

# Project Report



AS2022432 – I.P.Wickramaarachchi

## 1. Introduction

The DINEDECK Restaurant Table Reservation System is a user-friendly application designed to streamline table reservations for customers and optimize operations for restaurant staff. With flexible reservation options and intuitive interfaces, it offers a seamless dining experience while ensuring efficient management of tables, customers, and reservations. This project report provides an overview of the system and also covers the functional and non-functional requirements of the system.

## 2. Objectives

- Incorporate the MVC architecture to ensure the distinct separation of responsibilities and simplify the maintenance process.
- Provide user-friendly interface for customers to reserve tables based on specific requirements.
- Address functional requirements by incorporating features to manage table details, customer details, and booking details effectively.
- Address non-functional requirements by ensuring system performance, security and usability.

## 3. System Architecture

The DINEDECK Restaurant Table Reservation System follows the MVC architecture.

- **Model** : The model Represents the data and business logic of the application. It's like the brain of the system, storing and managing information. In this system, the 'Customer', 'Tables' and 'Reservations' hold data about restaurant tables and customer bookings.
- **View** : The view is responsible for rendering the user interface and displaying the data to the user. The view component does not contain any business logic but instead relies on the model for data. In this system, providing interfaces for make reservations and manage table details and reservation details.
- **Controller** : The controller act as an intermediary between the model and the view. It handles user input, processes user actions, and updates the model or view accordingly.

#### 4. Assumptions

- The system has two privilege levels namely customer and administrator.
- Customer can make reservations for single date, for continuous time period and for specific days.
- Administrator can add, update, search and remove the customers, tables and reservations.

#### 5. Functional Requirements

- Customers should be able to register with the system by providing details.
- Customers should be able to log in using their email and password.
- Customers should be able to reserve tables for single date, for continuous time period and for specific days.
- Customers should be able to cancel the reservations.
- Administrator should have special privileges for managing the system, including user management and access to administrative functionalities.

#### 6. Non-Functional Requirements

- **Security** : User authentication should be secure to prevent unauthorized access to customer accounts and administrative functionalities.
- **Performance** : The system should handle a large number of concurrent users without significant degradation in performance.
- **Reliability** : Data integrity should be maintained to prevent loss or corruption of customer information, table reservation and administrative data.
- **Usability** : The user interface should be intuitive and easy to navigate, facilitating seamless interactions for customers and administrators.
- **Scalability** : The system should be scalable to accommodate growth in the number of users, tables and reservations over time.
- **Maintainability** : The system should be modular and well-documented, facilitating ease of maintenance and future enhancements.

## 7. System Implementation

- **Requirement Analysis** : Review the functional and non-functional requirements, gather stakeholder input and create detailed document outlining system functionality, user roles and technical specifications.
- **Design** : Design the system architecture, define data models, database schema and user interface based on MVC architecture.
- **Development** : Develop the system components, implement authentication, reservation functionalities and apply security measures.
- **Testing** : Integrate developed components, conduct integration and system testing to ensure functionality and performance meet requirements.
- **Deployment** : Prepare, set up infrastructure, deploy, configure conduct final testing in production environment
- **Maintenance** : Monitor performance, provide support, update with new features and conduct periodic reviews for system maintenance and improvement.

## 8. Test Cases

<b>Test Case ID</b>	: 01
<b>Test Case Scenario</b>	: Make sure signup is working
<b>Test Case</b>	: Enter user details with valid email and valid password
<b>Pre-Conditions</b>	: Need valid email and password
<b>Test Steps</b>	: <ul style="list-style-type: none"><li>i) Enter user details</li><li>ii) Enter valid email</li><li>iii) Enter valid password</li><li>iv) Confirm password</li><li>v) Sign up</li></ul>
<b>Test Data</b>	: <valid email> <valid password>
<b>Expected Result</b>	: Successful signup
<b>Post-Condition</b>	: Login page is shown
<b>Actual Result</b>	: Login page is shown
<b>Status</b>	: Pass

