

“TRAILBLAZER”
BACHELOR OF TECHNOLOGY
in
COMPUTER SCIENCE AND ENGINEERING



RGUKT

Rajiv Gandhi University of Knowledge Technologies

R.K.VALLEY

Submitted by

B SumaJyothi – R170416

Under the Esteemed guidance of

Mrs.V. Sravani

RGUKT RK Valley.

DECLARATION

We hereby declare that the report of the B.Tech Major Project Work entitled “TRAILBLAZER” which is being submitted to Rajiv Gandhi University of Knowledge Technologies, RK Valley, in partial fulfillment of the requirements for the award of Degree of Bachelor of Technology in Computer Science and Engineering, is a bonafide report of the work carried out by us. The material contained in this report has not been submitted to any university or institution for award of any degree.

B SumaJyothi -- R170416

Dept. Of Computer
Science and Engineering

RAJIV GANDHI UNIVERSITY OF KNOWLEDGE TECHNOLOGIES



RGUKT

(A.P. Government Act 18 of 2008)
RGUKT, RK VALLEY

Department of Computer Science and Engineering

CERTIFICATE FOR PROJECT COMPLETION

This is to certify that project entitled “TRAILBLAZER” submitted by B SumaJyothi (R170416) under our guidance and supervision for the partial fulfillment for the degree Bachelor of Technology in Computer Science and Engineering during the academic semester 2 2022-2023 at RGUKT, RK VALLEY. To the best of my knowledge, the results embodied in this dissertation work have not been submitted to any University or Institute for the award of any degree or diploma.

Project Internal Guide

Mrs.V. Sravani

Assistant Professor

RGUKT, RK Valley

Head of the Department

Mr.N.Satya Nandaram

HOD Of CSE

RGUKT, RK Valley

Index

1. Introduction -----	5
2. Abstraction -----	6
3. Objectives -----	7
4. Intended Audience -----	7
5. Product Vision -----	7
6. Scope -----	8
7. Modules -----	9
8. Technologies -----	9
8.1 HTML -----	9
8.2 CSS -----	9
8.5 DJANGO -----	10
8.6 MySQL -----	10-11
9. System in context -----	12
10. Functional Requirements -----	12-16
11. Starting Project -----	17
12. Admin Page -----	17-18
13. Source Code -----	19-20
14. Web Pages -----	21-25
15. References-----	25

1.Introduction

The “TRAILBLAZER” demonstrates how to make projects. This application helps the user to create new projects and add existing projects. The user can check the projects according to their requirements like according to the departments or according to their domains. This application gives guidance for the users to prepare projects.

The application is reduced as much as possible to avoid errors while entering the data. It also provides error message while entering invalid data. No formal knowledge is needed for the user to use this system. Thus by this all it proves it is user-friendly. Trailblazer, as described above, can lead to error free, secure, reliable and fast management system. It can assist the user to concentrate on their other activities rather to concentrate on the record keeping. Thus it will help the users in better utilization of resources.

Trailblazer, whether big or small, has challenges to overcome and managing the information of Activity, Projects, Domains. Trailblazer has different modules therefore we design pre-requisites for different departments, project ideas that are adapted to your managerial requirements. This is designed to assist in strategic planning, and will help you ensure that your organization is equipped with the right level of information and details for your future goals. Also, for those busy executive who are always on the go, our systems come with remote access features, which will allow you to manage your workforce anytime, at all times. These systems will ultimately allow you to better manage resources.

2. Abstract of the Trailblazer:

The purpose of Trailblazer is to automate the existing manual system by the help of computerized equipments and full-fledged computer software, fulfilling their requirements, so that their valuable data/information can be stored for a longer period with easy accessing and manipulation of the same. The required software and hardware are easily available and easy to work with.

Trailblazer, as described above, can lead to error free, secure, reliable and fast management system. It can assist the user to concentrate on their other activities rather to concentrate on the record keeping. Thus it will help user in better utilization of resources. The organization can maintain computerized records without redundant entries. That means that one need not be distracted by information that is not relevant, while being able to reach the information.

The aim is to automate its existing manual system by the help of computerized equipments and full-fledged computer software, fulfilling their requirements, so that their valuable data/information can be stored for a longer period with easy accessing and manipulation of the same. Basically the project describes how to manage for good performance and better services for the users.

3.Objective of Project on Trailblazer:

The main objective of the project on Trailblazer is to provide the project details by using different domains and project ideas. The project is totally built at administrative end and thus only the administrator is guaranteed the access. The purpose of the project is to build an application program to provide the overview of the projects. It contains some inbuilt projects details and stores the project details uploaded by the users.

4.Intended Audience:

The intended audience will be the users who can sign up to access the Trailblazer and also users can check the uploaded projects by the host and the users and the user can upload the project reports which they have done

5.Product Vision:

Vision Statement:

The product vision is to develop an Trailblazer, which is a user friendly and easily accessible. This Trailblazer helps to provide services to user which will decrease manual work for the user and make project successfully.

6.Scope of the Trailblazer :

It may help collecting Project details in detail. In a very short time, the collection will be obvious, simple and sensible. It will help a person to know the projects perfectly and vividly. It also helps the user to get an about project and project ideas.

- Our project aims to provide the projects in different domains, i.e., we have tried to computerize various various types of projects.
- In computer system, the person has to fill the various forms & number of copies of the forms can be easily generated at a time.
- The system generates types of information that can be used for various purposes.
- It satisfy the user requirement.
- Be easy to understand by the user and operator.
- Be easy to operate.
- Have a good user interface.
- Be expandable.

7.Modules of Trailblazer :

- Prerequisites Module: Used for managing the project details.
- Project ideas : Used for managing the project ideas.
- Project Overview : Used for the source code of the projects.
- Sign Up Module: Used for managing the Sign Up details.
- Login Module: Used for managing the Login details.
- Home Module: Used for managing the Web Page details.
- About Us Module: Used for managing the Managers details.
- Registration Module: Used for managing the Registration details.

8.Technologies:

- Html
- CSS
- Django
- MySQL

8.1 Html:

HTML is a Hyper Text Markup Language used to create electronic documents (called pages) that are displayed on the World Wide Web. Each page contains several connections to other pages called hyperlinks. Every web page you see was written using one version of HTML.

