests.

5. Improve the Guest Experience

Offer self-service options for check-in, check-out, and service requests, reducing wait times and ensuring faster responses.

6. Optimize Operational Efficiency

Minimize manual errors and improve real-time communication and task tracking among staff.

7. Increase Guest Satisfaction and Retention

Provide personalized services based on guest preferences and allow instant feedback to address concerns promptly.

6.0 Scopes

6.1 Project Scope

The scope of this hotel booking system covers all essential hotel operations to ensure smooth functioning and high-quality service. This comprehensive hotel booking system ensures smooth operations by automating processes, improving staff coordination, and providing high-quality services, ultimately enhancing guest satisfaction and business performance.

1. Room Booking Management

Automates room availability tracking, reservations, and booking validation.

2. Guest Check-in and Check-out

Streamlines guest management with self-service portals and real-time updates for staff.

3. Housekeeping Management

Assigns cleaning tasks based on room status and check-out schedules, with prioritization for urgent tasks through the dashboard.

4. Luggage Handling and Delivery

Bellboys use QR codes to identify guest rooms and update luggage delivery status in real-time.

5. Room Service Requests

Provides guests with self-service options to request services through mobile application and the room service staff will receive the request.

6. Staff Task Monitoring

Real-time task tracking via system application ensures staff know what to do, when to do it, and how to complete it.

7. Real-Time Communication and Alerts

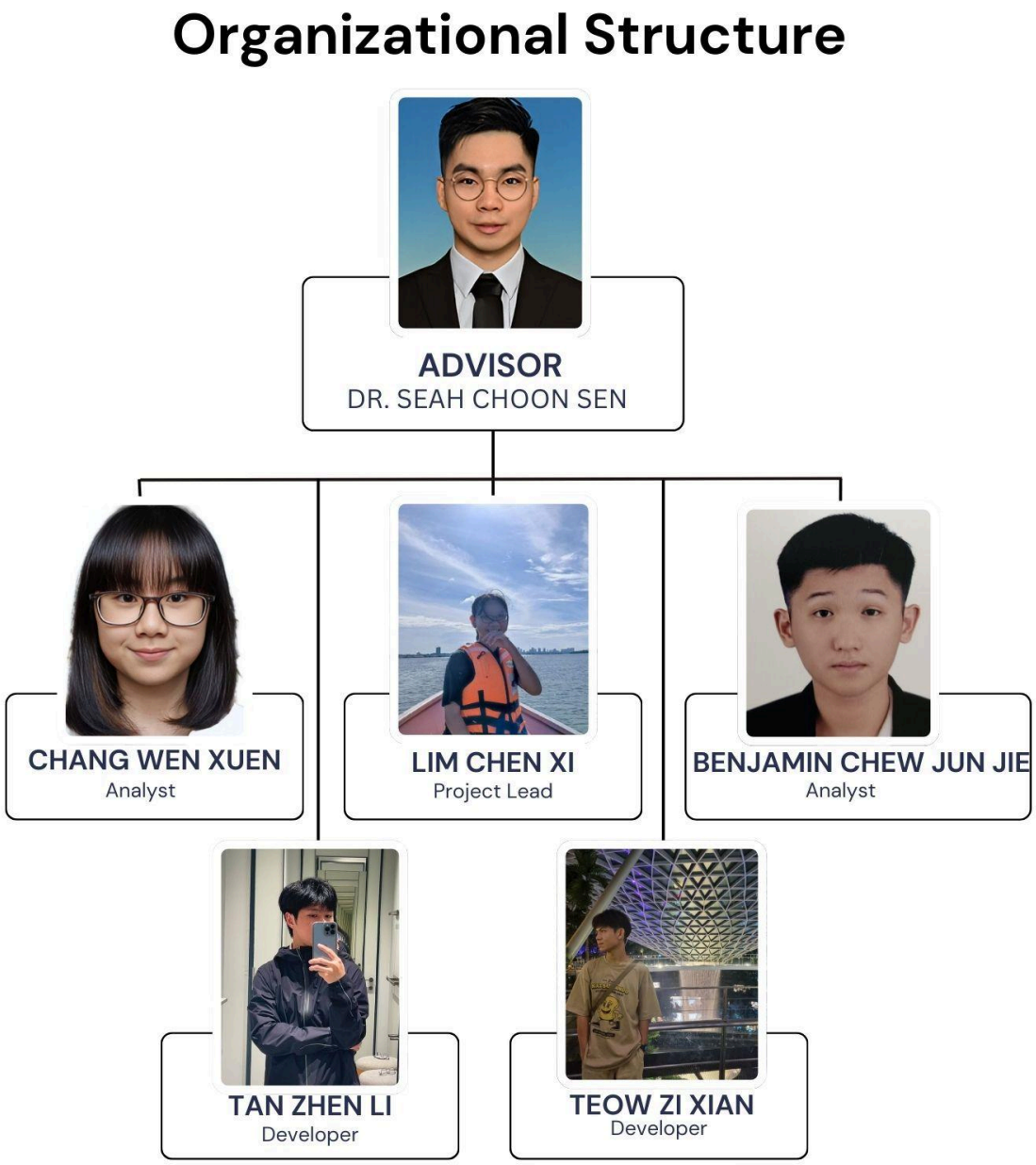
Keeps all staff informed of tasks and status changes, ensuring timely action and minimizing delays.

6.2 System Boundaries

The hotel booking system includes several key functions. It manages room availability and bookings by handling reservations and modifications in real-time. The system also supports guest check-ins and check-outs, updating room status automatically. Through housekeeping and task management, it assigns and tracks cleaning tasks, ensuring rooms are prepared on time. A luggage tracking system using QR codes ensures accurate and efficient delivery of guest luggage to the correct rooms. Guests can use the mobile app or web portal for self-service and provide feedback. The system's database management stores all relevant data, such as bookings, guest details, room status, and task assignments. Staff use the internal dashboard of mobile apps to monitor tasks and coordinate operations efficiently.

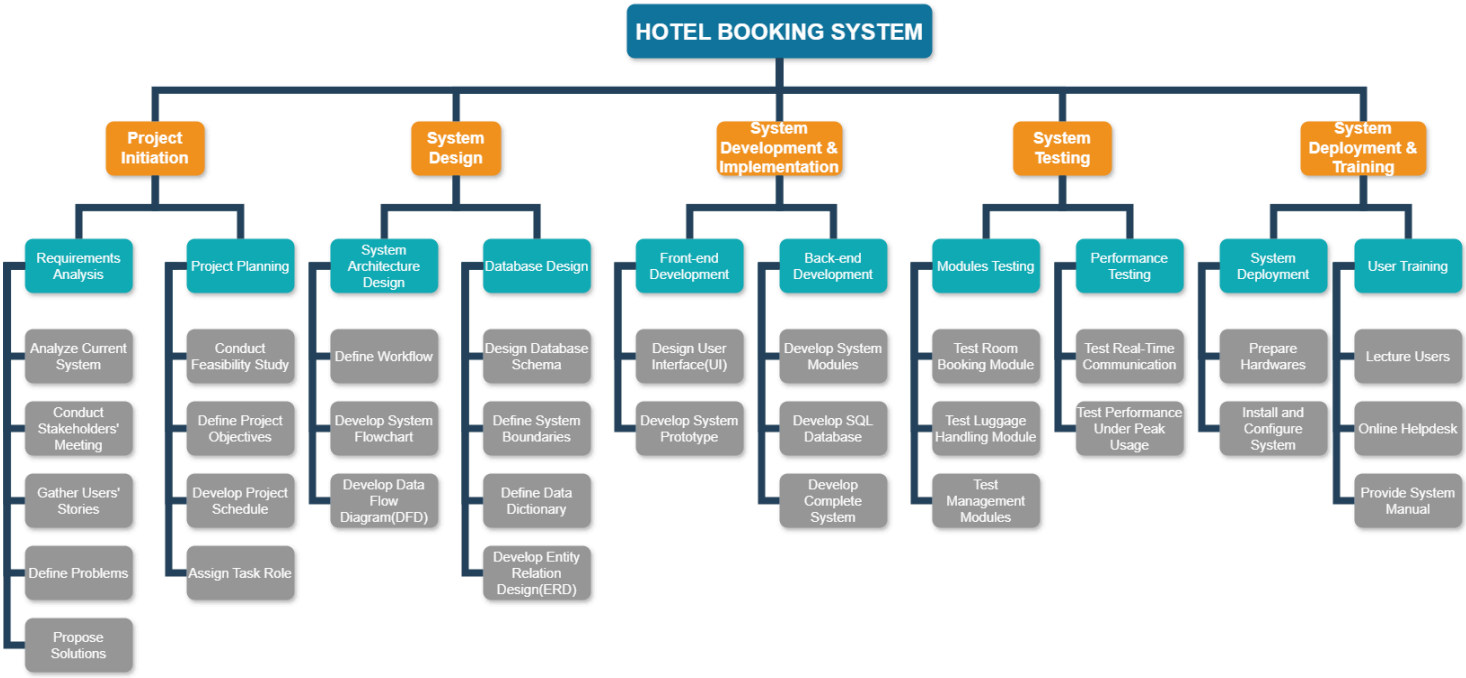
7.0 Planning

7.1 Human Resource



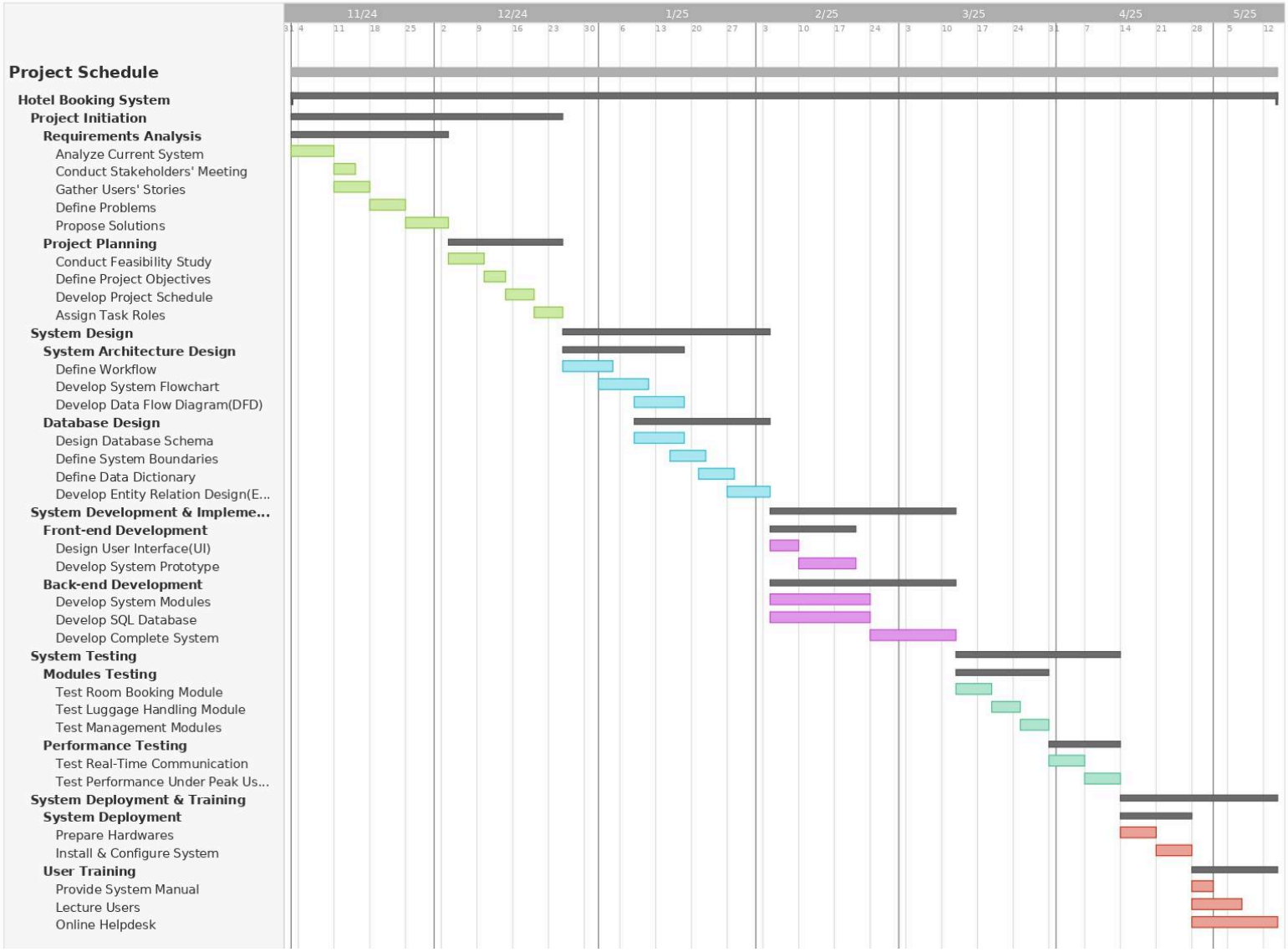
7.2 Work Breakdown Structure (WBS)

We breakdown the works of the Hotel Booking System project by dividing the project into 5 main phases: Project Initiation, System Design, System Development & Implementation, System Testing and System Deployment & Training.



7.3 Gantt Chart

The Gantt Chart below shows the project schedule which start from 01 NOV 2024 until 13 MAY 2025 (Takes about 28 weeks):



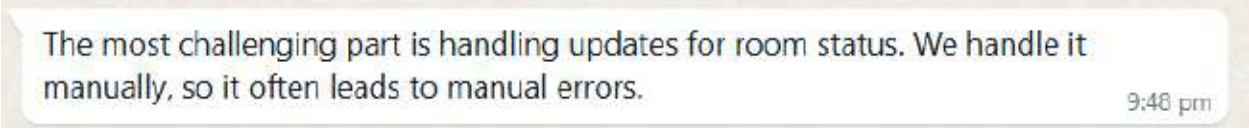
7.4 Methods for Information Gathering

Information gathering is foundational for effective decision-making and comprehensive research. To learn more about this system, we have conducted an interview containing both closed and open-ended design questions. This interview is conducted via a communication application, Whatsapp with a current hotel staff.

Question 1:

What is the most challenging part of your daily tasks when using the current hotel management system?

Ans:

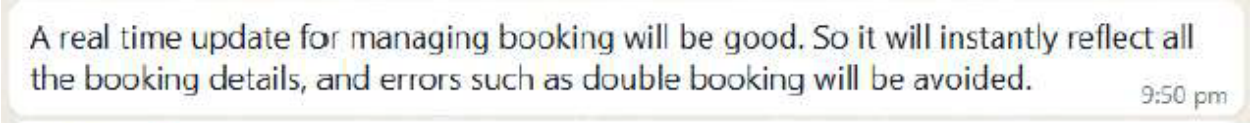


The response shows that the current hotel system implements manual updates to room status which results in mistakes that interrupt workflow and lag in processes.

Question 2:

What improvements would you like to see in managing bookings and cancellations?

Ans:



The respondent was hoping for a real-time updates system so that would prevent double booking errors and improve accuracy of booking data.

Question 3:

What steps do you currently take to ensure room availability updates are accurate and timely?

Ans:

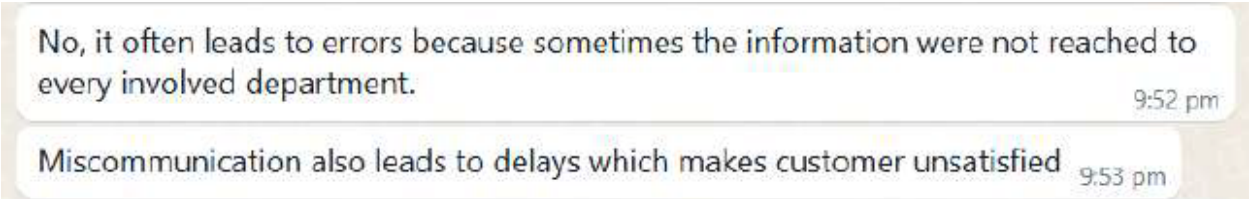
From the response we acknowledged that reliance on manual methods, such as phone calls and

logs, will cause delays and inefficiencies in updating room statuses.

Question 4:

Does the current system allow effective communication between departments (e.g., front desk, housekeeping)?

