



Sri Lanka Institute of Information Technology

Hotel Management System for Hotel Aronway-Mawanella Project Report

Information Technology Project 2020

Project ID: **ITP_KDY_2020_WD05**

Submitted by:

1. IT19210520 – (M.R.U.M Senevirathna)
2. IT19152110 – (K.D Rajapaksha)
3. IT19207100 – (A.G.S.P Deshapriya)
4. IT19210698 – (J.M.V.L Jayasinghe)
5. IT19127538 – (A.R.W.M.B.W.D.B Jayawardana)
6. IT19207964 – (K.T.N Weerasooriya)
7. IT19211688 – (C.D Weerasinghe)
8. IT19242200 – (K.K.A.E.N Aththanayaka)
9. IT19213590 – (G.A.D.K.M Gardiarachchi)

23th of October 2020

Declaration

We declare that the this project report or part of it was not a copy of a document done by any organization, university any other institute or a previous student project group at SLIIT and was not copied from the Internet or other sources.

Project Details

Project Title	Hotel Management System
Project ID	ITP_KDY_2020_WD05

Group Members

Reg. No	Name	Signature
IT19210520	M.R.U.M Senevirathna	
IT19152110	K.D Rajapaksha	
IT19207100	A.G.S.P Deshapriya	
IT19210698	J.M.V.L Jayasinghe	
IT19127538	A.R.W.M.B.W.D.B Jayawardana	
IT19207964	K.T.N Weerasooriya	
IT19211688	C.D Weerasinghe	
IT19242200	K.K.A.E.N Aththanayaka	
IT19213590	G.A.D.K.M Gardiarachchi	

Abstract

Today, with the development of the hospitality industry around the world, the tendency of using hotel management systems has increased tremendously. This is because it allows hotels to easily and conveniently performs all the functions of a hotel from a guest reservation to an employee departure also to an importation of necessary supply and so on. Therefore, hotel management systems have become an integral part of hospitality operations and have yielded many beneficial results in day-to-day operations as a whole.

This project is developed for the hotel named ‘Aronway’ which is newly built with 15 rooms including a restaurant, bar and banquet facilities. Currently, they use a file-based manual system to execute their hotel operations on a daily basis and everything has to be recorded manually which requires a lot of effort. At the same time, they face issues such as wastage of time, cost and resources, data storage issues, employee information records, inventory information retention, overlapping reservations, and insecurity of sensitive data.

The system we have developed, helps to overcome all the challenges that our client currently faces as it includes the functions such as, inventory management, hall reservation, employee data handling and salary management, room reservation, order handling, payments handling, expenses recording, menus handling and travel package handling. And by that, the client can have the benefits of proper management of reservations, reduction of time spent on management functions, generating reports, efficient customer service and etc.

This system has been developed by using Java programming language and user interfaces were designed by using Java Swing frame work. At the same time, NetBeans was used as the integrated development environment and in order to design the database MySQL has been used.

The main objective of the entire project is to automate the process of day to day activities of hotel Aronway and this hotel management system will indeed help the hotel management and its esteemed staff members to manage and steer the hotel’s functionality effectively and efficiently while helping to maximize its competence in the hotel business field.

Acknowledgement

First and foremost, we would like to express our special gratitude towards our lecturer in charge of IT Project, Ms. Chathurika Koswatte, for her proper guidance, continues support and wealth of knowledge which has contributed immensely for every stage of this project.

Secondly, we would like to offer our special thanks to lecturer Ms. Chathurika Pinnaduwage, for her great assistance from the very beginning of this project. She gave us support whenever we needed it, and we are grateful for that.

Advice given by the supervisor in-charge of our IT Project group, Ms. Shanika Abeysinghe and her quick responses for our questions regarding the project, have been a great help in completion of this project, hence we would like to extend our thank to her as well.

Furthermore, we would like to thank the other academic and non-academic staff of Sri Lanka Institute of Information Technology, Kandy for providing us guidance, good environment and facilities to complete this assignment.

Moreover, we would like to thank our colleague who gave us their immense support through the entire project.

Finally, we convey our special thanks and appreciation to our family members for their support and understanding towards us in completing this project.

Table of Contents

1. Introduction.....	1
1.1 Problem Statement.....	2
1.2 Product Scope	4
1.2.1 Overall Scope	4
1.2.2 Individual Scope.....	5
1.3 Project Report Structure	14
2. Methodology	15
2.1 Requirements and Analysis	15
2.1.1 Inventory Management	17
2.1.2 Hall Reservation.....	18
2.1.3 Employee Handling and Salary Management	19
2.1.4 Room Reservations	20
2.1.5 Order Handling.....	21
2.1.6 Expences Handling.....	22
2.1.7 Payment Handling.....	23
2.1.8 Menus Handling.....	24
2.1.9 Travel Packeges Handling.....	25
2.2 Design.....	26
2.2.1 Class Diagram	26
2.2.2 ER Diagram.....	27
2.2.3 Sequence Diagrams	28
2.2.4 Communication Diagrams.....	38
2.2.5 State Chart Diagrams	47
2.2.6 User Intefaces.....	56
2.3 Implementation.....	72
2.3.1 Database Management System.....	72
2.3.2 Implementation Language.....	72
2.4 Testing	73
2.4.1 Inventory Management	73
2.4.2 Hall Reservation.....	75
2.4.3 Employee Handling And Salary Management	77
2.4.4 Room Reservation.....	79
2.4.5 Order Handling.....	81
2.4.6 Expences Handling.....	83
2.4.7 Payment Handling.....	85
2.4.8 Menus Handling.....	87
2.4.9 Travel Package Handling	89
3. Conclusion	91
3.1 Weakness and Limitation	91
4. References	92

List of Figures

<i>Figure 1: System Overview</i>	4
<i>Figure 2: Use Case Diagram.....</i>	16
<i>Figure 3: Activity Diagram of Generate Least Used Report</i>	17
<i>Figure 4: Activity Diagram of Generate Hall Reservation Report.....</i>	18
<i>Figure 5: Activity Diagram of Calculate Employee's Salary</i>	19
<i>Figure 6: Activity Diagram of Room Reservations.....</i>	20
<i>Figure 7: Activity Diagram of Add Order</i>	21
<i>Figure 8: Activity Diagram of Expences Handling</i>	22
<i>Figure 9: Activity Diagram of Update Payment Details</i>	23
<i>Figure 10: Activity Diagram of Add Menu</i>	24
<i>Figure 11: Activity Diagram of Travel Packages Handling.....</i>	25
<i>Figure 12: Class Diagram</i>	26
<i>Figure 13: ER Diagram</i>	27
<i>Figure 14: Sequence Diagram of Login</i>	28
<i>Figure 15: Sequence Diagram of Generate least used report</i>	29
<i>Figure 16: Sequence Diagram of Hall Reservation</i>	30
<i>Figure 17: Sequence Diagram of Calculate Employee's Salary</i>	31
<i>Figure 18: Sequence Diagram of Room Reservations.....</i>	32
<i>Figure 19: Sequence Diagram of Add Order</i>	33
<i>Figure 20: Sequence Diagram of Expences Handling</i>	34
<i>Figure 21: Sequence Diagram of Update Payment</i>	35
<i>Figure 22: Sequence Diagram of Menus Handling</i>	36
<i>Figure 23: Sequence Diagram of Travel Packages Handling</i>	37
<i>Figure 24 : Communication Diagram of Generate Lest Used Ingredient Report</i>	38
<i>Figure 25: Communication Diagram of Hall Reservation</i>	39
<i>Figure 26: Communication Diagram of Calculate Employee's Salary</i>	40
<i>Figure 27: Communication Diagram of Room Reservations</i>	41
<i>Figure 28: Communication Diagram of Add Order</i>	42
<i>Figure 29: Communication Diagram of Expences Handling</i>	43
<i>Figure 30: Communication Diagram of Update Payment</i>	44
<i>Figure 31: Communication Diagram of Menus Handling</i>	45
<i>Figure 32: Communication Diagram of Travel Package Handling</i>	46
<i>Figure 33: State Chart Diagram of Inventory Management</i>	47
<i>Figure 34 : State Chart Diagram of Hall Reservation</i>	48
<i>Figure 35: State Chart Diagram of Employee Handling and Salary Management</i>	49
<i>Figure 36: State Chart Diagram of Room Reservation</i>	50
<i>Figure 37: State Chart Diagram of Order Handling.....</i>	51
<i>Figure 38: State Chart Diagram of Expences Handling</i>	52
<i>Figure 39: State Chart Diagram of Payment Handling</i>	53
<i>Figure 40: State Chart Diagram of Order Handling.....</i>	54
<i>Figure 41: State Chart Diagram of Travel Package Handling</i>	55
<i>Figure 42: Home Page.....</i>	56
<i>Figure 43: Inventory Management Interface.....</i>	57
<i>Figure 44: Least Used Ingridients Interface</i>	57
<i>Figure 45: Running Out Ingridients Interface</i>	58
<i>Figure 46: Used History Interface.....</i>	58
<i>Figure 47: Hall Reservation Interface.....</i>	59
<i>Figure 48: Hall Adding Interface</i>	59
<i>Figure 49: Hall Reservation Report Generate Interface</i>	60
<i>Figure 50: Employee Handling Interface</i>	61
<i>Figure 51: Leaves Adding Interface</i>	61

<i>Figure 52: OT Adding Interface</i>	62
<i>Figure 53: Salary Generator Interface</i>	62
<i>Figure 54: Room Reservation Interface</i>	63
<i>Figure 55 : Delete / Update Room Interface</i>	63
<i>Figure 56 : Room Handling Interface</i>	64
<i>Figure 57: Room Checking Interface</i>	64
<i>Figure 58 : Order Handling Interface</i>	65
<i>Figure 59: Generate Report Interface</i>	65
<i>Figure 60: Expences Handling Interface.....</i>	66
<i>Figure 61: Expences Category Handling Interface.....</i>	66
<i>Figure 62: Payment Handling Interface.....</i>	67
<i>Figure 63: Payment History Handling Interface.....</i>	67
<i>Figure 64 : Update / Delete Menu Interface</i>	68
<i>Figure 65 : Search and Generate Report Interface</i>	68
<i>Figure 66: Menu Adding Interface</i>	69
<i>Figure 67: Travel Package Handling Interface.....</i>	70
<i>Figure 68: Travel Package Search Interface.....</i>	70
<i>Figure 69: Travel Package Generate Report Interface</i>	71
<i>Figure 70: Test Case of Used History deduct.....</i>	73
<i>Figure 71: Test Case of Storing State.....</i>	73
<i>Figure 72: Hall Reservation Passed Test Case</i>	75
<i>Figure 73: Hall Reservation Passed Test Case</i>	75
<i>Figure 74: Add Employee Passed Test Case</i>	77
<i>Figure 75: Add Employee Passed Test Case</i>	77
<i>Figure 76: Room Reservation Passed Test Case.....</i>	79
<i>Figure 77: Room Reservation Passed Test Case</i>	79
<i>Figure 78: Add Order Passed Test Case</i>	81
<i>Figure 79: Add Order Passed Test Case</i>	81
<i>Figure 80: Add Expences Passed Test Case</i>	83
<i>Figure 81: Add Expences Passed Test Case</i>	83
<i>Figure 82: Add Payment Passed Test Case</i>	85
<i>Figure 83: Add Payment Failed Test Case</i>	85
<i>Figure 84: Add Menu Passed Test Case</i>	87
<i>Figure 85: Add Menu Passed Test Case</i>	87
<i>Figure 86: Add Travel Package Passed Test Case</i>	89
<i>Figure 87: Add Travel Package Failed Test Case</i>	89
<i>Figure 88: Physical Diagram</i>	93
<i>Figure 89: OT Time Adding Passed Test Case</i>	95
<i>Figure 90: Menu Deleting Passed Test Case</i>	95
<i>Figure 91: Menu Updating Passed Test Case</i>	96
<i>Figure 92:Expences Deleting Passed Test Case</i>	96
<i>Figure 93: trigger of deduct used quantity from available quanitity of an ingridient- Inventory Management.....</i>	97
<i>Figure 94:trigger of increase vailable quantity of a ingridient according to an order-Inventory Management.....</i>	97
<i>Figure 95: trigger of detecting storing state – Inventory Management</i>	98
<i>Figure 96: Special Code - Hall Reservation.....</i>	98
<i>Figure 97:Special Code of Image Uploading - Employee Management</i>	99
<i>Figure 98: Special Code for Generate auto ID -Employee Management</i>	99
<i>Figure 99: Special Code for Calculate Total OT Time -Salary Management.....</i>	100
<i>Figure 100: Special Code for Calculate Net Salary -Salary Management</i>	100
<i>Figure 101:Special Code for Payment Calculation:Payment Management</i>	101

List of Tables

<i>Table 1: Table of Abbreviation</i>	vii
<i>Table 2: Individual Functions</i>	5
<i>Table 3: Test Cases for Inventory Management</i>	74
<i>Table 4: Test Cases for Hall Reservation</i>	76
<i>Table 5: Test Cases for Employee Handling and Salary Management</i>	78
<i>Table 6: Test Cases for Room Reservation</i>	80
<i>Table 7: Test Cases for Order Handling</i>	82
<i>Table 8: Test Cases for Expences Handling</i>	84
<i>Table 9: Test Cases for Payement Handling</i>	86
<i>Table 10: Test Cases for Menus Handling</i>	88
<i>Table 11: Test Cases for Travel Package Handling</i>	90

List of Acronyms and Abbreviations

Abbreviation	Description
DBMS	Database Management System
ER	Entity – Relationship Diagram
SQL	Structured Query Language

Table 1: Table of Abbreviation

1. Introduction

Hotel Aronway is one of the most popular hotels located in Mawanella area among both national and international guests. The reason behind its popularity is the location as there is a nice river flow by and there are very popular tourist attractions nearby. Of this hotel, there are 15 rooms in three different styles are available to meet the needs of any type of traveler, while the food in their restaurant and bar will satisfy any palette. As well as finely featuring banquet facility has become one of the main reasons behind its popularity. At present, there are 45 employees working under 05 main departments of the hotel.

Currently, Hotel Aronway is doing its every function manually, such as taking reservations, updating room availability/function hall availability, recording guest details, guest meal plan details, status of rooms (clean, vacant, out of order or any other), maintaining inventories, personnel management and even billing. So, they are using documents for every function and keeping/filing records. They have to put an extra effort to secure their records as they need a separate space to store them for a long time, protect them from any damages like fire, insects, or any other disaster and some of the information is confidential as well. At the same time, it wastes the most valuable thing in an organization which is man-hours.

Earlier it was said that the most significant part of the hotel industry is associated with the human touch or it is labor-intensive. But today it is realized that hotel management systems play a significant role as equal to its human touch. The system we will be developing helps to overcome all the challenges that our client is currently facing.

It helps the client in their every day to day function. It would handle all the front desk activities (guest profiles, room types, room/function hall availability, room status), reservation details, guest meal plans, inventory management, personnel management, cashiering and other accounting functions.

All of the above information will be stored in the system database and can be reverted whenever needed. As well as, the interface will be designed user-friendly and functions will be displayed in a simple manner and will ensure that easy to learn. Switching to our system will save time, manpower, space wastage and ensure smooth operation in day to day activities and ultimately will help to achieve organizational goals and objectives effectively and efficiently.

1.1 Problem Statement

- Time consuming

To find the relevant information. It will take much time to search documents in the cabinet. Sometimes some services go out next day such as delivery services so that customers not satisfied about poor service. Using this system, they can provide quick and good service to customers and they can improve their productivity.

- Higher cost and wastage

Every information in the company is stored in much papers so that we need to buy ink, papers, cabinets and cupboards to store that documents. That Documentation and record keeping process is very costly in manual system. Using this system, they can reduce their expenses.

- Record employees' details and attendance.

Using a file-based system have limited the ability of making changes on data. Employees' salaries calculate based on their attendance. Therefore, hotel's management must record attendance of every employees. Using our system, they can easily add an employee to the data base and maintain their details. Also, they can easily take attendance of employees and calculate their salaries by using our system.

- Calculate bills manually

Calculating bills manually take serious amount of time. We cannot always trust the accuracy of the manual calculation by reason of human errors. Moreover, keep track of income using manual bills is a challenging task. Our system will reduce employee's workload and will provide a reliable income report at the end of the month.

- Keep track of inventories.

Inventory records are the most essential things to maintain in a Hotel system. Using a file-based system to keep track of inventories produce miscalculations. Our system will provide user friendly interfaces to manage all the inventories in effective manner. Moreover, our system will detect items that are needed to be restored and will notify the corresponding employee.

- Reservation overlap.

Current system of the hotel constantly leads to a reservation overlap among deferent customers due to poor data handling. Using our system, they can manage customers' reservations in effective manner.

- Lack of security

Employee's data and Management data was being misplaced. That misplaced documents can easily take into wrong hands. If lost the documents somewhere, there is no chance to get them back. We are at risk of using manual system. By using our system, they can prevent unauthorized access to company information.

1.2 Product Scope

1.2.1 Overall Scope



Figure 1: System Overview

1.2.2 Individual Scope

Table 2: Individual Functions

	Name with Initials	Brief Description of the Function
1.	M.R.U.M Senevirathna	<p>Inventory management. The purpose of this function is to manage Items and ingredients in the inventory. The function mainly focuses on adding, updating, and removing Item details from the system. As an additional feature of the function, helps to detect items which are running out of the stock currently using available quantity of the items. Also function records items used history with used quantity and date. Moreover, if user enter a received supply order for a certain item into the system by using order handling function, the quantity of the item will auto update according to the order.</p> <p>CRUD Operations</p> <ul style="list-style-type: none"> • Add item details. • Update item details • Delete item details. • Calculate new quantity and automatically update inventory table according to the Order. <p>Reports</p> <ul style="list-style-type: none"> • As a report user can get a list of running out items with available quantity in the stock. • User can get a report of least used ingredients in a month

2.	K.D Rajapaksha	<p>Hall reservation. Hotel has halls which are used to held events. Through this function user should be able to handle these halls reserving. By using this function, user should be able to add a reservation for a hall, delete a reservation and update a reservation. Also there have search function to check availability by entering a date. Also, function calculate total amount of payment that customer needed to pay when reserve the hall using number of participants, menu amount and amount for the hall.</p> <p>CRUD Operations</p> <ul style="list-style-type: none"> • Add a hall reservation. • Delete a hall reservation. • Update a hall reservation. • Search available rooms. • Calculate total amount. <p>Reports</p> <ul style="list-style-type: none"> • At end of the month user can get report about “most reserved hall in month”.
----	----------------	--

3.	A.G.S.P Deshapriya	<p>Employee handling and salary management. This function holds all the details about employees. If a new employee joins to the hotel as permanent, admin (manager or owner) can insert his or her details to the systems by using this function. Inserting details are employee id, employee name, employee NIC, date of birth, basic salary etc. Apart from inserting, user also can update and delete employee details by using features of this function.</p> <p>For salary management part, if employee need to get a leave, user can check their available leaves and just deduct it from that employee's eligible leaves.</p> <p>Also, user can include bonus salary, OT hour, OT salary and user can deduct some amount for the ETF, EPF.</p> <p>CRUD Operations</p> <ul style="list-style-type: none"> • Insert employee details. • Update employee details. • Delete employee details. • Search employee available leaves. • Insert OT details. • Search salary details needed for salary calculation. • Insert other Salary details and calculate salary. <p>Reports</p> <ul style="list-style-type: none"> • Salary report which is about the how much pay for salaries per month. • Pay sheet, which is about the employee salary including their leaves, OT hour, OT salary, bonus salary, Basic salary etc.
----	--------------------	--

4.	J.M.V.L Jayasinghe	<p>Rooms reservations. Customers can reserve rooms and those detail add to the system. Also, by using the function user can update and delete reservation details. Also, when a customer who reserved a room, used an additional feature like foods, there is a way to store those details as well. This function also calculates total amount of room reservation as well. Also, user can search available rooms by using this function</p> <p>CRUD Operations</p> <ul style="list-style-type: none"> • Add a room reservation. • Update a room reservation • Delete a room reservation. • Search available rooms. <p>Reports</p> <ul style="list-style-type: none"> • User can get a report which include details of lowly reserved room types of the month.
----	--------------------	--

5.	A.R.W.M.B.W.D.B Jayawardana	<p>Orders handling. The function should be able to add Items ordered details to the database, update ordered details and delete ordered details. Also, should notify the order details before the receiving date. And should be able to include states where received or not to the system. By entering a time period on the search bar user can retrieve order details for that specific period. Other than that, the order handling function stores the details of suppliers.</p> <p>CRUD Operations</p> <ul style="list-style-type: none"> • Add order details. • Update order details. • Delete order details. • Search order details. <p>Reports</p> <ul style="list-style-type: none"> • The system will generate a report of the total order cost and most expensive order for each month.
----	--------------------------------	---

6.	K.T.N Weerasooriya	<p>This function is about Expences handling. It includes two table. One table is expenses table. Other one is income table. Expenses table is about the details of monthly expenses in the hotel. income table is about the details of monthly income in the hotel. It also will generate report of month is profitable or not. Admin is the one who handle above function. He can add, delete edit and update. This function is along with payment, salary management and order handling. Salaries and orders details have taken by expenses table automatically and the customer payment details have taken by income table automatically. Admin can add other expenses details that are not included in the order and salary management function and the also admin can add other income details that are not included in the function.</p> <ul style="list-style-type: none"> • CRUD Operations • Add additional expenses to the system. • Update Accounting table. • Delete mistakenly added records. • Search income for a day. <p>Reports</p> <ul style="list-style-type: none"> • User can get a report which include details about a certain month, whether it is profitable or not by comparing total expenses and total income.
----	--------------------	---

7.	C.D Weerasinghe	<p>Payments handlings. This function will handle all the payment information of the system. This will allow user to do all the normal operations such as insert, update and Delete. This function will automate all the billing processors and calculate the total cost of the bill for user. Moreover, this function will provide a search function to retrieve payment histories by entering a date.</p> <p>CRUD Operations</p> <ul style="list-style-type: none"> • Insert payment information. • Update payment details. • Delete payment details. • Calculate the total cost of bill. <p>Reports</p> <ul style="list-style-type: none"> • At the end of the month user can get a report that shows “Best source of income of the month”, which generated by comparing incomes from Rooms, Halls and Travel packages.
----	-----------------	---

8.	K.K.A.E.N Aththanayaka	<p>Menus handling. If a customer reserved a hall for an event, customer could choose a one of food menu that available in the menu list. The menu includes price and some dishes. Function also must be able to add new menu to the list, delete a menu and update a menu. Also for the room reservations there are different menus besides menus for hall reservations. There are several sections in the menu selection. They are breakfast, lunch, and dinner. There are separate sub sections for each section, for example Only vegetables, chicken, beef, pork, and everything else are on the menu.</p> <p>User can search menu by name and get all dishes which were included in each menu.</p> <p>CRUD Operations</p> <ul style="list-style-type: none"> • Add a new menu to the list. • Update a menu. • Delete a menu. • Search a menu <p>Reports</p> <ul style="list-style-type: none"> • At the end of the month and year user can generate report which include most prepared menu of customers by considering count of each menu selected.
----	---------------------------	---

9.	G.A.D.K.M Gardiarachchi	<p>Travel packages handling. This function is used to store travel information of customers to the system. This function should be able to insert travel details, update and delete those details. In this function, Travel starting date, Travel ending date, how many participants per travel, Guider id (Employee id) from Employee function, Customer Name and Email from reservation function are taken from customer. There is a dropdown menu which is used to get travel location. According to travel location, travel charge will be calculated.</p> <p>CRUD Operations</p> <ul style="list-style-type: none"> • Insert travel details. • Update travel details. • Delete travel details. • Calculate travel charge. • Calculate Monthly Revenue. <p>Reports</p> <ul style="list-style-type: none"> • There are two reports in this function. First one is the most frequently traveled place by customers. Using this function, hotels can identify what the most favorite traveling place is among customers. Then the hotel can charge more than other places. Next report is what place earns the highest income. From this report the hotel can estimate monthly earnings only from the travel package function.
----	----------------------------	--

1.3 Project Report Structure

The rest of this report is deliberately lined up in order to provide a precise idea about the hotel management system. Initially, as a sub topic of Methodology, identified requirement are pointed out with a use case diagram which were created using the information gathered in the requirement analysis phase. After that individual use cases are analyzed in detail through the medium of activity diagram for each function.

Secondly, Under the design topic, utilize of object-oriented concepts is denoted using a class diagram. The design of the database is exhibited using an ER diagram. The design of each individual function is demonstrated with the aid of sequence diagram, communication diagram, state diagram and the snapshots of user interfaces.

As the third phase of this report, choice of database management system and reasons behind chosen implementation language are briefly explained. Afterward, performed test case of each function are demonstrated using tabular format and screen shots captured during the execution of each test cases. Finally, the entire report ceases with Conclusion, References and Appendix. A summary of the project is noted down under conclusion and a physical diagram of the system along with some additional test cases, calculations, and some screen captured of special codes are exemplified under the subtopic Appendix.

2. Methodology

2.1 Requirements and Analysis

As the first step in the SDLC we decided to gather the required information, we needed to design our system.

We visited our client and we were able to interact with many of the employees working at the restaurant and gather the required information.

To gather the information, we used 2 requirements gathering methods,

- Interviews
- Observations

Since it was initially decided to develop the system according to departments, we were given the opportunity to interview the heads of the departments and get a thorough understanding about the flow of each department. After the interviews we decided to gather information by observation, we were given permission to observe the work flow of the company and get a solid understanding about how the company works and how each department is important for a successful work flow.

After a few brainstorming sessions with our team members we were able to eliminate all the unnecessary requirements and clear all the doubts we had about the unclear requirements.

And then we moved into the documenting phase of requirements gathering.

