SALON RESERVATION SYSTEM

GROUP NAME: ReserveIT

TEB1043 – Object-Oriented Programming

May. 2024

**Submitted by**

| 24000182 | NURUL HANIIZATI BINTI HAZLI |
| --- | --- |
| 24000574 | SARA EUDORA BINTI SAID |
| 22006373 | DANIA ADRIANA BINTI MOHD FAIZAL |
| 24000918 | ZULAIKHA BINTI MOHD AZHAR |
| 22005979 | RAILYNN NISHALIA BINTI RHYMEI |
|  |  |

**Submitted to**

Dr M Nordin B Zakaria

CIS Department

Universiti Teknologi PETRONAS, Malaysia

**CONTENTS**

| **No.** |  | **Page** |
| --- | --- | --- |
| 1. | PROJECT TITLE | 3 |
| 2. | PROJECT OBJECTIVES | 3 |
| 3. | MARKET POTENTIAL | 4 |
| 4. | PROJECT DETAILS | 5 |
| 5. | PROJECT CODE | 6 |
| 6. | USER MANUAL | 7 |
| 7. | PROJECT UML DIAGRAM | 14 |

**PROJECT TITLE**

Salon Reservation System

**PROJECT OBJECTIVES**

1. **Develop a User-Friendly and Efficient Reservation System**

Design a user-friendly client application for customers to easily find and select an available stylist based on their preferences, choose a convenient time slot that fits their schedule, browse and select from a variety of services including haircuts, hair coloring, hair styling and facials, enabling customers to conveniently manage their reservations. This application should provide a seamless and intuitive user experience to enhance customer satisfaction and streamline the reservation process.

1. **Leverage Advanced Java Programming Techniques and Best Practices**

Implement a modular and maintainable codebase by applying fundamental object-oriented programming principles. Moreover, it is to enhance user interaction and experience by creating a visually appealing and intuitive graphical user interface (GUI).

1. **Provide Valuable Business Insights**

Through comprehensive reporting and analytics, the system offers salon owners and managers insights into client preferences and business performance. These data-driven insights help in making informed decisions that can drive business growth and success.

**MARKET POTENTIAL**

**Market Demand**

An efficient and convenient system that will help salons improve appointment management and reduce manual customer entries. Furthermore, it will increase gadgets penetration with online booking trends to minimize customer’s time spent in a salon while enhancing customer amenities.

**Target Market**

Our system focuses on hair and beauty salons where it is a business that heavily relies on appointments and client schedule management. The need for a reservation system is to predict customers traffic throughout the day, allowing for better staff scheduling, inventory management, and increased revenue.

The Salon Reservation System is designed to cater to a diverse range of beauty and wellness businesses, including independent salons, salon chains, and franchises. It serves small to medium-sized salons seeking to streamline their booking process, as well as large multi-location chains requiring centralized management. The system is ideal for mobile and home-based stylists who need a flexible scheduling solution, as well as full-service spas and day spas looking to manage a wide array of services efficiently. Additionally, it supports beauty schools and training centers in managing student appointments, hotel and corporate office salons in coordinating bookings with existing systems, and event and wedding planners in organizing beauty services for special occasions. By addressing the specific needs of these various segments, the Salon Reservation System enhances operational efficiency, client satisfaction, and overall business performance.

**PROJECT DETAILS**

Salon Reservation System will operate in the following procedure:

**Input***:* Customers can input their reservation details such as the date, time and preferred stylist. Customer details can also be inserted such as their name, contact, email and type of services. Additionally, employees have the ability to update their availability.

**Process***:* The program has a user-friendly and minimalistic design to ensure straightforward navigation and usability for users for all technical skill levels. Customer details and submitted information will be meticulously processed and stored in the database upon submission. The system supports adding, removing, searching, and printing customer details. Staff can manage and supervise reservations, to meet customers' needs and regulate stylist availability as required.

**Output***:* A list of client information, reservations and stylist information will be displayed. A confirmation message will also be displayed on the window saying that the data has either been successfully added or removed. Additionally, an option to cancel the reservation is provided.

**Database**: Stores persistent data related to clients, stylists, and reservations. The MySQL database is accessible through **phpMyAdmin** for easy management and administration.

