



**INSTITUTE OF TECHNOLOGY,
NIRMA UNIVERSITY**

2CSDE86 Application Development Framework

PRACTICAL 4

Name : Jaimik S. Chauhan

Roll No. : 20BCE503

Semester : 7th

Teacher's Signature :

Aim

Apply concept of Class Based Views and demonstrate the CRUD operations for registration page designed. Also, design the app/module for posts.

Design a view, templates and filters for different modules. Ensure that your application has following features:

1. Login and registration page
2. The dashboard which displays the latest post
3. Every post can be liked, shared by others
4. Another user is permitted to comment on displayed post

1. ...Continue from Practical 3

2. CRUD Operation on User

/p3/settings.py

```
# Application definition

INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'social',
    'state',
    'posts'
]
```

/p3/urls.py

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

urlpatterns = [
    path('user/', include('social.urls')),
    path('admin/', admin.site.urls),
    path("", include('posts.urls')),
]
```

User App

/p3/social/models.py

```
import datetime
from django.db import models

# Create your models here.
class User(models.Model):
    CHOICE = (
        ("Male", "Male"),
        ("Female", "Female")
    )

    name = models.CharField(max_length=250)
    email = models.EmailField()
    password = models.CharField(max_length=18)
    phone = models.CharField(max_length=200)
    username = models.CharField(max_length=20)
    birthdate = models.DateField("Date", default=datetime.date.today())
    gender = models.CharField(choices=CHOICE, max_length=100)
```

/p3/social/admin.py

```
from django.contrib import admin

# jaimik : Admin@123
# Register your models here.
from .models import User
admin.site.register(User)
```

/p3/social/urls.py

```
from django.urls import path, include
from . import views

urlpatterns = [
    path('', views.home, name='home'),
    path('create/', views.CreateUser.as_view(), name="CreateUser"),
    path('list/', views.ListUser.as_view(), name='ListUsers'),
    path("user/<int:pk>", views.UserDetail.as_view(), name="UserDetail"),
    path('update/<int:pk>', views.UpdateUser.as_view(), name='UpdateUser'),
    path('delete/<int:pk>', views.RemoveUser.as_view(), name='RemoveUser')
]
```

/p3/social/views.py

```
from http.client import HTTPResponse
from django.shortcuts import render
from django.views.generic import ListView, DetailView
from django.views.generic.edit import CreateView, UpdateView, DeleteView
from .models import User
from django.urls import reverse_lazy
from django.http import HttpResponse
from django.contrib.admin.widgets import AdminDateWidget

# Create your views here.

# Base View
def home(request):
    # template_name = 'home.html'
    return render(request, 'templates/home.html')

# List Users View
class ListUser(ListView):
    model = User
    query_results = User.objects.all()
    template_name = 'ListUsers.html'

class UserDetail(DetailView):
    model = User
    template_name = 'UserDetail.html'

# Create User
class CreateUser(CreateView):
    model = User
    fields = '__all__'
    template_name = 'CreateUser.html'
    success_url = reverse_lazy('ListUsers')

    # to display date picker widget in form
    def get_form(self, form_class=None):
        form = super(CreateUser, self).get_form(form_class)
        form.fields['birthdate'].widget = AdminDateWidget(attrs={'type':
'date'})
        return form

# Update User View
class UpdateUser(UpdateView):
    model = User
    fields = '__all__'
    template_name = 'UpdateUser.html'
    success_url = reverse_lazy('ListUsers')

# Remove User View
class RemoveUser(DeleteView):
    model = User
    template_name = 'RemoveUser.html'
    success_url = reverse_lazy('ListUsers')
```

