

**BTIS 3102**

**FINAL YEAR PROJECT I**

**Hotel Management System**

**Ang Sheng Yang (B180226C)**

**Lecturer: Mr. So Yong Quay**

**27th June 2019**

**Department of Computer Science**

**Faculty of Engineering and Information Technology**

# Abstract

The Hotel Management System is a web-based system that will let the hotel company to deal with the works in hotel. The receptionist can add the additional service and print the invoice for customer. Furthermore, the admin is able to manage the inventory stock of hotel with the Hotel Management System. Moreover, the membership of the customer and promotions also can be managed by the Hotel Management System to easily offer the discount for the customer. Besides that, the admin is able to manage the suppliers such as the laundry supplier or the inventory supplier and create purchase order to them. Last but not least, the room status are shown to the housekeeping department and easily to let them to do their work. The customer can check the availability of room before book the room. The customer is able to do the online booking with their internet connection and complete the payment by online.

Keyword: Print Invoice, Inventory, Purchase order, Supplier, Customer Membership, Housekeeping, Search Availability Room, Online Reservation and Online Payment.

# Acknowledgements

First of all, I would like to thank my lecturers that help me in completing this project in Southern University College. I want to especially thank to my lecturer Mr So Yong Quay for the endless amount of advice given to me to help improve this project. Mr So always provides the best solution and advises and help me to have the more understanding of the system in nowadays.

Secondly, I would also like to thank to my mother that providing her experience of the business process that can give a lot of help in designing the system. Furthermore, I would also like to thank to my father that provide the financial help to me for my studying. When I encountered the problems, they always provide their opinion without any of unwilling.

Last but not least, I would like to thank to my classmates and friends that offering the help to me when I face the problems. I feel that I am grateful in making the friendship with them. It is because they always offer the help in answering my confusion in the question.

Contents

[Abstract I](#_Toc19000887)

[Acknowledgements II](#_Toc19000888)

[1.0 Introduction 1](#_Toc19000889)

[2.0 Literature review 4](#_Toc19000890)

[3.0 Methodology 13](#_Toc19000891)

[4.0 Design 15](#_Toc19000892)

[**4.1 DFD diagram** 15](#_Toc19000893)

[***4.1.1 Context level*** 15](#_Toc19000894)

[***4.1.2 DFD Level 0*** 16](#_Toc19000895)

[***4.1.3 DFD Level 1*** 17](#_Toc19000896)

[**4.2 ER-diagram** 29](#_Toc19000897)

[**4.3 Class diagram** 30](#_Toc19000898)

[**4.4 Use case diagram** 31](#_Toc19000899)

[***4.4.1 Front-desk system*** 31](#_Toc19000900)

[***4.4.2 Payment system*** 32](#_Toc19000901)

[***4.4.3 Reservation detail system*** 33](#_Toc19000902)

[***4.4.4 Customer account system*** 34](#_Toc19000903)

[***4.4.5 Inventory system*** 35](#_Toc19000904)

[***4.4.6 Purchase order system*** 36](#_Toc19000905)

[***4.4.7 Supplier system*** 37](#_Toc19000906)

[***4.4.8 Room information system*** 38](#_Toc19000907)

[***4.4.9 Housekeeping system*** 39](#_Toc19000908)

[***4.4.10 Employee system*** 40](#_Toc19000909)

[***4.4.11 Promotion and reward system*** 41](#_Toc19000910)

[***4.4.12 Online booking system*** 42](#_Toc19000911)

[**4.5 Use case description** 43](#_Toc19000912)

[***4.5.1 Front-desk system*** 43](#_Toc19000913)

[***4.5.2 Payment system*** 47](#_Toc19000914)

[***4.5.3 Reservation detail system*** 49](#_Toc19000915)

[***4.5.4 Customer account system*** 51](#_Toc19000916)

[***4.5.5 Inventory system*** 53](#_Toc19000917)

[***4.5.6 Purchase order system*** 57](#_Toc19000918)

[***4.5.7 Supplier system*** 64](#_Toc19000919)

[***4.5.8 Room information system*** 66](#_Toc19000920)

[***4.5.9 Housekeeping system*** 69](#_Toc19000921)

[***4.5.10 Employee system*** 72](#_Toc19000922)

[***4.5.11 Promotion reward system*** 74](#_Toc19000923)

[***4.5.12 Online booking system*** 78](#_Toc19000924)

[**4.6 Activity diagram** 81](#_Toc19000925)

[***4.6.1 Front-desk system*** 81](#_Toc19000926)

[***4.6.2 Payment system*** 82](#_Toc19000927)

[***4.6.3 Reservation detail system*** 83](#_Toc19000928)

[***4.6.4 Customer account system*** 84](#_Toc19000929)

[***4.6.5 Inventory system*** 85](#_Toc19000930)

[***4.6.6 Purchase order system*** 86](#_Toc19000931)

[***4.6.7 Supplier system*** 88](#_Toc19000932)

[***4.6.8 Room information system*** 89](#_Toc19000933)

[***4.6.9 Housekeeping system*** 90](#_Toc19000934)

[***4.6.10 Employee system*** 91](#_Toc19000935)

[***4.6.11 Promotion reward system*** 92](#_Toc19000936)

[***4.6.12 Online booking system*** 93](#_Toc19000937)

[**4.7 Sequence diagram** 94](#_Toc19000938)

[***4.7.1 Front-desk system*** 94](#_Toc19000939)

[***4.7.2 Payment system*** 95](#_Toc19000940)

[***4.7.3 Reservation detail system*** 96](#_Toc19000941)

[***4.7.4 Customer account system*** 97](#_Toc19000942)

[***4.7.5 Inventory system*** 98](#_Toc19000943)

[***4.7.6 Purchase order system*** 99](#_Toc19000944)

[***4.7.7 Supplier system*** 100](#_Toc19000945)

[***4.7.8 Room information system*** 101](#_Toc19000946)

[***4.7.9 Housekeeping system*** 102](#_Toc19000947)

[***4.7.10 Employee system*** 103](#_Toc19000948)

[***4.7.11 Promotion reward system*** 104](#_Toc19000949)

[***4.7.12 Online booking system*** 105](#_Toc19000950)

[5.0 Implementation 106](#_Toc19000951)

[6.0 Testing 111](#_Toc19000952)

[7.0 Evaluation 114](#_Toc19000953)

[8.0 Recommendations 115](#_Toc19000954)

[9.0 Conclusion 116](#_Toc19000955)

[References 117](#_Toc19000956)

[APPENDIX A – Curriculum Vitae 118](#_Toc19000957)

[APPENDIX B – User Guide 121](#_Toc19000958)

[APPENDIX C – Project Diaries 146](#_Toc19000959)

# Introduction

Nowadays ICT is already developed into high technologies era, there have being an increase in the amount of hotel because of the increase of tourists around the world. So, the hotel will need a very good in managing the information of their daily works.

Hotel Management System is an automated system which is developed for the hotel and the customer. The Hotel Management System is a web-based system and it is accessed by the internet connection to retrieve and store the data. So, the Hotel Management System will focus in two modules which are the customer module and admin module.

For the hotel, the hotel admin is able to manage the booking details easily compared to the traditional way of writing down the booking details in a piece of paper. Furthermore, hotel admin also can know the current status of room easily and assign the work for housekeeping by using the management system and do not need to go and check manually by person. Besides that, the admin can manage the inventory stock easily. The admin is able to create purchase order and receive delivery order from the supplier system instead of using paper. Moreover, the admin also can manage the discount for the membership of customer.

The customer module will allow the customer to do the searching of availability of the room on the certain date. Customer also is able to make the online reservation of the room through the website. Next, the customer also can do the online payment through the website instead of queue to make payment when check in to the hotel. Customer also can exchange the reward points of the membership with the items.

**1.1 Project Objectives**

After developing this project, I can achieve the objectives as shown as below:

* To provide a platform to let the customer online booking the hotel room easily.
* To provide a platform to let the hotel staff manage the daily operations easily.

**1.2 Personal Objectives**

After developing this project, I can achieve the personal objectives as shown as below:

* To increase my web programming skills.
* To improve the design and analysis skills of the project.
* To learn how to allocate the time and resources to the project and produce a well-planned schedule.
* To learn how to develop a formal report

**1.3 Rationale for project choice**

My rationale for this project choice is not only to improve my programming skill, but also improve my skill in handling a project. Furthermore, I can also gain the knowledge of how the hotel is processing their works in the daily. The most of the important skill that I need to improve is the analysis and design skill. It is because the hotel has some new complex processes and will also lead to the difficulty when designing the system. So, I will like to learn how to design an efficient database and improve the designing skill of database during this project.

I choose this topic because I also would like to develop a hotel management system that I do not develop before and I am interested in the hotel business processes. Although I will have some difficulties in develop the hotel system that I do not have any related knowledge to the context of hotel business, but I would still like to put more efforts in understanding of the hotel business and develop a usability of system to the hotel.

**1.4 Background Research**

Tune Hotels [7] – A web-based system



*Figure 1.4.1 Tune Hotels Logo*

Tune Hotels is a chain of hotels that offer accommodation facilities in a range of countries for travalers. Tune Hotels core proposition is to offer a great night’s sleep at a great price. All of the Tune Hotels feature space-efficient, streamlined rooms focusing on high-quality basics: 5-star beds and powerful hot showers.

Tune Hotels system provides the customer to search the availability of the room and shows the prices for the different types of room. The customer also can choose the range of check-in and check-out date for their booking. The customers are able to choose whether they want to book for the normal, or overnight, or the day use rooms only. Furthermore, customer also can enter the promo code that given by the tune hotels to get the discount for the booking room process. Once the customer confirm with their options, customer need to create the account and provide their personal information in the field provided to complete the booking process. Customer can choose their prefer option to complete the payment.

Hence, Tune Hotels requires network connection to view the booking details that booked by the customer. This system may use following tool or technique MySQL, PHP , Javascript and web service. Therefore, I may use the same technology to create the system in this project.

In conclusion, I will decide to develop a hotel management system that will fit to the other hotels in nowadays. The hotel management system should be able to reduce the efforts in completing the booking process and the hotel processes in daily.

# 2.0 Literature review

In this part, I am going to explore the research related to my project that have been conducted by other academic previously. Hotel management system is a web-based system which comprises PHP, MySQL, and Web Service. A web-based system is one of the system that can be accessed through a web browser and typically off-site or off of the network that you are currently on. Hence, I have one academic scholar and one academic book related to the hotel to review which are Hotel Operations: A room with a technological view by Robert Grimes and Check in Check Out by S.Ghosh.

**2.1 Hotel Operations: A room with a technological**

According to the Hotel Operations: A room with a technological view by Robert Grimes [2], he mentioned about the hotel systems can be very complex. It is because the hotel systems is depended on the characteristics of the hotel. For example, he said that the hotel is like a small city and it can contain restaurants, retail stores and even gymnasiums. The one of the important of hotel features is the front desk. The front desk operation should be able to help the customer to check-in, check-out and assign room for them. In the past, the front desk does not have the system that helps the customer to do those operations but using the room card number to know whether the rooms are assigned or not. It is very inefficiency when sometimes customer do not return the room card number to the front desk when checking out and assign the room for the new incoming of customer.

Furthermore, the telephone system within the room has data ports for computer link-ups as well as voice-mail and voice messaging. Some telephone systems are usually used by the housekeeping to allow the maids to enter the room and then send room status information to the property management system. Moreover, the guest histories should also be tracked. By tracking the guest histories, the hotel can know the preferences of the guests and provide the better service for the next time arrival. So, the guest histories features should be implemented in the front desk to allow the receptionist to assign the type of room for the guests according to their previous preferences. By doing in this way, the regular customers are able to reduce the time when making the reservation of room.

**2.2 Check in Check Out**

Next, according to the Check in Check Out by S.Ghosh [1], the staff of the reservation department should always maintain the accurate room inventory information and also maintain an up-to-date inventory list of available rooms for the various dates. The room information is one of the most important information needed to provide to the customer. It is because the reception staffs need to assign the rooms to guest coming with reservation as well as the guests coming without reservation. So, the front desk system will help the reception staffs to deal with these problems. Besides that, there are a several type of rooms in hotel, below are the information of some type of rooms:

1. Single room

* A single room is a room which has a single bed. Normally assigned to one person.

1. Double room

* A double room is a room which has a double or a composite bed. Normally assigned to two person.

1. Twin room

* A twin room is a room which has two single bed. Normally assigned to two person.

1. Twin double room

* A twin room is a room which two double beds separated from each other. Normally assigned to four person. Also called as family room.

1. King room

* A king room is a room with a king-sized bed. It may be occupied by one or more persons.

1. Queen room

* A queen room is a room with a queen-sized bed. It may be occupied by one or more persons.

In conclusion, the hotel management is more difficult to handle than I thought. Hence, I decide to create a hotel management system for some of the hotel to handle their daily operations easily in this era of advanced cloud technology.

**2.3 Research**

***2.3.1 PHP: Hypertext Preprocessor(PHP)***



*Figure 2.3.1: Logo of PHP*

PHP [4] is a widely-used open source general-purpose scripting language that is especially suited for web development and can be embedded into HTML. Since the PHP is an open source language, so it will be easier for me to develop the hotel management system because many discussions and solutions are provided on the forum. Moreover, hotel management system is a web-based system and it will need the PHP to do server side scripting.

***2.3.2 MySQL***



*Figure 2.3.2: Logo of MySQL*

MySQL is an open-source relational database management system (RDBMS).[3] Its name is a combination of "My", the name of co-founder Michael Widenius's daughter, and "SQL", the abbreviation for Structured Query Language.

MySQL is free and open-source software under the terms of the GNU General Public License, and is also available under a variety of proprietary licenses. Since it is free and open source, I will use MySQL as my RDBMS to save the cost in purchase a paid RDBMS. MySQL is easier to find solution on the internet because it is also an open source software.

***2.3.3 Web Service***

A **Web Service** is a client server application or application component for communication. It is also a method of communication between two devices over network. Moreover, web service is a collection of standards or protocols for exchanging information between two devices or application. For example, java application can interact with Java, .Net and PHP applications. So web service is a language independent way of communication.

**2.4 Outline of scope of the system**

|  |
| --- |
| **Project Title:** Hotel Management System |
| **Project Justification:** Provides management system for hotel staff and online booking service for customer. |
| **Product Characteristics and Requirements:**   1. Front-desk system 2. Sales system 3. Reservation detail system 4. Customer account system 5. Inventory system 6. Purchase order system 7. Supplier system 8. Room information system 9. Housekeeping system 10. Promotion and rewards system 11. Online booking system |
| **Detail**   1. **Front-desk system**  * Assign room and perform check-in, check-out processes.  1. **Sales system**  * Add additional service and print invoice.  1. **Reservation detail system**  * View, edit and delete the reservation details.  1. **Customer account system**  * View, edit and delete the user account details.  1. **Inventory system**  * View, edit and delete the inventory details.  1. **Purchase order system**  * Create purchase order to the supplier  1. **Supplier system**  * View, edit and delete the supplier details.  1. **Room information system**  * View, edit and delete the room details.  1. **Housekeeping system**  * Assign the housekeeping to the room  1. **Promotion and reward system**  * View, edit and delete the promotion detail * View, edit and delete the reward detail  1. **Online Booking system**  * Book the reservation online. * View the booked room details * Search the availability of the room |

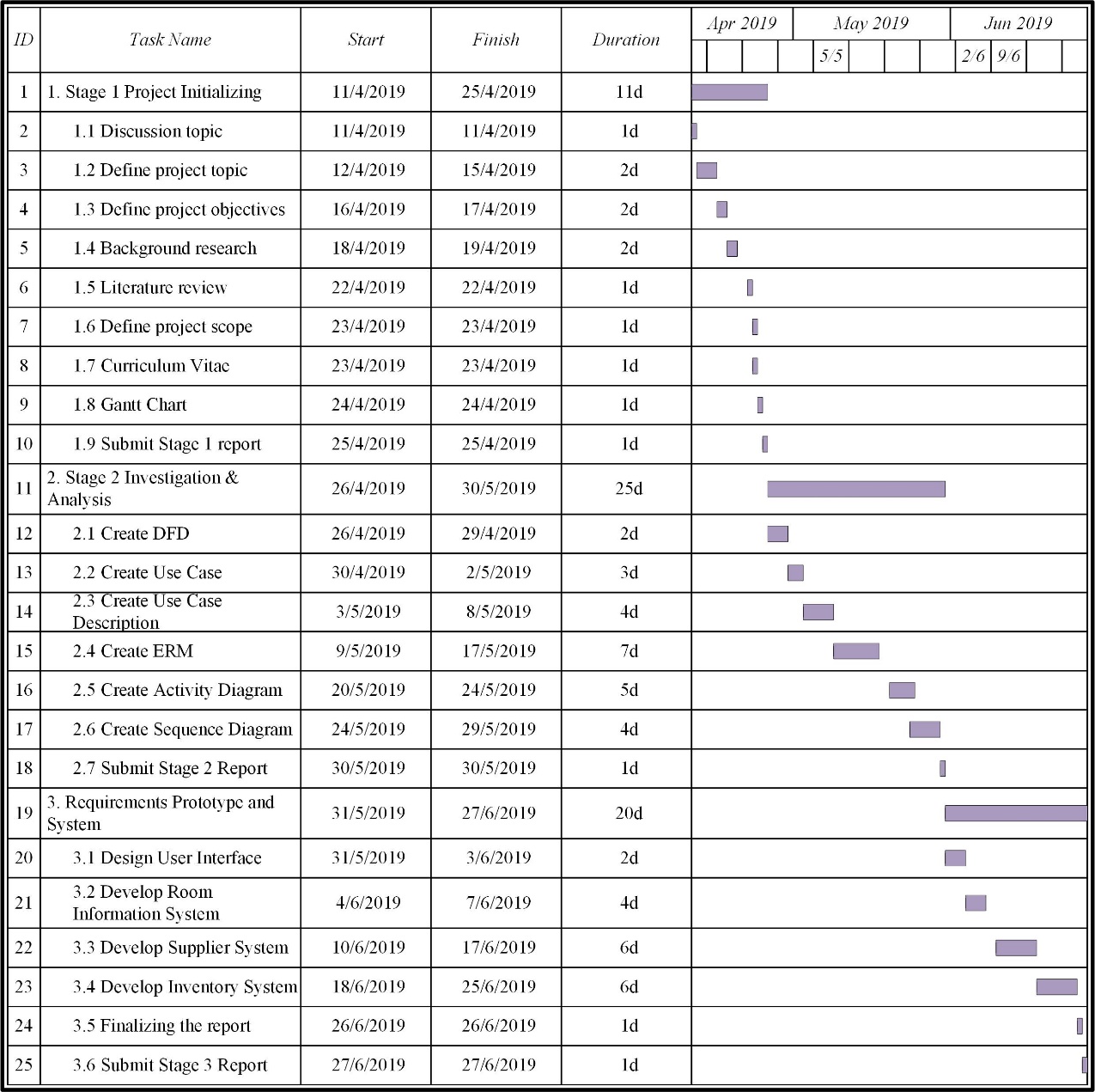
My project will focus on a small hotel management system and the system consists of two modules which are admin module and customer module. The front-desk system will allow the receptionist to assign the room for the customer that without make the reservation. The receptionist can also enter the check-in date and check-out date for the customer by the front-desk system. The receptionist can add the additional service such as add additional bed to the booked room if allowable and print the invoice for the customer.

Reservation detail system allows the staff to view and manage the reservation details from the customer. Customer account system allows the staff to view and manage the customer account that registered by them. Inventory system allows the staff to view and manage the inventory such as toiletries. Purchase order system allows the staff to create purchase order to the toiletries supplier and laundry supplier. Supplier system allows the staff to view and manage the supplier details that they prefer to make order with them. Room information system allows the staff to view and manage the room details such as the type of room and rates of room. Housekeeping system will allow the staff to assign the housekeeping service to the room that needed. Promotion system allows the admin to view and manage the promotion details.

Online booking system is the system which is also a website for the customer use only. The customer needs to register an account to do the reservation online. Customer also can search the availability room before proceed to the booking process. After the customer enters the date that they prefer to make the reservation, the price and available room will be shown to the customer. After the customer confirm with their prefer price and available room, they can proceed to complete the payment process. Furthermore, the customer can accumulate the point when booking the reservation of room and they can claim the rewards from the website too.**2.5 Financial Justification**

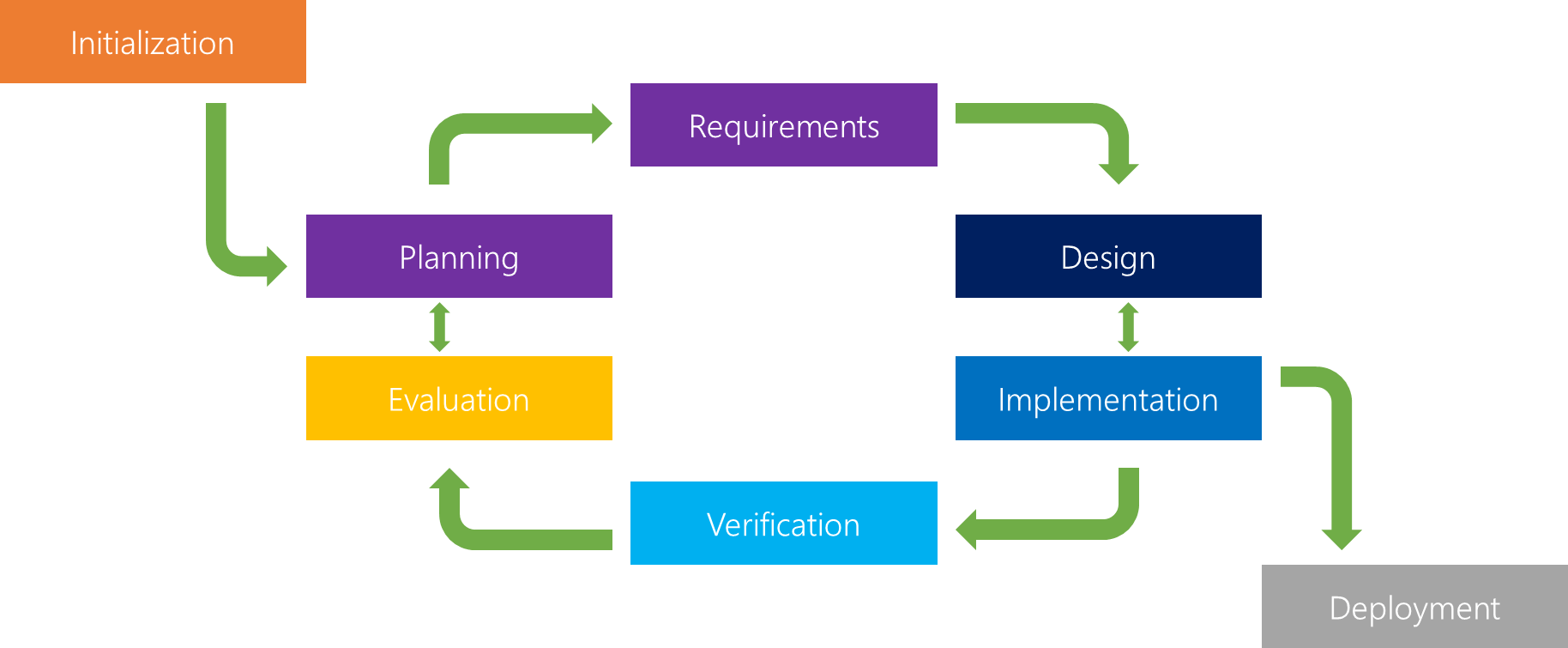
|  |  |  |  |
| --- | --- | --- | --- |
|  | **Item** | **Qty** | **Total(RM)** |
| **Hardware** | **ASUS computer** | **1** | **RM 2300.00** |
| **Software** | **Microsoft Office** | **1** | **RM 150.00** |
|  | **Microsoft Visio 2010** | **1** | **RM 30.00** |
|  | **Sublime Text** | **1** | **RM 0.00** |
|  | **MySQL** | **1** | **RM 0.00** |
|  | **PHP** | **1** | **RM 0.00** |
| **Other** | **Transportation Fee** | **-** | **RM 300.00** |
|  | **Internet Subscription Fee** | **-** | **RM 300.00** |
|  | **Printing Cost** | **-** | **RM 500.00** |
|  |  | **Total:** | **RM 3480.00** |

**2.6 Project Schedule**

****

# Methodology

**3.1 Iterative model**



*Figure 3.1 Iterative model*

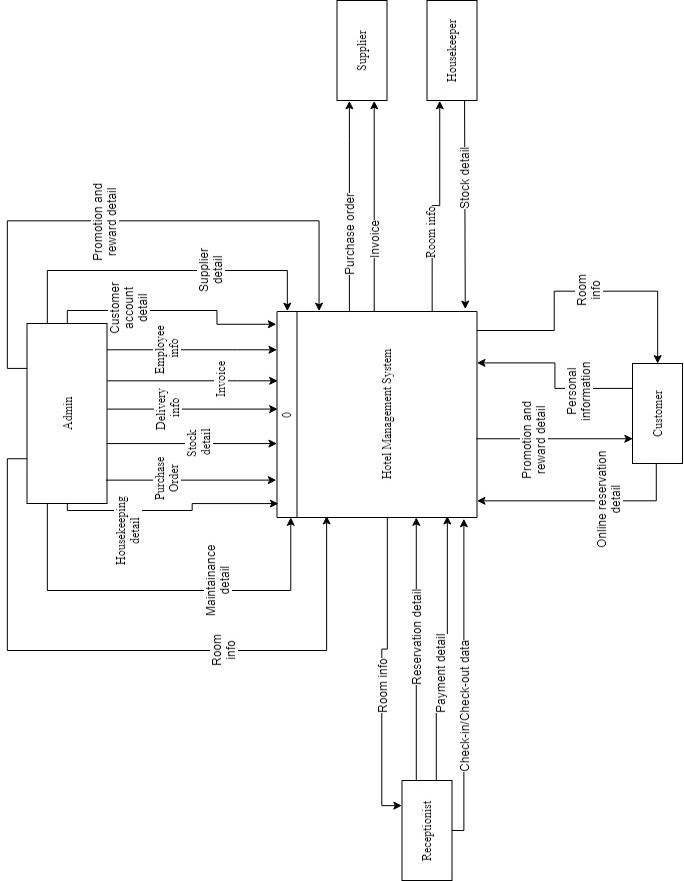
Since I am not familiar with the requirements of hotel management system, so I will use the iterative model to develop the system. According to the iterative model [5], an iterative life cycle model does not start with a full specification of requirements. In this model, the development begins by specifying and implementing just part of the software, which is then reviewed in order to identify further requirements. Moreover, in iterative model, the iterative process starts with a simple implementation of a small set of the software requirements, which iteratively enhances the evolving versions until the complete system is implemented and ready to be deployed. Each release of Iterative Model is developed in a specific and fixed time period, which is called iteration.

The process of Iterative Model is cyclic, unlike the more traditional models that focus on a rigorous step-by-step process of development. So, the iterative model is suitable to make changes in requirements when I have misunderstood the requirements of the hotel management system. In this process, once the initial planning is complete, a handful of phases are repeated again and again, with the completion of each cycle incrementally improving and iterating on the software. Other phases of the iterative model are described below:

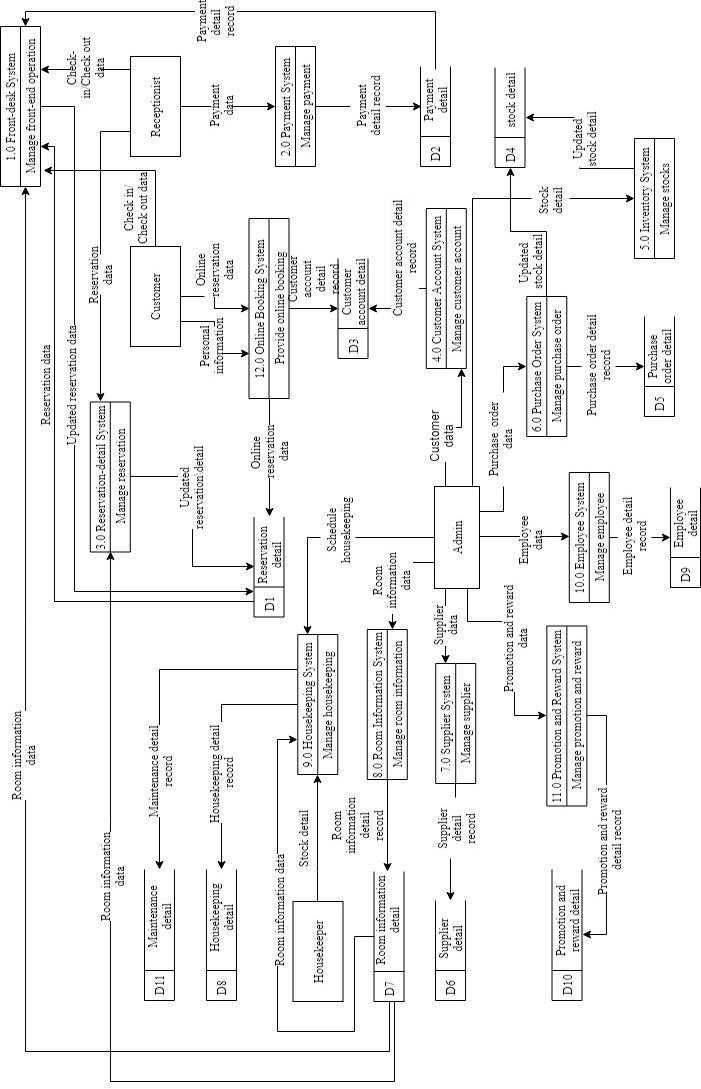
1. **Planning Phase:** This is the first stage of the iterative model, where proper planning is done by the team, which helps them in mapping out the specifications documents, establish software or hardware requirements and generally prepare for the upcoming stages of the cycle.
2. **Analysis and Design Phase:** Once the planning is complete for the cycle, an analysis is performed to point out the appropriate business logic, database models and to know any other requirements of this particular stage. Moreover, the design stage also occurs in this phase of iterative model, where the technical requirements are established that will be utilized in order to meet the need of analysis stage.
3. **Implementation Phase:** This is the third and the most important phase of the iterative model. Here, the actual implementation and coding process is executed. All planning, specification, and design documents up to this point are coded and implemented into this initial iteration of the project.
4. **Testing Phase:** After the current build iteration is coded and implemented, testing is initiated in the cycle to identify and locate any potential bugs or issues that may have been in the software.
5. **Evaluation Phase:** The final phase of the Iterative life cycle is the evaluation phase, where the entire team along with the client, examine the status of the project and validate whether it is as per the suggested requirements.

# Design

## **4.1 DFD diagram**

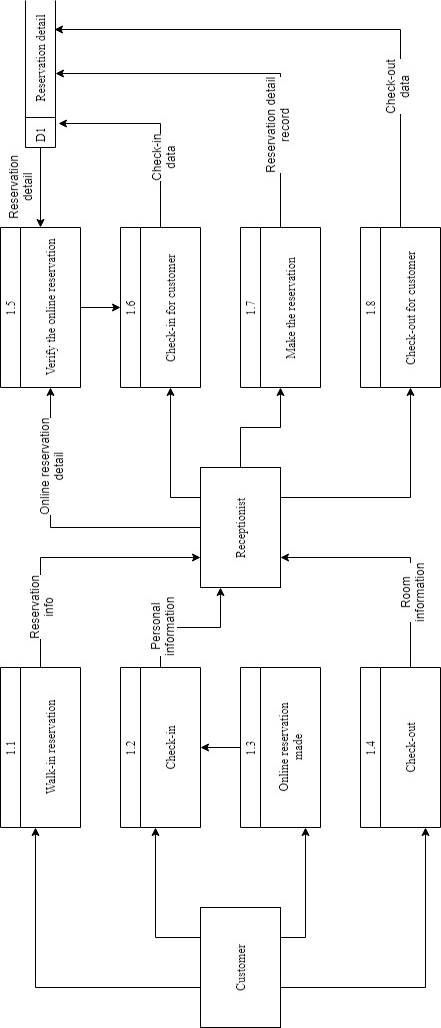
***4.1.1*** ***Context level***

***4.1.2 DFD Level 0***

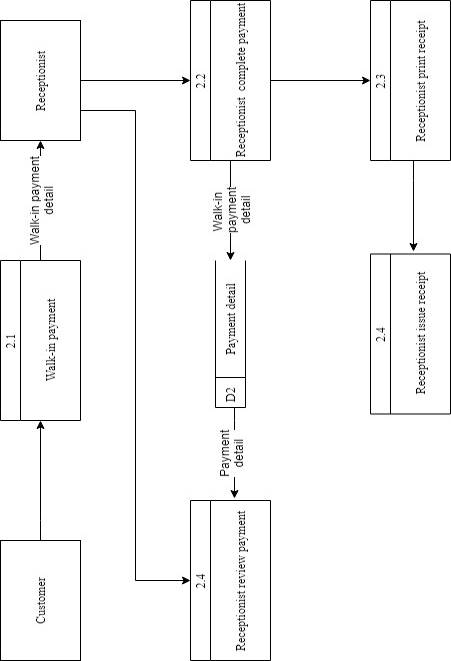


### ***4.1.3 DFD Level 1***

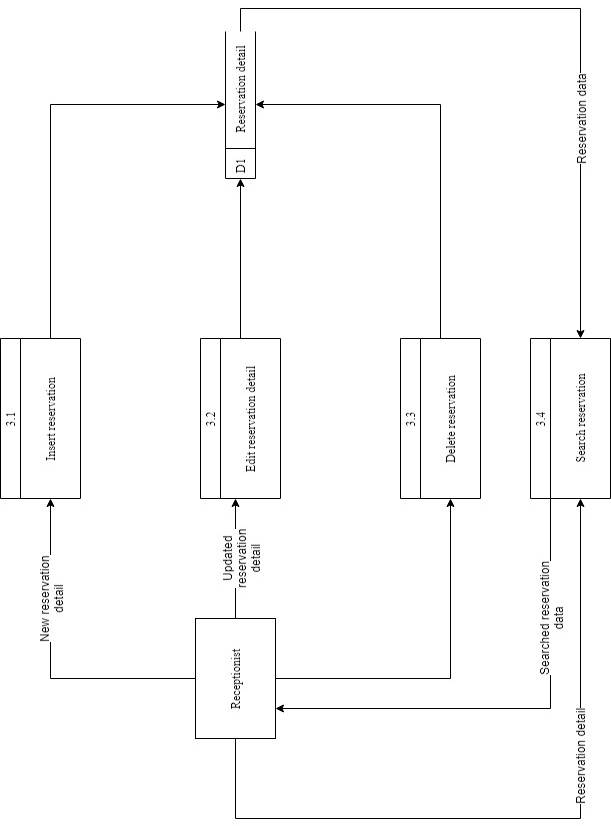
#### **4.1.3.1 Level 1Front-Desk System**



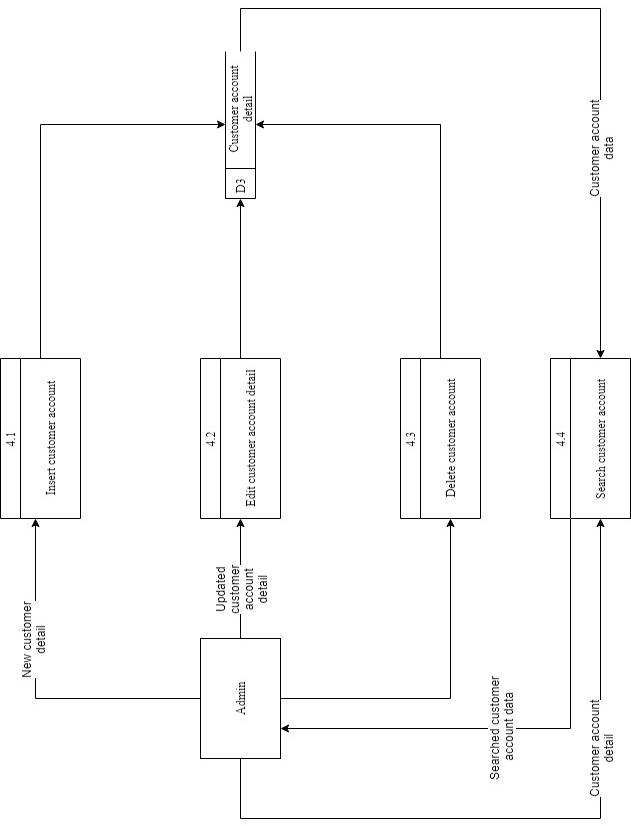
#### **4.1.3.2 Level 1Payment System**



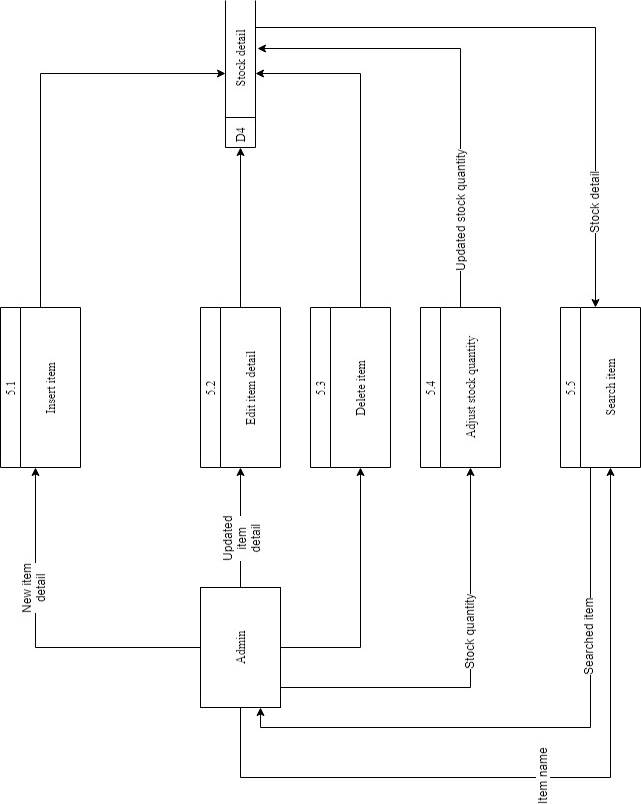
#### **4.1.3.3 Level 1 Reservation Detail System**



