MINI PROJECT
(2020-2021)

Instant Shop

MID-TERM REPORT

Institute of Engineering & Technology

<u>Submitted by-</u>

Tarun (181500756)

Naman Gupta (181500410)

Amber Srivastava (181500082)

Devesh Yadav (181500214)

Nitin Pratap Singh (181500436)

<u>Supervised by-</u>

Mr. Anand Gupta
Technical Trainer
Department of Computer Engineering & Applications

Contents

Abstract

In this mini project, we are building a website using Django, HTML5, CSS3, JavaScript and Bootstrap. After creating this website it will be hosted on pythonanywhere.com .

It will be a non-profit website, which can be used by everyone.

This website will solve the problem of those people who are trying to sell their old products and people who wants to buy a used product, as this website will save their time to find each other and also the deal can be done on a good price.

HTML5, CSS3, JavaScript and Bootstrap makes it possible to design a good UI for the users which is easy to use.

Django is used to implement the backend of the website. It delivers users the functionality to login, logout, register, email verification, searching, list products, book product and chat**.**

Introduction

The Instant shop is a web-based (website) project, which provides solution to the people who wants to sell or buy used products. This website is being developed for sellers who wants to sell their used products at good price and the buyer who wants a good second hand product in their budget. The traditional pattern for buying/selling used product can become a very complex task and time consuming and totally dependent on manual searching technique. After keep in mind all these problems. We make our site that can solve all those problems of such customers easily in an elegant & efficient manner.

Hardware Requirements

- Memory 4GB Ram or higher
- A decent processor

Technology Requirements

- Python 3.6
- Django 3.1
- HTML5
- CSS3
- JavaScript
- Bootstrap
- Font Awesome Icons

Objectives

- Provides an interactive interface with the help of Html, CSS, JavaScript and Bootstrap behaves in a sufficient manner that is easy to use and understand.
- All the information regarding products and users applied online.
- Provide an advanced feature for product management.
- Provide a secured environment for accessing database wherever necessary.
- Provide a module to update the product information.
- To store details of the buyers and sellers.

Implementation Details

- **Part 1: To build Home page.**

With the use of HTML5, CSS3, JavaScript, Bootstrap and Font awesome icons we will design the home page which will be having a carousel and navigation bar.

- **Part 2: To design Login and Signup form in Django.**

With the help of Django, HTML5, CSS3, JavaScript and Bootstrap we will design the login and signup form. These forms will also be having the server side validation. After validation the data will be used for further use.

- **Part 3: To implement Login, Signup and Logout.**

This will be implemented with Django only. After validating the data from Signup form we will be saving the data in database and after doing so we will register the user.

After validating data from Login form the user will be logged in to the server.

A logout button will be provided, so whenever user wants to logout, it can be done by pressing the button.

- **Part 4: To implement the e-mail verification of user.**

At the time of signup an e-mail with a link is sent to the user’s email address. That link is is used to verify the user’s mail address. When user will click on that link the user’s mail address will be verified**.**

- **Part 5: To build about us page.**

With the use of HTML5, CSS3, JavaScript, Bootstrap and Font awesome icons we will design the about-us page which will be having details about each team member and mentor with their social links.

- **Part 6: To build a shopping page.**

With the use of Django, HTML5, CSS3, JavaScript, Bootstrap and Font awesome icons we will design the shopping page which will be having search bar to search products, and list of searched products.

- **Part 7: To build profile page.**

With the use of Django, HTML5, CSS3, JavaScript, Bootstrap and Font awesome icons we will design the profile page which will be having all the details of logged in user, editing the profile detail and list of all the products purchased/uploaded by the user.

- **Part 8: To implement chat feature.**

With the help of Django, Django Channels we will be implementing the chat feature between two users**.**

- **Part 9: To implement comment feature.**

With the help of Django we will implement the comment on products feature.

- **Part 10: To implement cart and checkout feature.**

With the use of Django, HTML5, CSS3, JavaScript, Bootstrap and Font awesome icons we will design the cart and checkout feature (checkout feature will not be having any payment partner) which will be having all the details of logged in user, editing the profile detail and list of all the products purchased/uploaded by the user.

Progress

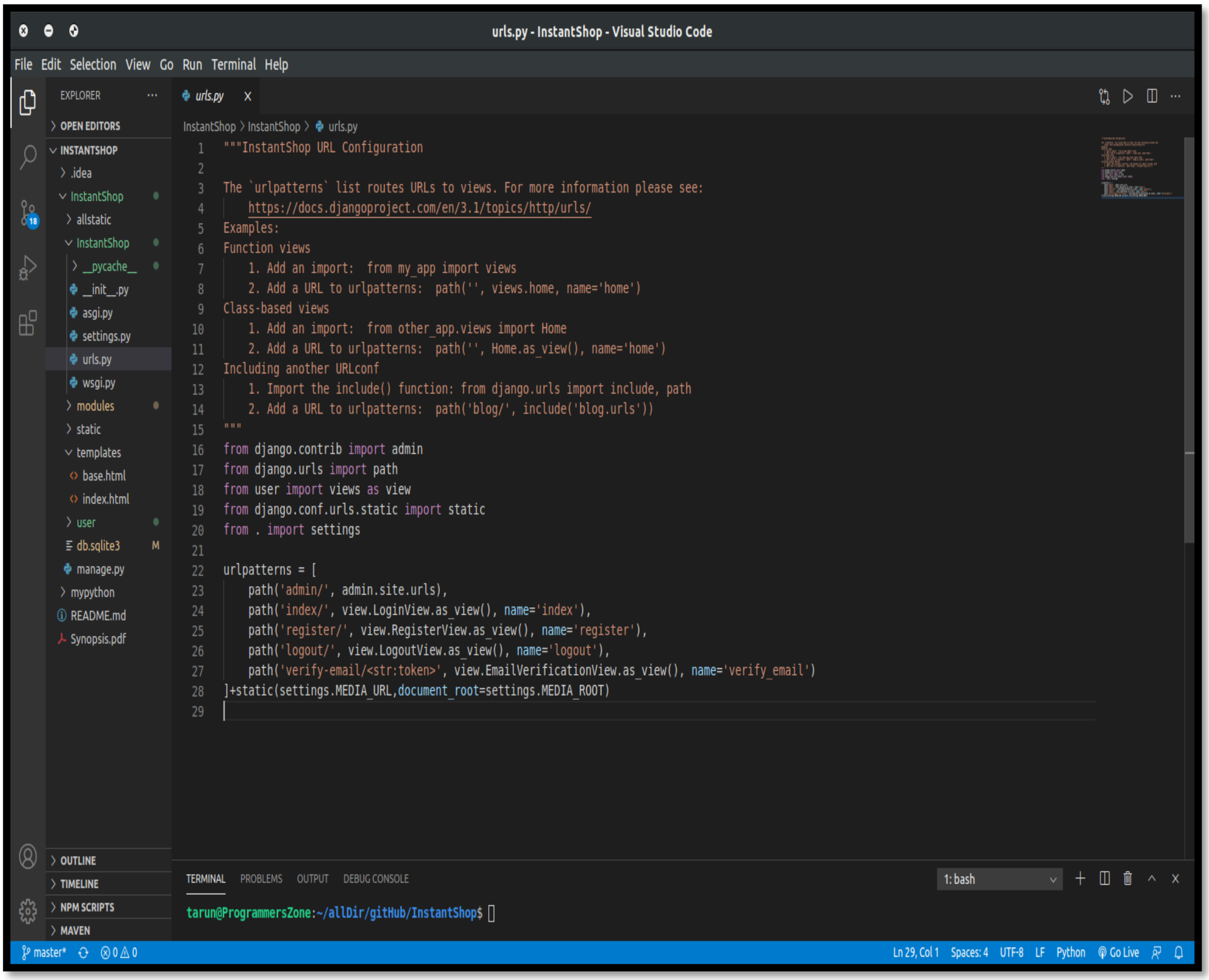
Part 1 is completed.

Part 2 is completed.

Part 3 is completed.

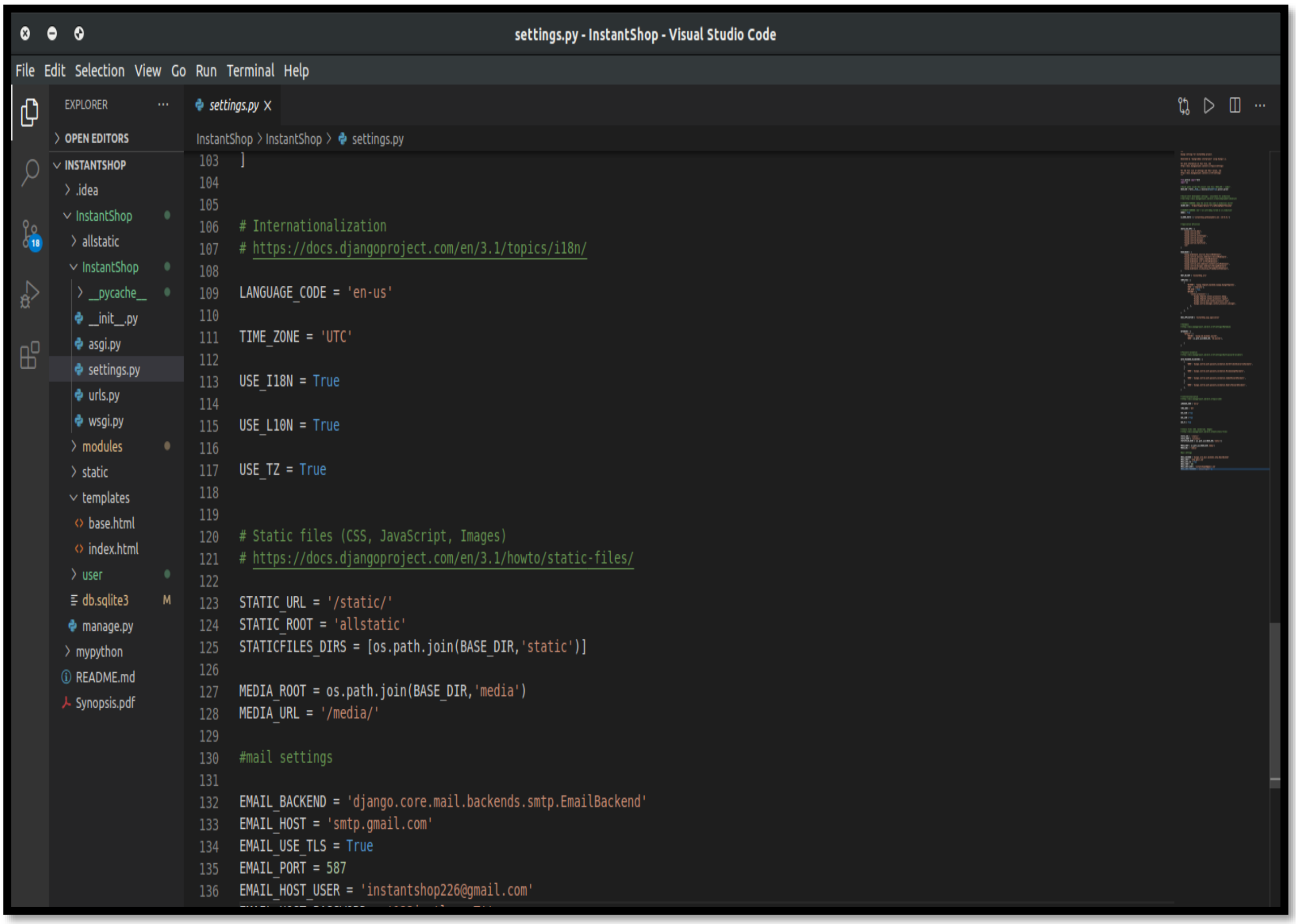
Part 4 is completed.

