UM - Internship || WEB DEVELOPMENT || (Full Stack JavaScript, MEAN, MERN) 1 Month

INTERNSHIP ID: UMIP23139

DIGITAL E GRAM PANCHAYAT PROJECT REPORT

ADVANCE PROJECT

Project Title: Digital E Gram Panchayat

Developed By: Deeksha Chauhan

Table of Contents

1. Introduction

The **Digital E Gram Panchayat** is a web-based application designed to digitalize the services provided by gram panchayats, such as birth registration, marriage registration, caste certificates, income certificates, and more. This system aims to simplify the interaction between citizens and panchayat staff by automating the application and service management processes.

The system is designed to handle both user-side requests and administrative tasks like verifying and managing applications using a Firebase backend.

---

2. Objectives

- To automate and digitalize the gram panchayat services.
- To create an efficient workflow for staff and admin to verify, process, and approve applications.
- To provide citizens with an easy-to-use portal for applying for various services.
- To integrate Firebase for seamless database management and real-time updates.

---

3. Technologies Used

The project was developed using modern web development technologies:

- **Frontend**: HTML, CSS, JavaScript
- **Backend**: Firebase (Authentication, Realtime Database)
- **Version Control**: GitHub
- **Other Libraries**: Firebase JS SDK

---

4. System Architecture

The system consists of three main modules:

1. **User Module**:
    - Users can register and log in via Firebase Authentication.
    - After logging in, users can apply for various services.
2. **Admin Module**:

    - Admins manage the overall system, verify applications, and update statuses.
3. **Staff Module**:
    - Staff members handle service applications, including viewing, verifying, and rejecting services.

The Firebase Realtime Database is structured to store application details under user IDs categorized by service type.

---

5. System Modules

5.1 User Module

Users can:

- Register and log in to the system using their email.
- Apply for services like birth registration, marriage registration, caste certificate, etc.
- View the status of their applied services in the "Application Status" page.
- Receive an automatic application number and status updates.

5.2 Admin Module

Admins can:

- View all applications submitted by users.
- Approve or reject applications based on the validation of submitted data.
- Manage user profiles and provide updates on application statuses.

5.3 Staff Module

Staff members can:

- View the list of applications submitted under their assigned services.
- Verify or reject services based on submitted documents.
- Access details of each service from the "Manage Services" page.

---

6. Firebase Integration

The project uses Firebase for:

- **Authentication**: Secure login and registration for users, staff, and admins.

- **Realtime Database**: Storing user data, applications, and their statuses in a structured format.
- **Hosting**: Deployed the project on Firebase hosting. For personal use: I have stored admin email id as(myadmin@gmail.com) and the password (123456) in the realtime database project. So admin get access to the portal with given login details.
-

---

7. Features and Implementation

1. **Service Management**:
    - Users can apply for services like marriage registration, caste certificates, birth registeration, pension registration, ews certificates and income certificates etc. Users can apply for the services by registering themselves inside the register window and hence after registering they can login with the registered email and passaword through user login window.
    - Once you applied the for any service the application got submitted and stored in database. And users can also check the status of their submitted application with given view application status button in the user portal.
    - Applications are managed by staff members only they can verify or reject them. And access to the staff portal is given only for those members who are authenticated inside the realtime database in firebase project with valid email id and password database manually. Hence no outside users can access the staff portal.
        - For personal use: I have stored staff email id as(mystaff@gmail.com) and the password (123456) in the realtime database project. So staff member can get access to the portal with given login details.
    - And at the final stage applications are approved by the admin. Admin get access to admin portal by logging with the authenticated email id and password stored inside the realtime database of firebase project.
        - For personal use: I have stored admin email id as(myadmin@gmail.com) and the password (123456) in the realtime database project. So admin can get access to the portal with given login details.

2. **Application Status Page**:

- Users can view the status of all submitted applications.
- The page dynamically loads data from Firebase and allows staff to update the verification status.

3. **User-Friendly UI**:
   - The interface is designed to be simple and easy to navigate, ensuring an efficient user experience for both citizens and staff members.
   - Modal windows are used for providing additional details about services.
4. **Dynamic Actions for Admin and Staff**:
   - Admins can approve or reject applications with a single click.
   - Realtime status updates are provided to users.

