

**INSTITUTE OF TECHNOLOGY,**

**NIRMA UNIVERSITY**

**2CSDE86 Application Development Framework**

**PRACTICAL 5**

**Name :** Jaimik S. Chauhan

**Roll No. :** 20BCE503

**Semester :** 7th

**Teacher’s Signature :**

**Aim**

Continue with the social media application using the Django platform.

Create Forms File in Social Media Application

1. **…Continue from Practical 4**
2. **Django Forms in User App**

## /p3/social/forms.py

from django import forms

from .models import User

class CreateUserForm(forms.ModelForm):

    class Meta:

        model = User

        fields= '\_\_all\_\_'

## /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, FormView

from .models import User

from django.urls import reverse\_lazy

from django.http import HttpResponse

from django.contrib.admin.widgets import AdminDateWidget

from social.forms import CreateUserForm

from django.shortcuts import render, redirect

from .models import User

from django.contrib import messages

from django import forms

# 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

    form\_class = CreateUserForm

    # 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\_\_'

    form\_class = CreateUserForm

    template\_name = 'UpdateUser.html'

    success\_url = reverse\_lazy('ListUsers')

# Remove User View

class RemoveUser(DeleteView):

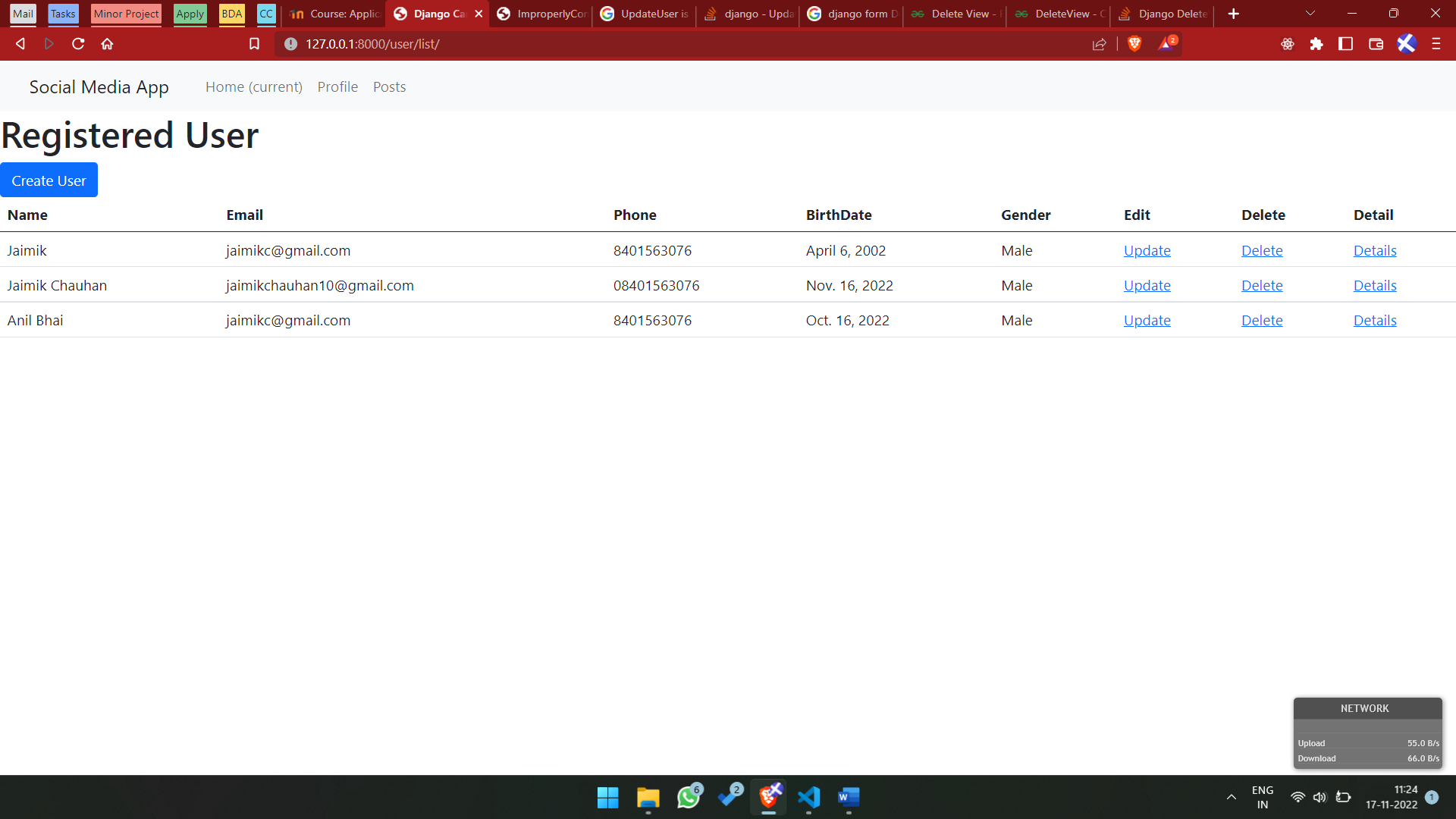
    model = User

    form\_class = CreateUserForm

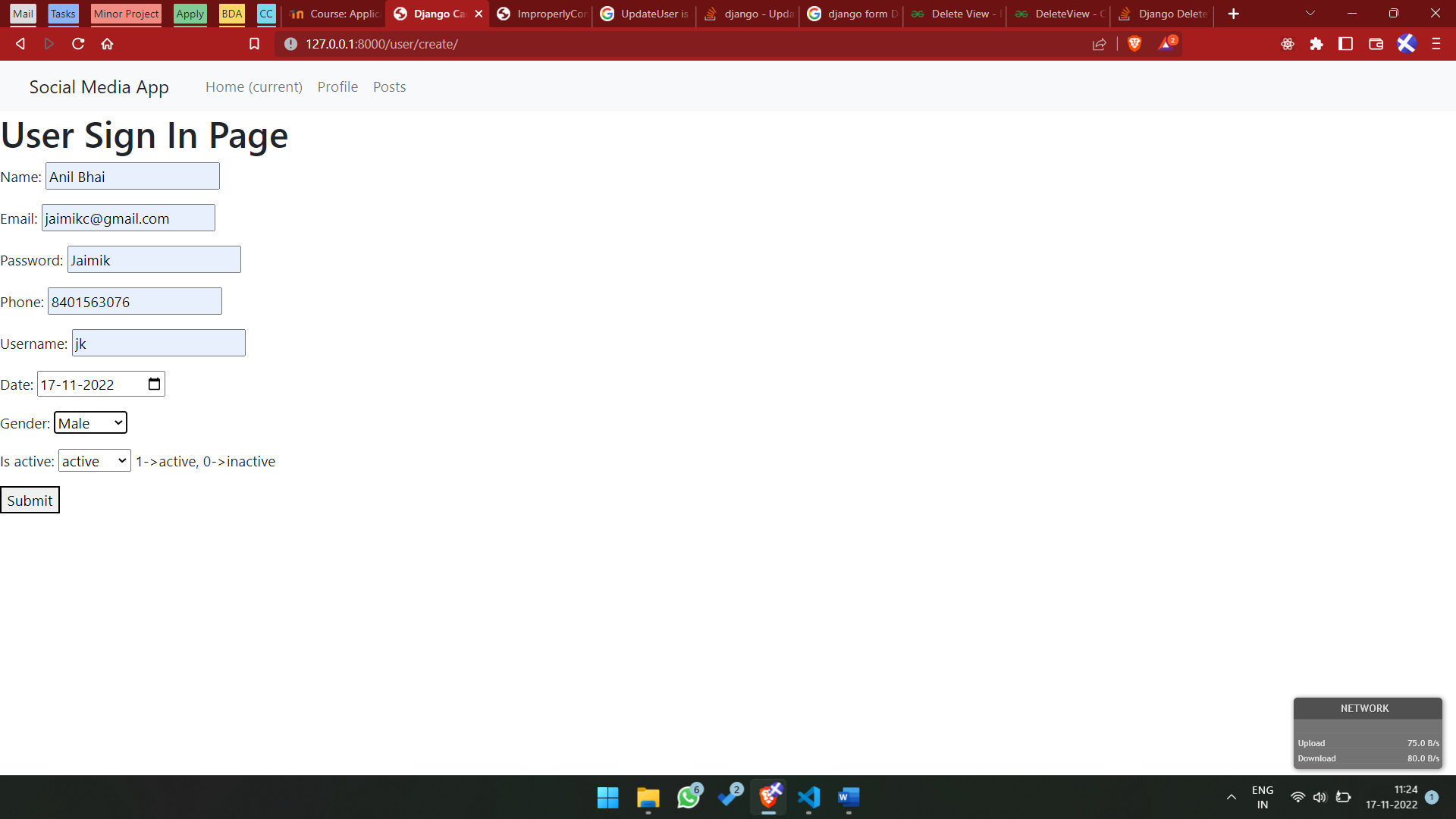
    template\_name = 'RemoveUser.html'

