Faculty of Technology
Department of Computer Engineering
Dharmsinh Desai University

A
PROJECT REPORT ON

Online appointment Management System

By

Dhruvin Gamit	**CE077**	**21CEUTS109**
Abhishek Jamkar	**CE087**	**21CEUTF069**

B. Tech. CE Semester – VI
Subject: System Design Practices

Guided by:
Prof. Niyati J. Buch
Assistant Professor
Department of Computer Engineering

Faculty of Technology
Department of Computer Engineering
Dharmsinh Desai University

CERTIFICATE

This is to certify that the practical / term work carried out in the subject of **System Design Practices** and recorded in this journal is the bonafide work of

Dhruvin Gamit	**CE077**	**21CEUTS109**
Abhishek Jamkar	**CE087**	**21CEUTF069**

of B.Tech semester **VI** in the branch of **Computer Engineering** during the academic year **2023-2024**

Prof. Niyati J. Buch Assistant Professor, Dept. of Computer Engg. Faculty of Technology Dharmsinh Desai University, Nadiad	Dr. C. K. Bhensdadia Head of the Department, Dept. of Computer Engg. Faculty of Technology Dharmsinh Desai University, Nadiad

Table of Contents

Abstract

The Online Appointment Management System (OAMS) is a robust software solution designed to simplify and optimize the organization and administration of appointments across various service providers and clients. In today's fast-paced world, efficient management tools are essential to ensure smooth coordination and seamless scheduling of appointments.

1. Introduction

1.1 Brief Introduction:-

In today's fast-paced world, managing appointments efficiently has become a critical aspect of various sectors, including healthcare, education, and professional services. With the increasing reliance on technology, there is a growing demand for robust online appointment management systems to streamline the booking process and enhance overall efficiency.

The Online Appointment Management System (OAMS) is designed to address the challenges associated with scheduling and managing appointments in various domains. Whether it's booking doctor's appointments, scheduling meetings with advisors, or arranging consultations with service providers, OAMS provides a centralized platform for seamless appointment scheduling and management.

With OAMS, users can easily book appointments online, view available time slots, and receive automated reminders to ensure they never miss an appointment. The system also offers features such as calendar integration, real-time availability updates, and secure payment processing, making the appointment booking process convenient and hassle-free for both service providers and clients.

By leveraging technology to streamline appointment management processes, OAMS aims to enhance productivity, improve customer satisfaction, and optimize resource allocation across different sectors. Whether it's optimizing clinic workflows, improving student advisory services, or maximizing business efficiency, OAMS serves as a versatile solution to meet the diverse needs of appointment-based industries.

1.2 Tools, Technology and Platform used:-

1) Programming Languages: JavaScript, HTML and CSS(MERN stack)

2) IDE: Visual Studio Code

3) **MongoDB** as Database server

4) **Express.js** as web application framework for Node.js.

5) **React** for building user interfaces

6) **Node.js** is a server-side JavaScript runtime built on Chrome's V8 JavaScript engine.

2. Software Requirements Specifications

2.1 Purpose

The purpose of this document is to outline the requirements and scope for the development of the Online Appointment Management System (OAMS). OAMS is designed to streamline the process of scheduling and managing appointments for various service providers and clients. This document aims to provide a comprehensive overview of the system's functionalities and features to guide the development process effectively.

2.2 Product Scope

OAMS will serve as a comprehensive software solution for managing appointments across different sectors, including healthcare, education, and professional services. The system will facilitate appointment scheduling, modification, and cancellation for both service providers and clients.

2.3 Product Features

The key features of OAMS include:

- **User Authentication**: Secure login and registration functionality for both service providers and clients.
- **Service Browsing**: Capability for clients to browse available services and service providers.
- **Appointment Booking**: Ability for clients to schedule appointments with service providers based on availability.
- **Appointment Management**: Tools for service providers to manage their schedules, view appointment details, and update availability.
- **Appointment History**: Access to appointment history for both service providers and clients to track past appointments.
- **Appointment Cancellation**: Option for clients to cancel pending appointments as needed.

2.4 System Functional Requirements

R.1 User Authentication

R.1.1 Service Providers Login:
- **Description:** Service Providers log in to the system using their existing accounts.
- **Input:** Admin username and password.
- **Processing:** Authenticating login details.
- **Output:** Service Providers access the system upon successful login.

R.1.2 User Login:
- **Description:** Users log in to the system using their valid credentials.
- **Input:** User's valid username and password.
- **Processing:** Authenticating user login details.
- **Output:** Users redirected to the site upon successful login.

R.2 Add New Service:
- **Description:** User adds a new service type or model to the system.
- **Input:** Service details.
- **Processing:** New service added to the database.
- **Output:** User views newly added service details.

R.3 Update Service Details:
- **Description:** User modifies details of an existing service.
- **Input:** Updated service details.
- **Processing:** Previous service details replaced with the new information in the database.
- **Output:** Admin views the updated service details.

R.4 Remove Service:

- Description: User removes a service model or an entire service type from the system.

- Input: Service removal request.

- Processing: Service model/type removed from the database.

- Output: Admin can no longer view removed services in the database.

R.5 User Registration:

- Description: Users create new accounts by providing valid information.

- Input: Name, contact number, email address, and address.

- Processing: Authenticate and register customer details.

- Output: Admin/Manager views customer profiles.

R.6 View Service Details:

- Description: Admin views detailed information about a selected service.

- Input: Selection of the service.

- Processing: Fetching all service details from the database.

- Output: Details of the service, including name, picture, description, address, cost and contact information.

R.7 Book Service

R.7.1 Select Service:

- Description: Users select the service they want.

- Input: Selected service.

- Output: Prompt users to provide details.

R.7.2 Provide Details:

- Description: Users provide necessary details for booking the service.

- Input: User details for service booking.

- Output: Users prompted with payment options.

R.7.3 Select Payment Option:

- Description: Users select their preferred payment method.

- Input: Selected payment option.

- Output: Provide the requested payment method to pay the bill.

R.8 Cancel Service Booking:

- Description: Users can cancel their booked service.

- Input: Click on the cancel button.

- Processing: Check for existing booking and initiate refund process if applicable.

- Output: Refund money for cancelled bookings.

R.9 Service Provider Accept/Reject :-

R.9.1 Accept Service Request:

- Description: Users can accept their booked service.

- Input: Click on the accept button.

- Processing: The company sends the service requester an email that their request has been accepted.

- Output: Email sent.

R.9.2 Reject Service Request:

- Description: Users can reject their booked service.

- Input: Click on the reject button.

- Processing: Check for existing booking and initiate refund process if applicable. The company sends the service requester an email that their request has been rejected.

- Output: Email sent and refund.

2.5 Other Non-functional Requirements

1. **Performance Requirements**
    - Response Time: The system must respond to user actions within 3 seconds under typical load conditions.
    - High Performance: Task execution should be swift, with complex tasks taking around 50 seconds and simpler tasks taking 20-25 seconds.
2. **Safety Requirements**
    - Risk Indication: The system should warn users about potential risks related to actions.
    - Data Security: Validate user input to prevent data injection attacks.
    - Error Handling: Implement error recovery processes for system errors and exceptions.
3. **Security Requirements**
    - Data Encryption: Utilize encryption to safeguard sensitive information, like user passwords.
    - Regular Security Assessments: Conduct routine vulnerability assessments and penetration tests.
4. **Software Quality Attributes**
    - High Availability: Aim for system uptime of at least 99.5%.
    - Security: Ensure robust protection against unauthorized access and data breaches.
    - Scalability: The system should handle increased traffic and data storage efficiently.

