

# **HOTEL MANAGEMENT SYSTEM**

**Group - 5 - Members**

**AM.EN.U4CSE20348 - NIRMAL KUMAR K**

**AM.EN.U4CSE20364 - SIDHARTH M DINESH**

**AM.EN.U4CSE20365 - SIVASANKAR A**

S3 CSE Department Amrita Vishwa Vidya Peetham, Amritapuri - 2021

# INDEX

CHAPTER	TITLE	PAGE
1	INTRODUCTION	3
2	ABSTRACT	4
3	LIST OF FUNCTIONALITIES	5
	1.1 FUNCTIONAL REQUIREMENTS	
	1.2 USE CASES	
	1.3 ACTORS	
4	USE CASE DESCRIPTION	6
5	SYSTEM DESIGN	
	1.1 USE CASE DIAGRAM	9
	1.2 CLASS DIAGRAM	10
6	CLASS DESCRIPTION	11
7	DATABASE DETAILS	16
8	USER INTERFACE	
	8.1 HOME PAGE	18
	8.2 USER LOGIN	18
	8.3 ADMIN LOGIN	19
	8.4 ADMIN VIEW	
	8.4.1 ADMIN FUNCTIONALITIES	20
	8.5 ADMIN ASSISTANT	23
	8.6 BRANCH MANAGER	24
	8.7 HEAD CHEF	24
	8.8 CHEF	25
	8.9 WAITER	26
	8.10 CUSTOMER	26
	8.11 MENU	27
	8.12 PAYMENT	27
9	TOOLS USED	28
10	CONCLUSION	29
11	REFERENCES	30
12	APPENDIX	31

## CHAPTER 1

## INTRODUCTION

This project mainly focusses on concepts based on OOP to make a Java Application for a group of hotels. The Hotel Management System provides a path for the employees in the Hotel that helps them to ease their access to information and implement assigned functionalities.

The application is meant to be used by the Admin of the restaurant, Assistant Admin, Branch Managers, Chefs, Head Chefs, Waiters as well as the customers of the restaurant. Each of the Employees have a customized view of the Application based upon their designation in the Hotel.

Java Swings have been used in this project to implement the UI phase for the application. The project uses Java Database Connectivity(JDBC) to insert, access, update and delete user defined data from the Hotel Management System database.

## CHAPTER 2

## ABSTRACT

An entrepreneur starts a group of hotels in important cities of different states. We created an automation software that can manage all his requirements to run the hotels.

Each hotel has its' own id, name, start\_date, revenue\_planned, expenditure\_expected, city, and state. The chefs in each branch of the hotel are identified by a social security number (SSN), and are also described by a name, date of join and a cuisine in which he/she specializes. The chefs may be asked to work in more than one branch if shortage demands. There is a specific menu card for each branch in which a manager is one who takes care of the specific branch requirements. Also, the administrative assistant who serves as an assistant to admin who can manage a small group of hotel. Each food item in the Menu is then assigned to a chef based on his specialization. The performance of each hotel can be analyzed by their profitability and their respective average customer review for Food quality, ambiance, and service.

### **Functional Requirements of the System**

1. User other than the customers need to login with the system before using the application.
2. User can view personal details like SSN, address, salary, etc.
3. Admin can update Hotel details whenever required.
4. Customers can pay with ease.
5. User can update their personal details.
6. User can logout of the application.

### **Use cases**

1. Login
2. Enter or view personal details.
3. Branch details
4. Update branch
5. Chef Admission
6. Chef Termination
7. View official details
8. Update official details
9. Salary
10. Hotel Menu
11. Update Menu
12. Update prices
13. Last Maintenance
14. Service Ratings
15. Payment
16. Profitability
17. Revenue
18. Logout

### **Actors**

1. Administrator
2. Administrative assistant
3. Branch Manager
4. Head Chef
5. Chef
6. Waiter
7. Customer

## CHAPTER 4

## USE CASE DESCRIPTION

Use Case	Login
Actors	Administrator, Administrative assistant, Branch Manager, Head Chef, Chef, Waiter
Inputs	Username and Password
Outputs	The system will state whether inputs are correct or not.
Description	Allows user to login to the system using username and password which later verified to gain access to the system using verify login. Login error message is displayed if the credentials are incorrect.

Use Case	Edit or view personal details
Actors	Administrative assistant, Branch Manager, Head Chef, Chef, Waiter
Inputs	Name, SSN, DOJ, Phone, Address, Title
Outputs	The system will display/edit changes in personal details.
Description	Allows user to enter/update personal details in the system.

Use Case	Branch details
Actors	Administrator, Administrative assistant, Branch Manager
Description	Allows user to view important details about the branch

Use Case	Update branch
Actors	Administrator, Administrative assistant, Branch Manager
Inputs	Expected revenue, planned expenditure,
Outputs	The system will display the results after updating.
Description	Allows user to update any credentials of a particular branch in the system.

Use Case	Chef Admission
Actors	Administrator
Inputs	SSN, Name, address, Phone, Cuisine, Hotel id
Outputs	The system will add the details of the chef.
Description	Allows user to add details of new chef to a particular branch in the system.

Use Case	Chef Termination
Actors	Administrator
Outputs	The system will remove the details of the chef.
Description	Allows user to terminate a chef in a particular branch in the system.

Use Case	Salary
Actors	Administrator, Administrative assistant
Description	Allows user to access salary details of employees.

Use Case	View official details
Actors	Administrator, Administrative assistant, Branch Manager, Head Chef, Chef
Description	Allows user to view their salary, ratings, chef branch, and other official details, it acts as a parent function for update official details. Parent function of Update official details.

Use Case	Update official details
Actors	Administrator, Administrative assistant
Inputs	Salary, Revenue, Expenditure, Chef branch
Outputs	The system will update the details and present the details.
Description	Allows users to update salary, chef branch and other official details of employees.

Use Case	Hotel Menu
Actors	Administrator, Administrative assistant, Branch Manager, Head Chef, Chef, Waiter, Customer
Description	Allows user to view food menu of each hotel.

Use Case	Update Menu
Actors	Administrator, Administrative assistant, Branch Manager, Head Chef, Chef
Inputs	Dish, Cuisine
Outputs	The system will update the details of the dish and the respective cuisine.
Description	Allows user to add/remove a dish from Hotel Menu

Use Case	Update Price
Actors	Administrator, Administrative assistant, Branch Manager
Inputs	Dish Id
Outputs	The system will update the price of the dish .
Description	Allows user to update dish price from Hotel Menu

Use Case	Service Ratings
Actors	Customer
Inputs	User Ratings.
Outputs	The system will save the ratings obtained for the service.
Description	Allows user to give a review for the service.