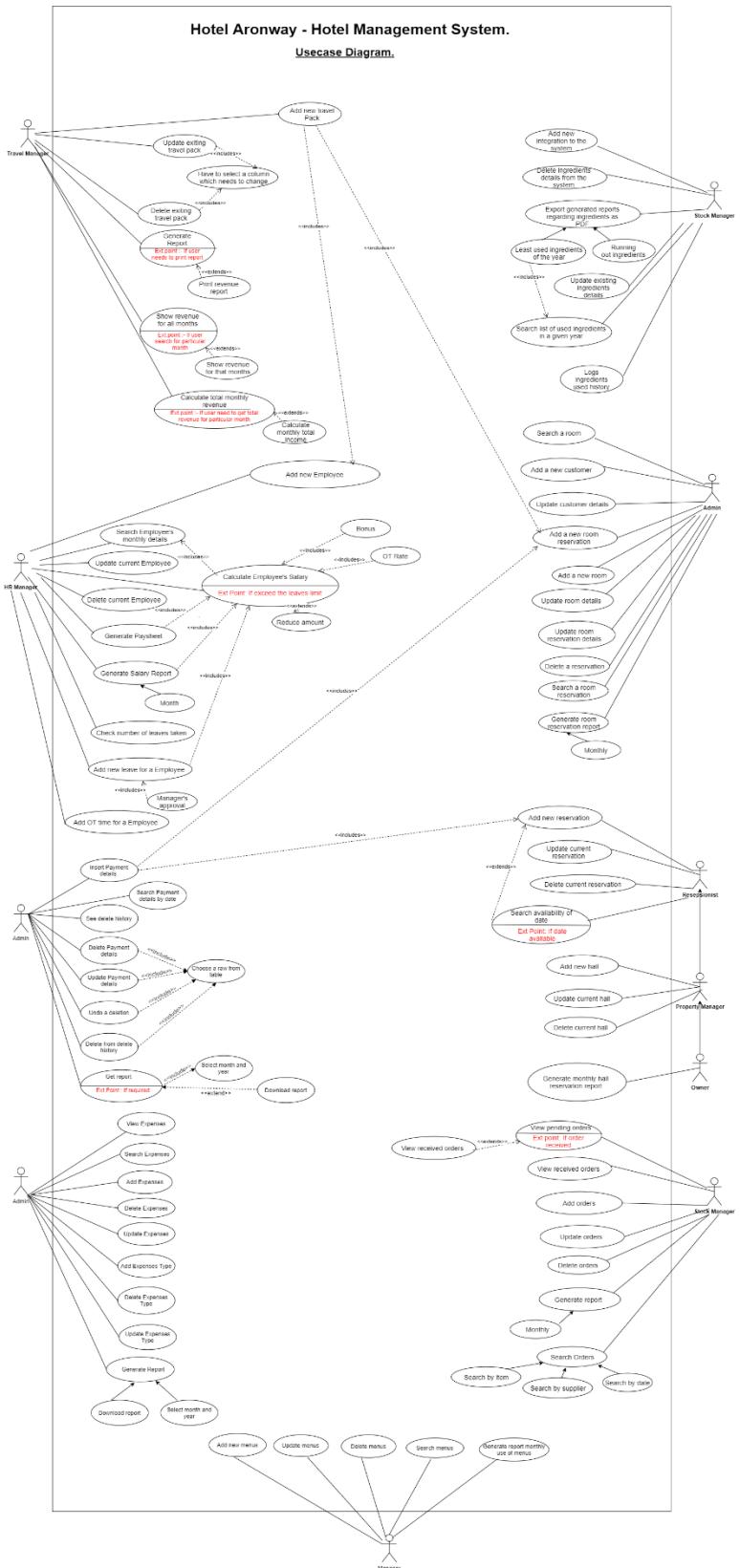


Figure 2: Use Case Diagram

2.1.1 Inventory Management

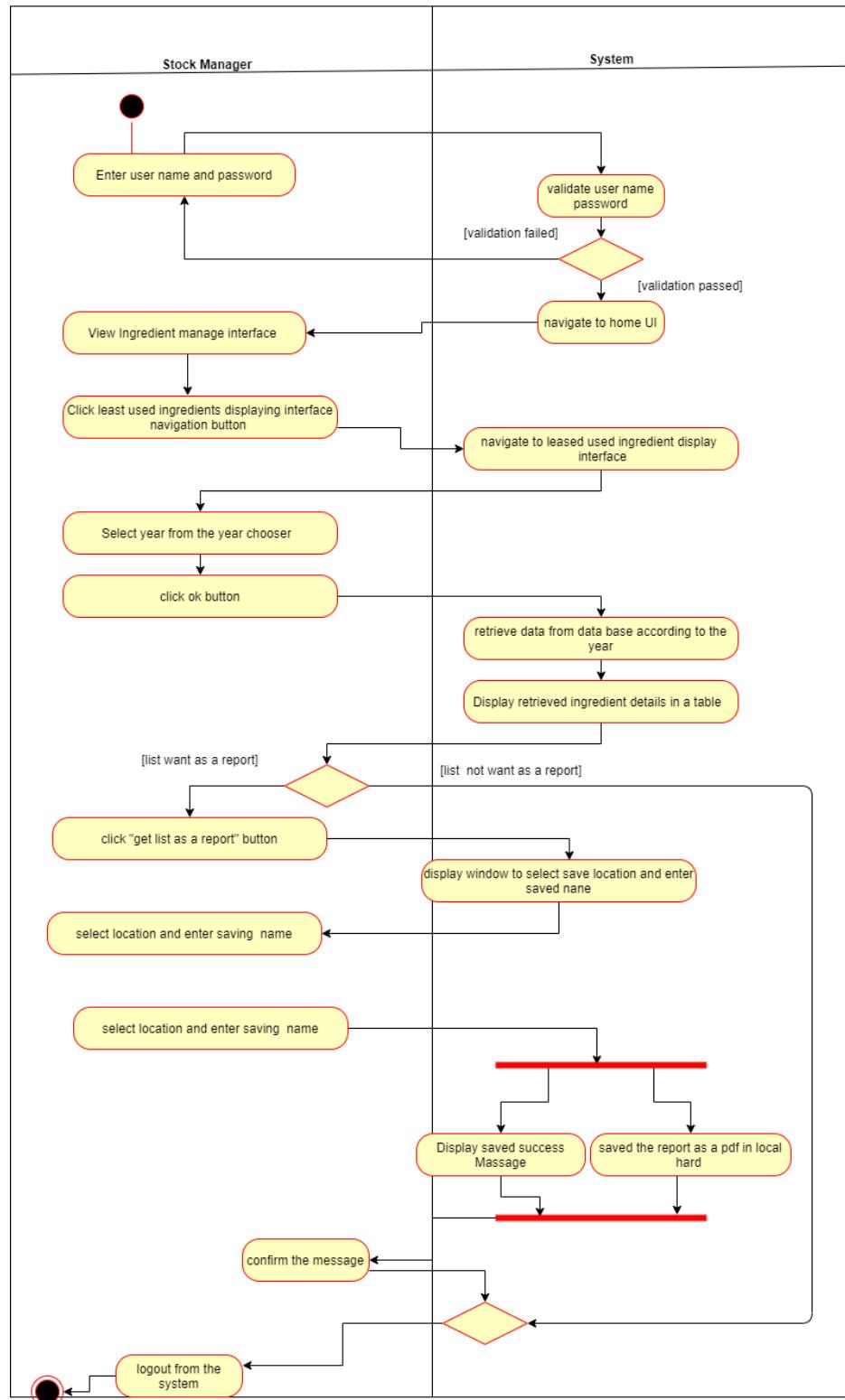


Figure 3: Activity Diagram of Generate Least Used Report

2.1.2 Hall Reservation

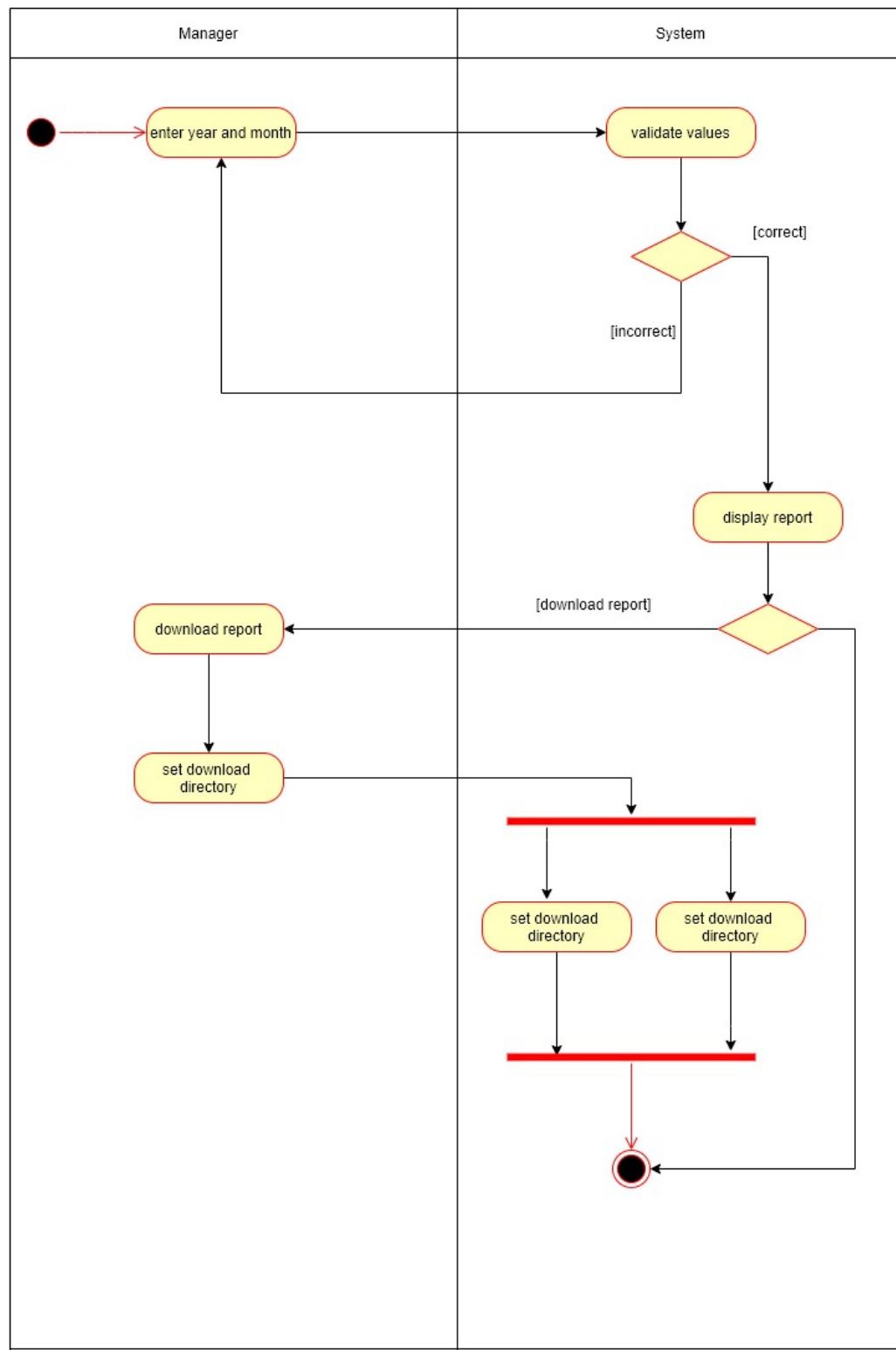


Figure 4: Activity Diagram of Generate Hall Reservation Report

2.1.3 Employee Handling and Salary Management

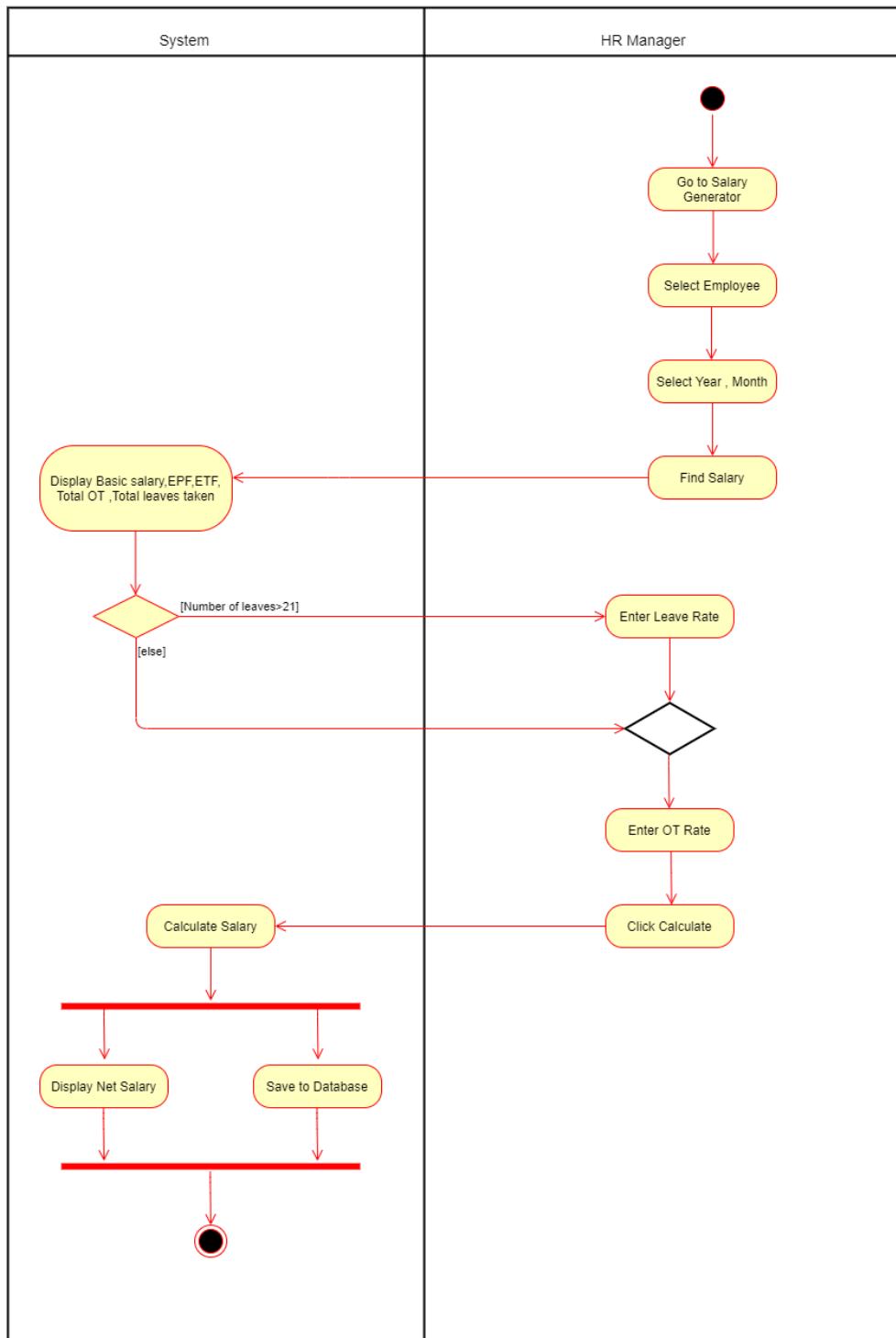


Figure 5: Activity Diagram of Calculate Employee's Salary

2.1.4 Room Reservations

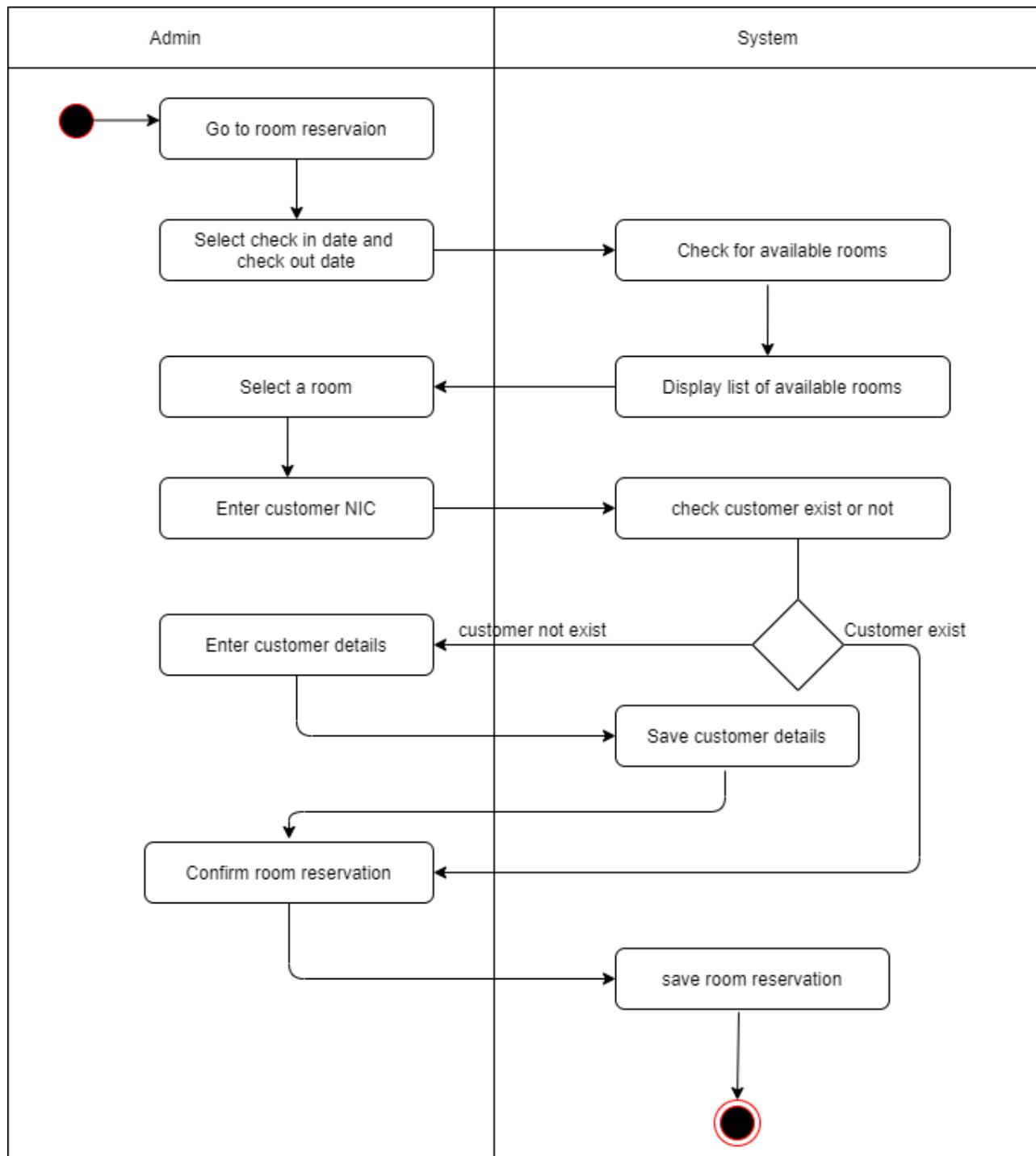


Figure 6: Activity Diagram of Room Reservations

2.1.5 Order Handling

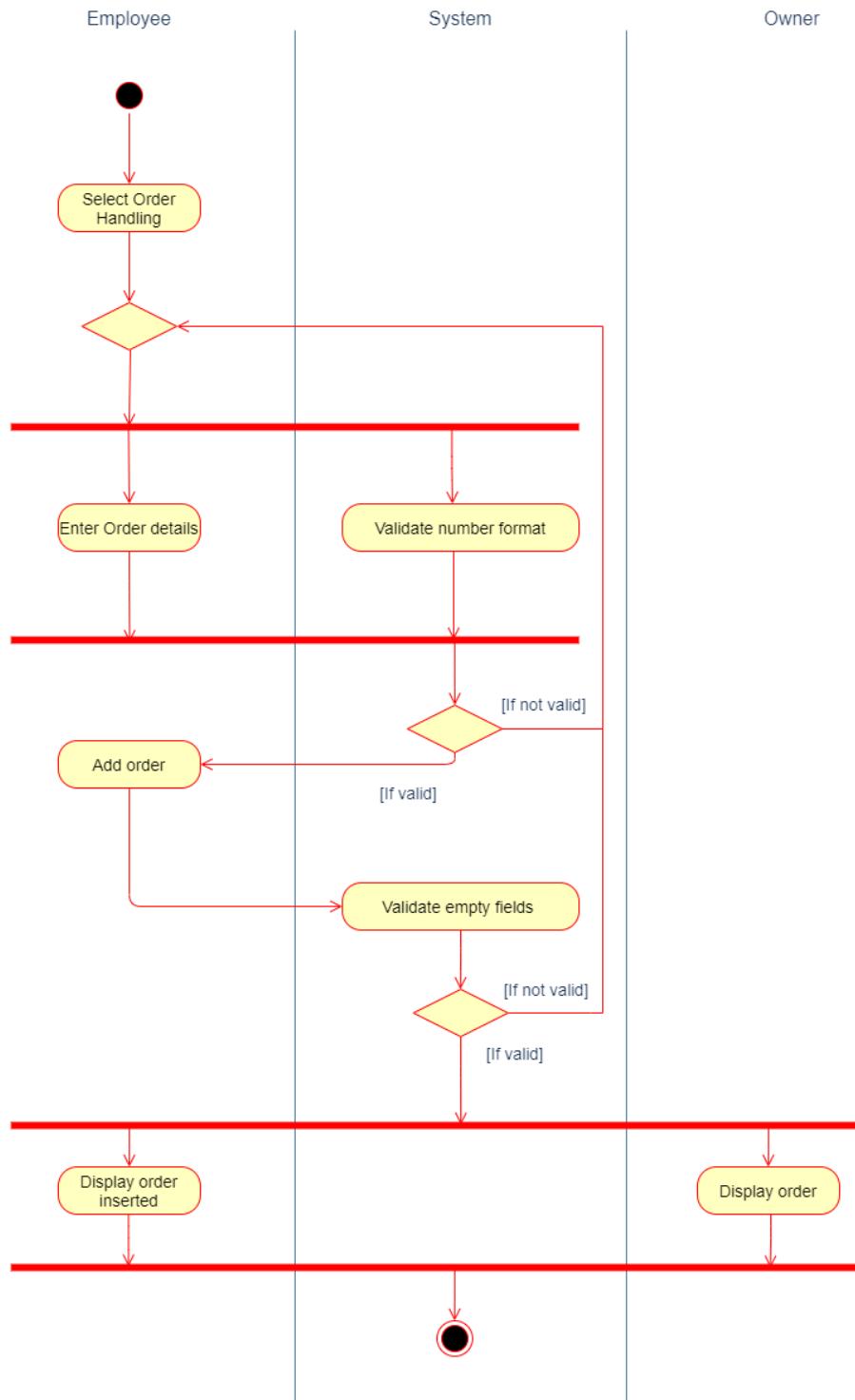


Figure 7: Activity Diagram of Add Order

2.1.6 Expences Handling

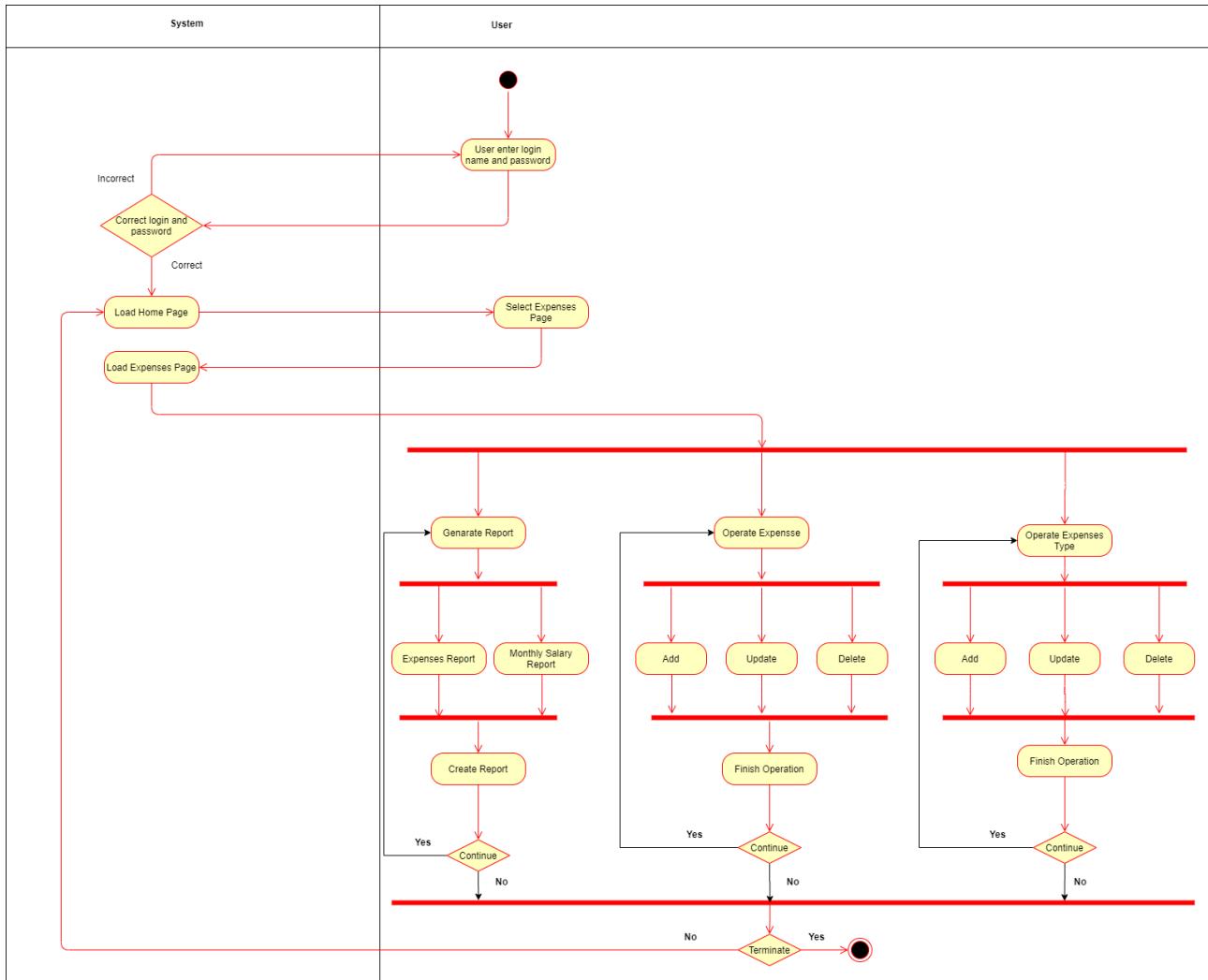


Figure 8: Activity Diagram of Expences Handling

2.1.7 Payment Handling

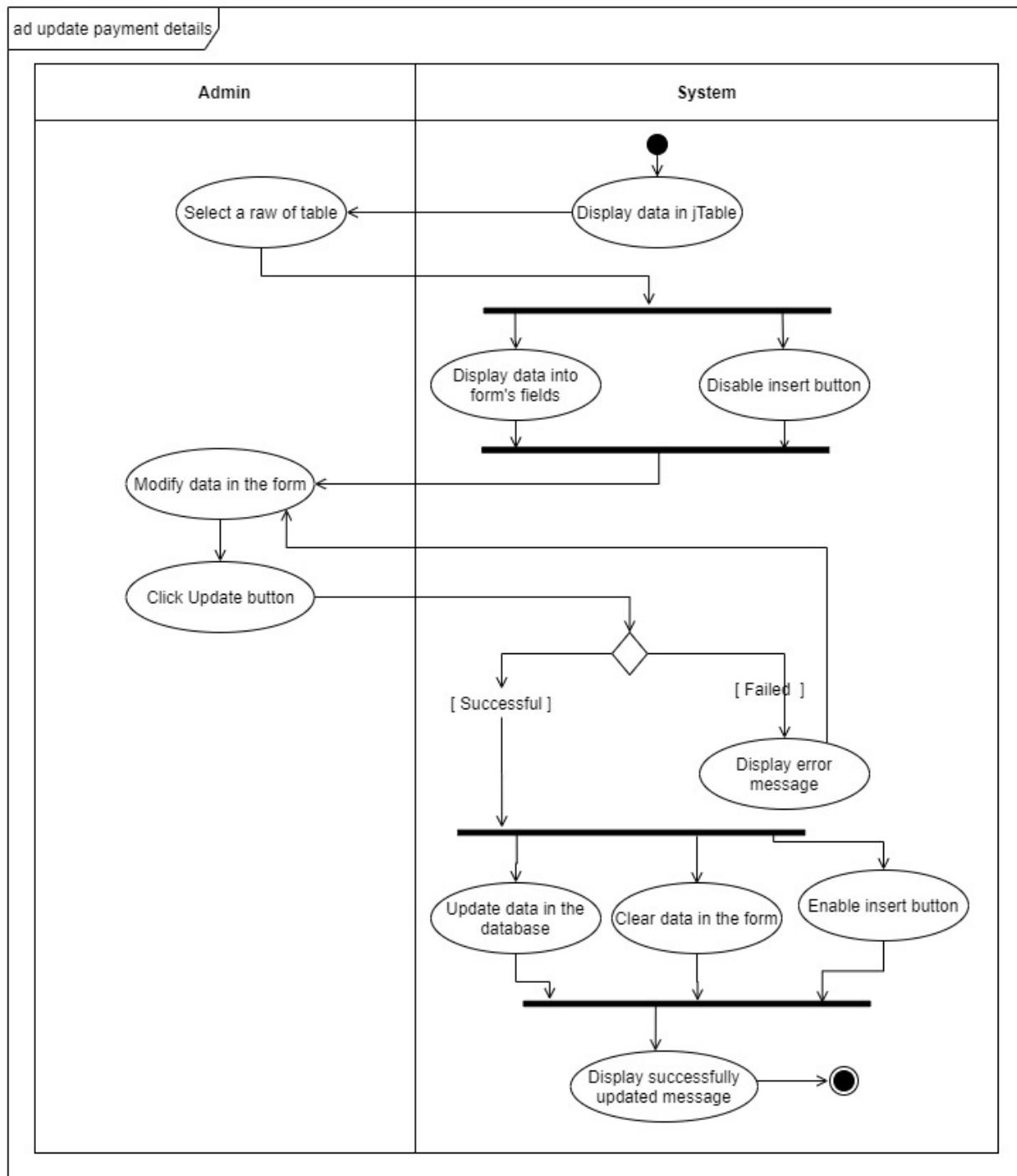


Figure 9: Activity Diagram of Update Payment Details

2.1.8 Menus Handling

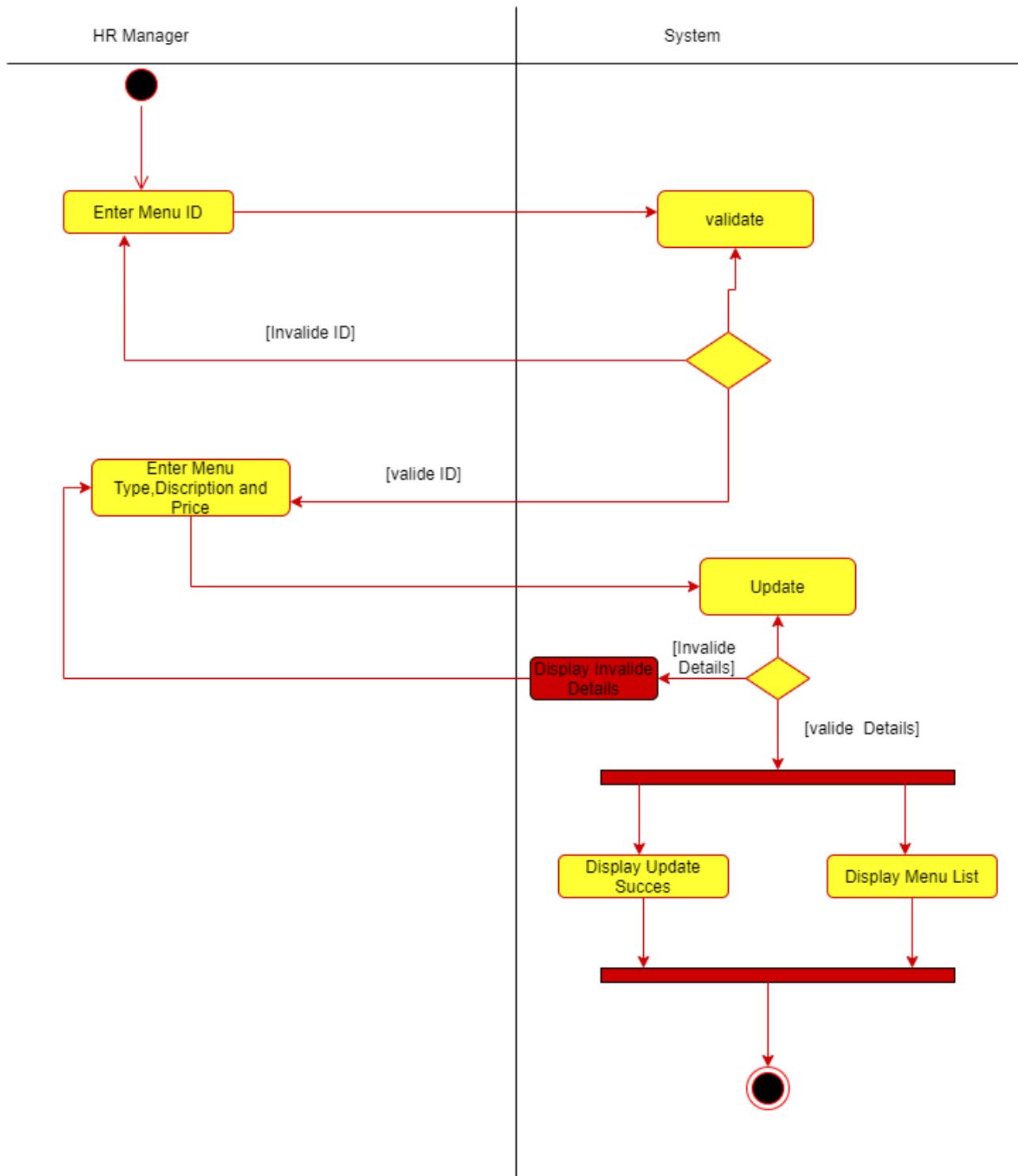


Figure 10: Activity Diagram of Add Menu

2.1.9 Travel Packages Handling

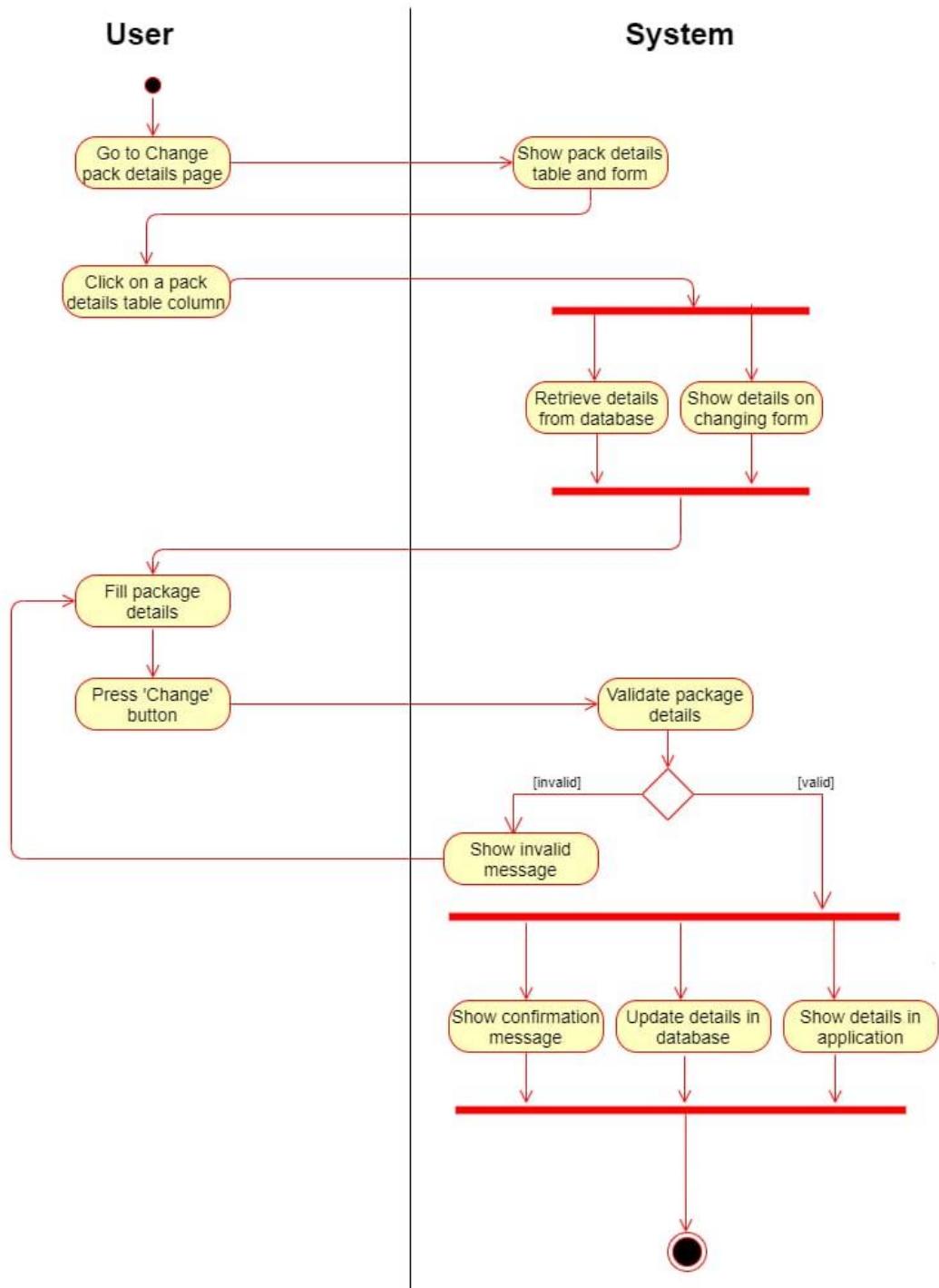


Figure 11: Activity Diagram of Travel Packages Handling

2.2 Design

2.2.1 Class Diagram

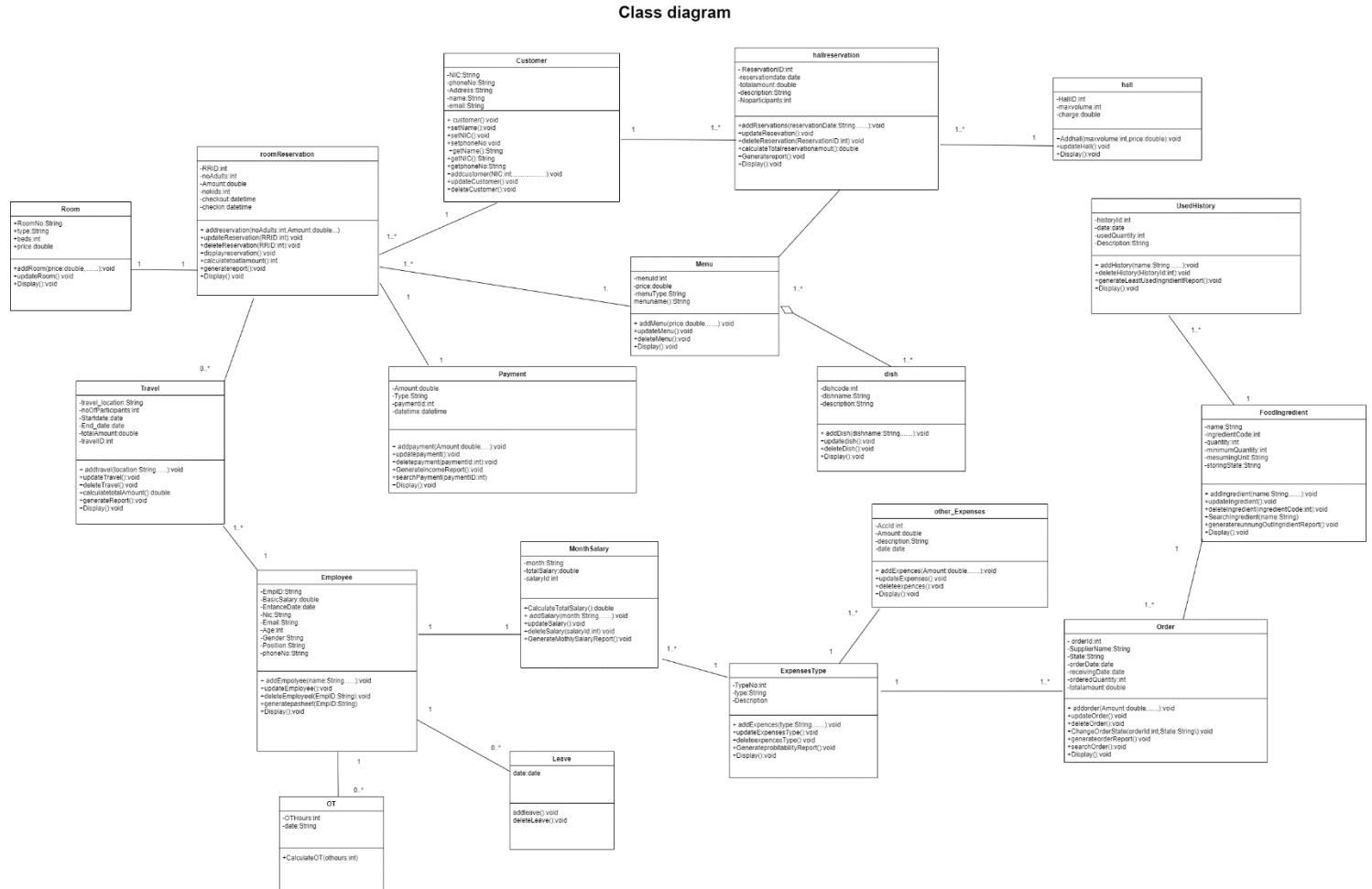


Figure 12: Class Diagram

2.2.2 ER Diagram

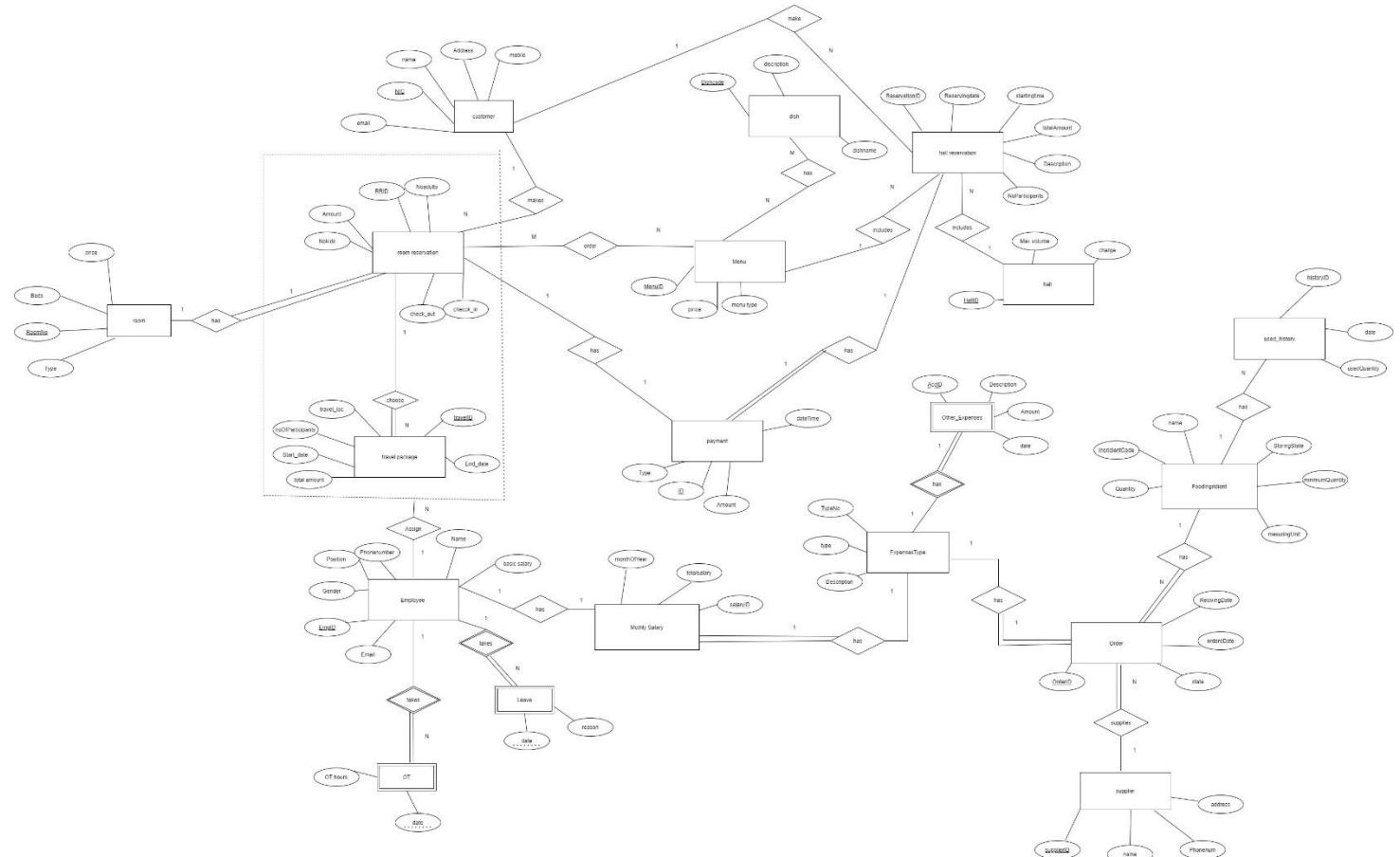


Figure 13: ER Diagram

2.2.3 Sequence Diagrams

2.2.3.1 Inventory Management

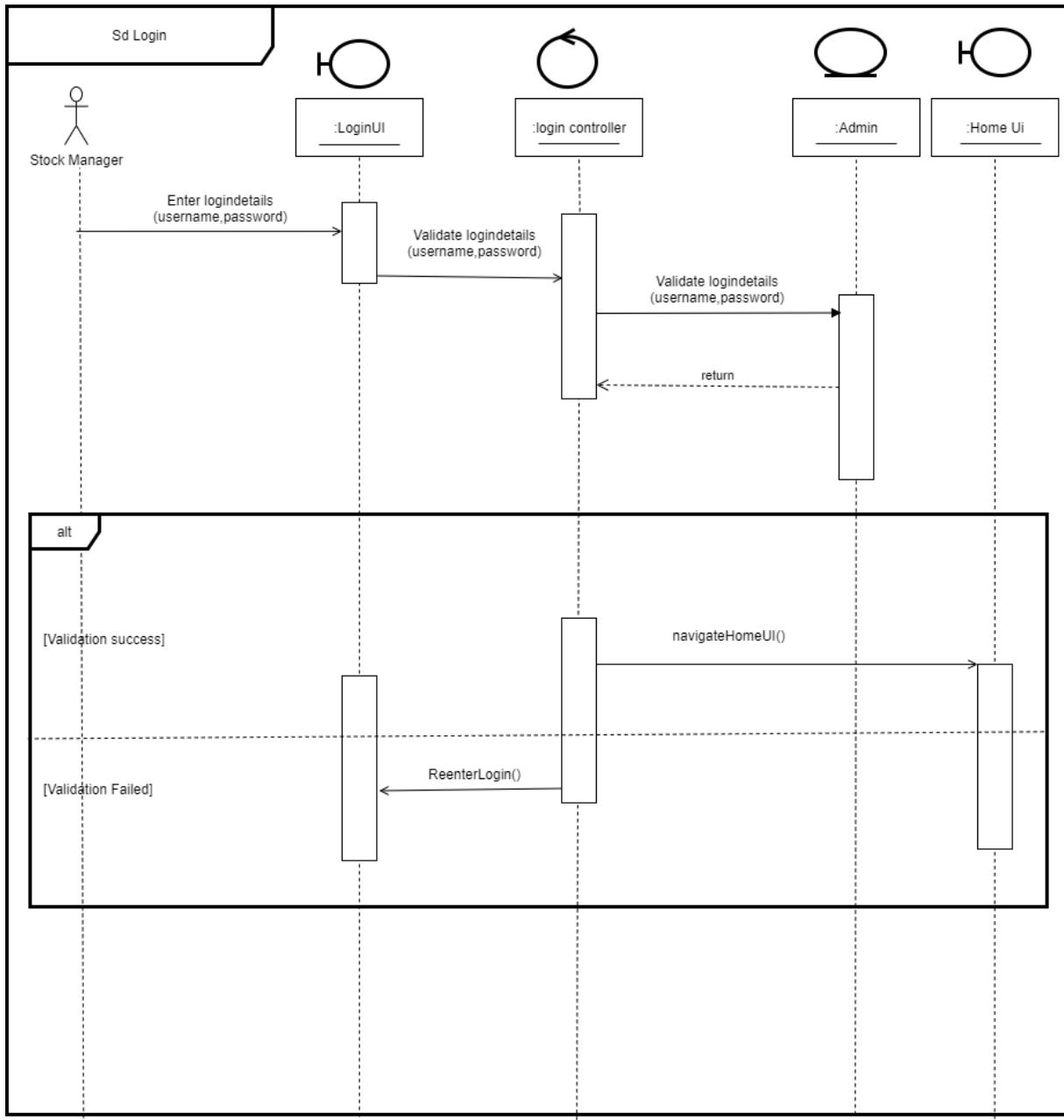


