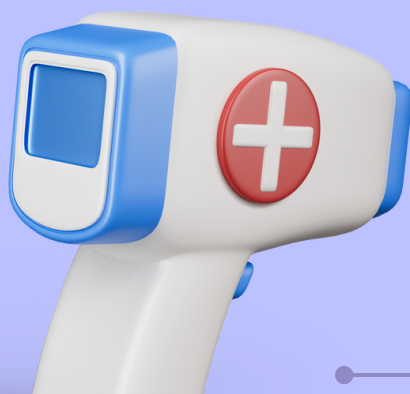




DocCARE MEDICAL SERVICES

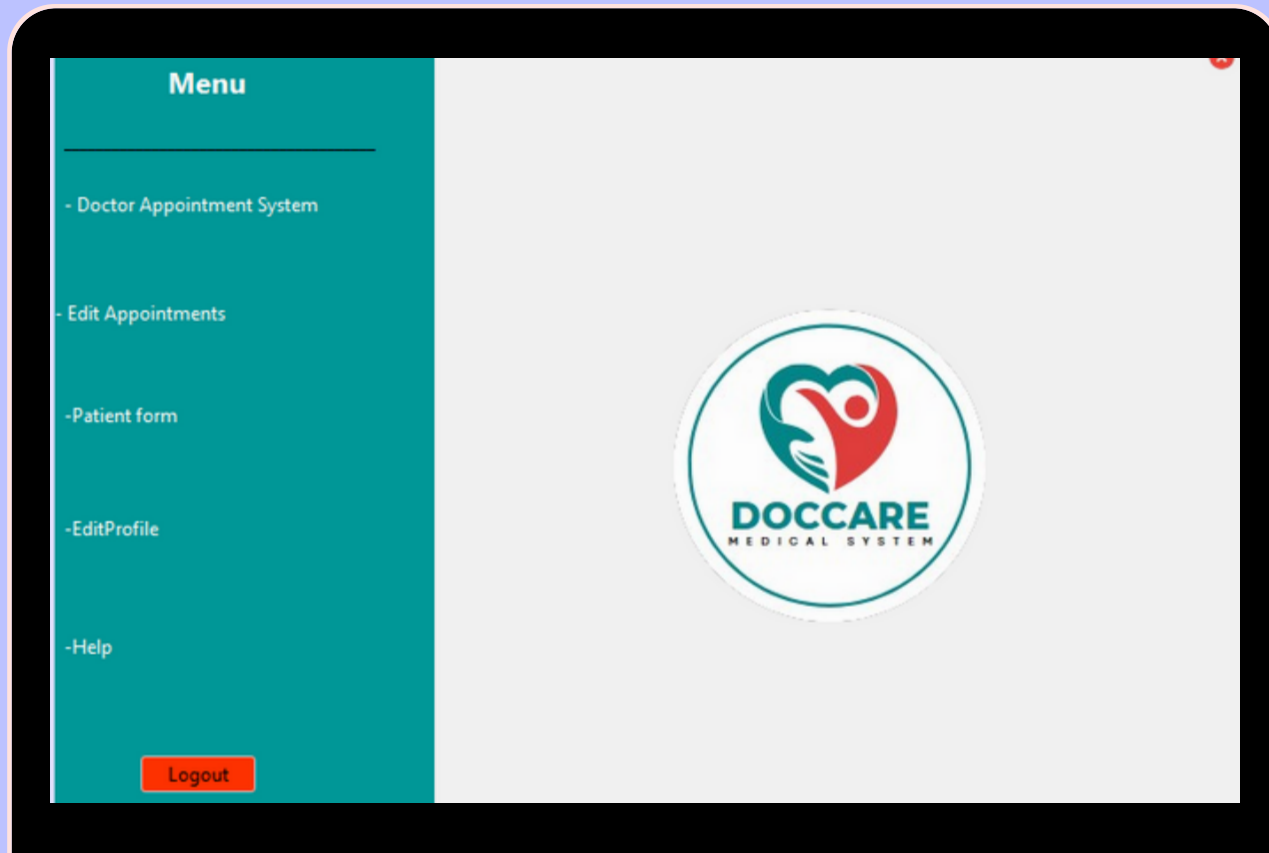
MADE BY

**Leen Humaidan
Hamzeh Hamzeh
Fatima Saleh
Youhana Nageeb**



Introduction

- **DocCare is an innovative Java-based healthcare management system designed to streamline the interaction between patients and healthcare providers. This comprehensive system integrates various modules to offer a seamless experience for both patients and doctors.**



