**Online Hotel Management System**

**Low Level Design(LLD)**

**Date:**

**Submitted BY**

**Submitted To**

DOCUMENT APPROVAL

**Approvers of this document**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Department** | **Role** | **Signature** | **Date** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Document Change History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Document** **Version #** | **Author** | **Date** | **Description** |
| 1.0 |  | 16/05/2022 | Online Hotel Management System LLD |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Table of Content**

**1.0 Document Purpose**

**2.0 Intended Audience**

**3.0 Project Background, Objectives**

**3.1 Project Background**

**3.2 Project Objective**

**4.0 Design Pattern**

**5.0 Introduction**

**5.1 Purpose**

**5.2 Scope of The Project**

**5.3 User Classes and Characteristics**

**5.3.1 User Classes**

**5.3.2 User Characteristics**

**5.3.3 Operating Environment**

**5.4 Assumptions and Dependencies**

**6.0 Solution Diagram**

**7.0 Solution Steps**

**8.0 Data Model/Tables**

**9.0 Activity Diagram**

**10.0 Sequence Diagram**

**11.0 UML Diagrams**

**12.0 Status Codes**

# Document Purpose

This document describes the solution architecture for Online Hotel Management System

# Intended Audience

This document is intended as a reference for the following roles and stakeholders who are interested in the Online Hotel Management System technical architecture.

|  |  |
| --- | --- |
| Role | Nature of Engagement in WB Classics Portal Technical Architecture |
| Product Owners/SME | Key stakeholder to ensure that the architecture is aligned with business goals. |
| Business Analysts | Business analysts are one of the stakeholders who are informed with the key architectural decisions. |
| Enterprise Architects | To enforce Customer management Platform Architecture is aligned to business goals and architecture, architectural guidelines. |
| Solution Architects | To ensure solution design and architecture is aligned to business requirements, architectural guidelines. |
| Developers | Use Technical Architecture Document as the guiding document for detail design and implantation approach to align with Customer management Microservice |

# Project Background, Objective(s)

## Project Background

Online Hotel management System Management of Customers details where one can register themselves and perform various operations

## Project Objective

Online Hotel management System​ will perform various operations like Inserting , updation and deletion of Guest Details.

Receptionist can enter the Guest Details and then he/she can perform all the operations.

# Design Pattern

|  |  |  |
| --- | --- | --- |
| # | Name | Description |
| 1 | API | Using HTTP requests, we will use the respective action to trigger various operations |

# Introduction

## 5.1 Purpose

The Software Requirements Specification (SRS) will provide a detailed description of the requirements for the Hotel Management System (HMS). This SRS will allow for a complete understanding of what is to be expected from the newly introduced system which is to be constructed. The clear understanding of the system and its’ functionality will allow for the correct software to be developed for the end user and will be used for the development of the future stages of the project. This SRS will provide the foundation for the project. From this SRS, the Hotel Management System can be designed, constructed, and finally tested.

**5.2 Scope of The Project**

The introducing software, Hotel Management System which is going to be implemented for Hotel will automate the major operations of the hotel. The Reservation System is to keep track in room and hall reservation and check availability. The Room Management System is for manage all room type of room services. The Inventory Control System will keep track in all inventories of the hotel and guest details will handled by guest management. Administration department will monitor the all. There is three End Users for HMS. The End Users Are Owner, Manager and Receptionist. Owner can access to all system functionalities without any restrictions. Manager can access to all system functionalities with limited restrictions. Receptionist can only access to the Reservation management section. To keep restrictions for each End User levels HMS can create different Login functions.

The objectives of the automated Hotel Management System is to simplify the day to day processes of the hotel. The system will be able to handle many services to take care of all customers in a quick manner. As a solution to the large amount of file handling happening at the hotel, this software will be used to overcome those drawbacks. Safety, easiness of using and most importantly the efficiency of information retrieval are some benefits the development team going to present with this system. The system should be user appropriate, easy to use, provide easy recovery of errors and have an overall end user high subjective satisfaction.

## User Roles and Characteristics

### 5.3.1 User Roles

There are three user levels in Hotel Management System of Hotel.

1. Owner
2. Manager
3. Receptionist

### Characteristics of User Classes

**Owner:-**

Hotel owner has the privilege of Monitoring and authorization of all the tasks handle by the system. He can access every function performed by the system. Owner of the company as well as the system can access to the administration panel which is consider the core of the system. As the main authorized person of the company owner gets the ability to manage the other users including their user levels and privileges. Taking backups of the system and restoring system can also be done by the Owner. Meanwhile he will be able to take all the kinds of reports available in the system. As the owner of the system and the company he has the power to set room rates as well. Hotel owner has the sole right of deleting a staff member from the system database.