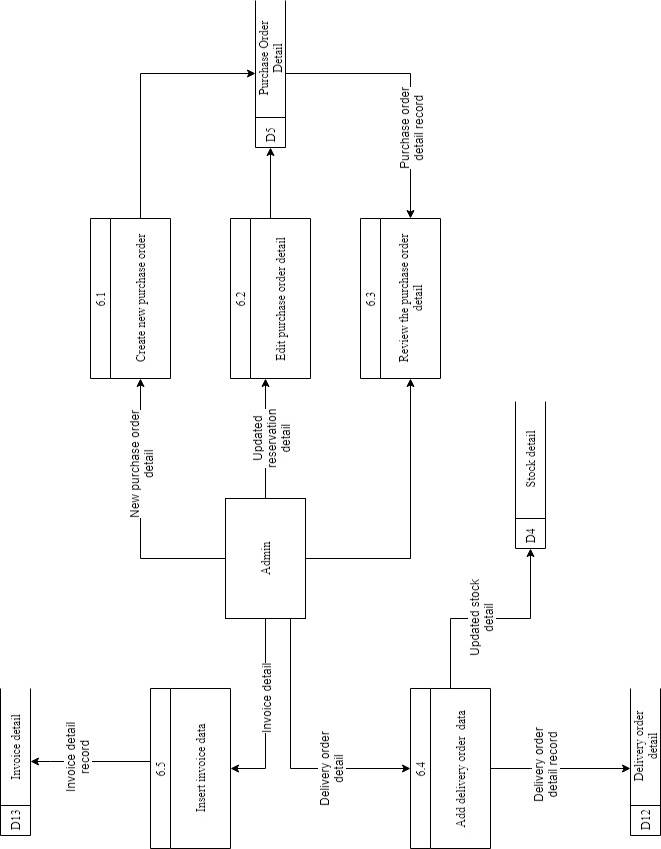
#### **4.1.3.4 Level 1 Customer Account System**



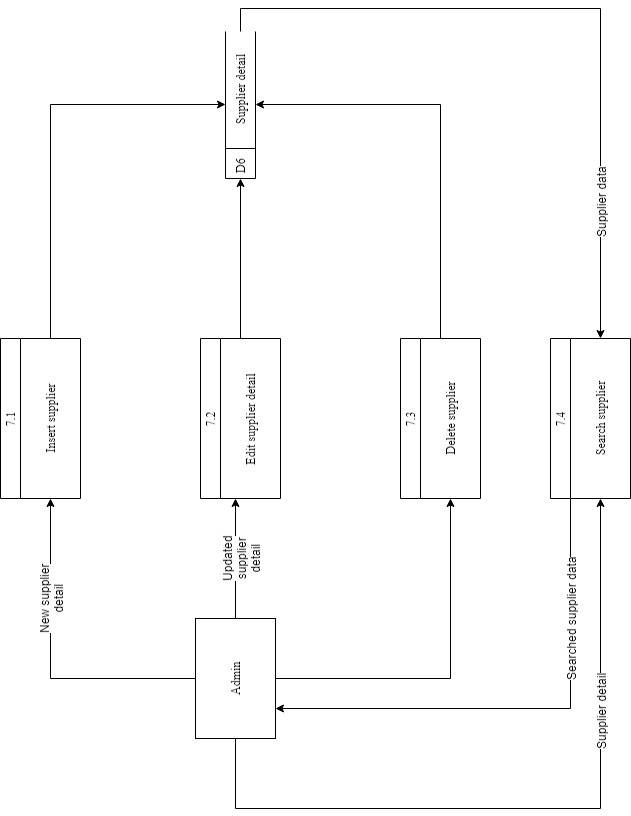
#### **4.1.3.5 Level 1 Inventory System**



#### **4.1.3.6 Level 1 Purchase Order System**



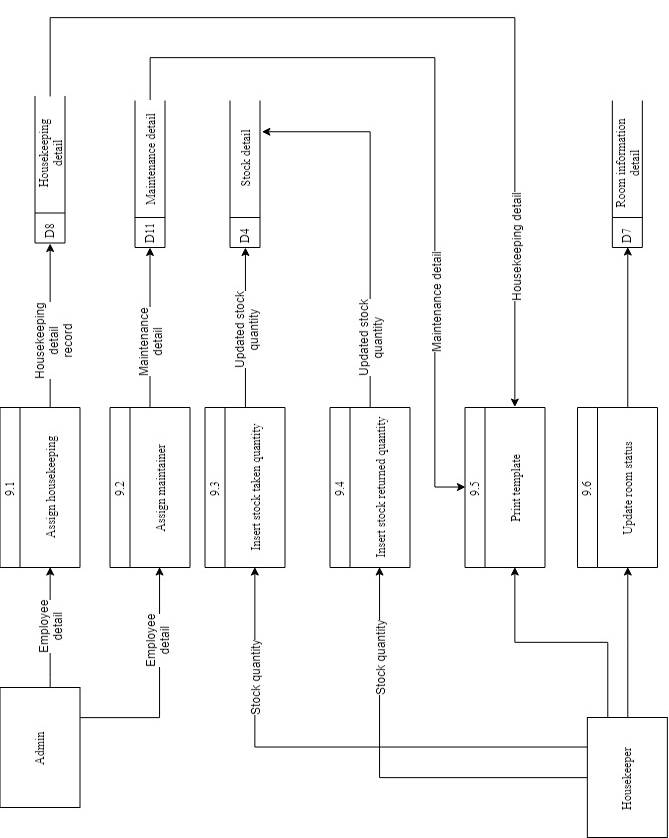
#### **4.1.3.7 Level 1 Supplier System**



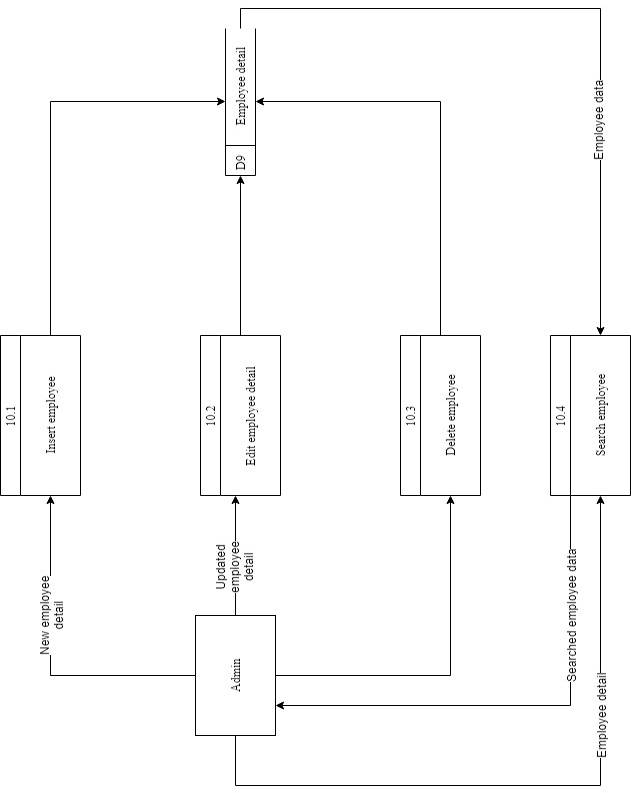
#### **4.1.3.8 Level 1 Room Information System**



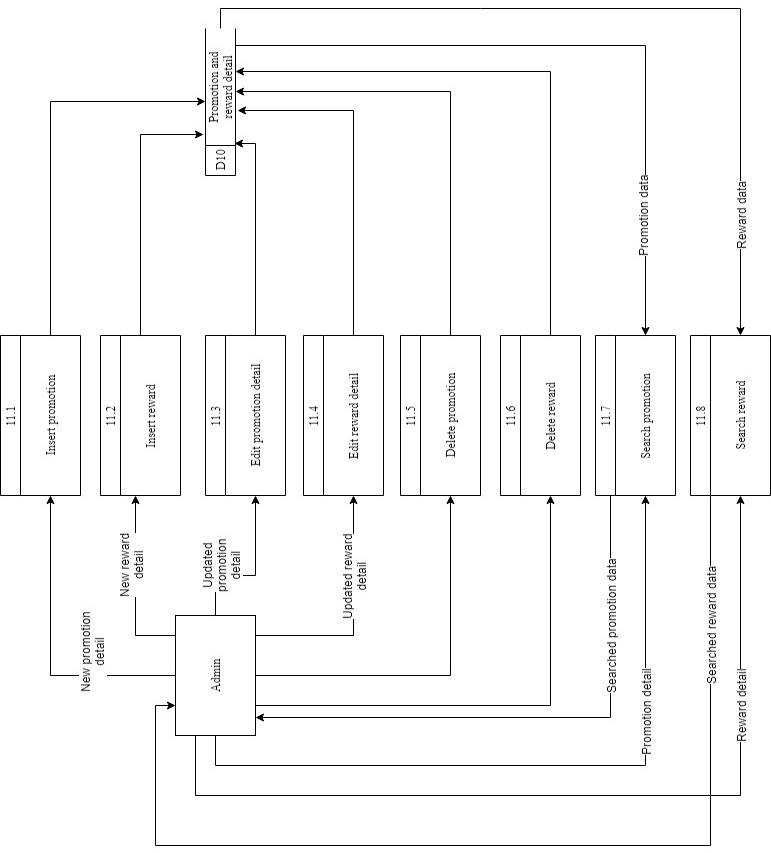
#### **4.1.3.9 Level 1 Housekeeping System**



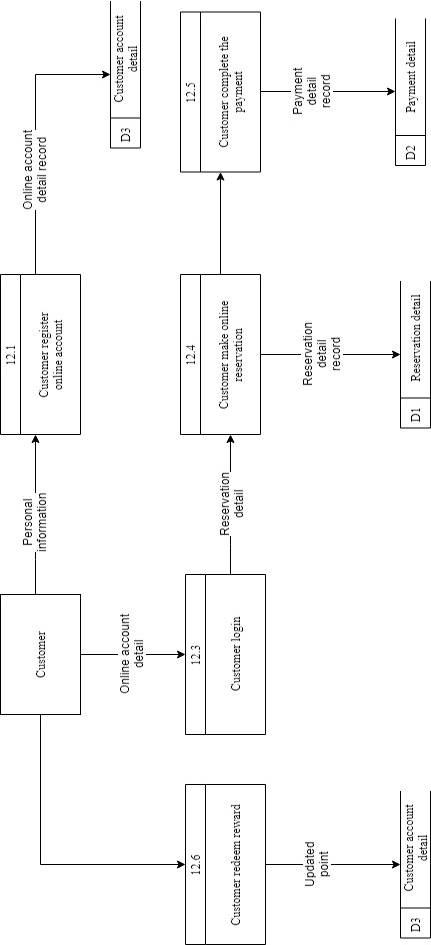
#### **4.1.3.10 Level 1 Employee System**



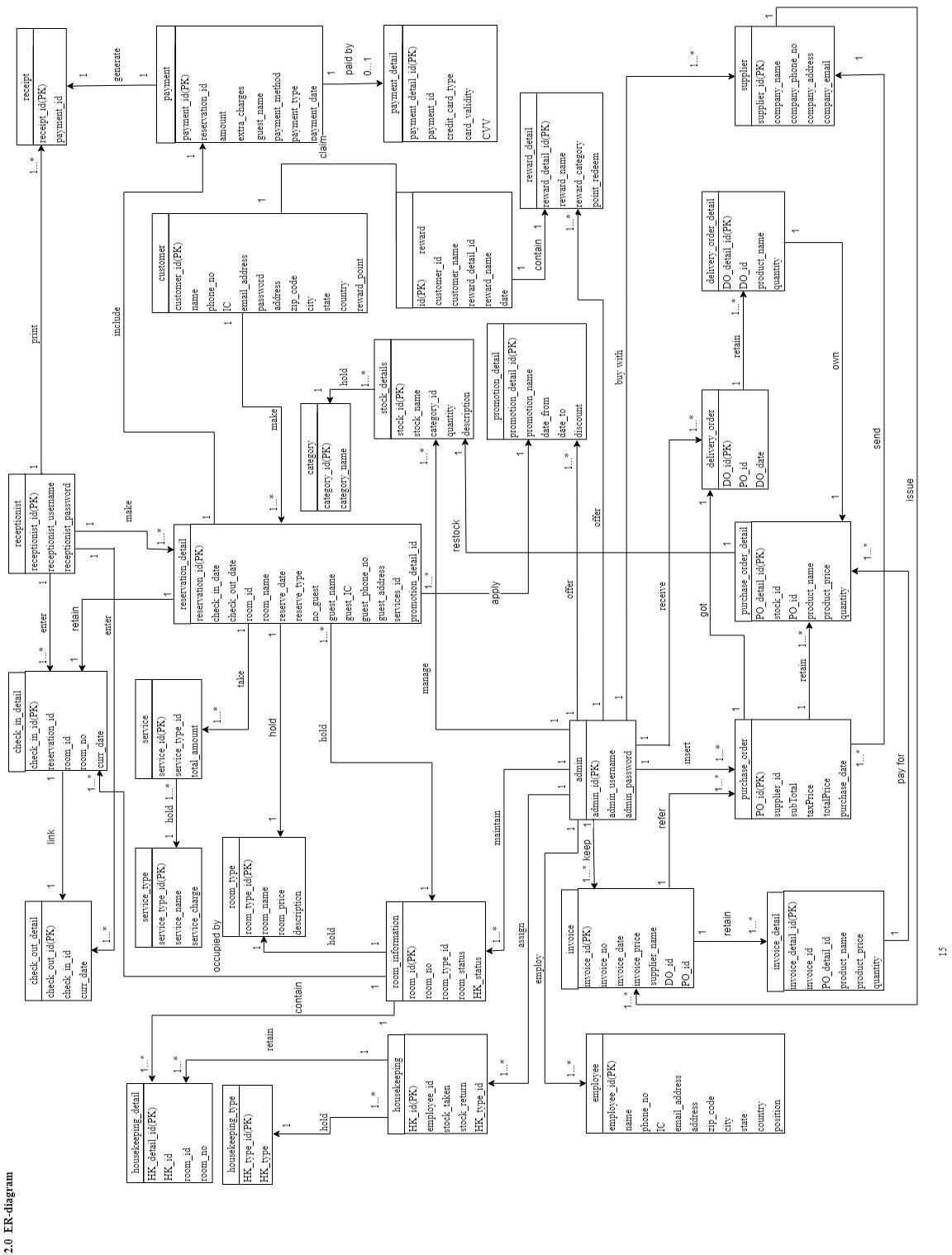
#### **4.1.3.11 Level 1 Promotion Reward System**



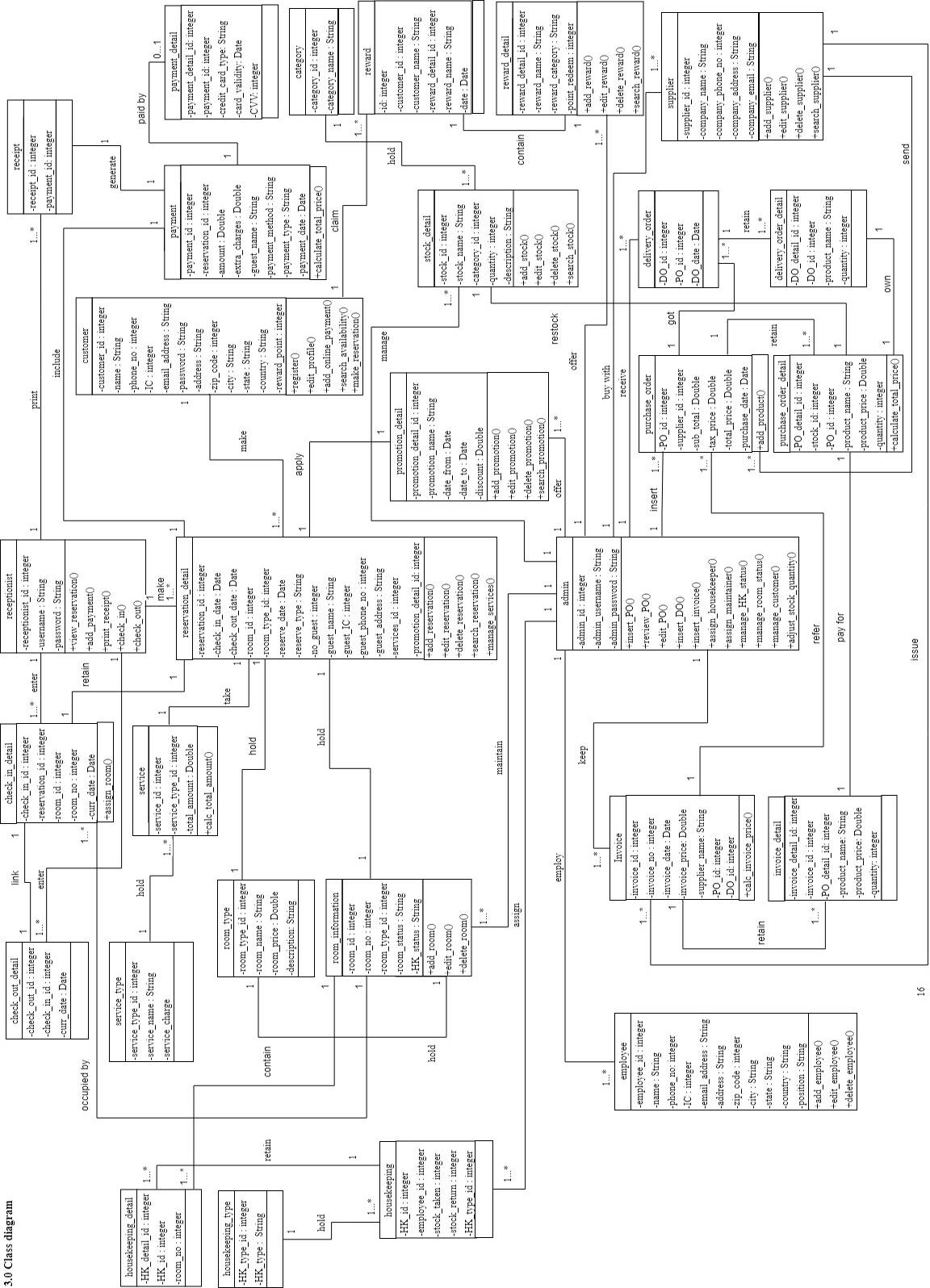
#### **4.1.3.12 Level 1 Online Booking System**



## **4.2 ER-diagram**



## **4.3 Class diagram**

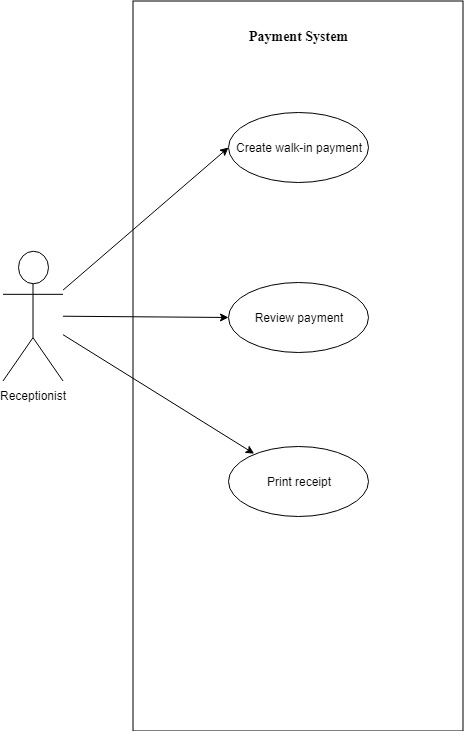


## **4.4 Use case diagram**

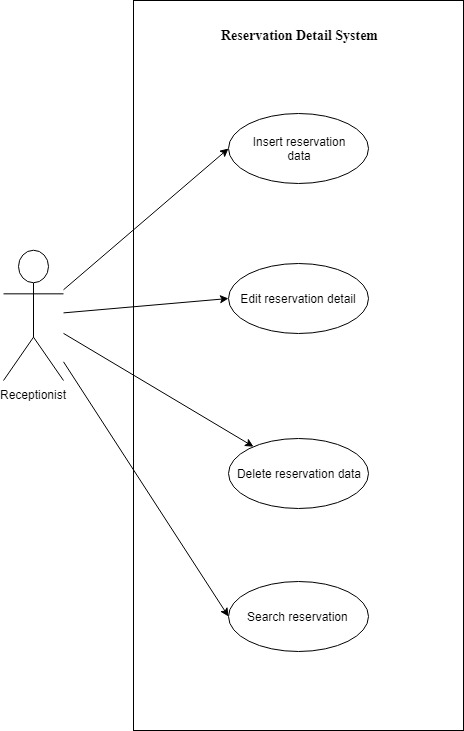
### ***4.4.1 Front-desk system***

****

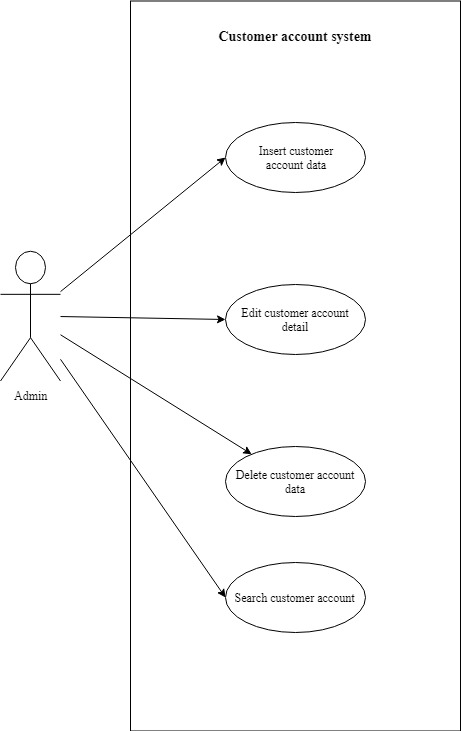
### ***4.4.2 Payment system***

****

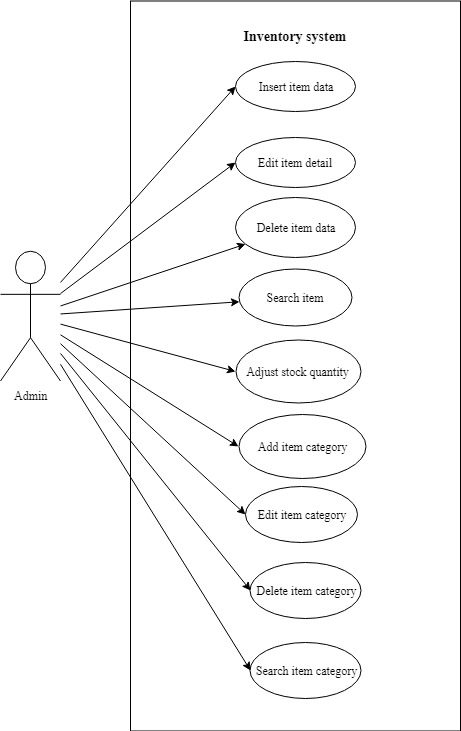
### ***4.4.3 Reservation detail system***

****

### ***4.4.4 Customer account system***



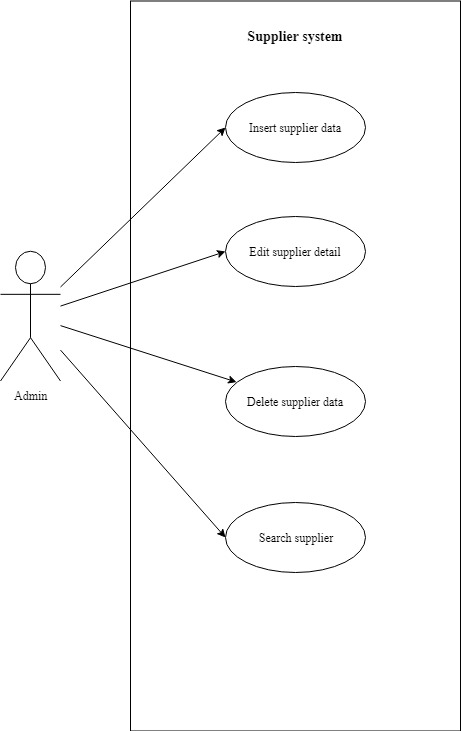
### ***4.4.5 Inventory system***

****

### ***4.4.6 Purchase order system***

****

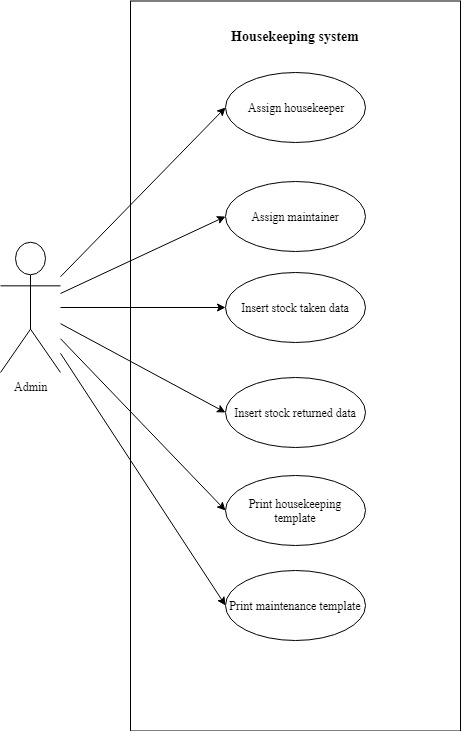
### ***4.4.7 Supplier system***

****

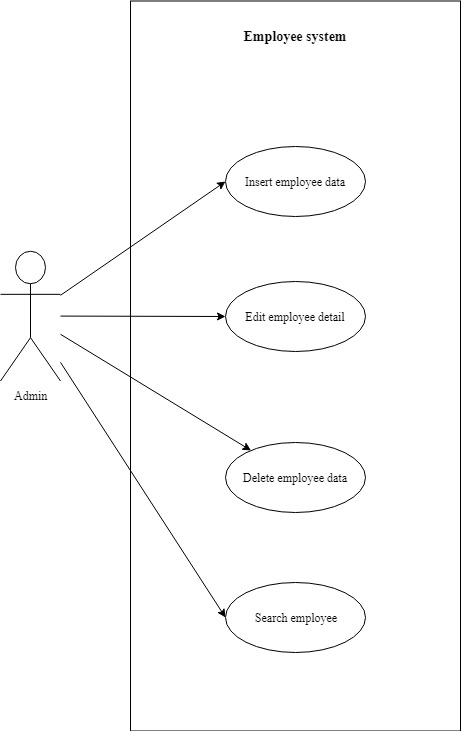
### ***4.4.8 Room information system***

****

### ***4.4.9 Housekeeping system***

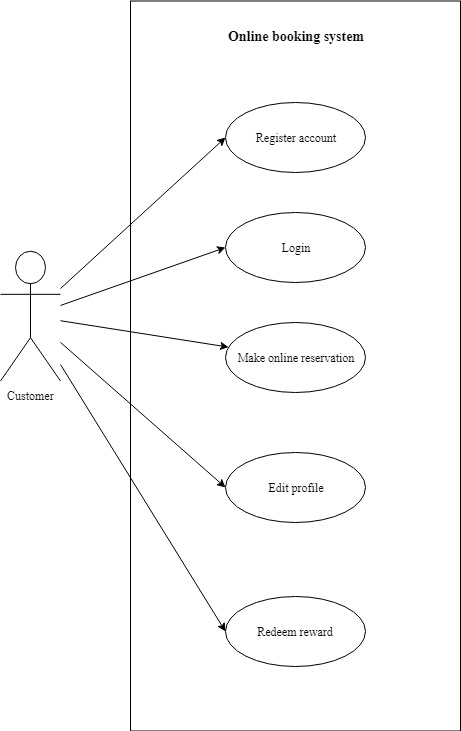
****

### ***4.4.10 Employee system***

****

### ***4.4.11 Promotion and reward system***

****

***4.4.12 Online booking system***

## **4.5 Use case description**

### ***4.5.1 Front-desk system***

|  |  |  |
| --- | --- | --- |
| Use case: Insert reservation data (Front-desk system) | | |
| Overview: This case describes how the receptionist inserts reservation data. | | |
| Pre – conditions:   1. The receptionist must login | | |
| Post – conditions:   1. Successful condition  * Receptionist has inserted reservation data. | | |
| Event | | Response |
| 1 | Receptionist login to front-desk system. | * Perform receptionist account validation. |
| 2 | Receptionist selects “Reservation” | * Enter to the reservation list webpage. |
| 3 | Select “New reservation” | * Enter to the reservation form webpage. |
| 4 | Enter all the required information | * Perform information validation. |
| 5 | Receptionist has made the reservation successfully | * Reservation data is inserted successfully. |

|  |  |  |
| --- | --- | --- |
| Use case: Edit reservation detail (Front-desk system) | | |
| Overview: This case describes how the receptionist edits the reservation detail. | | |
| Pre – conditions:   1. The receptionist must login | | |
| Post – conditions:   1. Successful condition  * Receptionist has updated the reservation detail successfully. | | |
| Event | | Response |
| 1 | Receptionist login to front-desk system. | * Perform receptionist account validation. |
| 2 | Receptionist selects “Reservation” | * Enter to the reservation list webpage. |
| 3 | Receptionist searches reservation | * Show the searched results. |
| 4 | Select “Edit” button | * Enter to the reservation form webpage. |
| 5 | Enter the updated information | * Perform information validation. |
| 6 | Receptionist has updated the reservation successfully | * Reservation detail is updated successfully. |

|  |  |  |
| --- | --- | --- |
| Use case: Delete reservation data (Front-desk system) | | |
| Overview: This case describes how the receptionist deletes reservation data. | | |
| Pre – conditions:   1. The receptionist must login | | |
| Post – conditions:   1. Successful condition  * Receptionist has deleted reservation data. | | |
| Event | | Response |
| 1 | Receptionist login to front-desk system. | * Perform receptionist account validation. |
| 2 | Receptionist selects “Reservation” | * Enter to the reservation list webpage. |
| 3 | Receptionist searches reservation | * Show the searched results. |
| 4 | Select “Delete” button | * Prompt out the confirm message to delete the reservation data. |
| 5 | Receptionist has deleted the reservation successfully | * Reservation data is deleted successfully. |

|  |  |  |
| --- | --- | --- |
| Use case: View reservation detail (Front-desk system) | | |
| Overview: This case describes how the receptionist views the reservation detail. | | |
| Pre – conditions:   1. The receptionist must login | | |
| Post – conditions:   1. Successful condition  * Receptionist is able to view the reservation detail successfully. | | |
| Event | | Response |
| 1 | Receptionist login to front-desk system. | * Perform receptionist account validation. |
| 2 | Receptionist selects “Reservation” | * Enter to the reservation list webpage. |
| 3 | Receptionist searches reservation | * Show the searched results. |
| 4 | Select “View” button | * Enter to the webpage that display the detail of the selected reservation. |
| 5 | Receptionist has updated the reservation successfully | * Reservation detail is displayed successfully. |

|  |  |  |
| --- | --- | --- |
| Use case: Complete check in process (Front-desk system) | | |
| Overview: This case describes how the receptionist completes the checks in process. | | |
| Pre – conditions:   1. The receptionist must login | | |
| Post – conditions:   1. Successful condition  * Receptionist has checked in for guest successfully. | | |
| Event | | Response |
| 1 | Receptionist login to front-desk system. | * Perform receptionist account validation. |
| 2 | Receptionist selects “Check-in” | * Enter to the list of room webpage. * If the customer is walk-in, then proceed to step 3, else proceed to step 4 for the customer that made the reservation. |
| 3 | Receptionist selects “New customer” | * Enter to the check-in form webpage. * Perform information validation. |
| 4 | Receptionist searches reservation | * Show the searched room result. |
| 4 | Select “Check-in” button | * Prompt out the confirm message to check in. |
| 5 | Receptionist assign room for the guest | * + Display the check-in form.   + Perform room availability validation. |
| 6 | Receptionist has completed the check in process successfully. | * Receptionist checked in for guest successfully. |

|  |  |  |
| --- | --- | --- |
| Use case: Complete check out process (Front-desk system) | | |
| Overview: This case describes how the receptionist completes the check out process. | | |
| Pre – conditions:   1. The receptionist must login | | |
| Post – conditions:   1. Successful condition  * Receptionist has checked out for guest successfully. | | |
| Event | | Response |
| 1 | Receptionist login to front-desk system. | * Perform receptionist account validation. |
| 2 | Receptionist selects “Check-out” | * Enter to the list of room webpage. |
| 3 | Receptionist searches room number | * Show the searched room result. |
| 4 | Select “Check-out” button | * Prompt out the confirm message to checkout. |
| 5 | Receptionist enters the unpaid amount value and click “Check-out” | * Perform information validation. |
| 6 | Receptionist has completed the check out process successfully. | * Receptionist checked out for guest successfully. |

### ***4.5.2 Payment system***

|  |  |  |
| --- | --- | --- |
| Use case: Create walk-in payment (Payment system) | | |
| Overview: This case describes how the receptionist creates the walk-in payment. | | |
| Pre – conditions:   1. The receptionist must login | | |
| Post – conditions:   1. Successful condition  * Receptionist has created the payment successfully. | | |
| Event | | Response |
| 1 | Receptionist login to front-desk system. | * Perform receptionist account validation. |
| 2 | Receptionist selects “Payment” | * Enter to the list of payment webpage. |
| 3 | Select “Create payment” button | * Enter to the payment form webpage. |
| 4 | Enter all the required information | * Perform information validation. |
| 5 | Receptionist has created the payment successfully | * Payment is created successfully. |

|  |  |  |
| --- | --- | --- |
| Use case: Review payment (Payment system) | | |
| Overview: This case describes how the receptionist reviews the payment detail. | | |
| Pre – conditions:   1. The receptionist must login | | |
| Post – conditions:   1. Successful condition  * Receptionist is able to view the payment detail successfully. | | |
| Event | | Response |
| 1 | Receptionist login to front-desk system. | * Perform receptionist account validation. |
| 2 | Receptionist selects “Payment” | * Enter to the list of payment webpage. |
| 3 | Receptionist searches payment | * Show the searched payment result. |
| 4 | Select “View” button | * Show the detail of the payment result. |
| 5 | Receptionist has viewed the payment detail successfully | * Payment detail is displayed successfully. |

|  |  |  |
| --- | --- | --- |
| Use case: Print receipt (Payment system) | | |
| Overview: This case describes how the receptionist prints the receipt. | | |
| Pre – conditions:   1. The receptionist must login | | |
| Post – conditions:   1. Successful condition  * Receptionist prints the receipt successfully. | | |
| Event | | Response |
| 1 | Receptionist login to front-desk system. | * Perform receptionist account validation. |
| 2 | Receptionist selects “Payment” | * Enter to the list of payment webpage. |
| 3 | Receptionist searches payment | * Show the searched payment result. |
| 4 | Select “Print” button | * Prompt out the print setup page. |
| 5 | Receptionist printed the receipt successfully | * Receipt is printed successfully. |

### ***4.5.3 Reservation detail system***

|  |  |  |
| --- | --- | --- |
| Use case: Insert reservation data (Reservation detail system) | | |
| Overview: This case describes how the receptionist inserts reservation data. | | |
| Pre – conditions:   1. The receptionist must login | | |
| Post – conditions:   1. Successful condition  * Receptionist has inserted reservation data. | | |
| Event | | Response |
| 1 | Receptionist login to front-desk system. | * Perform receptionist account validation. |
| 2 | Receptionist selects “Reservation” | * Enter to the reservation list webpage. |
| 3 | Select “New reservation” | * Enter to the reservation form webpage. |
| 4 | Enter all the required information | * Perform information validation. |
| 5 | Receptionist make the reservation successfully | * Reservation data is inserted successfully. |

|  |  |  |
| --- | --- | --- |
| Use case: Edit reservation detail (Reservation detail system) | | |
| Overview: This case describes how the receptionist edits the reservation detail. | | |
| Pre – conditions:   1. The receptionist must login | | |
| Post – conditions:   1. Successful condition  * Receptionist has updated the reservation detail successfully. | | |
| Event | | Response |
| 1 | Receptionist login to front-desk system. | * Perform receptionist account validation. |
| 2 | Receptionist selects “Reservation” | * Enter to the reservation list webpage. |
| 3 | Receptionist searches reservation | * Show the searched reservation results. |
| 4 | Select “Edit” button | * Enter to the reservation form webpage. |
| 5 | Enter the updated information | * Perform information validation. |
| 6 | Receptionist has updated the reservation successfully | * Reservation detail is updated successfully. |

|  |  |  |
| --- | --- | --- |
| Use case: Delete reservation data (Reservation detail system) | | |
| Overview: This case describes how the receptionist deletes reservation data. | | |
| Pre – conditions:   1. The receptionist must login | | |
| Post – conditions:   1. Successful condition  * Receptionist has deleted reservation data. | | |
| Event | | Response |
| 1 | Receptionist login to front-desk system. | * Perform receptionist account validation. |
| 2 | Receptionist selects “Reservation” | * Enter to the reservation list webpage. |
| 3 | Receptionist searches reservation | * Show the searched reservation results. |
| 4 | Select “Delete” button | * Prompt out the confirm message to delete the reservation data. |
| 5 | Receptionist has deleted the reservation successfully | * Reservation data is deleted successfully. |

|  |  |  |
| --- | --- | --- |
| Use case: Search reservation (Reservation detail system) | | |
| Overview: This case describes how the receptionist searches the reservation. | | |
| Pre – conditions:   1. The receptionist must login | | |
| Post – conditions:   1. Successful condition  * Receptionist has searched the reservation successfully. | | |
| Event | | Response |
| 1 | Receptionist login to front-desk system. | * Perform receptionist account validation. |
| 2 | Receptionist selects “Reservation” | * Enter to the reservation list webpage. |
| 3 | Receptionist searches reservation | * Show the searched reservation results. |
| 4 | Select “View” button | * Enter to the webpage that display the detail of the selected reservation. |
| 5 | Receptionist searched the reservation successfully | * Reservation detail is displayed successfully. |