---

8. User Interface Design

The project uses a clean and modern UI:

- **Responsive Design**: The UI is fully responsive and adjusts to different screen sizes.
- **Table Layout**: Service details and application statuses are displayed in tables, making it easier for users and staff to view and manage information.
- **Modals for Detailed View**: Detailed information about services is displayed in modal pop-ups.
- **Use of Icons**: Icons are integrated for better visual appeal.

---

9. Challenges Faced

- **Firebase Integration**: The initial setup for Firebase Authentication and Realtime Database required careful planning and debugging.
- **Real-time Status Updates**: Ensuring that updates to applications are reflected in real time across different user interfaces was a challenge.
- **Form Validation**: Ensuring correct form validation to avoid incomplete or invalid applications being submitted.

---

10. Conclusion

The **Digital E Gram Panchayat** project successfully digitalizes the services provided by gram panchayats, offering an efficient platform for citizens to apply for services and for staff to manage applications. This system streamlines the workflow of service management, making it quicker and more transparent.

---

11. Future Enhancements

- **Mobile Application**: Developing a mobile version of the platform for easier access.
- **Automated Notifications**: Implementing email or SMS notifications for status updates on applications.
- **Enhanced Security**: Further securing the system with multi-factor authentication (MFA) and other security measures.

Appendices

Appendix A: Home Page And Firebase Database Structure

Appendix B: Screenshots of the Application UI

Source Code Of The Home Page Saved As The Name Of MAININDEX.Html File

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Digital E Gram Panchayat</title>
  <link rel="stylesheet" href="../static/port.css">
  <style>
    * {
      margin: 0;
      padding: 0;
```

```
    box-sizing: border-box;
}
body {
    font-family: 'Arial', sans-serif;
    background: linear-gradient(135deg, #e3f2fd, #bbdefb);
    color: #333;
    line-height: 1.6;
}
header {
    background: #1976d2;
    color: white;
    padding: 20px 0;
    text-align: center;
    box-shadow: 0 2px 10px rgba(0, 0, 0, 0.2);
}
header h1 {
    font-size: 2.5em;
    margin-bottom: 10px;
    text-transform: uppercase;
}
nav {
    margin: 20px 0;
}
nav ul {
    list-style: none;
}
nav ul li {
```

```
    display: inline;
 margin: 0 20px;
}
nav ul li a {
    color: white;
    text-decoration: none;
    font-size: 1.2em;
    padding: 8px 15px;
    border-radius: 5px;
    transition: background 0.3s ease;
    background-color: #6f6c6c;
}
nav ul li a:hover {
    background: rgba(57, 54, 54, 0.3);
}
/* Main Section Styles */
main {
    max-width: 900px;
    margin: 40px auto;
    padding: 20px;
}
/* Intro Section Styles */
.intro {
    text-align: center;
    background: #64b5f6;
    padding: 40px 20px;
    border-radius: 10px;
```

```
    color: white;
    margin-bottom: 40px;
    text-transform: uppercase;
}
.intro a {
    text-transform: uppercase;
}
.intro h2 {
    font-size: 1.5em;
    margin-bottom: 20px;
    text-transform: uppercase;
}
.intro p {
    font-size: 1.3em;
    margin-bottom: 20px;
    text-transform: uppercase;
}
.intro .btn {
    display: inline-block;
    background: #ffffff;
    color: #1c1e1f;
    padding: 10px 20px;
    border-radius: 5px;
    text-decoration: none;
    transition: background 0.3s;
}
.intro .btn:hover {
```

```
    background: #e0e0e0;
}
.about {
    text-align: center;
    margin-bottom: 40px;
}
.about h2 {
    font-size: 2em;
    color: #1976d2;
    text-transform: uppercase;
}
.about p {
    font-size: 1.2em;
}
.modules {
    margin-bottom: 40px;
}
.modules h2 {
    font-size: 1.5em;
    text-align: center;
    margin-bottom: 20px;
    text-transform: uppercase;
}
.module-container {
    display: flex;
    justify-content: space-around;
    flex-wrap: wrap;
```