8.2 CSS:

Cascading Style Sheets (CSS) is a styleshet language used to describe the presentation of a document written in HTML or XML. CSS describes how elements should be rendered on screen, on paper, in speech, or on other media.

8.3: Django:

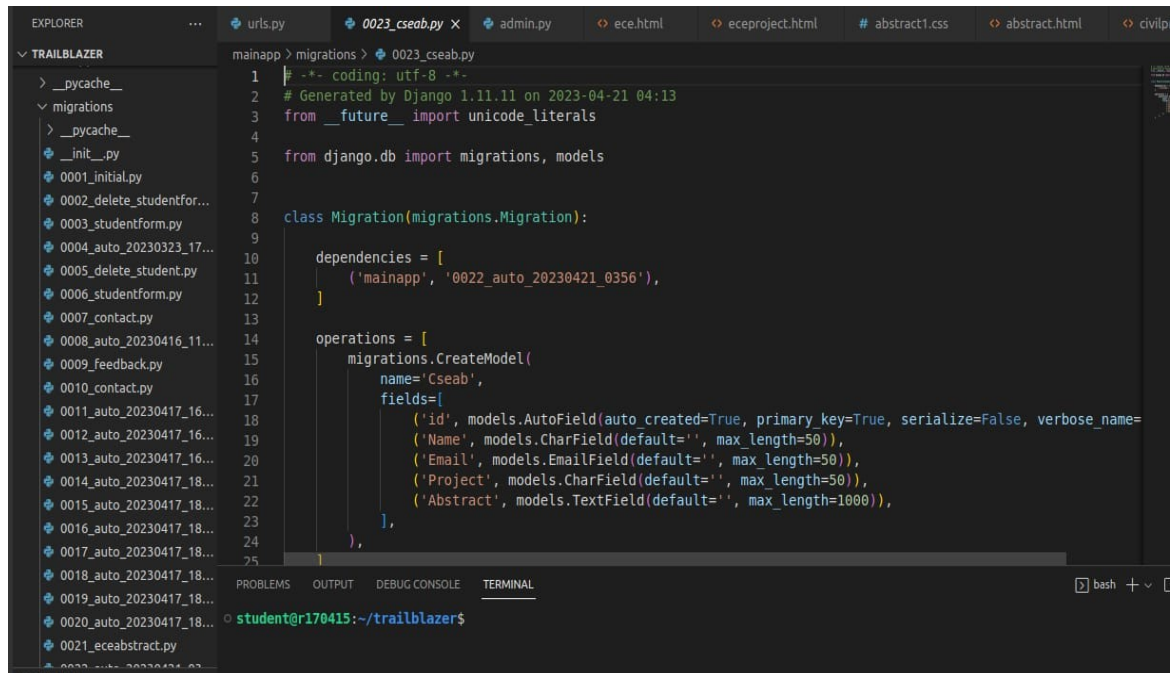
Django is a high-level Python web framework tht enables rapid development of secure and maintainable websites. Built by experienced developers, Django takes care of much of the hassle of web development, so you can focus on writing your app without needing to reinvent the wheel.

8.4: MySQL:

MySQL is a relational database management system

The database structure is organized into physical files optimized for speed. The logical data model, with objects such as data tables, views, rows, and columns, offers a flexible programming environment.

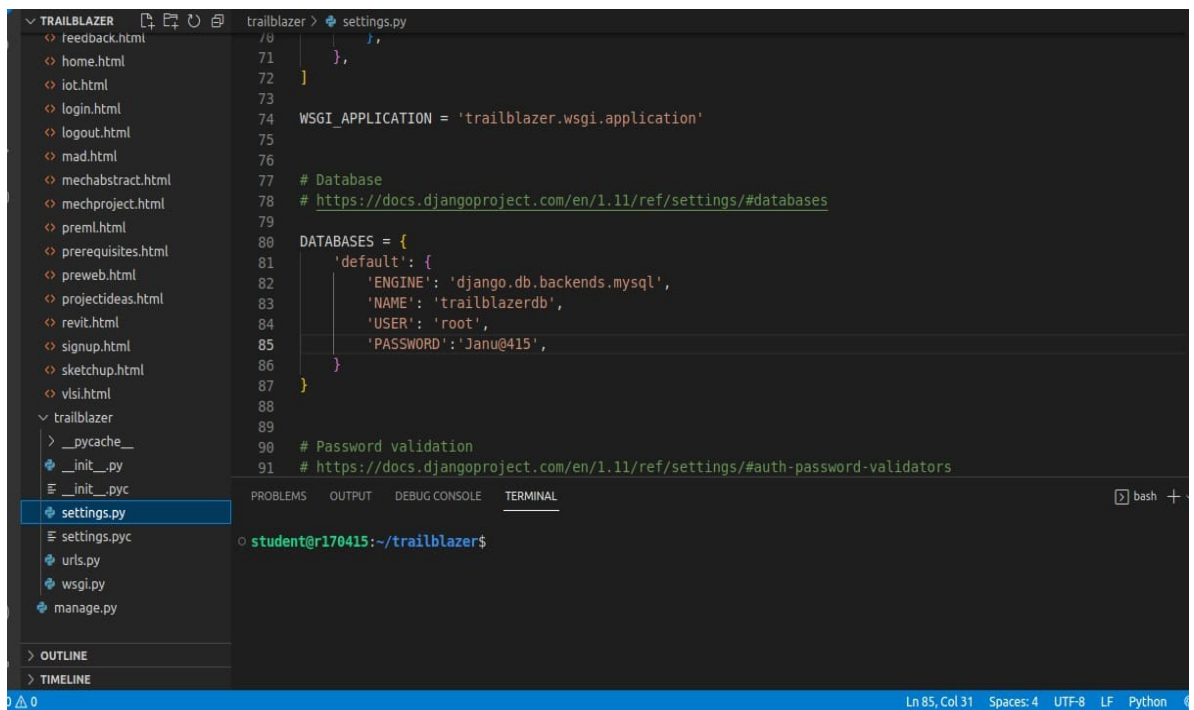
Connection of DBSQLite in Django :



