# SOFTWARE ENGINEERING FINAL REPORT

HOTEL MANAGEMENT SOFTWARE

*Instructor*: **GV NGUYỄN NGỌC PHIÊN** *Executor*:  **LÊ QUANG HUY - 521H0238**

Course: **25**

## HO CHI MINH CITY,2024

# SOFTWARE ENGINEERING FINAL REPORT

HOTEL MANAGEMENT SOFTWARE

*Instructor*: **GV NGUYỄN NGỌC PHIÊN** *Executor*:  **LÊ QUANG HUY - 521H0238**

Course: **25**

## HO CHI MINH CITY,2024

# THANK YOU

First and foremost, we extend our heartfelt gratitude to Professor Nguyen Ngoc Phien for his unwavering support and invaluable guidance throughout our research and the completion of this end-of-term report.

We would also like to express our appreciation to the Department of Information Technology at Ton Duc Thang University for providing the facilities and environment conducive to our studies and research. The faculty has consistently been eager to impart essential knowledge and share document reference experiences, which have been crucial not only in executing and finalizing research projects but also in our overall learning and practical training at the university.

Finally, after a period of diligent study and class engagement, we have completed this research project under the tutelage and wisdom of our educators. Despite our best efforts, we acknowledge that there are still areas of improvement due to our limited knowledge and reasoning capabilities. We humbly seek further guidance and contributions to enhance our research. We are also thankful for the constructive suggestions from our teachers and peers, which will undoubtedly aid us in improving future research endeavors. We hope that all our supportive teachers and friends remain in good health and spirits.

## WE THANK YOU!

# PROJECT COMPLETED AT TON DUC THANG UNIVERSITY

I hereby declare that this is my own project and is guided by Mr Nguyen Ngoc Phien. The research contents and results in this topic are honest and have not been published in any form before. The data in the tables for analysis, comments and evaluation are collected by the author himself from different sources, clearly stated in the reference section.

In addition, the project also uses a number of comments, assessments as well as data of other authors, other agencies and organizations, with citations and source annotations.

If I find any fraud, I will take full responsibility for the content of my project. Ton Duc Thang University is not related to copyright and copyright violations caused by me during the implementation process (if any).

*Ho Chi Minh City, day month year*

## TEACHER'S CONFIRMATION AND ASSESSMENT SECTION

### The confirmation part of the instructor















*Ho Chi Minh City, day month year*

*Author*

*(Sign and write full name)*

## The evaluation part of the teacher marks the exam















*Ho Chi Minh City, day month year Author*

*(Sign and write full name)*

# SUMMARY

The report is made using the knowledge learned and materials provided by the lecturer, in addition to the reference of documents at Ton Duc Thang University Library.

# TABLE OF CONTENTS

[THANK YOU i](#_gjdgxs)

[PROJECT COMPLETED AT TON DUC THANG UNIVERSITY ii](#_30j0zll)

[TEACHER'S CONFIRMATION AND ASSESSMENT SECTION 3](#_1fob9te)

[SUMMARY 4](#_3znysh7)

[TABLE OF CONTENTS 5](#_2et92p0)

[LIST OF TABLES 9](#_tyjcwt)

[LIST OF IMAGES 10](#_3dy6vkm)

[CONTENTS OF THE REPORTS 12](#_1t3h5sf)

[CHAPTER I: INTRODUCTION 12](#_4d34og8)

[Topic name 12](#_2s8eyo1)

1. [Reasons to choose a topic 12](#_17dp8vu)
2. [Requirement engineering 12](#_3rdcrjn)
   1. [Require function 12](#_26in1rg)
   2. [Non-functional requirements 12](#_lnxbz9)
      1. [Performance requirements 12](#_35nkun2)
      2. [Safety requirements 13](#_1ksv4uv)
      3. [Confidentiality 13](#_44sinio)
      4. [Software quality attributes 13](#_2jxsxqh)
3. [Business process 13](#_z337ya)
4. [Fact survey 14](#_3j2qqm3)

[CHAPTER II: SPECIFIC DESCRIPTION 17](#_4i7ojhp)

1. [System specification 17](#_2xcytpi)
2. [Target audience 17](#_1ci93xb)
3. [Range 17](#_3whwml4)
4. [Possibility 17](#_2bn6wsx)
5. [Actors in the system 18](#_qsh70q)
6. [Use Cases in the system 18](#_1pxezwc)
7. [System functions 19](#_2p2csry)

[CHAPTER III: DETAILED DESIGN 21](#_3o7alnk)

1. [General use case diagram 21](#_23ckvvd)
2. [Use case diagram 22](#_32hioqz)
   1. [Use case Sign Up, Sign In 22](#_1hmsyys)
   2. [Use case Management of personal information 23](#_2grqrue)
   3. [Use case Cart management, Buy Items 23](#_3fwokq0)
   4. [Use case Payment 24](#_4f1mdlm)
   5. [Use case View products 24](#_19c6y18)
   6. [Use case Search for products 25](#_28h4qwu)
   7. [Use case View orders 25](#_37m2jsg)
   8. [Use case Customer support 26](#_46r0co2)
   9. [Use case Manage member lists 26](#_111kx3o)
   10. [Use case Product portfolio management 27](#_206ipza)
   11. [Use case Receiving and processing orders 27](#_2zbgiuw)
3. [Use case specification 28](#_3ygebqi)
   1. [Use case Sign Up, Sign In 28](#_2dlolyb)
   2. [Use case Management of personal information 29](#_3cqmetx)
   3. [Use case Cart management, Buy Items 30](#_4bvk7pj)
   4. [Use case Payment 31](#_1664s55)
4. [Use case View products 32](#_25b2l0r)
5. [Search for products 33](#_34g0dwd)
6. [View orders 34](#_43ky6rz)
7. [Customer support 35](#_xvir7l)
8. [Manage member lists 36](#_1x0gk37)
9. [Product portfolio management 37](#_2w5ecyt)
10. [Receiving and processing orders 38](#_3vac5uf)
11. [Activity diagram 40](#_pkwqa1)
    1. [Sign up 40](#_39kk8xu)
    2. [Sign in 41](#_48pi1tg)
    3. [Management of personal information 42](#_1302m92)
    4. [Payment and Buy Item 43](#_2250f4o)
    5. [View products 44](#_319y80a)
    6. [Search for products 45](#_40ew0vw)
    7. [View orders 46](#_upglbi)
    8. [Manage member lists 47](#_1tuee74)
    9. [Product portfolio management 48](#_2szc72q)
    10. [Receiving and processing orders 49](#_3s49zyc)
12. [Sequence diagram 50](#_meukdy)
    1. [Sign up 50](#_36ei31r)
    2. [Sign In 51](#_45jfvxd)
    3. [Management of personal information 51](#_zu0gcz)
    4. [Payment and Buy Item 52](#_1yyy98l)
    5. [View products 53](#_2y3w247)
13. [Search for products 54](#_3x8tuzt)
14. [View orders 55](#_rjefff)
15. [Customer support 56](#_1qoc8b1)
16. [Product portfolio management 57](#_2pta16n)
17. [Manage member lists 58](#_3oy7u29)
18. [Receiving and processing orders 59](#_j8sehv)
19. [Class diagram 60](#_1idq7dh)
20. [Entity Relationship Diagram and Relational Data Model 61](#_2hio093)
    1. [Relation Data Model 61](#_wnyagw)
    2. [Entity Relationship Diagram 62](#_1vsw3ci)

[CHAPTER IV: ACTUAL PROCESS 62](#_2uxtw84)

* + 1. [Software development life cycle model 62](#_1a346fx)
       1. [Introduction to the waterfall model 62](#_3u2rp3q)
       2. [Advantages 63](#_2981zbj)
       3. [Defect 64](#_odc9jc)
       4. [Reason for choosing model 64](#_38czs75)
    2. [Architecture use 64](#_1nia2ey)

[CHAPTER V: DEMO CODE 65](#_2mn7vak)

1. [Customer interface 65](#_11si5id)
2. [Admin interface 70](#_16x20ju)

[CHAPTER VI: TEST CASE AND UNIT TEST 72](#_44bvf6o)

[REFERENCES 74](#_ymfzma)

# LIST OF TABLES

[Table 1. Actual interview table 17](#_1y810tw)

[Table 2. Agent table in the system 18](#_3as4poj)

[Table 3. Table of use cases in the system 19](#_49x2ik5)

[Table 4. Table of functions in the system 20](#_147n2zr)

[Table 5. Specification sheet use case sign up, sign in 29](#_sqyw64)

[Table 6. Specification sheet Use case Management of personal information 30](#_1rvwp1q)

[Table 7. Specification sheet Use case Cart management, Buy Items 31](#_2r0uhxc)

[Table 8. Specification sheet Use case Payment 32](#_3q5sasy)

[Table 9. Specification sheet Use case View products 33](#_kgcv8k)

[Table 10. Specification sheet Search for products 34](#_1jlao46)

