

Software Project Development On

“Government Scheme Portal”

In Partial Fulfilment of

Bachelor of Computer Application (BCA)



Gujarat University (2022-2023)
(Third Year Bachelor of Computer Application)

Project created by
GROUP NO: 02
Bhavesh Prajapati: (202012101986)
Suraj Gupta: (202012101930)
Prem Patel: (202012101978)

Rudra Goswami College of Computer Application (RGCCA)
Naroda - Dehgam Rd, Vahelal,
Ahmedabad – 382330.

Company letter



INFOLABZ IT SERVICES PVT. LTD.
WEB DEVELOPMENT | APP DEVELOPMENT | DATA SCIENCE | IOT



DATE: 13 / 07 / 2022

To,

Rudra Goswami College of Computer Application,
At Post Vahelal Naroda,
Dahegam Road, Dascroi-382330

Subject: Confirmation letter of internship.

Respected Sir/Ma'am,

We are pleased to inform you that we have granted permission to the students of your college to work on their project on "**PORTAL FOR GOVERNMENT SCHEMES**" at our premises **INFOLABZ IT SERVICES PVT LTD**. They will work on PYTHON along with Web Designing.

We will provide best guidance from our side and expect good outcomes from students as well. They will work 6 hours for learning, 4 hours for Assignments and 4 hours for project work, totaling 14 hours per week. We will issue a completion certificate once they complete the project.

Name of Students:

No	Name	Enrollment
1	Patel Prem Rajeshbhai	202012101978
2	Gupta Suraj Saktiprashad	202012101930
3	Prajapati Bhavesh Vijaybhai	202012101986

If you have any questions, please feel free to contact Ms Twinkle Shah. We are pleased you've decided to join INFOLABZ IT SERVICES PVT. LTD.



Sincerely,

Twinkle Shah (7984187048)
Senior Developer, InfoLabz, Ahmedabad



+91 8866662662
+91 8141236662



contact@infolabz.in
www.infolabz.in



405 Vraj Avenue, Above SAM'S Pizza
Nr, Commerce Six Rd, Navrangpura,
Ahmedabad, Gujarat 380009

College certificate



RUDRA GOSWAMI COLLEGE OF COMPUTER APPLICATION (BCA)

(Promoted by Shree Monark Education Trust)

(Affiliated to Gujarat University)

Website : www.rgcca.org

College Code : 745

E-mail : rudrabca745@gmail.com

Ref. No. : RGCCA /AOM /2022-23 /TR1.LET/52

Date : 8/8/2022

TO WHOMSOEVER IT MAY CONCERN

Sub: -Permission for Project of B.C.A.-3rd Year(5th/6th Semester). Students.

We are glad to introduce "**RUDRA GOSWAMI COLLEGE OF COMPUTER APPLICATION**" as a fast growing engineering institute affiliated to Gujarat University.

We appreciate positive gesture of your reputed organization which is offering Project to our student. Our following students of 5th /6th semester B.C.A. would like to take Project from your organization. We declare that the below listed student is bonafide student of our college and college has **no objection against** this student taking Project at your organization/ in your company.

- 1) PATEL PREM (202012101978)
- 2) PRAJAPATI BHAVESH (202012101986)
- 3) GUPTA SURAJ (202012101930)

We seek your help and co-operation for permitting our student in your esteemed organization. We honestly request you to issue the certificates to our student after completion of the Project successfully.

Thank you.

With warm regards,

Principal

Principal

RUDRA GOSWAMI COLLEGE OF
COMPUTER APPLICATION
At. Vahelal, Naroda-Dahegam Road,
Ta. Dascroi, Dist. Ahmedabad-382330

At. & Post : Vahelal, Naroda-Dahegam Road, Ta. : Dascroi, Dist. : Ahmedabad-382 330. (Gujarat)
Phone No. : (02718) 247138, Fax No. 02718-247131, M : 9824561940, 9924125030

Acknowledgement

- Towards of my Bachelor of Computer Application project training in Rudra Goswami College Of Computer Application, Ahmedabad, we feel greatly obliged to certain Specials.
- We are sincerely gratitude to **Prof. VANDANA MA'AM** (Internal Guide) of Rudra Goswami College Of Computer Application, Ahmedabad, who providing me this opportunity and necessary support during the Analysis of the project. We are also grateful to all staff members of this company for providing all the technical assistance and helping in solving problems arising during Analysis.
- We have no other words to express my sincere thanks to all my lecturers of INFOLABZ IT SERVICE, Ahmedabad for their kind co-operation and able guidance. Especially to **Ms. TWINKLE SHAH MA'AM** my External project guide in Internship.

INDEX

CHAPTER 1 :- INTRODUCTION	7
1.1 COMPANY PROFILE	8
1.2 PROJECT DEFINITION	9
CHAPTER 2 :- ANALYSIS	10
2.1 EXISTING SYSTEM.....	11
2.2 NEED FOR NEW SYSTEM	11
CHAPTER 3 :- SCREEN LAYOUT	12
3.1 USER SIDE SCREEN LAYOUT BEFORE USER LOGIN	12
INDEX PAGE HEADER	12
INDEX PAGE BODY	13
FOOTER	14
SCHEMES PAGE.....	15
SCHEMES DETAIL.....	16
ABOUT PAGE	17
CONTACT PAGE	18
LOGIN AND SING-UP PAGE	19
3.2 USER SIDE SCREEN LAYOUT AFTER USER LOGIN	20
INDEX PAGE.....	20
SCHEMES PAGE.....	22
SCHEMES DETAIL.....	24
APPLIED SCHEMES VIEW	25
RELEVANT SCHEMES	26
FEED-BACK PAGE.....	27
ADD YOUR DETAIL	28
ABOUT PAGE	29
CONTACT PAGE	30
3.3 ADMIN SIDE SCREEN LAYOUT	31
ADMIN LOGIN PAGE	31
DASHBOARD	32

AADHAR-CARD DATA (filter)	33
APPLICATION DATA	34
CONTACT DATA.....	35
FEEDBACKS DATA	36
POLICY DATA	37
USER SIGN-UP DATA	38
ALL ADMIN DATA	39
PASSWORD CHANGE PAGE.....	40
CHAPTER 4 :- SAMPLE CODING.....	41
BACKEND - CODEING	41
FRONTEND - CODING	45
CHAPTER 5:- FUTURE ENHANCEMENT	47
Points	47
CHAPTER 6:- CONCLUSION	48
Points	48
CHAPTER 7 :- BIBLIOGRAPHY	49

CHAPTER 1 :- INTRODUCTION



1.1 COMPANY PROFILE



INFOLABZ IT SERVICES AHMEDABAD

COMPANY OVERVIEW

Established in 2016, incorporation with our patent IT company, INFOLABZ has managed to make it's own position in IT Sector. We are involved in Web Development, App Development, Progressive Web Application Development, IOT solutions, Graphics & Designing, Corporate & Project training, Digital Marketing, Domain & Hosting services, SMS services etc

In the span of four years we have managed to deliver all projects on time with utmost accuracy to our clients across the globe. We have dedicated teams of experienced and hard working developers, our developers who are always willing to take new challenges and looking forward to learn new things, are heart of this company.

WE ARE COMMITTED TO...

- ON TIME DELIVERY
- QUALITY WORK
- TECHNOLOGICALLY ADVANCE WORK
- BEST OUTPUT FROM OUR STRONG DEVELOPERS TEAM
- PROVIDE BEST TECHNICAL EDUCATION
- MAKE STUDENTS INDUSTRY READY

MS. TWINKLE SHAH

PROJECT CO-ORDINATOR | 7984187048

405, VRAJ AVENUE, ABOVE SAM'S PIZZA
NR. COMMERCE SIX ROAD, NAVRANGPURA
AHMEDABAD, 380009



**APPLICATION
DEVELOPMENT**



**WEBSITE
DEVELOPMENT**



**IOT
SOLUTIONS**



**PLACEMENTS &
TRAINING**



**DIGITAL
MARKETING**

8866662662

WWW.INFOLABZ.IN

CONTACT@INFOLABZ.IN

1.2 PROJECT DEFINITION

Title: Government Scheme Portal

- This website is for Needy person who are unaware about government existing schemes. Our website will help those people who need these schemes very much.
- Our website will also notify about upcoming government schemes in which the people can enroll and take benefit of those schemes.
- Our website will also help those people who are desperately in need of government and cannot get the right place to complain or get help from.
- Our website will guide them to there nearby AMC office from there they can get help from.

CHAPTER 2 :- ANALYSIS



2.1 EXISTING SYSTEM

The existing system is manual and the manual system works in the following way :

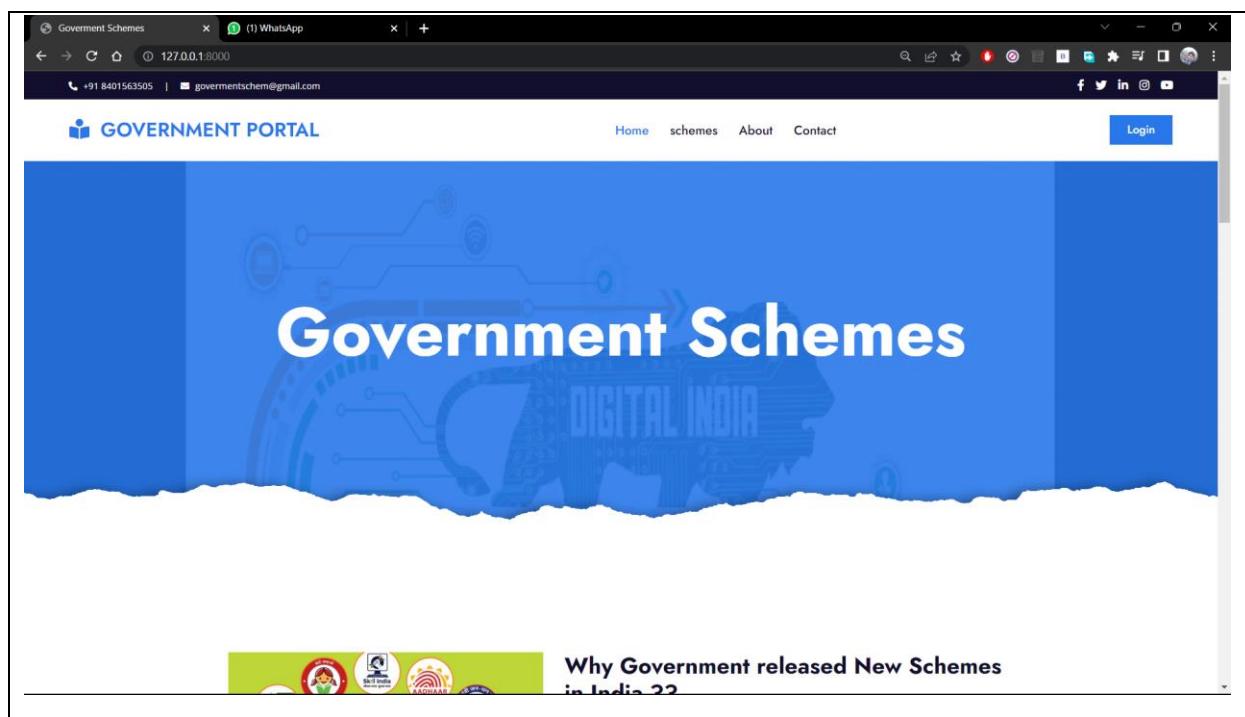
- Current system is a manual one in every panchayat displays the list of available schemes provided by the Government.
- It invites the applications from the eligible users based on certain criteria .
- All the applications are verified at panchayat level and then approved applications are sent to District level for approval.
- All the approved users avail the benefits of the scheme.
- Its takes more time to complete the process.

2.2 NEED FOR NEW SYSTEM

- Our website will help people who are in need of various Government schemes.
- our website is made for those people who need benefit of such scheme and don't want to go through lot's of creepy website .
- this website has also a feature where a person who find a specific scheme very helpful the person can apply using online form by provided by GOI.
- this website will also provide your nearest municipal office address, In case the person need any further assistance regarding any scheme.

CHAPTER 3 :- SCREEN LAYOUT

3.1 USER SIDE SCREEN LAYOUT BEFORE USER LOGIN



INDEX PAGE BODY

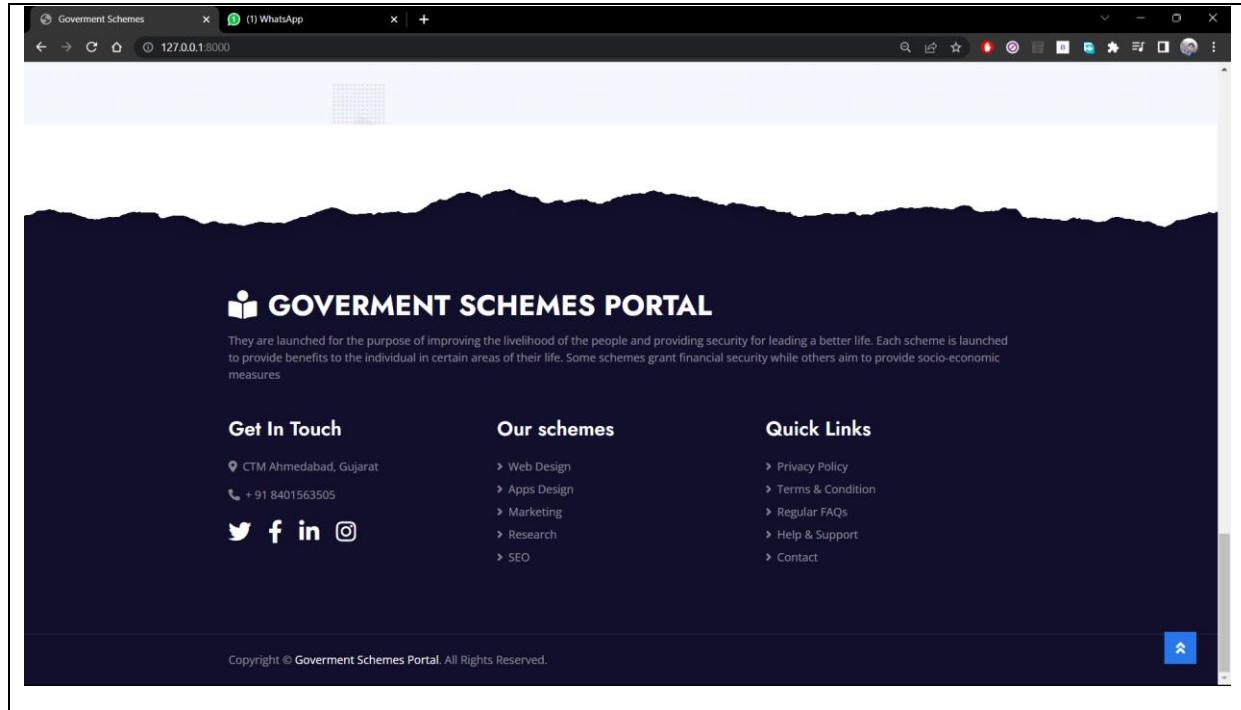
The screenshot shows a web browser window with two tabs: "Government Schemes" and "(2) WhatsApp". The main content area features a large green circular graphic titled "GOVERNMENT SCHEMES, MISSIONS, CAMPAIGNS & PROGRAMMES IN INDIA FOR CIVIL SERVICES EXAMINATIONS". The graphic is surrounded by various small icons representing different government programs like Pradhan Mantri Awas Yojana, National Health Mission, etc. To the right of the graphic, the text "Why Government released New Schemes in India ??" is displayed, followed by a detailed paragraph explaining the reasons. Below this, there is a summary table with four colored boxes: green (3 Agriculture), blue (1 Education), red (2 Health), and yellow (4 Finance). The URL in the address bar is 127.0.0.1:8000.