3. Design

3.1 Use Case Diagram

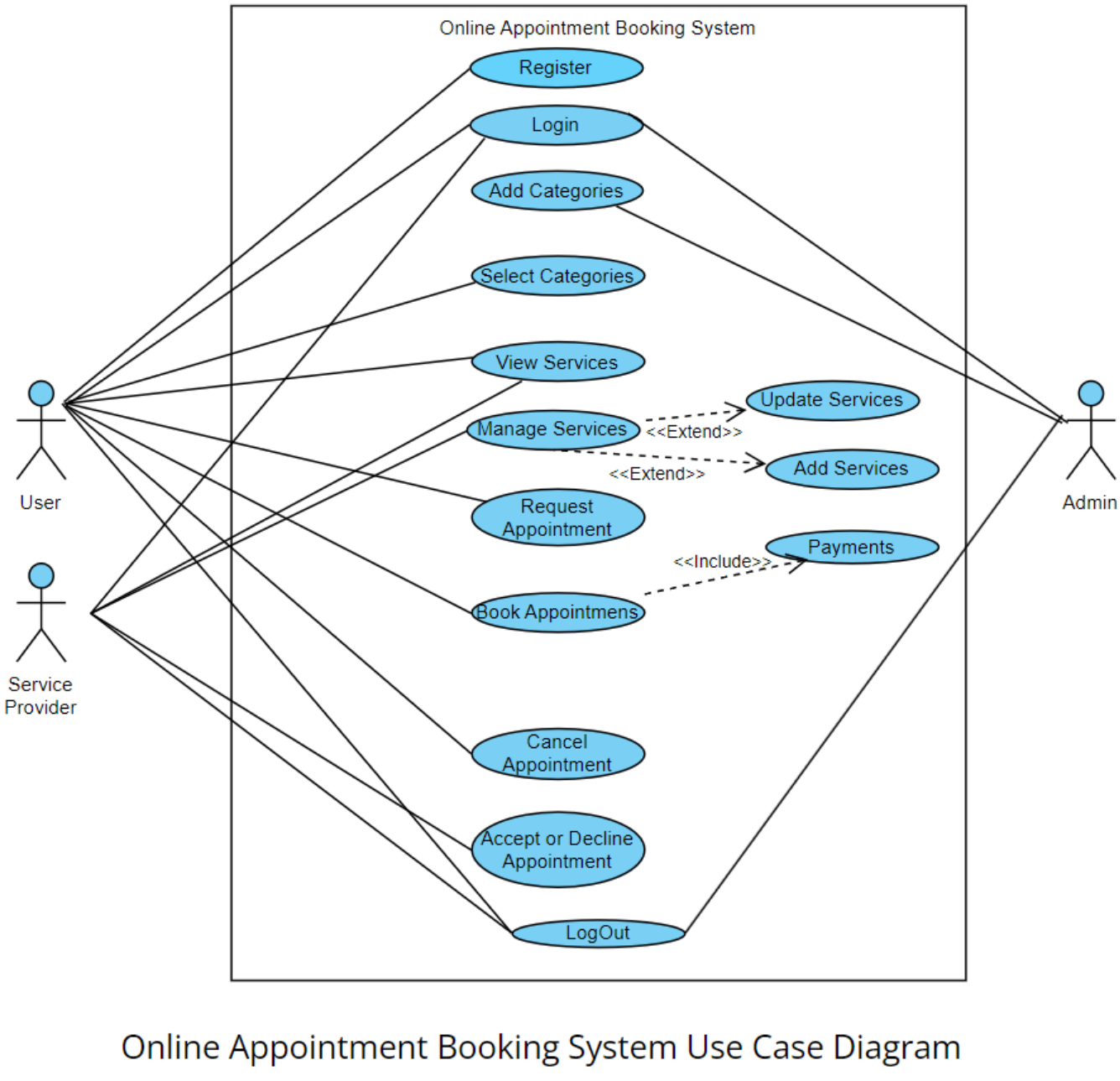
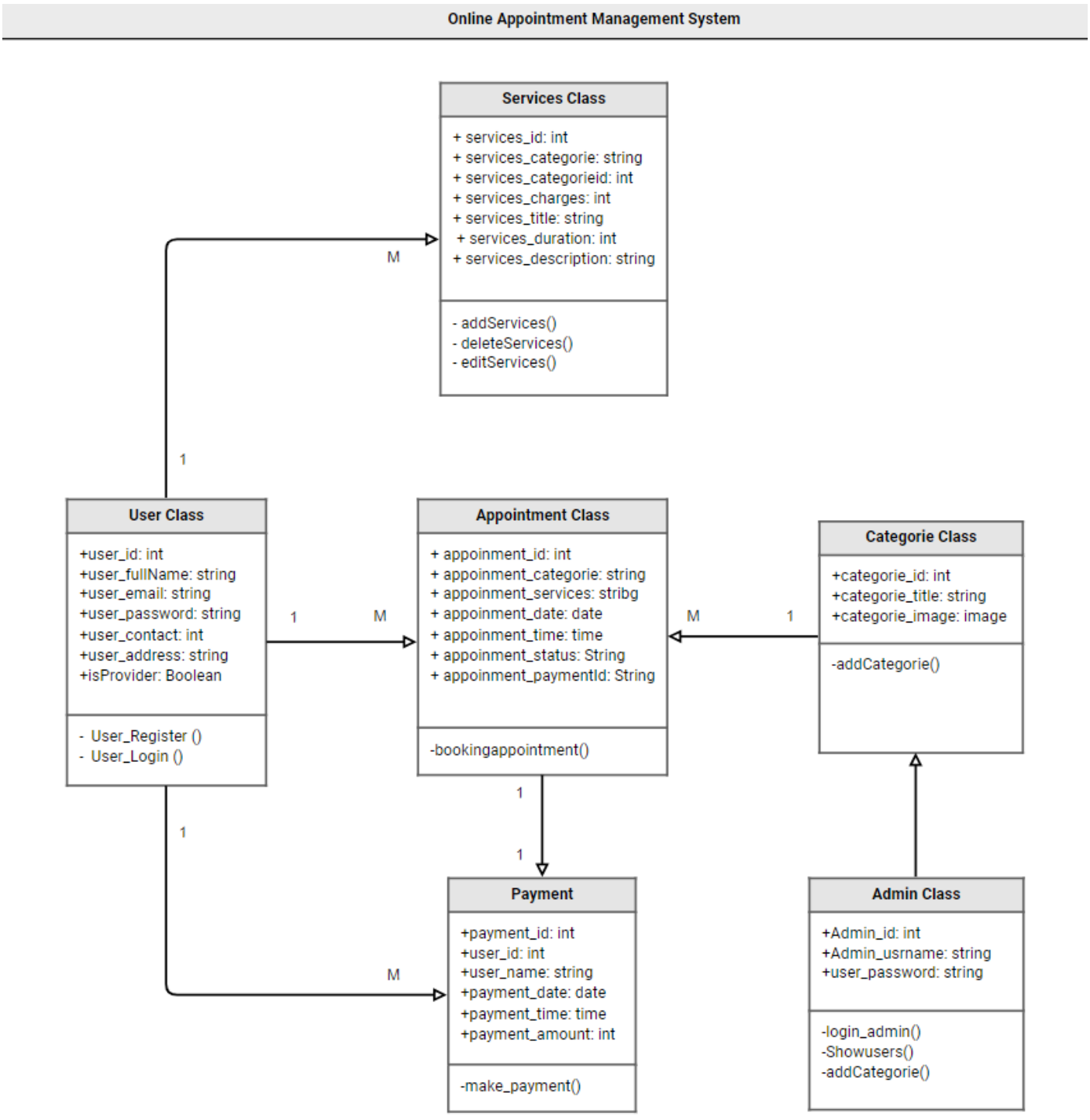