Figure 14: Sequence Diagram of Login

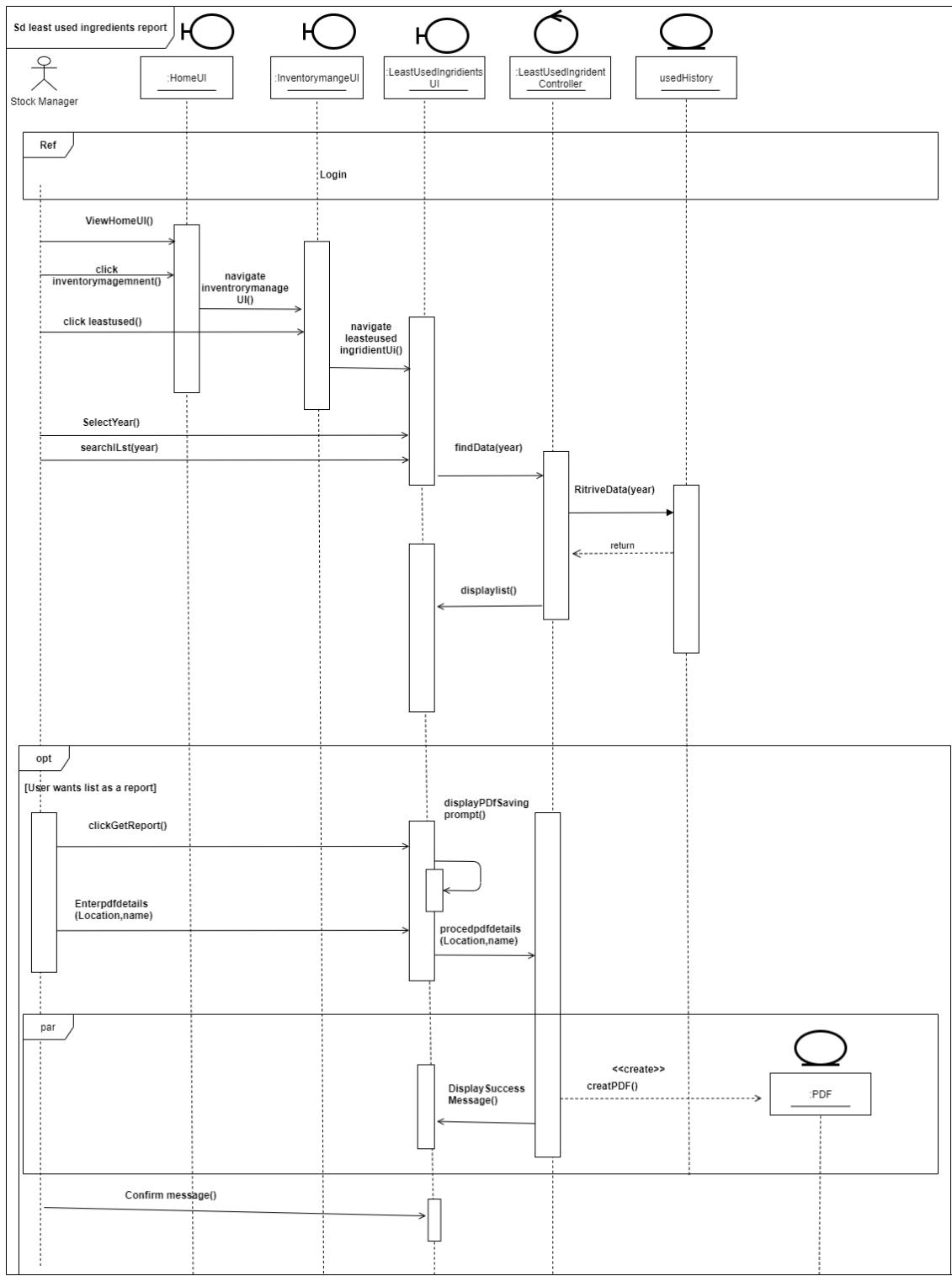


Figure 15: Sequence Diagram of Generate least used report

2.2.3.2 Hall Reservation

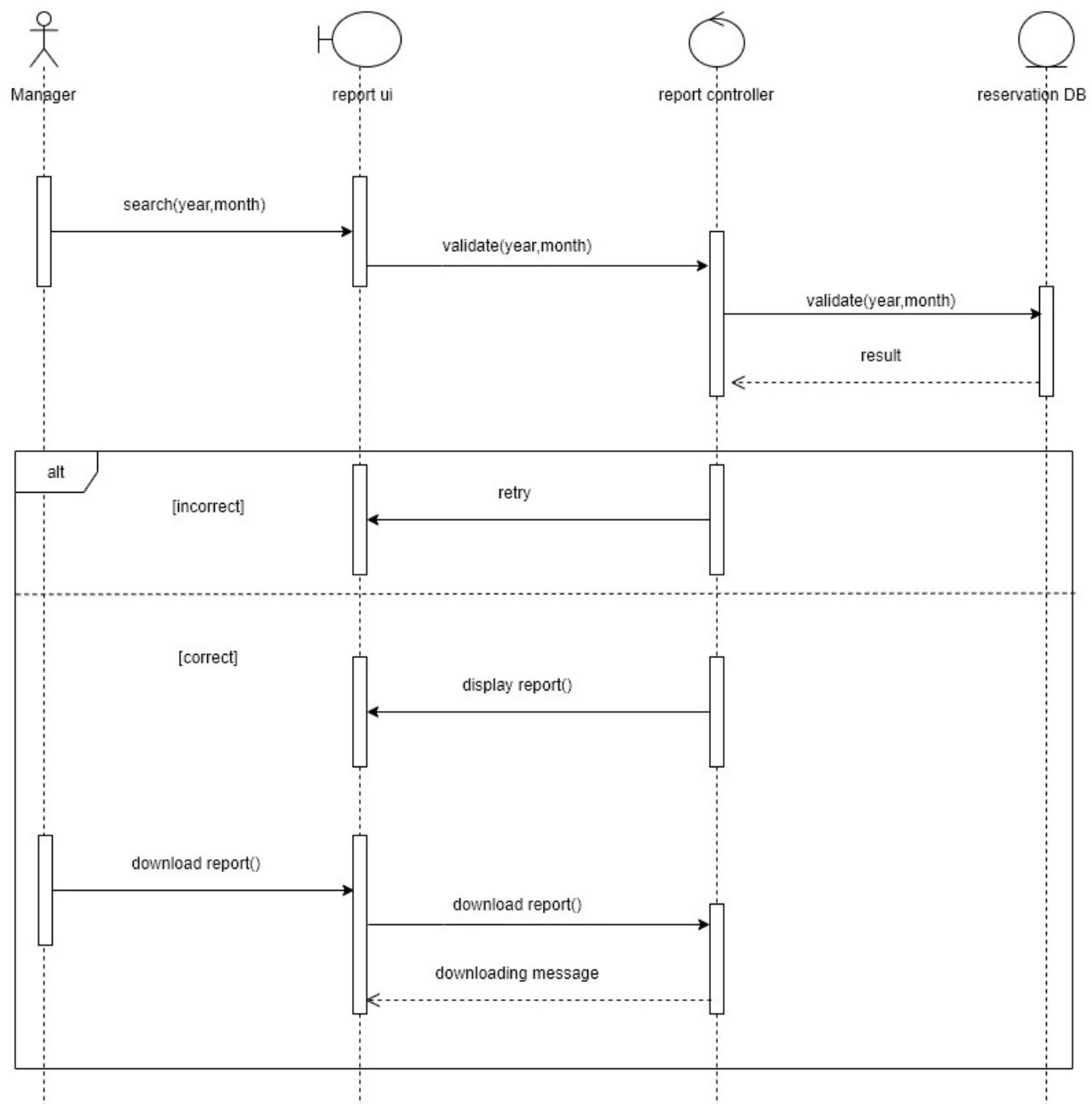


Figure 16: Sequence Diagram of Hall Reservation

2.2.3.3 Employee Handling and Salary Management

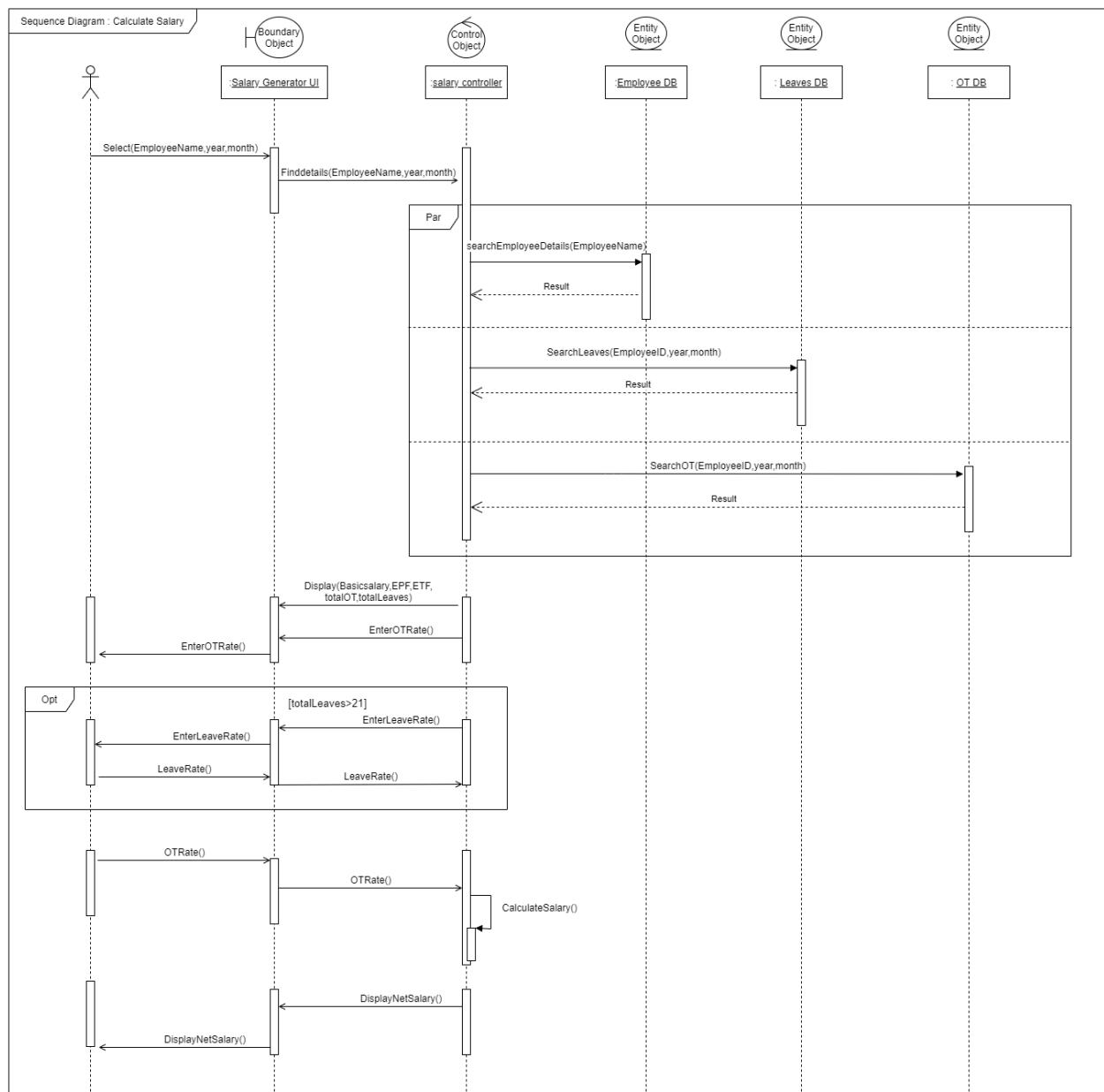


Figure 17: Sequence Diagram of Calculate Employee's Salary

2.2.3.4 Room Reservation

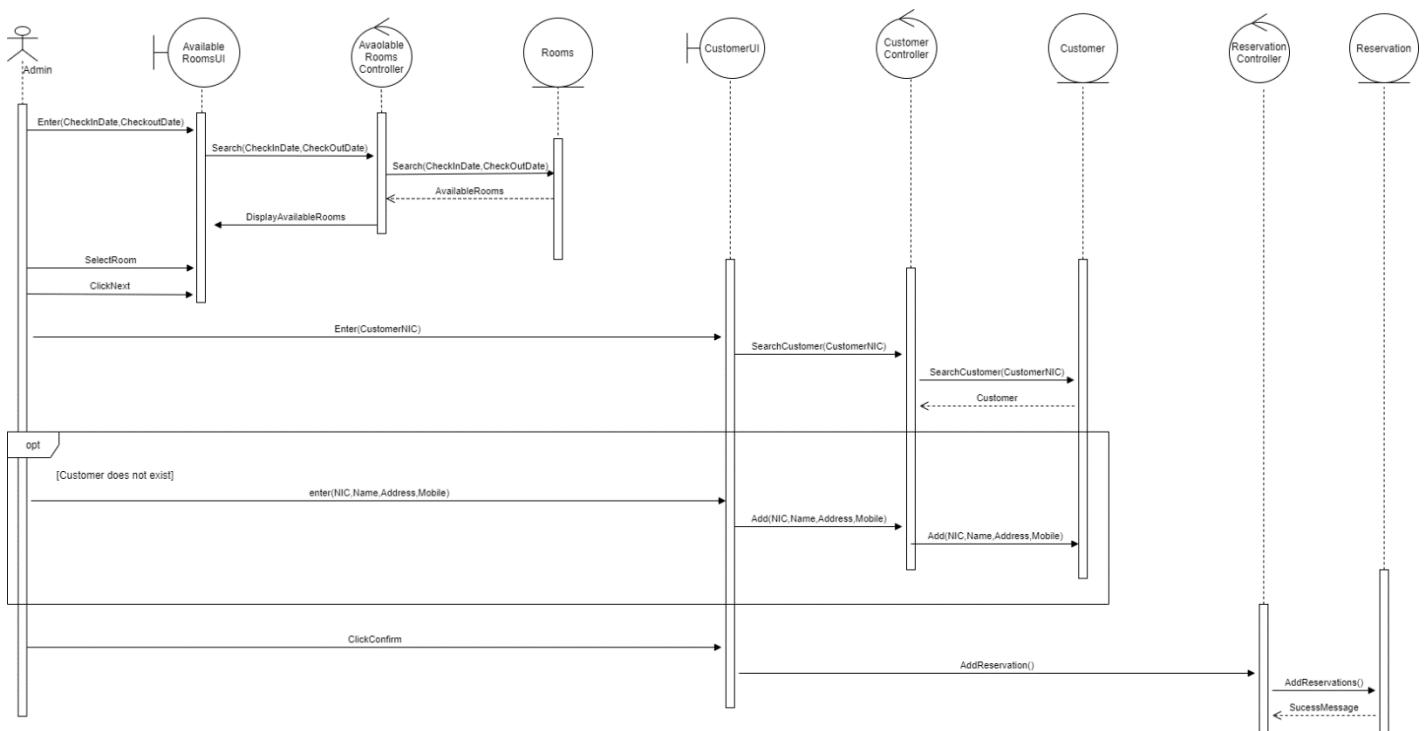


Figure 18: Sequence Diagram of Room Reservations

2.2.3.5 Order Handling

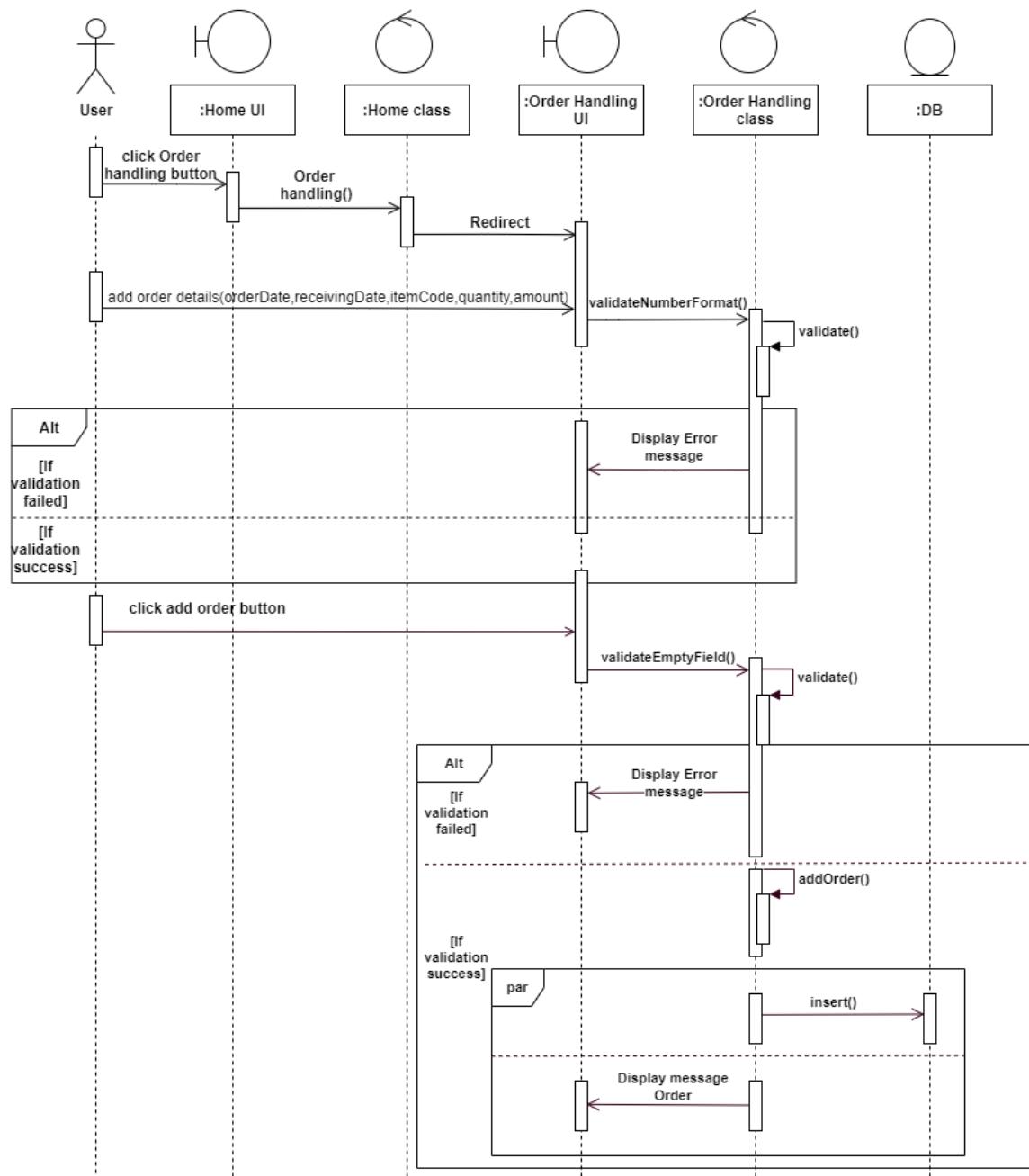


Figure 19: Sequence Diagram of Add Order

2.2.3.6 Expenses Handling

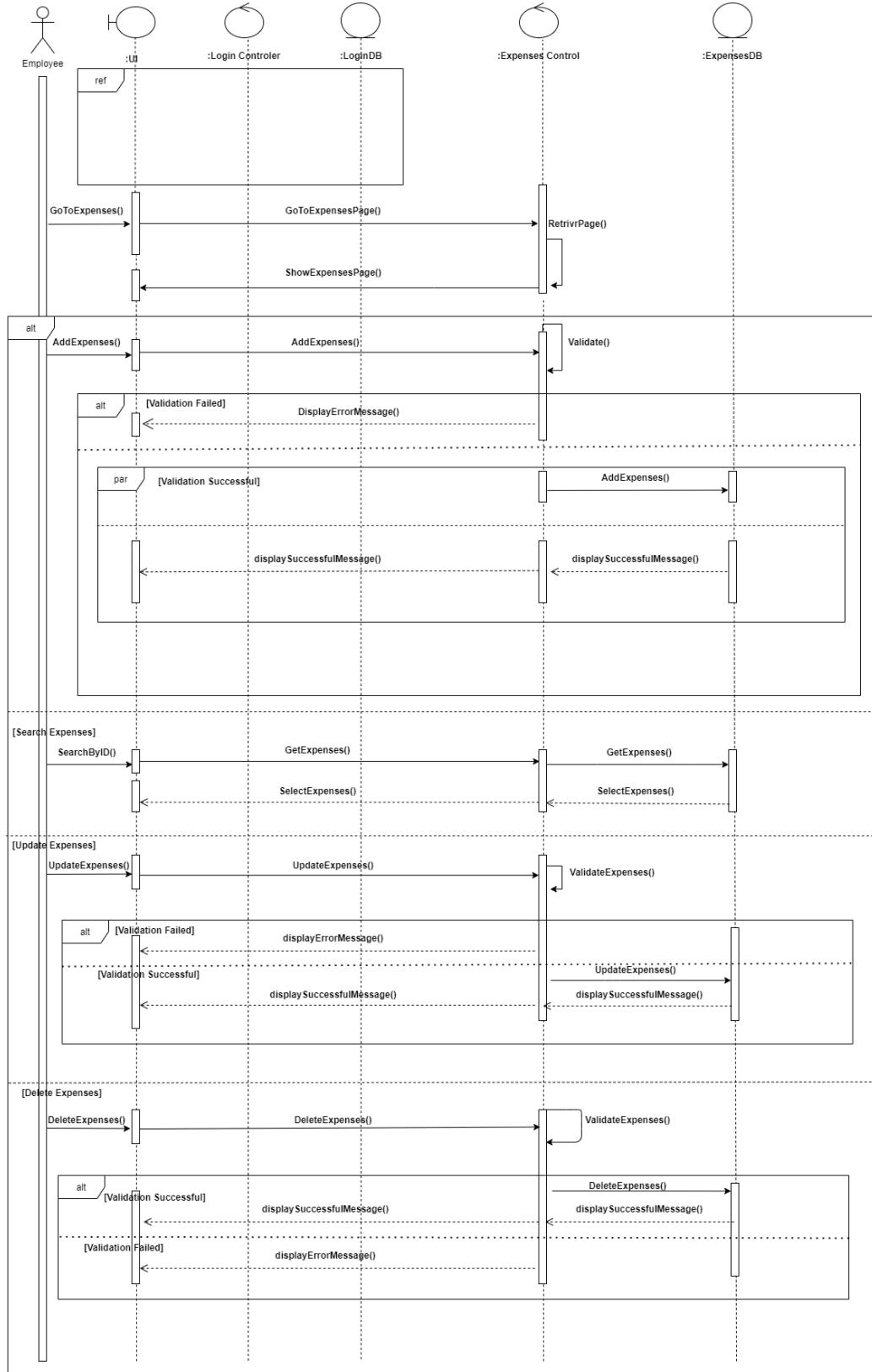


Figure 20: Sequence Diagram of Expenses Handling

2.2.3.7 Payment Handling

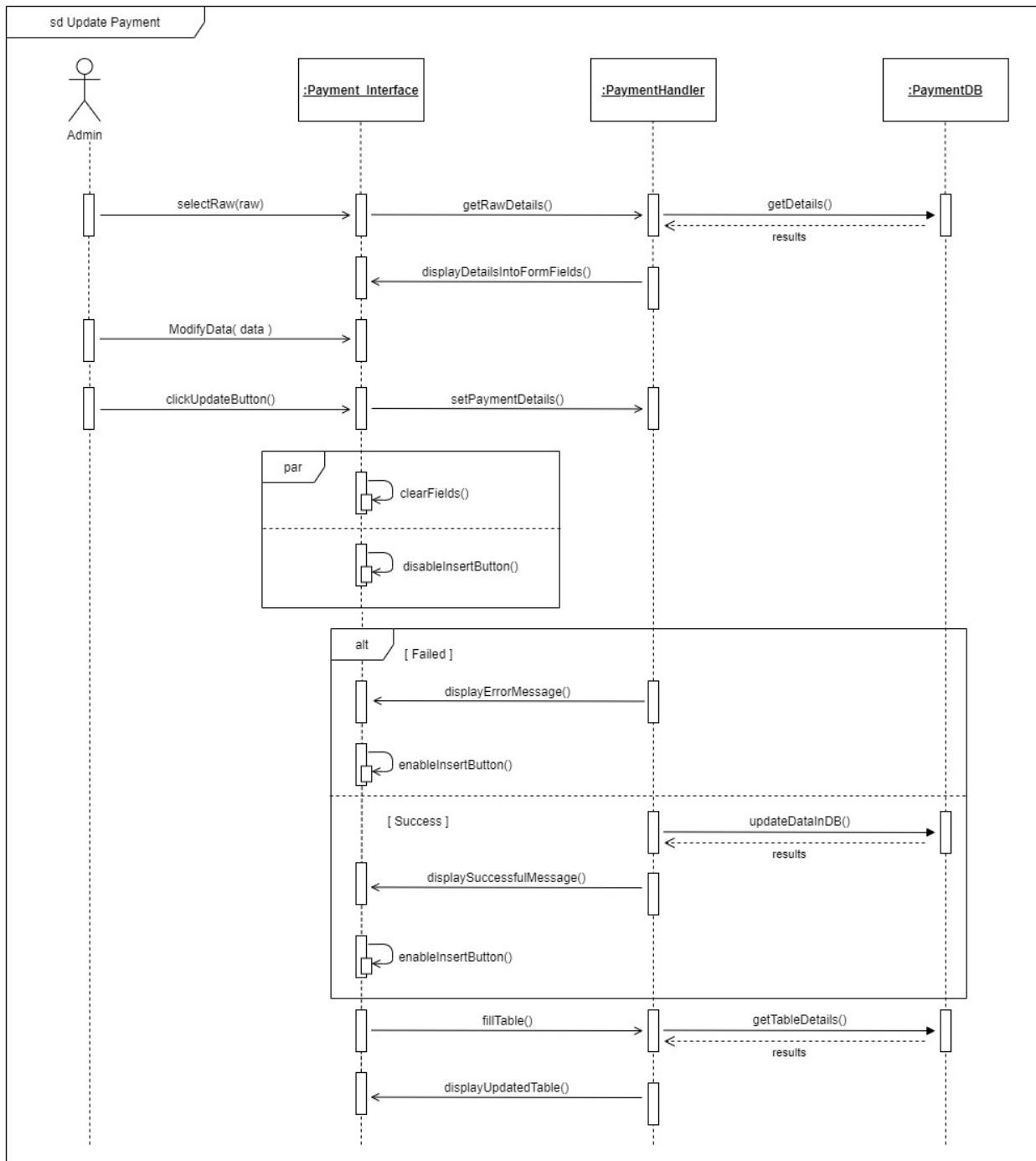


Figure 21: Sequence Diagram of Update Payment

2.2.3.8 Menus Handling

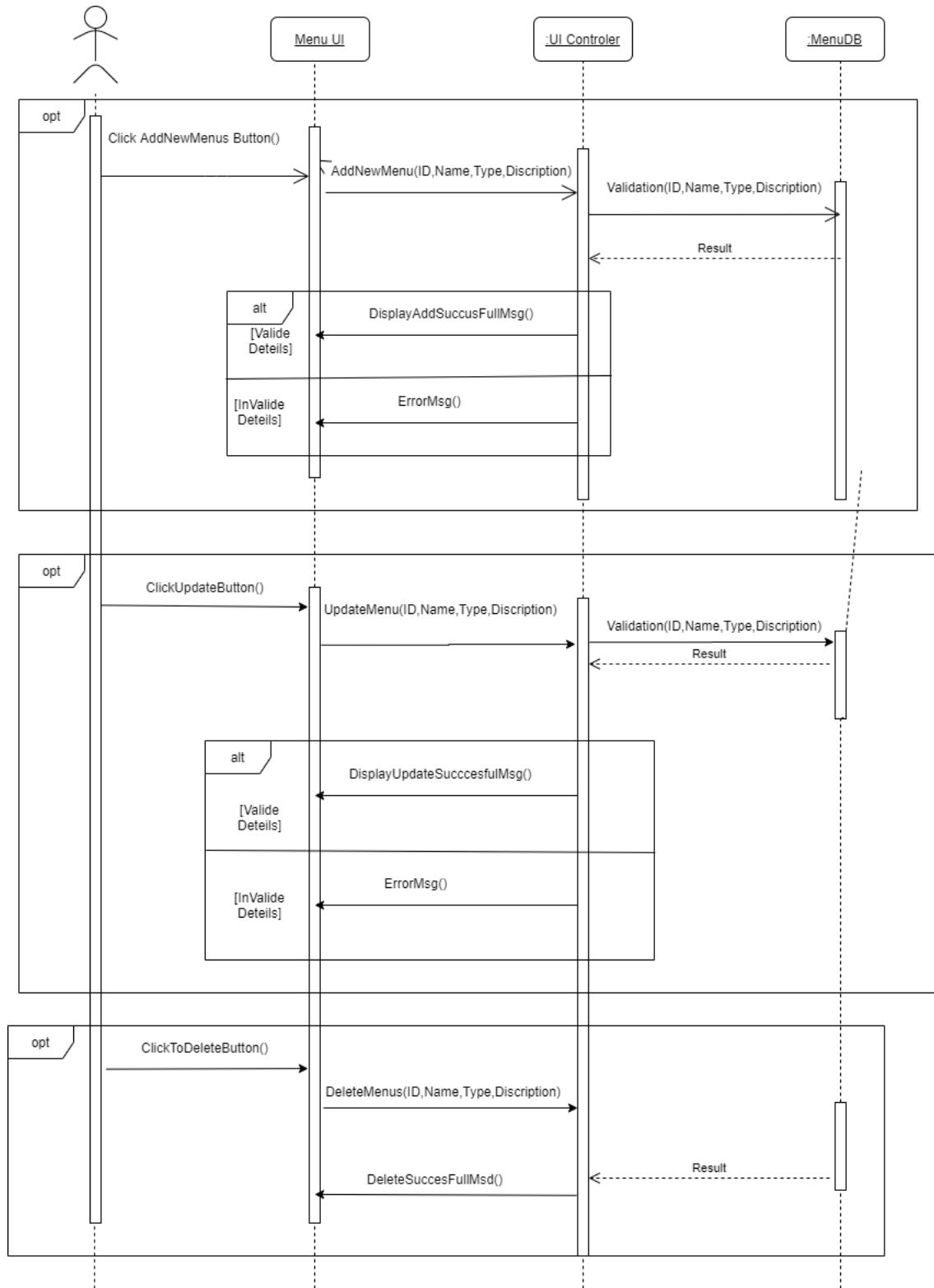


Figure 22: Sequence Diagram of Menus Handling

2.2.3.9 Travel Package Handling

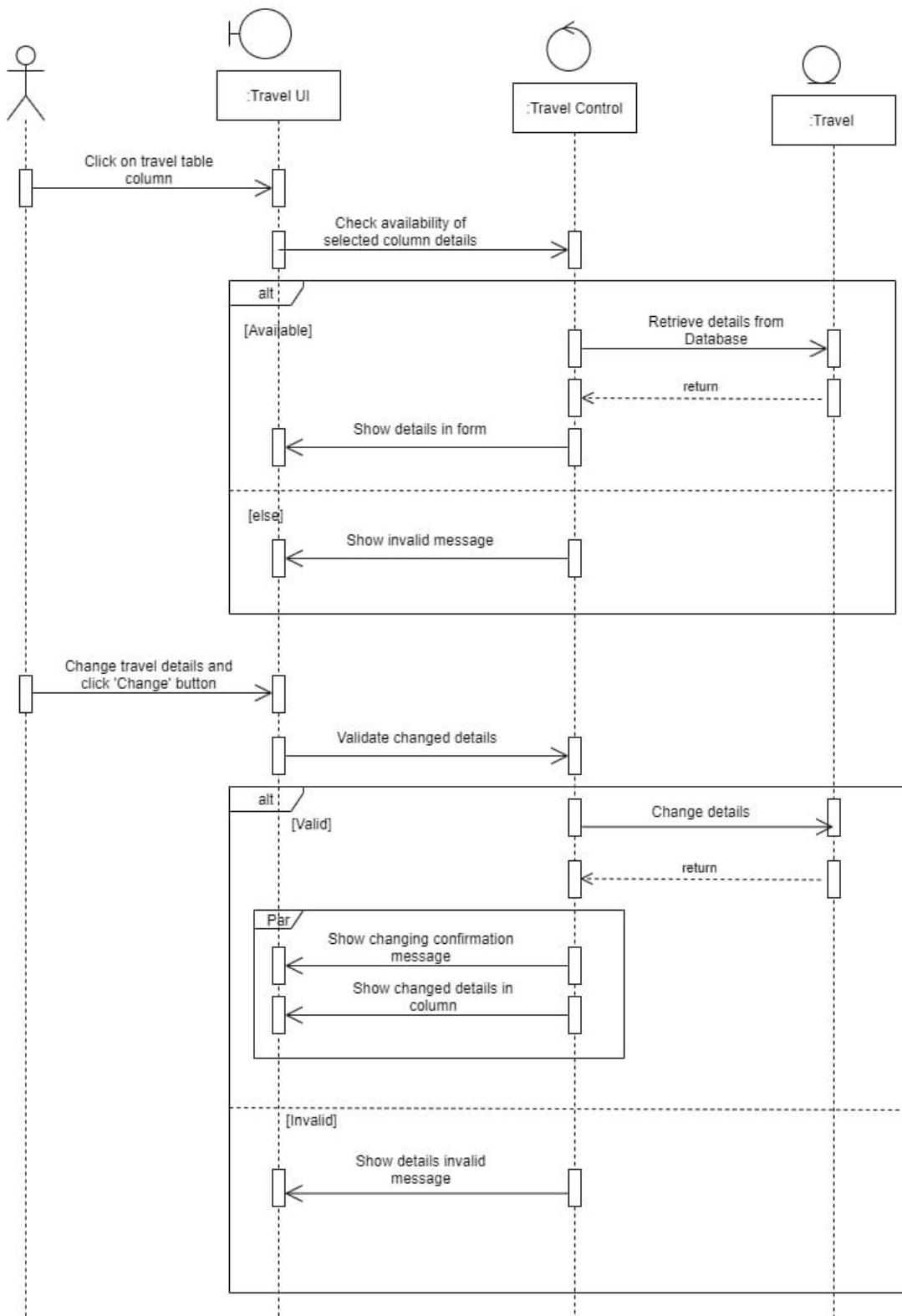


Figure 23: Sequence Diagram of Travel Packages Handling

2.2.4 Communication Diagrams

2.2.4.1 Inventory Management

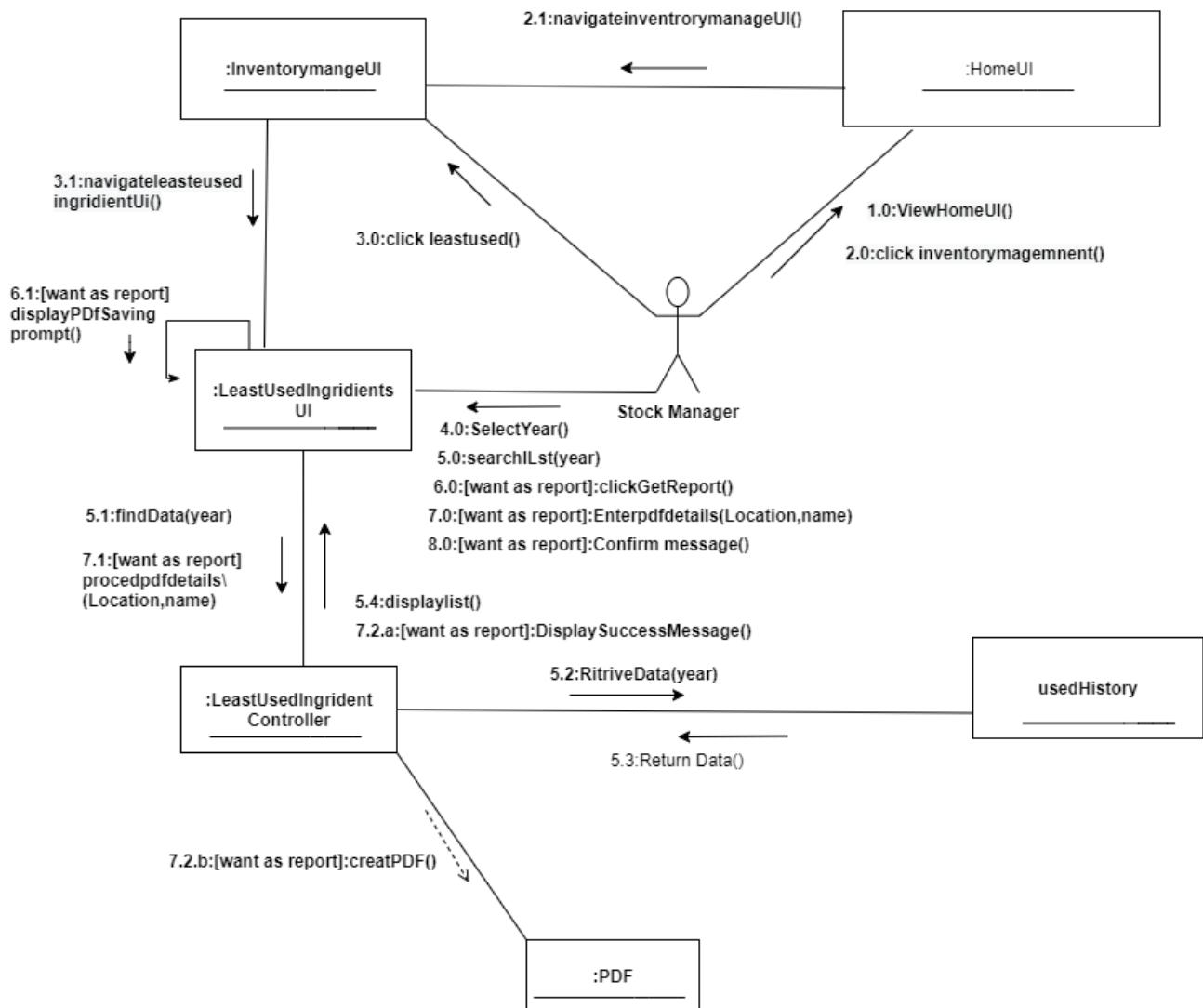


Figure 24 : Communication Diagram of Generate Lest Used Ingredient Report

2.2.4.2 Hall Reservation

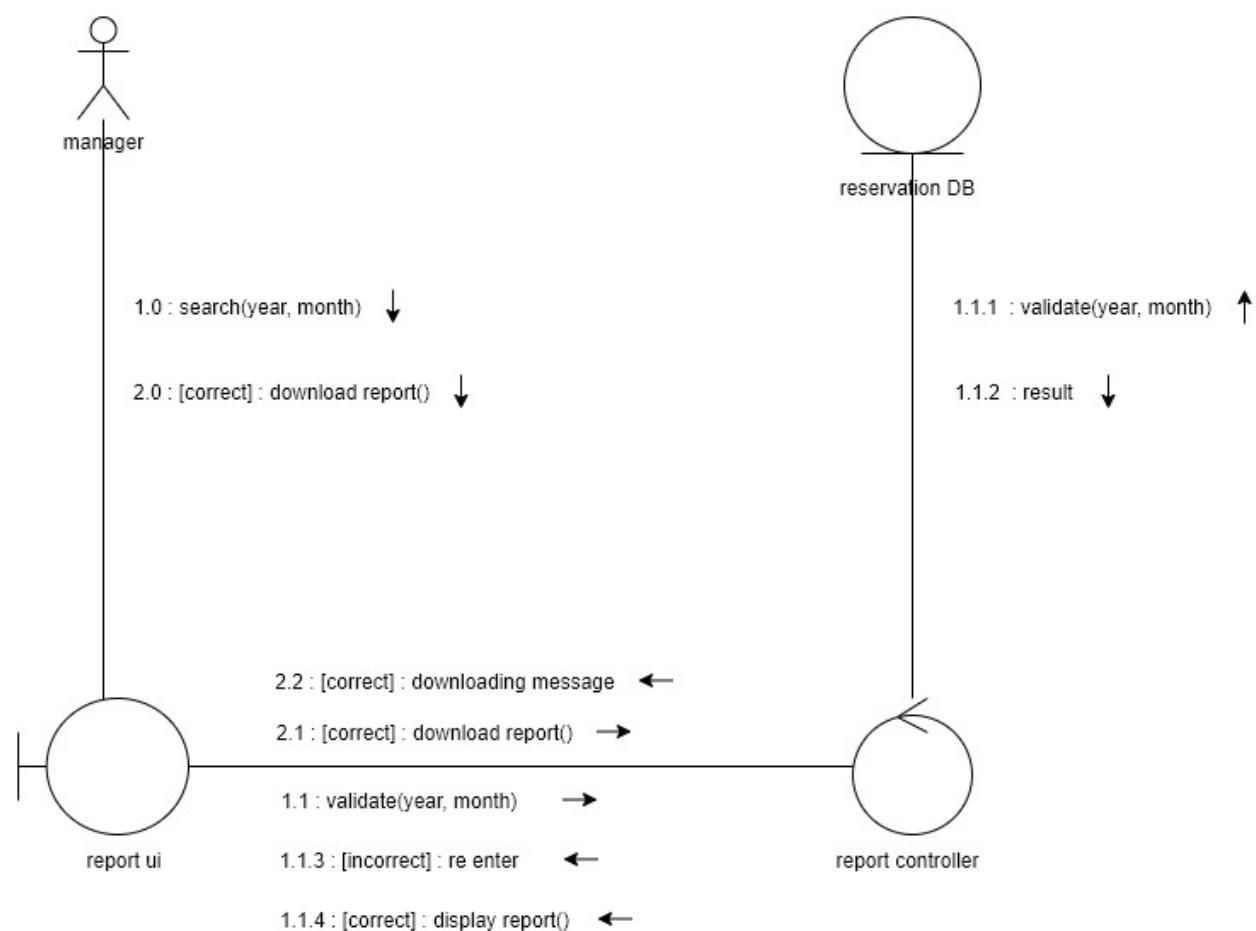


Figure 25: Communication Diagram of Hall Reservation

2.2.4.3 Employee Handling and Salary Management

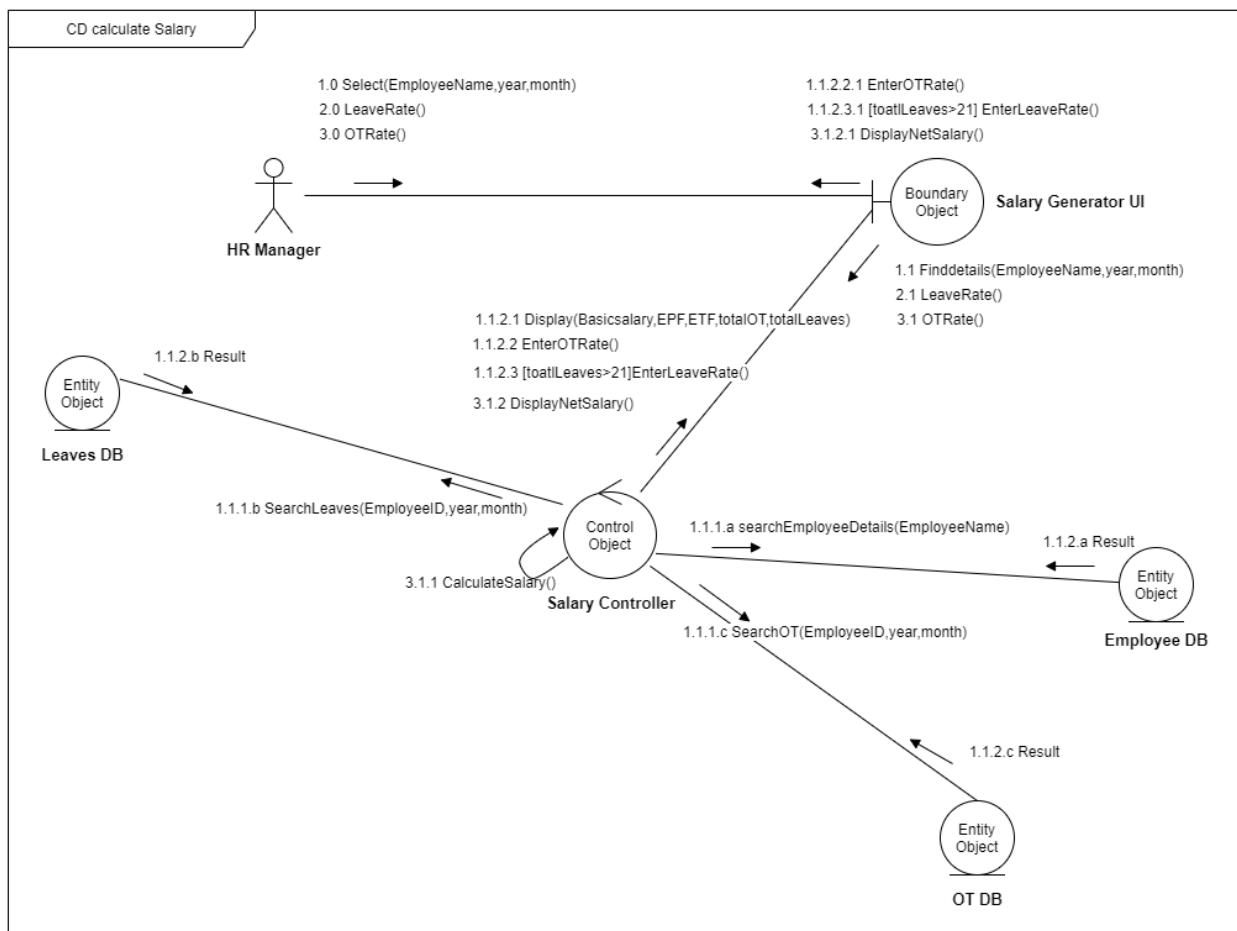