The screenshot shows a VS Code editor with a file explorer on the left and a code editor on the right. The file explorer shows a project named 'TRAILBLAZER' with a 'migrations' folder. The code editor shows the file '0023_cseab.py' with the following content:

```
1  # -*- coding: utf-8 -*-
2  # Generated by Django 1.11.11 on 2023-04-21 04:13
3  from __future__ import unicode_literals
4
5  from django.db import migrations, models
6
7
8  class Migration(migrations.Migration):
9
10     dependencies = [
11         ('mainapp', '0022_auto_20230421_0356'),
12     ]
13
14     operations = [
15         migrations.CreateModel(
16             name='Cseab',
17             fields=[
18                 ('id', models.AutoField(auto_created=True, primary_key=True, serialize=False, verbose_name='ID')),
19                 ('Name', models.CharField(default='', max_length=50)),
20                 ('Email', models.EmailField(default='', max_length=50)),
21                 ('Project', models.CharField(default='', max_length=50)),
22                 ('Abstract', models.TextField(default='', max_length=1000)),
23             ],
24         ),
25     ]
```

The terminal at the bottom shows the command prompt: `student@r170415:~/trailblazer$`.



The screenshot shows a VS Code editor with a file explorer on the left and a code editor on the right. The file explorer shows a project named 'TRAILBLAZER' with a 'settings.py' file. The code editor shows the file 'settings.py' with the following content:

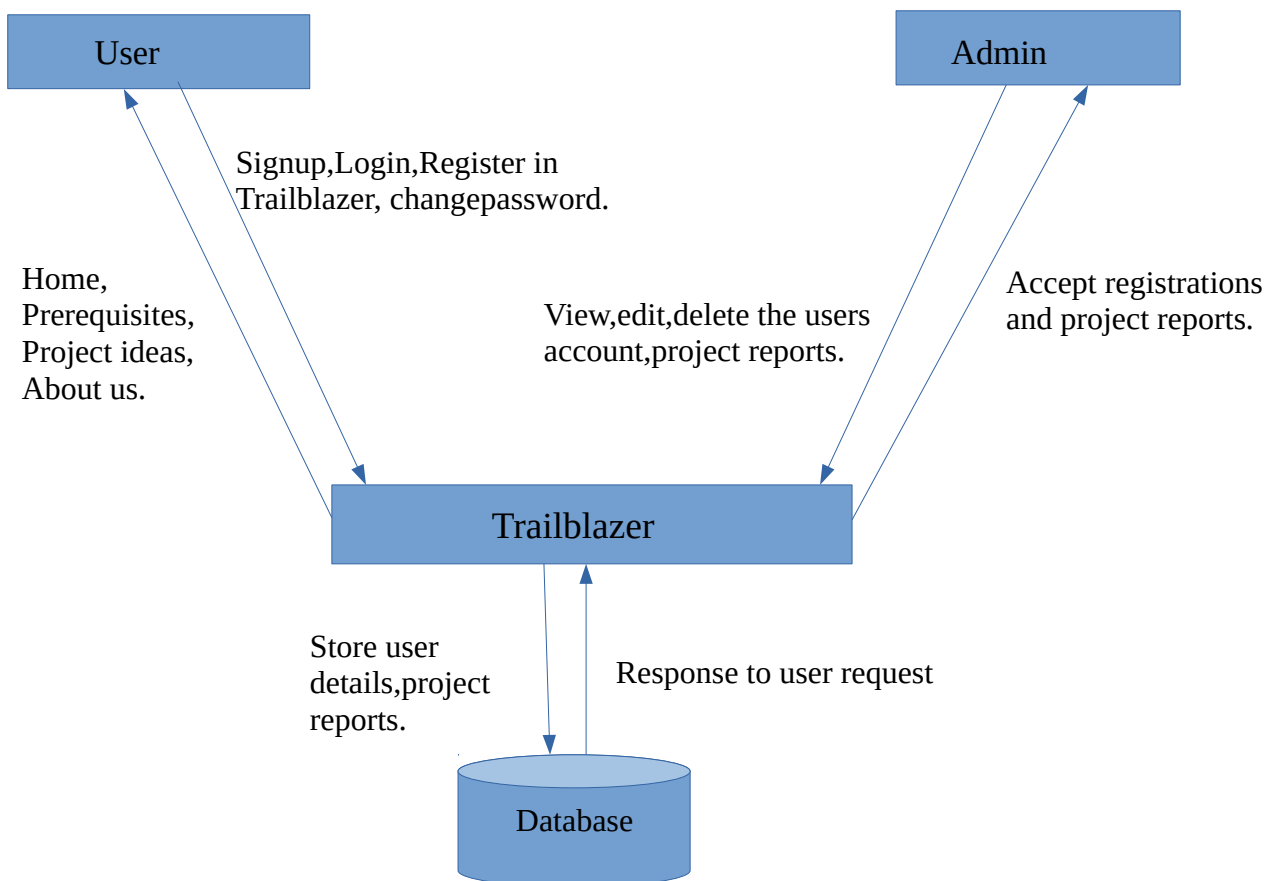
```
1  # Django settings for trailblazer.
2  #
3  # Generated by Django 1.11.11 on 2023-04-21 04:13
4
5  import os
6
7  # Build paths inside the project like this: os.path.join(BASE_DIR, 'path')
8  BASE_DIR = os.path.dirname(os.path.dirname(os.path.abspath(__file__)))
9
10 # Quick-start development settings - unsuitable for production
11 # See https://docs.djangoproject.com/en/1.11/ref/settings/#databases
12
13 # SECURITY WARNING: don't run with debug turned on in production!
14 DEBUG = True
15
16 # Application definition
17 INSTALLED_APPS = (
18     'django.contrib.admin',
19     'django.contrib.auth',
20     'django.contrib.contenttypes',
21     'django.contrib.sessions',
22     'django.contrib.messages',
23     'django.contrib.staticfiles',
24     'mainapp',
25 )
26
27 # Database
28 # https://docs.djangoproject.com/en/1.11/ref/settings/#databases
29
30 DATABASES = {
31     'default': {
32         'ENGINE': 'django.db.backends.mysql',
33         'NAME': 'trailblazerdb',
34         'USER': 'root',
35         'PASSWORD': 'Janu@415',
36     }
37 }
38
39 # Password validation
40 # https://docs.djangoproject.com/en/1.11/ref/settings/#auth-password-validators
41
42 AUTH_PASSWORD_VALIDATORS = (
43     'django.contrib.auth.password_validation.UserAttributeSimilarityValidator',
44     'django.contrib.auth.password_validation.MinimumLengthValidator',
45     'django.contrib.auth.password_validation.CommonPasswordValidator',
46     'django.contrib.auth.password_validation.NumericPasswordValidator',
47 )
```

The terminal at the bottom shows the command prompt: `student@r170415:~/trailblazer$`.