**Manager:**

Manager is responsible for managing resources available in hotel management system. Manager also has most of the privileges mentioned above except the things regarding the payment handling. The reason for using a Manager is to reduce the work load done by the owner that cannot be assigned to the receptionist, as those tasks seem much responsible. The user level, Manager has the authority to take all the reports available in the system but here also except the reports related to financial stuff, hotel income. Manager has other abilities that receptionist, user level has. Such as, adding new staff member to the system, Modifying them or removing them, Adding new guests to the system, Modifying them and removing them from the system, Adding new inventory to the system, Modifying them and removing them. Adding new room types to the system, modifying them and removing them

**Receptionist:**

As a hotel receptionist, he or her role will be to attain the goals of bookings and to ensure that all guests are treated with a high standard of customer service. Hierarchically receptionist role has the least accessibility to the system functions. Receptionist plays the boundary role of the system .He or she can perform limited functions such as registering new guest to the system, make reservations, Sending e-mail reminders to clients for booking confirmation. Management of hotel will prefer to hire receptionist who have a good standard of general education and possibly in subjects such as English, math and IT.

## Operating Environment

Hardware and software requirements

**Hardware:-**

1. **Operating System** Supports all known operating systems, such as Windows, Linux
2. **Computer** 512GB+8GB(RAM)

**Software:-**

1. Software is designed to run on any platform above Microsoft Windows.
2. Microsoft .NET core 4.0
3. Microsoft SQL Server Management Studio Express 2019.
   1. **Assumptions and Dependencies**

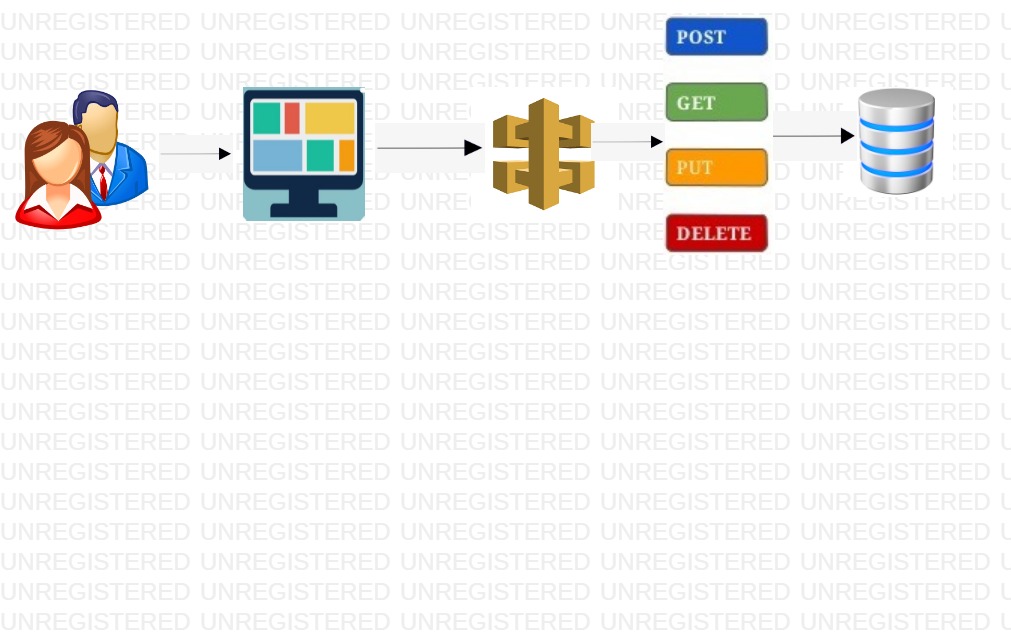
Some software used in implementing the system is with high cost and the client has

agreed to afford the amount of money needed to purchase them. It’s assumed that

client won’t change that decision on the next phases of the software development.

Although we assume that client is using windows 7 or windows 8. Otherwise if client use an open source operating system, there is a need of changing the SRS accordingly.

**6.0 Solution Diagram**

****

**7.0 Solution Steps**

**Make reservations:**

Reservations will be done by the receptionist, one of the three end-users.

The receptionist will add a new reservation for the guests away from the hotel making reservations.

The receptionist will enter the required details such as the unique code of the guest, number of children to stay, number of adults to stay, check-in date along with time, check-out date along with time, the status of booking and staying, number of nights to stay at the hotel.

Once the details have been entered the system searches for the room details present in the hotel, details like the of rooms available, types of rooms available, price range, etc.

The receptionist asks the guest to select any particular room from the available list of rooms and the guest selects one room and confirms the tariff.

The system records guest's entered details and the receptionist confirms the booking on the system.

After the completion of the reservation, the system generates the confirmation receipt and makes changes to the database.