**Running The System:** To run the Salon Management System, the BlueJ IDE is used.

Follow these steps to set up and run the project:

1. Open BlueJ: Launch the **BlueJ IDE** on your computer.
2. Open the Project: Go to **Project -> Open Project** and select the directory containing the Salon Management System source code.
3. Compile the Project: Click on **Compile** to compile all classes in the project.
4. Run the FrontEndClient: Right-click on the **FrontEndClient** class and select **Run JavaFX Application** to run the application. This will launch the user interface, allowing you to interact with the system.

**PROJECT CODE**

**Key Classes:**

**Client**: Represents individual clients of the salon. Stores client-specific information such as name, contact details, and appointment history.

*Methods*: getName(), getPhoneNumber(), getEmail()

**Clients**: Manages a collection of Client objects. Provides methods to add, remove, print, and find clients.

*Methods*: addClient(), removeClient(), findClient()

**Reservation**: Represents an appointment or booking between a client and a stylist. Stores details such as appointment date, time, service type, and status.

*Methods*: getDate(), getTime(), getServiceType(), getClientName(), getPhoneNum(), getEmail()

**Reservations**: Manages a collection of Reservation objects. Provides methods to add, remove, print, and find reservations.

*Methods*: addReservation(), removeReservationByName(), findReservationByName(), printReservations()

**Stylist**: Represents individual stylists working at the salon. Stores stylist-specific information such as name, specialty, and availability schedule.

*Methods*: getName(), getRate(), getReservation(),

**Stylists**: Manages a collection of Stylist objects. Provides methods to add, remove, print, and find stylists.

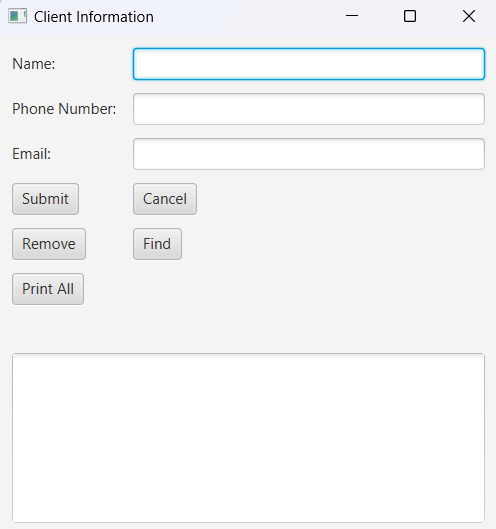
*Methods*: addStylist(), removeStylist(), findStylist(), printStylist()

**USER MANUAL**

**Add**

Client Information:

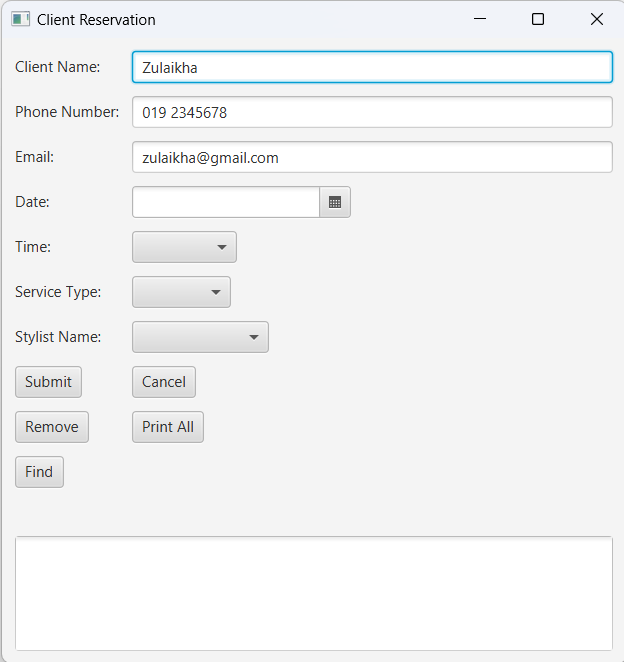
1. The customer provides their name, phone number, and email address, then confirms the information by clicking **Submit**.



