

ADF Practical – 5 Steps

- Submitted by: - Sanket Chheladiya(18bce042)

-Submitted To: - Prof. Dvijesh Bhatt

Q1:-

1. Create a signup and login page using Django Admin Auth package. Ask user to include the the following information during the registration:

a. User ID, Email Address, Password and Confirm Password {Django Form Fields}

base.html

- Base.html provide view page according to situations like if we are not authenticated into site the page will be shown having signin and signup button.
- elseif user authenticated then it will show us home page having all the functionality of site.

```
<!doctype html>
<html lang="en">
  <head>
    <!-- Required meta tags -->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">

    <!-- Bootstrap CSS -->
    <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/css/bootstrap.min.c
ss" rel="stylesheet" integrity="sha384-
F3w7mX95PdgyTmZZMECAngseQB83DfGTowi0iMjiWaeVhAn4FJkqJByhZMI3AhiU"
crossorigin="anonymous">

    <title>{% block title %}{% endblock title %}</title>
  </head>
  <body>
    <nav class="navbar navbar-expand-lg navbar-dark bg-dark">
      <div class="container-fluid">
        <a class="navbar-brand" href="/">Social Media Site</a>
        <button class="navbar-toggler" type="button" data-bs-
toggle="collapse" data-bs-target="#navbarSupportedContent" aria-
```

```

controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle
navigation">
    <span class="navbar-toggler-icon"></span>
</button>
<div class="collapse navbar-collapse"
id="navbarSupportedContent">
    <ul class="navbar-nav me-auto mb-2 mb-lg-0">
</ul>

    {% if user.is_authenticated %}
    <a class="btn btn-primary mx-2" href="{% url 'user_add_post'
%}">
        Add Post
    </a>
    <a class="btn btn-primary" href="{% url 'user_post_list' %}">
        Post list
    </a>
    <ul class="navbar-nav ml-auto">
    <li class="nav-item dropdown">

        <a class="nav-link dropdown-toggle" href="#"
id="navbarDropdown" role="button" data-bs-toggle="dropdown" aria-
expanded="false">
            Welcome {{ request.user }}
        </a>
        <ul class="dropdown-menu" aria-labelledby="navbarDropdown">
            <li><a class="dropdown-item" href="{% url 'logout'
%}">Logout</a></li>
        </ul>
    </li>
    </ul>
    {% else %}
    <a class="btn btn-primary mx-2" href="{% url 'user_register'
%}">
        Sign up
    </a>
    <a class="btn btn-primary" href="{% url 'login' %}">
        Sign In
    </a>
    {% endif %}
    </div>
</div>
</nav>
<div class="container">
    {% for message in messages %}
    <div class="alert alert-{{ message.tags }} alert-dismissible fade
show" role="alert">
        <strong>Message </strong>{{ message }}
        <button type="button" class="btn-close" data-bs-dismiss="alert"
aria-label="Close"></button>
    </div>
    {% endfor %}
</div>
    {% block body %}{% endblock body %}
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/js/bootstrap.bundle.
min.js" integrity="sha384-
/bQdsTh/da6pkI1MST/rWKFNjaCP5gBSY4sEBT38Q/9RBh9AH40zEOg7Hlq2THRZ"
crossorigin="anonymous"></script>
</body>
</html>

```

Home.html

```
{% extends 'base.html' %}
{% block title %}Home{% endblock title %}
```

urls.py

```
from django.contrib import admin
from django.urls import path,include
from user import views

from django.conf import settings
from django.conf.urls.static import static

urlpatterns = [
    path('admin/', admin.site.urls),
    path("accounts/",include('django.contrib.auth.urls')),
    path('register/',views.registerUser,name="user_register"),
    path('add_post/',views.add_post,name="user_add_post"),
    path('show_post/',views.show_post,name="user_post_list"),
    path('delete_post/<str:postname>',views.delete_post,name="deletepost"),
    path('update_post/<int:id>',views.update_post,name="updatepost"),
    path('registration/emailVerification/<uidb64>/<token>', views.activate,
name='emailActivate'),
    path('user_profile/',views.view_profile,name="user_profile"),
    path('',views.home,name="home"),
]
if settings.DEBUG:
    urlpatterns +=
static(settings.MEDIA_URL,document_root=settings.MEDIA_ROOT)
```

admin.py

```
from django.contrib import admin
from django.contrib.auth.models import User
from .models import Post,UserProfile

@admin.register(Post)
class UserAdmin(admin.ModelAdmin):
    list_display = ('id','postusername' , 'postname', 'description')

@admin.register(UserProfile)
class UserAdmin1(admin.ModelAdmin):
    list_display = ('id','username' , 'email',
'birthdate','mobilen0','gender', 'city', 'pincode','profile','document')
```