9.System in Context:

This Trailblazer provide the list of contents which having the user ,admin,trailblazer and database.

Context Diagram:



The below table provides a set of user visible that define the functionalities that are in **Trailblazer**

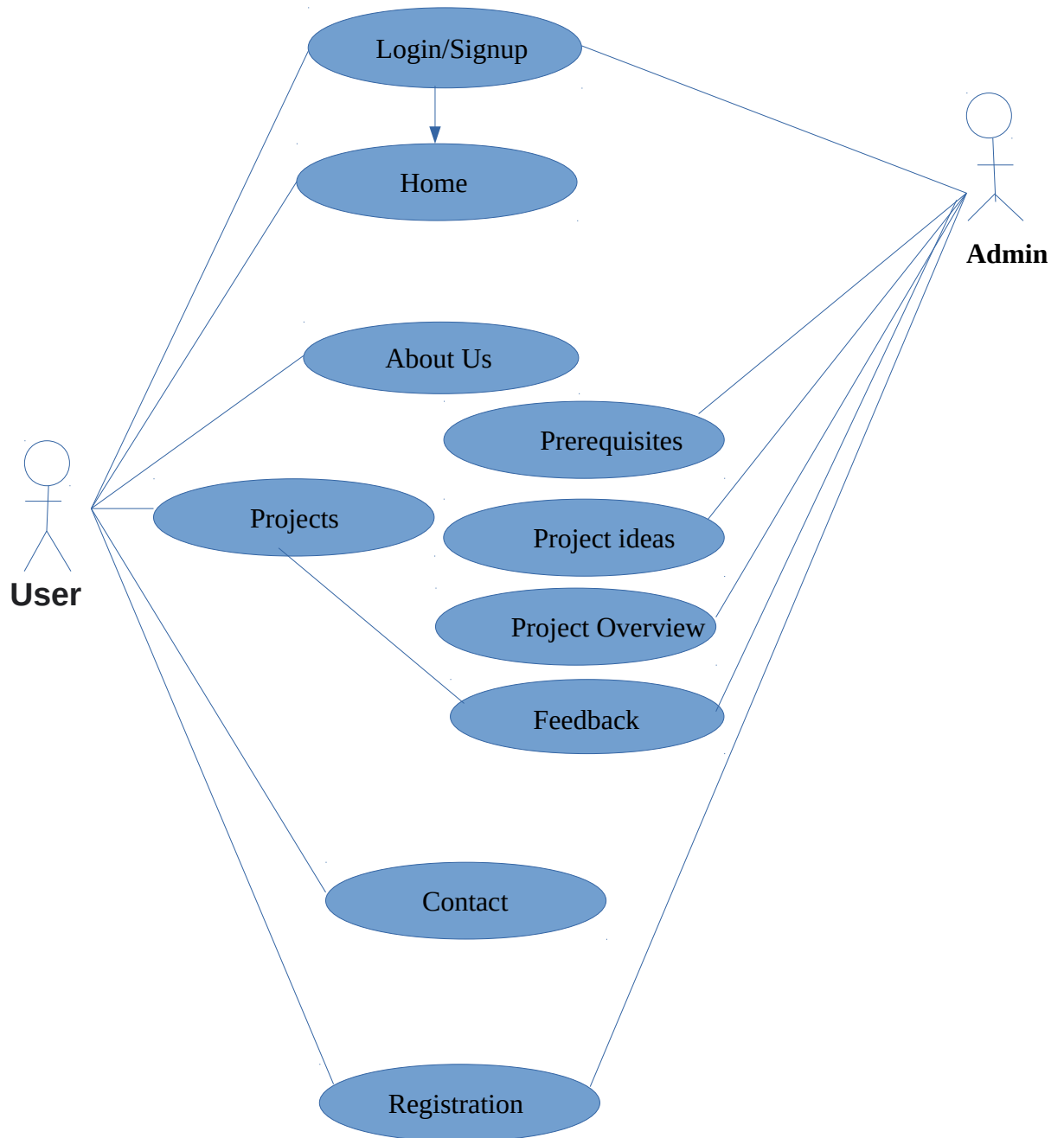
		Actor	Action	Object	Frequen cy	Arrival Pattern	Respons e
1.		User	onclick	Signup	1/day	Episodic	It asks userna me first name last name email passwor d
2.		User	onclick	Login	1/day	Episodic	It asks userna me passwor d
3.		User	onclick	Main	1/day	Episodic	It displays all modules in the software
4.		User	onclick	Home	1/day	Episodic	It displays informati on about software
5.		User	onclick	About us	1/day	Episodic	It displays informati on about

							managers
6.		User	onclick	Prerequisites	1/day	Episodic	Branch and domain
7.		User	onclick	Project Ideas	1/day	Episodic	It displays info about projects
8.		User	onclick	Project Overview	1/day	Episodic	It displays the source code of the project
9.		User	onclick	upload	1/day	Episodic	It displays upload option

10.Functional Requirements:

Use Case Diagram:

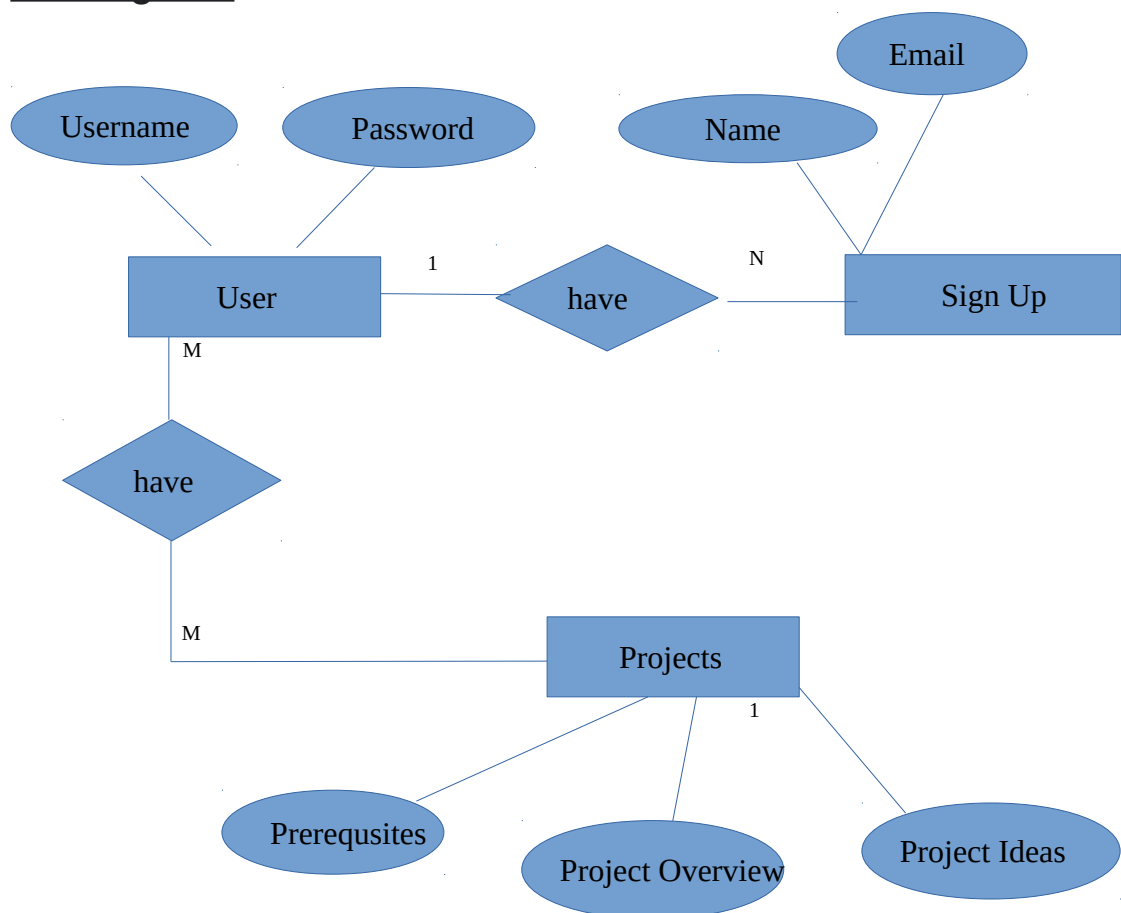
Trailblazer :



Use Case Overview:

	Use-case ID	Use-case Name	Priority	Stability	Verifiable
1.	UC-TB-PR	Prerequisites	High	Stable	Verifiable
2.	UC-TB-RG	Registrations	High	Not Stable	Verifiable
3.	UC-TB-PI	Project Ideas	High	Stable	Verifiable
4.	UC-TB-UP	Project Overview.	High	Stable	Verifiable

ER Diagram:



11. Starting Project:

Steps:

1.Install mysql in the system

2.Install django by using pip

`pip install django`

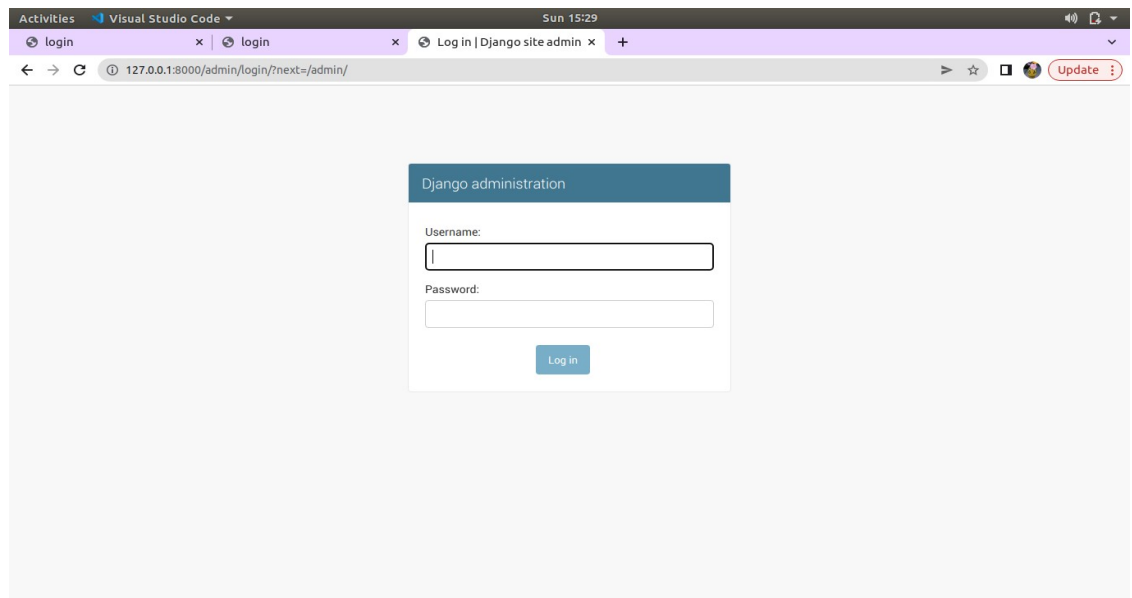
3.Once we installed django in our system,we will get 'django-admin' command line tool ,which can be used to create our Django project. Create a project using : **django-admin startproject projectName**

By default some of the python files will be installed and database dbsqlite too. Open the project folder using visual studio code.