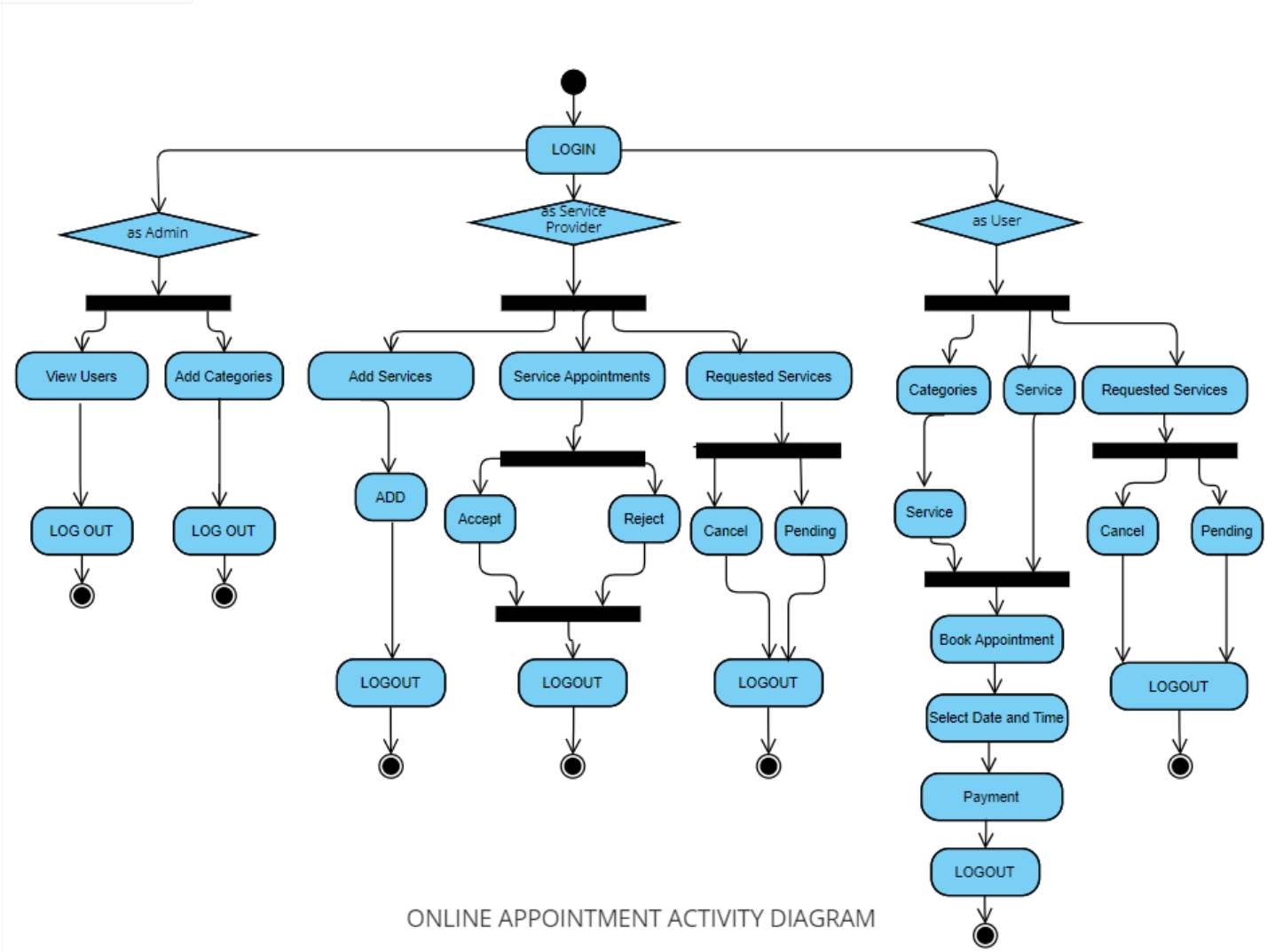
Fig. 3.1: Use Case Diagram

3.2 Class Diagram



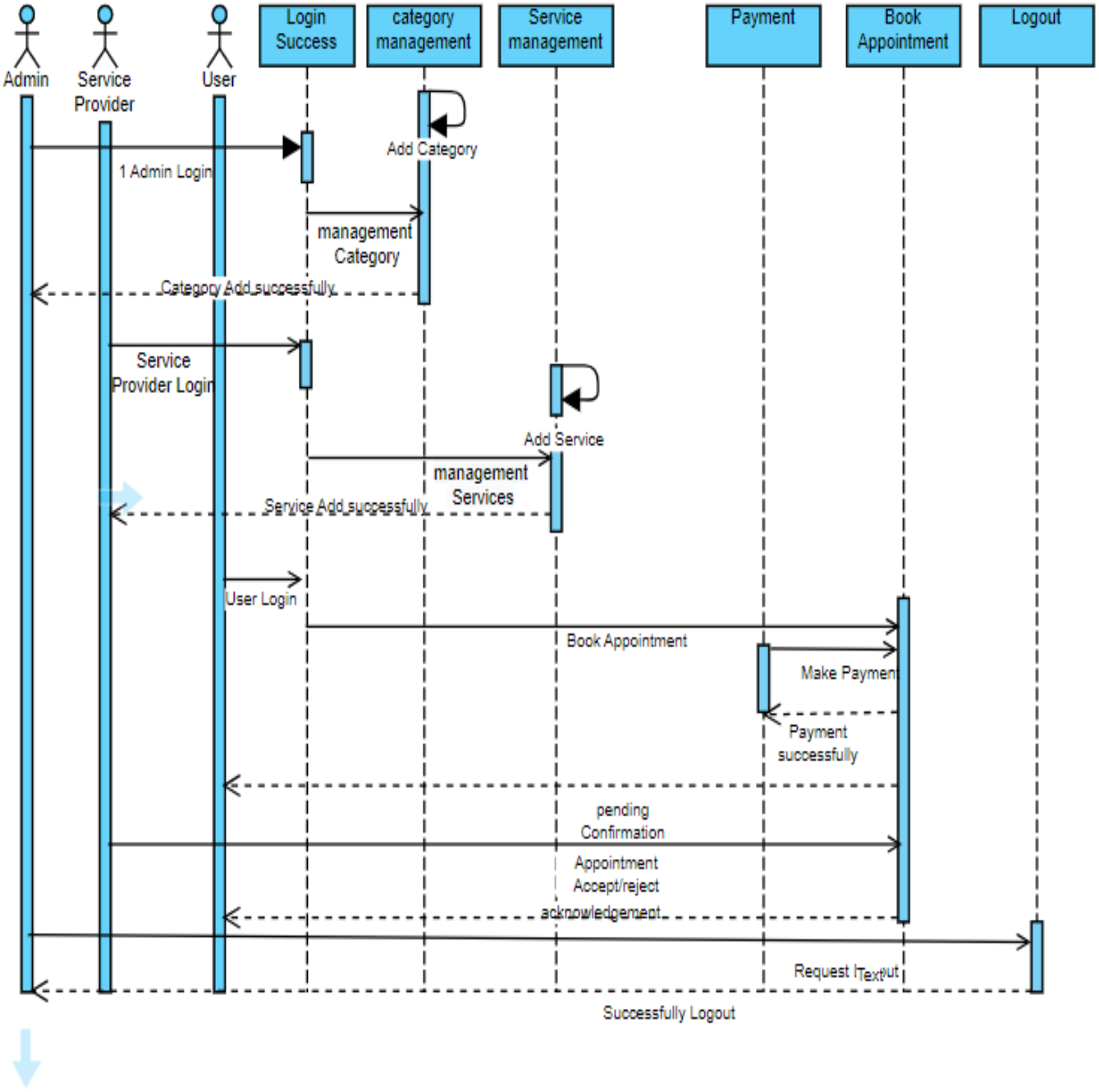
3.2 Class Diagram

3.3 Activity Diagram



3.3 Activity diagram

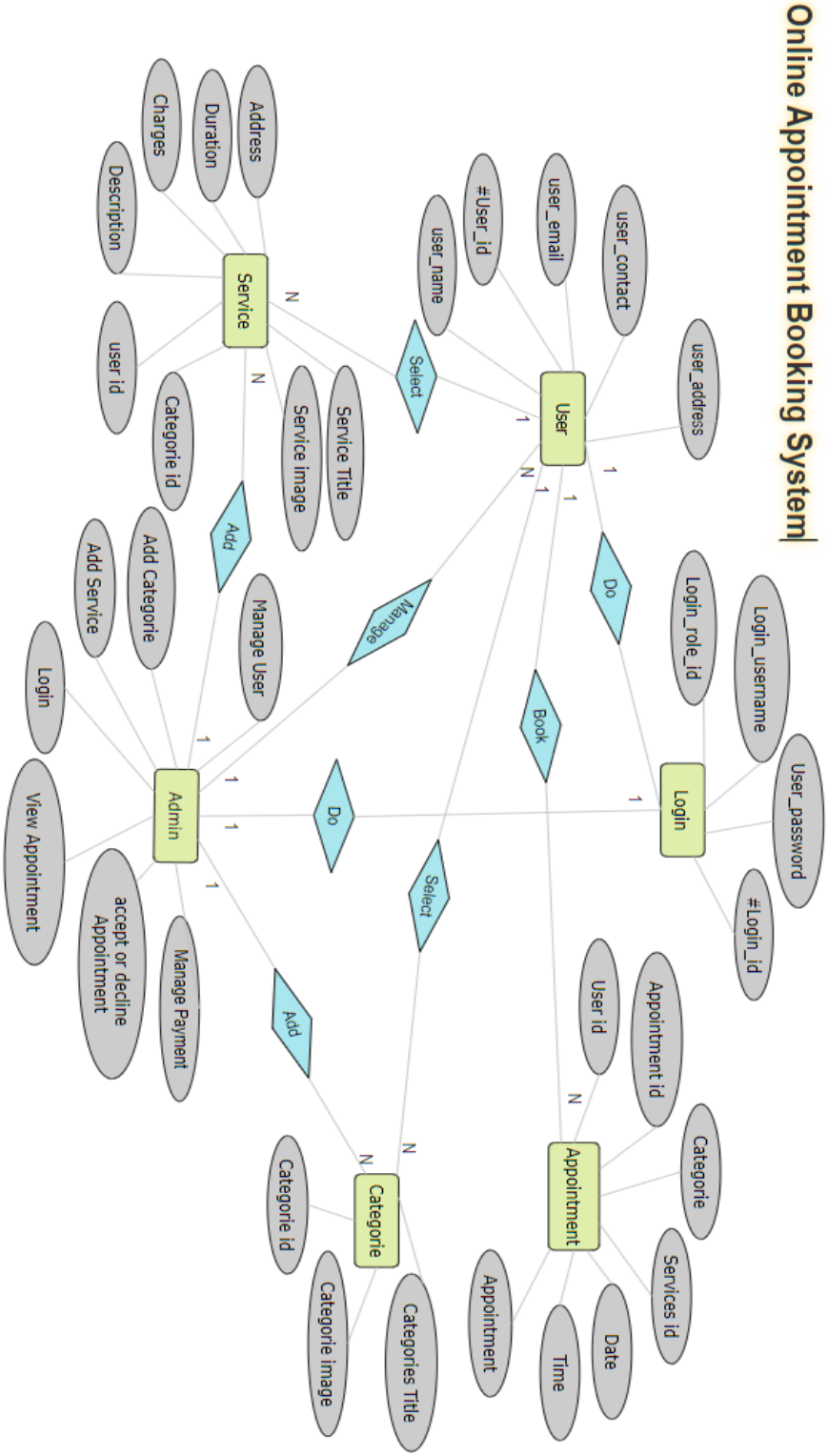
3.4 Sequence Diagram



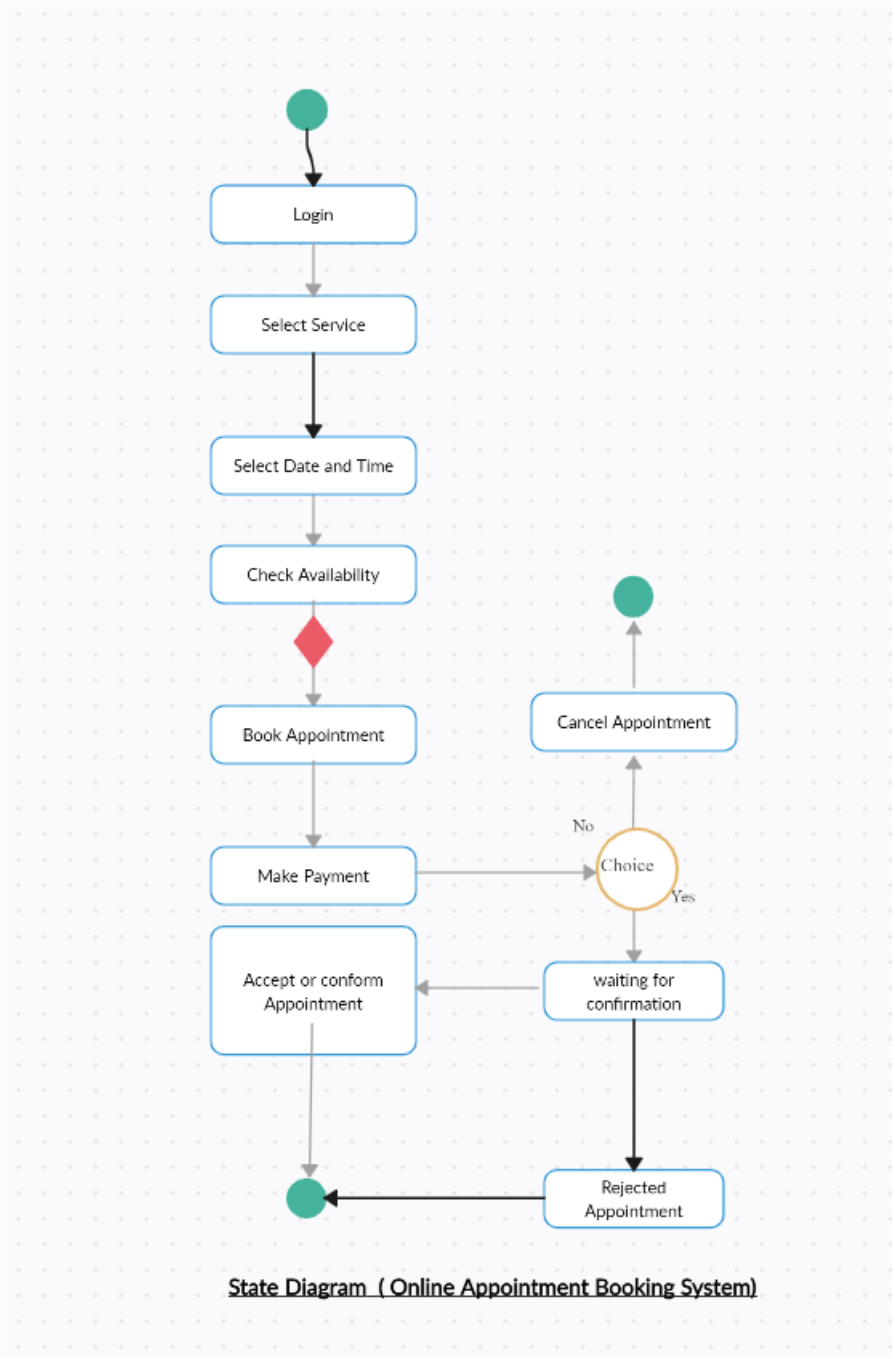
Online Appointment Booking System Sequence Diagram

3.4 Sequence Diagram

3.5 ER Diagram



3.6 State Diagram



State Diagram (Online Appointment Booking System)

4. Implementation Details

The implementation of the Online Appointment Management System (OAMS) was realized using a modern technology stack and frameworks to ensure efficiency, scalability, and maintainability.

4.1 Technology Stack and Frameworks

For the implementation of various functionalities and features, the following technologies and frameworks were utilized:

MongoDB: As the NoSQL database, MongoDB provided a flexible and scalable solution for storing data related to appointments, users and service providers.

Express.js: Used to build the backend server, Express.js provided a robust framework for creating RESTful APIs to handle requests and responses between the frontend and the database. It facilitated the implementation of user authentication and service booking.

React.js: Employed for building the frontend user interface, React.js facilitated the development of interactive and responsive components for managing appointments, user profiles, and service details.