Figure 26: Communication Diagram of Calculate Employee's Salary

2.2.4.4 Room Reservations

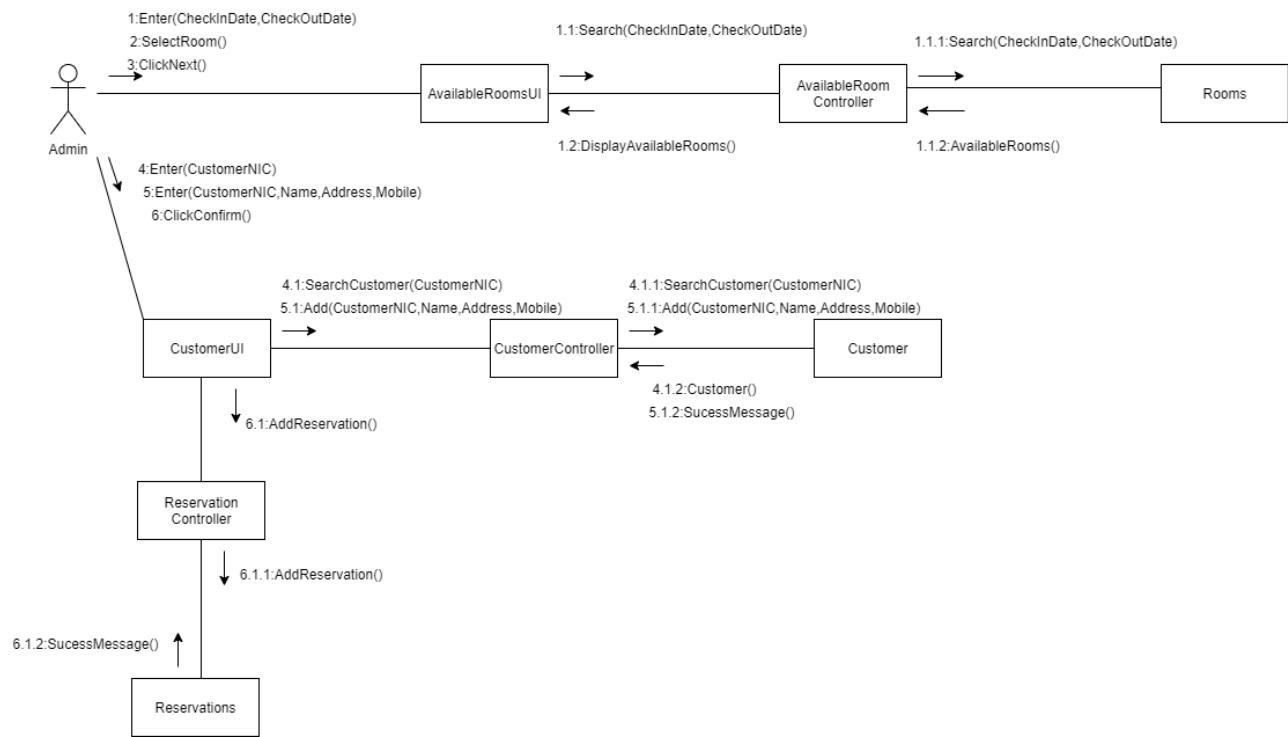


Figure 27: Communication Diagram of Room Reservations

2.2.4.5 Order Handling

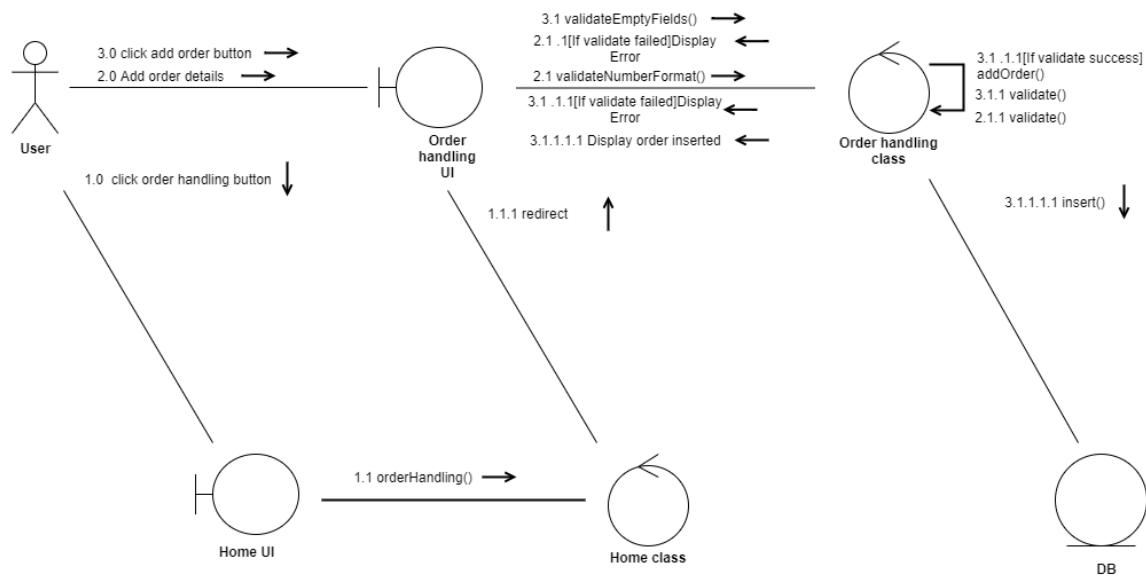


Figure 28: Communication Diagram of Add Order

2.2.4.6 Expenses Handling

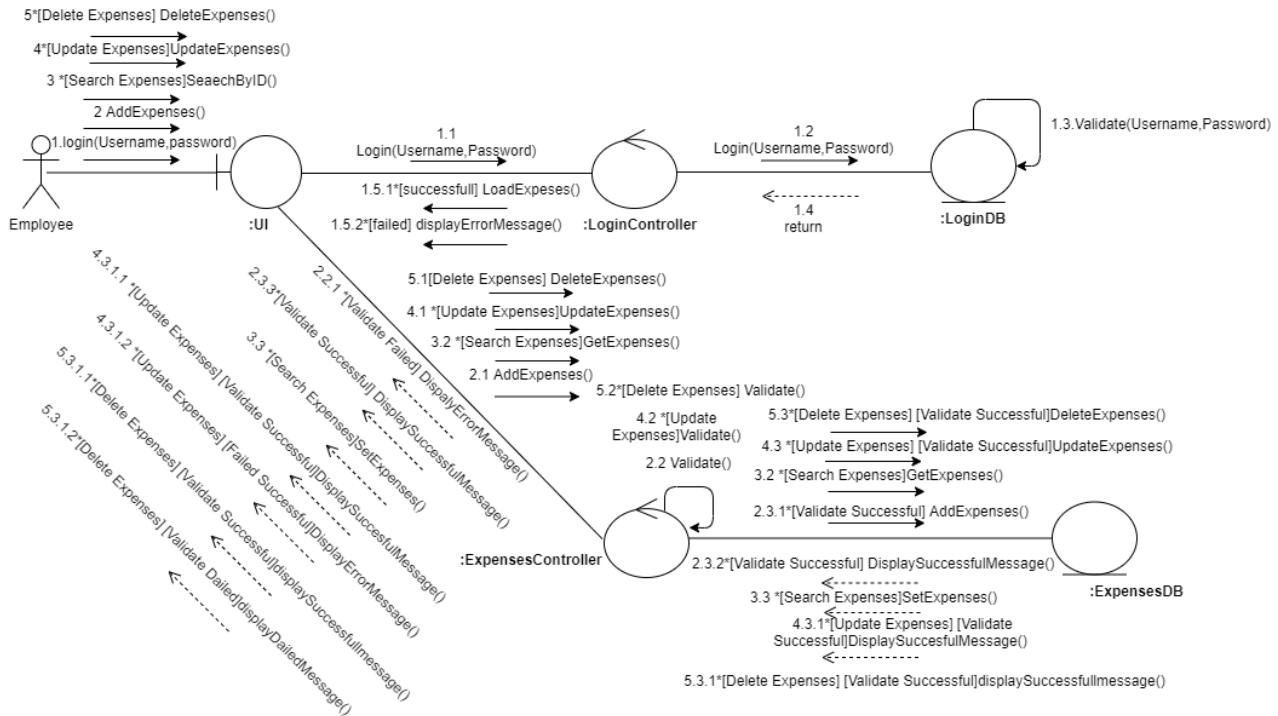


Figure 29: Communication Diagram of Expenses Handling

2.2.4.7 Payment Handling

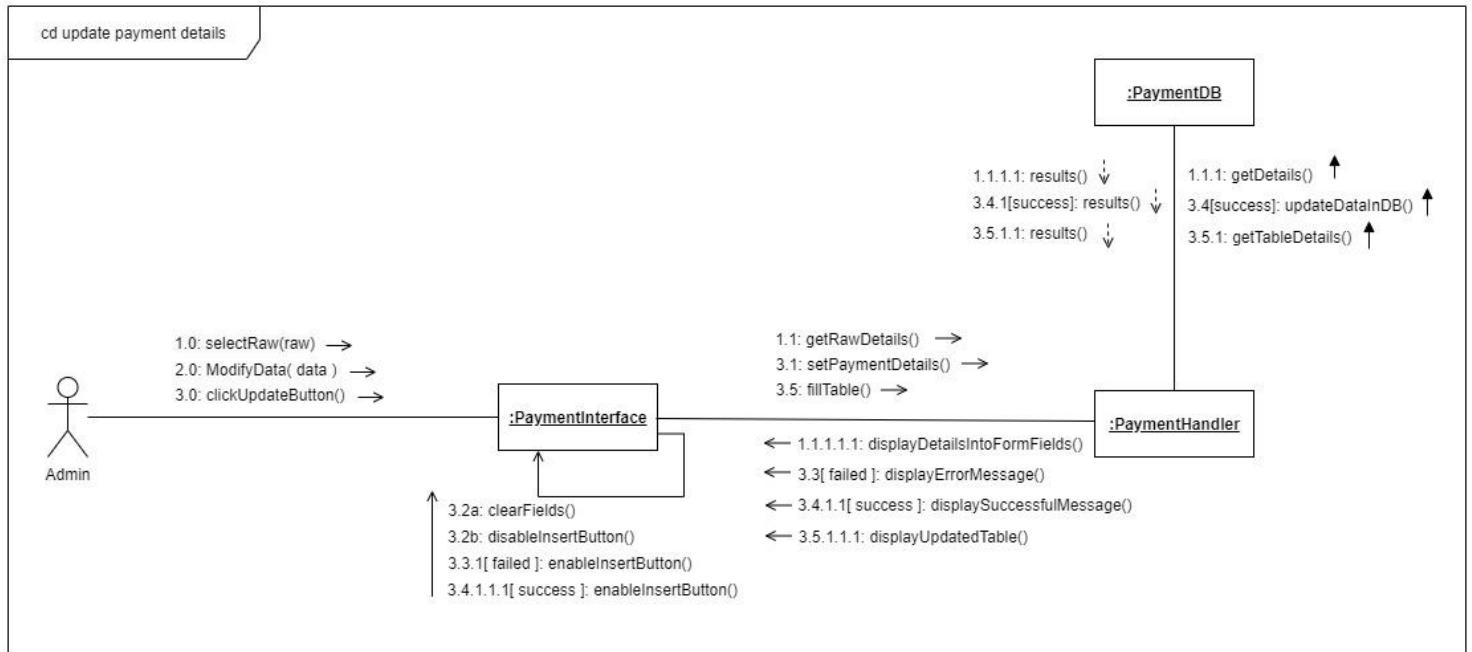


Figure 30: Communication Diagram of Update Payment

2.2.4.8 Menus Handling

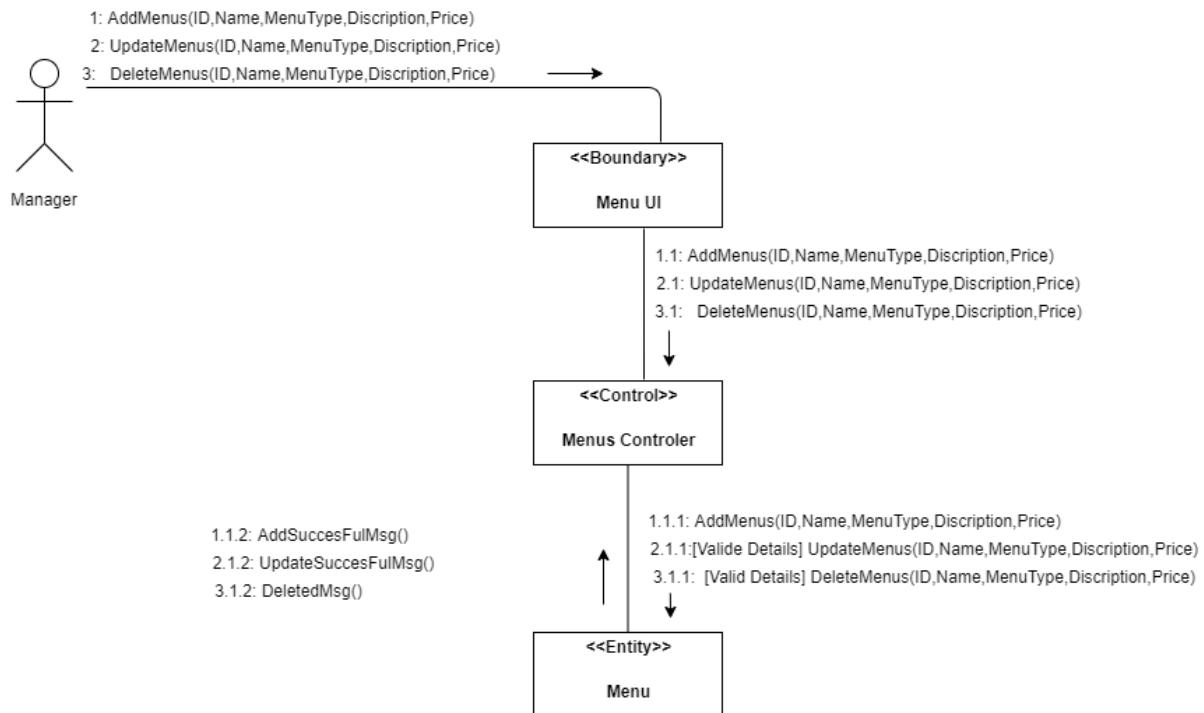


Figure 31: Communication Diagram of Menus Handling

2.2.4.9 Travel Package Handling

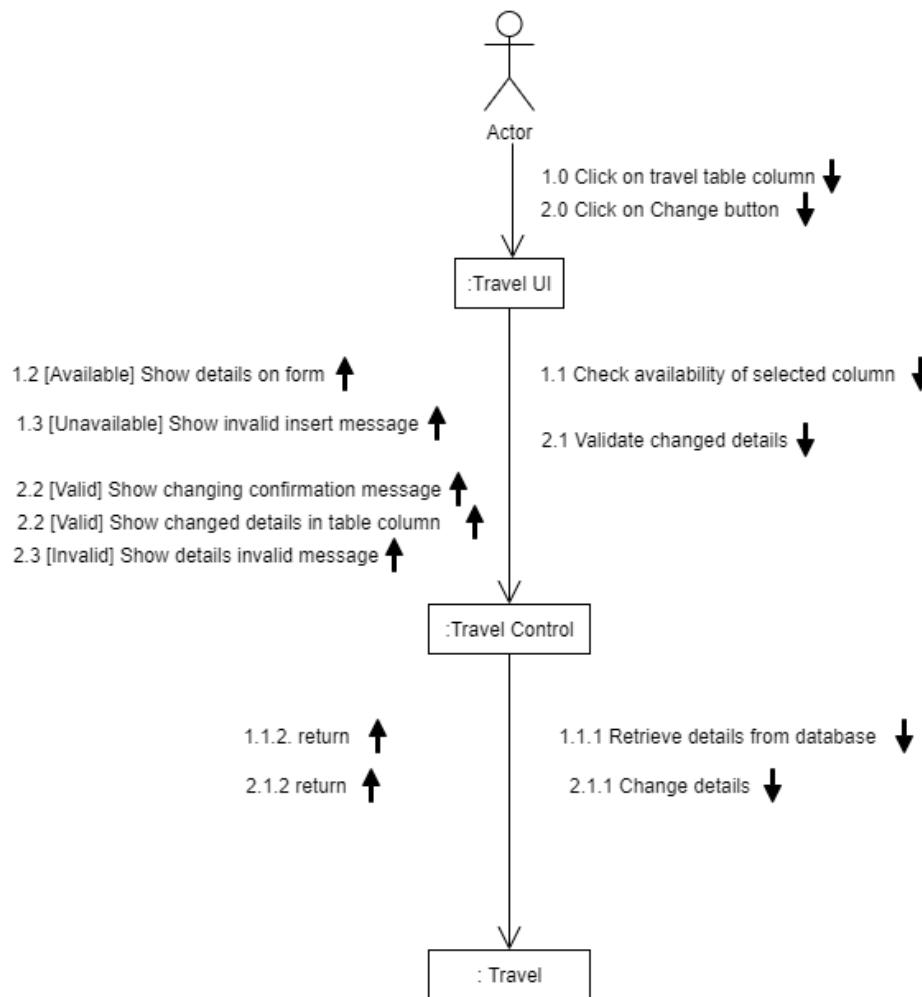


Figure 32: Communication Diagram of Travel Package Handling

2.2.5 State Chart Diagrams

2.2.5.1 Inventory Management

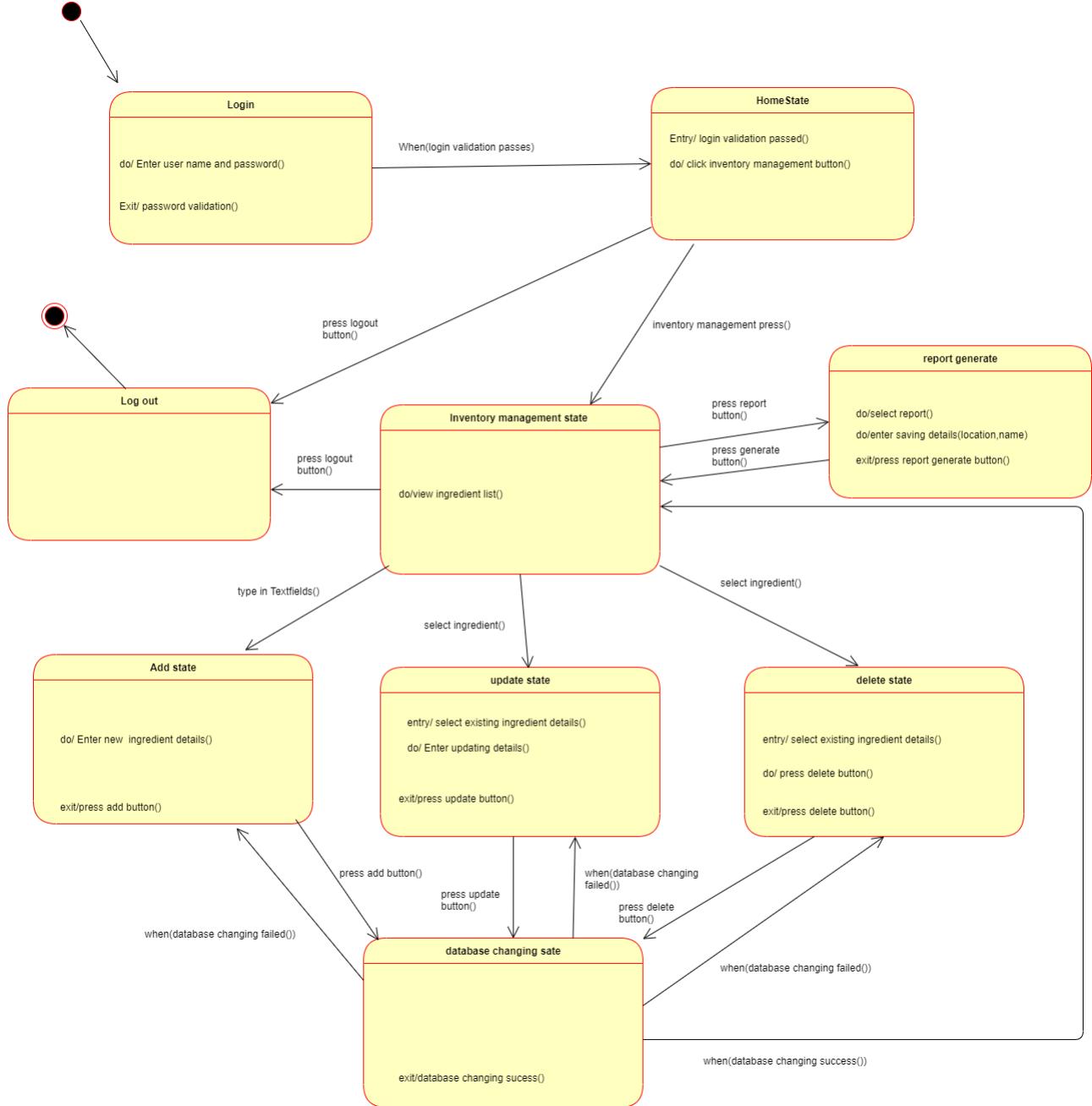


Figure 33: State Chart Diagram of Inventory Management

2.2.5.2 Hall Reservation

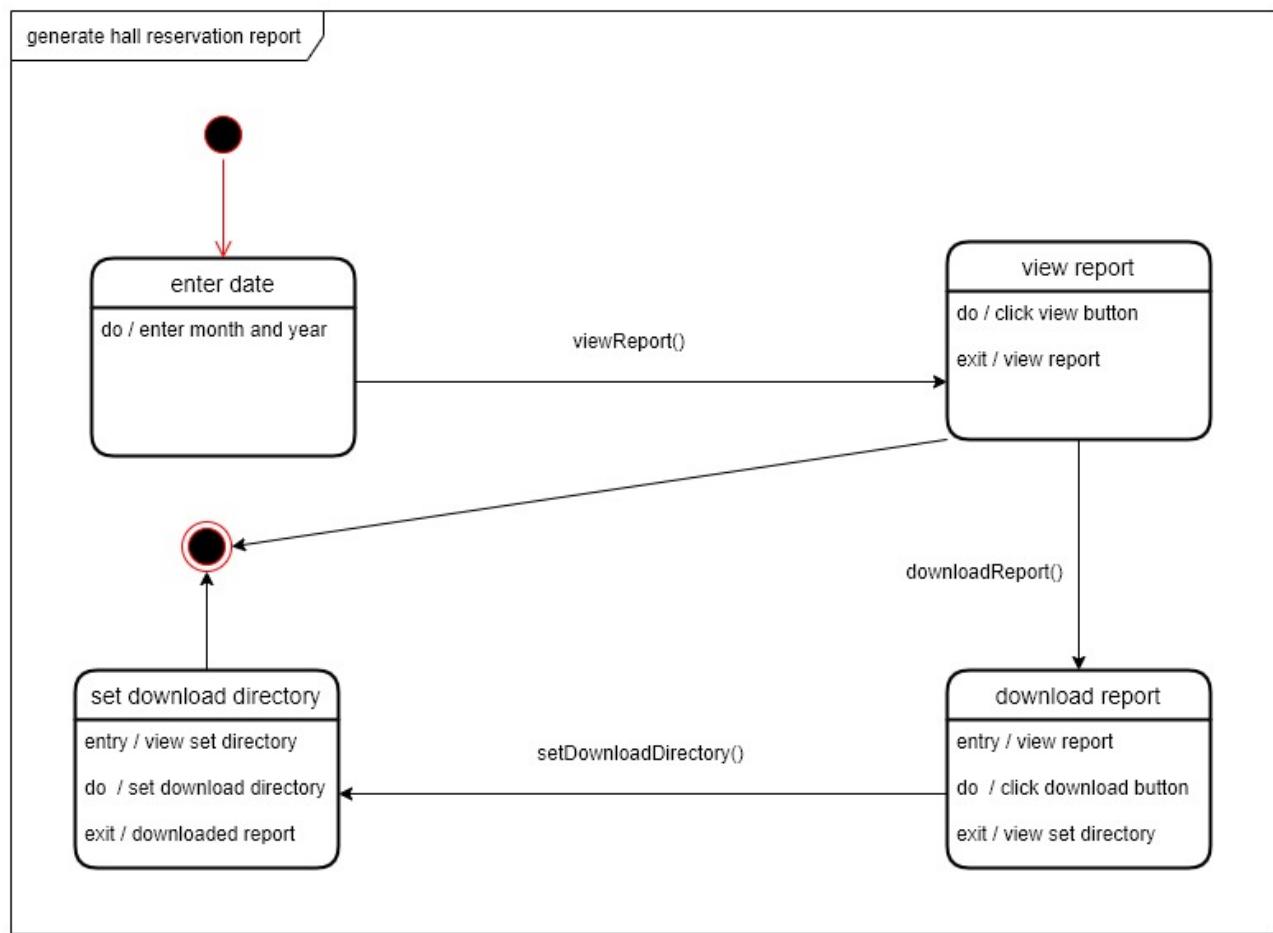


Figure 34 : State Chart Diagram of Hall Reservation

2.2.5.3 Employee Handling and Salary Management

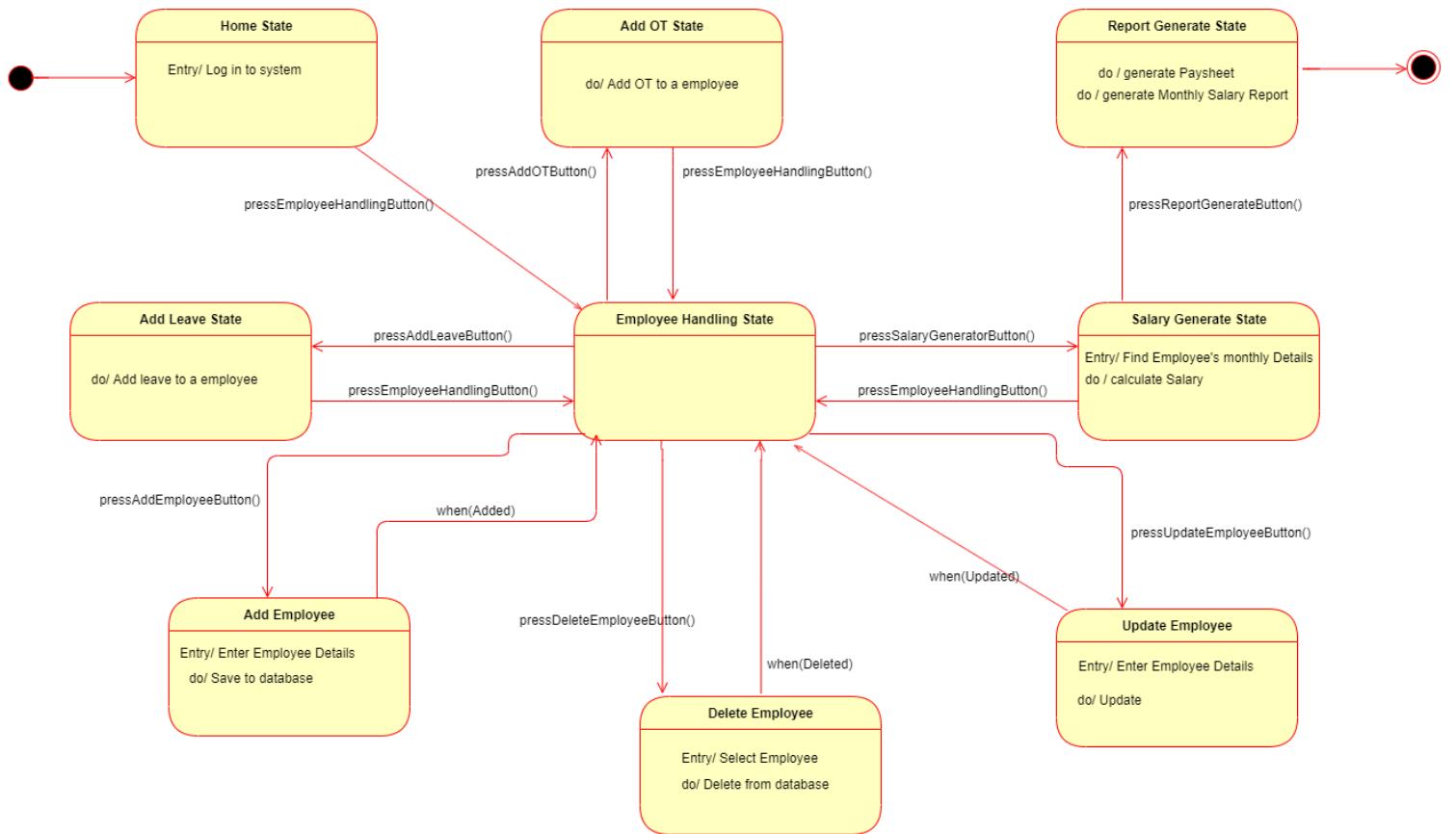


Figure 35: State Chart Diagram of Employee Handling and Salary Management

2.2.5.4 Room Reservations

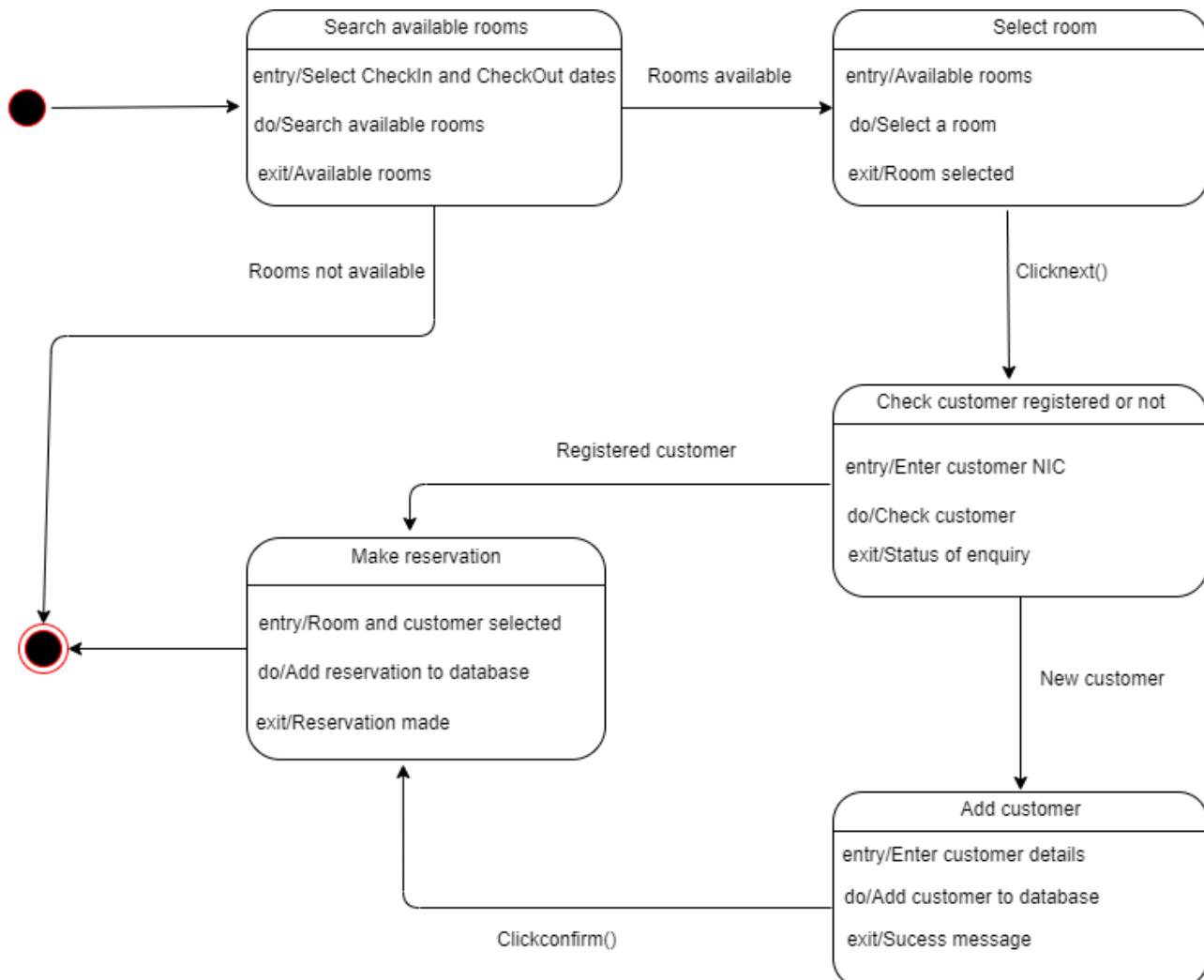


Figure 36: State Chart Diagram of Room Reservation

2.2.5.5 Order Handling

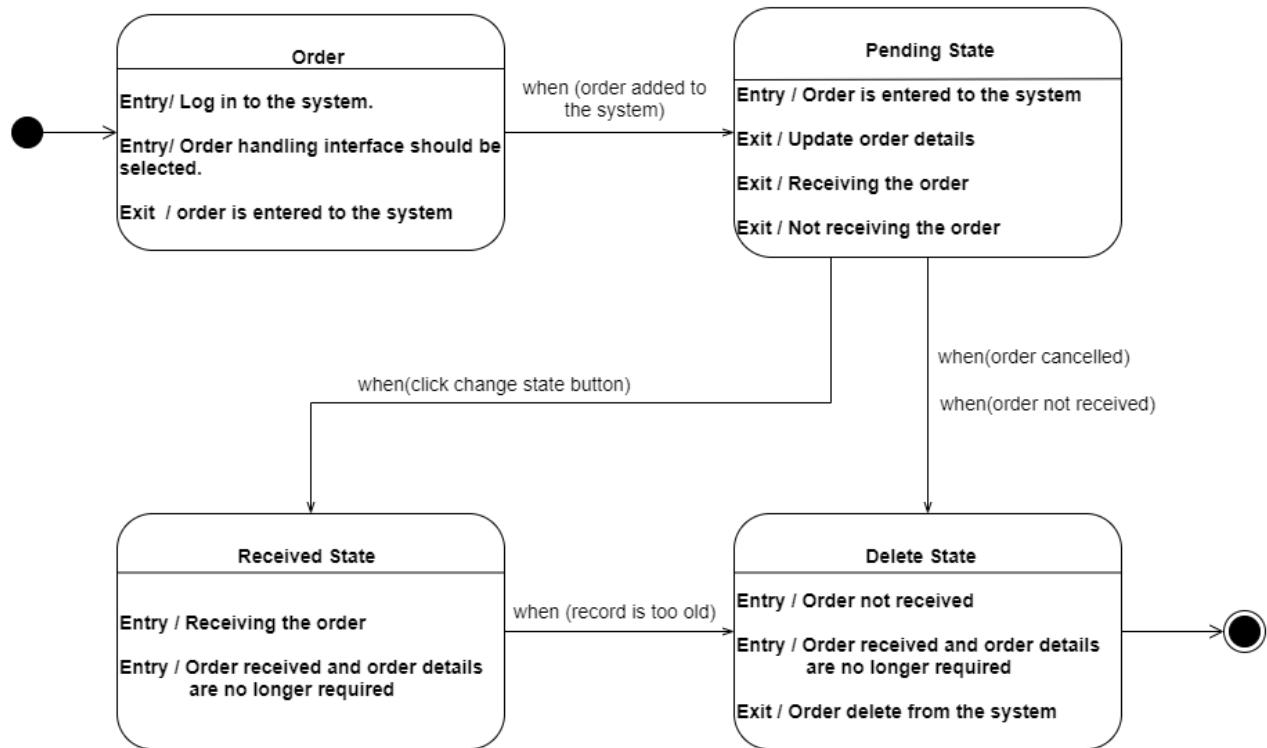