Ans:

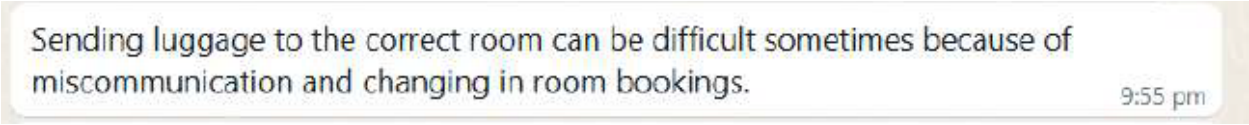


Based on the response, we know that the current hotel management system leads to ineffective communication between departments resulting in errors, delays, and unsatisfactory customer service.

Question 5:

What difficulties have you encountered when delivering luggage or navigating large hotel premises?

Ans:

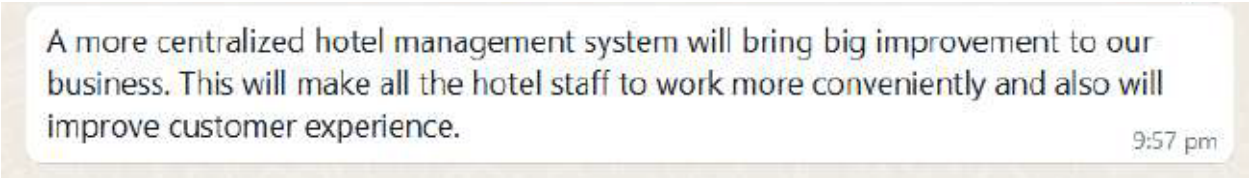


The respondent stated that miscommunication and last-minute room booking changes create challenges in delivering luggage accurately.

Question 6:

What changes would you recommend to improve the guest experience, particularly in terms of speed, personalization, or service quality?

Ans:



A centralized management system would enhance staff convenience and significantly improve the guest experience based on the response.

7.5 User View

Manager view

The Manager View is to support hotel staff in their daily work throughout hotel operations. Managers oversee staff performance, review and assign tasks, and ensure smooth hotel operations by monitoring the management system and addressing issues with IT. They also receive alerts about critical updates, such as low inventory or delays, to take timely action.

Receptionist view

The receptionist view serves as a central interface for front desk staff, allowing them to manage reservations, guest interactions, and room assignments. Key features include reservation and check-in management to simplify check-ins and check-outs, access to guest profiles for personalized data, and room assignment to handle room allocations and changes.

Guest view

The Guest View provides a personalized experience for guests, from booking to checkout. Room Search and Booking helps guests find and reserve rooms, while reservation management enables them to modify or cancel reservations. Additionally, convenient check-ins and check-outs options reduce front desk visits, and service requests allow guests to order room service. The view also allows guests to provide feedback and reviews to let guests share their experiences, to support hotel improvements.

Housekeeper view

The housekeeping view is designed to ensure room cleanliness and efficient management of cleaning tasks. It includes room status updates, which display the current status of each room (e.g., “occupied,” “vacant,” “needs cleaning,” or “ready for check-in”). The view also includes task assignment and tracking which allows specific housekeepers to be assigned to rooms, enabling them to mark tasks as “cleaned” or “maintenance needed” upon completion. The view also facilitates maintenance requests, enabling staff to report any issues, such as broken appliances or plumbing problems, directly to the maintenance team. Additionally, schedule management which provides daily cleaning schedules for the housekeeping.

Bell Boy view

The Bell Boy View in a hotel management system is tailored to enhance the efficiency of bellhops. This view includes room and guest information, providing details on upcoming guest arrivals, room numbers, and preferences, which helps bellhops anticipate needs and prepare accordingly. A key feature is Luggage and Task Tracking, where each piece of luggage is tagged with a QR code. When bellhops scan the QR code, they can instantly access information about the corresponding room and all relevant guest details, simplifying the luggage delivery process. The view also includes Requests and Notifications to allow bellhops to receive real-time alerts about guest needs. Additionally, it facilitates communication with the front desk to keep bellboys updated on guest check-in and check-out times.

Room Service Staff view

The Room Service Staff view is designed to ensure completing tasks such as delivering food and beverages to guest rooms, responding to service requests, and addressing any maintenance issues within the rooms. They ensure that guests receive efficient service, assisting with room-related needs.

7.6 Benefit and Summary of the Proposed System

Benefits of the Proposed System

1.Improved Efficiency:

The system automates key processes like room booking, check-ins, and housekeeping, reducing manual errors and saving valuable time for staff.

2.Enhanced Communication and Coordination:

Departments like reception, housekeeping, and bell services can stay aligned with real-time updates and notifications, ensuring smooth operations.

3.Better Guest Experience:

Guests will enjoy faster check-ins, self-service options for service requests, and quicker responses to their needs, creating a seamless and enjoyable stay.

4.Accurate Luggage Tracking:

The QR code-based tracking system eliminates delays and errors in luggage delivery, improving efficiency and guest satisfaction.

5.Increased Revenue and Cost Savings:

By preventing errors like double bookings and improving task coordination, the system helps optimize resources and boosts operational performance.

6.Future Scalability:

The system is designed to adapt to the hotel's growth, ensuring it remains effective as the business expands.

Summary of the Proposed System

The proposed hotel booking system is a comprehensive solution designed to address inefficiencies in current hotel operations. It centralizes critical functions, such as room booking, housekeeping management, and guest service requests, into one integrated platform. With features like real-time updates, QR code luggage tracking, and a mobile app for guests, the system ensures smooth communication across departments and a high level of guest satisfaction.

This system not only reduces operational errors but also improves staff productivity and enhances the guest experience. It's a future-proof solution that supports growth, ensures better use of resources, and positions the hotel to meet modern demands in a highly competitive industry.

With a profitability index of **1.23**, the system offers a solid return on investment, making it a practical and impactful choice for the hotel.

8.0 Requirement Analysis

8.1 Current Business Process

Currently, room bookings at the hotel are handled manually through phone calls, emails, or walk-ins at the front desk. Staff must check room availability by consulting a paper log or a basic spreadsheet system. This increases the chances of errors associated with manual data update with each new booking, cancellation or modification. When a booking is canceled or rescheduled, staff need to manually adjust the availability, increasing the chances of double bookings or inaccurate records of available rooms.There is no common, up-to-date system for managing available rooms in order to avoid the conflicts and confusion with room assignment.

At the moment, the hotel organization employs a conventional way of documenting the check-in and check-out of the guests. Upon checking in, guests' particulars are recorded manually on a Written Registry, either on paper or on a spreadsheet or both and the check out date is noted and checked manually.When a guest checks out, the room status is updated by marking it as available, but this can lead to delays in preparing rooms for new guests, as the process relies heavily on staff intervention Furthermore, maintaining guest information manually increases the risk of data entry errors and inconsistent record-keeping, making it challenging to track guest history or preferences accurately. This approach also reduces the efficiency of the staff and the effectiveness of response to guests' information necessary for a quicker provision of services.

Data collection is done by management usually through paper and pen for compiling occupancy reports, revenue summaries among other reports and bookings trends results in mostly are tabulated using spreadsheets. The problem with this process is that it is usually slow and therefore reports can be prepared after long intervals while the data collected may be more recent. Most of the analytical information is past oriented; having access to timely reports is difficult, thus, limiting management's ability to make decisions before certain events, such as changes in prices, promotional campaigns, or resources, occur. Additionally, it is likely to have mathematical calculation mistakes when data are entered manually, therefore producing misleading analysis and decision-making results.

As a result of the present work, several cases have been pointed out where manual methods result in inefficiency as well as mistakes. These issues are poorly attended to by current

systems, and the proposed Hotel Booking System which runs on a database environment seeks to eradicate disparities in issues of booking, checking, charging, and bookkeeping. This will lessen on operation time, increase on accuracy of information and give hotel management real time information or to do better the services and the operations of the hotel.

Summary: Current Business Process

Room Booking Process:

- Managed manually via phone calls, emails, or walk-ins.
- Availability checked using paper logs or spreadsheets, increasing error risks.
- Manual updates for cancellations or rescheduling lead to double bookings and inaccurate records.

Guest Check-In/Check-Out Process:

- Guest details recorded manually in a registry or spreadsheet.
- Room status updates depend on staff intervention, causing delays.
- Manual record-keeping risks data entry errors and inconsistent tracking of guest history/preferences.

Data Collection and Reporting:

- Reports prepared using paper and pen, later tabulated in spreadsheets.
- Reports are slow to produce, with analytical data often outdated.
- Manual calculations can lead to errors and inaccurate decision-making.

Identified Challenges:

- Inefficiencies in booking, check-in/out processes, and data management.
- Manual systems hinder operational accuracy and timeliness.
- Lack of real-time insights limits management's ability to make proactive decisions.

Proposed Solution:

- A database-driven Hotel Booking System to eliminate inefficiencies.
- Enhances operation time, accuracy, and real-time management of bookings and services.

8.2 Functional and Non-functional Requirements

8.2.1 Functional Requirements

The proposed Hotel Booking System will incorporate essential **functional requirements** to streamline booking, guest information management, payment tracking, and reporting processes.Some popular features are **Room Booking Management** where members can schedule rooms, check available accommodations, manage booking cancellation as well as modify booking schedules. Additionally, the system will support **Guest Information Management**, storing detailed guest records that can be retrieved and updated as needed, along with a **Check-In and Check-Out Processing** feature to efficiently manage room occupancy status in real time.

Other key functions include **Room Availability Tracking** to prevent double bookings by updating room status in real time, **Reporting and Analytics Tools** to generate data-driven insights, such as occupancy rates and revenue trends, aiding management in making informed business decisions. The system will also feature **User Management** to support various user roles with role-specific access rights, as well as **Notifications and Alerts** to inform staff about upcoming check-ins, check-outs, and other critical updates.

Summary: Functional Requirements that will be included in the system:

- **Room Booking Management:** Scheduling rooms, checking availability, managing cancellations, and modifying bookings.
- **Guest Information Management:** Storing, retrieving, updating detailed guest records.
- **Check-In and Check-Out Processing:** Real-time management of room occupancy status.
- **Room Availability Tracking:** Preventing double bookings through real-time updates of room status.
- **Reporting and Analytics:** Providing insights into occupancy rates and revenue trends.
- **User Management:** Supporting role-specific access rights for different user roles.
- **Notifications and Alerts:** Informing staff about upcoming check-ins, check-outs, and other critical updates.

8.2.2 Non-functional Requirements

As far as non-functional requirements are concerned, the focus in this system is on the usability and performance, offering explicit and uncluttered interface, and requiring not more than 3 seconds to complete the routine tasks. It will provide future expansion of bookings and room inventory and maintain data accuracy and 99.9% availability with a backup to avoid data loss. Privacy will be the major concern of the application, including secure communication, user identification, and authorization with the purpose of protecting guest and payment data according to contemporary legislation.

The other non-functional requirements include that the system must be easy to maintain to allow easy identification of faults and to permit easy updates.The information has to be accurate to avoid customers being booked at the same time as another customer and for proper charging of customers. The system must be accessible,it means that it has to fulfill the ADA2009 standards for people with disabilities. Last but not least, in order to avoid the loss of critical data a backup and recovery will be conducted to ensure that if servers shut down, data can be retrieved within a reasonable amount of time. All these functional requirements and nonfunctional requirements will enhance the effective and secure running of the system in regard to efficiency, reliability and expandability, hence a strong boost for the Hotels ability to offer an effective solution that will make it easier for the customer or employee to use the system.

Summary: Non Functional Requirements that will be included in the system:

- **Usability and Performance:** Intuitive interface, completion of routine tasks within 3 seconds.
- **Expandability:** Supports future expansion for bookings and room inventory.
- **Data Accuracy and Availability:** Ensures 99.9% uptime with data backups to prevent data loss.
- **Privacy and Security**: Secure communication, user identification, and authorization to protect guest and payment data.
- **Maintainability**: Easy fault identification and system updates.
- **Accessibility:** Compliance with ADA2009 standards for people with disabilities.
- **Backup and Recovery:** Data recovery ensured within a reasonable timeframe in case of server shutdowns.

8.3 Data Flow Diagram (DFD)

8.3.1 Context Diagram

Figure above shows the context diagram of the hotel booking management system

8.3.2 Diagram 0

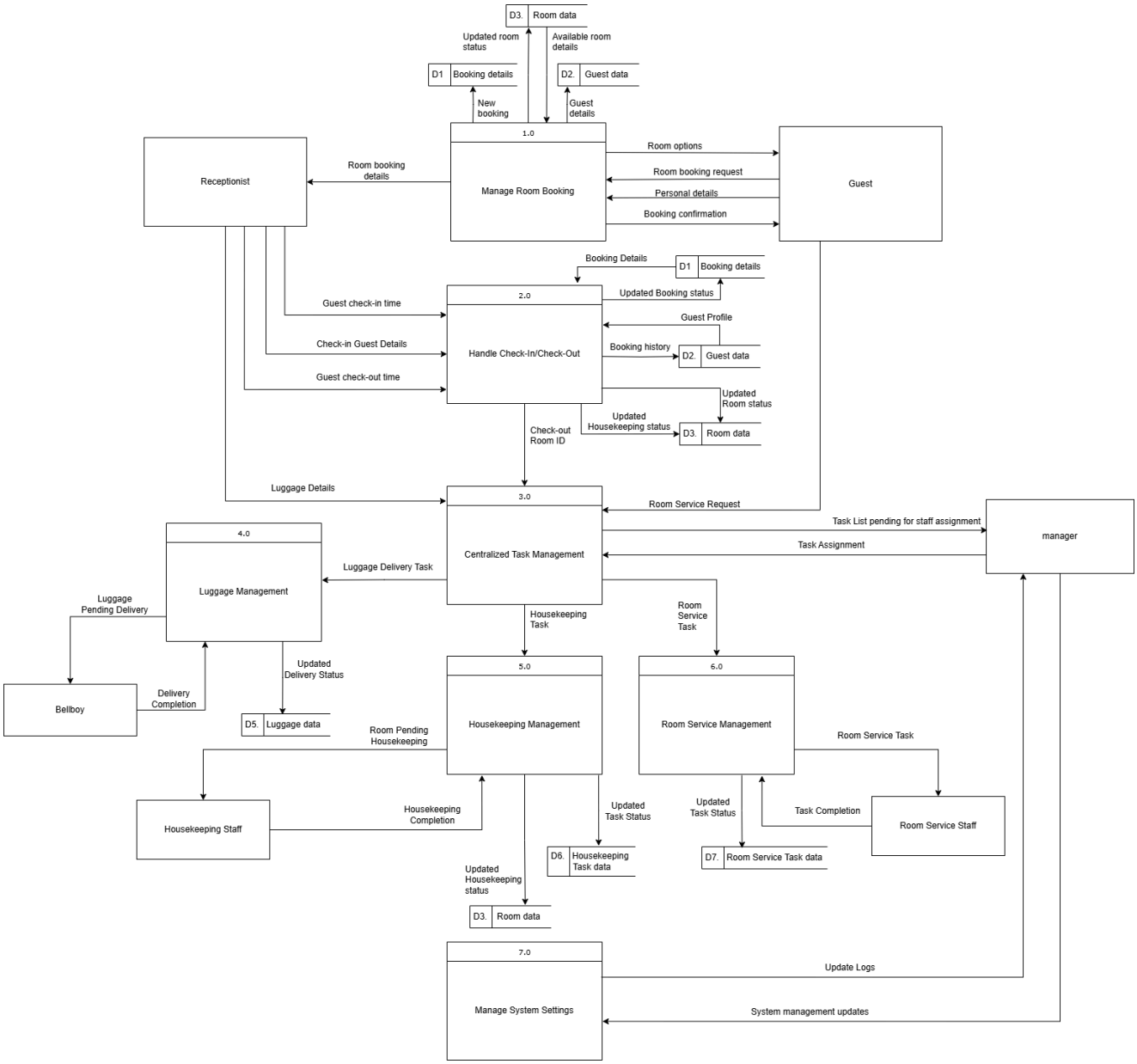
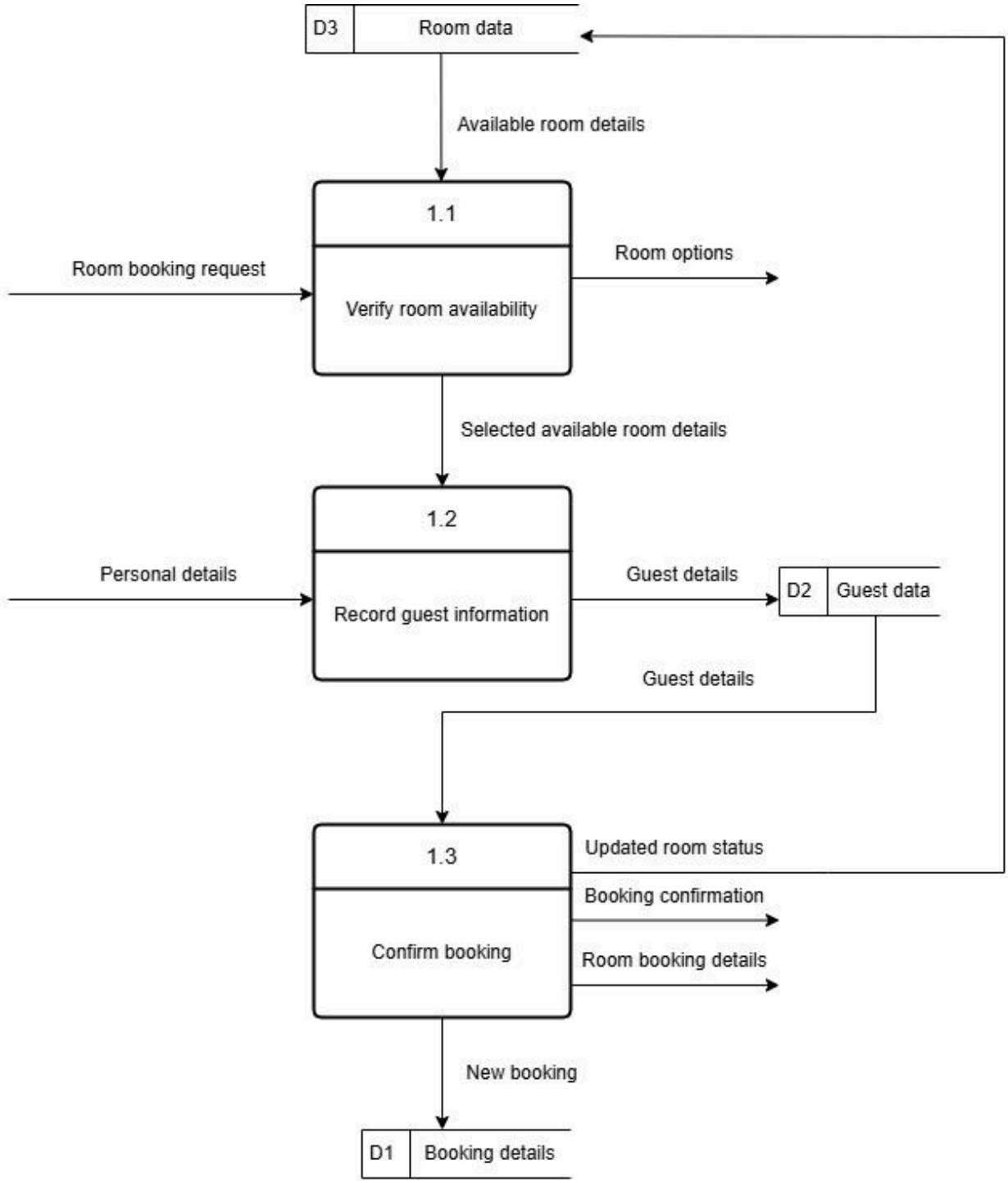