views.py

```
from django.shortcuts import render,HttpResponse
from .forms import RegisterForm,ProfileForm
from django.contrib import messages
from django.shortcuts import render,redirect
from .forms import addPost
from django.contrib.auth.models import User
from .models import Post,UserProfile
```

```

from django.conf import settings
from django.core.mail import send_mail
from django.utils.http import urlsafe_base64_encode, urlsafe_base64_decode
from django.contrib.sites.shortcuts import get_current_site
from .token import account_activation_token
from django.core.mail import send_mail
from django.template.loader import render_to_string
from django.utils.encoding import force_bytes, force_text

def registerUser(request):
    if request.method=="POST":
        form=RegisterForm(request.POST)
        if form.is_valid():
            email = request.POST['email']
            user = form.save(commit=False)
            user.is_active = False
            user.save()
            current_site = get_current_site(request)
            Msub = 'Activate your account on Social Media Site'
            msg = render_to_string('registration/acc_active_email.html', {
                'user': user,
                'domain': current_site.domain,
                'uid': urlsafe_base64_encode(force_bytes(user.pk)),
                'token': account_activation_token.make_token(user),
            })
            email_from = settings.EMAIL_HOST_USER
            recipient_list = [email,]
            send_mail(Msub, msg, email_from, recipient_list)
            return HttpResponse("Please confirm your email address to complete the registration!")
        else:
            context={'form':form}
            messages.error(request,'There is Error in your information...kindly refill the form')
            render(request,'registration/signup.html',context)
            form=RegisterForm()
            context={'form':form}
            return render(request,'registration/signup.html',context)

def activate(request, uidb64, token):
    try:
        uid = force_text(urlsafe_base64_decode(uidb64))
        user = User.objects.get(pk=uid)
    except(TypeError, ValueError, OverflowError, User.DoesNotExist):
        user = None
    if user is not None and account_activation_token.check_token(user, token):
        user.is_active = True
        user.save()
        messages.success(request, 'Thank you for your email confirmation. Now you can login your account.')
        messages.success(request, 'Please complete your profile in profile section after login.')
        return redirect('login')
    else:
        return HttpResponse('Activation link is invalid!')

def add_post(request):
    pi = User.objects.get(username=request.user)

```

```

if(request.method == 'POST'):
    fm = addPost(request.POST)
    if(fm.is_valid()):
        pname = fm.cleaned_data['postname']
        desc = fm.cleaned_data['description']
        reg = Post(postusername=pi, postname=pname, description=desc)
        reg.save()
        fm = addPost()
    else:
        fm = addPost()
    return render(request, 'registration/addPost.html', {'form':fm})

def show_post(request):
    stud = Post.objects.filter(postusername=request.user)
    return render(request, 'registration/showPost.html' , {'stu':stud})

def delete_post(request, postname):
    if request.method == 'POST':
        pi = Post.objects.get(postusername=request.user, postname=postname)
        pi.delete()
        stud = Post.objects.filter(postusername=request.user)
        return render(request, 'registration/showPost.html' , {'stu':stud})

def update_post(request, id):
    if request.method == 'POST':
        pi = Post.objects.get(pk=id, postusername=request.user)
        fm = addPost(request.POST, instance=pi)
        if fm.is_valid():
            fm.save()
        else:
            pi = Post.objects.get(pk=id, postusername=request.user)
            fm = addPost(instance=pi)
        return render(request, 'registration/updatePost.html', {'form':fm})

def view_profile(request):
    if not UserProfile.objects.filter(username=request.user).exists():
        if request.method=="POST":
            fm=ProfileForm(request.POST,request.FILES)
            if fm.is_valid():
                uname = request.user
                email = uname.email
                bdate = fm.cleaned_data['birthdate']
                mobno = fm.cleaned_data['mobilenno']
                gender = fm.cleaned_data['gender']
                city = fm.cleaned_data['city']
                pincode = fm.cleaned_data['pincode']
                profile = fm.cleaned_data['profile']
                document = fm.cleaned_data['document']
                reg = UserProfile(username=uname,
email=email,birthdate=bdate,mobileno=mobno,
gender=gender,city=city,pincode=pincode,profile=profile,document=document)
                reg.save()
                fm = ProfileForm()
                return redirect('user_profile')
            else:
                context={'form':fm}
                messages.error(request,'There is Error in your
information...kindly refill the form')
                render(request,'registration/userProfile.html',context)
                fm=ProfileForm()

```

```

        return render(request, 'registration/userProfile.html',
{'form':fm})

    stud = UserProfile.objects.filter(username=request.user)
    return render(request, 'registration/showProfile.html' , {'stu':stud})

def home(request):
    return render(request, 'home.html')

```

forms.py

```

from django.contrib.auth.forms import UserCreationForm
from django.contrib.auth.models import User
from django.forms import widgets
from .models import Post, UserProfile
from django import forms

class RegisterForm(UserCreationForm):
    class Meta:
        model = User
        fields = ["username", "password1", "password2", "email",
"first_name", "last_name"]

class addPost(forms.ModelForm):
    class Meta:
        model = Post
        fields = ["postname", "description"]
        widgets = {
            'postname': forms.TextInput(attrs={'class': 'form-control'}),
            'description': forms.TextInput(attrs={'class': 'form-
control'}),
        }

class ProfileForm(forms.ModelForm):
    class Meta:
        model = UserProfile
        fields = ["birthdate", "mobilen", "gender",
"city", "pincode", "profile", "document"]
        widgets = {
            'birthdate': forms.TextInput(attrs={'class': 'form-control'}),
            'mobilen': forms.TextInput(attrs={'class': 'form-control'}),
            'gender': forms.TextInput(attrs={'class': 'form-control'}),
            'city': forms.TextInput(attrs={'class': 'form-control'}),
            'pincode': forms.TextInput(attrs={'class': 'form-control'}),
            #'profile' : forms.ImageField(),
        }