```
}
.module {
    background: #bbdefb;
    border-radius: 10px;
    padding: 20px;
    margin: 10px;
    flex: 1;
    min-width: 200px;
    box-shadow: 0 2px 10px rgba(0, 0, 0, 0.1);
}
.module h3 {
    text-align: center;
    color: #1976d2;
}
.module ul {
    list-style: none;
    padding: 0;
}
.module ul li {
    padding: 5px 0;
}
.cta {
    text-align: center;
    margin-bottom: 40px;
}

.cta h2 {
```

```
    font-size: 2em;
    margin-bottom: 20px;
}

.cta .btn {
    display: inline-block;
    background: #1976d2;
    color: white;
    padding: 10px 20px;
    border-radius: 5px;
    text-decoration: none;
    transition: background 0.3s;
}

.cta .btn:hover {
    background: #1565c0;
}

footer {
    text-align: center;
    padding: 15px 0;
    background: #1976d2;
    color: white;
    position: relative;
    bottom: 0;
    width: 100%;
}
```

```
    footer p {
        margin: 0;
        font-size: 1em;
    }
  </style>
</head>
<body>
  <header>
    <h1>Welcome to Digital E Gram Panchayat</h1>
    <nav>
      <ul>
        <li><a href="userlogin.html">User Portal</a></li>
        <li><a href="adminlogin.html">Admin Portal</a></li>
        <li><a href="stafflogin.html">Staff Portal</a></li>
      </ul>
    </nav>
  </header>
  <main>
    <section class="intro">
      <h2>Welcome to the Digital Village Portal</h2>
      <p>Empowering Gram Panchayats with Digital Services.</p>
      <a href="register.html" class="btn">Get Started! Register now</a>
    </section>
    <section class="about">
      <h2>Our Mission</h2>
      <p>We aim to improve the delivery of citizen services in villages by computerizing the application process
```

```
        for Gram Panchayat services.</p>
    </section>
    <section class="modules">
      <h2>System Modules</h2>
      <div class="module-container">
        <div class="module">
          <h3>User</h3>
          <ul>
            <li>Register & Login</li>
            <li>Apply for Services</li>
            <li>Track Application Status</li>
            <li>Manage Profile</li>
          </ul>
        </div>
        <div class="module">
          <h3>Admin</h3>
          <ul>
            <li>Create Services</li>
            <li>Update/Delete Services</li>
            <li>Manage Users</li>
          </ul>
        </div>
        <div class="module">
          <h3>Staff</h3>
          <ul>
            <li>View Services</li>
            <li>Update Application Status</li>
```

```
            </ul>
          </div>
        </div>
      </section>
      <section class="cta">
        <h2>Ready to Get Started?</h2>
        <a href="register.html" class="btn">Register Now</a>
      </section>
    </main>
    <footer>
      <p>&copy; 2024 Digital E Gram Panchayat</p>
    </footer>
    <script type="module">
      import { initializeApp } from "https://www.gstatic.com/firebasejs/10.14.0/firebase-app.js";
      import { getDatabase, ref, set, get, child } from "https://www.gstatic.com/firebasejs/10.14.0/firebase-database.js";
      const firebaseConfig = {
        apiKey: "AIzaSyCVwyQ6nlGzFcO9DMYGWRADLeGjCFxqOl0",
        authDomain: "e-gram-2da6d.firebaseapp.com",
        projectId: "e-gram-2da6d",
        storageBucket: "e-gram-2da6d.appspot.com",
        messagingSenderId: "406178850869",
        appId: "1:406178850869:web:cbecc9674508f7909be97c",
      };
      const app = initializeApp(firebaseConfig);
      const db = getDatabase(app);
      document.getElementById('loginForm').addEventListener('submit', function (e) {
```

