**APPENDIX**

**APPENDIX A**

**Dialogue transcript of interaction with client (after first meeting)**

Me: Hi sir, good to see you again. After our initial discussion, I have grasped the gist of the issue you are facing and the direction we should be heading in. Now, I would like to dive deeper into the specifics. Could you elaborate more on the challenges you're encountering and what exactly you are looking for in the solution?

Client: Sure thing. So, you see, I've been running my dental clinic for quite some time now, and while I have got the equipment, managing patient records has become a bit of a hassle. I'm still doing everything manually, jotting down details in a book and even mailing myself the pictures I need. It's time-consuming and prone to errors. I want a web app that can handle all this digitally, from patient management to appointments and picture storage.

Me: Got it. Thanks for sharing that. I have a solution in mind that could streamline your clinic operations. Here's what you can expect from the app:

1. You will have the ability to create an account with your own unique credentials.

2. The app will ensure security with two-factor email authentication during signup.

3. You can easily add new patients to the system.

4. All your patients will be neatly displayed on a separate page.

5. You will be able to search and filter patients based on their name, age, or gender.

6. You can add adding additional diagnosis for each patient.

7. You can upload and store patient pictures within the app.

8. These pictures will be displayed as a carousel.

9. Managing appointments will be integrated into the system, allowing you to add, remove, or update appointment times.

10. The app will prevent any overlap in appointment timings, ensuring smooth scheduling.

11. Key metrics like the total number of patients and appointments will be showcased on the dashboard.

12. The dashboard will also highlight appointments for the current day.

Client: That sounds good. Thanks for putting this together.

**APPENDIX C**

main,py

from dental\_clinic import app, db

with app.app\_context():

    db.create\_all()

if \_\_name\_\_ == '\_\_main\_\_':

    app.run(*debug*=True)

dental\_clinic/static/style.css

body {

    overflow-y: hidden;

}

h3{

    font-size: 24px;

}

.sidebar {

    height: 100%;

    width: 300px;

    position: fixed;

    top: 0;

    left: 0px;

    background-color: #0096c8;

    overflow-x: hidden;

    transition: left 0.3s ease;

}

.sidebar-header {

    padding: 20px;

    text-align: center;

    color: #fff;

    margin-bottom: 20px;

}

.nav-links, span {

    list-style-type: none;

    padding: 0 0 0 5px;

}

.nav-links li {

    font-size: 20px;

    padding: 10px;

}

.nav-links a {

    color: #fff;

    text-decoration: none;

    display: flex;

    align-items: center;

    transition: all linear 0.1s;

}

.nav-links a:hover {

    background-color: #028cba;

}

.nav-links a i {

    margin-right: 10px;

}

#menu{

    color: #fff;

    margin-bottom: 10px

}

.submenu, .submenu-2{

    display: none;

    padding: 0px 0;

    padding-left: 20px;

    list-style: none;

}

.submenu-toggle, .submenu-toggle-2{

    background-color: transparent;

    color: #fff;

    border: none;

}

.submenu a, .submenu-2 a{

    font-size: 18px;

    display: block;

}

.submenu li, .submenu-2 li{

    margin-bottom: -10px;

}

.submenu.show, .submenu-2.show {

    display: block;

}

.flash{

    margin-left: 300px;

    margin-bottom: -65px;

}

.alert{

    height: 65px;

    font-size: 18px;

    border-radius: 0%;

}

.container{

    height: 100vh;

}

.row{

    margin-top: 15%;

    margin-left: 260px;

}

.table{

    overflow-y: scroll;

}

#table{

    border: none;

}

#table-heading{

    background-color: transparent;

    margin-top: -10%;

}

#search-box, #search-btn{

    border-radius: 0%;

}

#table-link{

    text-decoration: none;

    color: #fff;

}

#link{

    color: #0096c8;

    text-decoration: underline;

    cursor: pointer;

}

.btn-primary{

    background-color: #0096c8;

    border: 1px solid #0096c8;

}

.btn-primary:hover{

    background-color: #0a83ac;

    border: 1px solid #0a83ac;

    cursor: pointer;

}

#pictures{

    height: 800px;

    width: 110%;

    overflow: none;

    border-radius: 0px;

    margin-top: -5%;

}

#pictures-header{

    color: #fff;

    background-color: #0096c8;

    border: 2px solid #0096c8;

    border-radius: 0%;

}

#pictures-body{

    background-color: #f2f2f2;

}

.prev-icon .fas,

.next-icon .fas {

    color: black;

    font-size: 32px;

}

dental\_clinic/templates/appointments/add.html

{% extends "layout.html" %}

{% block content %}

<div class="row justify-content-md-center">

    <div class="col-md-6">

        <div class="card">

            <div class="card-header">

                <h3 class="text-center">Add Appointment</h3>

            </div>

            <div class="card-body">

                <form method="POST" action="{{ url\_for('add\_appointment') }}">

                    {{ form.hidden\_tag() }}

                    <div class="form-group mb-3">

                        {{ form.patient\_id.label(class="form-label") }}

                        {% if form.patient\_id.errors %}

                            {{ form.patient\_id(class="form-control is-invalid") }}

                            <div class="invalid-feedback">

                                {% for error in form.patient\_id.errors %}

                                    <span>{{ error }}</span>

                                {% endfor %}

                            </div>

                        {% else %}

                            {{ form.patient\_id(class="form-control") }}

                        {% endif %}

                    </div>

                    <div class="form-group mb-3">

                        {{ form.date.label(class="form-label") }}

                        {% if form.date.errors %}

                            {{ form.date(class="form-control is-invalid") }}

                            <div class="invalid-feedback">

                                {% for error in form.date.errors %}

                                    <span>{{ error }}</span>

                                {% endfor %}

                            </div>

                        {% else %}

                            {{ form.date(class="form-control") }}

                        {% endif %}

                    </div>

                    <div class="form-group mb-3">

                        {{ form.start\_time.label(class="form-label") }}

                        {% if form.start\_time.errors %}

                            {{ form.start\_time(class="form-control is-invalid") }}

                            <div class="invalid-feedback">

                                {% for error in form.start\_time.errors %}

                                    <span>{{ error }}</span>

                                {% endfor %}

                            </div>

                        {% else %}

                            {{ form.start\_time(class="form-control") }}

                        {% endif %}

                    </div>

                    <div class="form-group mb-3">

                        {{ form.end\_time.label(class="form-label") }}

                        {% if form.end\_time.errors %}

                            {{ form.end\_time(class="form-control is-invalid") }}

                            <div class="invalid-feedback">

                                {% for error in form.end\_time.errors %}

                                    <span>{{ error }}</span>

                                {% endfor %}

                            </div>

                        {% else %}

                            {{ form.end\_time(class="form-control") }}

                        {% endif %}

                    </div>

                    <button type="submit" class="btn btn-primary btn-block">{{ form.submit.label }}</button>