    success\_url = reverse\_lazy('ListUsers')

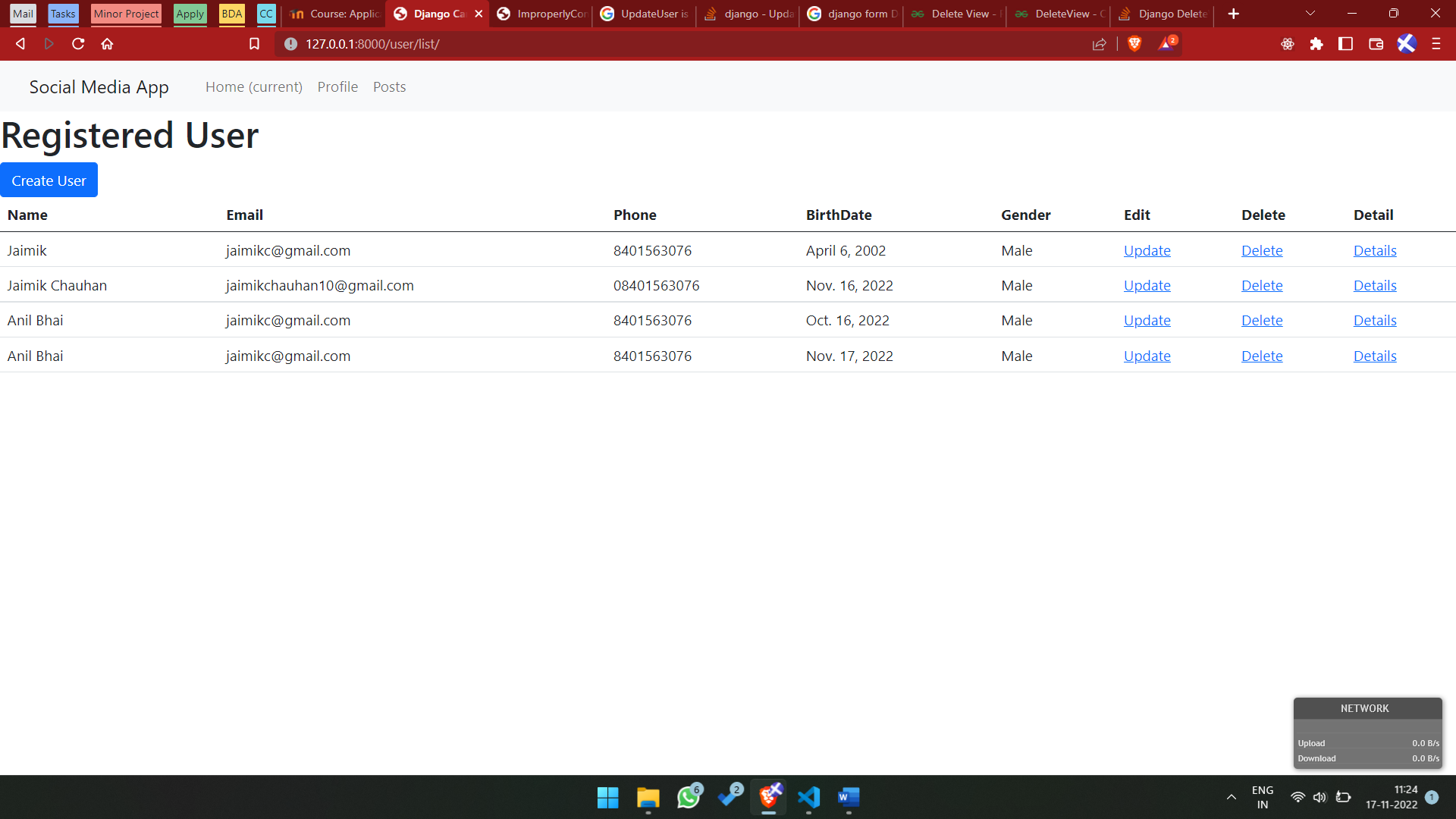
**Output :**



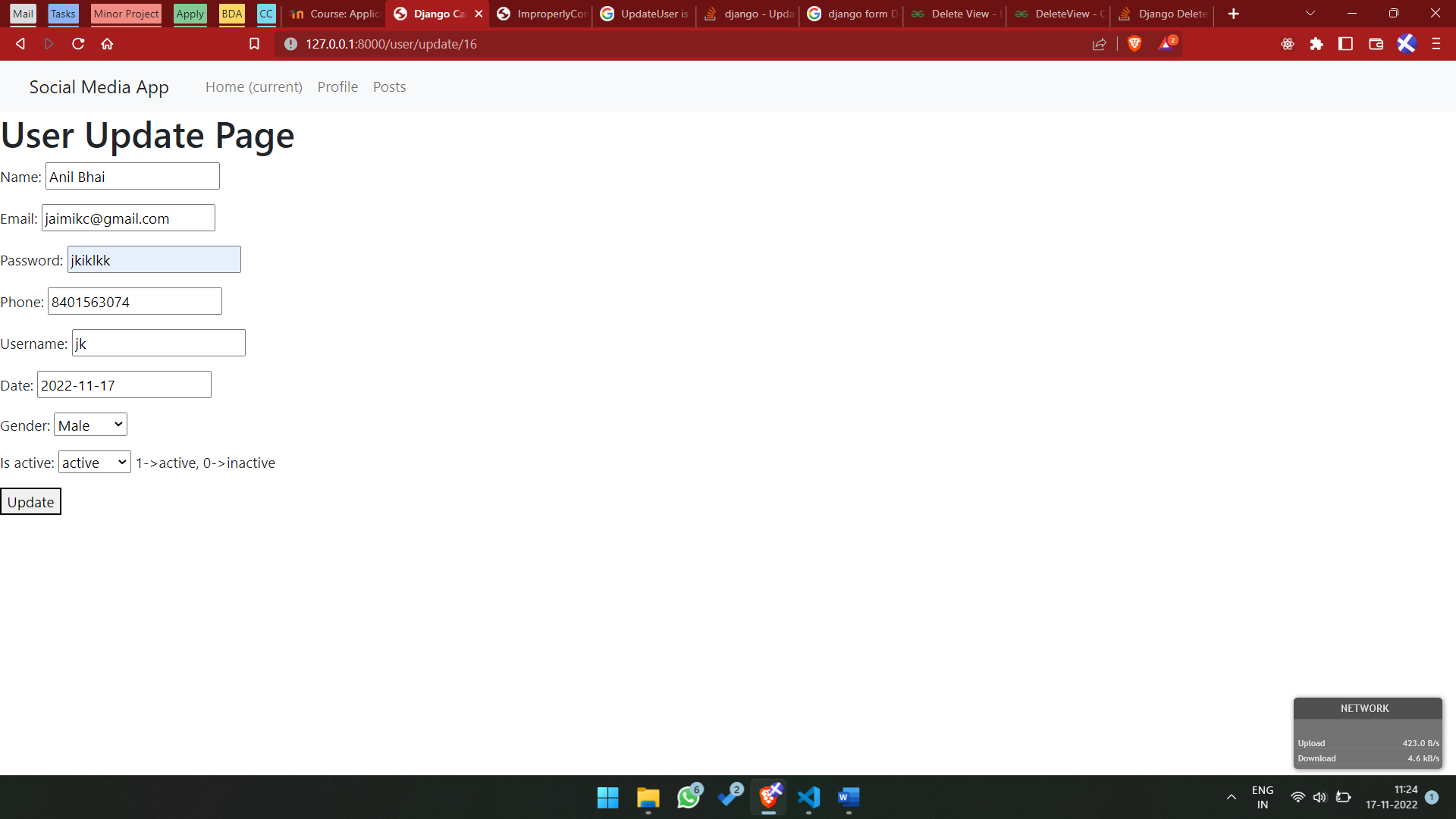
Output 1. Home Page For User



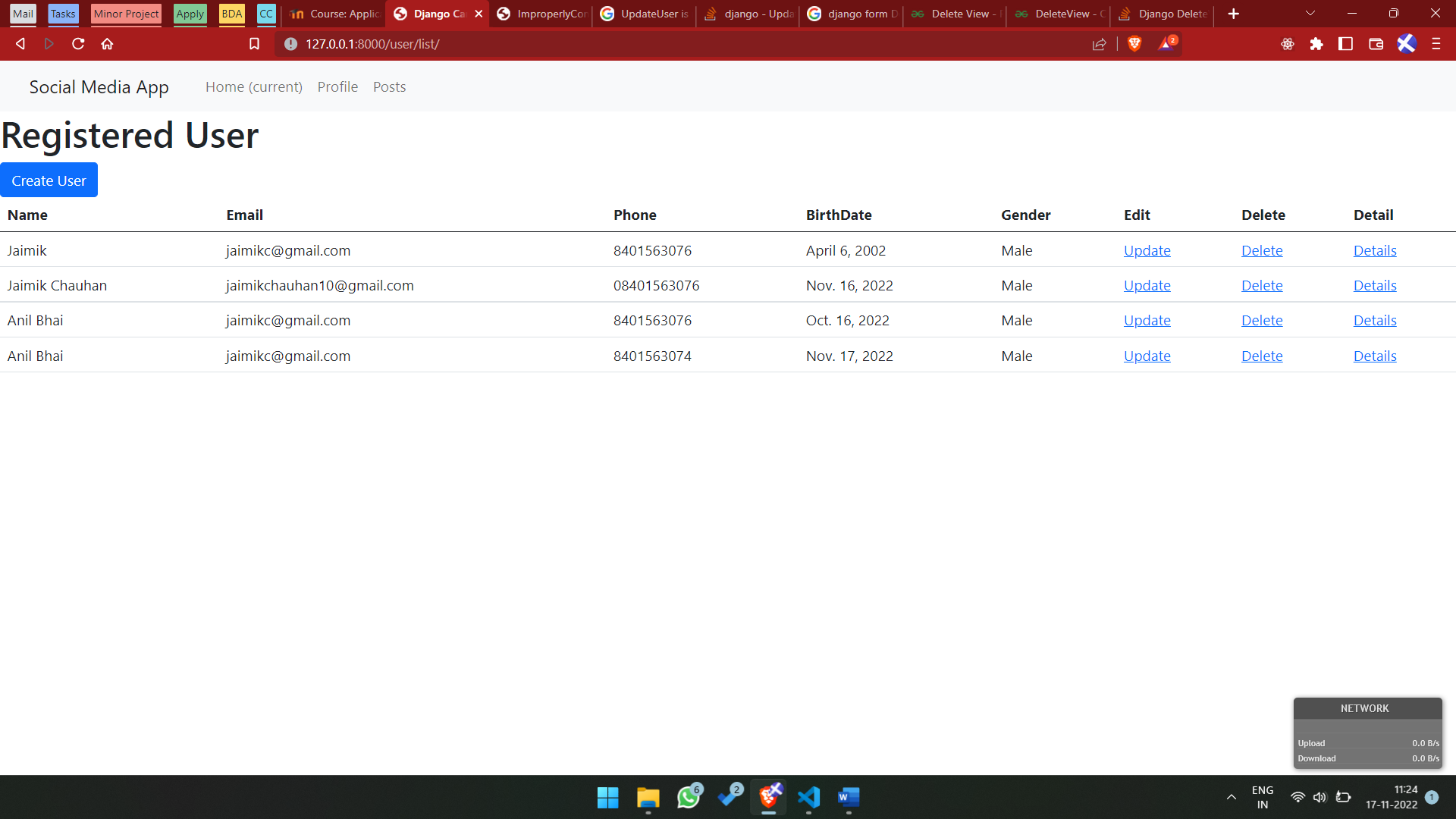
Output 1. User Creation Page