```
        e.preventDefault();
        const email = document.getElementById('email').value;
        const password = document.getElementById('password').value;
        firebase.auth().signInWithEmailAndPassword(email, password)
            .then((userCredential) => {
                alert('Login Successful');


                window.location.href = 'user.html';
            })
            .catch((error) => {
                alert('Login Failed: ' + error.message);
            })
    });
    document.getElementById('registerBtn').addEventListener('click', function (e) {
        e.preventDefault();
        window.location.href = 'register.html';
    });
  </script>
</body>
</html>
```

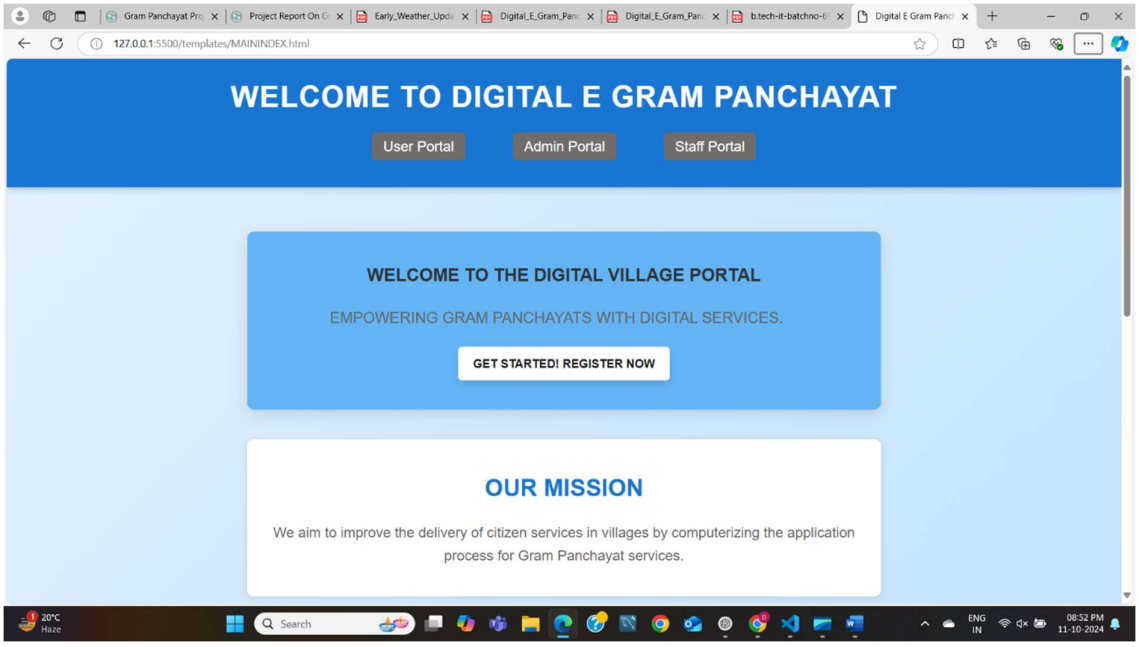
FIREBASE DATABASE STRUCTURE

```
<script type="module">
    import { initializeApp } from "https://www.gstatic.com/firebasejs/10.14.0/firebase-app.js";
    import { getDatabase, ref, set, get, child } from "https://www.gstatic.com/firebasejs/10.14.0/firebase-database.js";

    const firebaseConfig = {
      apiKey: "AIzaSyCVwyQ6nlGzFcO9DMYGWRADLeGjCFxqOl0",
      authDomain: "e-gram-2da6d.firebaseapp.com",
      projectId: "e-gram-2da6d",
      storageBucket: "e-gram-2da6d.appspot.com",
      messagingSenderId: "406178850869",
      appId: "1:406178850869:web:cbecc9674508f7909be97c",
    };
    const app = initializeApp(firebaseConfig);

    const db = getDatabase(app);
    document.getElementById('loginForm').addEventListener('submit', function (e) {
      e.preventDefault();