SCREENSHOTS

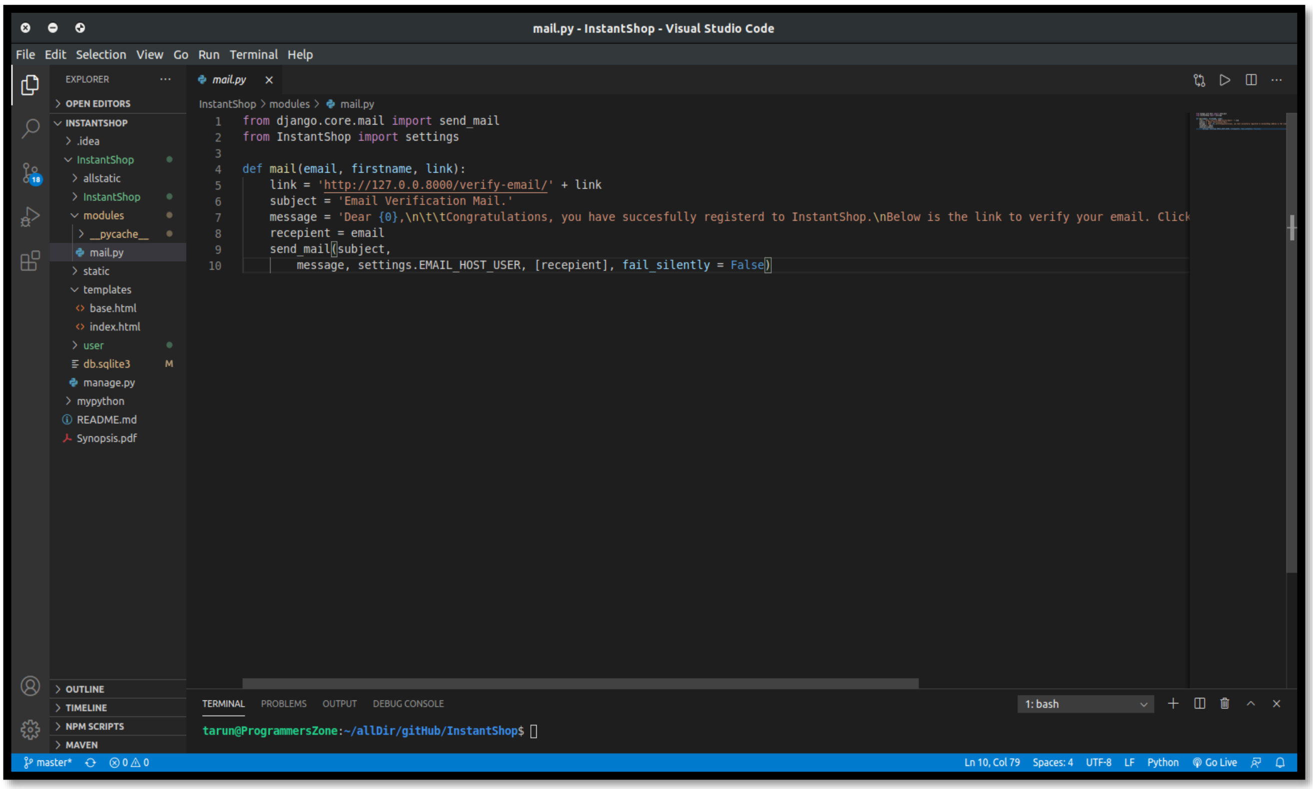


```python
"""InstantShop URL Configuration

The `urlpatterns` list routes URLs to views. For more information please see:
    https://docs.djangoproject.com/en/3.1/topics/http/urls/
Examples:
Function views
    1. Add an import:  from my_app import views
    2. Add a URL to urlpatterns:  path('', views.home, name='home')
Class-based views
    1. Add an import:  from other_app.views import Home
    2. Add a URL to urlpatterns:  path('', Home.as_view(), name='home')
Including another URLconf
    1. Import the include() function: from django.urls import include, path
    2. Add a URL to urlpatterns:  path('blog/', include('blog.urls'))
"""
from django.contrib import admin
from django.urls import path
from user import views as view
from django.conf.urls.static import static
from . import settings

urlpatterns = [
    path('admin/', admin.site.urls),
    path('index/', view.LoginView.as_view(), name='index'),
    path('register/', view.RegisterView.as_view(), name='register'),
    path('logout/', view.LogoutView.as_view(), name='logout'),
    path('verify-email/<str:token>', view.EmailVerificationView.as_view(), name='verify_email')
]+static(settings.MEDIA_URL,document_root=settings.MEDIA_ROOT)
```

```
tarun@ProgrammersZone:~/allDir/gitHub/InstantShop$
```

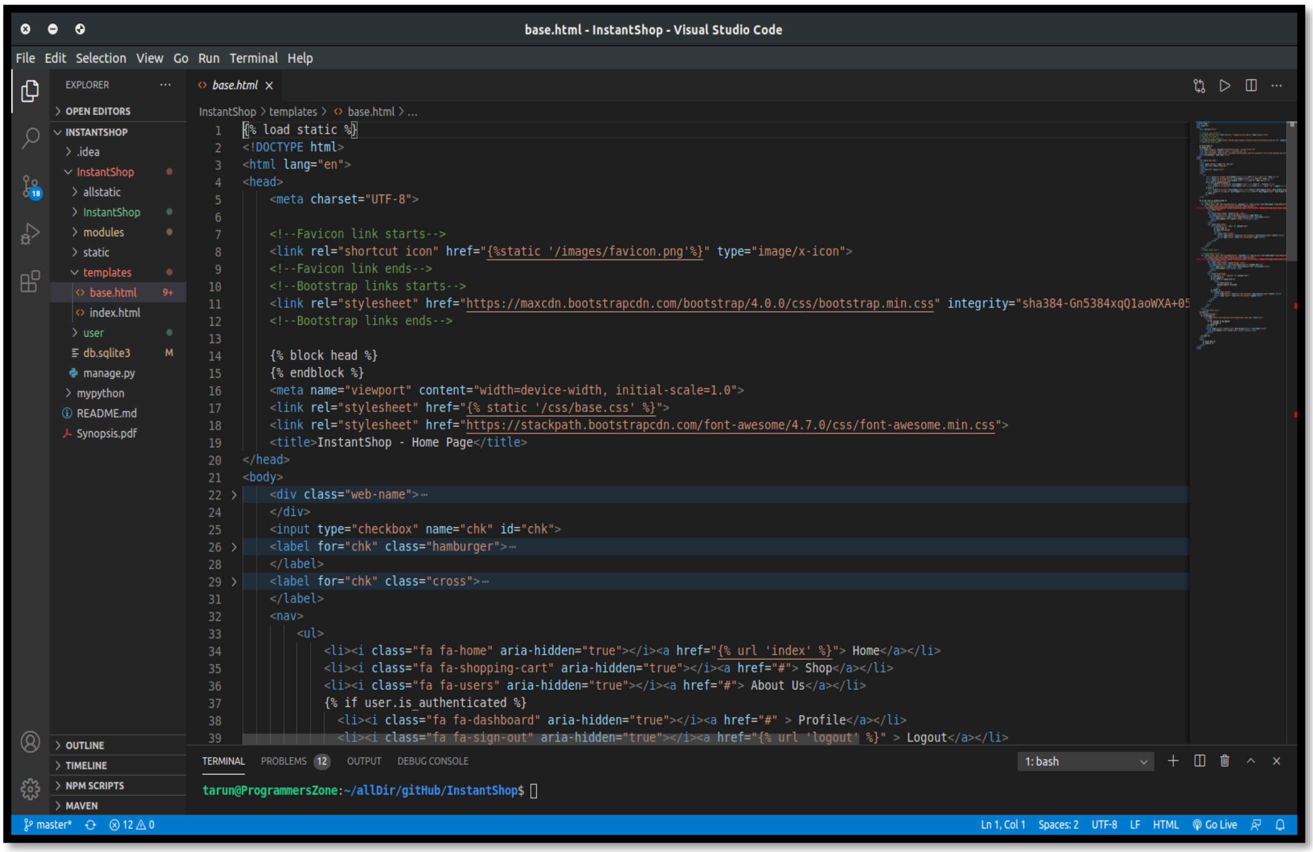


```
]


# Internationalization
# https://docs.djangoproject.com/en/3.1/topics/i18n/

LANGUAGE_CODE = 'en-us'

TIME_ZONE = 'UTC'

USE_I18N = True

USE_L10N = True

USE_TZ = True


# Static files (CSS, JavaScript, Images)
# https://docs.djangoproject.com/en/3.1/howto/static-files/

STATIC_URL = '/static/'
STATIC_ROOT = 'allstatic'
STATICFILES_DIRS = [os.path.join(BASE_DIR,'static')]

MEDIA_ROOT = os.path.join(BASE_DIR,'media')
MEDIA_URL = '/media/'

#mail settings

EMAIL_BACKEND = 'django.core.mail.backends.smtp.EmailBackend'
EMAIL_HOST = 'smtp.gmail.com'
EMAIL_USE_TLS = True
EMAIL_PORT = 587
EMAIL_HOST_USER = 'instantshop226@gmail.com'
```