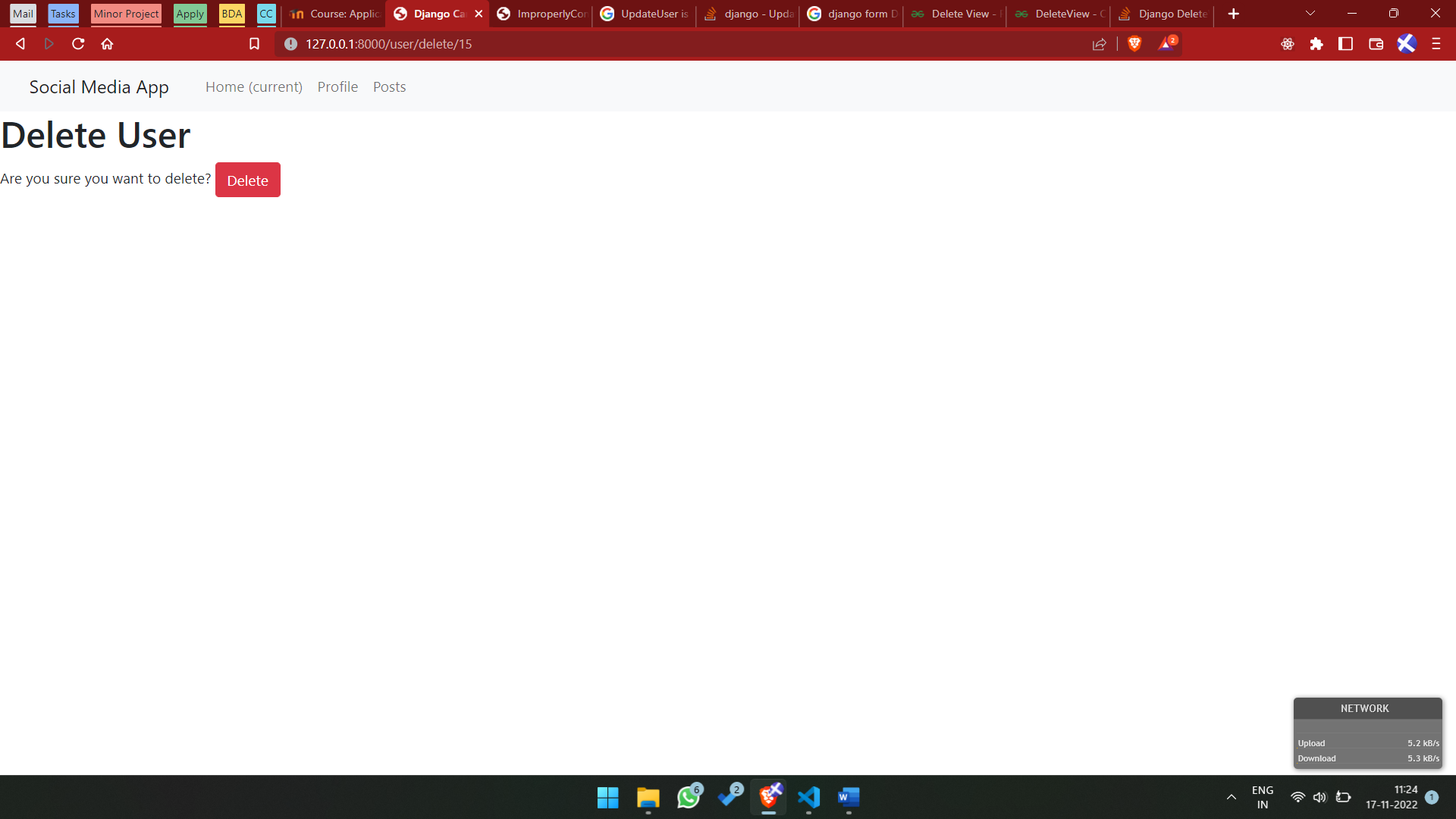
Output 1. New User Created details



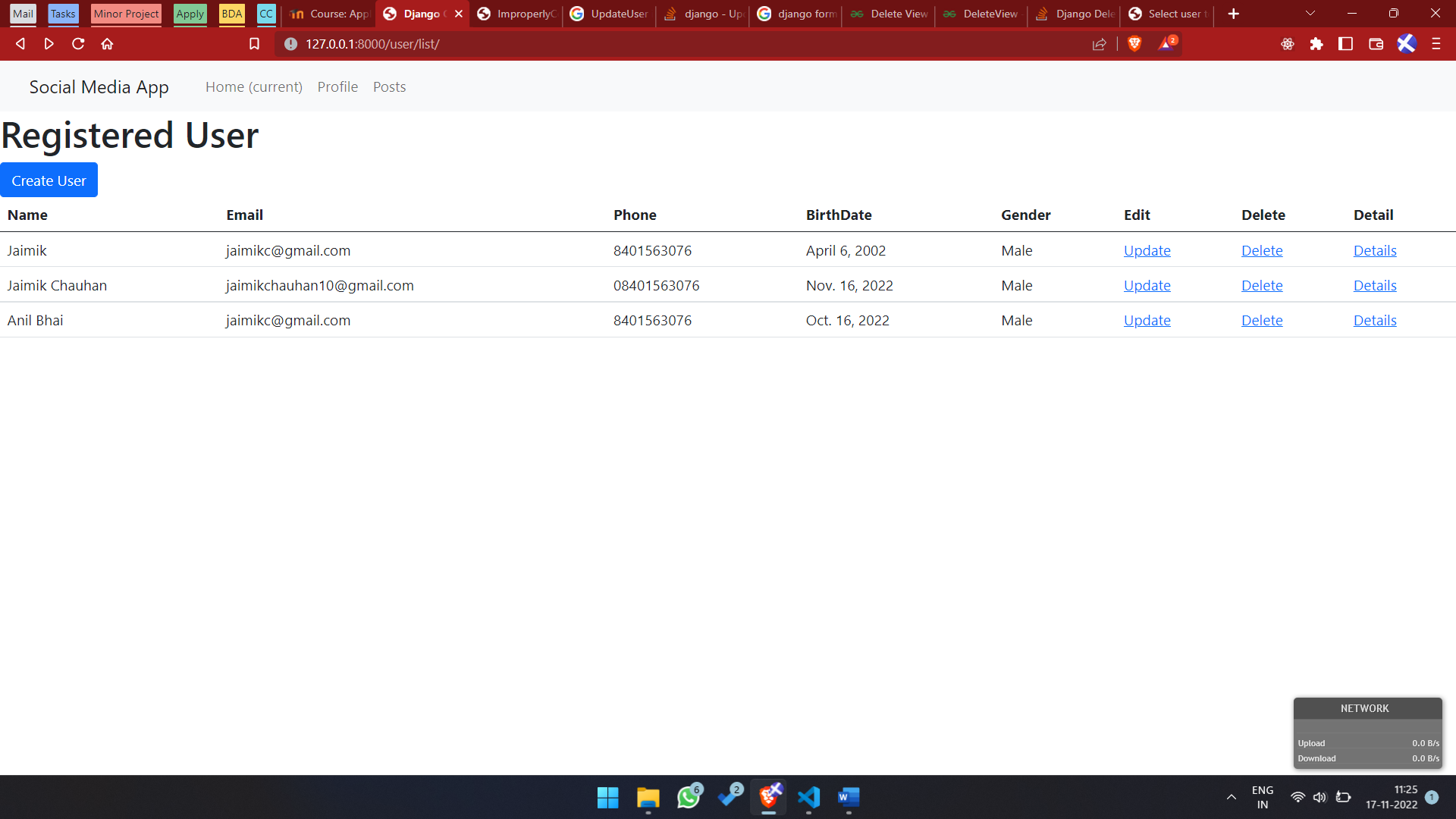
Output 1. Update User Details Page



Output 1. After Updating User Details



Output 1. Delete User Page



Output 1. After Deletion of User

1. **Forms.py in Post App**

## /p3/posts/forms.py

from django import forms

from .models import Posts

class CreatePostForm(forms.ModelForm):

    class Meta:

        model = Posts

        fields= '\_\_all\_\_'