```
<html lang="en">
<head>
  <meta charset="UTF-8">
  <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0/dist/css/bootstrap.min.css"
    rel="stylesheet"
    integrity="sha384-
wEmeIV1mKuiNpC+IOBJI7aAzPcEZeedi5yW5f2yOq55WWLwNGmvvx4Um1vskeMj0"
    crossorigin="anonymous">
  <title>Django Cards</title>
</head>
<body>
  <nav class="navbar navbar-expand-lg navbar-light bg-light">
    <a class="navbar-brand" href="/" style="margin:0 2%">Social Media
App</a>
    <button class="navbar-toggler" type="button" data-toggle="collapse"
data-target="#navbarSupportedContent" aria-controls="navbarSupportedContent"
aria-expanded="false" aria-label="Toggle navigation">
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse" style=""
id="navbarSupportedContent">
      <ul class="navbar-nav mr-auto">
        <li class="nav-item active">
          <a class="nav-link" href="#">Home <span class="sr-
only">(current)</span></a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="user/list/">Profile</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="/">Posts</a>
        </li>
      </ul>
    </div>
  </nav>
  <div style="display: flex; flex-direction: row; flex-wrap: wrap">
    {% comment %} {% block %}
    {% endblock %} {% endcomment %}
  </div>
</body>
</html>
```

/p3/social/templates/CreateUser.html

```
{% include 'home.html' %}
<h1>User Sign In Page</h1>
<form method="POST">
  {% csrf_token %}
  {{form.as_p}}
  <button type="submit">Submit</button>
</form>
```

Social Media App Home (current) Profile Posts

User Sign In Page

Name:

Email:

Password:

Phone:

Username:

Date: 15-11-2022

Gender:

127.0.0.1:8000/user/create/

NETWORK
Upload 0.0 B/s
Download 0.0 B/s

ENG IN 19:32 15-11-2022

/p3/social/templates/ListUsers.html

```
<!-- CSS only -->
{% include 'home.html' %}

<h1> Registered User </h1>
<a href="/user/create/" class="btn btn-primary">Create User </a>
<table class="table">
  <thead>
    <tr>
      <th>Name</th>
      <th>Email</th>
      <th>Phone</th>
      <th>BirthDate </th>
      <th>Gender</th>
      <th>Edit</th>
      <th>Delete </th>
      <th>Detail </th>
    </tr>
  </thead>
```