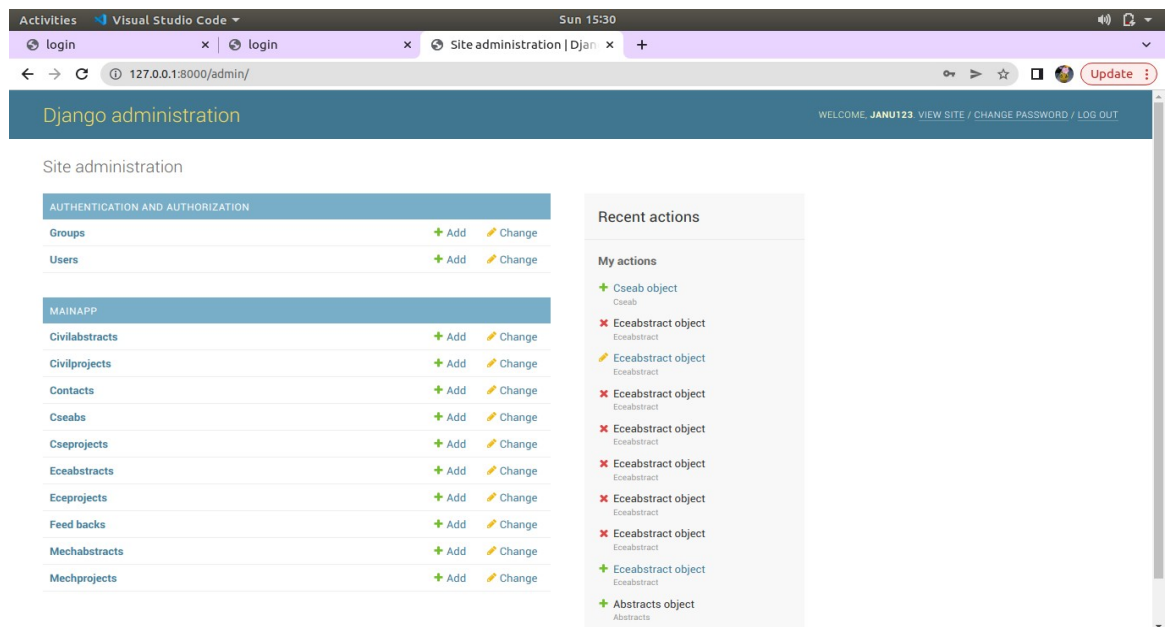
4.Create an application in that project folder

12. Admin Page:

Django provides a default admin interface which can be used to perform create, read, update and delete operations on the model directly. It reads set of data that explain and gives information about data from the model, to provide an instant interface where the user can adjust contents of the application.



Admin Login Page



13. Source Code:

Main.html:

```
<!DOCTYPE html>
{%load staticfiles%}
<html>
<head>
<title>home page</title>
<link rel="stylesheet" href="{% static 'css/home1.css' %}">
</head>
<body>
<b>
<div class="menu-bar">
<ul>
<div class="logo">

</div>
<li class="active"><i class="fa fa-home"></i><a href="{% url 'projectideas' %}">ProjectIdeas</a></li>
<li class="active"><i class="fa fa-home"></i><a href="{% url 'preview' %}">PreRequisites</a></li>
<li class="active"><i class="fa fa-home"></i><a href="#">ProjectOverview</a>
<div class="sub-menu">
<ul>
<li><a href="{% url 'cseview' %}">CSE</a></li>
<li><a href="{% url 'eceview' %}">ECE</a></li>
<li><a href="#">CIVIL</a></li>
<li><a href="#">MECHANICAL</a></li>
</ul>
</div>
</li>
<li><a href="{% url 'about' %}">About us</a></li>
<li><a href="{% url 'contact' %}">Contact</a></li>
<li><a href="{% url 'feed' %}">Feedback</a></li>
<li><a href="{% url 'loginaction' %}">Logout</a></li>
</ul>
</div>
</b>
<div class="scroll">
<marquee scrollamount="15px" direction="right" width="200%">WELCOME TO TRAILBLAZER</marquee>
<marquee scrollamount="15px" id="caption">explore your thoughts</marquee>

</div>

</body>
```

</html>

views.py (main):

-*- coding: utf-8 -*-

from __future__ import unicode_literals

```
from django.shortcuts import render, redirect, HttpResponseRedirect
from django.contrib.auth import authenticate, login
from django.contrib.auth.forms import AuthenticationForm
from django.core import serializers
from django.contrib import messages
from .forms import SignupForm
from django.core.files.storage import FileSystemStorage
from django.http import HttpResponse, HttpResponseNotFound
from .functions import handle_uploaded_file
from .models import
FeedBack, Contact, Abstracts, Eceabstract, Cseab, Civilabstract, Mechabstract
from .models import Cseproject, Eceproject, Civilproject, Mechproject
# Create your views here.
def loginaction(request):
    if request.method == "POST":
        fm = AuthenticationForm(request=request, data=request.POST)
        if fm.is_valid():
            uname=fm.cleaned_data['username']
            passwd=fm.cleaned_data['password']
            user= authenticate(username=uname,password=passwd)
            if user is not None:
                login(request,user)
                return HttpResponseRedirect('home')
            else:
                messages.error(request,'please Enter correct username and password')
                return HttpResponseRedirect('login')
            else:
                fm=AuthenticationForm()
                return render(request,'login.html',{'form':fm})
    def signupaction(request):
        if request.method == "POST" :
            fm = SignupForm(request.POST)
            if fm.is_valid():
                messages.success(request,'Account created successfully')
                fm.save()
                fm=SignupForm()
            else:
                fm = SignupForm()
                return render(request,'signup.html',{'form':fm})
    def home(request):
```

```

return render(request,"home.html")
def projectideas(request):
return render(request,'projectideas.html')
def pdf_view(request):
fs=FileSystemStorage()
filename = 'mypdf.pdf'
if fs.exists(filename):
with fs.open(filename) as pdf:
response = HttpResponse(pdf,content_type='application/pdf')
response['content-Disposition']='inline; filename=mypdf.pdf'
return response
else:
return HttpResponseRedirect("the request pdf is not found in the server")
def abstract(request):
if request.method=='POST':
Name=request.POST['name']
Email=request.POST['email']
Project=request.POST['project']
Abstract=request.POST['abstract']
messages.success(request,'Abstract Submitted successfully')
obj=Eceabstract()
obj.Name=Name
obj.Email=Email
obj.Project=Project
obj.Abstract=Abstract
obj.save()
ab = serializers.serialize("python",Eceabstract.objects.all())
my_dict={
'ab': ab,
}
return render(request,'abstract.html',context=my_dict)