Node.js: As the server-side runtime environment, Node.js enabled the execution of JavaScript code on the server, powering the backend logic and communication with the database. It facilitated real-time interactions between users and service providers, ensuring smooth and efficient appointment management processes.

By leveraging the MERN (MongoDB, Express.js, React.js, Node.js) stack, the Online Appointment Management System was developed to meet the requirements for performance, security, usability, reliability, scalability, and compatibility, providing users with a seamless and efficient platform for booking services and providing feedback.

4.2 Modules created and brief description of each module:

1. **User Authentication:**
    - **Description:** Manages user authentication processes, including login and logout functionalities. Ensures secure access to the system using valid credentials.

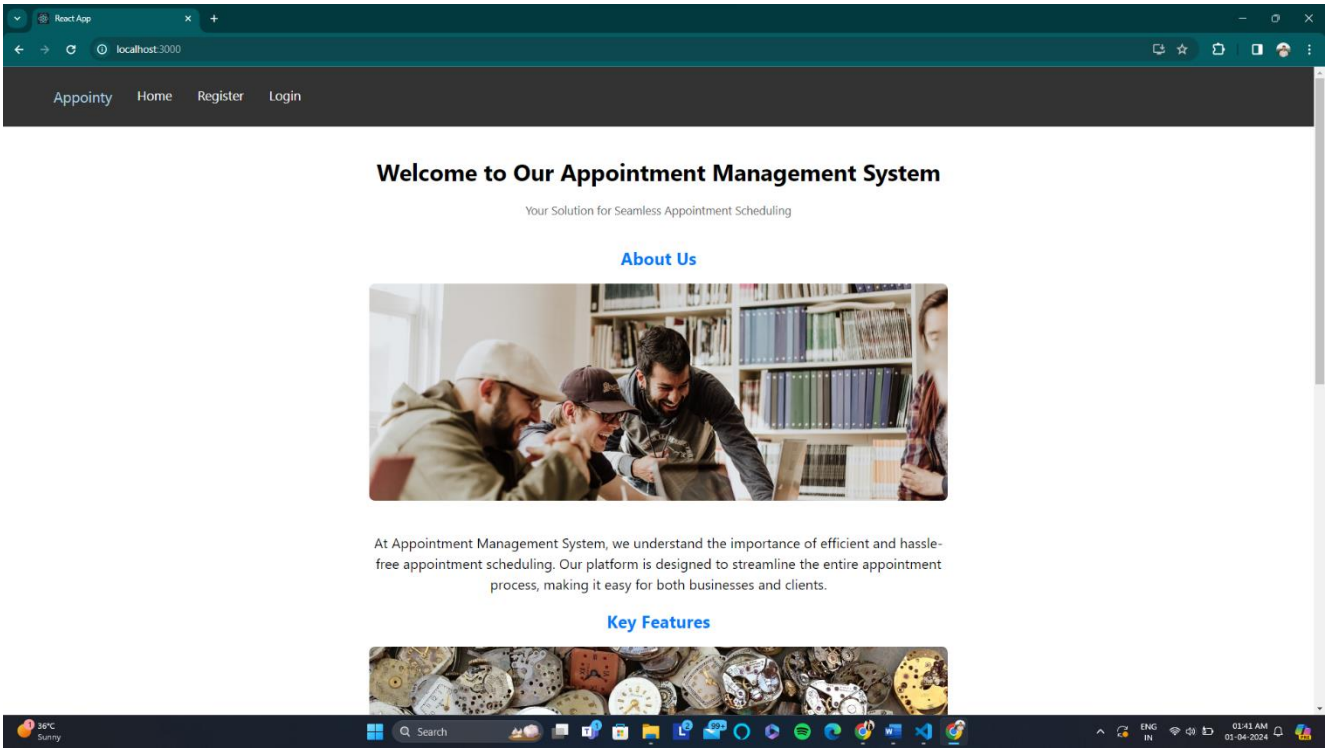
2. **Service Provider Management:**
    - **Description:** Handles service provider accounts and profiles. Service providers can log in, view appointment details, update availability, and manage service offerings.
3. **Appointment Booking:**
    - **Description:** Enables users to schedule appointments for various services. Users can search for available services, select desired appointments, provide necessary details, and choose payment options.
4. **Notification System:**
    - **Description**: Sends notifications to users and service providers regarding appointment confirmations, cancellations, and other relevant updates.