                </form>

            </div>

        </div>

    </div>

</div>

{% endblock %}

dental\_clinic/templates/appointments/update.html

{% extends "layout.html" %}

{% block content %}

<div class="row justify-content-md-center">

    <div class="col-md-8">

        <div class="card">

            <div class="card-header">

                <h3 class="text-center">Update Appointment</h3>

            </div>

            <div class="card-body">

                <form method="POST" action="">

                    {{ form.hidden\_tag() }}

                    <div class="mb-3">

                        <label for="text" class="form-label">Patient Name</label>

                        <input type="text" class="form-control" value="{{ patient }}" readonly>

                    </div>

                    <br>

                    <div>Previous Date: {{ appointment.date.strftime("%d %B, %Y") }}</div>

                    <div class="mb-3">

                        {{ form.date.label(class="form-label") }}

                        {{ form.date(class="form-control") }}

                        {% for error in form.date.errors %}

                        <div class="invalid-feedback">{{ error }}</div>

                        {% endfor %}

                    </div>

                    <br>

                    <div>Previous Start-Time: {{ appointment.start\_time.strftime("%I:%M %p") }}</div>

                    <div class="mb-3">

                        {{ form.start\_time.label(class="form-label") }}

                        {{ form.start\_time(class="form-control") }}

                        {% for error in form.start\_time.errors %}

                        <div class="invalid-feedback">{{ error }}</div>

                        {% endfor %}

                    </div>

                    <br>

                    <div>Previous End-Time: {{ appointment.end\_time.strftime("%I:%M %p") }}</div>

                    <div class="mb-3">

                        {{ form.end\_time.label(class="form-label") }}

                        {{ form.end\_time(class="form-control") }}

                        {% for error in form.end\_time.errors %}

                        <div class="invalid-feedback">{{ error }}</div>

                        {% endfor %}

                    </div>

                    {{ form.submit(class="btn btn-primary") }}

                </form>

            </div>

        </div>

    </div>

</div>

{% endblock %}

dental\_clinic/templates/appointments/view.html

{% extends "layout.html" %}

{% block content %}

<div class="row justify-content-md-center">

    <div class="card" id="table">

        <div class="card-header" id="table-heading">

            <h3 class="text-center">View Appointments</h3>

        </div>

        <div class="card-body">

            <form method="GET" action="{{ url\_for('view\_appointments') }}">

            <div class="input-group mb-3">

                <input type="text" class="form-control" placeholder="Search by date (YYYY-MM-DD)" id="search-box" name="search" value="{{ search\_term }}">

                <div class="input-group-append">

                <button class="btn btn-outline-secondary" type="submit" id="search-btn">Search</button>

                </div>

            </div>

            </form>

            <table class="table table-striped">

                <thead>

                    <tr>

                        <th scope="col">ID</th>

                        <th scope="col">Patient</th>

                        <th scope="col">Date</th>

                        <th scope="col">Timing</th>

                        <th scope="col">Update (?)</th>

                    </tr>

                </thead>

                <tbody>

                    {% for appointment in appointments %}

                    <tr>

                        <td>{{ appointment.id }}</td>

                        <td>{{ Patient.query.get\_or\_404(appointment.patient\_id).name }}</td>

                        <td>{{ appointment.date.strftime("%d %B, %Y") }}</td>

                        <td>

                            {{ appointment.start\_time.strftime("%I:%M %p") }} to {{ appointment.end\_time.strftime("%I:%M %p") }}

                        </td>

                        <td>

                            <button class="btn btn-primary btn-sm">

                                <a id="table-link" href="{{ url\_for('update\_appointment', appointment\_id=appointment.id) }}">

                                    Update Appointment

                                </a>

                            </button>

                        </td>

                    </tr>

                    {% endfor %}

                </tbody>

            </table>

        </div>

    </div>

</div>

{% endblock content %}

dental\_clinic/templates/patients/pictures/add.html

{% extends "layout.html" %}

{% block content %}

<div class="row justify-content-md-center">

    <div class="col-md-6">

        <div class="card">

            <div class="card-header">

                <h3 class="text-center">Add Picture</h3>

            </div>

            <div class="card-body">

                <form method="POST" action="{{ url\_for('add\_picture') }}" enctype="multipart/form-data">

                    {{ form.hidden\_tag() }}

                    <div class="form-group mb-3">

                        {{ form.patient\_id.label(class="form-label") }}

                        {% if form.patient\_id.errors %}

                            {{ form.patient\_id(class="form-control is-invalid") }}

                            <div class="invalid-feedback">

                                {% for error in form.patient\_id.errors %}

                                    <span>{{ error }}</span>

                                {% endfor %}

                            </div>

                        {% else %}

                            {{ form.patient\_id(class="form-control") }}

                        {% endif %}

                    </div>

                    <div class="form-group mb-3">

                        {{ form.picture.label(class="form-label") }}

                        {% if form.picture.errors %}

                            {{ form.picture(class="form-control is-invalid") }}

                            <div class="invalid-feedback">

                                {% for error in form.picture.errors %}

                                    <span>{{ error }}</span>

                                {% endfor %}

                            </div>

                        {% else %}

                            {{ form.picture(class="form-control") }}

                        {% endif %}

                    </div>

                    <button type="submit" class="btn btn-primary btn-block">{{ form.submit.label }}</button>

                </form>

            </div>

        </div>

    </div>

</div>

{% endblock %}

dental\_clinic/templates/patients/pictures/view.html

{% extends "layout.html" %}

{% block content %}

<div class="row justify-content-md-center">

    <div class="col-md-12">

        <div class="card" id="pictures">

            <div class="card-header" id="pictures-header">

                <h3 class="text-center">View pictures for {{ patient.name }}</h3>

            </div>

            <div class="card-body" id="pictures-body">

                <button class="carousel-control-prev" type="button" data-bs-target="#carouselExampleIndicators" data-bs-slide="prev">

                    <span class="prev-icon" aria-hidden="true"><i class="fas fa-chevron-left"></i></span>

