**TABLE OF CONTENTS**

* INTRODUCTION
* PROBLEM DEFINITION
* OBJECTIVE
* REQUIREMENT SPECIFICATION
* MODULE DISCRIPTION
* DATA FLOW DIAGRAM
* CODING
* FUTURE SCOPE OF THE PROJECT
* REFERENCE

**INTRODUCTION**

In the field of education, study materials and when you stuck on something if it is resolved by someone is very helpful for the students to learn more efficiently about the topic.

QnA platform generally assists the students and questioner to enhance their [learning](https://blog.teachmint.com/assessment-for-learning/)process. This encourages students and questioner to learn new concepts that significantly enhance their knowledge.

For example, if you are a student and preparing for [exams](https://www.teachmint.com/glossary/e/exam/)you need notes and you need to ask certain questions that can help you clear the exam. In this scenario student or learner can reach to QnA platform where he/she can find their queries which may be already present or asked by someone else or he/she can ask the same query.

**PROBLEM DEFINITION**

Now days when a person need certain answers for his/her questions he/she goes directly to internet and ask the question there. This platform is providing them a secure area where they can ask their queries.

**Purpose**

The purpose of this document is to specify requirements and to give guidelines for the development of above said project. In particular, it gives guidelines on how to prepare the above said project.

**Scope**

As this is generic software it can be used by a wide variety of room to automate the process of manually maintaining the records related to the subject of maintaining the entries and updating these records that too in a single click.

**OBJECTIVE**

Today’s world is computer world because most of work is doing with the help of computer. Dependency on computer is behind the few reasons. We cannot easily manage to store large number of data or information single handle. If we will be needing some information or data in urgency, then we cannot manage in manually these works are very difficult if we cannot use computer.

As the generic software it can be used by a wide verity of rooms to automate the process of manually maintaining records related to the subject of maintain the records of each personnel in room.

This software is basically updating the manual work of systems. So that organization can manage their record in efficient and organize them.

* The main objective of the project is to develop software that facilitates the data storage, data maintenance and its retrieval for the Questions and answers in an igneous way.
* To store the records of the users, queries and answers and feedback and admin that has the privileges to access, modify and delete any record.
* It will speed the processing of data and transaction.
* It will provide best security features such as provisions of passwords
* To develop a user-friendly system that requires minimal user training.

**System Context: -**

This section clearly depicts the environment and boundaries of QnA platform and the entities with which it interacts. It helps us see how the system fits into the existing scheme of things.

**Functional Requirement: -**

This Software must request Username and Password for access to data, only after authentication will allow access to the system.

**REQUIREMENTS SEPECIFICATIONS**

**HARDWARE AND SOFTWARE SPECIFICATION**

1. **Hardware Requirements**

* Pentium IV Processor
* 512 MB RAM
* 40 GB HDD
* Color Monitor
* Keyboard, Mouse

1. **Software Requirements:-**

* Python 3 (Any version)
* Any text editor (PyCharm, Notepad++ etc.)

**MODULE DESCRIPTION**

* **Index:**

The QnA Platform First Shows the index page. Here the user can see newly added questions and there answers and from there he/she can register to the new account or can login in order to get access to the full functionality of the application.

* **LOGIN:**

The QnA Platform first activates the login page. Here the user enters USER NAME AND PASSWORD and the system starts authentication process in which the USER NAME AND PASSWORD is matched with the existing USER NAME AND PASSWORD in the database. If the password matches, then it is allowed to the main page else it warns the user for invalid USER NAME AND PASSWORD.

**ADMIN:**

This module has software configuration that only admin can access this module. Here admin performs certain operations like adding new questions, answering some questions, deleting or seeing the information of users etc.

**USER**:

User are only seeing the all function.

**PROBLEM SPECIFICATION**

The definition of our problem lies in manual system and a fully automated system.

**1. Manual System: -**

The system is very time consuming and lazy. This system is more prone to error and sometimes the approach to various problems is unstructured.

**2. Technical System: -**

With the advent of latest technology if we do not update our system then our business result in losses gradually with time. The technical system contains the tools of latest trend i.e. computers, printers, FAX, Internet etc the system with the technology are very fast, accurate, user friendly and reliable.

**3. Need of QnA platform System: -**

QnA PLATFORM website is very needy for various people. This website helps them to rectify their problems and to get ahead if they stuck on some particular problems here they can ask them.

**4. The Proposed System: -**

The proposed system helps them in many ways. It helps them to see if their problem is already asked and solved by someone or they can ask them, people can also write blogs here.

**Data Flow Diagram:-**

ADMIN

QnA plateform

USER

View user

Feed Back

Change password

Signup

View vbogs

Add question

View question

Write blogs

View blogs

**CODING:-**

**Backend Coding: -**

**Views.py file: -**

from django.shortcuts import render, redirect

from django.contrib.auth.models import User, auth

from django.contrib.auth import authenticate, login, logout

from .models import \*

import random

def index(request):

    data = Questions.objects.all()

    usr\_data = Signup.objects.all()

    lst = []

    cnt = 1

    for i in data:

        if i.que is not None and i.que != "" and cnt <= 5:

            lst.append([i, str(i.id)])

        cnt += 1

    for i in range(len(lst)-1):

        random\_index = random.randint(0, len(lst)-1)

        temp = lst.pop(random\_index)

        lst.append(temp)

    print(len(lst))

    d = {'data': data, 'lst': lst, 'usr\_data': usr\_data}

    return render(request, 'index.html', d)

def logout(request):

    auth.logout(request)

    return redirect('/')

def questions(request):

    data = Questions.objects.all()

    d = {'data': data}

    return render(request, 'questions.html', d)

def login(request):

    error = ""

    if request.method == "POST":

        u = request.POST['uname']

        p = request.POST['pwd']

        user = auth.authenticate(username=u, password=p)

        try:

            if user.is\_staff:

                auth.login(request, user)

                error = "no"

            elif user is not None:

                auth.login(request, user)

                return redirect('user\_home')

                error = "not"

            else:

                error = "yes"

        except:

            error = "yes"

    d = {'error': error}

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

def admin\_home(request):

    if not request.user.is\_authenticated:

        return redirect('login')

    data = Questions.objects.all()

    usr\_data = Signup.objects.all()

    lst = []

    for i in data:

        if i.que is not None and i.que != "":

            lst.append([i, str(i.id)])

    d = {'data': data, 'lst': lst, 'usr\_data': usr\_data}

    return render(request, 'admin\_home.html', d)

def signup(request):

    error = ""

    if request.method == 'POST':

        f = request.POST['fname']

        l = request.POST['lname']

        e = request.POST['email']

        con = request.POST['contact']

        p = request.POST['pwd']

        gen = request.POST['gender']

        d = request.POST['dob']

        try:

            user = User.objects.create\_user(

                first\_name=f, last\_name=l,

username=e, password=p

)

            Signup.objects.create(

user=user, mobile=con, gender=gen,

                                  dob=d, follower=0, following=0, que\_asked=0,

answered=0

)

            error = "no"

        except:

            error = "yes"

    d = {'error': error}

    return render(request, 'signup.html', d)

def add\_que\_admin(request):

    if not request.user.is\_authenticated:

        return redirect('login')

    error = ""

    if request.method == 'POST':

        t = request.POST['topic']

        q = request.POST['que']

        try:

            Questions.objects.create(que=q, topic=t)

            error = "no"

        except:

            error = "yes"

    d = {'error': error}

    return render(request, 'add\_que\_admin.html', d)

def user\_home(request):

    data = Questions.objects.all()

    user = request.user

    # print(user.id)

    all\_user\_data = Signup.objects.all()

    user\_data = Signup.objects.all().filter(user\_\_exact=user.id)[0]

    # print(user\_data)

    lst = []

    for i in data:

        if i.que is not None and i.que != "":

            lst.append([i, str(i.id)])

    # print(len(lst))

    for i in range(len(lst)-1):

        random\_index = random.randint(0, len(lst)-1)

        temp = lst.pop(random\_index)

        lst.append(temp)

    d = {'data': data, 'lst': lst, 'u\_data': user\_data, 'all\_usr': all\_user\_data}

    if not request.user.is\_authenticated:

        return redirect('login')

    return render(request, 'user\_home.html', d)

def add\_que\_user(request):

    if not request.user.is\_authenticated:

        return redirect('login')

    error = ""

    if request.method == 'POST':

        t = request.POST['topic']

        q = request.POST['que']

        user = request.user

        user\_name = user.first\_name + " " + user.last\_name

        print(user\_name)

        try:

            Questions.objects.create(

                que=q, topic=t, user=user\_name, user\_id=user.id)

            error = "no"

        except:

            error = "yes"

    d = {'error': error}

    return render(request, 'add\_que\_user.html', d)

def view\_que\_admin(request):

    if not request.user.is\_authenticated:

        return redirect('login')

    data = Questions.objects.all()

    d = {'data': data}

    return render(request, 'view\_que\_admin.html', d)

def view\_que\_user(request):

    data = Questions.objects.all()

    hp = Helpful.objects.all()

    use\_data = Signup.objects.all()

    ur = request.user

    lst = []

    for i in data:

        if i.que is not None and i.que != "":

            lst.append([i, str(i.id)])

    d = {'data': data, 'data2': hp, 'ur': ur, 'usr\_data': use\_data, 'lst': lst}

    return render(request, 'view\_que\_user.html', d)

def add\_ans\_user(request, pid):

    if not request.user.is\_authenticated:

        return redirect('login')

    data = Questions.objects.get(id=pid)

    user = request.user

    user\_name = user.first\_name + " " + user.last\_name

    # print(user)

    if request.method == 'POST':

        a = request.POST['ans']

        t = request.POST['atopic']

        try:

            Questions.objects.create(

                ans=a, user=user\_name, user\_id=user.id, atopic=t, h=0, nh=0)

        except:

            pass

    d = {'data': data}

    return render(request, 'add\_ans\_user.html', d)

def view\_user(request):

    if not request.user.is\_authenticated:

        return redirect('login')

    data = Signup.objects.all()

    d = {'data': data}

    return render(request, 'view\_user.html', d)

def delete\_user(request, id):

    if not request.user.is\_authenticated:

        return redirect('login')

    student = User.objects.get(id=id)

    student.delete()

    return redirect('view\_user')

def delete\_que(request, id):

    if not request.user.is\_authenticated:

        return redirect('login')

    student = Questions.objects.get(id=id)

    student.delete()

    return redirect('admin\_home')

def delete\_ans(request, id):

    if not request.user.is\_authenticated:

        return redirect('login')

    student = Questions.objects.get(id=id)

    student.delete()

    return redirect('admin\_home')

def delete\_feedback(request, id):

    if not request.user.is\_authenticated:

        return redirect('admin\_login')

    feedback = Feedback.objects.get(id=id)

    feedback.delete()

    return redirect('view\_feedback')

def view\_feedback(request):

    if not request.user.is\_authenticated:

        return redirect('admin\_login')

    data = Feedback.objects.all()

    d = {'data': data}

    return render(request, 'view\_feedback.html', d)

def change\_password(request):

    if not request.user.is\_authenticated:

        return redirect('login')

    error = ""

    if request.method == "POST":

        c = request.POST['currentpassword']

        n = request.POST['newpassword']

        try:

            u = User.objects.get(id=request.user.id)

            if u.check\_password(c):

                u.set\_password(n)

                u.save()

                error = "no"

            else:

                error = "not"

        except:

            error = "yes"

    d = {'error': error}

    return render(request, 'change\_password.html', d)

def feedback(request):

    error = ""

    if request.method == 'POST':

        n = request.POST['fname']

        p = request.POST['fphone']

        e = request.POST['femail']

        c = request.POST['fcomment']

        try:

            Feedback.objects.create(

                feedback\_name=n, feedback\_contact=p,

feedback\_email=e, feedback\_comment=c

)

            error = "no"

        except:

            error = "yes"

    d = {'error': error}

    return render(request, 'feedback.html', d)

def helpful(request, id):

    que = Questions.objects.get(id=id)

    hp = Helpful.objects.all()

    for i in hp:

        n = i.num

        n1 = i.numid

        print(id)

        if n1 == id:

            try:

                i.numid = id

                i.num = n+1

                i.save()

                break

            except:

                pass

    else:

        Helpful.objects.create(numid=id, num=1)

        # break

    return redirect('view\_que\_user')

def write\_blog(request):

    if not request.user.is\_authenticated:

        return redirect('login')

    user = request.user

    u\_name = user.first\_name + " " + user.last\_name

    blog\_obj = None

    if request.method == 'POST':

        t = request.POST['btopic']

        bc = request.POST['blog']

        try:

            blog\_obj = Blog.objects.create(

                user\_id=user.id, user\_name=u\_name, topic=t, blog\_c=bc)

            return blog(request, blog\_obj.id)

        except:

            pass

    return render(request, 'write\_blog.html')

def view\_blogs(request):

    data = Blog.objects.all()

    usr\_data = Signup.objects.all()

    data.reverse()

    d = {'data': data, 'usr\_data': usr\_data}

    return render(request, 'view\_blogs.html', d)

def view\_blog\_admin(request):

    data = Blog.objects.all()

    usr\_data = Signup.objects.all()

    data.reverse()

    d = {'data': data, 'usr\_data': usr\_data}

    return render(request, 'view\_blog\_admin.html', d)

def delete\_blog(request, id):

    if not request.user.is\_authenticated:

        return redirect('login')

    blog = Blog.objects.get(id=id)

    blog.delete()

    return redirect('view\_blog\_admin')

def view\_que\_visitor(request):

    data = Questions.objects.all()

    hp = Helpful.objects.all()

    d = {'data': data, 'data2': hp}

    return render(request, 'view\_que\_visitor.html', d)

def add(request, id):

    all = Questions.objects.all()

    for i in all:

        if i.id == id:

            try:

                i.h = i.h+1

                i.save()

                break

            except:

                pass

    return redirect('view\_que\_user')

def add\_on\_admin(request, id):

    all = Questions.objects.all()

    for i in all:

        if i.id == id:

            try:

                i.h = i.h+1

                i.save()

                break

            except:

                pass

    return redirect('admin\_home')

def addd(request, id):

    all = Questions.objects.all()

    for i in all:

        if i.id == id:

            try:

                i.nh = i.nh+1

                i.save()

                break

            except:

                pass

    return redirect('view\_que\_user')

def addd\_on\_admin(request, id):

    all = Questions.objects.all()

    for i in all:

        if i.id == id:

            try:

                i.nh = i.nh+1

                i.save()

                break

            except:

                pass

    return redirect('admin\_home')

def follow\_inc(request, id):

    if not request.user.is\_authenticated:

        return redirect('login')

    user = request.user

    jiska\_follow\_inc\_krna = Signup.objects.all().filter(user\_\_exact=id)[0]

    jiska\_following\_inc\_krna = Signup.objects.all().filter(

        user\_\_exact=user.id)[0]

    # print(jiska\_following\_inc\_krna.following)

    try:

        jiska\_following\_inc\_krna.following = jiska\_following\_inc\_krna.following + 1

        jiska\_following\_inc\_krna.save()

        jiska\_follow\_inc\_krna.follower = jiska\_follow\_inc\_krna.follower + 1

        jiska\_follow\_inc\_krna.save()

    except:

        pass

    return redirect('user\_home')

def blog(request, id):

    blog = Blog.objects.all().filter(id\_\_exact=id)[0]

    usr\_data = Signup.objects.all().filter(user\_\_exact=blog.user\_id)[0]

    print(usr\_data)

    return render(request, 'blog.html', {'blog\_obj': blog, 'user\_data': usr\_data})

Admin.py file: -

from django.contrib import admin

from .models import \*

admin.site.register(Signup)

admin.site.register(Questions)

admin.site.register(Feedback)

admin.site.register(Helpful)

admin.site.register(Blog)

Urls.py file: -

from django.contrib import admin

from django.urls import path

from study.views import \*

from django.conf import settings

from django.conf.urls.static import static

urlpatterns = [

    path('admin/', admin.site.urls),

    path('', index, name="index"),

    path('login', login, name="login"),

    path('admin\_home', admin\_home, name="admin\_home"),

    path('logout', logout, name="logout"),

    path('questions', questions, name="questions"),

    path('signup', signup, name="signup"),

    path('user\_home', user\_home, name="user\_home"),

    path('add\_que\_admin', add\_que\_admin, name="add\_que\_admin"),

    path('add\_que\_user', add\_que\_user, name="add\_que\_user"),

    path('view\_que\_visitor', view\_que\_visitor, name="view\_que\_visitor"),

    path('view\_que\_admin', view\_que\_admin, name="view\_que\_admin"),

    path('view\_que\_user', view\_que\_user, name="view\_que\_user"),

    path('add\_ans\_user/<int:pid>', add\_ans\_user, name="add\_ans\_user"),

    path('view\_user', view\_user, name="view\_user"),

    path('delete\_user/<int:id>', delete\_user, name="delete\_user"),

    path('delete\_que/<int:id>', delete\_que, name="delete\_que"),

    path('delete\_ans/<int:id>', delete\_ans, name="delete\_ans"),

    path('change\_password', change\_password, name="change\_password"),

    path('feedback', feedback, name="feedback"),

    path('view\_feedback', view\_feedback, name="view\_feedback"),

    path('delete\_feedback/<int:id>', delete\_feedback, name="delete\_feedback"),

    path('view\_que\_admin', view\_que\_admin, name="view\_que\_admin"),

    path('helpful/<int:id>', helpful, name="helpful"),

    path('add/<int:id>', add, name="add"),

    path('add\_on\_admin/<int:id>', add\_on\_admin, name="add\_on\_admin"),

    path('addd/<int:id>', addd, name="addd"),

    path('addd\_on\_admin/<int:id>', addd\_on\_admin, name="addd\_on\_admin"),

    path('write\_blog', write\_blog, name="write\_blog"),

    path('view\_blogs', view\_blogs, name="view\_blogs"),

    path('view\_blog\_admin', view\_blog\_admin, name="view\_blog\_admin"),

    path('delete\_blog/<int:id>', delete\_blog, name="delete\_blog"),

    path('follow\_inc/<int:id>', follow\_inc, name="follow\_inc"),

    path('blog/<int:id>', blog, name="blog"),

] + static(settings.MEDIA\_URL, document\_root=settings.MEDIA\_ROOT)

Settings.py file: -

"""

Django settings for study\_material project.

Generated by 'django-admin startproject' using Django 3.1.6.

For more information on this file, see

https://docs.djangoproject.com/en/3.1/topics/settings/

For the full list of settings and their values, see

https://docs.djangoproject.com/en/3.1/ref/settings/

"""

from pathlib import Path

import os

# Build paths inside the project like this: BASE\_DIR / 'subdir'.

BASE\_DIR = Path(\_\_file\_\_).resolve().parent.parent

# Quick-start development settings - unsuitable for production

# See https://docs.djangoproject.com/en/3.1/howto/deployment/checklist/

# SECURITY WARNING: keep the secret key used in production secret!

SECRET\_KEY = 'kg+5-tua@g1pm)x(pw9)t9dlx-v-t2w\*g^q=-d)zgm^+^9-0^x'

# SECURITY WARNING: don't run with debug turned on in production!

DEBUG = True

ALLOWED\_HOSTS = []

# Application definition

INSTALLED\_APPS = [

    'study',

    'django.contrib.admin',

    'django.contrib.auth',

    'django.contrib.contenttypes',

    'django.contrib.sessions',

    'django.contrib.messages',

    'django.contrib.staticfiles',

    'livereload',

    'ckeditor',

]

MIDDLEWARE = [

    'django.middleware.security.SecurityMiddleware',

    'django.contrib.sessions.middleware.SessionMiddleware',

    'django.middleware.common.CommonMiddleware',

    'django.middleware.csrf.CsrfViewMiddleware',

    'django.contrib.auth.middleware.AuthenticationMiddleware',

    'django.contrib.messages.middleware.MessageMiddleware',

    'django.middleware.clickjacking.XFrameOptionsMiddleware',

    'livereload.middleware.LiveReloadScript',

]

ROOT\_URLCONF = 'study\_material.urls'

TEMPLATES = [

    {

        'BACKEND': 'django.template.backends.django.DjangoTemplates',

        'DIRS': [],

        'APP\_DIRS': True,

        'OPTIONS': {

            'context\_processors': [

                'django.template.context\_processors.debug',

                'django.template.context\_processors.request',

                'django.contrib.auth.context\_processors.auth',

                'django.contrib.messages.context\_processors.messages',

            ],

        },

    },

]

WSGI\_APPLICATION = 'study\_material.wsgi.application'

# Database

# https://docs.djangoproject.com/en/3.1/ref/settings/#databases

DATABASES = {

    'default': {

        'ENGINE': 'django.db.backends.sqlite3',

        'NAME': BASE\_DIR / 'db.sqlite3',

    }

}

# Password validation

# https://docs.djangoproject.com/en/3.1/ref/settings/#auth-password-validators

AUTH\_PASSWORD\_VALIDATORS = [

    {

        'NAME': 'django.contrib.auth.password\_validation.UserAttributeSimilarityValidator',

    },

    {

        'NAME': 'django.contrib.auth.password\_validation.MinimumLengthValidator',

    },

    {

        'NAME': 'django.contrib.auth.password\_validation.CommonPasswordValidator',

    },

    {

        'NAME': 'django.contrib.auth.password\_validation.NumericPasswordValidator',

    },

]

# 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/'

MEDIA\_ROOT = os.path.join(BASE\_DIR, 'media')

MEDIA\_URL = 'media/'

**Frontend coding: -**

Add\_answer\_user.html

{% include 'user\_navigation.html' %}

<style>

  form {

    display: flex;

    width: 80%;

    margin: 60px auto;

  }

  .ques-ask {

    width: 40%;

    font-size: 1.5rem;

    font-weight: bold;

  }

  button {

    padding: 20px;

    background-color: #626ee3;

    color: white;

    border: none;

    font-size: 1.5rem;

    font-weight: bold;

    margin: 10px 0;

    border-radius: 5px;

    cursor: pointer;

  }

</style>

<form method="post">

  {% csrf\_token %}

  <div class="ques-ask">

    <p>{{ data.que }}</p>

    <input

      type="text"

      value="{{ data.id }}"

      name="atopic"

      readonly

      style="display: none"

    />

  </div>

  <div class="ans-area">

    <textarea id="mytex" name="ans" cols="30" rows="10"></textarea>

    <div style="display: flex; justify-content: flex-end">

      <button>Submit</button>

    </div>

  </div>

</form>

{% include 'footer.html' %}

Add\_question\_user.html: -

{% load static %} {% include 'user\_navigation.html' %}

<style>

  .frm {

    width: 60%;

    margin: 40px auto;