### ***4.5.4 Customer account system***

|  |  |  |
| --- | --- | --- |
| Use case: Insert customer account data (Customer account system) | | |
| Overview: This case describes how the admin inserts customer account data. | | |
| Pre – conditions:   1. The admin must login | | |
| Post – conditions:   1. Successful condition  * Admin has inserted customer account data. | | |
| Event | | Response |
| 1 | Admin login to the system. | * Perform admin account validation. |
| 2 | Admin selects “Customer” | * Enter to the customer list webpage. |
| 3 | Select “New customer” | * Enter to the customer account form webpage. |
| 4 | Enter all the required information | * Perform information validation. |
| 5 | Admin has made the customer account successfully | * Customer account data is inserted successfully. |

|  |  |  |
| --- | --- | --- |
| Use case: Edit customer account detail (Customer account system) | | |
| Overview: This case describes how the admin edits the customer account detail. | | |
| Pre – conditions:   1. The admin must login | | |
| Post – conditions:   1. Successful condition  * Admin has updated the reservation detail successfully. | | |
| Event | | Response |
| 1 | Admin login to the system. | * Perform admin account validation. |
| 2 | Admin selects “Customer” | * Enter to the customer list webpage. |
| 3 | Admin searches customer account | * Show the searched customer result. |
| 4 | Select “Edit” button | * Enter to the customer account form webpage. |
| 5 | Enter the updated information | * Perform information validation. |
| 6 | Admin has updated the customer account successfully | * Customer account detail is updated successfully. |

|  |  |  |
| --- | --- | --- |
| Use case: Delete customer account data (Customer account system) | | |
| Overview: This case describes how the admin deletes the customer account data. | | |
| Pre – conditions:   1. The admin must login | | |
| Post – conditions:   1. Successful condition  * Admin has deleted customer account data. | | |
| Event | | Response |
| 1 | Admin login to the system. | * Perform admin account validation. |
| 2 | Admin selects “Customer” | * Enter to the customer list webpage. |
| 3 | Admin searches customer account | * Show the searched customer result. |
| 4 | Select “Delete” button | * Prompt out the confirm message to delete the customer account data. |
| 5 | Admin has deleted the customer account successfully | * Customer account data is deleted successfully. |

|  |  |  |
| --- | --- | --- |
| Use case: Search customer account (Customer account system) | | |
| Overview: This case describes how the admin searches the customer account. | | |
| Pre – conditions:   1. The admin must login | | |
| Post – conditions:   1. Successful condition  * Admin has searched the customer account successfully. | | |
| Event | | Response |
| 1 | Admin login to the system. | * Perform admin account validation. |
| 2 | Admin selects “Customer” | * Enter to the customer list webpage. |
| 3 | Admin searches customer account | * Show the searched customer result. |
| 4 | Select “View” button | * Enter to the webpage that display the detail of the selected customer. |
| 5 | Admin has searched the customer account successfully | * Customer detail is displayed successfully. |

### ***4.5.5 Inventory system***

|  |  |  |
| --- | --- | --- |
| Use case: Insert item data (Inventory system) | | |
| Overview: This case describes how the admin adds new item data | | |
| Pre – conditions:   1. The admin must login. | | |
| Post – conditions:   1. Successful condition  * Admin has added new item data successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Inventory” | * Enter to the inventory system webpage. |
| 3 | Admin selects ”View Stock” | * Enter to the stock list webpage. |
| 4 | Admin selects “Add new item” | * Enter to add new product webpage. |
| 5 | Admin enters all the item information | * Perform information validation. |
| 6 | Admin adds the item data successfully | * New item is added successfully. * Return to inventory system. |

|  |  |  |
| --- | --- | --- |
| Use case: Edit item data(Inventory system) | | |
| Overview: This case describes how the admin edits item data | | |
| Pre – conditions:   1. The admin must login. | | |
| Post – conditions:   1. Successful condition  * Admin has edited the item data successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Inventory” | * Enter to the inventory system webpage. |
| 3 | Admin selects ”View Stock” | * Enter to the stock list webpage. |
| 4 | Search by item name | * Show the searched item result. |
| 5 | Admin selects “Edit” | * Enter to the stock form webpage. |
| 6 | Admin edits the product details. | * Perform information validation. |
| 7 | Admin edits the product successfully | * The record is edited successfully. * Return to view item list. |

|  |  |  |
| --- | --- | --- |
| Use case: Delete item data (Inventory system) | | |
| Overview: This case describes how the admin deletes product data | | |
| Pre – conditions:   1. The admin must login. | | |
| Post – conditions:   1. Successful condition  * Admin has deleted the item data successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Inventory” | * Enter to the inventory system webpage. |
| 3 | Admin selects ”View Stock” | * Enter to the stock list webpage. |
| 4 | Search by item name | * Show the searched item result. |
| 5 | Select “Delete” | * Prompt out the confirm message to delete the item data. |
| 6 | Admin deletes the item successfully | * The record is deleted successfully. * Return to view item list. |

|  |  |  |
| --- | --- | --- |
| Use case: Search item (Inventory system) | | |
| Overview: This case describes how the admin searches item | | |
| Pre – conditions:   1. The admin must login. | | |
| Post – conditions:   1. Successful condition  * Admin had searched the item successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Inventory” | * Enter to the inventory system webpage. |
| 3 | Admin selects ”View Stock” | * Enter to the stock list webpage. |
| 4 | Admin searches the item name | * Show the searched item result. |
| 5 | Admin searched the item successfully | * Display different available alternative. |

|  |  |  |
| --- | --- | --- |
| Use case: Add item category (Inventory system) | | |
| Overview: This case describes how the admin adds item category | | |
| Pre – conditions:   1. The admin must login. | | |
| Post – conditions:   1. Successful condition  * Admin has added the item category successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Inventory” | * Enter to the inventory system webpage. |
| 3 | Admin selects ”View Category” | * Enter to the category list webpage. |
| 3 | Select “Add category” | * Enter to add category webpage. |
| 4 | Admin enters item category name | * Perform information validation. |
| 5 | Admin added the item category successfully | * New item category is added successfully. * Return to view item category list |

|  |  |  |
| --- | --- | --- |
| Use case: Edit item category (Inventory system) | | |
| Overview: This case describes how the admin edits item category | | |
| Pre – conditions:   1. The admin must login. | | |
| Post – conditions:   1. Successful condition  * Admin has edited the item category successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Inventory” | * Enter to the inventory system webpage. |
| 3 | Admin selects ”View Category” | * Enter to the item category list webpage. |
| 4 | Admin searches item category name | * Show the searched item result. |
| 5 | Admin selects “Edit” | * Enter to the item category form webpage. |
| 6 | Admin edits the item category name | * Perform duplication validation of item category name. |
| 7 | Admin edited the item category successfully | * The record is edited successfully. * Return to view item category list. |

|  |  |  |
| --- | --- | --- |
| Use case: Delete item category (Inventory system) | | |
| Overview: This case describes how the admin deletes item category | | |
| Pre – conditions:   1. The admin must login. | | |
| Post – conditions:   1. Successful condition  * Admin has deleted the product category successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Inventory” | * Enter to the inventory system webpage. |
| 3 | Admin selects ”View Category” | * Enter to the item category list webpage. |
| 4 | Admin searches item category name | * Show the searched item result. |
| 5 | Admin selects “Delete” | * Prompt out the confirm message to delete the item category data. |
| 6 | Admin delete the item category successfully | * The record is deleted successfully. * Return to view item category list. |

|  |  |  |
| --- | --- | --- |
| Use case: Search item category (Inventory system) | | |
| Overview: This case describes how the admin searches item category | | |
| Pre – conditions:   1. The admin must login. | | |
| Post – conditions:   1. Successful condition  * Admin had searched the item category successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Inventory” | * Enter to the inventory system webpage. |
| 3 | Admin selects ”View Category” | * Enter to the item category list webpage. |
| 4 | Admin searches the item category name | * Show the searched item category result. |
| 5 | Admin searched the item category successfully | * Display different available alternative. |

### ***4.5.6 Purchase order system***

|  |  |  |
| --- | --- | --- |
| Use case: Create purchase order (Purchase order system) | | |
| Overview: This case describes how the admin creates purchase order | | |
| Pre – conditions:   1. Successful condition  * The admin must login. | | |
| Post – conditions:   1. Successful condition  * Admin has created the purchase order successfully | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Purchase Order” | * Enter purchase order system webpage. |
| 3 | Admin selects “Create Purchase Order” | * Enter to view the supplier list webpage. |
| 4 | Admin selects “Create PO” | * Enter purchase order form webpage. |
| 5 | Admin enters all the order information | * Perform PO information validation. |
| 6 | Admin selects “Save” | * Prompt out “New Purchase order is created successfully”. |
| 7 | Admin has made the purchase order successfully | * New purchase order is made successfully. |

|  |  |  |
| --- | --- | --- |
| Use case: Edit purchase order detail (Purchase order system) | | |
| Overview: This case describes how the admin edits purchase order detail. | | |
| Pre – conditions:   1. Successful condition  * The admin must login. * Admin must create a purchase order | | |
| Post – conditions:   1. Successful condition  * Admin has edited the purchase order detail successfully | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Purchase order” | * Enter purchase order system webpage. |
| 3 | Admin selects “View/Edit Purchase order” | * Enter view purchase order webpage. |
| 4 | Admin selects “Edit PO” | * Enter purchase order form webpage. |
| 5 | Admin edits the purchase order data. | * Perform PO information validation. |
| 6 | Admin has edited the purchase order successfully | * The purchase order is edited successfully. * Return to view purchase order webpage. |

|  |  |  |
| --- | --- | --- |
| Use case: Review purchase order (Purchase order system) | | |
| Overview: This case describes how the admin reviews purchase order. | | |
| Pre – conditions:   1. Successful condition  * The admin must login. * Admin must create a purchase order | | |
| Post – conditions:   1. Successful condition  * Admin is able to review the purchase order successfully | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Purchase order” | * Enter purchase order system webpage. |
| 3 | Admin selects “View/Edit Purchase order” | * Enter view purchase order webpage. |
| 4 | Admin selects “Review PO” | * The detail of purchase order is displayed. |
| 5 | Admin reviews the purchase order detail successfully | * The purchase order is reviewed successfully. |

|  |  |  |
| --- | --- | --- |
| Use case: Search purchase order (Purchase order system) | | |
| Overview: This case describes how the admin searches purchase order. | | |
| Pre – conditions:   1. Successful condition  * The admin must login. * Admin must create a purchase order | | |
| Post – conditions:   1. Successful condition  * Admin is able to search the purchase order successfully | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Purchase order” | * Enter purchase order system webpage. |
| 3 | Admin selects “View/Edit Purchase order” | * Enter view purchase order webpage. |
| 4 | Admin searches the purchase order | * Show the searched purchase order result. |
| 5 | Admin searches the purchase order successfully | * The purchase order is searched successfully. |

|  |  |  |
| --- | --- | --- |
| Use case: Insert delivery order data (Purchase order system) | | |
| Overview: This case describes how the admin inserts delivery order data | | |
| Pre – conditions:   1. Successful condition  * The admin must login. * Admin must create a purchase order | | |
| Post – conditions:   1. Successful condition  * Admin has inserted the delivery order data successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Purchase order” | * Enter purchase order system webpage. |
| 3 | Admin selects “Delivery Order” | * Enter delivery order webpage. |
| 4 | Admin selects “Insert DO” | * Enter delivery order form webpage. |
| 5 | Admin enters DO number and DO date | * Perform DO information validation. |
| 6 | Admin enters delivery order details | * Perform information DO validation and stock calculation. |
| 7 | Admin selects “Save” | * Items have added into Stock. * Return to view delivery order webpage. |

|  |  |  |
| --- | --- | --- |
| Use case: Edit delivery order detail (Purchase order system) | | |
| Overview: This case describes how the admin inserts delivery order data | | |
| Pre – conditions:   1. Successful condition  * The admin must login. * Admin must has a delivery order | | |
| Post – conditions:   1. Successful condition  * Admin has edited the delivery order detail successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Purchase order” | * Enter purchase order system webpage. |
| 3 | Admin selects “Delivery Order” | * Enter delivery order webpage. |
| 4 | Admin selects “Edit” | * Enter delivery order form webpage. |
| 5 | Admin enters the updated delivery order details | * Perform DO information validation and stock calculation. |
| 6 | Admin selects “Save” | * Delivery order detail is updated. * Return to view delivery order webpage. |

|  |  |  |
| --- | --- | --- |
| Use case: Search delivery order (Purchase order system) | | |
| Overview: This case describes how the admin searches delivery order. | | |
| Pre – conditions:   1. Successful condition  * The admin must login. * Admin must has a delivery order | | |
| Post – conditions:   1. Successful condition  * Admin is able to search the delivery order successfully | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Purchase order” | * Enter purchase order system webpage. |
| 3 | Admin selects “Delivery Order” | * Enter delivery order webpage. |
| 4 | Admin searches the delivery order | * Show the searched delivery order result. |
| 5 | Admin searches the delivery order successfully | * The delivery order is searched successfully. |

|  |  |  |
| --- | --- | --- |
| Use case: Insert invoice data (Purchase order system) | | |
| Overview: This case describes how the admin inserts invoice data | | |
| Pre – conditions:   1. Successful condition  * The admin must login. * Admin must have a delivery order | | |
| Post – conditions:   1. Successful condition  * Admin has inserted the invoice data successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Purchase order” | * Enter purchase order system webpage. |
| 3 | Admin selects “Invoice” | * Enter invoice webpage. |
| 4 | Admin selects “Insert invoice” | * Enter invoice form webpage. |
| 5 | Admin selects the DO | * Display the selected DO. * Perform total amount calculation. |
| 6 | Admin enters invoice number and invoice date | * Perform invoice information validation. |
| 7 | Admin selects “Save” | * Invoice data is inserted successfully. * Return to view invoice webpage. |

|  |  |  |
| --- | --- | --- |
| Use case: Edit invoice detail (Purchase order system) | | |
| Overview: This case describes how the admin edits the invoice detail | | |
| Pre – conditions:   1. Successful condition  * The admin must login. * Admin must has an invoice | | |
| Post – conditions:   1. Successful condition  * Admin has edited the invoice detail successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Purchase order” | * Enter purchase order system webpage. |
| 3 | Admin selects “Invoice” | * Enter invoice webpage. |
| 4 | Admin selects “Edit” | * Enter invoice form webpage. |
| 5 | Admin enters the updated invoice details | * Perform invoice information validation and total amount calculation. |
| 6 | Admin selects “Save” | * Invoice detail is updated. * Return to view invoice webpage. |

|  |  |  |
| --- | --- | --- |
| Use case: Search invoice (Purchase order system) | | |
| Overview: This case describes how the admin searches invoice. | | |
| Pre – conditions:   1. Successful condition  * The admin must login. * Admin must has an invoice | | |
| Post – conditions:   1. Successful condition  * Admin is able to search the invoice successfully | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Purchase order” | * Enter purchase order system webpage. |
| 3 | Admin selects “Invoice” | * Enter invoice webpage. |
| 4 | Admin searches the invoice | * Show the searched invoice result. |
| 5 | Admin searches the invoice successfully | * The invoice is searched successfully. |

### ***4.5.7 Supplier system***

|  |  |  |
| --- | --- | --- |
| Use case: Insert supplier data (Supplier system) | | |
| Overview: This case describes how the admin inserts new supplier | | |
| Pre – conditions:   1. The admin must login. | | |
| Post – conditions:   1. Successful condition  * Admin has inserted new supplier data successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Supplier” | * Enter to supplier system webpage. |
| 3 | Admin selects “New Supplier” | * Enter to add new supplier webpage. |
| 4 | Admin enters all the supplier information | * Perform information validation. |
| 5 | Admin inserts the supplier data successfully | * New supplier data is inserted successfully. * Return to view supplier list. |

|  |  |  |
| --- | --- | --- |
| Use case: Edit supplier detail(Supplier system) | | |
| Overview: This case describes how the staff edits supplier | | |
| Pre – conditions:   1. The admin must login. | | |
| Post – conditions:   1. Successful condition  * Admin has edited supplier successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Supplier” | * Enter to supplier system webpage. |
| 3 | Search the supplier and select “Edit” | * Enter to edit supplier webpage. |
| 4 | Admin edits the supplier data. | * Perform data validation. |
| 5 | Admin edits the supplier successfully | * The record is edited successfully. * Return to view supplier list. |

|  |  |  |
| --- | --- | --- |
| Use case: Delete supplier data (Supplier system) | | |
| Overview: This case describes how the admin deletes supplier | | |
| Pre – conditions:   1. The admin must login. | | |
| Post – conditions:   1. Successful condition  * Admin has deleted supplier successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Supplier” | * Enter to supplier system webpage. |
| 3 | Search the supplier and select “delete” | * Prompt out the confirm message to delete the supplier data. |
| 4 | Admin has deleted the supplier successfully | * The record is deleted successfully. * Return to view supplier list. |

|  |  |  |
| --- | --- | --- |
| Use case: Search supplier (Supplier system) | | |
| Overview: This case describes how the admin searches supplier | | |
| Pre – conditions:   1. The admin must login. | | |
| Post – conditions:   1. Successful condition  * Admin has searched the supplier successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Supplier” | * Enter to supplier system webpage. |
| 3 | Search the supplier name | * Show the searched supplier result. |
| 4 | Admin has searched the supplier successfully. | * Display different available alternative. |

### ***4.5.8 Room information system***

|  |  |  |
| --- | --- | --- |
| Use case: Insert room data (Room information system) | | |
| Overview: This case describes how the admin inserts new room data | | |
| Pre – conditions:   1. The admin must login. | | |
| Post – conditions:   1. Successful condition  * Admin has inserted new room data successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Room” | * Enter to room information system webpage. |
| 3 | Admin selects “New room” | * Enter to add new room webpage. |
| 4 | Admin enters all the room information | * Perform information validation. |
| 5 | Admin inserts the room data successfully | * New room data is inserted successfully. * Return to view room list. |

|  |  |  |
| --- | --- | --- |
| Use case: Edit room detail(Room information system) | | |
| Overview: This case describes how the admin edits room detail | | |
| Pre – conditions:   1. The admin must login. | | |
| Post – conditions:   1. Successful condition  * Admin has edited room details successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Room” | * Enter to room information system webpage. |
| 3 | Search the room and select “Edit” | * Enter to edit room webpage. |
| 4 | Admin edits the room details. | * Perform data validation. |
| 5 | Admin edits the room details successfully | * The record is edited successfully. * Return to view room list. |

|  |  |  |
| --- | --- | --- |
| Use case: Delete room data (Room information system) | | |
| Overview: This case describes how the admin deletes room data | | |
| Pre – conditions:   1. The admin must login. | | |
| Post – conditions:   1. Successful condition  * Admin has deleted room data successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Room” | * Enter to room information system webpage. |
| 3 | Search the room and select “delete” | * Display different available alternative. |
| 4 | Admin has deleted the supplier successfully | * The record is deleted successfully. * Return to view room list. |

|  |  |  |
| --- | --- | --- |
| Use case: Adjust room status (Room information system) | | |
| Overview: This case describes how the admin adjusts room status | | |
| Pre – conditions:   1. The admin must login. | | |
| Post – conditions:   1. Successful condition  * Admin has adjusted the room status successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Room” | * Enter to room information system webpage. |
| 3 | Search the room | * Show the searched room result. |
| 4 | Choose the status of the room from the option given | * The options in dropdown status are shown. |
| 5 | Admin selects “Save” | * Prompt out the successful of saving message. |
| 6 | Admin has adjusted the room status successfully | * The room status is adjusted successfully. * Return to view room list. |

|  |  |  |
| --- | --- | --- |
| Use case: Search room (Room information system) | | |
| Overview: This case describes how the admin searches room | | |
| Pre – conditions:   1. The admin must login. | | |
| Post – conditions:   1. Successful condition  * Admin has searched the room successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Room” | * Enter to room information system webpage. |
| 3 | Search the room number | * Show the searched room result. |
| 4 | Admin has searched the room successfully. | * Display different available alternative. |

### ***4.5.9 Housekeeping system***

|  |  |  |
| --- | --- | --- |
| Use case: Assign housekeeper (Housekeepingsystem) | | |
| Overview: This case describes how the admin assigns housekeeper | | |
| Pre – conditions:   1. The admin must login. | | |
| Post – conditions:   1. Successful condition  * Admin has assigned the housekeeper successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Housekeeping” | * Enter to housekeeping system webpage. |
| 3 | Admin selects “Create housekeeper” | * Enter to housekeeper form webpage. |
| 4 | Admin enters the housekeeper information | * Perform housekeeper information validation. |
| 5 | Admin clicks “Save” | * Housekeeper is assigned successfully. * Return to view room list. |

|  |  |  |
| --- | --- | --- |
| Use case: Assign maintainer (Housekeepingsystem) | | |
| Overview: This case describes how the admin assigns maintainer | | |
| Pre – conditions:   1. The admin must login. | | |
| Post – conditions:   1. Successful condition  * Admin has assigned the maintainer successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Housekeeping” | * Enter to housekeeping system webpage. |
| 3 | Admin selects “Create maintainer” | * Enter to maintainer form webpage. |
| 4 | Admin enters the maintainer information | * Perform maintainer information validation. |
| 5 | Admin clicks “Save” | * Maintainer is assigned successfully. * Return to view room list. |

|  |  |  |
| --- | --- | --- |
| Use case: Insert stock taken (Housekeepingsystem) | | |
| Overview: This case describes how the admin inserts stock taken data | | |
| Pre – conditions:   1. The admin must login. | | |
| Post – conditions:   1. Successful condition  * Admin has inserted the stock taken data successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Housekeeping” | * Enter to housekeeping system webpage. |
| 3 | Admin searches record and enter stock taken quantity | * Show the searched housekeeping record. |
| 4 | Admin clicks “Save” | * Record is updated successfully. * Return to view housekeeper list. |

|  |  |  |
| --- | --- | --- |
| Use case: Insert stock returned (Housekeepingsystem) | | |
| Overview: This case describes how the admin inserts stock returned data | | |
| Pre – conditions:   1. The admin must login. | | |
| Post – conditions:   1. Successful condition  * Admin has inserted the stock returned data successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Housekeeping” | * Enter to housekeeping system webpage. |
| 3 | Admin searches record and enter stock returned quantity | * Show the searched housekeeping record. |
| 4 | Admin clicks “Save” | * Record is updated successfully. * Return to view housekeeper list. |

|  |  |  |
| --- | --- | --- |
| Use case: Print housekeeping template (Housekeepingsystem) | | |
| Overview: This case describes how the admin prints housekeeping template | | |
| Pre – conditions:   1. The admin must login. | | |
| Post – conditions:   1. Successful condition  * Admin has printed the housekeeping template successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Housekeeping” | * Enter to housekeeping system webpage. |
| 3 | Admin searches housekeeper record and click “view” | * Show the searched housekeeper record. |
| 4 | Admin clicks “print” | * Housekeeping template is printed successfully. * Return to view housekeeping list. |

|  |  |  |
| --- | --- | --- |
| Use case: Print maintainer template (Housekeepingsystem) | | |
| Overview: This case describes how the admin prints maintainer template | | |
| Pre – conditions:   1. The admin must login. | | |
| Post – conditions:   1. Successful condition  * Admin has printed the housekeeping template successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Housekeeping” | * Enter to housekeeping system webpage. |
| 3 | Admin searches maintainer record and click “view” | * Show the searched maintainer record. |
| 4 | Admin clicks “print” | * Maintainer template is printed successfully. * Return to view housekeeping list. |

### ***4.5.10 Employee system***

|  |  |  |
| --- | --- | --- |
| Use case: Insert employee data (Employee system) | | |
| Overview: This case describes how the admin inserts new employee data | | |
| Pre – conditions:   1. The admin must login. | | |
| Post – conditions:   1. Successful condition  * Admin has inserted new employee data successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Employee” | * Enter to supplier system webpage. |
| 3 | Admin selects “New employee” | * Enter to add new employee webpage. |
| 4 | Admin enters all the employee information | * Perform information validation. |
| 5 | Admin inserts the employee data successfully | * New employee data is inserted successfully. * Return to view employee list. |

|  |  |  |
| --- | --- | --- |
| Use case: Edit employee detail(Employee system) | | |
| Overview: This case describes how the admin edits employee details | | |
| Pre – conditions:   1. The admin must login. | | |
| Post – conditions:   1. Successful condition  * Admin has edited employee detail successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Employee” | * Enter to employee system webpage. |
| 3 | Search the employee and select “Edit” | * Enter to edit employee webpage. |
| 4 | Admin edits the employee details. | * Perform data validation. |
| 5 | Admin edits the employee successfully | * The record is edited successfully. * Return to view employee list. |

|  |  |  |
| --- | --- | --- |
| Use case: Delete employee data (Employee system) | | |
| Overview: This case describes how the admin deletes employee data | | |
| Pre – conditions:   1. The admin must login. | | |
| Post – conditions:   1. Successful condition  * Admin has deleted employee data successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Employee” | * Enter to employee system webpage. |
| 3 | Search the employee and select “delete” | * Show the searched employee result. |
| 4 | Admin has deleted the employee data successfully | * The record is deleted successfully. * Return to view employee list. |

|  |  |  |
| --- | --- | --- |
| Use case: Search employee (Employee system) | | |
| Overview: This case describes how the admin searches employee | | |
| Pre – conditions:   1. The admin must login. | | |
| Post – conditions:   1. Successful condition  * Admin has searched the employee successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Employee” | * Enter to employee system webpage. |
| 3 | Search the employee name | * Show the searched employee result. |
| 4 | Admin has searched the employee successfully. | * The employee record is searched successfully. |

### ***4.5.11 Promotion reward system***

|  |  |  |
| --- | --- | --- |
| Use case: Insert promotion data (Promotion reward system) | | |
| Overview: This case describes how the admin inserts new promotion data | | |
| Pre – conditions:   1. The admin must login. | | |
| Post – conditions:   1. Successful condition  * Admin has inserted new promotion data successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Promotion/Reward” | * Enter to promotion reward system webpage. |
| 3 | Admin selects “Promotion” | * Enter to promotion webpage |
| 4 | Admin selects “New promotion” | * Enter to add new promotion webpage. |
| 5 | Admin enters all the promotion information | * Perform information validation. |
| 6 | Admin inserts the promotion data successfully | * New promotion data is inserted successfully. * Return to view promotion list. |

|  |  |  |
| --- | --- | --- |
| Use case: Insert reward data (Promotion reward system) | | |
| Overview: This case describes how the admin inserts new reward data | | |
| Pre – conditions:   1. The admin must login. | | |
| Post – conditions:   1. Successful condition  * Admin has inserted new reward data successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Promotion/Reward” | * Enter to promotion reward system webpage. |
| 3 | Admin selects “Reward” | * Enter to reward webpage |
| 4 | Admin selects “New reward” | * Enter to add new reward webpage. |
| 5 | Admin enters all the reward information | * Perform information validation. |
| 6 | Admin inserts the reward data successfully | * New reward data is inserted successfully. * Return to view promotion list. |

|  |  |  |
| --- | --- | --- |
| Use case: Edit promotion detail (Promotion reward system) | | |
| Overview: This case describes how the admin edits promotion details | | |
| Pre – conditions:   1. The admin must login. | | |
| Post – conditions:   1. Successful condition  * Admin has edited promotion detail successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Promotion/Reward” | * Enter to promotion reward system webpage. |
| 3 | Admin selects “Promotion” | * Enter to promotion webpage |
| 4 | Search the promotion and select “Edit” | * Enter to edit promotion webpage. |
| 5 | Admin edits the promotion details. | * Perform data validation. |
| 6 | Admin edits the promotion successfully | * The record is edited successfully. * Return to view promotion list. |

|  |  |  |
| --- | --- | --- |
| Use case: Edit reward detail (Promotion reward system) | | |
| Overview: This case describes how the admin edits reward details | | |
| Pre – conditions:   1. The admin must login. | | |
| Post – conditions:   1. Successful condition  * Admin has edited reward detail successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Promotion/Reward” | * Enter to promotion reward system webpage. |
| 3 | Admin selects “Reward” | * Enter to reward webpage |
| 4 | Search the reward and select “Edit” | * Enter to edit reward webpage. |
| 5 | Admin edits the reward details. | * Perform data validation. |
| 6 | Admin edits the reward successfully | * The record is edited successfully. * Return to view reward list. |

|  |  |  |
| --- | --- | --- |
| Use case: Delete promotion data (Promotion reward system) | | |
| Overview: This case describes how the admin deletes promotion data | | |
| Pre – conditions:   1. The admin must login. | | |
| Post – conditions:   1. Successful condition  * Admin has deleted promotion data successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Promotion/Reward” | * Enter to promotion reward system webpage. |
| 3 | Admin selects “Promotion” | * Enter to promotion webpage |
| 4 | Search the promotion and select “delete” | * Show the searched promotion result. * Prompt out the confirm message to delete the promotion data. |
| 5 | Admin has deleted the promotion data successfully | * The record is deleted successfully. * Return to view promotion list. |

|  |  |  |
| --- | --- | --- |
| Use case: Delete reward data (Promotion reward system) | | |
| Overview: This case describes how the admin deletes reward data | | |
| Pre – conditions:   1. The admin must login. | | |
| Post – conditions:   1. Successful condition  * Admin has deleted reward data successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Promotion/Reward” | * Enter to promotion reward system webpage. |
| 3 | Admin selects “Reward” | * Enter to reward webpage |
| 4 | Search the reward and select “delete” | * Show the searched reward result. * Prompt out the confirm message to delete the reward data. |
| 5 | Admin has deleted the reward data successfully | * The record is deleted successfully. * Return to view reward list. |

|  |  |  |
| --- | --- | --- |
| Use case: Search promotion (Promotion reward system) | | |
| Overview: This case describes how the admin searches promotion | | |
| Pre – conditions:   1. The admin must login. | | |
| Post – conditions:   1. Successful condition  * Admin has searched the promotion successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Promotion/Reward” | * Enter to promotion reward system webpage. |
| 3 | Admin selects “Promotion” | * Enter to promotion webpage. |
| 4 | Search the promotion name | * Show the searched promotion data. |
| 5 | Admin has searched the promotion successfully. | * The promotion data is searched successfully. |

|  |  |  |
| --- | --- | --- |
| Use case: Search reward (Promotion reward system) | | |
| Overview: This case describes how the admin searches reward | | |
| Pre – conditions:   1. The admin must login. | | |
| Post – conditions:   1. Successful condition  * Admin has searched the reward successfully. | | |
| Event | | Response |
| 1 | Admin logins to the system | * Perform admin account validation. |
| 2 | Admin selects “Promotion/Reward” | * Enter to promotion reward system webpage. |
| 3 | Admin selects “Reward” | * Enter to reward webpage |
| 4 | Search the reward name | * Show the searched reward data. |
| 5 | Admin has searched the reward successfully. | * The reward data is searched successfully. |

### ***4.5.12 Online booking system***

|  |  |  |
| --- | --- | --- |
| Use case: Register account (Online booking system) | | |
| Overview: This case describes how the customer registers account. | | |
| Pre – conditions: - | | |
| Post – conditions:   1. Successful condition  * Customer has registered an account and able to login. | | |
| Event | | Response |
| 1 | Customer enters the online webpage. | * Display the homepage of the online booking system. |
| 2 | Select “register” | * Enter to the signup webpage. |
| 3 | Enter all the required information | * Perform information validation. |
| 4 | Customer registers successfully | * Account is created. |

|  |  |  |
| --- | --- | --- |
| Use case: Login (Online booking system) | | |
| Overview: This case describes how the customer logins. | | |
| Pre – conditions:   1. Customer has a registered account. | | |
| Post – conditions:   1. Successful condition  * Customer login to their account successfully. | | |
| Event | | Response |
| 1 | Customer enters the online webpage. | * Display the homepage of the online booking system. |
| 2 | Select “login” | * Enter to the login webpage. |
| 3 | Enter all the required information and click “login” | * Perform information validation. |
| 4 | Customer logins successfully | * Account is logged in successfully. |

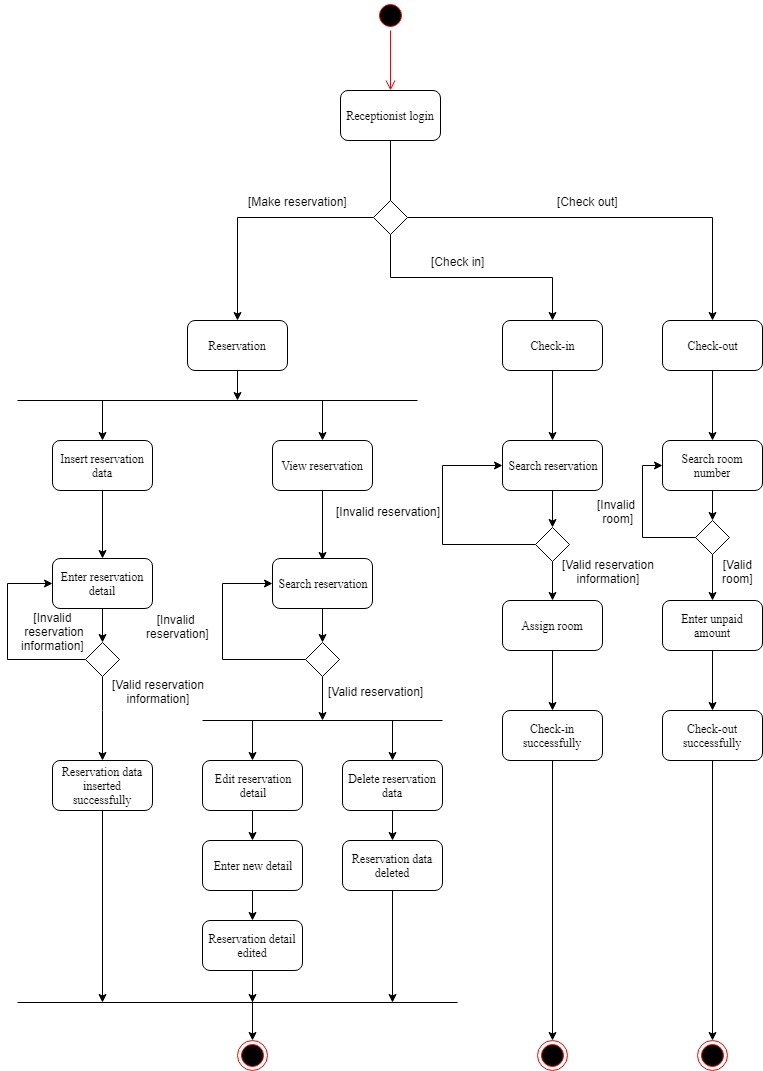
|  |  |  |
| --- | --- | --- |
| Use case: Make online reservation (Online booking system) | | |
| Overview: This case describes how the customer makes online reservation. | | |
| Pre – conditions:   1. Customer has a registered account. | | |
| Post – conditions:   1. Successful condition  * Customer made the online reservation successfully. | | |
| Event | | Response |
| 1 | Customer enters the online webpage. | * Display the homepage of the online booking system. |
| 2 | Select “Book a reservation” | * Enter to the online reservation webpage. |
| 3 | Enter all the required information and click “Payment” | * Perform information validation. |
| 4 | Enter the payment information and click “submit” | * Perform information validation. |
| 5 | Customer has made the online reservation successfully | * Online reservation is made successfully. |

|  |  |  |
| --- | --- | --- |
| Use case: Edit profile (Online booking system) | | |
| Overview: This case describes how the customer edits profile. | | |
| Pre – conditions:   1. The customer must login. | | |
| Post – conditions:   1. Successful condition  * Customer has updated their profile successfully. | | |
| Event | | Response |
| 1 | Customer logins to the online webpage | * Perform customer account validation. |
| 2 | Customer selects “Profile” | * Enter to the profile webpage. |
| 3 | Customer selects “Edit” | * Enter to the profile form webpage. |
| 4 | Customer enter details that want to edit | * Perform information validation. |
| 5 | Customer selects “Update” | * Prompt out the successful of updating message. |
| 6 | Customer has updated information successfully | * Updated successfully. * Prompt out success message. * Return to the webpage. |

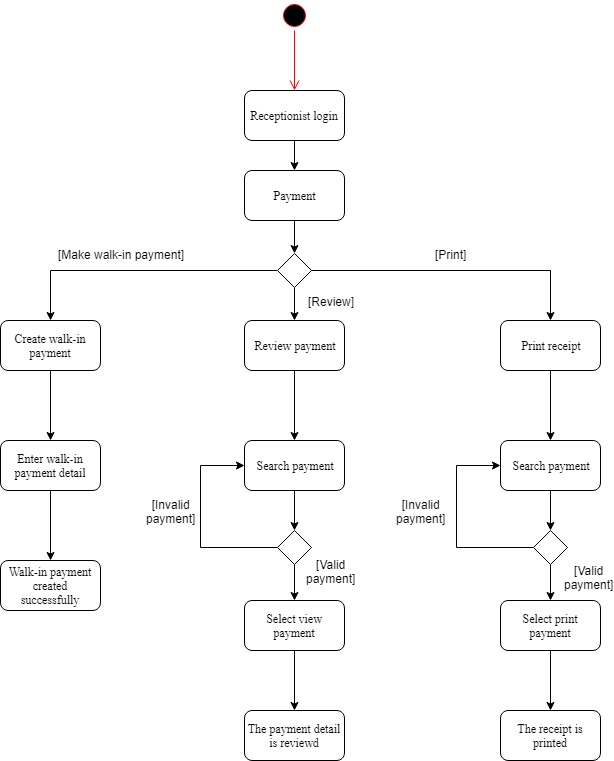
|  |  |  |
| --- | --- | --- |
| Use case: Redeem reward (Online booking system) | | |
| Overview: This case describes how the customer redeems the reward. | | |
| Pre – conditions:   1. The customer must login. 2. The customer must has redeem points. | | |
| Post – conditions:   1. Successful condition  * Customer has redeemed the reward successfully. | | |
| Event | | Response |
| 1 | Customer logins to the online webpage | * Perform customer account validation. |
| 2 | Customer selects “Redeem” | * Enter to the redeem webpage. |
| 3 | Customer selects a reward item and click “Redeem” button. | * Display different available alternative. |
| 4 | Customer has redeemed the reward successfully and able to pick up the redeemed item in the hotel | * Redeemed successfully. * Prompt out success message. * Return to the webpage. |