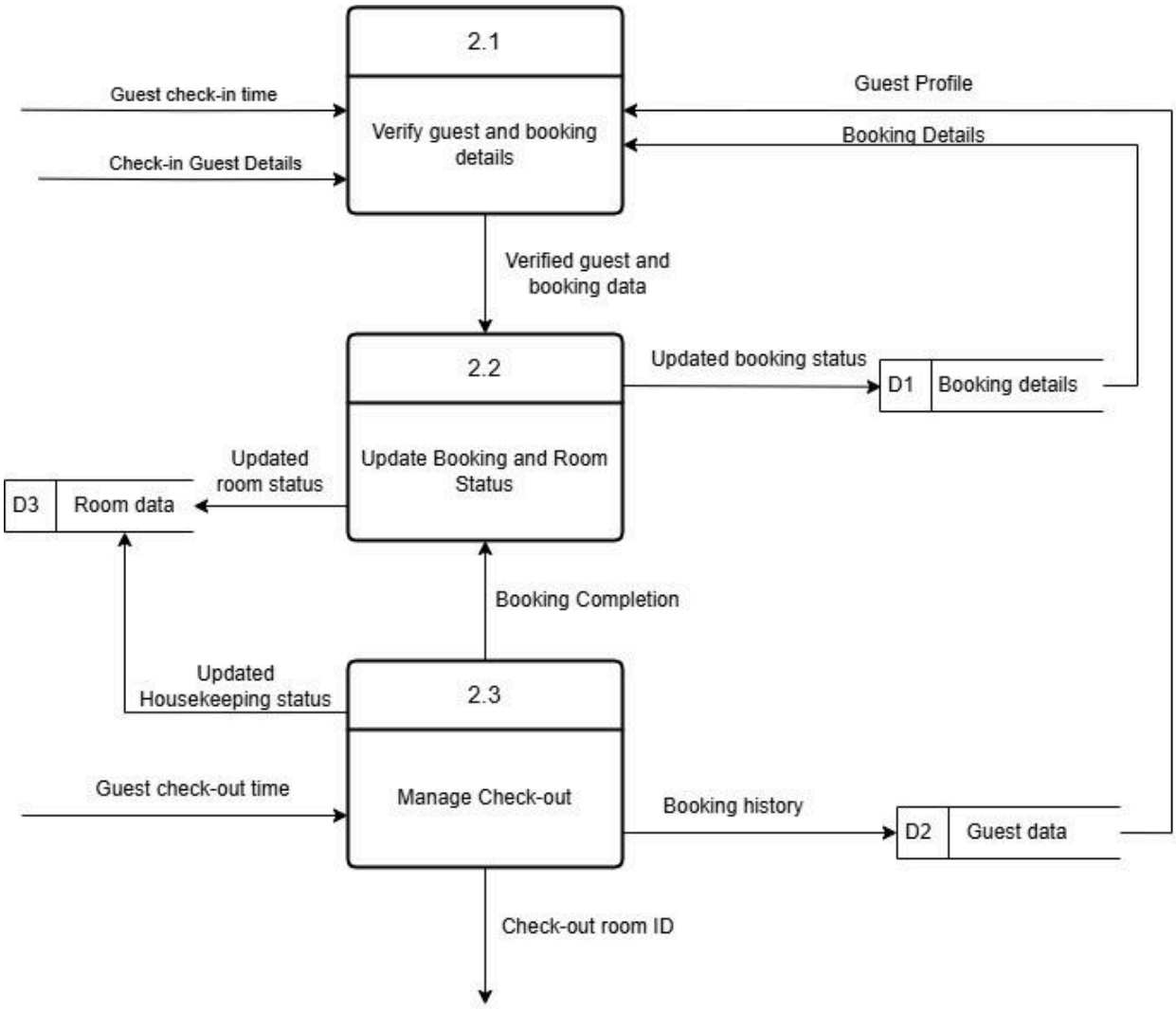
Figure above shows the diagram 0 of the hotel booking management system