## /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 HttpResponse

from .forms import CreatePostForm

class ListPosts(ListView):

    model = Posts

    query\_resuls = Posts.objects.all()

    template\_name = "ListPosts.html"

class CreatePost(CreateView):

    # model = Posts

    # fields = '\_\_all\_\_'

    form\_class = CreatePostForm

    template\_name = 'CreatePost.html'

    success\_url = reverse\_lazy('ListPosts')

class UpdatePost(UpdateView):

    model = Posts

    # fields = '\_\_all\_\_'

    form\_class = CreatePostForm

    template\_name = 'UpdatePost.html'

    success\_url = reverse\_lazy('ListPosts')

class RemovePost(DeleteView):

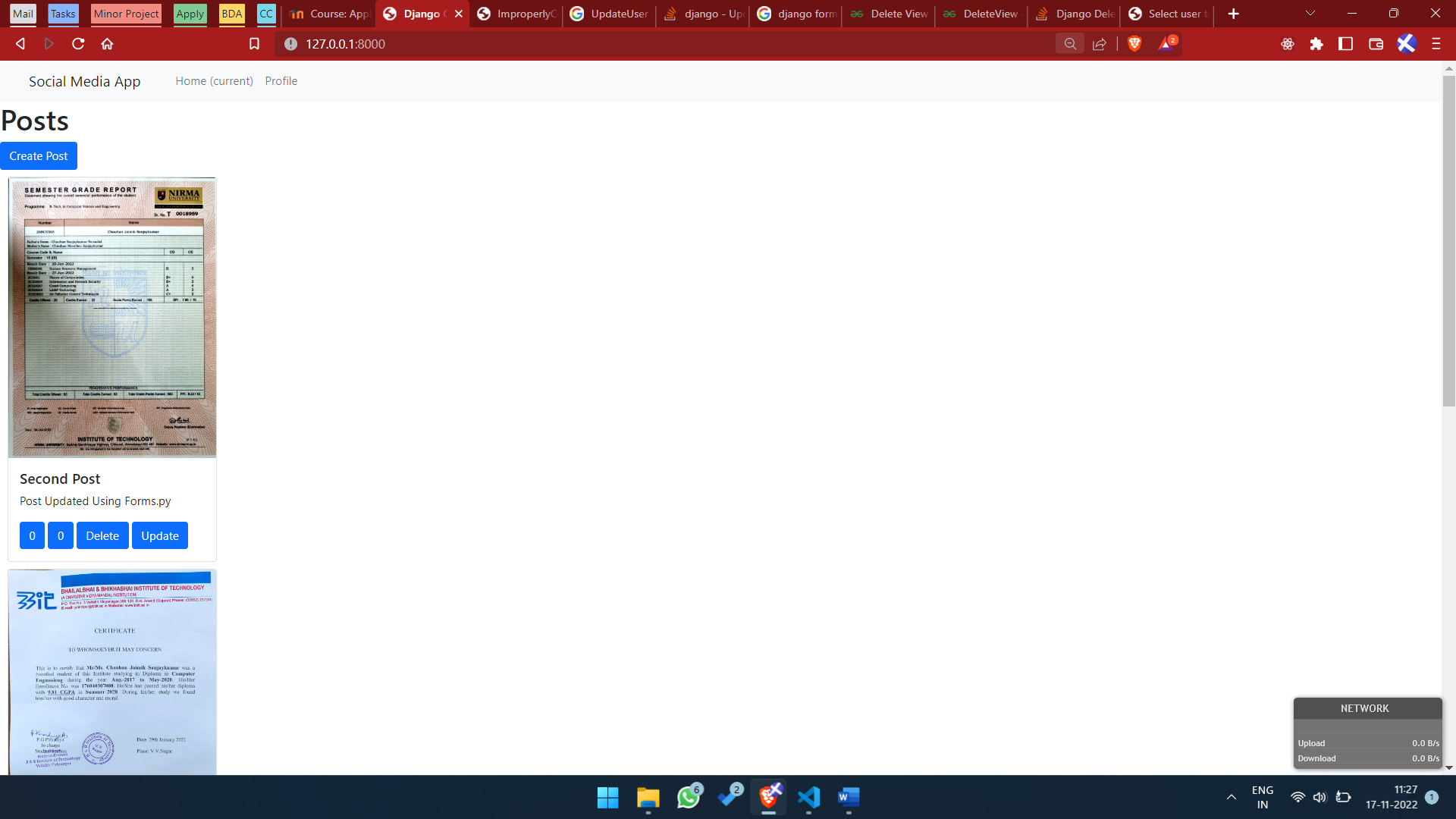
    model= Posts

    form\_class = CreatePostForm

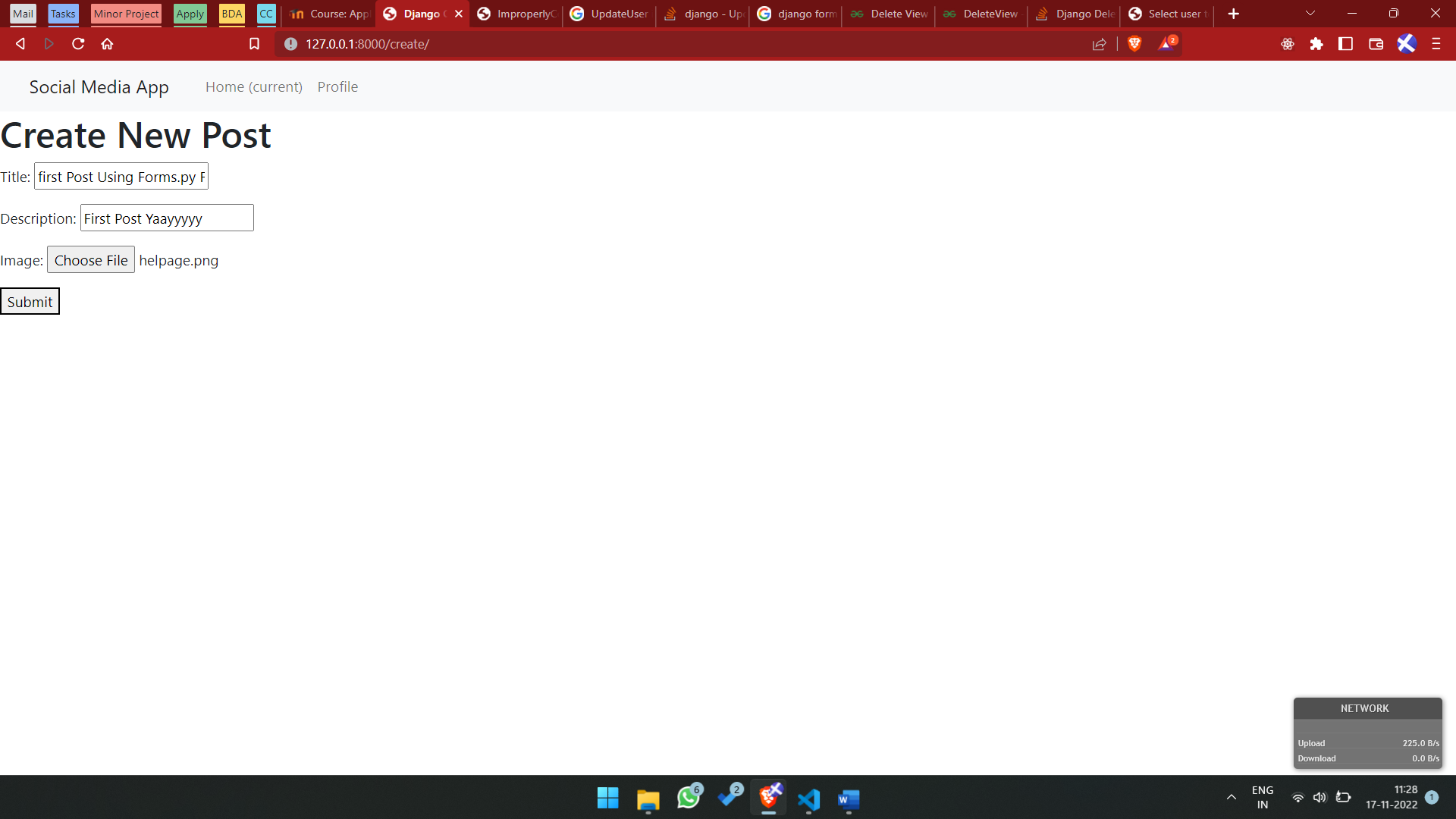
    template\_name = 'RemovePost.html'