      const email = document.getElementById('email').value;
      const password = document.getElementById('password').value;
      firebase.auth().signInWithEmailAndPassword(email, password)
        .then((userCredential) => {
          alert('Login Successful');
          window.location.href = 'user.html';
        })
        .catch((error) => {
          alert('Login Failed: ' + error.message);
```

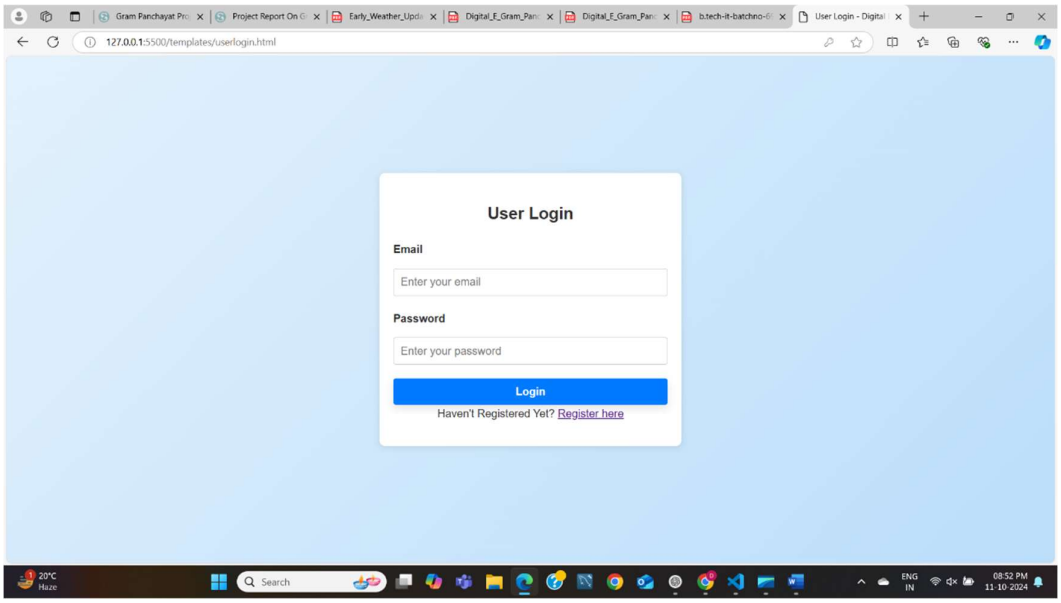
```
        })
    });
    document.getElementById('registerBtn').addEventListener('click', function (e) {
        e.preventDefault();
        window.location.href = 'register.html';
    });
</script>