## **4.6 Activity diagram**

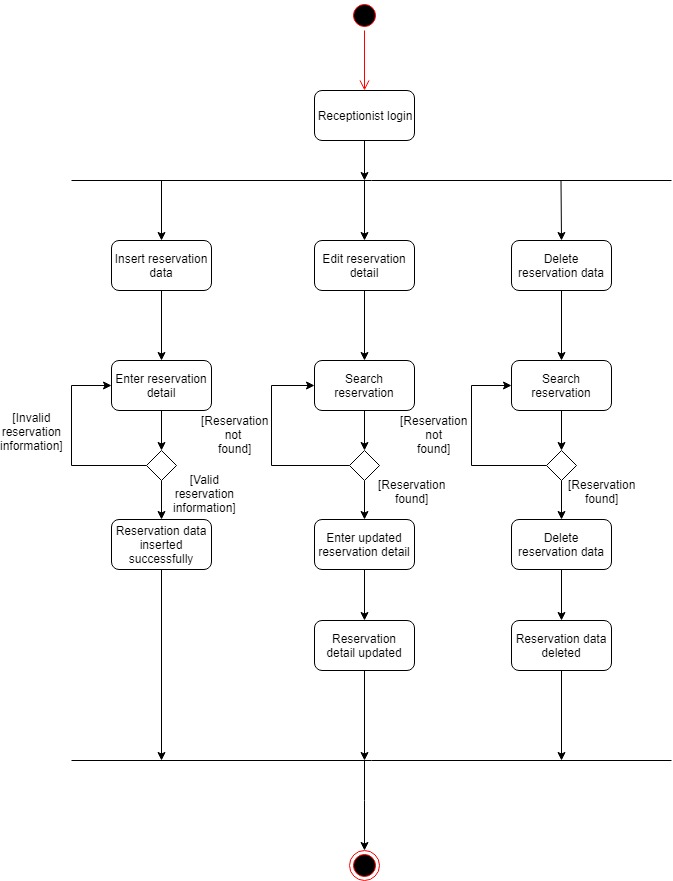
### ***4.6.1 Front-desk system***



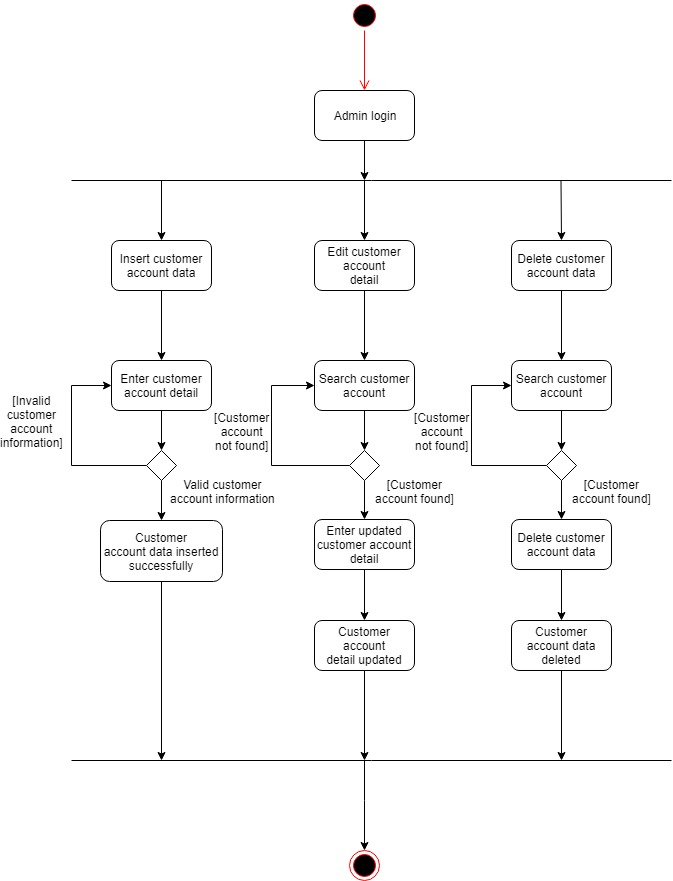
### ***4.6.2 Payment system***

****

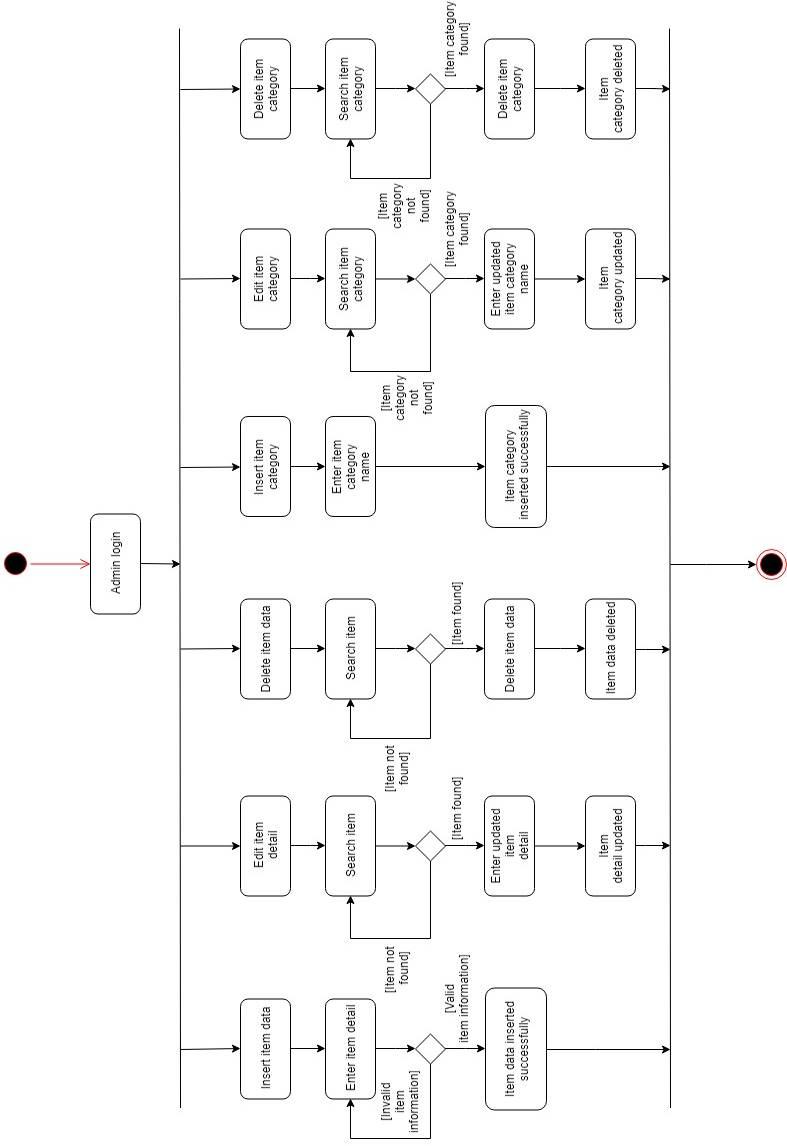
### ***4.6.3 Reservation detail system***

****

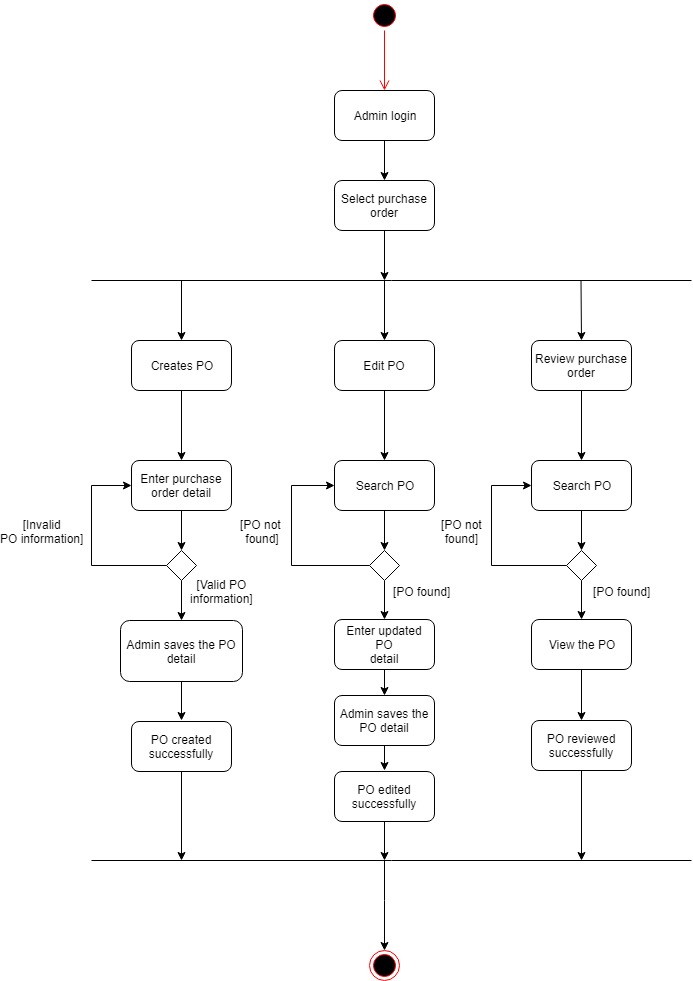
### ***4.6.4 Customer account system***

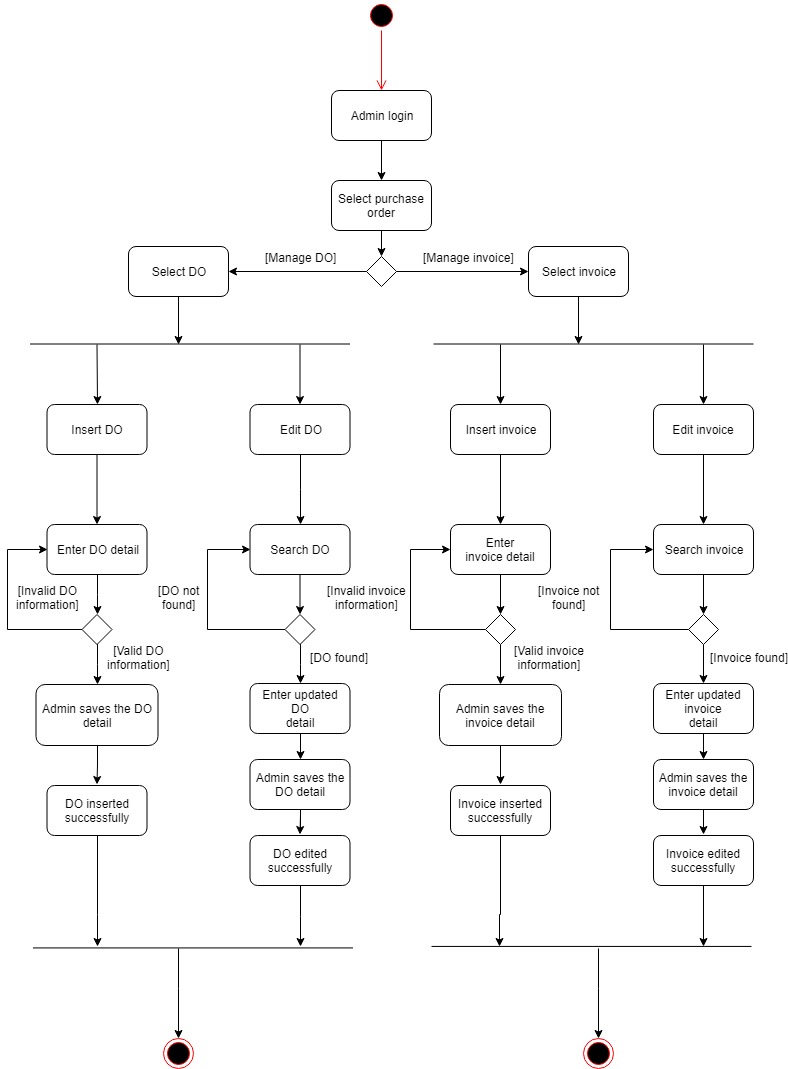
****

### ***4.6.5 Inventory system***

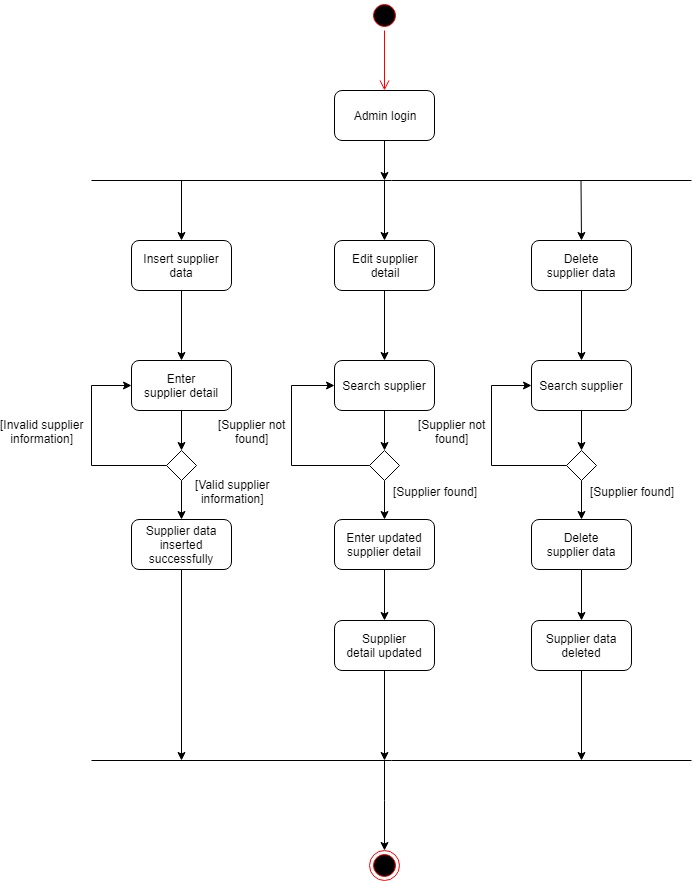
****

### ***4.6.6 Purchase order system***

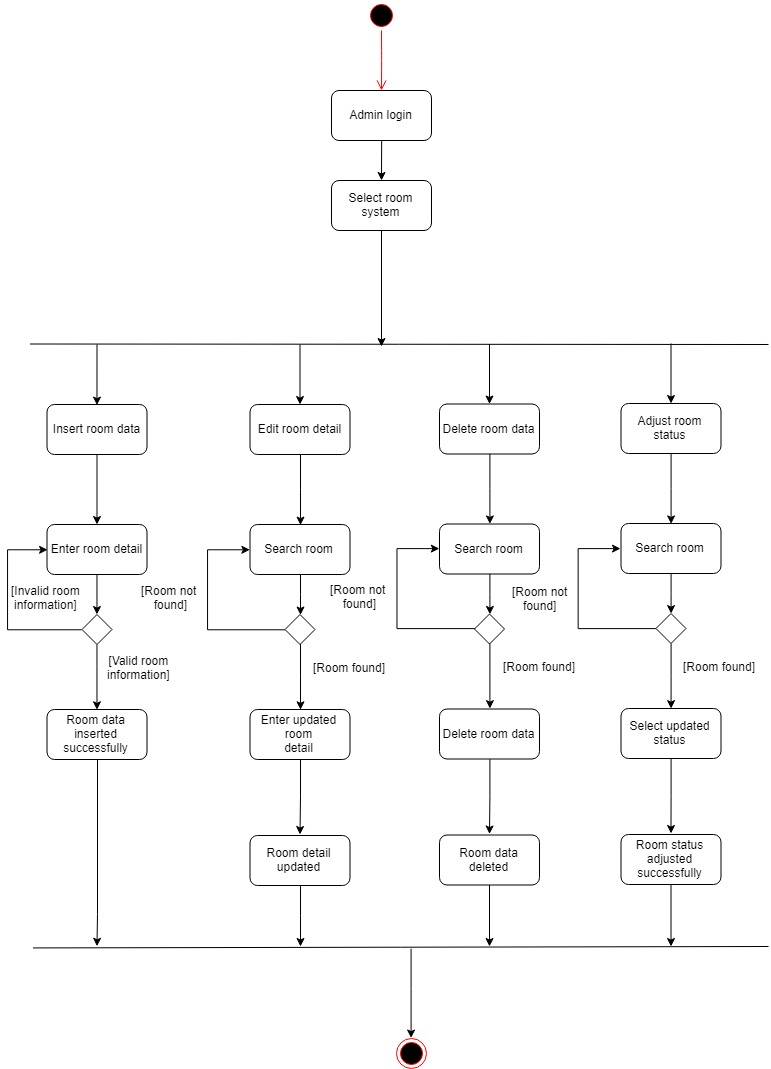
****

****

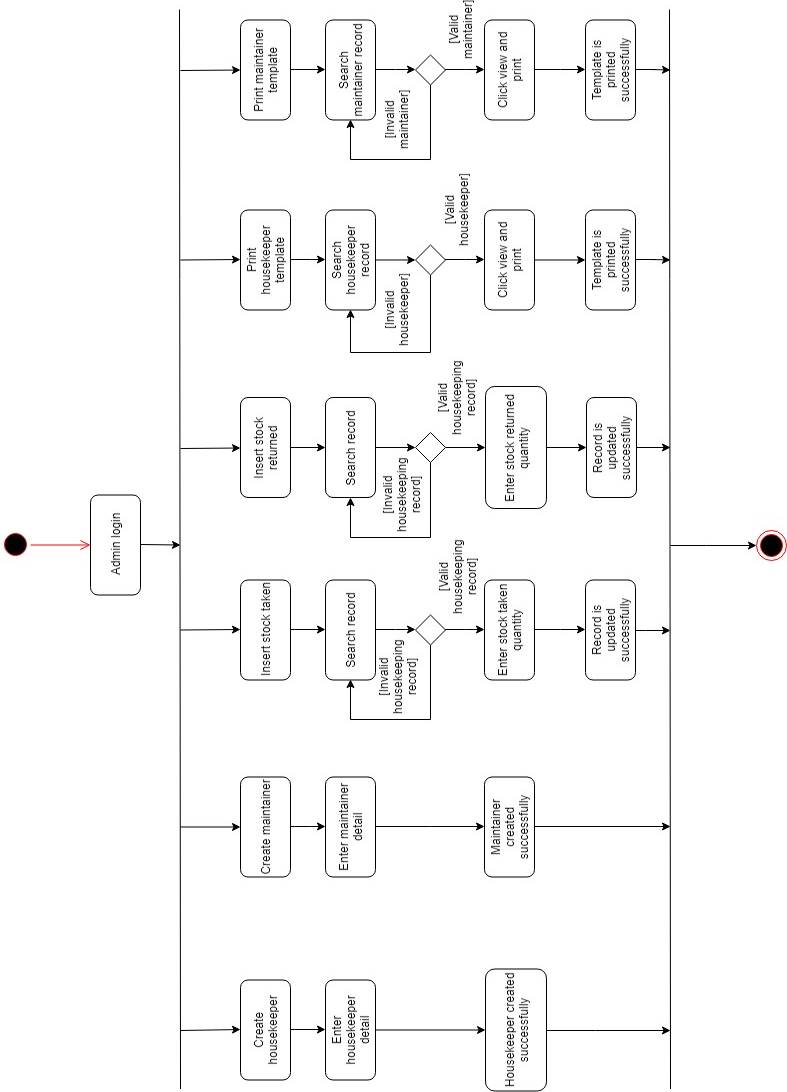
### ***4.6.7 Supplier system***

****

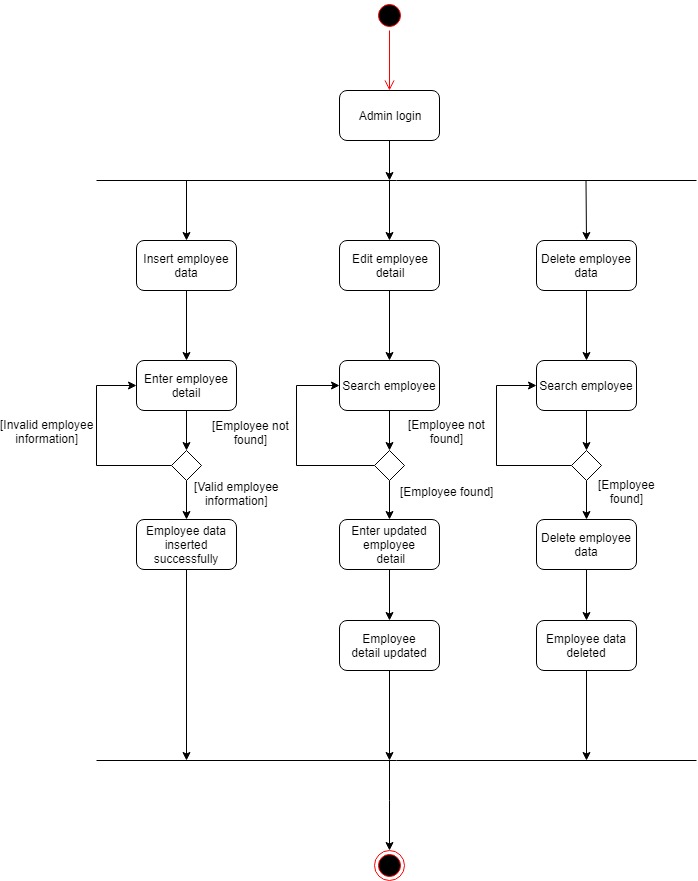
### ***4.6.8 Room information system***

****

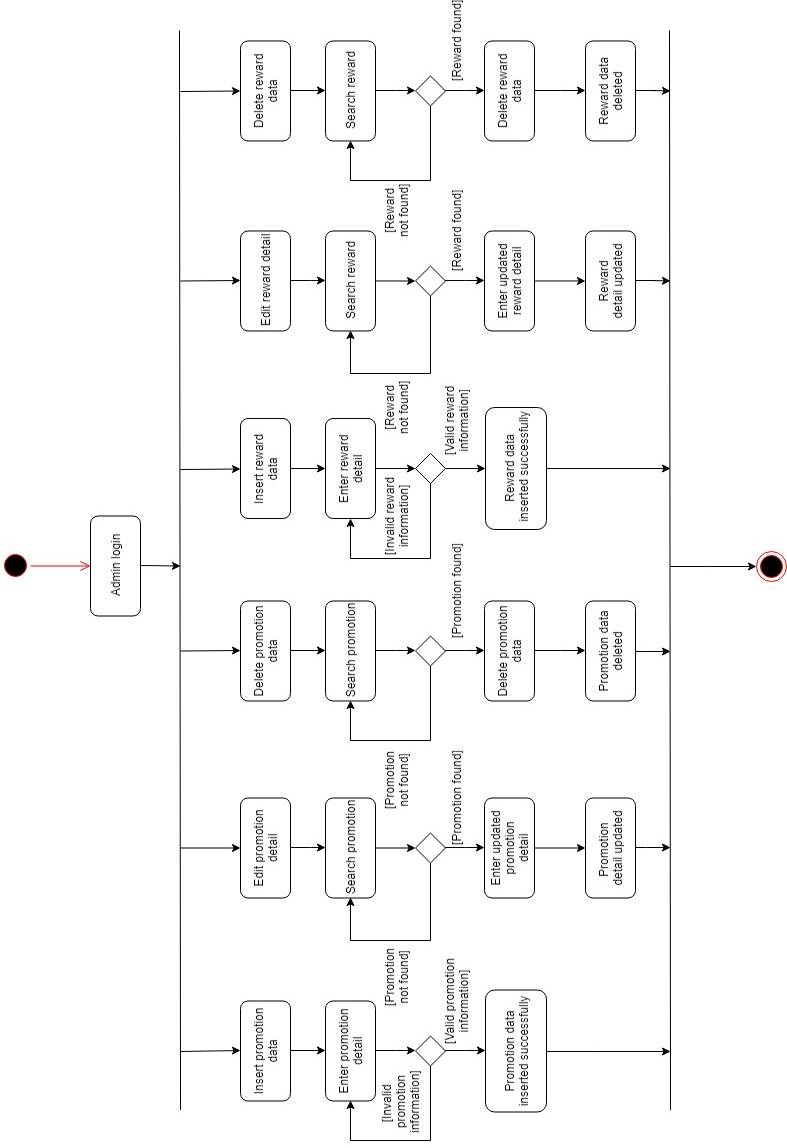
### ***4.6.9 Housekeeping system***

****

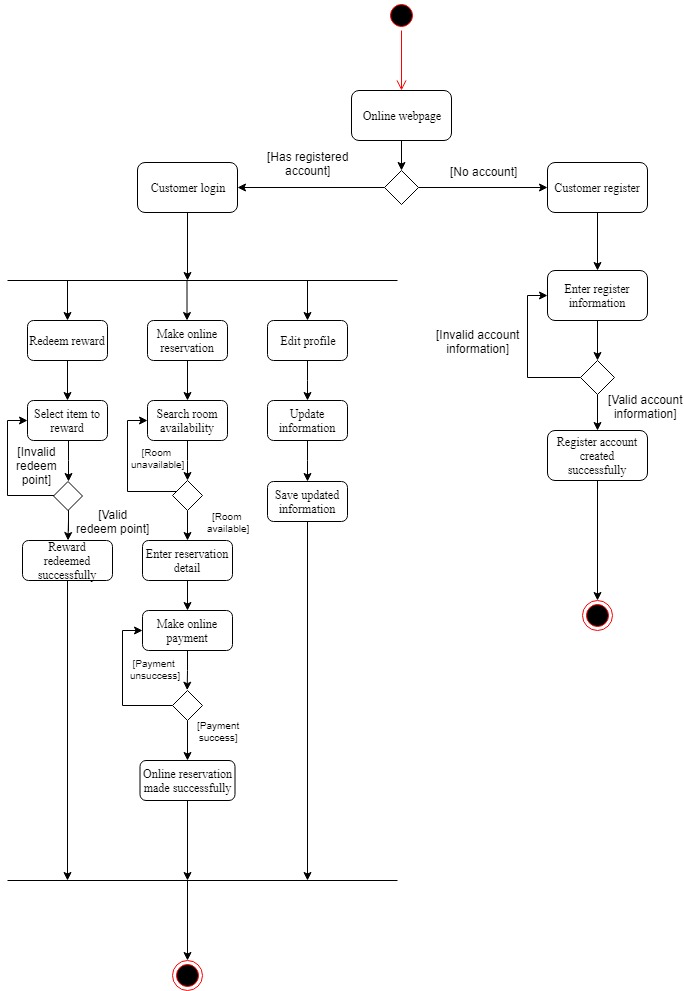
### ***4.6.10 Employee system***

****

### ***4.6.11 Promotion reward system***

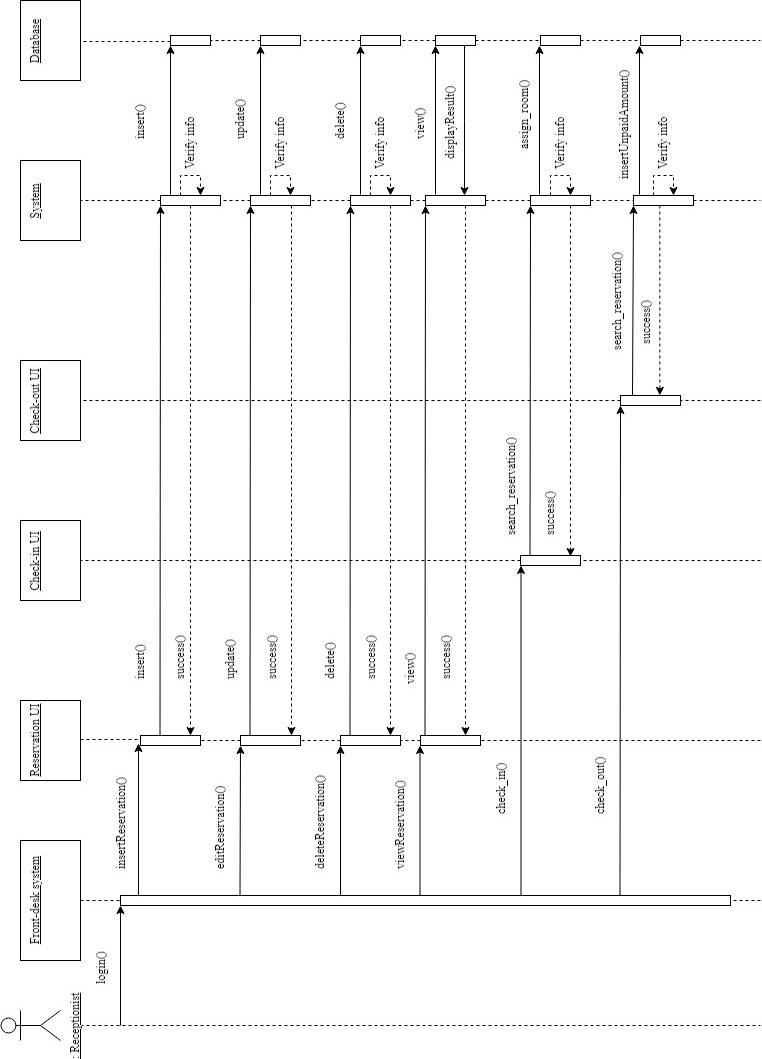
****

### ***4.6.12 Online booking system***

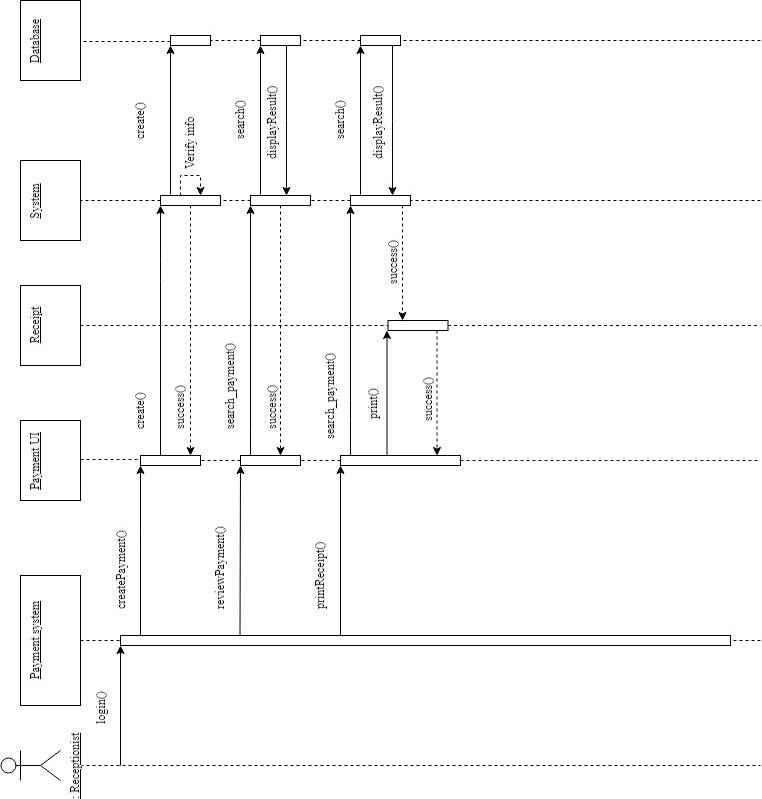


## **4.7 Sequence diagram**

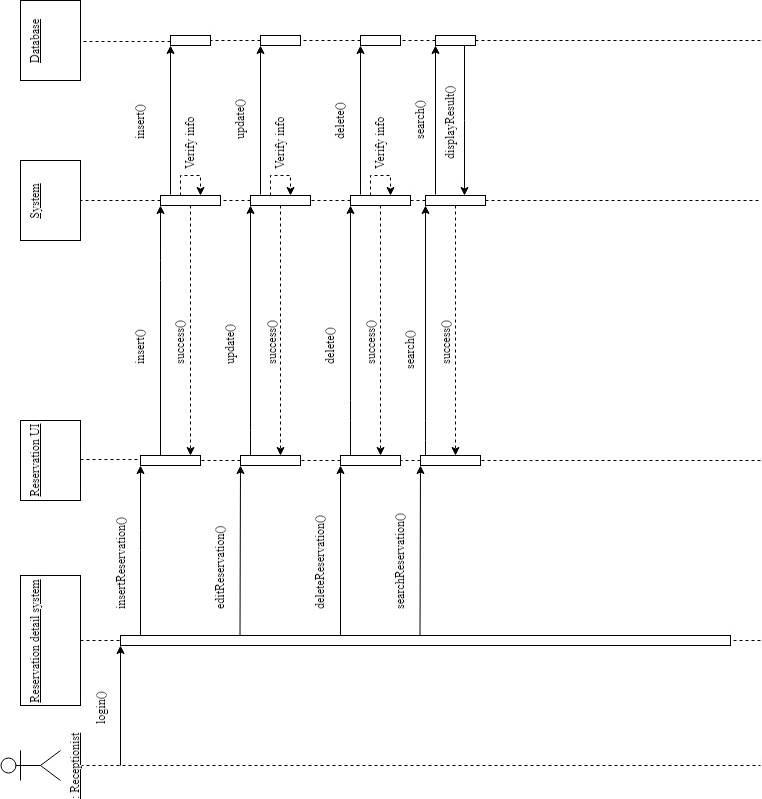
### ***4.7.1 Front-desk system***



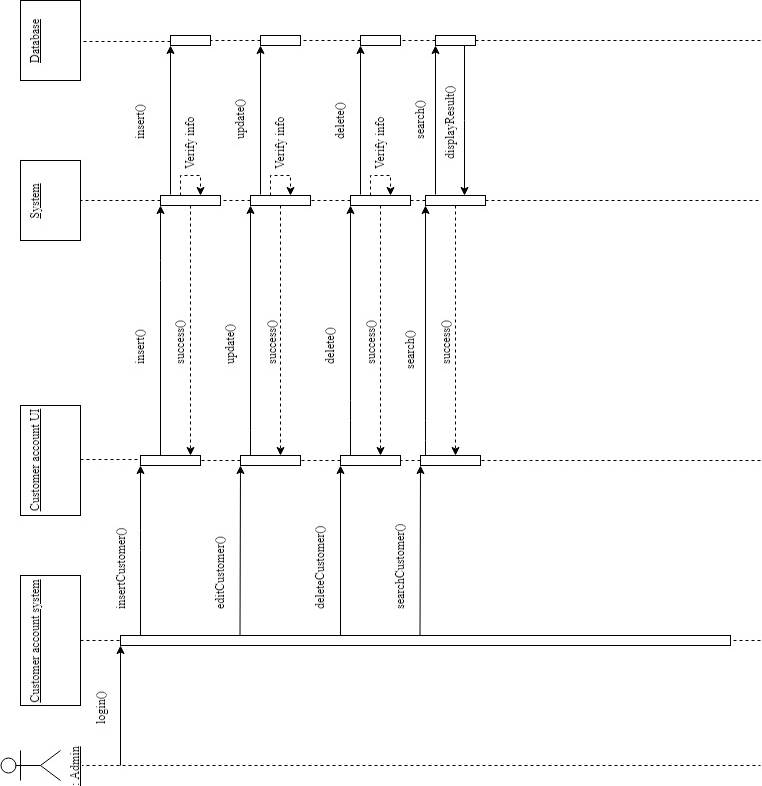
### ***4.7.2 Payment system***



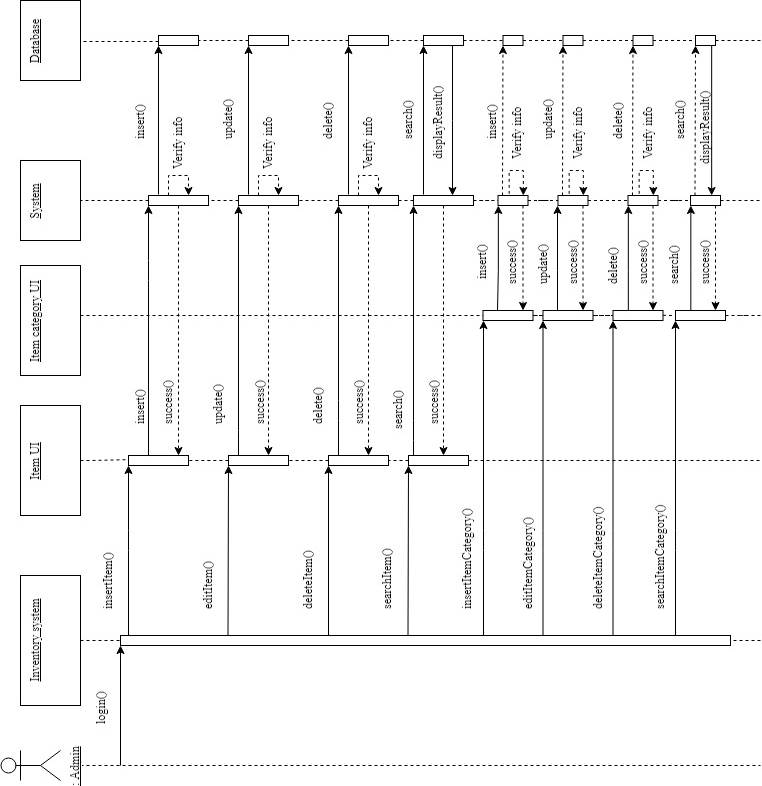
### ***4.7.3 Reservation detail system***