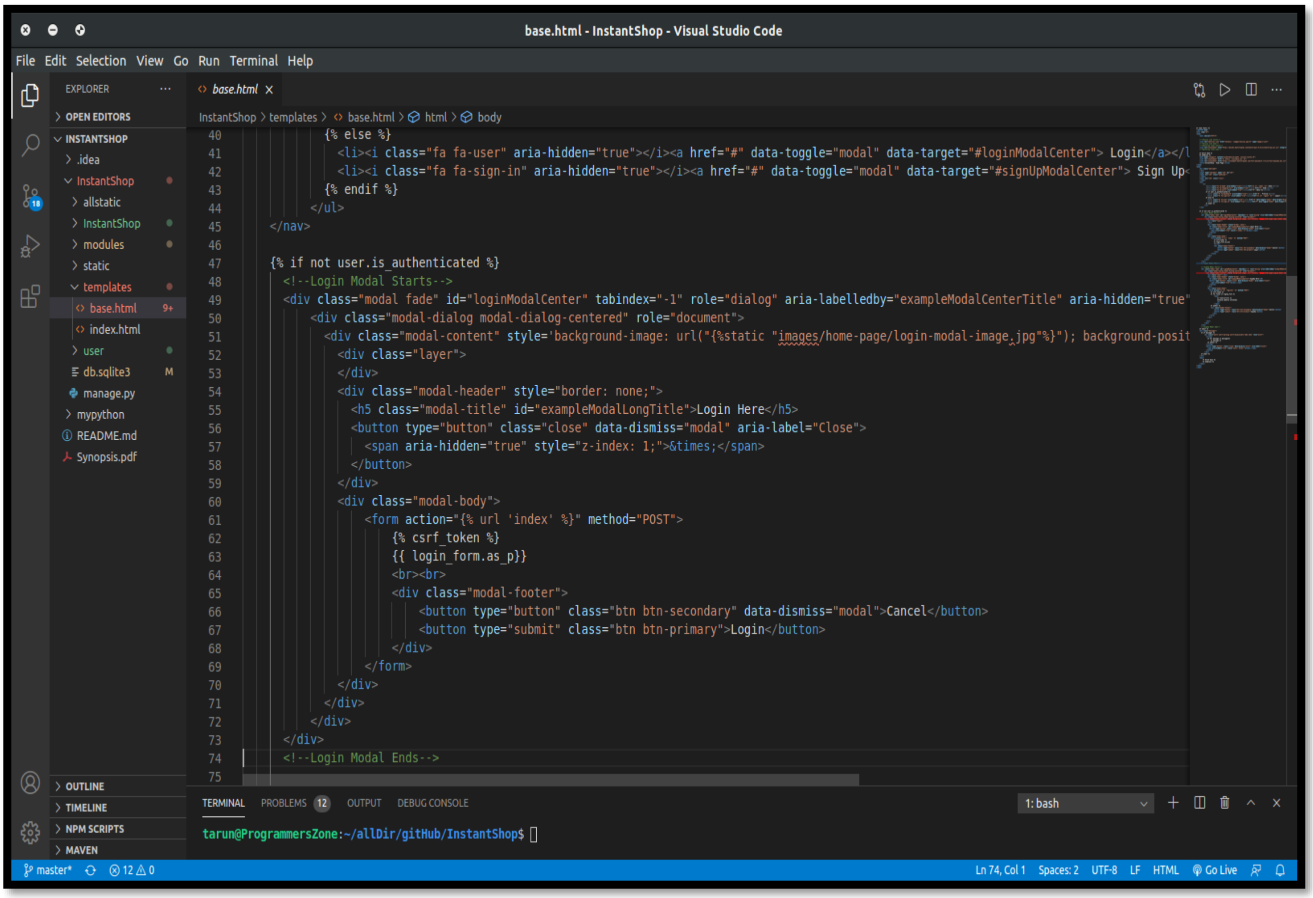
```python
from django.core.mail import send_mail
from InstantShop import settings

def mail(email, firstname, link):
    link = 'http://127.0.0.8000/verify-email/' + link
    subject = 'Email Verification Mail.'
    message = 'Dear {0},\n\t\tCongratulations, you have succesfully registerd to InstantShop.\nBelow is the link to verify your email. Click
    recepient = email
    send_mail(subject,
        message, settings.EMAIL_HOST_USER, [recepient], fail_silently = False)
```

```
tarun@ProgrammersZone:~/allDir/gitHub/InstantShop$ 
```


base.html - InstantShop - Visual Studio Code
File Edit Selection View Go Run Terminal Help
EXPLORER
OPEN EDITORS
INSTANTSHOP
.idea
InstantShop
allstatic
InstantShop
modules
static
templates
base.html
index.html
user
db.sqlite3
manage.py
mypython
README.md
Synopsis.pdf
OUTLINE
TIMELINE
NPM SCRIPTS
MAVEN
InstantShop > templates > base.html > ...
TERMINAL PROBLEMS 12 OUTPUT DEBUG CONSOLE
1: bash
master* ⊗ 12 ⚠ 0
Ln 1, Col 1 Spaces: 2 UTF-8 LF HTML Go Live

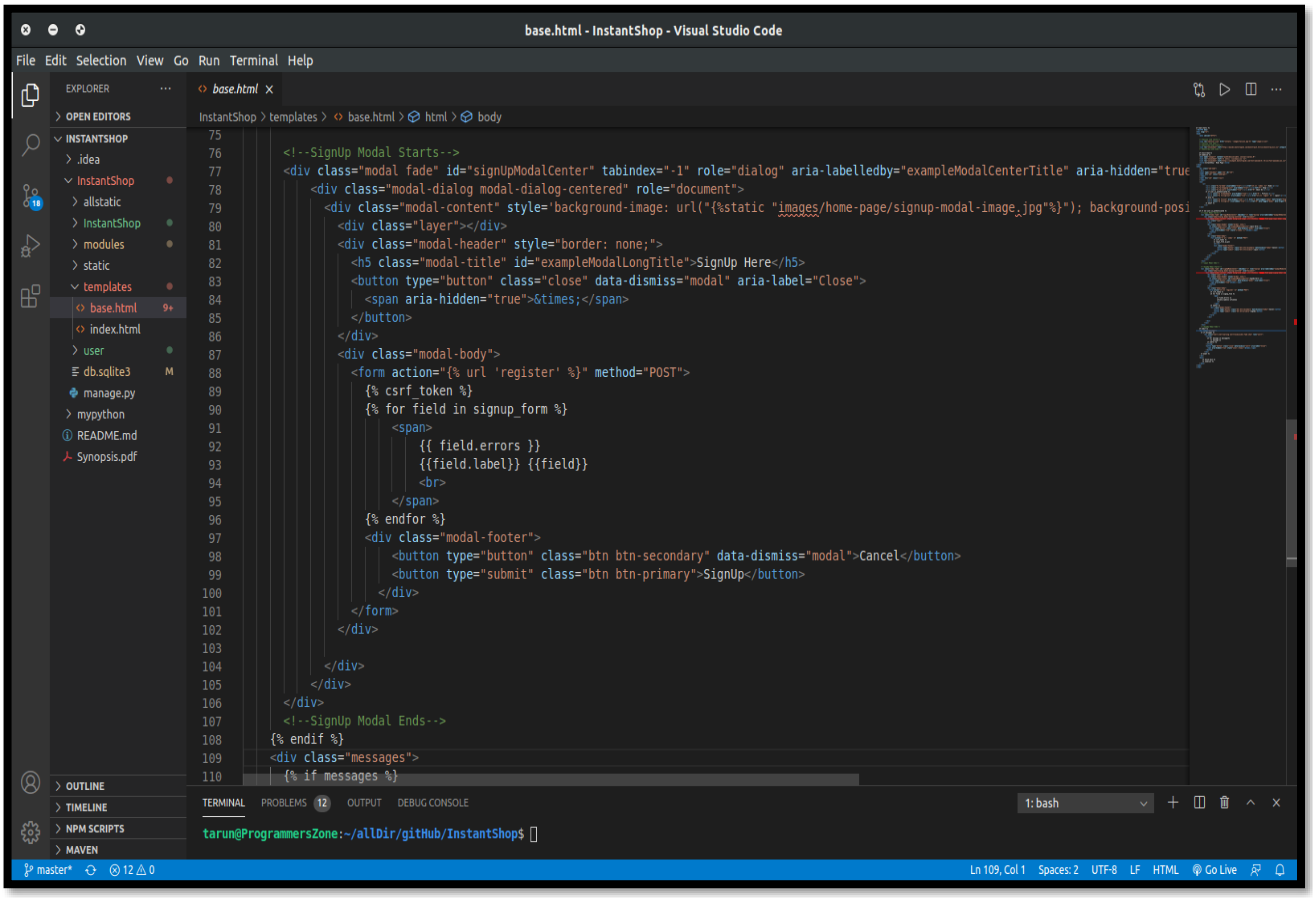
```html
{% load static %}
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">

    <!--Favicon link starts-->
    <link rel="shortcut icon" href="{%static '/images/favicon.png'%}" type="image/x-icon">
    <!--Favicon link ends-->
    <!--Bootstrap links starts-->
    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css" integrity="sha384-Gn5384xqQ1aoWXA+05
    <!--Bootstrap links ends-->

    {% block head %}
    {% endblock %}
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="stylesheet" href="{% static '/css/base.css' %}">
    <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/font-awesome/4.7.0/css/font-awesome.min.css">
    <title>InstantShop - Home Page</title>
</head>
<body>
    <div class="web-name">...
    </div>
    <input type="checkbox" name="chk" id="chk">
    <label for="chk" class="hamburger">...
    </label>
    <label for="chk" class="cross">...
    </label>
    <nav>
        <ul>
            <li><i class="fa fa-home" aria-hidden="true"></i><a href="{% url 'index' %}"> Home</a></li>
            <li><i class="fa fa-shopping-cart" aria-hidden="true"></i><a href="#"> Shop</a></li>
            <li><i class="fa fa-users" aria-hidden="true"></i><a href="#"> About Us</a></li>
            {% if user.is_authenticated %}
              <li><i class="fa fa-dashboard" aria-hidden="true"></i><a href="#" > Profile</a></li>
              <li><i class="fa fa-sign-out" aria-hidden="true"></i><a href="{% url 'logout' %}" > Logout</a></li>
```

```
tarun@ProgrammersZone:~/allDir/gitHub/InstantShop$ 
```