**Test Case ID** : 02  
**Test Case Scenario** : Make sure login is working  
**Test Case** : Enter email and password  
**Pre-Conditions** : Need valid email and password  
**Test Steps** :  
    i) Select user type  
    ii) Enter valid email  
    iii) Enter valid password  
    iv) Log in  
        a. Customer log in  
        b. Administrator log in  
  
**Test Data** : <user type> <valid email> <valid password>  
**Expected Result** : Successful login  
  
**Post-Condition** : a. Customer main menu page is shown  
                    b. Administrator main menu page is shown  
  
**Actual Result** : a. Customer main menu page is shown  
                    b. Administrator main menu page is shown  
**Status** : Pass

**Test Case ID** : 03  
**Test Case Scenario** : Make sure customer details update is working  
**Test Case** : Enter new user details with valid email and valid password  
**Pre-Conditions** : Need login to the system  
**Test Steps** :  
    i) Enter new user details  
    ii) Update user details  
  
**Test Data** : <valid email> <valid password>  
**Expected Result** : Successfully update user account with new details  
**Post-Condition** : Updated user details  
**Actual Result** : Updated user details  
**Status** : Pass

**Test Case ID** : 04  
**Test Case Scenario** : Make sure reservation is working  
**Test Case** : Enter reservation details  
**Pre-Conditions** : Chose a reservation type  
**Test Steps** : i) Chose a reservation type  
ii) Enter valid reservation details  
iii) Chose a table from available table list  
iv) Reserve a table  
  
**Test Data** : <reservation type> <start date> <start time> <end time>  
<end date>  
  
**Expected Result** : Successful reservation  
**Post-Condition** : Ask to download reservation slip  
**Actual Result** : Reservation successful and ask to download reservation slip  
**Status** : Pass

**Test Case ID** : 05  
**Test Case Scenario** : Make sure cancel reservation is working  
**Test Case** : Select upcoming reservation from table  
**Pre-Conditions** : Go to the my reservation details page  
**Test Steps** : i) Select a reservation from upcoming reservation table  
ii) Cancel the reservation  
  
**Test Data** : <selected row>  
  
**Expected Result** : Successfully cancel reservation  
**Post-Condition** : Refresh the upcoming booking table  
**Actual Result** : Cancel the reservation and refresh table is shown  
**Status** : Pass

**Test Case ID** : 06  
**Test Case Scenario** : Make sure customer deletion is working  
**Test Case** : Select a customer from table  
**Pre-Conditions** : Need to login as administrator and go to customer details page  
**Test Steps** : i) Log in as administrator  
ii) Select a customer from customer table  
iii) Delete the customer  
  
**Test Data** : <selected row>  
**Expected Result** : Successfully delete a customer  
**Post-Condition** : Refresh the customer details table  
**Actual Result** : Delete the customer and refresh the table  
**Status** : Pass

**Test Case ID** : 07  
**Test Case Scenario** : Make sure add a new table is working  
**Test Case** : Add new table details  
**Pre-Conditions** : Need to login as administrator and go to table details page  
**Test Steps** : i) Log in as administrator  
ii) Enter new table details  
iii) Add the table  
  
**Test Data** : <table id>  
**Expected Result** : Successfully add a new table  
**Post-Condition** : Refresh the table details table  
**Actual Result** : Add a new table and refresh the table details table  
**Status** : Pass