[Table 11. Specification sheet use case View orders 35](#_2iq8gzs)

[Table 12. Specification sheet use case Customer support 35](#_3hv69ve)

[Table 13. Specification sheet use case Manage member lists 37](#_4h042r0)

[Table 14. Specification sheet Product portfolio management 38](#_1baon6m)

[Table 15. Specification sheet use case Product portfolio management 39](#_2afmg28)

[Table 16. Table of programming languages and libraries used 64](#_47hxl2r)

[Table 17. Test Case and Unit Test 73](#_2jh5peh)

# LIST OF IMAGES

[Image 1. General use case diagram 21](#_ihv636)

[Image 2. Use case Sign up, Sign In 22](#_41mghml)

[Image 3. Use case Management of personal information 23](#_vx1227)

[Image 4. Use case Cart management, Buy Items 23](#_1v1yuxt)

[Image 5. Use case Payment 24](#_2u6wntf)

[Image 6. Use case View product 24](#_3tbugp1)

[Image 7. Use case Search for products 25](#_nmf14n)

[Image 8. Use case view orders 25](#_1mrcu09)

[Image 9. Use case Customer support 26](#_2lwamvv)

[Image 10. Use case manage member lists 26](#_3l18frh)

[Image 11. Use case Product portfolio management 27](#_4k668n3)

[Image 12. Use case Receiving and processing orders 27](#_1egqt2p)

[Image 13. Sign up activity diagram 40](#_1opuj5n)

[Image 14. Sign in activity diagram 41](#_2nusc19)

[Image 15. Management of personal information activity diagram 42](#_3mzq4wv)

[Image 16. Payment and Buy Item activity diagram 43](#_haapch)

[Image 17. View product activity diagram 44](#_1gf8i83)

[Image 18. Search for products activity diagram 45](#_2fk6b3p)

[Image 19. View orders activity diagram 46](#_3ep43zb)

[Image 20. Manage member lists activity diagram 47](#_4du1wux)

[Image 21. Product portfolio management activity diagram 48](#_184mhaj)

[Image 22. Receiving and processing orders activity diagram 49](#_279ka65)

[Image 23. Sign up sequence diagram 50](#_1ljsd9k)

[Image 24. Sign In sequence diagram 51](#_2koq656)

[Image 25. Management of personal information sequence diagram 51](#_3jtnz0s)

[Image 26. Payment and Buy Item sequence diagram 52](#_4iylrwe)

[Image 27. View products sequence diagram 53](#_1d96cc0)

[Image 28. Search for products sequence diagram 54](#_2ce457m)

[Image 29. View orders sequence diagram 55](#_3bj1y38)

[Image 30. Customer support sequence diagram 56](#_4anzqyu)

[Image 31. Product portfolio management sequence diagram 57](#_14ykbeg)

[Image 32. Manage member lists sequence diagram 58](#_243i4a2)

[Image 33. Receiving and processing orders sequence diagram 59](#_338fx5o)

[Image 34. Class diagram 60](#_42ddq1a)

[Image 35. Relation Data model 61](#_3gnlt4p)

[Image 36. Entity Relationship Diagram 62](#_4fsjm0b)

[Image 37. Login screen 65](#_3ls5o66)

[Image 38. Register screen 1 65](#_20xfydz)

[Image 39. Register screen 2 66](#_4kx3h1s)

[Image 40. Home screen 66](#_302dr9l)

[Image 41. Product details screen 1 67](#_1f7o1he)

[Image 42. Product details screen 2 67](#_3z7bk57)

[Image 43. Website info bar 67](#_2eclud0)

[Image 44. Personal information screen 68](#_thw4kt)

[Image 45. Cart screen 68](#_3dhjn8m)

[Image 46. Order screen 68](#_1smtxgf)

[Image 47. Payment screen 69](#_4cmhg48)

[Image 48. Shopping history screen 69](#_2rrrqc1)

[Image 49. Admin homepage 70](#_3qwpj7n)

[Image 50. Product management 70](#_261ztfg)

[Image 51. User management 71](#_l7a3n9)

[Image 52. Order management 71](#_356xmb2)

[Image 53. Add category page 72](#_1kc7wiv)

# CONTENTS OF THE REPORTS

### CHAPTER I: INTRODUCTION

### Topic name:

Hotel management software

### oReasons to choose a topic:

* + - * **Practical Necessity**: As the hospitality industry grows, so does the complexity of managing a hotel's daily operations. Hotel management software simplifies this by automating processes like bookings, customer management, and billing, making it essential for modern hotels to operate efficiently.
      * **Operational Challenges:** Running a hotel involves countless tasks that can overwhelm even the most experienced staff. Using management software helps streamline these tasks, allowing staff to focus more on guest experience rather than back-end operations.
      * **Market Demand:** The demand for hotel management software is on the rise as hoteliers recognize the need for technology that can handle complex operations and provide a competitive edge. This makes it a timely and relevant topic for study and discussion.
      * **Innovation and Updates**: The field of hotel management software is continuously evolving, with new features and capabilities being developed regularly. Staying updated on these innovations is crucial for anyone involved in the hospitality industry.
      * **Ease of Use**: Modern hotel management systems are designed with user-friendly interfaces, making it easier for staff at all levels to adapt and use the software effectively, hence reducing training time and costs.
      * **Enhanced Customer Service:** By automating routine tasks, hotel management software frees up staff to provide better, more personalized service to guests, directly impacting customer satisfaction and loyalty.

These reasons highlight the practical implications and benefits of adopting hotel management software, making it an essential tool for enhancing operational efficiency and guest satisfaction in the hospitality industry.

### Requirement engineering

### Require function

Admin: Customer consultation, order management, product management, user information list management

### Non-functional requirements

### Performance requirements

* + - * Data is updated quickly, with complete consistency.
      * Information received quickly and effectively through the website.
      * Friendly interface, easy to use, fast response speed (under 1s for 1000 requests).
      * Capable of serving a large amount of traffic at the same time.

### Safety requirements

* + - * Capable of storing state, backing up the system and restoring data when having problems (network attack, connection failure,hardware failure,...).

### Confidentiality

* + - * Strict decentralization: Manage user rights through functions (screens) and data. Independently designed View, Add, Edit, Delete functions give customers more flexibility in organizing multiple users and controlling data.

### Software quality attributes

* + - * Built on web platform and compatible with many browsers (Chrome, Opera, Firefox, mobile browsers,...) to use anytime, anywhere..
      * Easy to maintain, upgrade, develop software after deployed. Interoperability, maintainability, portability, reliability, reusability, robustness, testability.
      * The ability to reuse the software for a number of other information management purposes is very powerful.
      * Absolute data control, safe and secure.

### Business process

Hotel management software streamlines hotel operations, enhancing efficiency and guest experiences. It manages reservations, including online booking and channel synchronization, and optimizes front desk operations with quick check-in/check-out and room allocation. Housekeeping benefits from room status tracking and task management, while POS systems handle restaurant, bar, spa, and retail sales. CRM functions store guest profiles, manage loyalty programs, and collect feedback. Revenue management adjusts pricing dynamically and provides financial forecasting. Financial management includes billing, accounts, and reporting. HR features support staff scheduling, payroll, and performance tracking. Maintenance management handles work orders and preventive tasks. Reporting and analytics offer insights through customizable dashboards. Key features include system integration, mobile access, user-friendly interfaces, security, and customization, ultimately boosting efficiency, satisfaction, and profitability.

### 

### CHAPTER II: SPECIFIC DESCRIPTION

### System specification

The system "Hotel Management Software" includes three main objects: Customer, Admin, and Hotel Staff.

On the Customer side: Customers can book rooms through the hotel's website. To make a booking, they must create an account by registering beforehand. Customers can view available rooms, choose their desired dates, and complete their booking by paying with a credit card or other accepted payment methods. After their stay, customers can leave reviews and feedback about their experience.

On the Admin side: Admins manage suppliers, room inventories, customer accounts, and personal data. They also oversee the pricing of rooms, special promotions, and overall hotel operations. When a customer makes a booking, the Admin receives the booking details, processes the payment, and ensures the reservation is confirmed. Admins also handle customer service inquiries and resolve any issues that may arise.

On the Hotel Staff side: Hotel Staff are responsible for day-to-day operations, including front desk management, housekeeping, and maintenance. When a guest checks in, front desk staff verify their booking details and assist with the check-in process. Housekeeping staff update the room status and ensure rooms are clean and well-maintained. Maintenance staff handle any repair or maintenance requests. Additionally, Hotel Staff can update the system with real-time information about room availability and service requests.

### Target audience

The Hotel Management Software is aimed at hotel managers who want to streamline their operations, improve efficiency, enhance guest experiences, and optimize business profits.