The screenshot shows a web browser window with two tabs: "Government Schemes" and "(1) WhatsApp". The main content area features a section titled "OUR SCHEMES" with a sub-section header "Checkout New Releases Schemes". Below this, there are four cards representing different government schemes:

- Kisan Credit Card Scheme**: Features an image of farmers in a field, a Kisan Credit Card logo, and the text "Farmers".
- National Health Mission**: Features an image of a family, a sun icon, and the text "Poor and vulnerable populations".
- National Rural Livelihood Mission (NRLM)**: Features an image of rural households, a green and orange circular logo, and the text "Rural households".
- Pradhan Mantri Awas Yojana-Urban**: Features an image of a stylized person, the text "Urban poor", and the text "प्रधान मंत्री आवास योजना-शहरी Pradhan Mantri Awas Yojana-Urban".

The URL in the address bar is 127.0.0.1:8000.

FOOTER



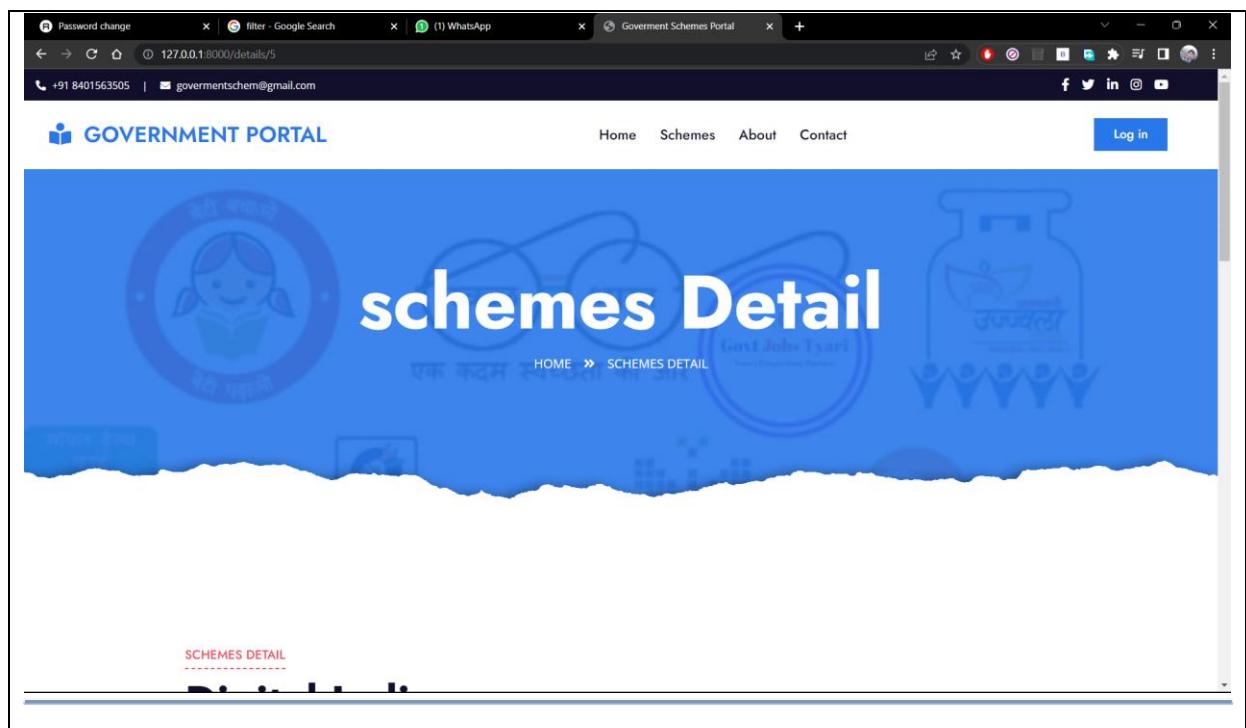
→ SAME FOOTER IN ALL PAGE

SCHEMES PAGE

The screenshot shows a web browser window with the URL 127.0.0.1:8000/schemes. The page has a blue header with the text "GOVERNMENT PORTAL" and "Login". Below the header is a large banner featuring a circular logo with a child reading a book, the word "schemes", and the text "एक कदम स्वयंसेवा की ओर" (One step towards self-help). The main content area has a title "Checkout Releases Government Schemes" and a sub-header "127.0.0.1:8000/schemes".

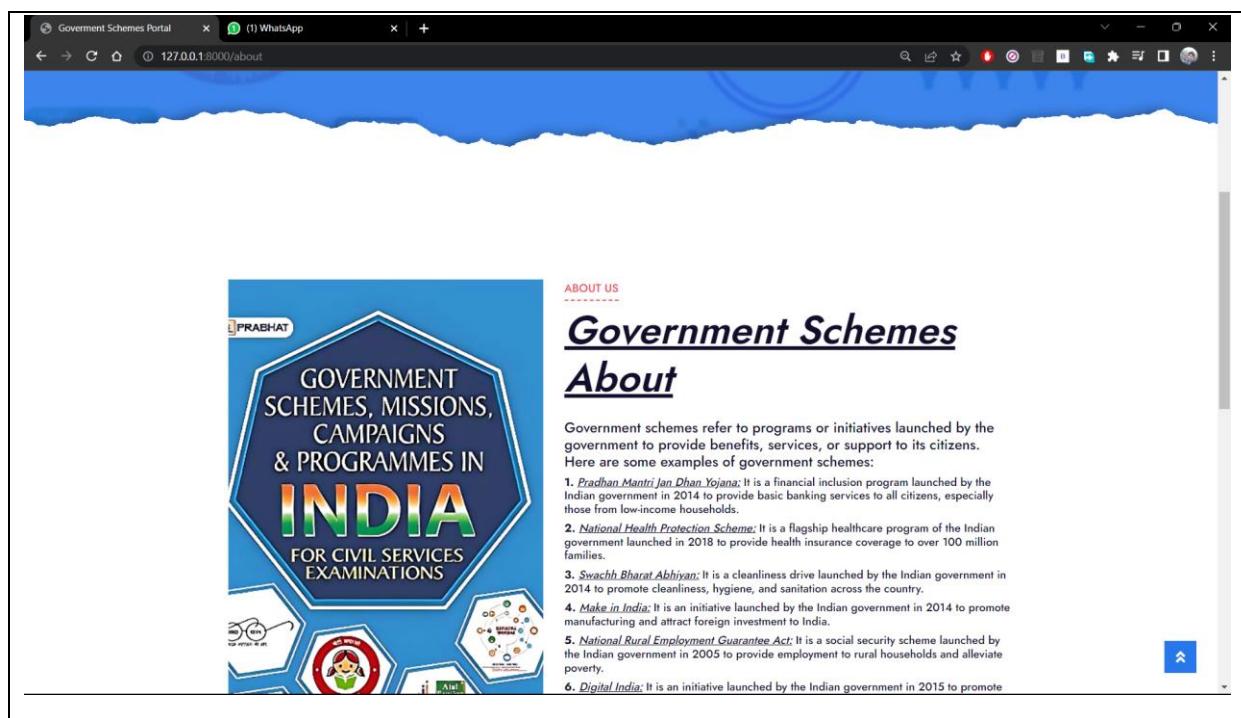
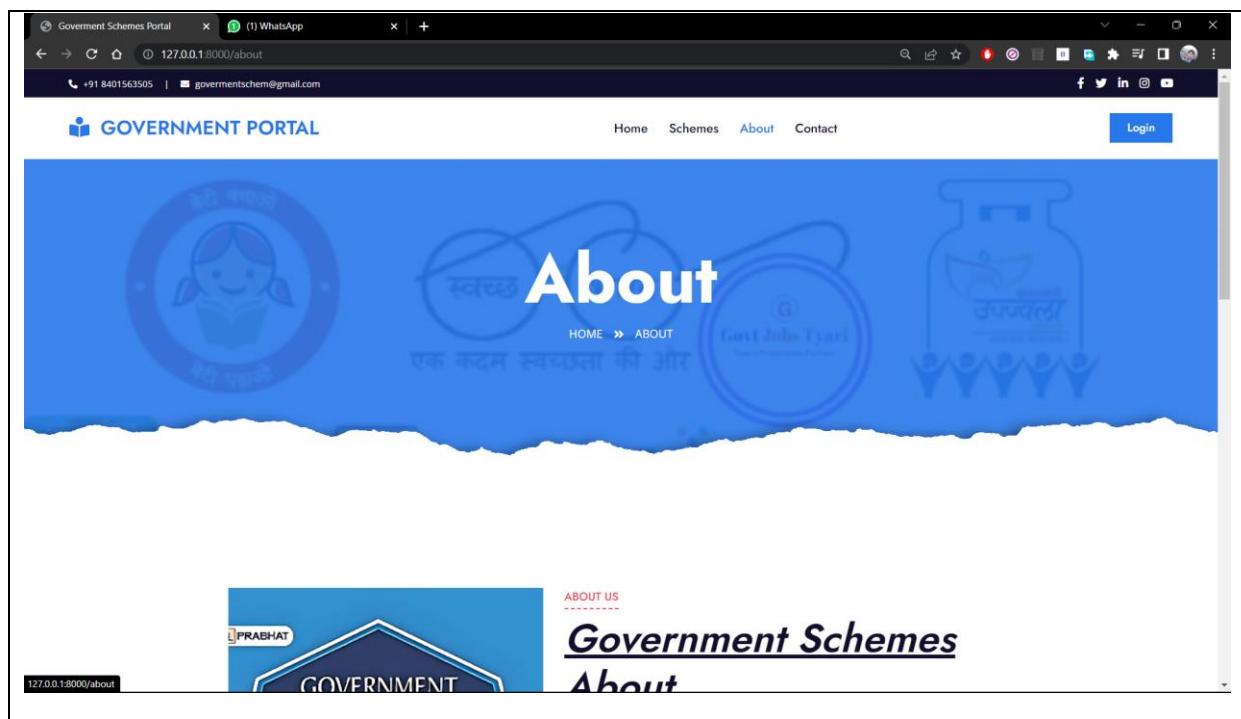
The screenshot shows a web browser window with the URL 127.0.0.1:8000/details/2. It displays five cards representing different government schemes: Atal Pension Yojana (Finance), Beti Bachao Beti Padhao (Education), Deen Dayal Upadhyaya Grameen Kaushalya Yojana (Skill Development and Entrepreneurship), Digital India (Digital India), and Kisan Credit Card Scheme (Kisan Credit Card Scheme). The National Health Mission logo is also visible in the bottom right corner.