```

tokens.py

```

from django.contrib.auth.tokens import PasswordResetTokenGenerator
import six
class TokenGenerator(PasswordResetTokenGenerator):
    def _make_hash_value(self, user, timestamp):
        return (
            six.text_type(user.pk) + six.text_type(timestamp) +
            six.text_type(user.is_active)
        )
account_activation_token = TokenGenerator()

```

models.py

```

from django.db import models

class Post(models.Model):
    postusername = models.CharField(max_length=20)
    postname = models.CharField(max_length=200)
    description = models.CharField(max_length=200)

class UserProfile(models.Model):
    username = models.CharField(max_length=20)
    email = models.EmailField(max_length = 254)
    birthdate = models.CharField(max_length=10)
    mobileno = models.CharField(max_length=10)
    gender = models.CharField(max_length=6)
    city = models.CharField(max_length=10)
    pincode = models.CharField(max_length=6)
    profile = models.ImageField(upload_to = "images/")
    document = models.FileField(upload_to= "docs/")

```

signup.html

```

{% extends 'base.html' %}
{% block title %}Singup{% endblock title %}
{% block body %}
    <div class="container my-2">
        <h2>Register Here</h2>
        <form method="post">
            {%csrf_token%}
            {{form.as_p}}
            <button type="submit" class="btn btn-primary">Register</button>
        </form>
    </div>
{% endblock body %}

```

Login.html

```

{% extends 'base.html' %}
{% block title %}Login{% endblock title %}
{% block body %}
    <div class="container">
        <h2>Login Page</h2>
        <form method="post">
            {%csrf_token%}
            {{ form.as_p }}
            <button type="submit">Log In</button>
        </form>
    </div>

```

```
</form>
</div>

{% endblock body %}
```

Screenshots (signin/signup page)

A screenshot of a web browser window showing a registration form titled 'Register Here'. The form includes the following fields and instructions:

- Username:** A text input field with a note: 'Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.'
- Password:** A text input field with a list of requirements:
 - 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.
- Password confirmation:** A text input field with a note: 'Enter the same password as before, for verification.'
- Email address:** A text input field.
- First name:** A text input field.
- Last name:** A text input field.
- Register:** A blue button at the bottom of the form.

The browser's address bar shows the URL '127.0.0.1:8000/register/'. The top navigation bar is dark with 'Social Media Site' and 'Sign up'/'Sign in' buttons.

The screenshot shows a web browser with multiple tabs. The active tab is titled '127.0.0.1:8000/register/'. The browser's address bar shows the URL '127.0.0.1:8000/register/'. The page has a dark header with the text 'Social Media Site' and two buttons: 'Sign up' and 'Sign In'. The main content area is titled 'Register Here'. It contains a registration form with the following fields and labels:

- Username:** Sanket543. Below the field is the text: 'Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.'
- Password:** A field with masked characters (dots).
- Password confirmation:** A field with masked characters (dots). Below it is the text: 'Enter the same password as before, for verification.'
- Email address:** sanketchheladiya1904200
- First name:** Sanket
- Last name:** Patel

Below the form is a blue button labeled 'Register'.

Q2:-

Verify the user mail id by sending them verification mail. User must verify his/her mail id before create/react/view any post on social media platform

The screenshot shows a web browser with multiple tabs. The active tab is titled '127.0.0.1:8000/register/'. The browser's address bar shows the URL '127.0.0.1:8000/register/'. The page has a dark header with the text 'Social Media Site' and two buttons: 'Sign up' and 'Sign In'. The main content area displays the message: 'Please confirm your email address to complete the registration!'.

- acc_active_sent.html

```
{% extends 'base.html' %}

{% block content %}
<p>Please confirm your email address to complete the registration. </p>
{% endblock %}
```