                    <span class="visually-hidden">Previous</span>

                </button>

                <button class="carousel-control-next" type="button" data-bs-target="#carouselExampleIndicators" data-bs-slide="next">

                    <span class="next-icon" aria-hidden="true"><i class="fas fa-chevron-right"></i></span>

                    <span class="visually-hidden">Next</span>

                </button>

                <div id="carouselExampleIndicators" class="carousel slide d-flex align-items-center" data-bs-ride="carousel">

                    <div class="carousel-indicators">

                        {% for picture in pictures %}

                            <button style="background-color: black" type="button" data-bs-target="#carouselExampleIndicators" data-bs-slide-to="{{ loop.index0 }}" {% if loop.first %} class="active" {% endif %} aria-current="true" aria-label="Slide {{ loop.index }}"></button>

                        {% endfor %}

                    </div>

                    <div class="carousel-inner d-flex align-items-center" style="height: 710px;">

                        {% for picture in pictures %}

                            <div class="carousel-item {% if loop.first %} active {% endif %}">

                                <img src="{{ url\_for('retrieve\_picture', picture\_id=picture.id) }}" class="d-block w-100" alt="Slide {{ loop.index }}">

                            </div>

                        {% endfor %}

                    </div>

                </div>

            </div>

        </div>

    </div>

</div>

{% endblock content %}

dental\_clinic/templates/patients/add.html

{% extends "layout.html" %}

{% block content %}

<div class="row justify-content-md-center">

    <div class="col-md-6">

        <div class="card">

            <div class="card-header">

                <h3 class="text-center">Add Patient</h3>

            </div>

            <div class="card-body">

                <form method="POST" action="{{ url\_for('add\_patient') }}">

                    {{ form.hidden\_tag() }}

                    <div class="form-group mb-3">

                        {{ form.name.label(class="form-label") }}

                        {% if form.name.errors %}

                            {{ form.name(class="form-control is-invalid") }}

                            <div class="invalid-feedback">

                                {% for error in form.name.errors %}

                                    <span>{{ error }}</span>

                                {% endfor %}

                            </div>

                        {% else %}

                            {{ form.name(class="form-control") }}

                        {% endif %}

                    </div>

                    <div class="form-group mb-3">

                        {{ form.gender.label(class="form-label") }}

                        {% if form.gender.errors %}

                            {{ form.gender(class="form-control is-invalid") }}

                            <div class="invalid-feedback">

                                {% for error in form.gender.errors %}

                                    <span>{{ error }}</span>

                                {% endfor %}

                            </div>

                        {% else %}

                            {{ form.gender(class="form-control") }}

                        {% endif %}

                    </div>

                    <div class="form-group mb-3">

                        {{ form.age.label(class="form-label") }}

                        {% if form.age.errors %}

                            {{ form.age(class="form-control is-invalid") }}

                            <div class="invalid-feedback">

                                {% for error in form.age.errors %}

                                    <span>{{ error }}</span>

                                {% endfor %}

                            </div>

                        {% else %}

                            {{ form.age(class="form-control") }}

                        {% endif %}

                    </div>

                    <div class="form-group mb-3" id="diagnosisField" style="display: none;">

                        {{ form.diagnosis.label(class="form-label") }}

                        {% if form.diagnosis.errors %}

                            {{ form.diagnosis(class="form-control is-invalid") }}

                            <div class="invalid-feedback">

                                {% for error in form.diagnosis.errors %}

                                    <span>{{ error }}</span>

                                {% endfor %}

                            </div>

                        {% else %}

                            {{ form.diagnosis(class="form-control") }}

                        {% endif %}

                    </div>

                    <div class="form-label" id="toggle-heading">Diagnosis</div>

                    <button type="button" class="btn btn-secondary btn-sm" id="toggleDiagnosis">Add Diagnosis (Optional)</button>

                    <br>

                    <br>

                    <button type="submit" class="btn btn-primary btn-block">{{ form.submit.label }}</button>

                </form>

            </div>

        </div>

    </div>

</div>

<script>

    document.addEventListener('DOMContentLoaded', function() {

        const diagnosisField = document.querySelector('#diagnosisField');

        const toggleButton = document.querySelector('#toggleDiagnosis');

        const toggleHeading = document.querySelector('#toggle-heading');

        toggleButton.addEventListener('click', function() {

            if (diagnosisField.style.display === 'none' || diagnosisField.style.display === '') {

                diagnosisField.style.display = 'block';

                toggleHeading.style.display = 'none';

                toggleButton.textContent = 'Remove Diagnosis';

            } else {

                diagnosisField.style.display = 'none';

                toggleButton.textContent = 'Add Diagnosis (Optional)';

                toggleHeading.style.display = 'block';

                document.querySelector('#diagnosis').value = '';

            }

        });

    });

</script>

{% endblock %}

dental\_clinic/templates/patients/diagnosis.html

{% extends "layout.html" %}

{% block content %}

<div class="row justify-content-md-center">

    <div class="col-md-8">

        <div class="card">

            <div class="card-header">

                <h3 class="text-center">Update Diagnosis</h3>

            </div>

            <div class="card-body">

                <form method="POST" action="">

                    {{ form.hidden\_tag() }}

                    <div class="mb-3">

                        <label for="text" class="form-label">Patient Name</label>

                        <input type="text" class="form-control" value="{{ patient\_info.name }}" readonly>

                    </div>

                    <div class="mb-3">

                        <label for="text" class="form-label">Patient Gender</label>

                        <input type="text" class="form-control" value="{{ patient\_info.gender }}" readonly>

                    </div>

                    <div class="mb-3">

                        <label for="text" class="form-label">Patient Age</label>

                        <input type="text" class="form-control" value="{{ patient\_info.age }}" readonly>

                    </div>

                    <div class="mb-3">

                        {{ form.diagnosis.label(class="form-label") }}

                        {{ form.diagnosis(class="form-control") }}

                        {% for error in form.diagnosis.errors %}

                        <div class="invalid-feedback">{{ error }}</div>

                        {% endfor %}

                    </div>

                    {{ form.submit(class="btn btn-primary") }}

                </form>

            </div>

        </div>

    </div>

</div>

{% endblock %}

dental\_clinic/templates/patients/view.html

{% extends "layout.html" %}

{% block content %}

<div class="row justify-content-md-center">

    <div class="card" id="table">

        <div class="card-header" id="table-heading">

            <h3 class="text-center">View Patients</h3>

        </div>

        <div class="card-body">

            <form method="GET" action="{{ url\_for('view\_patients') }}">

            <div class="input-group mb-3">

                <input type="text" class="form-control" placeholder="Search by name" id="search-box" name="search" value="{{ search\_term }}">