SCHEMES DETAIL

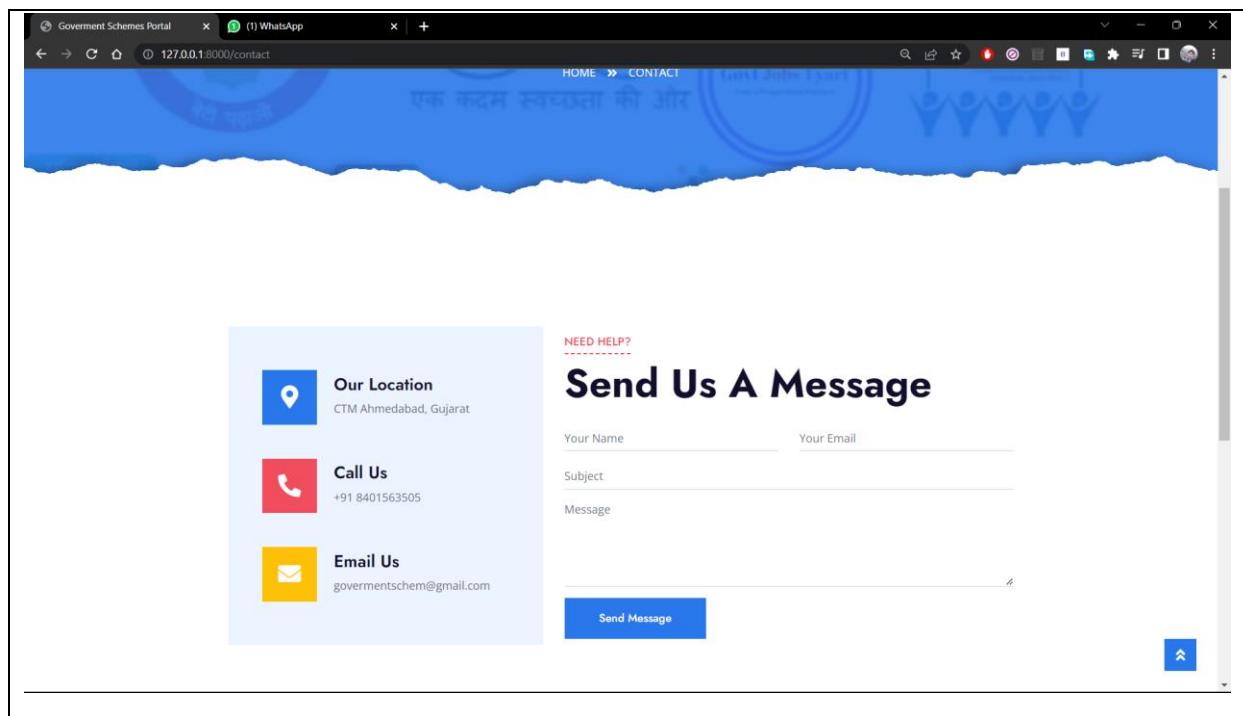
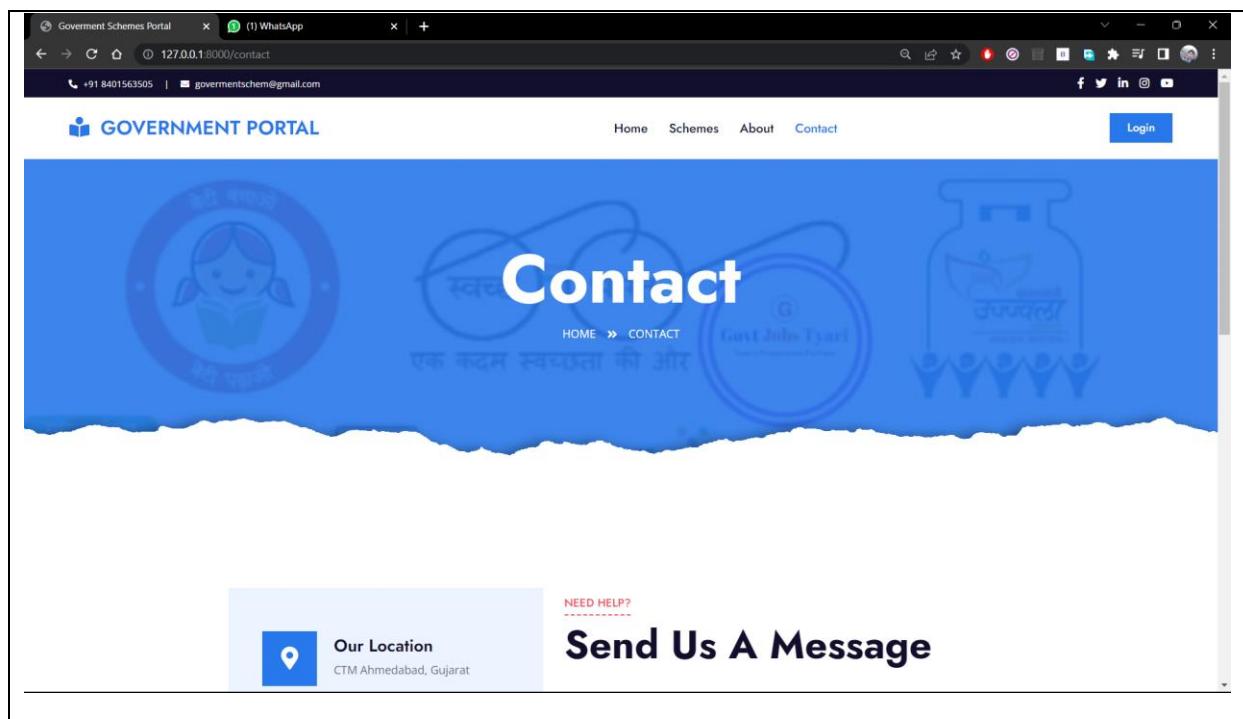


A screenshot of a web browser window titled "Government Schemes Portal". The URL in the address bar is "127.0.0.1:8000/details/5". The page displays a detailed view of the "Digital India" scheme. The title "Digital India" is prominently displayed at the top. Below the title, there are several sections with headings and descriptions: "Policy Description" (This scheme aims to digitize the country's infrastructure and provide online services to citizens.), "Suitable for" (All individuals), "Policy Type" (Technology), "Policy Agency" (Ministry of Electronics and Information Technology), "Policy Eligible Castes" (All Castes), "Policy Applicable State" (All States), "Policy Residence Area" (Both), and "Policy Disability Status" (Nn). A small blue arrow icon is located in the bottom right corner of the page content area.

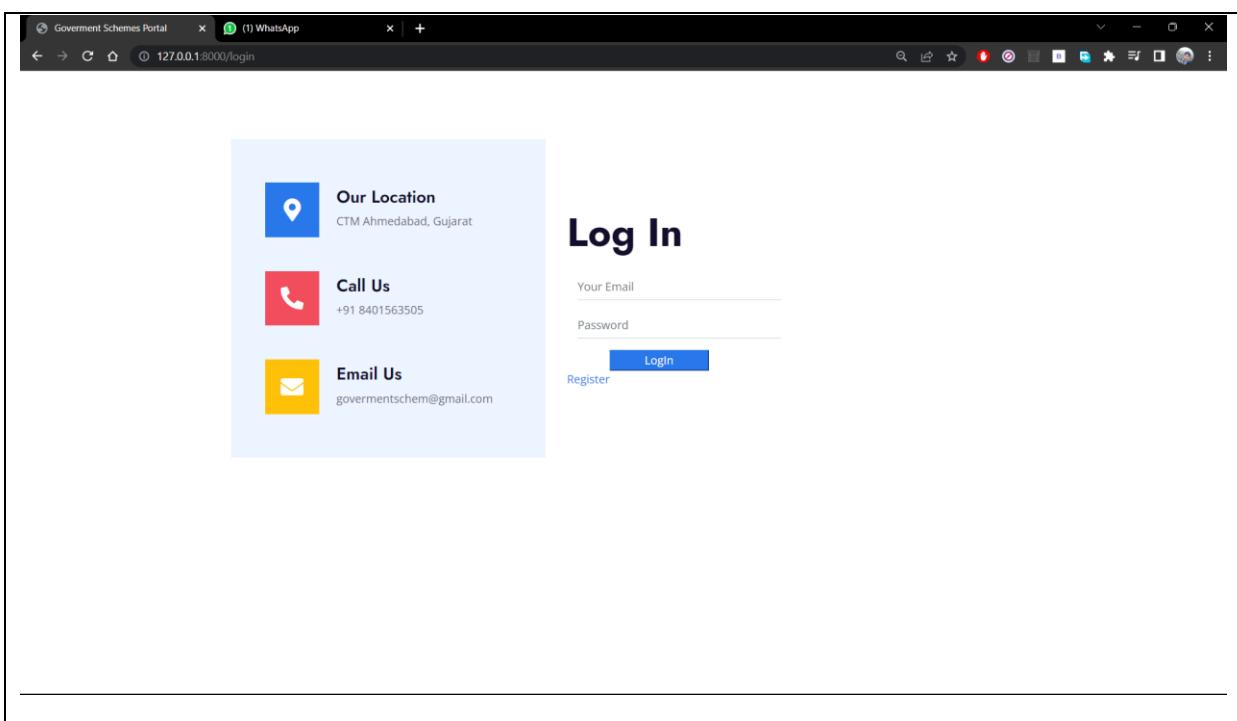
ABOUT PAGE



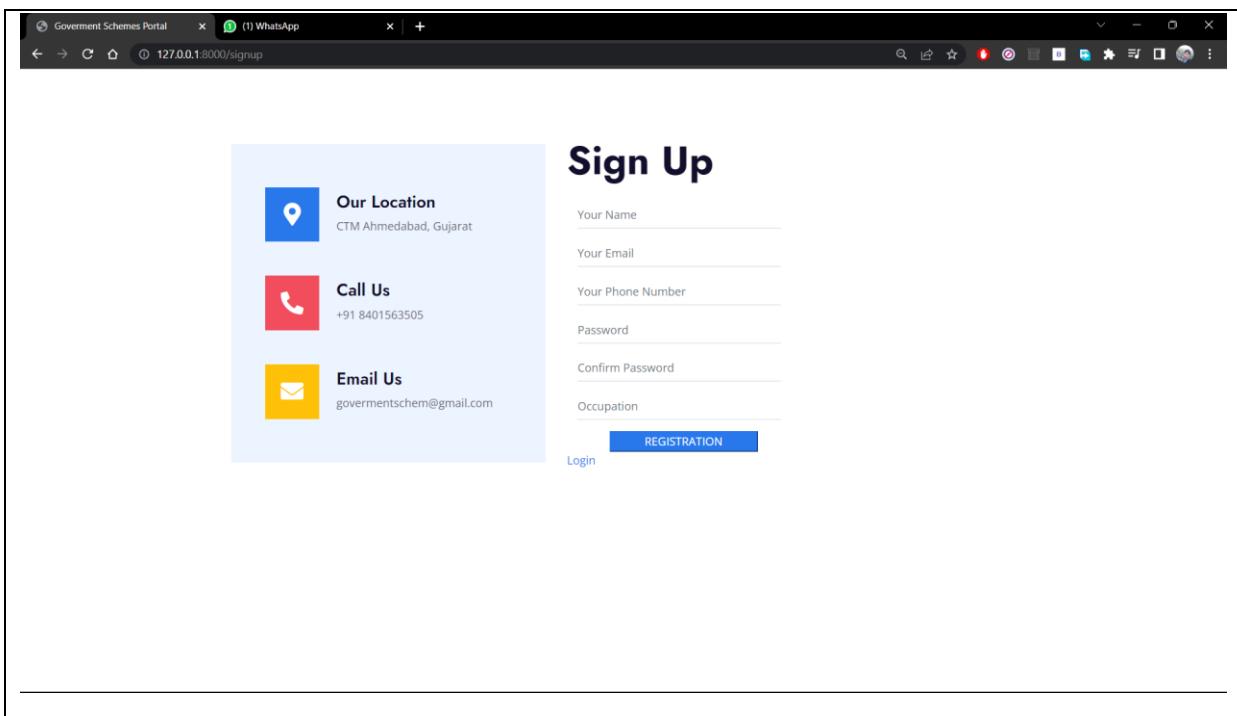
CONTACT PAGE



LOGIN AND SING-UP PAGE

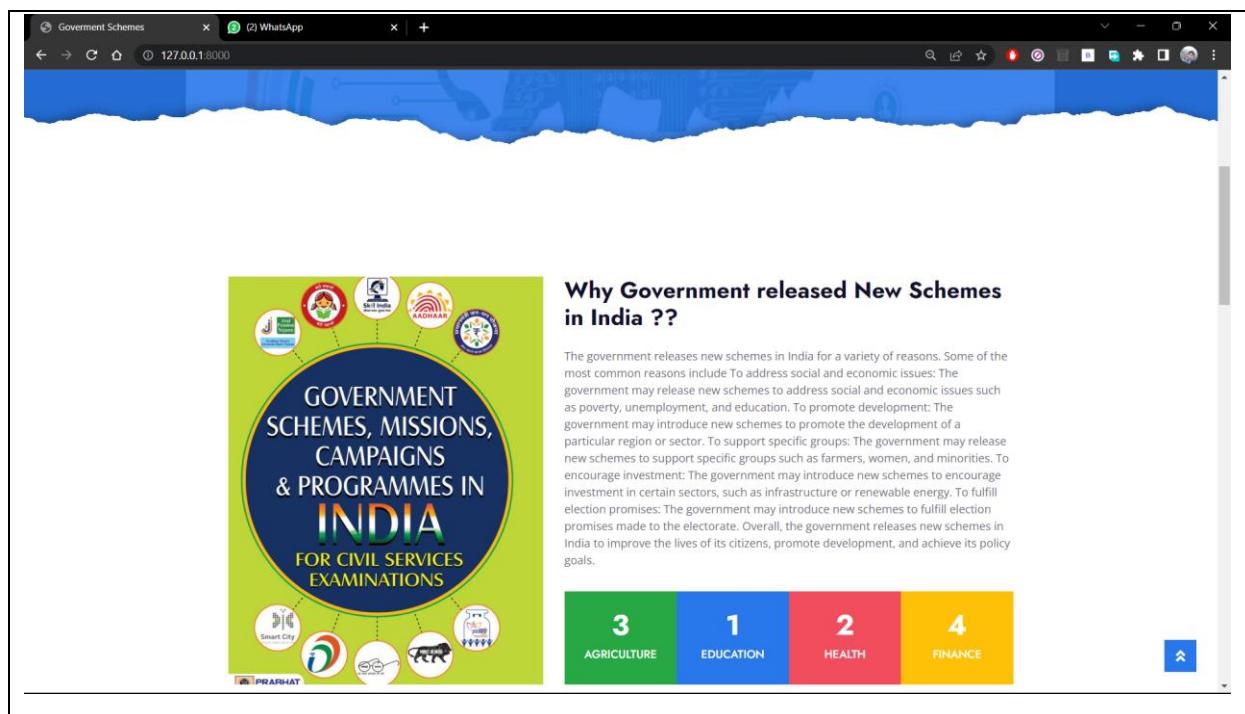
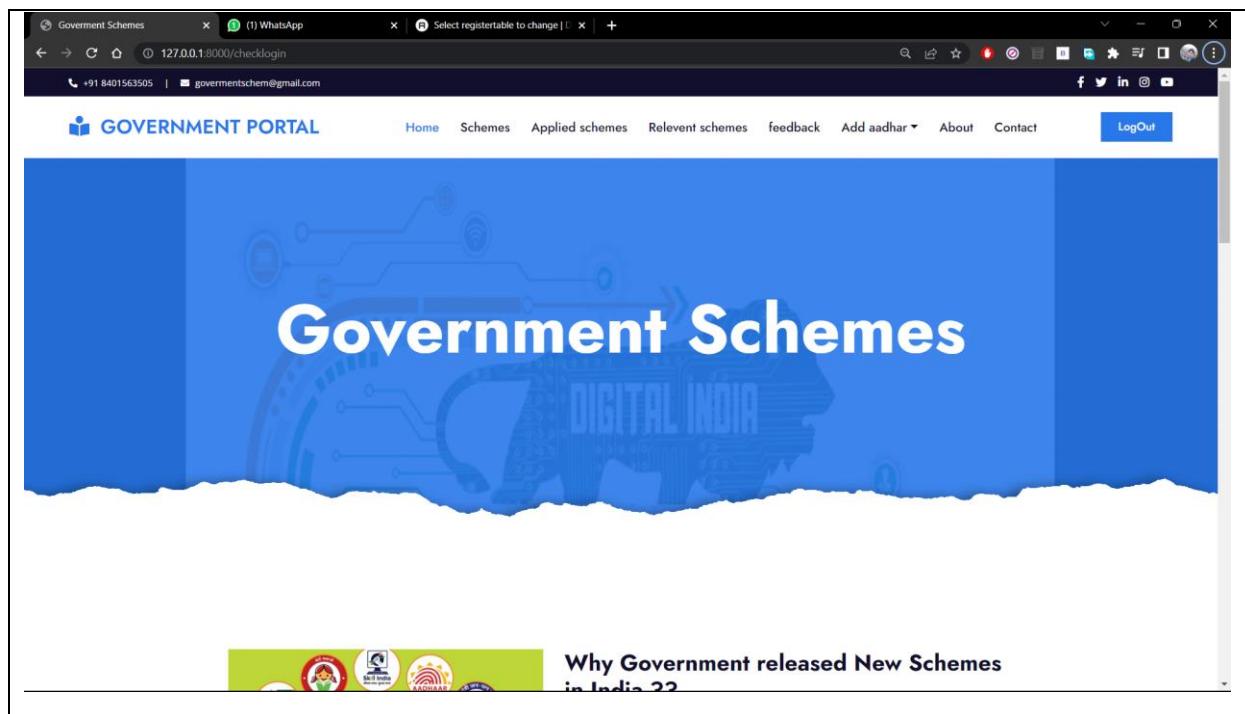


