

Project Name: Digital Grampanchayat

Documentation On

**“Digital Grampanchayat”**

PG-DAC SEPT 2022

**Guided By: Snehal Somawanshi**

Submitted By :

**Group No: 13**

* **Suraj Chechare 220943020103**
* **Sanket Huljute 220943020038**
* **Varma Abhishek 220943020107**
* **Diksha Upadhyay 220943020027**

**Table of Contents**

**1. Introduction 4**

1.1 Document Purpose **4**

1.2 Project Background **4**

1.3 Goals of the project **4**

**2. Business Requirements Overview**…………………………………………………………**5**

**3.Functional Requirements Overview** **6**

3.1 User Module **6**

3.2 Admin Module **6**

3.3 PDO Module **6**

**4.** **Non-Functional Requirement………………………………………………………………7**

**5. Use Case Diagram 8**

5.1 User **9**

5.2 Admin **10**

5.3 PDO **11**

**6. Database Design 12**

1 User **12**

2 User\_types **12**

3 Userproblem.…………………………………………………………………………… **13**

4 Schemes **13**

5 Birthdeath **13**

6 Btype **13**

7 Register\_report **13**

7 Report **13**

**7. E-R Diagram………………………………………………………………………………………14**

**8. Snapshots 15**

**9. Conclusion 20**

**List of Figures**

**Use Case Diagrams 17**

Fig 1 User 8

Fig 2 Admin 9

Fig 3 PDO 10

Fig 4 ER Diagram 14

1. **Introduction:**
   1. **Document Purpose:**

This document communicates with the rural area people for solving their problems and give awareness about new technologies/schemes.

## Project Background:

Digital Gram Panchayat is a web application that is specially, designed to establish an environment, wherein villagers can easily communicate with the village councils. Our application features: administrator, PDO (Panchayat Development Officer) and village public. Administrator will handle the whole website and manage the activities of users. PDO will update panchayat services, application, payments and complaints filled by public. The village public will register to the application and avail for the facilities. This post has complete information of Digital Gram Panchayat Project Documentation.

## Aim & Objectives:

Our project is aimed at achieving the solution for the problems that are faced by the public in villages. Through this application public can easily contact the PDO regarding any issues such as availing for services, complaints filling, bill payments, application filling and feedbacks. The PDO can easily keep track on the village problems and managements.

1. **Business Requirements Overview:**

* Digital Gram Panchayat is the public web application.
* Digital Gram Panchayat will be available for every village in India, but in the phase 1, the main target is for a particular village.
* There are mainly three types of user. Admin, User (public) and PDO.
* User (Villagers) can search for the schemes and register their problems.
* Admin (Village Head) will use the interface for solving the problems.
* Digital Gram Panchayat will be maintained by Head-administrator (PDO) .
* Digital Gram Panchayat provides the functions, which connect the Users, Admin and the PDO efficiently.

# Functional Requirements Overview:

* Digital Grampanchayat System consists of four modules described as below.

1. User Module
2. Admin Module
3. PDO Module

# **User Module**

* Login: After user successful registration, user can login.
* Apply services: The user can apply for services by filling the application.
* Post complaints and feedback: Here user can post the complaints on the Panchayat portal.
* Birth and Death information: This module enables the user to update the details of birth and death.
* News and Notification: This module enables the user to see the PDO updated news and notifications.
* Change password: Here user can change old password into new password.

# **3.2 Admin Module**

* Login: In this module, the admin can be login.
* Manage public: Admin can manage the activities of the public, such as manage their Services, Complaint, Feedback, Birth and Death details.
* Manage User: Manages the User details.
* Approval: Manages approval for Birth and death certificates.
* Manage house details: The PDO will manage the public like updating, adding a family member etc...
* Report preview: The admin can preview the report of the application such as manage house details, updated services details, new schemes and notification.
* Register: In this module, the user can be register.

# **3.3 PDO Module**

* Login: Here PDO can be login.
* Update services details: Here PDO update the services details.
* Manage application: Here PDO can manage the Admin level issue.
* Update news and notification: Here the PDO will update all news and notifications that are to be seen by users.
* Complaints and feedback: The PDO will check the user updated complaints and feedback.
* Login: In this module, the admin can be login.
* Manage public: Admin can manage the activities of the public, such as manage their Services, Complaint, Feedback, Birth and Death details.
* Manage User: Manages the User details.
* Approval: Manages approval for Birth and death certificates.
* Manage house details: The PDO will manage the public like updating, adding a family member etc...
* Report preview: The admin can preview the report of the application such as manage house details, updated services details, new schemes and notification.
* Register: In this module, the user can be register.

1. **Non-Functional Requirement:**

* The website should use professional design, look and feel and color scheme.
* Users will have no limitations for accessing the application through Internet. The portal being an internet application, it is difficult specify exact number of visitor or users. Hence, we will target the system to support between five and ten million users on launch of phase 1.
* Being a public website, the site must follow general usability guidelines for menus, navigation, colors, links and other actions provided on the screens.
* The system should be designed in such a manner that user will be able to complete tasks in minimum number of steps.

## Use-Case Diagram

**5.1 User:**

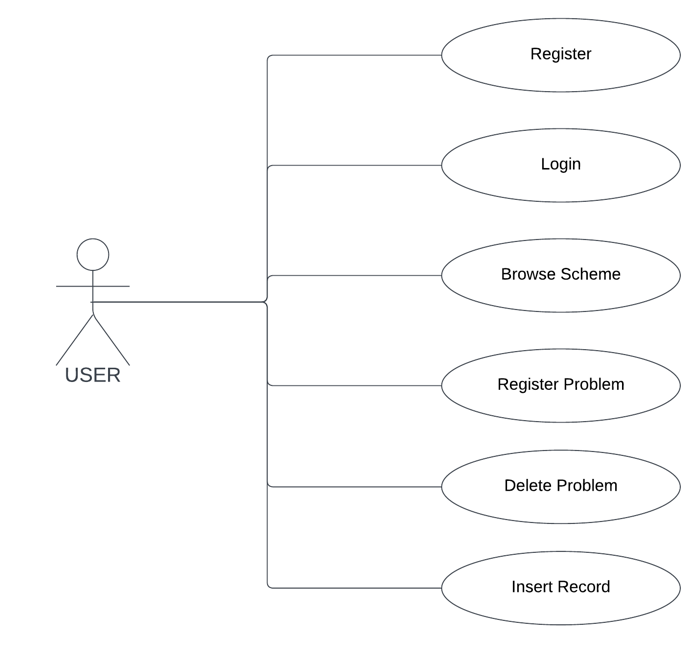


Fig. Use-Case Diagram for User

## 5.2 Admin:

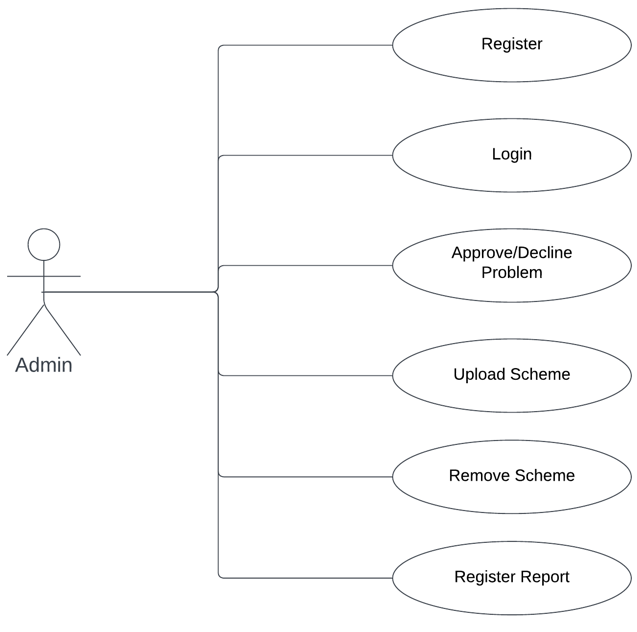


Fig. Use-Case Diagram for Admin

**5.3 PDO:**

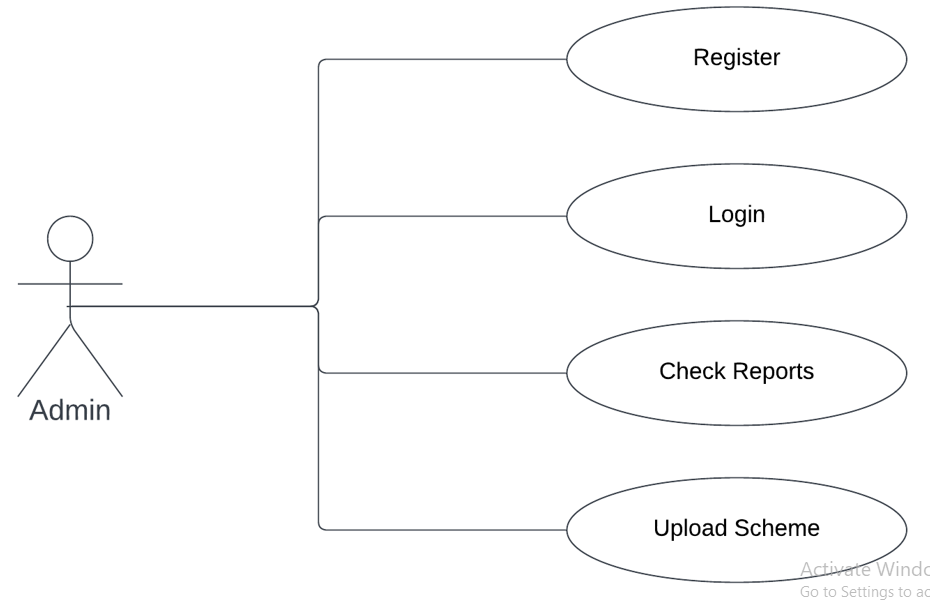


Fig. Use-Case Diagram for PDO

**6. Database Design:**

**1] User table**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Field** | | **Type** | | **Null** | | **Key** | | **Default** | | **Description** |
| user\_id | | Int | | No | | PK | | AI | | User ID |
| first\_name | | Varchar(255) | | No | |  | | Null | | First Name |
| type | | Int | | No | | FK | | Null | | Types of user |
| aadharcard\_no | | Int | | No | |  | | Null | | Aadharcard No. Of User |
| address | | Varchar(255) | | No | |  | | Null | | Address |
| contact\_no | | Varchar(255) | | No | |  | | Null | | Contact No. Of User |
| email | | Varchar(255) | | No | |  | | Null | | Email ID Of User |
| last\_name | | Varchar(255) | | No | |  | | Null | | Last Name |
| password | Varchar(255) | | No | |  | | Null | | Account Password | |
| username | | Varchar(255) | | No | |  | | Null | | User Name |