  }

  select {

    padding: 10px;

    width: 250px;

    margin-bottom: 15px;

  }

  button {

    padding: 20px;

    background-color: #626ee3;

    color: white;

    border: none;

    font-size: 1.5rem;

    font-weight: bold;

    margin: 10px 0;

    border-radius: 5px;

    cursor: pointer;

  }

</style>

<form class="frm" method="post">

  {% csrf\_token %}

  <select class="form-control col-10" name="topic">

    <option value="Science">Science</option>

    <option value="Engineering">Engineering</option>

    <option value="Personal">Personal</option>

    <option value="Mathematics">Mathematics</option>

    <option value="Tourism">Tourism</option>

  </select>

  <textarea name="que" id="mytex"></textarea>

  <div><button>Submit</button></div>

</form>

{% if error == "no" %}

<script>

  alert("Question Added Successfully")

  window.location = ('{% url 'add\_que\_user' %}')

</script>

{% endif %} {% if error == "yes" %}

<script>

  alert("Something Went Wrong try again...")

  window.location = ('{% url 'add\_que\_user' %}')

</script>

{% endif %}

{% include 'footer.html' %}

Admin\_home.html: -

<html>

  <head>

    {% load static %}

    <title>Study Material</title>

    <meta charset="utf-8" />

    <meta name="viewport" content="width=device-width, initial-scale=1" />

    <link rel="stylesheet"

      href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css"/>

    <link rel="stylesheet" href="{% static 'css/admin\_home.css' %}" />

  </head>

  <body>

    <div class="main-container">

      <div class="body-container">

        <a href="{% url 'index' %}"><p class="logo">Q n A</p></a>

        <div class="all-content">

          {% for i, j in lst %}

          <div class="que-cont">

            <div class="que">

              <p>Q.&nbsp&nbsp</p>

              <p>{{ i.que|safe }}</p>

            </div>

            <div class="asker">

              <div class="info">

                <div class="img-usr">

                  {% for m in usr\_data %} {% if m.user.id == i.user\_id %}

                  <img src="{{ MEDIA\_URL }}/media/{{ m.usr\_img }}" alt="" />

                  {% endif %} {% endfor %}

                </div>

                {{ i.user }}

              </div>

              <div class="del-que">

                <a

                  href="{% url 'delete\_que' i.id %}"

                  onclick="return confirm('Are you Sure?')"

                  ><button>Delete this question</button></a

                >

              </div>

            </div>

          </div>

          {% for k in data %} {% if k.atopic == j %}

          <div class="ans-cont">

            <div class="anss-img">

              {% for m in usr\_data %} {% if m.user.id == k.user\_id %}

              <img src="{{ MEDIA\_URL }}/media/{{ m.usr\_img }}" alt="" />

              {% endif %} {% endfor %}

            </div>

            <div class="ans">

              <p>{{ k.ans|safe }}</p>

              <div class="strip">

                <p>

                  <span

                    ><a href="{% url 'add\_on\_admin' k.id %}"

                      ><i class="fa fa-thumbs-up"></i

                    ></a>

                    {{ k.h }} &nbsp&nbsp</span

                  ><span

                    ><a href="{% url 'addd\_on\_admin' k.id %}"

                      ><i class="fa fa-thumbs-down"></i

                    ></a>

                    {{ k.nh }}</span

                  >

                </p>

                <p>

                  <a href="{% url 'add\_ans\_user' i.id%}"

                    ><button class="but" style="background: #626ee3">

                      Answer

                    </button></a

                  >

                  <a

                    href="{% url 'delete\_ans' k.id %}"

                    onclick="return confirm('Are you Sure?')"

                    ><button class="but" style="background-color: #eb5353">

                      delete Answer

                    </button></a

                  >

                </p>

              </div>

            </div>

          </div>

          {% endif %} {% endfor %} {% endfor %}

        </div>

      </div>

      <div class="menu-admin">

        <div class="admin-navigation">

          <p>

            Welcome,

            <span style="font-weight: bold; font-size: 1.3rem">

              {{ user }}</span

            >

          </p>

          <ul class="admin-nav">

            <a href="{% url 'admin\_home' %}"

              ><li class="nav-items-admin active">Home</li></a

            >

            <a href="{% url 'view\_user' %}"

              ><li class="nav-items-admin">View Users</li></a

            >

            <a href=""><li class="nav-items-admin">View Blogs</li></a>

            <a href="{% url 'change\_password' %}"

              ><li class="">Change Password</li></a

            >

            <a href="{% url 'view\_feedback' %}"

              ><li class="nav-items-admin">View Feedback</li></a

            >

          </ul>

          <a href="{% url 'logout' %}"

            ><button class="buttn">Log Out</button></a

          >

        </div>

      </div>

    </div>

  </body>

</html>

Blog.html: -

{% load static %} {% include 'user\_navigation.html' %}

<link rel="stylesheet" href="{% static 'css/blog.css' %}" />

<div class="banner">

  <h1>{{ blog\_obj.topic }}</h1>

</div>

<div class="img-cont">

  <img src="{{ MEDIA\_URL }}/media/{{ user\_data.usr\_img }}" alt="" />

</div>

<div class="u\_name">

  <p>{{ blog\_obj.user\_name }}</p>

</div>

<div class="blog-cont">

  <p>{{ blog\_obj.blog\_c|safe }}</p>

</div>

{% include 'footer.html' %}

Change\_password.html: -

<html>

  <head>

    {% load static %}

    <title>Study Material</title>

    <meta charset="utf-8" />

    <meta name="viewport" content="width=device-width, initial-scale=1" />

    <link

      rel="stylesheet"

      href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css"

    />

    <link rel="stylesheet" href="{% static 'css/admin\_home.css' %}" />

    <style>

      .all-content {

        margin-left: 80;

        margin-right: 20px;

      }

      .active::before {

        content: "";

        height: 100%;

        width: 100%;

        position: absolute;

        background-color: #f7f7f7;

        left: -170px;

        top: 0;

      }

      form {

        display: flex;

        flex-direction: column;

        align-items: center;

      }

      .custom-field {

        position: relative;

        font-size: 14px;

        padding-top: 20px;

        margin-bottom: 5px;

      }

      .custom-field > input {

        border: 2px solid #3a5ba0;

        appearance: none;

        padding: 12px;

        width: 250px;

        border-radius: 3px;

        background: #f2f2f2;

        outline: none;

        display: block;

      }

      .custom-field .placeholder {

        position: absolute;

        left: 12px;

        top: calc(50% + 10px);

        transform: translateY(-50%);

        color: #aaa;

        font-family: "Poppins";

        transition: top 0.3s ease, font-size 0.3s ease, color 0.3s ease;

      }

      .custom-field input:valid + .placeholder,

      .custom-field input:focus + .placeholder {

        top: 20px;

        font-size: 14px;

        color: #222;

        background: #f7f7f7;

        padding: 4px;

      }

      .custom-field.one input:valid,

      .custom-field.one input:focus {

        border-color: var(--primary);

        transition-delay: 0.1s;

      }

      .inp {

        background: var(--primary);

        border: none;

        font-size: 1rem;

        color: white;

        margin-top: 8px;

        cursor: pointer;

        appearance: none;

        padding: 12px;

        width: 250px;

        border-radius: 3px;

        outline: none;

        display: block;

      }

    </style>

    <script>

      function checkpass() {

        if (

          document.changepassword.newpassword.value !=

          document.changepassword.confirmpassword.value

        ) {

          alert("New Password and Confirm Password field does not match");

          document.changepassword.newpassword.focus();

          return false;

        }

        return true;

      }

    </script>

  </head>

  <body>

    <div class="main-container">

      <div class="body-container">

        <a href="{% url 'index' %}"><p class="logo">Q n A</p></a>

        <div class="all-content">

          <form

            name="changepassword"

            method="post"

            onsubmit="return checkpass()"

          >

            {% csrf\_token %}

            <label class="custom-field one">

              <input type="password" name="currentpassword" required />

              <span class="placeholder">Old password</span>

            </label>

            <label class="custom-field one">

              <input type="password" name="newpassword" required />

              <span class="placeholder">New Password</span>

            </label>

            <label class="custom-field one">

              <input type="password" name="confirmpassword" required />

              <span class="placeholder">Confirm Password</span>

            </label>

            <input class="inp" type="submit" value="Submit" />

          </form>

          {% if error == "no" %}

          <script>

            alert("Password Changed Successfully")

            window.location = ('{% url 'Logout' %}')

          </script>

          {% endif %} {% if error == "yes" %}

          <script>

            alert("Something went wrong, Try Again");

          </script>

          {% endif %} {% if error == "not" %}

          <script>

            alert("Your Current Password is wrong");

          </script>

          {% endif %}

        </div>

      </div>

      <div class="menu-admin">

        <div class="admin-navigation">

          <p>

            Welcome,

            <span style="font-weight: bold; font-size: 1.3rem">

              {{ user }}</span

            >

          </p>

          <ul class="admin-nav">

            <a href="{% url 'admin\_home' %}"><li class="">Home</li></a>

            <a href="{% url 'view\_user' %}"><li class="">View Users</li></a>

            <a href="{% url 'view\_blog\_admin' %}"

              ><li class="">View Blogs</li></a

            >

            <a href=""><li class="active">Change Password</li></a>

            <a href="{% url 'view\_feedback' %}"

              ><li class="">View Feedback</li></a

            >

          </ul>

          <a href="{% url 'logout' %}"

            ><button class="buttn">Log Out</button></a

          >

        </div>

      </div>

    </div>

  </body>

</html>

Feedback.html: -

<!DOCTYPE html>

<html lang="en">

  <head>

    {% load static %}

    <meta charset="UTF-8" />

    <meta http-equiv="X-UA-Compatible" content="IE=edge" />

    <meta name="viewport" content="width=device-width, initial-scale=1.0" />

    <title>Feedback</title>

    <link rel="stylesheet" href="{% static 'css/feedback.css' %}" />

    <script

      src="https://cdn.tiny.cloud/1/no-api-key/tinymce/6/tinymce.min.js"

      referrerpolicy="origin"

    ></script>

    <script>

      tinymce.init({

        selector: "#mytex",

      });

    </script>

  </head>

  <body>

    <a href="{% url 'index' %}"><div class="logo">Q n A</div></a>

    <form method="post">

      {% csrf\_token %}

      <div class="info">

        <div>

          <input

            type="text"

            name="fname"

            id=""

            placeholder="Enter Name"

            required

          />

          <input

            type="number"

            name="fphone"

            id=""

            placeholder="Phone Number"

            required

          />

          <input

            type="email"

            name="femail"

            id=""

            placeholder="email"

            required

          />

        </div>

        <div>

          <textarea name="fcomment" id="mytex" cols="30" rows="10"></textarea>

          <button class="btn">Submit</button>

        </div>

      </div>

    </form>

    {% if error == "no" %}

    <script>

      alert("Feedback Submitted Successfully")

      window.location = ('{% url 'feedback' %}')

    </script>

    {% endif %} {% if error == "yes" %}

    <script>

      alert("Something went wrong, Try Again");

    </script>

    {% endif %}

  </body>

</html>

Footer.html: -

{% load static %}

<style>

    .footer{

        width: 100%;

        height: 300px;

        background: #626ee3;

        display: flex;

        align-items: center;

        justify-content: center;

        color: white;

        margin-top: 50px;