Figure 37: State Chart Diagram of Order Handling

2.2.5.6 Expenses Handling

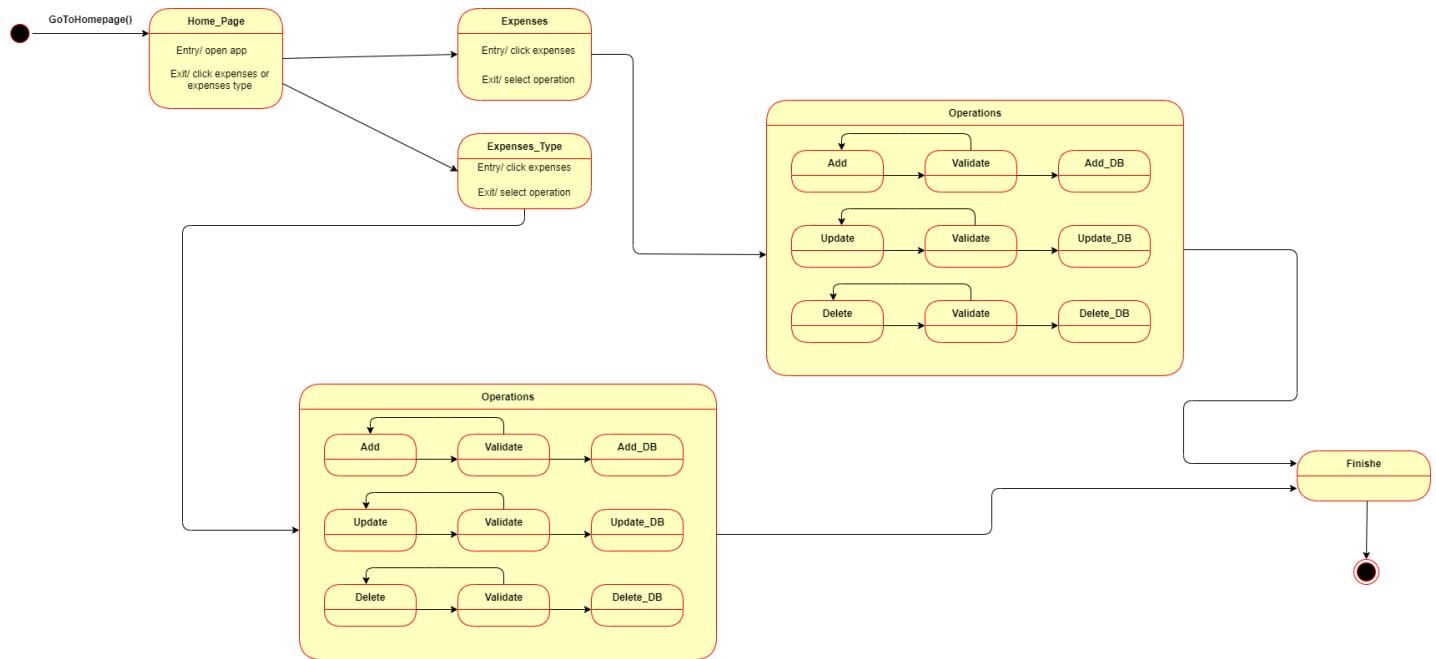


Figure 38: State Chart Diagram of Expenses Handling

2.2.5.7 Payment Handling

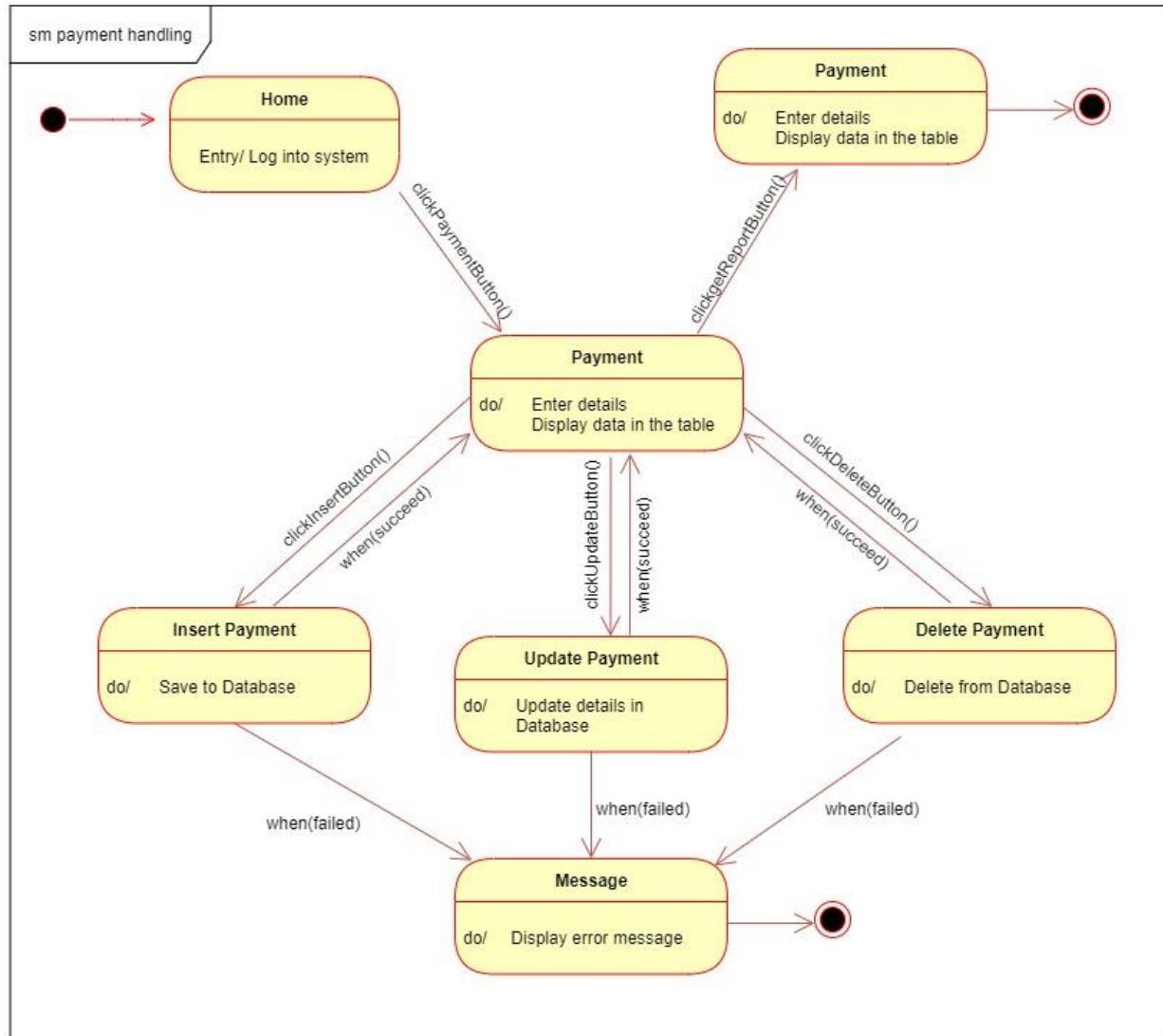


Figure 39: State Chart Diagram of Payment Handling

2.2.5.8 Menus Handling

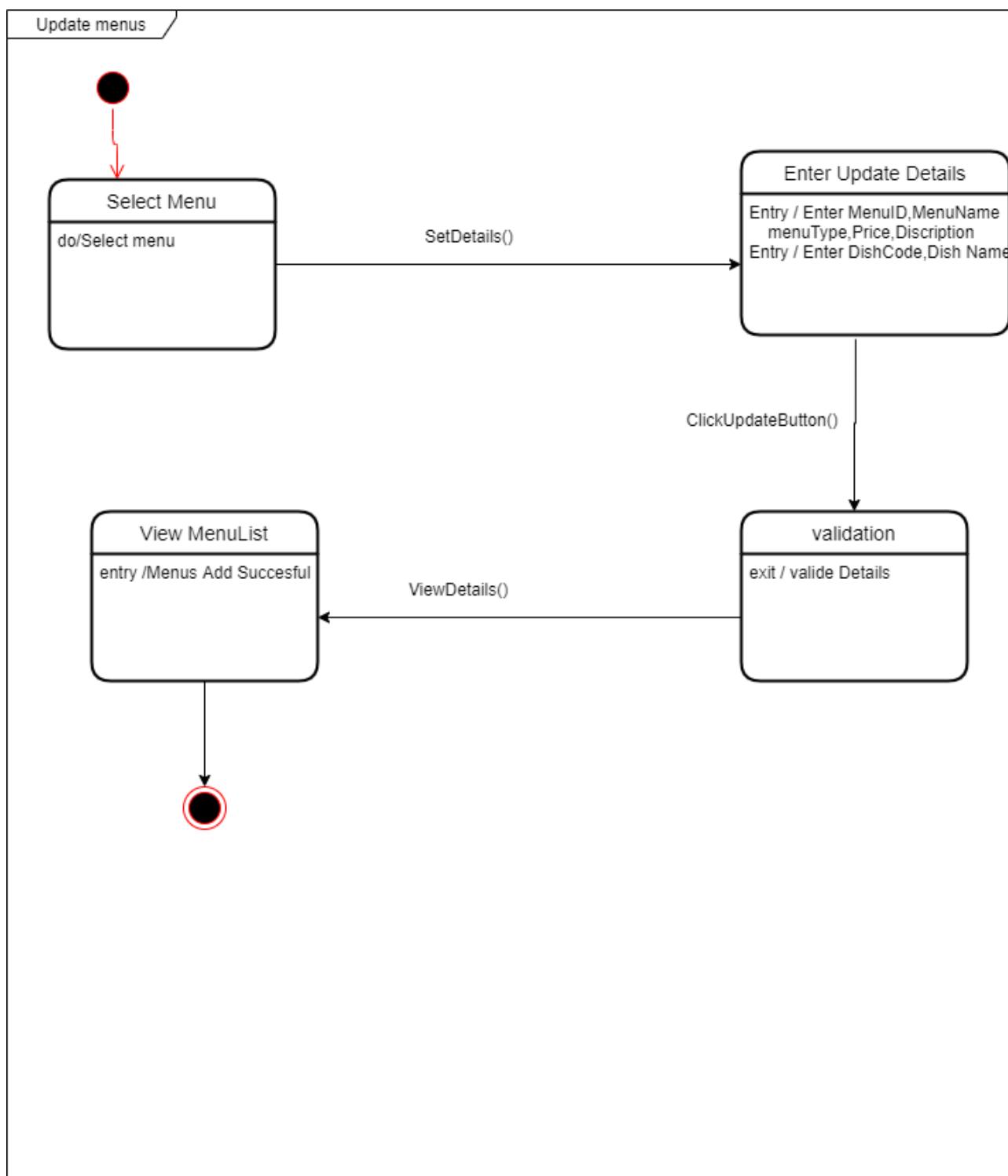


Figure 40: State Chart Diagram of Order Handling

2.2.5.9 Travel Package Handling

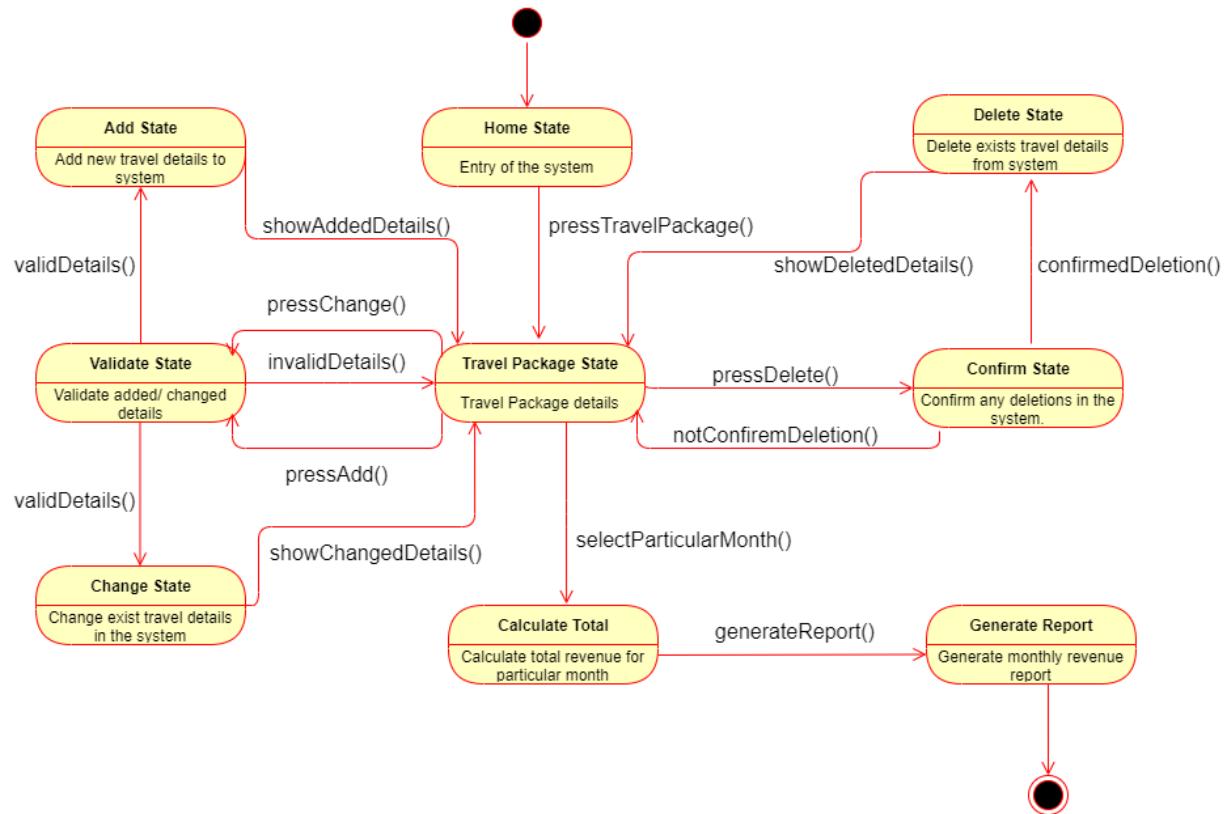


Figure 41: State Chart Diagram of Travel Package Handling

2.2.6 User Interfaces

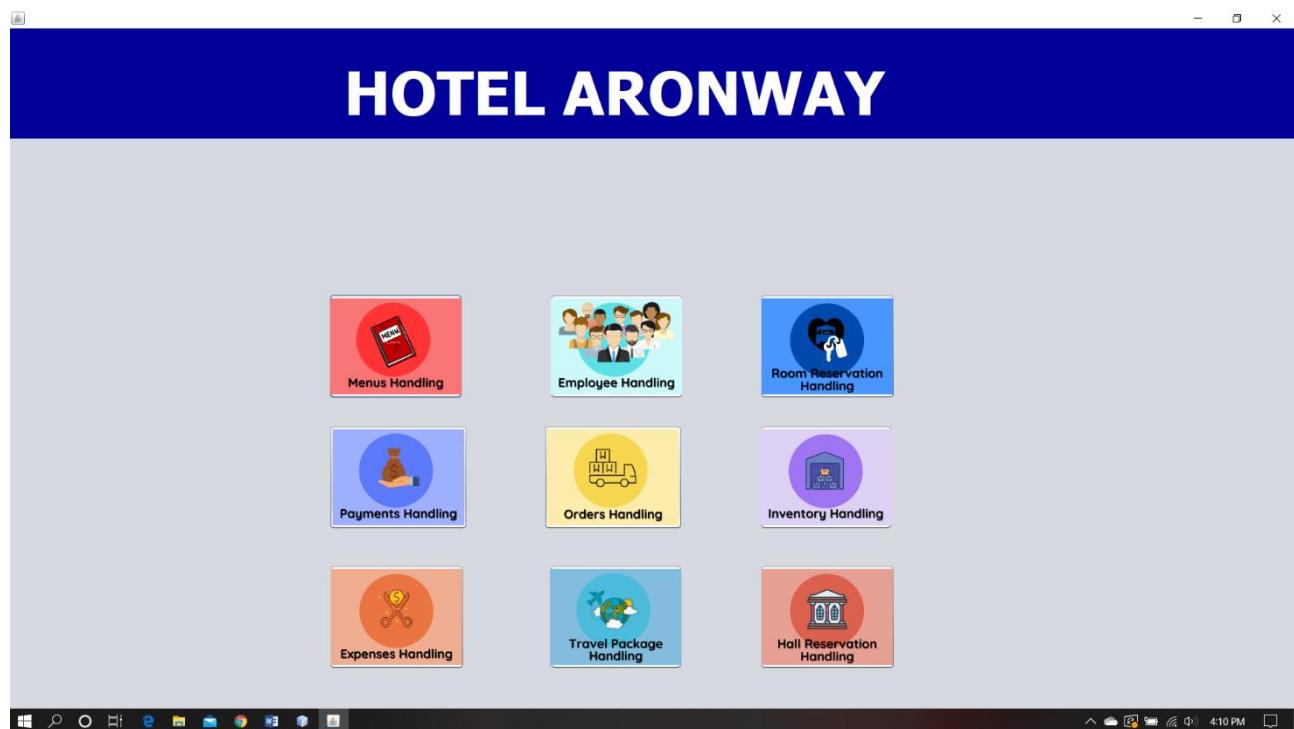


Figure 42: Home Page

2.2.6.1 Inventory Management Interfaces

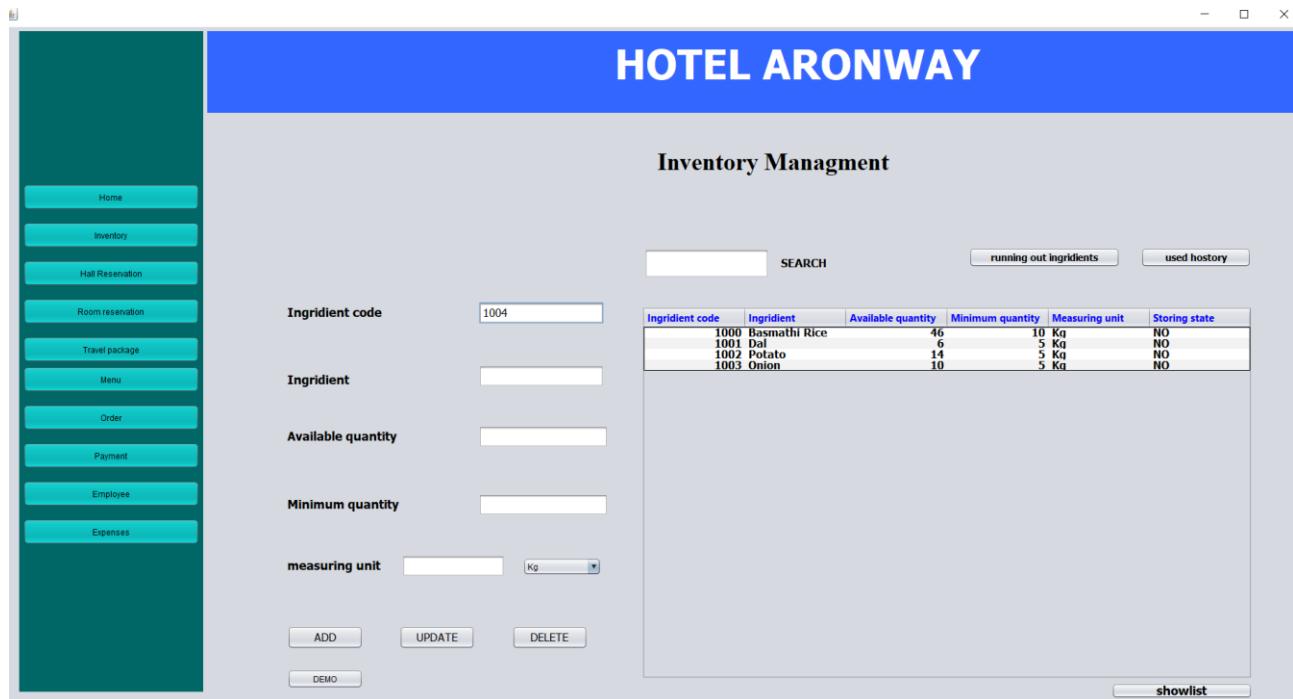


Figure 43: Inventory Management Interface



Figure 44: Least Used Ingredients Interface

HOTEL ARRONWAY

Running out Ingredients

Get the list

Ingrident code	Ingrident	Available quantity	Minimum quantity	Measuring unit	Storing state
1001	Dal	4	2	5 Kg	YES
1002	Potato	2		5 Kg	YES

Figure 45: Running Out Ingridents Interface

HOTEL ARRONWAY

Used History

Ingredient Name Ingredient code used Quantity date Description

No	ingridient	code	Used quantity	date	Description
1	Basmathi Rice	1000	2	2020-08-10	for today event
4	Dal	1001	2	2020-08-10	for today event
5	Dal	1001	11	2020-09-21	for event
6	Dal	1000	2	2020-09-16	for wedding event
7	Basmathi Rice	1000	4	2020-10-17	for event
8	Potato	1002	1	2020-10-27	for event
11	Dal	1001	4	2020-10-14	for event

[Least used ingredients](#)