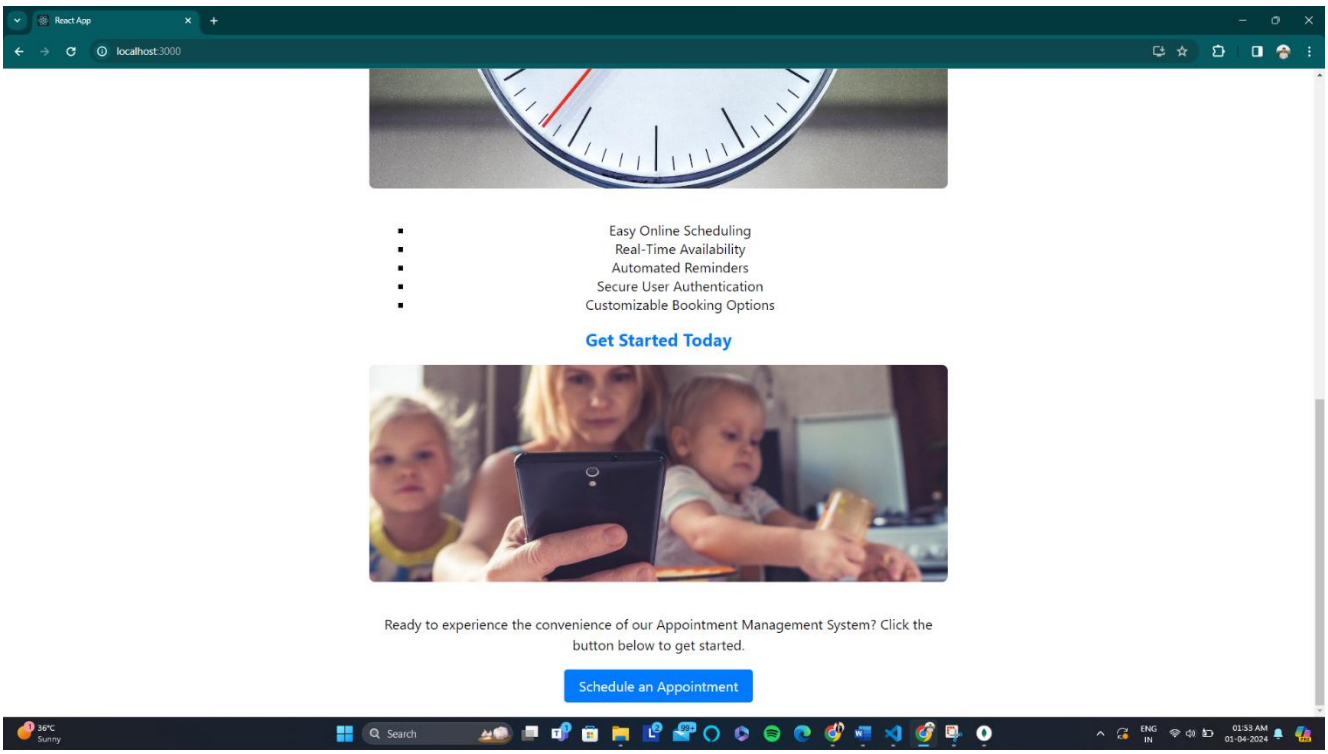
5. Testing

Sr No.	Test Scenario	Expected Result	Actual Result	Status
1	Register With valid details	Users should be able to register successfully.	Success message should be displayed: "Registration successful".	Pass
2	Login As admin	Admin should be able to login.	Success message: “ Admin login successfully”	Pass
3	Add Category	Category was successfully added by the Admin.	Success message: “ Category added successfully.	pass
4	Login as a Service Provider	Service Provider login.	Success message: “login successfully”	Pass
5	Add Service	Service was successfully added by the Service Provider.	Success message: “ Service added successfully.	Pass
6	Login with incorrect credentials	User should not be able to login	Error message: “Invalid credentials!”	Pass
7	Login with correct credentials	User should be able to login	Success message: “login successfully”	Pass
8	View Service and category	All Users should be able to view the Service and category.	Service and category is displayed correctly.	Pass
9	Select Service	All Users should be able to select Service And Book Appointments.	User has selected a service.	Pass
10	Make Payment	All users should be able to make payment.	Payment Successfully	Pass
11	Appointment status pending	Waiting for confirmation by the service provider.	Appointment Accepted.	Pass
12	Appointment Book	Appointment Booking.	Appointment Booking Successfully.	Pass

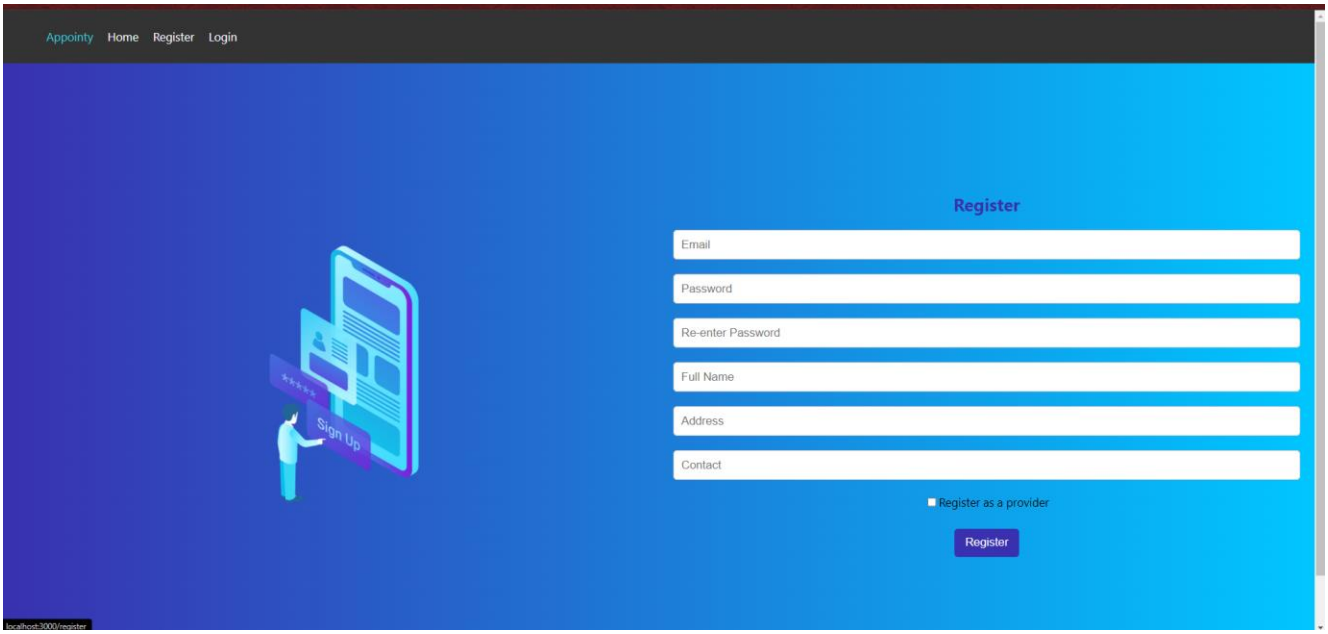
6. Screenshots

Introduction page :

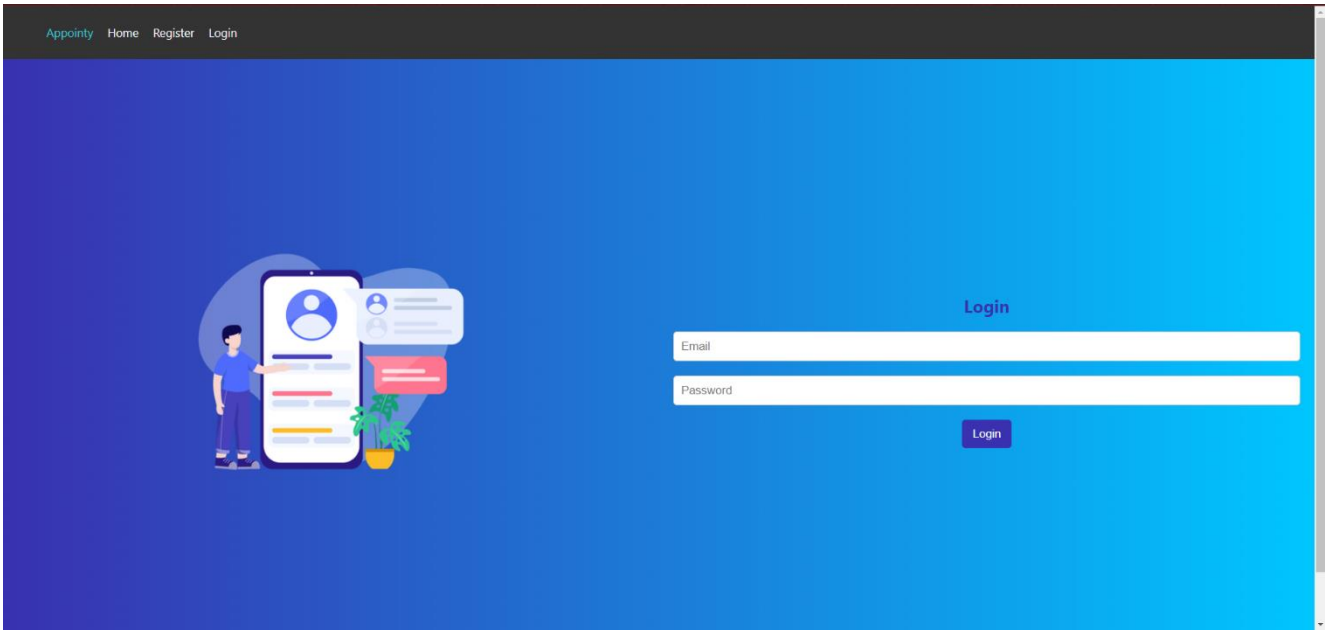




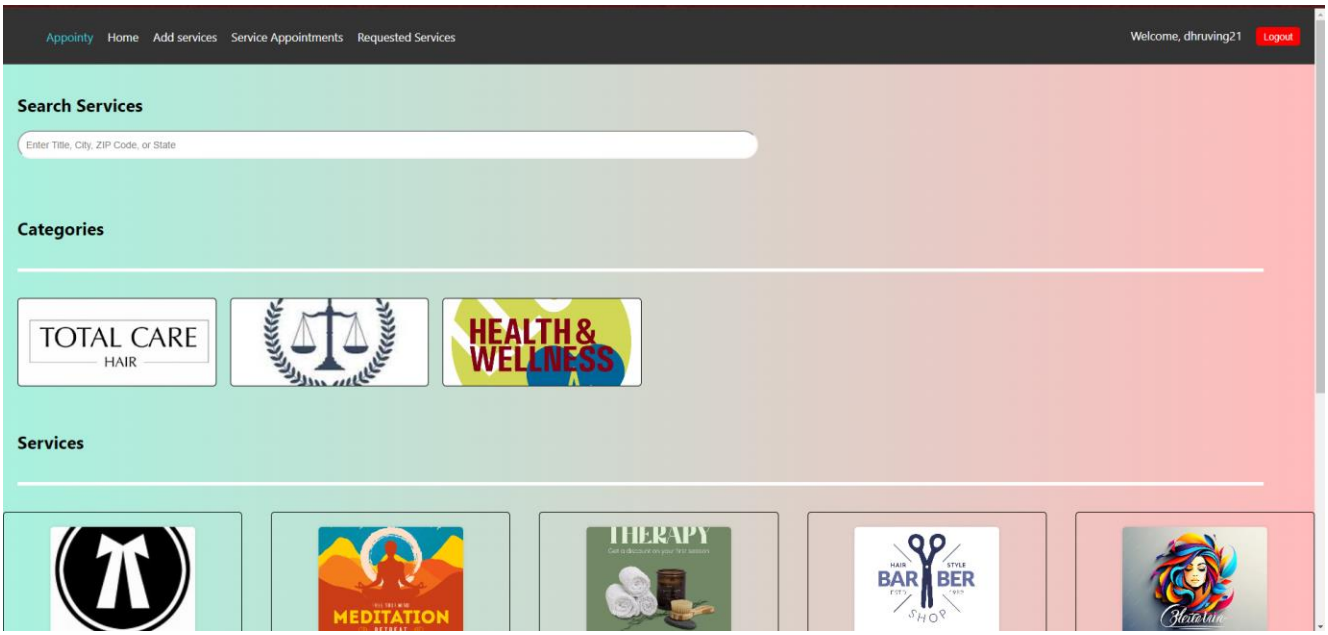
Registration page : user can create an account