```

<tbody>
  {% for users in object_list %}
  <tr>
    <td>{{users.name}}</td>
    <td>{{users.email}}</td>
    <td>{{users.phone}}</td>
    <td>{{users.birthdate}}</td>
    <td>{{users.gender}}</td>
    <td>
      <a href="{% url 'UpdateUser' users.id %}">Update</a>
    </td>
    <td>
      <a href="{% url 'RemoveUser' users.id %}">Delete </a>
    </td>
    <td>
      <a href="{% url 'UserDetail' users.id %}">Details </a>
    </td>
  </tr>
  {% endfor %}
</tbody>
</table>

```

The screenshot shows a web browser window with the address bar displaying '127.0.0.1:8000/user/list/'. The page title is 'Social Media App' with navigation links for 'Home (current)', 'Profile', and 'Posts'. Below the title is a section for 'Registered User' with a 'Create User' button. A table lists the details of a registered user.

Name	Email	Phone	BirthDate	Gender	Edit	Delete	Detail
Jaimik	jaimikk@gmail.com	8401563076	April 6, 2002	Male	Update	Delete	Details

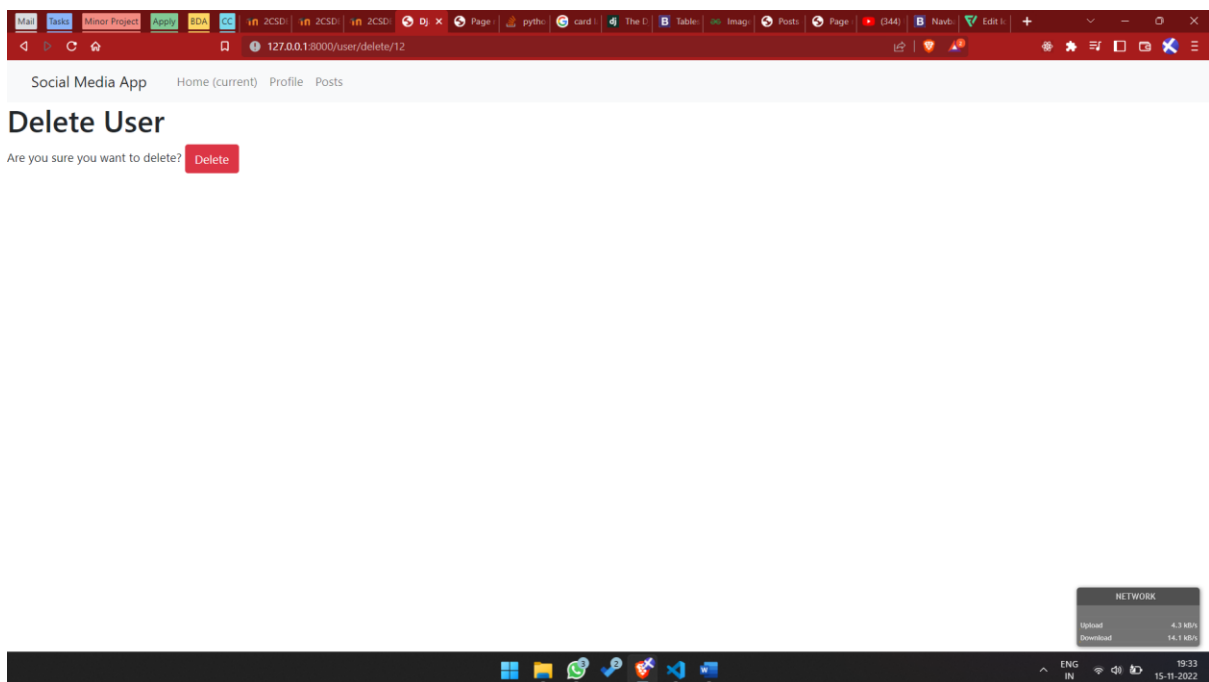
At the bottom of the browser window, a 'NETWORK' panel shows upload and download speeds, and the Windows taskbar displays the time as 19:33 on 15-11-2022.

/p3/social/templates/RemoveUser.html

```
{% include 'home.html' %}

<h1> Delete User </h1>

<form method="POST">
    {% csrf_token %}
    Are you sure you want to delete?
    <button type="submit" class="btn btn-danger">Delete</button>
</form>
```



/p3/social/templates/UpdateUser.html