- acc_active_email.html

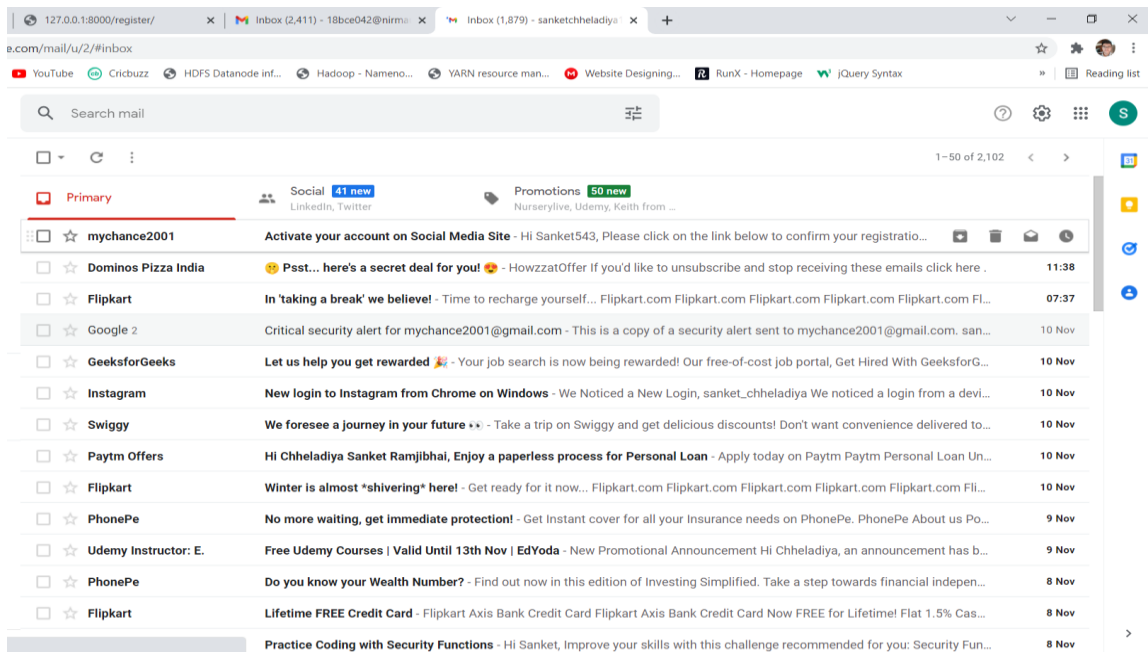
```
{% autoescape off %}
Hi {{ user.username }},

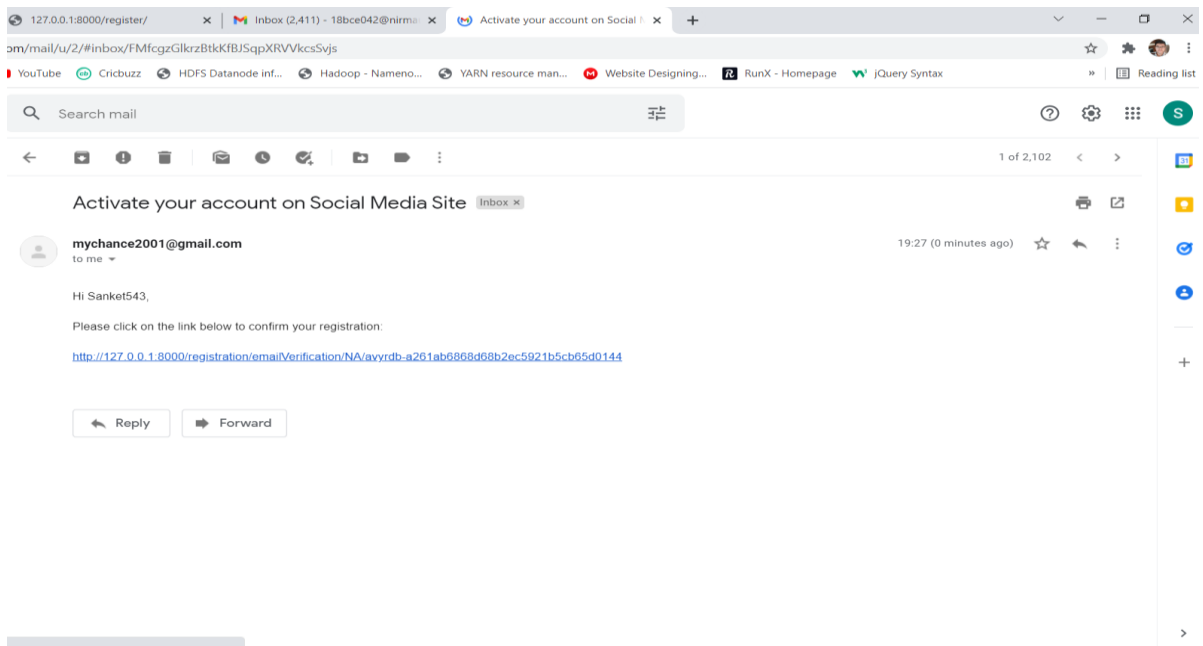
Please click on the link below to confirm your registration:

http://{{ domain }}{% url 'emailActivate' uidb64=uid token=token %}
{% endautoescape %}
```

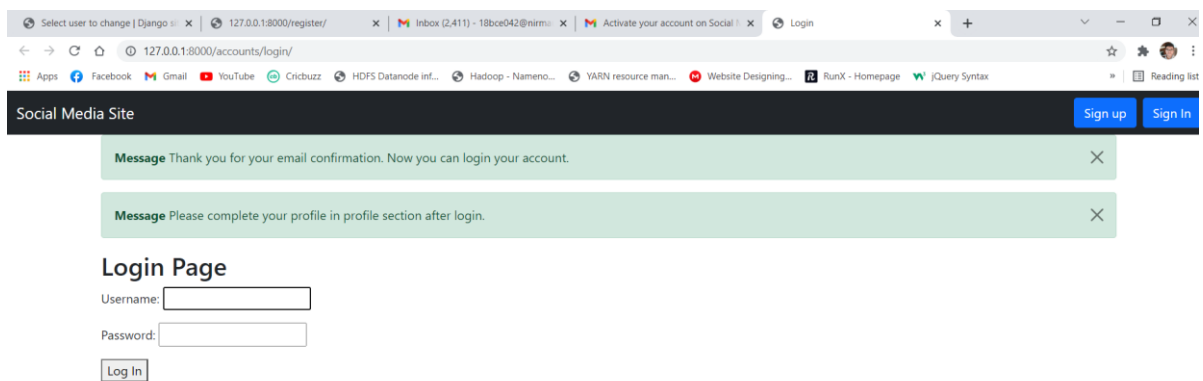
- emailVerification.html

```
Hi ,
Please click on the link to confirm your registration,
http://{{ domain }}/emailVerification/{{ uid }}/{{ token }}
```