**2] User\_type table**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| type | Int | No | PK | AI | Type |
| user\_type | Varchar(45) | No |  | Null | User type |

**3] Userproblem table**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| pid | Int | No | PK | AI | Problem ID |
| area | Varchar(255) | No |  | Null | Problem in which area |
| date | Datetime(6) | No |  | Null | Date of the problem |
| probdesc | Varchar(255) | No |  | Null | Problem in description |
| status | Varchar(255) | No |  | Null | status |
| useridfk | Int | No |  | Null | Problem solved date |

**4] schemes table**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| sid | Int | No | PK | AI | Schemes ID |
| sname | Varchar(100) | No |  | Null | Scheme name |
| Description | Varchar(200) | No |  | Null | schemes in details |
| link | Varchsar(100) | No |  | Null | Link of the scheme |
| startdate | Date | No |  | Null | Starting date of the schemes |
| endtime | Date | No |  | Null | End date of the schemes |

**5] birthdeath table**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| bid | Int | No | PK | Null | Birth ID |
| bname | Varchar(255) | No |  | Null | Birth Name |
| bplace | Varchar(255) | No |  | Null | Place of the birth |
| date | Datetime(6) | No |  | Null | Birth date |
| hospitalname | Varchar(45) | No |  | Null | Hospital of the birth |
| bdtype | Int | No |  | Null | Birth-Death type |
| useridfk | Int | No | FK | Null | User ID |

**6] btype table**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| bdtype | Int | No | PK | AI | Type |
| bdtypecat | Varchar(45) | No |  | Null | Birth-Death type |

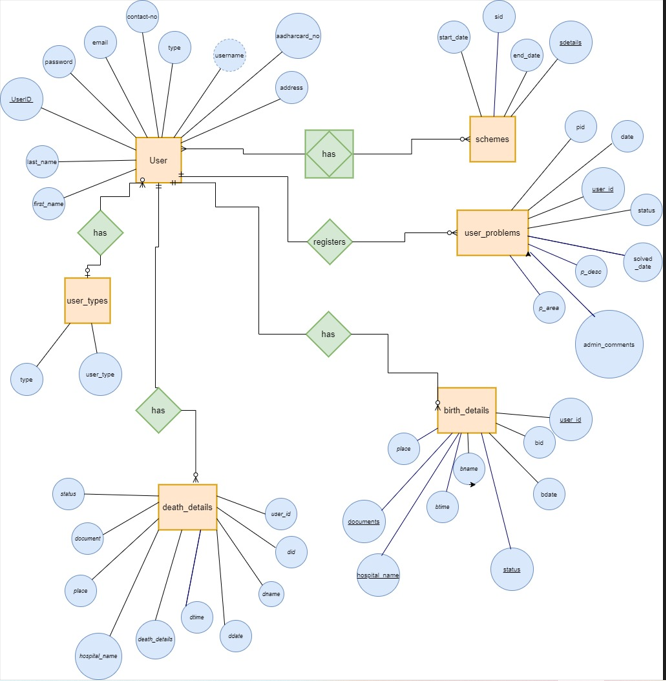
**7] Register\_report table**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| id | Int | No | PK | AI | Register Report ID |
| fname | Varchar(45) | No |  | Null | First name |
| lname | Varchar(45) | No |  | Null | Last Name |
| Work\_desc | Varchar(45) | No |  | Null | Work Description |
| status | Int | No |  | Null | Status of the report |
| link | Varchar(45) | No |  | Null | Link of the work |

**8] Report table**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Key** | **Default** | **Description** |
| id | Int | No | PK | AI | Report ID |
| fname | Varchar(255) | No |  | Null | First name |
| link | Varchar(255) | No |  | Null | Link of the work |
| lname | Varchar(255) | No |  | Null | Last Name |
| status | Int | No |  | Null | Status of the report |
| Work\_desc | Varchar(255) | No |  | Null | Description of the work |
| type | Int | No |  | Null | Type of the report |