    }

</style>

<div class="footer">

    <p>&#169 copyright 2022 QnA private organisation.</p>

</div>

(Home page) Index.html: -

{% load static %}

{% include 'navigation.html' %}

<div class="main-container">

    <div class="main-left">

        <div class="left-container">

            <p class="ttl">A Platform<br/> where people can:</p>

            <div class="word-slider">

                <ul class="words">

                    <li class="word">ASK QUESTIONS.</li>

                    <li class="word">WRITE ANSWERS.</li>

                    <li class="word">POST BLOGS.</li>

                    <li class="word">AND MANY MORE.</li>

                    <!-- <li class="word">ASK QUESTIONS.</li> -->

                </ul>

             </div>

        </div>

    </div>

    <div class="main-right">

        <!-- <div class="circle"></div> -->

        <div class="square s1"></div>

        <div class="square s2"></div>

        <div class="square s3"></div>

        <div class="square s4"></div>

        <div class="square s1"></div>

        <div class="square s2"></div>

        <div class="square s3"></div>

        <div class="square s4"></div>

        <div class="square s1"></div>

        <!-- <div class="square s2"></div>

        <div class="square s3"></div>

        <div class="square s4"></div> -->

    </div>

</div>

<h1 class="ntxt">Newly Asked Questions</h1>

<div class="line2"></div>

{% for i, j in lst %}

<div class="questions">

    <div class="usr-details">

        <div class="img-usr">

            {% for m in usr\_data %}

            {% if m.user.id == i.user\_id %}

            <img src="{{ MEDIA\_URL }}/media/{{ m.usr\_img }}" alt="">

            {% endif %}

            {% endfor %}

        </div>

        <p class="usr-name">{{ i.user }}</p>

    </div>

    <div class="ques-details">

        <div class="date-det-con">

            <div class="date">

                <p>Asked: 22/05/2012</p>

            </div>

            <div class="cate">

                <p>In: <span style="color: blue;">{{ i.topic }}</span></p>

            </div>

        </div>

        <p class="question">{{ i.que }}</p>

        {% for k in data %}

        {% if k.atopic == j %}

        <div class="answers">

            <div class="ans-usr">

                <div class="ans-usr-img">

                    {% for m in usr\_data %}

                    {% if m.user.id == k.user\_id %}

                    <img src="{{ MEDIA\_URL }}/media/{{ m.usr\_img }}" alt="">

                    {% endif %}

                    {% endfor %}

                </div>

                <div class="ans-usr-name">{{ k.user }}</div>

            </div>

            <div class="usr-ans">

                <p class="ans">{{ k.ans|safe }}</p>

            </div>

            <div class="ans-strip">

                <div class="vote">

                    <a href='{% url 'login' %}'><span class="thum"><i class="fa fa-thumbs-up"></i> {{ k.h }}</span></a>

                    <a href='{% url 'login' %}'><span class="thum"><i class="fa fa-thumbs-down"></i> {{ k.nh }}</span></a>

                </div>

                <div class="ask">

                    <a href='{% url 'login' %}'>Answer This Question.</a>

                </div>

            </div>

        </div>

        {% endif %}

        {% endfor %}

    </div>

</div>

{% endfor %}

{% include 'footer.html' %}

Login.html: -

<html>

  <head>

    {% load static %}

    <title>Log In</title>

    <meta charset="utf-8" />

    <meta name="viewport" content="width=device-width, initial-scale=1" />

    <link

      rel="stylesheet"

      href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css"

    />

    <link rel="stylesheet" href="{% static 'css/main.css' %}" />

  </head>

  <body>

    <header>

      <a href="{% url 'index' %}"><div class="logo">Q n A</div></a>

      <div class="log">

        <a href="{% url 'signup' %}"

          ><button class="log-btn">

            <i class="fa fa-user-plus"></i> Sign Up

          </button></a

        >

      </div>

    </header>

    <div class="form-container">

      <div class="left-form">

        <div class="cont-form">

          <img src="../static/h.png" alt="" />

          <p>

            Login to see full features where you can add your questions and you

            can answer questions you can post your own blog isn't it great.<br />Facing

            any problems or have any queries please contact us.

          </p>

        </div>

      </div>

      <div class="right-form">

        <div class="login-txt">Log In</div>

        <form class="for" method="post">

          {% csrf\_token %}

          <label class="custom-field one">

            <input type="text" name="uname" required />

            <span class="placeholder">User name</span>

          </label>

          <label class="custom-field one">

            <input type="password" name="pwd" required />

            <span class="placeholder">password</span>

          </label>

          <input class="inp" type="submit" value="Login" />

        </form>

      </div>

    </div>

    {% if error == "no" %}

    <script>

      window.location = ('{% url 'admin\_home' %}')

    </script>

    {% endif %} {% if error == "not" %}

    <script>

      window.location = ('{% url 'user\_home' %}')

    </script>

    {% endif %} {% if error == "yes" %}

    <script>

      alert("Username Passowrd Not Match")

      window.location = ('{% url 'admin\_home' %}')

    </script>

    {% endif %}

  </body>

</html>

Navigation.html: -

<html>

  <head>

    {% load static %}

    <title>QnA-Home</title>

    <meta charset="utf-8" />

    <meta name="viewport" content="width=device-width, initial-scale=1" />

    <link

      rel="stylesheet"

      href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css"

    />

    <link rel="stylesheet" href="{% static 'css/main.css' %}" />

  </head>

  <body>

    <header>

      <a href="{% url 'index' %}"><div class="logo">Q n A</div></a>

      <div class="log">

        <a href="{% url 'login' %}"

          ><button class="log-btn" style="background: none; color: #626ee3">

            <i class="fa fa-sign-in"></i> Log in

          </button></a

        >

        <a href="{% url 'signup' %}"

          ><button class="log-btn">

            <i class="fa fa-user-plus"></i> Sign Up

          </button></a

        >

      </div>

    </header>

  </body>

</html>

Signup.html: -

<html>

  <head>

    {% load static %}

    <title>QnA-Home</title>

    <meta charset="utf-8" />

    <meta name="viewport" content="width=device-width, initial-scale=1" />

    <link

      rel="stylesheet"

      href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css"

    />

    <link rel="stylesheet" href="{% static 'css/main.css' %}" />

    <script>

      function checkpass() {

        if (document.signup.pwd.value != document.signup.cpwd.value) {

          alert("Password Does Not Match");

          document.signup.cpwd.focus();

          return false;

        }

        return true;

      }

    </script>

  </head>

  <body>

    <header>

      <a href="{% url 'index' %}"><div class="logo">Q n A</div></a>

      <div class="log">

        <a href="{% url 'index' %}"

          ><button class="log-btn"><i class="fa fa-home"></i> Home</button></a

        >

      </div>

    </header>

    <div class="sign-container">

      <div class="left-sign">

        <p>Create an Account</p>

        <div class="line"></div>

        <form

          class="sign-form"

          onsubmit="return checkpass();"

          name="signup"

          method="post"

          enctype="multipart/form-data"

        >

          {% csrf\_token %}

          <div class="form-outer">

            <div class="left-c1">

              <label class="custom-field one">

                <input type="text" name="fname" required />

                <span class="placeholder">first name</span>

              </label>

              <label class="custom-field one">

                <input type="number" name="contact" required />

                <span class="placeholder">phone no.</span>

              </label>

              <label class="custom-field one">

                <input type="password" name="pwd" required />

                <span class="placeholder">password</span>

              </label>

              <label class="custom-field one">

                <input type="date" name="dob" required />

              </label>

            </div>

            <div class="left-c2">

              <label class="custom-field one">

                <input type="text" name="lname" required />

                <span class="placeholder">last name</span>

              </label>

              <label class="custom-field one">

                <input type="email" name="email" required />

                <span class="placeholder">email address</span>

              </label>

              <label class="custom-field one">

                <input type="password" name="cpwd" required />

                <span class="placeholder">repeat password</span>

              </label>

              <label class="rad"

                >Gender :

                <input

                  type="radio"

                  name="gender"

                  value="Male"

                  id="customRadioInLine1"

                />

                <label class="custom-control-label" for="customRadioInLine1"

                  >Male</label

                >

                <input

                  type="radio"

                  name="gender"

                  value="Female"

                  id="customRadioInLine2"

                />

                <label class="custom-control-label" for="customRadioInLine2"

                  >Female</label

                >

              </label>

            </div>

          </div>

          <input class="inp sgn-btn" type="submit" value="Sign up" />

        </form>

      </div>

      <div class="right-sign">

        <p>Already have an account.</p>

        <a href="{% url 'login' %}"

          ><button class="inp btn-log">

            <i class="fa fa-sign-in"></i> Log In

          </button></a

        >

      </div>

    </div>

    {% if error == "no" %}

    <script>

      alert("Signup Successfull")

      window.location = ('{% url 'login' %}')

    </script>

    {% endif %} {% if error == "yes" %}

    <script>

      alert("Somthing Wrong, Try Again...");

    </script>

    {% endif %}

  </body>

</html>

user\_home.html: -

{% load static %} {% include 'user\_navigation.html' %}

<link rel="stylesheet" href="{% static 'css/userhome.css' %}" />

<div class="main-user-view">

  <div class="left-user-main">

    <div class="search">

      <span>Filter</span>

      <i class="fa fa-filter"></i>

      <input class="srch" type="search" name="" id="" />

      <div class="ask-btn">

        <a href="{% url 'add\_que\_user' %}"><button>Ask a Question</button></a>

      </div>

    </div>

    {% for i, j in lst %}

    <div class="usr-ques">

      <div class="ques-wrapper">

        <div class="quess">{{ i.que }}</div>

      </div>

      <div class="ans-btn">

        <div class="ques-askes-inf">

          <div class="ans-usr-img">

            {% for k in all\_usr %} {% if k.user.id == i.user\_id %}

            <img src="{{ MEDIA\_URL }}/media/{{ k.usr\_img }}" alt="" />

            {% endif %} {% endfor %}

          </div>

          <div class="ans-usr-name">{{ i.user }}</div>

          <div class="follow">

            <a href="{% url 'follow\_inc' i.user\_id %}"

              ><i class="fa fa-user-plus"></i> FOLLOW</a

            >

          </div>

        </div>

        <a href="{% url 'add\_ans\_user' i.id%}"><button>Answer</button></a>

      </div>

    </div>

    {% endfor %}

  </div>

  <div class="right-user-main">

    <div class="usr-img">

      <img src="{{ MEDIA\_URL }}/media/{{ u\_data.usr\_img }}" alt="" />

    </div>

    <div class="follow-container">

      <div class="box">

        <p>{{ u\_data.follower }} Followers</p>

      </div>

      <div class="box">

        <p>{{ u\_data.following }} Followings</p>

      </div>

    </div>

    <div class="asked">

      <p>Ques. Answered : {{ u\_data.answered}}</p>

      <p>Questions Asked : {{ u\_data.que\_asked }}</p>

    </div>

  </div>

</div>

{% include 'footer.html' %}

User\_navigation.html:-

<!DOCTYPE html>

<html>

  <head>

    {% load static %}

    <title>Study Material</title>

    <meta charset="utf-8" />

    <meta name="viewport" content="width=device-width, initial-scale=1" />

    <script

      src="https://kit.fontawesome.com/0da782d01f.js"

      crossorigin="anonymous"

    ></script>

    <script

      src="https://cdn.tiny.cloud/1/no-api-key/tinymce/6/tinymce.min.js"

      referrerpolicy="origin"

    ></script>

    <link

      rel="stylesheet"

      href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css"