### Range

The Hotel Management Software includes:

* Reservation Management: Online bookings, channel management, group reservations.
* Front Desk Operations: Check-in/check-out, room allocation, guest registration.
* Housekeeping Management: Room status tracking, task assignment, inventory management.
* Point of Sale (POS): Orders and billing for restaurants, bars, spa services, retail shops.
* Customer Relationship Management (CRM): Guest profiles, loyalty programs, feedback analysis.
* Revenue Management: Dynamic pricing, yield management, financial forecasting.
* Accounting and Financial Management: Billing, invoicing, accounts management, financial reporting.
* Human Resources Management: Staff scheduling, payroll, performance tracking.
* Maintenance Management: Work orders, asset management, preventive maintenance.
* Reporting and Analytics: Customizable dashboards, data analytics.
* Security: Data encryption, role-based access, multi-factor authentication, regulatory compliance.
* Integration Capabilities: CRM, accounting software, OTAs integration.
* Mobile Accessibility: Access from mobile devices.
* User-Friendly Interface: Intuitive and easy to navigate.

### Possibility

The Hotel Management Software streamlines operations, enhances guest experiences, optimizes revenue with dynamic pricing, and provides real-time insights. It integrates with CRM, accounting, and OTA systems, offers mobile access, and ensures robust security. Additionally, it manages staff, maintenance, and inventory, gathers customer feedback, and ensures regulatory compliance, boosting efficiency, satisfaction, and profitability.

### Actors in the system

| **Actor** | **Describe** | **Function** |
| --- | --- | --- |
| Customer | As regular customers, visit the website but do not register for an account and search for necessary information | View information |
| Search room |
| Book room |
| Exchange opinions, feedback |
| Admin | As an admin: Manage all system activities | Manage rooms |
| Manage categories |
| Manage system permissions |
| Manage services |
| Manage staff |
| System login/logout |
| Hotel Staff | Process customer information requests | System login/logout |
| Invoice payment |
| Manage bookings |
| Manage invoices |

*Table 2. Agent table in the system*

### Use Cases in the system

| **Ordinal**  **number** | **Use case Name** | **Describe** |
| --- | --- | --- |
| UC01 | Login/Logout | Allows users (receptionists, managers) to access the system by entering their credentials (username and password), which are validated by the system. Successful login grants access to system functionalities. Enables users to securely exit the system, ending their session to protect their account from unauthorized access. |
| UC02 | Manage employee information | Allows users to enter, update, and track employee details, including personal information, job roles, and schedules. |
| UC03 | Fill in customer information | Enables users to enter and update customer details, storing them in the system for bookings and personalized services. |
| UC04 | View Room List | Allows users to see a list of all available rooms and their current status. |

| UC05 | View Room Details | Allows users to see detailed information about a specific room, including amenities and current status. |
| --- | --- | --- |
| UC06 | Add Room | Enables users to add new rooms to the system, specifying details like room type, number, and features. |
| UC07 | Payment Method | Allows users to select and manage payment options for guest transactions, such as credit card, debit card, or digital payments. |

*Table 3. Table of use cases in the system*

### System functions

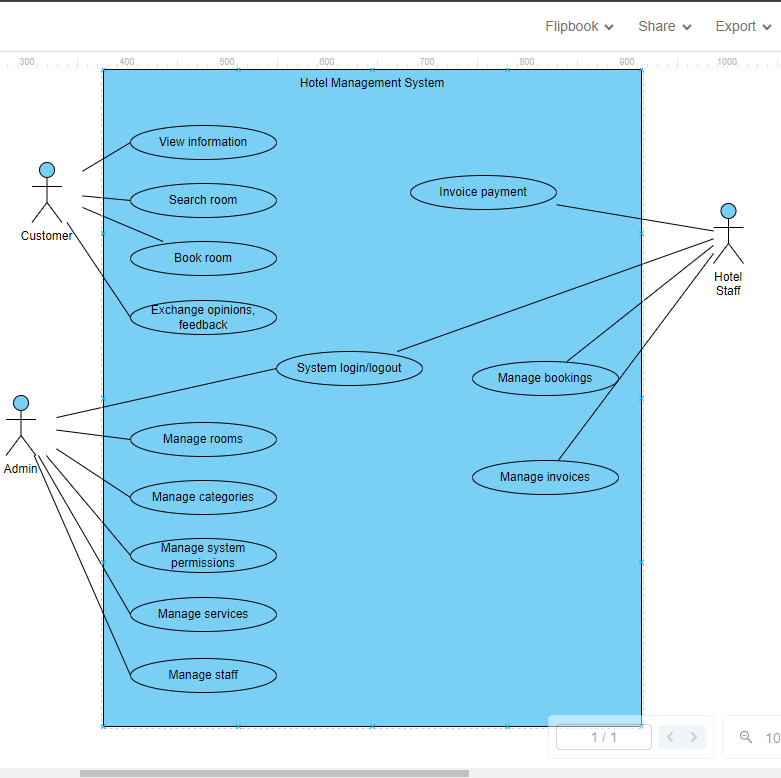
| **Ordinal**  **number** | **Function** | **Describe** |
| --- | --- | --- |
| 1 | Login | In a hotel management system, the receptionist or manager enters their username and password. The system validates these credentials and grants access if correct, otherwise, it displays an error message. |

| 2 | Room type | A classification defining the different categories of rooms available in a hotel, typically characterized by size, bed type, amenities, and occupancy capacity. Examples include Single, Double, Suite, Deluxe, and Family rooms.. |
| --- | --- | --- |
| 3 | View employee | Function allowing managers to access and review employee details, including personal information, job role, work schedule, and performance records. |
| 4 | Service | Offerings provided to guests to enhance their stay, such as room service, housekeeping, laundry, concierge, and spa services. |
| 5 | Order | Request made by guests for services or items, such as food, beverages, or amenities, which is processed and fulfilled by hotel staff. |
| 6 | Invoice | A detailed bill provided to guests, listing all charges for accommodation, services, and any additional expenses incurred during their stay. |
| 7 | Customer Detail | Allows admin to track, manage, and categorize customer information, including personal details, booking history, and preferences. |
| 8 | Log out | Process where users securely exit the system, ending their session and protecting their account from unauthorized access. |