**Manage guest:**

Add guest operation will be done by the receptionist.

Here the new guests, as well as reserved guests, checks into the hotel.

The receptionist selects add guest option and enters the prompt details for the reserved guests with the particular code.

The system validates the details and checks the database for any reservations, if yes displays the successful message and makes the booking, and the steps follow.

If the guest hasn't made any reservations then the receptionist adds the guest details such as code, phone number, company, name, e-mail, gender, address, and the steps follow.

The receptionist can also search for any particular guest with a particular code and the system validates the inputs and displays the guest details. If the guest details are incorrect it shows an "incorrect details entered" message.

Any guest details can be deleted with the delete guest option displayed on the user interface. Particular guest details can be entered and the system shows the details of that guest and the receptionist can confirm the details and delete them from the database.

There is another option to edit guest information, with this we can change any guest details by searching for that particular guest details and make the corrections as required, and after submitting it will be reflected in the database.

**Manage staff members:**

Add staff members function will be performed by the manager, one of the three end-users.

The manager has access to the staffing methods.The manager adds a new staff by entering some particular details like code, employee name, employee address, NIC, salary, employee age, occupation, email, etc.The system validates particular details and adds the details to the database.

The manager can also view the details of the staff by using the option view staff where the user has to enter the details of any particular staff and submit them. The system validates the inputs and displays the details.

There are other options like updating and deleting staff with this w the manager can update staff information after making changes and can delete any particular staff information and after submitting the data will be saved into the database.

**Manage rooms:**

The manage rooms function will be managed by the manager.The manager can use the options such as adding a  room, updating a room or deleting a room presented in the user interface.

The system prompts the manager to fill out room details and then the system validates the new room information and it creates a new room and the database will be updated.It displays the successful message after the successful addition of a room. Update room options will be used to change the room properties.

The manager will be asked to select the room and when submitted the system shows the room properties to edit them, after making the required changes the room properties will be saved into the database.

The delete room option will be used to delete any room.

The manager needs to select any particular room and marks it, the system will then delete that room and updates the database.

**Search rooms:**

The search room function will be performed by the receptionist.

There will be an option on the user interface where available rooms can be displayed so that the receptionist can make reservations and bookings for the guests.

When the search rooms option is selected it shows the list of types of rooms available in the hotel and once clicked on any one of the types the system shows the list of rooms available for booking so that the receptionist can select from them.

**Add payment and issue bills:**

This function will be done by the receptionist.

This step comes after entering the details of the guests and selecting a room and booking that room.

The add payment option will be displayed when clicked it asks for the mode of payment like cash, UPI, credit or debit cards, etc.

The receptionist will ask the guest for their preferred payment method and processes the operation according to the decision by the guest.

And the bill will be generated after the successful payment with some specific details of the guests along with payment time, and with a physical signature of the receptionist on the receipt.

**Manage inventory:**

This operation will be done by the manager.

The inventory referred to the number of rooms available, equipment or furniture in the rooms, and every resource used in a hotel on a day-to-day basis.

The manager will be provided with an option to add, edit, and delete under the manage inventory option where the manager can make changes to the inventory of the hotel.

By clicking ok the add option he can update the newly purchased inventory and the system will update the database.

With the edit option he can change the status of the inventory like quantity brought, quantity utilized, and quantity left.

With the delete option, the manager can change the list of inventory if any particular items are damaged or out of stock.

**Set room rate:**

The set room rate function will be done by the manager.

The manager can access the user interface to change the rate of current rooms and can also set the rate for any newly added rooms in the hotel by clicking the room properties of any particular room.

Then the manager will be provided with a set rate option with that the manager can change the rate of that particular room.

The system then changes the rate of the rooms and makes changes in the database.

**View reports:**

The view reports function will be performed by the owner, one of the three end-users.

This option is used to view all the financial reports for a specific time.

When this option is clicked the system prompts the owner to select two dates ie, the starting date and the ending date.

The system displays all the revenue between these two dates.

**Manage users:**

The manager users function will be only performed by the owner. He is the only one with access to this function.

The add, update, and delete users options will be displayed under the manage users by this the owner can perform all operators on the staff of the hotel.

By using add option he can add new users like another owner, a new manager, and a new receptionist.

By using the update option he can update the current users' details.

By using the delete option he can delete any particular user from the list.

The system validates the data entered by the owner and saves it into the database accordingly.

**E-mail notifications:**

E-mail notifications are an alert function called when any specified event has occurred.

Many events require an acknowledgment, to make this happen e-mail notifications function is used.

When any guest makes the payment for the room booking an email will be sent to the guest's particular email address as an acknowledgment along with the e-bill.

When there are no rooms available then an email will be sent to the manager's email from the system that no rooms are available right now.