    success\_url = reverse\_lazy('ListPosts')

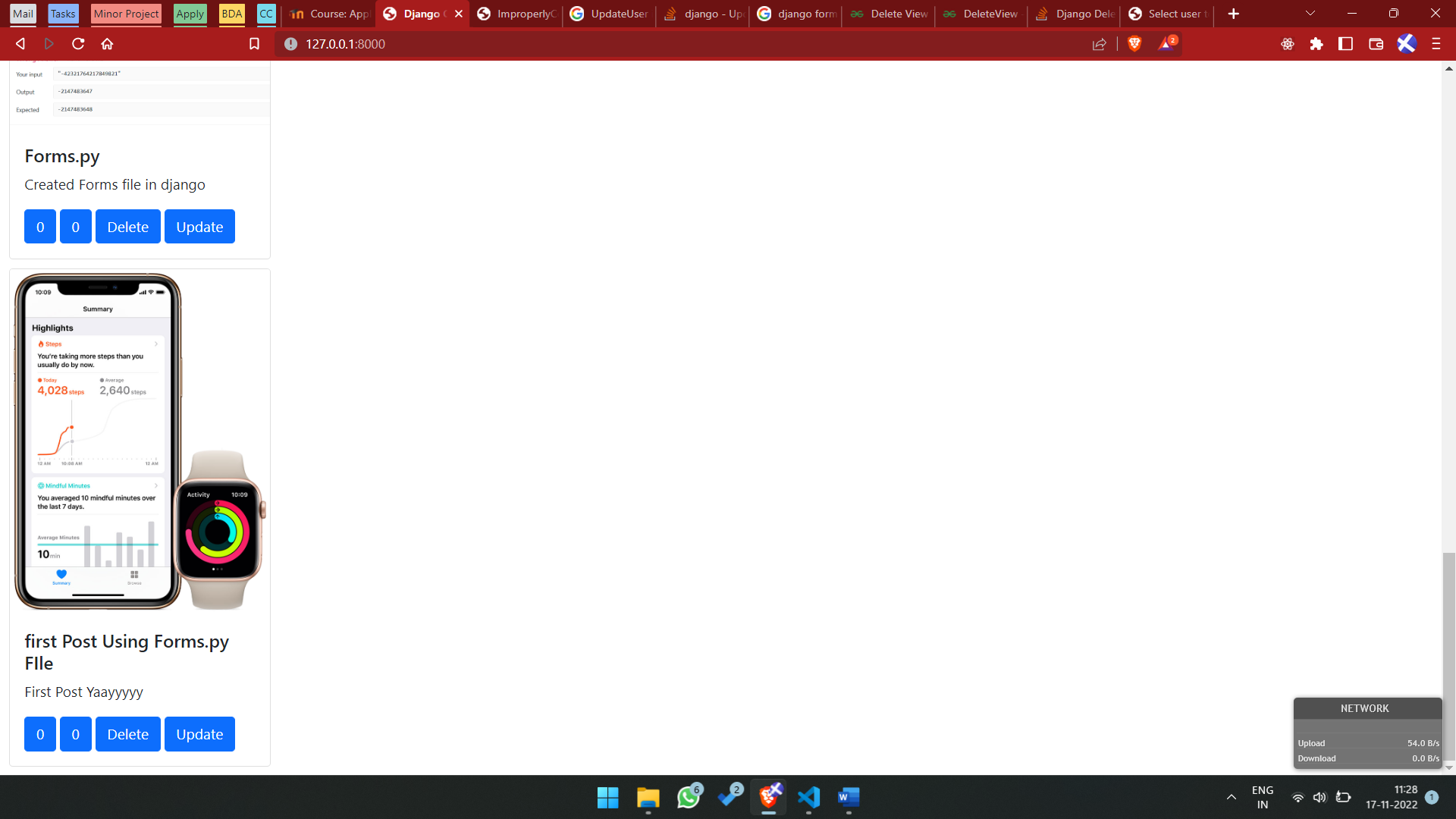
**Output :**



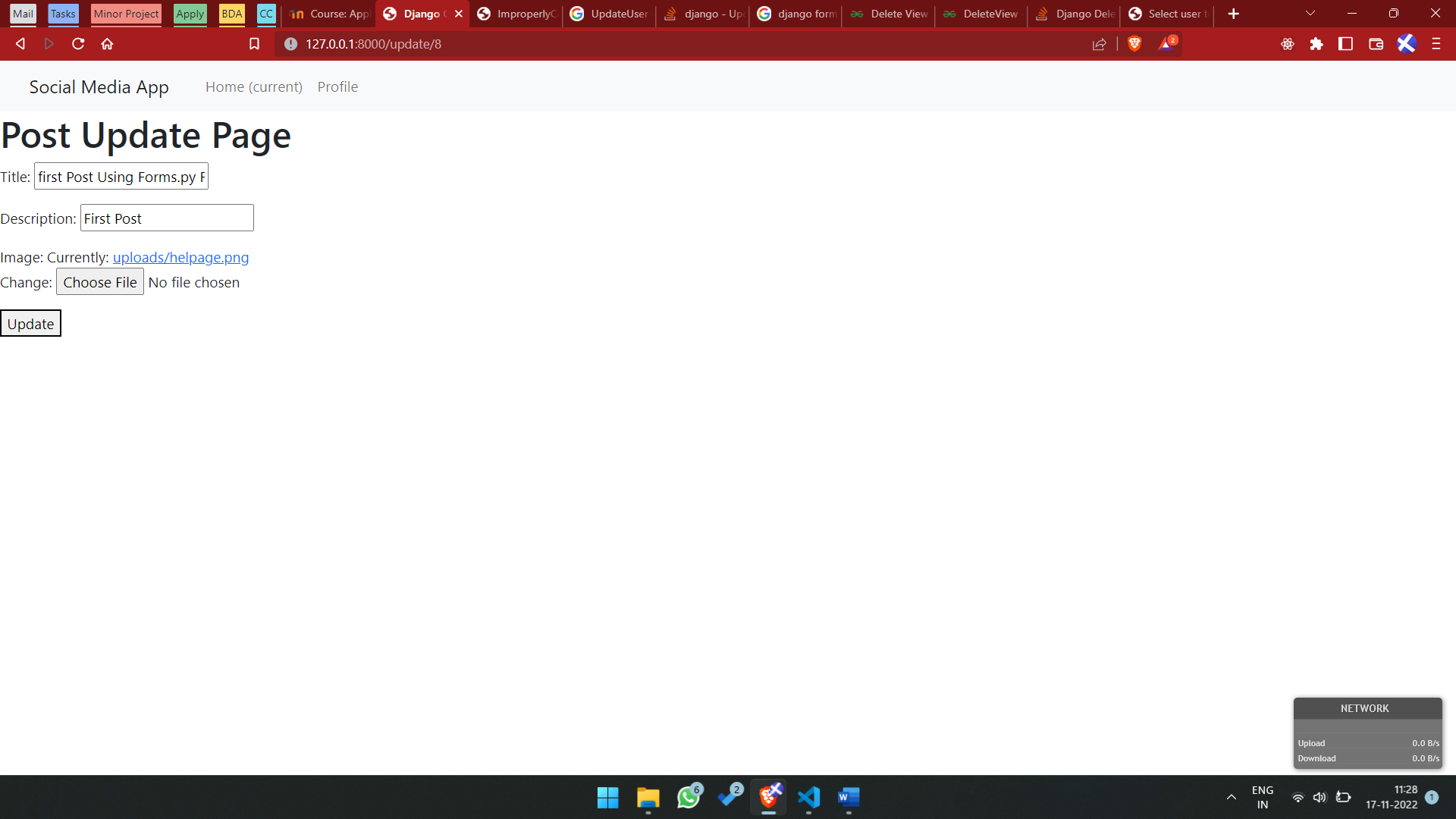
Output 2. Post Home Page