Login and Register

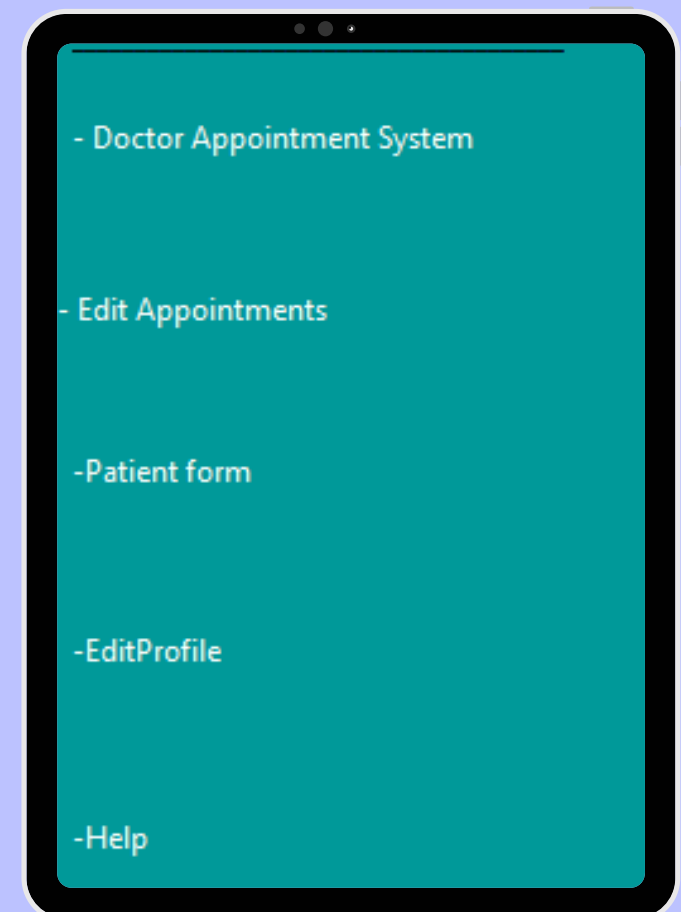
- **DocCare ensures secure and user-friendly authentication with a robust login and registration system. User data is securely stored in a database, allowing for efficient management of patient and doctor profiles.**

LOGIN	Signup
<p>UserName</p> <input type="text"/>	<p>Full name</p> <input type="text"/>
<p>Password</p> <input type="password"/>	<p>UserName</p> <input type="text"/>
<p>Login</p>	<p>Password</p> <input type="password"/>
<p>I don't have an account Sign Up</p>	<p>Sign Up</p>
	<p>I have an account Login</p>