8.3.3 Child Diagram

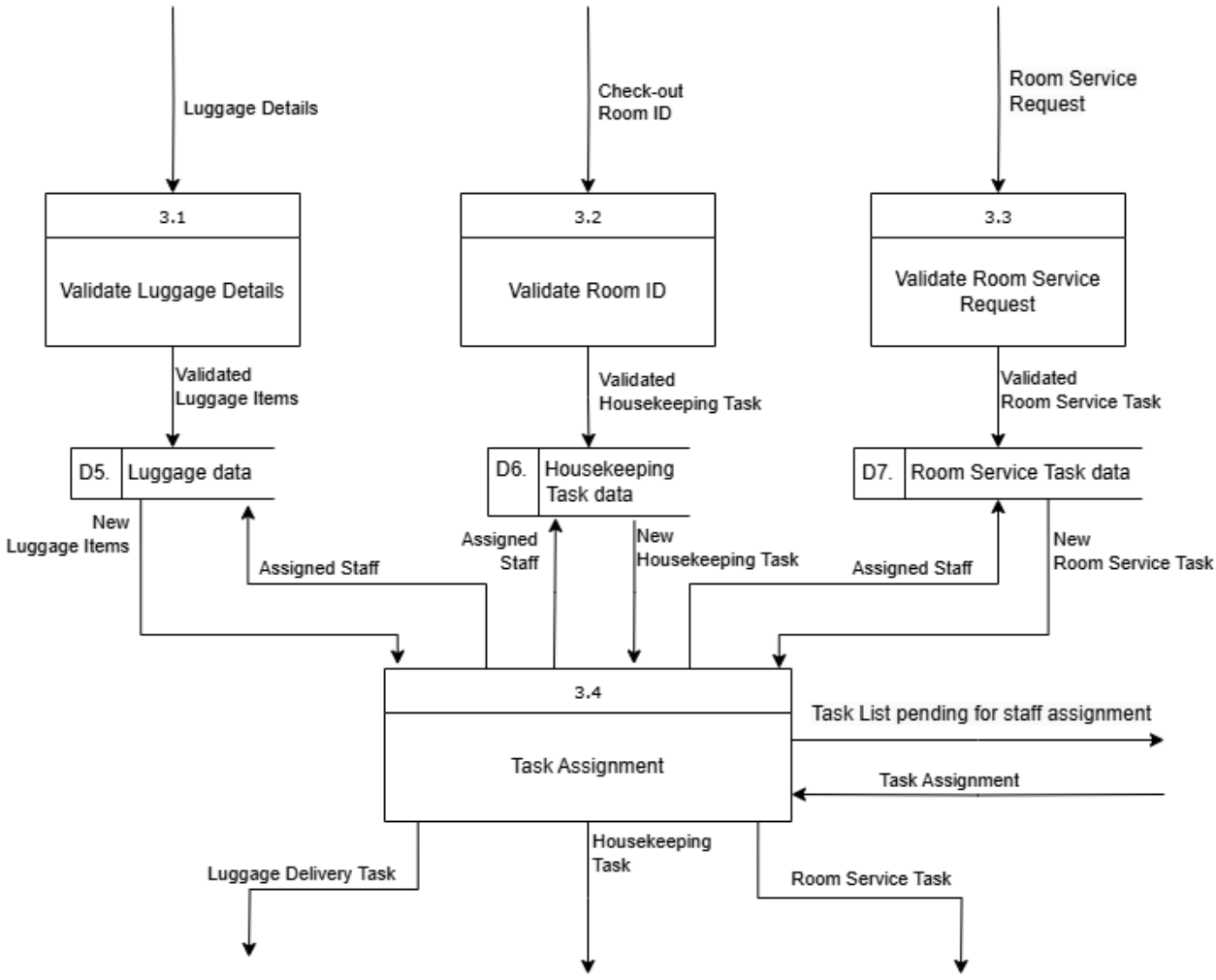
Process 1: Manage Room Booking



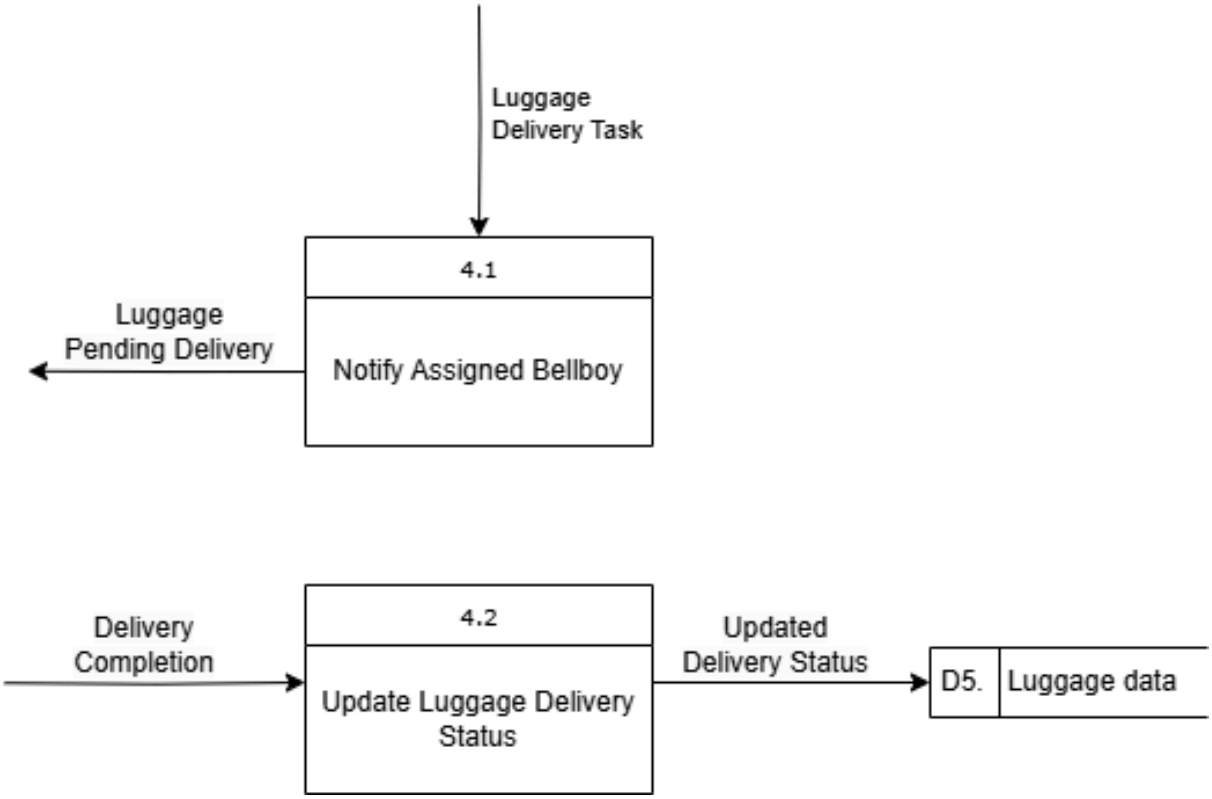
Process 2: Handle Check-In/Check-Out



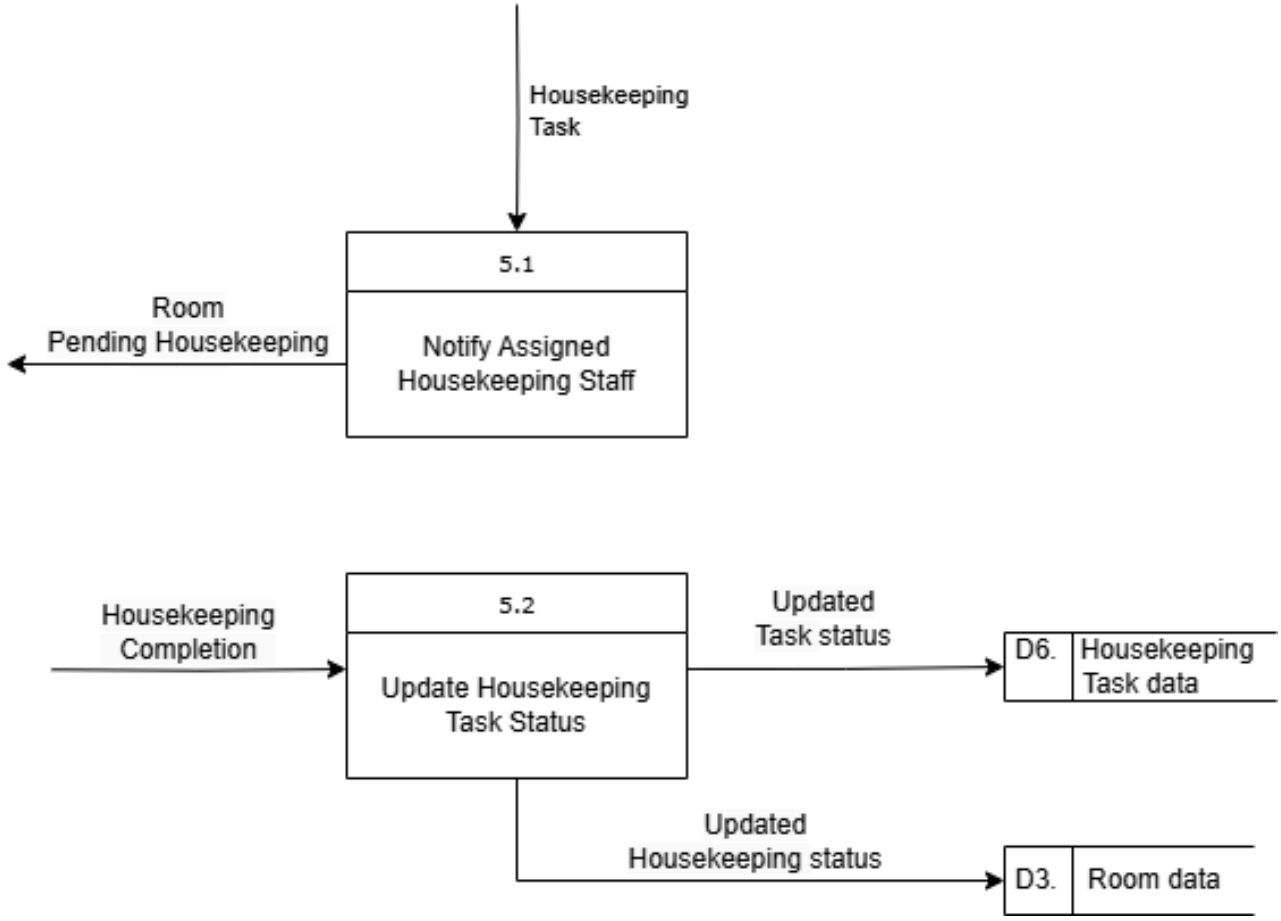
Process 3: Centralized Task Management



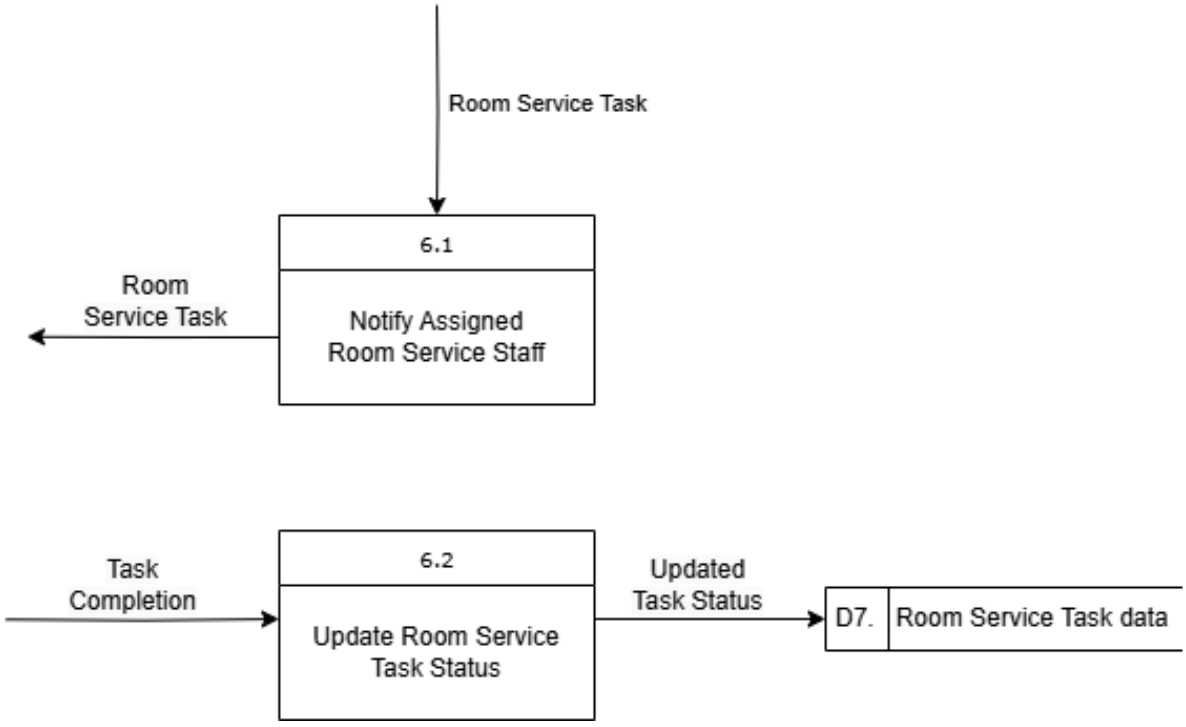
Process 4: Luggage Management



Process 5: Housekeeping Management



Process 6: Room Service Management



9.0 Proposed Business Rules

1. **Room Availability**
    - Rooms can only be booked if they're marked as "Available."
    - No double bookings allowed—the system checks before confirming.
    - Once a guest checks in, the room status switches to "Occupied." After they check out, it's set to "Available" again.

2. **Changing Bookings**
    - Guests can update their bookings (like dates or room types) up to 24 hours before check-in, as long as there's availability.
    - Canceling within 24 hours of check-in? There might be a fee.

3. **Protecting Guest Info**
    - All personal data is securely stored and follows privacy laws (like GDPR).
    - Guests can ask for their data to be deleted after their stay, but financial records will be kept for reporting.

4. **Staff Tasks**
    - Housekeeper gets cleaning assignments based on room statuses (like "Needs Cleaning" after a check-out).
    - Bellboys get luggage delivery tasks using QR codes, and they confirm when it's done.

5. **Guest Requests**
    - Guests can make service requests using the mobile app or by asking the receptionist.
    - Requests should be acknowledged by room service staff within 10 minutes.

6. **Tracking Luggage**
    - QR codes on luggage link it to the guest's room in the system.
    - Deliveries are logged in the system to ensure nothing gets lost.

7. **Notifications and Alerts**
    - Staff get real-time updates for tasks like cleaning or check-outs.
    - If something major goes wrong (like a system error), managers are notified immediately.

8. **Access Control**
    - Everyone gets system access based on their role
    - Only authorized staff can do high-level tasks like adding room types or changing prices.

10.0 Data & transaction requirement

10.1 Data requirements

1. Guest

- Guest ID: A unique identifier for each guest. (Primary Key)
- Name: Full name of the guest.
- Contact Information: Email address and phone number for communication.
- Booking History: Past bookings, check-in/check-out records.

2. Room

- Room ID: Unique identifier for each room. (Primary Key)
- Room Type: Type of room (e.g., single, double, suite).
- Room Status: Current status of the room (available, reserved, occupied, under maintenance).
- Room rate: Price per night based on room type and season.
- Housekeeping Status (Clean, Dirty, Under Maintenance)

3. Booking

- Booking ID: Unique identifier for each booking. (Primary Key)
- Guest ID (Foreign Key to Guest)
- Room ID (Foreign Key to Room)
- Receptionist ID (Foreign Key to Staff for Receptionist role)
- Booking Dates: Check-in and check-out dates.
- Booking Status: Confirmed, Pending, Cancelled.

4. Staff

- Staff ID: Unique identifier for each staff member. (Primary Key)
- Staff Name
- Role: Receptionist, Bellboy, Housekeeper, Room Service Staff.
- Contact Information: Phone number and email address.
- Shift Details: Work schedules and availability of staff. (Morning, Evening, Night)

- Manager ID (Foreign Key to Manager)

  4.1 Receptionist

  - Desk Location (Specific desk area for managing front desk operations)

  4.2 Housekeeper

  - Assigned Floor (Floor or area assigned for cleaning tasks)

  4.3 Bellboy

  - Shift Zone (Specific area or floors assigned to the bellboy)
  - Luggage Capacity (Maximum number of luggage items they can handle at a time)

  4.4 Room service staff

  - Service Floor (Floor or area assigned for room service tasks)

5. Manager

- Manager ID: Unique identifier for each manager. (Primary Key)
- Name: Manager's full name.
- Contact Information: Email and phone number.
- Task Assignments: Monitor staff performance, review and approve staff task assignments and reassign tasks if necessary.

6. Housekeeping Task

- Task ID: Unique identifier for each housekeeping task. (Primary Key)
- Room ID (Foreign Key to Room)
- Task Description: Details of the housekeeping task (e.g., cleaning, maintenance).
- Task Status: Status of the task (pending, completed).

7. Luggage Tracking

- Luggage ID: Unique QR code/identifier assigned to each piece of luggage. (Primary Key)
- Room ID (Foreign Key to Room)
- Delivery Status: Status of the luggage (in transit, delivered).

8. Service Request

- Request ID: Unique identifier for each service request. (Primary Key)

- Room ID (Foreign Key to Room)
- Request type: Type of service (e.g., Food delivery, maintenance issue).
- Request Description: Details of the service request.
- Request Status: Pending, in progress, or completed.

10.2 Transaction requirement

The Hotel Booking System will support various transaction types to handle data entry, updates, deletions, and queries.It will ensure efficient and reliable data management for core hotel operations. Data entry transactions are essential for recording new information in the system, such as room bookings, guest registrations, and payment details. When a new booking is created, the system will capture essential information, including room details, guest information, check-in/check-out dates, and payment method, validating room availability before finalizing the booking. As with the registration of guests, guest details will be entered and recorded the same when checking in, as will invoicing and records of each payment transaction.

Data update transactions will allow staff to modify existing records as needed, ensuring the system reflects any changes in booking details or guest status. For example, the system will support booking modifications, allowing updates to check-in/check-out dates or room types if necessary. During check-out, the transaction will update the room's availability status and complete any pending payments, marking the booking as finalized. Additionally, adjustments to payments, such as refunds or additional charges can be processed in real time, ensuring the financial records remain current.

Data delete transactions allow the system to remove records that are no longer needed, such as outdated or canceled bookings. For example, when a booking is canceled, the system will delete the booking record and automatically update the room status to show it as available for new reservations. Additionally, if a guest requests that their personal information be removed from the system, it can be securely deleted, while keeping important historical data (like revenue records) intact for reporting purposes.

Lastly, data queries will offer quick access to information for hotel staff and management. The system will enable queries on room availability, allowing staff to check availability by date, room type, and occupancy for easy booking management. Guest reference options are going to help staff search for guests by their name or reservation number to help with the check-in and implement more targeted guest experiences. For the repeat customers, the booking history queries will let the customers have the records of their bookings, their choices and even their previous bills which in one way or the other can improve or personalise the guest experience.

These transaction requirements ensure that the Hotel Booking System will manage data accurately and securely, supporting smooth hotel operations and providing real-time information

to both staff and management.

Summary: 10.2 Transaction Requirements

- Data Entry Transactions:
    - Record new information such as room bookings, guest registrations, and payment details.
    - Validate room availability before finalizing bookings.
    - Capture essential details (room, guest, check-in/check-out dates, payment method).
- Data Update Transactions:
    - Modify records to reflect changes in bookings or guest status.
    - Support updates to check-in/check-out dates, room types, and financial adjustments (e.g., refunds, additional charges).
    - Update room availability during check-out and finalize bookings.
- Data Delete Transactions:
    - Remove outdated or canceled bookings and update room availability.
    - Securely delete personal guest information upon request while retaining historical data for reporting.
- Data Queries:
    - Enable quick access to room availability based on date, type, and occupancy.
    - Allow staff to search guest information by name or reservation number.
    - Provide booking history to personalize guest experiences and improve services.
- Overall System Benefits:
    - Accurate and secure data management.
    - Real-time information for smooth operations and decision-making.

11.0 Database conceptual design

11.1 Updated business rules

The use of the Proposed Hotel Booking System on the overall hotel operations is that all operations will be done electronically making it easier. Real time room availability will be used to avoid occurrences of room duplication by signaling if a room is already booked. Customers will have access to a special interface where they will be able to change or cancel a given slot, with the changes appearing in the program's database immediately. Everyone will check in and check out which will have automatic change of room status whilst entire transactions will be written through electronically to reduce chances of errors.

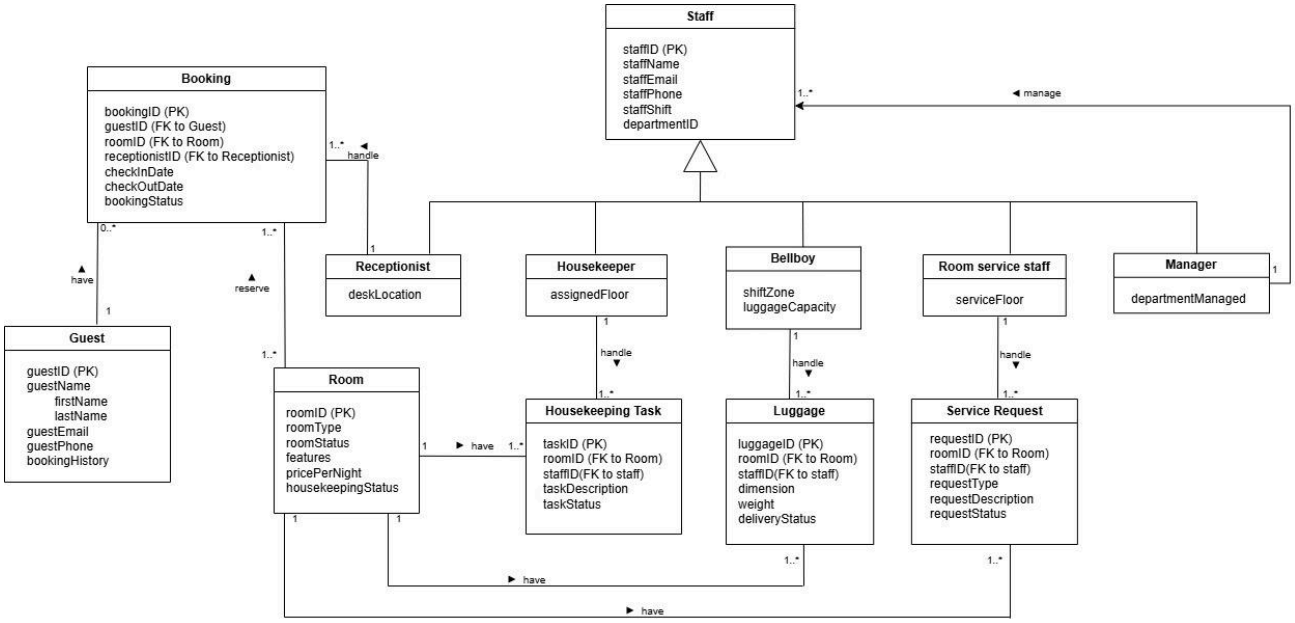
Guest data such as their preferences and history will be safely saved within the system to ensure that guest experience is personalized and there is quick decision making. Cleaning assignments will be made with a real time display of room status, while the ability to input task completion status will help in fast room turn-over. When it comes to luggage handling, the use of the QR code that will be linked with the guest profiles will be used effectively to increase efficiency by eliminating delays and mistakes. Implementation of automatic reports will enhance the real time reporting system for occupancy rate revenue and guest satisfaction to be availed by the management through automated reports for decision making. Implementing the role-based access control policy, staff will only be able to get to those functions that are necessary for their line of work and particular data will remain safe from prying eyes.

Furthermore, there will be the possibility to send real time notifications to the staff, when it is about new bookings or check-outs soon. It will also monitor important errors such as when the server is down and alert the management. The data compliance and security will be maintained with the help of installed encryption, data access log, practices regarding privacy laws including GDPR. That way, data will be backed up and there will be little loss in the instance of technical glitches. These modified business rules will address key issues of hotel growth, customer benefits, and decision making.