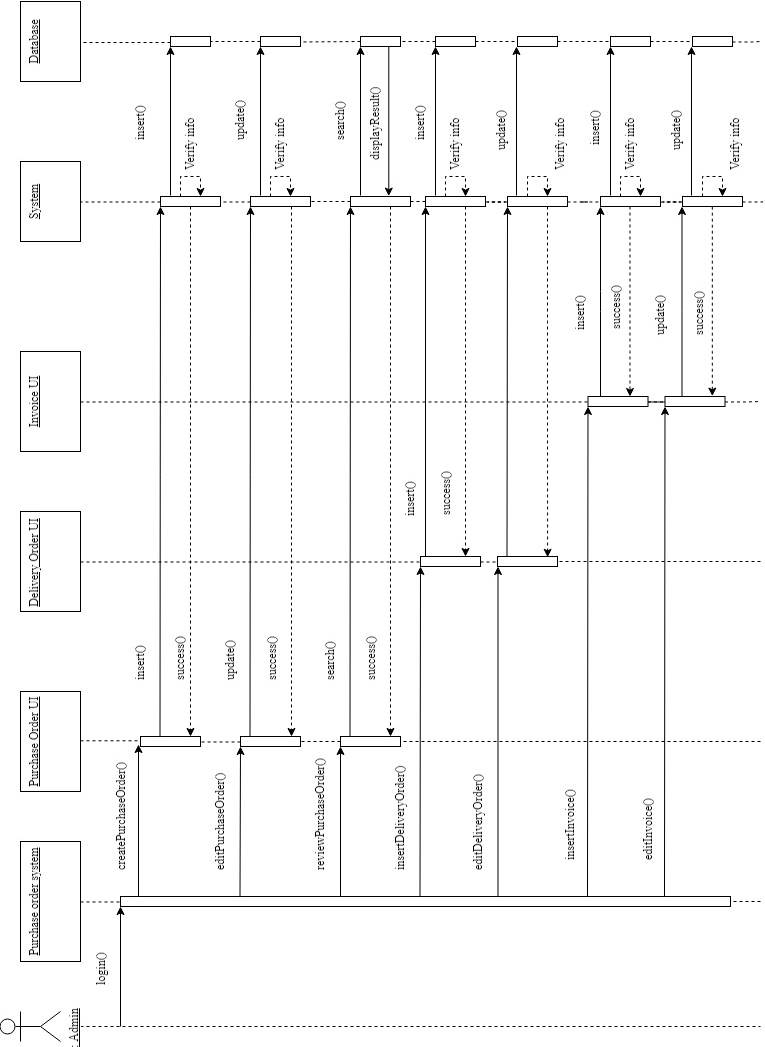
### ***4.7.4 Customer account system***



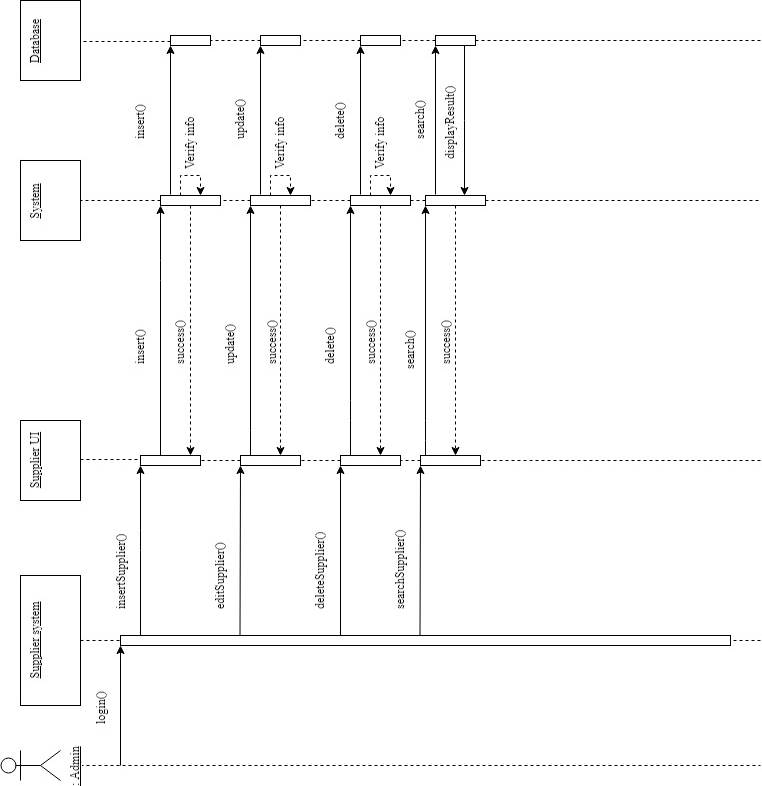
### ***4.7.5 Inventory system***



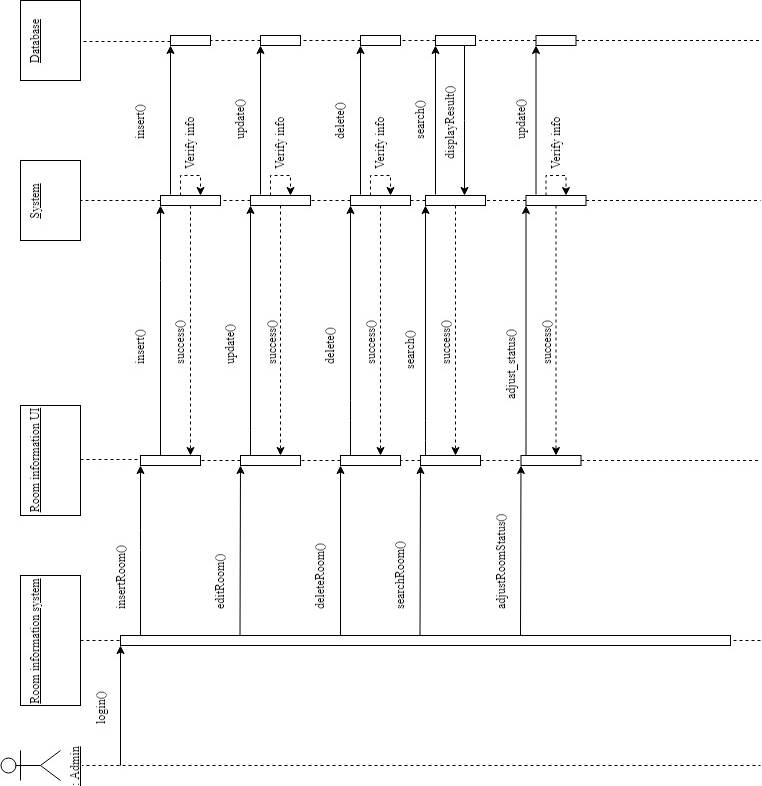
### ***4.7.6 Purchase order system***



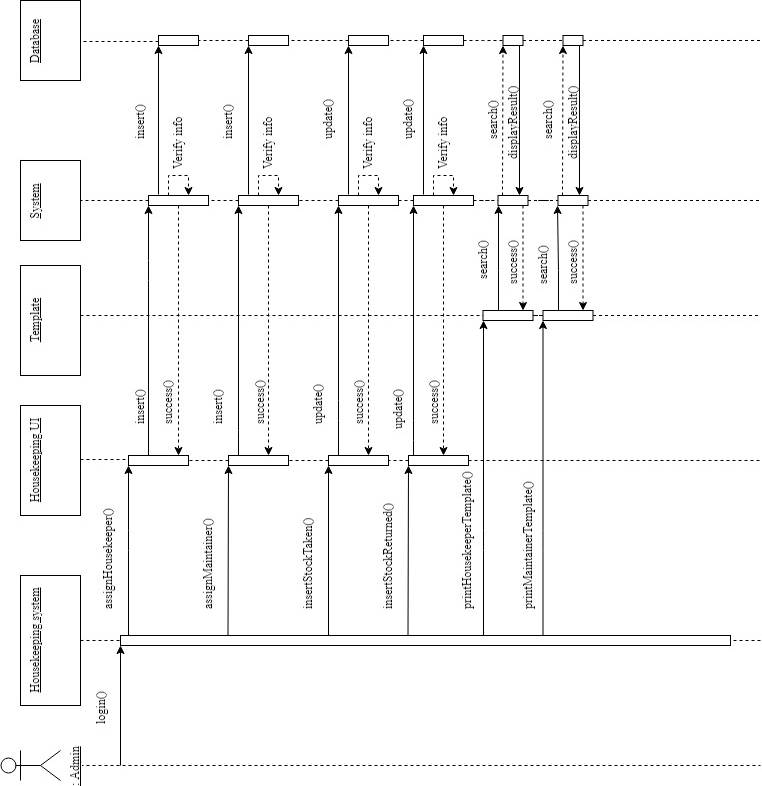
### ***4.7.7 Supplier system***



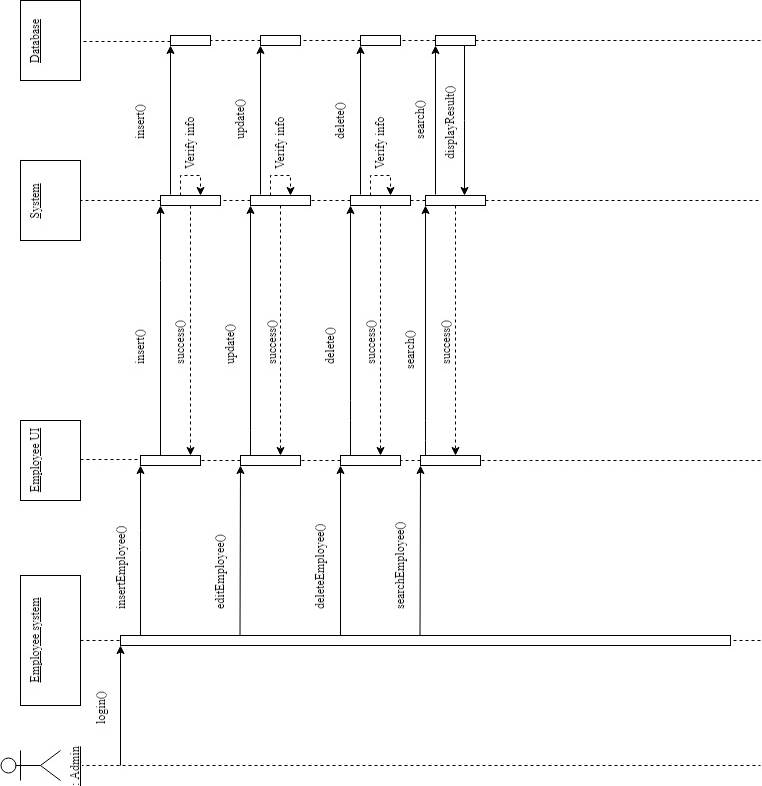
### ***4.7.8 Room information system***



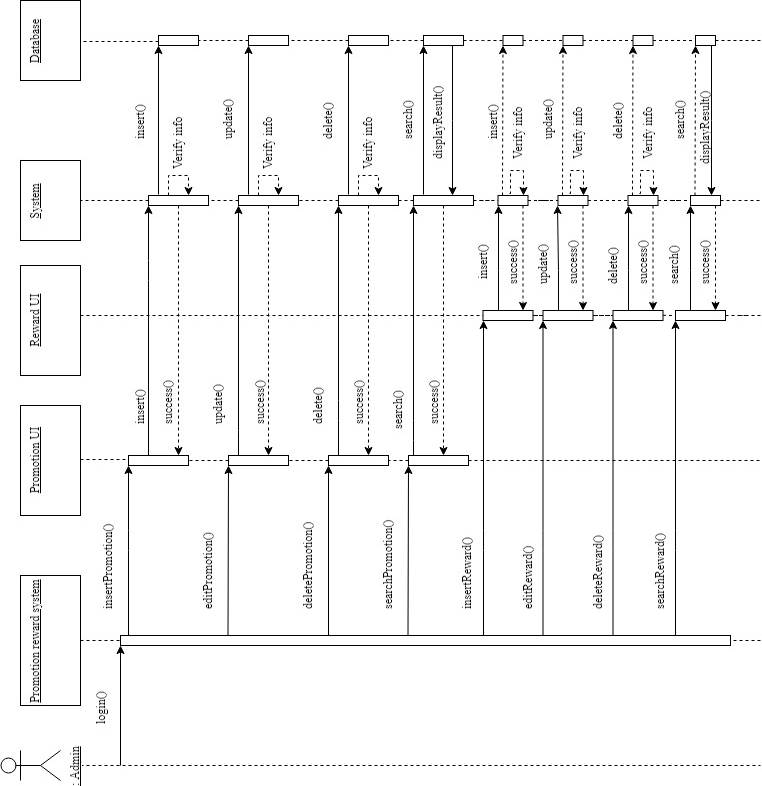
### ***4.7.9 Housekeeping system***



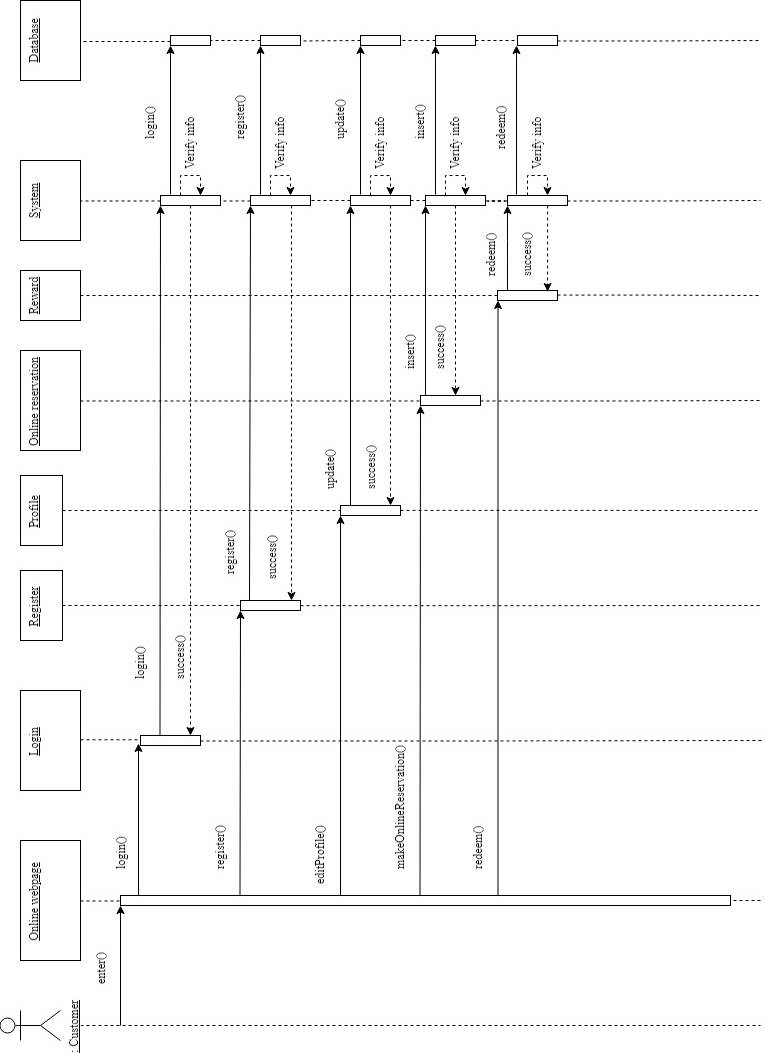
### ***4.7.10 Employee system***



### ***4.7.11 Promotion reward system***



### ***4.7.12 Online booking system***



# Implementation

The following section will describe the implementation of system.

**5.1 System Interface Design**

According to Margaret Rouse [6], she said that the user interface (UI) is everything designed into an information device with which a person may interact. This can include display screens, keyboards, a mouse and the appearance of a desktop. It is also the way through which a user interacts with an application or a website. The growing dependence of many companies on web applications and mobile applications has led many companies to place increased priority on UI in an effort to improve the user's overall experience. So, UI is very important for the user to interact with the system.

Below are the main characteristics that applied to hotel management admin web-based system:

1. Clarity (Clear)

* With the clarity of user interface system, admin user is able to interact with the system easily and does not spend excess time in interacting the system. So, the characteristic of clarity of user interface in hotel management system is able to allow the admin to perform the functions easily in the daily.

1. Consistent

* The arrangement and patterns of the interface elements must be consistent to prevent the confusion of the admin user when he/she interacts to the system. Furthermore, the consistent of the interface elements also can speed up the daily process due to the familiar of the user interface.

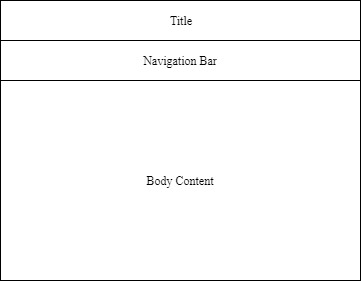
***5.1.1 Admin Web-Based System Interface Design***

In hotel management system, the web-based system for admin is designed by using HTML 5 and CSS. In this web-based system, it consists of three parts, which are title bar, navigation bar and the body content.

For the navigation bar, it is placed below of the title bar of the system. It consists of buttons for selections to the sub-system. The corresponding page will displayed in body content when the button is clicked.

For the title bar, it is placed on the top of the screen of the system. It contains title of the page and it is saw easily.

For the body content, it displays the data. It is also contains few buttons to allow admin performs the functions. The layout is shown as below.



**5.2 System Catalogue**

1. Admin

|  |  |  |  |
| --- | --- | --- | --- |
| No | Field | Type | Extra |
| 1. 1 | admin\_id (primary key) | Int (10) | Auto\_increment |
| 1. 2 | admin\_username | Varchar (50) | - |
| 1. 3 | admin\_password | Varchar (50) | - |

1. Customer

|  |  |  |  |
| --- | --- | --- | --- |
| No | Field | Type | Extra |
| 1. 1 | customer\_id (primary key) | Int (10) | Auto\_increment |
| 1. 2 | name | Varchar (50) | - |
| 1. 3 | phone\_no | Varchar (20) | - |
|  | IC | Varchar (30) | - |
|  | email\_address | Varchar (50) | - |
|  | password | Varchar (30) | - |
|  | address | Varchar (200) | - |
|  | zip\_code | Int (10) | - |
|  | city | Varchar (20) | - |
|  | state | Varchar (20) | - |
|  | country | Varchar (20) | - |
|  | online\_account\_status | Varchar (20) | - |
|  | reward\_point | Int (20) | - |

1. Employee

|  |  |  |  |
| --- | --- | --- | --- |
| No | Field | Type | Extra |
| 1. 1 | customer\_id (primary key) | Int (10) | Auto\_increment |
| 1. 2 | name | Varchar (50) | - |
| 1. 3 | phone\_no | Varchar (20) | - |
|  | IC | Varchar (30) | - |
|  | email\_address | Varchar (50) | - |
|  | address | Varchar (200) | - |
|  | zip\_code | Int (10) | - |
|  | city | Varchar (20) | - |
|  | state | Varchar (20) | - |
|  | country | Varchar (20) | - |
|  | position | Varchar (50) | - |

1. Room\_information

|  |  |  |  |
| --- | --- | --- | --- |
| No | Field | Type | Extra |
| 1. 1 | room\_id (primary key) | Int (10) | Auto\_increment |
| 1. 2 | room\_no | Int (10) | - |
| 1. 3 | room\_type\_id | Int (10) | - |
|  | room\_status | Varchar (10) | - |
|  | HK\_status | Varchar (10) | - |

1. Room\_type

|  |  |  |  |
| --- | --- | --- | --- |
| No | Field | Type | Extra |
| 1. 1 | supplier\_id (primary key) | Int (10) | Auto\_increment |
| 1. 2 | room\_name | Varchar (20) | - |
| 1. 3 | room\_price | Double(10,2) | - |
|  | description | Varchar (100) | - |

1. Supplier

|  |  |  |  |
| --- | --- | --- | --- |
| No | Field | Type | Extra |
| 1. 1 | supplier\_id (primary key) | Int (10) | Auto\_increment |
| 1. 2 | company\_name | Varchar (100) | - |
| 1. 3 | company\_phone\_no | Varchar (20) | - |
|  | company\_address | Varchar (200) | - |
|  | company\_email | Varchar (50) | - |

# Testing

Testing is an important part of software of software development. It is an activity that allow us to test and evaluate our system that we develop to meet its required results. It is crucial for us to find any remaining bugs or error in the system before being release to the general public. A test should be held when there any slight change in the source. To follow this ideology we will be testing every single function in our system.

**6.1 Approach**

Since this is final year project 1 and the system is not completed yet, so I decide to choose the unit testing to test each features separately. Unit testing is a level of software testing where individual units or components of the software is tested. A unit is the smallest testable part of any software. It usually has one or a few inputs and usually a single output. Hence, I perform the unit testing by manually.

**6.2 Features to be tested**

Hotel Management System features to be tested.

|  |  |
| --- | --- |
| Features | Description |
| Login | Login as admin user |
| Logout | Logout from the management system |
| Room information system | Add/edit/delete the room data and detail |
| Supplier system | Add/edit/delete the supplier data and detail |
| Employee system | Add/edit/delete the employee data and detail |
| Customer account system | Add/edit/delete the room data and detail |

**6.3 System testing**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **System Testing of Hotel Management System (login)** | | | | |
| No | Test action | Expected Result | Pass/Fail | Comments |
| 1 | Enter admin username and admin password | Login to the management system | Pass | - |
| 2 | Press “Logout” button | Logging out from the management system and not being able to return back | Pass | - |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **System Testing of Hotel Management System (Room information system)** | | | | |
| No | Test action | Expected Result | Pass/Fail | Comments |
| 1 | Create room data | New room shows up with all the newly entered information | Pass | - |
| 2 | Edit room detail | Room information shown with the updated details. | Pass | - |
| 3 | Delete room data | Removal of a room data | Pass | - |
| 4 | Adjust room status | The status of the room is updated. | Pass | - |
| 5 | Search room | Only rooms with the similar search keyword are shown | Pass | - |
| 6 | Create room type data | New room type shows up with all the newly entered information | Pass | - |
| 7 | Edit room type detail | Room type detail shown with the updated details. | Pass | - |
| 8 | Delete room type data | Removal of a room type data | Pass | - |
| 9 | Search room type | Only room type names with the similar searched keyword are shown | Pass | - |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **System Testing of Hotel Management System (Supplier system)** | | | | |
| No | Test action | Expected Result | Pass/Fail | Comments |
| 1 | Create supplier data | New supplier shows up with all the newly entered information | Pass | - |
| 2 | Edit supplier detail | Supplier detail shown with the updated details. | Pass | - |
| 3 | Delete supplier data | Removal of a supplier data | Pass |  |
| 4 | Search supplier | Only supplier name with the similar search keyword are shown | Pass |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **System Testing of Hotel Management System (Employee system)** | | | | |
| No | Test action | Expected Result | Pass/Fail | Comments |
| 1 | Create employee data | New employee shows up with all the newly entered information | Pass | - |
| 2 | Edit employee detail | Employee detail shown with the updated details. | Pass | - |
| 3 | Delete employee data | Removal of a employee data | Pass |  |
| 4 | Search employee | Only employee name with the similar search keyword are shown | Pass |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **System Testing of Hotel Management System (Customer account system)** | | | | |
| No | Test action | Expected Result | Pass/Fail | Comments |
| 1 | Create customer account data | New customer account shows up with all the newly entered information | Pass | - |
| 2 | Edit customer account detail | Customer account detail shown with the updated details. | Pass | - |
| 3 | Delete customer account data | Removal of a customer account data | Pass |  |
| 4 | Search customer account | Only customer account name with the similar search keyword are shown | Pass |  |

# 7.0 Evaluation

According to the main objective within the project Hotel Management System which is developing a system that allows the hotel admin to manage the daily operations easily and allows the customer to online make the reservation. This system has satisfied the requirements of having a reservation detail system and online booking system that allow the customer to make the reservation via online. It also keeps track of everything nicely.

This project also includes a customer account system for the admin to view the history of customer’s reservations and track the customer’s preference. So, it is also help the hotel to increase the sales by providing better services for the customer. Moreover, it is also reduce the time of check in process when the customer account is existed in the system. In conclusion this system meets its initial requirements.

Overall this final year project 1 still is not a complete system but it still has lots and lots of improvement that I can do in the future to make this an even better system. Hotel Management System should continue to be improved so that I can stand our ground against others that have been release to the public.

# 8.0 Recommendations

This Section will be dedicated to list my several improvements which our Hotel Management System can be explored in future development

Make a mobile version of the online booking system

With the rate of smartphone being improve on and how smartphone has become part of our daily life. Because of this and the general portability of smartphone make me consider to port into the android platform. So the customer can use it wherever they.

Develop a Sale report generator

To let me have a better idea of how’s the hotel business doing a sale report generator will let me have a general idea of how’s the hotel business being doing and how to improve it much more quickly.

Improve security

With hackers all around the world it would be the best for me to put my focus on my system security sometime in the future. Like checking for loopholes in my system or exploit in my system to prevent hacker from getting the upper hand on us.

# 9.0 Conclusion

Hotel management system the main objective for me is to have an ease of use to the system for hotel admin to manage the daily operations without facing lots of problem. Hotel management System will help admin to keep track of the rooms status in the hotel daily. Hotel Management System is a very large project for me and this is my first time developing a complete system.

In this semester I spend about 2 weeks to think up a system that I want to do and all the background research, investigation and analysis how does a hotel management system works. After I completed all my research we spend 2 months to draw all the require diagram and finish up the system.

During the developing phase I meet a lot of problem either it on my design diagram or codes but I still press on by read up the solution via online, asking my friends and lecturers for advices and solutions. Beside that during in my time of developing this system, I learnt a lot of new ways to approach a problem and new programming skills.

In conclusion, I am very happy to have a chance to develop my Hotel Management System. This project gave a slight idea on the how much hard work it takes to develop a full fledge system. With the experience I got from this project I hope become more efficient in developing a system.

# References

1. Ghosh, S. (2008). *Check in Check Out.* Global Media.
2. Grimes, R. (1994). Hotel operations: A room with a technological view. *Nation's Restaurant News; New York*, 47.
3. Oracle. (2019). *What is MySQL?* Retrieved from MySQL: http://dev.mysql.com/doc/refman/4.1/en/what-is-mysql.html
4. *PHP*. (2019). Retrieved from https://www.php.net/manual/en/intro-whatis.php
5. ProfessionalQA. (31 December, 2018). *What is Iterative Model?* Retrieved from ProfessionalQA.com: http://www.professionalqa.com/iterative-model
6. Rouse, M. (November, 2016). *User Interface (UI)*. Retrieved from Search Microservices Web site: https://searchmicroservices.techtarget.com/definition/user-interface-UI
7. *Tune Hotels*. (2019). Retrieved from https://www.tunehotels.com/

# APPENDIX A – Curriculum Vitae



Name : Ang Sheng Yang

Age : 21

I/C No : 980108-01-6351

Address : 46, Jalan Pulai,

76, Taman Pulai,

Utama, 81300

Skudai, Johor.

H/P Number : 0167368892

Class : BoSE-18C2

Student ID : B180226C

Email : [angshengyang@gmail.com](mailto:angshengyang@gmail.com)

Education Qualifications:

|  |  |  |
| --- | --- | --- |
| **Name of Institution** | **From-To** | **Highest Qualification Attained** |
| SJK(C) Pulai | 2005-2010 | UPSR |
| SMK Taman Desa Skudai | 2011-2015 | SPM |
| Southern University College | 2016-2018 | Diploma in Computer Science |
| Southern University College | 2019 | Bachelor in Information Technology (Hons) in Software Engineering |

Ability Skill : Microsoft Office

Adobe Photoshop

HTML 5

Java Programming

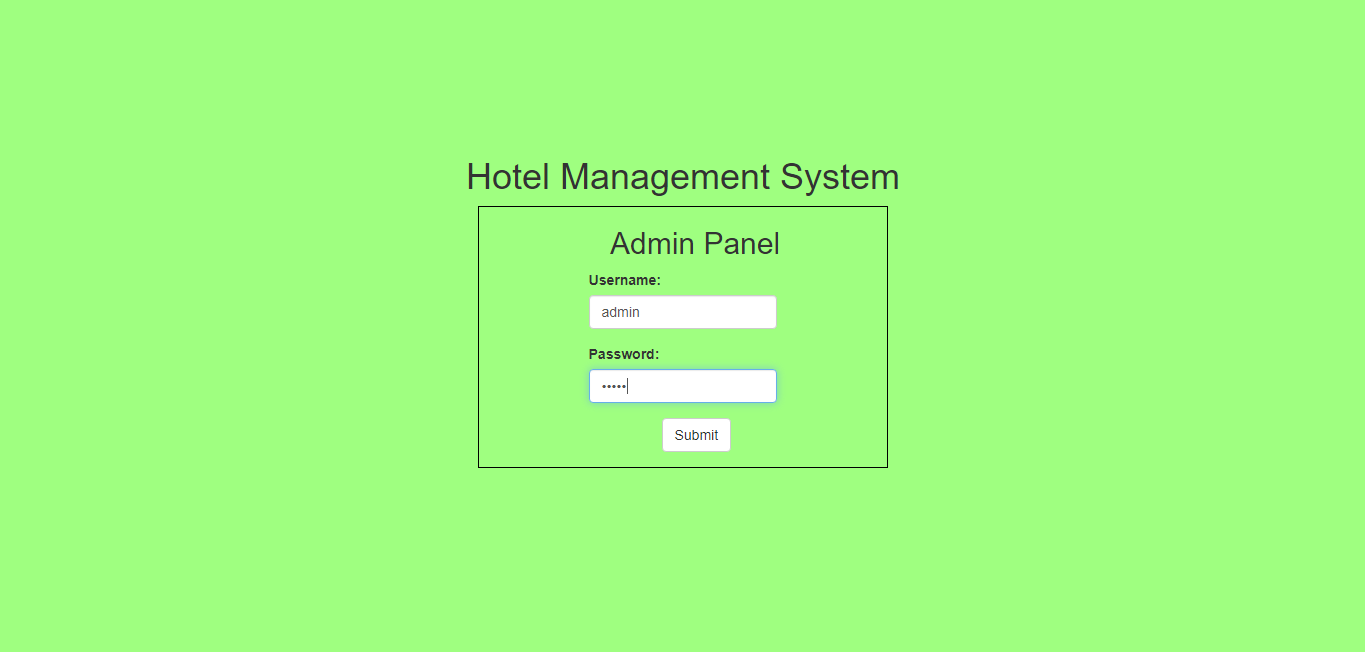
PHP & MySQL

Language : (Proficiency: Best=10, Worst=1)

|  |  |
| --- | --- |
| **Language** | **Proficiency** |
| Malay | 7 |
| English | 7 |
| Mandarin | 7 |
| Hokkien | 6 |

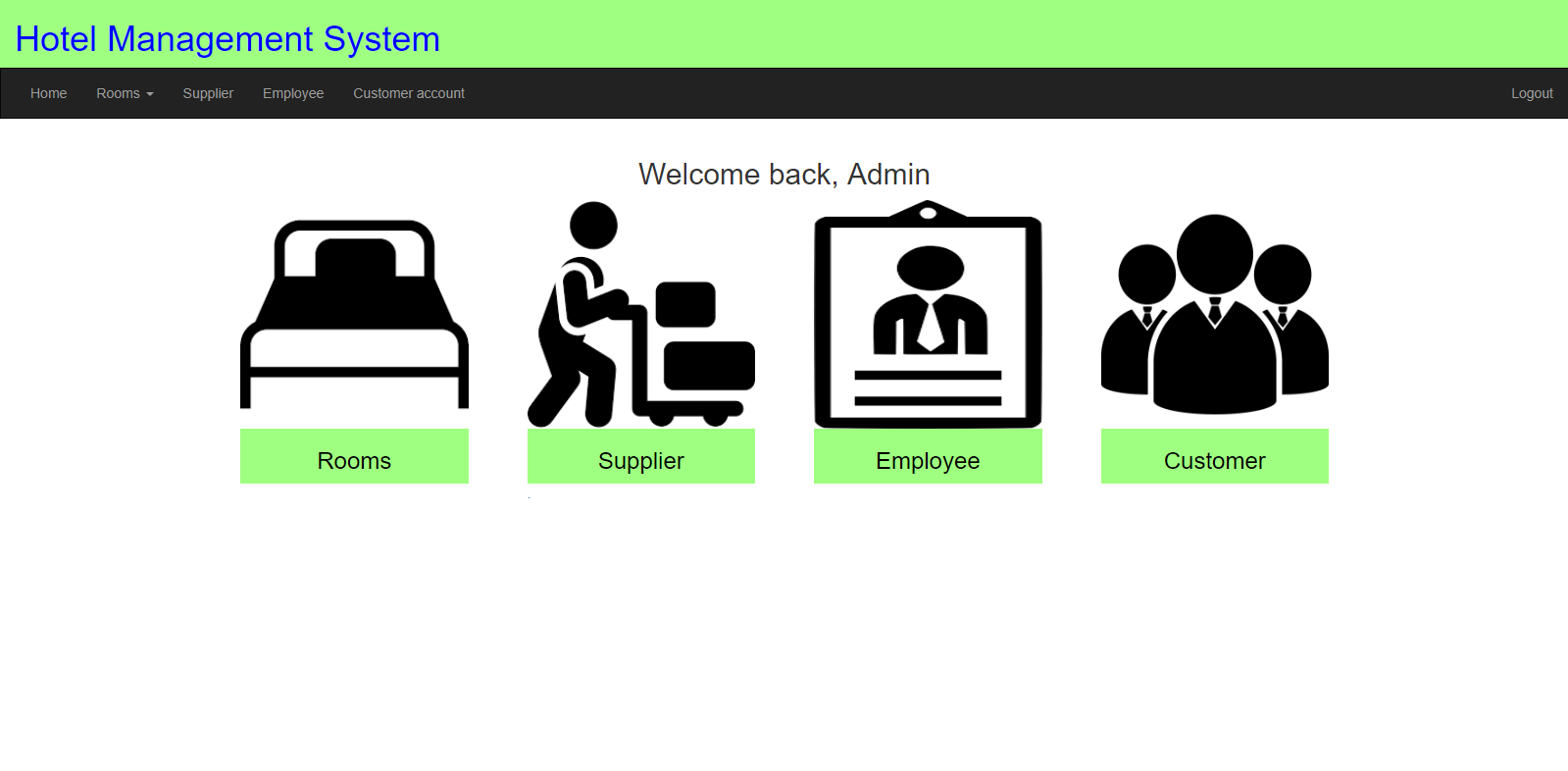
# APPENDIX B – User Guide

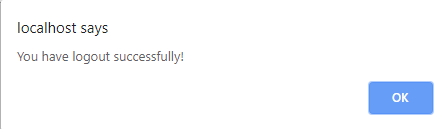
**Figure 1.0 Login Page**



**Figure 1.0** is the login page to the admin panel of Hotel Management System. If the username or password is wrong it will just refresh the page instead of entering the admin panel of Hotel Management System.

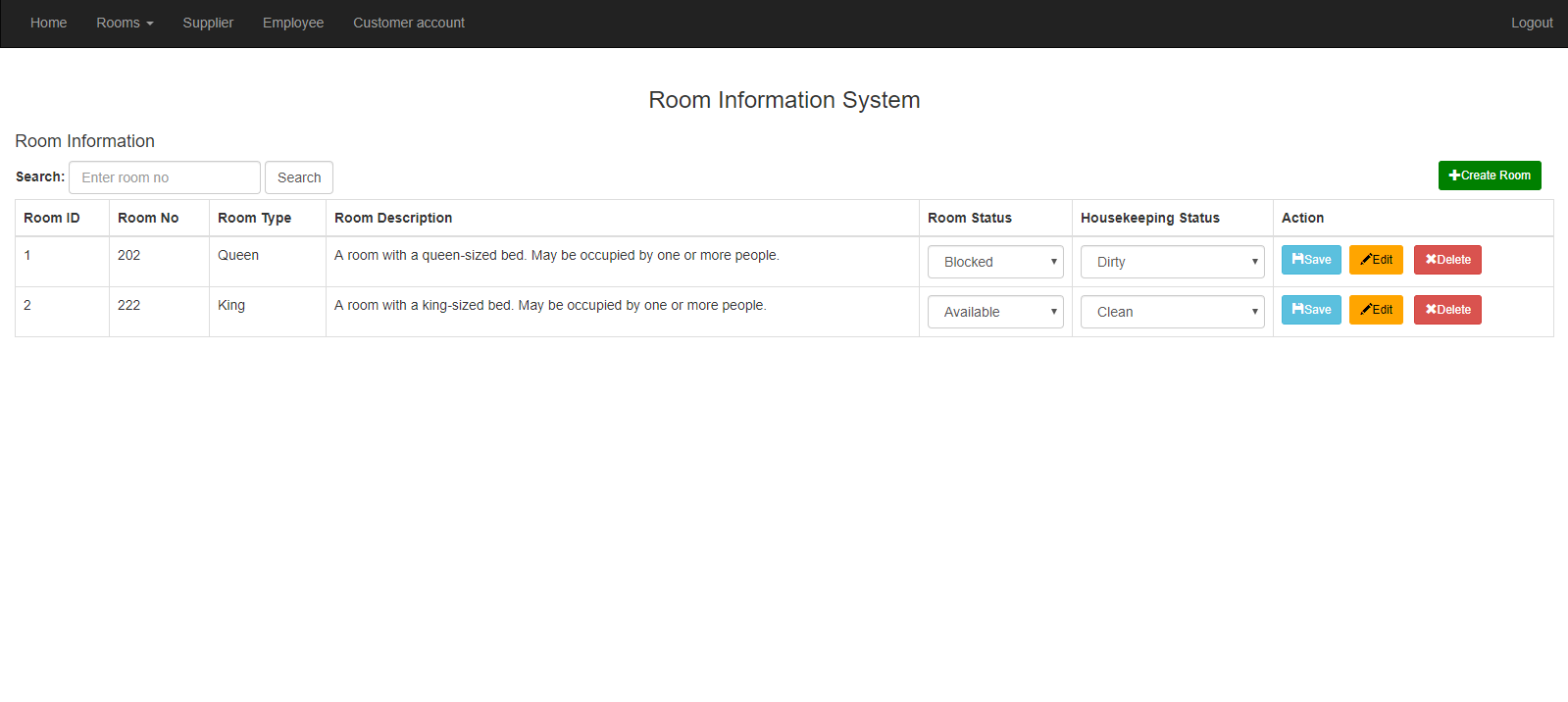
**Figure 2.0 Hotel Management System**





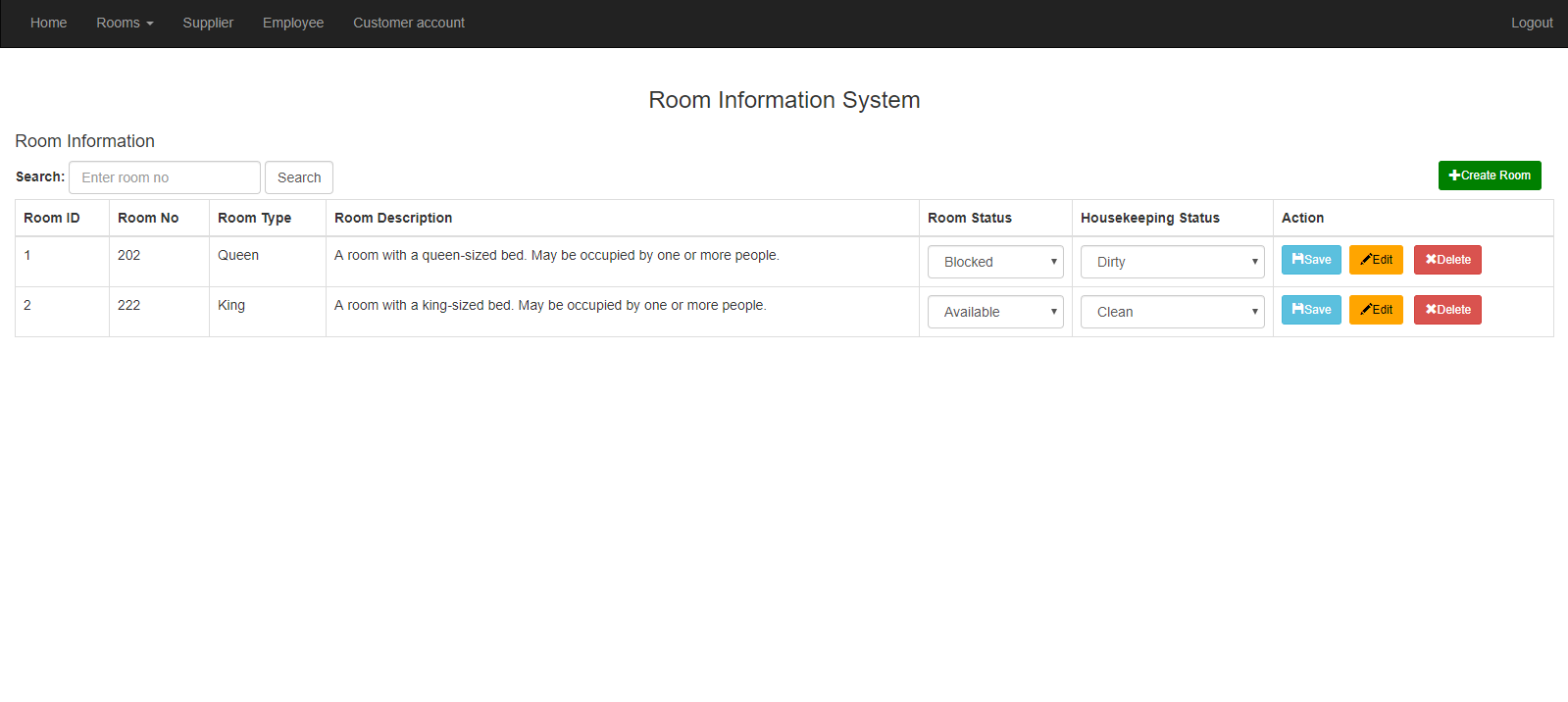
**Figure 2.0** shows the home page of our inventory management system. By pressing the Logout button on the top right it will show a pop up a notification to tell you that you have log out and redirect you to the login screen.