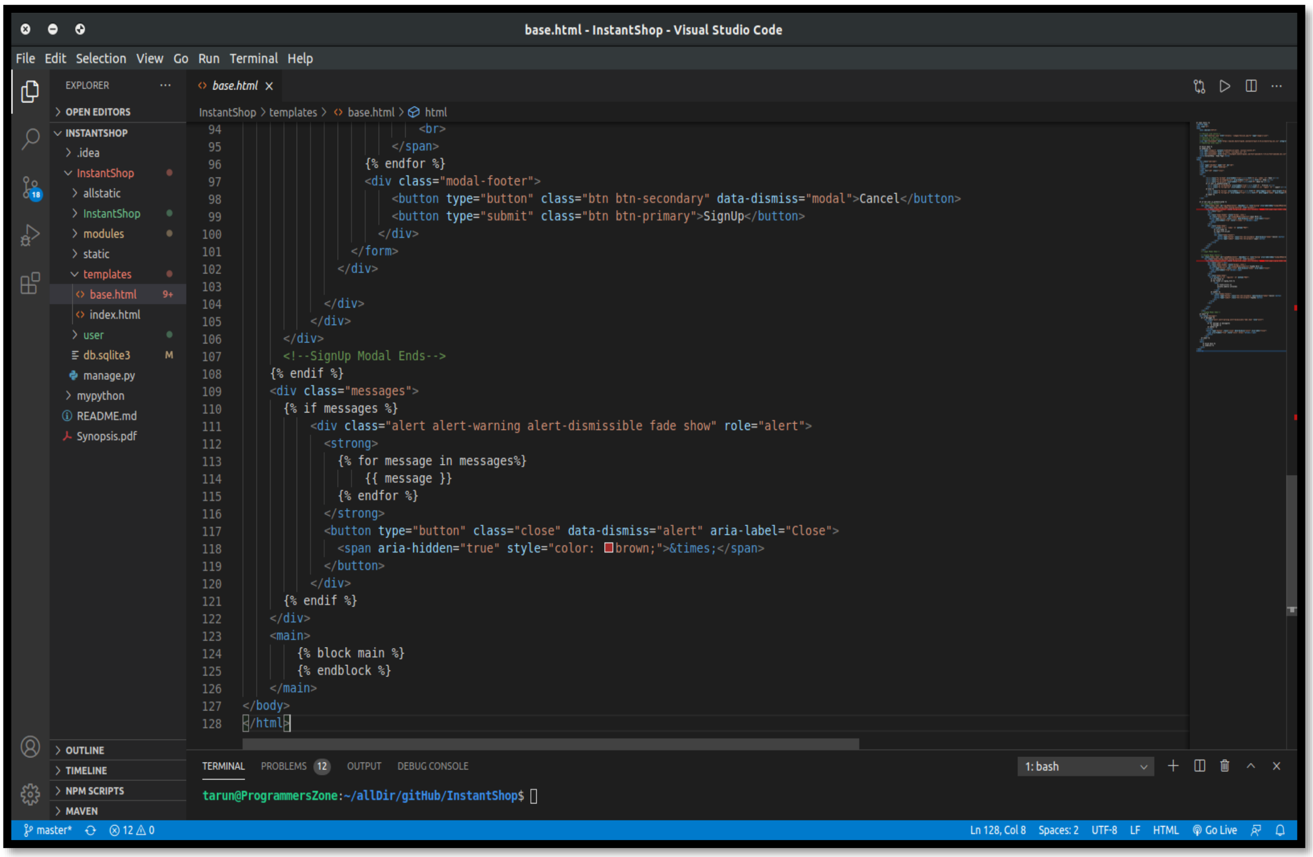
```html
        {% else %}
          <li><i class="fa fa-user" aria-hidden="true"></i><a href="#" data-toggle="modal" data-target="#loginModalCenter"> Login</a></l
          <li><i class="fa fa-sign-in" aria-hidden="true"></i><a href="#" data-toggle="modal" data-target="#signUpModalCenter"> Sign Up<
        {% endif %}
      </ul>
</nav>

{% if not user.is_authenticated %}
  <!--Login Modal Starts-->
  <div class="modal fade" id="loginModalCenter" tabindex="-1" role="dialog" aria-labelledby="exampleModalCenterTitle" aria-hidden="true"
      <div class="modal-dialog modal-dialog-centered" role="document">
        <div class="modal-content" style='background-image: url("{%static "images/home-page/login-modal-image.jpg"%}"); background-posit
          <div class="layer">
          </div>
          <div class="modal-header" style="border: none;">
            <h5 class="modal-title" id="exampleModalLongTitle">Login Here</h5>
            <button type="button" class="close" data-dismiss="modal" aria-label="Close">
              <span aria-hidden="true" style="z-index: 1;">&times;</span>
            </button>
          </div>
          <div class="modal-body">
              <form action="{% url 'index' %}" method="POST">
                  {% csrf_token %}
                  {{ login_form.as_p}}
                  <br><br>
                  <div class="modal-footer">
                      <button type="button" class="btn btn-secondary" data-dismiss="modal">Cancel</button>
                      <button type="submit" class="btn btn-primary">Login</button>
                  </div>
              </form>
          </div>
        </div>
      </div>
  </div>
  <!--Login Modal Ends-->
```

```
tarun@ProgrammersZone:~/allDir/gitHub/InstantShop$
```



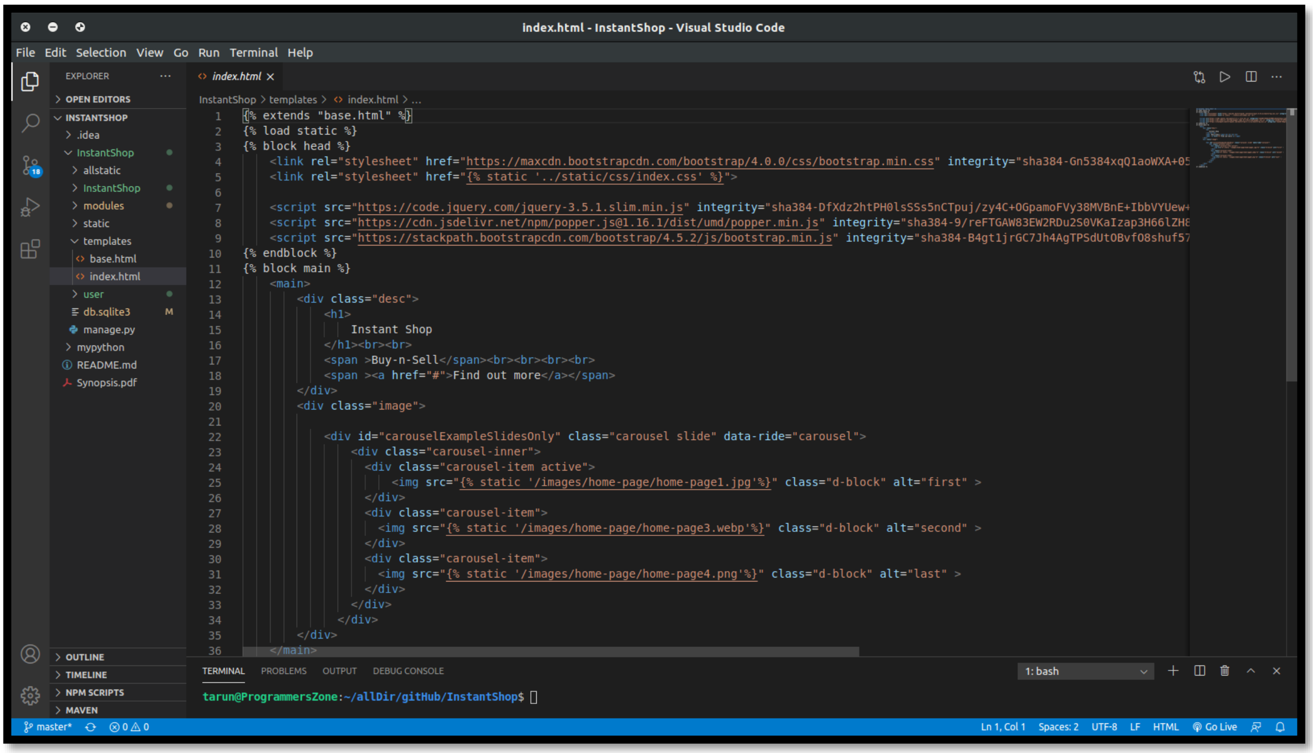
```
  <!--SignUp Modal Starts-->
  <div class="modal fade" id="signUpModalCenter" tabindex="-1" role="dialog" aria-labelledby="exampleModalCenterTitle" aria-hidden="true
      <div class="modal-dialog modal-dialog-centered" role="document">
        <div class="modal-content" style='background-image: url("{%static "images/home-page/signup-modal-image.jpg"%}"); background-posi
          <div class="layer"></div>
          <div class="modal-header" style="border: none;">
            <h5 class="modal-title" id="exampleModalLongTitle">SignUp Here</h5>
            <button type="button" class="close" data-dismiss="modal" aria-label="Close">
              <span aria-hidden="true">&times;</span>
            </button>
          </div>
          <div class="modal-body">
            <form action="{% url 'register' %}" method="POST">
              {% csrf_token %}
              {% for field in signup_form %}
                  <span>
                      {{ field.errors }}
                      {{field.label}} {{field}}
                      <br>
                  </span>
              {% endfor %}
              <div class="modal-footer">
                  <button type="button" class="btn btn-secondary" data-dismiss="modal">Cancel</button>
                  <button type="submit" class="btn btn-primary">SignUp</button>
                </div>
            </form>
          </div>

        </div>
      </div>
  </div>
  <!--SignUp Modal Ends-->
{% endif %}
<div class="messages">
  {% if messages %}
```

```
tarun@ProgrammersZone:~/allDir/gitHub/InstantShop$ 
```



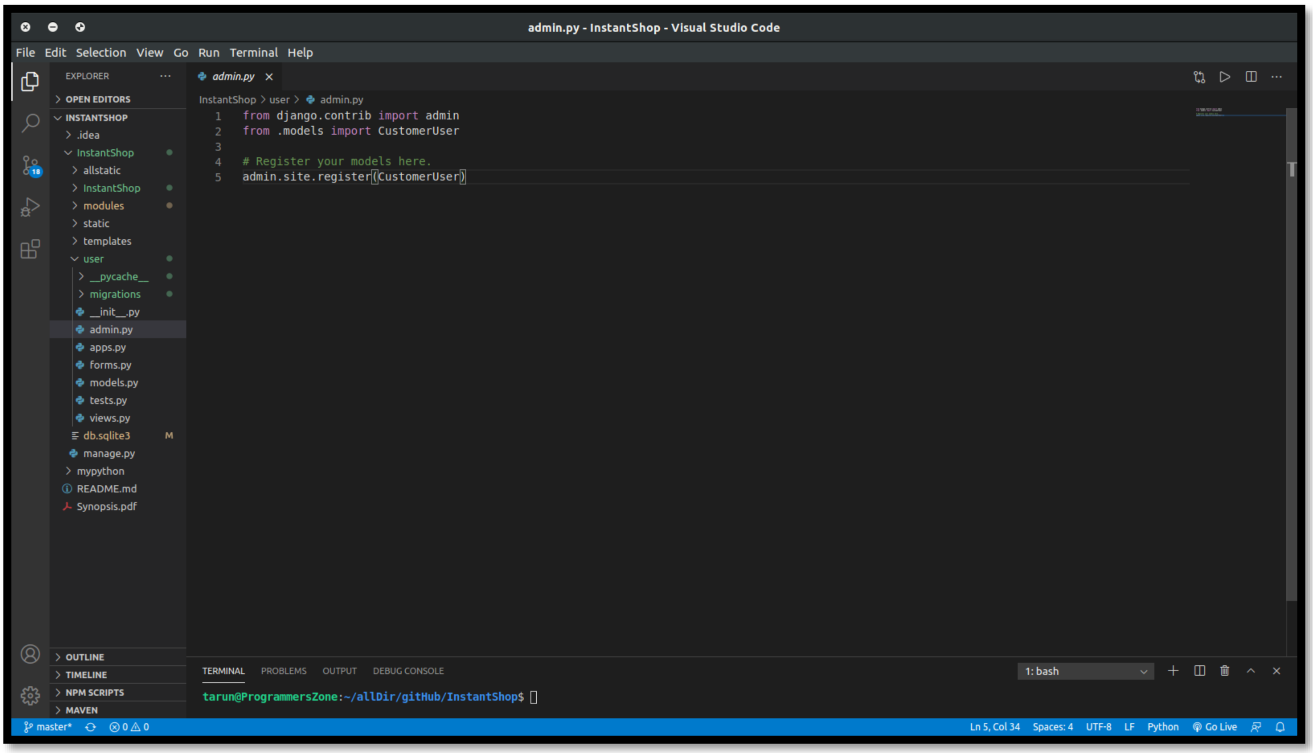
```
                          <br>
                      </span>
                  {% endfor %}
                  <div class="modal-footer">
                      <button type="button" class="btn btn-secondary" data-dismiss="modal">Cancel</button>
                      <button type="submit" class="btn btn-primary">SignUp</button>
                    </div>
                </form>
              </div>

            </div>
          </div>
      </div>
      <!--SignUp Modal Ends-->
    {% endif %}
    <div class="messages">
      {% if messages %}
          <div class="alert alert-warning alert-dismissible fade show" role="alert">
            <strong>
              {% for message in messages%}
                  {{ message }}
              {% endfor %}
            </strong>
            <button type="button" class="close" data-dismiss="alert" aria-label="Close">
              <span aria-hidden="true" style="color: ■brown;">&times;</span>
            </button>
          </div>
      {% endif %}
    </div>
    <main>
        {% block main %}
        {% endblock %}
    </main>
</body>
</html>
```

```
tarun@ProgrammersZone:~/allDir/gitHub/InstantShop$ 
```