# 7.0 Classes/function

|  |  |  |
| --- | --- | --- |
| **#** | **Class** | **Description** |
| 1 | Guest.cs | Model holds the Guest schema details |
| 2 | Inventory.js | The handler to handle the registration of customers which calls the registerCustomerService class |
| 3 | Invoice.cs | It contains the core business logic for the registration of customers. Which calls the registrationCustomerRepository class to create the customer in database |
| 4 | MakeReservation.cs | This class deals with the data accessibility for customer registration |
| 5 | PaymentDetails.cs | The handler to handle the listing of customers. which calls the listCustomerService class |
| 6 | Room.cs | It contains the core business logic for the registration of customers. Which calls the listCustomerRepository class to list the customer from database |
| 7 | ServiceBill.cs | This class deals with data accessibility for customer list |
| 8 | Staff.cs | The handler to handle the deletion of customers. which calls the removeCustomerService class |
| 9 | TypeOfRoom.cs | It contains the core business logic for the registration of customers. Which calls the removeCustomerRepository class to remove the customer from database |

**8.0 Data Model/ Table**

**8.1 Guest:**

|  |  |  |
| --- | --- | --- |
| S.No | ColumnName | Datatype |
| 1(PK) | MemberCode | int |
| 2 | Name | nvarchar(50) |
| 3 | Age | int |
| 4 | MobileNumber | nvarchar |
| 5 | Gender | nvarchar |
| 6 | City | nvarchar |
| 7 | Pincode | nvarchar |
| 8 | Citizenship | nvarchar |
| 9 | Occupation | nvarchar |
| 10 | DrivingLicence | nvarchar |
| 11 | Email | nvarchar |

**8.2. Room:**

|  |  |  |
| --- | --- | --- |
| S.no | ColumnName | DataTypes |
| 1(PK) | RoomId | int |
| 2 | RoomName | nvarchar |
| 3 | BuildingNumber | int |
| 4 | FloorNumber | int |
| 5 | Status | nvarchar |
| 6(FK) | RoomType | nvarchar |

**8.3. TypeOfRoom:**

|  |  |  |
| --- | --- | --- |
| S.no | ColumnName | DataTypes |
| 1(PK) | RoomType | nvarchar |
| 2 | Adults | int |
| 3 | Child | int |
| 4 | StandardPrice | int |

**8.3. MakeReservation:**

|  |  |  |
| --- | --- | --- |
| S.no | ColumnName | DataType |
| 1(PK) | ReservationCode | int |
| 2(FK) | RoomId | int |
| 3(FK) | MemberCode | int |
| 4 | Check-inTime | datetime |
| 5 | Check-outTime | datetime |
| 6 | NoOfNights | int |
| 7 | RoomPrice | int |

**8.4. ServicesBill:**

|  |  |  |
| --- | --- | --- |
| S.no | ColumnName | DataTypes |
| 1(PK) | BillNo | int |
| 2 | Date | datetime |
| 3 | Price | int |
| 4 | Tax | int |
| 5 | Units | int |
| 6 | ItemName | nvarchar |
| 7(FK) | MemberCode | int |
| 8(FK) | ReservationCode | int |
| 9 | Amount | int |

**8.5. Invoice:**

|  |  |  |
| --- | --- | --- |
| S.no | ColumnName | DataType |
| 1(PK) | InvoiceId | int |
| 2(FK) | MemberCode | int |
| 3 | RoomPrice | int |
| 4 | ServicesCost | int |
| 5 | Total | int |

**8.6. PaymentDetails:**

|  |  |  |
| --- | --- | --- |
| S.no | ColumnName | DataType |
| 1(PK) | PaymentId | int |
| 2 | PaymentTime | datetime |
| 3 | PaymentMethod | nvarchar |
| 4(FK) | MemberCode | int |
| 5(FK) | InvoiceId | int |
| 6 | Total | int |