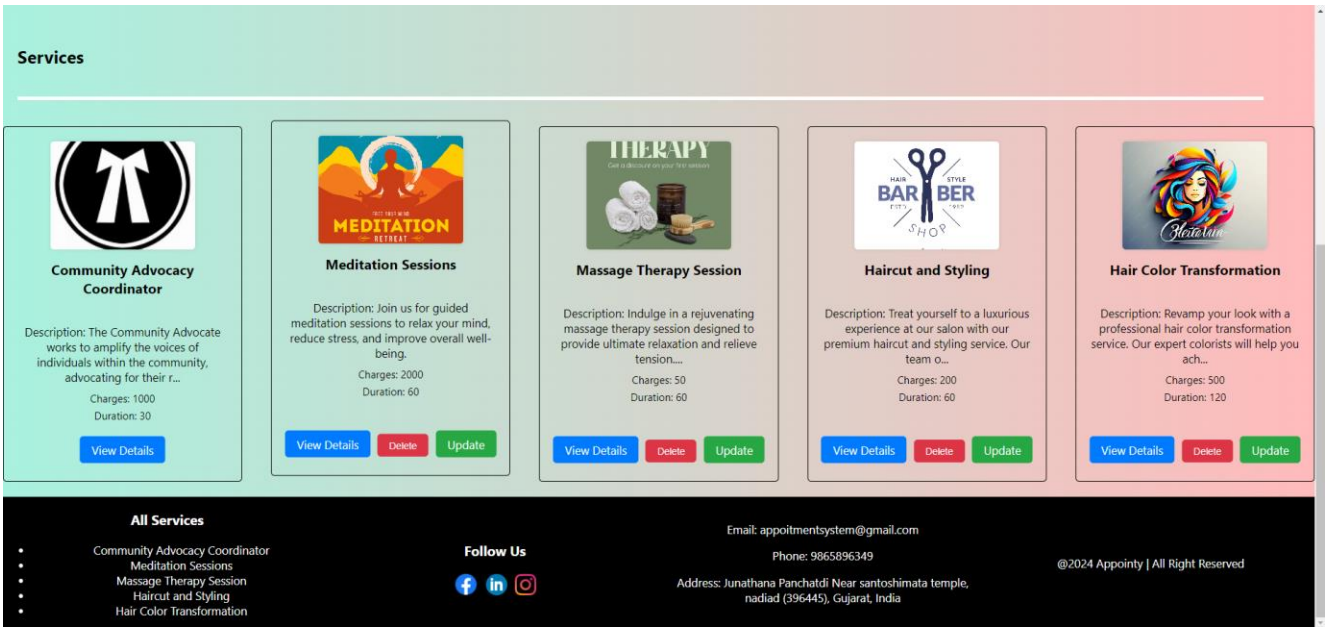


Login page : to access functionality user have to login

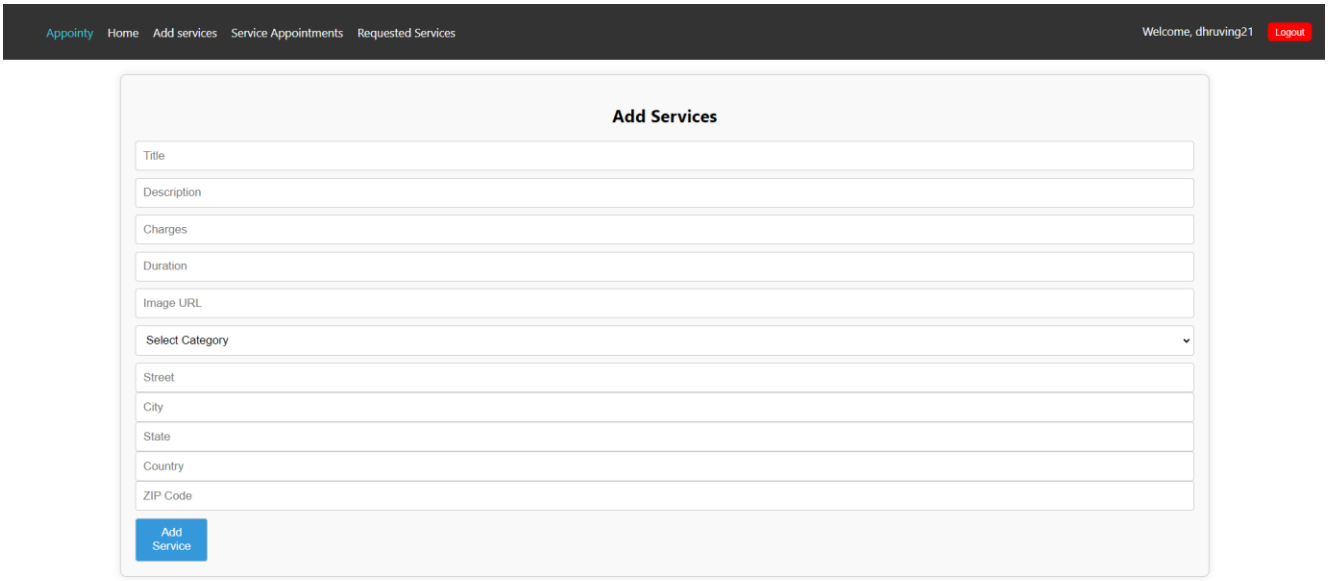


Home page :

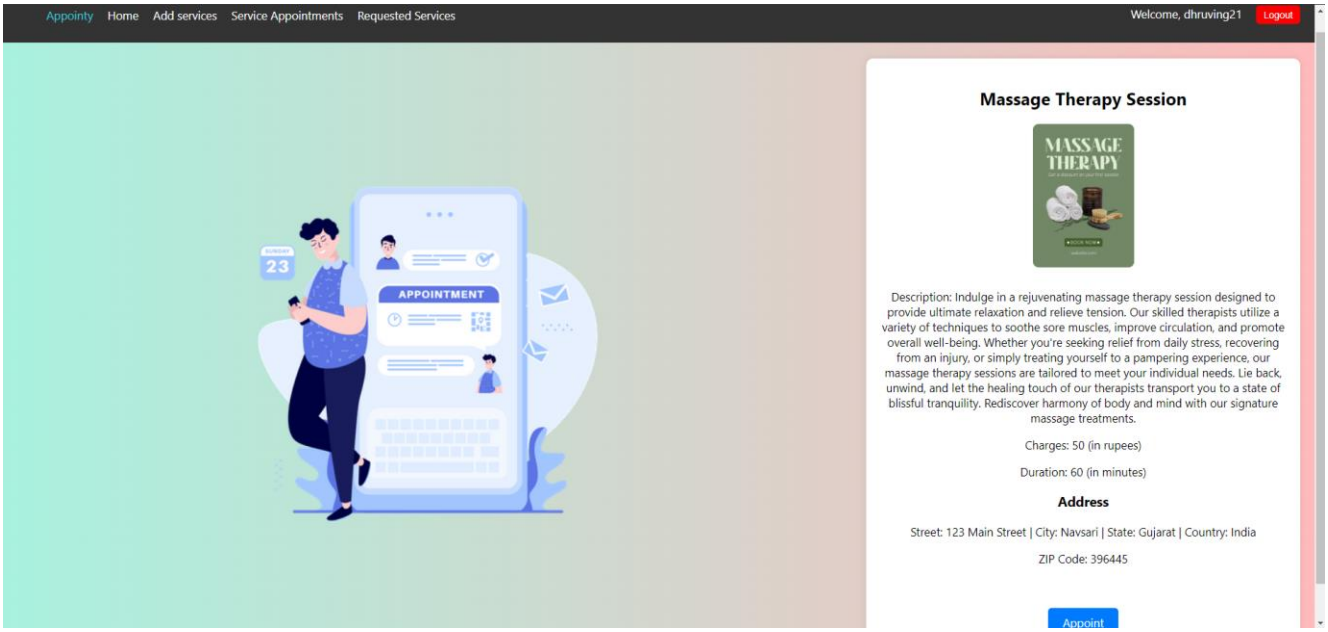

Appointy
Home
Add services
Service Appointments
Requested Services
Welcome, dhruving21
Logout
Search Services
Enter Title, City, ZIP Code, or State
Categories
TOTAL CARE
HAIR
HEALTH &
WELLNESS
Services
MEDITATION
THERAPY
BARBER
SHOP


Services
Community Advocacy Coordinator
Description: The Community Advocate works to amplify the voices of individuals within the community, advocating for their r...
Charges: 1000
Duration: 30
View Details
Meditation Sessions
Description: Join us for guided meditation sessions to relax your mind, reduce stress, and improve overall well-being.
Charges: 2000
Duration: 60
View Details
Delete
Update
Massage Therapy Session
Description: Indulge in a rejuvenating massage therapy session designed to provide ultimate relaxation and relieve tension....
Charges: 50
Duration: 60
Haircut and Styling
Description: Treat yourself to a luxurious experience at our salon with our premium haircut and styling service. Our team o...
Charges: 200
Duration: 60
Hair Color Transformation
Description: Revamp your look with a professional hair color transformation service. Our expert colorists will help you ach...
Charges: 500
Duration: 120
All Services
Community Advocacy Coordinator
Meditation Sessions
Massage Therapy Session
Haircut and Styling
Hair Color Transformation
Follow Us
Email: appoitmentsystem@gmail.com
Phone: 9865896349
Address: Junathana Panchatdi Near santoshimata temple, nadiad (396445), Gujarat, India
@2024 Appointy | All Right Reserved