                <div class="input-group-append">

                <button class="btn btn-outline-secondary" type="submit" id="search-btn">Search</button>

                </div>

            </div>

            </form>

            <table class="table table-striped">

                <thead>

                    <tr>

                        <th scope="col">ID</th>

                        <th scope="col">Name</th>

                        <th scope="col">Diagnosis</th>

                        <th scope="col">Update (?)</th>

                        <th scope="col">Pictures</th>

                    </tr>

                </thead>

                <tbody>

                    {% for patient in patients %}

                    <tr>

                        <td>{{ patient.id }}</td>

                        <td>{{ patient.name }}</td>

                        <td>

                            {% if patient.diagnosis == "" %}

                                -

                            {% else %}

                                {{ patient.diagnosis }}

                            {% endif %}

                        </td>

                        <td><button class="btn btn-primary btn-sm"><a id="table-link" href="{{ url\_for('update\_diagnosis', patient\_id=patient.id) }}">Update Diagnosis</a></button></td>

                        <td><button class="btn btn-primary btn-sm"><a id="table-link" href="{{ url\_for('view\_pictures', patient\_id=patient.id) }}">View Pictures</a></button></td>

                    </tr>

                    {% endfor %}

                </tbody>

            </table>

        </div>

    </div>

</div>

{% endblock content %}

dental\_clinic/templates/dashboard.html

{% extends "layout.html" %}

{% block content %}

<div class="container mt-4">

    <div class="row">

        <div class="col-md-3">

            <div class="card text-center mb-4">

                <div class="card-header">

                    <h5 class="text-center">Total Patients</h5>

                </div>

                <div class="card-body">

                    <i class="fas fa-user fa-3x mb-2"></i>

                    <h4 class="card-title">{{ num\_patients }}</h4>

                </div>

            </div>

            <div class="card text-center">

                <div class="card-header">

                    <h5 class="text-center">Total Appointments</h5>

                </div>

                <div class="card-body">

                    <i class="far fa-calendar-alt fa-3x mb-2"></i>

                    <h4 class="card-title">{{ num\_appointments }}</h4>

                </div>

            </div>

        </div>

        <div class="col-md-9">

            <div class="card">

                <div class="card-header">

                    <h3 class="text-center">Today's Appointments</h3>

                </div>

                <div class="card-body">

                    <div class="list-group" style="border-radius: 0px;">

                        {% for appointment in appointments\_for\_today %}

                            <div class="list-group-item" style="font-size: 18px;">

                                <div class="d-flex justify-content-between align-items-center">

                                    <div>

                                        <h5>Appointment for patient: <span style="font-weight: bold; padding-left: 0px;">{{ appointment.patient.name }}</span></h5>

                                        <p class="mb-0"><strong>Date:</strong> {{ appointment.date.strftime("%d %B, %Y") }}</p>

                                        <p style="display: none;" class="mb-0" id="time{{ loop.index }}"><span style="font-weight: bold; padding-left: 0px;">Timing: </span>{{ appointment.start\_time.strftime("%I:%M %p") }} to {{ appointment.end\_time.strftime("%I:%M %p") }}</p>

                                    </div>

                                    <button type="button" class="btn btn-primary btn-sm" onclick="toggleTime('{{ loop.index }}')">Details</button>

                                </div>

                            </div>

                        {% endfor %}

                    </div>

                </div>

            </div>

        </div>

    </div>

</div>

<script>

    function toggleTime(*index*) {

        var time = document.getElementById('time' + index);

        if (time.style.display === 'none') {

            time.style.display = 'block';

        } else {

            time.style.display = 'none';

        }

    }

</script>

{% endblock %}

dental\_clinic/templates/layout.html

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-GLhlTQ8iRABdZLl6O3oVMWSktQOp6b7In1Zl3/Jr59b6EGGoI1aFkw7cmDA6j6gD" crossorigin="anonymous">

    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.3/css/all.min.css">

    <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>

    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/js/bootstrap.bundle.min.js"></script>

    <link rel="stylesheet" type="text/css" href="{{ url\_for('static', filename='style.css') }}">

    <title>Dental Clinic</title>

</head>

<body>

    <header>

        <nav class="sidebar">

            <div class="sidebar-header mt-4">

                <h3>Dental Clinic</h3>

            </div>

            <ul class="nav-links">

                {% if current\_user.is\_authenticated %}

                <li><a href="{{ url\_for('home') }}"><i class="fas fa-home"></i> Dashboard</a></li>

                <li>

                    <i class="fas fa-users" id="menu"></i><span style="color: #fff"> Manage Patients</span><button class="submenu-toggle" data-target="submenu"><i class="fas fa-caret-down"></i></button>

                    <ul class="submenu">

                        <li><a href="{{ url\_for('view\_patients') }}">View Patients</a></li>

                        <li><a href="{{ url\_for('add\_patient') }}">Add Patient</a></li>

                        <li><a href="{{ url\_for('add\_picture') }}">Add Pictures</a></li>

                    </ul>

                </li>

                <li>

                    <i class="fas fa-calendar-check" id="menu"></i><span style="color: #fff"> Manage Appointments</span><button class="submenu-toggle" data-target="submenu-2"><i class="fas fa-caret-down"></i></button>

                    <ul class="submenu-2">

                        <li><a href="{{ url\_for('view\_appointments') }}">View Appointments</a></li>

                        <li><a href="{{ url\_for('add\_appointment') }}">Add Appointments</a></li>

                    </ul>

                </li>

                <li><a href="{{ url\_for('signout') }}"><i class="fas fa-sign-out-alt"></i> Sign Out</a></li>

                {% else %}

                <li><a href="{{ url\_for('home') }}"><i class="fas fa-home"></i> Home Page</a></li>

                <li><a href="{{ url\_for('signin') }}"><i class="fas fa-sign-in-alt"></i> Sign In</a></li>

                <li><a href="{{ url\_for('signup') }}"><i class="fas fa-user-plus"></i> Sign Up</a></li>