```
{% include 'home.html' %}

<h1>User Sign In Page</h1>
<form method="POST">
    {% csrf_token %}
    {{form.as_p}}
    <button type="submit">Update</button>
</form>
```

Social Media App Home (current) Profile Posts

User Update Page

Name:

Email:

Password:

Phone:

Username:

Date:

Gender:

127.0.0.1:8000/user/update/12

NETWORK

Upload	371.0 B/s
Download	123.0 B/s

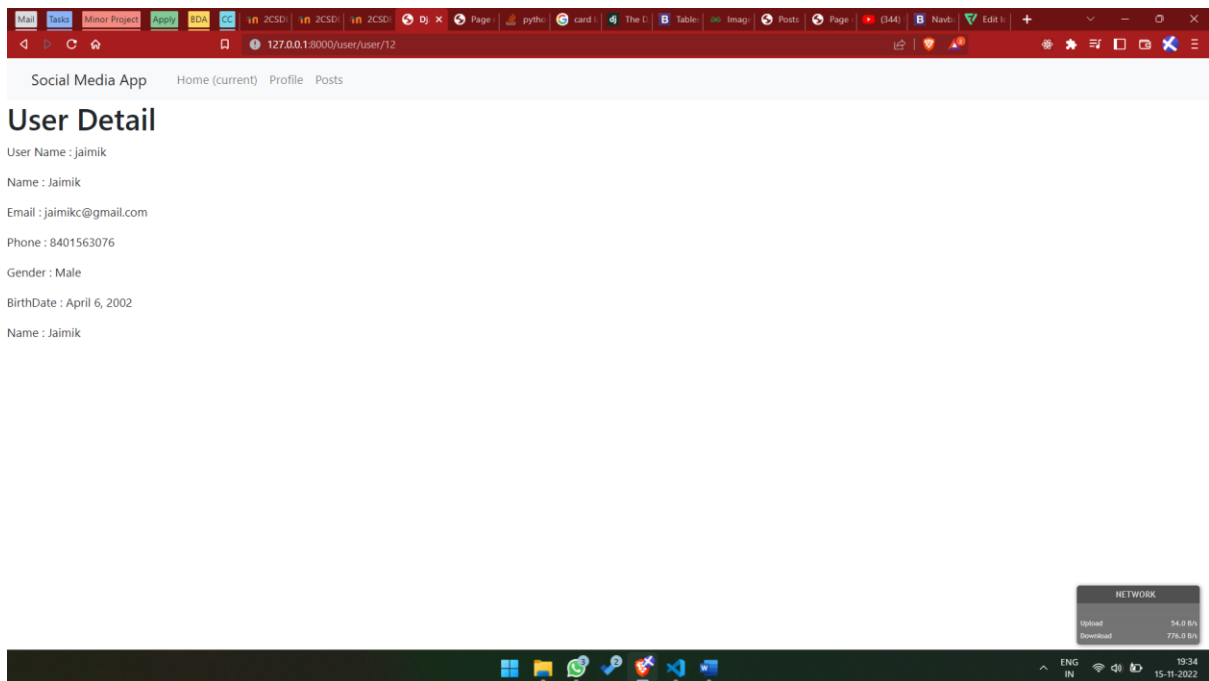
ENG IN 19:34 15-11-2022

/p3/social/templates/UserDetail.html

```
{% include 'home.html' %}

<h1> User Detail </h1>

<p> User Name : {{object.username}}</p>
<p> Name : {{object.name}}</p>
<p> Email : {{object.email}}</p>
<p> Phone : {{object.phone}}</p>
<p> Gender : {{object.gender}}</p>
<p> BirthDate : {{object.birthdate}}</p>
<p> Name : {{object.name}}</p>
```



Post App

/p3/posts/models.py

```
from django.db import models

# Create your models here.

class Posts(models.Model):
    title = models.CharField(max_length=250)
    description = models.CharField(max_length=250)
    like = models.IntegerField(max_length=10, default=0, editable=False)
    dislike = models.IntegerField(max_length=10, default=0, editable=False)
    image = models.ImageField(max_length=200, upload_to='uploads/')
```

/p3/posts/urls.py

```
from django.urls import path
from . import views
from django.conf import settings
from django.conf.urls.static import static

urlpatterns = [
    path('', views.ListPosts.as_view(), name="ListPosts"),
    path('create/', views.CreatePost.as_view(), name="CreatePost"),
    path('update/<int:pk>', views.UpdatePost.as_view(), name="UpdatePost"),
    path('delete/<int:pk>', views.RemovePost.as_view(), name="RemovePost")
]+ static(settings.MEDIA_URL, document_root=settings.MEDIA_URL)
```

/p3/posts/admin.py

```
from django.contrib import admin

# Register your models here.
from .models import Posts
admin.site.register(Posts)
```

/p3/posts/views.py

```
from django.shortcuts import render
from django.shortcuts import render
from django.views.generic import ListView, DetailView
from django.views.generic.edit import CreateView, UpdateView, DeleteView
from .models import Posts
from django.urls import reverse_lazy
from django.http import HttpResponseRedirect

# Create your views here.

class ListPosts(ListView):
    model = Posts
    query_results = Posts.objects.all()
    template_name = "ListPosts.html"