## 7. ER-Diagram:

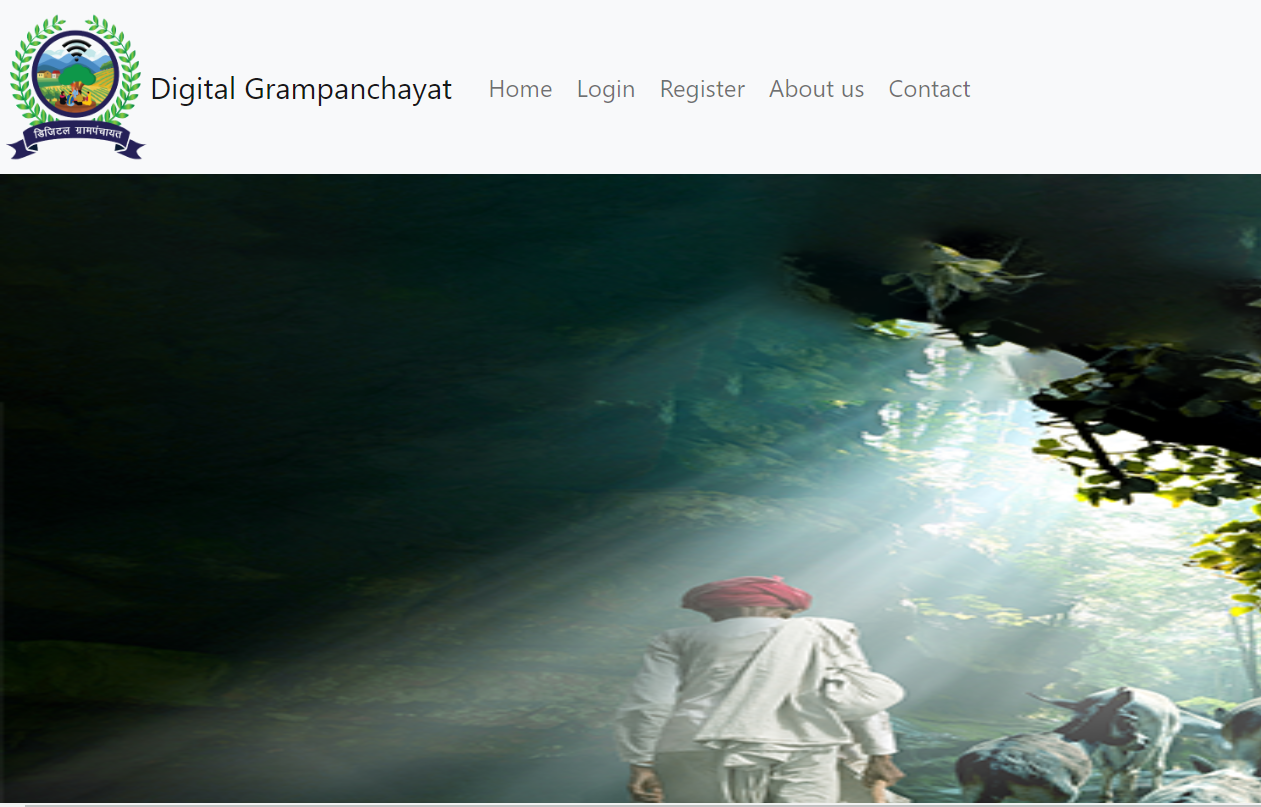


**E-R diagram shows database of Digital Grampanchayat**

**8. Snapshots:**

**8.1 Home Page:**

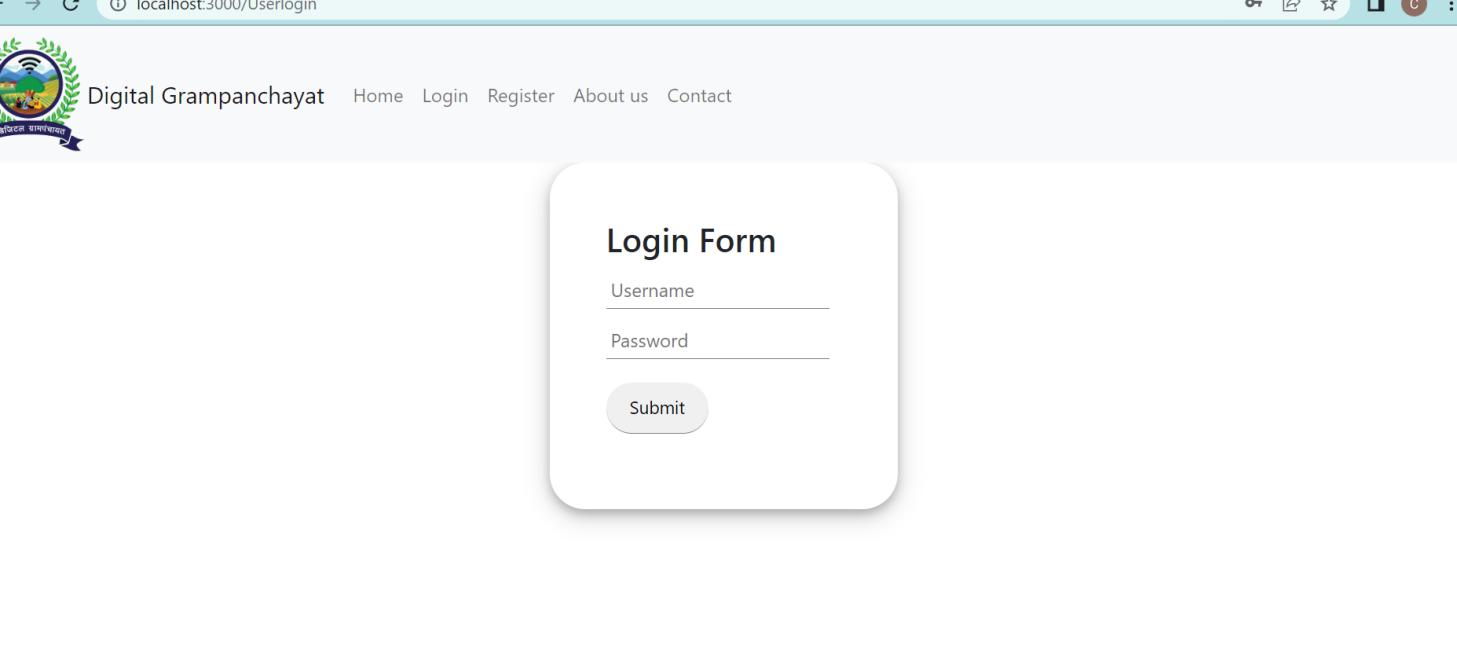
Following snapshot shows the Home page for Digital Grampanchayat