```

SCREENSHOTS OF THE APPLICATION UI

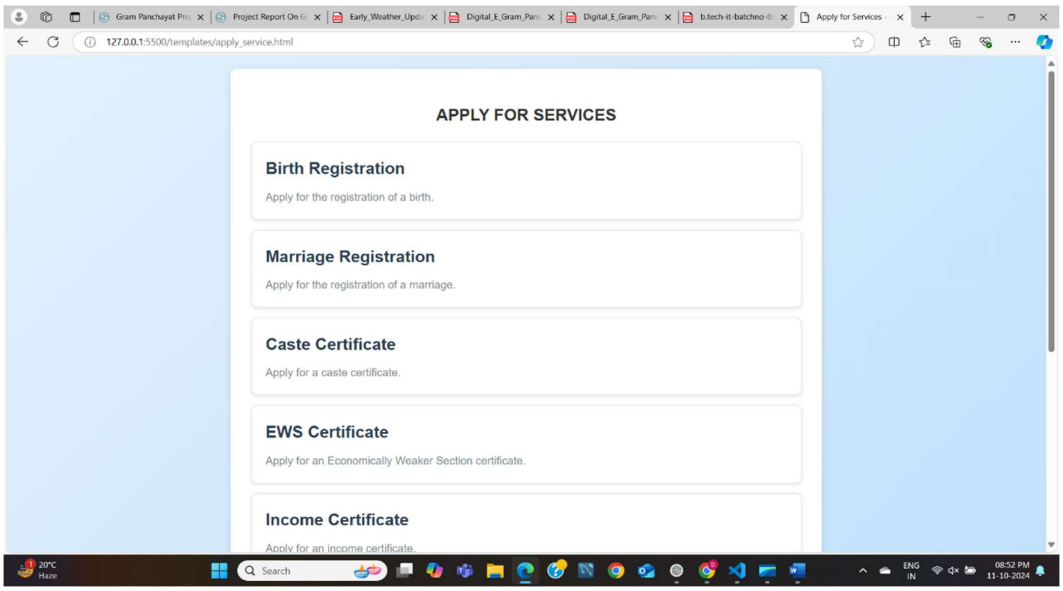
MAIN HOME PAGE VIEW



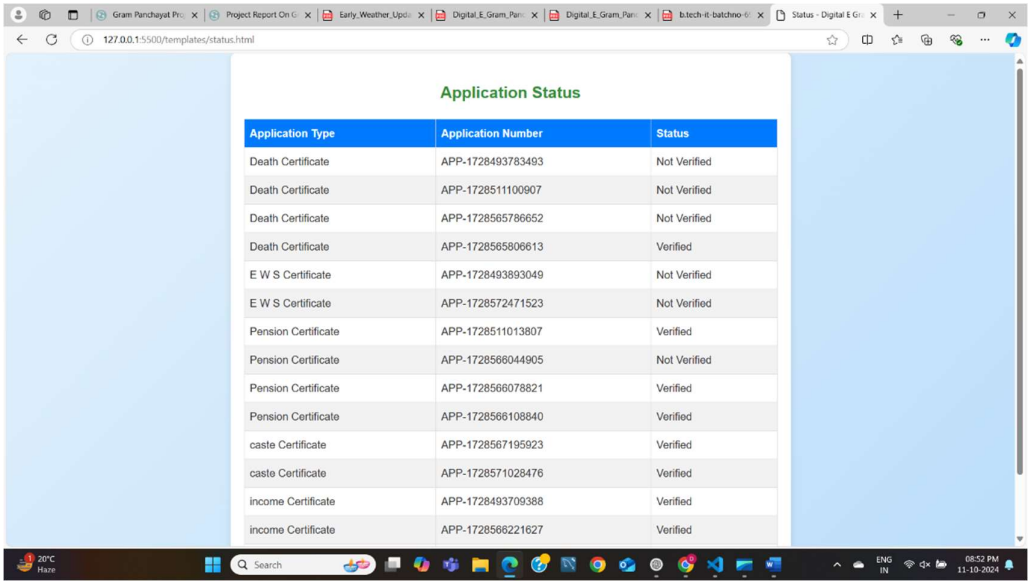
USER LOGIN PAGE



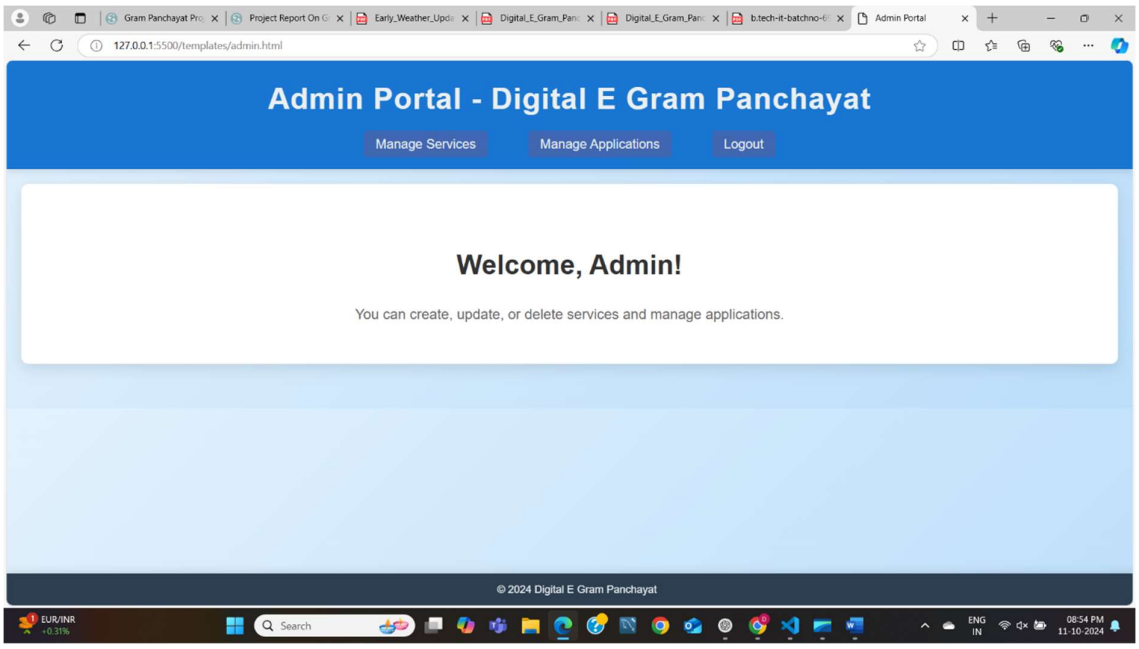
APPLY FOR SERVICES USER PORTAL



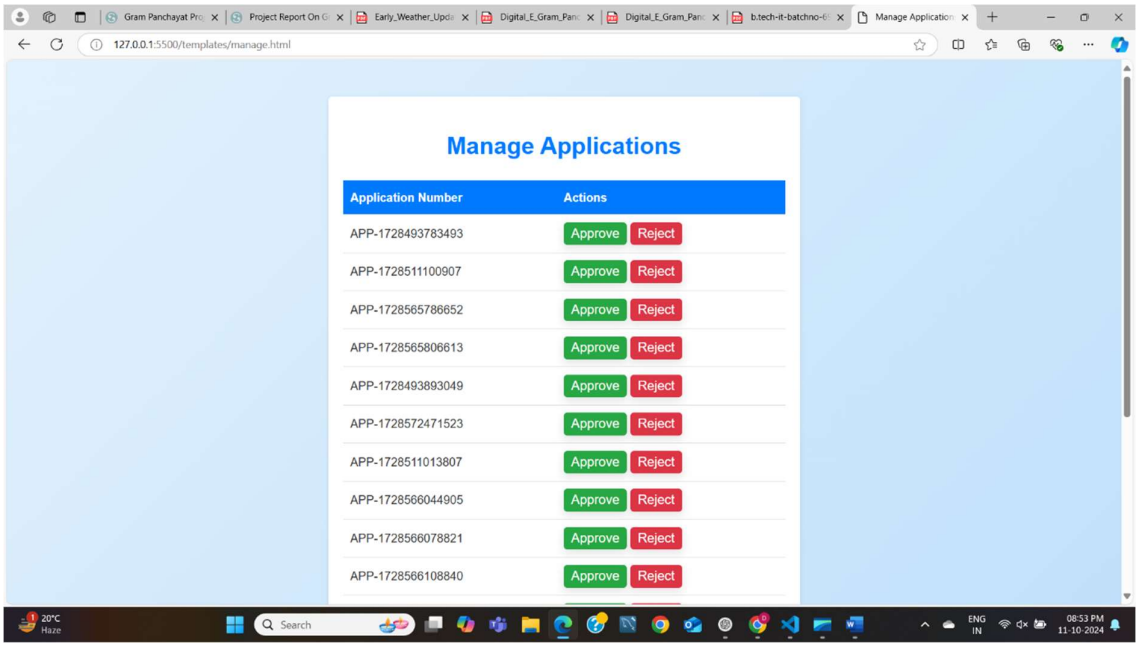
VIEW APPLICATION STATUS FOR USERS