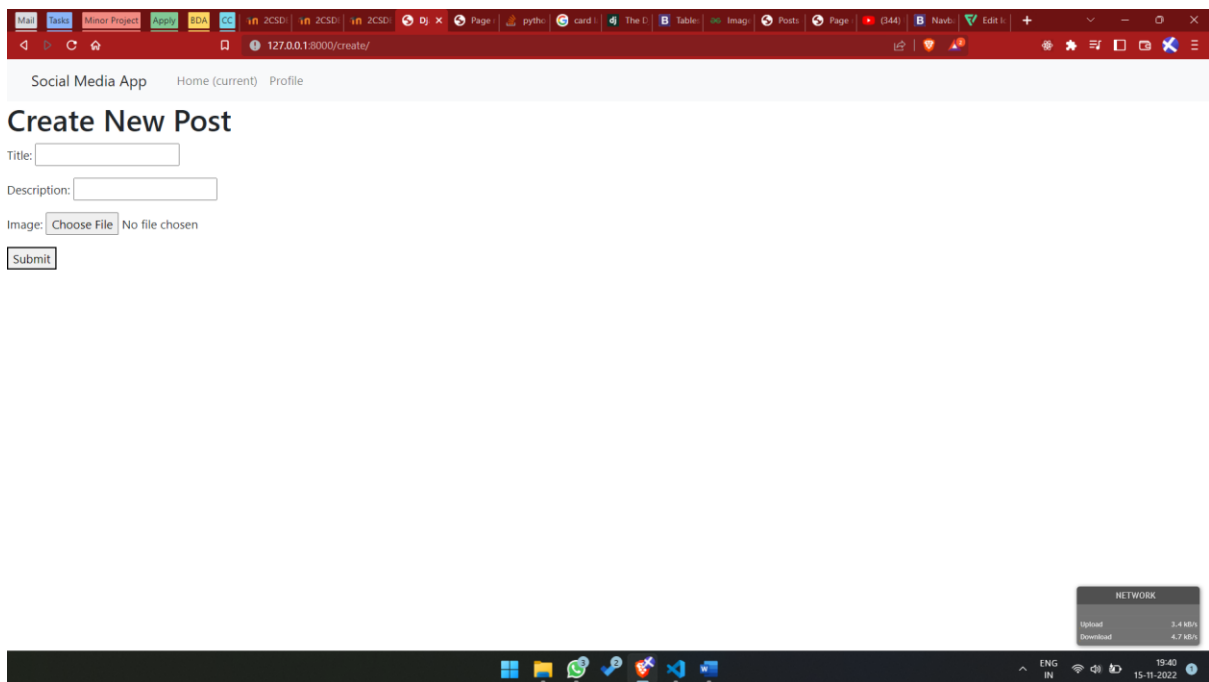
class CreatePost(CreateView):
    model = Posts
    fields = '__all__'
    template_name = 'CreatePost.html'
    success_url = reverse_lazy('ListPosts')

class UpdatePost(UpdateView):
    model = Posts
    fields = '__all__'
    template_name = 'UpdatePost.html'
    success_url = reverse_lazy('ListPosts')

class RemovePost(DeleteView):
    model= Posts
    template_name = 'RemovePost.html'
    success_url = reverse_lazy('ListPosts')
```

/p3/posts/templates/CreatePost.html

```
{% include 'index.html' %}
<h1>Create New Post</h1>
<form method="POST" enctype="multipart/form-data">
    {% csrf_token %}
    {{form.as_p}}
    <button type="submit">Submit</button>
</form>
```