    />

    <link rel="stylesheet" href="{% static 'css/userhome.css' %}" />

    <script>

      tinymce.init({

        selector: "#mytextarea",

        width: 1000,

        height: 600,

      });

    </script>

    <script>

      tinymce.init({

        selector: "#mytex",

      });

    </script>

  </head>

  <body>

    <div class="cross menu--1">

      <label>

        <input

          class="ac"

          onclick="func()"

          type="checkbox"

          style="display: none"

        />

        <svg viewBox="0 0 100 100" xmlns="http://www.w3.org/2000/svg">

          <circle cx="50" cy="50" r="30" />

          <path class="line--1" d="M0 40h62c13 0 6 28-4 18L35 35" />

          <path class="line--2" d="M0 50h70" />

          <path class="line--3" d="M0 60h62c13 0 6-28-4-18L35 65" />

        </svg>

      </label>

      <div class="menu" id="m">

        <ul class="lst">

          <a href="{% url 'user\_home' %}"><li>Home</li></a>

          <a href="{% url 'view\_que\_user' %}"><li>view Questions</li></a>

          <a href="{% url 'view\_blogs' %}"><li>view blog</li></a>

          <a href="{% url 'write\_blog' %}"><li>write blog</li></a>

          <a href="{% url 'feedback' %}"><li>Feedback</li></a>

        </ul>

      </div>

    </div>

    <header>

      <a href="{% url 'user\_home' %}"><div class="logo">Q n A</div></a>

      <div class="log-out">

        <p>

          Hello,

          <span style="font-weight: bold"

            >{{ user.first\_name }} {{ user.last\_name }}</span

          >

        </p>

        <a href="{% url 'logout' %}"

          ><p class="lout">Log out <i class="fa fa-sign-out"></i></p

        ></a>

      </div>

    </header>

    <script>

      function func() {

        let toggle = document.querySelector(".ac");

        let menu = document.querySelector("#m");

        let lii = document.querySelector(".lst");

        if (toggle.checked == true) {

          menu.style.width = "100vw";

          lii.style.display = "block";

        } else {

          menu.style.width = "0";

          lii.style.display = "none";

        }

      }

    </script>

  </body>

</html>

View\_Blog\_admin.html: -

<html>

  <head>

    {% load static %}

    <title>Study Material</title>

    <meta charset="utf-8" />

    <meta name="viewport" content="width=device-width, initial-scale=1" />

    <link

      rel="stylesheet"

      href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css"

    />

    <link rel="stylesheet" href="{% static 'css/admin\_home.css' %}" />

    <style>

      .active::before {

        content: "";

        height: 100%;

        width: 150%;

        position: absolute;

        background-color: #f7f7f7;

        left: -170px;

        top: 0;

      }

      .card-cont {

        width: 90%;

        text-align: justify;

        margin-bottom: 50px;

      }

      .card {

        display: flex;

        flex-direction: column;

      }

      .topic {

        font-size: 1.5rem;

        font-weight: bold;

      }

      .del {

        display: flex;

        justify-content: flex-end;

        margin-top: 20px;

      }

      .del > a > button {

        padding: 15px;

        border: none;

        border-radius: 3px;

        color: #f7f7f7;

        cursor: pointer;

      }

    </style>

  </head>

  <body>

    <div class="main-container">

      <div class="body-container">

        <a href="{% url 'index' %}"><p class="logo">Q n A</p></a>

        <div class="all-content">

          {% for i in data %}

          <div class="card-cont">

            <div class="card">

              <div class="blog-cont">

                <div class="topic">{{ i.topic }}</div>

              </div>

            </div>

            <div class="del">

              <a href="{% url 'blog' i.id %}"

                ><button style="background: #626ee3; margin-right: 5px">

                  view Blog

                </button></a

              >

              <a href="{% url 'delete\_blog' i.id %}"

                ><button style="background: #eb5353">Delete Blog</button></a

              >

            </div>

          </div>

          {% endfor %}

        </div>

      </div>

      <div class="menu-admin">

        <div class="admin-navigation">

          <p>

            Welcome,

            <span style="font-weight: bold; font-size: 1.3rem">

              {{ user }}</span

            >

          </p>

          <ul class="admin-nav">

            <a href="{% url 'admin\_home' %}"

              ><li class="nav-items-admin">Home</li></a

            >

            <a href="{% url 'view\_user' %}"

              ><li class="nav-items-admin">View Users</li></a

            >

            <a href=""><li class="active">View Blogs</li></a>

            <a href="{% url 'change\_password' %}"

              ><li class="nav-items-admin">Change Password</li></a

            >

            <a href=""><li class="nav-items-admin">View Feedback</li></a>

          </ul>

          <a href="{% url 'logout' %}"

            ><button class="buttn">Log Out</button></a

          >

        </div>

      </div>

    </div>

  </body>

</html>

View\_Blogs.html: -

{% load static %} {% include 'user\_navigation.html' %}

<link rel="stylesheet" href="{% static 'css/view\_blogs.css' %}" />

{% for i in data %}

<div class="main-cont">

  <div class="card-cont">

    <div class="card">

      <div class="img-sec">

        {% for j in usr\_data %} {% if j.user.id == i.user\_id %}

        <div class="img">

          <img src="{{ MEDIA\_URL }}/media/{{ j.usr\_img }}" alt="" />

        </div>

        <p style="margin-top: 5px; color: white">

          {{ j.user.first\_name }} {{ j.user.last\_name }}

        </p>

        {% endif %} {% endfor %}

      </div>

      <div class="blog-cont">

        <a href="{% url 'blog' i.id %}"

          ><div class="topic">{{ i.topic }}</div></a

        >

        <div class="blog">{{ i.blog\_c|safe }}</div>

        <a href="{% url 'blog' i.id %}"

          ><div class="view-more">view more</div></a

        >

      </div>

    </div>

  </div>

</div>

{% endfor %}

{% include 'footer.html' %}

View\_Feedback.html: -

<html>

  <head>

    {% load static %}

    <title>Study Material</title>

    <meta charset="utf-8" />

    <meta name="viewport" content="width=device-width, initial-scale=1" />

    <link

      rel="stylesheet"

      href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css"

    />

    <link rel="stylesheet" href="{% static 'css/admin\_home.css' %}" />

    <style>

      .all-content {

        margin-left: 80;

        margin-right: 20px;

      }

      .active::before {

        content: "";

        height: 100%;

        width: 150%;

        position: absolute;

        background-color: #f7f7f7;

        left: -170px;

        top: 0;

      }

      .feed {

        display: flex;

        padding: 20px;

        border: 1px solid #626ee3;

        width: 80%;

        margin: 0 auto 20px auto;

      }

      .info-feed {

        display: flex;

        flex-direction: column;

        width: 30%;

        text-align: justify;

        justify-content: center;

        margin-top: 8px;

        /\* align-items: center; \*/

      }

      .feed-cont {

        display: flex;

        justify-content: center;

        width: 70%;

        font-size: 1.3rem;

        font-weight: bold;

        /\* margin-left: 30px; \*/

        text-align: justify;

        /\* align-items: center; \*/

        /\* padding: 40px; \*/

      }

      .info-feed > a > button {

        width: 40%;

        padding: 15px;

        font-size: 1.2rem;

        background: #eb5353;

        color: white;

        border: none;

        margin-top: 30px;

        border-radius: 3px;

        cursor: pointer;

      }

    </style>

  </head>

  <body>

    <div class="main-container">

      <div class="body-container">

        <a href="{% url 'index' %}"><p class="logo">Q n A</p></a>

        <div class="all-content">

          {% for i in data %}

          <div class="feed">

            <div class="info-feed">

              <div>

                <p>NAME: {{ i.feedback\_name }}</p>

                <p>email: {{ i.feedback\_email }}</p>

                <p>contact: {{ i.feedback\_contact }}</p>

              </div>

              <a

                href="{% url 'delete\_feedback' i.id %}"

                onclick="return confirm('Are you Sure?')"

                ><button>Delete</button></a

              >

            </div>

            <div class="feed-cont">{{ i.feedback\_comment|safe }}</div>

          </div>

          {% endfor %}

        </div>

      </div>

      <div class="menu-admin">

        <div class="admin-navigation">

          <p>

            Welcome,

            <span style="font-weight: bold; font-size: 1.3rem">

              {{ user }}</span

            >

          </p>

          <ul class="admin-nav">

            <a href="{% url 'admin\_home' %}"><li class="">Home</li></a>

            <a href="{% url 'view\_user' %}"><li class="">View Users</li></a>

            <a href="{% url 'view\_blog\_admin' %}"

              ><li class="">View Blogs</li></a

            >

            <a href="{% url 'change\_password' %}"

              ><li class="">Change Password</li></a

            >

            <a href=""><li class="active">View Feedback</li></a>

          </ul>

          <a href="{% url 'logout' %}"

            ><button class="buttn">Log Out</button></a

          >

        </div>

      </div>

    </div>

  </body>

</html>

View\_question\_user.html: -

{% load static %}

{% include 'user\_navigation.html' %}

<link rel="stylesheet" href="{% static 'css/view\_ques\_user.css' %}">

{% for i, j in lst %}

<div class="main-cont">

    <div class="ques-cont">

        <div class="img-asked-by">

            {% for m in usr\_data %}

            {% if m.user.id == i.user\_id %}

            <img src="{{ MEDIA\_URL }}/media/{{ m.usr\_img }}" alt="">

            {% endif %}

            {% endfor %}

        </div>

        <div class="ques">

            <p>{{ i.que }}</p>

            <div class="cat">Category : {{ i.topic }}</div>

    </div>

        </div>

    <div class="ans-cont">

        {% for k in data %}

        {% if k.atopic == j %}

        <div class="img-flo">

            <div class="ans-usr">

                {% for m in usr\_data %}

                {% if m.user.id == k.user\_id %}

                <img src="{{ MEDIA\_URL }}/media/{{ m.usr\_img }}" alt="">

                <p>{{ m.user.first\_name }} {{ m.user.last\_name }}</p>

                {% endif %}

                {% endfor %}

            </div>

            <a href="{% url 'follow\_inc' k.user\_id %}"><i class="fa fa-user-plus"></i></a>

        </div>

        <div class="ans">

            <p>{{ k.ans|safe }}</p>

            <div class="like">

                <a href="{% url 'add' k.id %}"><i class="fa fa-thumbs-up"></i><span> {{ k.h }}</span></a>

                <a href="{% url 'addd' k.id %}"><i class="fa fa-thumbs-down mar-like"></i><span> {{ k.nh }}</a>

            </div>

        </div>

        {% endif %}

        {% endfor %}

    </div>

</div>

{% endfor %}

{% include 'footer.html' %}

View\_User.html: -

<html>

  <head>

    {% load static %}

    <title>Study Material</title>

    <meta charset="utf-8" />

    <meta name="viewport" content="width=device-width, initial-scale=1" />

    <link

      rel="stylesheet"

      href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css"

    />

    <link rel="stylesheet" href="{% static 'css/admin\_home.css' %}" />

    <style>

      .all-content {

        margin-left: 80;

        margin-right: 20px;

      }

      #example {

        border-collapse: collapse;

      }

      #example th {

        padding: 20px;

        text-align: left;

        color: white;

      }

      #example tr:nth-child(even) {

        background-color: #f2f2f2;

      }

      #example td,

      #example th {

        border: 1px solid #ddd;

        padding: 15px 10px 15px 10px;

      }

      .active::before {

        content: "";

        height: 100%;

        width: 150%;

        position: absolute;

        background-color: #f7f7f7;

        left: -170px;

        top: 0;

      }

    </style>

  </head>

  <body>

    <div class="main-container">

      <div class="body-container">

        <a href="{% url 'index' %}"><p class="logo">Q n A</p></a>

        <div class="all-content">

          <table class="" id="example">

            <thead>

              <tr style="background-color: #626ee3; padding: 10px">

                <th>S. No.</th>

                <th>Full Name</th>

                <th>Email ID</th>

                <th>Contact</th>

                <th>Gender</th>

                <th>DOB</th>

                <th>Action</th>

              </tr>

            </thead>

            <tbody>

              {% for i in data %}

              <tr>

                <td>{{forloop.counter}}</td>

                <td>{{i.user.first\_name}} {{i.user.last\_name}}</td>

                <td>{{i.user.username}}</td>

                <td>{{i.mobile}}</td>

                <td>{{i.gender}}</td>

                <td>{{i.dob}}</td>

                <a

                  href="{% url 'delete\_user' i.user.id %}"

                  onclick="return confirm('Are you Sure?')"