Use Case	Payment
Actors	Customer
Inputs	Customer objects.
Outputs	The system will show the output after payment.
Description	Allows user to make online payments and payment is related to Revenue.

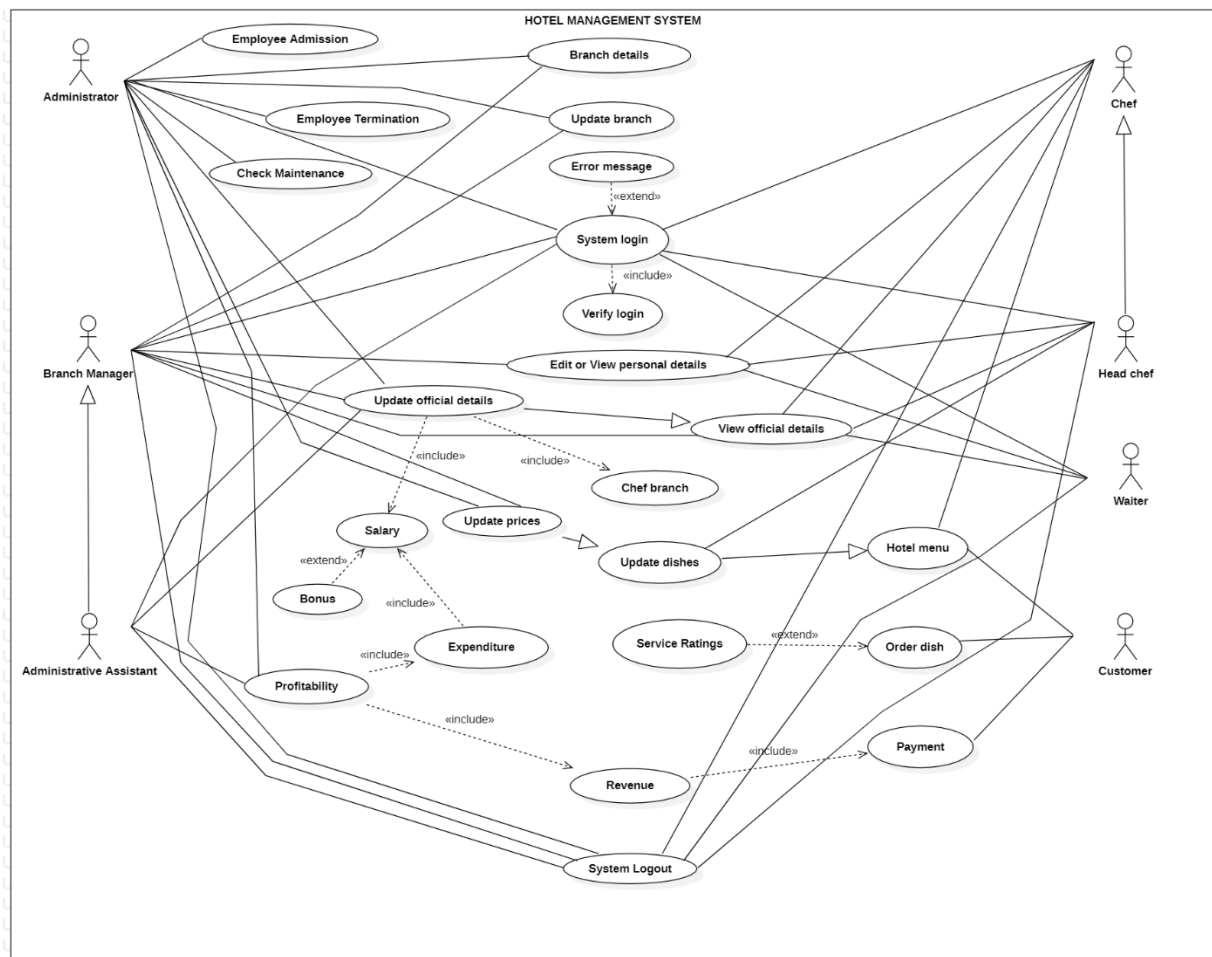
Use Case	Last Maintenance date
Actors	Administrator, Administrative assistant, Branch Manager
Description	Allows user to view the last maintenance date of the respective hotel

Use Case	Profitability
Actors	Administrator, Administrative assistant, Branch Manager
Description	Allows user to determine the total revenue of the branch which includes expenditure and salary.

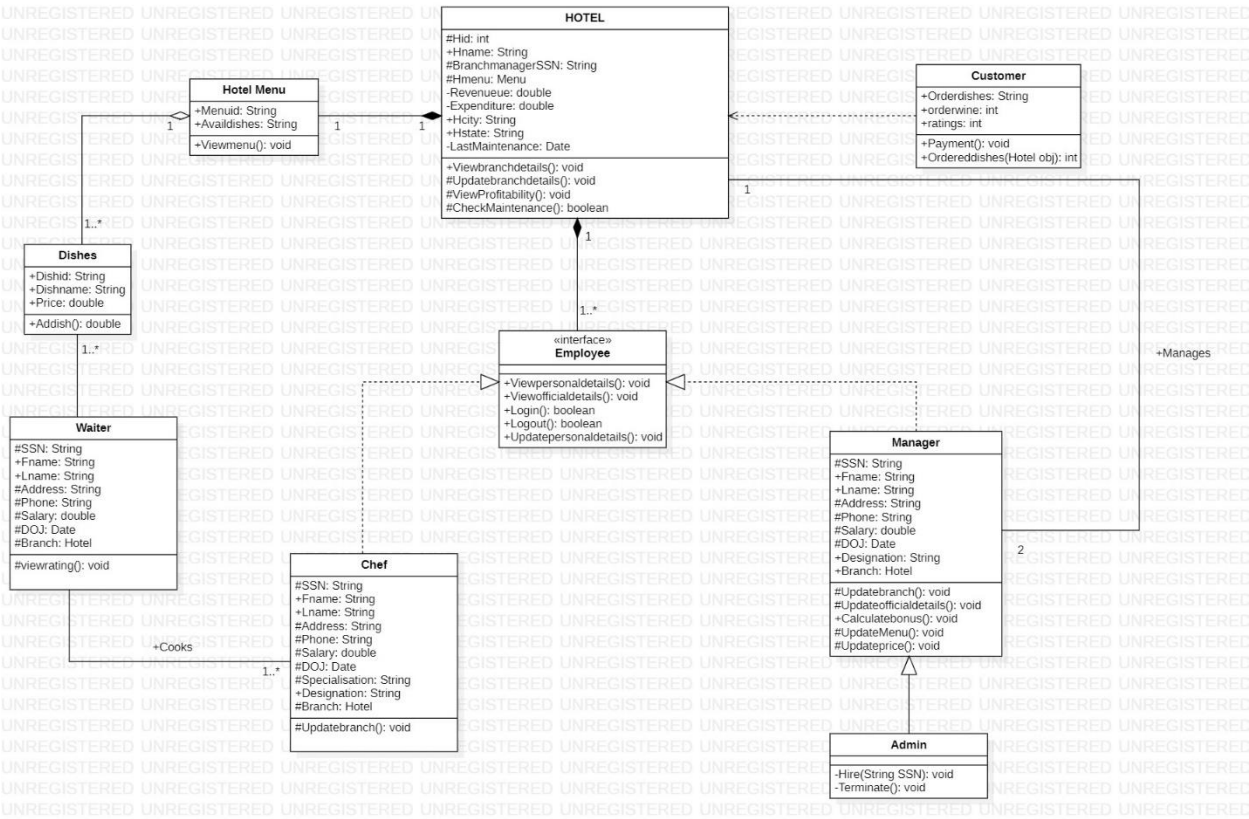
Use Case	Revenue
Actors	Administrator, Administrative assistant, Branch Manager
Description	Allows user to update revenues and expenditure from payment.