*Table 4. Table of functions in the system*

### CHAPTER III: DETAILED DESIGN

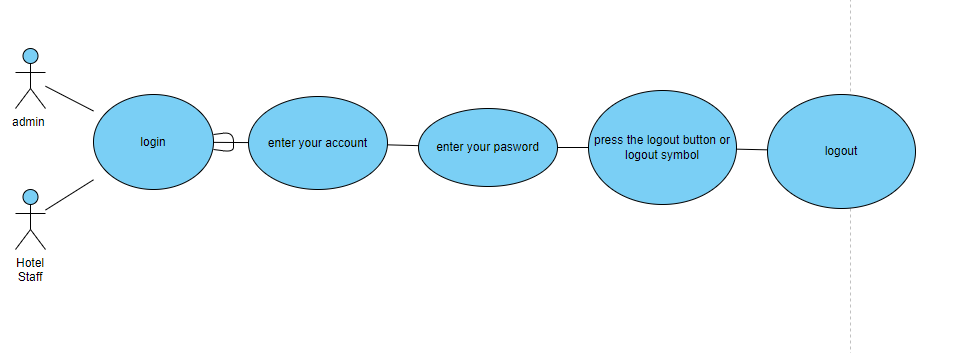
* 1. **General use case diagram**

****

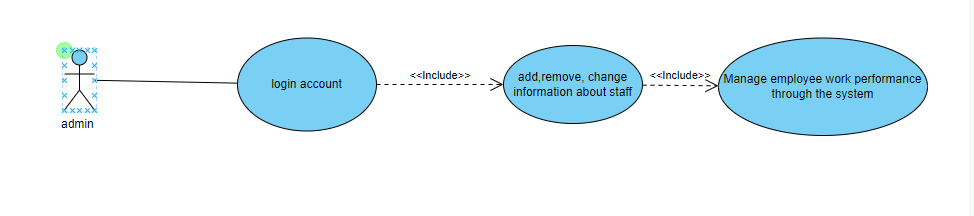
*Image 1. General use case diagram*

### Use case diagram

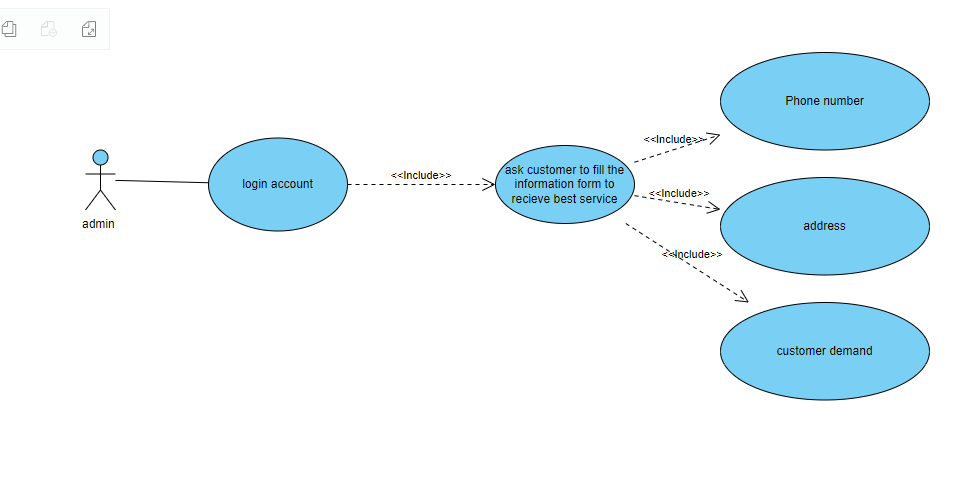
* 1. Login/Logout



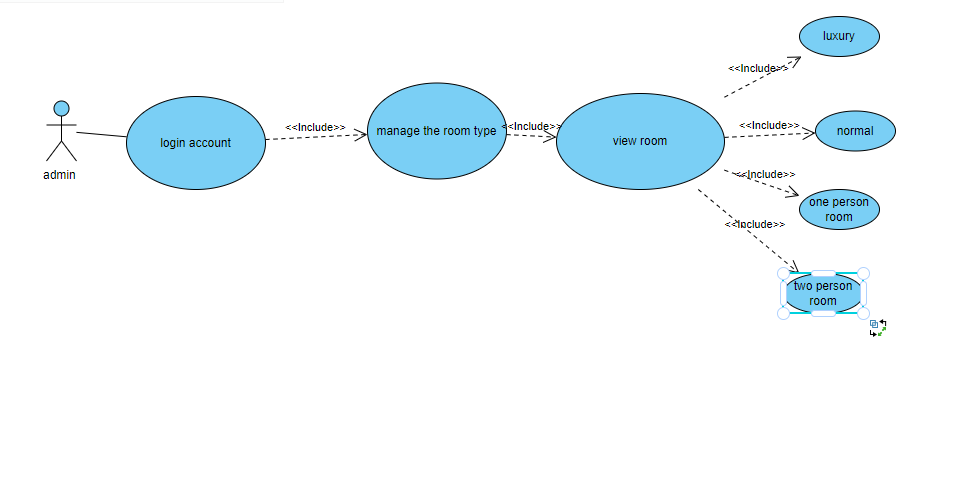
* 1. Manage employee information



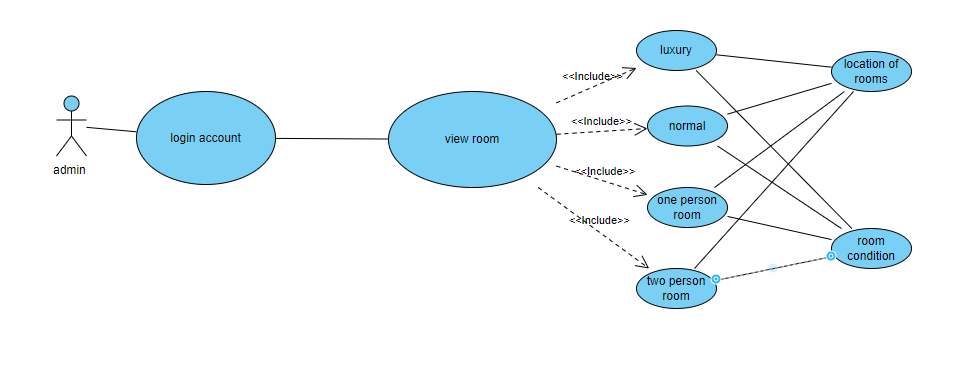
* 1. Fill in customer information



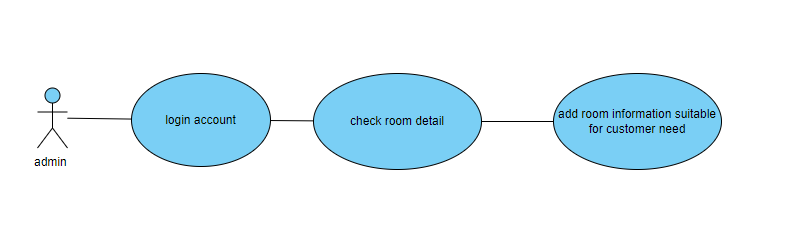
* 1. View Room List



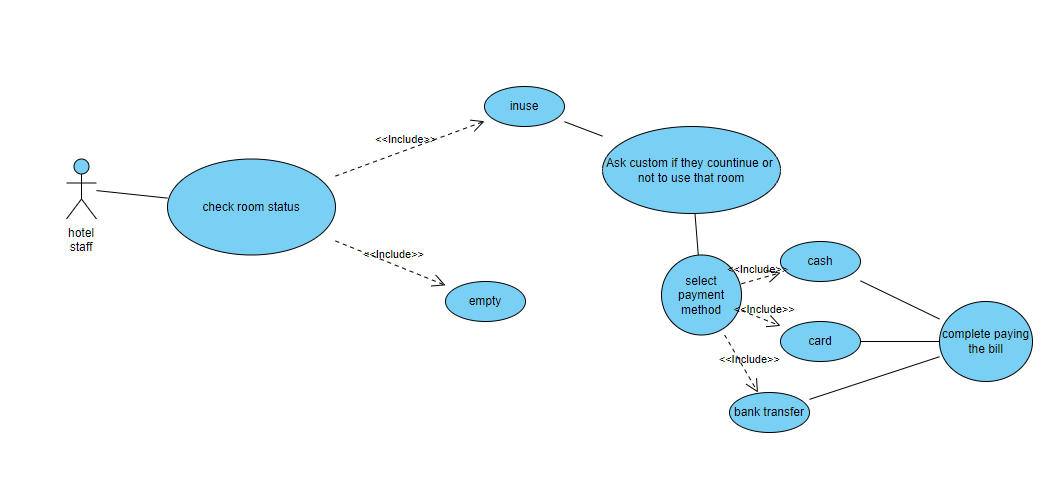
* 1. View Room Details



* 1. Add Room



* 1. Payment Method



### Use case specification

* 1. Manage employee information

| Use Case Name | Manage employee information |
| --- | --- |
| Scenario | Admin wants to enter, update, and track employee details |
| Triggering event | Admin logs in and needs to manage employee information |
| Brief description | Allows admins to manage employee details, including personal information, job roles, and schedules |
| Actors | Admin |
| Related use cases | Must have |
| Stakeholders | HR Department |
| Preconditions | -Admin successfully entered the system  - Admin has internet access |
| Post conditions | Employee details are saved and updated successfully |
| Flow of activities(Actor) | 1. Admin selects "Employee Management"  1.2 Admin chooses to add, update, or view employee details |
| Flow of activities(System) | 1.1 Display "Employee Management" interface  1.2 Display options for adding, updating, or viewing employee details  1.3 Save updated or new employee details to the system |

| Exception  conditions | 1.2.1 If the user has no previous order, these functions will  have no data |
| --- | --- |

*Table 11. Specification sheet use case View orders*

### Customer support

| Use Case Name | Customer support |
| --- | --- |
| Scenario | Customer needs assistance or has an inquiry |
| Triggering event | Customer encounters an issue or has a question |
| Brief description | Allows customers to contact support for assistance or inquiries |
| Actors | Customer, Support Agent |
| Related use cases | Must have |
| Stakeholders | Customer Service Department |
| Preconditions | - Customer has successfully logged into the system  - Customer has internet access |
| Post conditions | - Customer inquiry is logged and addressed by a support agent |
| Flow of  activities(Actor) | 1. Customer selects "Contact Support"  1.2 Customer submits an inquiry or request for assistance |
| Flow of activities(System) | 1.1 Display "Contact Support" interface  1.2 Log the customer inquiry  1.3 Assign the inquiry to a support agent  1.4 Notify the customer of receipt and expected response time |