**8.7. Staff:**

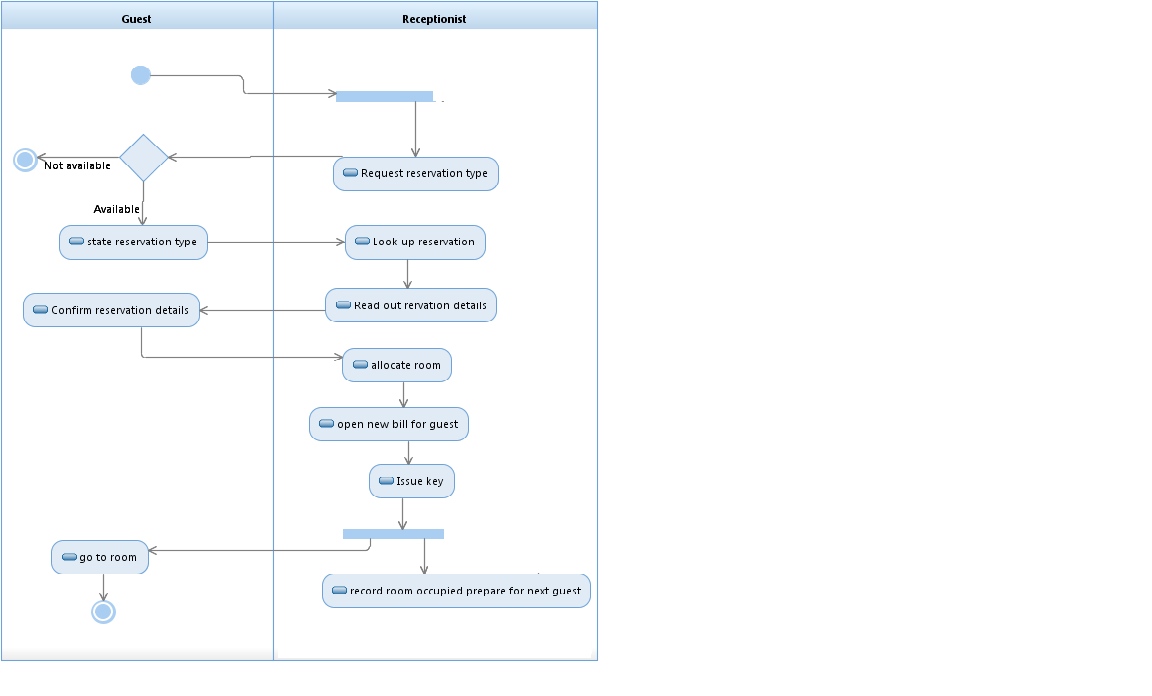
|  |  |  |
| --- | --- | --- |
| S.no | ColumnName | DataTypes |
| 1(PK) | UserId | int |
| 2 | Name | nvarchar |
| 3 | MobileNumber | int |
| 4 | Email | nvarchar |
| 5 | Gender | nvarchar |
| 6 | Salary | int |
| 7 | NIC | int |
| 8 | Address | nvarchar |
| 9 | Password | nvarchar |
| 10 | Role | nvarchar |

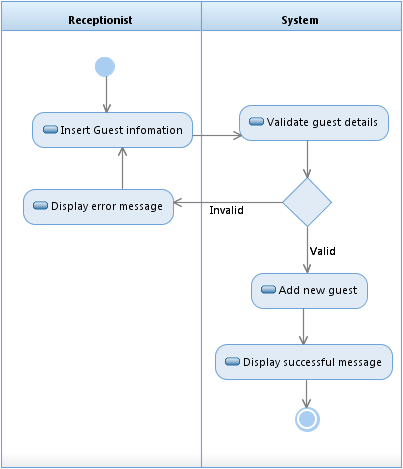
**8.9 Inventory:**

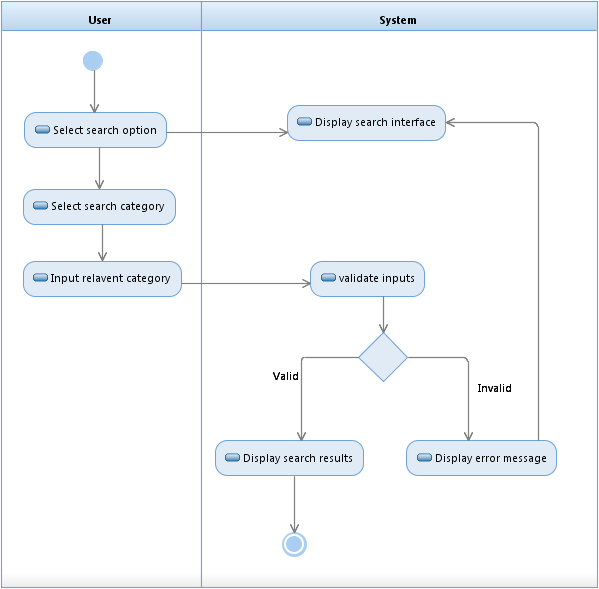
|  |  |  |
| --- | --- | --- |
| S.no | ColumnName | DataTypes |
| 1(PK) | InventoryId | int |
| 2 | InventoryName | nvarchar |
| 3 | Quantity | int |
| 4 | UnitPrice | int |