****

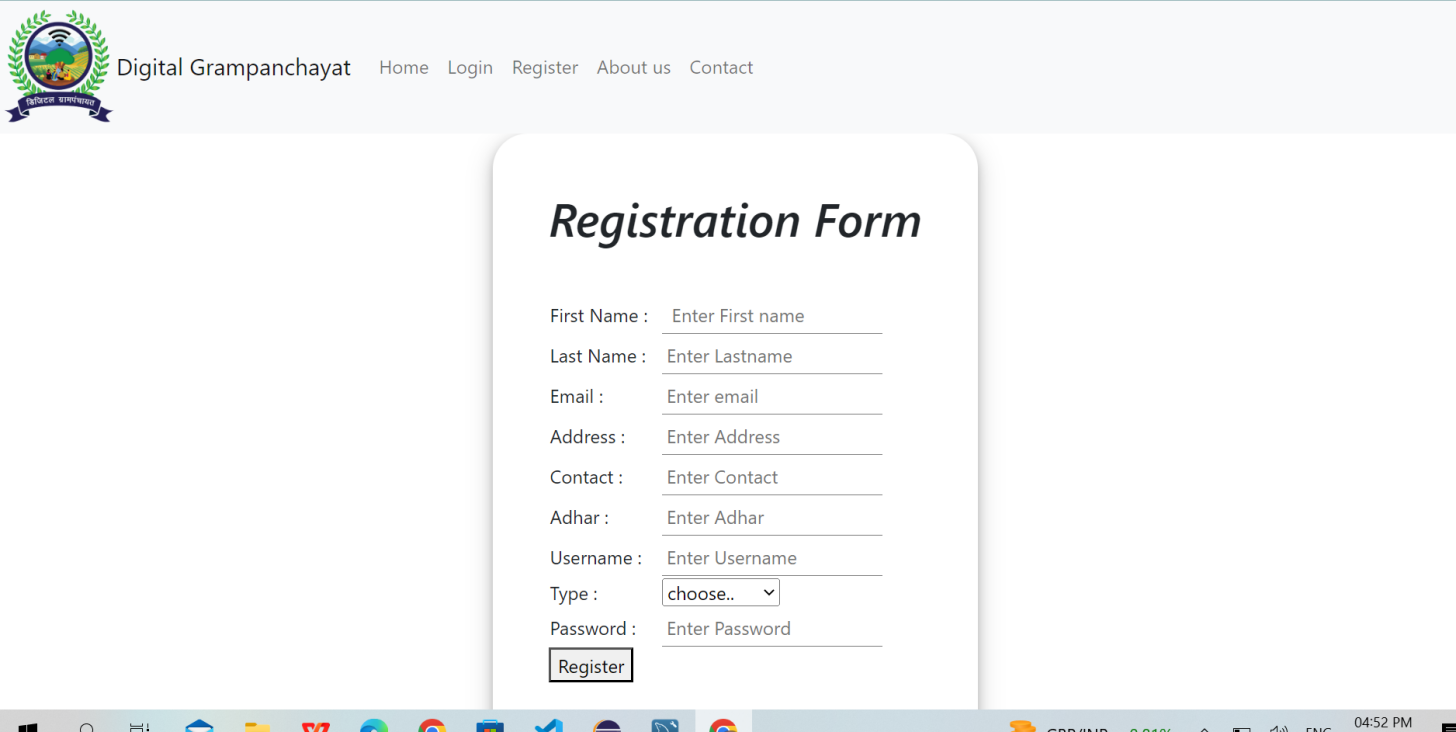
This page contains following controls

* Home
* Login
* Register
* About Us
* Contact

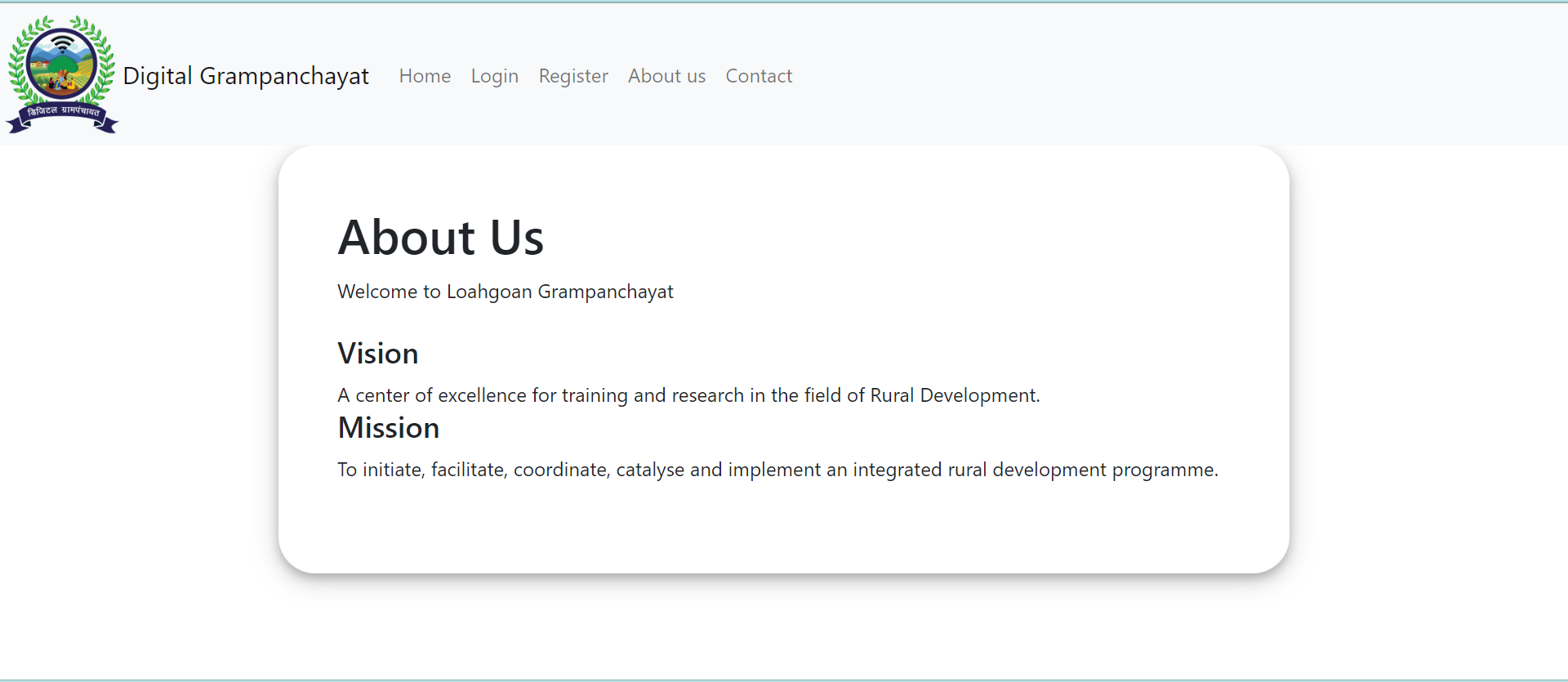
**8.2 Login Page:**

****

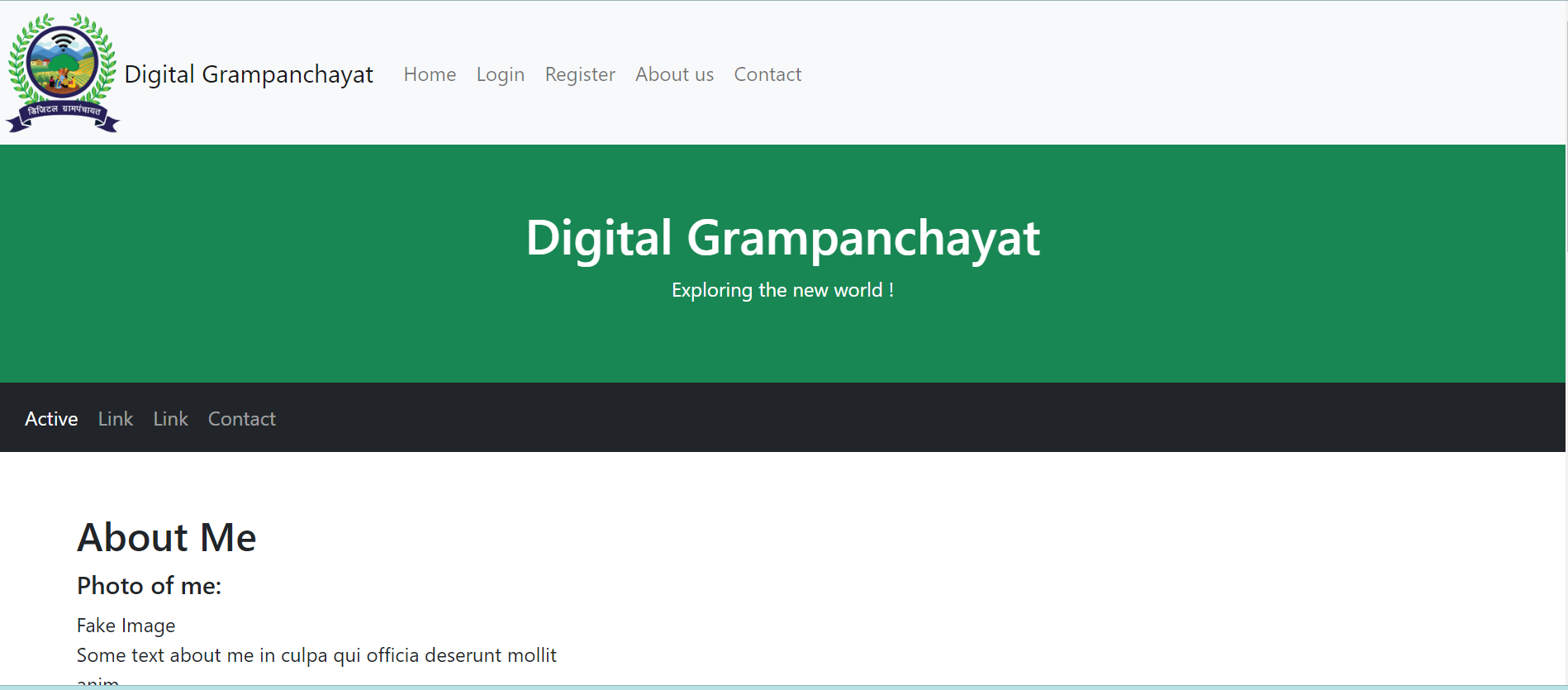
**8.3 Register Page:**

****

**8.4 About us Page:**

****

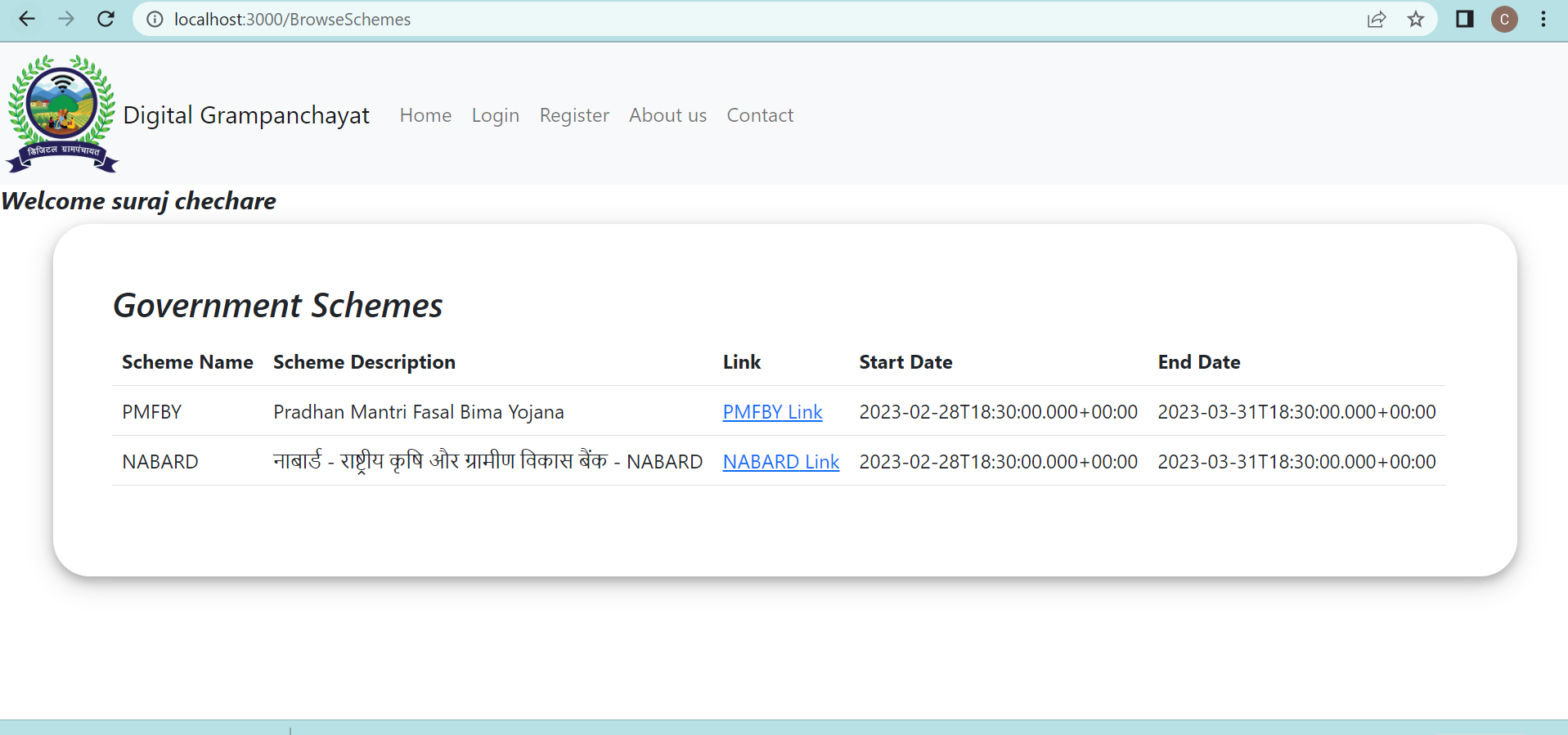
**8.5 Contact Page:**

****

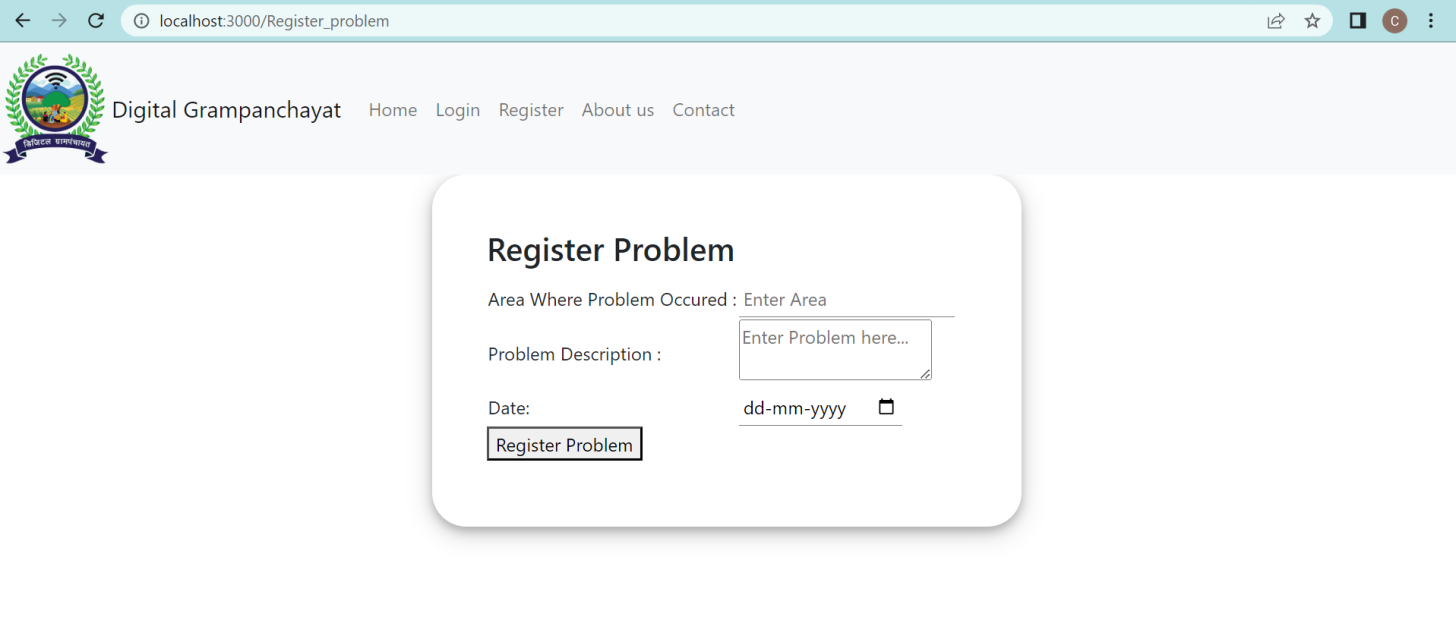
**User**

Following snapshot shows the User Related cases