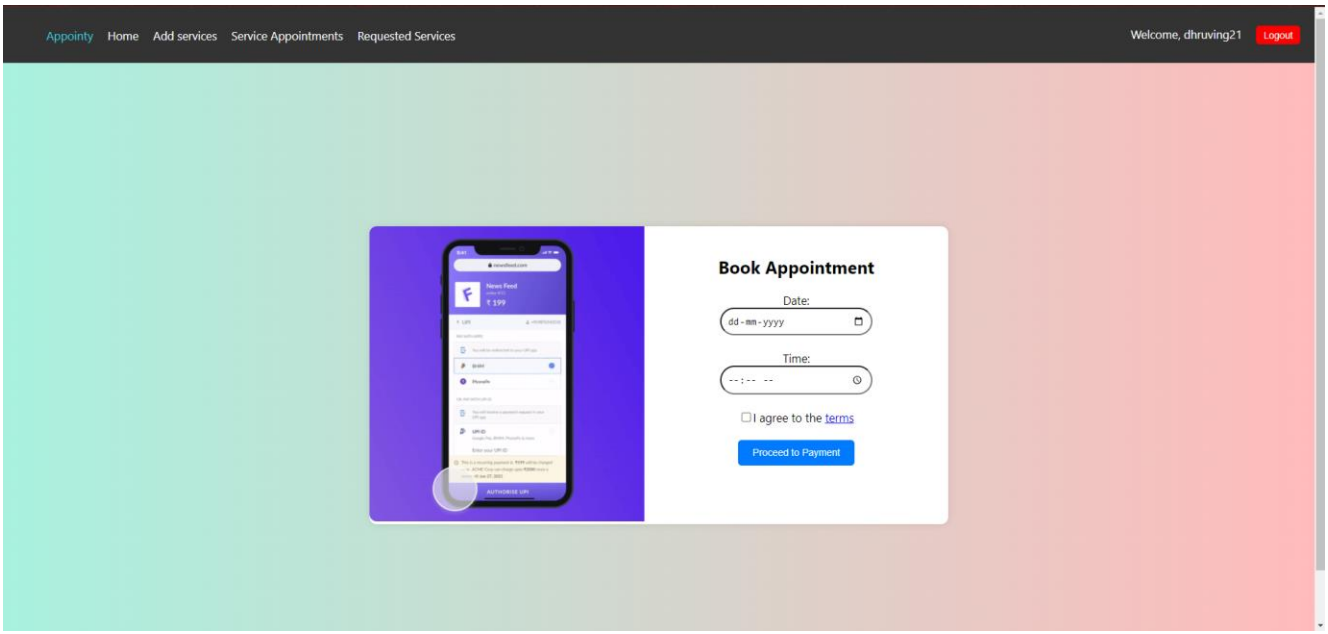
addService : service provider can add his/her services



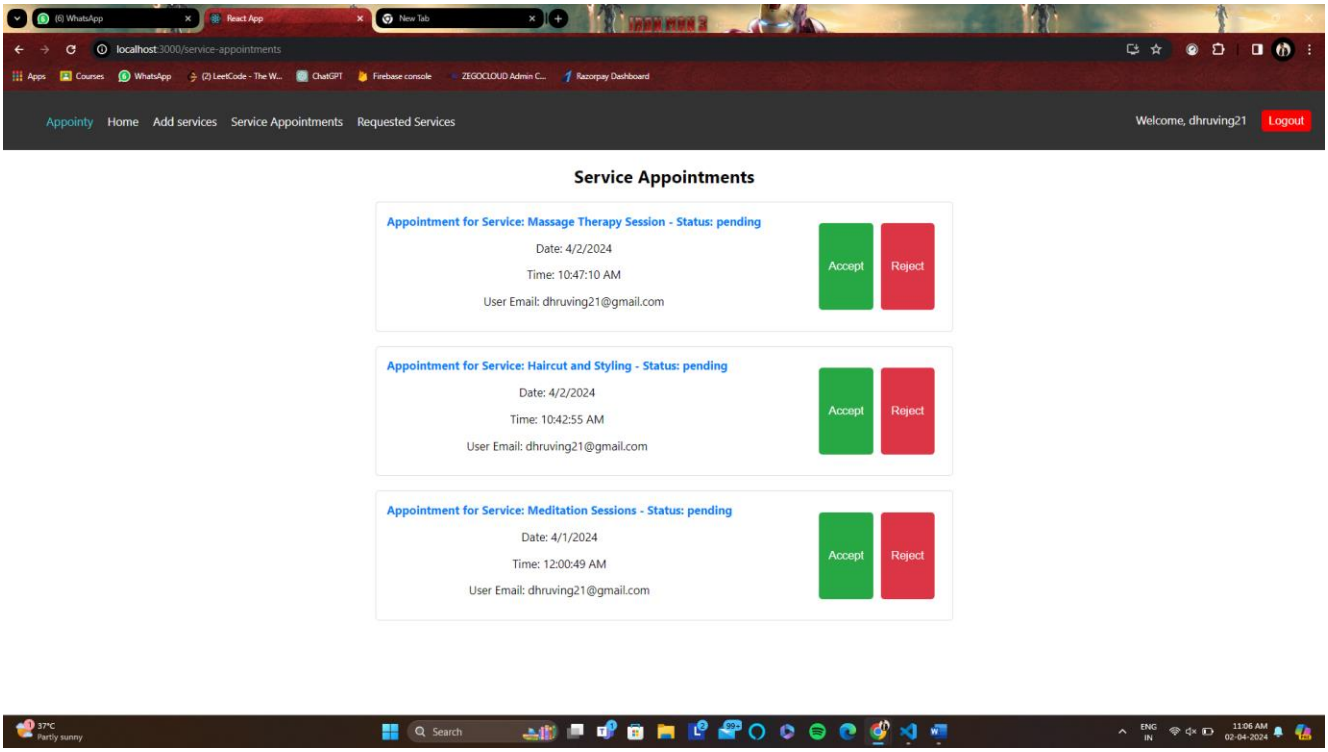
service details :



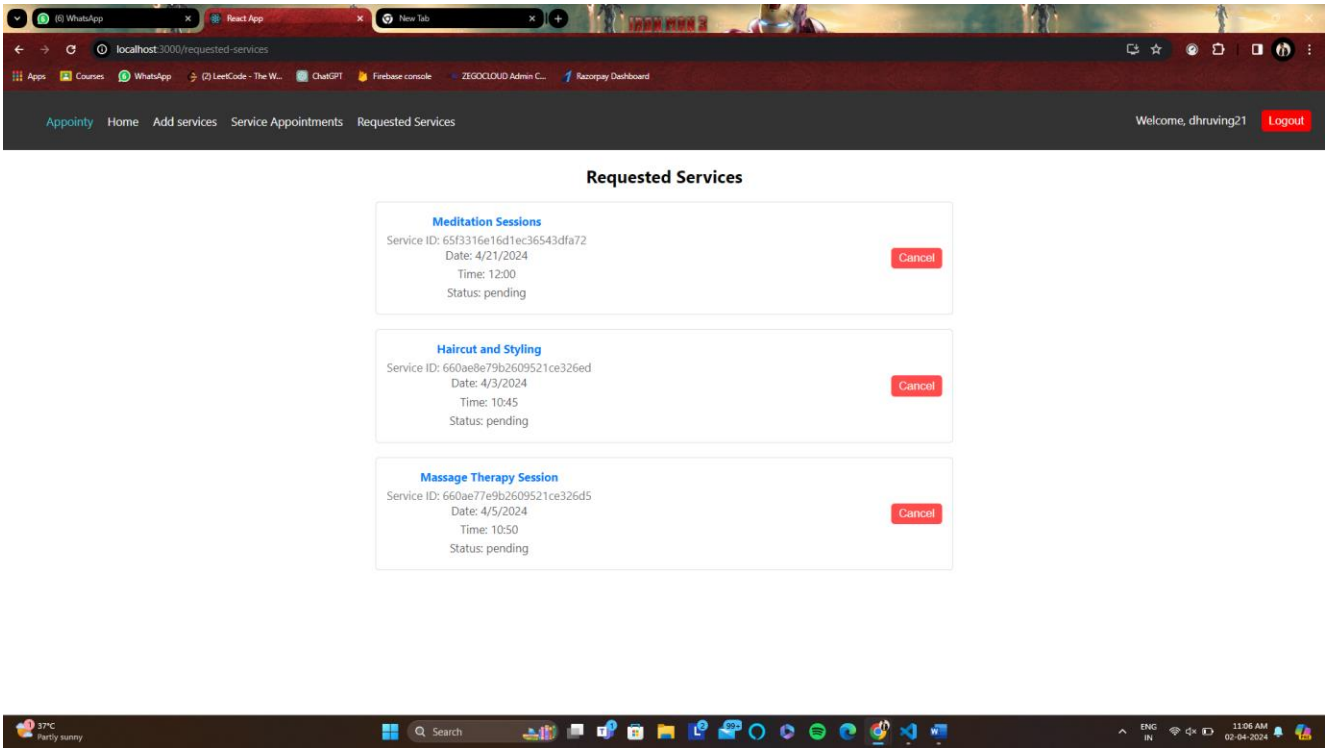
Book Appointment : Select date and time for booking