                {% endif %}

            </ul>

        </nav>

    </header>

    <div class="flash">

        {% with messages = get\_flashed\_messages(with\_categories=true) %}

        {% if messages %}

            {% for category, message in messages %}

            <div class="alert alert-{{ category }}">

                {{ message }}

            </div>

            {% endfor %}

        {% endif %}

        {% endwith %}

    </div>

    <main role="main" class="container">

        {% block content %}{% endblock %}

    </main>

    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/js/bootstrap.bundle.min.js" integrity="sha384-w76AqPfDkMBDXo30jS1Sgez6pr3x5MlQ1ZAGC+nuZB+EYdgRZgiwxhTBTkF7CXvN" crossorigin="anonymous"></script>

    <script>

        document.addEventListener("DOMContentLoaded", function() {

            const submenuToggles = document.querySelectorAll('.submenu-toggle');

            submenuToggles.forEach(function(*toggle*) {

                toggle.addEventListener('click', function() {

                    const target = toggle.getAttribute('data-target');

                    const submenu = document.querySelector('.' + target);

                    submenu.classList.toggle('show');

                    if (submenu.classList.contains('show')) {

                        localStorage.setItem(target, 'show');

                    } else {

                        localStorage.setItem(target, 'hide');

                    }

                });

            });

            submenuToggles.forEach(function(*toggle*) {

                const target = toggle.getAttribute('data-target');

                const submenuState = localStorage.getItem(target);

                const submenu = document.querySelector('.' + target);

                if (submenuState === 'show') {

                    submenu.classList.add('show');

                }

            });

        });

    </script>

</body>

</html>

dental\_clinic/templates/signin.html

{% extends "layout.html" %}

{% block content %}

<div class="row justify-content-md-center">

    <div class="col-md-6">

        <div class="card">

            <div class="card-header">

                <h3 class="text-center">Sign In</h3>

            </div>

            <div class="card-body">

                <form method="POST" action="{{ url\_for('signin') }}">

                    {{ form.hidden\_tag() }}

                    <div class="form-group mb-3">

                        {{ form.email.label(class="form-label") }}

                        {% if form.email.errors %}

                            {{ form.email(class="form-control is-invalid") }}

                            <div class="invalid-feedback">

                                {% for error in form.email.errors %}

                                    <span>{{ error }}</span>

                                {% endfor %}

                            </div>

                        {% else %}

                            {{ form.email(class="form-control") }}

                        {% endif %}

                    </div>

                    <div class="form-group mb-3">

                        {{ form.password.label(class="form-label") }}

                        {% if form.password.errors %}

                            {{ form.password(class="form-control is-invalid") }}

                            <div class="invalid-feedback">

                                {% for error in form.password.errors %}

                                    <span>{{ error }}</span>

                                {% endfor %}

                            </div>

                        {% else %}

                            {{ form.password(class="form-control") }}

                        {% endif %}

                    </div>

                    <button type="submit" class="btn btn-primary btn-block">{{ form.submit.label }}</button>

                </form>

            </div>

        </div>

    </div>

</div>

{% endblock %}

dental\_clinic/templates/signup.html

{% extends "layout.html" %}

{% block content %}

<div class="row justify-content-md-center">

    <div class="col-md-6">

        <div class="card">

            <div class="card-header">

                <h3 class="text-center">Sign Up</h3>

            </div>

            {% if not valid %}

                <div id="info-form">

                    <div class="card-body">

                        <form method="POST" action="{{ url\_for('signup') }}">

                            {{ form.hidden\_tag() }}

                            <div class="form-group mb-3">

                                {{ form.username.label(class="form-label") }}

                                {% if form.username.errors %}

                                    {{ form.username(class="form-control is-invalid") }}

                                    <div class="invalid-feedback">

                                        {% for error in form.username.errors %}

                                            <span>{{ error }}</span>

                                        {% endfor %}

                                    </div>

                                {% else %}

                                    {{ form.username(class="form-control") }}

                                {% endif %}

                            </div>

                            <div class="form-group mb-3">

                                {{ form.email.label(class="form-label") }}

                                {% if form.email.errors %}

                                    {{ form.email(class="form-control is-invalid") }}

                                    <div class="invalid-feedback">

                                        {% for error in form.email.errors %}

                                            <span>{{ error }}</span>

                                        {% endfor %}

                                    </div>

                                {% else %}

                                    {{ form.email(class="form-control") }}

                                {% endif %}

                            </div>

                            <div class="form-group mb-3">

                                {{ form.password.label(class="form-label") }}

                                {% if form.password.errors %}

                                    {{ form.password(class="form-control is-invalid") }}

                                    <div class="invalid-feedback">

                                        {% for error in form.password.errors %}

                                            <span>{{ error }}</span>

                                        {% endfor %}

                                    </div>

                                {% else %}

                                    {{ form.password(class="form-control") }}

                                {% endif %}

                            </div>

                            <div class="form-group mb-3">

                                {{ form.confirm\_password.label(class="form-label") }}

                                {% if form.confirm\_password.errors %}

                                    {{ form.confirm\_password(class="form-control is-invalid") }}

                                    <div class="invalid-feedback">

                                        {% for error in form.confirm\_password.errors %}

                                            <span>{{ error }}</span>

                                        {% endfor %}

                                    </div>

                                {% else %}

                                    {{ form.confirm\_password(class="form-control") }}

                                {% endif %}

                            </div>

                            <button type="submit" class="btn btn-primary btn-block">{{ form.submit.label }}</button>

                        </form>

                    </div>

                </div>

            {% elif valid %}

                <div id="otp-form">

                    <div class="card-body">

                        <form method="POST" action="{{ url\_for('check\_otp') }}">

                            {{ form.hidden\_tag() }}

                            <input type="hidden" name="username" value="{{ form.username.data }}">

                            <input type="hidden" name="email" value="{{ form.email.data }}">

                            <input type="hidden" name="password" value="{{ form.password.data }}">

                            <input type="hidden" name="password" value="{{ form.confirm\_password.data }}">

                            <div class="form-group">

                                <label for="otp\_input">Enter OTP</label>

                                    <input type="text" class="form-control" id="otp\_input" name="otp\_input">

                            </div>

                            <br>

                            <button type="submit" class="btn btn-primary">Check OTP</button>

                        </form>

                    </div>

                </div>

            {% endif %}

        </div>

    </div>

</div>

{% endblock %}

dental\_clinic/\_\_init\_\_.py

from flask import Flask

from flask\_sqlalchemy import SQLAlchemy

from flask\_bcrypt import Bcrypt

from flask\_login import LoginManager

from flask\_mail import Mail

app = Flask(\_\_name\_\_)

app.config['SECRET\_KEY'] = "jbsusdkffdhbfi3979938sihjbdad"