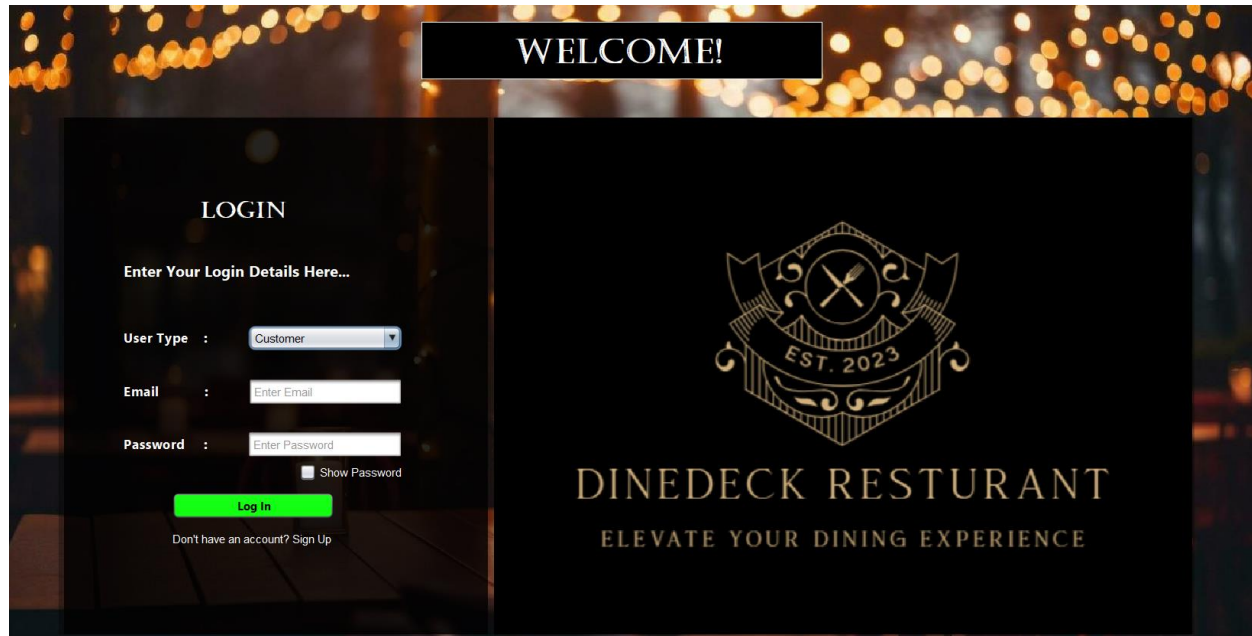
**Test Case ID** : 08  
**Test Case Scenario** : Make sure update table is working  
**Test Case** : Update table details  
**Pre-Conditions** : Search a table using table id  
**Test Steps** : i) Log in as administrator  
ii) Search a table using table id  
iii) Update table details  
  
**Test Data** : <table id>  
**Expected Result** : Successfully update a table  
**Post-Condition** : Refresh the table details table  
**Actual Result** : Update a table and refresh the table details table  
**Status** : Pass

**Test Case ID** : 09  
**Test Case Scenario** : Make sure table deletion is working  
**Test Case** : Delete a table  
**Pre-Conditions** : Search a table using table id  
**Test Steps** : i) Log in as administrator  
ii) Search a table using table id  
iii) Delete table details  
  
**Test Data** : <table id>  
**Expected Result** : Successfully delete a table  
**Post-Condition** : Refresh the table details table  
**Actual Result** : Delete a table and refresh the table details table  
**Status** : Pass