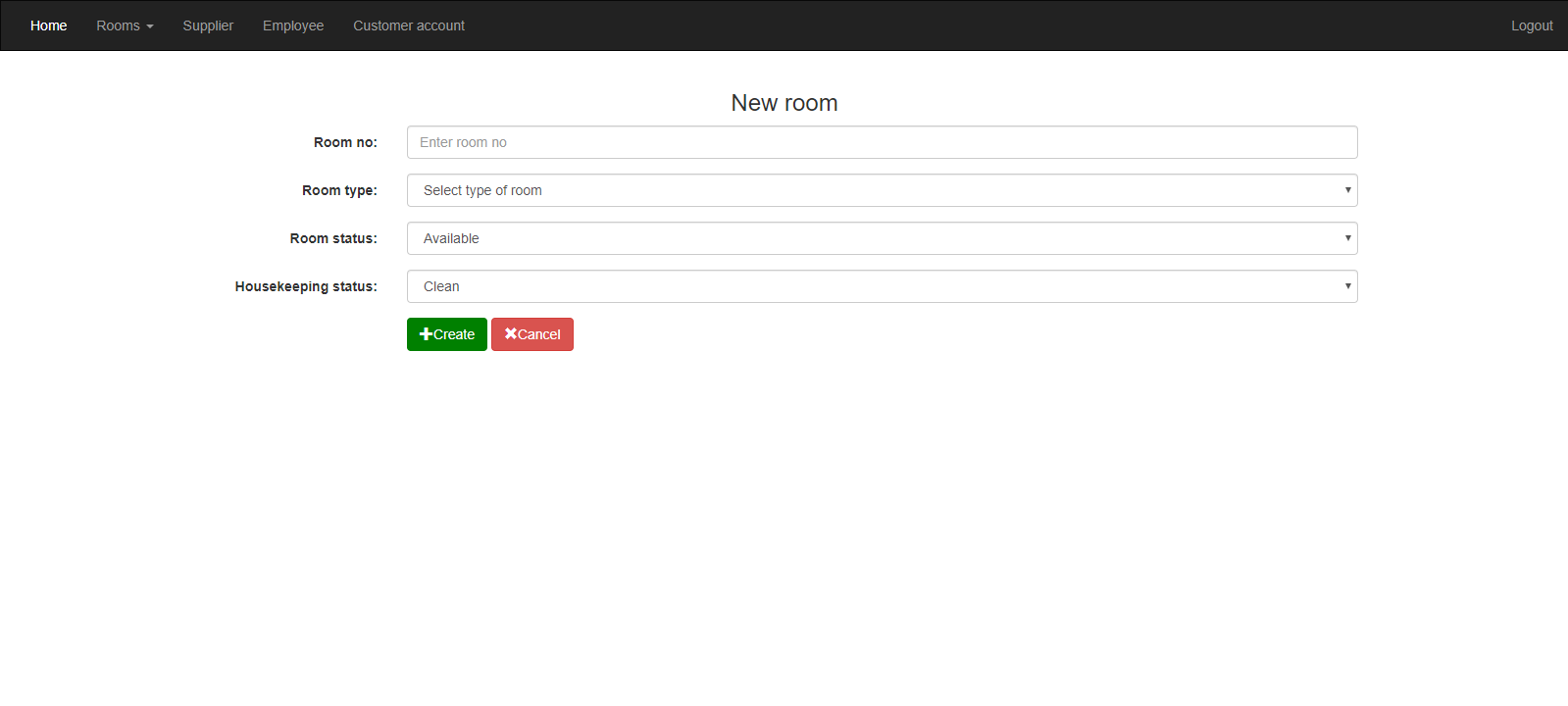
**Figure 3.0 Room information System**

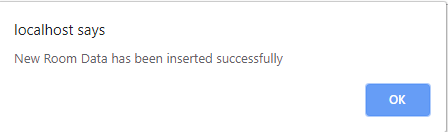


**Figure 3.0** show our room information System, it will display all the existing room and our admin is able to create/edit/or delete room information by using the create room/edit/delete button above. Furthermore, the room status and housekeeping status also able to be managed by clicking the save button and changes the current status of the room.

**Figure 3.1 Insert room data function**

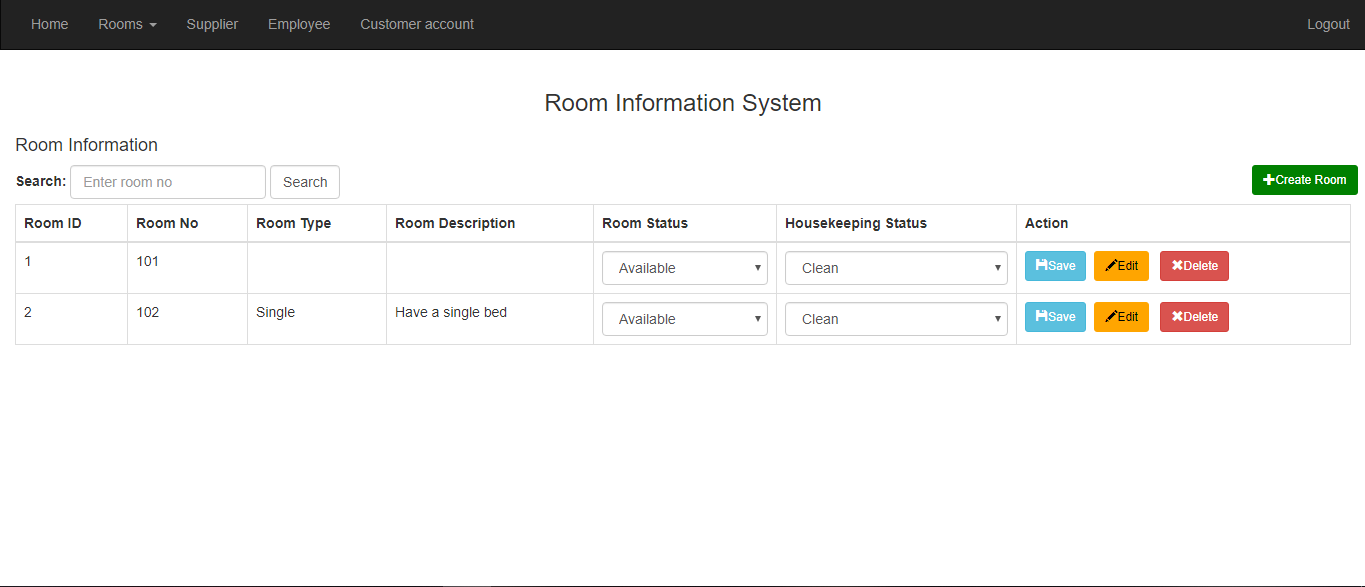


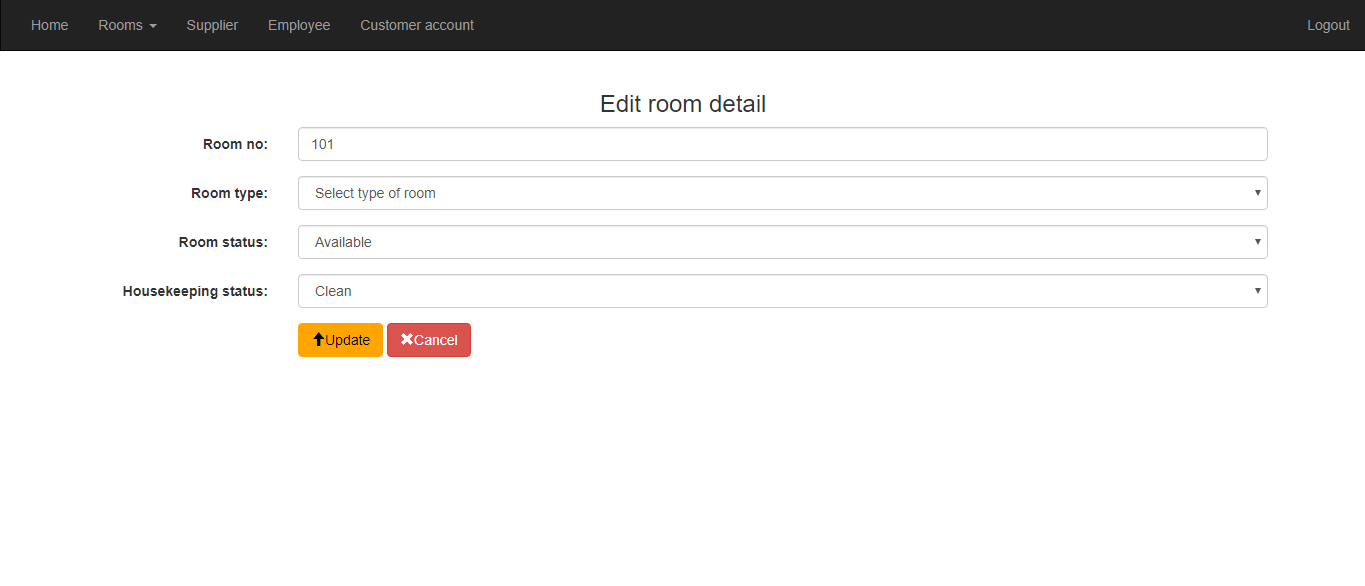




**Figure 3.1** show the form for creating the new room information for the hotel. The options of room type are able to be created in the room type system. If the all inputs are valid, the message above will pop out and room data is created successfully.

**Figure 3.2 Edit room detail function**

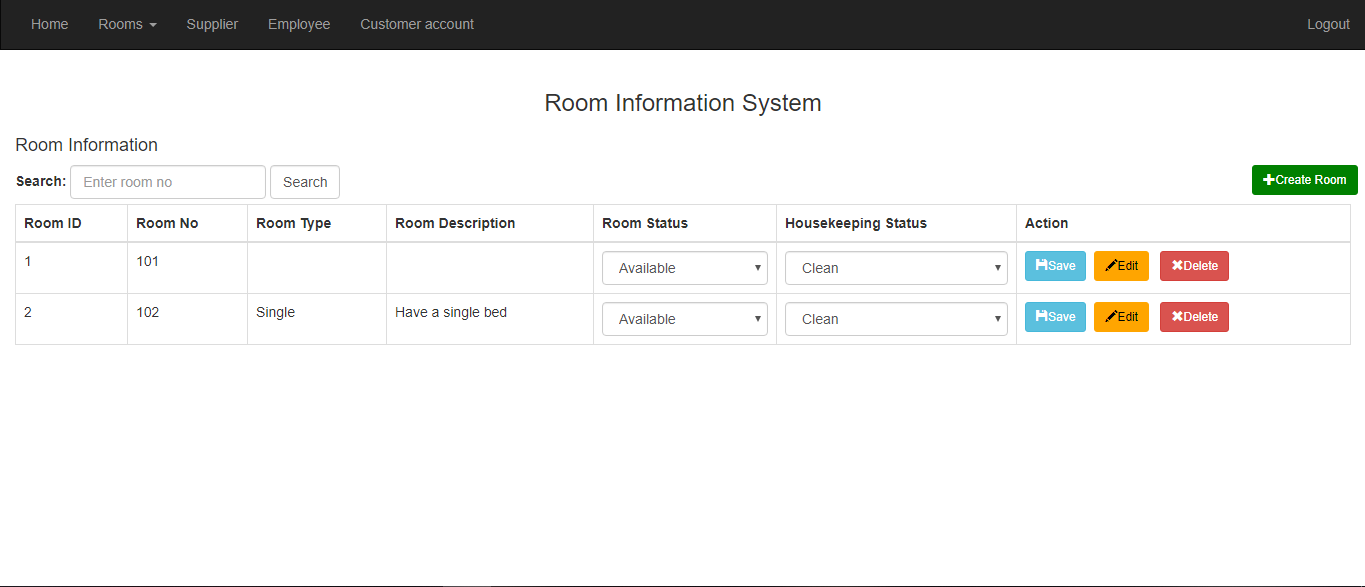


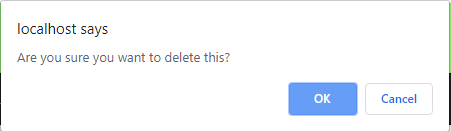


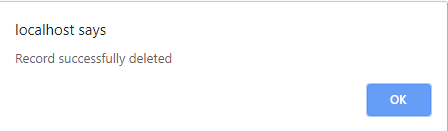


**Figure 3.2** shows our edit room detail function by pressing the edit button and it will take us to the edit room page with all the current information. After we finish editing it will take back to the room information system and the room will be successfully updated with the latest information.

**Figure 3.3 Delete room data function**

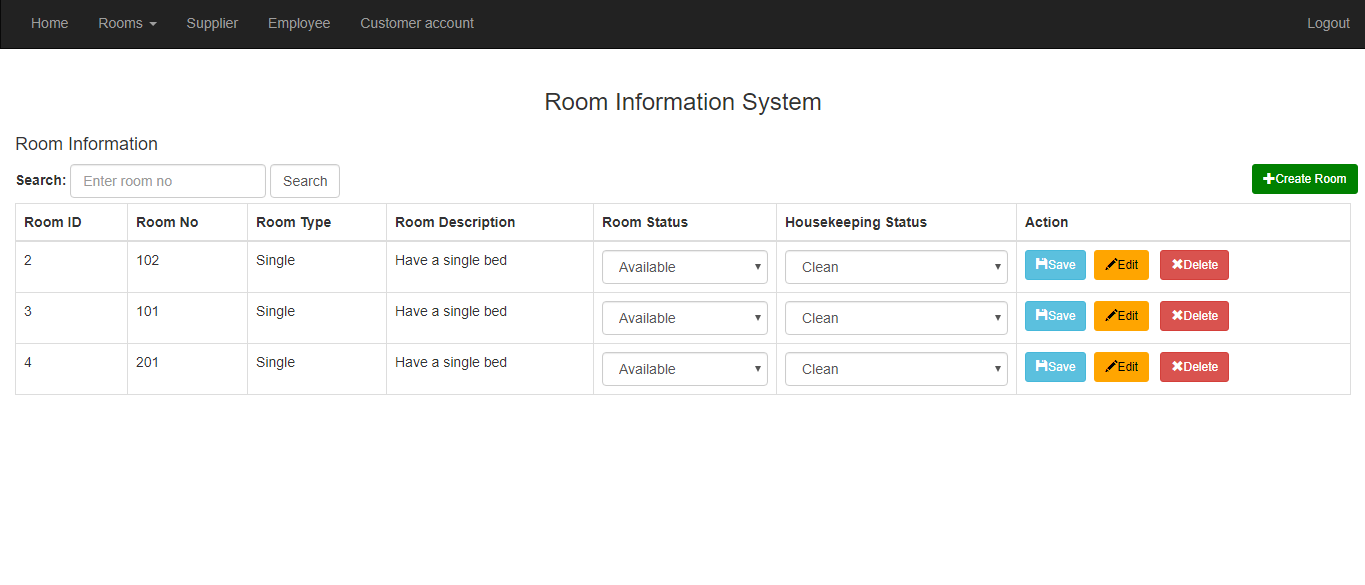


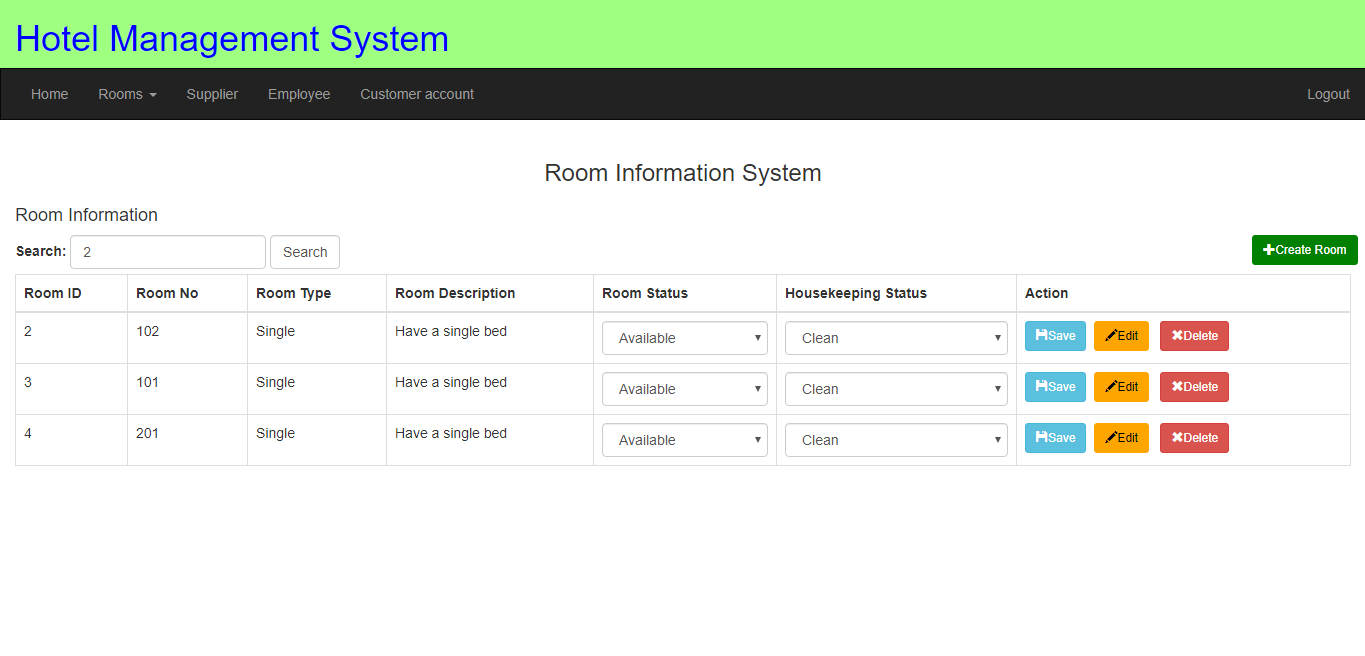


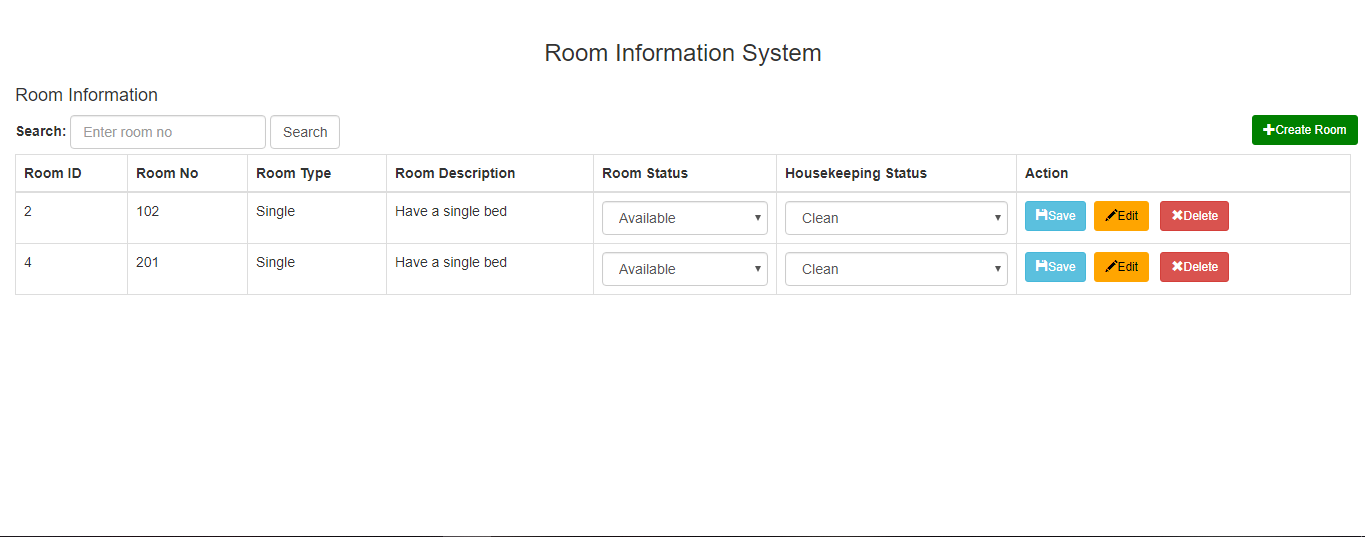


**Figure 3.3** shows our delete room data function by pressing the delete button it will pop up a confirmation to delete that room. After pressing OK, the room data will be deleted and pop out a notification telling us that it has been successfully deleted.

**Figure 3.4 Search room function**

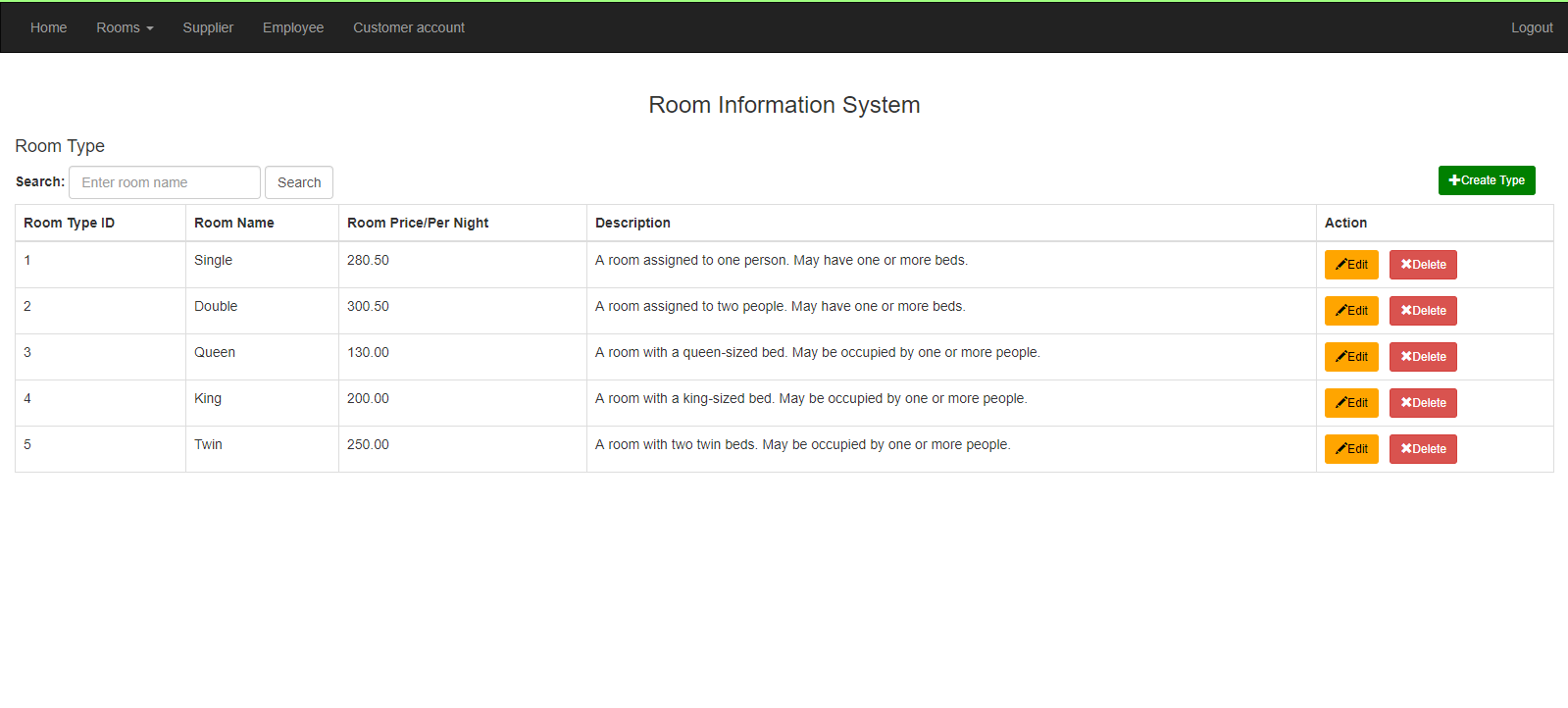






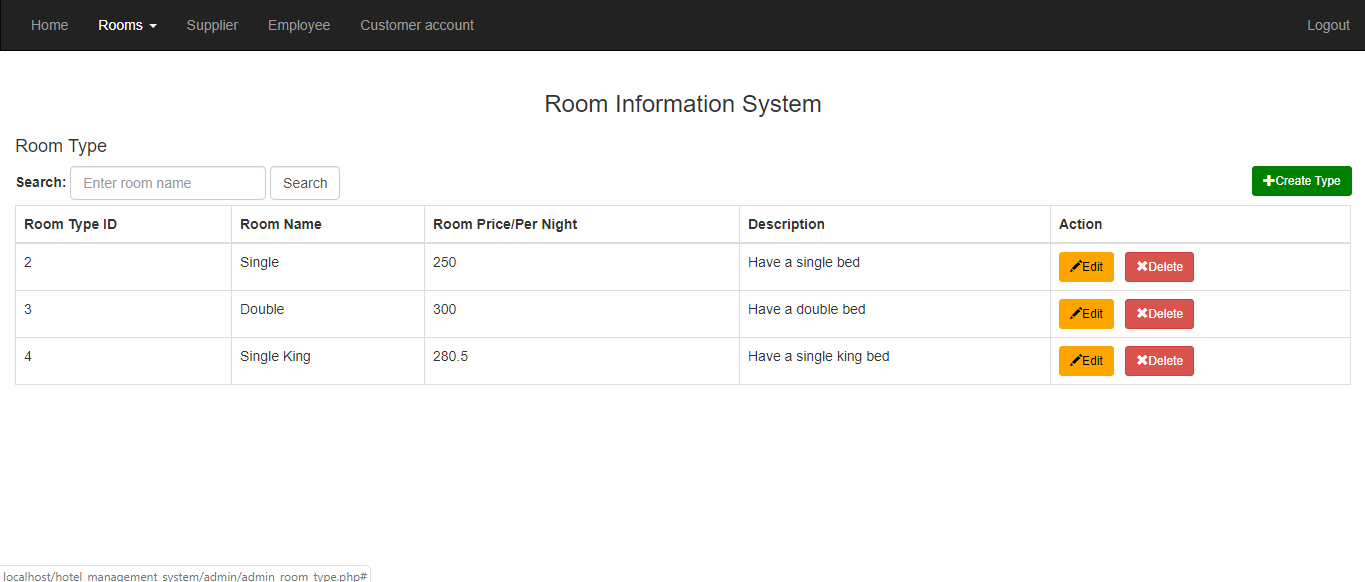
**Figure 3.4** shows our search room function by typing the room number in the provided search box. After the search button is clicked, all room number with the similar name will show up for to admin to view.

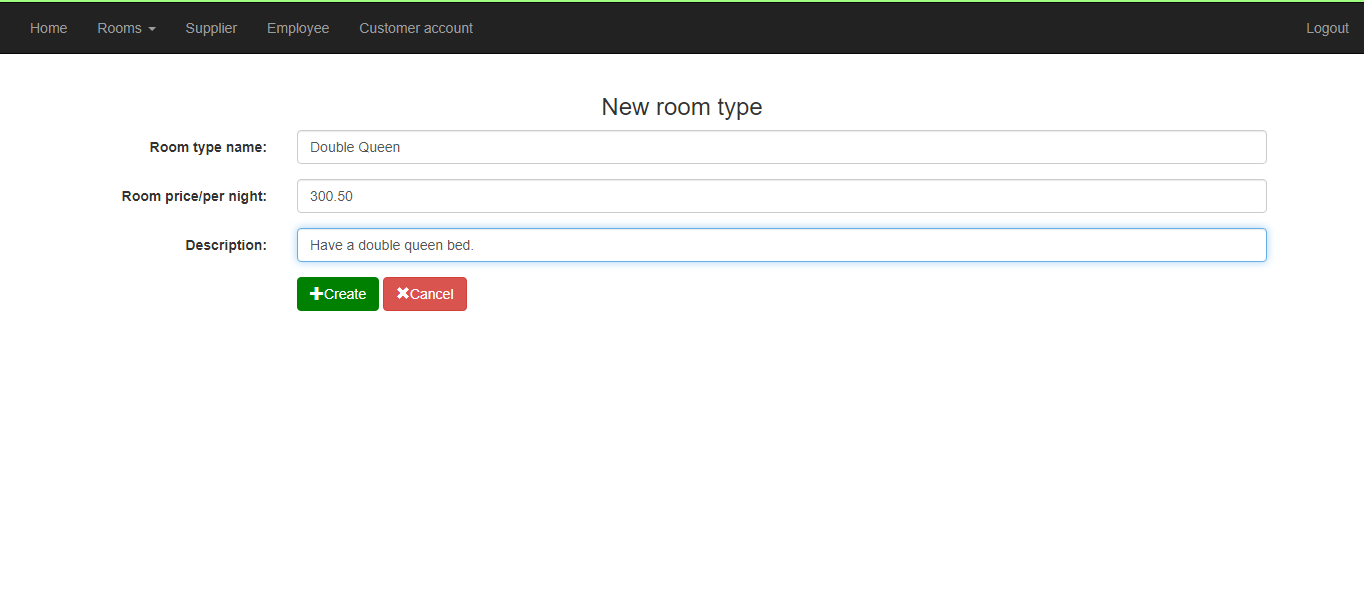
**Figure 3.5 Manage room type function**

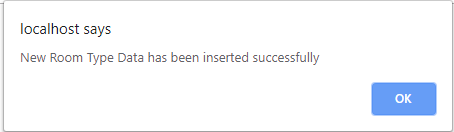


**Figure 3.5** also shows our room type system. The admin can easily to create the new type of room if the hotel provides a new room type.

**Figure 3.6 Insert room type data function**

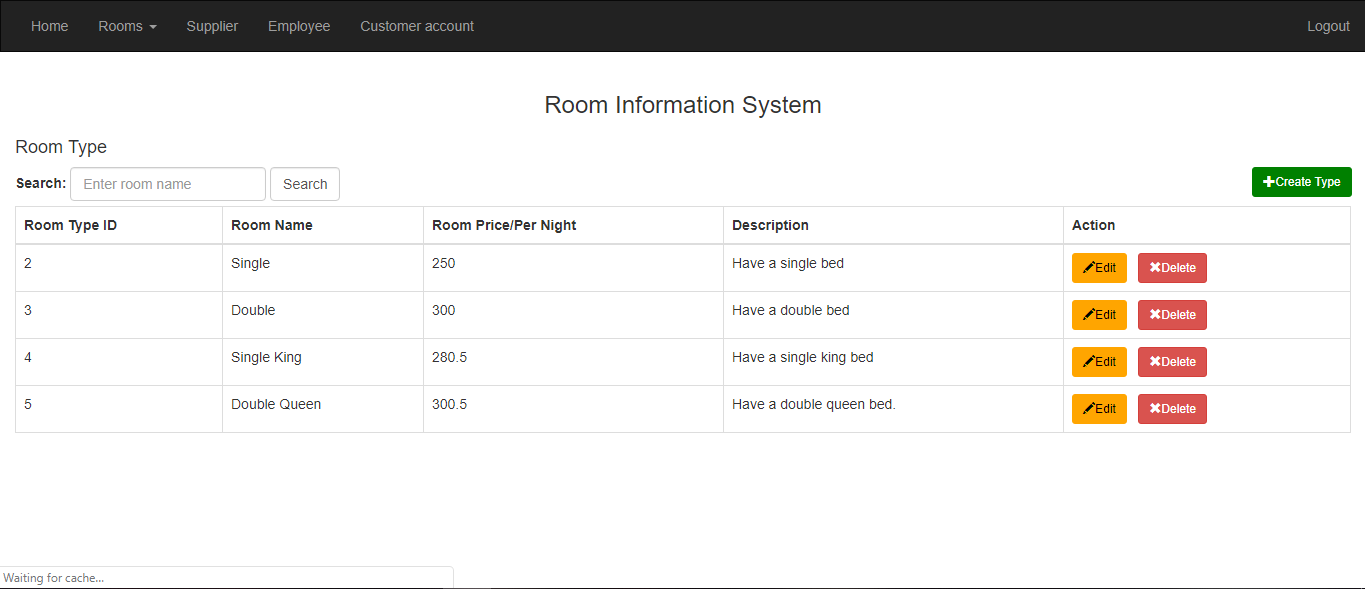


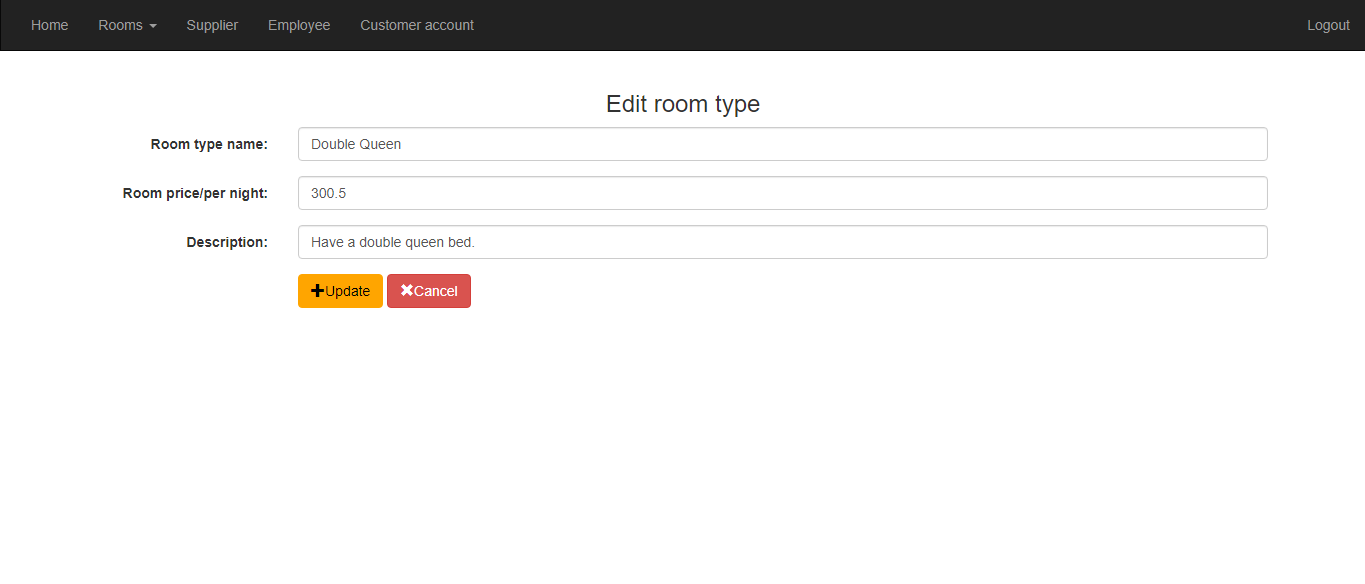




**Figure 3.6** show the form for creating the new room type data for the hotel. The room type is able to be inserted once the hotel provides new room type. If the all inputs are valid, the message above will pop out and room type data is created successfully.

**Figure 3.7 Edit room type detail function**

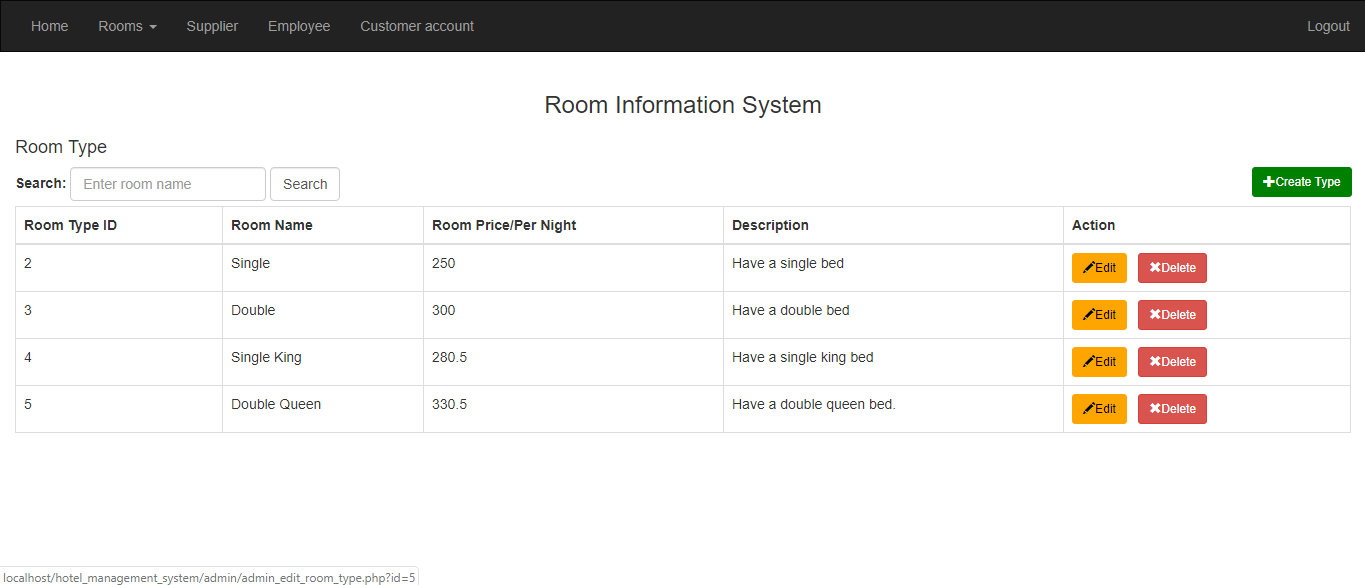


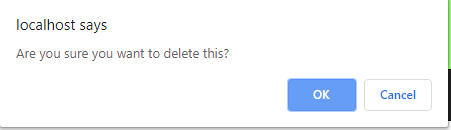


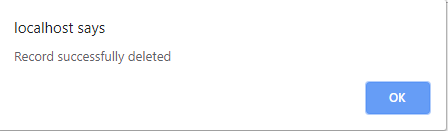


**Figure 3.7** shows our edit room type detail function by pressing the edit button and it will take us to the edit room type detail page with all the current information. After we finish editing it will take back to the room type system and the room type will be successfully updated with the latest information.