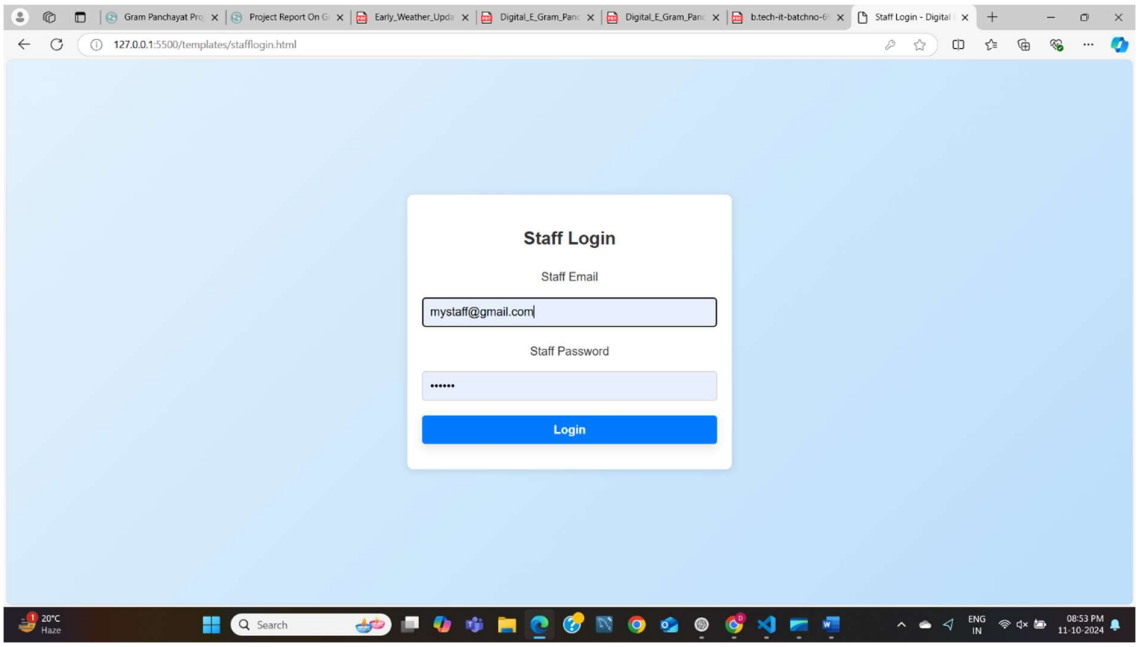
ADMIN PORTAL(LOGIN WITH EMAIL:myadmin@gmail.com AND PASSWORD:123456)



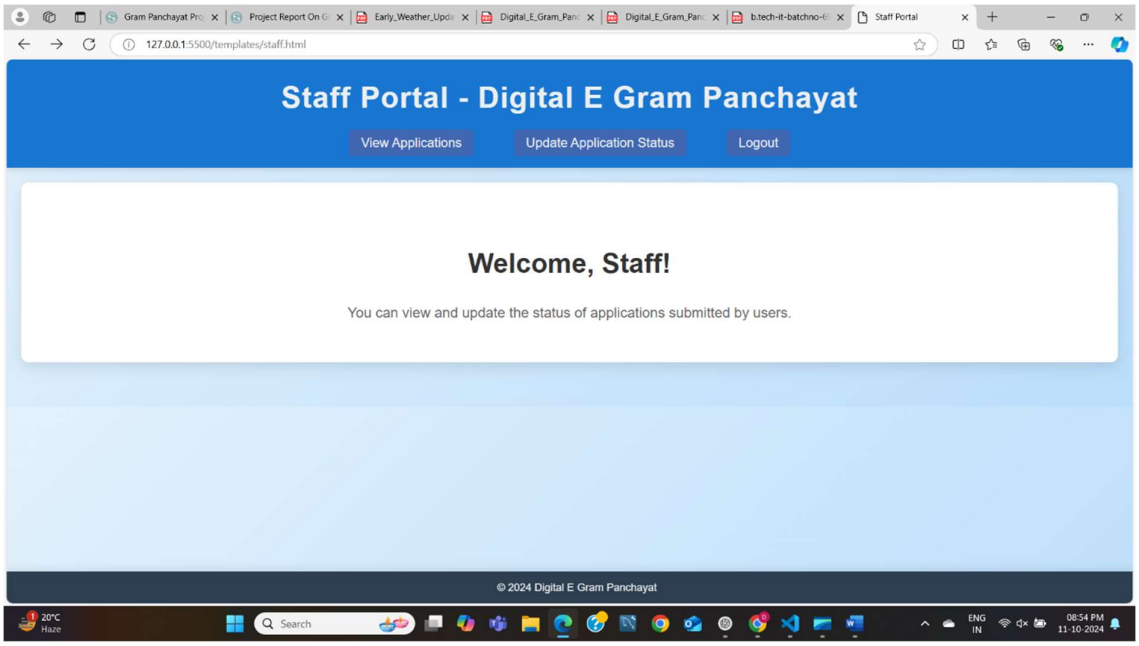
APPICATION OF THE USERS TO BE APPROVED BY ADMIN



STAFF LOGIN(LOGIN WITH EMAIL:mystaff@gmail.com AND PASSWORD:123456)



STAFF PORTAL



STAFF PORTAL TO VERIFY THE APPLICATIONS