```

14. Web Pages:

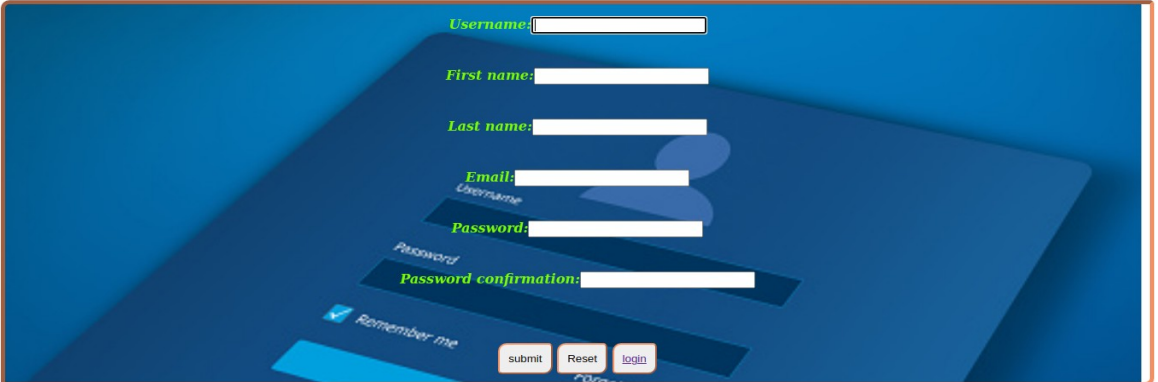
14.1 Sign Up Page:

Activities Visual Studio Code Sun 16:19

login x home page x Screenshot from 2023-04- x Screenshot from 2023-04- x Sign Up Form x +

127.0.0.1:8000/mainapp/signup/ Update

Sign Up Form



The Sign Up Form is displayed on a blue background with a laptop graphic. It contains the following fields and elements:

- Username:
- First name:
- Last name:
- Email:
- Password:
- Password confirmation:
- ☒ Remember me
- submit, Reset, login buttons

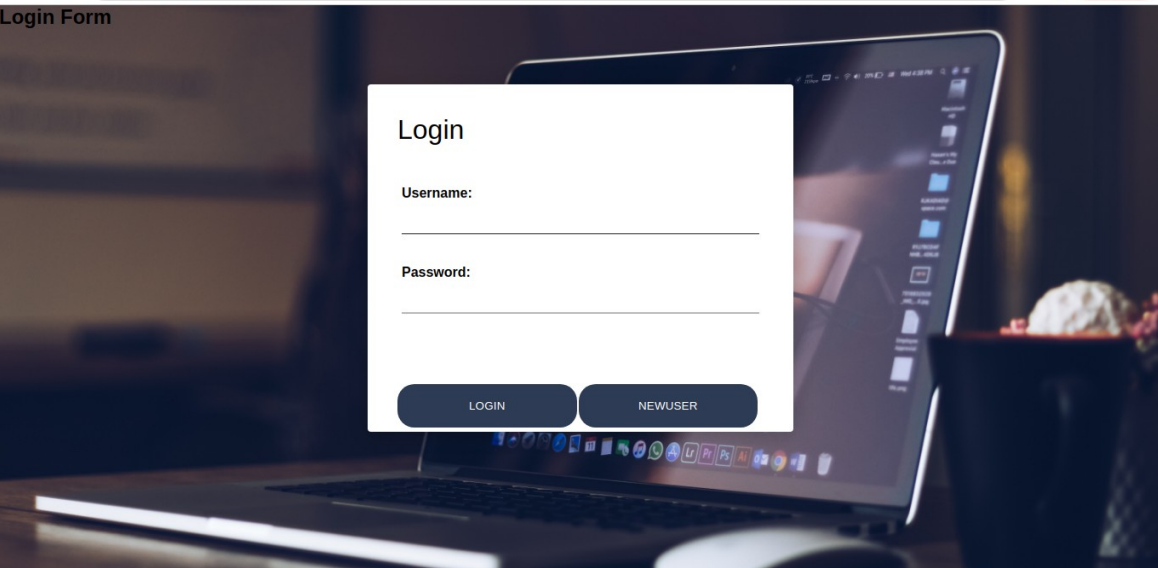
14.2 Login Page:

Activities Visual Studio Code Sun 16:19

login x home page x Screenshot from 2023-04- x Screenshot from 2023-04- x login x +