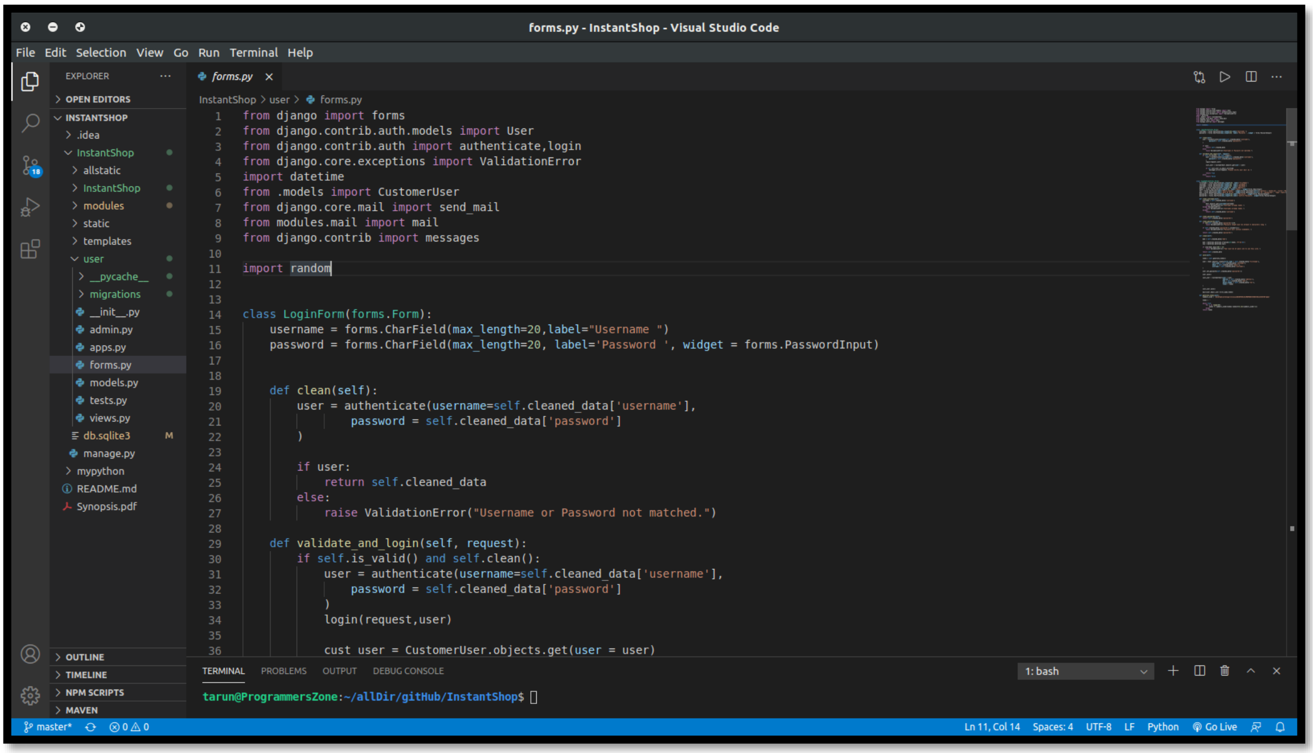
```
{% extends "base.html" %}
{% load static %}
{% block head %}
    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css" integrity="sha384-Gn5384xqQ1aoWXA+05
    <link rel="stylesheet" href="{% static '../static/css/index.css' %}">

    <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js" integrity="sha384-DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+
    <script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js" integrity="sha384-9/reFTGAW83EW2RDu2S0VKaIzap3H66lZH8
    <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js" integrity="sha384-B4gt1jrGC7Jh4AgTPSdUtOBvfO8shuf57
{% endblock %}
{% block main %}
    <main>
        <div class="desc">
            <h1>
                Instant Shop
            </h1><br><br>
            <span >Buy-n-Sell</span><br><br><br><br>
            <span ><a href="#">Find out more</a></span>
        </div>
        <div class="image">

            <div id="carouselExampleSlidesOnly" class="carousel slide" data-ride="carousel">
                <div class="carousel-inner">
                  <div class="carousel-item active">
                      <img src="{% static '/images/home-page/home-page1.jpg'%}" class="d-block" alt="first" >
                  </div>
                  <div class="carousel-item">
                    <img src="{% static '/images/home-page/home-page3.webp'%}" class="d-block" alt="second" >
                  </div>
                  <div class="carousel-item">
                    <img src="{% static '/images/home-page/home-page4.png'%}" class="d-block" alt="last" >
                  </div>
                </div>
              </div>
        </div>
    </main>
```

```
tarun@ProgrammersZone:~/allDir/gitHub/InstantShop$ 
```



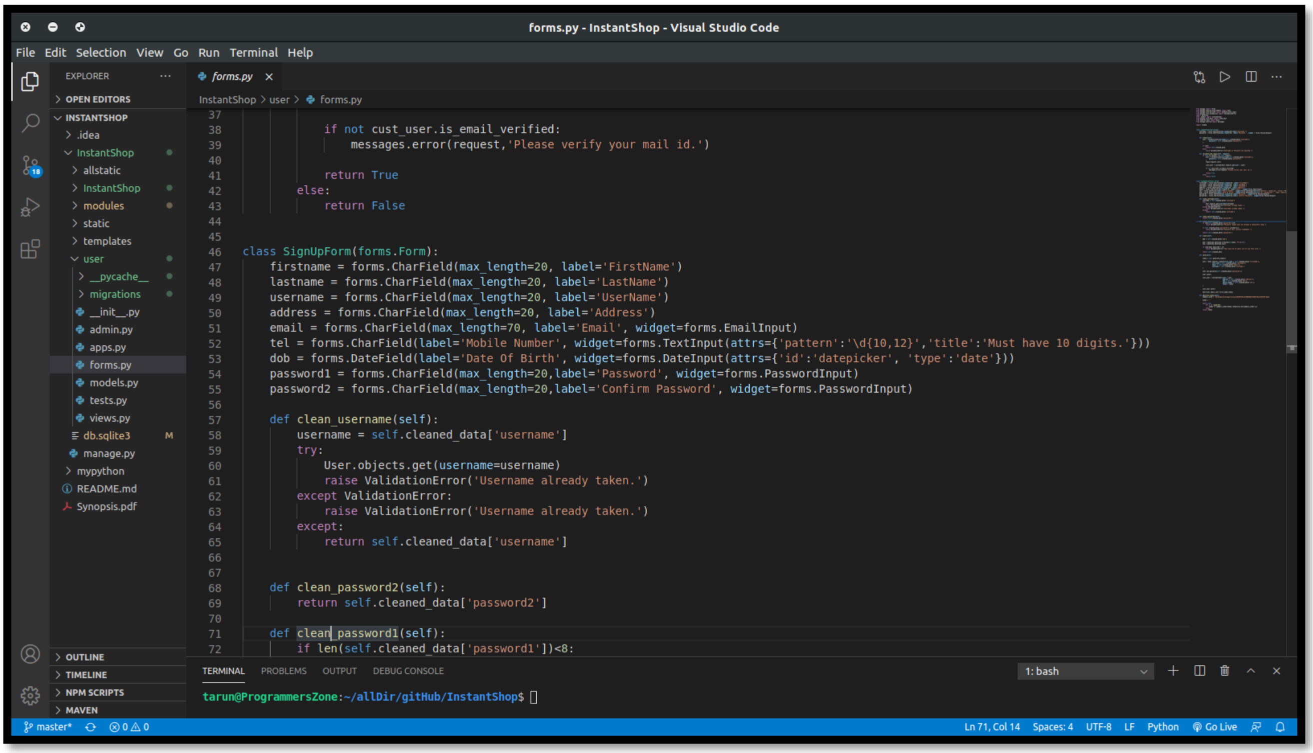
```python
from django.contrib import admin
from .models import CustomerUser

# Register your models here.
admin.site.register(CustomerUser)
```

```
tarun@ProgrammersZone:~/allDir/gitHub/InstantShop$ 
```



