1. Introduction

The Digital eGram Panchayat system is a web-based application designed to digitize various services offered by a Gram Panchayat.

It provides a portal for citizens to apply for services, raise grievances, track applications, and receive digital certificates.

Staff members can manage services, update application statuses, and communicate announcements effectively. This system improves efficiency, transparency, and accessibility within the local governance framework.

2. Objective

The main objectives of the project are:

- To create a centralized system for managing rural services.
- To digitize the manual procedures of Gram Panchayat offices.
- To improve communication between citizens and Panchayat staff.
- To enhance transparency and reduce paperwork.

3. Tools and Technologies Used

Frontend:

- React.js

Backend:

- Firebase Firestore (Database)
- Firebase Authentication
- Firebase Hosting

Others:

- JavaScript, HTML, CSS, Lucide Icons, Tailwind CSS

4. System Architecture

The project follows a client-server architecture:

- The frontend is built using React.js and hosted on Firebase Hosting.
- Firebase Authentication handles user login and registration.
- Firebase Firestore acts as the NoSQL backend database storing all data.
- The application interacts with Firebase services using SDK and axios.

5. Modules

Citizen Module:

- Register/Login
- Search Services
- Apply for Services
- Track Application Status
- Submit Grievances
- Download Certificates

Staff Module:

- Login
- Create/Update/Delete Services
- View Applications and Update Status
- View/Manage Grievances

- Make Announcements

Admin Module (Future Scope):

- Manage Staff and Users

- Dashboard Analytics

6. Features

- Responsive Design

- Role-based Access Control

- Realtime Service Updates

- Grievance Redressal

- Firebase Storage for uploading files (photos, documents)

- Service Application with Unique IDs

7. Database Structure

Collections:

- users (citizen details)

- staff (staff details)

- services (service details)

- applyservices (application records)

- complaint (grievance records)

- announcements (notices)

Each collection includes structured fields for easy access and querying.

8. Implementation Steps

1. Initialize Firebase project and configure Firestore & Authentication.

2. Build frontend pages using React.

3. Integrate Firebase Auth in Login/Register pages.

4. Store and retrieve data using Firestore queries.

5. Implement dynamic routing and dashboards for different roles.

6. Deploy app using Firebase Hosting.

9. Deployment

- Firebase Hosting is used to deploy the live version of the application.

- The deployed site can be accessed at: https://digitalgrampanchayat-e1d66.web.app

- Changes in the codebase can be redeployed using `firebase deploy`.

10. Challenges Faced

- Authentication validation using Firebase

- Real-time updates and state management in React

- Role-based routing and data filtering

- Firestore security rules and performance tuning

11. Future Enhancements

- Integration with Aadhar/e-KYC systems

- Mobile App development using React Native

- Advanced Analytics Dashboard for Admin

- SMS/Email Notifications for service updates

12. Conclusion

The Digital eGram Panchayat system successfully addresses the gap between citizens and rural administrative bodies through a user-friendly and modern web interface.

It promotes digital empowerment in villages, streamlines governance processes, and aligns with India's vision of Digital India.