**9.0 Activity Diagram**

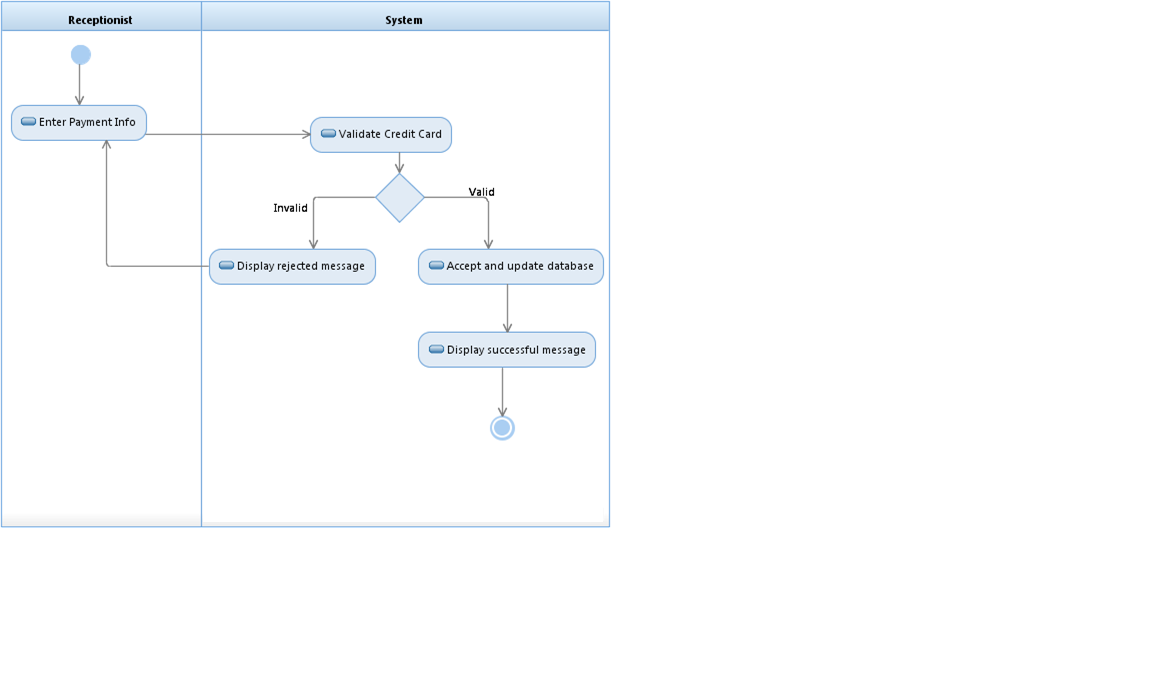
**9.1 Make Reservation**

****

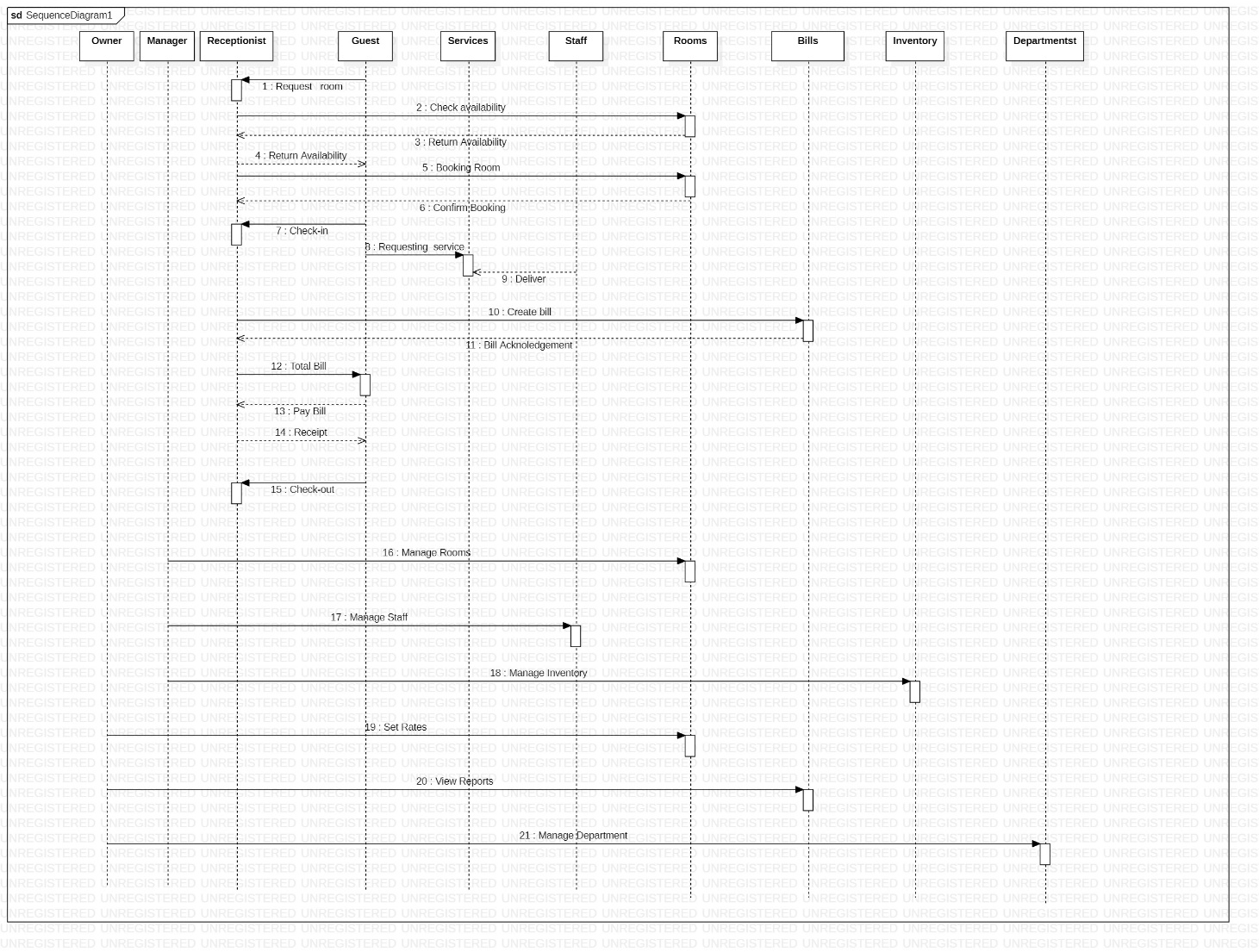
 **9.2 Add Guest**

**9.3 Search**

**9.4 Add Payment**

****

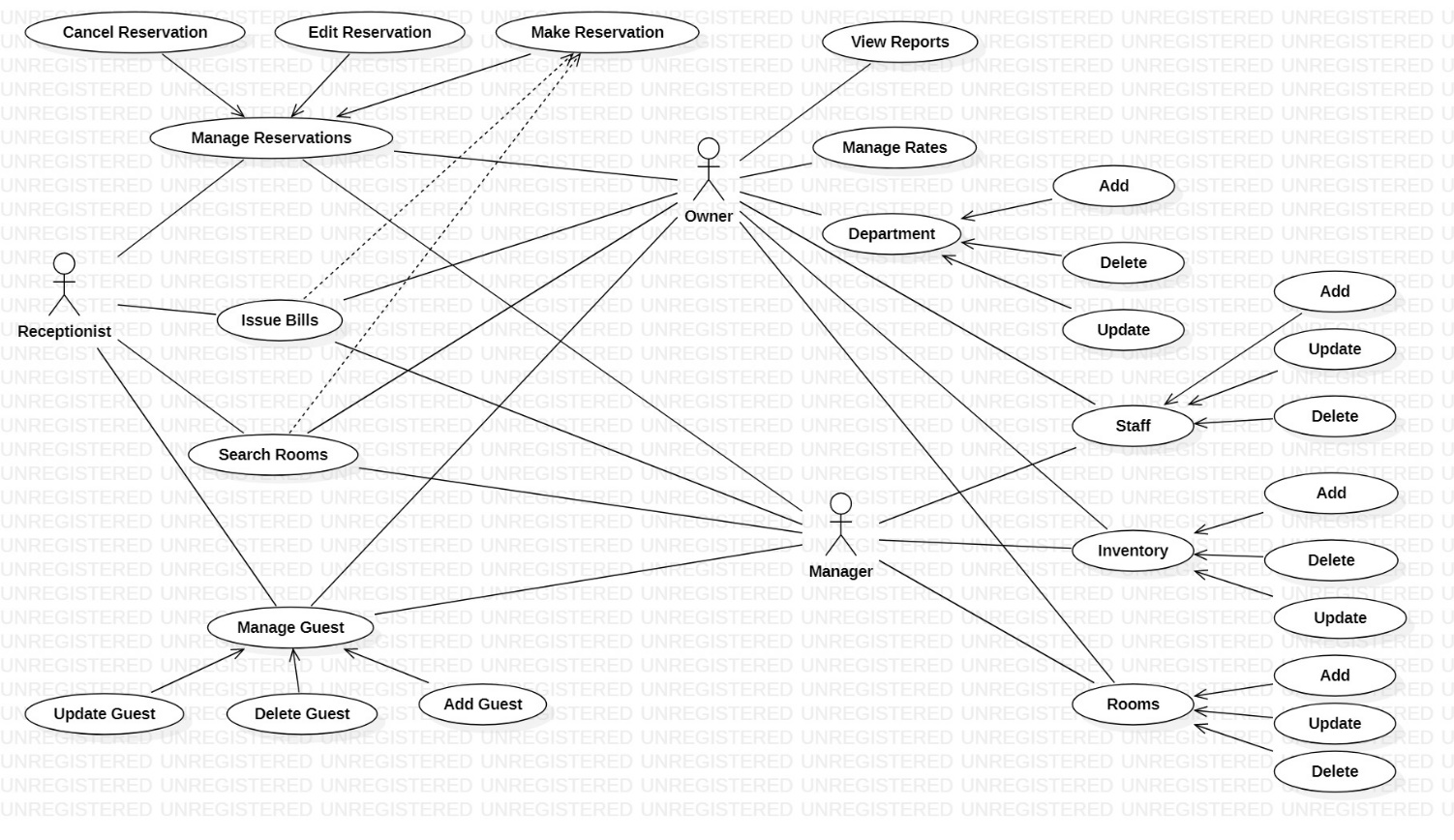