## 9. System GUI

- Login form



**For User Logins,**

**For customer login :**

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

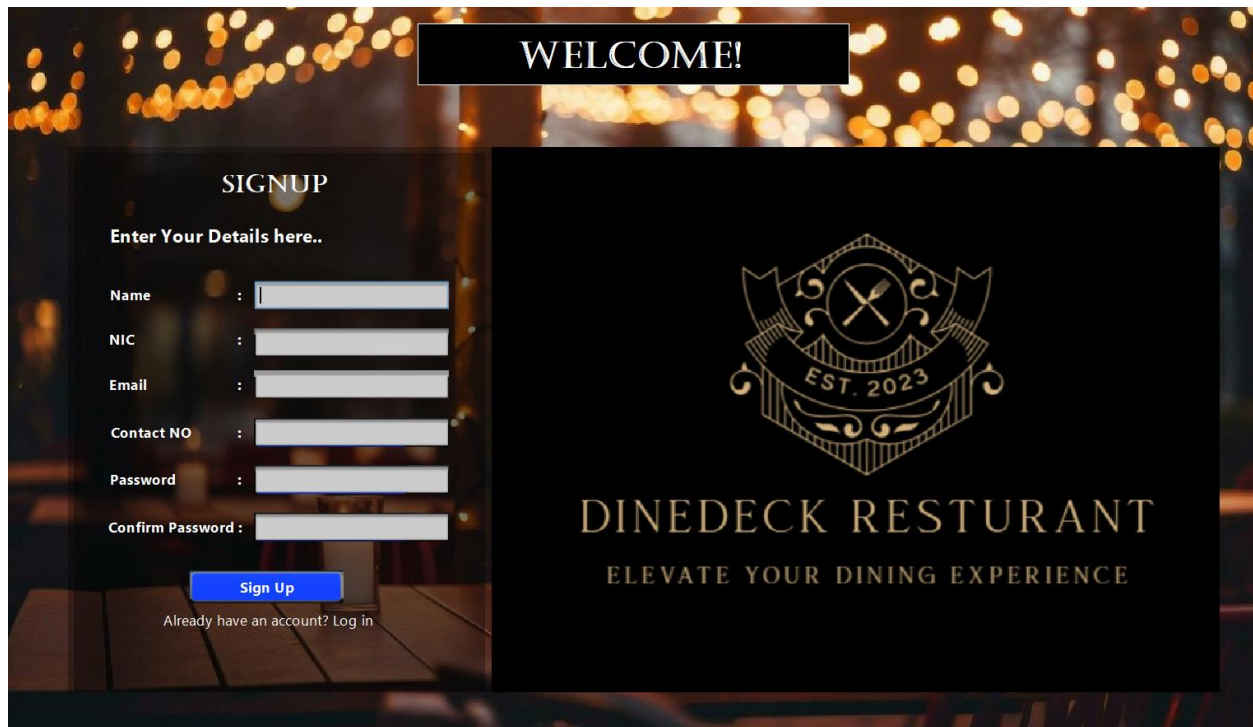
Password : Isira@123

**For administrator login :**

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

Password : dinedeckAdmin@123

- Signup form



The image shows a web page for DINEDECK RESTURANT. At the top, a black banner with the word "WELCOME!" in white capital letters is centered. Below this, the page is divided into two main sections. The left section is a dark grey box with the title "SIGNUP" in white. Below the title, it says "Enter Your Details here..". There are six input fields, each with a label and a colon: "Name", "NIC", "Email", "Contact NO", "Password", and "Confirm Password". Each field has a white input box. Below the input fields is a blue button with the text "Sign Up" in white. Below the button, it says "Already have an account? Log in". The right section is a dark grey box with a logo at the top. The logo is a shield-shaped emblem with a fork and knife crossed inside, and the text "EST. 2023" below it. Below the logo, the text "DINEDECK RESTURANT" is written in a large, serif font, and "ELEVATE YOUR DINING EXPERIENCE" is written in a smaller, sans-serif font below it.

WELCOME!

**SIGNUP**

Enter Your Details here..

Name :

NIC :

Email :

Contact NO :

Password :

Confirm Password :