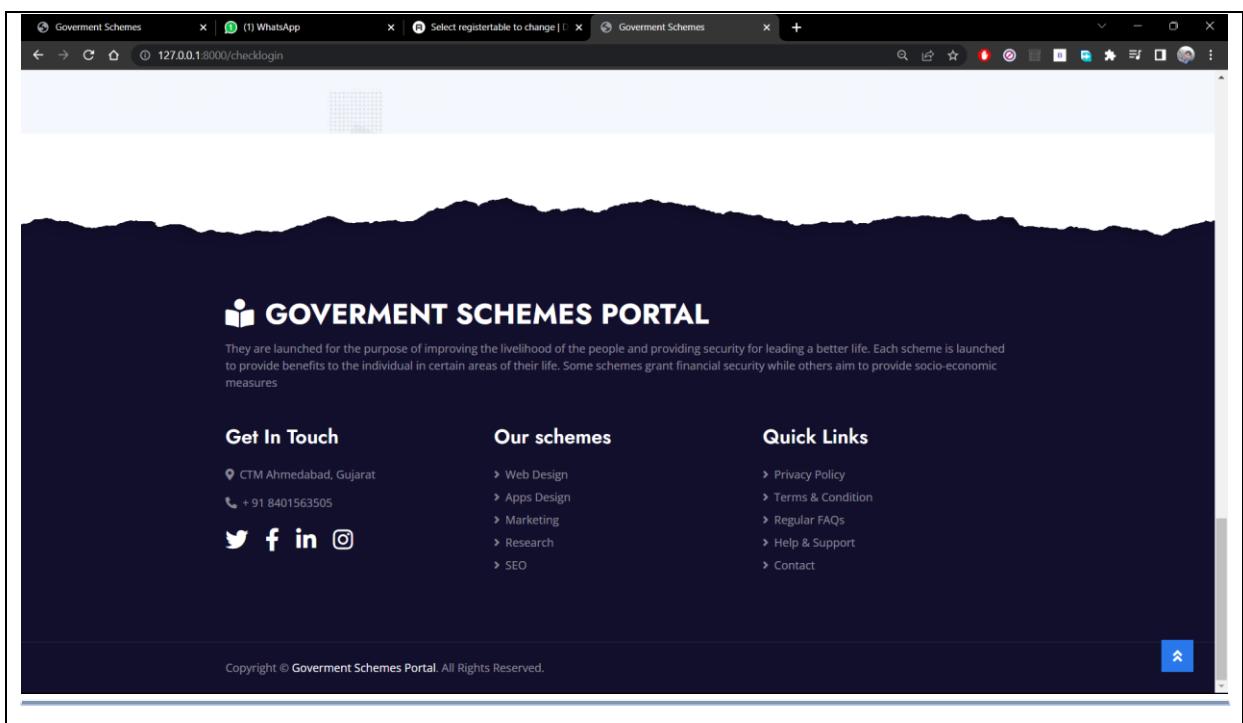
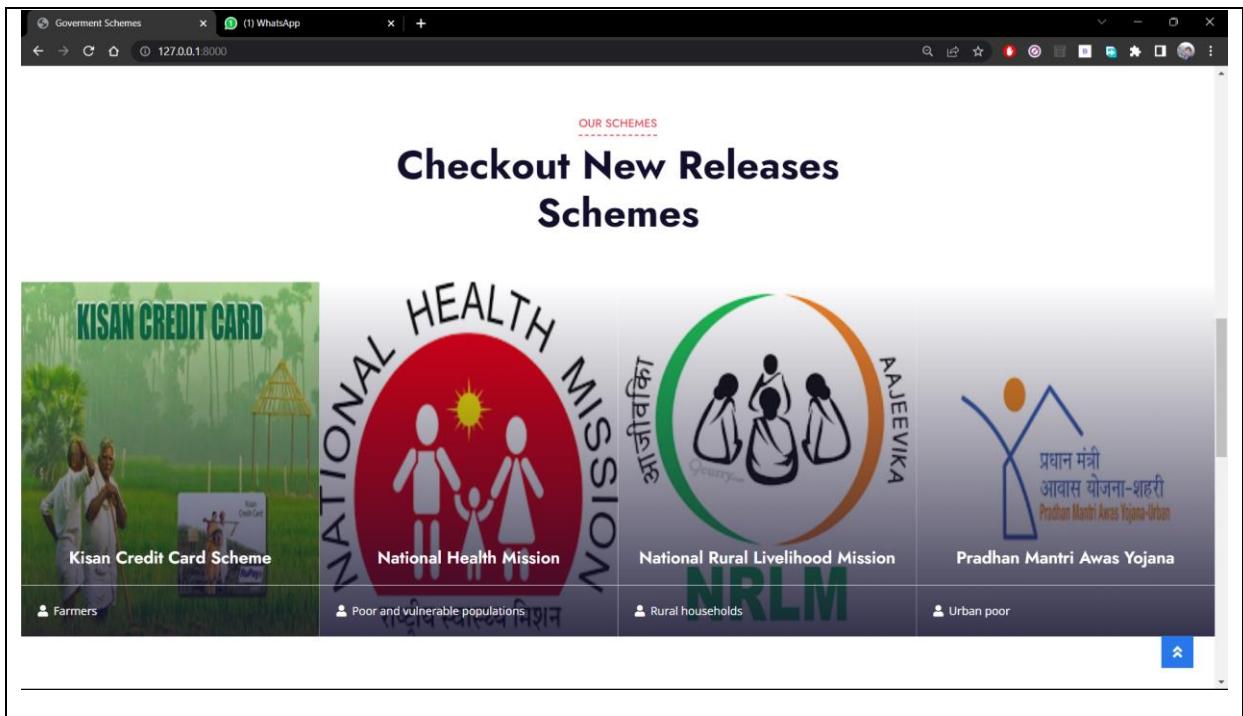
The screenshot shows a web browser window with two tabs: "Government Schemes Portal" and "(1) WhatsApp". The active tab is "127.0.0.1:8000/login". The page has a header "Log In" and a sidebar with contact information: "Our Location" (CTM Ahmedabad, Gujarat), "Call Us" (+91 8401563505), and "Email Us" (governmentschem@gmail.com). The main form fields are "Your Email" and "Password", with a "Login" button and a "Register" link.



The screenshot shows a web browser window with two tabs: "Government Schemes Portal" and "(1) WhatsApp". The active tab is "127.0.0.1:8000/signup". The page has a header "Sign Up" and a sidebar with contact information: "Our Location" (CTM Ahmedabad, Gujarat), "Call Us" (+91 8401563505), and "Email Us" (governentschem@gmail.com). The main form fields are "Your Name", "Your Email", "Your Phone Number", "Password", "Confirm Password", and "Occupation", with a "REGISTRATION" button and a "Login" link.

3.2 USER SIDE SCREEN LAYOUT AFTER USER LOGIN

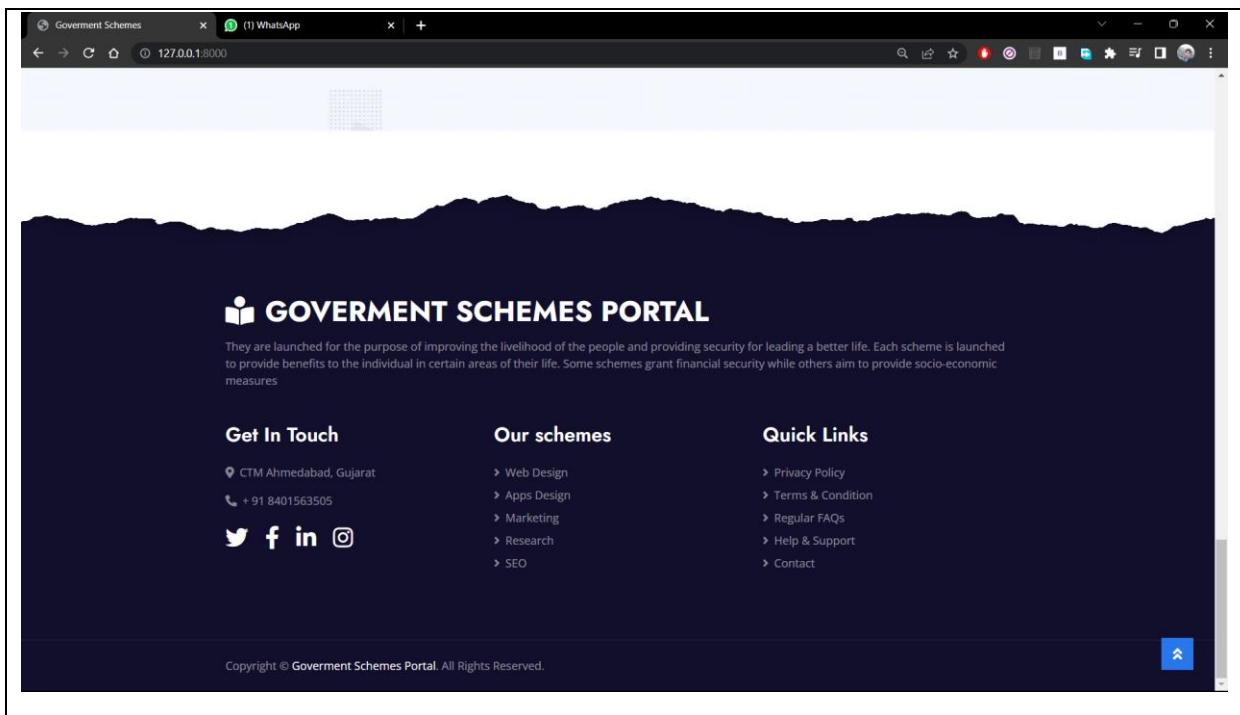
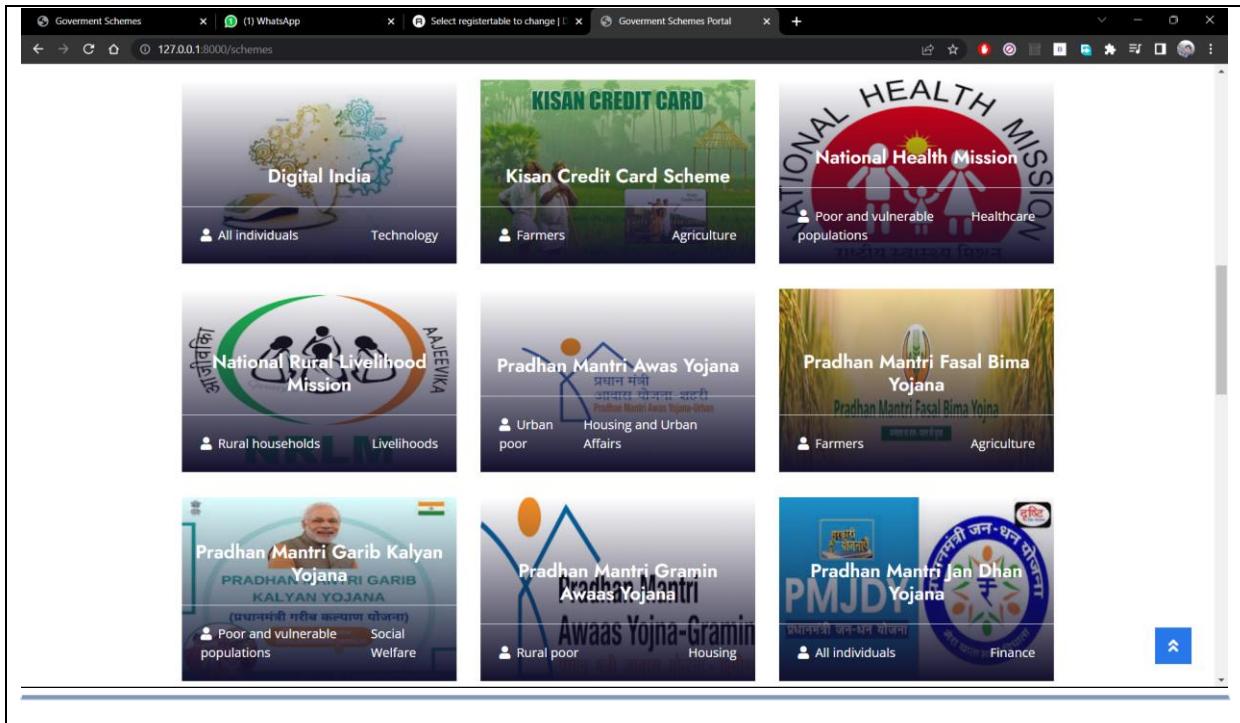




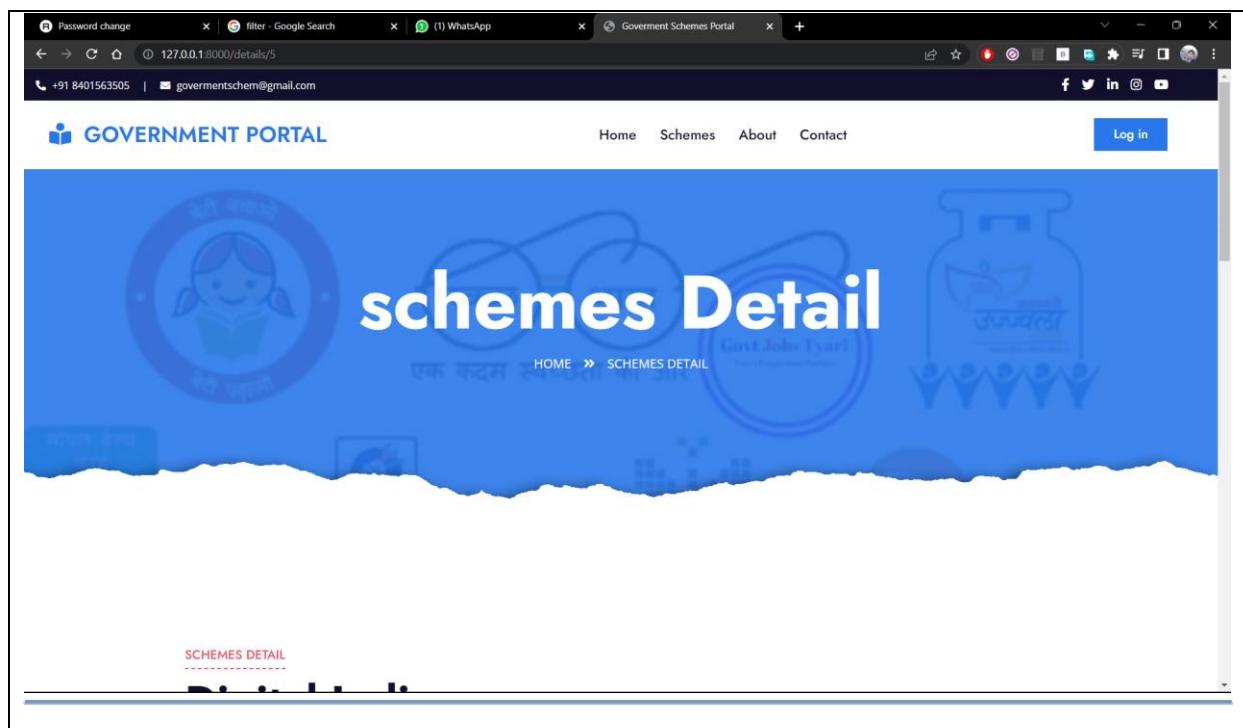
SCHEMES PAGE

The screenshot shows a web browser window with the URL 127.0.0.1:8000/schemes. The page has a blue header with the text "GOVERNMENT PORTAL" and navigation links for Home, Schemes, Applied schemes, Relevant schemes, feedback, Add aadhar, About, Contact, and LogOut. The main content area features a large blue banner with the word "schemes" in white. Below the banner, there are three circular icons representing different government programs: "Atal Pension Yojana" (a person holding a pension card), "Beti Bachao, Beti Padhao" (two girls holding hands), and "Deen Dayal Upadhyaya Grameen Kaushalya Yojana" (a group of people). The banner also includes the text "एक कदम सरकारी योजना" (One Step Government Scheme) and "Govt Jobs Tyari". At the bottom of the page, the text "Checkout Releases" is displayed.