**9.0 Sequence Diagram:**

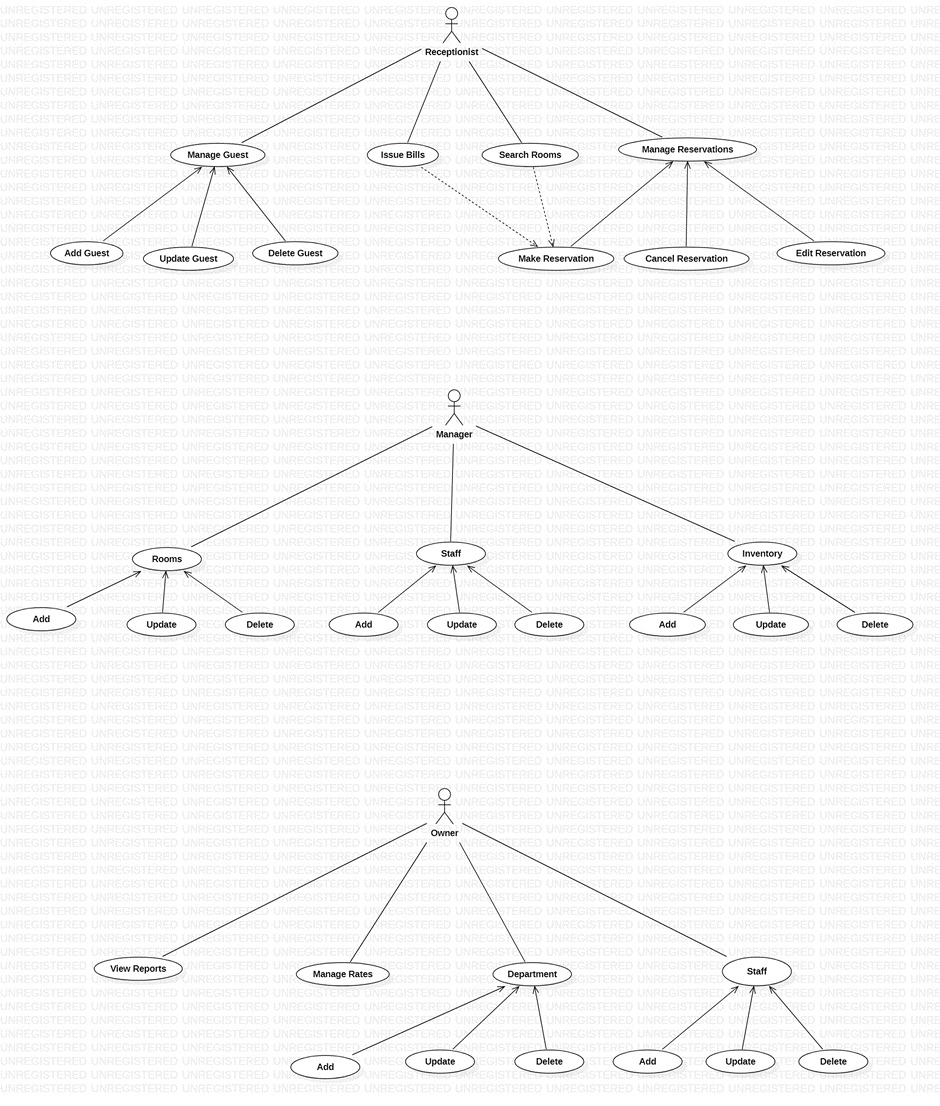


**10.0 UML Diagrams:**

UML diagrams to understand the system in a better and simple way. A single diagram is not enough to cover all the aspects of the system. UML defines various kinds of diagrams to cover most of the aspects of a system.

You can also create your own set of diagrams to meet your requirements. Diagrams are generally made in an incremental and iterative way.





**12.0 Status Codes**

201 –User Registered

200 - Request succeeded

400 – Inputs are invalid

404 –User Not found

502 – Bad gateway