127.0.0.1:8000 Update

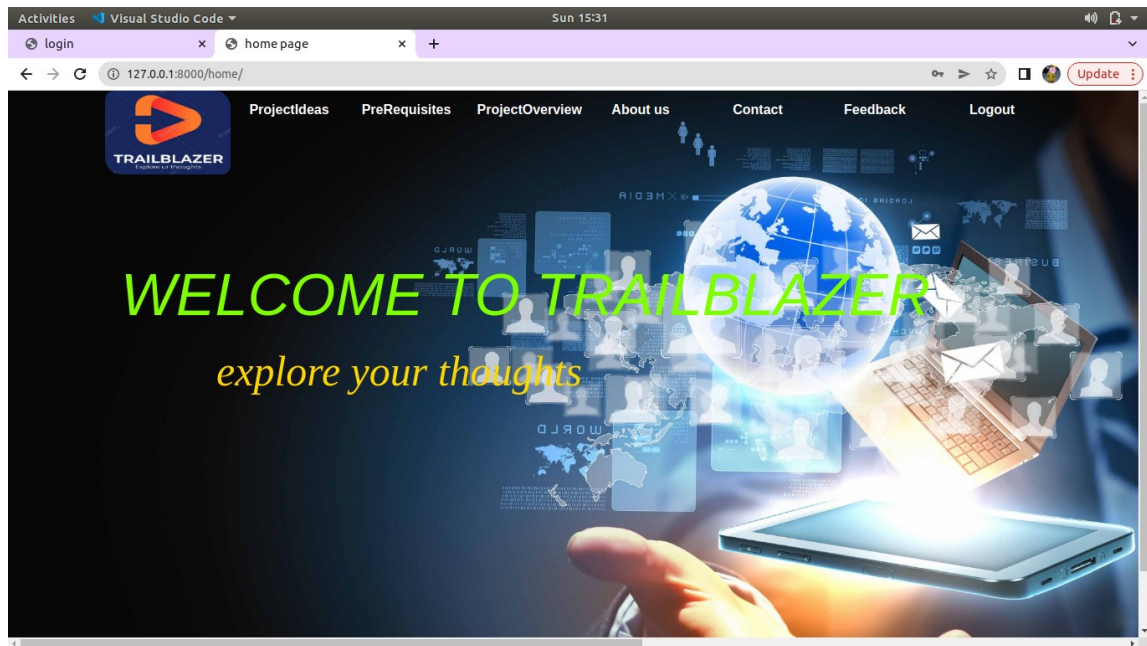
Login Form



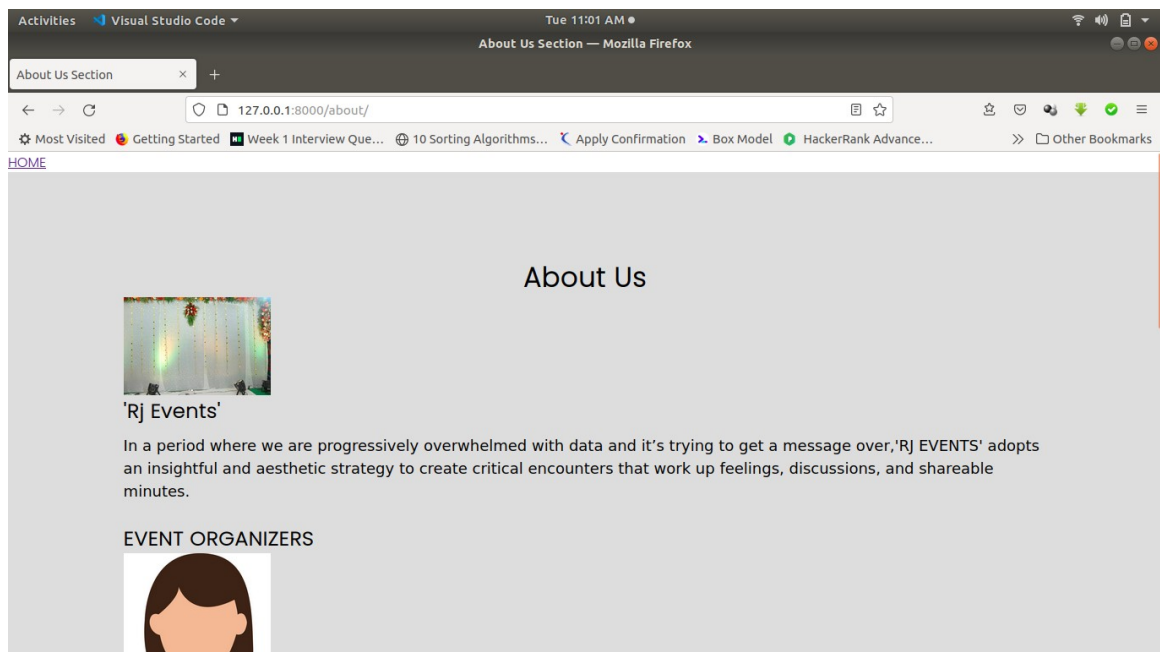
The Login Form is displayed on a dark background with a laptop graphic. It contains the following fields and elements:

- Login
- Username:
- Password:
- LOGIN, NEWUSER buttons

14.3 Home Page:



14.4 About Us Page:



14.5 Prerequisites Page:

Activities Visual Studio Code Sun 16:22


login x home page x Screenshot from 2023-04- x Screenshot from 2023-04- x Prerequisites x +

127.0.0.1:8000/mainapp/preview

Trailblazer HOME CSE ECE CIVIL MECHANICAL

Computer Science Engineering

Students who are beginner in doing a project.This article might be helpful them.Prerequisites like technonologies,databases,frameworks which is needed for beginners to do pprojects in different domain is available here.



127.0.0.1:8000/mainapp/preview#CSE

and Communication Engineering

14.6 Project Ideas


Activities Visual Studio Code Sun 16:22

login x home page x Screenshot from 2023-04- x Screenshot from 2023-04- x Projectideas x +

127.0.0.1:8000/mainapp/projectideas

Trailblazer HOME CSE ECE CIVIL MECHANICAL

2/6



ECE

Computer Science Engineering

Students who are beginner in doing a project.This article might be helpful them.Project Ideas for beginners and final year project ideas in different domain is available here.

14.7 Contact US

The screenshot shows a web browser window with the URL `127.0.0.1:8000/mainapp/contact`. The page is titled "Contact Us" and features a dark background with a world map and a "Send Message" modal form. The modal form has fields for "Full Name", "Email", and "Type Your Message...", and a "Submit" button. On the left side of the page, there is contact information:

- Address:** RGUKT RK Valley, Idupulapaya, Vempalli, Kadapa, 516330
- Phone:** 9100810469, 9502836884, 9553209846
- Email:** r170829@rguktrkv.ac.in, r170415@rguktrkv.ac.in, r170416@rguktrkv.ac.in

14.8 Feedback

The screenshot shows a web browser window with the URL `127.0.0.1:8000/mainapp/feed`. The page is titled "Feedback Form" and features a light green background. The form has fields for "Name" (filled with "Ex:-Jahnavi"), "Email" (filled with "Ex:-ekotjahnavi0504@gmail.com"), and a large text area for "Feedback:". Below the form, there are buttons for "Submit Feedback" and "Reset". At the bottom of the page, there is a list of feedback comments:

- jayasree: [this site is very useful.thank you](#)
- janu: [nice](#)
- janu: [nice](#)
- jaya: [useful](#)

15. References:

1. https://www.udemy.com/share/101soi3@8EC6KVhGb09rQm0rwqy6ID6GET0ufkpd_V64dUG4wYhbyCgileSXLuiZoUz1BbBazw==/
2. https://www.w3schools.com/django/django_intro.php
3. <https://www.digitalocean.com/community/tutorials/how-to-install-the-django-web-framework-on-ubuntu-18-04>
4. <https://stackoverflow.com/questions/36797051/django-login-error-attempt-to-write-a-readonly-database>