The screenshot shows a web browser window with the URL 127.0.0.1:8000/schemes. The main content area features a large title "Checkout Releases Government Schemes" in bold black text. Below the title, there are four cards representing different government schemes: "Atal Pension Yojana" (Unorganized sector workers - Finance), "Beti Bachao, Beti Padhao" (Girls Bachao-Beti Padhao - Education), "Deen Dayal Upadhyaya Grameen Kaushalya Yojana" (Rural youth - Skill Development and Entrepreneurship), and "KISAN CREDIT CARD" (Agriculture). A partial view of the "AL HEALTH" scheme is visible at the bottom right.



SCHEMES DETAIL



A screenshot of a web browser displaying the 'Digital India' policy details. The title bar shows multiple tabs including 'Password change', 'filter - Google Search', '(1) WhatsApp', 'Government Schemes Portal', and others. The address bar shows the URL '127.0.0.1:8000/details/5'. The page title is 'SCHEMES DETAIL' followed by 'Digital India'. The main content area includes sections for 'Policy Description' (digitizing infrastructure and providing online services), 'Suitable for' (all individuals), 'Policy Type' (Technology), 'Policy Agency' (Ministry of Electronics and Information Technology), 'Policy Eligible Castes' (All Castes), 'Policy Applicable State' (All States), 'Policy Residence Area' (Both), and 'Policy Disability Status' (No). A blue arrow icon is visible in the bottom right corner of the content area.

APPLIED SCHEMES VIEW

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "Goverment Schemes Portal" and displays the URL "127.0.0.1:8000/viewpolicy". The page header includes a "GOVERNMENT PORTAL" logo and navigation links for Home, Schemes, Applied schemes, Relevant schemes, feedback, Add aadhar, About, Contact, and LogOut. The main content area has a blue background featuring icons related to government schemes like a child reading and a family. The title "Applied Schemes" is prominently displayed in white. Below it, there is a "HOME" link and some smaller text in Hindi. A central message reads "Viewed Your Applied Policy's". Below this, the policy name "Policy name : Beti Bachao Beti Padhao" is listed. The address bar at the top also shows the URL "127.0.0.1:8000/viewpolicy".

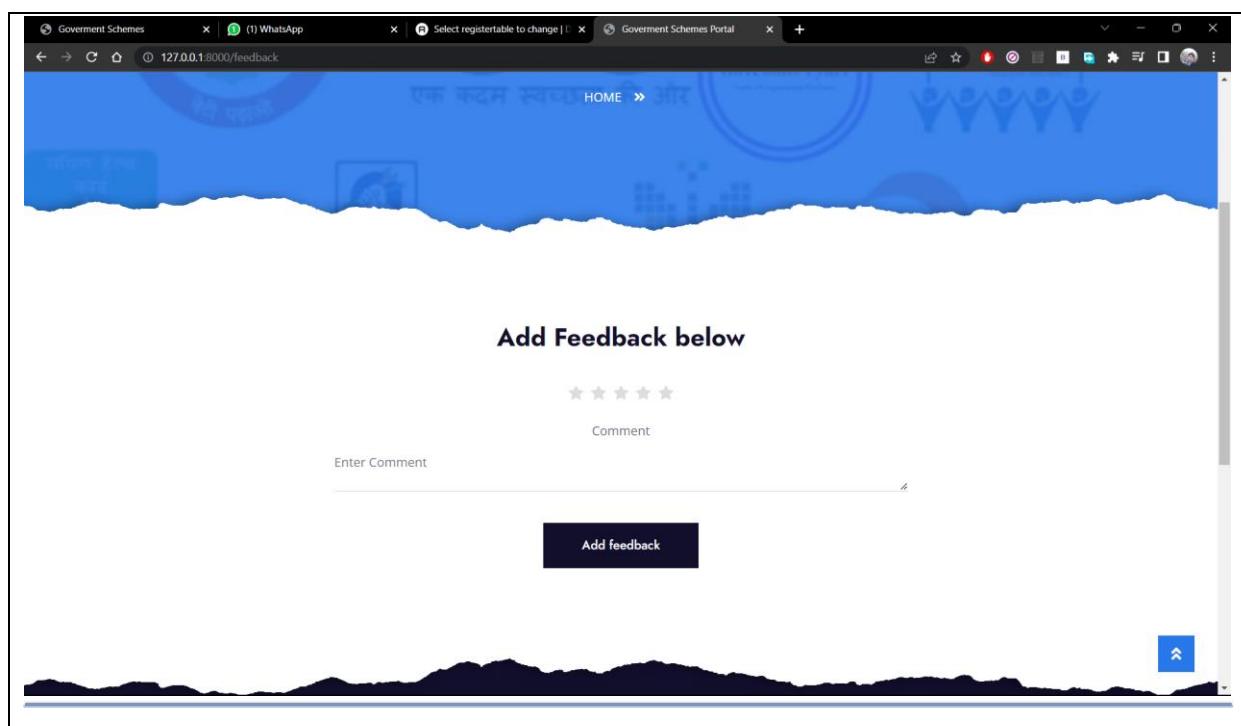
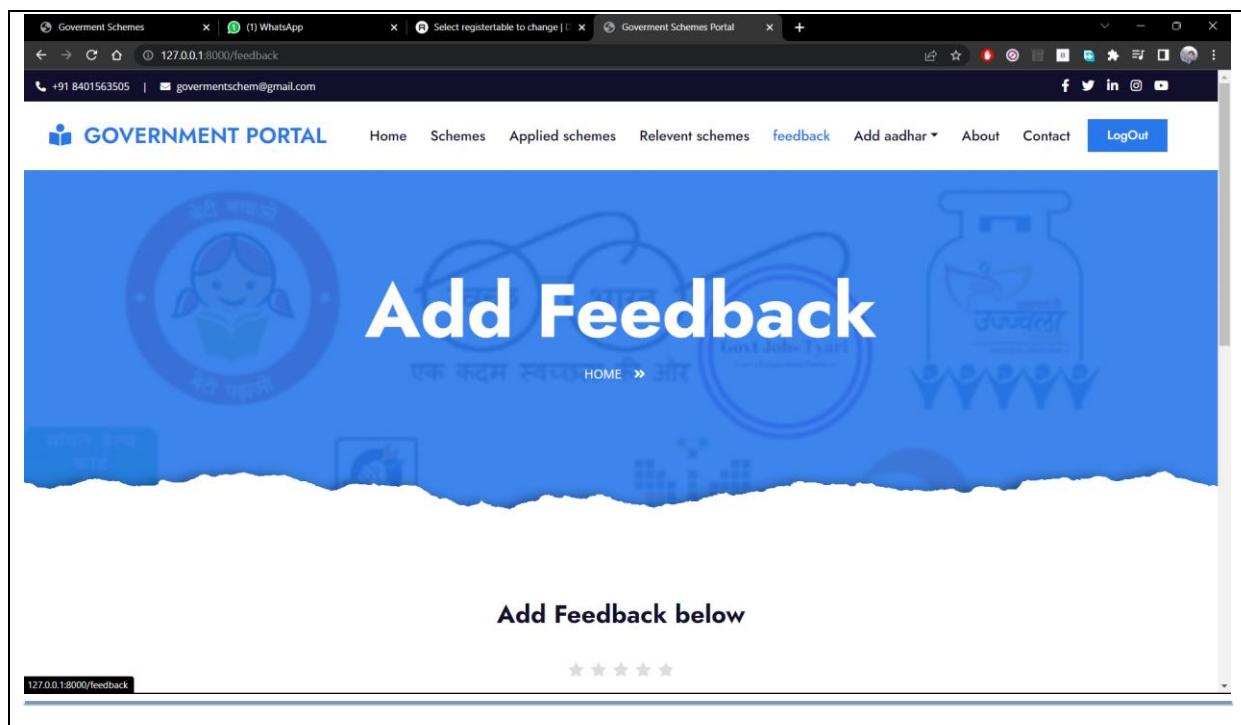
This screenshot shows the same "Applied Schemes View" page as the previous one, but with additional user-specific information. It displays the "User Name : Newuser" and a blue "ADD Policy" button. The rest of the page structure, including the header, title, and main message, remains the same.

RELEVANT SCHEMES

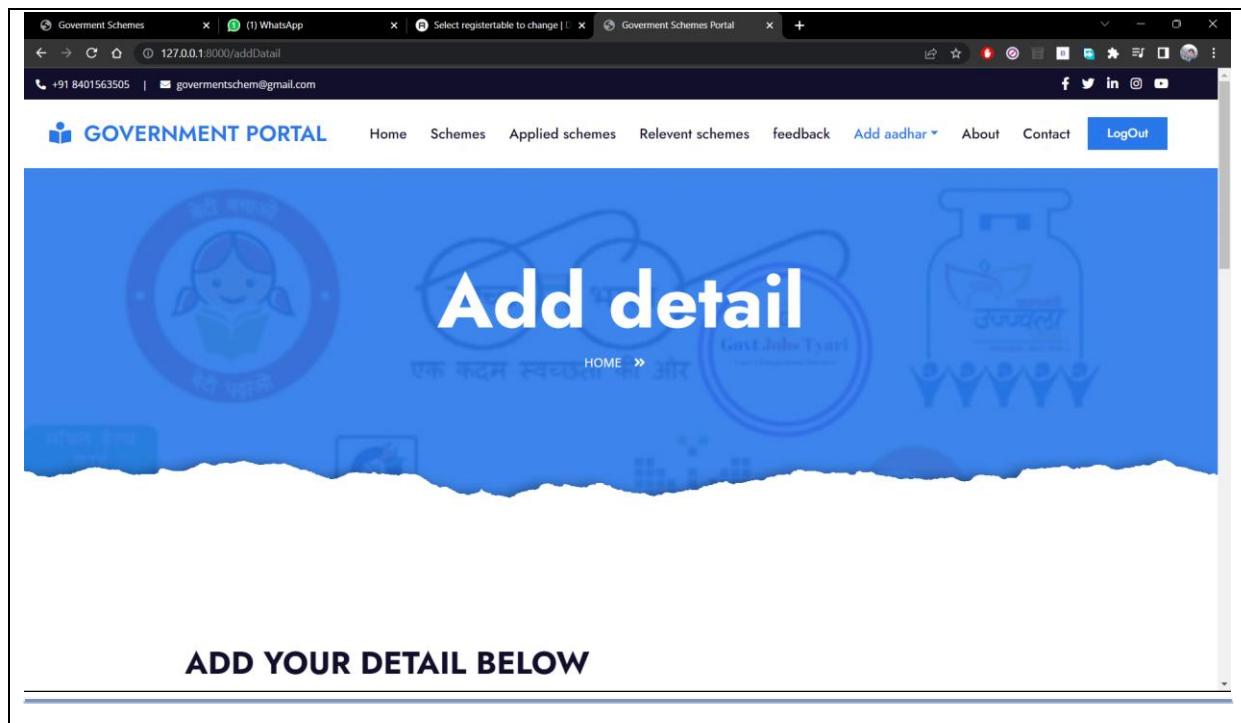
The screenshot shows a web browser window with multiple tabs open. The active tab is titled 'Relevant Schemes' and has the URL '127.0.0.1:8000/releventpolicy'. The page header includes a logo for 'GOVERNMENT PORTAL' and navigation links for Home, Schemes, Applied schemes, Relevant schemes, feedback, Add aadhar, About, Contact, and LogOut. The main content area features a large blue banner with the text 'Relevant Schemes' in white. Below the banner, there's a red link labeled 'OUR POLICIES' and a URL '127.0.0.1:8000/releventpolicy'.

The screenshot shows a web browser window with multiple tabs open. The active tab is titled 'Select registrable to change' and has the URL '127.0.0.1:8000/releventpolicy'. The page header includes a logo for 'GOVERNMENT PORTAL' and navigation links for Home, Schemes, Applied schemes, Relevant schemes, feedback, Add aadhar, About, Contact, and LogOut. The main content area features a red link labeled 'OUR POLICIES' and a large black text 'Checkout your relevant Policies'. Below this, there are three cards: 'Kisan Credit Card Scheme' (with an image of two people in a field), 'Pradhan Mantri Fasal Bima Yojana' (with an image of a green field and a logo), and 'Pradhan Mantri Krishi Sinchai Yojana' (with an image of Prime Minister Narendra Modi and a field being irrigated).