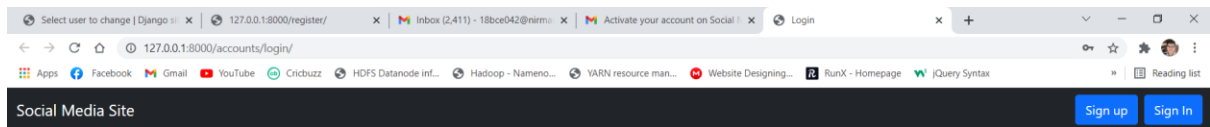
- After Clicking the link redirect to login page



Q3:-

Once the user verifies his/her email id, redirect them to fetch the extra information about users using Django. Forms- Extra information of users that are to be collected are as mentioned below:

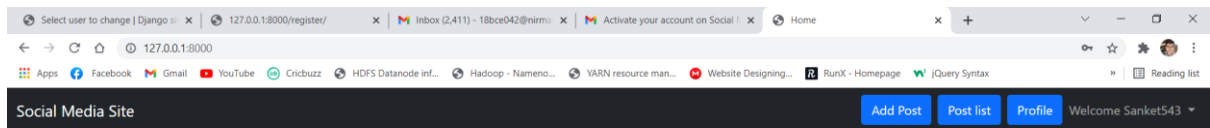
a. Birth Date, Calculate Age based on entered date, mobile no, gender, city, pincode, state {Use Django Fields, Widgets and Attributes}



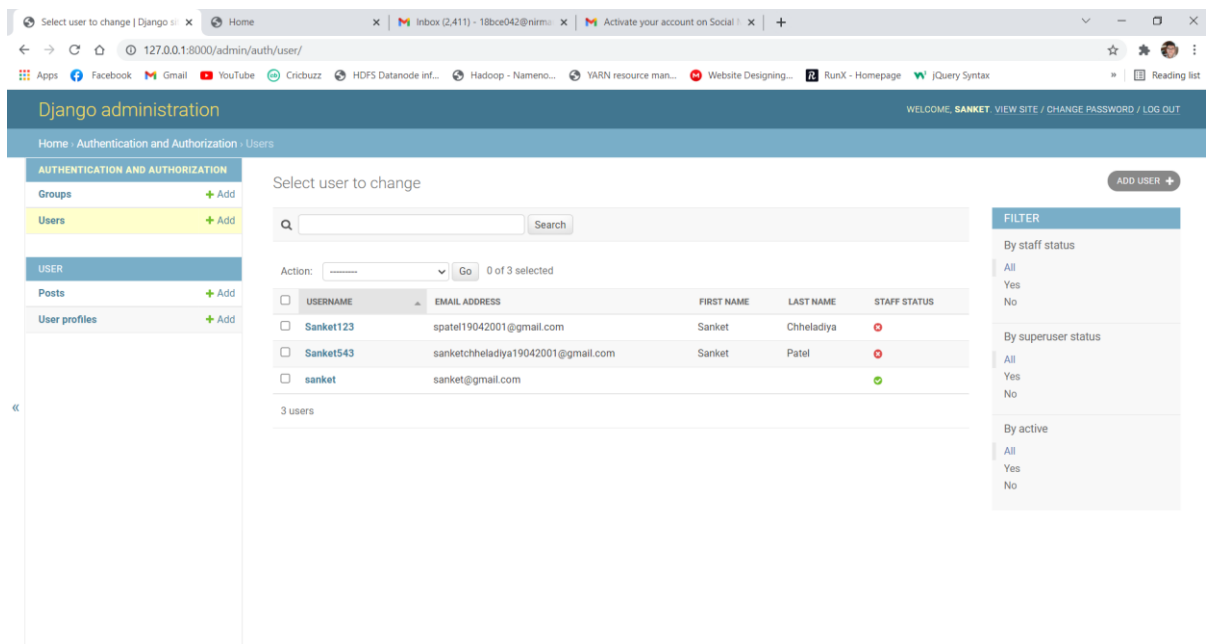
Login Page

Username:

Password:



-Check User will be added to Django admin



- userProfile.html

```
{% extends 'base.html' %}
{% block title %}User Profile{% endblock title %}
{% block body %}
    <div class="container my-2">
        <h2>Complete your Profile</h2>
        <form method="post" enctype="multipart/form-data">
            {% csrf_token %}
            {{ form.as_p }}
            <button type="submit" class="btn btn-primary">Submit</button>
        </form>
    </div>
{% endblock body %}
```

-Add Extra Information about user and make profile

The screenshot shows a web browser window with the address bar displaying '127.0.0.1:8000/user_profile/'. The browser's tab bar shows several open tabs, including 'Apps', 'Facebook', 'Gmail', 'YouTube', 'Cricbuzz', 'HDFS Datanode inf...', 'Hadoop - Namen...', 'YARN resource man...', 'Website Designing...', 'RunX - Homepage', and 'jQuery Syntax'. The browser's address bar also shows a 'Reading list' icon.

The web page has a dark header bar with the text 'Social Media Site' on the left and three blue buttons ('Add Post', 'Post list', 'Profile') on the right. The 'Profile' button is highlighted. To the right of the buttons, the text 'Welcome Sanket543' is displayed with a dropdown arrow.

The main content area is titled 'Complete your Profile'. It contains several form fields with labels and input boxes:

- Birthdate:
- Mobileno:
- Gender:
- City:
- Pincode:
- Profile: No file chosen
- Document: No file chosen

At the bottom of the form is a blue 'Submit' button.