Use Case	Logout
Actors	Administrator, Administrative assistant, Branch Manager, Head Chef, Chef, Waiter
Outputs	The user will get logged out of the system.
Description	System allows the user to logout.





# CLASS DIAGRAM



**Hotel Class**

- Integer Hid which is the hotel id, declared as protected data member to provide partial data hiding.
- String Hname which is Hotel name, declared as a public data member.
- String BranchmanagerSSN that is SSN of Branch Manager, declared as a protected data member.
- Object of menu class Hmenu ,declared as protected data member.
- double Revenue, declared as private data member.
- double Expenditure, declared as private data member.
- String Hcity that is the city where the hotel is in, declared as a public data member.
- String HState that is the state where the hotel is in, declared as a public data member.
- void Viewbranchdetails(), which prints details of the branch details, declared as a public method.
- void, Updatebranchdetails , which enables the user to update branch details, declared as a protected data member.
- Date LastMaintenancedate is the date when last maintenance check was done, declared as a private data member.
- void ViewProfitability, which checks the hotel is running on a profit or not by comparing the revenue to the expenditure of the hotel.

**Employee Class**

The employee interface is a part of hotel class, that is, the employee class exists only if the hotel class exists and hence the relationship between employee interface and hotel class is a composition. Since one hotel contains several employees, its multiplicity is defined as one to many.

The employee class is defined as an interface class and has some methods defined in it. All the methods in the interface are abstract methods and will be implemented in both Chef and Manager with their own specifications.

It also provides abstraction to various details of Manager and chef.

- void Viewpersonaldetails(), which gives the personal details of the employee. It is declared as a public method.
- void viewofficialdetails(), which gives the official details of the employee. It is declared as public method.
- Bool Login (), enables the user to gain access to the Hotel Management System. Its declared as a public
- Bool Logout (), enables the user to end a session of system. Its declared as a public
- void Updatepersonaldetails(), enables user to update any personal details. It is defined as a public.

## Branch Manager Class

The manager class implements the employee interface and hence all the abstract methods defined in the employee interface is implemented here and it has an interface realization also shares a manages relationship with Hotel class that has a multiplicity of 1 to 2, which means that a hotel can have 2 managers which are Assistant Administrator and Branch Manager. The data members defined in it are:

- String SSN, which is the social security number of the Manager. It is defined as a protected data member.
- String Fname , is the first name of the manager. It is declared as a public data member.
- String Lname , is the last name of the manager. It is declared as a public data member.
- String Address, is the address of the manager. It is declared as a protected data member.
- String Phone, is the contact number of the employee. It is declared as a protected data member.
- double Salary, is the salary which the manager earns. It is declared as a protected data member.
- Date DOJ, is the date of joining of the manager. It is declared as protected data member.
- String Designation, is the designation of the manager. It is declared as a public data member.
- Object of hotel class, Branch, is the Branch of the manager. It is declared as public data member.
- Date LastMaintenancedate is the date when last maintenance check was done, declared as a private data member.
- In addition to the abstract methods defined in employee interface, this class has certain other methods defined in it
- void Updatebranch(), is a method that enables the manger to update the branch of manager. It is declared as a protected method.

## Chef Class

The Chef class implements the employee interface and hence all the abstract methods defined in the employee interface is implemented here and it has an interface realization. It also shares a cook's relationship with Dishes class that has a multiplicity of many to many, which means that a hotel can have several dishes and several chefs.

- String SSN, which is the social security number of the chef. It is defined as a protected data member.
- String Fname , is the first name of the chef. It is declared as a public data member.
- String Lname , is the last name of the chef. It is declared as a public data member.
- String Address, is the address of the chef. It is declared as a protected data member.
- String Phone, is the contact number of the chef. It is declared as a protected data member.
- double Salary, is the salary which the chef earns. It is declared as a protected data member.
- Date DOJ, is the date the chef joined the hotel. It is declared as protected data member.
- Void Updatebranch() , which allows the chef to update the Branch-it is declared as a protected method.

## Waiter Class

The Chef class implements the employee interface and hence all the abstract methods defined in the employee interface is implemented here and it has an interface realization. It also shares a cook's relationship with Dishes class that has a multiplicity of many to many, which means that a hotel can have several dishes and several chefs.

- String SSN, which is the social security number of the Waiter. It is defined as a protected data member.
- String Fname , is the first name of the Waiter. It is declared as a public data member.
- String Lname , is the last name of the Waiter. It is declared as a public data member.
- String Address, is the address of the Waiter. It is declared as a protected data member.
- String Phone, is the contact number of the Waiter. It is declared as a protected data member.
- double Salary, is the salary which the Waiter earns. It is declared as a protected data member.
- Date DOJ, is the date the Waiter joined the hotel. It is declared as protected data member.
- Object of hotel class, Branch, is the Branch of the manager. It is declared as public data member.
- Void Updatebranch() , which allows the Waiter to update the Branch-it is declared as a protected method.
- Calculatetip(), is the tip offered to Waiter from Customer It is a protected data member.

## Admin Class

The Admin class inherits the data members and member functions of the class manager. It also has some member functions of its own.

- void Terminate(String SSN) member function allows the Admin to Fire Employees which in turn delete them from the database. Its declares as private.
- void Hire () member function allows the Admin to add new employees to the database. Its declares as private.
- String SSN, which is the social security number of the Admin. Tt is defined as a protected data member.
- String Fname , is the first name of the Admin. It is declared as a public data member.
- String Lname , is the last name of the Admin. It is declared as a public data member.
- void Updateprice(), is a method that enables the Admin to update the price of the dishes. It is declared as a protected method.
- Date LastMaintenancedate is the date when last maintenance check was done, declared as a private data member.
- In addition to the abstract methods defined in employee interface, this class has certain other methods defined in it
- void Updatebranch(), is a method that enables the Admin to update the branch of Admin. It is declared as a protected method.