FEED-BACK PAGE

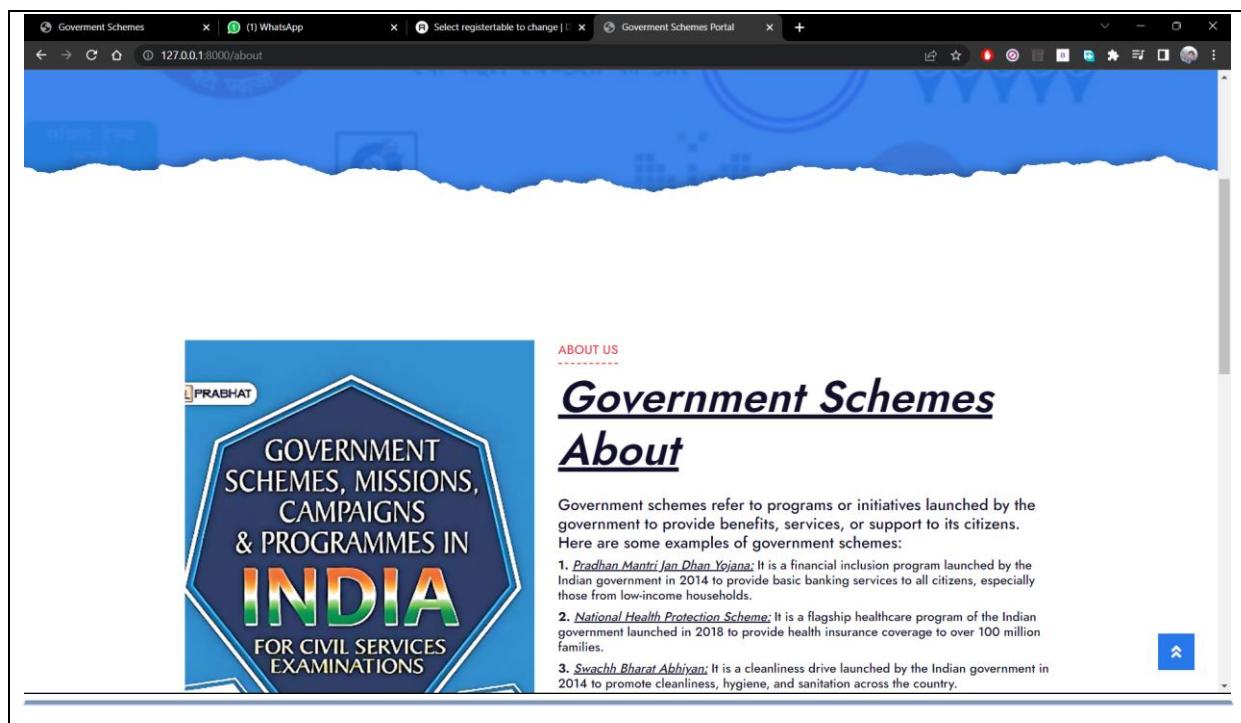
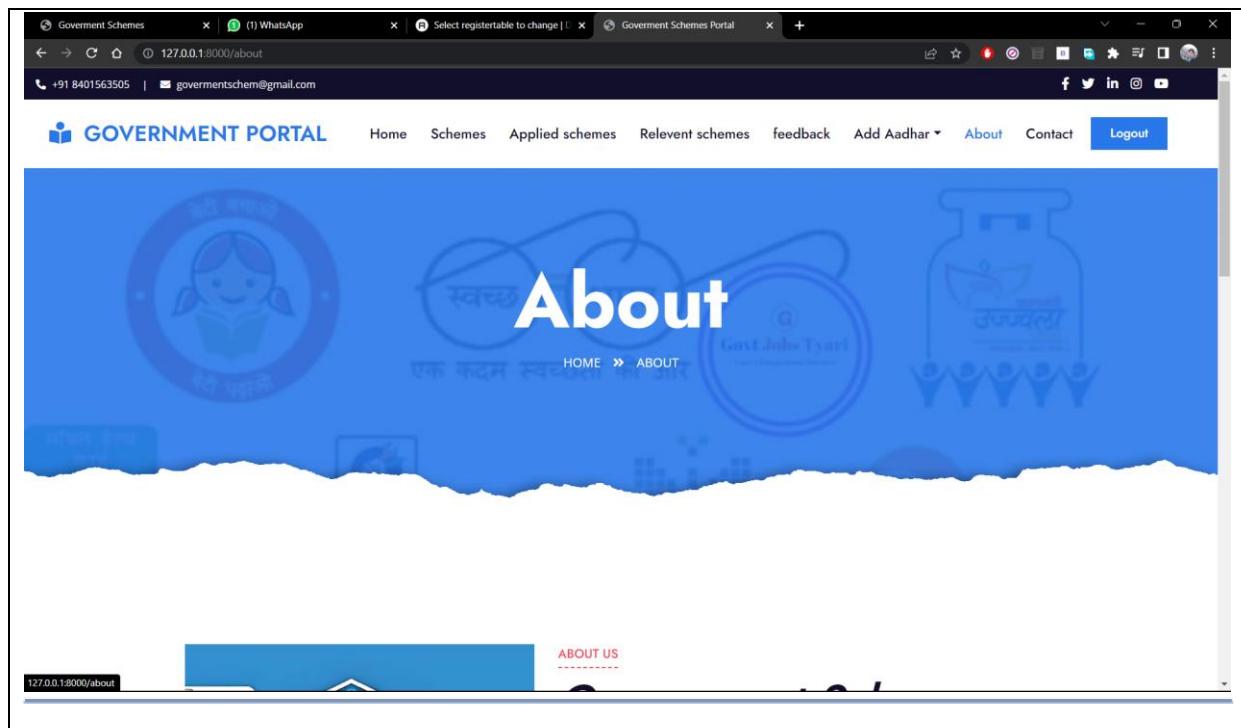


ADD YOUR DETAIL

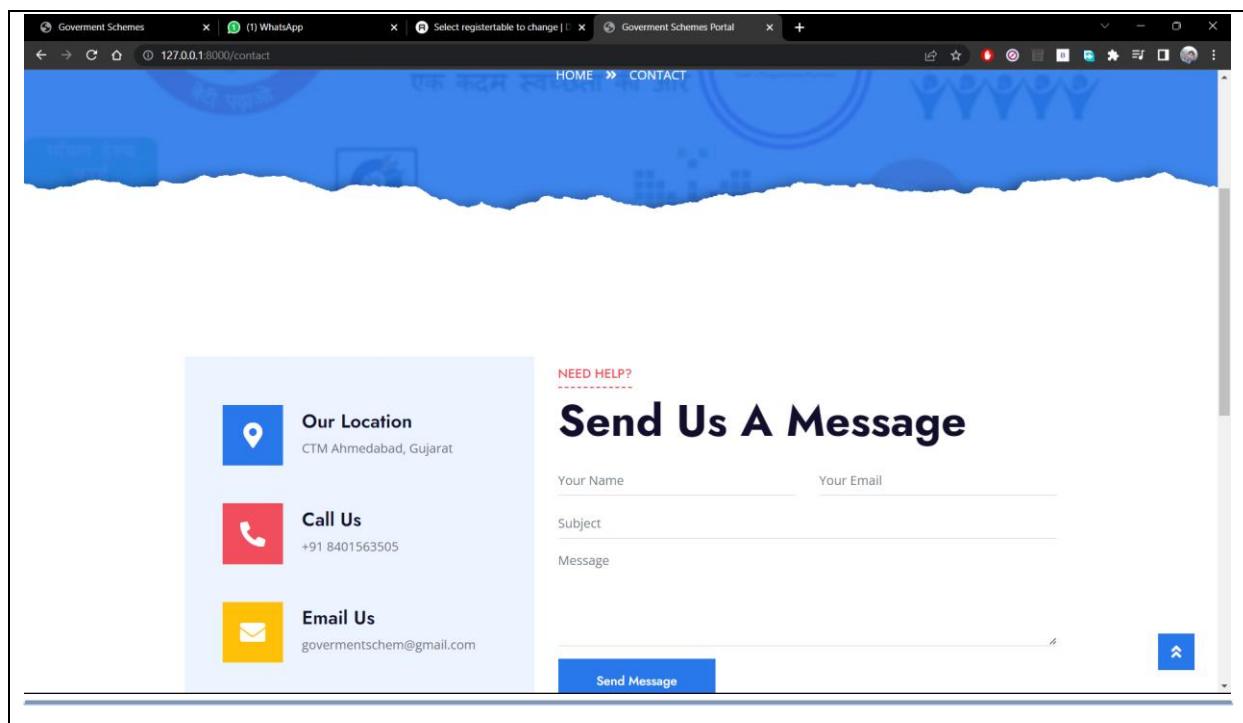
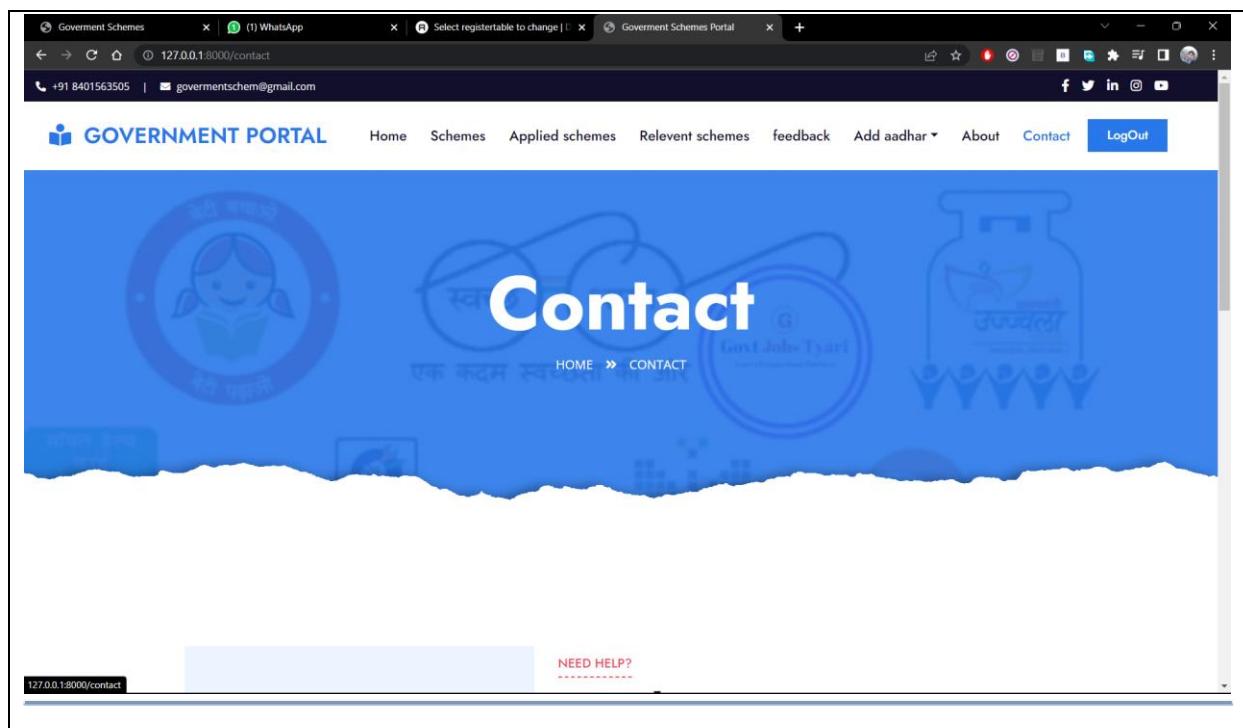


The screenshot shows the same web browser window with the URL 127.0.0.1:8000/addDetail. The main content area now displays a form titled "ADD YOUR DETAIL BELOW". The form includes input fields for "Your Aadhar Number", "Your First Name", "Your Middle Name", "Your Last Name", "Your Phone Number", and a date input field labeled "dd-mm-yyyy". There are also dropdown menus for "Gender" (Male, Female, Other) and "Cast" (All Cast). A small "X" icon is visible in the bottom right corner of the form area.

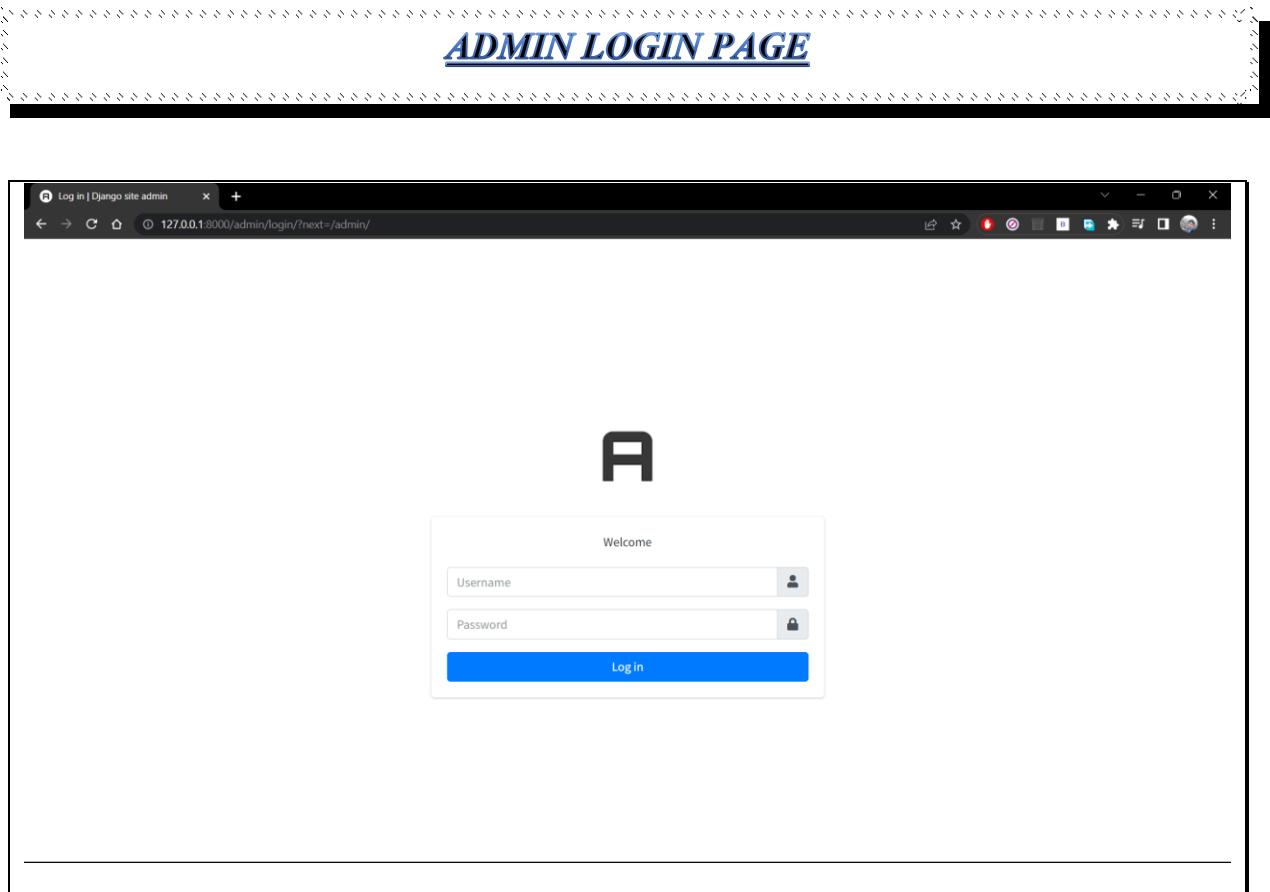
ABOUT PAGE



CONTACT PAGE



3.3 ADMIN SIDE SCREEN LAYOUT



DASHBOARD

The screenshot shows the Django admin interface at the URL `127.0.0.1:8000/admin/`. The left sidebar is dark-themed and lists the following sections:

- Authentication and Authorization**
 - Groups
 - Users
- Goverment_App**
 - Aadharcards
 - Applicationtables
 - Contactables
 - Feedbacks
 - Policytables
 - Registertables

The main dashboard area has three columns:

- Authentication and Authorization**: Contains links for Groups and Users, each with "Add" and "Change" buttons.
- Goverment_App**: Contains links for Aadharcards, Applicationtables, Contactables, Feedbacks, Policytables, and Registertables, each with "Add" and "Change" buttons.
- Recent actions**: A list of six recent operations, each with a red circular icon and a timestamp: "registerd 3 weeks, 3 days ago able object (1)" through "(6)".