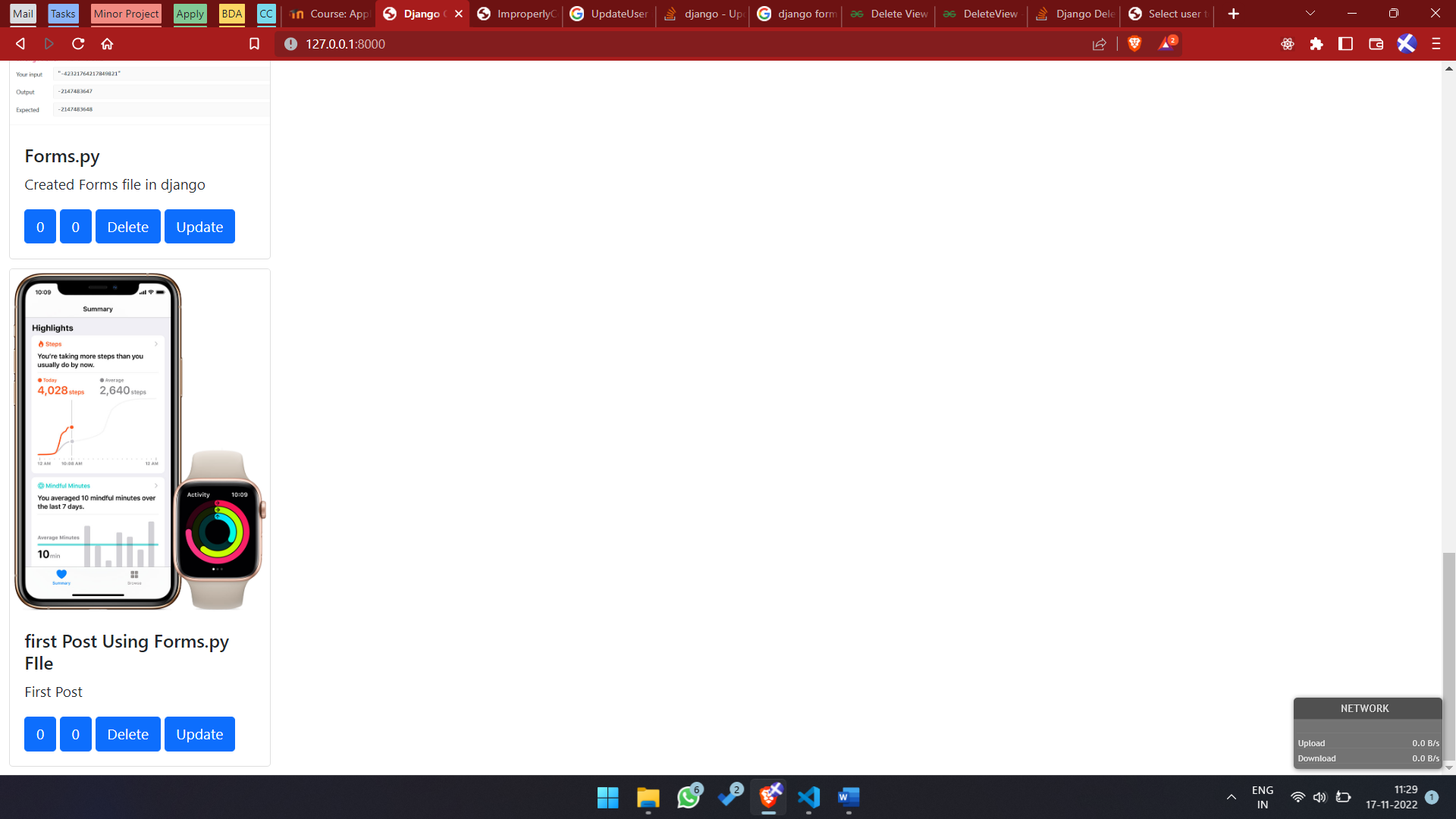
Output 2. Create New Post Form



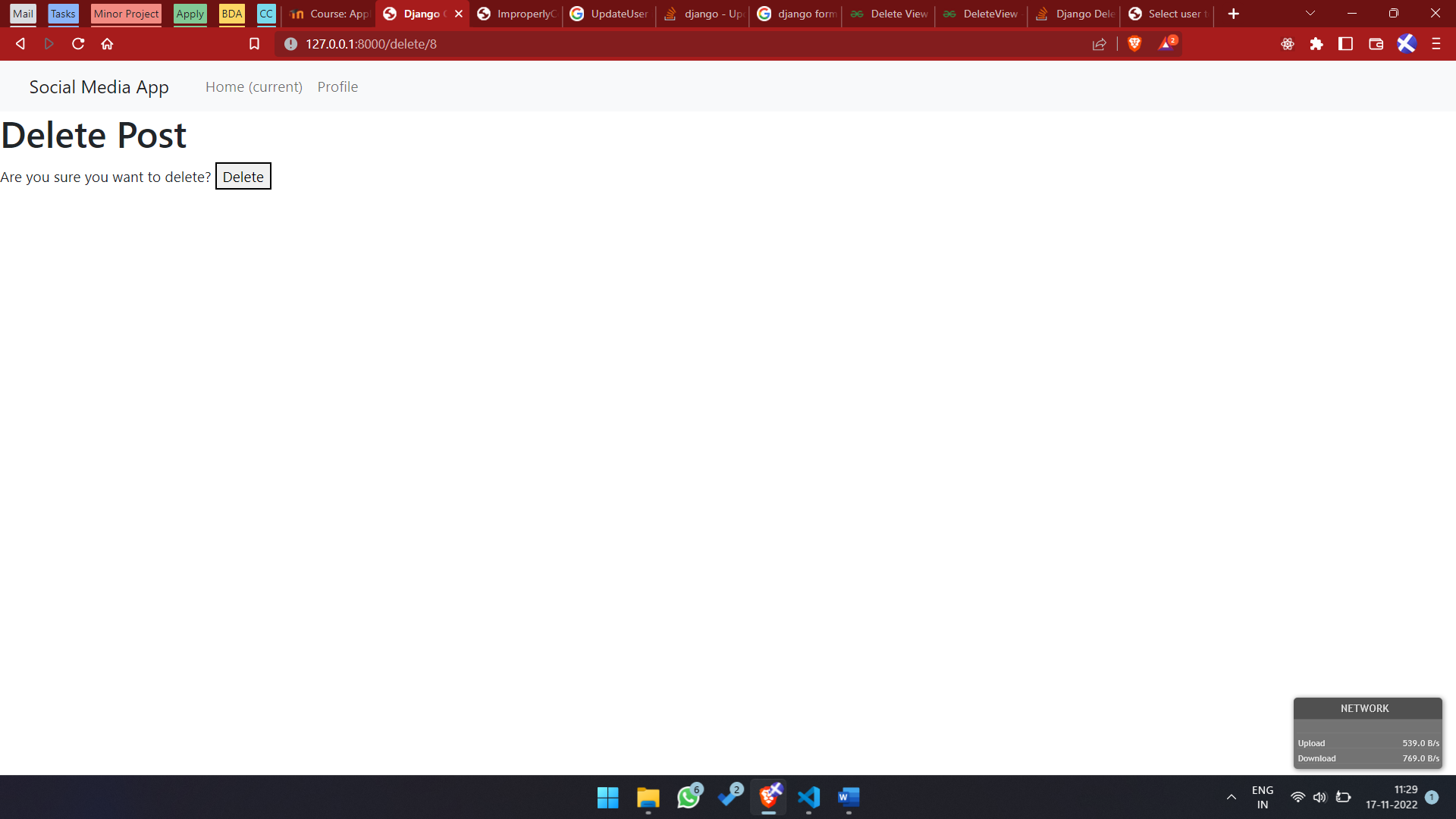
Output 2. Post Created