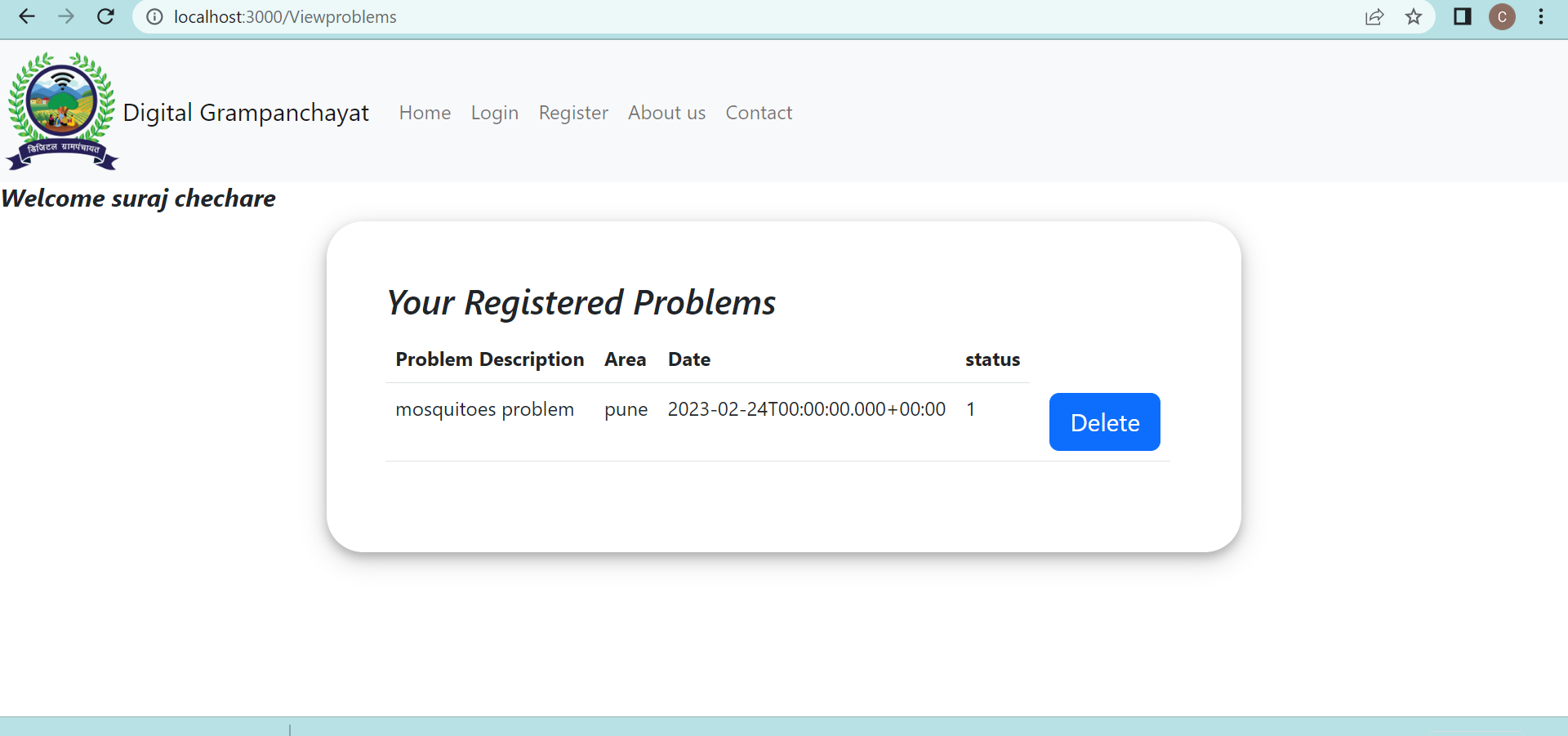
**8.6 BrowseSchemes Page**



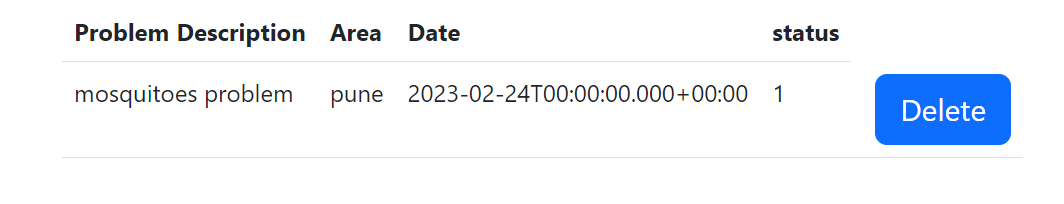
**8.7 Register Problem Page**



**8.8 View Problem Page**

****

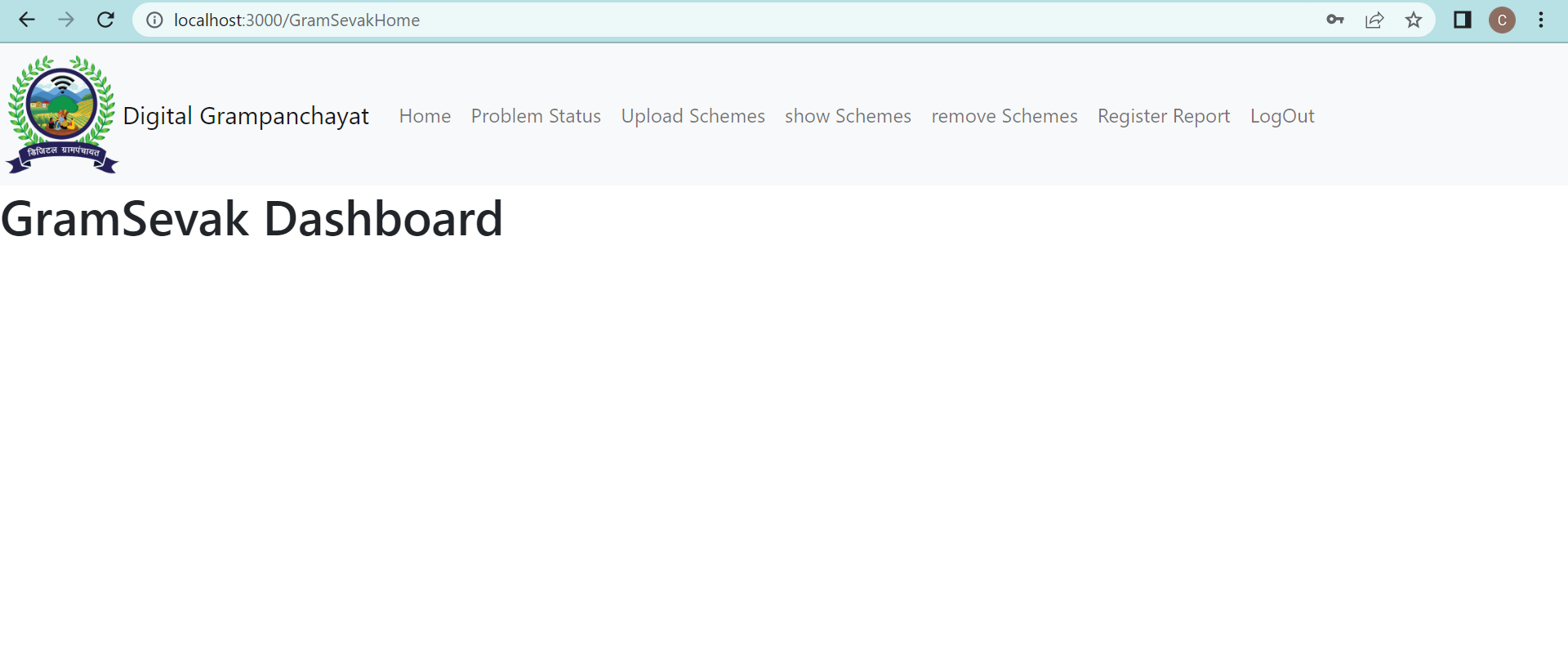
**8.9 Delete Problem Page**



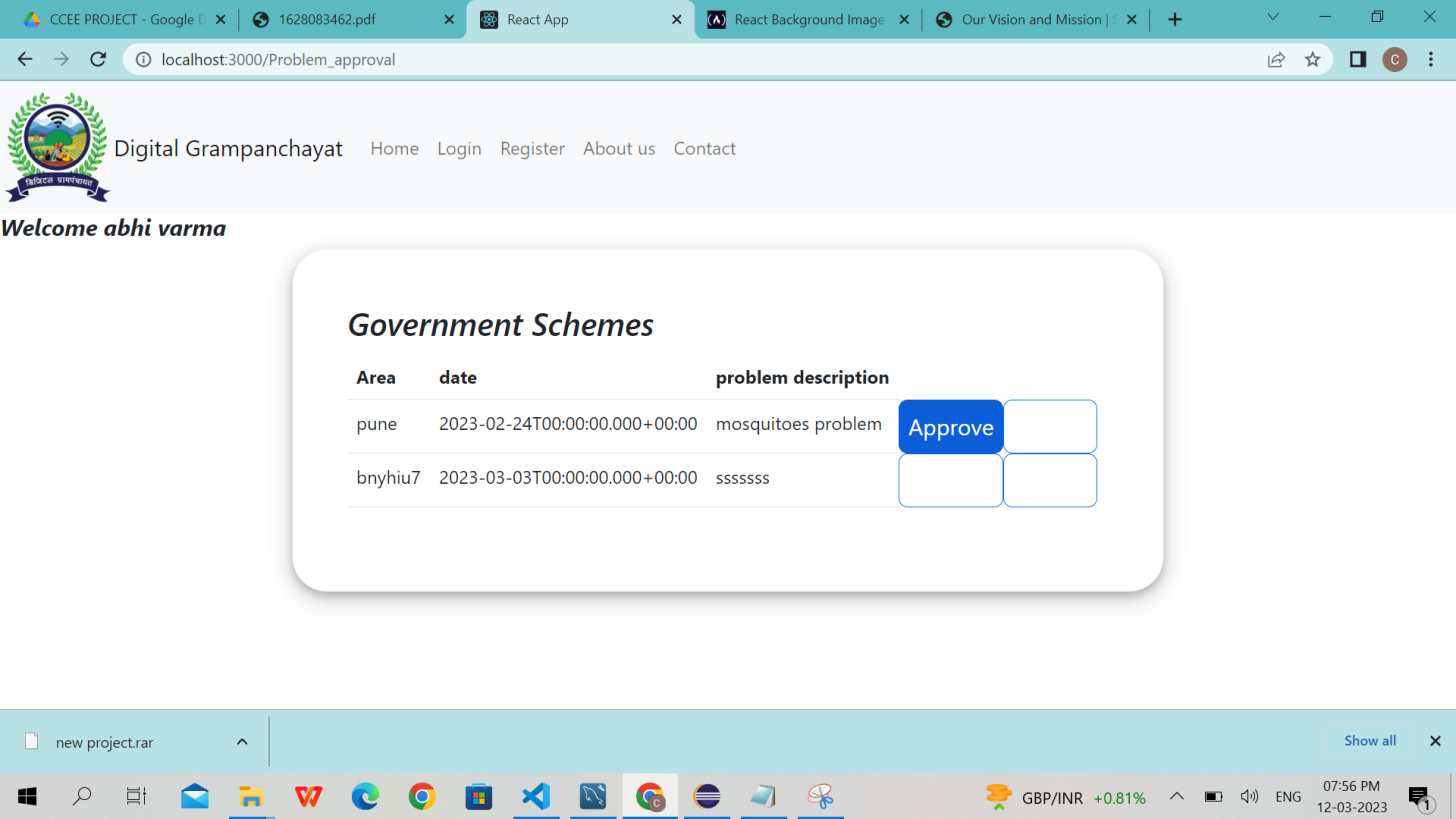
**Admin (Gramsevak)**

Following snapshot shows the Admin Related cases

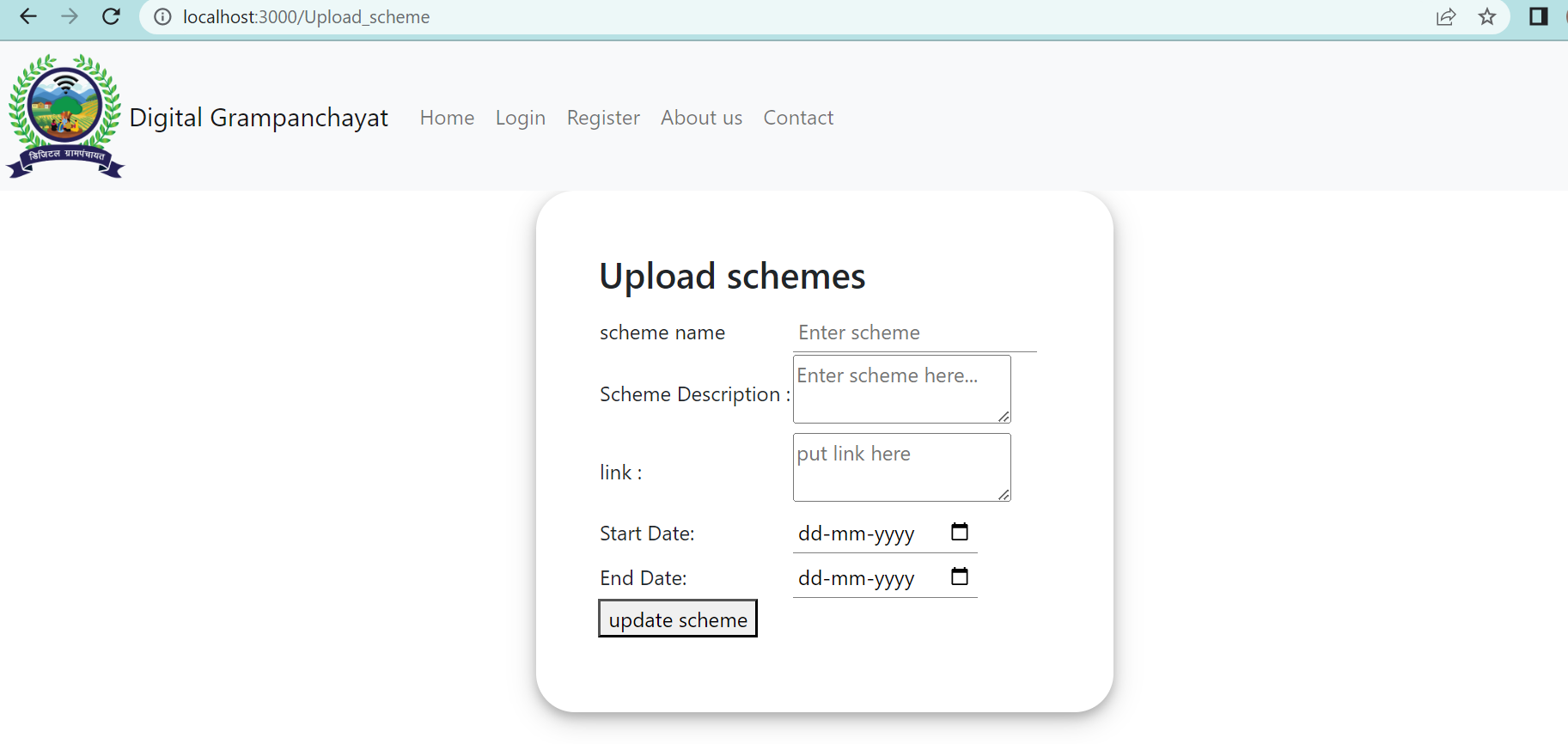