AADHAR-CARD DATA (filter)

The screenshot shows the Django admin interface for a 'Goverment_App' application. The left sidebar has a dark theme with a navigation menu. The 'Aadharcardtables' item is selected, highlighted in blue. The main content area displays a table of data with one row. The table columns are: AadharNumber, Firstname, MiddleName, Lastname, Address, PhoneNumber, Dob, Cast, Gender, Document, ResidenceArea, and Disability. The single row of data is: 111122220000, New, user, testing, Ahmedabad, 8885552222, Jan. 31, 1996, General, female, img.jpg, Urban, NO.

AadharNumber	Firstname	MiddleName	Lastname	Address	PhoneNumber	Dob	Cast	Gender	Document	ResidenceArea	Disability
111122220000	New	user	testing	Ahmedabad	8885552222	Jan. 31, 1996	General	female	img.jpg	Urban	NO

APPLICATION DATA

The screenshot shows the Django admin interface for the 'Applicationtables' model. The URL in the browser is `127.0.0.1:8000/admin/government_app/applicationtable/`. The left sidebar has a dark theme with the 'Applicationtables' item selected under the 'Goverment_App' app. The main content area displays a table with two rows:

<input type="checkbox"/>	Regid	Name	Policyid	Status
<input type="checkbox"/>	Newuser	Newuser	Beti Bachao Beti Padhao	1
<input type="checkbox"/>	User	User	Pradhan Mantri Garib Kalyan Yojana	1

Below the table, it says '2 applicationtables'. At the bottom of the page, there is a copyright notice 'Copyright © 2023 . All rights reserved.' and a footer note 'Jazzmin version 2.6.0'.

CONTACT DATA

The screenshot shows the Django admin interface for the 'Contacttables' model. The left sidebar has a dark theme with a navigation menu. The 'Contacttables' item is highlighted with a blue background. The main content area displays a table with three rows of data:

<input type="checkbox"/>	Name	Email	Timestamp	Subject	Feedbackdesk
<input type="checkbox"/>	testing	test@gmail.com	March 22, 2023, 11:09 a.m.	testin	Testing purpose!!!
<input type="checkbox"/>	testing	newuser@gmail.com	March 22, 2023, 11:06 a.m.	testin	testing purpose
<input type="checkbox"/>	twinkle	twinkle@gmail.com	March 17, 2023, 12:17 p.m.	kjd	kjklsdf

Below the table, it says "3 contacttables". At the bottom of the page, there is copyright information and a note about the Jazzmin version.

Copyright © 2023 . All rights reserved.
Jazzmin version 2.6.0

FEEDBACKS DATA

The screenshot shows the Django admin interface for a 'Feedbacks' model. The left sidebar has a dark theme with a 'Feedbacks' link highlighted in blue. The main content area displays a table with 6 rows of feedback data. Each row includes a checkbox, a user's name ('Regid'), their rating ('Rating'), and their comment ('Comments').

	Regid	Rating	Comments
<input type="checkbox"/>	suraj	3	Test
<input type="checkbox"/>	suraj	4	Awsome site
<input type="checkbox"/>	suraj	2	GREAT JOB
<input type="checkbox"/>	Newuser	2	Great experience
<input type="checkbox"/>	Newuser	2	Great experience
<input type="checkbox"/>	Newuser	4	Awsome Site

Below the table, it says '6 feedbacks'. At the bottom of the page, there is copyright information and a note about the Jazzmin version.

Copyright © 2023 . All rights reserved.
Jazzmin version 2.6.0

POLICY DATA

Screenshot of the Django admin interface showing the 'Policytables' list view.

The URL is 127.0.0.1:8000/admin/government_app/policytable/

The page title is 'Policytables'.

There is a green button labeled '+ Add policytable'.

The table has the following columns:

	Policyname	Policydetails	Policytype	Admin photo	PolicyAgency	PolicyTargetAudience	PolicyEligibleCastes	PolicyApplicableState
<input type="checkbox"/>	Swachh Bharat Abhiyan	This scheme aims to clean up and sanitize public places.	Sanitation		Ministry of Housing and Urban Affairs	All individuals	All Castes	All States
<input type="checkbox"/>	Sukanya Samridhi Yojana	This scheme aims to promote the welfare of the girl child by providing a long-term investment option.	Finance		Ministry of Finance	Girl children	Girl children	All Castes
<input type="checkbox"/>	Pradhan Mantri Ujjwala Yojana	This scheme aims to provide LPG connections	Energy		Ministry of Petroleum and Natural Gas	Women from Below Poverty Line households	All Castes	All States

USER SIGN-UP DATA

The screenshot shows the Django admin interface for a 'Registertables' model. The left sidebar has a dark theme with a navigation bar for 'Django administration' and 'admin'. Under 'Goverment_App', 'Registertables' is selected, highlighted with a blue background. The main content area displays a table with 5 registered users. The table columns are: Full Name, Email, Password, Mobile no, Occupation, Status, Role, and Aadharid. The data is as follows:

	Full Name	Email	Password	Mobile no	Occupation	Status	Role	Aadharid
<input type="checkbox"/>	udit	udit@gmail.com	123456	4561237415	Study	-	0	-
<input type="checkbox"/>	Newuser	newuser@gmail.com	newuser	8885552222	Agriculture	-	0	-
<input type="checkbox"/>	User	user@gmail.com	user	8888888888	student	-	-	-
<input type="checkbox"/>	suraj	sun@gmail.com	1234567	1234567	2345678	-	-	-
<input type="checkbox"/>	23456789	wertyui@gmail.com	123456789	5678994535678	qwertyui	-	-	-

5 registertables

Copyright © 2023 . All rights reserved. Jazzmin version 2.6.0

ALL ADMIN DATA

The screenshot shows the Django admin interface at the URL `127.0.0.1:8000/admin/auth/user/`. The left sidebar has a dark theme with the following navigation items:

- Dashboard
- Authentication and Authorization
- Groups
- Users** (selected)
- Goverment_App
 - Aadharcardtables
 - Applicationtables
 - Contactables
 - Feedbacks
 - Policytables
 - Registertables

The main content area is titled "Users" and shows a list of users. There are three users listed:

<input type="checkbox"/>	Username	Email address	First name	Last name	Staff status
<input type="checkbox"/>	Admin				✓
<input type="checkbox"/>	admin	admin123@gmail.com			✓
<input type="checkbox"/>	admin1	admin@gmail.com			✓

Below the table, it says "3 users". At the bottom of the page, there is a footer with the text "Copyright © 2023 . All rights reserved." and "Jazzmin version 2.6.0".

PASSWORD CHANGE PAGE

The screenshot shows the Django admin interface with a password change form. The URL in the browser is `127.0.0.1:8000/admin/password_change/`. The left sidebar lists various models: Dashboard, Groups, Users, Aadharcards, Applicationtables, Contactables, Feedbacks, Policytables, and Registratables. The main content area is titled "Password change" and contains instructions: "Please enter your old password, for security's sake, and then enter your new password twice so we can verify you typed it in correctly." It features three input fields: "Old password" (marked with a red asterisk), "New password" (marked with a red asterisk), and "New password confirmation" (marked with a red asterisk). Below the "New password" field is a list of validation rules:

- Your password can't be too similar to your other personal information.
- Your password must contain at least 8 characters.
- Your password can't be a commonly used password.
- Your password can't be entirely numeric.

A blue "Change password" button is located below the validation rules. At the bottom of the page, there is a copyright notice: "Copyright © 2023 . All rights reserved." and "Jazzmin version 2.6.0".

CHAPTER 4 :- SAMPLE CODING

BACKEND - CODEING

The screenshot shows a code editor with the file `models.py` open. The code defines a model named `aadharcardtable` with various fields:

```
from django.utils.safestring import mark_safe
from django.db import models

CHOICE1 = [
    {'0', 'inactive'},
    {'1', 'active'}
]

CHOICE2 = [
    {'0', 'male'},
    {'1', 'female'}
]

class aadharcardtable(models.Model):
    aadharnumber = models.BigIntegerField(null=True)
    firstname = models.CharField(max_length=30)
    middlename = models.CharField(max_length=30, null=True)
    lastname = models.CharField(max_length=30)
    address = models.TextField(null=True)
    phononenumber = models.BigIntegerField()
    dob = models.DateField()
    cast = models.CharField(max_length=30)
    gender = models.CharField(null=True, max_length=10)
    document = models.FileField(null=True)
    ResidenceArea = models.CharField(max_length=30, null=True)
    DisabilityStatus = models.CharField(max_length=30, null=True)
    MinorityStatus = models.CharField(max_length=30, null=True)
    BPLStatus = models.CharField(max_length=30, null=True)
```

The code editor interface includes a project tree on the left, a toolbar at the top, and various status indicators at the bottom.

The screenshot shows a code editor with the file `models.py` open. It contains two models: `registrable` and `policyable`.

```
return self.firstname

class registrable(models.Model):
    Full_Name = models.CharField(max_length=30)
    Email = models.EmailField()
    Password = models.CharField(max_length=32)
    Mobile_no = models.BigIntegerField()
    Occupation = models.CharField(max_length=40)
    status = models.CharField(null=True, choices=CHOICE1, max_length=30)
    role = models.IntegerField(null=True)
    aadharid = models.ForeignKey(aadharcardtable, on_delete=models.CASCADE, null=True)

    def __str__(self):
        return self.Full_Name

class policyable(models.Model):
    policyname = models.CharField(max_length=120, null=True)
    policydetails = models.TextField(null=True)
    policytype = models.CharField(max_length=100, null=True)
    policyphoto = models.ImageField(upload_to='photos', null=True)
    policyAgency = models.CharField(max_length=100, null=True)
    policyTargetAudience = models.CharField(max_length=100, null=True)
    policyEligibleCastes = models.CharField(max_length=30, null=True)
    policyApplicableState = models.CharField(max_length=30, null=True)
    policyResidenceArea = models.CharField(max_length=30, null=True)
    policyDisabilityStatus = models.CharField(max_length=30, null=True)
    policyMinorityStatus = models.CharField(max_length=30, null=True)
    policyBPLStatus = models.CharField(max_length=30, null=True)
    policyurl = models.CharField(max_length=100, null=True)

    def admin_photo(self):
```

The code editor interface includes a project tree on the left, a toolbar at the top, and various status indicators at the bottom.

```
File Edit View Navigate Code Refactor Run Tools Git Window Help goverment [C:\Django-Project\goverment scheme\goverment] - models.py
goverment > goverment_app > models.py
Project goverment C:\Django-Project\goverment scheme\goverment
  > migrations
  > static
  > templates
    __init__.py
    admin.py
    apps.py
    models.py
    tests.py
    urls.py
    views.py
  > goverment_scheme
  > Lib
  > media
  > Scripts
  > ven
    .gitignore
    activate
    db.sqlite3
    main.py
    manage.py
    pyenv.cfg
    requirements.txt
External Libraries
Scratches and Consoles

45     aadharid = models.ForeignKey(aadharcardtable, on_delete=models.CASCADE, null=True)
46
47     def __str__(self):
48         return self.Full_Name
49
50     class policytable(models.Model):
51         policymname = models.CharField(max_length=120, null=True)
52         policydetails = models.TextField(null=True)
53         policytype = models.CharField(max_length=100, null=True)
54         policyphoto = models.ImageField(upload_to='photos', null=True)
55         policyAgency = models.CharField(max_length=100, null=True)
56         policyTargetAudience = models.CharField(max_length=100, null=True)
57         policyEligibleCastes = models.CharField(max_length=30, null=True)
58         policyApplicableState = models.CharField(max_length=30, null=True)
59         policyResidenceArea = models.CharField(max_length=30, null=True)
60         policyDisabilityStatus = models.CharField(max_length=30, null=True)
61         policyMinorityStatus = models.CharField(max_length=30, null=True)
62         policyBPLStatus = models.CharField(max_length=30, null=True)
63         policyurl = models.CharField(max_length=100, null=True)
64
65     def admin_photo(self):
66         return mark_safe(''.format(self.policyphoto.url))
67
68     policyphoto.allow_tags = True
69
70     def __str__(self):
71         return self.policynname
72
73
74     class applicationtable(models.Model):
75         regid = models.ForeignKey(registertable, on_delete=models.CASCADE)
76         name = models.CharField(max_length=30)
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95

Git Python Packages TODO Python Console Problems Terminal Services
Invalid VCS root mapping: The directory C:\Django-Project\goverment scheme is registered as a Git root, but no Git repositories were found there. // Configure... (today 20:09) 13:2 CRLF UTF-8 4 spaces Python 3.11 main
```

```
File Edit View Navigate Code Refactor Run Tools Git Window Help goverment [C:\Django-Project\goverment scheme\goverment] - models.py
goverment > goverment_app > models.py
Project goverment C:\Django-Project\goverment scheme\goverment
  > migrations
  > static
  > templates
    __init__.py
    admin.py
    apps.py
    models.py
    tests.py
    urls.py
    views.py
  > goverment_scheme
  > Lib
  > media
  > Scripts
  > ven
    .gitignore
    activate
    db.sqlite3
    main.py
    manage.py
    pyenv.cfg
    requirements.txt
External Libraries
Scratches and Consoles

69
70     def __str__(self):
71         return self.policynname
72
73
74     class applicationtable(models.Model):
75         regid = models.ForeignKey(registertable, on_delete=models.CASCADE)
76         name = models.CharField(max_length=30)
77         # dob = models.DateField(null=True)
78         policyid = models.ForeignKey(policytable, on_delete=models.CASCADE, null=True)
79         status = models.IntegerField()
80
81
82     class contacttable(models.Model):
83         email = models.EmailField(null=True)
84         subject = models.CharField(max_length=50, null=True)
85         name = models.CharField(max_length=50, null=True)
86         feedbackdesk = models.TextField(null=True)
87         timestamp = models.DateTimeField(auto_now_add=True, null=True)
88
89
90     class feedback(models.Model):
91         regid = models.ForeignKey(registertable, on_delete=models.CASCADE)
92         rating = models.IntegerField(default=0)
93         comments = models.CharField(max_length=300)
94
95

Git Python Packages TODO Python Console Problems Terminal Services
Invalid VCS root mapping: The directory C:\Django-Project\goverment scheme is registered as a Git root, but no Git repositories were found there. // Configure... (today 20:09) 13:2 CRLF UTF-8 4 spaces Python 3.11 main
```