*Table 12. Specification sheet use case Customer support*

### 

### Receiving and processing customer booking

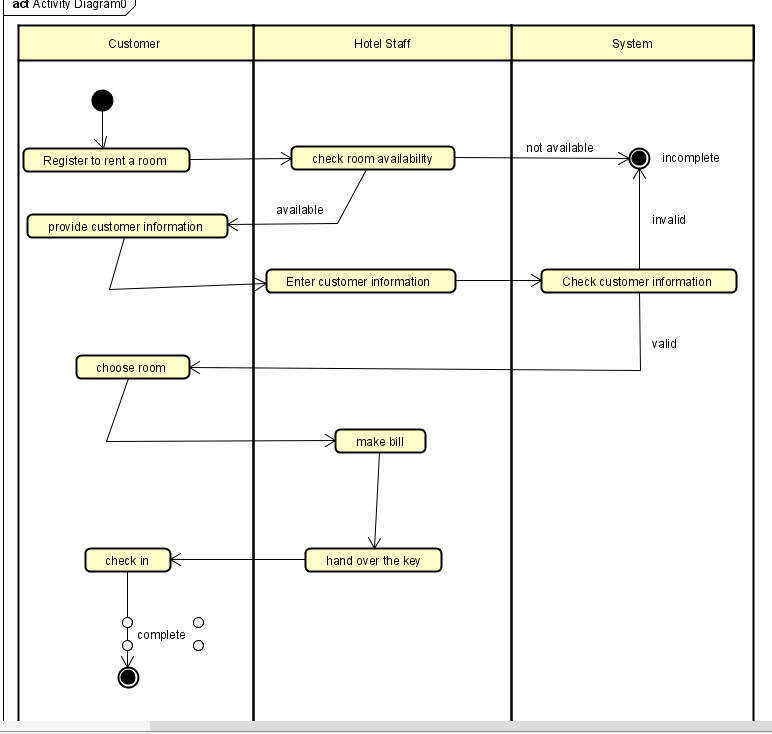
| Use Case Name | Hotel staff receives and processes guest booking |
| --- | --- |
| Scenario | Guest places a booking |
| Triggering event | Guest places an order |
| Brief description | Allows hotel staff to receive, process booking |
| Actors | Receptionist, Customer, Hotel Staff |
| Related use cases | Must have |
| Stakeholders | Hotel Management, Guests |
| Preconditions | - Guest has successfully placed an order  - Hotel staff has access to the system and internet |
| Post conditions | - Guest order is recorded and fulfilled |
| Flow of activities(Actor) | 1. Guest places an order  1.2 Receptionist or hotel staff receives the order  1.3 Hotel staff processes and fulfills the order |

|  |  |
| --- | --- |
| Flow of activities(System) | 1.1 Display "Order Management" interface  1.2 Log the guest order  1.3 Notify relevant hotel staff  1.4 Update order status as it is processed and fulfilled |

*Table 15. Specification sheet use case Product portfolio management*

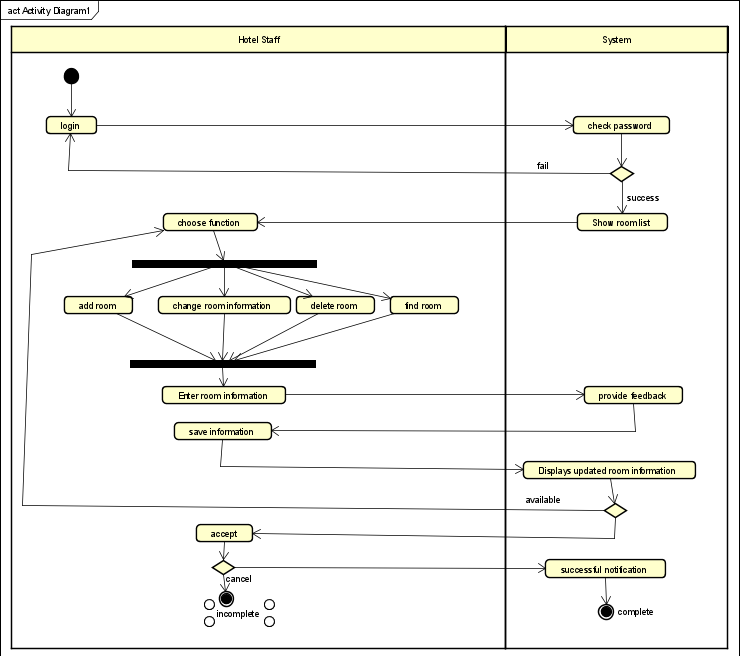
### Activity diagram

* 1. **Room rental management**

****

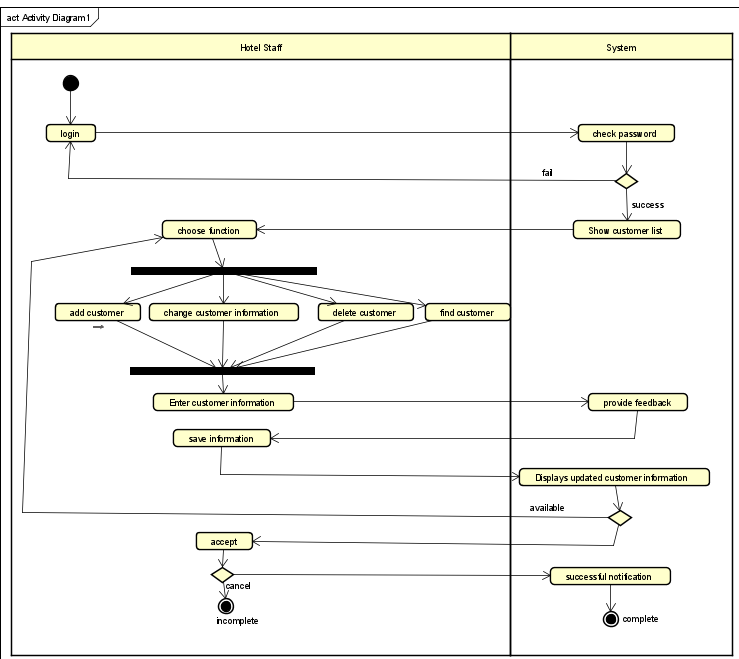
*Image 13. Room rental management activity diagram*

### Room Manager

****

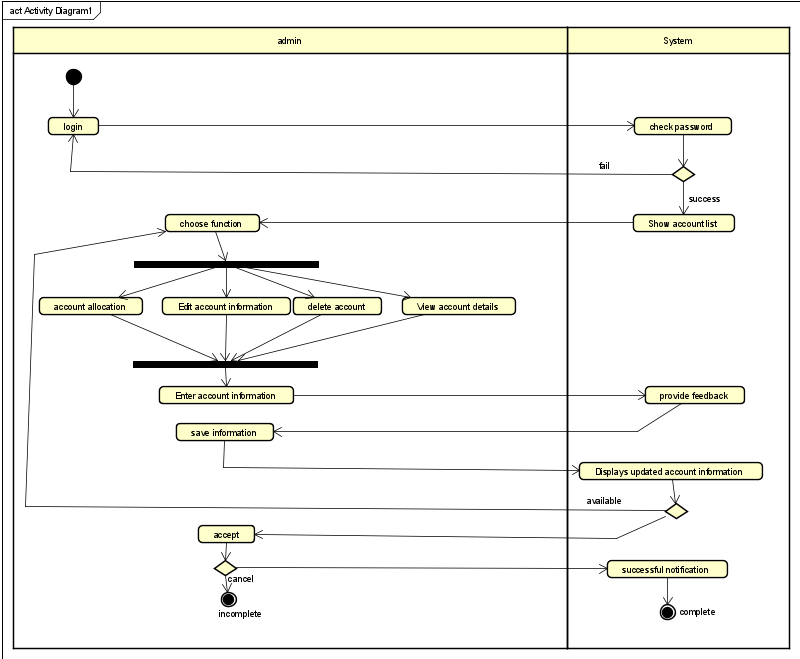
*Image 14. Room Manager activity diagram*

### Customer management



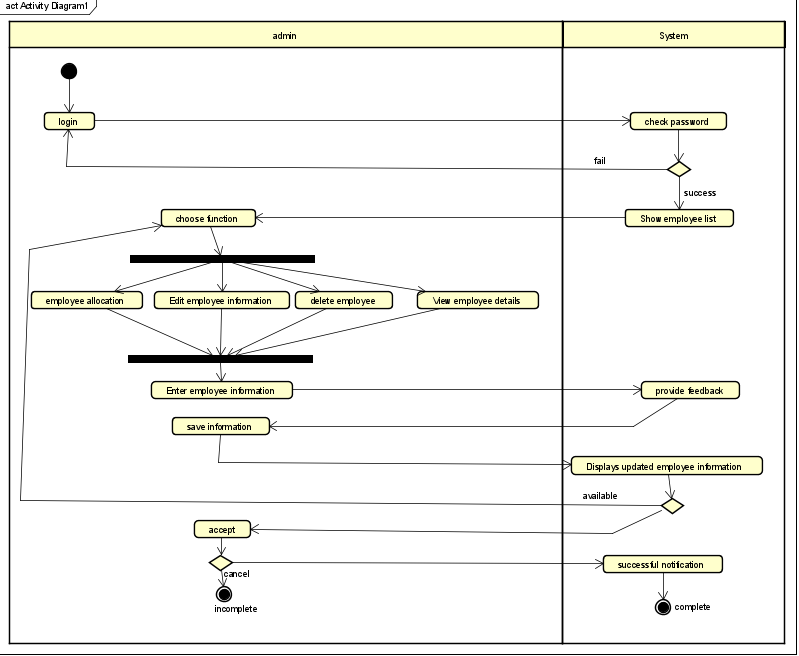
*Image 15. Customer management activity diagram*