**Figure 3.8 Delete room type data function**

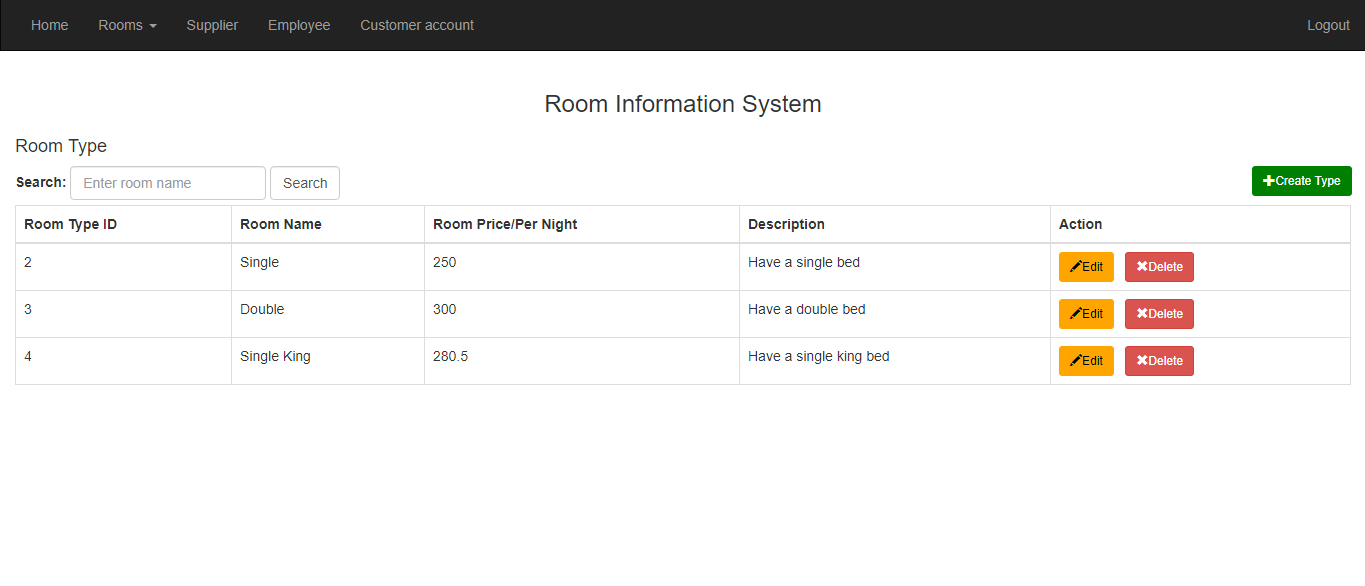
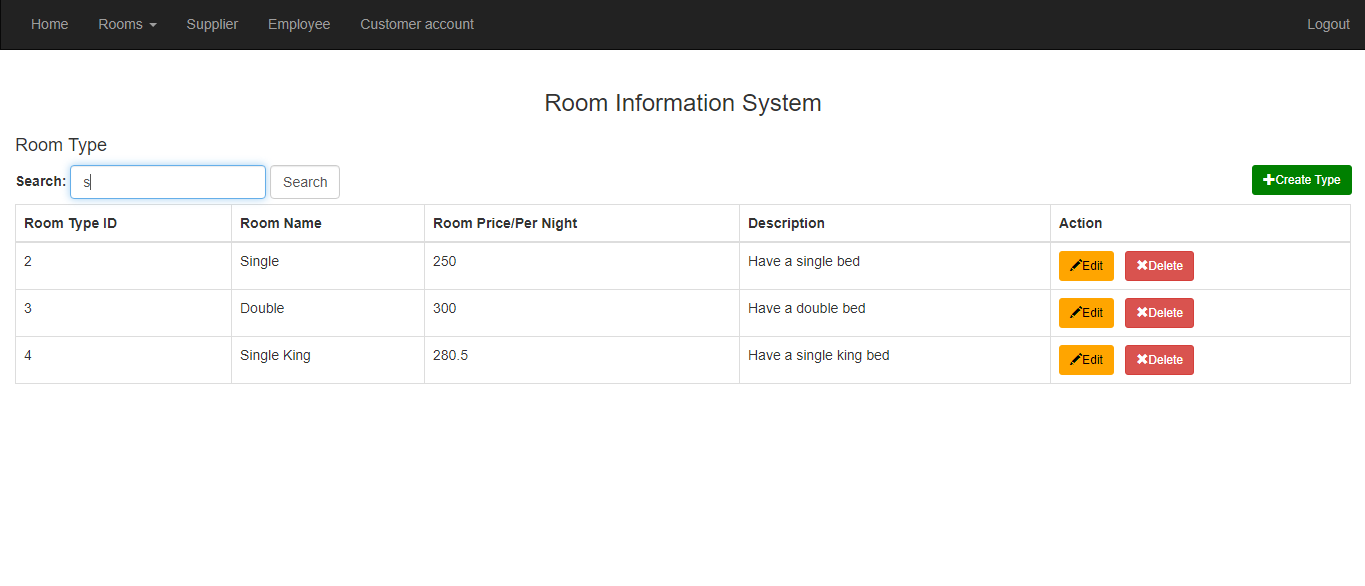


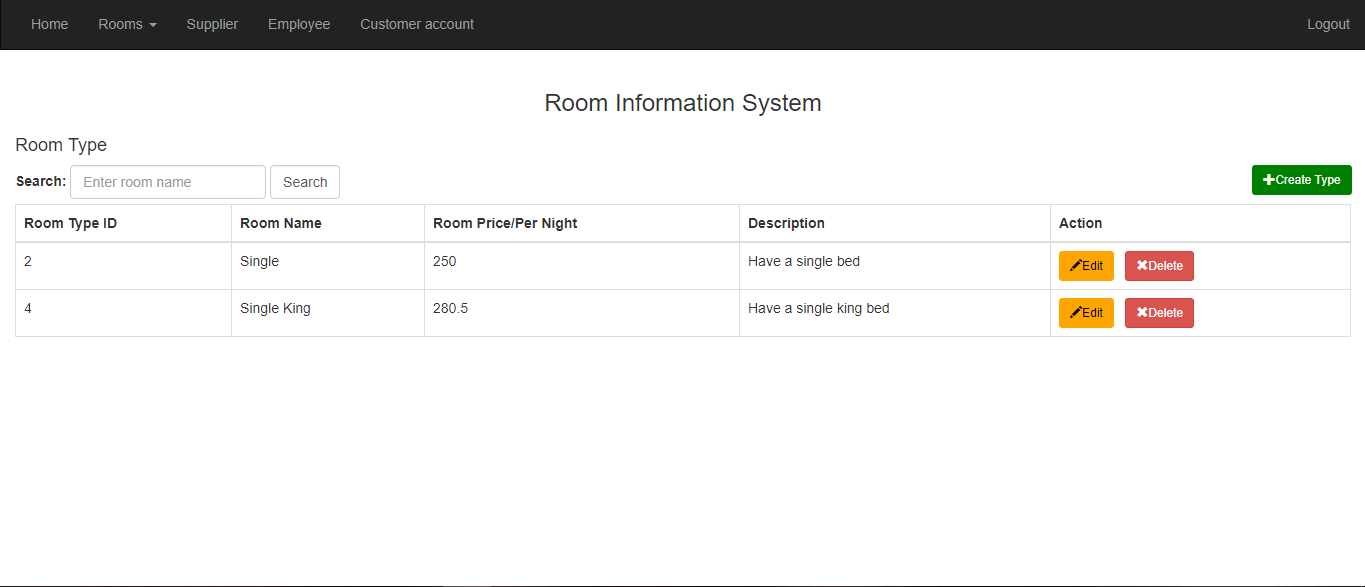




**Figure 3.8** shows our delete room type data function by pressing the delete button it will pop up a confirmation to delete that room type. After pressing OK, the room data will be deleted and pop out a notification telling us that it has been successfully deleted.

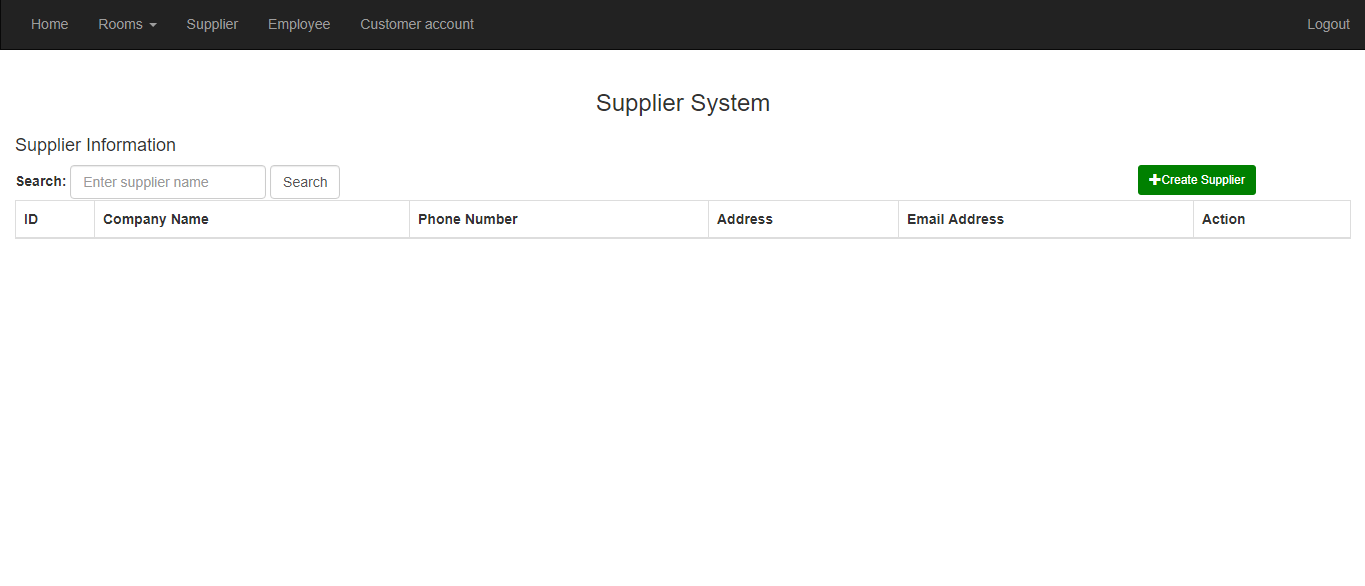
**Figure 3.9 Search room type function**



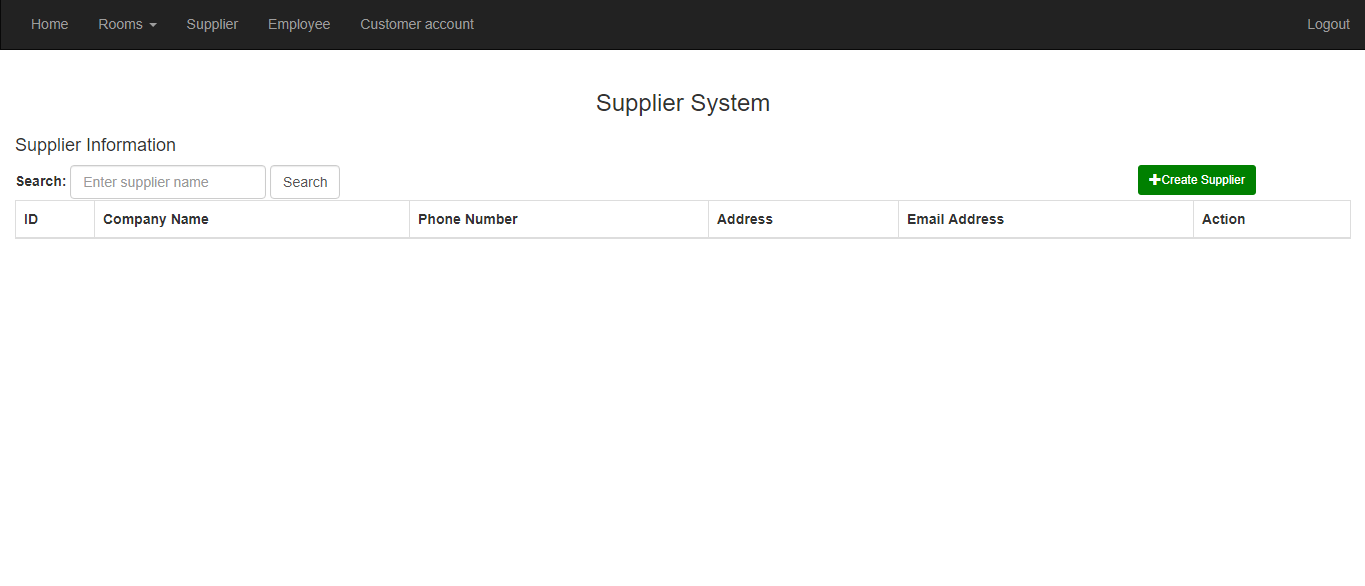
**Figure 3.9** shows our search room type function by typing the room name in the provided search box. After the search button is clicked, all room type names with the similar search keyword will show up for to admin to view.

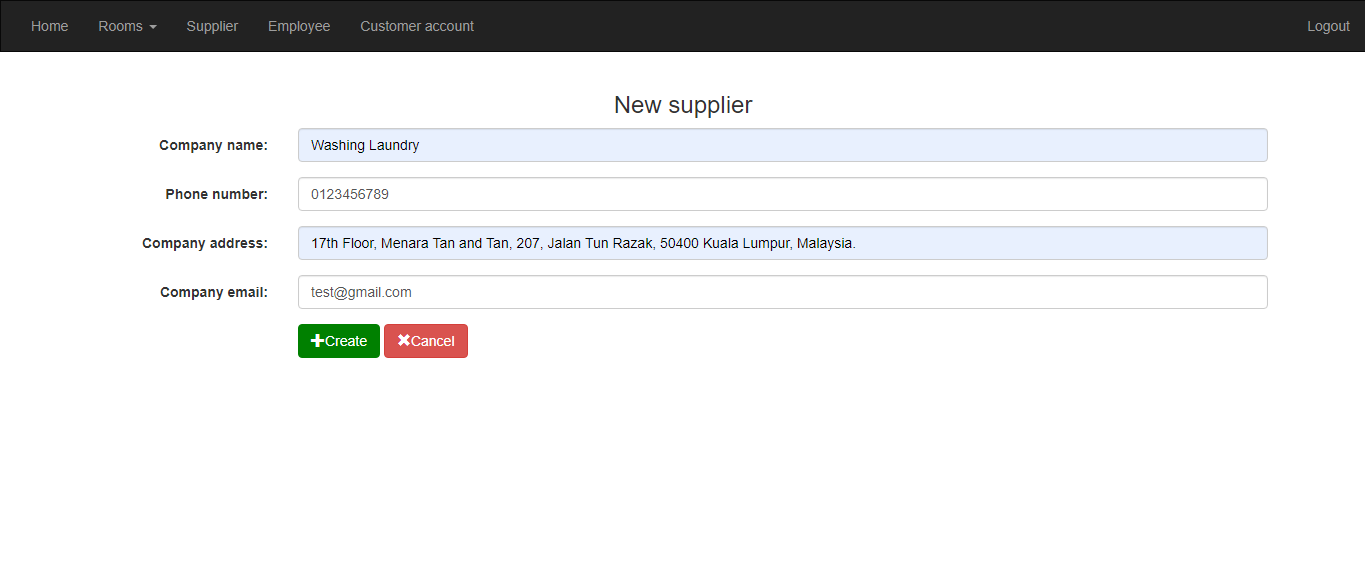
**Figure 4.0 Supplier System**

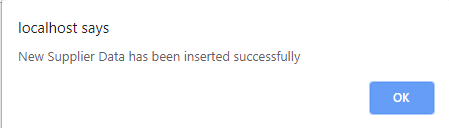


**Figure 4.0** shows our supplier system, it will display all the supplier information and our admin is able to create/edit/or delete supplier information by using the create room/edit/delete button above. Furthermore, supplier information is stored for the ease of contacting the supplier if the purchase order has encountered any problems.

**Figure 4.1 Insert supplier data function**

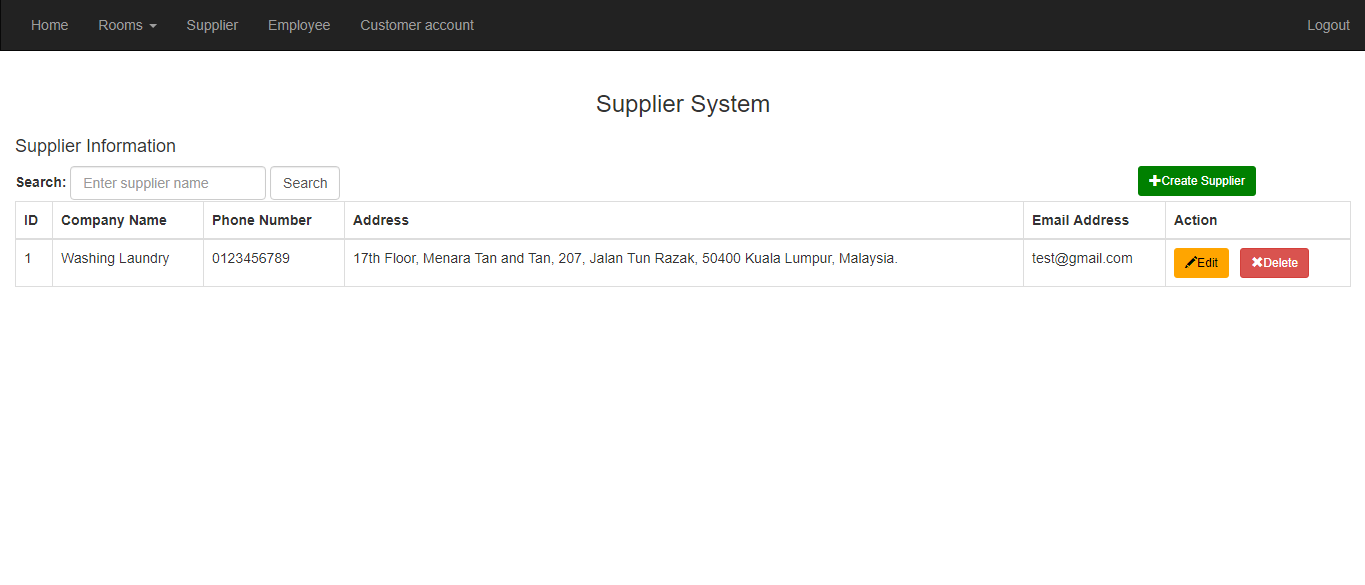


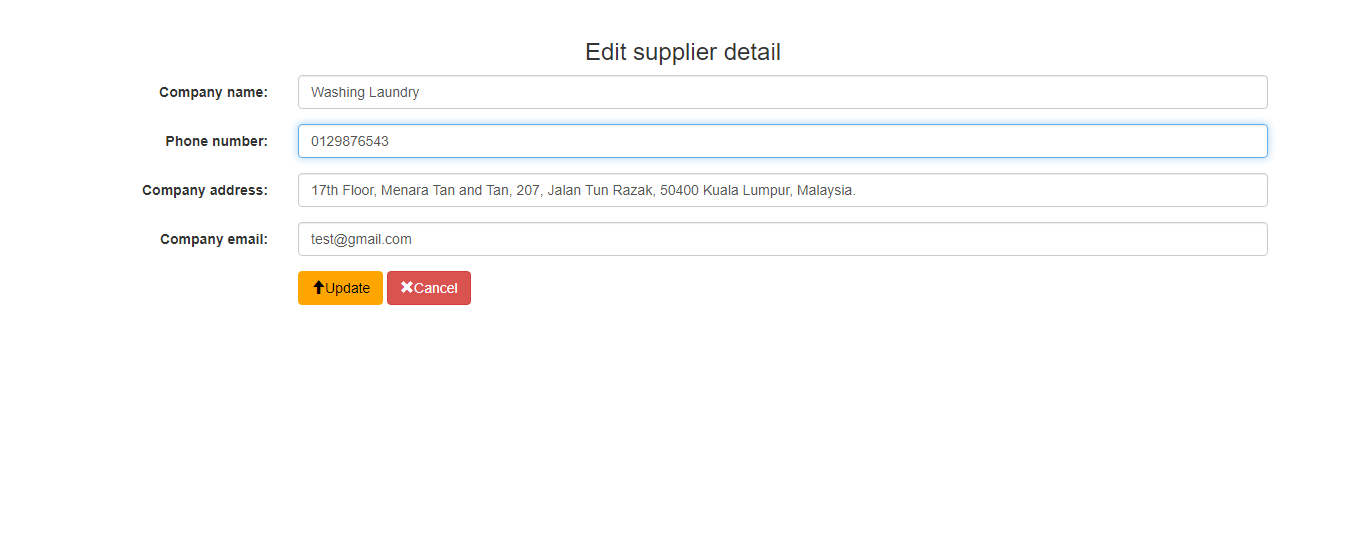


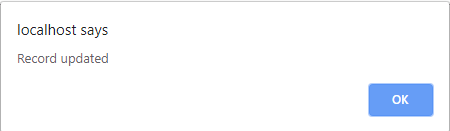


**Figure 4.1** shows the form for creating the new supplier data for the hotel. The supplier information is able to be inserted once the hotel has the new supplier to make the purchase order. If the all inputs are valid, the message above will pop out and room type data is created successfully.

**Figure 4.2 Edit supplier detail function**



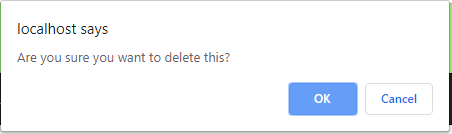


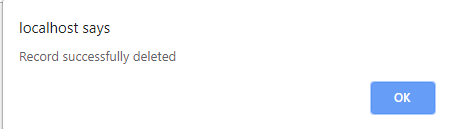


**Figure 4.2** shows our edit supplier detail function by pressing the edit button and it will take us to the edit supplier page with all the current information. After we finish editing it will take back to the supplier system and the supplier will be successfully updated with the latest information.

**Figure 4.3 Delete supplier data function**





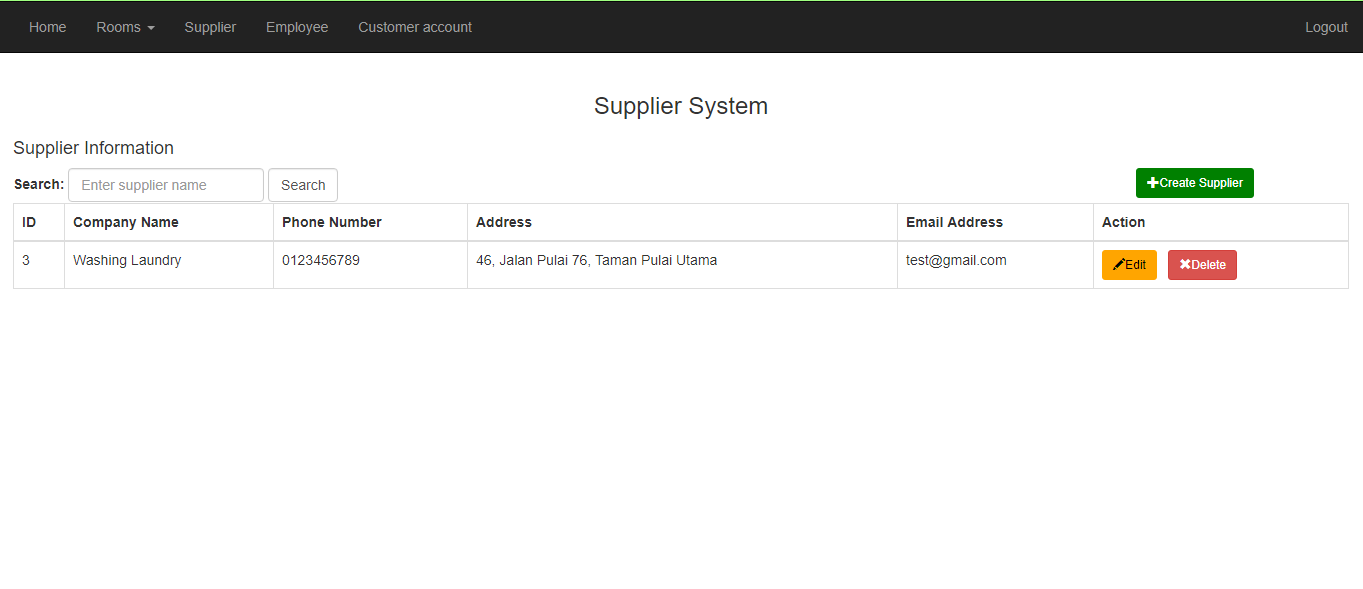


**Figure 4.3** shows our delete supplier data function by pressing the delete button it will pop up a confirmation to delete that supplier. After pressing OK, the supplier data will be deleted and pop out a notification telling us that it has been successfully deleted.

**Figure 4.4 Search supplier function**







**Figure 4.4** shows our search supplier function by typing the supplier name in the provided search box. After the search button is clicked, all suppliers with the similar name will show up for to admin to view.

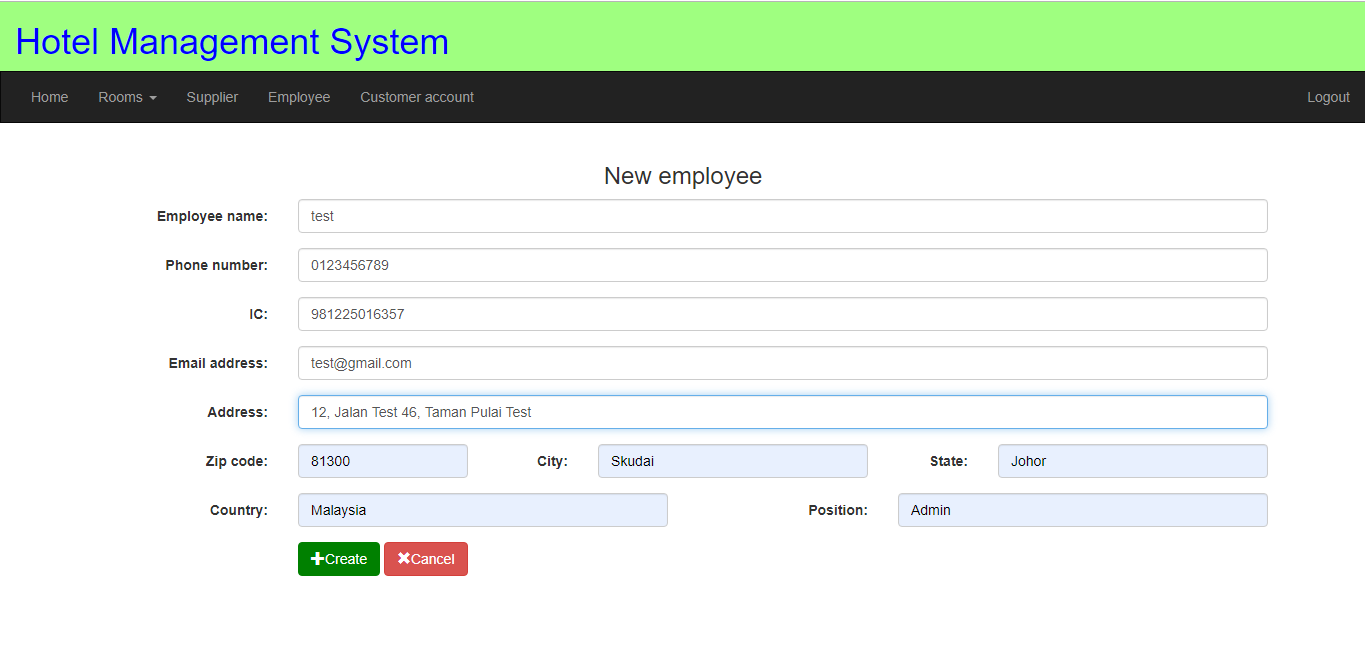
**Figure 5.0 Employee System**

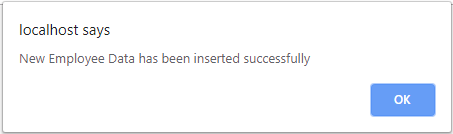


**Figure 5.0** shows our employee system, it will display all the employee information and our admin is able to create/edit/or delete employee information by using the create room/edit/delete button above.

**Figure 5.1 Insert employee data function**

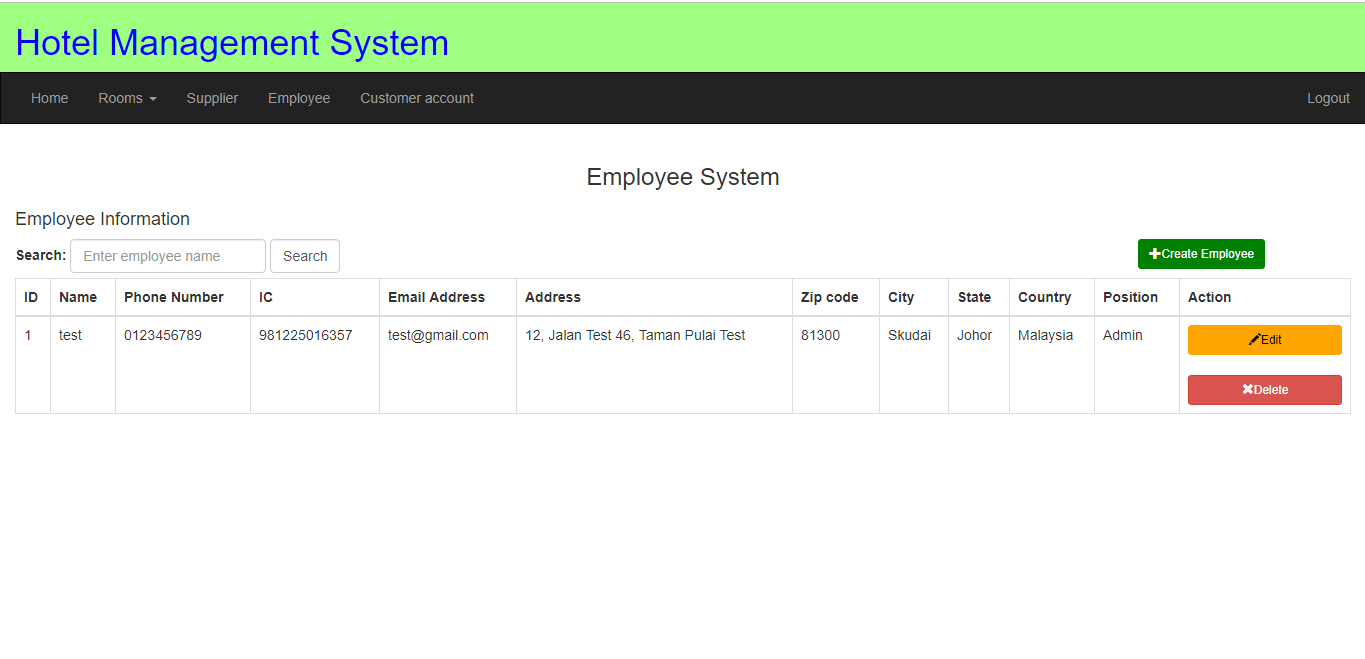


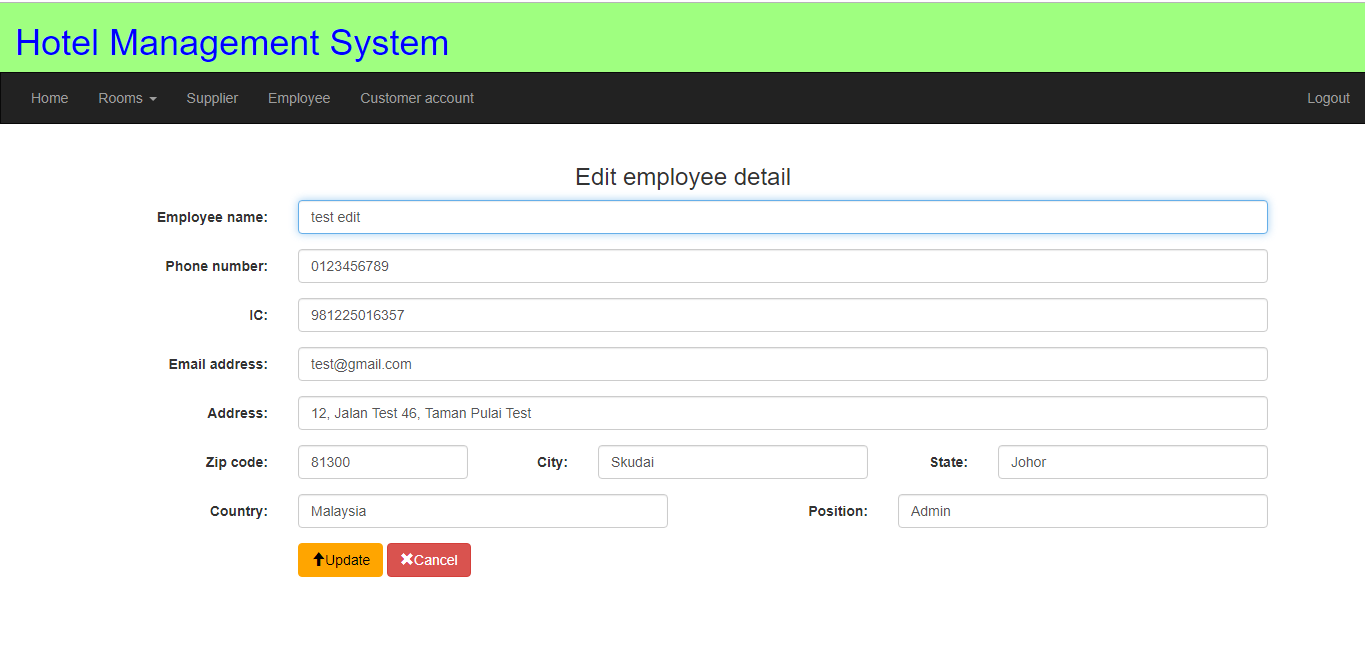


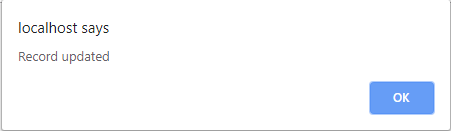


**Figure 5.1** shows the form for creating the new employee data for the employee in the hotel. The employee information is able to be inserted once the new employee comes to work in the hotel. If the all inputs are valid, the message above will pop out and employee data is created successfully.

**Figure 5.2 Edit employee detail function**



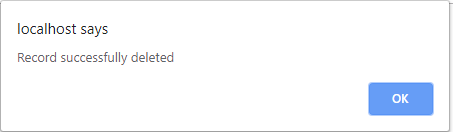
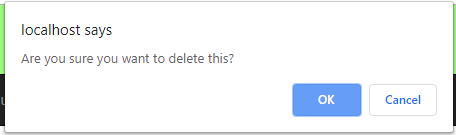




**Figure 5.2** shows our edit employee detail function by pressing the edit button and it will take us to the edit employee page with the employee’s current information. After we finish editing it will take back to the employee system and the employee will be successfully updated with the latest information.

**Figure 5.3 Delete employee data function**

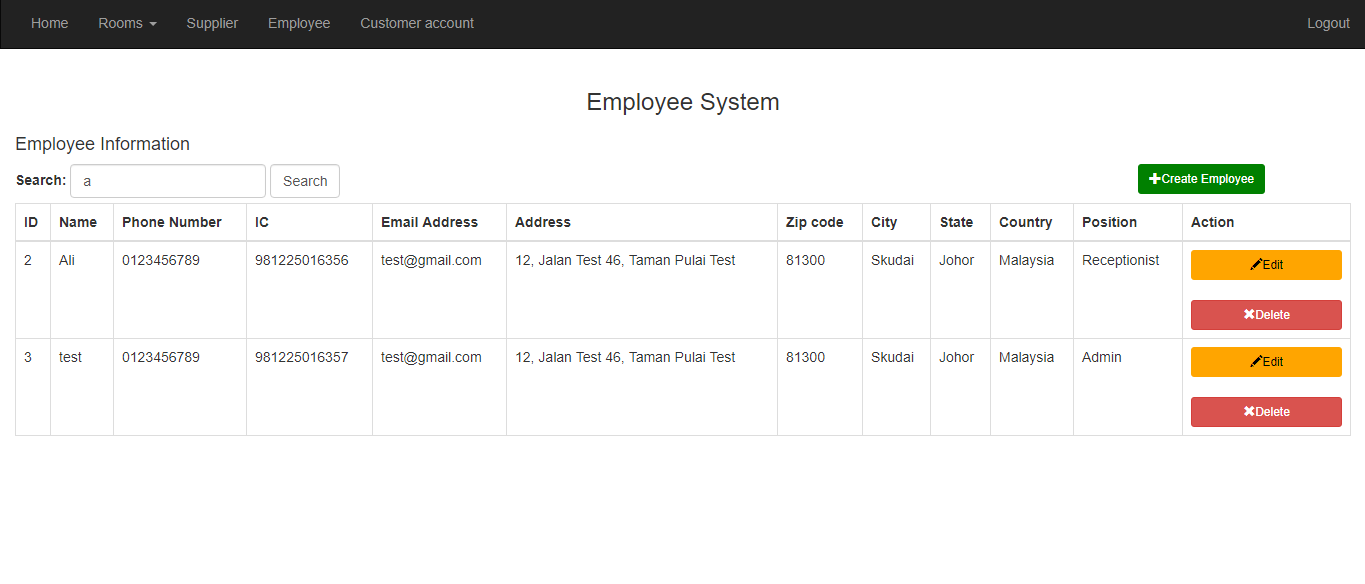


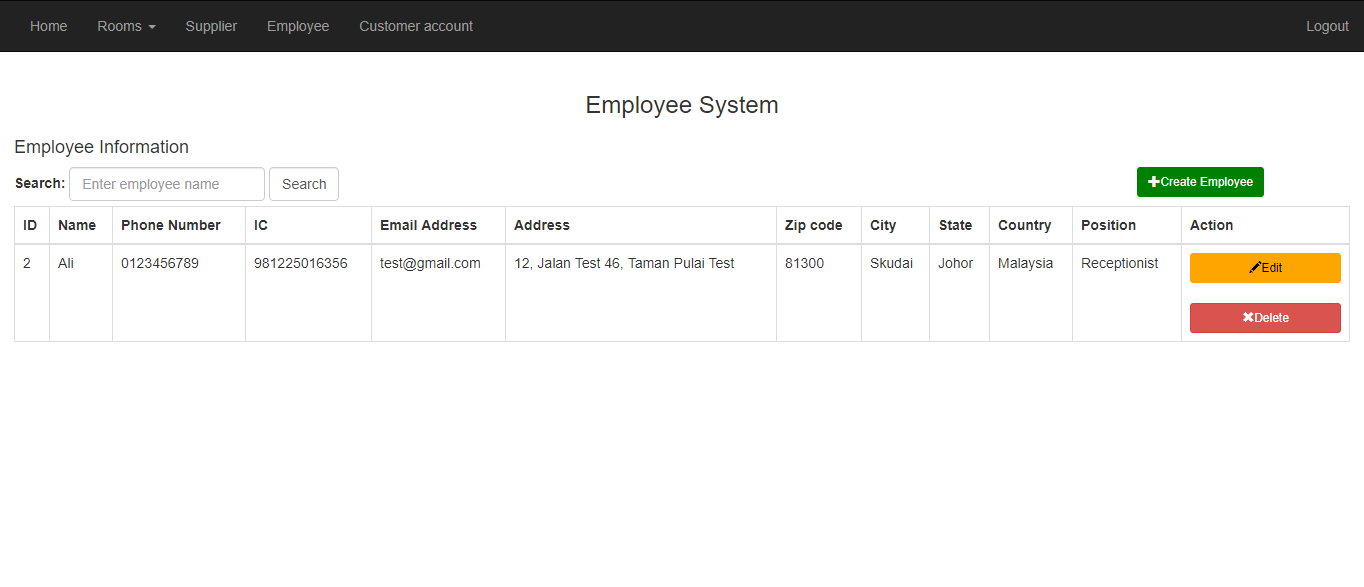


**Figure 5.3** shows our delete employee data function by pressing the delete button it will pop up a confirmation to delete that supplier. After pressing OK, the employee data will be deleted and pop out a notification telling us that it has been successfully deleted.