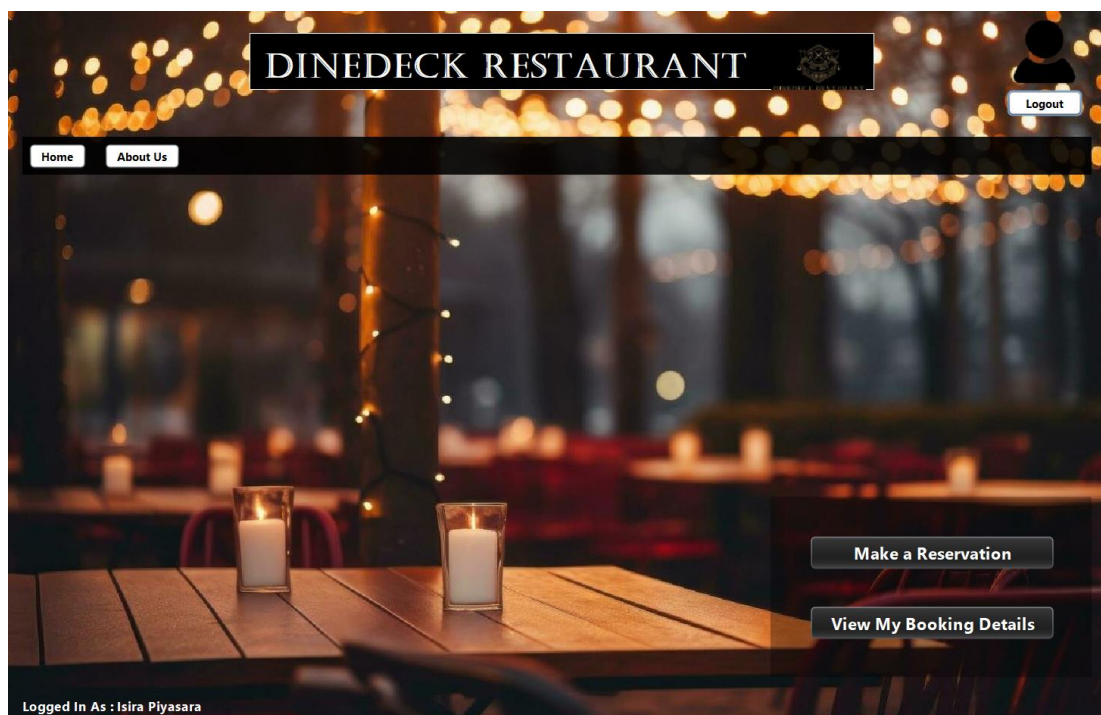
[Sign Up](#)

[Already have an account? Log in](#)