Client Information Interface

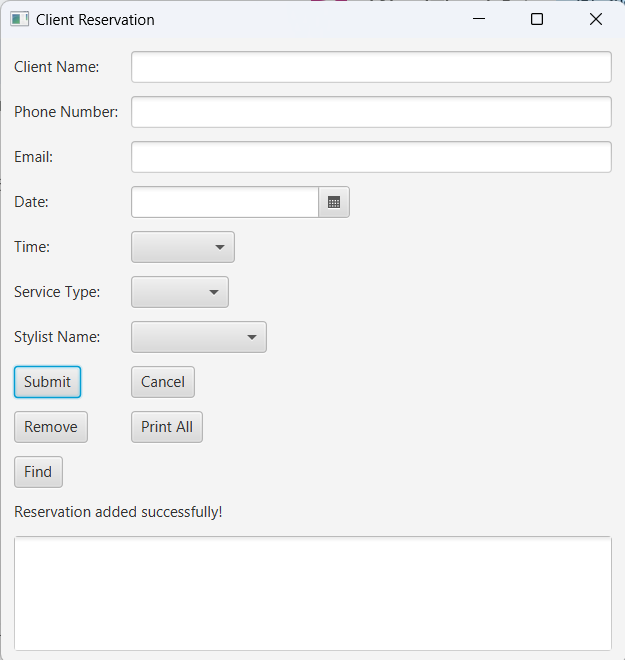
Client Reservation:

1. The customer enters the desired date, time, service type, and stylist for their appointment and then confirms their selection by clicking **Submit**.



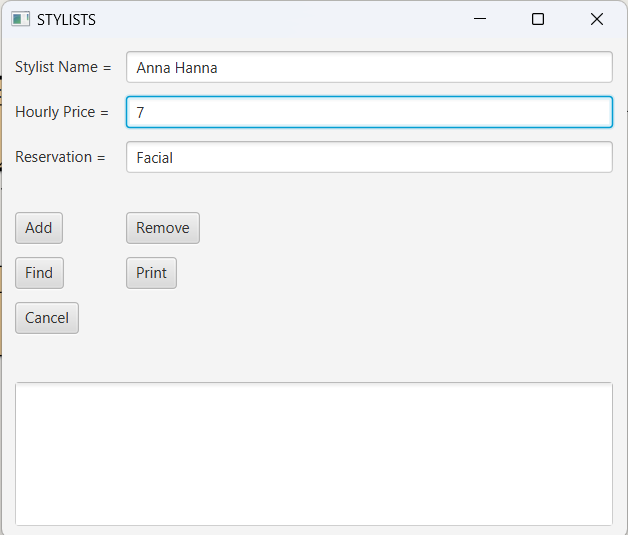
Client Reservation Interface

1. The customer’s reservation is finalized and stored in the system.



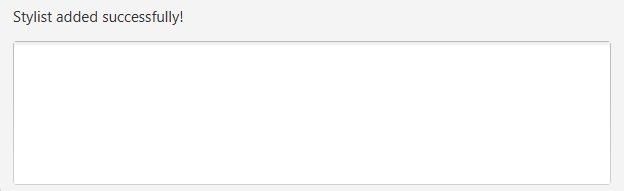
Stylist:

1. Stylists input their hourly charge and confirm by clicking **Add**.



Stylists Interface

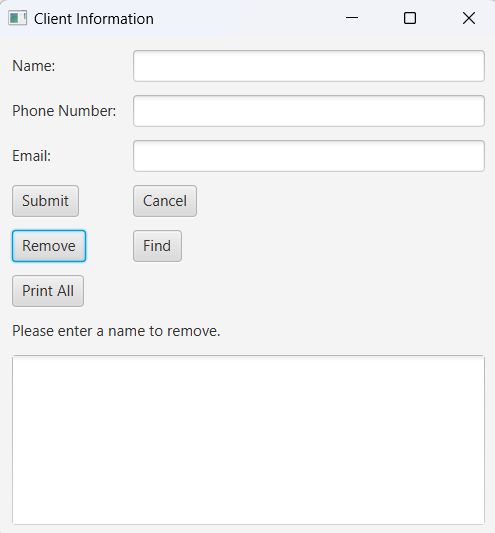
1. The stylist's information has been successfully recorded in the system.