## **Admin Assistant Class**

The Admin Assistant class inherits the data members and member functions of the class manager. It also has some member functions of its own.

- String SSN, which is the social security number of the Admin. It is defined as a protected data member.
- String Fname , is the first name of the Admin Assistant. It is declared as a public data member.
- String Lname , is the last name of the Admin Assistant. It is declared as a public data member.
- void Updateprice(), is a method that enables the Admin assistant to update the price of the dishes. It is declared as a protected method.
- Date LastMaintenancedate is the date when last maintenance check was done, declared as a private data member. In addition to the abstract methods defined in employee interface, this class has certain other methods defined in it
- void Updatebranch(), is a method that enables the Admin assistant to update the branch of assistant. It is declared as a protected method.

## **Customer Class**

The customer class is a dependency of the class hotel. This is because this class has a method(Ordereddishes) that uses the object of Hotel class.

- orderedDishes: This is an array of strings which stores the Dishid of the dishes that were ordered by the customer
- ratings: This is a variable used to keep track of the number of stars out of five by the customer.
- void viewandPay(): This function is used to display and then pay the bill after which the revenue is updated in the database.
- int Orderddishes(Hotel obj): This function allows the customer to order their dishes.

## **Hotel Menu Class**

Menu Class is a part of hotel class, that is, the menu class exists only if the hotel class exists and hence the relationship between employee interface and hotel class is a composition. Since one hotel has only 1 menu, the multiplicity is 1 to 1.

- String Menuid , is the id of the menu. It is declared as a public data member.
- String array Availdishes , stores all the dishes in the menu as an string array. It is declared as a public data member.
- void Viewmenu, used to view the menu. Its declared as a public method.

## **Dishes Class**

The dishes class is an aggregated class from the menu. It's an aggregation as dishes is a part of the menu and can exist independently even if it is not in a menu.

- String Dishid , is the id of each of the dish. It is declared as a public data member.
- String Dishname , is the name of the dish in the menu. It is declared as a public data member.
- double Price is the rate of each dish in the menu. It is declared as a public data member.
- void Addish() , which allows the user to add a dish to the menu. It is declared as a protected method.

## 1. HOTEL TABLE

Column name	Datatype
Hid	Varchar(20)
Hname	Varchar(20)
Sdate	Date
Revenue	Numeric(50,2)
Expenditure	Numeric(50,2)
City	Varchar(20)
State	Varchar(20)
Last Maintenance	Date

## 2. EMPLOYEE TABLE

Column name	Datatype
SSN	Varchar(4)
Fname	Varchar(20)
Lname	Varchar(20)
DOJ	Date
Salary	Numeric(25,2)
Designation	Varchar(20)
Phone	Varchar(10)
Address	Varchar(50)

## 3. CHEF TABLE

Column name	Datatype
SSN	Varchar(4)
Speciality	Varchar(20)



#### 4. HOTELEMP TABLE

Column	Datatype
SSN	Varchar(4)
HID	Varchar(4)

#### 5. MENU TABLE

Column name	Datatype
Mid	Varchar(20)
Hid	Varchar(20)
qty	integer

#### 6. DISH TABLE

Column name	Datatype
DID	Varchar(20)
Dname	Varchar(50)
Price	Numeric(9,2)

#### 7. PART OF TABLE

Column name	Datatype
Mid	Varchar(20)
DID	Varchar(20)

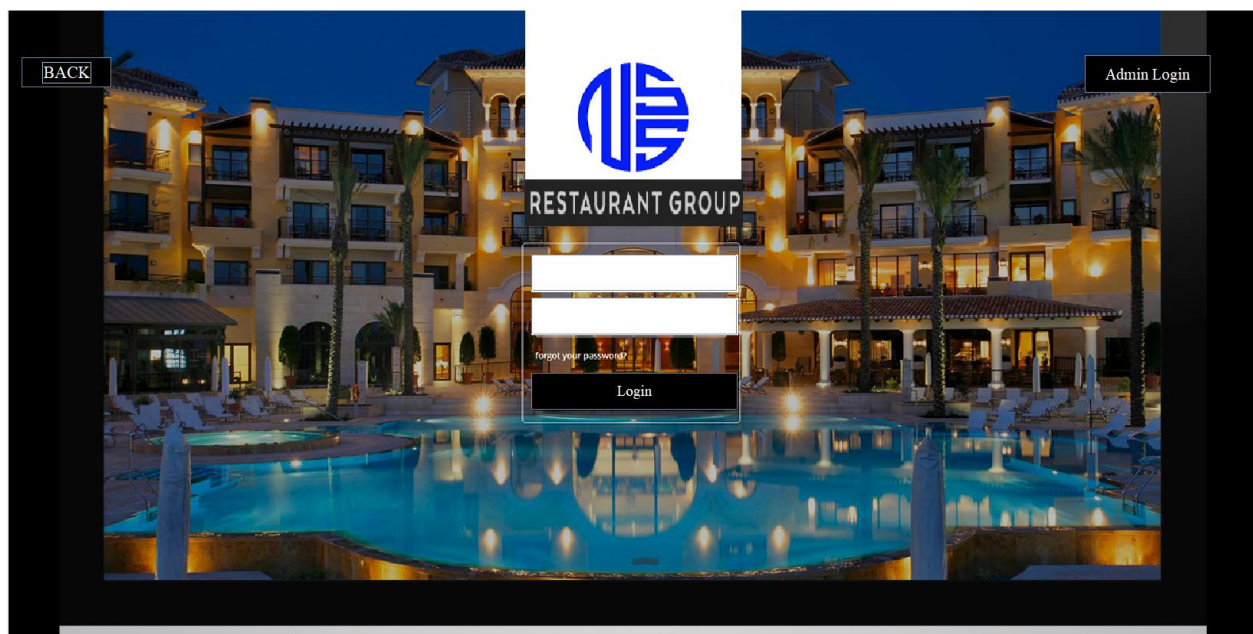
#### 8. COOKS TABLE

Column name	Datatype
DID	Varchar(20)
SSN	Varchar(20)