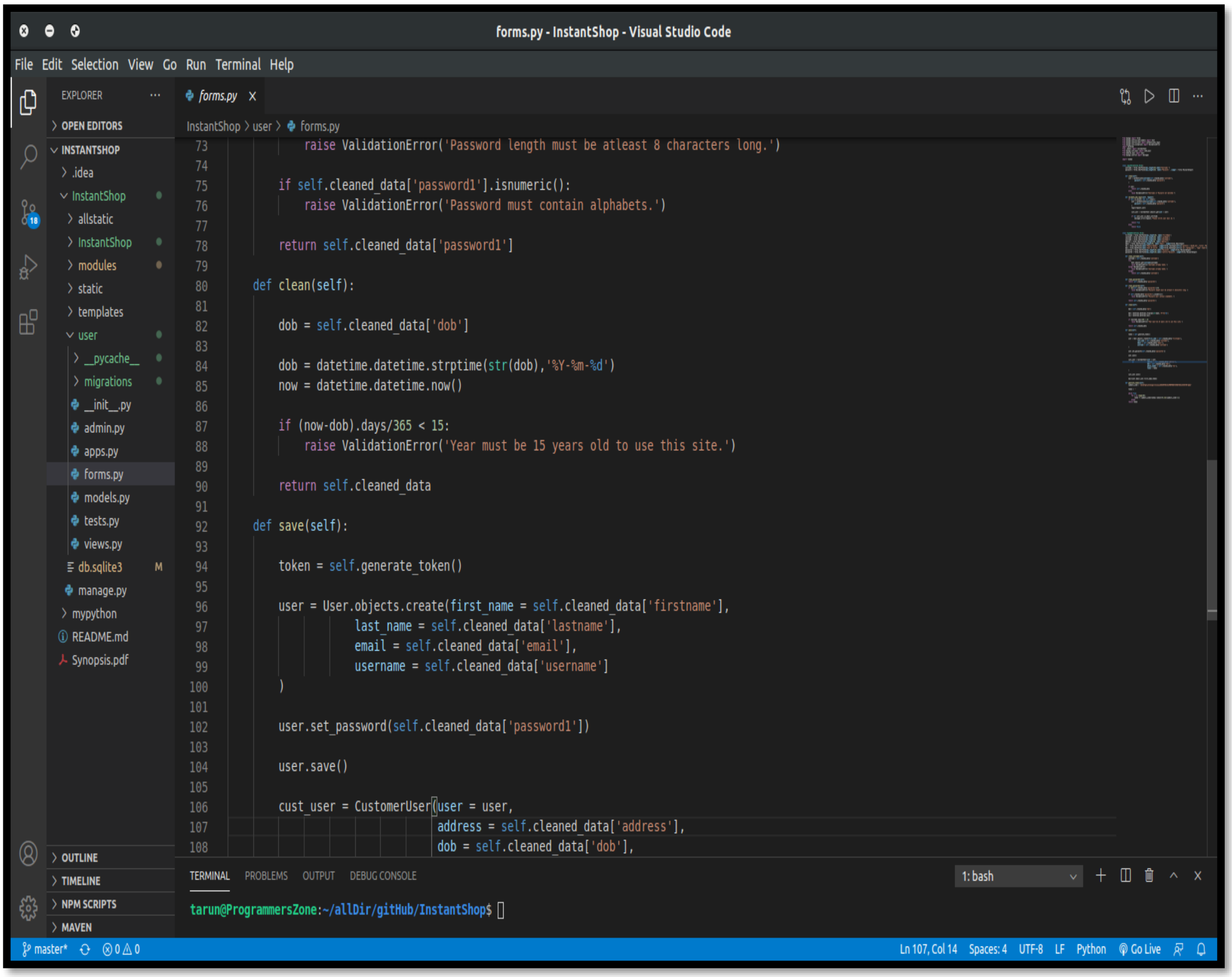
```python
from django import forms
from django.contrib.auth.models import User
from django.contrib.auth import authenticate,login
from django.core.exceptions import ValidationError
import datetime
from .models import CustomerUser
from django.core.mail import send_mail
from modules.mail import mail
from django.contrib import messages

import random


class LoginForm(forms.Form):
    username = forms.CharField(max_length=20,label="Username ")
    password = forms.CharField(max_length=20, label='Password ', widget = forms.PasswordInput)


    def clean(self):
        user = authenticate(username=self.cleaned_data['username'],
                password = self.cleaned_data['password']
        )

        if user:
            return self.cleaned_data
        else:
            raise ValidationError("Username or Password not matched.")

    def validate_and_login(self, request):
        if self.is_valid() and self.clean():
            user = authenticate(username=self.cleaned_data['username'],
                password = self.cleaned_data['password']
            )
            login(request,user)

            cust_user = CustomerUser.objects.get(user = user)
```

```
tarun@ProgrammersZone:~/allDir/gitHub/InstantShop$ 
```



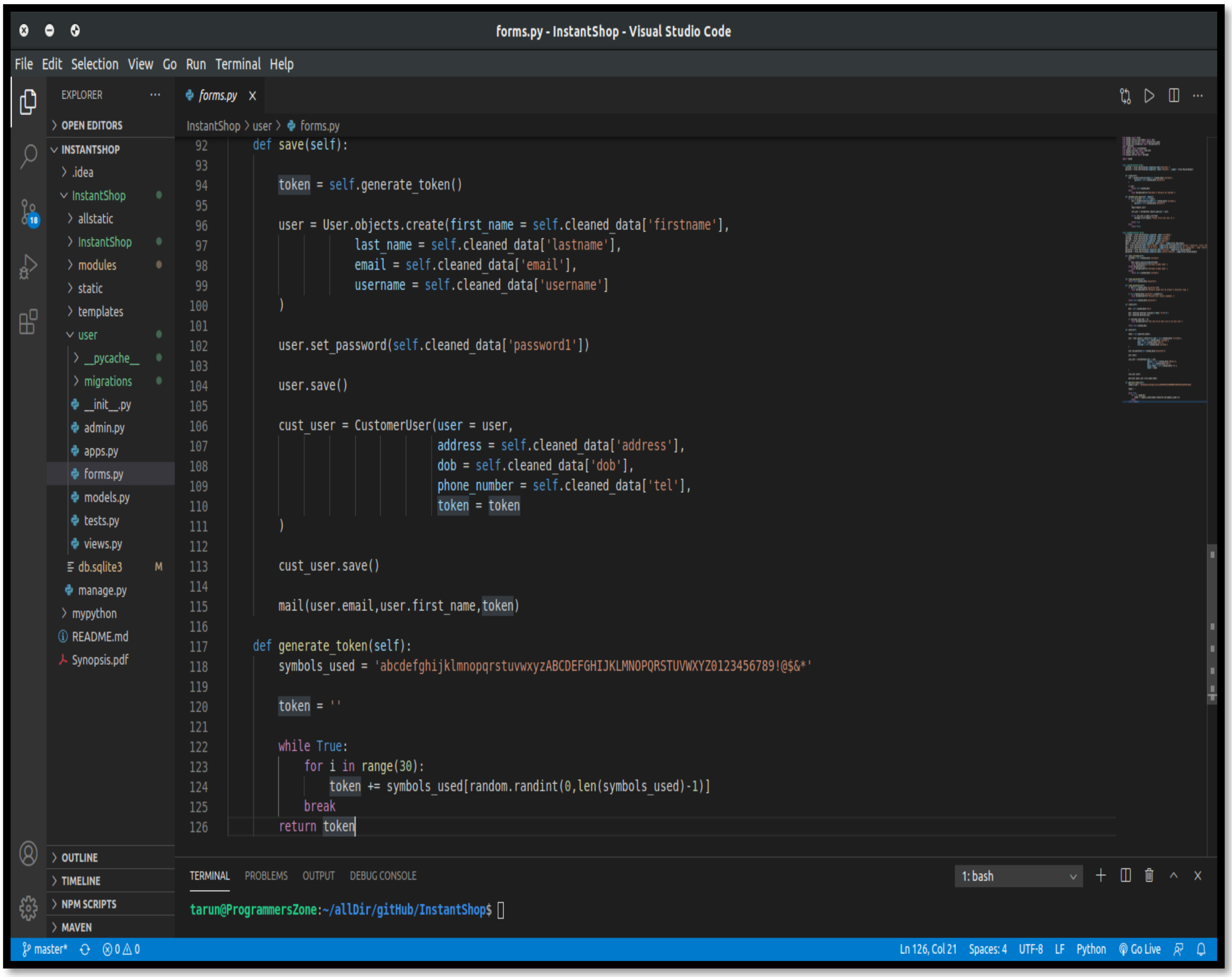
```python

            if not cust_user.is_email_verified:
                messages.error(request,'Please verify your mail id.')

            return True
        else:
            return False


class SignUpForm(forms.Form):
    firstname = forms.CharField(max_length=20, label='FirstName')
    lastname = forms.CharField(max_length=20, label='LastName')
    username = forms.CharField(max_length=20, label='UserName')
    address = forms.CharField(max_length=20, label='Address')
    email = forms.CharField(max_length=70, label='Email', widget=forms.EmailInput)
    tel = forms.CharField(label='Mobile Number', widget=forms.TextInput(attrs={'pattern':'\d{10,12}','title':'Must have 10 digits.'}))
    dob = forms.DateField(label='Date Of Birth', widget=forms.DateInput(attrs={'id':'datepicker', 'type':'date'}))
    password1 = forms.CharField(max_length=20,label='Password', widget=forms.PasswordInput)
    password2 = forms.CharField(max_length=20,label='Confirm Password', widget=forms.PasswordInput)

    def clean_username(self):
        username = self.cleaned_data['username']
        try:
            User.objects.get(username=username)
            raise ValidationError('Username already taken.')
        except ValidationError:
            raise ValidationError('Username already taken.')
        except:
            return self.cleaned_data['username']


    def clean_password2(self):
        return self.cleaned_data['password2']

    def clean_password1(self):
        if len(self.cleaned_data['password1'])<8:
```

```
tarun@ProgrammersZone:~/allDir/gitHub/InstantShop$ 
```