**8.10 Gramsevak Dashboard Page**



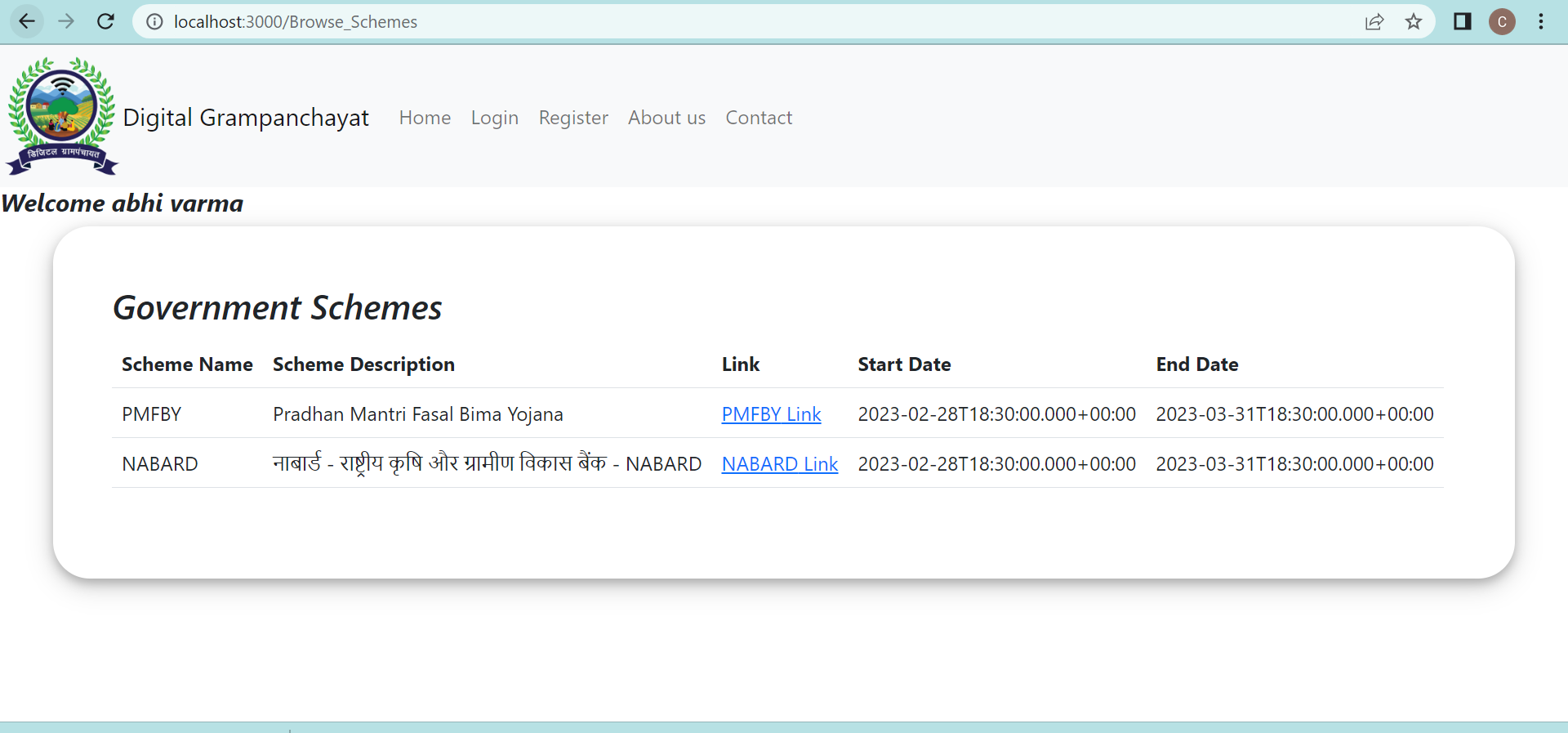
**8.11 Problem Status Page**

****

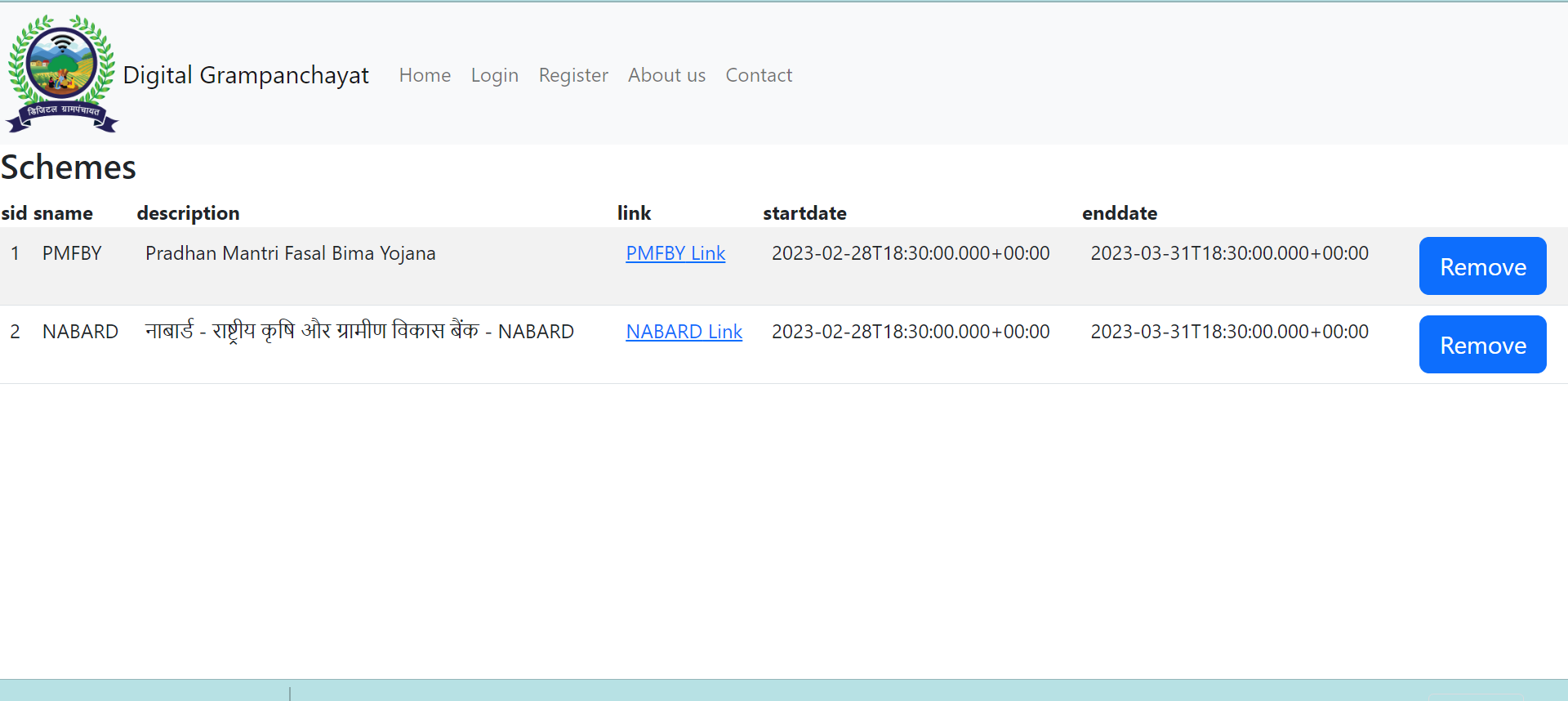
**8.12 Upload Schemes Page**

****

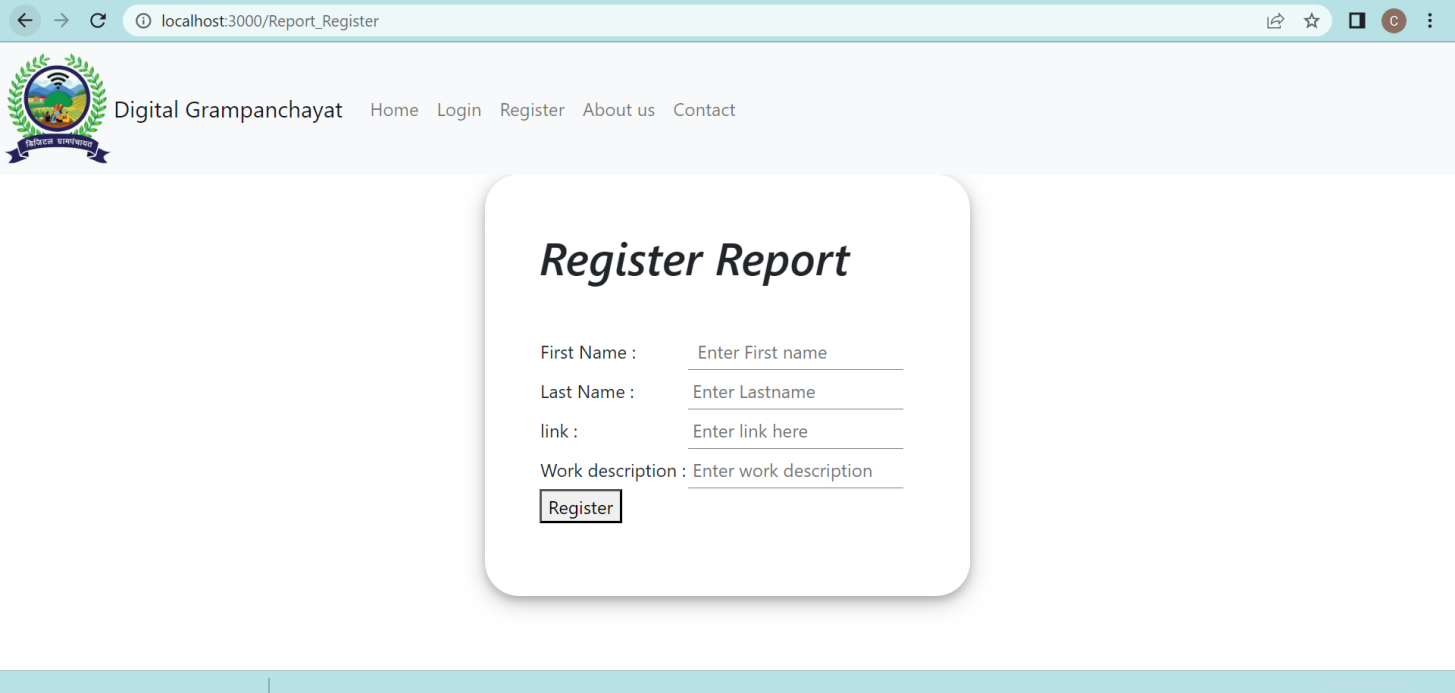
**8.13 Show Schemes Page**

****

**8.14 Remove Schemes Page**

****

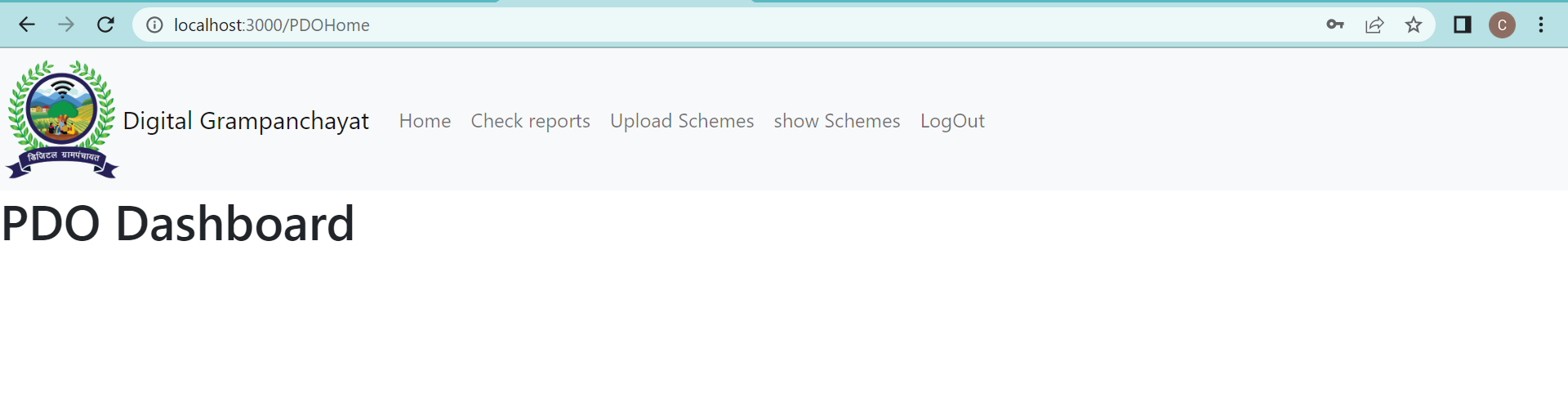
**8.15 Register Report Page**

****

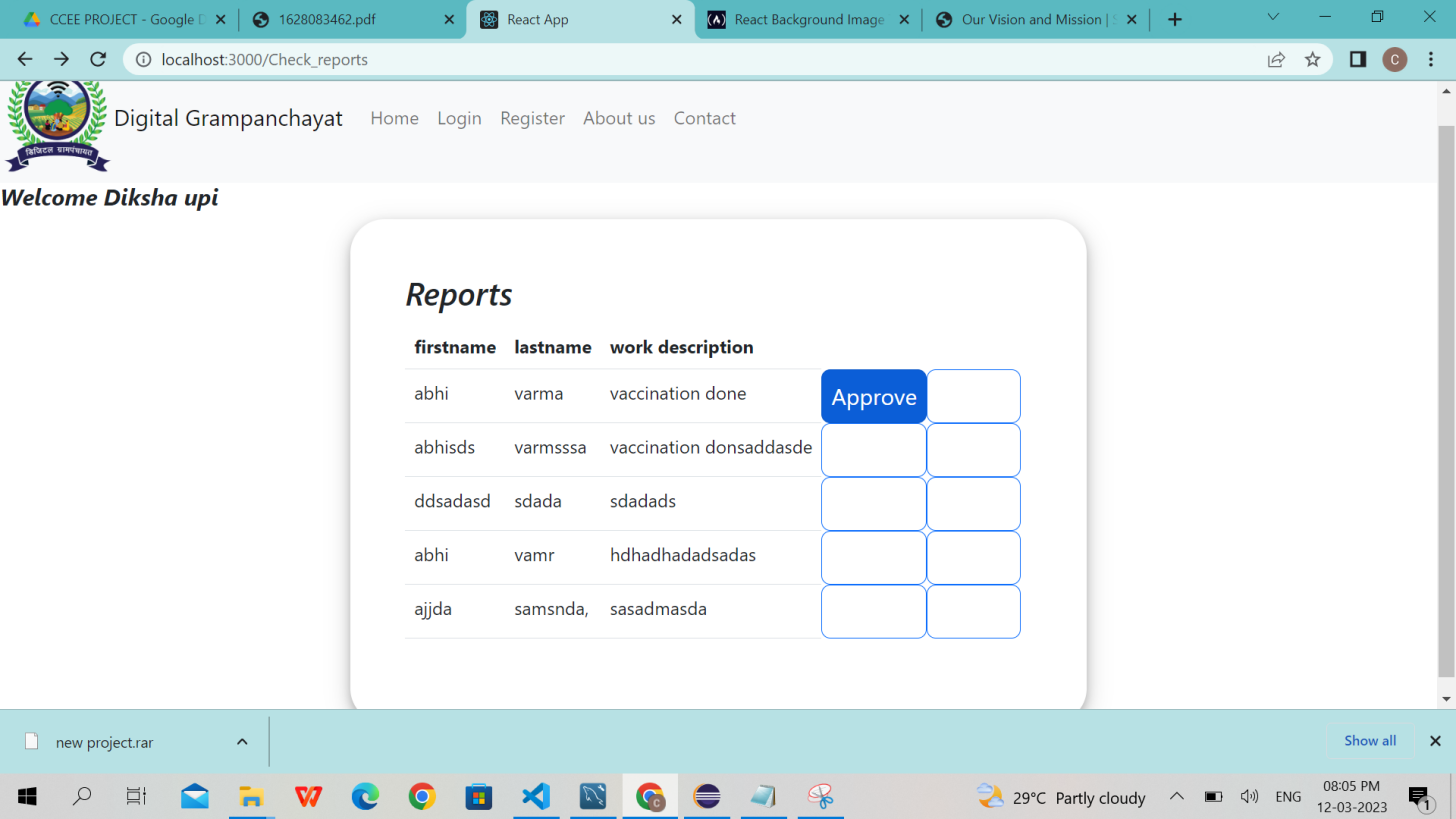
**PDO**

Following snapshot shows the PDO Related cases