Summary: 11.1 Updated Business Rules

- Electronic Operations:
    - All hotel operations will be conducted electronically for efficiency.
    - Real-time room availability ensures no duplication in bookings.
- Customer Interface:
    - Customers can modify or cancel bookings, with updates reflected immediately in the system.
- Automated Room Management:
    - Check-in/check-out processes automatically update room status.
    - Cleaning tasks linked to real-time room status for quicker turn-over.
- Guest Data Management:
    - Guest preferences and history saved for personalized experiences and quick decision-making.
- Luggage Handling:
    - QR codes linked to guest profiles streamline luggage management and reduce errors.
- Automated Reporting:
    - Real-time reports for occupancy rates, revenue, and guest satisfaction aid decision-making.
- Role-Based Access Control:
    - Staff access restricted to functions relevant to their roles for enhanced security.
- Real-Time Notifications:
    - Notifications sent for new bookings, upcoming check-outs, and server errors.
- Data Security and Compliance:
    - Encryption, data logs, and GDPR compliance protect guest and hotel data.
    - Regular backups minimize data loss during technical issues.

11.2 ERD


Staff
staffID (PK)
staffName
staffEmail
staffPhone
staffShift
departmentID
manage
Booking
bookingID (PK)
guestID (FK to Guest)
roomID (FK to Room)
receptionistID (FK to Receptionist)
checkInDate
checkOutDate
bookingStatus
handle
have
reserve
Guest
guestID (PK)
guestName
firstName
lastName
guestEmail
guestPhone
bookingHistory
Receptionist
deskLocation
Housekeeper
assignedFloor
Bellboy
shiftZone
luggageCapacity
Room service staff
serviceFloor
Manager
departmentManaged
Room
roomID (PK)
roomType
roomStatus
features
pricePerNight
housekeepingStatus
Housekeeping Task
taskID (PK)
roomID (FK to Room)
staffID(FK to staff)
taskDescription
taskStatus
Luggage
luggageID (PK)
roomID (FK to Room)
staffID(FK to staff)
dimension
weight
deliveryStatus
Service Request
requestID (PK)
roomID (FK to Room)
staffID(FK to staff)
requestType
requestDescription
requestStatus

12.0 Data dictionary

12.1 Description of Entity

ENTITY NAME	DESCRIPTION
Guest	Represents customers who book and stay in the hotel. Stores their personal and booking-related information.
Room	Represents rooms available in the hotel, including their type, status, and associated features.
Booking	Captures booking information such as guest, room allocation, check-in/out dates, and receptionist details.
Staff	Represents employees involved in various operations like reception, housekeeping, and luggage management.
Housekeeping Task	Tracks tasks assigned to staff, such as room cleaning or luggage delivery, and their current status.
Service Request	Captures guest-initiated requests, such as room service or maintenance, and tracks their resolution status.
Luggage	Tracks luggage linked to guests, using QR codes for efficient delivery and monitoring.
Receptionist	Handles guest bookings and maintains desk operations.
Bellboy	Responsible for handling luggage with specified capacity and shift zones.
Manager	Oversees hotel departments and staff.
Housekeeper	Assigned to maintain cleanliness and order on specific floors.

12.2 Description of Relationship

Entity 1	**Multiplicity**	**Relationship**	**Multiplicity**	**Entity 2**
Guest	1	Makes	0…*	Booking
Booking	1	reserves	1	Room
Staff	1	manages	*	Manager
Booking	0…*	handled by	1	Receptionist
Guest	1	Requests	*	ServiceRequest
Guest	1	Tracks	*	Luggage
Room	1	LinkedTo	0...*	ServiceRequest
Room	1	assigned to	1..*	Housekeeping Task
Luggage	1	handled by	1..*	Bellboy

12.3 Description of Attributes

Entity	Attribute Name	Description	Data Type	Constraints
Guest	GuestID	Unique identifier for each guest.	Integer	Primary Key
	Name	Full name of the guest.	String	Not Null
	guestEmail	Email address for communication.	String	Unique, Not Null
	guestPhone	Phone number for contact.	String	Unique, Not Null
	BookingHistory	Records of past bookings for the guest.	String	Optional
Room	RoomID	Unique identifier for each room.	Integer	Primary Key
	Type	Type of room	String	Not Null
	Status	Current status of the room	String	Not Null
	housekeepingStatus	Indicates cleaning status of the room.	String	Not Null
	features	Amenities available in the room.	String	Optional
	pricePerNight	Cost per night for the room.	Float	Positive
Staff	StaffID	Unique identifier for each staff member	Integer	Primary Key
	Name	Full name of the staff member.	String	Not Null
	staffEmail	Email address of the staff.	String	Not Null
	staffPhone	Phone number of the staff.	String	Unique, Not Null
	Shift	Assigned working	String	Not Null

		shift		
	departmentID	Department managed by the staff.	Integer	Foreign Key
Receptionist	deskLocation	Location of the receptionist's desk.	String	Not Null
Booking	BookingID	Unique identifier for each booking.	Integer	Primary Key
	GuestID	Links the booking to the guest who made it.	Integer	Foreign Key (Guest)
	RoomID	Links the booking to the allocated room.	Integer	Foreign Key (Room)
	CheckIn	Check-in date for the booking.	Date	Not Null
	CheckOut	Check-out date for the booking.	Date	Not Null
	Status	Current booking status	String	Not Null
	receptionistID	Links the booking to the receptionist handling it.	Integer	Foreign Key (Receptionist)
Housekeeper	assignedFloor	Floor assigned to the housekeeper.	Integer	Not Null
Bellboy	shiftZone	Shift area handled by the bellboy.	String	Not Null
	luggageCapacity	Number of luggage pieces the bellboy can handle.	Integer	Not Null
Manager	departmentManaged	Department supervised by the manager.	String	Not Null
Housekeeping Task	TaskID	Unique identifier for each task.	Integer	Primary Key
	taskDescription	Description of the housekeeping task.	String	Not Null

	roomID	Links the task to the relevant room.	Integer	Foreign Key (Room)
	staffID	Links the task to the assigned housekeeper.	Integer	Foreign Key (Staff)
	taskStatus	Current status of the task (e.g., Completed).	string	Not Null
ServiceRequest	RequestID	Unique identifier for each service request.	Integer	Primary Key
	RoomID	Links the request to a specific room.	Integer	Foreign Key (Room)
	requestType	Type of service requested	String	Not Null
	requestStatus	Current status of the request	String	Not Null
	staffID	Links the request to the staff assigned to handle it.	Integer	Foreign Key (Staff)
	requestDescription	Description of the service request.	string	Not Null
Luggage	LuggageID	Unique identifier for each piece of luggage.	String	Primary Key
	staffID	Links the luggage handling staff.	Integer	Foreign Key (Staff)
	RoomID	Links the luggage to the guest's assigned room.	Integer	Foreign Key (Room)
	DeliveryStatus	Status of luggage delivery	String	Not Null
	dimension	Dimensions of the luggage.	String	Optional
	weight	Weight of the luggage.	Float	Positive

13.0 SQL Statement

```
-- Create table for Staff (Superclass)
CREATE TABLE Staff (
    staffID INT,
    staffName VARCHAR(50) NOT NULL,
    staffEmail VARCHAR(100),
    staffPhone VARCHAR(15),
    staffShift VARCHAR(20),
    departmentID INT
);

-- Create table for Receptionist
CREATE TABLE Receptionist (
    staffID INT,
    deskLocation VARCHAR(50)
);

-- Create table for Housekeeper
CREATE TABLE Housekeeper (
    staffID INT,
    assignedFloor INT
);

-- Create table for Bellboy
CREATE TABLE Bellboy (
    staffID INT,
    shiftZone VARCHAR(50),
    luggageCapacity INT
);
```

```
-- Create table for RoomServiceStaff
CREATE TABLE RoomServiceStaff (
    staffID INT,
    serviceFloor INT
);

-- Create table for Manager
CREATE TABLE Manager (
    staffID INT,
    departmentManaged VARCHAR(50)
);

-- Create table for Room
CREATE TABLE Room (
    roomID INT,
    roomType VARCHAR(20),
    roomStatus VARCHAR(20),
    features TEXT,
    pricePerNight DECIMAL(10, 2),
    housekeepingStatus VARCHAR(20)
);

-- Create table for Guest
CREATE TABLE Guest (
    guestID INT,
    guestName VARCHAR(100),
    guestEmail VARCHAR(100),
    guestPhone VARCHAR(15),
    bookingHistory TEXT
);
```

```sql
-- Create table for Booking
CREATE TABLE Booking (
    bookingID INT,
    guestID INT,
    roomID INT,
    receptionistID INT,
    checkInDate DATE,
    checkOutDate DATE,
    bookingStatus VARCHAR(20)
);

-- Create table for Housekeeping Task
CREATE TABLE HousekeepingTask (
    taskID INT,
    roomID INT,
    staffID INT,
    taskDescription TEXT,
    taskStatus VARCHAR(20)
);

-- Create table for Luggage
CREATE TABLE Luggage (
    luggageID INT,
    roomID INT,
    staffID INT,
    dimension VARCHAR(50),
    weight DECIMAL(5, 2),
    deliveryStatus VARCHAR(20)
);
```

```sql
-- Create table for Service Request
CREATE TABLE ServiceRequest (
    requestID INT,
    roomID INT,
    staffID INT,
    requestType VARCHAR(50),
    requestDescription TEXT,
    requestStatus VARCHAR(20)
);
```

```
-- Add primary key constraints

ALTER TABLE Staff
ADD CONSTRAINT PK_Staff PRIMARY KEY (staffID);

ALTER TABLE Receptionist
ADD CONSTRAINT PK_Receptionist PRIMARY KEY (staffID);

ALTER TABLE Housekeeper
ADD CONSTRAINT PK_Housekeeper PRIMARY KEY (staffID);

ALTER TABLE Bellboy
ADD CONSTRAINT PK_Bellboy PRIMARY KEY (staffID);

ALTER TABLE RoomServiceStaff
ADD CONSTRAINT PK_RoomServiceStaff PRIMARY KEY (staffID);

ALTER TABLE Manager
ADD CONSTRAINT PK_Manager PRIMARY KEY (staffID);

ALTER TABLE Room
ADD CONSTRAINT PK_Room PRIMARY KEY (roomID);

ALTER TABLE Guest
ADD CONSTRAINT PK_Guest PRIMARY KEY (guestID);

ALTER TABLE Booking
ADD CONSTRAINT PK_Booking PRIMARY KEY (bookingID);

ALTER TABLE HousekeepingTask
ADD CONSTRAINT PK_HousekeepingTask PRIMARY KEY (taskID);
```

```
ALTER TABLE Luggage
ADD CONSTRAINT PK_Luggage PRIMARY KEY (luggageID);

ALTER TABLE ServiceRequest
ADD CONSTRAINT PK_ServiceRequest PRIMARY KEY (requestID);
```

```
-- Add foreign key constraints

ALTER TABLE Receptionist
ADD CONSTRAINT FK_Receptionist_Staff FOREIGN KEY (staffID)
REFERENCES Staff(staffID);

ALTER TABLE Housekeeper
ADD CONSTRAINT FK_Housekeeper_Staff FOREIGN KEY (staffID)
REFERENCES Staff(staffID);

ALTER TABLE Bellboy
ADD CONSTRAINT FK_Bellboy_Staff FOREIGN KEY (staffID)
REFERENCES Staff(staffID);

ALTER TABLE RoomServiceStaff
ADD CONSTRAINT FK_RoomServiceStaff_Staff FOREIGN KEY (staffID)
REFERENCES Staff(staffID);

ALTER TABLE Manager
ADD CONSTRAINT FK_Manager_Staff FOREIGN KEY (staffID)
REFERENCES Staff(staffID);

ALTER TABLE Booking
ADD CONSTRAINT FK_Booking_Guest FOREIGN KEY (guestID)
REFERENCES Guest(guestID);

ALTER TABLE Booking
ADD CONSTRAINT FK_Booking_Room FOREIGN KEY (roomID)
REFERENCES Room(roomID);

ALTER TABLE Booking
```

```
ADD CONSTRAINT FK_Booking_Receptionist FOREIGN KEY (receptionistID)
REFERENCES Receptionist(staffID);

ALTER TABLE HousekeepingTask
ADD CONSTRAINT FK_HousekeepingTask_Room FOREIGN KEY (roomID)
REFERENCES Room(roomID);

ALTER TABLE HousekeepingTask
ADD CONSTRAINT fk_housekeepingtask_staff FOREIGN KEY (staffID)
REFERENCES Staff(staffID);

ALTER TABLE Luggage
ADD CONSTRAINT FK_Luggage_Room FOREIGN KEY (roomID)
REFERENCES Room(roomID);

ALTER TABLE Luggage
ADD CONSTRAINT fk_luggage_staff FOREIGN KEY (staffID)
REFERENCES Staff(staffID);

ALTER TABLE ServiceRequest
ADD CONSTRAINT FK_ServiceRequest_Room FOREIGN KEY (roomID)
REFERENCES Room(roomID);

ALTER TABLE ServiceRequest
ADD CONSTRAINT fk_servicerequest_staff FOREIGN KEY (staffID)
REFERENCES Staff(staffID);
```

```
-- Insert data into Staff
INSERT INTO Staff (staffID, staffName, staffEmail, staffPhone, staffShift, departmentID)
VALUES
(001, 'Alice Johnson', 'alice.johnson@example.com', '1234567890', 'Morning', 101),
(002, 'Bob Smith', 'bob.smith@example.com', '9876543210', 'Evening', 101),
(003, 'Charlie Brown', 'charlie.brown@example.com', '4561237890', 'Evening', 102),
(004, 'Diana Ross', 'diana.ross@example.com', '7894561230', 'Morning', 102),
(005, 'Dan Prince', 'dian.prince@example.com', '1234567893', 'Morning', 103),
(006, 'Edward Stark', 'edward.stark@example.com', '1234567894', 'Evening', 103),
(007, 'Michael Wong', 'michaelwong@example.com', '7894561235', 'Morning', 104),
(008, 'Ben Chia', 'benchia@example.com', '7894561630', 'Evening', 104),
(009, 'Jason See', 'jasonsee@example.com', '7894563430', 'Evening', 105),
(010, 'Kelvin Smith', 'kelvin.smith@example.com', '9876541230', 'Morning', 105);

-- Insert data into Receptionist
INSERT INTO Receptionist (staffID, deskLocation)
VALUES
(001, 'Front Desk A'),
(002, 'Front Desk B');

-- Insert data into Housekeeper
INSERT INTO Housekeeper (staffID, assignedFloor)
VALUES
(003, 1),
(004, 2);

-- Insert data into Bellboy
INSERT INTO Bellboy (staffID, shiftZone, luggageCapacity)
VALUES
(005, 'East Wing', 5),
(006, 'West Wing', 4);
```

```
-- Insert data into RoomServiceStaff
INSERT INTO RoomServiceStaff (staffID, serviceFloor)
VALUES
(007, 2),
(008, 1);

-- Insert data into Manager
INSERT INTO Manager (staffID, departmentManaged)
VALUES
(009, 'Operations'),
(010, 'Tasking');

-- Insert data into Room
INSERT INTO Room (roomID, roomType, roomStatus, features, pricePerNight,
housekeepingStatus)
VALUES
(101, 'Single', 'Available', 'Wi-Fi, TV', 100.00, 'Clean'),
(102, 'Double', 'Occupied', 'Wi-Fi, TV, Mini-bar', 150.00, 'Clean'),
(103, 'Suite', 'Under Maintenance', 'Wi-Fi, TV, Jacuzzi', 300.00, 'Under Maintenance'),
(104, 'Single', 'Available', 'Wi-Fi, TV', 100.00, 'Clean'),
(105, 'Single', 'Occupied', 'Wi-Fi, TV, Mini-bar', 120.00, 'Dirty'),
(106, 'Double', 'Occupied', 'Wi-Fi, TV', 130.00, 'Dirty');

-- Insert data into Guest
INSERT INTO Guest (guestID, guestName, guestEmail, guestPhone, bookingHistory)
VALUES
(001, 'John Doe', 'john.doe@example.com', '5551234567', NULL),
(002, 'Jane Smith', 'jane.smith@example.com', '5559876543', NULL),
(003, 'June Leong', 'june.leong@example.com', '5559812345', NULL);
```

```
-- Insert data into Booking
INSERT INTO Booking (bookingID, guestID, roomID, receptionistID, checkInDate,
checkOutDate, bookingStatus)
VALUES
(1, 001, 102, 001, '2024-12-20', '2024-12-25', 'Confirmed'),
(2, 002, 105, 001, '2024-12-21', '2024-12-23', 'Pending'),
(3, 003, 106, 002, '2024-12-21', '2024-12-23', 'Confirmed');

-- Insert data into Housekeeping Task
INSERT INTO HousekeepingTask (taskID, roomID, staffID, taskDescription, taskStatus)
VALUES
(1, 102, 003, 'Clean the room after guest check-out.', 'Pending'),
(2, 106, 004, 'Change the bedsheet.', 'Pending'),
(3, 105, 003, 'Clean the room after guest check-out.', 'Pending');

-- Insert data into Luggage
INSERT INTO Luggage (luggageID, roomID, staffID, dimension, weight, deliveryStatus)
VALUES
(5001, 102, 005, '24x16x10 inches', 25.50, 'Delivered'),
(5002, 105, 006, '24x16x10 inches', 30.20, 'Pending'),
(5003, 105, 005, '24x16x10 inches', 15.50, 'Delivered');

-- Insert data into Service Request
INSERT INTO ServiceRequest (requestID, roomID, staffID, requestType, requestDescription,
requestStatus)
VALUES
(1, 102, 007, 'Food Delivery', 'Deliver breakfast to Room 101', 'Pending'),
(2, 106, 007, 'Food Delivery', 'Deliver lunch to Room 102', 'Pending'),
(3, 105, 008, 'Food Delivery', 'Deliver breakfast to Room 103', 'Delivered');
```