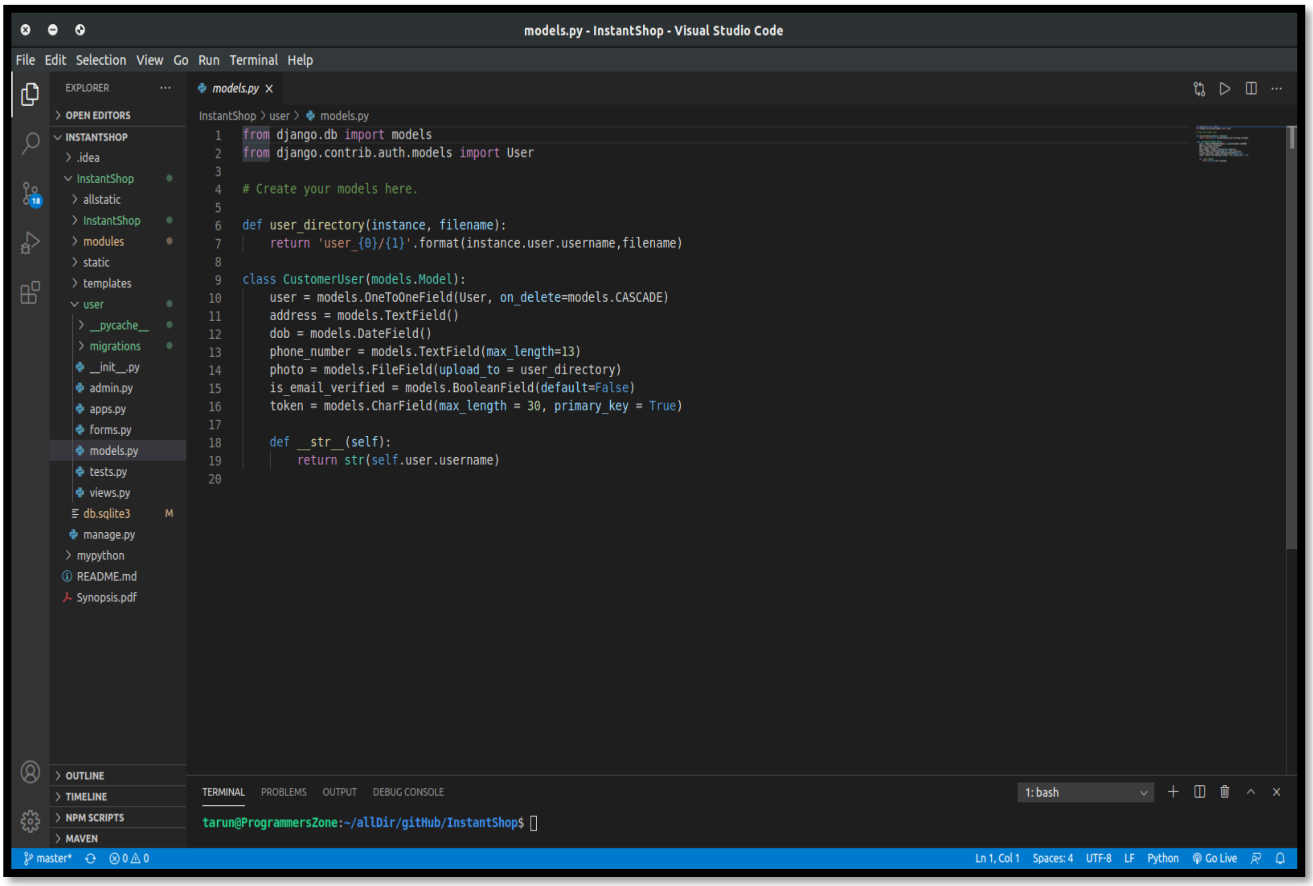
```python
            raise ValidationError('Password length must be atleast 8 characters long.')

        if self.cleaned_data['password1'].isnumeric():
            raise ValidationError('Password must contain alphabets.')

        return self.cleaned_data['password1']

    def clean(self):

        dob = self.cleaned_data['dob']

        dob = datetime.datetime.strptime(str(dob),'%Y-%m-%d')
        now = datetime.datetime.now()

        if (now-dob).days/365 < 15:
            raise ValidationError('Year must be 15 years old to use this site.')

        return self.cleaned_data

    def save(self):

        token = self.generate_token()

        user = User.objects.create(first_name = self.cleaned_data['firstname'],
                    last_name = self.cleaned_data['lastname'],
                    email = self.cleaned_data['email'],
                    username = self.cleaned_data['username']
        )

        user.set_password(self.cleaned_data['password1'])

        user.save()

        cust_user = CustomerUser(user = user,
                                 address = self.cleaned_data['address'],
                                 dob = self.cleaned_data['dob'],
```

```
tarun@ProgrammersZone:~/allDir/gitHub/InstantShop$ 
```



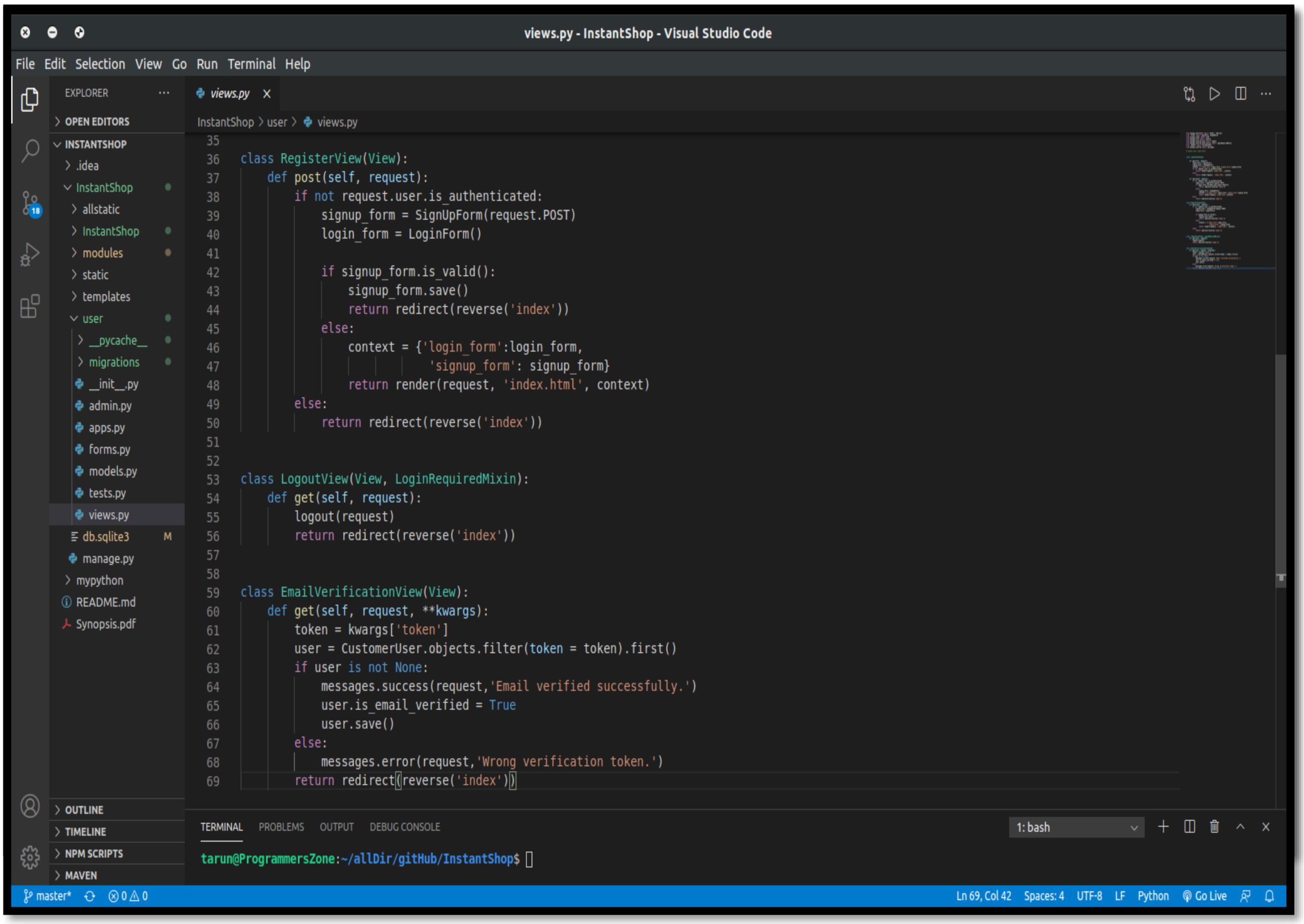
```
    def save(self):

        token = self.generate_token()

        user = User.objects.create(first_name = self.cleaned_data['firstname'],
                    last_name = self.cleaned_data['lastname'],
                    email = self.cleaned_data['email'],
                    username = self.cleaned_data['username']
        )

        user.set_password(self.cleaned_data['password1'])

        user.save()

        cust_user = CustomerUser(user = user,
                            address = self.cleaned_data['address'],
                            dob = self.cleaned_data['dob'],
                            phone_number = self.cleaned_data['tel'],
                            token = token
        )

        cust_user.save()

        mail(user.email,user.first_name,token)

    def generate_token(self):
        symbols_used = 'abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789!@$&*'

        token = ''

        while True:
            for i in range(30):
                token += symbols_used[random.randint(0,len(symbols_used)-1)]
            break
        return token
```

```
tarun@ProgrammersZone:~/allDir/gitHub/InstantShop$
```



```python
from django.db import models
from django.contrib.auth.models import User

# Create your models here.

def user_directory(instance, filename):
    return 'user_{0}/{1}'.format(instance.user.username,filename)

class CustomerUser(models.Model):
    user = models.OneToOneField(User, on_delete=models.CASCADE)
    address = models.TextField()
    dob = models.DateField()
    phone_number = models.TextField(max_length=13)
    photo = models.FileField(upload_to = user_directory)
    is_email_verified = models.BooleanField(default=False)
    token = models.CharField(max_length = 30, primary_key = True)

    def __str__(self):
        return str(self.user.username)

```

```
tarun@ProgrammersZone:~/allDir/gitHub/InstantShop$
```

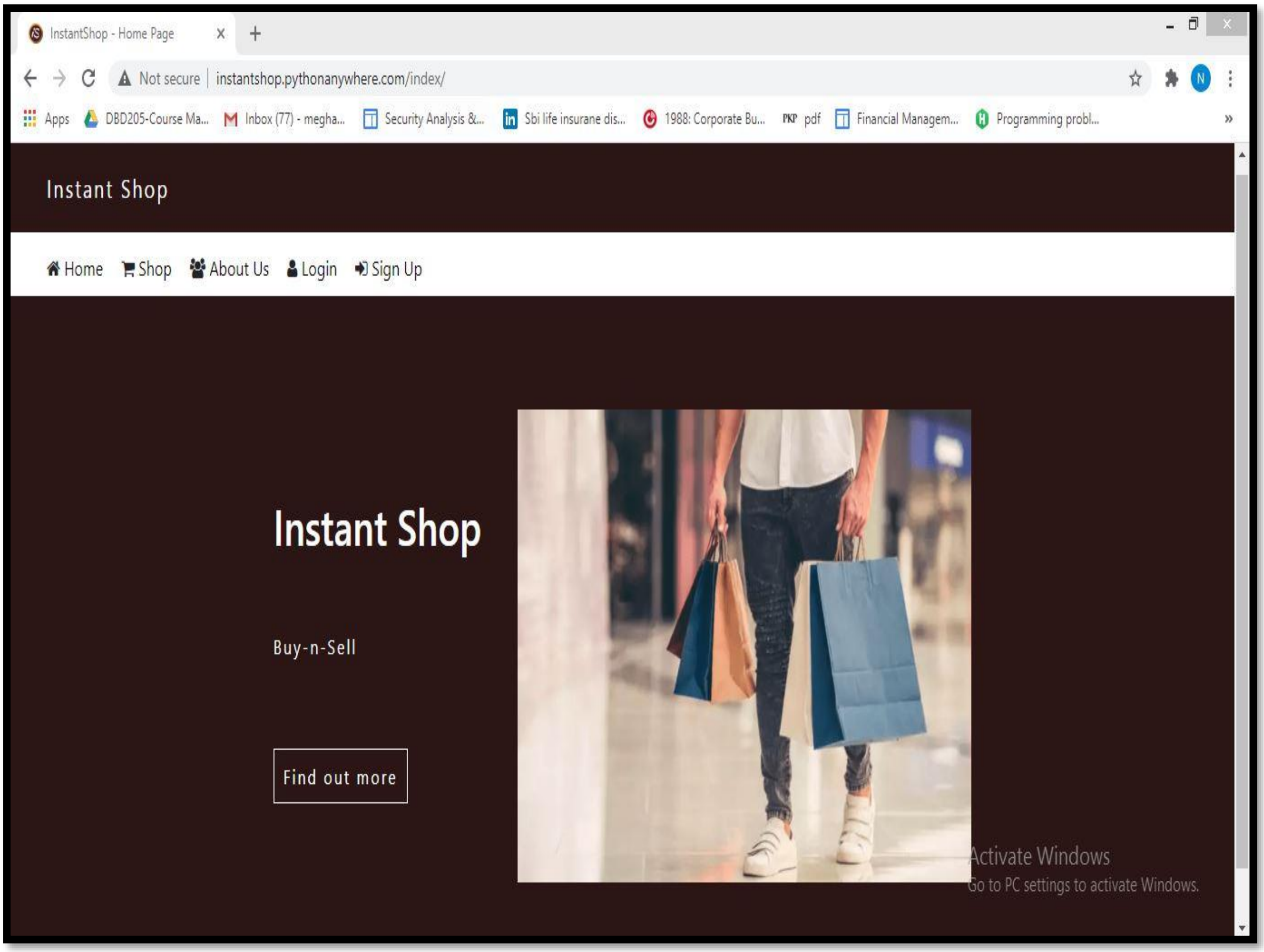


```python

class RegisterView(View):
    def post(self, request):
        if not request.user.is_authenticated:
            signup_form = SignUpForm(request.POST)
            login_form = LoginForm()

            if signup_form.is_valid():
                signup_form.save()
                return redirect(reverse('index'))
            else:
                context = {'login_form':login_form,
                            'signup_form': signup_form}
                return render(request, 'index.html', context)
        else:
            return redirect(reverse('index'))


class LogoutView(View, LoginRequiredMixin):
    def get(self, request):
        logout(request)
        return redirect(reverse('index'))


class EmailVerificationView(View):
    def get(self, request, **kwargs):
        token = kwargs['token']
        user = CustomerUser.objects.filter(token = token).first()
        if user is not None:
            messages.success(request,'Email verified successfully.')
            user.is_email_verified = True
            user.save()
        else:
            messages.error(request,'Wrong verification token.')
        return redirect(reverse('index'))
```