                  ><td

                    style="

                      background-color: #eb5353;

                      color: white;

                      cursor: pointer;

                    "

                  >

                    Delete

                  </td></a

                >

              </tr>

              {% endfor %}

            </tbody>

          </table>

        </div>

      </div>

      <div class="menu-admin">

        <div class="admin-navigation">

          <p>

            Welcome,

            <span style="font-weight: bold; font-size: 1.3rem">

              {{ user }}</span

            >

          </p>

          <ul class="admin-nav">

            <a href="{% url 'admin\_home' %}"><li class="">Home</li></a>

            <a href="{% url 'view\_user' %}">

<li class="active" style="color: black">View Users</li></a>

            <a href="{% url 'view\_blog\_admin' %}">

<li class="">View Blogs</li></a>

            <a href="{% url 'change\_password' %}"

              ><li class="">Change Password</li></a

            >

            <a href="{% url 'view\_feedback' %}"

              ><li class="">View Feedback</li></a

            >

          </ul>

          <a href="{% url 'logout' %}"

            ><button class="buttn">Log Out</button></a

          >

        </div>

      </div>

    </div>

  </body>

</html>

Write\_blog.html: -

{% include 'user\_navigation.html' %}

<style>

  .frm {

    display: flex;

    width: 100%;

    margin: 0 auto;

    flex-direction: column;

    justify-content: center;

    align-items: center;

    margin-top: 40px;

  }

  .inp {

    border: 2px solid #3a5ba0;

    appearance: none;

    padding: 12px;

    width: 450px;

    border-radius: 3px;

    background: #f2f2f2;

    outline: none;

    display: block;

    z-index: 99;

    margin-bottom: 50px;

  }

  button {

    padding: 30px;

    background-color: #626ee3;

    color: white;

    border: none;

    font-size: 1.5rem;

    font-weight: bold;

    margin: 40px 0;

    border-radius: 5px;

    cursor: pointer;

  }

</style>

<form class="frm" method="post">

  {% csrf\_token %}

  <input placeholder="Blog Topic Name" class="inp" type="text" name="btopic" />

  <div style="border: 1px solid #626ee3">

    <textarea name="blog" id="mytextarea" cols="120" rows="10"></textarea>

  </div>

  <button>Submit</button>

</form>

**Database Models code: -**

from email.policy import default

from django.db import models

from django.contrib.auth.models import User

from ckeditor.fields import RichTextField

class Signup(models.Model):

    user = models.ForeignKey(User, on\_delete=models.CASCADE)

    mobile = models.CharField(max\_length=20, null=True)

    gender = models.CharField(max\_length=10, null=True)

    dob = models.CharField(max\_length=10, null=True)

    follower = models.IntegerField(max\_length=200, null=True)

    following = models.IntegerField(max\_length=200, null=True)

    que\_asked = models.IntegerField(max\_length=100, null=True)

    answered = models.IntegerField(max\_length=100, null=True)

    usr\_img = models.ImageField(upload\_to="img/", default="")

    def \_str\_(self):

        return self.user.username

class Questions(models.Model):

    id = models.AutoField

    que = RichTextField(blank=True, null=True)

    topic = models.CharField(max\_length=100, null=True)

    ans = RichTextField(blank=True, null=True)

    user = models.CharField(max\_length=50, null=True)

    user\_id = models.IntegerField(max\_length=100, null=True)

    atopic = models.CharField(max\_length=100, null=True)

    h = models.IntegerField(max\_length=200, null=True)

    nh = models.IntegerField(max\_length=200, null=True)

    def \_\_str\_\_(self):

        return str(self.id)

class Feedback(models.Model):

    feedback\_name = models.CharField(max\_length=20, null=True)

    feedback\_contact = models.CharField(max\_length=30, null=True)

    feedback\_email = models.CharField(max\_length=10, null=True)

    feedback\_comment = RichTextField(blank=True, null=True)

class Helpful(models.Model):

    num = models.IntegerField(max\_length=200, null=True)

    num2 = models.IntegerField(max\_length=200, null=True)

    numid = models.IntegerField(max\_length=200, null=True)

class Blog(models.Model):

    id = models.AutoField

    user\_id = models.IntegerField(default=1, max\_length=200, null=True)

    user\_name = models.CharField(max\_length=200, null=True)

    topic = models.CharField(max\_length=200, null=True)

    blog\_c = RichTextField(blank=True, null=True)

**Styling: -**

Admin\_home.css: -

@import url("https://fonts.googleapis.com/css2?family=Fascinate&family=Poppins:wght@100;200;300;400;500;600;700;800;900&family=Ubuntu:wght@300;400;500;700&display=swap");

\* {

  margin: 0;

  padding: 0;

  box-sizing: border-box;

}

\*::after,

\*::before {

  box-sizing: border-box;

}

@font-face {

  font-family: gilroy;

  src: url(../font/Gilroy-ExtraBold.otf);

}

body {

  background-color: #f7f7f7;

  font-family: "Poppins";

}

:root {

  --primary: #626ee3;

}

img {

  max-width: 100%;

  max-height: 100%;

}

a {

  text-decoration: none;

  color: inherit;

}

ul {

  list-style: none;

}

.main-container {

  display: flex;

}

.all-content {

  margin-top: 150px;

  margin-left: 180px;

  margin-right: 40px;

}

.body-container {

  width: 80%;

}

.menu-admin {

  width: 30%;

}

.admin-navigation {

  position: sticky;

  width: 300px;

  height: 100vh;

  /\* max-height: 100vh; \*/

  top: 0;

  background: var(--primary);

  left: 100%;

  display: flex;

  align-self: flex-start;

  flex-direction: column;

  justify-content: space-between;

  align-items: center;

  padding: 25px 0;

  z-index: 99;

}

.admin-navigation > p {

  color: white;

}

.admin-nav {

  display: flex;

  justify-content: center;

  align-items: flex-end;

  flex-direction: column;

}

.admin-nav > a > li {

  padding: 12px;

  color: #f7f7f7;

  font-size: 1.2rem;

  font-family: "Ubuntu";

  cursor: pointer;

}

.active {

  background-color: #f7f7f7;

  color: black !important;

  position: relative;

}

.active::before {

  content: "";

  height: 100%;

  width: 230%;

  position: absolute;

  background-color: #f7f7f7;

  left: -170px;

  top: 0;

}

.buttn {

  padding: 13px;

  color: #626ee3;

  background-color: white;

  border: none;

  font-size: 1.5rem;

  font-weight: bold;

  border-radius: 5px;

  cursor: pointer;

}

.logo {

  font-family: "Fascinate", cursive;

  font-size: 3rem;

  font-weight: bolder;

  color: var(--primary);

  padding: 20px 0 0 60px;

  position: absolute;

  left: 0;

  top: 0;

}

.que-cont {

  display: flex;

  flex-direction: column;

  margin-top: 100px;

  margin-bottom: 50px;

}

.que {

  font-size: 1.3rem;

  font-weight: bold;

  text-align: justify;

  display: flex;

}

.asker {

  display: flex;

  justify-content: space-between;

  align-items: center;

  margin-top: 5px;

}

.info {

  display: flex;

}

.img-usr {

  width: 30px;

  height: 30px;

}

.ans-cont {

  margin: 30px 0 50px 70px;

  display: flex;

}

.anss-img {

  width: 30px;

  height: 30px;

  margin-right: 10px;

}

.del-que > a > button {

  padding: 15px;

  border: none;

  background: #eb5353;

  color: white;

  border-radius: 3px;

  font-size: 1rem;

  cursor: pointer;

}

.ans {

  width: 100%;

  text-align: justify;

}

.strip {

  display: flex;

  justify-content: space-between;

  align-items: center;

}

.but {

  padding: 10px;

  font-size: 0.9rem;

  color: white;

  border-radius: 3px;

  border: none;

  cursor: pointer;

}

Blog.css: -

@import url("https://fonts.googleapis.com/css2?family=Fascinate&family=Poppins:wght@100;200;300;400;500;600;700;800;900&family=Ubuntu:wght@300;400;500;700&display=swap");

\* {

  margin: 0;

  padding: 0;

  box-sizing: border-box;

}

\*::after,

\*::before {

  box-sizing: border-box;

}

@font-face {

  font-family: gilroy;

  src: url(../font/Gilroy-ExtraBold.otf);

}

body {

  background-color: #f7f7f7;

  font-family: "Poppins";

  overflow-x: hidden;

}

:root {

  --primary: #626ee3;

}

img {

  max-width: 100%;

  max-height: 100%;

}

a {

  text-decoration: none;

  color: inherit;

}

ul {

  list-style: none;

}

.banner {

  width: 100%;

  height: 400px;

  display: flex;

  justify-content: center;

  align-items: center;

  background: var(--primary);

  margin-top: 20px;

}

.banner > h1 {

  font-size: 3rem;

  color: #f7f7f7;

  width: 70%;

  text-align: center;

}

.img-cont {

  width: 150px;

  height: 150px;

  background-color: white;

  border-radius: 50%;

  position: relative;

  display: flex;

  align-items: center;

  justify-content: center;

  overflow: hidden;

  top: -70px;

  left: 50%;

  transform: translate(-50%);

}

.blog-cont {

  width: 60%;

  margin: 0 auto;

  text-align: justify;

}

.blog-cont > p {

  margin-bottom: 50px;

}

.u\_name {

  text-align: center;

  margin-top: -60px;

  margin-bottom: 50px;

  font-weight: bold;

  font-size: 1.5rem;

}

Feedback.css: -

@import url("https://fonts.googleapis.com/css2?family=Fascinate&family=Poppins:wght@100;200;300;400;500;600;700;800;900&family=Ubuntu:wght@300;400;500;700&display=swap");

\* {

  margin: 0;

  padding: 0;

  box-sizing: border-box;

}

\*::after,

\*::before {

  box-sizing: border-box;

}

@font-face {

  font-family: gilroy;

  src: url(../font/Gilroy-ExtraBold.otf);

}

body {

  background-color: #f7f7f7;

  font-family: "Poppins";

  overflow-x: hidden;

}

:root {

  --primary: #626ee3;

}

img {

  max-width: 100%;

  max-height: 100%;

}

a {

  text-decoration: none;

  color: inherit;

}

ul {

  list-style: none;

}

.logo {

  font-family: "Fascinate", cursive;

  font-size: 3rem;

  font-weight: bolder;

  color: var(--primary);

  padding: 20px 0px 0 50px;