-- View the Table

SELECT * FROM Staff;

```
mysql> SELECT  * FROM Staff;
+---------+--------------+---------------------------+------------+------------+--------------+
| staffID | staffName    | staffEmail                | staffPhone | staffShift | departmentID |
+---------+--------------+---------------------------+------------+------------+--------------+
|       1 | Alice Johnson | alice.johnson@example.com | 1234567890 | Morning    |          101 |
|       2 | Bob Smith     | bob.smith@example.com     | 9876543210 | Evening    |          101 |
|       3 | Charlie Brown | charlie.brown@example.com | 4561237890 | Evening    |          102 |
|       4 | Diana Ross    | diana.ross@example.com    | 7894561230 | Morning    |          102 |
|       5 | Dan Prince    | dian.prince@example.com   | 1234567893 | Morning    |          103 |
|       6 | Edward Stark  | edward.stark@example.com  | 1234567894 | Evening    |          103 |
|       7 | Michael Wong  | michaelwong@example.com   | 7894561235 | Morning    |          104 |
|       8 | Ben Chia      | benchia@example.com       | 7894561630 | Evening    |          104 |
|       9 | Jason See     | jasonsee@example.com      | 7894563430 | Evening    |          105 |
|      10 | Kelvin Smith  | kelvin.smith@example.com  | 9876541230 | Morning    |          105 |
+---------+--------------+---------------------------+------------+------------+--------------+
```

SELECT * FROM Receptionist;

```
mysql> SELECT  * FROM Receptionist;
+---------+--------------+
| staffID | deskLocation |
+---------+--------------+
|       1 | Front Desk A |
|       2 | Front Desk B |
+---------+--------------+
```

SELECT * FROM Housekeeper;

```
mysql> SELECT  * FROM Housekeeper;
+---------+---------------+
| staffID | assignedFloor |
+---------+---------------+
|       3 |             1 |
|       4 |             2 |
+---------+---------------+
```

SELECT * FROM Bellboy;

```
mysql> SELECT  * FROM Bellboy;
+---------+-----------+-----------------+
| staffID | shiftZone | luggageCapacity |
+---------+-----------+-----------------+
|       5 | East Wing |               5 |
|       6 | West Wing |               4 |
+---------+-----------+-----------------+
```

SELECT * FROM RoomServiceStaff;

```
mysql> SELECT  * FROM RoomServiceStaff;
+---------+--------------+
| staffID | serviceFloor |
+---------+--------------+
|       7 |            2 |
|       8 |            1 |
+---------+--------------+
```

SELECT * FROM Manager;

```
mysql> SELECT  * FROM Manager;
+---------+--------------------+
| staffID | departmentManaged  |
+---------+--------------------+
|       9 | Operations         |
|      10 | Tasking            |
+---------+--------------------+
```

SELECT * FROM Room;

```
mysql> SELECT  * FROM Room;
+--------+----------+------------------+--------------------+---------------+--------------------+
| roomID | roomType | roomStatus       | features           | pricePerNight | housekeepingStatus |
+--------+----------+------------------+--------------------+---------------+--------------------+
|    101 | Single   | Available        | Wi-Fi, TV          |        100.00 | Clean              |
|    102 | Double   | Occupied         | Wi-Fi, TV, Mini-bar |        150.00 | Clean              |
|    103 | Suite    | Under Maintenance | Wi-Fi, TV, Jacuzzi |        300.00 | Under Maintenance  |
|    104 | Single   | Available        | Wi-Fi, TV          |        100.00 | Clean              |
|    105 | Single   | Occupied         | Wi-Fi, TV, Mini-bar |        120.00 | Dirty              |
|    106 | Double   | Occupied         | Wi-Fi, TV          |        130.00 | Dirty              |
+--------+----------+------------------+--------------------+---------------+--------------------+
```

SELECT * FROM Guest;

```
mysql> SELECT  * FROM Guest;
+---------+------------+-----------------------+------------+----------------+
| guestID | guestName  | guestEmail            | guestPhone | bookingHistory |
+---------+------------+-----------------------+------------+----------------+
|       1 | John Doe   | john.doe@example.com  | 5551234567 | NULL           |
|       2 | Jane Smith | jane.smith@example.com | 5559876543 | NULL           |
|       3 | June Leong | june.leong@example.com | 5559812345 | NULL           |
+---------+------------+-----------------------+------------+----------------+
```

SELECT * FROM Booking;

```
mysql> SELECT  * FROM Booking;
+-----------+---------+--------+---------------+-------------+-------------+---------------+
| bookingID | guestID | roomID | receptionistID | checkInDate | checkOutDate | bookingStatus |
+-----------+---------+--------+---------------+-------------+-------------+---------------+
|         1 |       1 |    102 |             1 | 2024-12-20  | 2024-12-25   | Confirmed     |
|         2 |       2 |    105 |             1 | 2024-12-21  | 2024-12-23   | Pending       |
|         3 |       3 |    106 |             2 | 2024-12-21  | 2024-12-23   | Confirmed     |
+-----------+---------+--------+---------------+-------------+-------------+---------------+
```

SELECT * FROM HousekeepingTask;

```
mysql> SELECT  * FROM HousekeepingTask;
+--------+--------+---------+-----------------------------------------+------------+
| taskID | roomID | staffID | taskDescription                         | taskStatus |
+--------+--------+---------+-----------------------------------------+------------+
|      1 |    102 |       3 | Clean the room after guest check-out.   | Pending    |
|      2 |    106 |       4 | Change the bedsheet.                    | Pending    |
|      3 |    105 |       3 | Clean the room after guest check-out.   | Pending    |
+--------+--------+---------+-----------------------------------------+------------+
```

SELECT * FROM Luggage;

```
mysql> SELECT  * FROM Luggage;
+-----------+--------+---------+------------------+--------+----------------+
| luggageID | roomID | staffID | dimension        | weight | deliveryStatus |
+-----------+--------+---------+------------------+--------+----------------+
|      5001 |    102 |       5 | 24x16x10 inches  |  25.50 | Delivered      |
|      5002 |    105 |       6 | 24x16x10 inches  |  30.20 | Pending        |
|      5003 |    105 |       5 | 24x16x10 inches  |  15.50 | Delivered      |
+-----------+--------+---------+------------------+--------+----------------+
```

SELECT * FROM ServiceRequest;

```
mysql> SELECT  * FROM ServiceRequest;
+-----------+--------+---------+--------------+------------------------------+---------------+
| requestID | roomID | staffID | requestType  | requestDescription           | requestStatus |
+-----------+--------+---------+--------------+------------------------------+---------------+
|         1 |    102 |       7 | Food Delivery | Deliver breakfast to Room 101 | Pending       |
|         2 |    106 |       7 | Food Delivery | Deliver lunch to Room 102     | Pending       |
|         3 |    105 |       8 | Food Delivery | Deliver breakfast to Room 103 | Delivered     |
+-----------+--------+---------+--------------+------------------------------+---------------+
```

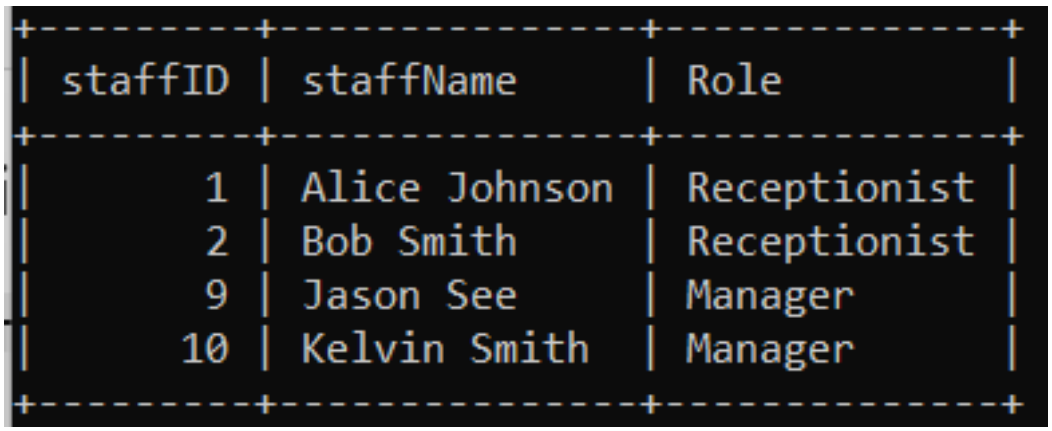
```sql
-- Different situation

-- Find Staff Members with Pending Tasks
SELECT S.staffID, S.staffName,
    CASE
        WHEN R.staffID IS NOT NULL THEN 'Receptionist'
        WHEN H.staffID IS NOT NULL THEN 'Housekeeper'
        WHEN B.staffID IS NOT NULL THEN 'Bellboy'
        WHEN RS.staffID IS NOT NULL THEN 'Room Service Staff'
        WHEN M.staffID IS NOT NULL THEN 'Manager'
        ELSE 'Unknown'
    END AS Role,
    COUNT(CASE
            WHEN HT.taskStatus = 'Pending' THEN HT.taskID
            WHEN LT.deliveryStatus = 'Pending' THEN LT.luggageID
            WHEN SR.requestStatus = 'Pending' THEN SR.requestID
        END) AS PendingTasks
FROM Staff S
LEFT JOIN Receptionist R ON S.staffID = R.staffID
LEFT JOIN Housekeeper H ON S.staffID = H.staffID
LEFT JOIN Bellboy B ON S.staffID = B.staffID
LEFT JOIN RoomServiceStaff RS ON S.staffID = RS.staffID
LEFT JOIN Manager M ON S.staffID = M.staffID
LEFT JOIN HousekeepingTask HT ON S.staffID = HT.staffID AND HT.taskStatus = 'Pending'
LEFT JOIN Luggage LT ON S.staffID = LT.staffID AND LT.deliveryStatus = 'Pending'
LEFT JOIN ServiceRequest SR ON S.staffID = SR.staffID AND SR.requestStatus = 'Pending'
GROUP BY S.staffID, S.staffName;
```

staffID	staffName	Role	PendingTasks
1	Alice Johnson	Receptionist	0
2	Bob Smith	Receptionist	0
3	Charlie Brown	Housekeeper	2
4	Diana Ross	Housekeeper	1
5	Dan Prince	Bellboy	0
6	Edward Stark	Bellboy	1
7	Michael Wong	Room Service Staff	2
8	Ben Chia	Room Service Staff	0
9	Jason See	Manager	0
10	Kelvin Smith	Manager	0

```
-- Find All Room Service Requests That Are Still Pending
SELECT SR.requestID, SR.requestType, SR.requestDescription,
    SR.requestStatus, R.roomID, R.roomType, S.staffName AS RoomServiceStaff
FROM ServiceRequest SR
INNER JOIN Room R ON SR.roomID = R.roomID
INNER JOIN Staff S ON SR.staffID = S.staffID
WHERE SR.requestStatus = 'Pending';
```

requestID	requestType	requestDescription	requestStatus	roomID	roomType	RoomServiceStaff
1	Food Delivery	Deliver breakfast to Room 101	Pending	102	Double	Michael Wong
2	Food Delivery	Deliver lunch to Room 102	Pending	106	Double	Michael Wong

```
-- Staff Who Have Not Been Assigned Any Tasks
SELECT S.staffID, S.staffName,
    CASE
        WHEN R.staffID IS NOT NULL THEN 'Receptionist'
        WHEN H.staffID IS NOT NULL THEN 'Housekeeper'
        WHEN B.staffID IS NOT NULL THEN 'Bellboy'
        WHEN RS.staffID IS NOT NULL THEN 'Room Service Staff'
        WHEN M.staffID IS NOT NULL THEN 'Manager'
        ELSE 'Unknown'
    END AS Role
FROM Staff S
LEFT JOIN Receptionist R ON S.staffID = R.staffID
LEFT JOIN Housekeeper H ON S.staffID = H.staffID
LEFT JOIN Bellboy B ON S.staffID = B.staffID
LEFT JOIN RoomServiceStaff RS ON S.staffID = RS.staffID
LEFT JOIN Manager M ON S.staffID = M.staffID
LEFT JOIN HousekeepingTask HT ON S.staffID = HT.staffID
LEFT JOIN Luggage L ON S.staffID = L.staffID
LEFT JOIN ServiceRequest SR ON S.staffID = SR.staffID
WHERE HT.taskID IS NULL
  AND L.luggageID IS NULL
  AND SR.requestID IS NULL;
```



staffID	staffName	Role
1	Alice Johnson	Receptionist
2	Bob Smith	Receptionist
9	Jason See	Manager
10	Kelvin Smith	Manager

-- Rooms with No Current Bookings

```
SELECT R.roomID, R.roomType, R.roomStatus, R.features, R.pricePerNight,
R.housekeepingStatus
FROM Room R
LEFT JOIN Booking B ON R.roomID = B.roomID AND B.bookingStatus = 'Confirmed'
WHERE R.roomStatus NOT LIKE 'Occupied'
  AND B.bookingID IS NULL;
```

roomID	roomType	roomStatus	features	pricePerNight	housekeepingStatus
101	Single	Available	Wi-Fi, TV	100.00	Clean
103	Suite	Under Maintenance	Wi-Fi, TV, Jacuzzi	300.00	Under Maintenance
104	Single	Available	Wi-Fi, TV	100.00	Clean