Home page

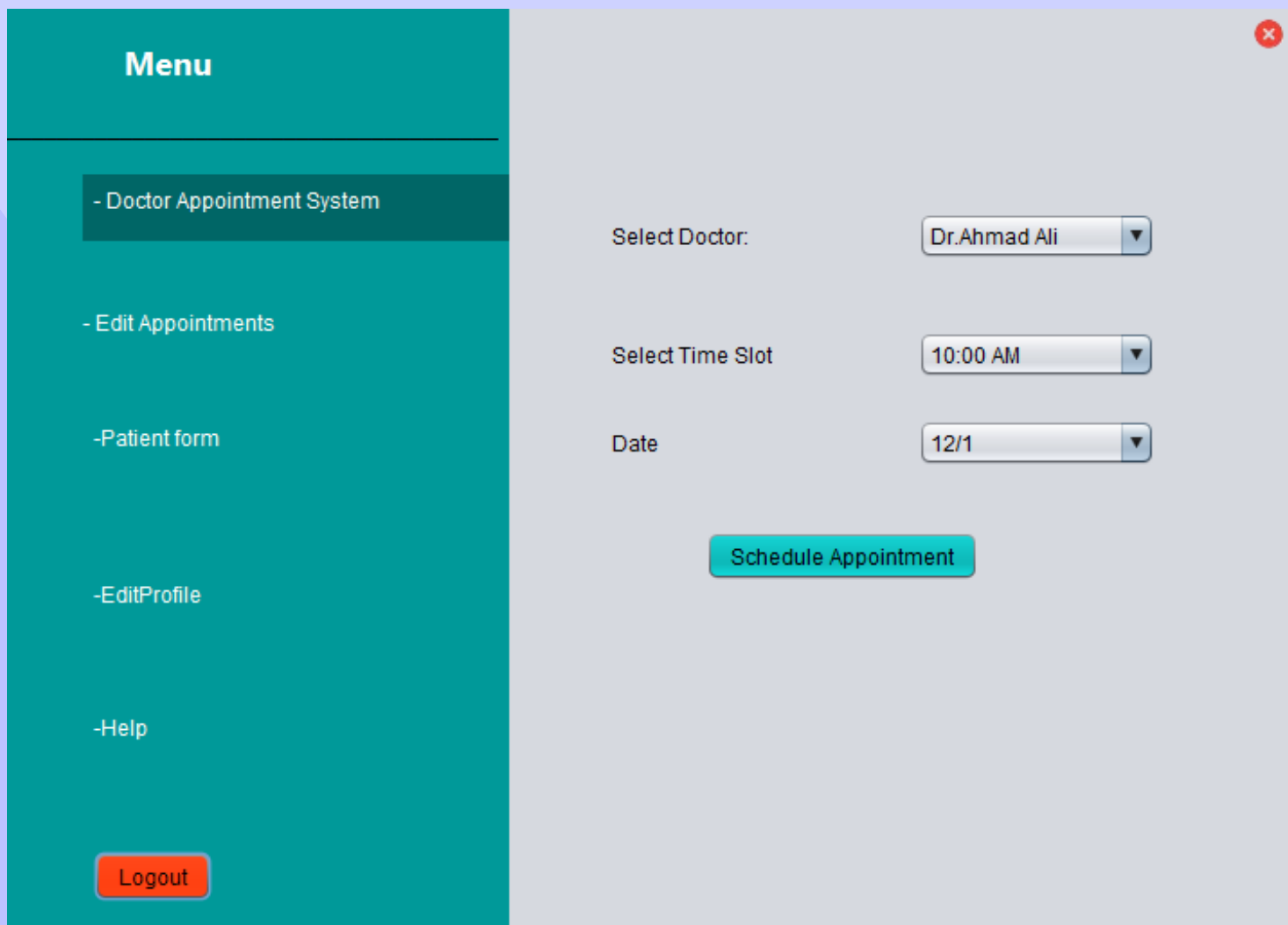
In Home page We have The MenuBar and the menu include:

- **Doctor Appointment System:**
1-Schedule and manage doctor appointments.
- **Edit Appontments:**
1-Edit existing appointments for flexibility.
- **Patient Form:**
1-Complete and update your patient information.
- **Edit Profile:**
1-Add More contact details, including phone and email.
- **Help:**
1-View our customer service email and get assistance.



Appintments

- **Schedule and manage doctor appointments easily. Patients can choose providers, select convenient time slots, and receive confirmation details.**



The screenshot displays a web application for scheduling appointments. On the left is a teal sidebar menu with the title 'Menu' at the top. The menu items are: '- Doctor Appointment System' (highlighted with a dark teal background), '- Edit Appointments', '-Patient form', '-EditProfile', and '-Help'. At the bottom of the sidebar is a red 'Logout' button. The main content area has a light gray background and a red close button in the top right corner. It contains three dropdown menus: 'Select Doctor:' with 'Dr.Ahmad Ali' selected, 'Select Time Slot' with '10:00 AM' selected, and 'Date' with '12/1' selected. Below these is a teal 'Schedule Appointment' button.

Menu

- Doctor Appointment System
- Edit Appointments
- Patient form
- EditProfile
- Help

Logout

Select Doctor: Dr.Ahmad Ali

Select Time Slot 10:00 AM

Date 12/1

Schedule Appointment

Appintments

- **Flexibility in managing scheduled appointments.** Users can modify or update details, allowing rescheduling or cancellations for adaptability. Ensures dynamic scheduling for patients and providers.