### Account management



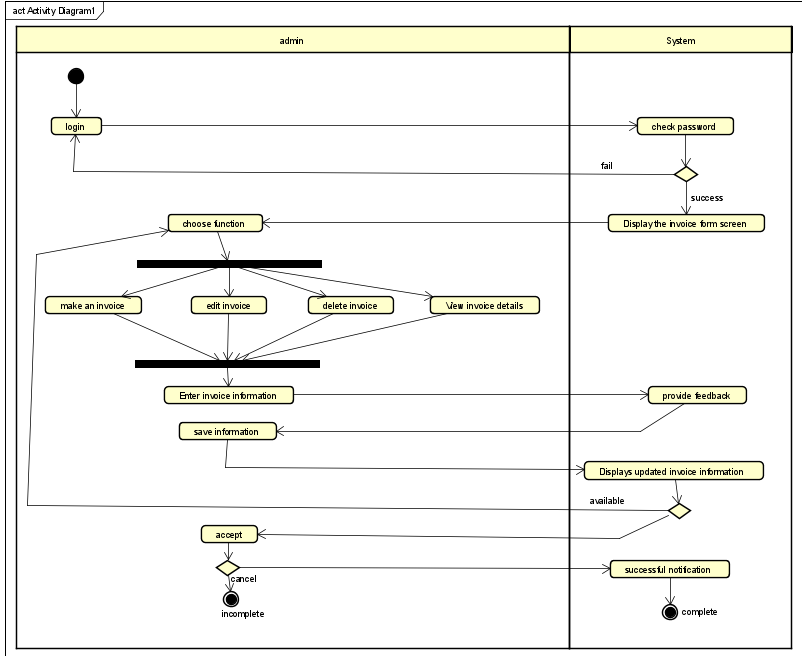
*Image 16. Account management activity diagram*

### Employee manager



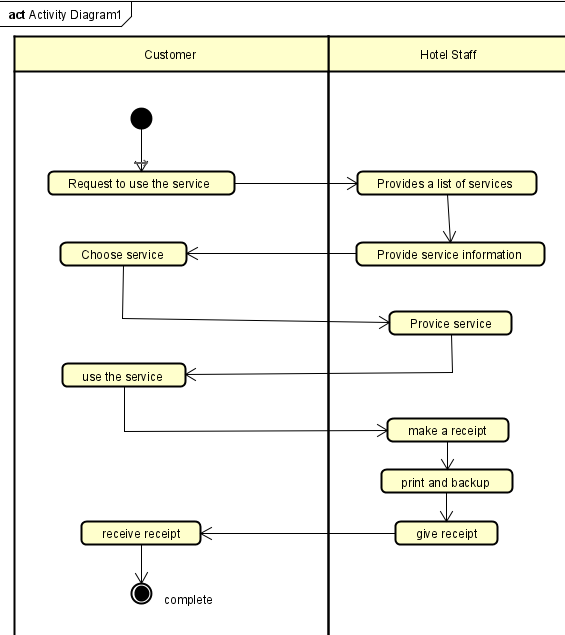
*Image 17. Employee manager activity diagram*

### Make an invoice

****

*Image 18. Make an invoice activity diagram*

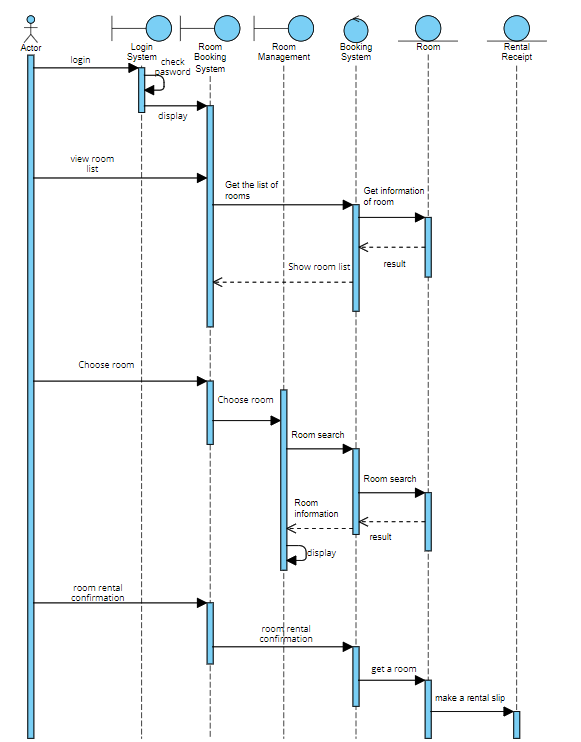
### Service rental management

****

*Image 19. Service rental management activity diagram*

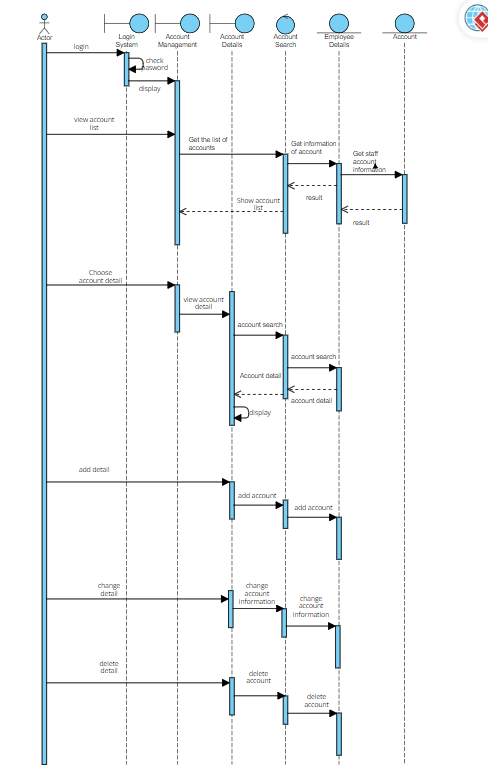
### Sequence diagram

### Room rental management

****

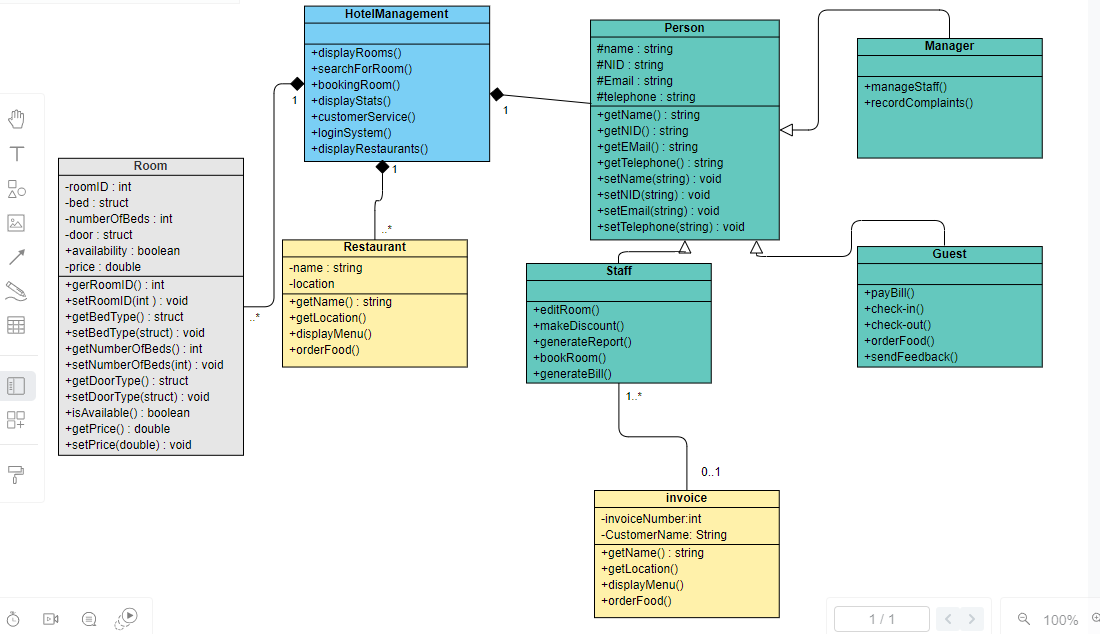
*Image 23. Room rental management sequence diagram*