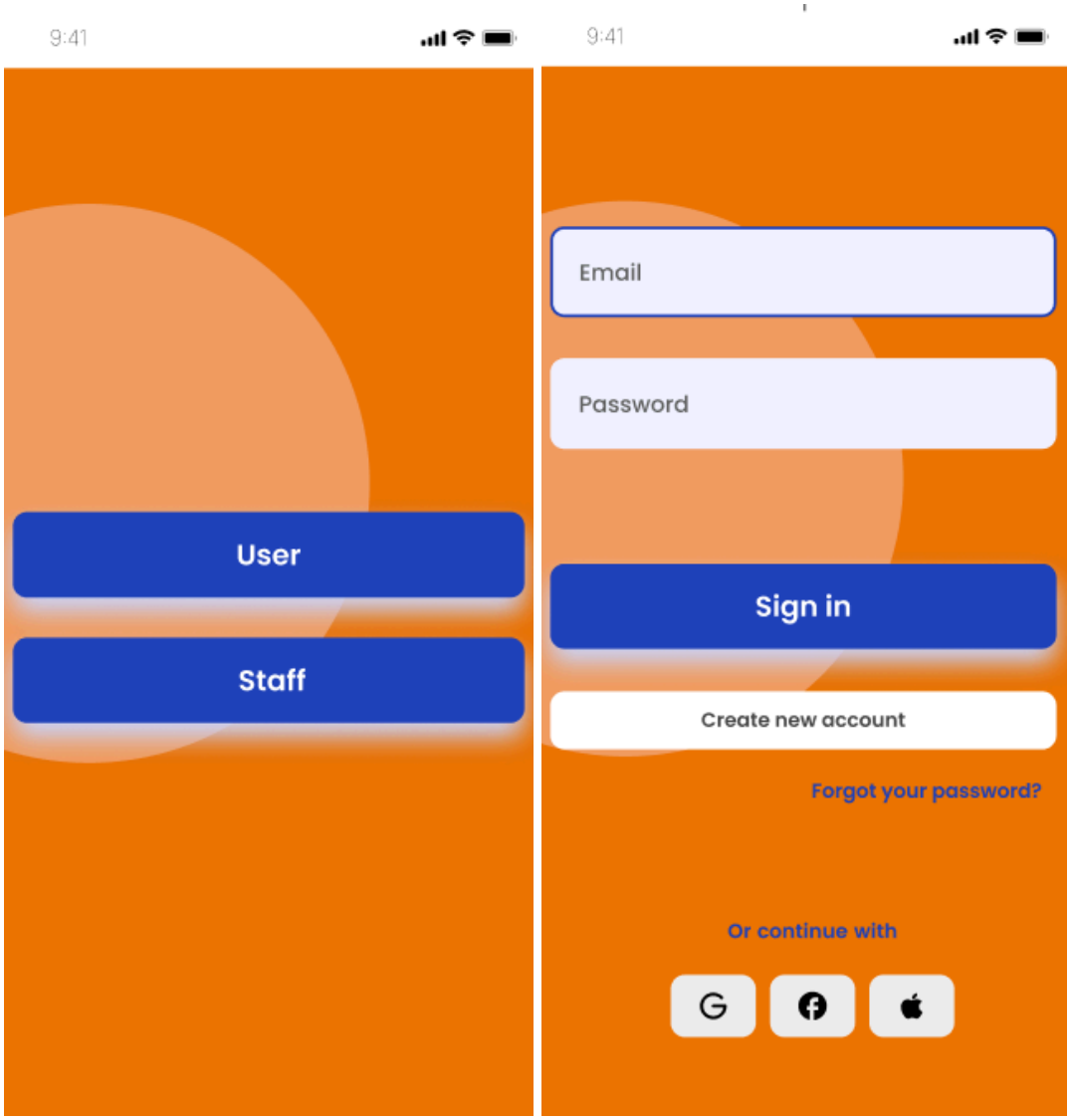
-- Bellboys Handling the Heaviest Luggage

```
SELECT B.staffID, S.staffName, MAX(L.weight) AS MaxWeight
FROM Bellboy B
JOIN Staff S ON B.staffID = S.staffID
JOIN Luggage L ON B.staffID = L.staffID
GROUP BY B.staffID, S.staffName
ORDER BY MaxWeight DESC;
```

staffID	staffName	MaxWeight
6	Edward Stark	30.20
5	Dan Prince	25.50

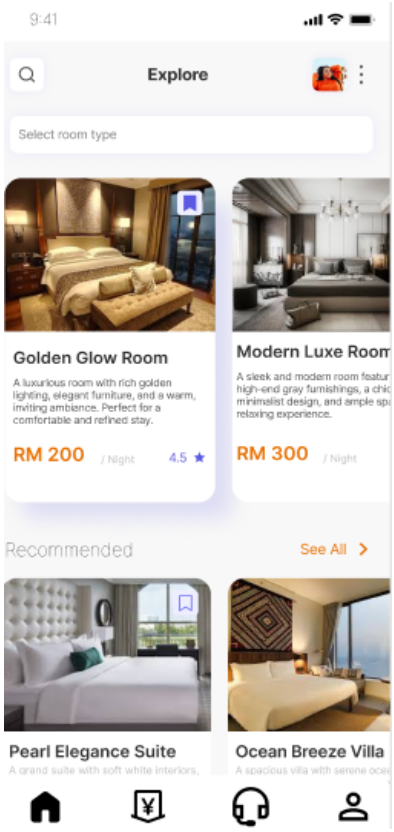
14.0 Interface

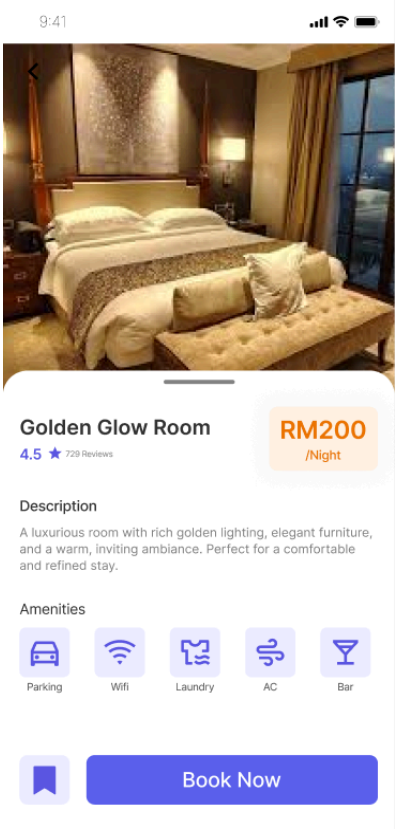
14.1 User Log in

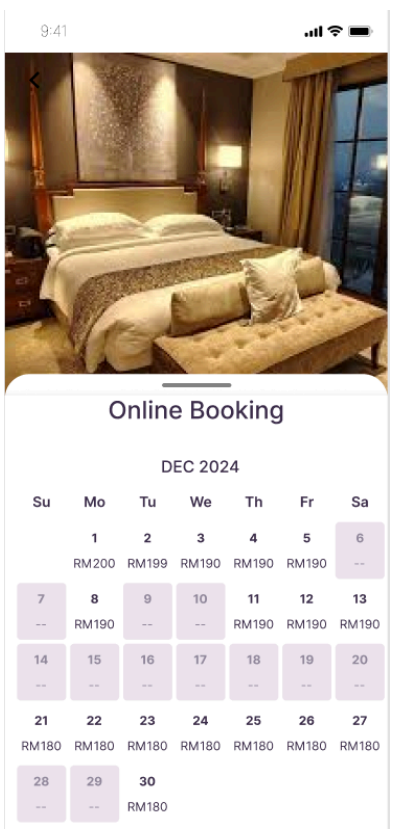


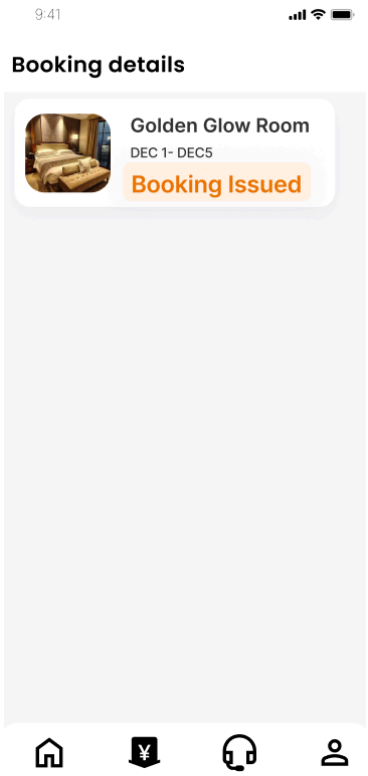
14.2 Guest View

14.2.1 Guest booking

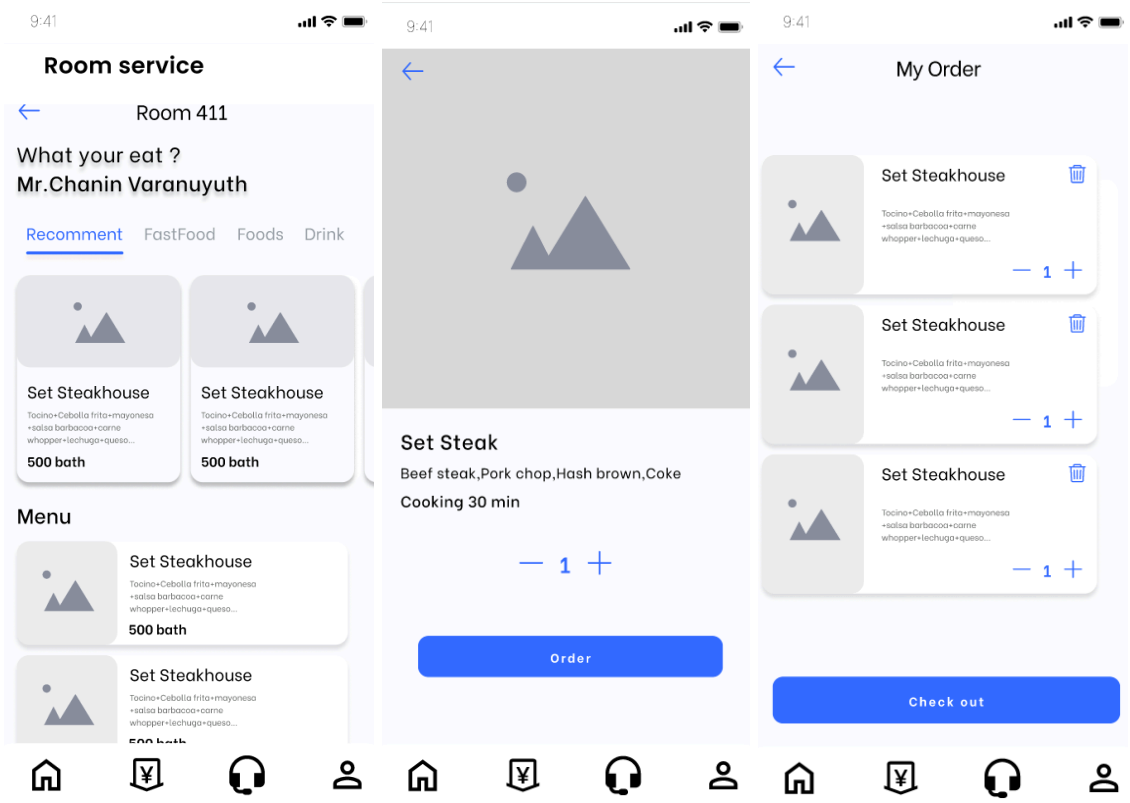




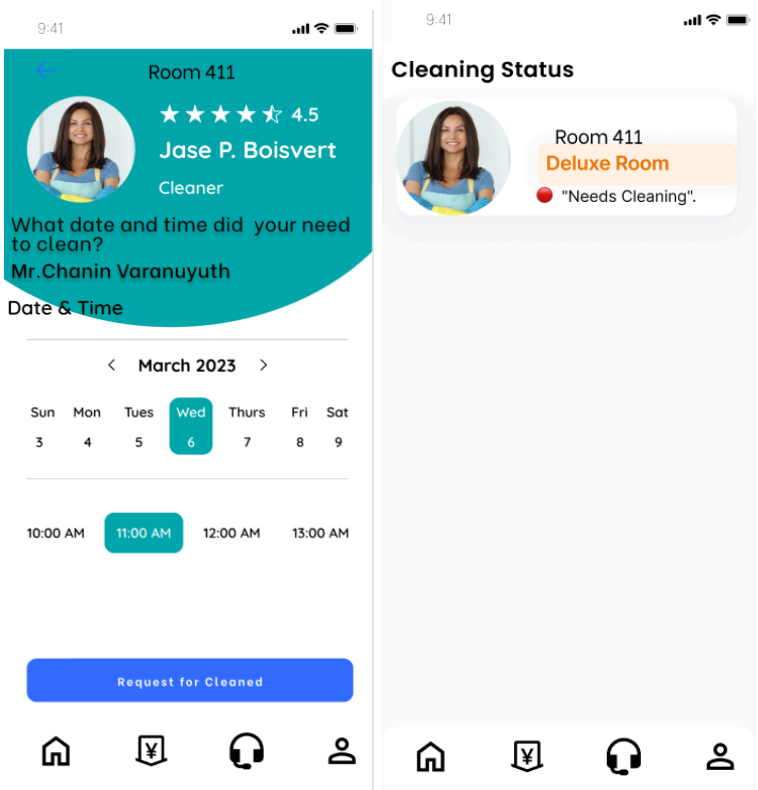




14.2.2 Room service food

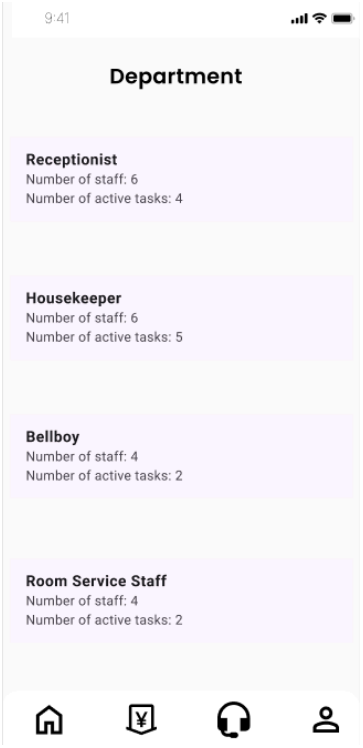


14.2.3 Room Clean Up

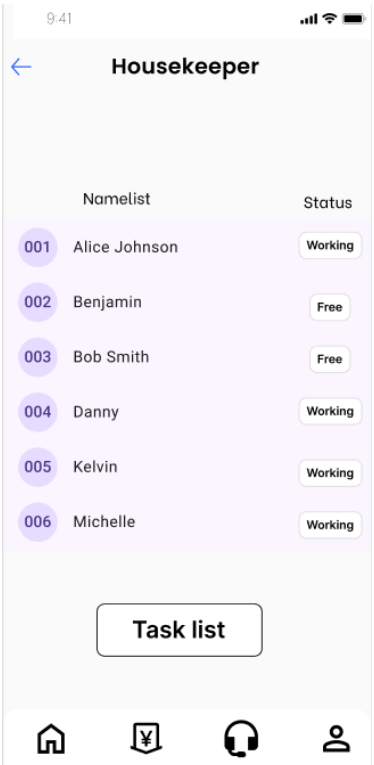


14.3 Manager View

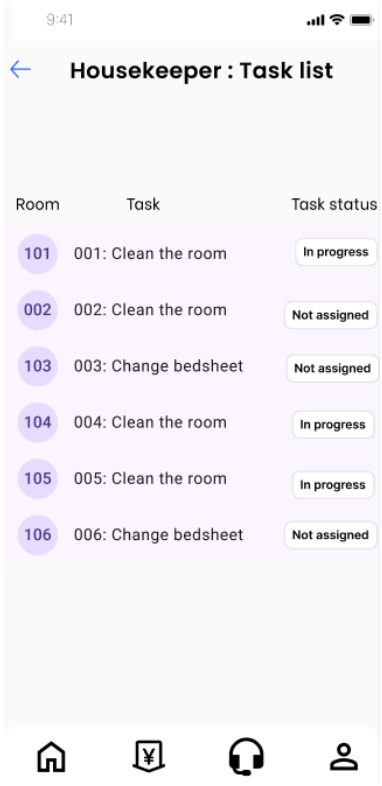
14.3.1 Manager view department



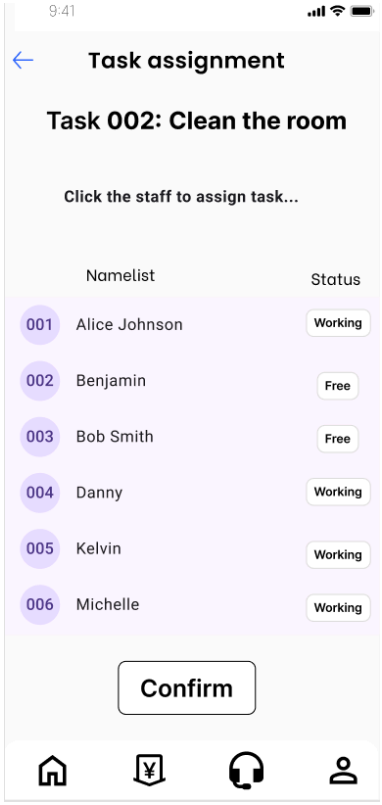
14.3.2 Manager view namelist of staff of a department