127.0.0.1:8000/user_profile/

Apps Facebook Gmail YouTube Cricbuzz HDFS Datanode inf... Hadoop - Namen... YARN resource man... Website Designing... RunX - Homepage jQuery Syntax Reading list

Social Media Site [Add Post](#) [Post list](#) [Profile](#) Welcome Sanket543

Complete your Profile

Birthdate:

Mobileno:

Gender:

City:

Pincode:

Profile: [Choose File](#) WIN_202001...1_31_Pro.jpg

Document: [Choose File](#) CamScanner... 15.11.14.pdf

[Submit](#)

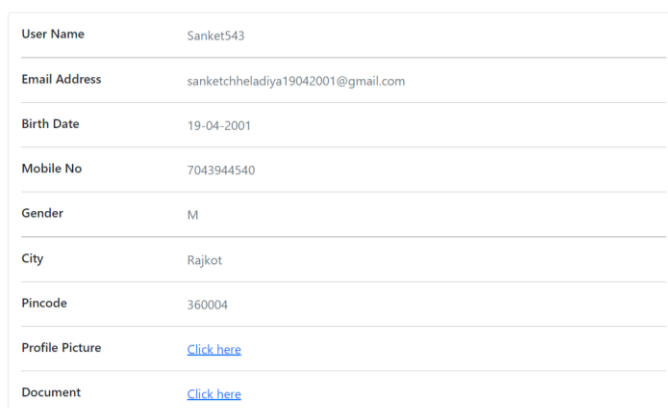
-After submit profile page looks like below we can see uploaded photo and documents about us by clicking [click here link](#).

-showProfile.html

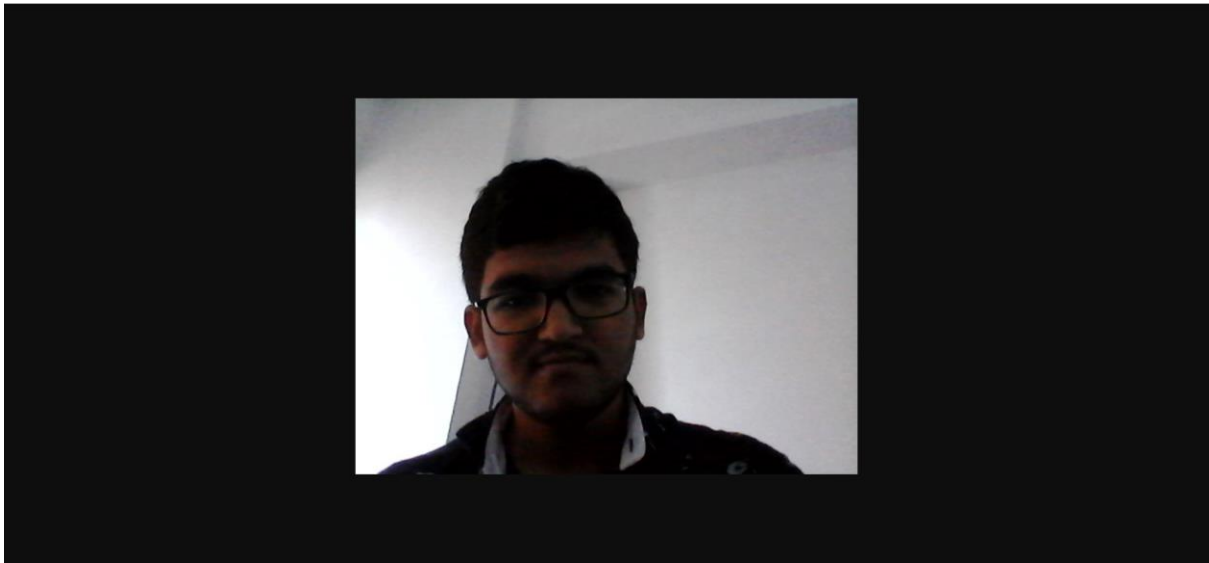
```
{% extends 'base.html' %}
{% block title %}Show Profile{% endblock title %}
{% block body %}
    <div class="container">
        <br><br><br><br>
        {% if stu %}
        <div class="main-body">
            <nav aria-label="breadcrumb" class="main-breadcrumb">
                </nav>
            {% for st in stu %}
                <div class="row gutters-sm">
                    <div class="col-md-4 mb-3">
                        <div class="card">
                            <div class="card-body">
                                {% comment %}
https://www.w3schools.com/howto/img_avatar.png {% endcomment %}
                                <div class="d-flex flex-column align-items-center text-center">
                                    
                                    <div class="mt-3">
                                        <h4>{{st.username}} </h4>
                                    </div>
                                </div>
                            </div>
                        </div>
                    </div>
                </div>
            {% endfor %}
        </div>
    </div>
</div>
```

```
<div class="col-md-8">
  <div class="card mb-3">
    <div class="card-body">
      <div class="row">
        <div class="col-sm-3">
          <h6 class="mb-0">User Name</h6>
        </div>
        <div class="col-sm-9 text-secondary">
          {{st.username}}
        </div>
      </div>
      <hr>
      <div class="row">
        <div class="col-sm-3">
          <h6 class="mb-0">Email Address</h6>
        </div>
        <div class="col-sm-9 text-secondary">
          {{st.email}}
        </div>
      </div>
      <hr>
      <div class="row">
        <div class="col-sm-3">
          <h6 class="mb-0">Birth Date</h6>
        </div>
        <div class="col-sm-9 text-secondary">
          {{st.birthdate}}
        </div>
      </div>
      <hr>
      <div class="row">
        <div class="col-sm-3">
          <h6 class="mb-0">Mobile No</h6>
        </div>
        <div class="col-sm-9 text-secondary">
          {{st.mobilenos}}
        </div>
      </div>
      <hr>
      <div class="row">
        <div class="col-sm-3">
          <h6 class="mb-0">Gender</h6>
        </div>
        <div class="col-sm-9 text-secondary">
          {{st.gender}}
        </div>
      </div>
      <hr>
      <div class="row">
        <div class="col-sm-3">
          <h6 class="mb-0">City</h6>
        </div>
        <div class="col-sm-9 text-secondary">
          {{st.city}}
        </div>
      </div>
      <hr>
      <div class="row">
        <div class="col-sm-3">
          <h6 class="mb-0">Pincode</h6>
        </div>
```