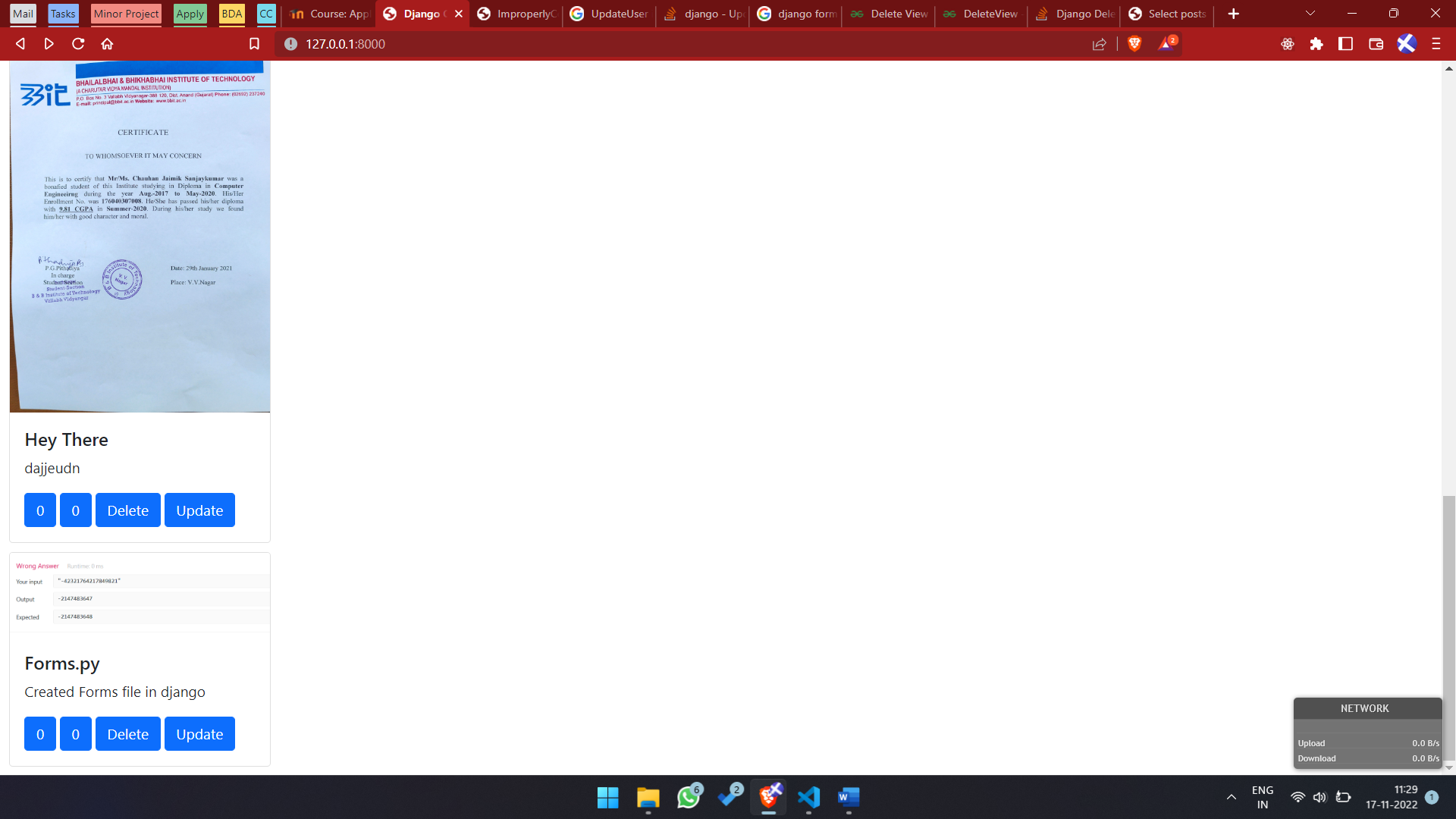
Output 2. Update Post Page



Output 2. Post Updated



Output 2. Delete Post page



Output 2. After Post Deletion, Home Page