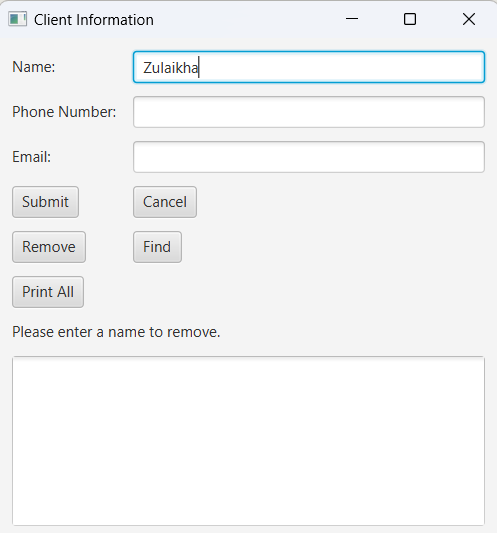
**Remove**

Customer Information:

1. To remove a customer's information, click the **Remove** button.



1. Enter the customer's name and click the **Remove** button again.

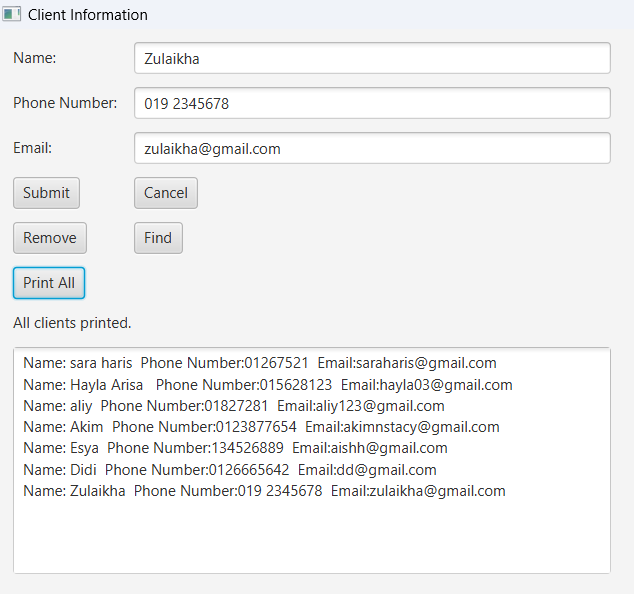


1. The customer’s information is successfully removed from the database.
2. To confirm the cancellation, click **Print All** to view the updated list of reservations.
3. Steps 1-4 are repeated for **Client Reservation** and **Stylist** to remove their information.

**Print**

Client Information:

1. Click **Print All** to generate the list of all customer information.

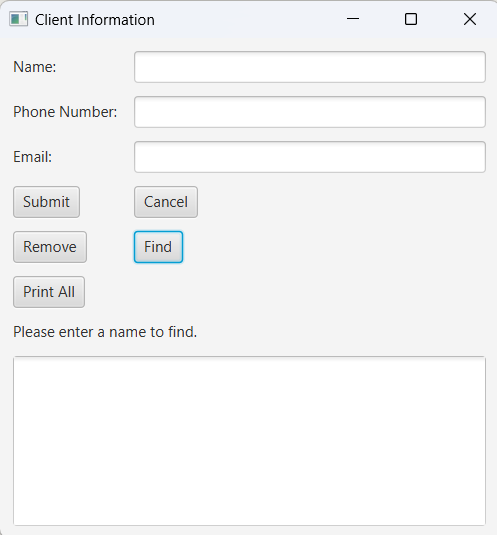


1. The same step is done for **Client Reservation** and **Stylist**.

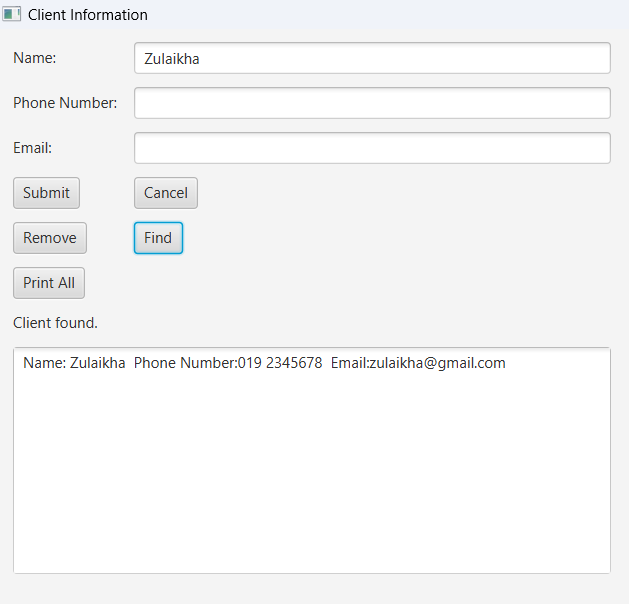
**Find**

Client Information:

1. Click the **Find** button to find a specific customer’s information.



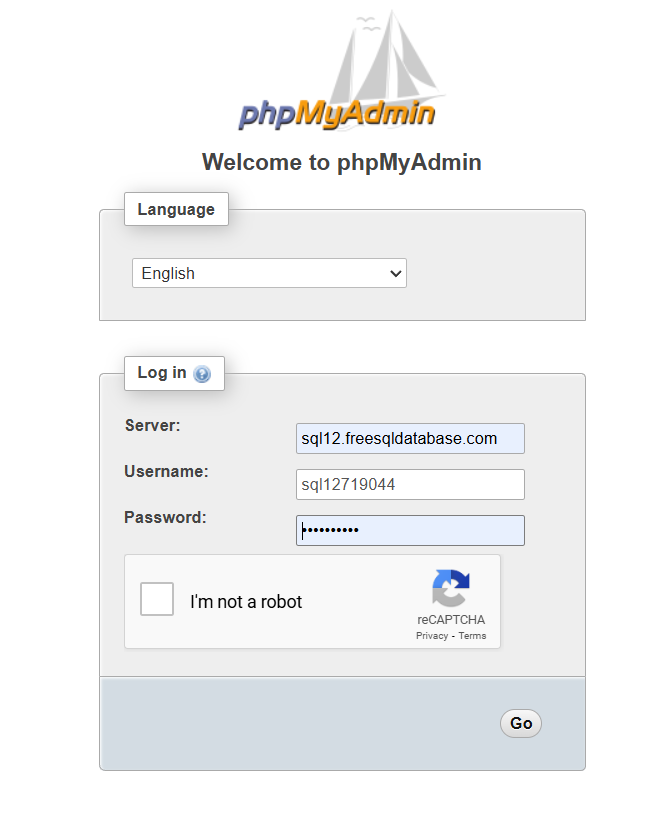
1. Enter the customer's name and click the **Find** button again.



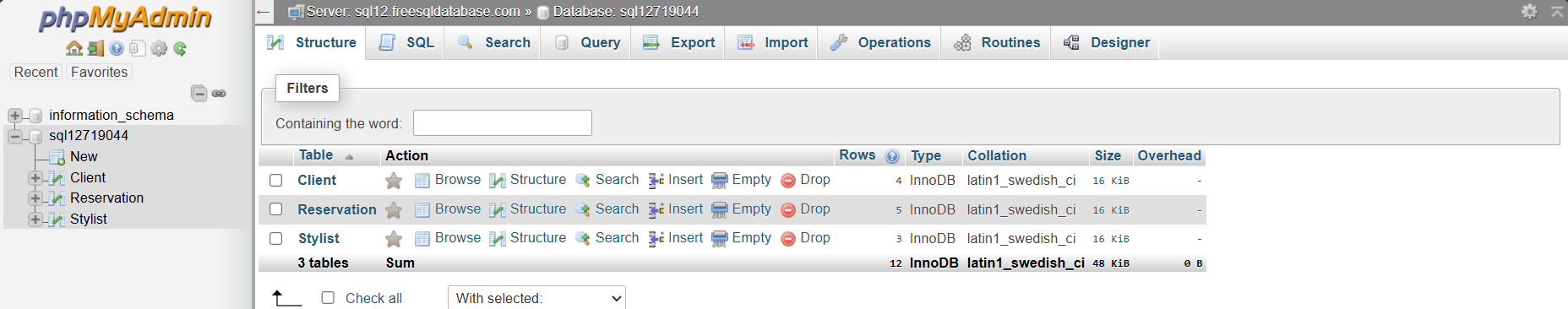
1. Client’s information is displayed.
2. Repeat steps 1-3 to find customer’s reservation details on **Client Reservation** and stylists details on **Stylist**.