app.config['SQLALCHEMY\_DATABASE\_URI'] = 'sqlite:///site.db'

app.config['MAIL\_SERVER'] = 'smtp.gmail.com'

app.config['MAIL\_PORT'] = 465

app.config['MAIL\_USERNAME'] = "dentalclinic123400@gmail.com"

app.config['MAIL\_PASSWORD'] = "wopw smno vgtp tyts"

app.config['MAIL\_USE\_TLS'] = False

app.config['MAIL\_USE\_SSL'] = True

mail = Mail(app)

db = SQLAlchemy(app)

bcrypt = Bcrypt(app)

login\_manager = LoginManager(app)

login\_manager.login\_view = 'signin'

app.app\_context().push()

from dental\_clinic import views

dental\_clinic/auth.py

import random

from flask\_mail import Message

from dental\_clinic import mail

def generate\_otp():

    return ''.join(str(random.randint(1, 6)) for \_ in range(5))

def send\_otp\_email(*user*, *email*, *otp*):

    otp\_email = Message('Authenticate your email', *sender*="mutyamkritin.29@gmail.com", *recipients*=[email])

    otp\_email.body = f"Hello {user},\n\nPlease use this OTP to authenticate your email: {otp}\n\nDo not share the OTP anywhere, also keep in mind that we do not call or message our clients to share the OTP with us.\n\n Warm Regards,\nDental Clinic"

    mail.send(otp\_email)

dental\_clinic/setup.py

from flask\_login import UserMixin

from flask\_wtf import FlaskForm

from flask\_wtf.file import FileField, FileAllowed, FileRequired

from wtforms import (

    StringField, PasswordField, SubmitField, SelectField,

    IntegerField, TextAreaField, DateField, TimeField

)

from wtforms.validators import (

    DataRequired, Length, Email, EqualTo, ValidationError, NumberRange

)

from dental\_clinic import db, login\_manager

@login\_manager.user\_loader

def load\_user(*user\_id*):

    return User.query.get(int(user\_id))

class User(db.Model, UserMixin):

    id = db.Column(db.Integer, *primary\_key*=True)

    patient = db.relationship('Patient', *backref*='author', *lazy*=True)

    username = db.Column(db.String(16), *unique*=True, *nullable*=False)

    email = db.Column(db.String(120), *unique*=True, *nullable*=False)

    password = db.Column(db.String(60), *nullable*=False)

class Patient(db.Model, UserMixin):

    id = db.Column(db.Integer, *primary\_key*=True)

    user\_id = db.Column(db.Integer, db.ForeignKey('user.id'), *nullable*=False)

    name = db.Column(db.String(16), *nullable*=False)

    age = db.Column(db.Integer, *nullable*=False)

    gender = db.Column(db.String(16), *nullable*=False)

    diagnosis = db.Column(db.Text)

    pictures = db.relationship('Picture', *backref*='patient', *lazy*=True)

class Picture(db.Model):

    id = db.Column(db.Integer, *primary\_key*=True)

    patient\_id = db.Column(db.Integer, db.ForeignKey('patient.id'), *nullable*=False)

    image\_data = db.Column(db.LargeBinary, *nullable*=False)

    image\_type = db.Column(db.String(10), *nullable*=False)

class Appointment(db.Model):

    id = db.Column(db.Integer, *primary\_key*=True)

    patient\_id = db.Column(db.Integer, db.ForeignKey('patient.id'), *nullable*=False)

    patient = db.relationship('Patient', *backref*='appointments', *lazy*=True)

    date = db.Column(db.Date, *nullable*=False)

    start\_time = db.Column(db.Time, *nullable*=False)

    end\_time = db.Column(db.Time, *nullable*=False)

class SignUpForm(FlaskForm):

    username = StringField('Enter Username', *validators*=[DataRequired(), Length(*min*=4, *max*=16)])

    email = StringField('Enter Email', *validators*=[DataRequired(), Email()])

    password = PasswordField('Enter Password', *validators*=[DataRequired(), Length(*min*=8, *max*=32)])

    confirm\_password = PasswordField('Confirm Password', *validators*=[DataRequired(), EqualTo('password')])

    submit = SubmitField('Sign Up')

    def validate\_username(*self*, *username*):

        if User.query.filter\_by(*username*=username.data).first():

            raise ValidationError('That username is already taken. Try Again.')

    def validate\_email(*self*, *email*):

        if User.query.filter\_by(*email*=email.data).first():

            raise ValidationError('That email is already taken. Try Again.')

class SignInForm(FlaskForm):

    email = StringField('Enter Email', *validators*=[DataRequired(), Email()])

    password = PasswordField('Enter Password', *validators*=[DataRequired()])

    submit = SubmitField('Sign In')

class AddPatientForm(FlaskForm):

    name = StringField('Full Name', *validators*=[DataRequired(), Length(*min*=2, *max*=50)])

    gender = SelectField('Gender', *choices*=[('male', 'Male'), ('female', 'Female'), ('other', 'Other')],

*validators*=[DataRequired()])

    age = IntegerField('Age', *validators*=[DataRequired(), NumberRange(*min*=0, *max*=150)])

    diagnosis = TextAreaField('Enter Diagnosis')

    submit = SubmitField('Add Patient')

class UpdateDiagnosisForm(FlaskForm):

    diagnosis = TextAreaField('Update Diagnosis', *validators*=[DataRequired()])

    submit = SubmitField('Submit')

class AddPictureForm(FlaskForm):

    def \_\_init\_\_(*self*, \**args*, \*\**kwargs*):

        super(AddPictureForm, self).\_\_init\_\_(\*args, \*\*kwargs)

        self.patient\_id.choices = [(patient.id, patient.name) for patient in Patient.query.all()]

    patient\_id = SelectField('Patient', *validators*=[DataRequired()])

    picture = FileField('Picture', *validators*=[FileRequired(), FileAllowed(['jpg', 'jpeg', 'png'], 'Only images can be uploaded')])

    submit = SubmitField('Submit')

class AppointmentForm(FlaskForm):

    def \_\_init\_\_(*self*, \**args*, \*\**kwargs*):

        super(AppointmentForm, self).\_\_init\_\_(\*args, \*\*kwargs)

        self.patient\_id.choices = [(patient.id, patient.name) for patient in Patient.query.all()]

    patient\_id = SelectField('Patient', *validators*=[DataRequired()])

    date = DateField('Date', *format*='%Y-%m-%d', *validators*=[DataRequired()])