```
<html lang="en">
<head>
  <meta charset="UTF-8">
  <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0/dist/css/bootstrap.min.css"
rel="stylesheet"
integrity="sha384-
wEmeIV1mKuiNpC+IOBjI7aAzPcEZeedi5yW5f2yOq55WWLwNGmvvx4Um1vskeMj0"
crossorigin="anonymous">
  <title>Django Cards</title>
</head>
<body>
  <nav class="navbar navbar-expand-lg navbar-light bg-light">
    <a class="navbar-brand" href="#" style="margin:0 2%">Social Media
App</a>
    <button class="navbar-toggler" type="button" data-toggle="collapse"
data-target="#navbarSupportedContent" aria-controls="navbarSupportedContent"
aria-expanded="false" aria-label="Toggle navigation">
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse" style=""
id="navbarSupportedContent">
      <ul class="navbar-nav mr-auto">
        <li class="nav-item active">
          <a class="nav-link" href="#">Home <span class="sr-
only">(current)</span></a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="user/list/">Profile</a>
        </li>
      </ul>
    </div>
  </nav>
  <div style="display: flex; flex-direction: row; flex-wrap: wrap">
    {% comment %} {% block %}
    {% endblock %} {% endcomment %}
  </div>
</body>
</html>
```

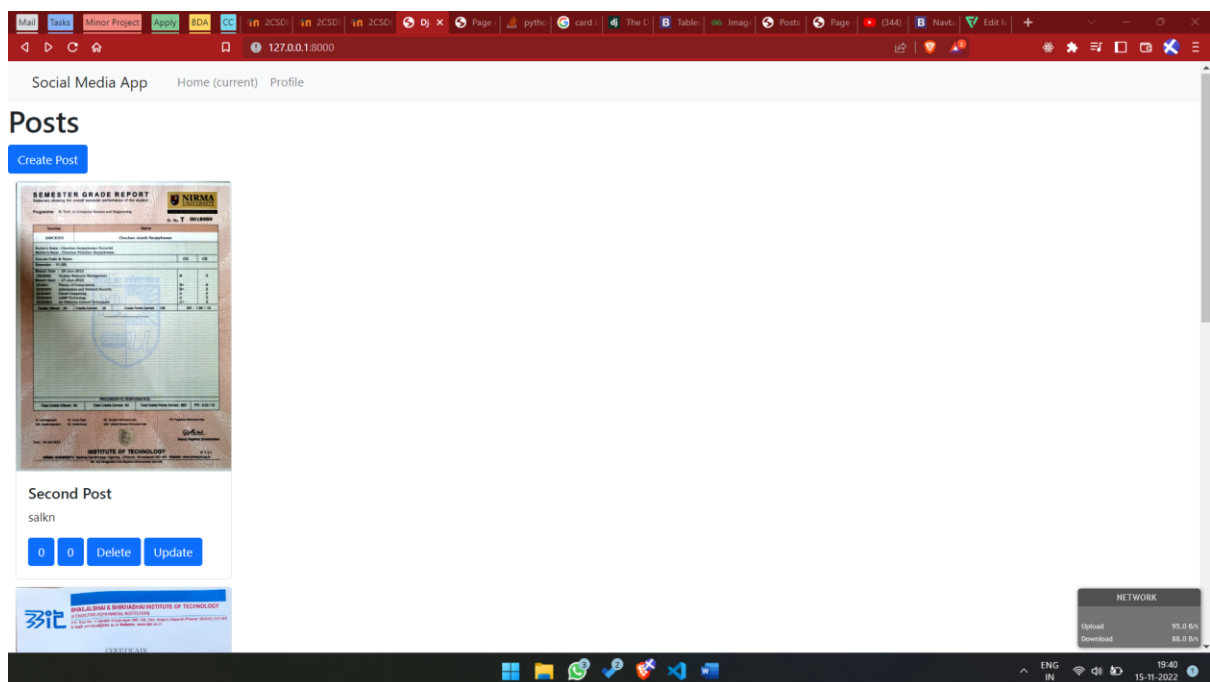
/p3/posts/templates/ListPosts.html

```
<!-- CSS only -->

{% include 'index.html' %}

<div>
  <h1> Posts </h1>
  <a class="btn btn-primary" href="/create">Create Post</a>
</div>