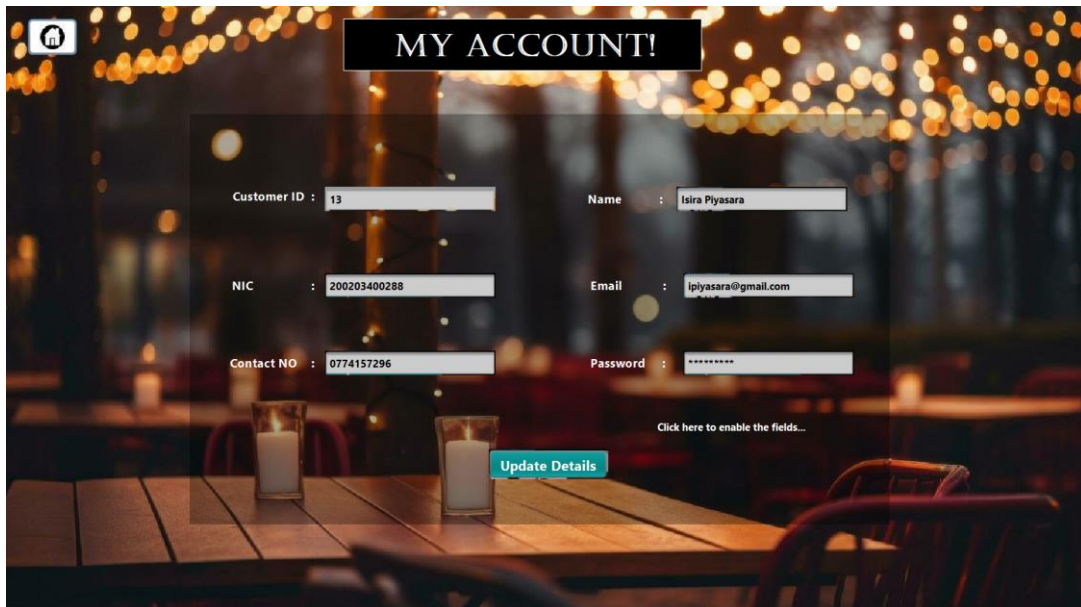
**DINEDECK RESTURANT**

ELEVATE YOUR DINING EXPERIENCE

- Customer main menu



- Customer account



**MY ACCOUNT!**

Customer ID : 13      Name : Isira Piyasara

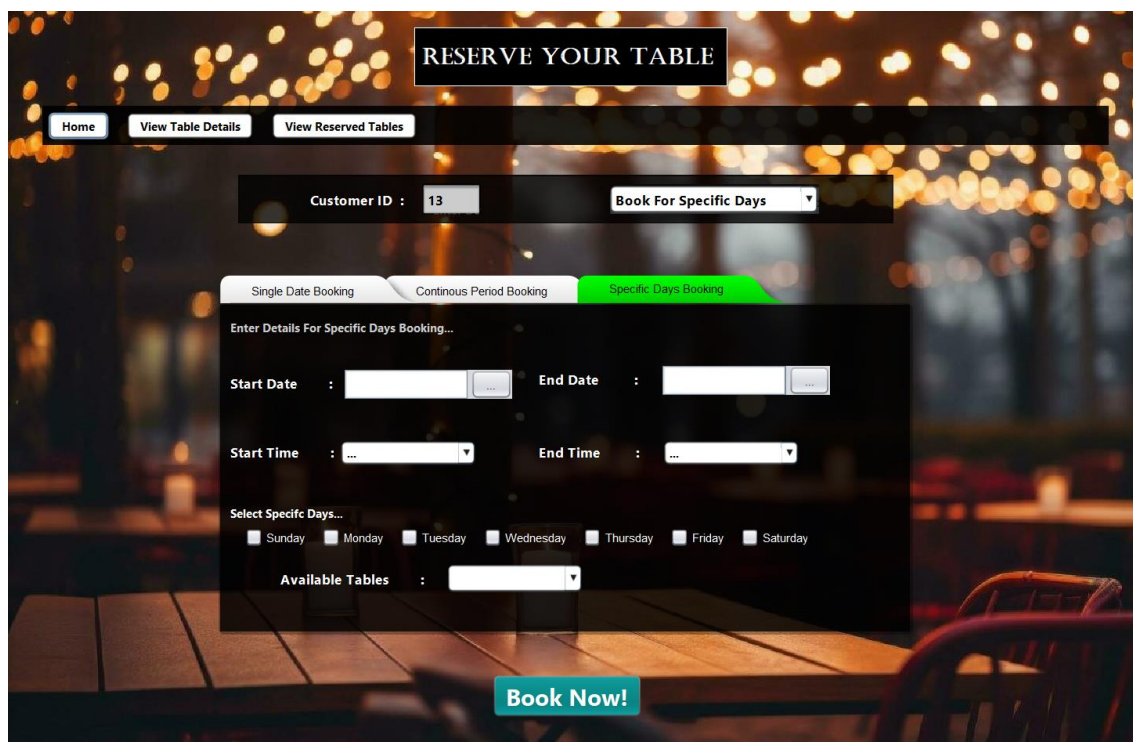
NIC : 200203400288      Email : lpiyasara@gmail.com

Contact NO : 0774157296      Password : \*\*\*\*\*

[Click here to enable the fields...](#)

[Update Details](#)

- Customer reservation form



**RESERVE YOUR TABLE**

[Home](#)   [View Table Details](#)   [View Reserved Tables](#)

Customer ID : 13      [Book For Specific Days](#)

Single Date Booking   Continuous Period Booking   **Specific Days Booking**

Enter Details For Specific Days Booking...

Start Date :       End Date :

Start Time : ...      End Time : ...

Select Specific Days...


☐ Sunday   ☐ Monday   ☐ Tuesday   ☐ Wednesday   ☐ Thursday   ☐ Friday   ☐ Saturday

Available Tables :

[Book Now!](#)



- Customer reservation details



## MY RESERVATIONS

Click table rows for get values...

Reservation ID : 
 Start Date :

Customer ID : 
 Start Time :

Table ID : 
 End Time :

Reservation Type : 
 End Date :

Payable Amount for table :

[Download Reservation Slip](#)

### Upoming Reservation...

Reservation...	TableID	Reservation...	StartDate	StartTime	EndTime	EndDate
29	T01	Book For C...	2024-05-23	19:00	20:00	2024-05-26
37	T04	Book For Sl...	2024-06-03	18:00	18:00	2024-06-03


[Cancel Reservation!](#)