### Account management

****

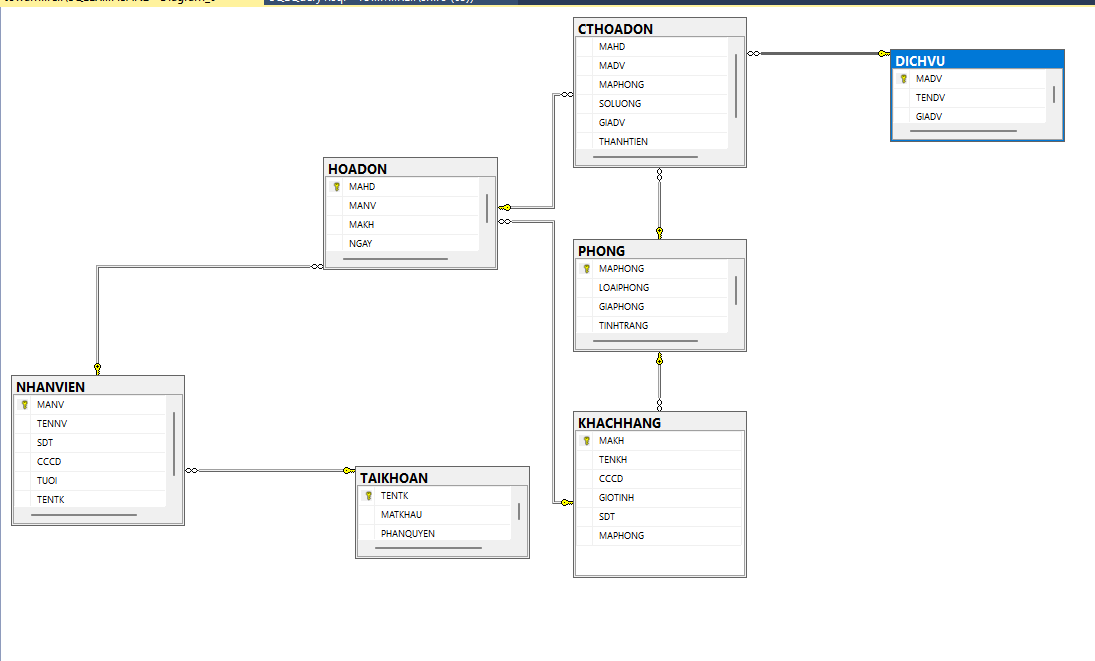
*Image 26. Account management sequence diagram*

### Class diagram

****

*Image 34. Class diagram*

### Database design

**

### 

**CHAPTER IV: ACTUAL PROCESS**

1. **Software development life cycle model**

The waterfall model is a traditional and highly structured software development methodology, introduced by Winston Royce in 1970. It is the earliest and one of the most straightforward models in the Software Development Life Cycle (SDLC). The waterfall model divides the software development process into a series of predefined, sequential phases, each of which must be completed before moving on to the next. This model is often likened to a waterfall, where each phase cascades into the next, ensuring a systematic and disciplined approach to software development. In the context of a hotel management system, the waterfall model can provide a clear roadmap for developing a robust and efficient system to manage hotel operations.

### Stages of Development in the Waterfall Model for a Hotel Management System

#### 1. Requirements Analysis

The first stage involves collecting and analyzing the system requirements. For a hotel management system, this includes understanding the needs of various stakeholders such as hotel staff, managers, and guests. Detailed requirements are gathered through interviews, surveys, and document reviews. This phase results in a comprehensive requirements specification document that outlines what the system should achieve. Key aspects might include:

* Guest check-in and check-out processes.
* Room booking and reservation management.
* Billing and invoicing.
* Inventory management.
* Employee management.

#### 2. System Design

In the system design phase, the software system is analyzed and designed based on the requirements gathered. This phase determines the overall architecture of the hotel management system. Key tasks include:

* Designing the user interface for different modules like reservations, billing, and employee management.
* Specifying the hardware and software requirements needed to support the system.
* Creating database schemas to handle guest information, room details, and financial transactions.

#### 3. Building the System (Implementation)

During the implementation phase, the system is developed in small units, which are later integrated. Each unit corresponds to a specific functionality of the hotel management system. For instance:

* Developing a module for managing room reservations.
* Implementing a billing and invoicing system.
* Creating an interface for managing employee schedules and payroll. Each unit is developed and tested independently in a process known as unit testing, ensuring that each component functions correctly before integration.

#### 4. System Testing

After individual units are tested, they are integrated into a complete system. This phase involves rigorous testing to ensure that all parts of the hotel management system work together seamlessly. Key activities include:

* Integration testing to check the interaction between different modules.
* System testing to validate the system's overall functionality against the requirements.
* Debugging any issues that arise during testing to ensure a reliable system.

#### 5. System Deployment

Once testing is complete and the system is verified to be functional and bug-free, it is deployed to the customer. For a hotel management system, deployment involves installing the system on the hotel's servers and setting up necessary hardware. Training sessions may also be conducted for hotel staff to ensure they can effectively use the system. The product is then released to the market for use in managing hotel operations.

#### 6. System Maintenance

After deployment, the system enters the maintenance phase. This involves regular updates and modifications to address any issues or changes requested by the client. For a hotel management system, maintenance activities might include:

1. Updating the system to incorporate new features based on user feedback.
2. Fixing bugs that are identified post-deployment.
3. Ensuring the system remains compatible with new hardware or software environments.

The waterfall model provides a clear and structured approach to software development, making it suitable for projects with well-understood requirements and little expected change. In the context of developing a hotel management system, the waterfall model ensures that each phase is thoroughly planned and executed, resulting in a comprehensive solution that meets the needs of hotel operations. By following the waterfall model, developers can systematically address each aspect of the system, from requirements analysis through to maintenance, ensuring a robust and efficient management system for hotels.

### Advantages of the Waterfall Model

The waterfall model offers several key advantages that make it an attractive choice for certain types of projects, especially in structured environments:

1. **Simplicity and Ease of Use**:
   * The waterfall model is straightforward, with each phase clearly defined and following a sequential order. This makes it easy to understand and apply, even for teams with less experience in project management.
2. **Ease of Management**:
   * Each phase in the waterfall model has specific deliverables and a review process. This fixed, linear approach simplifies project management, as progress can be easily tracked, and milestones are clearly defined.
3. **Explicit Phase Completion**:
   * Stages in the waterfall model are processed and completed one at a time. This explicit completion ensures that each phase is fully developed before moving on to the next, reducing the risk of overlooking critical elements.
4. **Comprehensive Documentation**:
   * The waterfall model emphasizes thorough documentation at each stage of the development process. This comprehensive documentation provides a clear record of the project’s progress and can be useful for training new team members or maintaining the system in the future.
5. **Suitability for Small Projects with Clear Requirements**:
   * For projects where requirements are well-understood from the outset and unlikely to change, the waterfall model is particularly effective. Its structured nature ensures that all aspects of the project are covered systematically.

### Defects of the Waterfall Model

Despite its advantages, the waterfall model has several significant drawbacks that can impact its effectiveness, particularly in more dynamic or complex project environments:

1. **Limited Flexibility**:
   * The rigidity of the waterfall model makes it difficult to accommodate changes once a phase has been completed. If an error or change in requirements is discovered mid-process, the project may need to revert to the initial stages, causing delays and increased costs.
2. **High Risk and Uncertainty**:
   * The waterfall model is less suited to projects where requirements may evolve over time. The inability to make incremental changes can result in a high risk of project failure if initial assumptions prove incorrect.
3. **Unsuitability for Complex Projects**:
   * Projects with high complexity or those requiring significant innovation may struggle under the waterfall model. The model’s linear approach does not easily accommodate iterative testing and refinement needed in complex projects.
4. **Difficulty in Measuring Progress**:
   * While the waterfall model’s stages provide clear checkpoints, it can be challenging to measure progress within each phase. Long phases without visible progress can lead to uncertainty and delays in the overall project timeline.

### Reason for Choosing the Waterfall Model

Despite its limitations, there are compelling reasons to choose the waterfall model for certain projects:

1. **Simplicity and Understandability**:
   * The waterfall model’s straightforward, sequential approach makes it easy to understand and implement. This simplicity is beneficial for teams or stakeholders who prefer a clear, linear progression of tasks.
2. **Effective Work Dissemination and Assignment**:
   * The model’s distinct phases allow for effective delegation of tasks. Team members can focus on specific stages without worrying about overlapping responsibilities, improving efficiency and accountability.
3. **Clear Phase Definition**:
   * Each phase in the waterfall model has well-defined objectives and deliverables. This clarity helps ensure that all necessary activities are completed thoroughly, reducing the likelihood of missed steps.
4. **Suitability for Projects with Stable Requirements**:
   * The waterfall model is particularly suitable for projects where the requirements are well-defined and unlikely to change. In such cases, the model’s structured approach ensures that all aspects of the project are comprehensively addressed from the beginning.
5. **Final Project Implementation**:
   * For final projects or those with minimal expected changes, the waterfall model provides a reliable framework. Its emphasis on upfront planning and detailed documentation ensures that the project is executed according to predefined specifications, with less need for mid-course corrections.

### Application in Hotel Management System

In the context of a hotel management system, the waterfall model can be particularly effective. Hotel management systems often have well-defined requirements, such as reservation management, billing, and customer information tracking, which do not change frequently. The waterfall model’s structured approach ensures that each functional area is thoroughly planned, implemented, and tested before moving on to the next, reducing the risk of integration issues and ensuring a high-quality final product.

By following the waterfall model, the development team can create a robust and reliable hotel management system that meets the specific needs of the hotel, with clear documentation and well-defined processes ensuring ease of maintenance and future enhancements.