The screenshot displays a web application for managing appointments. On the left is a teal sidebar menu with the title "Menu". The menu items are: "- Doctor Appointment System", "- Edit Appointments" (which is highlighted), "-Patient form", "-EditProfile", "-Help", and a red "Logout" button at the bottom. The main content area on the right is light gray and contains a table with three columns: "DoctorName", "Date", and "Time". Above the table is a refresh icon. Below the table are two buttons: "Delete" and "Add". A red close button (X) is in the top right corner of the main area.

DoctorName	Date	Time

Buttons: Delete, Add

Patient Form

- **DocCare simplifies the patient registration process with a user-friendly and comprehensive Patient Form. This form plays a crucial role in capturing essential information from individuals, ensuring that healthcare providers have a solid understanding of the patient's background.**

The image shows a web application interface for a patient form. On the left is a teal sidebar menu with the title "Menu" at the top. Below the title are five menu items: "- Doctor Appointment System", "- Edit Appointments", "-Patient form" (which is highlighted with a darker teal background), "-EditProfile", and "-Help". At the bottom of the sidebar is an orange "Logout" button. The main content area is light gray and contains a form with four input fields labeled "Name:", "Age:", "Gender", and "GuardianName". Each label is followed by a white rectangular input box. At the bottom of the form is a wide, light gray "Submit" button. A small red "x" icon is located in the top right corner of the main content area.

EditProfile

- **Add Phone and Email: Personalize profiles by adding or updating contact information. Users manage profiles by including phone numbers and email addresses, ensuring accurate communication.**

Menu

- Doctor Appointment System

- Edit Appointments

-Patient form

-EditProfile

-Help

Logout


EditProfile:

Email:

Submit

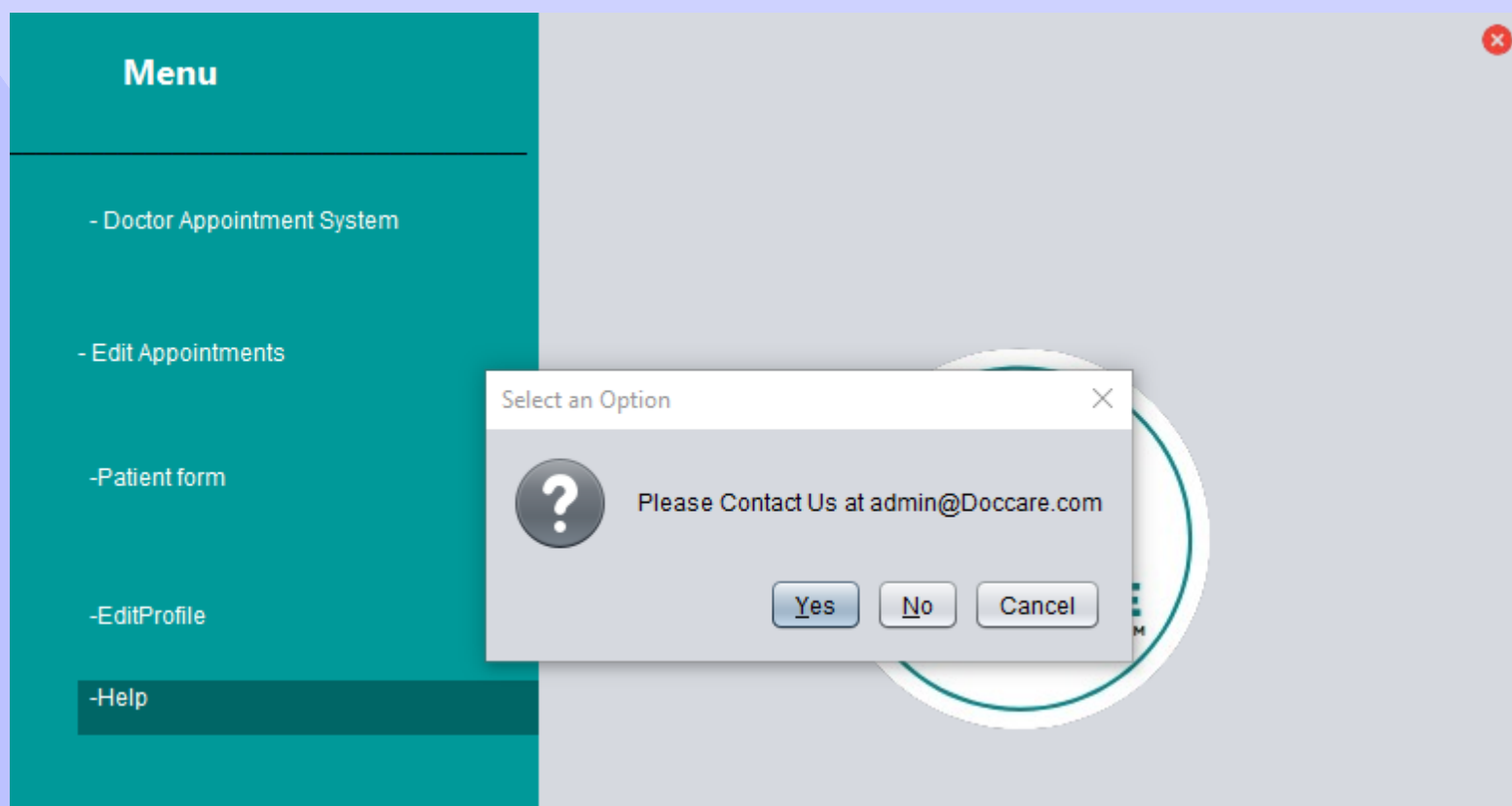
Phone:

+962



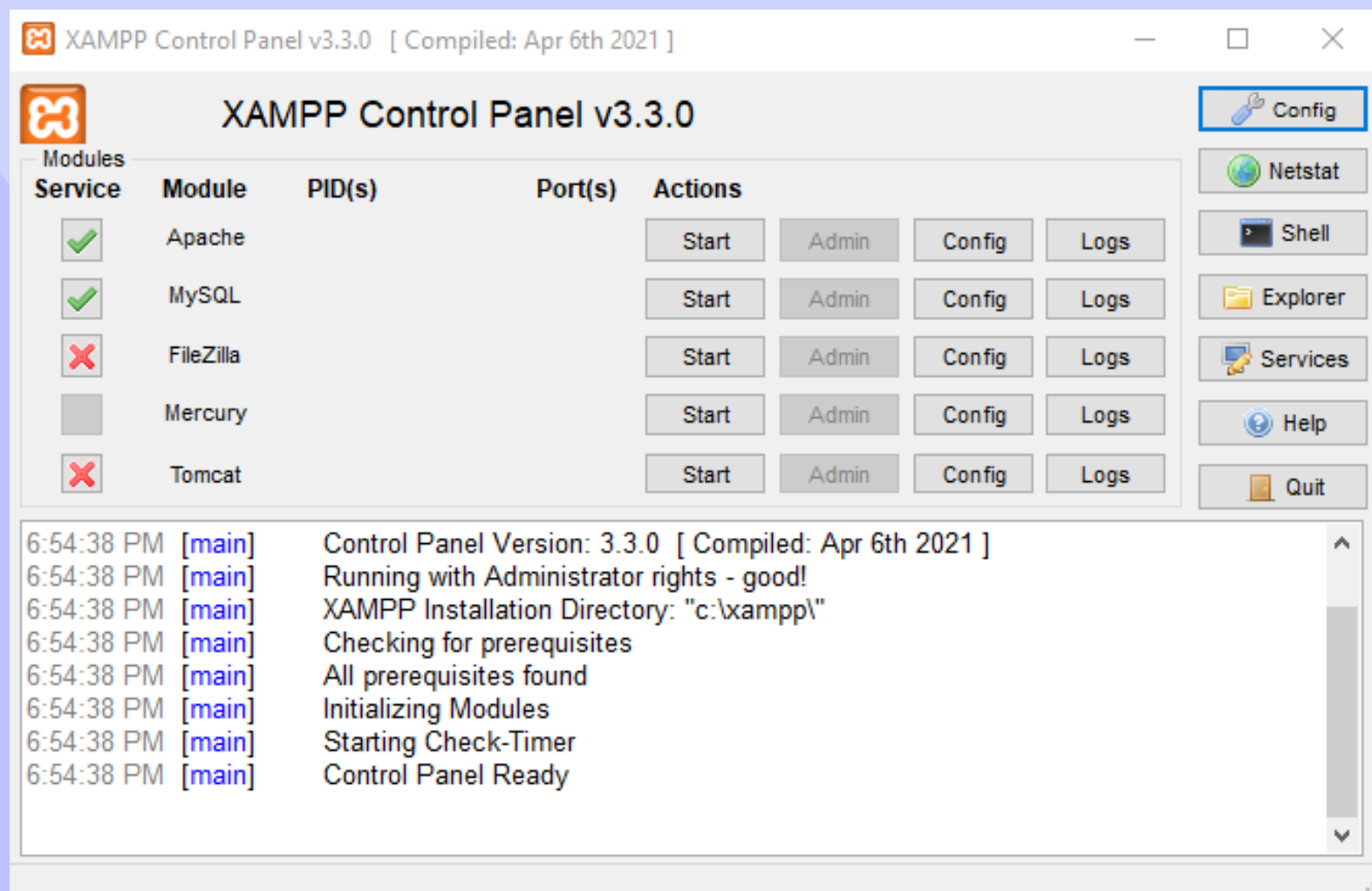
Help

- **View Our Customer Service Email: Access customer support information easily. The Help menu provides details on reaching customer service**



DataBase

- **DocCare relies on a robust and efficient database system powered by XAMPP. XAMPP, a cross-platform web server solution, provides the technological backbone for the storage, retrieval, and management of critical data within the DocCare System**

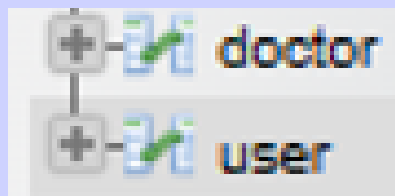


Our DataBase

- Our DataBase have two Tables:

1. User Table: For Login And Register and Also We use it in EditProfile Class.

2. Doctor Table: For Appintments

A screenshot of the phpMyAdmin interface. The left sidebar shows the database structure with 'login' and 'user' tables selected. The main panel displays the 'user' table structure and data. The table has columns: id, Name, User_Name, Email, Phone, and Password. The data row shows: 1, test_me, test, test@gmail.com, 0799332978, test123. The interface includes various navigation and query tools at the top and bottom.

phpMyAdmin

Server: 127.0.0.1 » Database: login » Table: user

Showing rows 0 - 3 (4 total, Query took 0.0015 seconds.)

`SELECT * FROM `user``

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

	id	Name	User_Name	Email	Phone	Password
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1	test_me	test	test@gmail.com	0799332978	test123

Check all | With selected: ☐ Edit ☐ Copy ☐ Delete ☐ Export

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Query results operations

Print | Copy to clipboard | Export | Display chart | Create view

Bookmark this SQL query

Label: ☐ Let every user access this bookmark

Bookmark this SQL query

GUI-Swing

- **JLabel:**
 - Used to display a non-editable text or image.
 - Provides a simple way to add descriptive text to a GUI.
- **JFrame:**