Figure 46: Used History Interface

2.2.6.2 Hall Reservation Interfaces

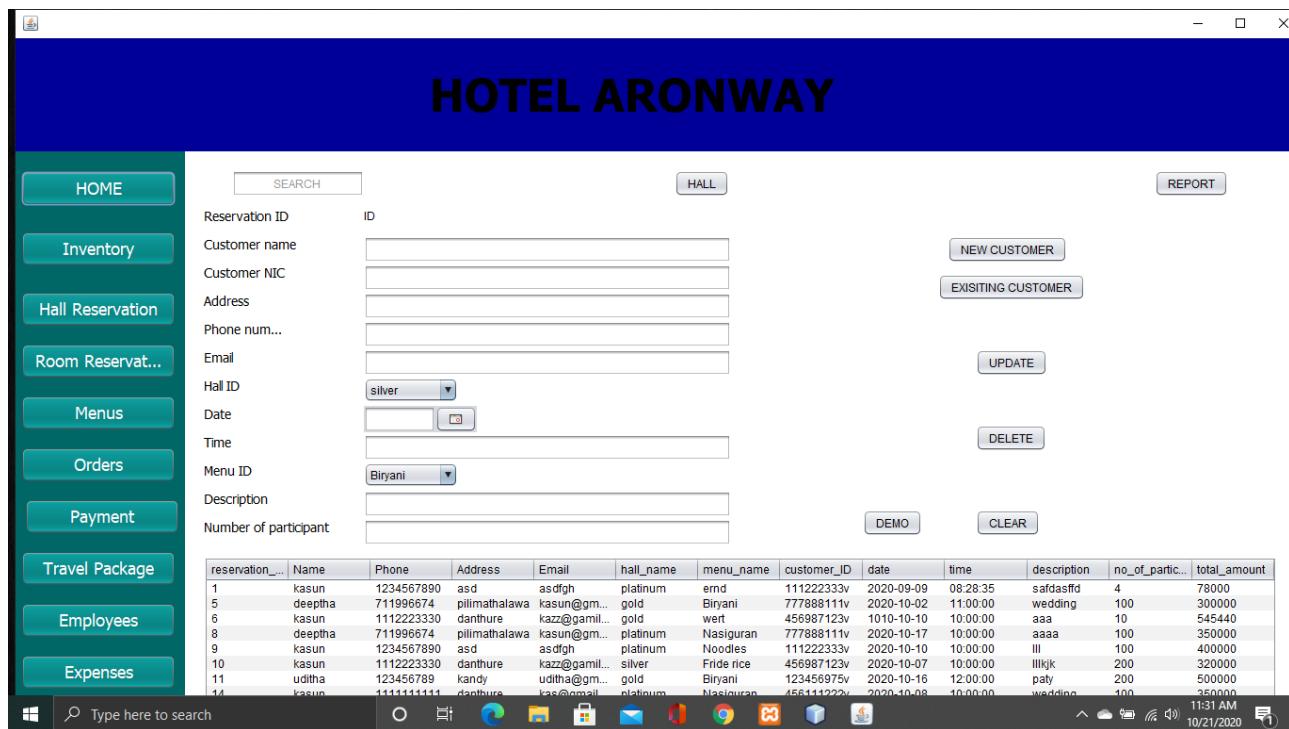


Figure 47: Hall Reservation Interface

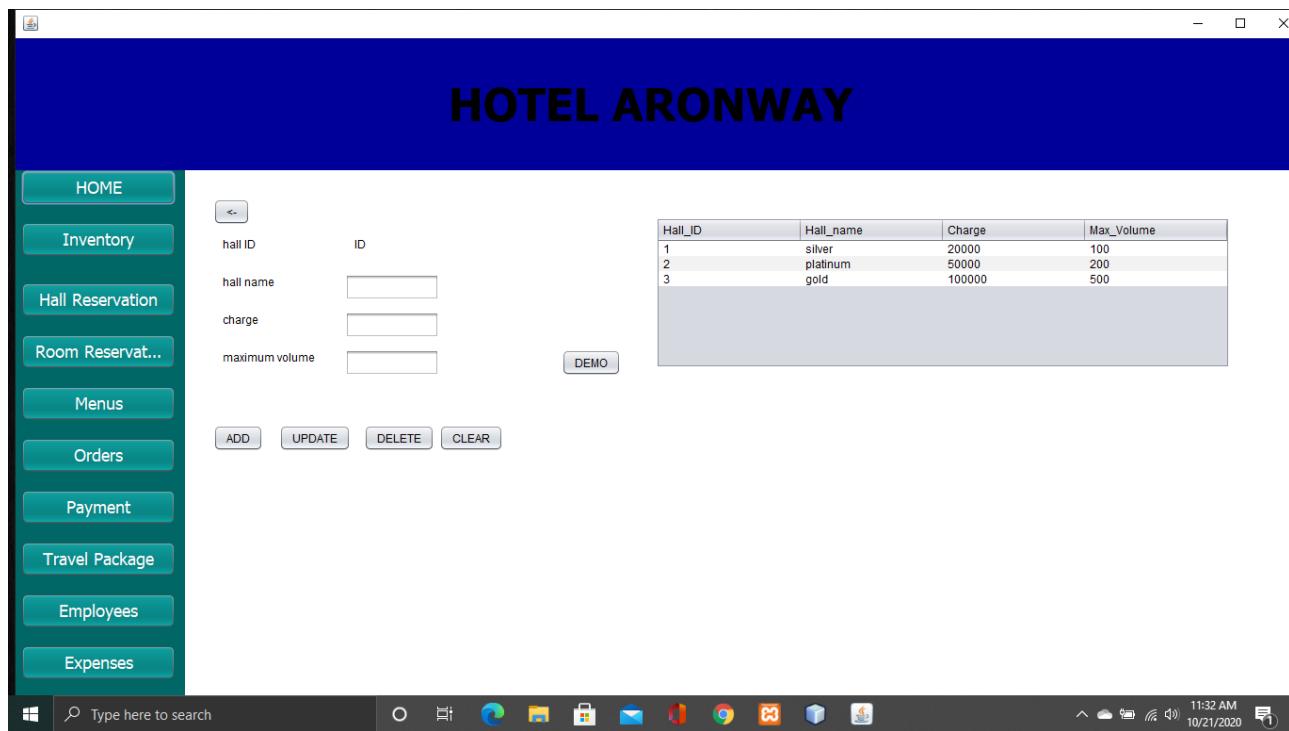


Figure 48: Hall Adding Interface

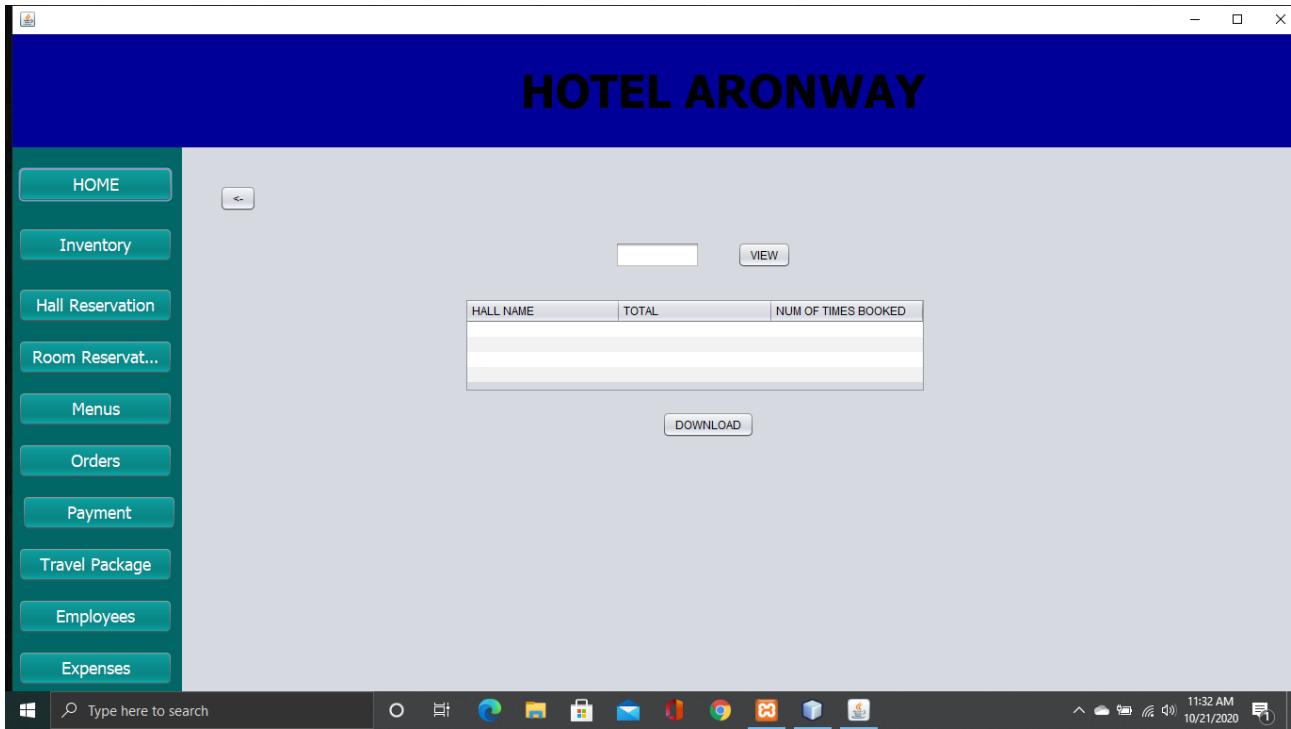


Figure 49: Hall Reservation Report Generate Interface

2.2.6.3 Employee Handling and Salary Management Interfaces

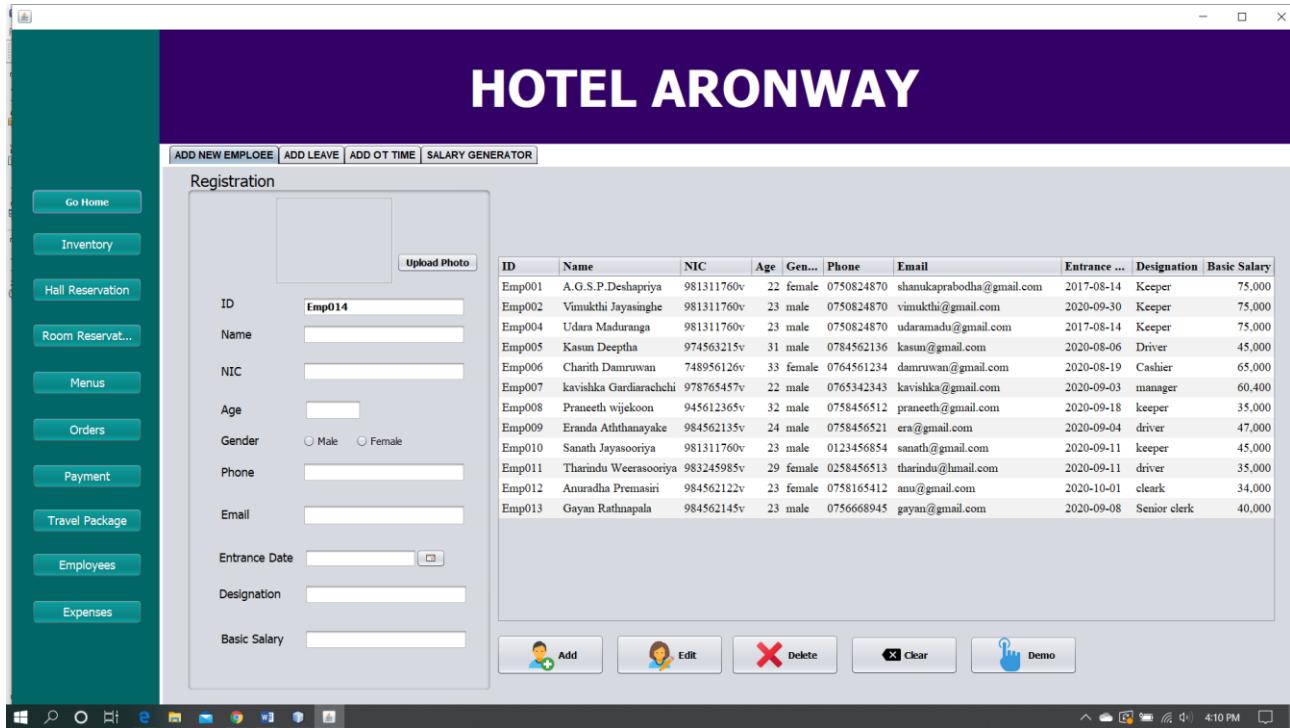


Figure 50: Employee Handling Interface

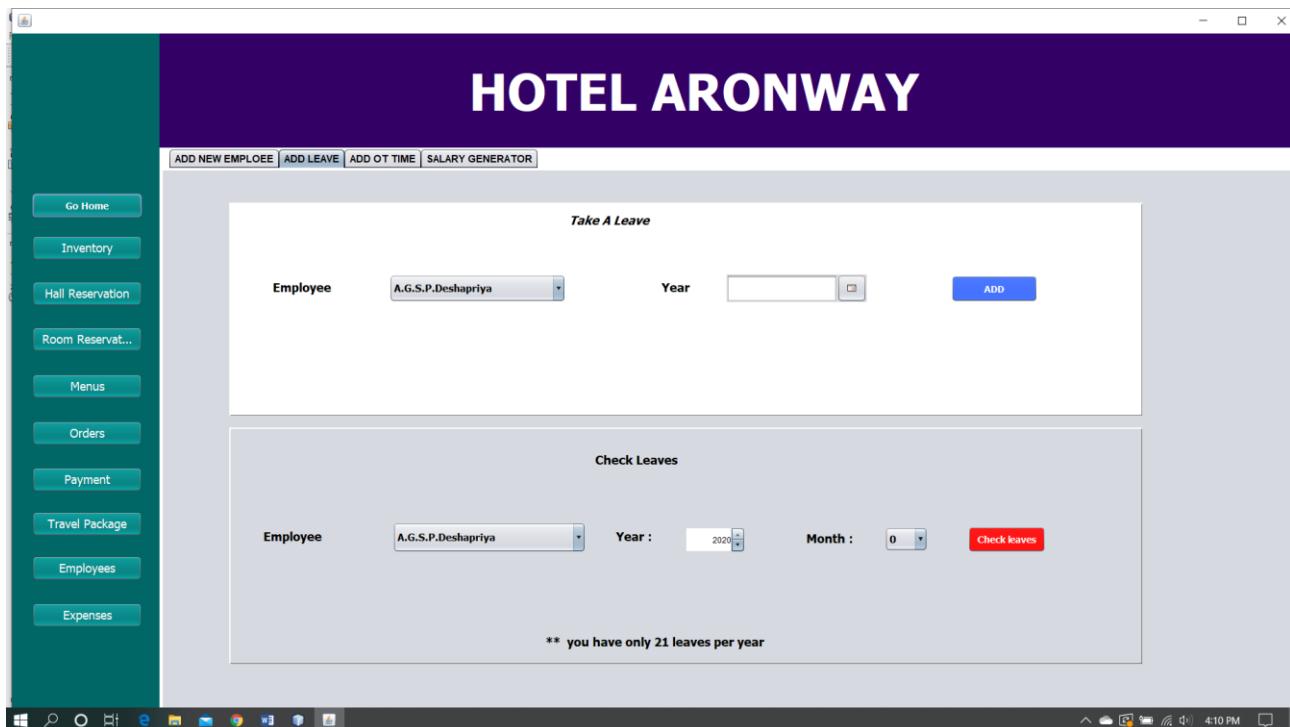


Figure 51: Leaves Adding Interface

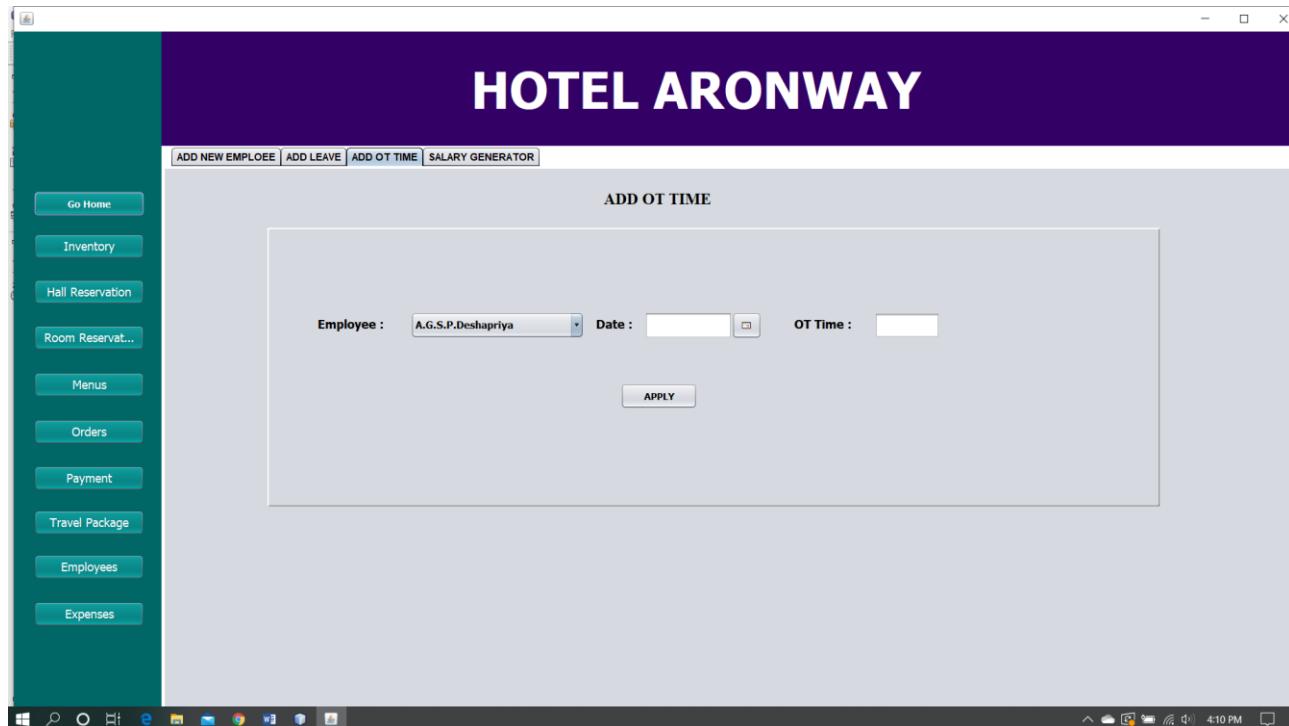


Figure 52: OT Adding Interface

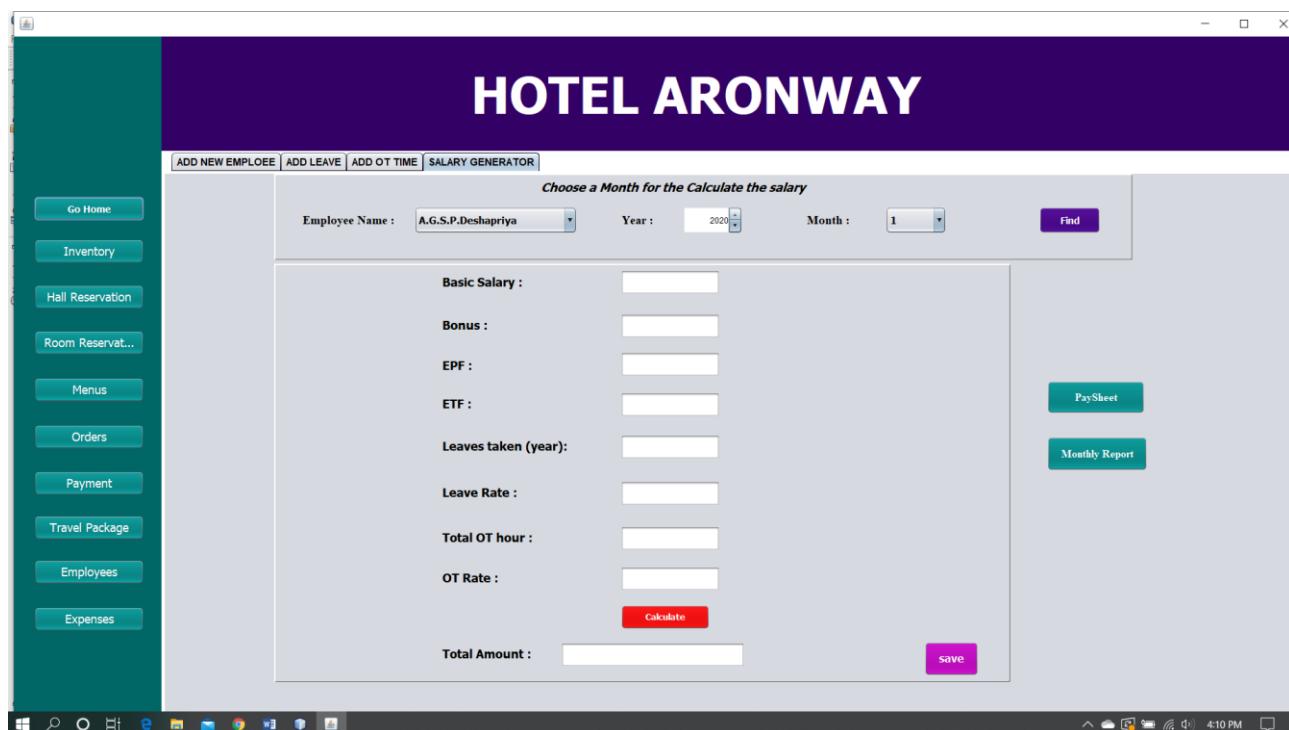


Figure 53: Salary Generator Interface

2.2.6.4 Room Reservation Interfaces

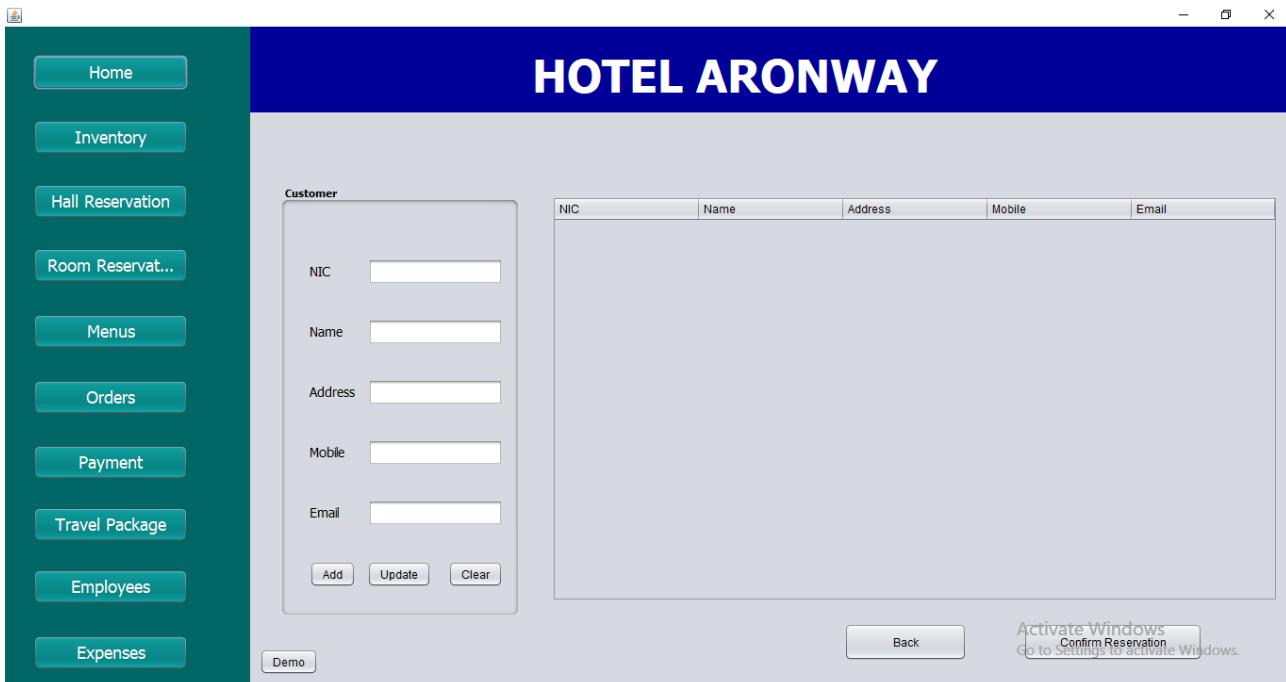


Figure 54: Room Reservation Interface

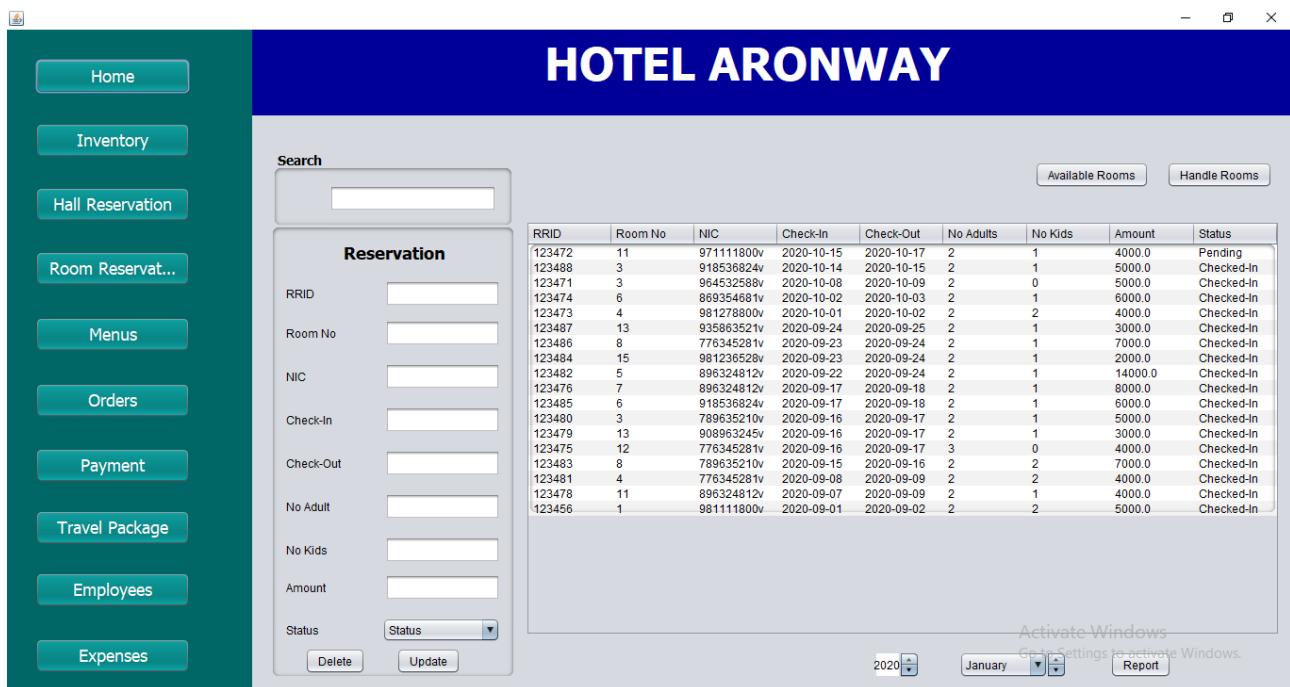


Figure 55 : Delete / Update Room Interface

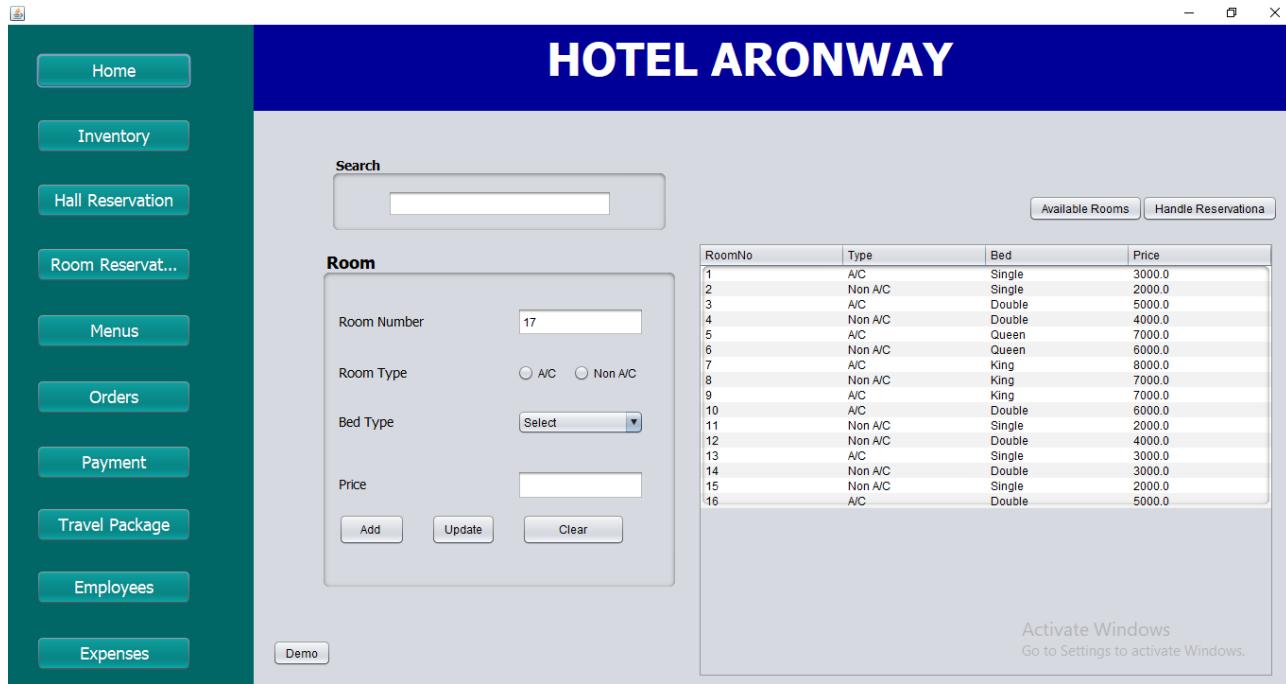


Figure 56 : Room Handling Interface

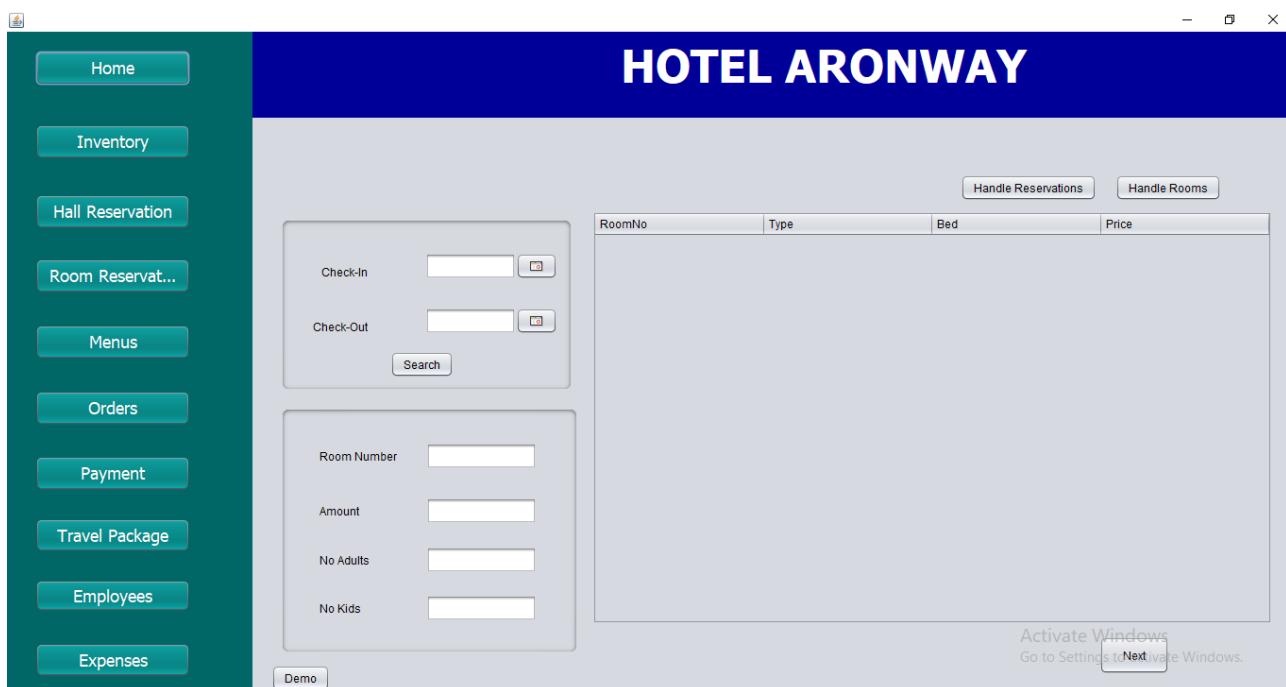


Figure 57: Room Checking Interface

2.2.6.5 Order Handling

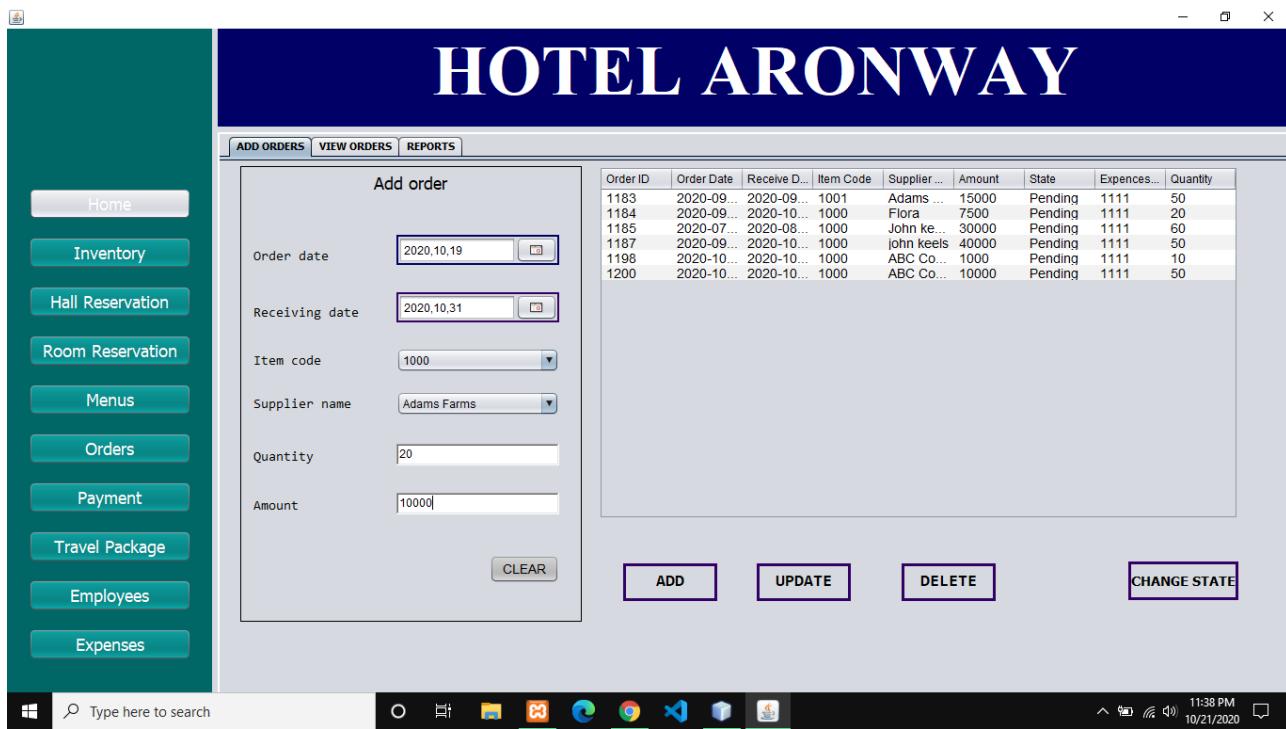


Figure 58 : Order Handling Interface

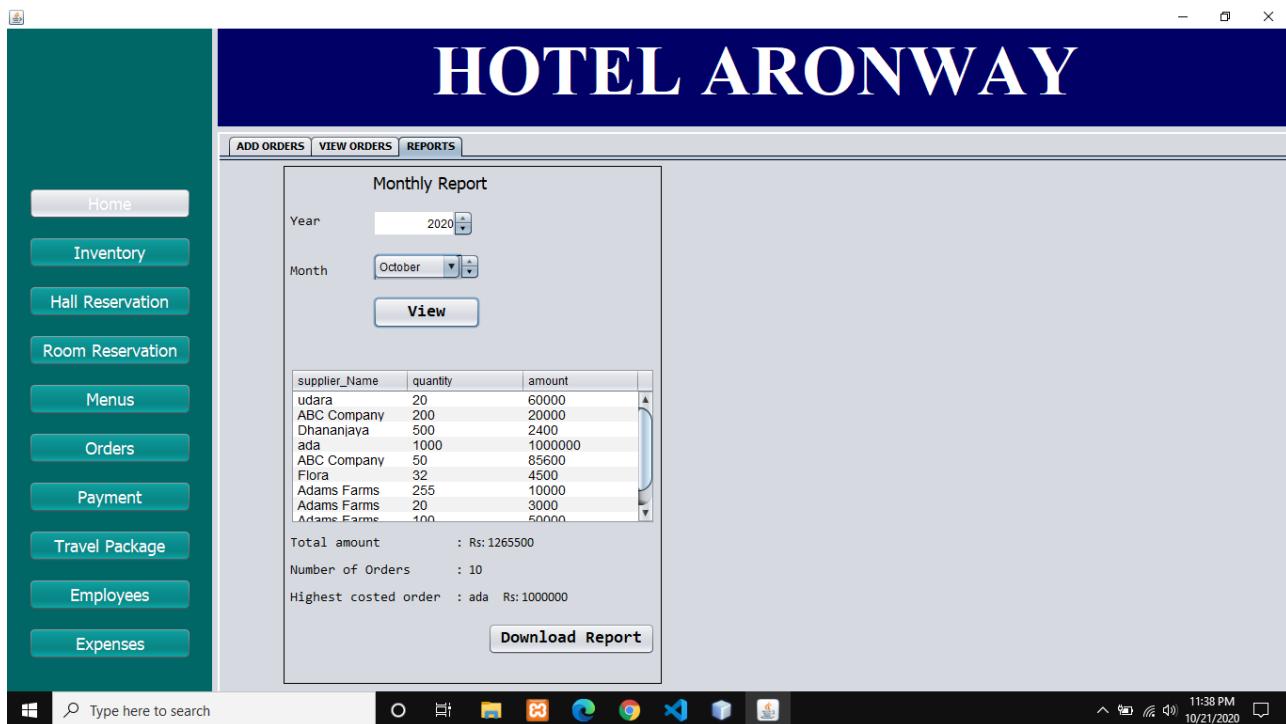


Figure 59: Generate Report Interface

2.2.6.6 Expenses Handling

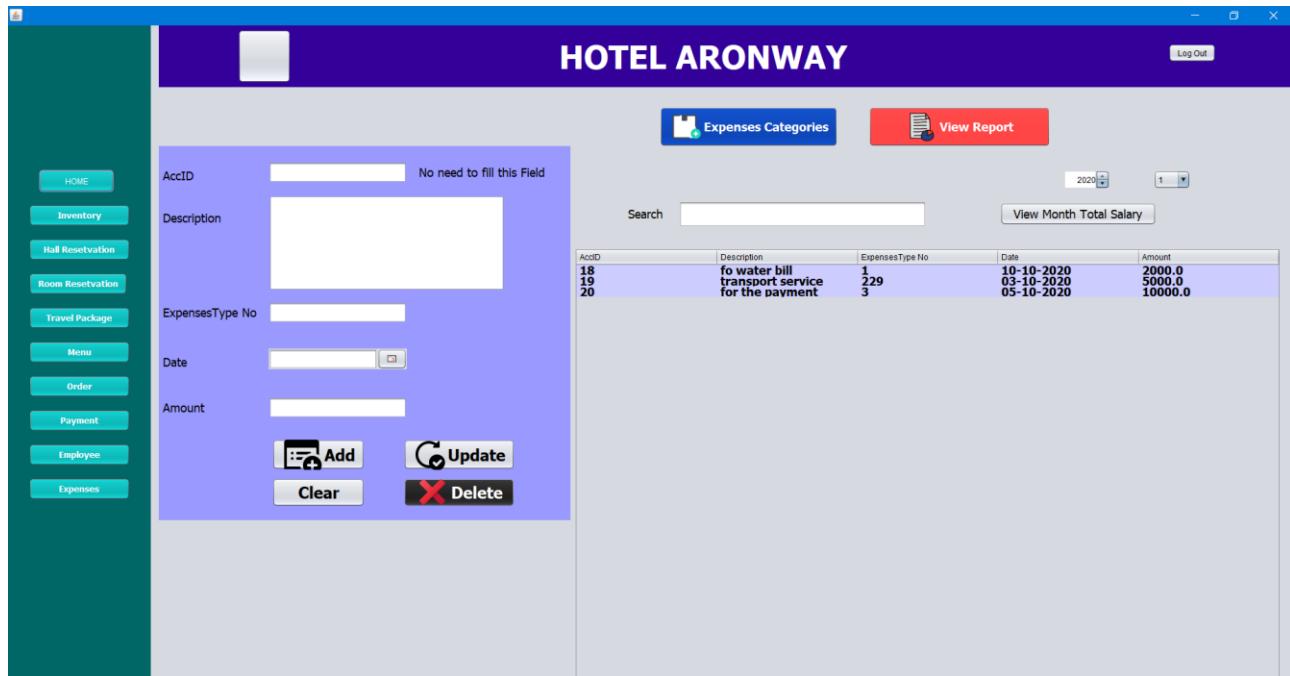


Figure 60: Expenses Handling Interface

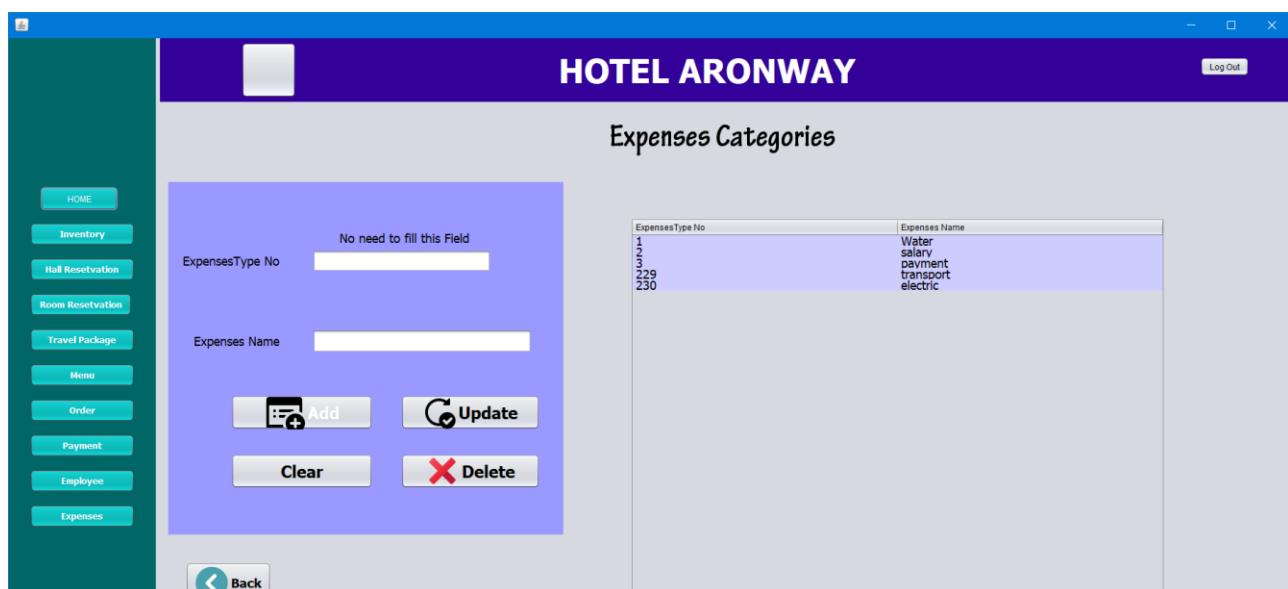


Figure 61: Expenses Category Handling Interface

2.2.6.7 Payment Handling

The screenshot shows the 'Payment Info' section of the Hotel Aronway system. On the left, a vertical menu bar lists various management functions: Home, Inventory, Hall Reservation, Room Reservation, Menus, Orders, Payment, Employees, Expenses, and Travel Package. The 'Payment' button is highlighted. The main area has a title 'HOTEL ARONWAY'. Below it, a 'Payment Info' form contains fields for Payment ID, Room Reservation ID, Date, Time, Hall Reservation ID, Payment Type (radio buttons for Cash or Card), and Final Amount. It includes four buttons: 'Add' (with a plus sign icon), 'UPDATE' (with a circular arrow icon), 'DELETE' (with a red X icon), and 'Clear' (with a broom icon). Above the form are date and time inputs ('2020-10-01 01') and a 'Get Report' button. To the right is a table titled 'Payment History' showing transaction details:

Payment ID	Room ID	Date	Time	Hall ID	Payment Type	Final Amount
37	123456	2020-10-01	07:46:40	0	cash	12945.0
38	123458	2020-10-02	07:46:40	0	cash	3000.0
39	123459	2020-10-03	07:46:40	0	card	12445.0
40	123460	2020-10-04	07:47:51	0	cash	6000.0
42	123461	2020-10-05	07:48:53	0	card	16000.0
43	0	2020-10-06	07:52:29	1	cash	80000.0
44	0	2020-10-07	10:49:00	3	card	320000.0
47	0	2020-10-13	09:59:17	15	card	450000.0

A search bar labeled 'Search By Date' is located at the top right, along with a 'GO' button. A 'Delete History' button is visible at the bottom right of the main panel.

Figure 62: Payment Handling Interface

This screenshot shows the 'Deleted Payment History' section. The left sidebar remains the same as in Figure 62. The main area has a title 'HOTEL ARONWAY'. Below it, a 'Deleted Payment History' form contains fields for Payment ID, Room Reservation ID, Date, Time, Hall Reservation ID, Payment Type (radio buttons for Cash or Card), and Final Amount. It includes three buttons: 'DELETE' (with a red X icon), 'Undo' (with a circular arrow icon), and 'Clear' (with a broom icon). Above the form are date and time inputs ('2020-10-01 01') and a 'Back' button. To the right is a table titled 'Deleted Payment History' showing the same transaction details as the main payment history table:

Payment ID	Room ID	Date	Time	Hall ID	Payment Type	Final Amount
32	0	2020-10-07	02:54:14	3	cash	720000.0
33	123456	2020-10-08	03:55:55	0	cash	4000.0
34	0	2020-10-13	05:11:28	3	card	720000.0
35	0	2020-10-13	06:29:11	3	cash	720000.0

A search bar labeled 'Search By Date' is located at the top right, along with a 'GO' button.

Figure 63: Payment History Handling Interface

2.2.6.8 Menus Handling

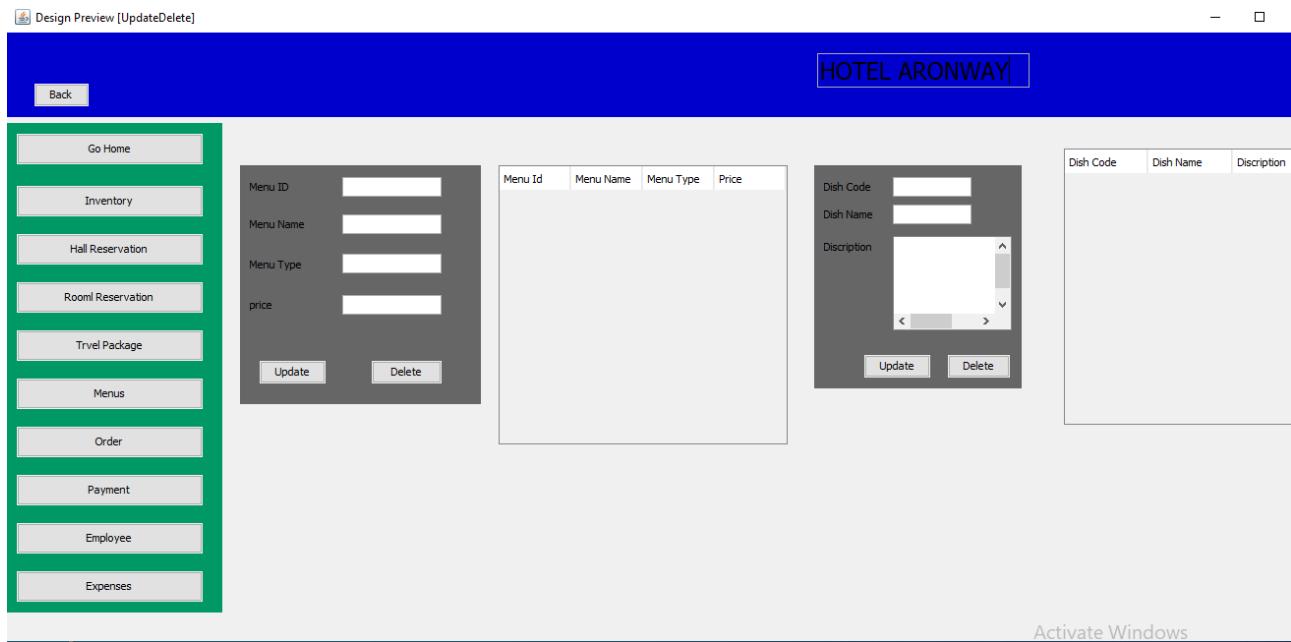


Figure 64 : Update / Delete Menu Interface

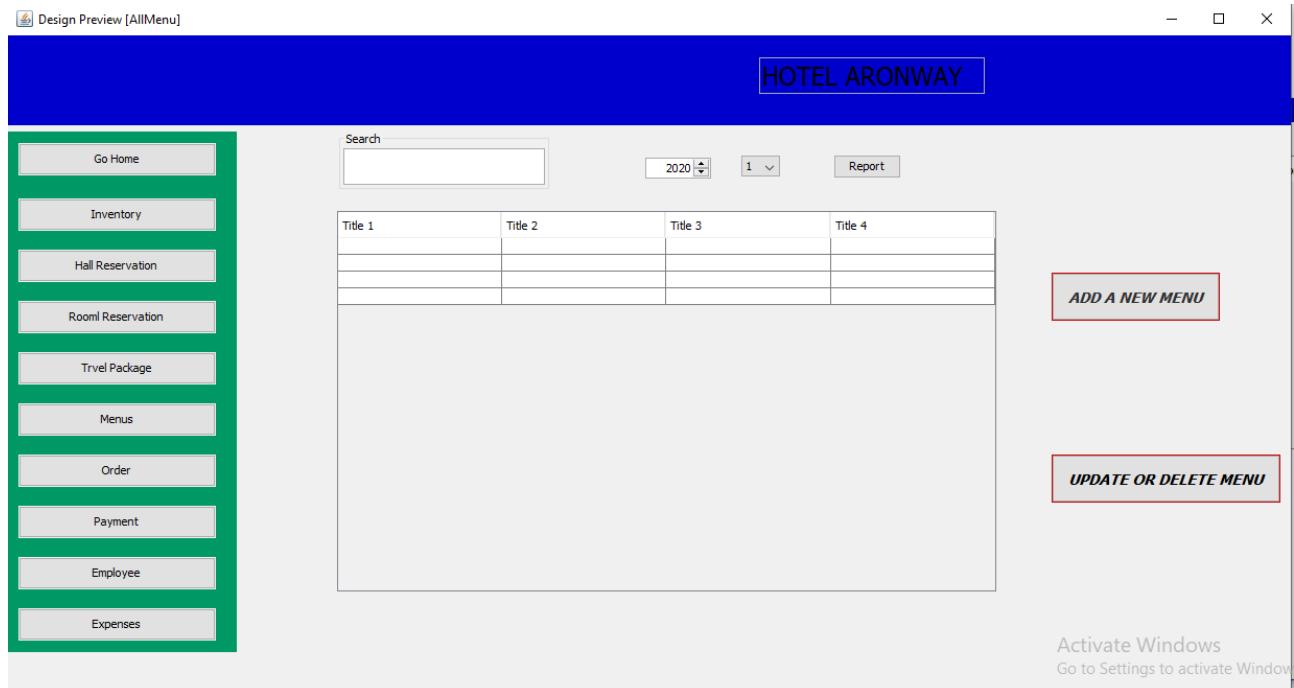


Figure 65 : Search and Generate Report Interface

Design Preview [AddNewMenu]

HOTEL ARONWAY

Back

ADD NEW MENU

Menu ID	Menu Name	Menu Type	Price
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Dish Code	Dish name	Description
<input type="text"/>	<input type="text"/>	<input type="text"/>
Description <input type="text"/>		
Add Dish		

Menu ID	Dish Code
<input type="text"/>	<input type="text"/>

Activate Windows
Go to Settings to activate Windows.

Go Home
Inventory
Hall Reservation
Room Reservation
Travel Package
Menus
Order
Payment
Employee
Expenses

Figure 66: Menu Adding Interface

2.2.6.9 Travel Package Handling Interfaces

The screenshot shows the 'Travel Package Handling' interface for Hotel Aronway. On the left is a vertical menu bar with buttons for Home, Inventory, Hall Reservation, Room Reservation, Menus, Orders, Payment, Travel Package, Employees, and Expenses. The main area has a purple header with the hotel name. Below it is a sub-header 'Travel Package Handling' with a 'Back' button. A form titled 'Insert Travel Details' contains fields for Room Reservation ID, Employee ID, Travel Location, Travel Starting Date, Travel Ending Date, Travel Fee, and No: of Participants. To the right of the form is a data grid displaying travel package records:

Travel ID	RRID	EID	Travel Locati.	Start Date	End Date	Fees	Participations
10026	1	emp001	Kandy	2020-09-11	2020-09-13	8500	2
10031	4	emp004	Jaffna	2020-09-11	2020-09-17	75546	1
10040	5	emp005	Kandy	2020-10-01	2020-10-02	8500	1
10041	2	emp004	Kandy	2020-10-08	2020-10-11	20000	2

At the bottom are buttons for Insert, Update, and Delete.

Figure 67: Travel Package Handling Interface

The screenshot shows the 'Travel Package Handling' interface for Hotel Aronway. The layout is similar to Figure 67, with a vertical menu on the left and a main area with a purple header. The sub-header 'Travel Package Handling' includes a 'Search by month:' input field. Below it is a data grid displaying travel package records, identical to the one in Figure 67. At the bottom of the main area, there is a message 'Total: Rs.112546.00' and a 'Print Report' button.

Travel ID	RRID	EID	Travel Location	Start Date	End Date	Fees	Participations
10026	1	emp001	Kandy	2020-09-11	2020-09-13	8500	2
10031	4	emp004	Jaffna	2020-09-11	2020-09-17	75546	1
10040	5	emp005	Kandy	2020-10-01	2020-10-02	8500	1
10041	2	emp004	Kandy	2020-10-08	2020-10-11	20000	2

Figure 68: Travel Package Search Interface

Travel ID	RRID	EID	Travel Location	Start Date	End Date	Fees	Participants
10026	1	emp001	Kandy	2020-09-11	2020-09-13	8500	2
10031	4	emp004	Jaffna	2020-09-11	2020-09-17	75546	1
10040	5	emp005	Kandy	2020-10-01	2020-10-02	8500	1
10041	2	emp004	Kandy	2020-10-08	2020-10-11	20000	2

Figure 69: Travel Package Generate Report Interface

2.3 Implementation

2.3.1 Database Management System

The choice of DBMS of this project was XAMPP. XAMPP is an open-source software developed by Apache Friends. The full form of XAMPP is X stands for Cross-platform, (A) Apache server, (M) MariaDB, (P) PHP and (P) Perl.[1]

Here are the reasons why our team decided to use XAMPP as our DBMS.

- XAMPP is easy to configure and use.
- It is cross-platform software available for all kinds of operating systems such as Linux and Windows.
- It has the essential modules such as PhpMyAdmin, OpenSSL and Joomla.
- Setup environment for development, testing and deployment is very simple and easy.[2]
- It handles many administrative tasks, such as status and security inspections.[2]

2.3.2 Implementation Language

The choice of implementation language of this project was JAVA. JAVA SWING was used as the graphical user interface widget toolkit.

Here are the reasons why our team decided to use JAVA as our implementation language.

- Multithreaded Environment
- Efficient Memory Storage
- Rich-set of APIs
- Independent Language
- Object-Oriented Nature of Language

2.4 Testing

2.4.1 Inventory Management

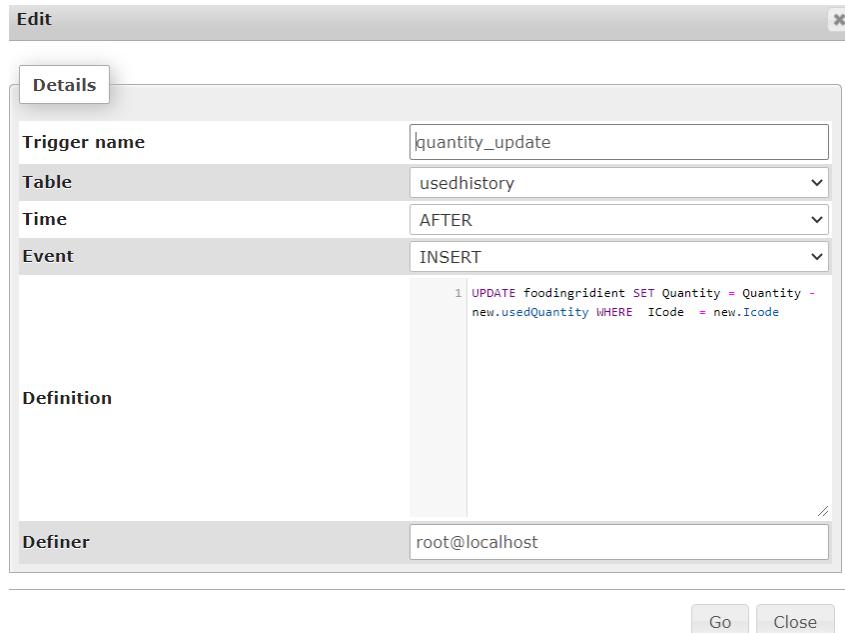


Figure 70: Test Case of Used History deduct

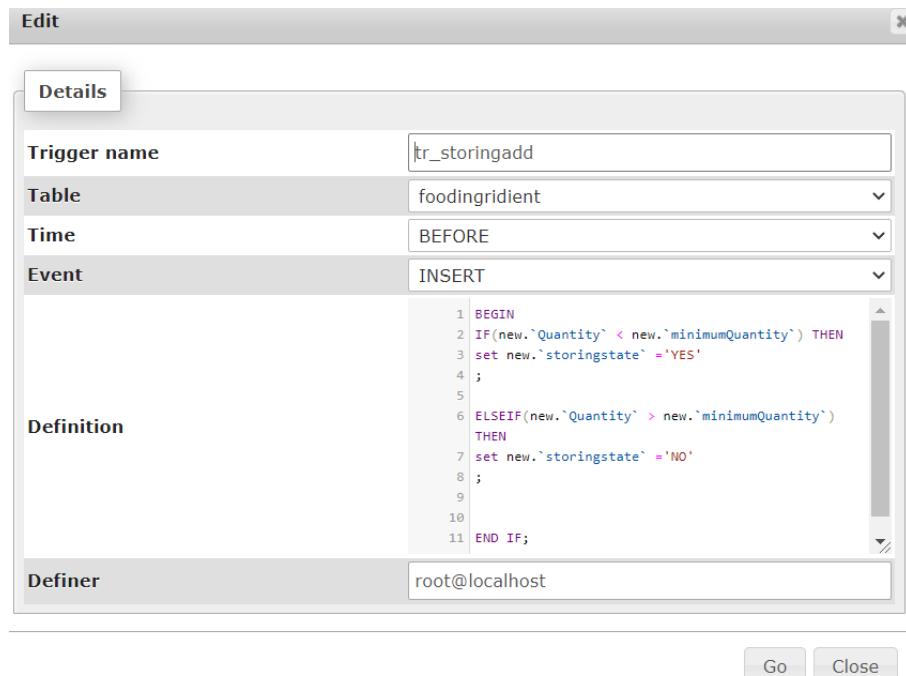


Figure 71: Test Case of Storing State

Table 3: Test Cases for Inventory Management

Test ID	Test Inputs	Expected Output	Actual Output	Test Result Pass/Fail	Description
TC01	Available quantity: 5 Minimum quantity :7	Storing state = YES	Storing state =Yes	Pass	If the available quantity of an ingredient is less than minimum quantity of that ingredient, the ingredient should restore so the storing state should be “YES”
TC02	Available quantity: 2 Used quantity: 2	new available quantity = 18	18	Pass	A particular ingredient used for some reason the used quantity must deduct from the current available quantity of that ingredient
TC03	Available quantity: 5 Used quantity: 6	error massage	-1	Fail	A particular ingredient used for some reason the used quantity must deduct from the current available quantity of that ingredient. But in this situation available quantity is less than input used quantity .So it must show an error message But the result shows as -1.Show test cased failed.