### Old Reservations...

Reservation...	TableID	Reservation...	StartDate	StartTime	EndTime	EndDate
4	T02	Book For Sl...	2024-05-03	18:00	19:00	2024-05-03
10	T01	Book For Sl...	2024-05-04	19:00	20:00	2024-05-04
11	T01	Book For C...	2024-05-05	17:00	18:00	2024-05-05
12	T02	Book For C...	2024-05-05	17:00	18:00	2024-05-05
22	T02	Book For S...	2024-05-14	16:00	17:00	2024-05-14
23	T02	Book For S...	2024-05-16	16:00	17:00	2024-05-16
24	T02	Book For S...	2024-05-18	16:00	17:00	2024-05-18

- Administrator main menu

# DINEDECK RESTAURANT



[Logout](#)


[Customer List Report](#)
[Daily Reservations Report](#)
[Table List Report](#)

[View Customer Details](#)

[View Reservation Details](#)

[View Table Details](#)

- Customer details



CUSTOMER DETAILS


Search for Customer :

cusID	Name	NIC	Email	ContactNO	Password
3	Sendini Siyathma	200206877965	sendi@gmail.com	0774987456	Sendi@987
8	Sithuni Gamman...	2002345698741	sithu@gmail.com	0769847854	Sithu@456
13	Isira Piyasara	200203400288	ipiyasara@gmail.c...	0774157296	Isira@123

Delete

Add Reservation

- Table details



VIEW TABLE DETAILS

Enter Details Here...

Table ID :

No Of Seats :

Charges Per Hour :

Add Table

Search for Table :

Table ID :

No Of Seats :

Charges Per Hour :

Update

Delete

Tables in the System...

tableID	NoOfSeats	ChargesPerHour
T01	2	1000.0
T02	4	2000.0
T03	6	3000.0
T04	8	4000.0
T05	7	3500.0

- Reservation details



MY RESERVATIONS

Click table rows for get values...

Reservation ID :

Start Date :

Customer ID :

Start Time :

Table ID :

End Time :

Reservation Type :

End Date :

Payable Amount for table :

Download Reservation Slip

Upoming Reservation...

Reservati...	Customer...	TableID	Reservati...	StartDate	StartTime	EndTime	EndDate	Reservati...
27	1	T01	Book For ...	2024-05-22	10:00	12:00	2024-05-22	2024-05-12
28	8	T02	Book For ...	2024-05-22	18:00	19:00	2024-05-22	2024-05-12
29	3	T01	Book For ...	2024-05-23	19:00	20:00	2024-05-26	2024-05-12
30	1	T01	Book For ...	2024-05-28	17:00	18:00	2024-05-28	2024-05-12
31	1	T01	Book For ...	2024-05-30	17:00	18:00	2024-05-30	2024-05-12
33	1	T03	Book For ...	2024-05-26	11:00	12:00	2024-05-29	2024-05-14
37	3	T04	Book For ...	2024-06-03	16:00	18:00	2024-06-03	2024-05-15
38	1	T01	Book For ...	2024-06-03	16:00	18:00	2024-06-03	2024-05-15
39	8	T01	Book For ...	2024-06-03	15:00	16:00	2024-06-03	2024-05-15
40	8	T01	Book For ...	2024-06-03	15:00	16:00	2024-06-03	2024-05-15