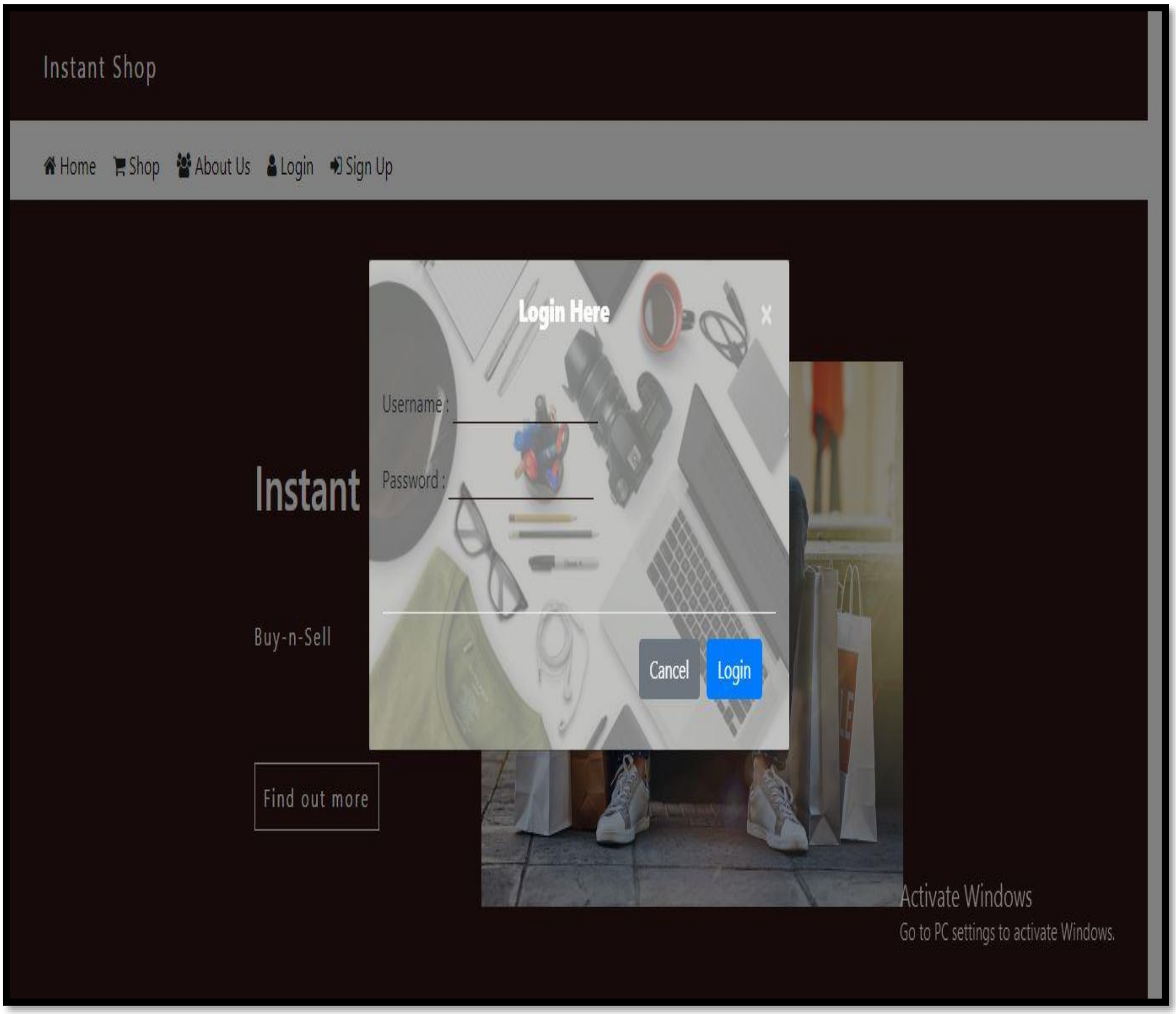
```
tarun@ProgrammersZone:~/allDir/gitHub/InstantShop$
```

Output Screenshots

Home Page

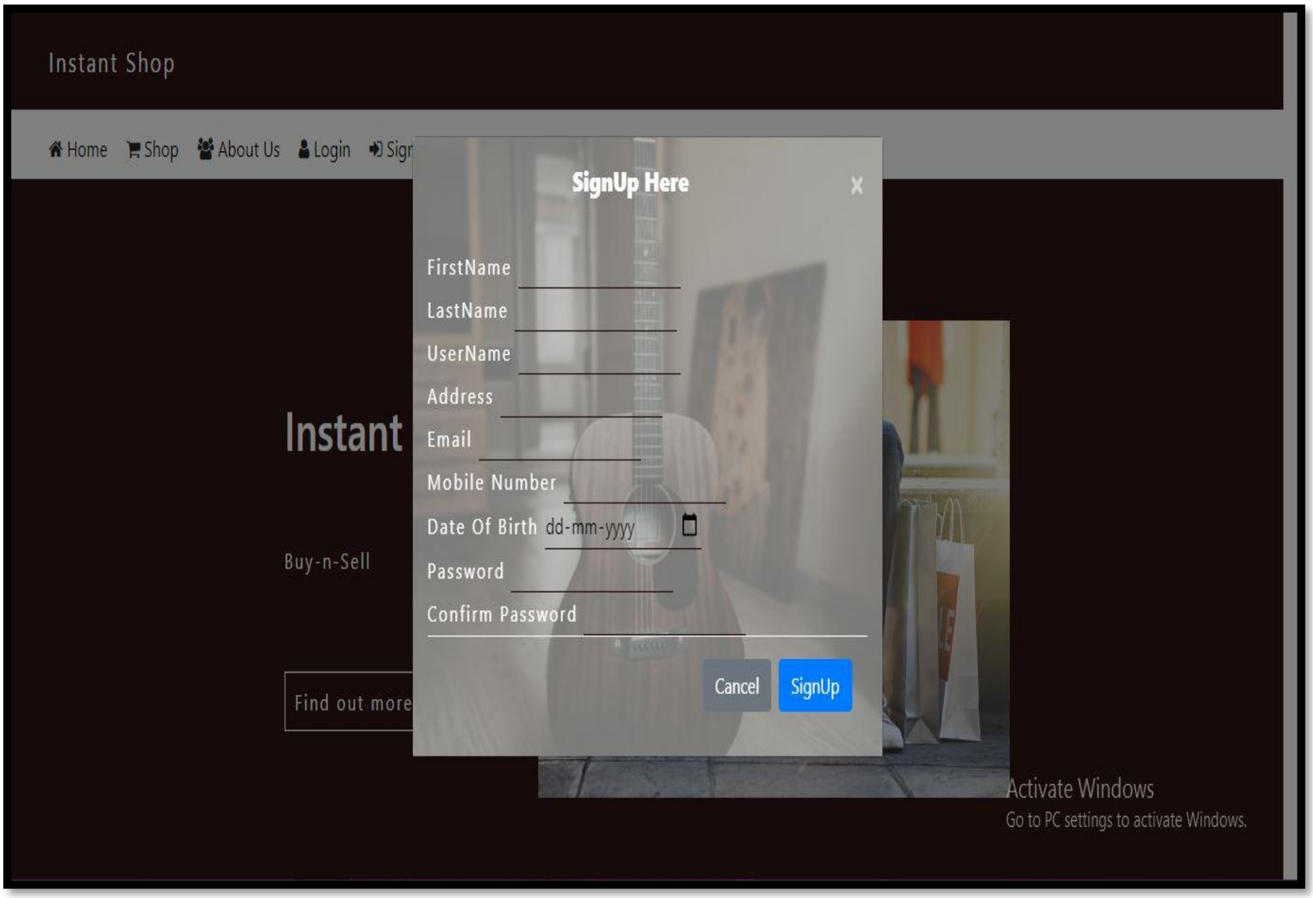


Login Page



Instant Shop
Home
Shop
About Us
Login
Sign Up
Instant
Buy-n-Sell
Find out more
Login Here
×
Username :
Password :
Cancel
Login
Activate Windows
Go to PC settings to activate Windows.

<u>Signup Page</u>


Instant Shop
Home
Shop
About Us
Login
Sign
SignUp Here
×
FirstName
LastName
UserName
Address
Email
Mobile Number
Date Of Birth dd-mm-yyyy
Password
Confirm Password
Cancel
SignUp
Instant
Buy-n-Sell
Find out more
Activate Windows
Go to PC settings to activate Windows.

References

- https://docs.djangoproject.com/en/3.1/topics/db/models/
- https://docs.djangoproject.com/en/3.1/topics/forms/
- https://docs.djangoproject.com/en/3.1/topics/class-based-views/
- https://docs.djangoproject.com/en/3.1/topics/http/urls/
- https://www.w3schools.com/html/
- https://www.w3schools.com/css/
- https://getbootstrap.com/