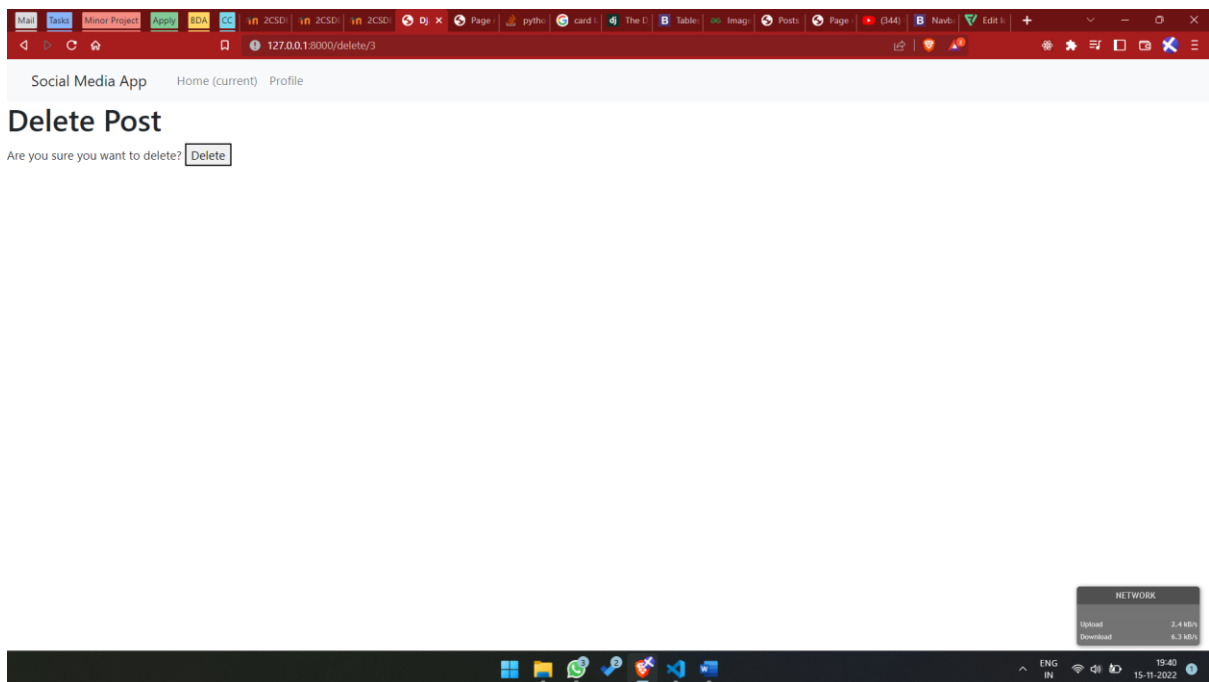
{% for posts in object_list %}
{% comment %} D:\BE\Sem-7\ADF\LAB\p3\media {% endcomment %}
<div class="card" style="width: 18rem; margin: 10px">
  
  <div class="card-body">
    <div style="display: flex; flex-direction: row">
      <h5 class="card-title">{{posts.title}}</h5>
    </div>
    <p class="card-text">{{posts.description}}</p>
    <a href="/like" class="btn btn-primary">{{posts.like}}</a>
    <a href="/dislike" class="btn btn-primary">{{posts.dislike}}</a>
    <a href="{% url 'RemovePost' posts.id %}" class="btn btn-
primary">Delete </a>
    <a href="{% url 'UpdatePost' posts.id %}" class="btn btn-
primary">Update</a>
  </div>
</div>
{% endfor %}
```



/p3/posts/templates/RemovePost.html

```
{% include 'index.html' %}
<h1> Delete Post </h1>

<form method="POST" enctype="multipart/form-data">
    {% csrf_token %}
    Are you sure you want to delete?
    <button type="submit">Delete</button>
</form>
```



/p3/posts/templates/UpdatePost.html

```
{% include 'index.html' %}
<h1>Post Update Page</h1>
<form method="POST" enctype="multipart/form-data">
    {% csrf_token %}
    {{form.as_p}}
    <button type="submit">Update</button>
</form>
```