Cancel Reservation

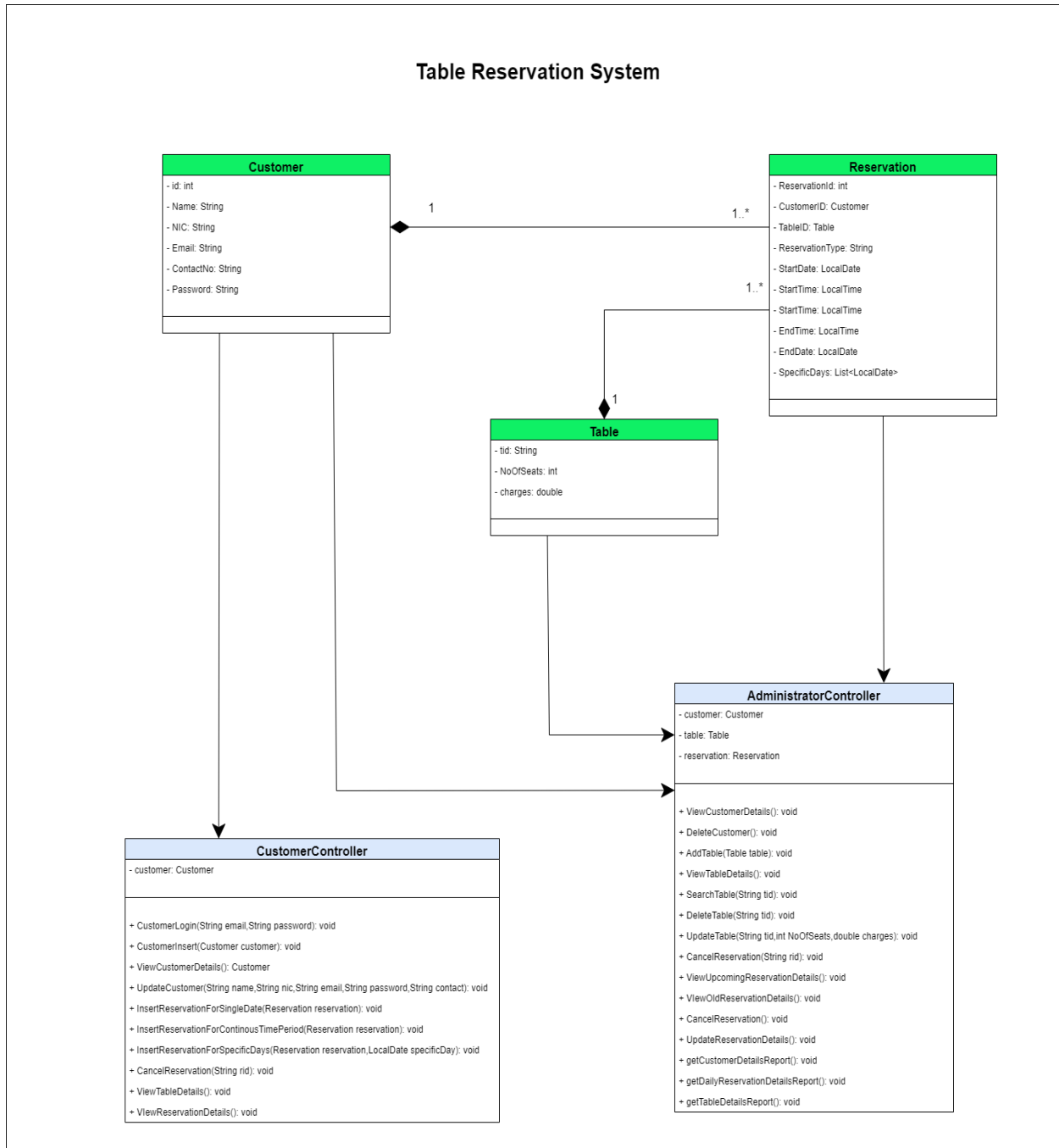
Old Reservations...

Reservati...	Custom...	TableID	Reservati...	StartDate	StartTime	EndTime	EndDate	Reservati...
2	1	T01	Book For ...	2024-05-01	16:30	17:30	2024-05-01	
3	1	T01	Book For ...	2024-05-01	16:30	17:30	2024-05-01	
4	3	T02	Book For ...	2024-05-03	18:00	19:00	2024-05-03	
5	1	T02	Book For ...	2024-05-03	18:00	19:00	2024-05-03	
6	1	T02	Book For ...	2024-05-03	16:00	17:00	2024-05-03	
7	1	T02	Book For ...	2024-05-03	17:00	18:00	2024-05-03	
8	1	T02	Book For ...	2024-05-04	17:00	18:00	2024-05-04	
9	1	T02	Book For ...	2024-05-04	18:00	19:00	2024-05-04	
10	3	T01	Book For ...	2024-05-04	19:00	20:00	2024-05-04	

## 10. Use Case Diagram



## 11. Class Diagram





## 12.ER Diagram