Service Appointments : for service provider all requests for his/her services will be show hear

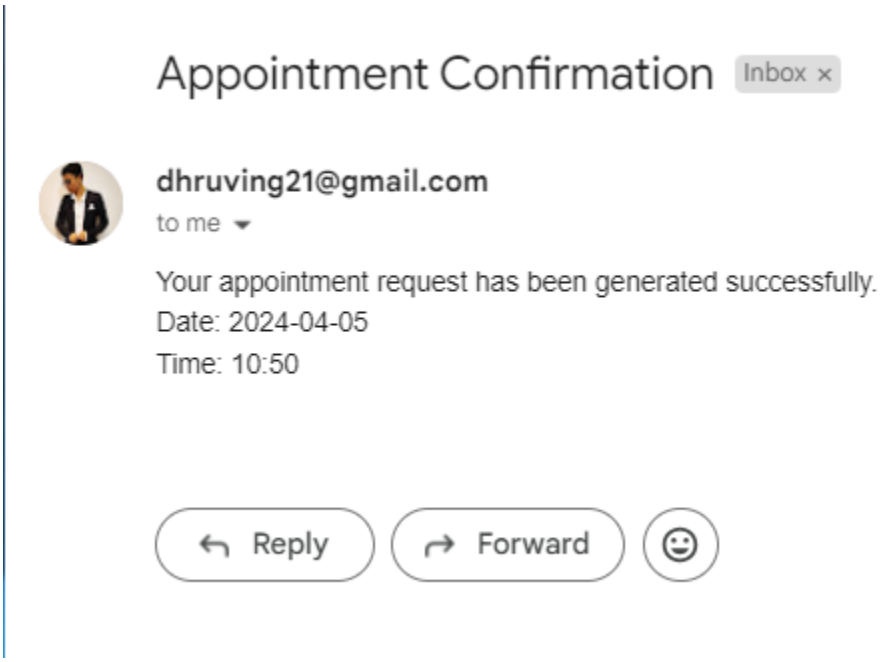


Requested Services : history of requests will be shown.

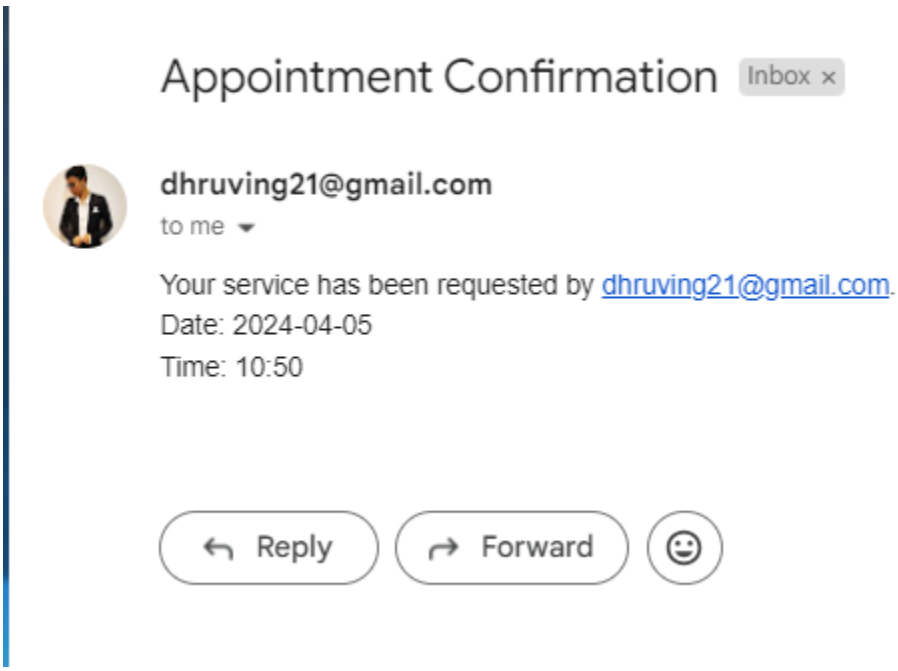


Emails :-

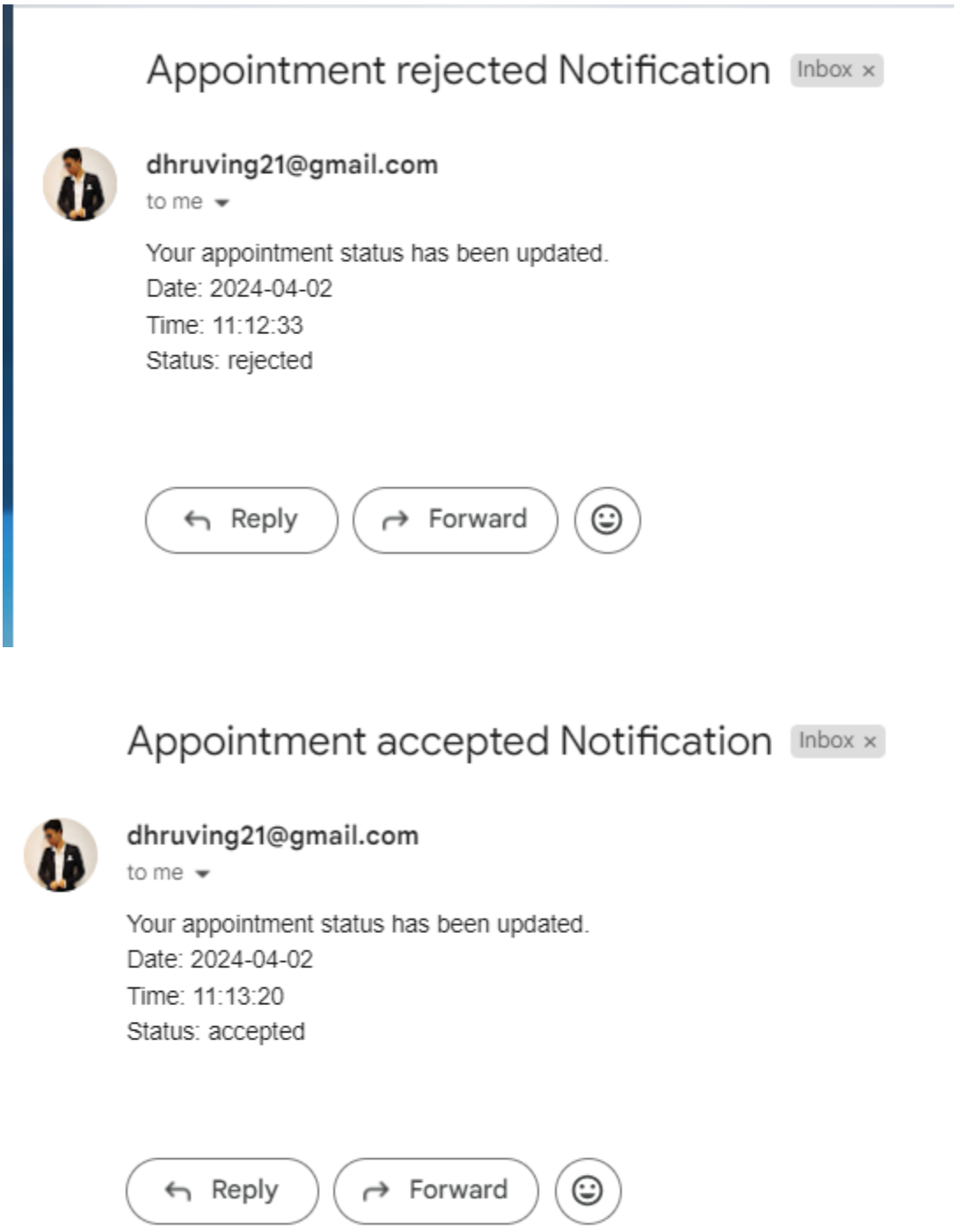
For service requester:



For service provider :



Emails after service provider has accepted/rejected :



7. Conclusion

The Online Appointment Management System (OAMS) serves as a crucial tool in streamlining appointment scheduling processes, enhancing user convenience, and improving service provider efficiency. By offering features such as easy appointment booking, user-friendly interfaces, and effective communication channels, OAMS aims to revolutionize the way appointments are managed in various service industries.

With its implementation, OAMS brings significant benefits to both service providers and users, including reduced wait times, better resource utilization, and improved overall service quality. The system's adoption of modern technologies and frameworks ensures reliability, scalability, and compatibility across different platforms, contributing to its effectiveness in meeting the evolving needs of users and service providers.

Overall, OAMS represents a milestone in optimizing appointment management workflows, fostering better user experiences, and driving operational efficiencies in service-oriented businesses.

8. Limitation and Future Extension

8.1 Limitation

- Dependency on internet connectivity may restrict access in areas with poor network coverage.
- The system may encounter performance issues under high load conditions, impacting response times.
- Limited customization options for specific industry requirements may restrict flexibility.

8.2 Future Extension

- Integration of artificial intelligence (AI) and machine learning (ML) algorithms for intelligent appointment scheduling and optimization.
- Expansion of the system to support additional features such as video consultations, resource management, and automated reminders.
- Development of mobile applications for iOS and Android platforms to provide on-the-go access to appointment management features.
- Incorporation of feedback analysis tools to gain insights from user reviews and enhance service provider performance based on customer feedback.

9. Bibliography

Following links and websites are referred during the development of this project for:-

Sources :

https://react.dev/

https://www.w3schools.com/nodejs/

https://razorpay.com/docs/payments/server-integration/nodejs/payment-gateway/build-integration/

https://www.youtube.com/watch?v=eC73edQ5jmw

https://www.mongodb.com/atlas/database