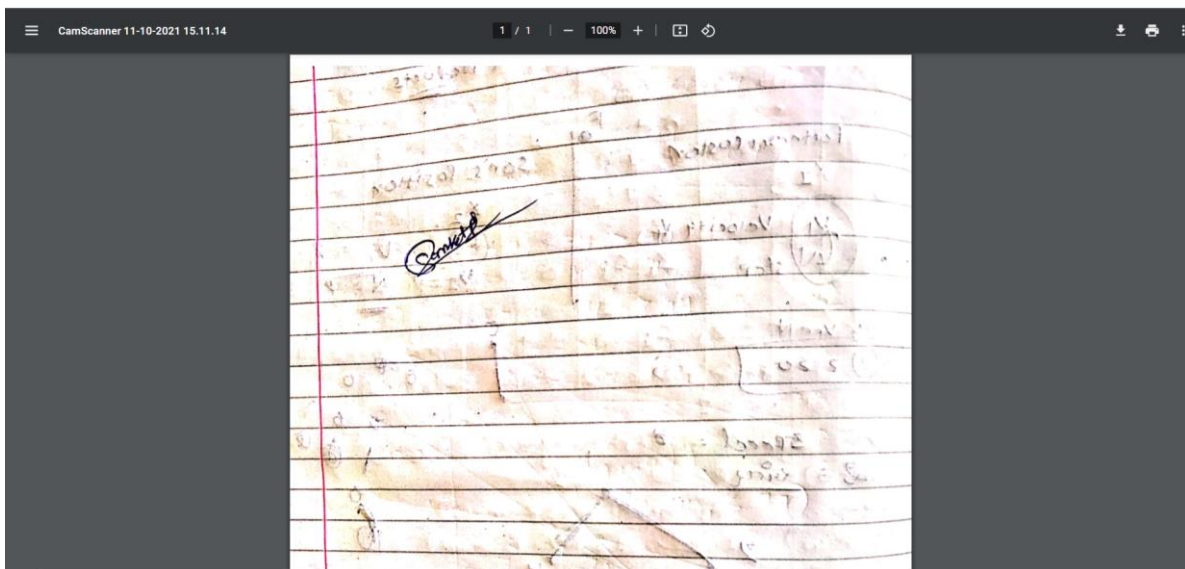
← → ↻ 🏠 127.0.0.1:8000/user_profile/ ☆ 🌙 🔍
📱 Apps 📘 Facebook 📧 Gmail 📺 YouTube 🎮 Cricbuzz 📊 HDFS Datanode info... 🔄 Hadoop - Namenode... 📄 YARN resource man... 📄 Website Designing... 📄 RunX - Homepage 📄 JQuery Syntax 📖 Reading list
Social Media Site Add Post Post list Profile Welcome Sanket543



-profile picture clicked uploaded photo display



-Document clicked uploaded document display



-We can see Django admin page updates with above changes.

Apps

Facebook

Gmail

YouTube

Cricbuzz

HDFS Datanode inf...

Hadoop - Nameno...

YARN resource man...

Website Designing...

RunX - Homepage

jQuery Syntax

Reading list

Django administration

WELCOME, SANKET VIEW SITE / CHANGE PASSWORD / LOG OUT

Home / User / User profiles

AUTHENTICATION AND AUTHORIZATION

Groups + Add

Users + Add

USER

Posts + Add

User profiles + Add

Select user profile to change

ADD USER PROFILE +

Action: Go 0 of 2 selected

	ID	USERNAME	EMAIL	BIRTHDATE	MOBILENO	GENDER	CITY	PINCODE	PROFILE	DOCUMENT
<input type="checkbox"/>	2	Sanket543	sanketchheladiya19042001@gmail.com	19-04-2001	7043944540	M	Rajkot	360004	images/WIN_20200118_10_01_31_Pro.jpg	docs/CamScanner_11-10-2021_15.11.14_JFMbvOI.p
<input type="checkbox"/>	1	Sanket123	spatel19042001@gmail.com	19-Apr-200	7043526125	M	Rajkot	360004	images/Sanket_Photo.jpeg	docs/CamScanner_11-10-2021_15.11.14.pdf

2 user profiles

-Now Add Post

-addPost.html

```

{% extends 'base.html' %}
{% block title %}Add Post{% endblock title %}
{% block body %}
    <div class="container">
        <h2>Add Post Page</h2>
        <form method="post">
            {% csrf_token %}
            {{ form.as_p }}
            <button type="submit">Submit</button>
        </form>
    </div>
{% endblock body %}

```

127.0.0.1:8000

Apps

Facebook

Gmail

YouTube

Cricbuzz

HDFS Datanode inf...