2.4.2 Hall Reservation

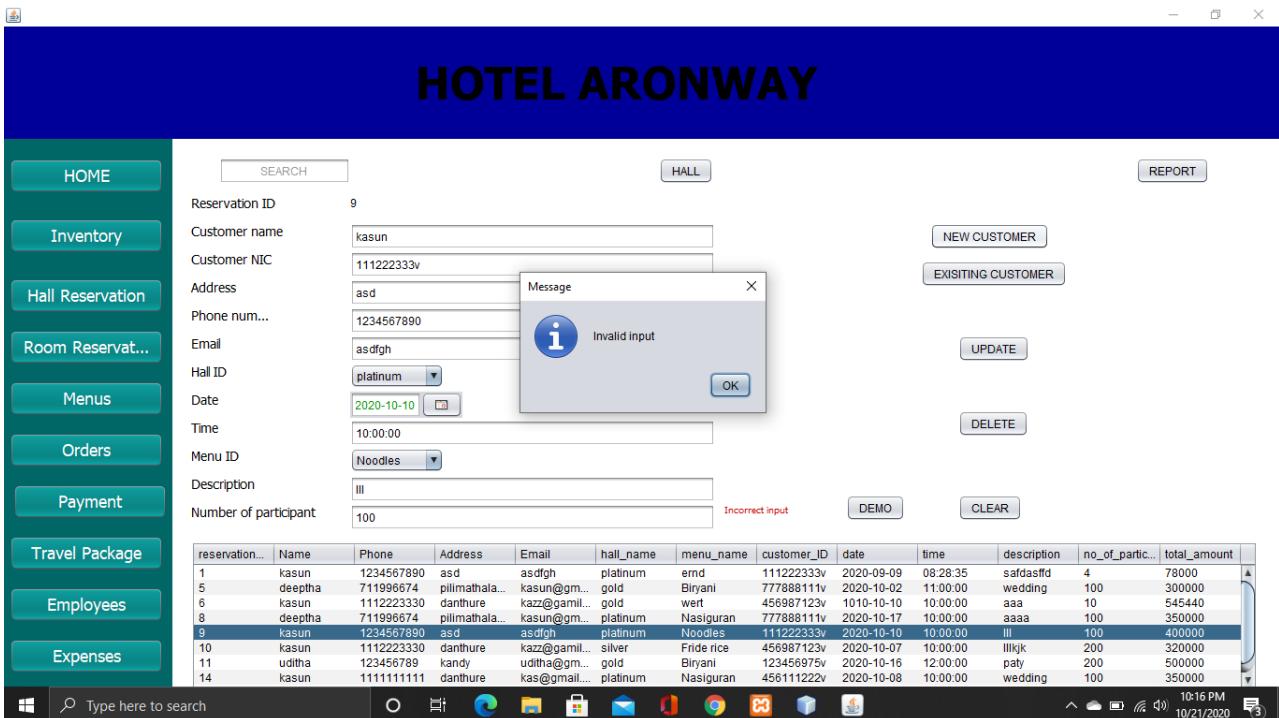


Figure 72: Hall Reservation Passed Test Case

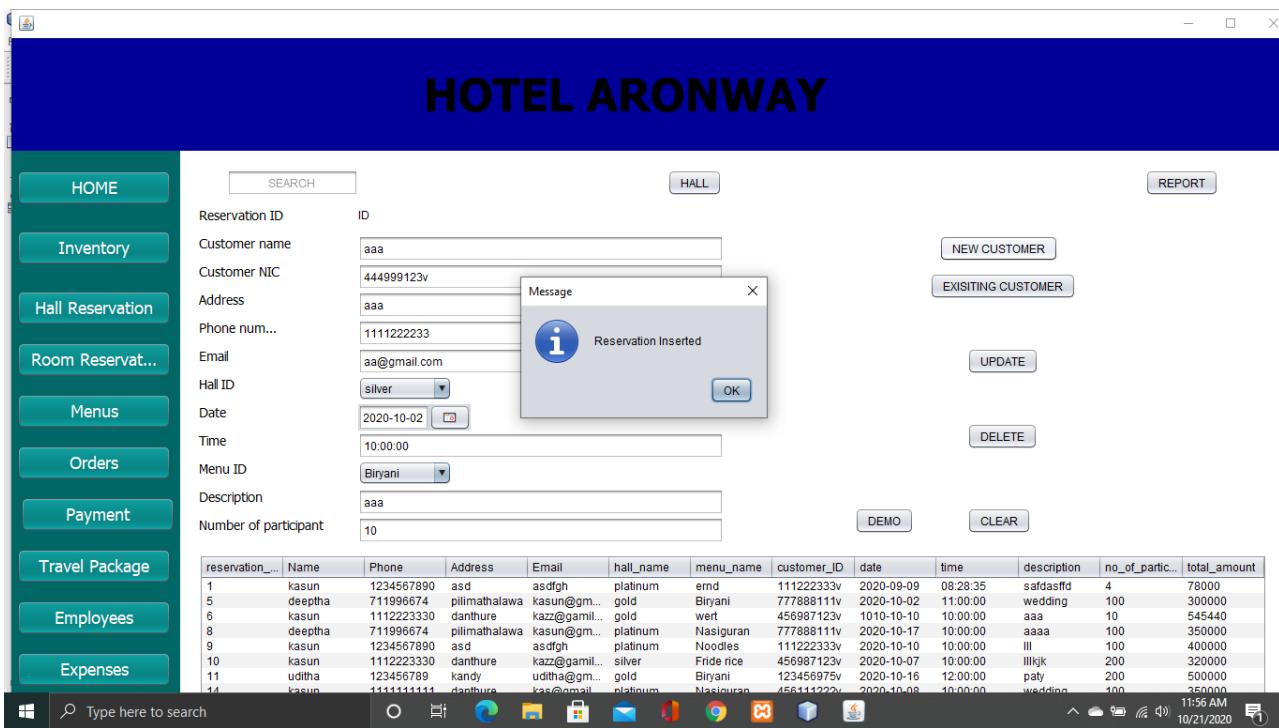


Figure 73: Hall Reservation Passed Test Case

Table 4: Test Cases for Hall Reservation

Test ID	Test Inputs	Expected Output	Actual Output	Test Result	Description
				Pass/Fail	
TC01	Hall name – gold Charge – 100000 Max volume - 500	Show success message after inserting data	Show “Hall added” message.	Pass	If all the details are correct input and there are no empty fields, hall will be added to the system.
TC02	Hall name – gold Charge – aaaa Max volume - 500	Show error message “invalid input ”	Show error message “invalid input ”	Pass	If enter the invalid inputs, it cannot add to the system

2.4.3 Employee Handling And Salary Management

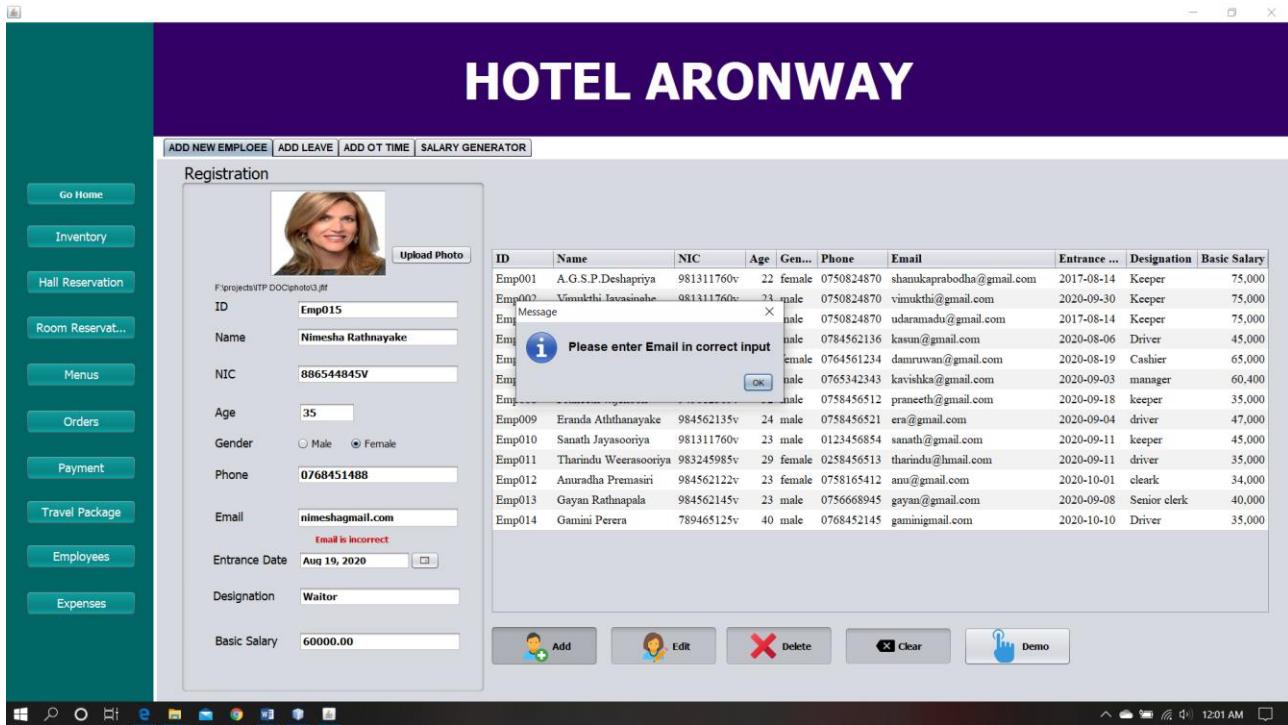


Figure 74: Add Employee Passed Test Case

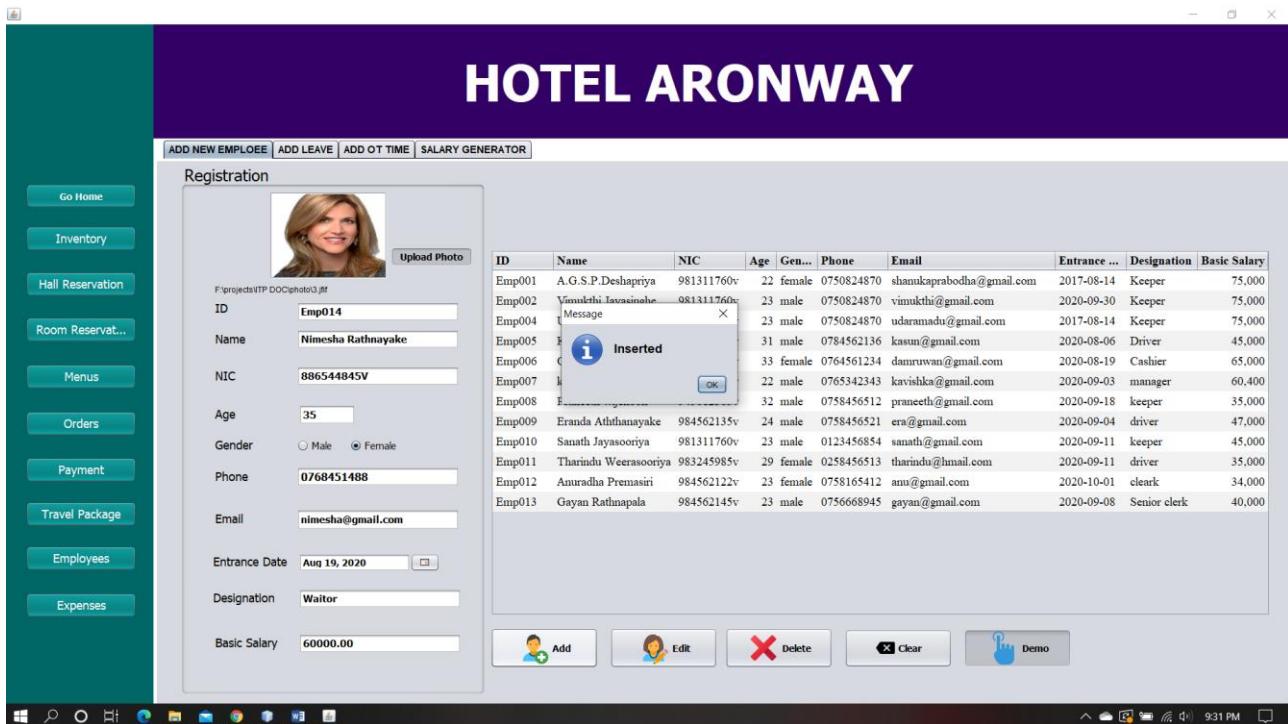


Figure 75: Add Employee Passed Test Case

Table 5: Test Cases for Employee Handling and Salary Management

Test ID	Test inputs	Expected output	Actual output	Result (Pass/Fail)	Description
T01	Name: Nimesha Rathnayake NIC:886544845v Age: 35 Gender: Female Email: nimehsa@gmail.com Entrance Date: Aug,19,2020 Designation:Waitor Basic Salary: 60000.00	Show successful message after add the Employee	Show “Inserted” message	Pass	If all the details are correct input and there are no empty fields, the Employee will be added to the system .
T02	Name: Nimesha Rathnayake NIC:886544845v Age: 35 Gender: Female Email: nimehsagmail.com Entrance Date: Aug,19,2020 Designation:Waitor Basic Salary: 60000.00	Show error message “Please enter Email in correct input”	Show error message “Please enter Email in correct input”	Pass	If enter the incorrect inputs, it cannot add to the system

2.4.4 Room Reservation

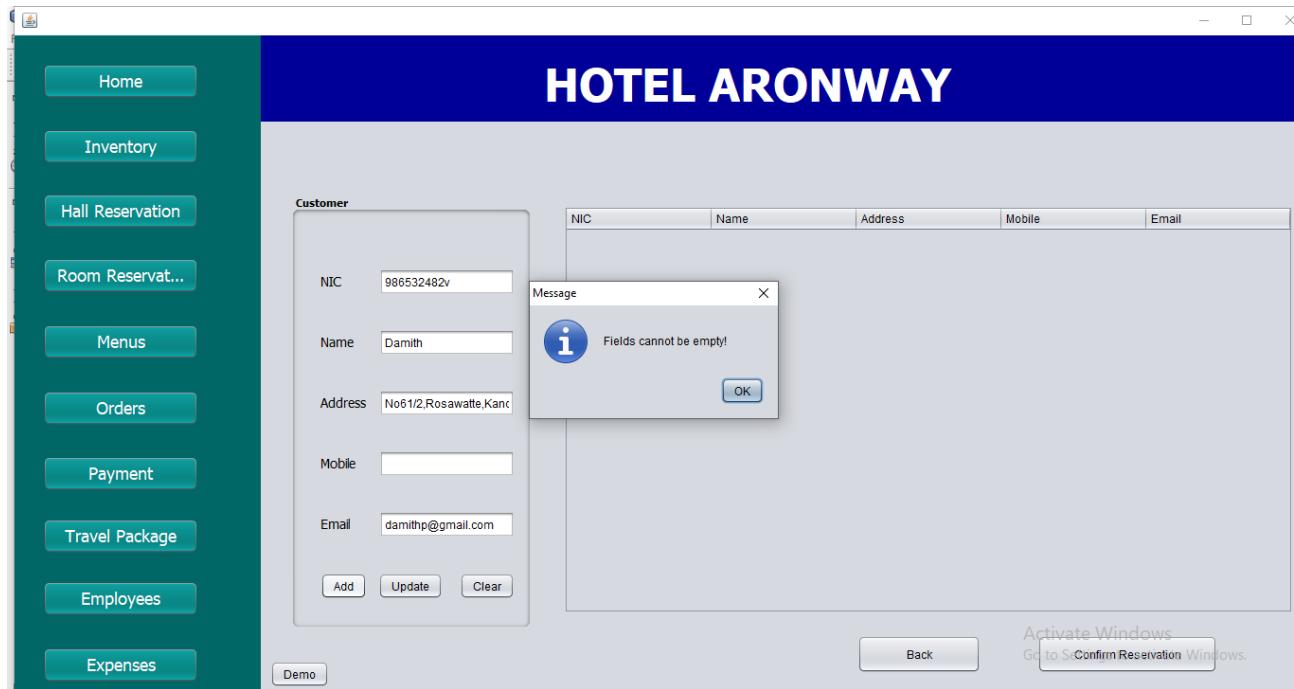


Figure 76: Room Reservation Passed Test Case

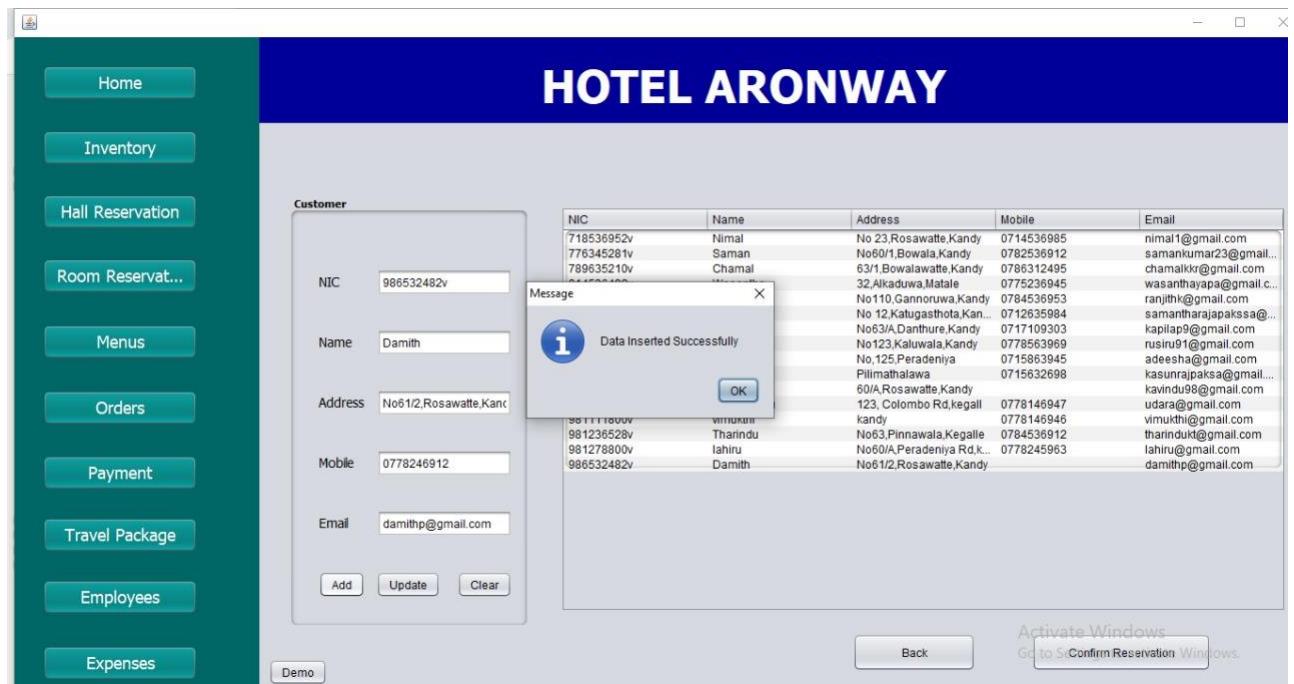


Figure 77: Room Reservation Passed Test Case

Table 6: Test Cases for Room Reservation

Test ID	Test inputs	Expected outputs	Actual output	Result (Pass/Fail)	Description
T01	NIC: 986532482v Name: Damith Address: No61/2,Rosawatte,Kandy Mobile: 0778246912 Email: damithp@gmail.com	Show success message after add the customer	Show success message	Pass	If all the inputs are correct and there are no empty fields, customer can add to the database
T02	NIC: 986532482v Name: Damith Address: No61/2,Rosawatte,Kandy Mobile: Email: damithp@gmail.com	Show error message “Fields cannot be empty”	Show error message “Fields cannot be empty”	Pass	If one or more fields are empty, customer cannot add to the database

2.4.5 Order Handling

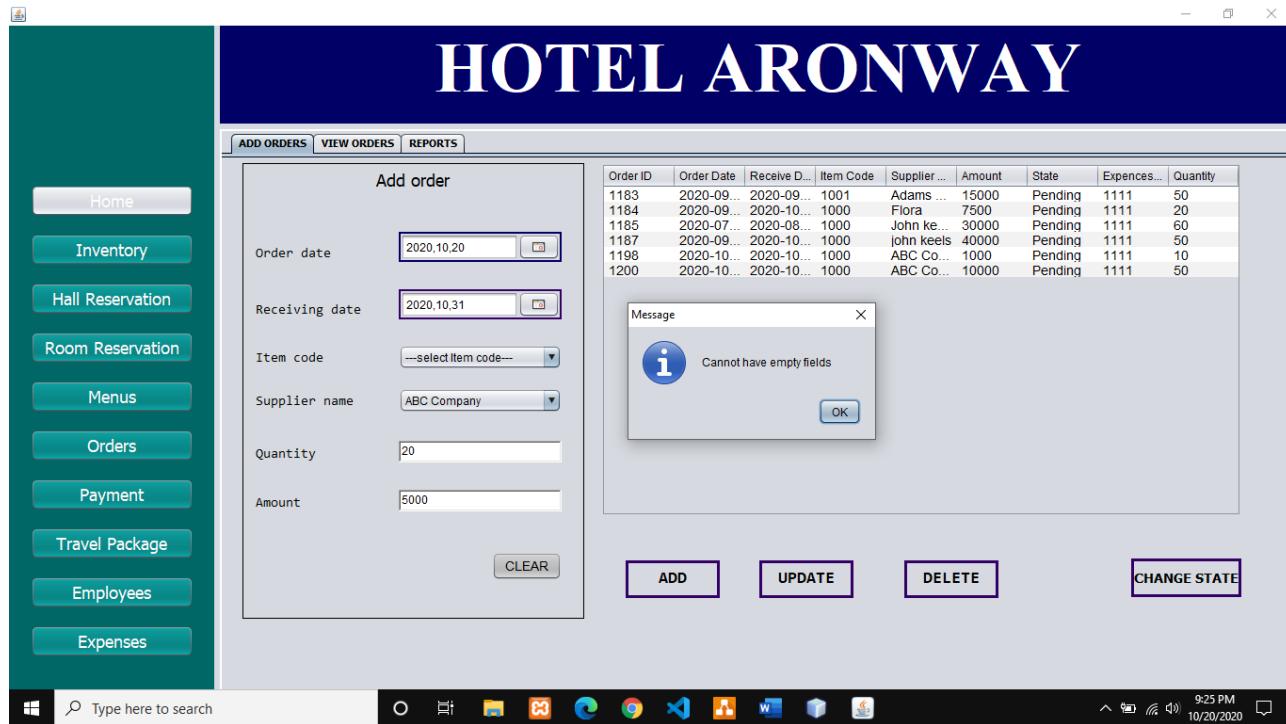


Figure 78: Add Order Passed Test Case

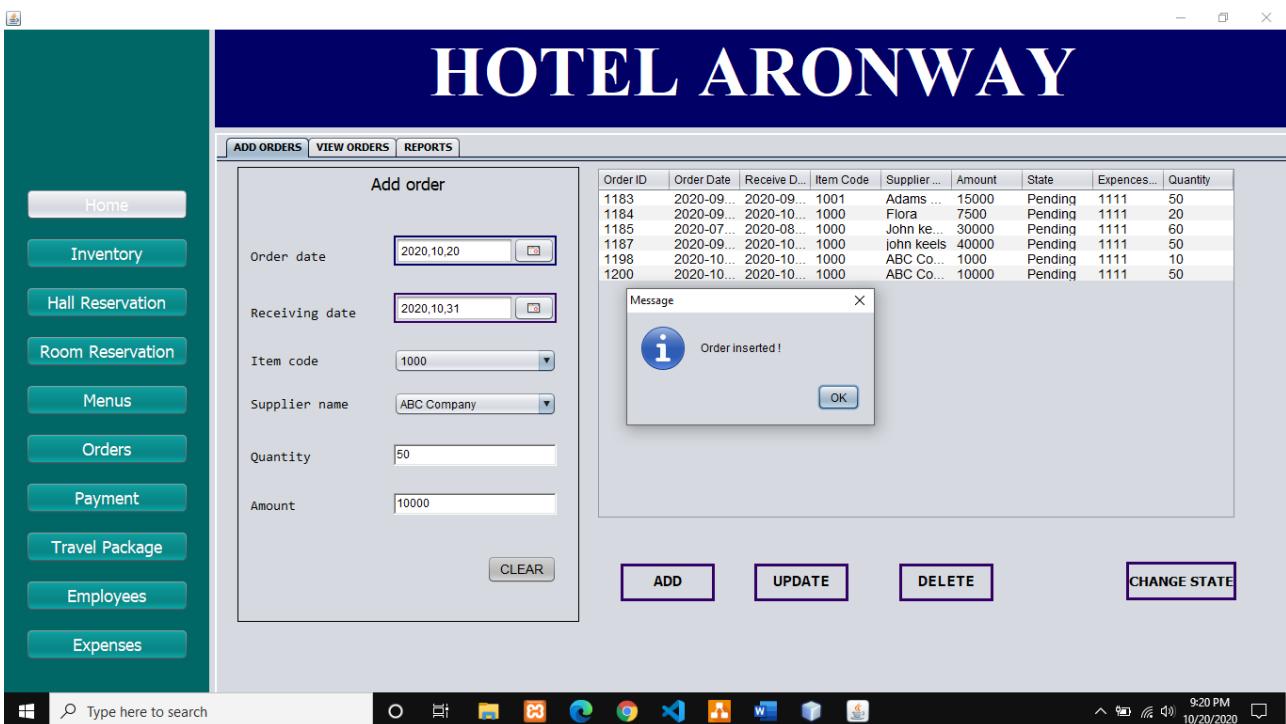


Figure 79: Add Order Passed Test Case

Table 7: Test Cases for Order Handling

Test ID	Test inputs	Expected output	Actual output	Result (Pass/Fail)	Description
T01	Order date: 2020/10/19 Receiving date: 2020/11/19 Item code: 1111 Supplier name: Adams Quantity: 100 Amount: 20000	Show successful message after add the order	Show successful message	Pass	If all the details are correct and there are no empty fields, the order will be added to the system
T02	Order date: 2020/10/19 Receiving date: 2020/11/19 Item code: Supplier name: Adams Quantity: 100 Amount: 20000	Show error message “Cannot have empty fields”	Show error message “Cannot have empty fields”	Pass	If one of the fields or more are empty the order cannot add to the system

2.4.6 Expences Handling

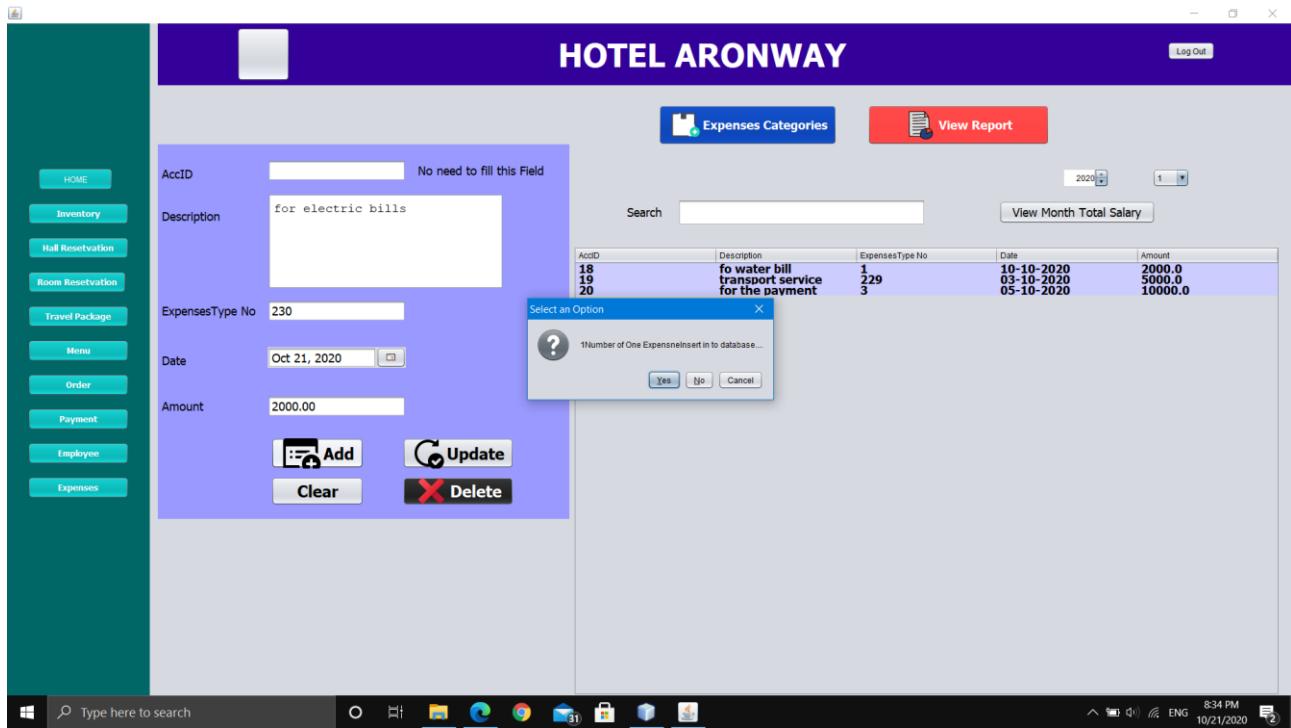


Figure 80: Add Expenses Passed Test Case

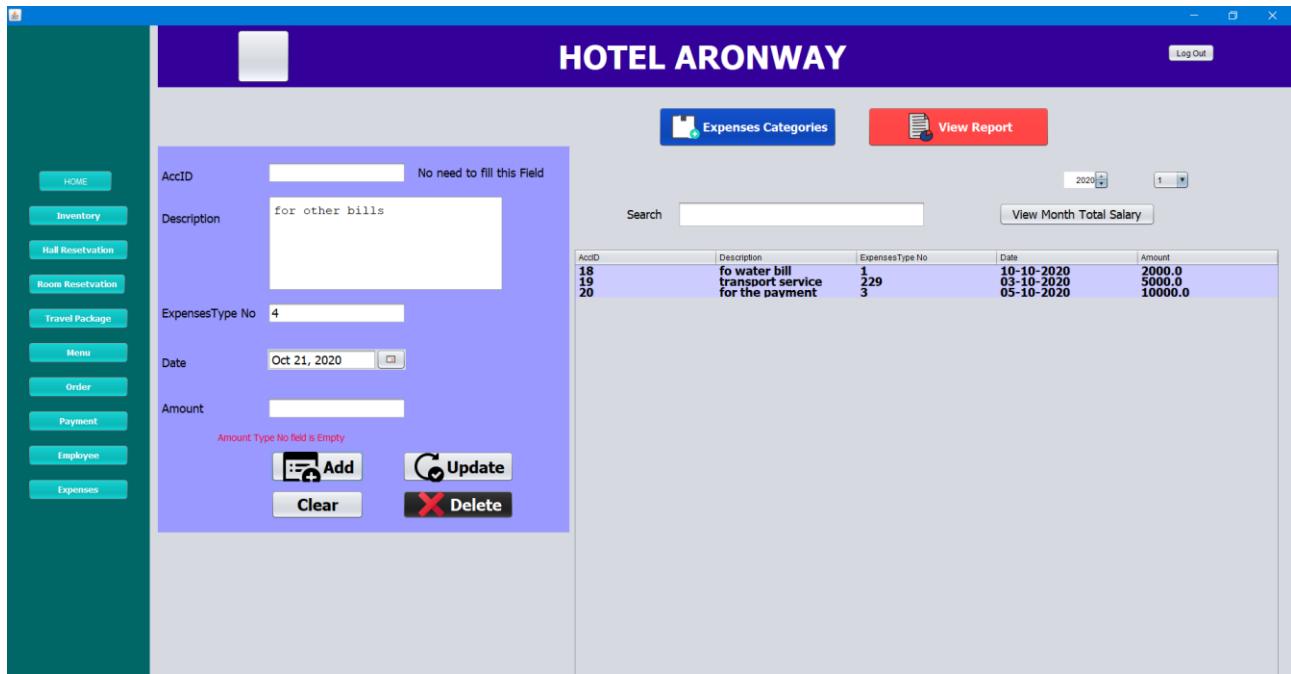


Figure 81: Add Expenses Passed Test Case

Table 8: Test Cases for Expences Handling

Test ID	Test Input	Expected Output	Actual output	Result (Pass/Fail)	Description
T01	Description : For other bills ExpensesType No : 4 Date : Oct 21,2020 Amont :	Show error message “Amont Type No field is empty”	Amount Type No field is empty	Pass	If one of the field or more field are empty expenses cannot add to the database.
T02	Description : For electric bills ExpensesType No : 230 Date : Oct 21, 2020 Amont : 2000.00	Show message “1 number of one expenses insert in to database”	1 number of one expenses insert in to database	pass	If all the details are correct and there are no empty fields, the expenses will be added in to database.

2.4.7 Payment Handling

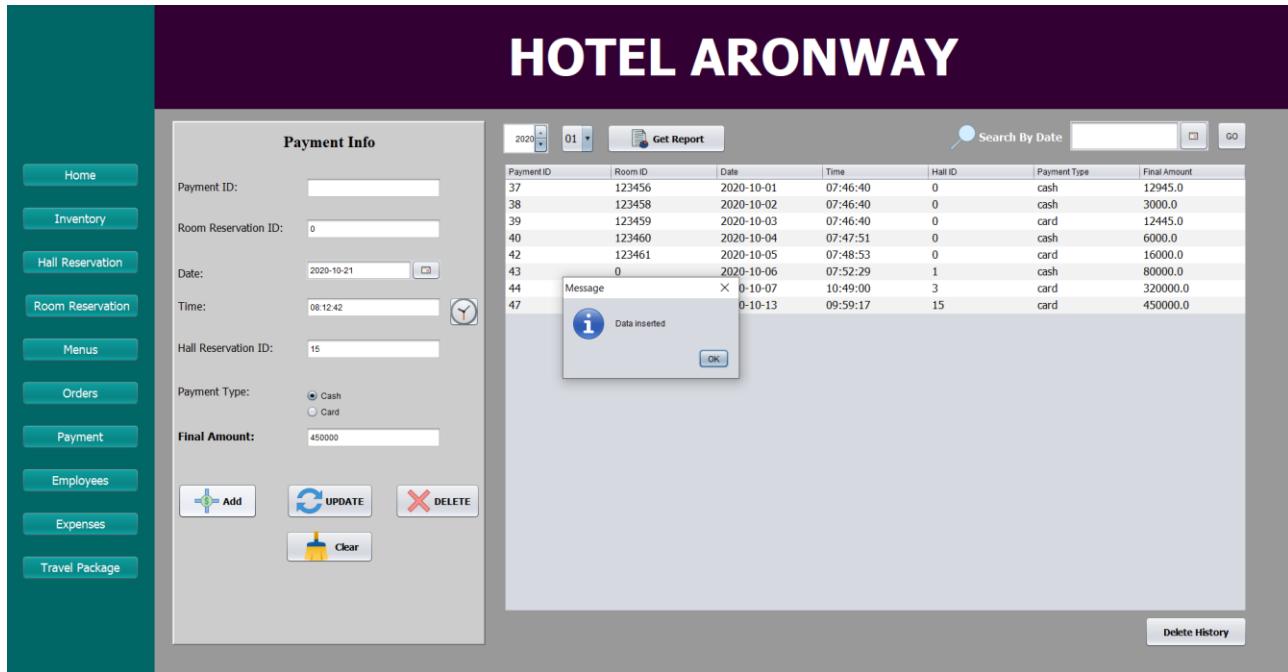


Figure 82: Add Payment Passed Test Case

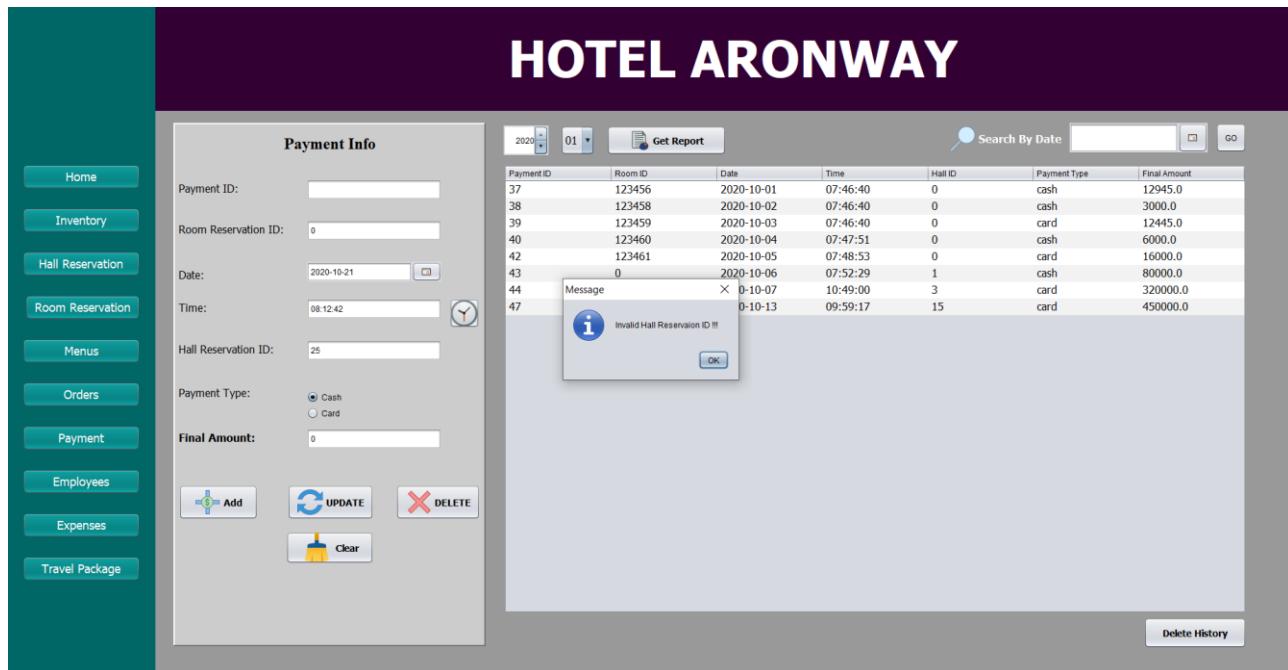


Figure 83: Add Payment Failed Test Case

Table 9: Test Cases for Payment Handling

Test ID	Test Input	Expected Output	Actual Output	Result	Description
P001	Room Reservation ID: 0 Date: 2020-10-21 Time: 08:12:42 Hall Reservation ID:25 Payment Type: cash	Display “Invalid Hall Reservation ID” Message	Display “Invalid Hall Reservation ID” Message	Pass	Entered hall reservation ID should already exists in Hall Reservation Database, else this should display an error message and Final amount should displayed as ‘0’.
P002	Room Reservation ID: 0 Date: 2020-10-21 Time: 08:12:42 Hall Reservation ID:15 Payment Type: cash	Display “Entered Hall Reservation ID already exists in Database” message	Display “Data Inserted” Message	Fail	Sine a reservation ID use only once, user should not be able to add another payment using already entered reservation ID.