}

form {

  width: 80%;

  margin: 40px auto 0 auto;

}

.info {

  display: flex;

  justify-content: center;

}

.info > div {

  display: flex;

  flex-direction: column;

  margin-right: 50px;

}

.info > div > input {

  padding: 10px;

  width: 300px;

  margin-bottom: 20px;

  border: 2px solid var(--primary);

  border-radius: 3px;

}

.btn {

  width: 20%;

  padding: 15px;

  margin-top: 20px;

  border: none;

  background: var(--primary);

  color: white;

  cursor: pointer;

}

Main.css: -

@import url("https://fonts.googleapis.com/css2?family=Fascinate&family=Poppins:wght@100;200;300;400;500;600;700;800;900&family=Ubuntu:wght@300;400;500;700&display=swap");

\* {

  margin: 0;

  padding: 0;

  box-sizing: border-box;

}

\*::after,

\*::before {

  box-sizing: border-box;

}

@font-face {

  font-family: gilroy;

  src: url(../font/Gilroy-ExtraBold.otf);

}

body {

  background-color: #f7f7f7;

  font-family: "Poppins";

  overflow-x: hidden;

}

:root {

  --primary: #626ee3;

}

img {

  max-width: 100%;

  max-height: 100%;

}

a {

  text-decoration: none;

  color: inherit;

}

ul {

  list-style: none;

}

header {

  display: flex;

  justify-content: space-between;

  align-items: center;

  height: 5rem;

}

/\* nav{

  display: flex;

  height: 100%;

  justify-content: center;

  align-items: center;

} \*/

.logo {

  font-family: "Fascinate", cursive;

  font-size: 3rem;

  font-weight: bolder;

  color: var(--primary);

  padding: 50px;

}

.log-btn {

  border: none;

  height: 100%;

  font-family: "Ubuntu", sans-serif;

  font-size: 1rem;

  color: white;

  padding: 0 20px 0 20px;

  background-color: var(--primary);

  letter-spacing: 0.1rem;

  cursor: pointer;

}

/\* .nav-bar{

  display: flex;

  height: 100%;

  justify-content: center;

  align-items: center;

  border-top: 2px solid var(--primary);

}

.nav-bar > li{

  margin: 0 10px;

  font-size: 1.2rem;

  color: var(--primary);

  font-family: 'Ubuntu', sans-serif;

  font-weight: 700;

} \*/

.main-container {

  display: flex;

}

.main-left {

  margin-left: 180px;

  margin-top: 150px;

}

.left-container {

  font-family: "Poppins", sans-serif;

}

.ttl {

  font-size: 3rem;

  font-weight: bolder;

}

/\* ----------------word slider ------------\*/

.word-slider {

  /\* display: inline-block; \*/

  height: 100px;

  overflow: hidden;

}

.words {

  display: inline-block;

  animation: wordSlider 10s ease-out infinite;

  padding: 0 !important;

  margin: 0;

  margin-block-start: 0;

  margin-block-end: 0;

  padding-inline-start: 0;

}

.word {

  font-size: 5rem;

  font-weight: bolder;

  color: var(--primary);

}

@keyframes wordSlider {

  0%,

  12.66%,

  100% {

    transform: translate3d(0, 0, 0);

  }

  16.66%,

  29.32% {

    transform: translate3d(0, -25%, 0);

  }

  33.32%,

  45.98% {

    transform: translate3d(0, -50%, 0);

  }

  49.98%,

  62.64% {

    transform: translate3d(0, -75%, 0);

  }

  66.64%,

  79.3% {

    transform: translate3d(0, -50%, 0);

  }

  83.3%,

  95.96% {

    transform: translate3d(0, -25%, 0);

  }

}

.circle {

  width: 100px;

  height: 100px;

  border: 2px solid var(--primary);

  border-radius: 50%;

}

.square {

  margin-left: 50px;

  margin-top: 10px;

}

.s1 {

  width: 60px;

  height: 60px;

  position: relative;

  left: 100px;

  transform: rotate(30deg);

  border: 2px solid var(--primary);

  animation: up1 1s infinite;

}

.s2 {

  width: 100px;

  height: 100px;

  position: relative;

  left: 200px;

  transform: rotate(-30deg);

  border: 4px solid var(--primary);

  animation: up 1s infinite;

}

.s3 {

  width: 30px;

  height: 30px;

  position: relative;

  left: 10px;

  transform: rotate(100deg);

  border: 3px solid var(--primary);

  animation: up 1s infinite;

}

.s4 {

  width: 20px;

  height: 20px;

  position: relative;

  left: 300px;

  transform: rotate(-100deg);

  border: 5px solid var(--primary);

  animation: up1 1s infinite;

}

@keyframes up {

  0% {

    transform: translateY(0px);

  }

  /\* 25%{transform: translateY(-5px);} \*/

  50% {

    transform: translateY(-10px);

  }

  /\* 75%{transform: translateY(5px);} \*/

  100% {

    transform: translateY(0px);

  }

}

@keyframes up1 {

  0% {

    transform: translateY(0px);

  }

  /\* 25%{transform: translateY(-5px);} \*/

  50% {

    transform: translateY(-5px);

    transform: rotate(180deg);

  }

  /\* 75%{transform: translateY(5px);} \*/

  100% {

    transform: translateY(0px);

  }

}

.form-container {

  width: 80vw;

  height: 80vh;

  margin: 30px auto 0 auto;

  display: flex;

}

.left-form {

  width: 50%;

  background-color: var(--primary);

  display: flex;

  justify-content: center;

  align-items: center;

}

.right-form {

  width: 50%;

  display: flex;

  flex-direction: column;

  justify-content: center;

  align-items: center;

  border: 2px solid var(--primary);

}

.for {

  display: flex;

  flex-direction: column;

  align-items: center;

}

.login-txt {

  font-size: 2.5rem;

  font-weight: bolder;

  color: var(--primary);

}

.cont-form {

  width: 70%;

}

.cont-form > p {

  font-family: "Poppins";

  color: white;

  font-weight: 300;

  font-size: 1rem;

  text-align: justify;

}

.custom-field {

  position: relative;

  font-size: 14px;

  padding-top: 20px;

  margin-bottom: 5px;

}

.custom-field > input {

  border: 2px solid #3a5ba0;

  appearance: none;

  padding: 12px;

  width: 250px;

  border-radius: 3px;

  background: #f2f2f2;

  outline: none;

  display: block;

}

.custom-field .placeholder {

  position: absolute;

  left: 12px;

  top: calc(50% + 10px);

  transform: translateY(-50%);

  color: #aaa;

  font-family: "Poppins";

  transition: top 0.3s ease, font-size 0.3s ease, color 0.3s ease;

}

.custom-field input:valid + .placeholder,

.custom-field input:focus + .placeholder {

  top: 20px;

  font-size: 14px;

  color: #222;

  background: #f7f7f7;

  padding: 4px;

}

.custom-field.one input:valid,

.custom-field.one input:focus {

  border-color: var(--primary);

  transition-delay: 0.1s;

}

.inp {

  background: var(--primary);

  border: none;

  font-size: 1rem;

  color: white;

  margin-top: 8px;

  cursor: pointer;

  appearance: none;

  padding: 12px;

  width: 250px;

  border-radius: 3px;

  outline: none;

  display: block;

}

.sign-container {

  width: 80vw;

  height: 85vh;

  margin: 10px auto 0 auto;

  display: flex;

}

.left-sign {

  width: 80%;

  height: 100%;

  display: flex;

  justify-content: center;

  flex-direction: column;

  align-items: center;

  border: 2px solid var(--primary);

}

.left-sign > p {

  font-size: 2rem;

  font-weight: bolder;

  color: var(--primary);

}

.line {

  width: 40%;

  height: 5px;

  background: var(--primary);

}

.right-sign {

  width: 40%;

  height: 100%;

  background: var(--primary);

  display: flex;

  flex-direction: column;

  justify-content: center;

  align-items: center;

}

.right-sign > p {

  font-size: 2rem;

  color: white;

  width: 75%;

}

.sign-form {

  width: 80%;

  margin-top: 20px;

}

.form-outer {

  display: flex;

}

.left-c1 {

  width: 50%;

  display: flex;

  flex-direction: column;

  justify-content: center;

  align-items: center;

}

.left-c2 {

  width: 50%;

  display: flex;

  flex-direction: column;

  justify-content: center;

  align-items: center;

  margin-top: -15px;

}

.rad {

  margin-top: 20px;

  padding-top: 10px;

}

.sgn-btn {

  margin: 30px auto 0 auto;

}

.btn-log {

  background: white;

  color: var(--primary);

}

.ntxt {

  text-align: center;

  font-family: "Ubuntu";

  font-size: 2.5rem;

  /\* color: var(--primary); \*/

  margin-top: 150px;

}

.line2 {

  width: 80%;

  height: 5px;

  margin: 15px auto 10px auto;

  background-color: var(--primary);

}

.questions {

  width: 60%;

  /\* height: 300px; \*/

  margin: 50px auto 10px auto;

  display: flex;

  border-bottom: 1px solid var(--primary);

}

.usr-details {

  width: 30%;

  height: 100%;

  /\* border: 2px solid var(--primary); \*/

  display: flex;

  flex-direction: column;

  justify-content: center;

  align-items: center;

  margin-top: 60px;

}

.ques-details {

  width: 70%;

  height: 100%;

  /\* border: 2px solid var(--primary);  \*/

  display: flex;

  flex-direction: column;

  padding: 20px 30px;

}

.img-usr {

  width: 100px;

  height: 100px;

  /\* display: flex; \*/

  /\* overflow: hidden; \*/

  /\* justify-content: center;

  align-items: center; \*/

}

.date-det-con {

  display: flex;

  font-size: 15px;

}

.date {

  color: #aaa;

}

.cate {

  margin-left: 50px;

}

.question {

  font-size: 1.8rem;

  font-weight: bold;

  margin: 10px 0;

}

.answers {

  padding-left: 30px;

  margin-bottom: 8px;

}

.ans-usr {

  display: flex;

  /\* justify-content: center; \*/

  align-items: center;

}

.ans-usr-img {

  width: 30px;

  height: 30px;

  border-radius: 50%;

  /\* background: var(--primary); \*/

  overflow: hidden;

  display: flex;

  justify-content: center;

  align-items: center;

}

.ans-usr-name {

  margin-left: 10px;

  font-family: "Ubuntu";

  font-size: 14px;

  font-weight: bold;

  text-decoration: underline;

}

.usr-ans {

  margin-top: 5px;

  padding-left: 39px;

  padding-right: 20px;

  text-align: justify;

}

.ans-strip {

  width: 100%;

  height: 50px;

  background: rgba(145, 9, 30, 0.38);

  margin: 30px 0;

  display: flex;

  justify-content: space-between;

  align-items: center;

}

.thum {

  color: #91091e;

  font-size: 1.3rem;

  margin: 0 20px;

  cursor: pointer;

}

.ask {

  color: #91091e;

  font-size: 1rem;

  text-decoration: underline;

  margin-right: 20px;

  cursor: pointer;

}

User\_home.css: -

@import url("https://fonts.googleapis.com/css2?family=Fascinate&family=Poppins:wght@100;200;300;400;500;600;700;800;900&family=Ubuntu:wght@300;400;500;700&display=swap");

\* {

  margin: 0;

  box-sizing: border-box;