    start\_time = TimeField('Start Time', *format*='%H:%M', *validators*=[DataRequired()])

    end\_time = TimeField('End Time', *format*='%H:%M', *validators*=[DataRequired()])

    submit = SubmitField('Submit')

class UpdateAppointmentForm(FlaskForm):

    date = DateField('Updated Date:', *format*='%Y-%m-%d', *validators*=[DataRequired()])

    start\_time = TimeField('Updated Start Time:', *format*='%H:%M', *validators*=[DataRequired()])

    end\_time = TimeField('Updated End Time:', *format*='%H:%M', *validators*=[DataRequired()])

    submit = SubmitField('Update')

dental\_clinic/views.py

from datetime import datetime, date

from io import BytesIO

import os

from flask import render\_template, url\_for, flash, redirect, request, send\_file, session

from flask\_login import login\_user, current\_user, logout\_user, login\_required

from werkzeug.utils import secure\_filename

from dental\_clinic import app, db, bcrypt

from dental\_clinic.setup import (

    User, Patient, Picture, Appointment, SignUpForm, SignInForm, AddPatientForm,

    UpdateDiagnosisForm, AddPictureForm, AppointmentForm, UpdateAppointmentForm

)

from dental\_clinic.auth import generate\_otp, send\_otp\_email

@app.route('/', *methods*=['POST', 'GET'])

@login\_required

def home():

    num\_patients = len(Patient.query.all())

    num\_appointments = len(Appointment.query.all())

    appointments\_for\_today = Appointment.query.filter\_by(*date*=date.today()).all()

    return render\_template('dashboard.html', *num\_patients*=num\_patients, *num\_appointments*=num\_appointments,

*appointments\_for\_today*=appointments\_for\_today)

@app.route('/patients/view', *methods*=['POST', 'GET'])

@login\_required

def view\_patients():

    patients = Patient.query.all()

    search\_term = request.args.get('search')

    if search\_term:

        patients = Patient.query.filter(Patient.name.ilike(f"%{search\_term}%")).all()

    else:

        search\_term = ""

    return render\_template('patients/view.html', *patients*=patients)

@app.route('/patients/add', *methods*=['POST', 'GET'])

@login\_required

def add\_patient():

    form = AddPatientForm()

    if form.validate\_on\_submit():

        patient = Patient(*name*=form.name.data, *gender*=form.gender.data, *age*=form.age.data, *diagnosis*=form.diagnosis.data, *author*=current\_user)

        db.session.add(patient)

        db.session.commit()

        flash(f'Patient "{form.name.data}" was added', 'success')

        return redirect(url\_for('view\_patients'))

    return render\_template('patients/add.html', *form*=form)

@app.route('/patients/<int:patient\_id>/pictures/view', *methods*=['POST', 'GET'])

@login\_required

def view\_pictures(*patient\_id*):

    patient = Patient.query.get\_or\_404(patient\_id)

    return render\_template('/patients/pictures/view.html', *pictures*=patient.pictures, *patient*=patient)

@app.route('/patients/pictures/add', *methods*=['POST', 'GET'])

@login\_required

def add\_picture():

    form = AddPictureForm()

    if form.validate\_on\_submit():

        print(form.picture.data)

        new\_picture = Picture(*patient\_id*=form.patient\_id.data, *image\_data*=form.picture.data.read(),

*image\_type*=os.path.splitext(secure\_filename(form.picture.data.filename))[1])

        patient = Patient.query.get\_or\_404(form.patient\_id.data).name

        db.session.add(new\_picture)

        db.session.commit()

        flash(f'The image has been successfully added for patient "{patient}"', 'success')

        return redirect(url\_for('view\_pictures', *patient\_id*=form.patient\_id.data))

    return render\_template('/patients/pictures/add.html', *form*=form)

@app.route('/pictures/<int:picture\_id>')

@login\_required

def retrieve\_picture(*picture\_id*):

    picture = Picture.query.get\_or\_404(picture\_id)

    return send\_file(BytesIO(picture.image\_data), *mimetype*=f'image/{picture.image\_type}')

@app.route('/patients/<int:patient\_id>/diagnosis', *methods*=['POST', 'GET'])

@login\_required

def update\_diagnosis(*patient\_id*):

    patient\_info = Patient.query.get\_or\_404(patient\_id)

    form = UpdateDiagnosisForm()

    if form.validate\_on\_submit():

        patient\_info.diagnosis = form.diagnosis.data

        flash(f'Diagnosis for patient "{patient\_info.name}" was updated', 'success')

        db.session.commit()

        return redirect(url\_for('view\_patients'))

    return render\_template('/patients/diagnosis.html', *patient\_info*=patient\_info, *form*=form)

@app.route('/appointments/add', *methods*=['POST', 'GET'])

@login\_required

def add\_appointment():

    form = AppointmentForm()

    if form.validate\_on\_submit():

        patient\_id = form.patient\_id.data

        date = form.date.data

        start\_time = form.start\_time.data

        end\_time = form.end\_time.data

        existing\_appointments = Appointment.query.filter\_by(*date*=date).all()

        for appointment in existing\_appointments:

            appointment\_start = datetime.combine(datetime.today(), appointment.start\_time)

            appointment\_end = datetime.combine(datetime.today(), appointment.end\_time)

            input\_start = datetime.combine(datetime.today(), start\_time)

            input\_end = datetime.combine(datetime.today(), end\_time)

            if input\_start < appointment\_end and input\_end > appointment\_start:

                flash('There is already an appointment scheduled at that time. Please choose another time.', 'danger')

                return redirect(url\_for('add\_appointment'))

        new\_appointment = Appointment(*patient\_id*=patient\_id, *date*=date, *start\_time*=start\_time, *end\_time*=end\_time)

        db.session.add(new\_appointment)

        db.session.commit()

        flash('Appointment created successfully', 'success')

        return redirect(url\_for('view\_appointments'))

    return render\_template('/appointments/add.html', *form*=form)

@app.route('/appointments/view', *methods*=['POST', 'GET'])

@login\_required

def view\_appointments():

    appointments = Appointment.query.all()

    search\_term = request.args.get('search')

    if search\_term:

        appointments = Appointment.query.filter(Appointment.date.ilike(f"%{search\_term}%")).all()

    else:

        search\_term = ""

    return render\_template('/appointments/view.html', *appointments*=appointments, *Patient*=Patient)

@app.route('/appointments/<int:appointment\_id>/update', *methods*=['POST', 'GET'])