### Architecture use

| **Programming language used** |
| --- |
| C# |

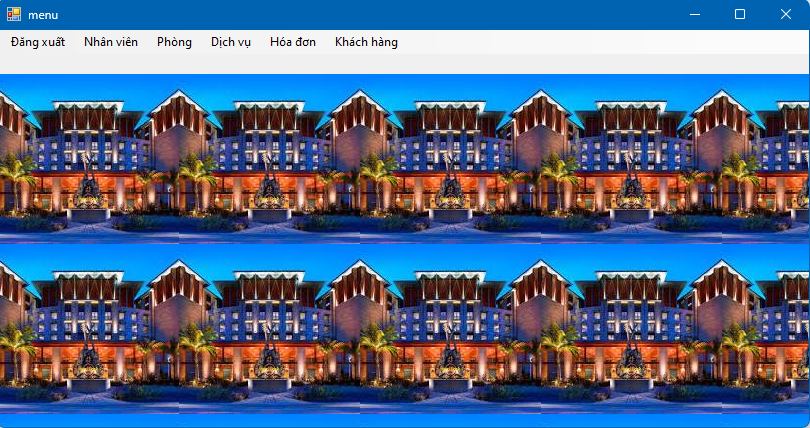
### CHAPTER V: DEMO CODE

1. **Customer interface**

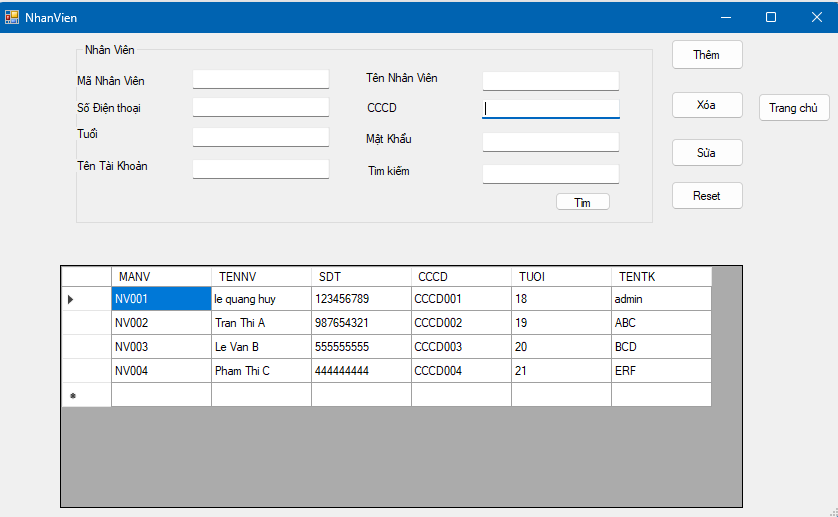
*.*

**

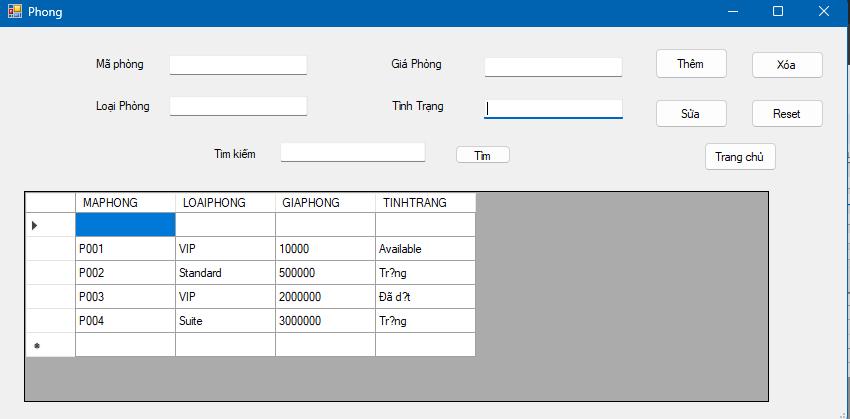
*Login screen*

**

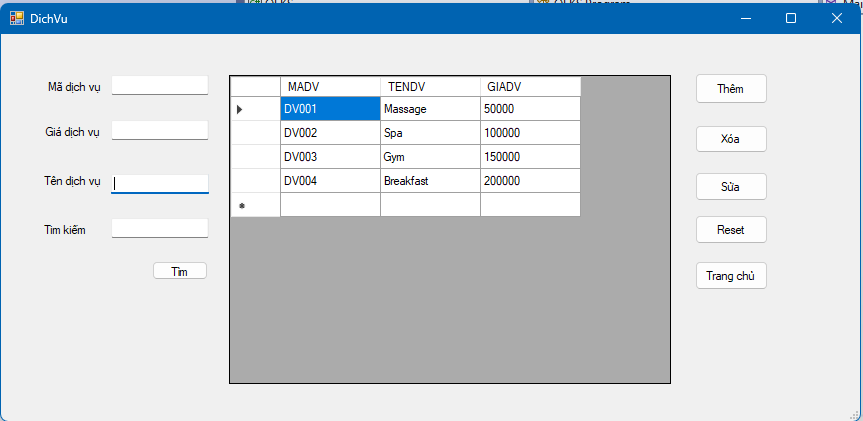
*Menu screen*

**

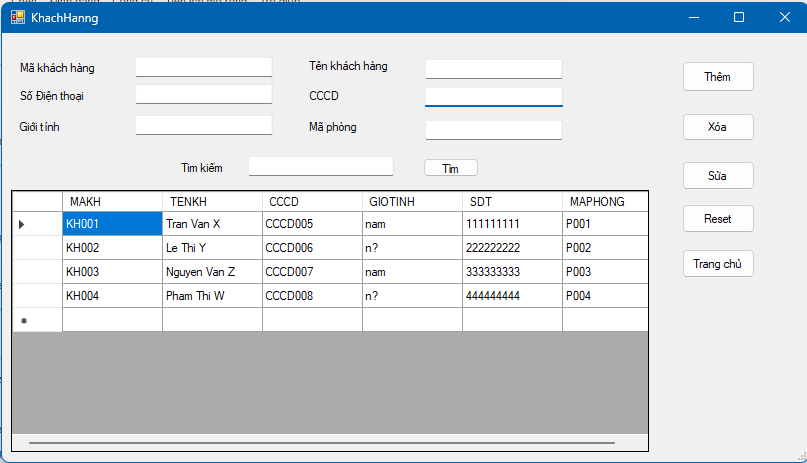
*Staff screen*

**

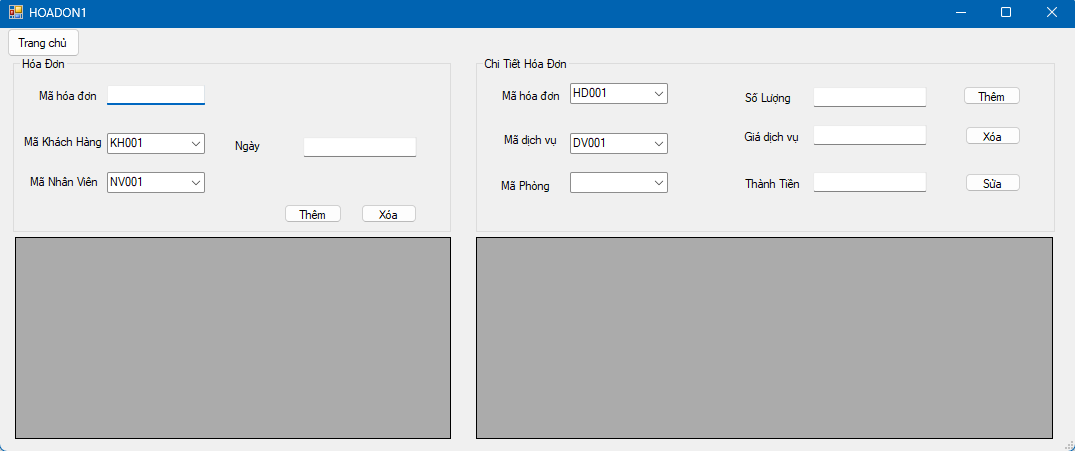
*Type room screen*

**

*service screen*

**

*customer management screen*

**

*payment screen*

### CHAPTER VI: TEST CASE AND UNIT TEST

| **ID** | **Unit test** | **Test case** | **Result** |
| --- | --- | --- | --- |
| 1 | Owner login | Enter “admin” for username and  “1234567” for password | Fail |
| 2 | Owner login | Enter “admin” for username and  “12345” for password | Success |
| 3 | Add Staff information | Press Staff button -> fill information \_> press add button | Success |
| 4 | Delete/change Staff information | Press Staff button -> fill information-> make a change | Success |
| 5 | Room type | Press room button -> fill information | fail |
| 6 | Service | Press service button -> fill information | fail |
| 7 | Make an invoice | Press invoice button -> fill information | Fail |
| 8 | Logout | Press logout button | Success |

REFERRENCE

<https://www.geeksforgeeks.org/>

<https://www.visual-paradigm.com/>

UPDATE

https://github.com/lequanghuy5/final-project-cnpm