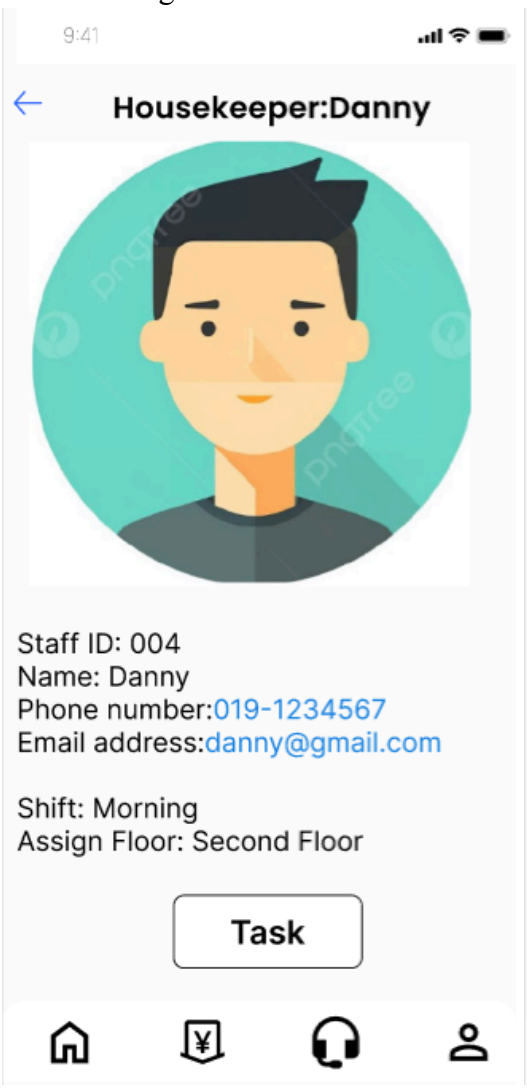
14.3.3 Manager view task list of a department

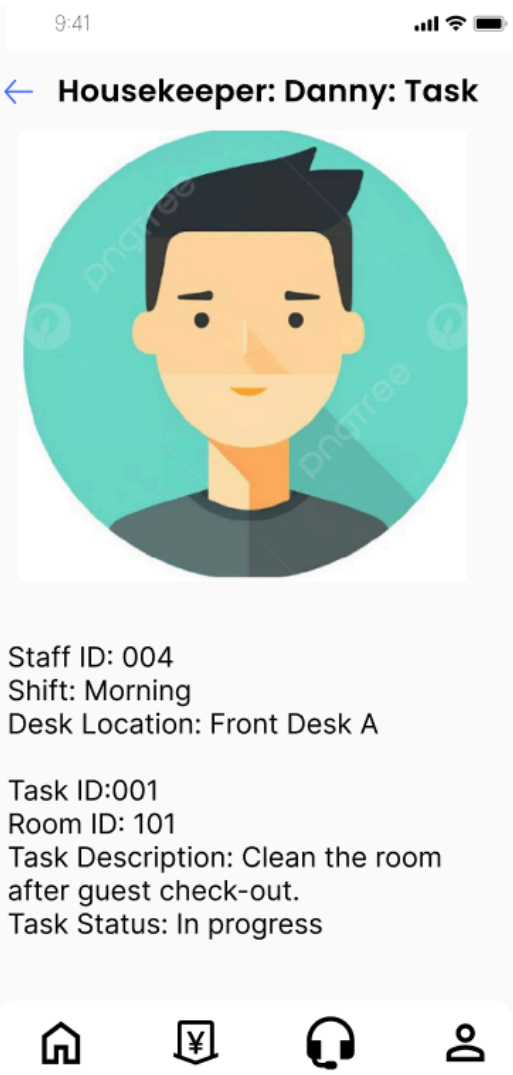


14.3.4 Manager handle task assignment



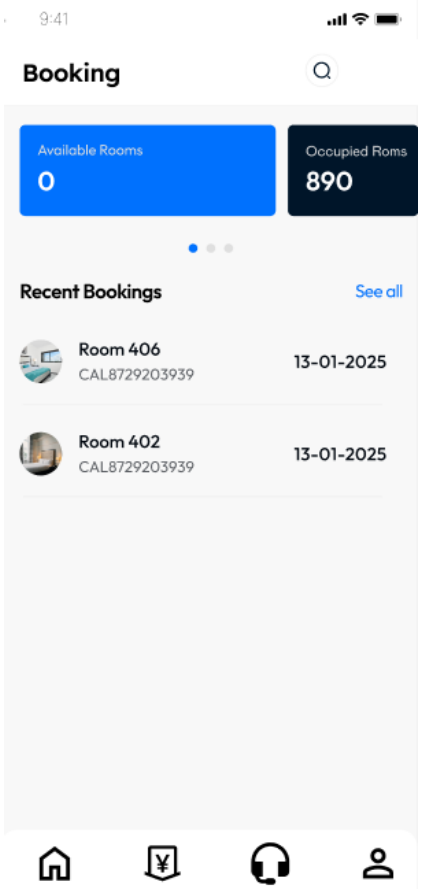
14.3.5 Manager view staff information

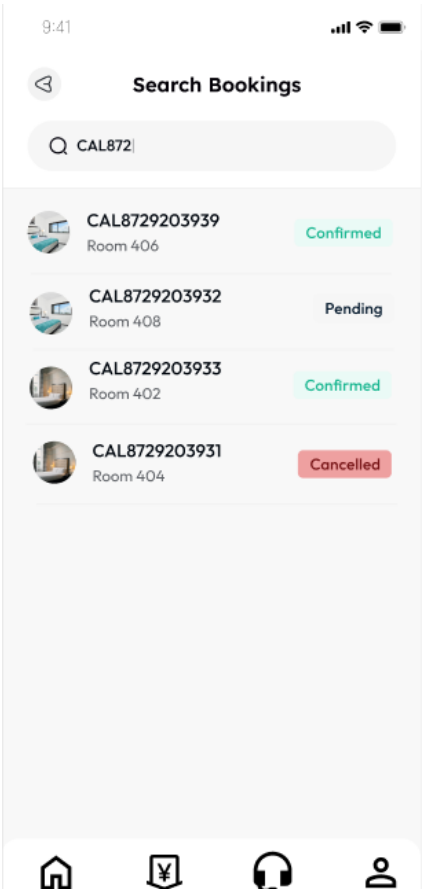


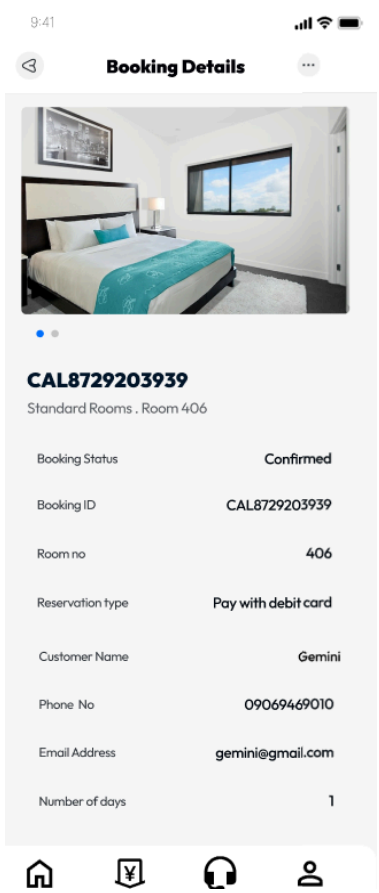


14.4 Receptionist View

14.4.1 Receptionist view booking

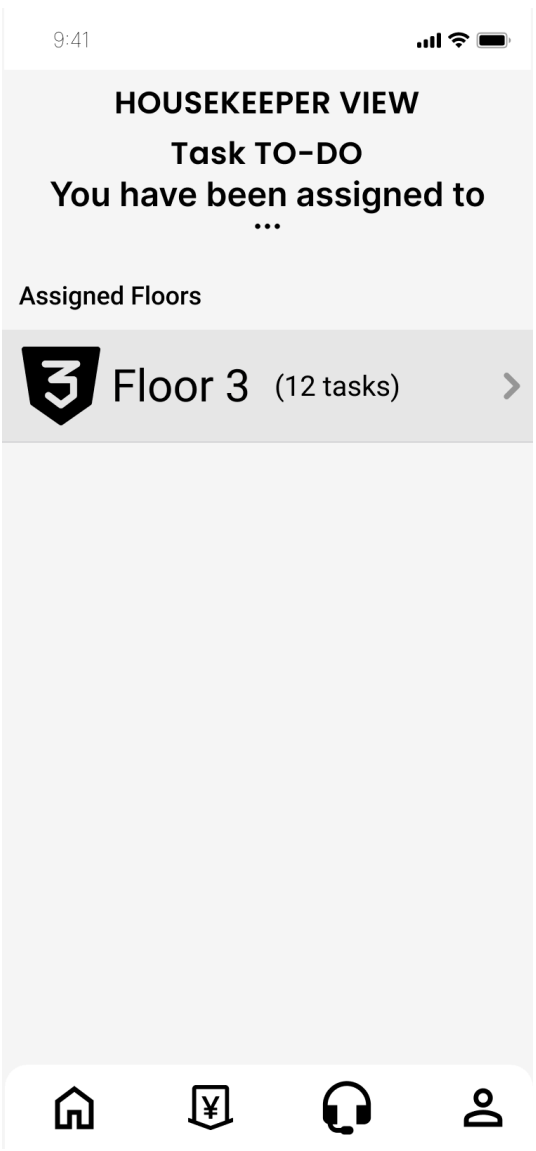


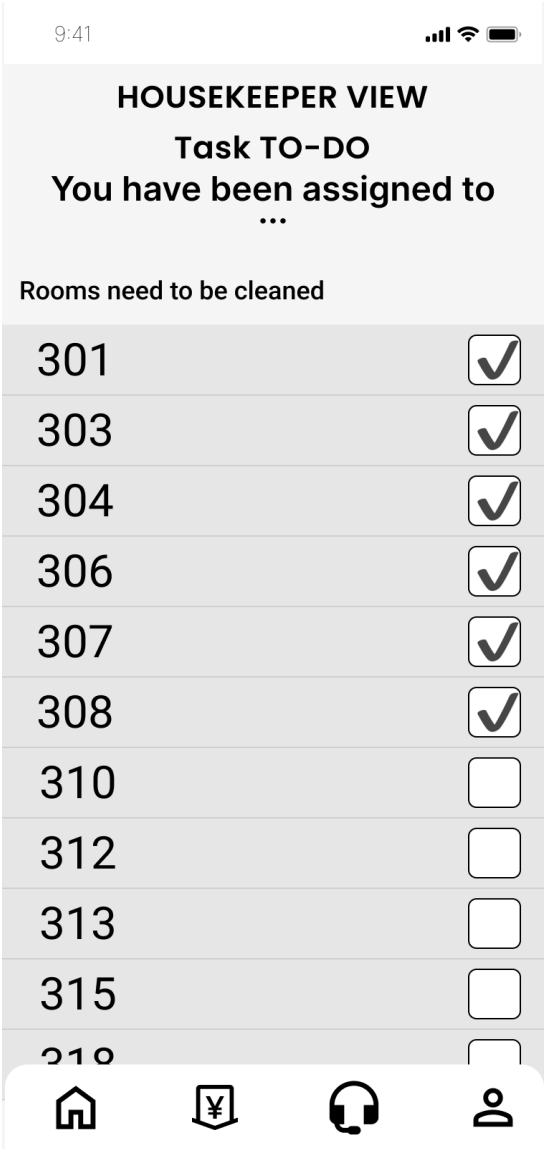




14.5 Housekeeper View

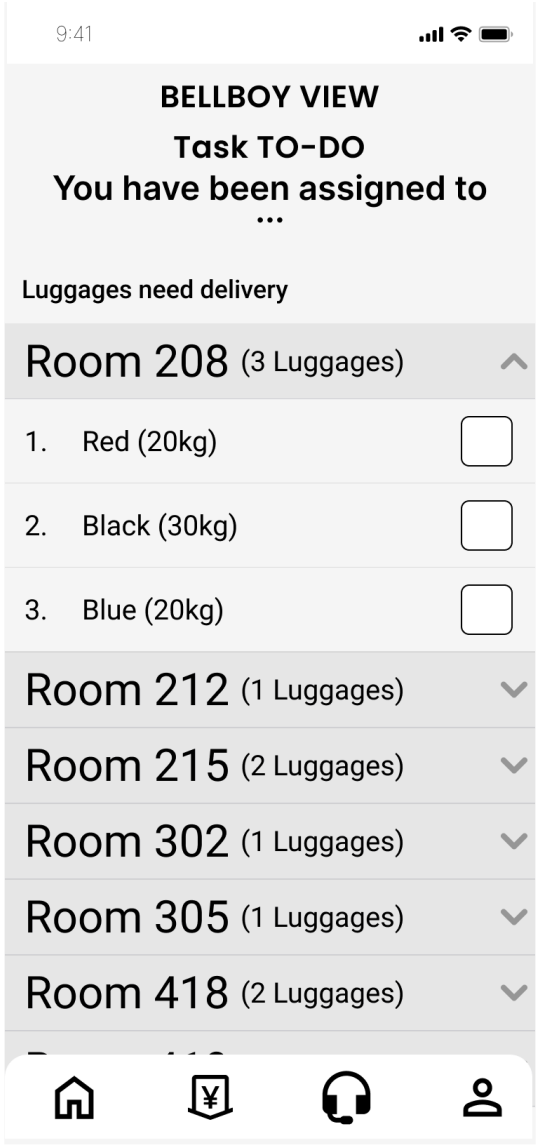
14.5.1 Housekeeper view task





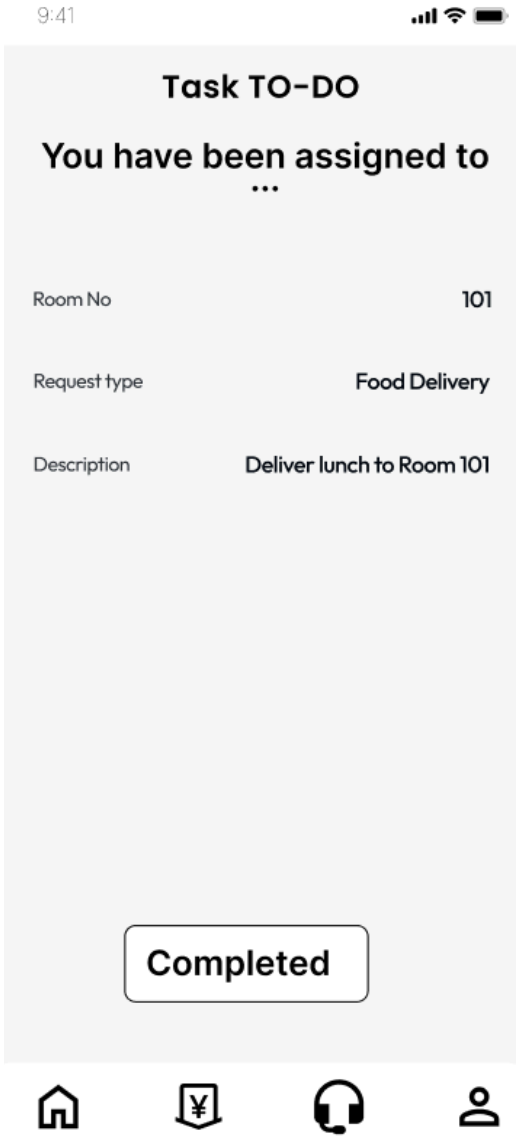
14.6 Bellboy View

14.6.1 Bellboy view task



14.7 Room Service Staff View

14.7.1 Room Service Staff view task



15.0 Summary

The idea of the Hotel Booking System project is to update the services that a hotel provides to guests, making the process fast and effective. Through the use of a relational database format in the design of the system, the work of the staff is made accurate and their work well coordinated. Major innovation solutions underpinning the program include an automated system for room bookings with real-time changes to eliminate the incidence of double bookings, an electronic task board to centralize work schedules and facilitate inter-organizational communication, and luggage tracking using QR codes to ensure accurate delivery timings. These services make the guests' lives easier and more convenient, they can check in and out, order services through a mobile app.Such things as wrong bookings, misunderstandings and eventual disappointments are minimized because the system is a central hub where everything is coordinated in real time. Its structure of its database is highly reliable and can be expandable for future growth of the hotels. The core values which the project brings are less problems during the operations, satisfied guests, and sound management of resources. This envisaged solution is therefore formed to assist the hotels deliver great services and sustain their operation in the competitive hospitality environment.