**Figure 5.4 Search employee function**

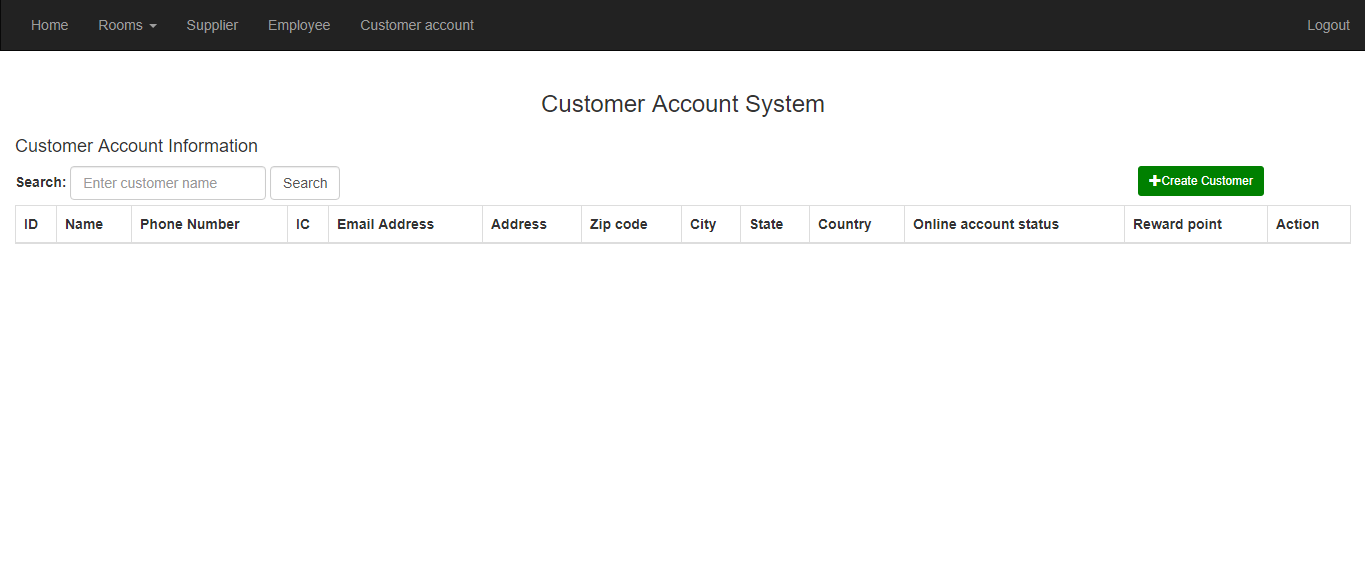






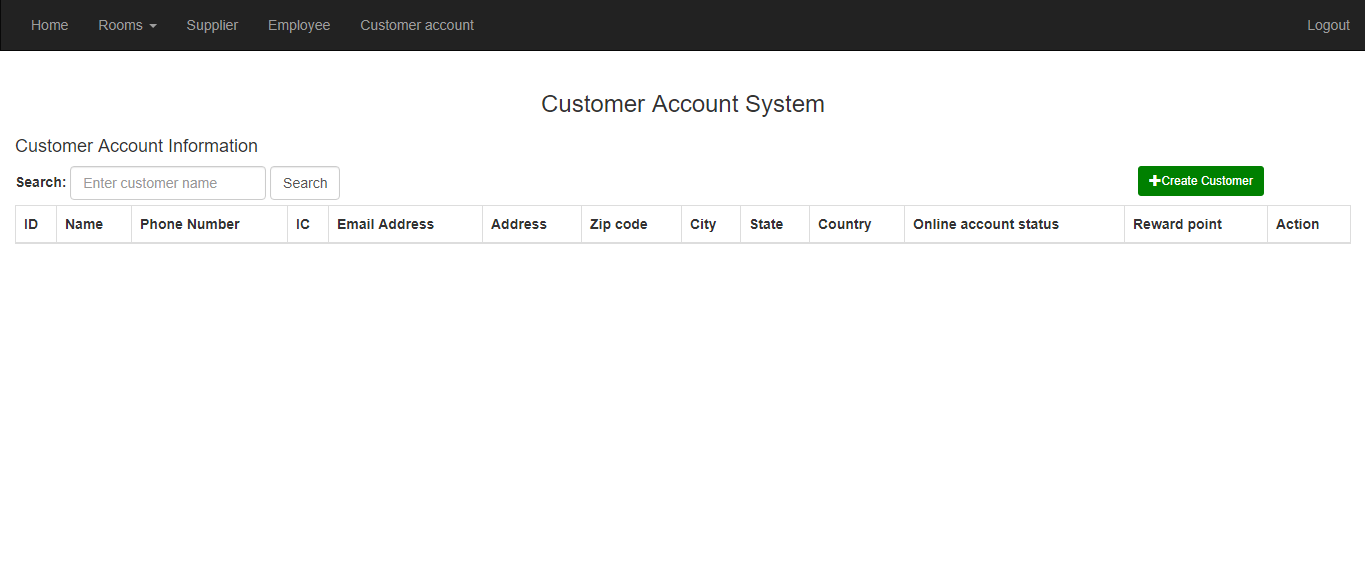
**Figure 5.4** shows our search employee function by typing the employee name in the provided search box. After the search button is clicked, all employees with the similar name will show up for to admin to view.

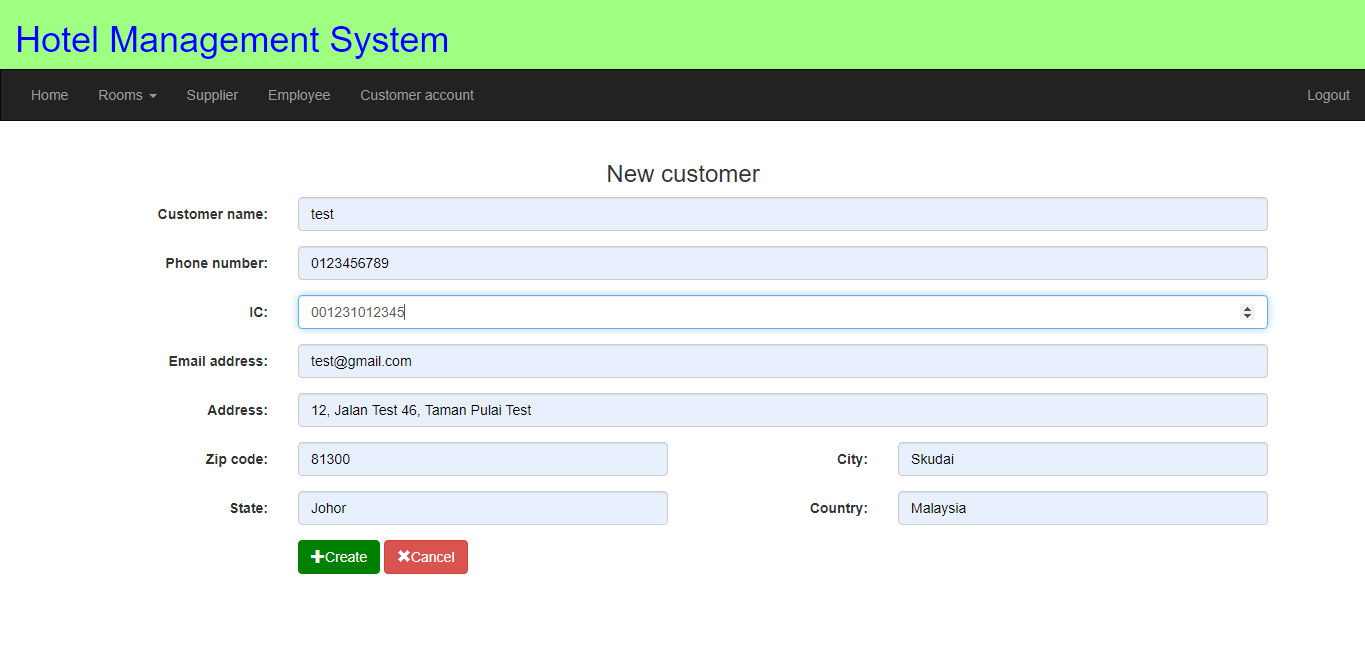
**Figure 6.0 Customer account system**

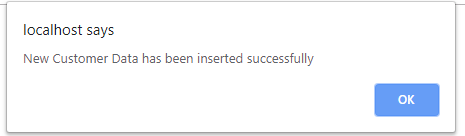


**Figure 6.0** shows our customer account system, it will display all the customer information when the customer makes the reservation for the room and our admin is able to create/edit/or delete employee information by using the create room/edit/delete button above.

**Figure 6.1 Insert customer account data function**

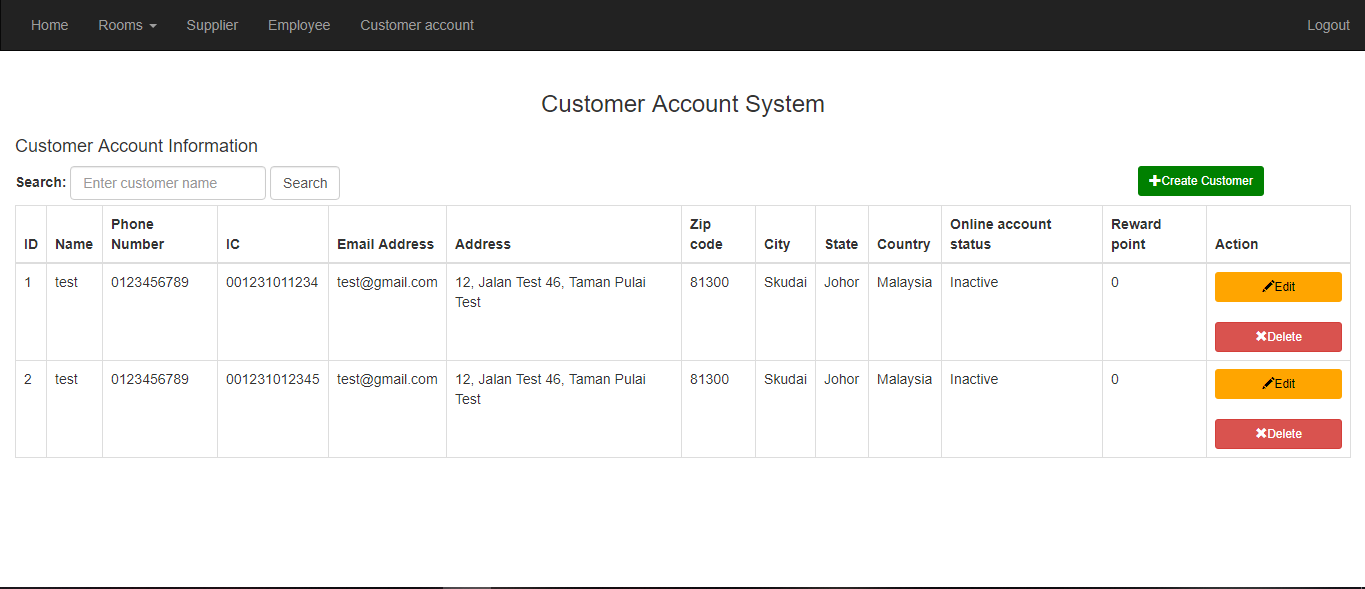


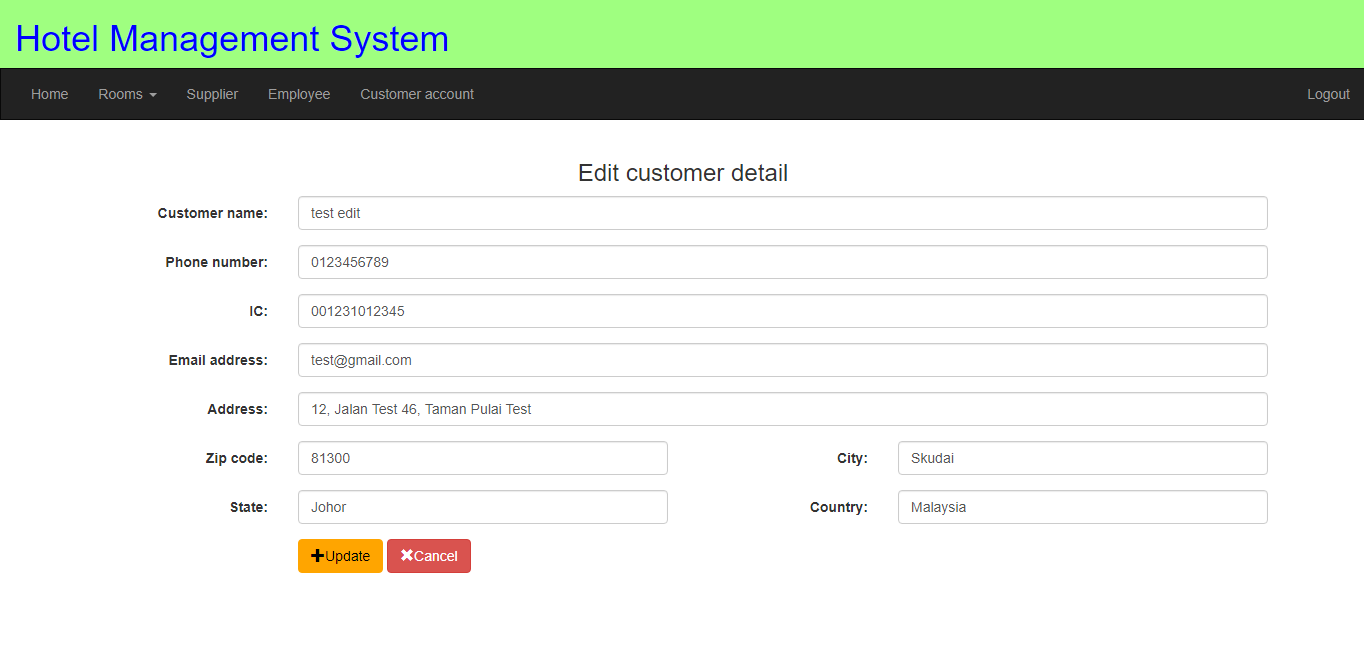




**Figure 6.1** shows the form for creating the new customer account data for the customer who has made the reservation in the hotel. The customer account information is able to be inserted if the customer is check in to the hotel first time. If the all inputs are valid, the message above will pop out and customer account data is created successfully.

**Figure 6.2 Edit customer account detail function**

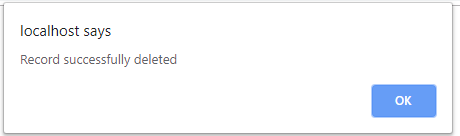
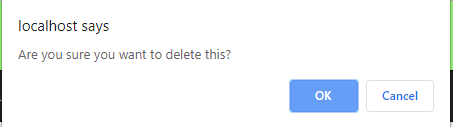






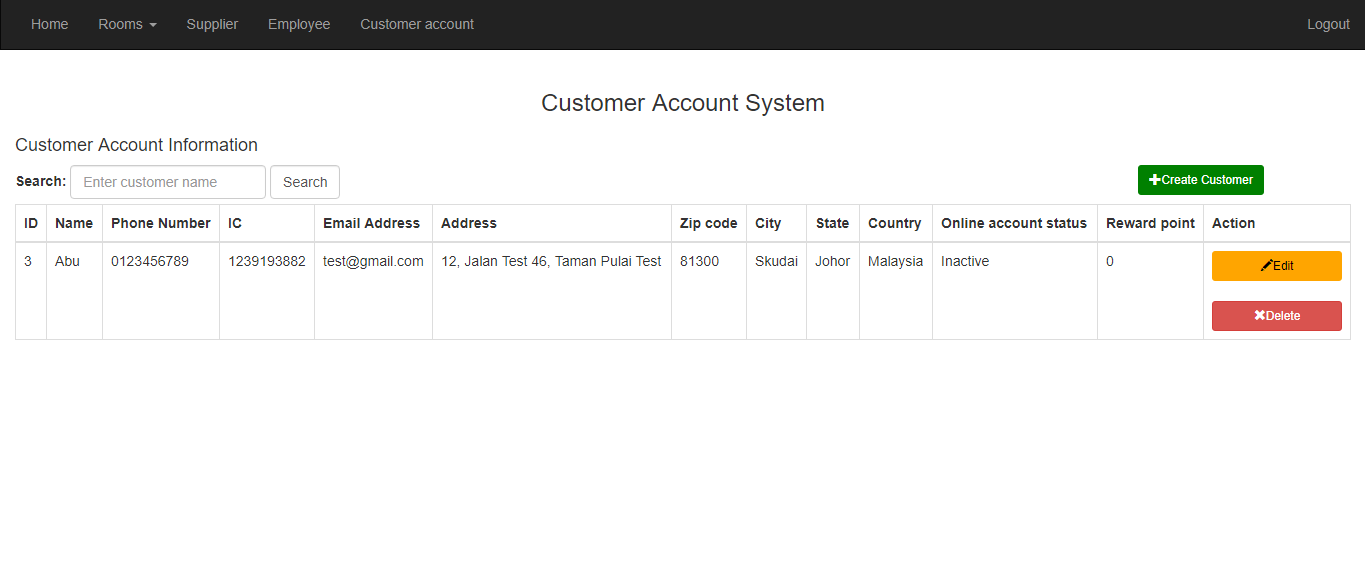
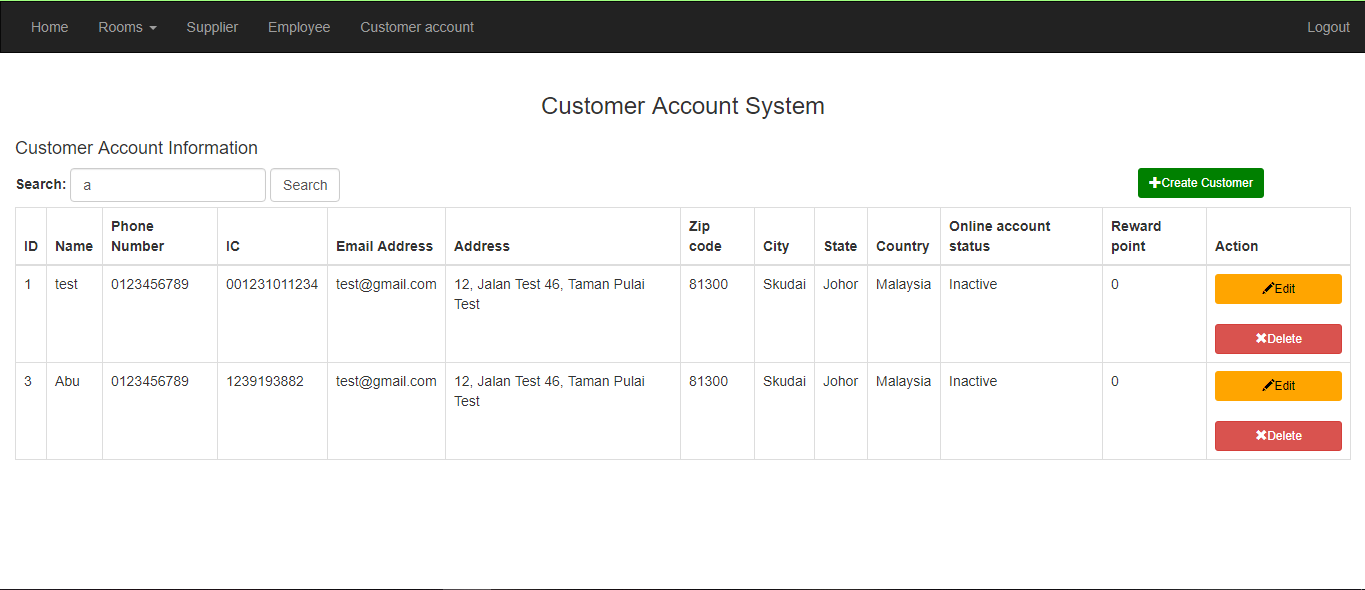
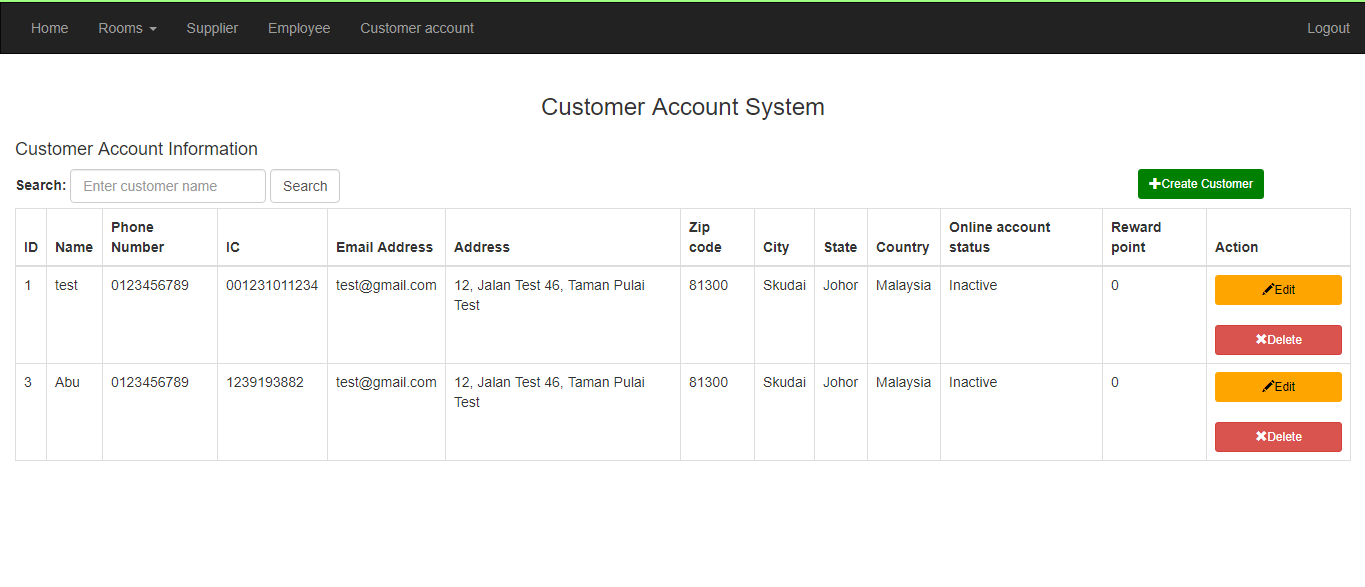
**Figure 6.2** shows our edit customer account detail function by pressing the edit button and it will take us to the edit customer account page with the latest information of the customer account. After we finish editing it will take back to the customer account system and the customer account data will be successfully updated with the latest information.

**Figure 6.3 Delete customer account data function**



**Figure 6.3** shows our delete customer account data function by pressing the delete button it will pop up a confirmation to delete that customer account. After pressing OK, the customer account data will be deleted and pop out a notification telling us that it has been successfully deleted.

**Figure 6.4 Search customer account function**



**Figure 6.4** shows our search customer account function by typing the customer name in the provided search box. After the search button is clicked, all customers with the similar name will show up for to admin to view.

# APPENDIX C – Project Diaries

Project Diary & Supervisor Feedback-1

|  |  |
| --- | --- |
| Student Name: | Ang Sheng Yang\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| Supervisor: | Mr So Yong Quay \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| Meeting Date: | 11/4/2019\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

**Issues identified in previous meeting (including personal development goals):**

This is the first project meeting, so no issue has been identified.

**Feedback received in previous meeting:**

This is the first project meeting, so no feedback has been identified.

**Action taken on feedback:**

This is the first project meeting, so no action has been taken.

**Matters to discuss:**

1. **Project topic**

* Firstly, I have several topics that I have familiar and want to use as my final year project topic. Below are the topics of final year project that I have figured:
  + Pet Café Management System
  + Hotel Management System
  + Cloth Shop Management System
* So, I decide to take the Pet Café Management System as my final year project title. Furthermore, I also prepare some materials that related to the system and discuss with Mr So about my ideas in the system.

1. **Mr So opinion**

* Mr So said that the concept of Pet Café Management System is similar to the Restaurant Management System that already taken by the other student. So, this system is not suitable as my project title.
* Secondly, Mr So said that the Cloth Shop Management System is too small as my project title. So, this system is not also suitable as my project title.
* Thirdly, Mr So said that the Hotel Management System is a quite big project title. The main functions of the system are able to manage the reservation by the customer, the management of housekeeping, the management of the inventory, and also the management of supplier.

1. **Conclusion**

* I decide to take the hotel management system as my project title and narrow down the hotel into the motel or inn.

**References Consulted:**

No References Consulted.

|  |  |
| --- | --- |
| **Student’s Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | **Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_** |
| **Supervisor’s Signature:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | **Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_** |

Project Diary & Supervisor Feedback - 2

|  |  |
| --- | --- |
| Student Name: | Ang Sheng Yang |
| Supervisor: | Mr So Yong Quay |
| Meeting Date: | 18/4/2019 |

**Issues identified in previous meeting (including personal development goals):**

1. **Collect the information of hotel management system**

* I do not have the insufficient information for the hotel business context.

**Feedback received in previous meeting:**

1. **Gather more information of the hotel management system from the internet**

* Gather the information from the exist hotel booking website.

**Action taken on feedback:**

1. **Gather the information of the hotel management system from the internet**

* Find some of the hotel booking website and analyse the backend process and frontend process of the hotel company.
* Write the abstract of the system and ready to discuss with Mr So in the next coming meeting.

**Matters to discuss:**

1. **Discuss the overall project scope**

* I have prepared the abstract of the system and ready to discuss with Mr So about my main functions of the system.

1. **Mr So Opinion**

* Firstly, Mr So advises me that I should think the backend system first before think about the frontend system.
* Mr So advises me that the hotel should consider the price management for the peak season, normal season, and close to the peak season.
* Furthermore, Mr So also said that the hotel management system should manage the promotion for the discount within the period.
* Moreover, the reward for the membership of the customer can be used from the online website.

1. **Conclusion**

* I have the overall concept of the hotel management system. So, I can complete the stage 1 report and submit the report on the next meeting.

**References Consulted:**

No References Consulted.

|  |  |
| --- | --- |
| **Student’s Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | **Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_** |
| **Supervisor’s Signature:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | **Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_** |

Project Diary & Supervisor Feedback-3

|  |  |
| --- | --- |
| Student Name: | Ang Sheng Yang |
| Supervisor: | Mr So Yong Quay |
| Meeting Date: | 25/4/2019 |

**Issues identified in previous meeting (including personal development goals):**

1. **Insufficient of the information of hotel management system**

* After I do some research about the online hotel system, I realised that the hotel should have the promotion management to manage their promotions in the certain period. Furthermore, the hotel consists of different type of rooms and the different rooms have the different rate of price.

**Feedback received in previous meeting:**

1. **Gather more information of the online hotel booking system from the internet**

* Gather the information from the exist hotel booking website.

**Action taken on feedback:**

1. **Do some correction of stage 1 report**

* Based on the more understanding of the online hotel booking system, I need to do some corrections about the scope of the system such as adding the system that should be included in the hotel management system.
* I also have to complete the stage 1 report and submit to the Mr So to review the stage 1 report.

**Matters to discuss:**

1. **Hotel Housekeeping**

* Mr So advises about the housekeeping system is able to assign the housekeepers and schedule the timetable for the housekeepers. Furthermore, Mr So also advises to do a template for recording the daily use of inventory by the housekeepers. The template should provide the housekeeper to prepare how many stocks have been taken from the inventory and how many should return back to the inventory. Moreover, Mr So also advises the housekeeper should check whether the drinks in the mini bar are used or not in each of the hotel room before the customer check out.

1. **Conclusion**

* I will start my project stage 2 which is design the data flow diagram and discuss to the Mr So in the next meeting.

**References Consulted:**

No References Consulted.

|  |  |
| --- | --- |
| **Student’s Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | **Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_** |
| **Supervisor’s Signature:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | **Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_** |

Project Diary & Supervisor Feedback-4

|  |  |
| --- | --- |
| Student Name: | Ang Sheng Yang |
| Supervisor: | Mr So Yong Quay |
| Meeting Date: | 2/5/2019 |

**Issues identified in previous meeting (including personal development goals):**

1. **The dataflow diagram**

* I am doing the context level diagram and the level 0 diagram of the system. I am not too sure about the correctness of the diagrams. Hence, I will submit to Mr So to review my diagrams.

**Feedback received in previous meeting:**

* No feedback has been identified.

**Action taken on feedback:**

* No action has been taken.

**Matters to discuss:**

1. **Stage 1 report**

* After Mr So has reviewed my stage 1 report, he said that my cover of report has wrong title and wrong course code. Furthermore, Mr So also helps me to check the grammar of the report and fount that there are many grammar errors in the report. Moreover, the references of the report do not have the numbering and there are no in-text citations in the paragraph.

1. **Add additional system**

* Mr So also advises that the hotel management system should be included the laundry and maintenance system for the hotel. The laundry system should able to record down the dirty laundries are sent to which supplier. The maintenance system is just assigns the employee to do the maintenance of the hotel room. The maintenance should consider about either is handled by the hotel employee or the supplier of maintenance.

1. **Submit the diagrams to review**

* Mr So said that the review of diagram needs to take some time, so he will review the diagram and discuss with me in the next meeting.

**References Consulted:**

No References Consulted.

|  |  |
| --- | --- |
| **Student’s Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | **Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_** |
| **Supervisor’s Signature:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | **Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_** |

Project Diary & Supervisor Feedback-5

|  |  |
| --- | --- |
| Student Name: | Ang Sheng Yang |
| Supervisor: | Mr So Yong Quay |
| Meeting Date: | 9/5/2019 |

**Issues identified in previous meeting (including personal development goals):**

1. **Grammar problem in stage 1 report**

* I have many grammar errors in my stage 1 report when Mr So returns back the stage 1 report to me.

1. **Laundry stock and maintenance problem**

* I do not have considered about the laundry stock and the maintenance of the hotel problem.

1. **The picture colour problem**

* I also realised that the background colour of the gantt chart and make the texts inside the gantt chart are too small.

**Feedback received in previous meeting:**

1. **Do the correction of grammar errors in stage 1 report**

* Mr So advises me to do the correction of grammar error in the stage 1 report.

1. **Include the management of laundry and maintenance**

* Mr So advises that I should consider about the laundry management and maintenance.

1. **Change the colour of the picture**

* Mr So advises that the picture should have the black colour in background and white colour of texts.

**Action taken on feedback:**

1. **Do the correction of grammar errors in stage 1 report**

* I will correct the grammar errors in the stage 1 report.

1. **Include the management of laundry and maintenance**

* I will include the laundry and maintenance in the housekeeping system.

1. **Change the colour of the picture**

* I will change the colour of the gantt chart and able to see it more clearly.

**Matters to discuss:**

1. **The data flow diagram**

* After Mr So has reviewed my level 0 diagram, he said that the flow of each process should consist the data only. Moreover, Mr So also noticed that there are some lack of connections between the processes. Furthermore, Mr So also advises some of the entity need not be included in the diagram.

**References Consulted:**

No References Consulted.

|  |  |
| --- | --- |
| **Student’s Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | **Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_** |
| **Supervisor’s Signature:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | **Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_** |

Project Diary & Supervisor Feedback-6

|  |  |
| --- | --- |
| Student Name: | Ang Sheng Yang |
| Supervisor: | Mr So Yong Quay |
| Meeting Date: | 16/5/2019 |

**Issues identified in previous meeting (including personal development goals):**

1. **Data flow diagram**

* There are still some connection errors between the processes. Furthermore, there are also some entities do no need to be included in the diagram. So, I have to do some corrections about the dataflow diagram.

**Feedback received in previous meeting:**

1. **Mistakes in level 1 diagram**

* Mr So advises to do the correction of the level 1 diagram.

**Action taken on feedback:**

1. **Do the correction of level 1 diagram**

* I will have to do the correction of level 1 diagram.

**Matters to discuss:**

1. **The data flow diagram**

* After I have made the correction in level 1 diagram, I discuss with Mr So again. Mr So said that is no much more problem in the data flow diagram.

1. **The use case diagram**

* Since my use case diagram is completed, I submit to the Mr So and let him to review my use case diagram and Mr So will discuss about my use case diagram in next week of meeting.

**References Consulted:**

No References Consulted.

|  |  |
| --- | --- |
| **Student’s Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | **Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_** |
| **Supervisor’s Signature:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | **Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_** |

Project Diary & Supervisor Feedback-7

|  |  |
| --- | --- |
| Student Name: | Ang Sheng Yang |
| Supervisor: | Mr So Yong Quay |
| Meeting Date: | 23/5/2019 |

**Issues identified in previous meeting (including personal development goals):**

* No issue has been identified.

**Feedback received in previous meeting:**

1. **Dataflow diagram**

* Mr So said that the data flow diagram does not have too much problem.

**Action taken on feedback:**

1. **The stage 2 report**

* I will continue to do the diagrams that have not been done yet.

**Matters to discuss:**

1. **The use case diagram**

* After Mr So has reviewed my use case diagram, he advises my inventory system should add new function which is the adjust stock quantity. Moreover, there are many systems that lack of the searching function and should be added in the systems. Furthermore, the online payment and walk-in payment should be specified clearly in the sale system and online booking system. Besides that, there are wrong terms used in the housekeeping system and should be corrected.

**References Consulted:**

No References Consulted.

|  |  |
| --- | --- |
| **Student’s Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | **Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_** |
| **Supervisor’s Signature:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | **Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_** |

Project Diary & Supervisor Feedback-8

|  |  |
| --- | --- |
| Student Name: | Ang Sheng Yang |
| Supervisor: | Mr So Yong Quay |
| Meeting Date: | 30/5/2019 |

**Issues identified in previous meeting (including personal development goals):**

1. **The use case diagram**

* There are still some mistakes in my use case diagrams, so I will have to correct the mistake according to the feedback received by Mr So.

**Feedback received in previous meeting:**

1. **The use case diagram**

* Mr So said that the inventory system should have the adjust stock quantity function.
* There are many systems that lack of the searching function and should be added in the systems.
* The online payment and walk-in payment should be specified clearly in the sale system and online booking system.
* There are wrong terms used in the housekeeping system and should be corrected.

**Action taken on feedback:**

1. **The use case diagram**

* I will do the correction for the use class diagrams.

1. **The stage 2 report**

* I will continue to do the diagrams that have not been done yet.

**Matters to discuss:**

1. **Submission of stage 2 report**

* Since it is the submission day for the stage 2 report, so I will submit my stage 2 report to Mr So and wait for Mr So to review my stage 2 report. I will proceed to the stage 3 which is implementation system stage and start my coding.

**References Consulted:**

No References Consulted.

|  |  |
| --- | --- |
| **Student’s Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | **Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_** |
| **Supervisor’s Signature:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | **Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_** |

Project Diary & Supervisor Feedback-9

|  |  |
| --- | --- |
| Student Name: | Ang Sheng Yang |
| Supervisor: | Mr So Yong Quay |
| Meeting Date: | 6/6/2019 |

**Issues identified in previous meeting (including personal development goals):**

Since this day is public holiday and my stage 2 report is submitted for review by Mr So, so I will continue to decide which system should be implemented first in my final project 1.

**Feedback received in previous meeting:**

There is no any discussion made with Mr So and there is no feedback received.

**Action taken on feedback:**

Since there is no feedback received, so no action has taken.

**Matters to discuss:**

Since this day is holiday, so there is no discussion for my project. If I encounter any problems in my project, I will have the discussion with Mr So in the next week.

**References Consulted:**

No References Consulted.

|  |  |
| --- | --- |
| **Student’s Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | **Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_** |
| **Supervisor’s Signature:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | **Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_** |

Project Diary & Supervisor Feedback-10

|  |  |
| --- | --- |
| Student Name: | Ang Sheng Yang |
| Supervisor: | Mr So Yong Quay |
| Meeting Date: | 13/6/2019 |

**Issues identified in previous meeting (including personal development goals):**

There are some problems encountered when designing the interface of the back-end system. So, I have searched some template online as reference that can give a lot of help in designing the UI of the system.

**Feedback received in previous meeting:**

There is no any discussion made with Mr So and there is no feedback received.

**Action taken on feedback:**

Since there is no feedback received, so no action has taken.

**Matters to discuss:**

1. Stage 2 report

* After the stage 2 report has reviewed by Mr So, Mr So said that the cover page has wrong title. Furthermore, the DFD diagram has only one mistake and the level 1 diagram of payment system should be corrected to allow the receptionist review the payment detail.
* Moreover, the most problem is my ER diagram. There are many lacks of relationships between the entities. The relationship identity also should be in unique form. The foreign key also should be excluded from the ER diagram.
* There are some lack of functions in my use case diagram and should be added.
* There are many grammar errors in my use case description and should be corrected.
* The decision should be specified either valid or invalid in my activity diagram.

**References Consulted:**

No References Consulted.

|  |  |
| --- | --- |
| **Student’s Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | **Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_** |
| **Supervisor’s Signature:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | **Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_** |