}

body {

  background-color: #f7f7f7;

  font-family: "Poppins";

  overflow-x: hidden;

}

:root {

  --primary: #626ee3;

}

img {

  max-width: 100%;

  max-height: 100%;

}

a {

  text-decoration: none;

  color: inherit;

}

ul {

  list-style: none;

}

.menu--1 > label {

  display: block;

  cursor: pointer;

  position: absolute;

  left: 0;

  top: 0;

  width: 100px;

  height: 100px;

  background-color: #f7f7f7;

  max-width: 150px;

  max-height: 150px;

}

path {

  fill: none;

  stroke: var(--primary);

  stroke-width: 3;

  stroke-linecap: round;

  stroke-linejoin: round;

  --length: 24;

  --offset: -38;

  stroke-dasharray: var(--length) var(--total-length);

  stroke-dashoffset: var(--offset);

  transition: all 0.8s cubic-bezier(0.645, 0.045, 0.355, 1);

}

circle {

  fill: #fff3;

  opacity: 0;

}

label:hover circle {

  opacity: 1;

}

.cross input:checked + svg .line--1,

.cross input:checked + svg .line--3 {

  --length: 22.627416998;

}

.cross input:checked + svg .line--2 {

  --length: 0;

}

.menu--1 .line--1,

.menu--1 .line--3 {

  /\* --total-length: 126.64183044433594; \*/

  --total-length: 150;

}

.menu--1 .line--2 {

  /\* --total-length: 70; \*/

  --total-length: 100;

}

.menu--1 input:checked + svg .line--1,

.menu--1 input:checked + svg .line--3 {

  --offset: -94.1149185097;

}

.menu--1 input:checked + svg .line--2 {

  --offset: -50;

}

.logo {

  font-family: "Fascinate", cursive;

  font-size: 3rem;

  font-weight: bolder;

  color: var(--primary);

  padding-top: 17px;

  margin-left: 150px;

}

.menu {

  position: fixed;

  left: 100px;

  top: 0;

  width: 0;

  height: 100vh;

  background: var(--primary);

  display: flex;

  justify-content: center;

  align-items: center;

  transition: width 1s ease;

  z-index: 99;

}

.menu > ul > a > li {

  color: white;

  font-size: 2rem;

  font-family: "Ubuntu";

  font-weight: bolder;

  margin-bottom: 20px;

  cursor: pointer;

  transition: transform 0.5s ease;

}

.menu > ul > a > li:hover {

  transform: scale(1.5);

}

.mm:hover {

  transform: scale(1.5);

  cursor: pointer;

}

.lst {

  display: none;

  text-align: center;

  margin-right: 80px;

}

header {

  display: flex;

  justify-content: space-between;

  align-items: center;

}

.log-out {

  display: flex;

  align-items: center;

  margin-top: 25px;

  margin-right: 50px;

}

.log-out > p {

  padding-right: 20px;

}

.lout {

  color: var(--primary);

  font-weight: bold;

  font-size: 1.2rem;

  margin-left: 20px;

  cursor: pointer;

}

.main-user-view {

  width: 100%;

  display: flex;

  margin-top: 30px;

}

.left-user-main {

  width: 73%;

  display: flex;

  flex-direction: column;

  /\* justify-content: center; \*/

  align-items: center;

  margin-top: 50px;

  /\* background-color: red; \*/

}

.right-user-main {

  width: 27%;

  height: 500px;

  background: var(--primary);

  border-radius: 10px 0 0 10px;

  display: flex;

  justify-content: center;

  align-items: center;

  flex-direction: column;

}

.usr-img {

  width: 100px;

  height: 100px;

  /\* border-radius: 50%;

  background-color: white; \*/

  margin-top: 90px;

  display: flex;

  justify-content: center;

  /\* align-items: center; \*/

}

.usr-img > i {

  font-size: 5rem;

}

.follow-container {

  display: flex;

  width: 80%;

  justify-content: space-between;

  margin-right: 15px;

  padding-left: 15px;

}

.box {

  display: flex;

  color: #f7f7f7;

  margin-top: 20px;

}

.asked {

  color: #f7f7f7;

  font-size: 1.3rem;

  margin-top: 100px;

}

.asked > p {

  margin-bottom: 15px;

}

.srch {

  border: 2px solid #3a5ba0;

  appearance: none;

  padding: 6px;

  height: 35px;

  padding-left: 25px;

  width: 500px;

  border-radius: 3px;

  background: #f2f2f2;

  outline: none;

  display: block;

  box-shadow: 0px -1px 50px 2px rgba(98, 110, 227, 0.75);

}

.search {

  display: flex;

  align-items: center;

  justify-content: space-between;

}

.fa-filter {

  position: relative;

  left: 20px;

  top: 0px;

  color: #aaa;

}

.usr-ques {

  display: flex;

  width: 80%;

  padding: 20px 20px;

  margin-top: 50px;

  display: flex;

  flex-direction: column;

  /\* background: rgba(105, 48, 195, 0.38); \*/

  color: var(--primary);

  font-weight: bold;

  border-radius: 5px;

  border: 2px solid var(--primary);

}

.quess {

  font-family: "Ubuntu";

  font-size: 1.2rem;

}

.ques-wrapper {

  display: flex;

  flex-direction: column;

}

.ans-btn {

  /\* margin-left: 89%; \*/

  display: flex;

  justify-content: space-between;

  align-items: center;

  margin-top: 10px;

}

.ans-btn > a > button {

  background: none;

  border: 1px solid var(--primary);

  padding: 10px;

  border-radius: 2px;

  font-size: 1rem;

  color: #f7f7f7;

  background: var(--primary);

  cursor: pointer;

  transition: background-color 0.4s ease, color 0.4s ease;

  margin-top: 20px;

}

.ans-btn > a > button:hover {

  background: white;

  color: var(--primary);

}

.ques-askes-inf {

  display: flex;

  justify-content: center;

  align-items: center;

  margin-top: 25px;

}

.ans-usr-img {

  width: 30px;

  height: 30px;

  border-radius: 50%;

  /\* background: var(--primary); \*/

  border: 2px solid black;

  display: flex;

  justify-content: center;

  align-items: center;

  overflow: hidden;

}

.ans-usr-name {

  margin-left: 10px;

  font-family: "Ubuntu";

  font-size: 14px;

  color: black;

  font-weight: bold;

  text-decoration: underline;

}

.follow {

  margin-left: 20px;

  cursor: pointer;

  font-size: 0.9rem;

  margin-top: 4px;

}

.ask-btn {

  margin-left: 48px;

}

.ask-btn > a > button {

  background: none;

  background-color: var(--primary);

  border: none;

  padding: 20px;

  color: white;

  font-size: 1rem;

  border-radius: 3px;

  cursor: pointer;

}

View\_blogs.css: -

@import url("https://fonts.googleapis.com/css2?family=Fascinate&family=Poppins:wght@100;200;300;400;500;600;700;800;900&family=Ubuntu:wght@300;400;500;700&display=swap");

\* {

  margin: 0;

  padding: 0;

  box-sizing: border-box;

}

\*::after,

\*::before {

  box-sizing: border-box;

}

@font-face {

  font-family: gilroy;

  src: url(../font/Gilroy-ExtraBold.otf);

}

body {

  background-color: #f7f7f7;

  font-family: "Poppins";

  overflow-x: hidden;

}

:root {

  --primary: #626ee3;

}

img {

  max-width: 100%;

  max-height: 100%;

}

a {

  text-decoration: none;

  color: inherit;

}

ul {

  list-style: none;

}

.main-cont {

  display: flex;

  justify-content: center;

  align-items: center;

  width: 100%;

  margin: 0 auto;

}

.card-cont {

  display: flex;

  flex-direction: column;

  width: 60%;

  margin-top: 80px;

}

.card {

  width: 100%;

  background: var(--primary);

  display: flex;

}

.img-sec {

  display: flex;

  flex-direction: column;

  justify-content: center;

  align-items: center;

  width: 30%;

}

.img {

  width: 100px;

  height: 100px;

  display: flex;

  align-items: center;

  justify-content: center;

  /\* background-color: #F7F7F7; \*/

}

.topic {

  font-size: 1.5rem;

  font-weight: bold;

}

.blog-cont {

  width: 70%;

  display: flex;

  flex-direction: column;

  color: #f7f7f7;

  padding: 40px 50px 40px 0;

  padding-right: 50px;

  padding-top: 40px;

}

.blog {

  text-align: justify;

  font-size: 0.8rem;

  height: 180px;

  overflow: hidden;

  margin-top: 20px;

}

.view-more {

  display: flex;

  justify-content: flex-end;

  text-decoration: underline;

  margin-top: 3px;

}

View\_ques\_user.css: -

@import url("https://fonts.googleapis.com/css2?family=Fascinate&family=Poppins:wght@100;200;300;400;500;600;700;800;900&family=Ubuntu:wght@300;400;500;700&display=swap");

\* {

  margin: 0;

  box-sizing: border-box;

}

body {

  background-color: #f7f7f7;

  font-family: "Poppins";

  overflow-x: hidden;

}

:root {

  --primary: #626ee3;

}

img {

  max-width: 100%;

}

a {

  text-decoration: none;

  color: inherit;

}

ul {

  list-style: none;

}

.main-cont {

  width: 90%;

  margin: 20px auto 0 auto;

  display: flex;

  border: 1px solid var(--primary);

}

.ques-cont {

  width: 50%;

  padding: 50px;

  display: flex;

  flex-direction: column;

}

.img-asked-by {

  width: 100px;

  margin: 0 auto;

  height: 100px;

  /\* background-color: var(--primary); \*/

  display: flex;

  justify-content: center;

  align-items: center;

}

.ques {

  /\* color: white; \*/

  font-size: 1.5rem;

  display: flex;

  flex-direction: column;

  margin: 20px auto 0 auto;

}

.cat {

  font-size: 1rem;

  margin-top: 20px;

}

.ans-cont {

  width: 50%;

  display: flex;

  flex-direction: column;

  padding: 40px 70px 40px 0;

  /\* margin-bottom: 30px; \*/

}

.img-flo {

  display: flex;

  align-items: center;

  margin-bottom: 10px;

}

.ans-usr {

  /\* width: 40px; \*/

  height: 40px;

  border-radius: 50%;

  display: flex;

}

.ans {

  background: var(--primary);

  padding: 20px;

  color: white;

  text-align: justify;

  margin-bottom: 30px;

  display: flex;

  flex-direction: column;

}

.ans-usr {

  display: flex;

  justify-content: center;

  align-items: center;

}

.fa-user-plus {

  margin-left: 20px;

}

.like {

  display: flex;

  justify-content: flex-end;

  font-size: 1.5rem;

  margin-top: 30px;

}

.mar-like {

  margin-left: 20px;

}

**FUTURE SCOPE OF THE PROJECT**

The proposed system helped people in many ways but it can also be extended to provide more functionality to the user.

A group module can be added so that a particular organization or group of people can only use the group.

Messaging feature can be added so that one user can send text to another user in real-time so that questioning answering becomes more efficient.

**References**

Reference Books

Head First PYTHON with django&SQLITE- By Lynn Beighley& Michael Morrison

The Complete Reference- PYTHON- with django(Guido van rossun)

Reference websites

* + [www.youtube.com](http://www.scribed.com)
  + [www.w3schools.com](http://www.w3schools.com)
  + [www.javatpoints.edu](http://www.javatpoints.edu)
  + <https://docs.djangoproject.com/en/4.0/>
  + https://stackoverflow.com/