Hadoop - Nameno...

YARN resource man...

Website Designing...

RunX - Homepage

jQuery Syntax

Reading list

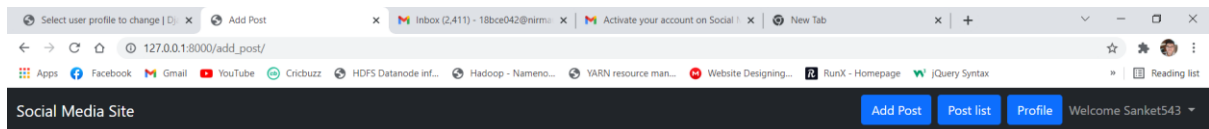
Social Media Site

Add Post

Post list

Profile

Welcome Sanket543



-showPost.html

```
{% extends 'base.html' %}
{% block title %}Show Post{% endblock title %}
{% block body %}
    <div class="container">
        <h2>Post List</h2>

        {% if stu %}
        <table class="table table-hover">
            <thead>
                <tr>
                    <th scope="col">Username</th>
                    <th scope="col">Post ID</th>
                    <th scope="col">Post Name</th>
                    <th scope="col">Post Description</th>
                    <th scope="col">Actions</th>
                </tr>
            </thead>
            <tbody>
                {% for st in stu %}
                <tr>
                    <th scope="row">{{st.postusername}}</th>
                    <td>{{st.id}}</td>
                    <td>{{st.postname}}</td>
                    <td>{{st.description}}</td>
                    <td>
                        {% comment %} {% url 'updatedata' st.id %} {%
                        <a href="{% url 'updatepost' st.id %}" class="btn
                        btn-warning btn-sm">Edit</a>
                        <form action="{% url 'deletepost' st.postname %}"
                        method="post" class="d-inline">
                            {% csrf_token %}
                            <input type="submit" value="Delete" class="btn
                        btn-danger btn-sm">
                        </form>
                    </td>
                </tr>
                {% endcomment %}
            </tbody>
        </table>
        {% else %}
        <p>No posts found.</p>
        {% endif %}
    </div>
{% endblock %}
```

```

                </td>
            </tr>
        {% endfor %}
    </tbody>
</table>
{% else %}
<h4 class="text-center alert alert-warning">No Records</h4>
{% endif %}

</div>

{% endblock body %}

```

Select user profile to change | D | Show Post | Inbox (2,411) - 18bce042@nirmal | Activate your account on Social | New Tab

127.0.0.1:8000/show_post/

Apps Facebook Gmail YouTube Cricbuzz HDFS Datanode inf... Hadoop - Nameno... YARN resource man... Website Designing... RunX - Homepage jQuery Syntax Reading list

Social Media Site [Add Post](#) [Post list](#) [Profile](#) Welcome Sanket543

Post List

Username	Post ID	Post Name	Post Description	Actions
Sanket543	1	Test_Upload_Post_By_Sanket	This is Testing for upload post	Edit Delete

-updatePost.html

```

{% extends 'base.html' %}
{% block title %}Update Post{% endblock title %}
{% block body %}
    <div class="container">
        <h2>Update Post Information</h2>
        <form method="post">
            {% csrf_token %}
            {{ form.as_p }}
            <button type="submit" class="btn btn-info">Submit</button>
            <a href="{% url 'user_post_list' %}" class="btn btn-info">Back
to Home</a>
        </form>
    </div>

{% endblock body %}

```

Select user profile to change | D | x Update Post x Inbox (2,411) - 18bce042@nirm... x Activate your account on Social | x Dashboard x +

127.0.0.1:8000/update_post/1

Apps Facebook Gmail YouTube Cricbuzz HDPS Datanode inf... Hadoop - Namen... YARN resource man... Website Designing... RunX - Homepage JQuery Syntax Reading list

Social Media Site Add Post Post list Profile Welcome Sanket543

Update Post Information

Postname:

Description:

Submit Back to Home

-Logout leads to home page


Select user profile to change | D | x Show Profile x Inbox (2,411) - 18bce042@nirm... x Activate your account on Social | x Dashboard x +

127.0.0.1:8000/user_profile/

Apps Facebook Gmail YouTube Cricbuzz HDPS Datanode inf... Hadoop - Namen... YARN resource man... Website Designing... RunX - Homepage JQuery Syntax Reading list

Social Media Site Add Post Post list Profile Welcome Sanket543

Logout

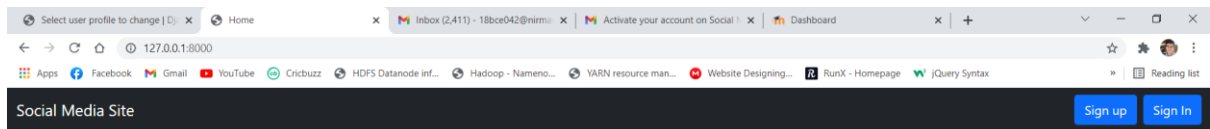


Sanket543

User Name	Sanket543
Email Address	sanketchheladiya19042001@gmail.com
Birth Date	19-04-2001
Mobile No	7043944540
Gender	M
City	Rajkot
Pincode	360004
Profile Picture	Click here
Document	Click here

Type here to search

27°C 20:10 11-11-2021



-----Thank You-----