**Database - phpMyAdmin**

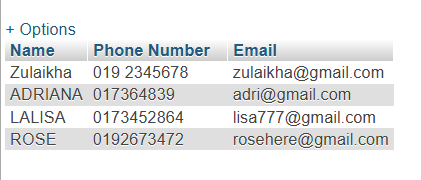
1. To access the database phpMyadmin is used.
2. Log into phpMyAdmin with Username: sql12719044 and Password: PQUSjvXBEP



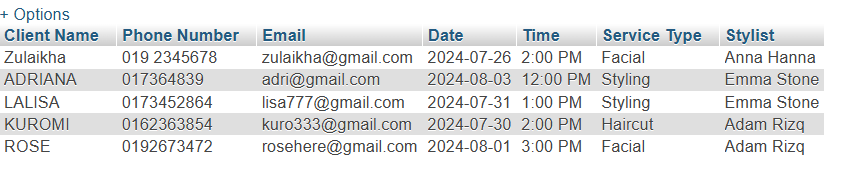
Log in to phpMyAdmin

****

Tables are created for each interface



Client Table

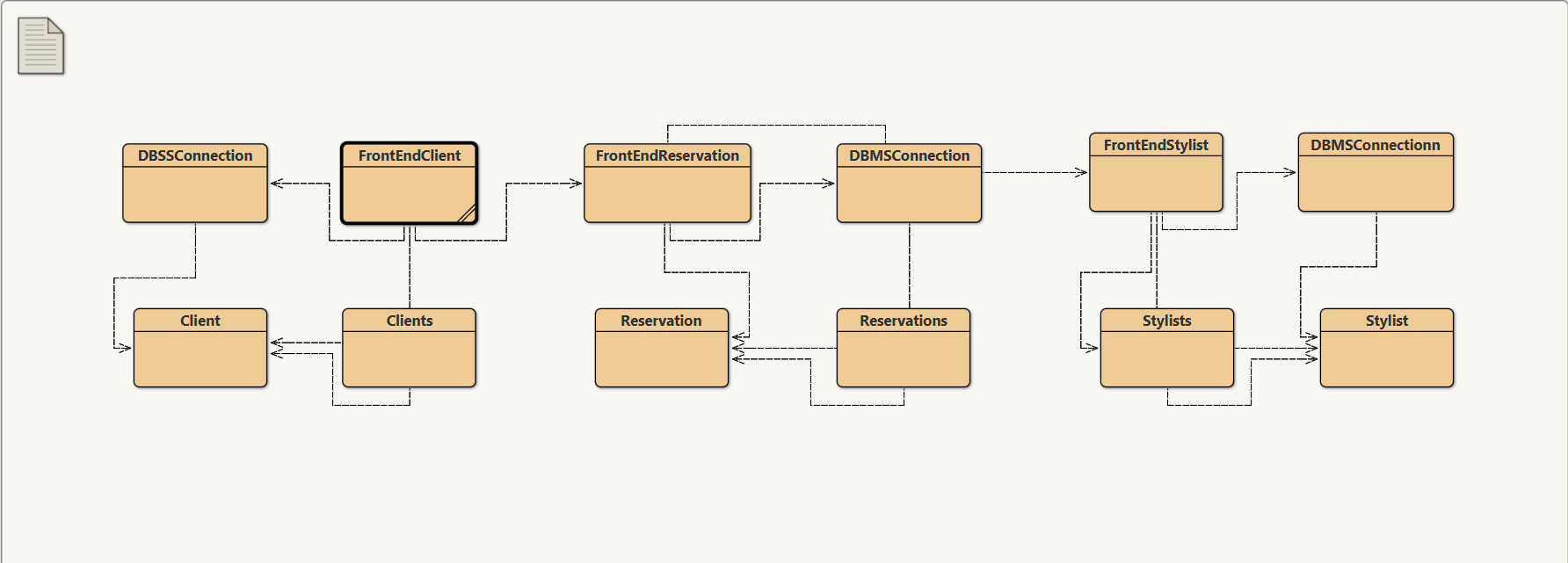


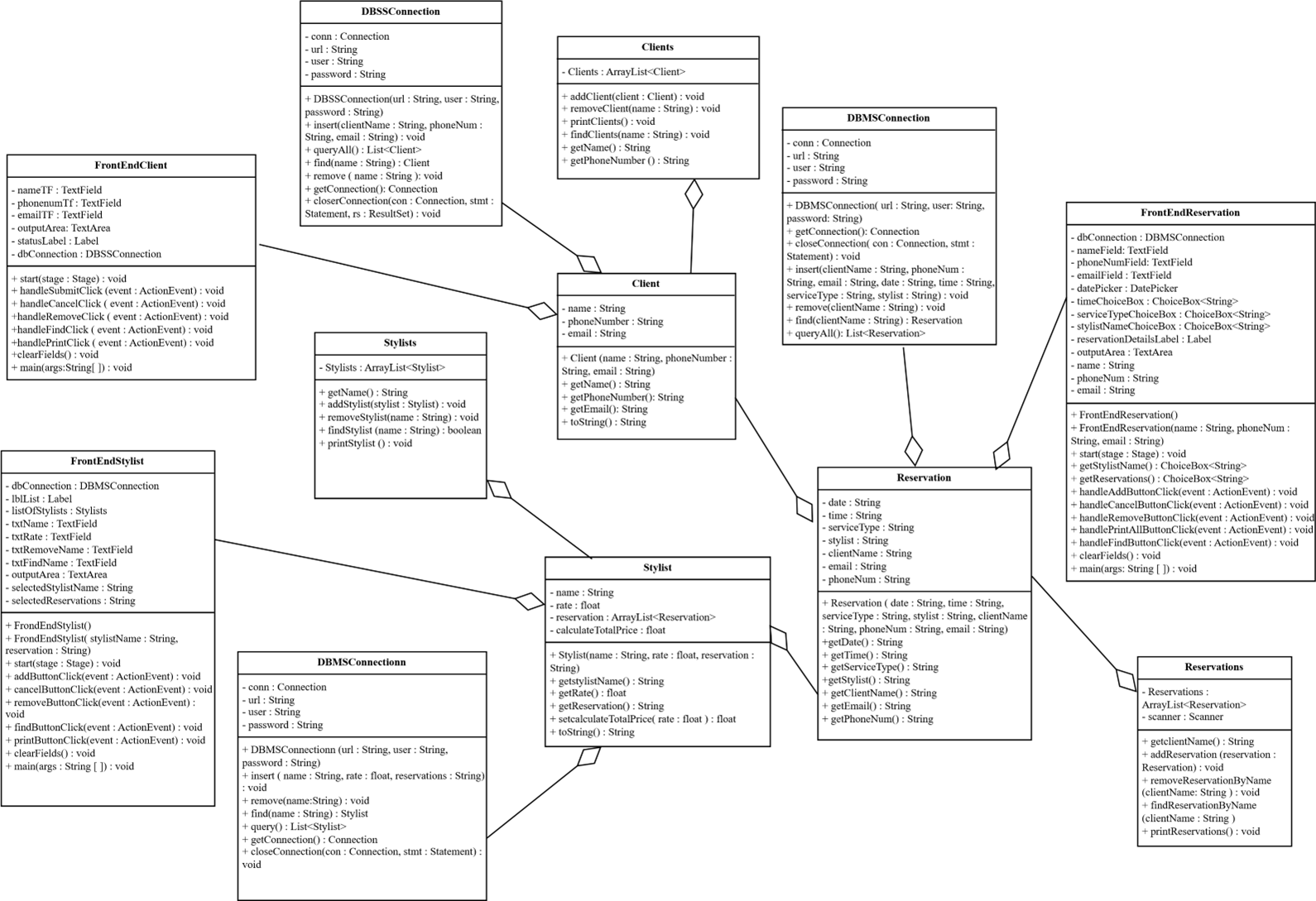
Reservation Table



Stylist Table

**PROJECT UML DIAGRAM**

****

****