2.4.8 Menus Handling

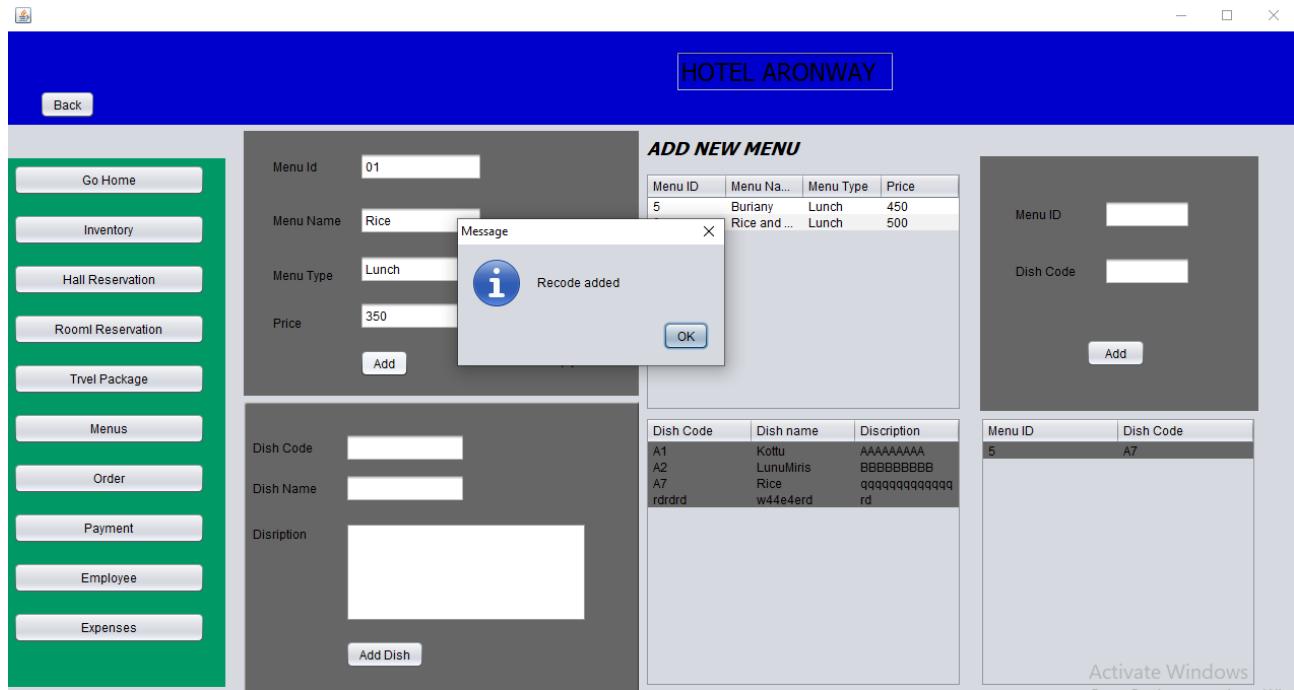


Figure 84: Add Menu Passed Test Case

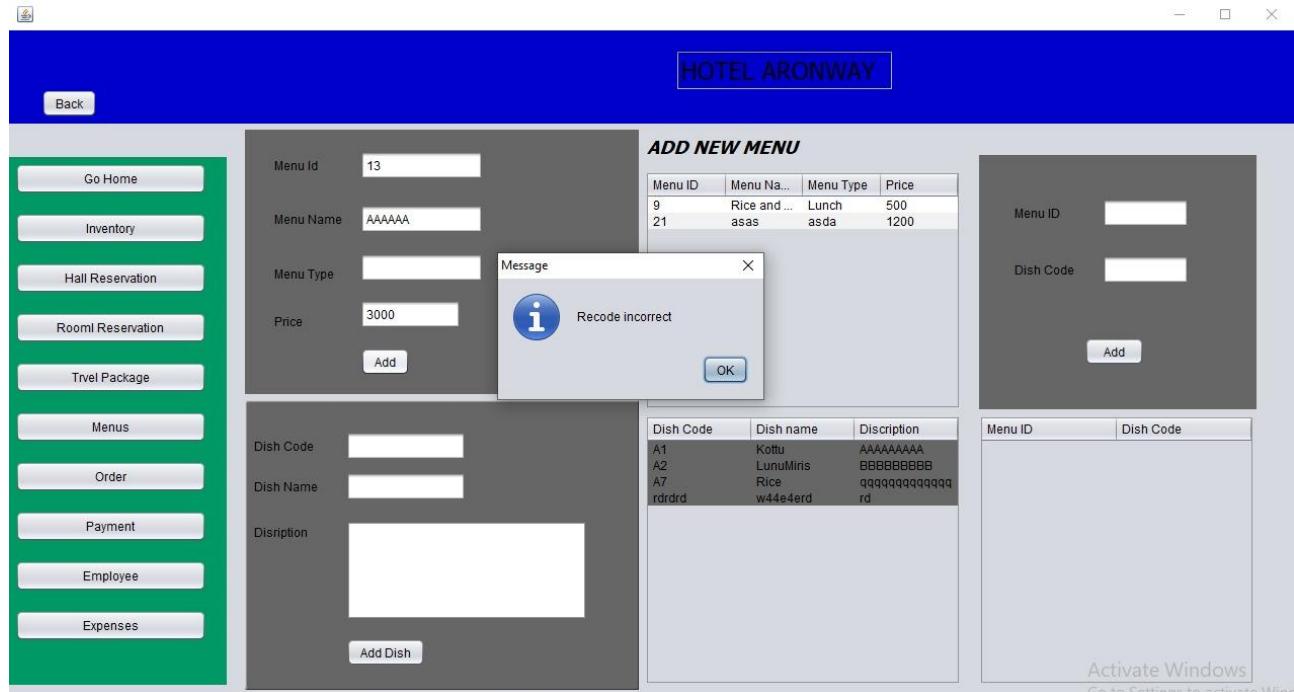


Figure 85: Add Menu Passed Test Case

Table 10: Test Cases for Menus Handling

Test ID	Test Inputs	Expected Output	Actual output	Result (Pass/Fail)	Description
A01	<ul style="list-style-type: none"> • Menu ID: 01 • Menu Name: Rice and Curry • Menu Type: Lunch • Description: AAAAAA • Dish Code: A01 • Dish Name: Rice, Parippu 	Show successful message after adding the menu	Show successful message	Pass	If enter Correct details the details will be added to the system and if there are any empty fields the data will be Cannot be added to the system
A02	<ul style="list-style-type: none"> • Menu ID: AA • Menu Name: Rice and Curry • Menu Type: Lunch • Description: AAAAAA • Dish Code: A01 • Dish Name: Rice, Parippu 	Show Error message “Menu ID is invalid”	Show Error message “Menu ID is invalid”	Pass	If enter Incorrect details cannot added to the system

2.4.9 Travel Package Handling

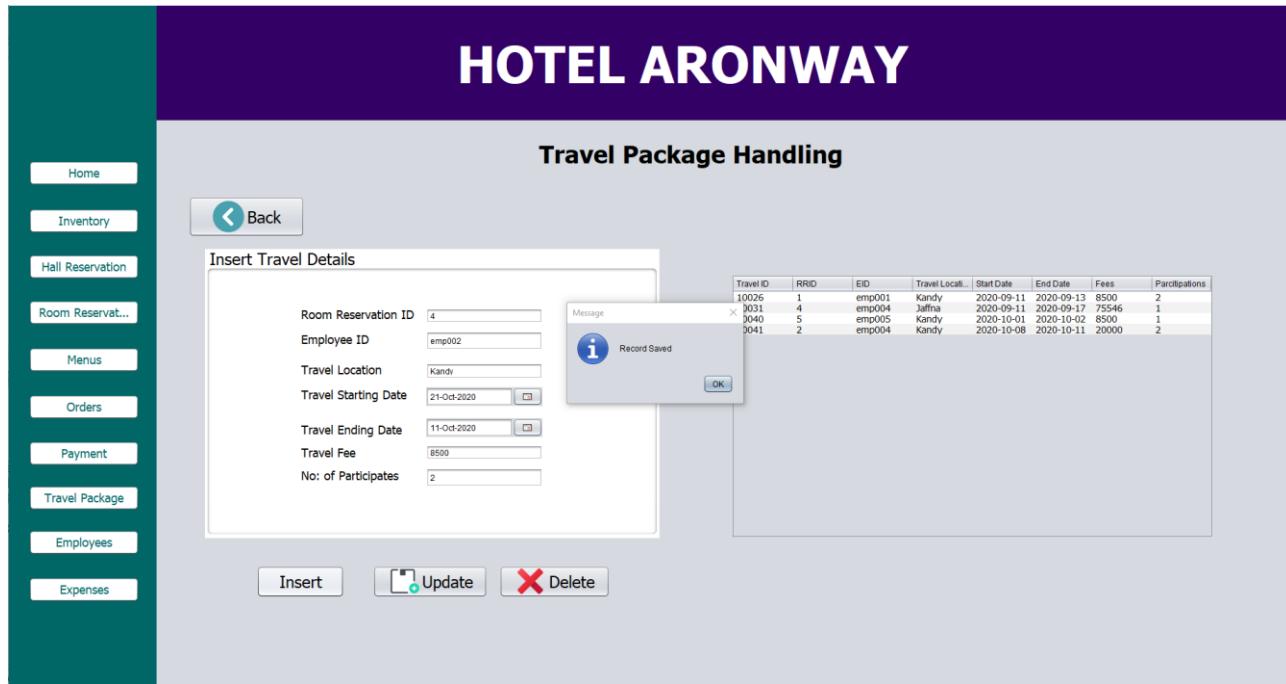


Figure 86: Add Travel Package Passed Test Case

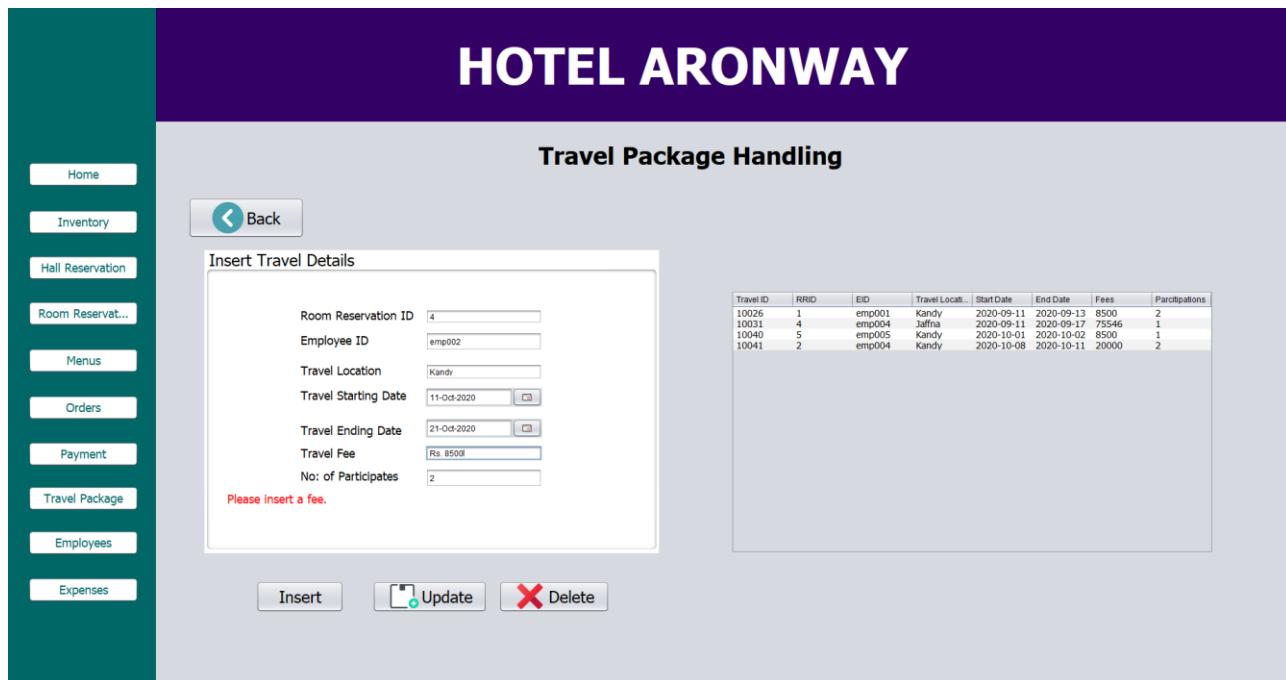


Figure 87: Add Travel Package Failed Test Case

Table 11: Test Cases for Travel Package Handling

Test ID	Test Inputs	Expected Outputs	Actual Outputs	Test Result	Description
Trav01	RRID: 4 EID: emp002 Location: Kandy Start Date: 11-Oct-2020 End Date: 21-Oct-2020 Fee: Rs. 8500 Participations: 2	Successful message that “Record saved.”	Show error message that “End date should be greater than start date”	Fail	If it is a valid insertion, the fee should be a double value. Then we can calculate the fee.
Trav02	RRID: 4 EID: emp002 Location: Kandy Start Date: 11-Oct-2020 End Date: 21-Oct-2020 Fee: 8500 Participations: 2	Show success message with new record.	Showed success message with new record.	Pass	If there is no error, records should be added to the database.

3. Conclusion

Conclusion has many different definitions. One may think the conclusion pushes beyond the boundaries of the prompt and allows you to consider broader issues, make new connections, and elaborate on the significance. After going through a starched requirement analysis phase, this system is developed to address main problems our client hotel ‘Aronway’ faced. Currently, this hotel staff handle all type of tasks by manually which make all these operations less diligent and also could lead to a customer dissatisfaction. This automated system gives solutions to all problems which were pointed out in the problem statement of this report. All those problems were written down below in a nutshell to for better clarification.

1. Time consuming
2. Higher cost and wastage
3. Record employees' details and attendances
4. Calculate bills manually
5. Keep track of inventories
6. Reservation overlap
7. Lack of security

Using information gathered in requirement analysis phase this entire project divided into nine different main functions which all together give solution to all problems which were mentioned above. To give a clear idea about current business progress and to support future growth of this business, each of these functions provide a meaningful report at the end of a particular month.

3.1 Weakness and Limitation

Even though the system is an automated system it is still being limited by some limitations,

- Technology is an ever-changing factor and there will always be better technologies, so with time the technology we used and methods we used will be outdated it is necessary to maintain and upgrade from time to time.
- Building a perfect errorless system is extremely hard, there will be unauthorized outside interference and it is important to keep the system up to date with security patches.

We always responsible and help our customer to maintain this system if an above limitation or weakness occurred. Also we are going to check the system every six months of time and do necessary updatings to keep the system in a secure and efficient way.

4. References

- [1] <https://www.wpblogx.com/what-is-xampp/>
- [2] <https://www.geeksforgeeks.org/how-to-install-xampp-on-windows/>

Appendix A: Design Diagrams

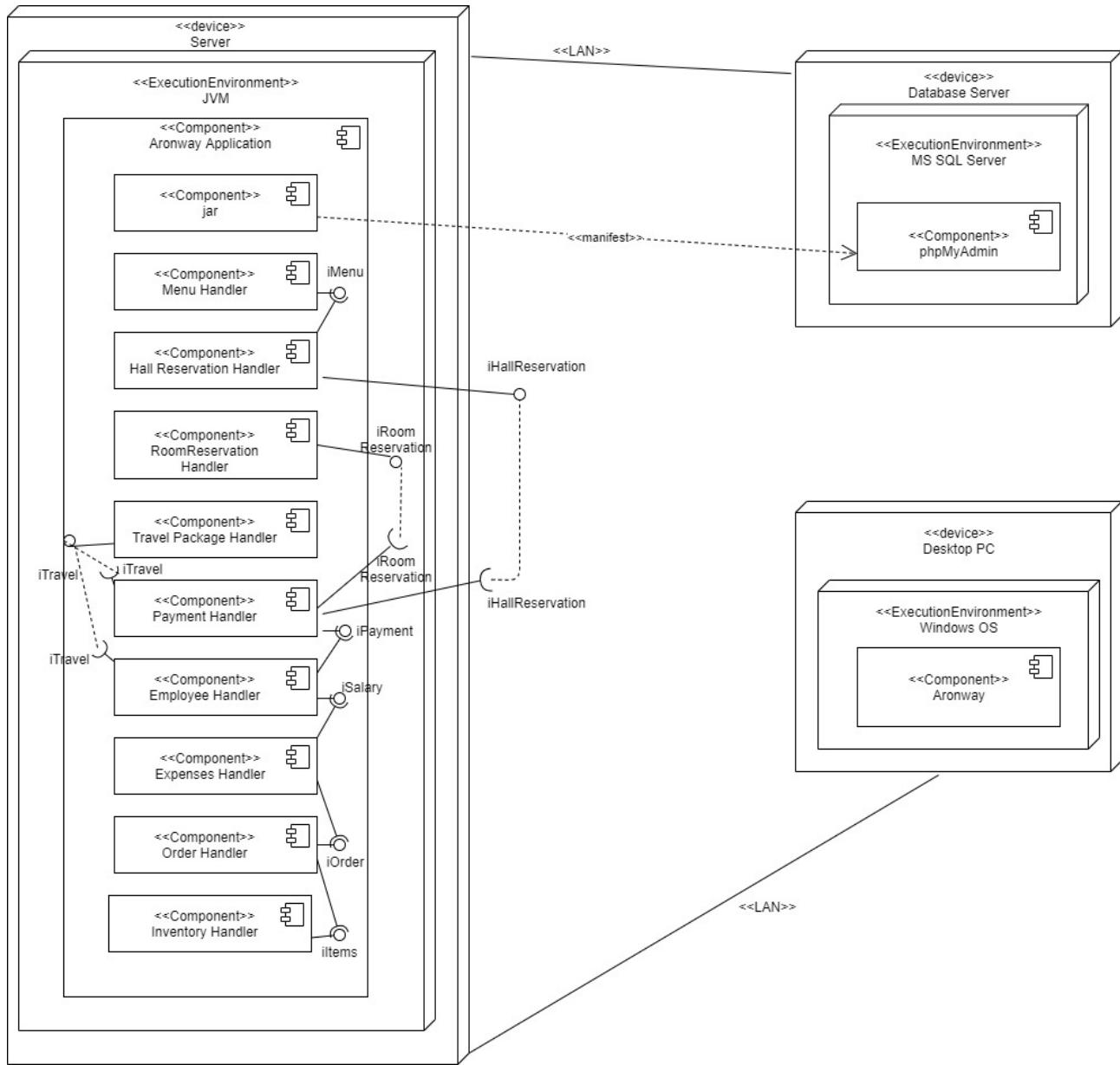


Figure 88: Physical Diagram

Appendix B: Test Results

Test ID	Test inputs	Expected output	Actual output	Result (Pass/Fail)	Description
T01	Name: Nimesha Rathnayake Date: Aug,19,2020 OT Time:2.30	Show “Inserted” message	Show “Inserted” message	Pass	If all the details are correct input and there are no empty fields, It will be added to the system
T02	Select row and update row	Show message “Menu Update Success”	Show message “Menu Update Success”	pass	selected details updated successfully
T03	Select row and Delete row	Show message “Deleted”	Show message “Deleted”	pass	selected row deleted successfully
T04	Select row and delete row	Show message “Expenses Delete Successfully”	Show message “Expenses Delete Successfully”	pass	selected correct row deleted successfull.

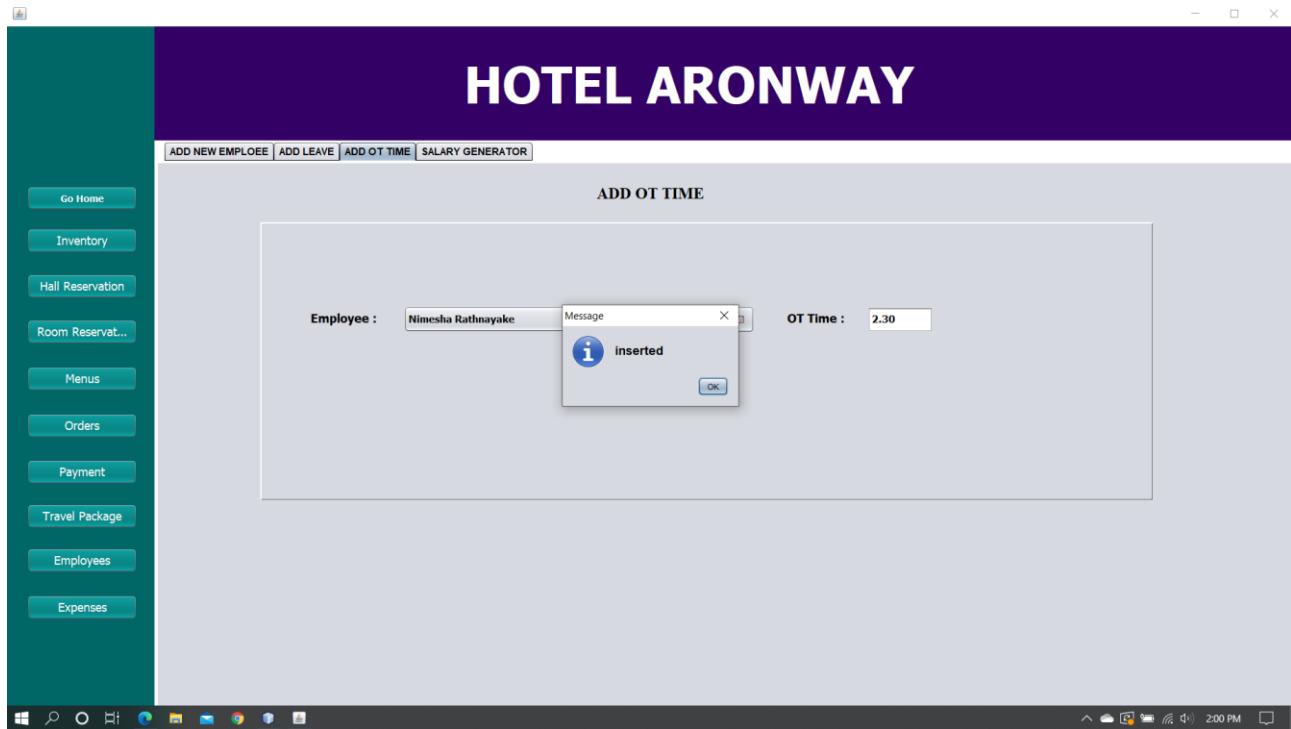


Figure 89: OT Time Adding Passed Test Case

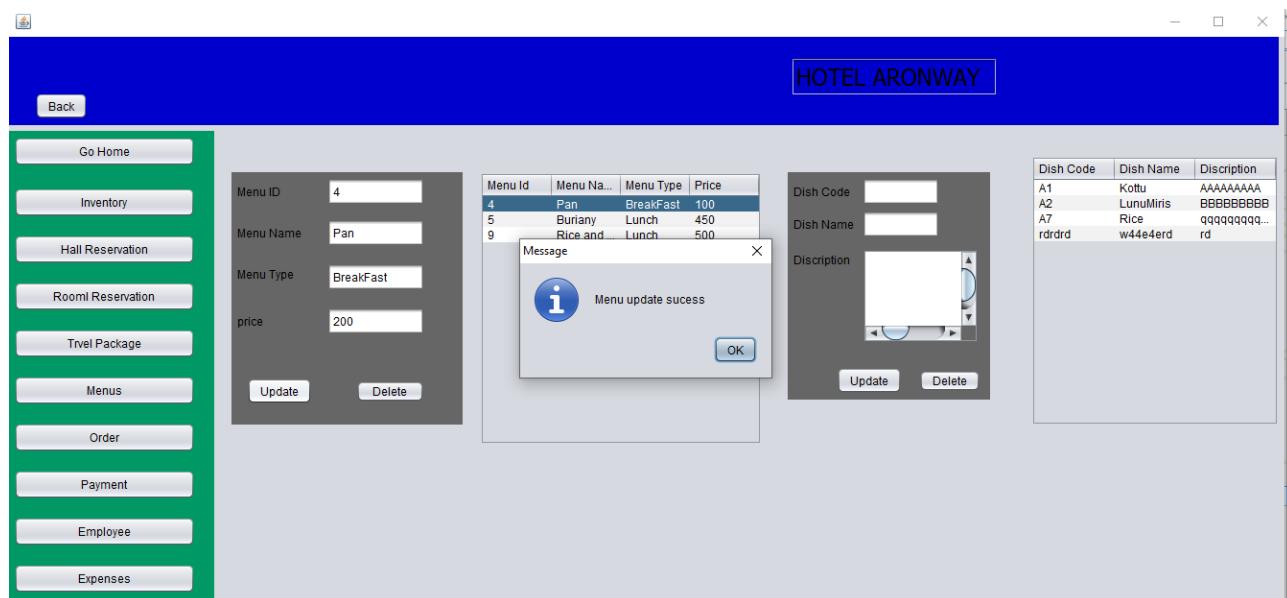


Figure 90: Menu Deleting Passed Test Case

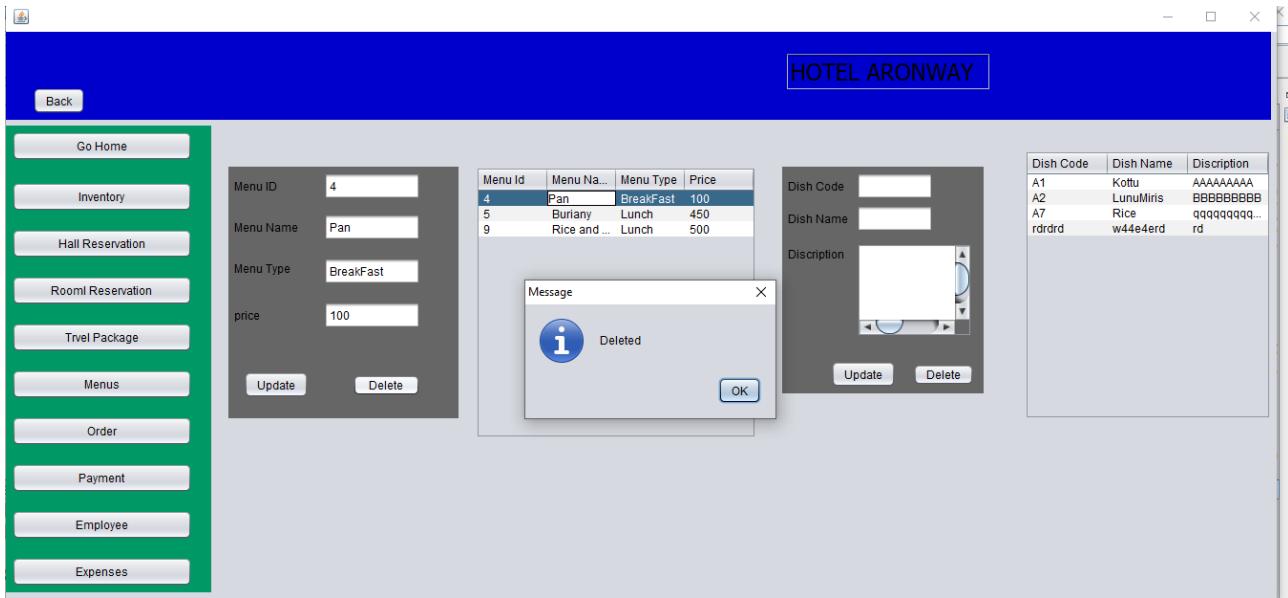


Figure 91: Menu Updating Passed Test Case

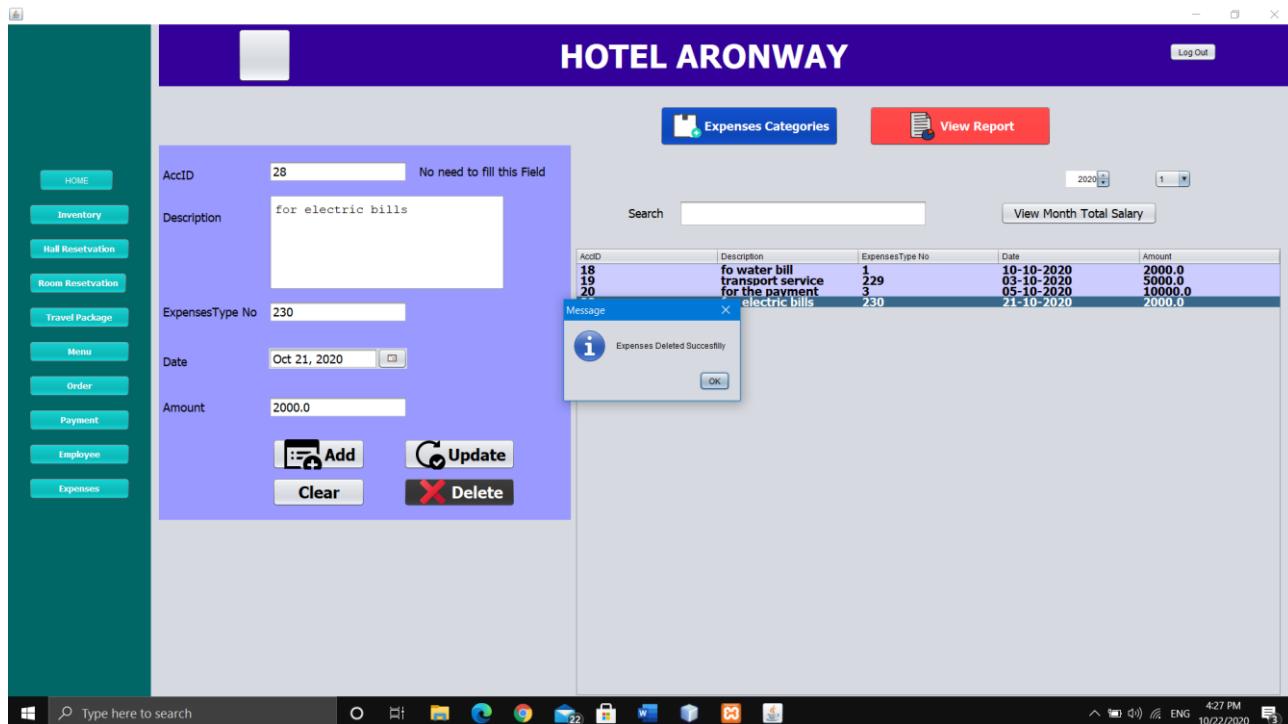


Figure 92:Expences Deleting Passed Test Case

Appendix C: Selected Code Listings

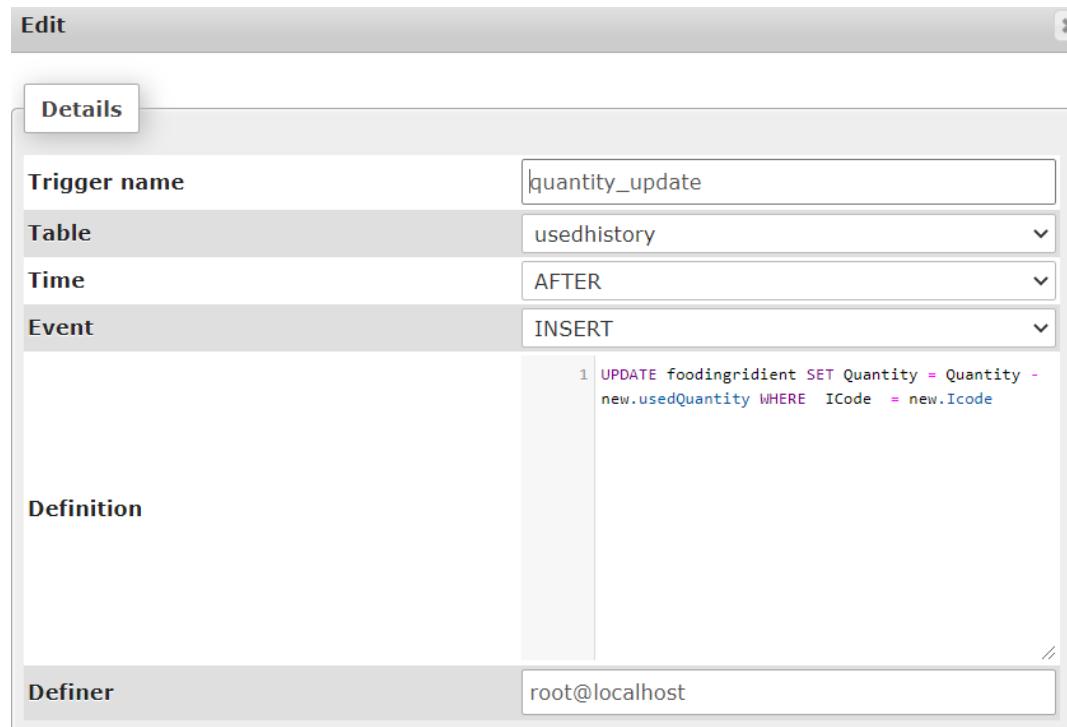


Figure 93: trigger of deduct used quantity from available quantity of an ingredient- Inventory Management



Figure 94:trigger of increase vailable quantity of a ingrident according to an order-Inventory Management

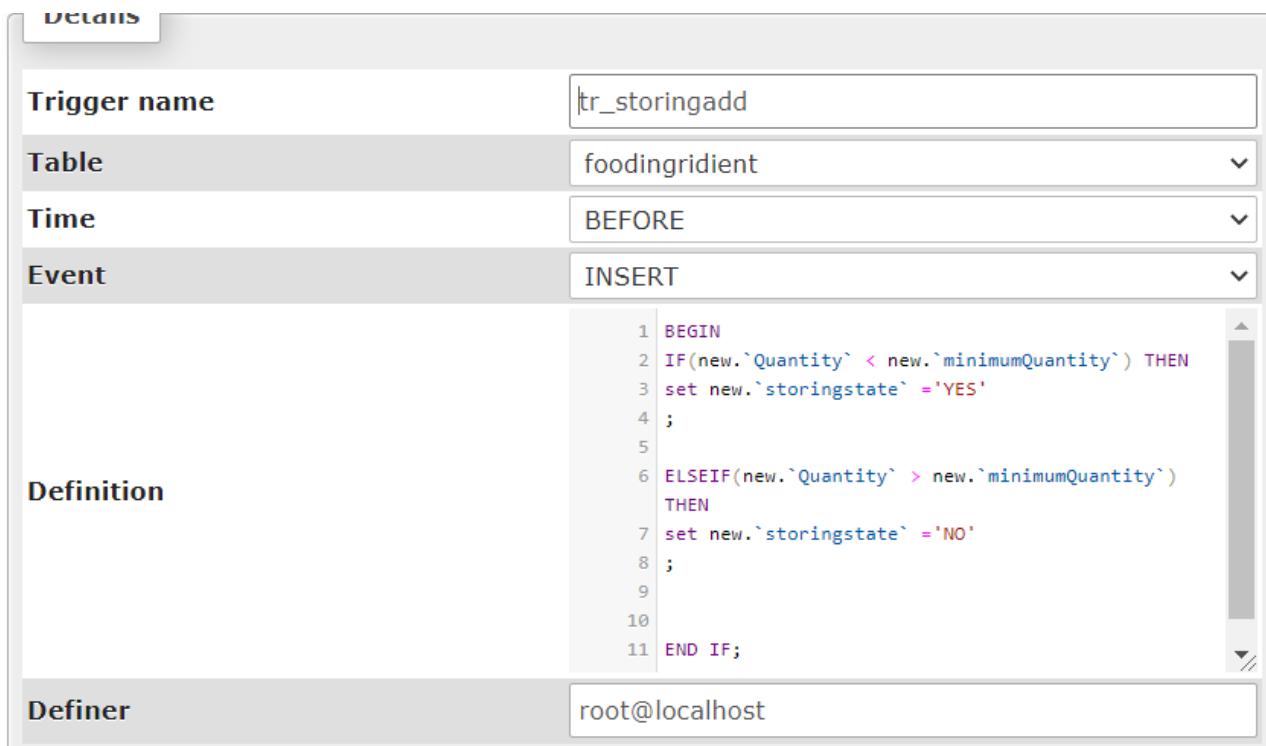


Figure 95: trigger of detecting storing state – Inventory Management

```

try {
    String sqlCus = "INSERT INTO customer(NIC, Name, Phone, Address, Email) VALUES('"+nic+"', '"+cName+"'";
    pst = conn.prepareStatement(sqlCus);
    pst.execute();

    pst = conn.prepareStatement("select * from menus where MenuName = '" +menuName+ "' ");
    rs = pst.executeQuery();

    while(rs.next()) {

        double mprice = rs.getDouble("Price");
        String menuID = rs.getString("MenuID");
        pst = conn.prepareStatement("select * from hall where Hall_name= '" +hname+ "'");
        rs = pst.executeQuery();
        rs.next();

        double hprice = rs.getDouble("Charge");
        String hallID = rs.getString("Hall_ID");
        double total = noParticipant * mprice +hprice;

        String sqltot = "INSERT INTO reservation (hall_ID,hall_name, menu_ID, menu_name, customer_ID, date"
                     + "           VALUES ('"+hallID+"','"+hname+"', '"+menuID+"', '"+menuName+"', '"+nic+"', '"+date+"')";
        pst = conn.prepareStatement(sqltot);
        pst.execute();
    }
}

```

Figure 96: Special Code - Hall Reservation

```

private void jToggleButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:

    String currentDirectorypath = "C:\\\\Users\\\\Public\\\\Pictures\\\\";
    JFileChooser file = new JFileChooser(currentDirectorypath);
    //    file.setCurrentDirectory(new File(System.getProperty("user.home")));
    //    file.setFileSelectionMode(JFileChooser.FILES_ONLY);

    FileNameExtensionFilter filter = new FileNameExtensionFilter("*image", "jpg", "png");
    file.addChoosableFileFilter(filter);

    int result = file.showSaveDialog(null);

    if (result == JFileChooser.APPROVE_OPTION) {

        File selectedFile = file.getSelectedFile();
        String path = selectedFile.getAbsolutePath();
        lbl_photo.setIcon(resetImageSize(path, null));
        this.photopath = path;
        //System.out.println(path);
        String imagefile = selectedFile.getName();
        ipath.setText(path);

    } else {

        System.out.println("no file selected");
    }
}

```

Figure 97:Special Code of Image Uploading - Employee Management

```

//<-- Employee ID auto generated-->
public void autoID() {

    try {

        Class.forName("com.mysql.jdbc.Driver");
        con = (Connection) DriverManager.getConnection("jdbc:mysql://localhost:3306/itp", "root", "");

        String sql = "select Max(id) from employees";
        Statement st = con.createStatement();
        ResultSet rs = st.executeQuery(sql);

        rs.next();
        rs.getString("Max(id)"); // get maxid in the database

        System.out.println(rs.getString("Max(id)"));
        if (rs.getString("Max(id)") == null) {

            txtid.setText("Emp001");
        } else {
            // Long ids= Long.parseLong(rs.getString("Max(id)").substring(3, rs.getString("Max(id)").length()));
            Long id = Long.parseLong(rs.getString("Max(id)").substring(3, rs.getString("Max(id)").length())); // id++;
            id++;

            txtid.setText("Emp" + String.format("%03d", id)); //Set next Employee id to the jfield
        }
    } catch (Exception e) {

        JOptionPane.showMessageDialog(null, e);
    }
}

```

Figure 98: Special Code for Generate auto ID -Employee Management

```

if (mnth > 0 & mnth < 10) {
    insert = con.prepareStatement("select sum(ottime) from ot where otdate like '" + yr + "-0" + mnth + "%' and id like '" + empid + "'");
    ResultSet rs2 = insert.executeQuery();
    rs2.next();

    double count2 = rs2.getDouble("sum(ottime)");
    // System.out.println(count2);

    String doubleAsString = String.valueOf(count2);
    int indexOfDecimal1 = doubleAsString.indexOf(".");
    double diff = Double.parseDouble(doubleAsString.substring(indexOfDecimal1));

    int time = (int) count2;

    // double diff = count2 - time;

    if (diff > 0.60) {

        double remain = diff - 0.60;
        time = time + 1;
        count2 = time + remain;
    } else {
        count2 = count2;
    }

    othour.setText(Double.toString(count2));
}

```

Figure 99: Special Code for Calculate Total OT Time -Salary Management

```

        double remain;
        int time = (int) totOT;
        System.out.println(time);
        double diff = (totOT - time) * 100;
        remain = diff / 60;

        System.out.println(diff);
        // System.out.println(remain);

        calot=(orate * time) + (orate * remain);
        calot= Math.round(calot*100.0)/100.0;

        if (totleave > 21) {
            if (leaverate.getText().toString().equals("")) {
                JOptionPane.showMessageDialog(null, "Enter the Leave Rate ");
            } else {
                double lrate = Double.parseDouble(leaverate.getText());
                double netsal = (basicosal + bonusl + (orate * time) + (orate * remain)) - (epfal + etfal + ((totleave - 21) * lrate));

                double r = netsal;
                r=Math.round(r*100.0)/100.0;
                total.setText(Double.toString(r));
                leavededuct=((totleave - 21) * lrate);

            }
        } else {

            double netsal = (basicosal + bonusl + (orate * time) + (orate * remain)) - (epfal + etfal);

            double r = netsal;
            r=Math.round(r*100.0)/100.0;
            total.setText(Double.toString(r));
            leavededuct=0;
        }
    }
}

```

Figure 100: Special Code for Calculate Net Salary -Salary Management

The screenshot shows a Java IDE interface with three tabs at the top: payment.java, Payment2.java, and paymentDelete.java. The payment.java tab is active, displaying the following code:

```
1433     String id= "0";
1434     txt_hallId.setText(id);
1435     txt_hallId.setEditable(false);

1436     int rod = Integer.parseInt(txt_roomId.getText());
1437     Connection con = MySqlConnection();

1438     try {
1439         String qry = "select sum(amount) from trav_pack_handle where RRID like '"+rod+"'";
1440         Statement st = con.createStatement();
1441         ResultSet rs =st.executeQuery(qry);
1442         rs.next();
1443         amount = 0;
1444         amount = Float.parseFloat(rs.getString("sum(amount)"));
1445     } catch (Exception e) {
1446         amount = 0;
1447     }
1448     try {
1449         String qry2 = "select Amount from roomreservation where RRID like '"+rod+"'";
1450         Statement st2 = con.createStatement();
1451         ResultSet rs2 =st2.executeQuery(qry2);
1452         rs2.next();
1453         amount2 = Float.parseFloat(rs2.getString("Amount"));
1454     } catch (Exception e) {
1455         amount2 = 0;
1456     }
1457     try {
1458         String qry3 = "select sum(Price) from roommenu rm,menus m, roomreservation r where rm.RRID = r.RRID and rm.MenuID =m.MenuID and r.RRID like '"+rod+"'";
1459         Statement st3 = con.createStatement();
1460         ResultSet rs3 =st3.executeQuery(qry3);
1461         rs3.next();
1462         amount3 = 0;
1463         amount3 = Float.parseFloat(rs3.getString("sum(Price)"));
1464     } catch (Exception e) {
1465         amount3 =0 ;
1466     }

1467     float finalA = amount + amount2 + amount3;
1468
1469     txt_final.setText(Float.toString(finalA));
1470
1471
1472 }
```

Figure 101: Special Code for Payment Calculation: Payment Management