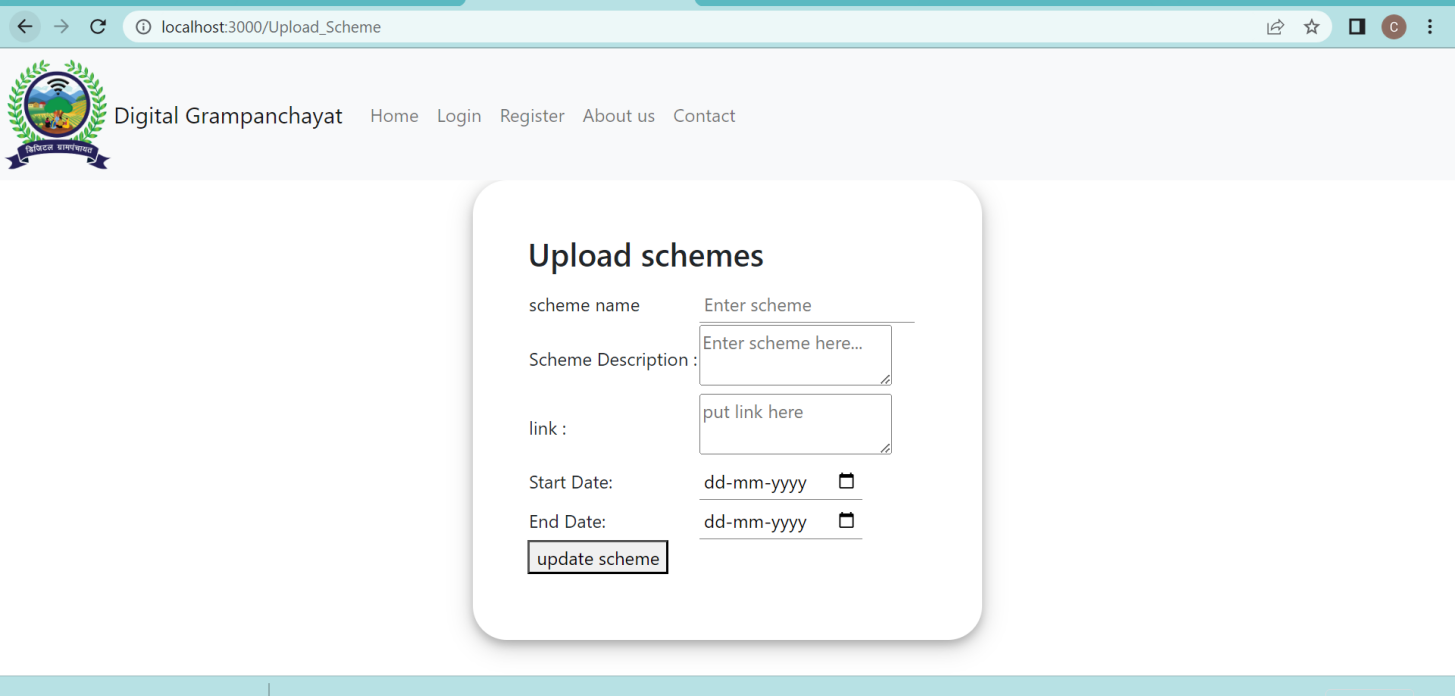
**8.16 Dashboard Page**

****

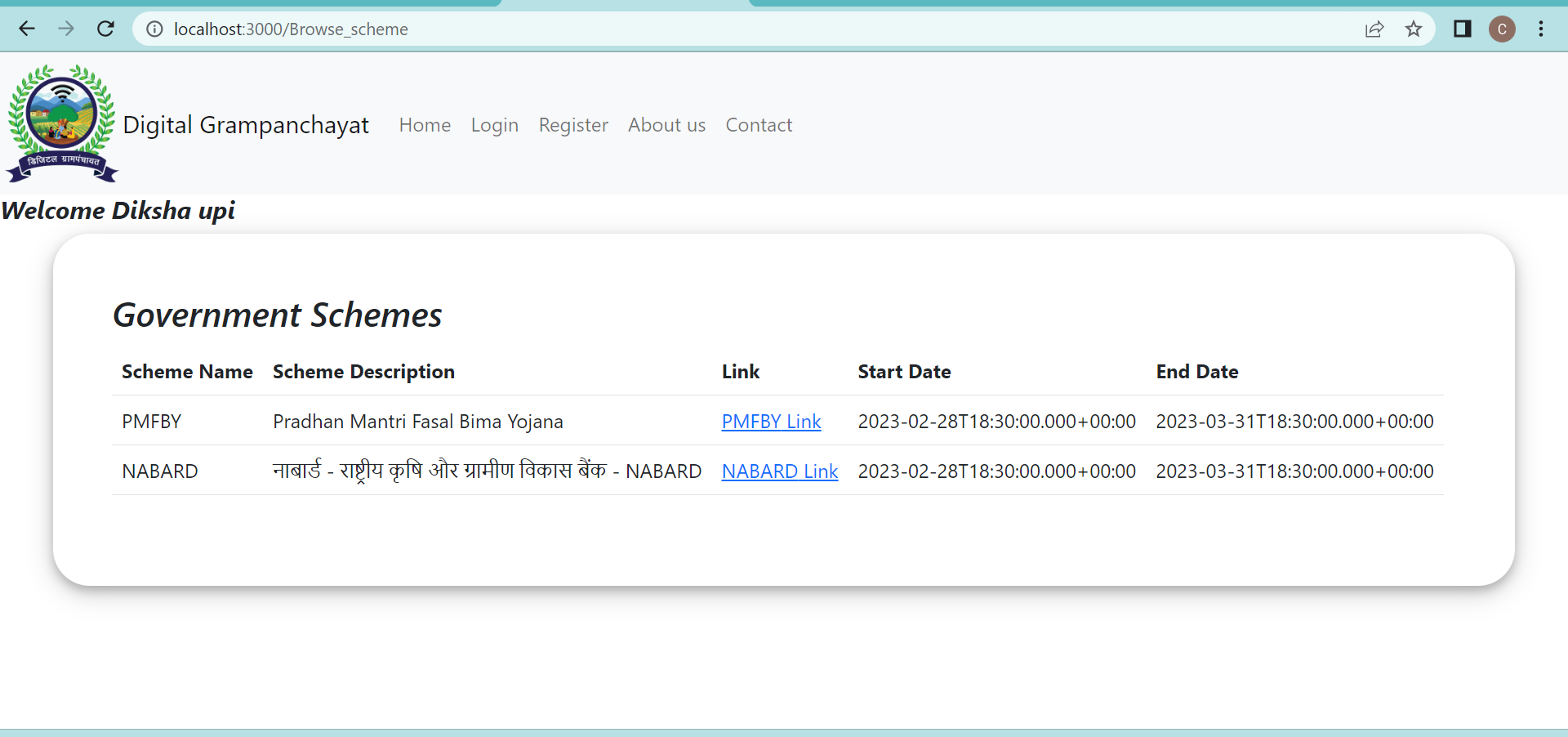
**8.17 Check Report Page**

****

**8.18 Upload Schemes Page**

****

**8.19 Show Schemes Page**

****

**9. CONCLUSION AND FUTURE SCOPE**

Paperless School Management System provides better education and platform to get connected with Teachers and Students efficiently.

Our System provides a very user-friendly platform where Student can easily read the study materials, check the attendance, raise the queries and view progress report, also Teacher will upload study materials, maintain attendance, answer queries and generate progress report.

Our system is aimed efficient management of various tasks like Generating time table for students on daily basis, Managing fees payment, Examination management. Digitally monitor daily activities along with managing all the resources and information on a single platform.