 - The main container representing the application window.
 - Provides the overall structure for GUI components.
- **JButton:**
 - Represents a clickable button.
 - Triggers actions when clicked, making it essential for user interaction.
- **JTextField:**
 - A single-line text input field.
 - Allows users to input and edit text.
- **JComboBox:**
 - A drop-down menu that allows the user to choose from a list of options.
 - Useful for selecting one option from a predefined set.
- **JPanel:**
 - A container used to organize and group components.
 - Can be added to a JFrame to organize the layout.
- **JDialog:**
 - A pop-up window that can be used for messages, input forms, or other temporary interactions.
 - Typically smaller than a JFrame and often modal.
- **JMenuBar, JMenu, JMenuItem:**
 - Components for creating menus in a GUI.
 - JMenuBar contains JMenu items, which, in turn, contain JMenuItem items for specific actions.

CODING



MainClasses



DocCare System

- Class DataBase
- Class Login
- Class Register
- Class Home
- Class Appointment
- Class edit Appointment
- Class PatientForm
- Class EditProfile

#JAVA

Some Methods

setBackground();

getText();

isEmpty();

showMessageDialog();

setVisible();

parseInt();

getSelectedItem();

dispose();

DataBase Conection

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;

public class db {
    public static Connection mycon() {

        Connection con = null;

        try {
            // Load the JDBC driver
            Class.forName("com.mysql.cj.jdbc.Driver");
            // Establish the connection
            con = DriverManager.getConnection("jdbc:mysql://127.0.0.1:3306/login", "root", "");

        } catch (ClassNotFoundException | SQLException e) {
            e.printStackTrace(); // Use printStackTrace() to print the exception details
        }

        return con;
    }
}
```