1. HOME PAGE



2. USER LOGIN PAGE



### 3. ADMIN LOGIN PAGE

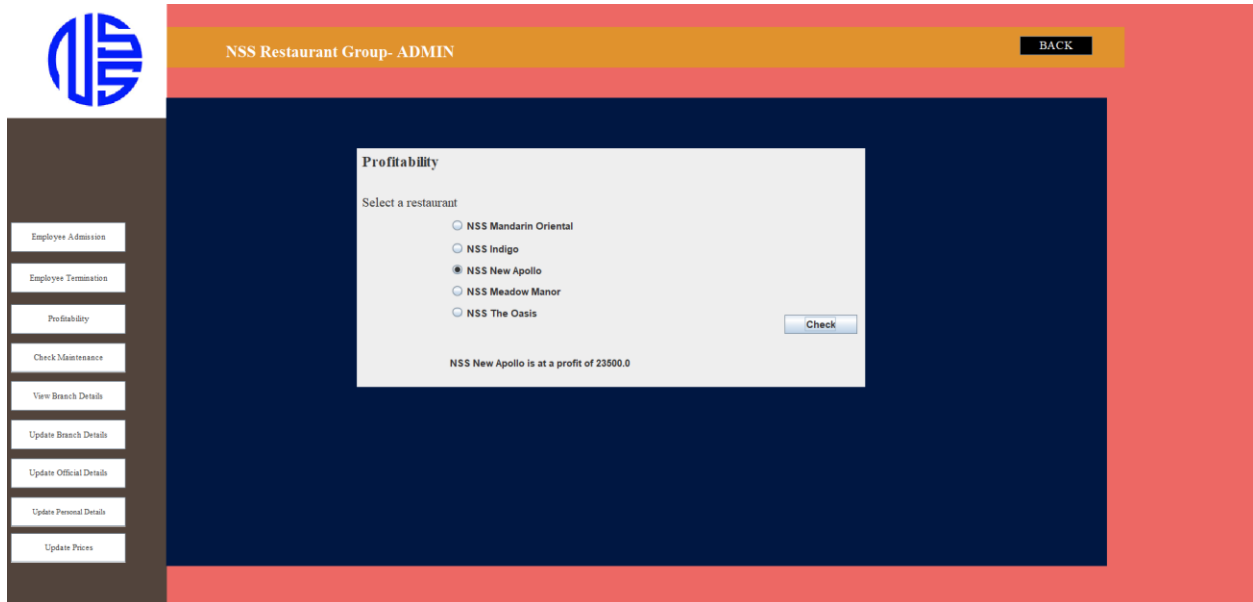


### 4. ADMIN VIEW

The image shows a screenshot of the admin view page for the NSS Restaurant Group. The page has a dark blue header with the NSS Restaurant Group logo on the left and the text 'NSS Restaurant Group- ADMIN' in the center. A black button with the word 'BACK' is on the right. Below the header is a dark blue sidebar with a list of menu items: 'Employee Admission', 'Employee Termination', 'Profitability', 'Check Maintenance', 'View Branch Details', 'Update Branch Details', 'Update Official Details', 'Update Personal Details', and 'Update Prices'. The main content area is white and contains a form titled 'Employee Admission'. The form has fields for 'SSN:', 'First Name:', 'Last Name:', 'Salary:', 'Designation:', 'Phone:', 'Address:', and 'Appointed Hotel id:'. An 'APPOINT' button is located at the bottom right of the form.

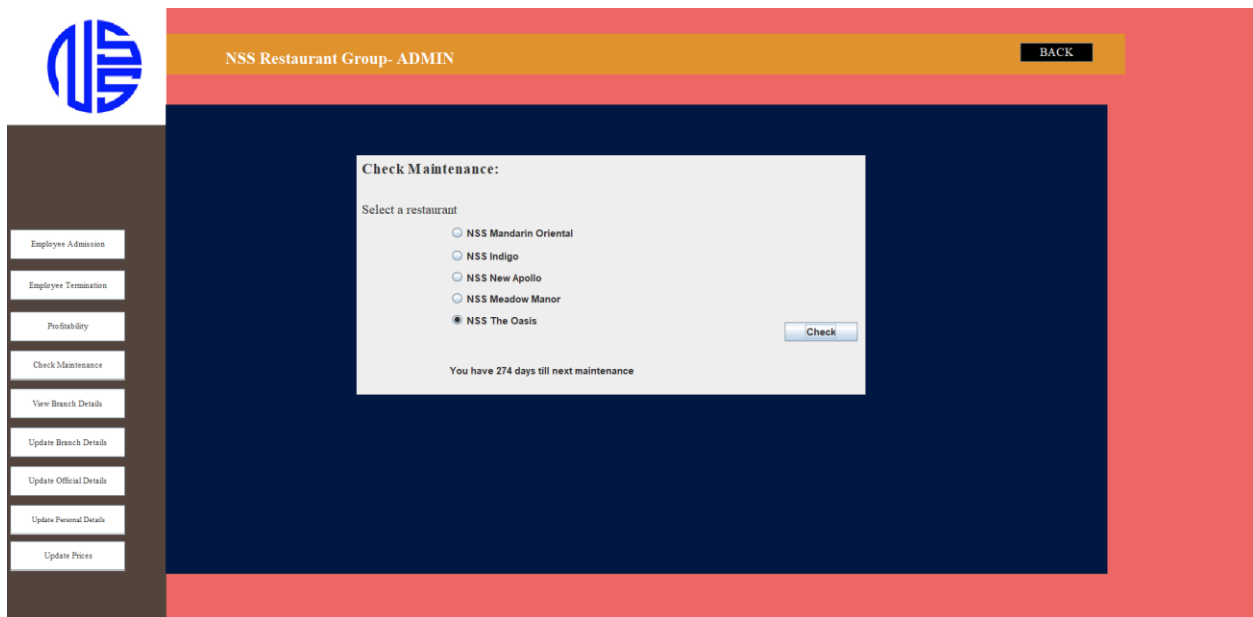
## ADMIN FUNCTIONALITIES:

### 4.1 PROFITABILITY



The screenshot displays the 'NSS Restaurant Group- ADMIN' interface. On the left is a vertical sidebar with a blue circular logo at the top and a list of menu items: Employee Admission, Employee Termination, Profitability, Check Maintenance, View Branch Details, Update Branch Details, Update Official Details, Update Personal Details, and Update Prices. The 'Profitability' item is highlighted. The main content area has a dark blue background and contains a white box titled 'Profitability'. Inside this box, it says 'Select a restaurant' followed by five radio button options: NSS Mandarin Oriental, NSS Indigo, NSS New Apollo (which is selected), NSS Meadow Manor, and NSS The Oasis. A 'Check' button is located to the right of these options. Below the options, it states 'NSS New Apollo is at a profit of 23500.0'. At the top right of the main area, there is a 'BACK' button.