@login\_required

def update\_appointment(*appointment\_id*):

    appointment = Appointment.query.get\_or\_404(appointment\_id)

    form = UpdateAppointmentForm()

    patient = Patient.query.get\_or\_404(appointment.patient\_id).name

    if form.validate\_on\_submit():

        existing\_appointments = Appointment.query.filter\_by(*date*=form.date.data).all()

        for exiting\_appointment in existing\_appointments:

            appointment\_start = datetime.combine(datetime.today(), exiting\_appointment.start\_time)

            appointment\_end = datetime.combine(datetime.today(), exiting\_appointment.end\_time)

            input\_start = datetime.combine(datetime.today(), form.start\_time.data)

            input\_end = datetime.combine(datetime.today(), form.end\_time.data)

            if input\_start < appointment\_end and input\_end > appointment\_start:

                flash('There is already an appointment scheduled at that time. Please choose another time.', 'danger')

                return redirect(url\_for('update\_appointment', *appointment\_id*=appointment\_id))

        else:

            appointment.date = form.date.data

            appointment.start\_time = form.start\_time.data

            appointment.end\_time = form.end\_time.data

            db.session.commit()

        flash(f'Appointment for patient "{patient}" was updated', 'success')

        return redirect(url\_for('view\_appointments'))

    return render\_template('/appointments/update.html', *form*=form, *patient*=patient, *appointment*=appointment)

@app.route('/signup', *methods*=['POST', 'GET'])

def signup():

    if current\_user.is\_authenticated:

        return redirect(url\_for('home'))

    form = SignUpForm()

    valid = False

    if form.validate\_on\_submit():

        valid = True

        otp = generate\_otp()

        session['otp'] = otp

        send\_otp\_email(form.username.data, form.email.data, otp)

        flash('Please check your email for the OTP', 'success')

    return render\_template('signup.html', *form*=form, *valid*=valid)

@app.route('/signup/otp', *methods*=['POST', 'GET'])

def check\_otp():

     if 'otp' not in session:

        flash('Invalid or expired OTP. Please try again.', 'danger')

        return redirect(url\_for('signup'))

     if request.form.get('otp\_input') == str(session['otp']):

        hashed\_password = bcrypt.generate\_password\_hash(request.form.get('password')).decode('utf-8')

        user = User(*username*=request.form.get('username'), *email*=request.form.get('email'), *password*=hashed\_password)

        db.session.add(user)

        db.session.commit()

        flash('Your account has been created', 'success')

        session.pop('otp', None)

        return redirect(url\_for('signin'))

     else:

        flash('Invalid OTP. Please try again.', 'danger')

        return redirect(url\_for('signup'))

@app.route('/signin', *methods*=['POST', 'GET'])

def signin():

    if current\_user.is\_authenticated:

        return redirect(url\_for('home'))

    form = SignInForm()

    if form.validate\_on\_submit():

        user = User.query.filter\_by(*email*=form.email.data).first()

        if user and bcrypt.check\_password\_hash(user.password, form.password.data):

            login\_user(user)

            next\_page = request.args.get('next')

            return redirect(next\_page) if next\_page else redirect(url\_for('home'))

        else:

            flash('Login Unsuccessful. Please check email and password', 'danger')

    return render\_template('signin.html', *form*=form)

@app.route("/signout")

def signout():

    logout\_user()

    return redirect(url\_for('signin'))

**APPENDIX E**

**Interview with the client (after testing)**

Me: Hi sir, how is the app working out for you?

Client: It's been pretty good. I'm finding it much easier to manage my patients and appointments now.

Me: That's good to hear! Is there anything you'd like to improve upon?

Client: Well, I was thinking it would be helpful if we could add a payment feature. That way, patients can pay online.

Me: Could you tell me more about how you think this feature should work?

Client: Sure. I think if we could have a simple payment gateway where patients can pay using their credit cards or digital wallets, it would make things more convenient for everyone.

Me: Understood. Besides that, is there anything else you think needs attention?

Client: No, everything else seems to be working fine for now.

Me: Okay, thanks for your input. I will look into adding the payment feature to the app.

**User test cases**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test No. | Test Description | Test Data | Planned Outcome | Actual Outcome | Pass/Fail |
| 1 | Client should be able to create an account with unique credentials. | Enter user details through signup form. | User can sign into their accounts after signing up. |  | Pass |
| 2 | The application must validate user signup with two-factor email authentication. | Email OTP to user’s email and validate signup. | User can input 5-digit OTP send to their email and create an account. |  | Pass |
| 3 | Client should be able to add new patients. | Enter patient details to store a new patient. | Patients are added after user enters their details in the add patient form. |  | Pass |
| 4 | The application must display all patients in a separate page. | Click on View Patients to see the list of existing patients. | All the added patients are visible with their details in the View Patients page. |  | Pass |
| 5 | Client will be able to filter and search for individual patients based on name, age, or gender. | Search for any patient based on the given criteria. | The application filters out the patients based on input by the user. |  | Pass |
| 6 | Client should be able to add additional diagnosis for each patient. | Enter additional diagnosis for patients in the Add Diagnosis and Update Diagnosis form. | Additional diagnosis can be added for all patients which will be displayed. |  | Pass |
| 7 | Client should be able to upload pictures for each patient. | Browse pictures and upload for patients. | Pictures can be uploaded for each patient in the Upload Pictures form. |  | Pass |
| 8 | Client should be able to view pictures for the corresponding patient as a carousel. | Click on View Pictures to see the pictures automatically scroll or scroll manually with arrows. | Uploaded pictures are displayed in the View Pictures page. |  | Pass |
| 9 | Client should be able to add or remove an appointment for the patients. | Enter appointment details in the Add Appointment form. | A new appointment is created after user inputs patient name and appointment timings. |  | Pass |
| 10 | Client should be able to update the appointment time for the patients. | User can choose to update appointment. | User needs to input new timings after which the appointment is updated. |  | Pass |
| 11 | The application must not allow for any overlap in timing among the existing and future appointments. | Appointment timings are checked for overlap. | Appointment is only created if it does not overlap with other appointments. |  | Pass |
| 12 | Display the grand total number of patients and appointments on the dashboard. | Number of patients and appointments are shown. | The dashboard shows the total number of Patients and Appointments accurately. |  | Pass |
| 13 | Display the appointments of the current day on the dashboard | The current’ days appointments are shown, | The dashboard shows the appointments scheduled for that day. |  | Pass |