A screenshot of a code editor showing the `urls.py` file from a Django project named "goverment". The file defines URL patterns for various views. The code includes imports from `django.conf` and `views`, and defines a list of URL patterns with their corresponding view functions and names.

```
from django.conf import settings

urlpatterns = [
    path('', views.indexpage, name="index"),
    path('Logout', views.logout, name="logout"),
    path('contact', views.contactpage, name="contact"),
    path('about', views.aboutpage, name="about"),
    path('schemes', views.schemespge, name="schemes"),
    path('detail', views.detailpage, name="detail"),
    path('details/<int:id>', views.detailpage, name="detail"),
    # path('feature', views.featurepage, name="feature"),
    # path('team', views.teampage, name="team"),
    # path('testimonial', views.testimonialpage, name="testimonial"),
    path('signup', views.signuppage, name="signup"),
    path('fetchdata', views.fetchdata, name='fetchdata'),
    path('checkLogin', views.checklogin, name='checklogin'),
    path('Login', views.loginpage, name='login'),
    path('addDetail', views.addDetailpage, name='addData'),
    path('addData', views.addData, name='addData'),
    path('feedback1', views.feedback1, name='feedback1'),
    path('feedback2', views.feedback2, name='feedback2'),
    path('applypolicy', views.applypolicy, name='applypolicy'),
    path('viewpolicy', views.viewpolicy, name='viewpolicy'),
    path('releventpolicy', views.releventpolicy, name='releventpolicy'),
]
```

A screenshot of a code editor showing the `admin.py` file from the "goverment" Django application. The file contains code to register various models in the Django admin interface, defining custom ModelAdmin classes for each table.

```
from django.contrib import admin
from .models import aadharcardtable, registertable, applicationtable, policytable, contacttable, feedback

# Register your models here.

class showaadharcardtable(admin.ModelAdmin):
    list_display = ['aadharnumber', 'firstname', 'middlename', 'lastname', 'address', 'phonenumber', 'dob', 'cast',
                   'gender', 'document', 'ResidenceArea', 'DisabilityStatus', 'MinorityStatus', 'BPLStatus']

class showregistrationtable(admin.ModelAdmin):
    list_display = ['Full_Name', 'Email', 'Password', 'Mobile_no', 'Occupation', 'status', 'role', 'aadharid']

class showapplicationtable(admin.ModelAdmin):
    list_display = ['regid', 'name', 'policyid', 'status']

class showpolicytable(admin.ModelAdmin):
    list_display = ['policyname', 'policydetails', 'policytype', 'admin_photo', 'policyAgency',
                   'policyTargetAudience', 'policyEligibleCastes', 'policyApplicableState', 'policyResidenceArea',
                   'policyDisabilityStatus', 'policyMinorityStatus', 'policyBPLStatus']

class showfeedbacktable(admin.ModelAdmin):
    list_display = ['regid', 'rating', 'comments']

class showcontacttable(admin.ModelAdmin):
    list_display = ['name', 'email', 'timestamp', 'subject', 'feedbackdesk']
```

Screenshot of PyCharm IDE showing the `views.py` file for the `goverment` application. The code handles POST requests for adding Aadhar details, saving them to the database, and displaying success messages. It also includes functions for handling feedback and fetching data.

```
phonenumber=phone_no,
dob=dob, gender=gender, cast=cast, DisabilityStatus=disabilityStatus,
MinorityStatus=minorityStatus, BPLStatus=BPLStatus, ResidenceArea=ResidenceArea)

query.save()
messages.success(request, 'AADHAR DETAILS ADDED SUCCESSFULLY!!')

else:
    pass

return render(request, 'addDetail.html')

def feedback1(request):
    if request.method == 'POST':
        name = request.POST.get('name')
        email = request.POST.get('email')
        subject = request.POST.get('subject')
        message = request.POST.get('message')
        queue = contacttable(name=name, email=email, subject=subject, feedbackdesk=message)
        queue.save()
        messages.success(request, 'ADDED SUCCESSFULLY!!')

    else:
        pass
    return render(request, 'contact.html')

def fetchdata(request):
    if request.method == 'POST':
        name = request.POST.get("name")
        email = request.POST.get("email")
        occup = request.POST.get("occupation")
        password = request.POST.get("con_password")
```

Screenshot of PyCharm IDE showing the `views.py` file for the `goverment` application. The code defines functions for the login page, adding details, and checking login credentials. It uses POST requests to handle user input and session management for logging in.

```
def loginpage(request):
    return render(request, 'login.html')

def addDetailpage(request):
    return render(request, 'addDetail.html')

def checkLogin(request):
    if request.method == 'POST':
        email = request.POST.get('email')
        password = request.POST.get('password')
        print(email)
        try:
            check = registertable.objects.get(Email=email, Password=password)
            request.session['loginid'] = check.id
            request.session['loginname'] = check.Full_Name

        except:
            check = None

        if check and check.id is not None:
            return render(request, "index.html")
        else:
            return render(request, "login.html")

    else:
        pass

    return render(request, 'login.html')
```

FRONTEND - CODING

The screenshot shows the PyCharm IDE interface with the file `index.html` open. The code is an HTML template for a government scheme website. It includes meta tags for charset, title, viewport, and keywords, as well as links to static files like favicon.ico and Google Web Fonts. It also includes CSS imports for Bootstrap and OwlCarousel, and a customized Bootstrap stylesheet. The body section starts with a Topbar Start div containing a row with a col-lg-6 text-center column. This column contains two small text blocks with phone and email icons.

```
<!DOCTYPE html>
<html lang="en">
    {% load static %}
    <head>
        <meta charset="utf-8">
        <title>Government Schemes</title>
        <meta content="width=device-width, initial-scale=1.0" name="viewport">
        <meta content="Free HTML Templates" name="keywords">
        <meta content="Free HTML Templates" name="description">
        <!-- Favicon -->
        <link href="{% static 'img/favicon.ico' %}" rel="icon">
        <!-- Google Web Fonts -->
        <link rel="preconnect" href="https://fonts.gstatic.com">
        <link href="https://fonts.googleapis.com/css2?family=Jost:wght@500;600;700&family=Open+Sans:wght@400;600&display=swap" rel="stylesheet">
        <!-- Font Awesome -->
        <link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.10.0/css/all.min.css" rel="stylesheet">
        <!-- Libraries Stylesheet -->
        <link href="{% static 'lib/owlcarousel/assets/owl.carousel.min.css' %}" rel="stylesheet">
        <!-- Customized Bootstrap Stylesheet -->
        <link href="{% static 'css/style.css' %}" rel="stylesheet">
    </head>
    <body>
        <!-- Topbar Start -->
        <div class="container-fluid bg-dark">
            <div class="row py-2 px-lg-5">
                <div class="col-lg-6 text-center text-lg-left mb-2 mb-lg-0">
                    <small><i class="fa fa-phone-alt mr-2"></i>+91 8401563505 </small>
                    <small><i class="fa fa-envelope mr-2"></i>governmentschem@gmail.com</small>
                </div>
            </div>
            <div class="col-lg-6 text-center text-lg-right">
                <div class="d-inline-flex align-items-center text-white">
                    <a class="text-white px-2" href="">
                        <i class="fab fa-facebook-f"></i>
                    </a>
                    <a class="text-white px-2" href="">
                        <i class="fab fa-twitter"></i>
                    </a>
                    <a class="text-white px-2" href="">
                        <i class="fab fa-linkedin-in"></i>
                    </a>
                    <a class="text-white px-2" href="">
                        <i class="fab fa-instagram"></i>
                    </a>
                    <a class="text-white pl-2" href="">
                        <i class="fab fa-youtube"></i>
                    </a>
                </div>
            </div>
        </div>
        <!-- Topbar End -->
```

This screenshot shows the same PyCharm IDE interface with the `index.html` file open, but with code folding expanded. The expanded code reveals the full structure of the Topbar Start and Topbar End sections, showing the detailed HTML and CSS for each element, such as the font icons and their corresponding classes.

```
<!-- Topbar Start -->
<div class="container-fluid bg-dark">
    <div class="row py-2 px-lg-5">
        <div class="col-lg-6 text-center text-lg-left mb-2 mb-lg-0">
            <small><i class="fa fa-phone-alt mr-2"></i>+91 8401563505 </small>
            <small><i class="fa fa-envelope mr-2"></i>governmentschem@gmail.com</small>
        </div>
    </div>
    <div class="col-lg-6 text-center text-lg-right">
        <div class="d-inline-flex align-items-center text-white">
            <a class="text-white px-2" href="">
                <i class="fab fa-facebook-f"></i>
            </a>
            <a class="text-white px-2" href="">
                <i class="fab fa-twitter"></i>
            </a>
            <a class="text-white px-2" href="">
                <i class="fab fa-linkedin-in"></i>
            </a>
            <a class="text-white px-2" href="">
                <i class="fab fa-instagram"></i>
            </a>
            <a class="text-white pl-2" href="">
                <i class="fab fa-youtube"></i>
            </a>
        </div>
    </div>
</div>
<!-- Topbar End -->
```

The screenshot shows the PyCharm IDE interface with the file `index.html` open. The code displays a navigation bar template for a government portal. It includes links for Home, Schemes, Applied schemes, Relevant schemes, Feedback, Add Aadhar, Add Detail, About, Contact, and Logout. The code uses Bootstrap classes and Django URL placeholders.

```
<!-- Navbar Start -->
<div class="container-fluid p-0">
    <nav class="navbar navbar-expand-lg bg-white navbar-light py-3 py-lg-0 px-lg-5">
        <a href="{% url 'index' %}" class="navbar-brand ml-lg-3">
            <h3 class="m-0 text-uppercase text-primary"><i class="fa fa-book-reader mr-3"></i>Government Portal</h3>
        </a>
        <button type="button" class="navbar-toggler" data-toggle="collapse" data-target="#navbarCollapse">
            <span class="navbar-toggler-icon"></span>
        </button>
        <div class="collapse navbar-collapse justify-content-between px-lg-3" id="navbarCollapse">
            <div class="navbar-nav mx-auto py-0">
                <a href="{% url 'index' %}" class="nav-item nav-link active">Home</a>
                <a href="{% url 'schemes' %}" class="nav-item nav-link">Schemes</a>
                {% if request.session.LoginId %}
                    <a href="{% url 'viewpolicy' %}" class="nav-item nav-link">Applied schemes</a>
                    <a href="{% url 'releventpolicy' %}" class="nav-item nav-link">Relevant schemes</a>
                    <a href="{% url 'feedback' %}" class="nav-item nav-link">Feedback</a>
                    <div class="nav-item dropdown">
                        <a href="#" class="nav-link dropdown-toggle" data-toggle="dropdown">Add Aadhar</a>
                        <div class="dropdown-menu m-0">
                            <a href="{% url 'addDetail' %}" class="dropdown-item">Add Detail</a>
                        </div>
                    </div>
                {% endif %}
                <a href="{% url 'about' %}" class="nav-item nav-link">About</a>
                <a href="{% url 'contact' %}" class="nav-item nav-link">Contact</a>
                {% if request.session.LoginId %}
                    <a href="{% url 'logout' %}" class="btn btn-primary py-2 px-4 d-none d-lg-block">Logout</a>
                {% else %}

```

The screenshot shows the PyCharm IDE interface with the file `index.html` open. The code has been extended to include a "Get In Touch" section with contact information for CTM Ahmedabad, Gujarat, and a phone number. Below this, there are sections for social media links (Facebook, Twitter, LinkedIn, Instagram) and quick links for Web Design, Apps Design, Marketing, Research, and SEO. The code uses Bootstrap and Font Awesome icons.

```
</div>
<div class="row">
    <div class="col-md-4 mb-5">
        <h3 class="text-white mb-4">Get In Touch</h3>
        <p><i class="fa fa-map-marker-alt mr-2"></i>CTM Ahmedabad, Gujarat</p>
        <p><i class="fa fa-phone-alt mr-2"></i>+ 91 8401563505</p>

        <div class="d-flex justify-content-start mt-4">
            <a class="text-white mr-4" href="#">
```

CHAPTER 5:- FUTURE ENHANCEMENT

Points

- Add Map to View User Nearest 10 KM all Government Offices.
- Add notification to send add new Relevant schemes to user.
- Change fetching Schemes database to API (JSON).

CHAPTER 6:- CONCLUSION

Points

- Our website is provide information of newly release schemes.
- Our website is an essential tool for promoting and managing any type of newly released government schemes
- It is use for check a relevant schemes and apply for New schemes. our website is user-friendly.
- In existing website user add personal data every time but our website add your detail only one time next time only you have to login our website.
- Our website send notifications to user relevant schemes new added.

CHAPTER 7 :- BIBLIOGRAPHY

▪ Books:

- CC-110 Database Management System-I
(ComputerWorld Publication)

- CC302 Python Programming
(ComputerWorld Publication)

▪ Websites:

- <https://github.com>
- <https://www.w3schools.com>
- <https://pypi.org>
- <https://www.geeksforgeeks.org>