### 4.2 LAST MAINTENANCE



This screenshot shows the same 'NSS Restaurant Group- ADMIN' interface as the previous one, but with the 'Check Maintenance' menu item highlighted in the sidebar. The main content area now displays a white box titled 'Check Maintenance:'. It contains the same 'Select a restaurant' section with five radio button options: NSS Mandarin Oriental, NSS Indigo, NSS New Apollo, NSS Meadow Manor, and NSS The Oasis (which is selected). The 'Check' button is still present. Below the options, it states 'You have 274 days till next maintenance'. The 'BACK' button remains at the top right.

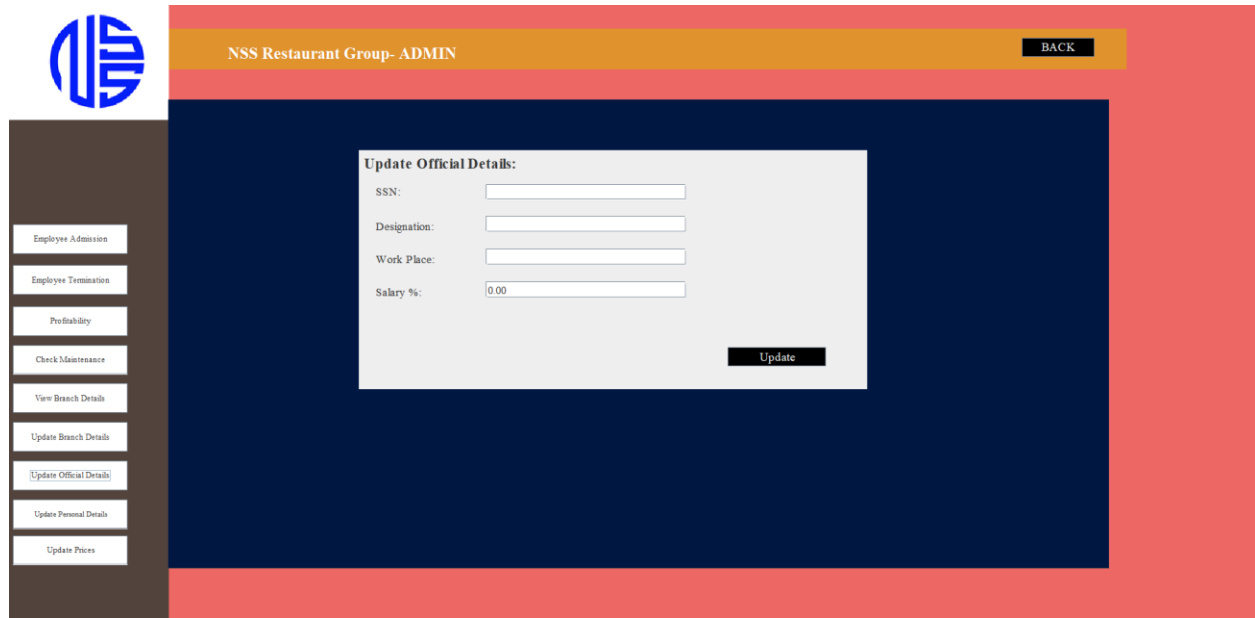
### 4.3 BRANCH DETAILS

The screenshot shows the 'View Branch Details' page. On the left is a sidebar with a logo and a menu of options: Employee Admission, Employee Termination, Profitability, Check Maintenance, View Branch Details (highlighted), Update Branch Details, Update Official Details, Update Personal Details, and Update Prices. The main content area has a header 'NSS Restaurant Group- ADMIN' with a 'BACK' button. Below this, the 'View Branch Details' section includes a 'Select a restaurant' dropdown with radio buttons for NSS Mandarin Oriental (selected), NSS Indigo, NSS New Apollo, NSS Meadow Manor, and NSS The Oasis. A 'View' button is to the right. Below the dropdown, the following details are listed: Hotel ID: H01, Name: NSS Mandarin Oriental, Start Date: 2016-12-07, Revenue: 61820.0, Expenditure: 35470.0, Location: Kochi, Kerala, and Last Maintenance on: 2021-04-22.

### 4.4 UPDATE BRANCH DETAILS

The screenshot shows the 'Update Branch Details' page. The sidebar is identical to the previous page. The main content area has the same header. The 'Update Branch Details' section includes the same 'Select a restaurant' dropdown with NSS Mandarin Oriental selected. Below this, there are two input fields: 'Updated Revenue:' and 'Updated Expenditure:', both with a value of '0'. An 'Update' button is located at the bottom right of the form area.

## 4.5 UPDATE OFFICIAL DETAILS



The screenshot displays the 'NSS Restaurant Group- ADMIN' interface. On the left is a vertical sidebar with a blue logo at the top and a list of menu items: Employee Admission, Employee Termination, Profitability, Check Maintenance, View Branch Details, Update Branch Details, Update Official Details (highlighted), Update Personal Details, and Update Prices. The main content area has a dark blue background. At the top of this area is a yellow header bar with the text 'NSS Restaurant Group- ADMIN' and a 'BACK' button. Centered on the page is a light gray form titled 'Update Official Details:'. The form contains four input fields: 'SSN:', 'Designation:', 'Work Place:', and 'Salary %:'. The 'Salary %' field is pre-filled with '0.00'. An 'Update' button is located at the bottom right of the form.

**Update Official Details:**

SSN:

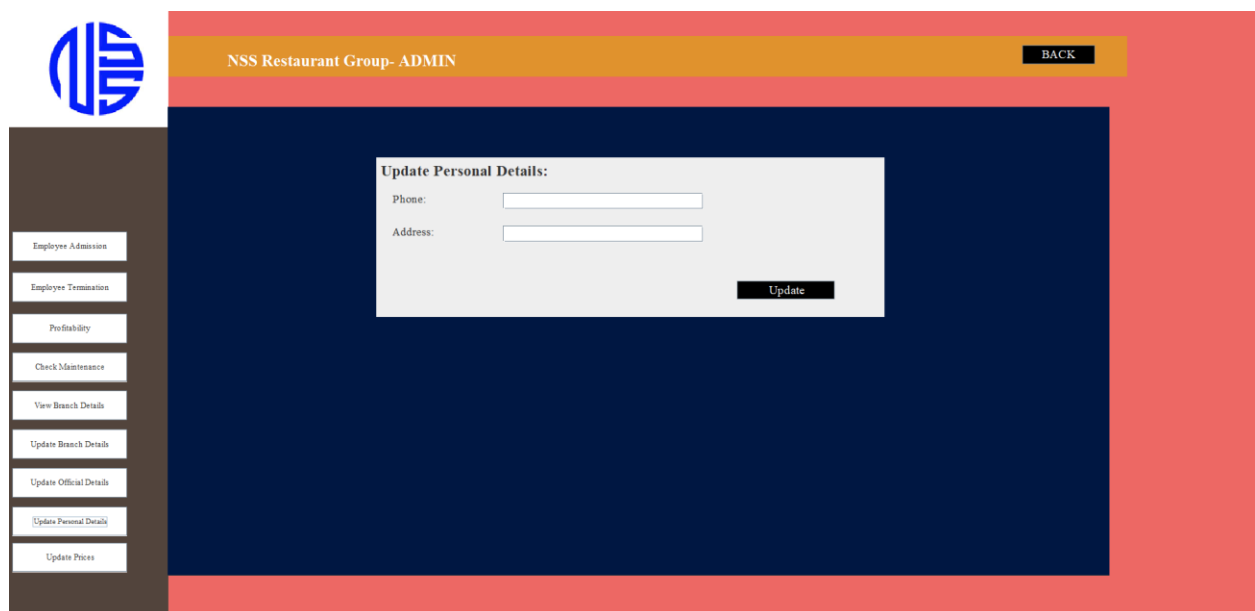
Designation:

Work Place:

Salary %:

**Update**

## 4.6 UPDATE PERSONAL DETAILS



This screenshot shows the same 'NSS Restaurant Group- ADMIN' interface as the previous one, but with the 'Update Personal Details' form displayed. The sidebar menu is identical, with 'Update Personal Details' highlighted. The main content area features the same yellow header bar and dark blue background. The light gray form titled 'Update Personal Details:' contains two input fields: 'Phone:' and 'Address:'. An 'Update' button is positioned at the bottom right of the form.

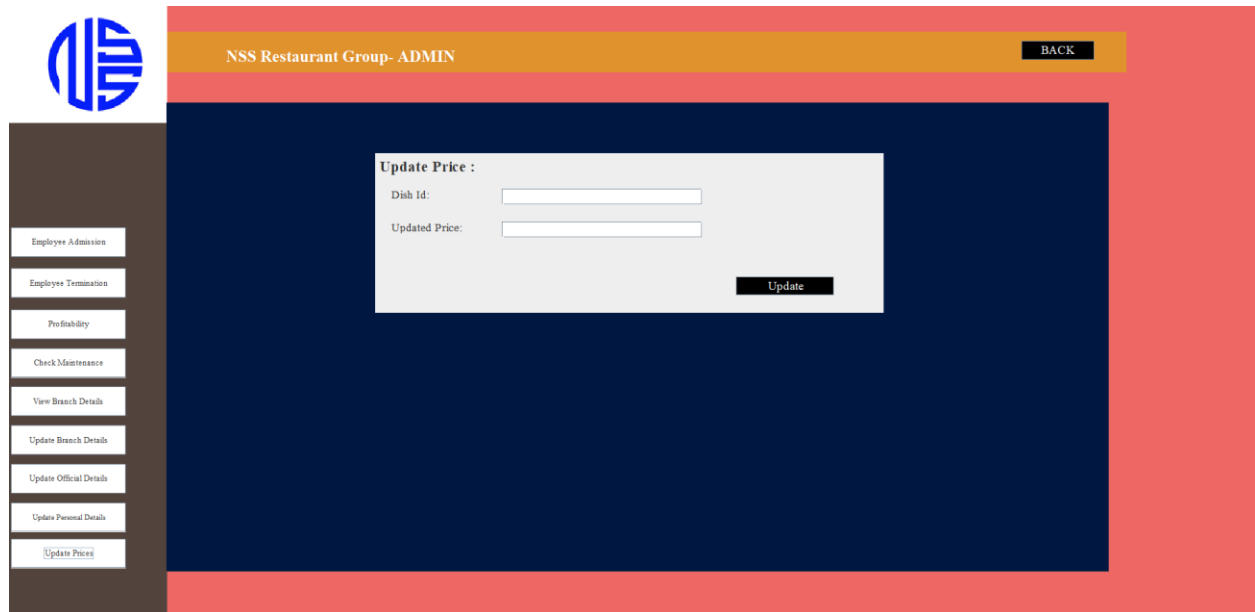
**Update Personal Details:**

Phone:

Address:

**Update**

## 4.7 UPDATE PRICE



The screenshot displays the 'NSS Restaurant Group- ADMIN' interface. On the left is a vertical sidebar with a blue logo at the top and a list of menu items: Employee Admins, Employee Termination, Profitability, Check Maintenance, View Branch Details, Update Branch Details, Update Official Details, Update Personal Details, and Update Prices. The main content area has a dark blue background. At the top of this area is a yellow header bar with the text 'NSS Restaurant Group- ADMIN' and a 'BACK' button. Centered in the main area is a light gray form titled 'Update Price :'. The form contains two input fields: 'Dish Id:' and 'Updated Price:'. Below these fields is an 'Update' button.

**NSS Restaurant Group- ADMIN** [BACK](#)

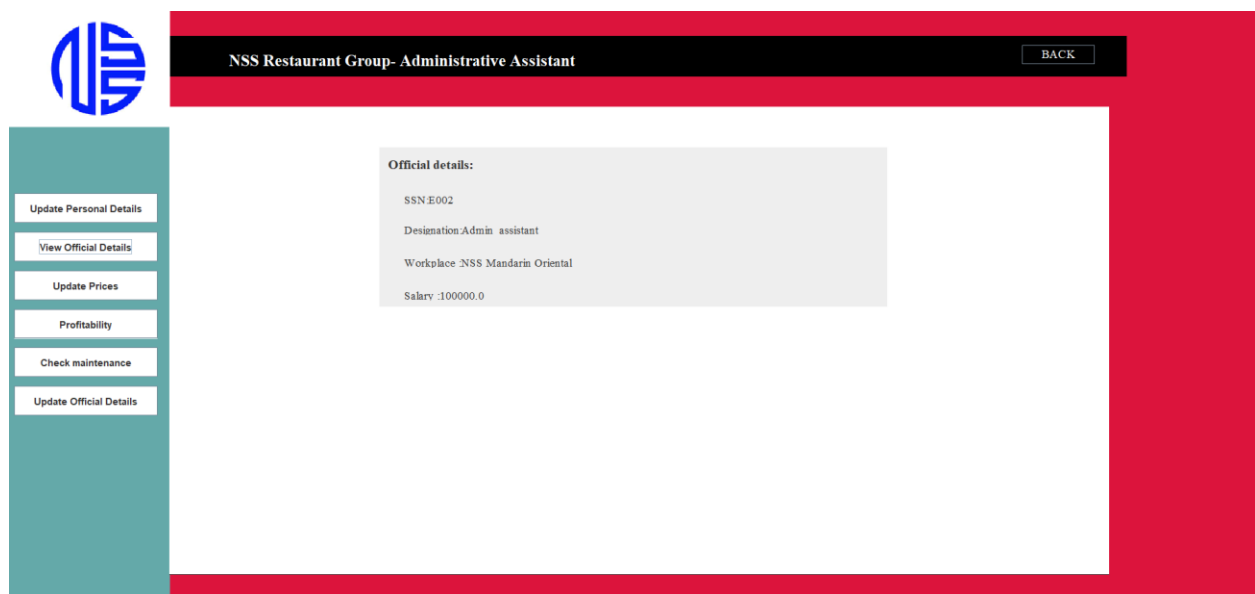
**Update Price :**

Dish Id:

Updated Price:

[Update](#)

## 5. ADMINISTRATIVE ASSISTANT VIEW



The screenshot displays the 'NSS Restaurant Group- Administrative Assistant' interface. On the left is a vertical sidebar with a blue logo at the top and a list of menu items: Update Personal Details, View Official Details, Update Prices, Profitability, Check maintenance, and Update Official Details. The main content area has a white background. At the top of this area is a red header bar with the text 'NSS Restaurant Group- Administrative Assistant' and a 'BACK' button. Centered in the main area is a light gray box titled 'Official details:'. Inside this box, the following information is displayed: SSNE002, Designation Admin assistant, Workplace NSS Mandarin Oriental, and Salary :100000.0.

**NSS Restaurant Group- Administrative Assistant** [BACK](#)

**Official details:**


SSNE002

Designation Admin assistant

Workplace NSS Mandarin Oriental

Salary :100000.0

## 7. BRANCH MANAGER VIEW



**NSS Restaurant Group- BRANCH MANAGER** **BACK**

Update Personal Details

View Official Details

Update Prices


**Update Price :**

Dish Id:

Updated Price:

Update

## 8. HEAD CHEF VIEW



**NSS Restaurant Group- HEAD CHEF** **BACK**

Update Personal Details

View Official Details

View Menu

Add Dishes

**Personal Details:**

Name: Sidharth Dinesh  
DOJ :2021-12-12  
Address :Kollam  
Phone :9916547521

**Update Personal Details:**

Phone:

Address:

Update




## 8.1 ADD DISHES

The screenshot shows a web application interface for a head chef. On the left is a vertical orange sidebar with a blue circular logo at the top and four buttons: 'Update Personal Details', 'View Official Details', 'View Menu', and 'Add Dishes'. The main content area has a blue background. At the top of this area is a dark grey header bar with the text 'NSS Restaurant Group- HEAD CHEF' and a 'BACK' button. In the center of the blue area is a white box titled 'Add dish'. Inside this box are three input fields labeled 'Dish id', 'Dish Name', and 'Dish Price', followed by an 'Add Dis...' button.

## 9. CHEF VIEW

The screenshot shows a web application interface for a chef. On the left is a vertical dark blue sidebar with a blue circular logo at the top and four buttons: 'View Branch Details', 'Update Personal Details', 'View Official Details', and 'View Menu'. The main content area has an orange background. At the top of this area is a black header bar with the text 'NSS Restaurant Group- CHEF' and a 'BACK' button. In the center of the orange area is a white box titled 'View Branch Details:'. Inside this box, the following information is displayed: 'Hotel ID: H01', 'Name: NSS Mandarin Oriental', 'Start Date: 2016/12/7', and 'Location: Kochi,Kerala'.

## 10. WAITER VIEW



**NSS Restaurant Group- Waiter** BACK

Update Personal Details

View Official Details

Review

Update Personal Details:

Phone:

Address:

Update

## 11. CUSTOMER VIEW



**NSS RESTAURANT GROUP** BACK

Select a restaurant

☐ NSS Mandarin Oriental

☐ NSS Indigo


☐ NSS New Apollo

☐ NSS Meadow Manor

☐ NSS The Oasis

Select

## 12. MENU



**NSS Restaurant Group- MENU**[BACK](#)

Number of wine bottles :

**ORDER DISHES**

Dish:  
mushroom risotto

Quantity:


[Add Item](#) [Place Order](#)

Your selected dish:

Quantity:

mushroom risotto	2
Alfredo Pasta	2

## 13. PAYMENT



**NSS Restaurant Group- Bill Payment**[Log Out](#)

**Ordered Dishes:**

Slno:	Dish Name:	Quantity:
1	Alfredo Pasta	2
2	mushroom risotto	2

**TOTAL AMOUNT :900.0**

[PAY BILL](#)

## CHAPTER 9

## TOOLS USED

**FRONT-END:** GUI – Java Swing components, Java, JDBC

**BACK-END:** PostgreSQL

This project is an excellent example of how we can use our technology to manage our daily needs to a large group of restaurants with ease and transparent. Hotel management system mainly facilitates the needs of each user to experience a easy and friendly approach within their tip of hands. In which Admin can manage the needs of each hotel instantly using this system rather than to visit each one. The customers with ease can order their respective dishes and pay for it without any delay, Employers can view their respective details and accredited menu transparently which is an important feature of the system. Currently the system works for limited number of users to work. In near future it will be extended for many users parallelly so that all users can experience an efficient application.

1. Developing Applications with Java™ and UML By Paul R. Reed Jr.
2. The Essence of Object Oriented Programming with Java and UML by Bruce E. Wampler, Ph.D.
3. Java How to Program, Early Objects, 11th Edition
4. Object-Oriented Analysis and Design with Applications Third Edition, Addison-Wesley Professional
5. Bryan Basham, Kathy Sierra ., Head First EJB ,O'Reilly Media; Second edition, 2008
6. Joel Murach, Michael Urban., JAVA servlets, Mike Murach, 2014
7. Java™2:The Complete Reference, Fifth Edition, Herbert Scheldt
8. Learn Java GUI Applications 11th Edition: A JFC Swing Tutorial, Philip Conrod, 2019
9. Database Programming with JDBC & Java, Second Edition, O'Reilly Media, Inc, 2000
10. JAVA GUI WITH POSTGRESQL: A Practical Approach to Build Database Project for Students and Programmers Kindle Edition, by Vivian Siahaan
11. <https://opus.govst.edu/cgi/viewcontent.cgi?article=1199&context=capstones>
12. <https://www.siteminder.com/r/trends-advice/hotel-management/hotel-management-definitions-operations-ideas-software/#hotel-management-software>
13. <https://pdfcoffee.com/hotel-management-system-10-pdf-free.html>
14. <https://code-projects.org/hotel-management-project-report-in-java-netbeans-ide-and-mysql-free-download/>
15. [https://www.academia.edu/30531515/Hotel\\_Management\\_System](https://www.academia.edu/30531515/Hotel_Management_System)

**Source Code:**

<https://github.com/sivasankarLM10/Hotel-Management-System>

\*\*\*\*\*