**Step-by-Step Guide: Projects & Authentication System**

**1. Ensure Virtual Environment is Activated**

Before making any changes, activate the virtual environment:

source venv/bin/activate # macOS/Linux

venv\Scripts\activate # Windows

**2. Navigate to the Project Directory**

Ensure you are in the correct project folder:

cd your\_project\_name

**PART 1: PROJECTS APP**

**3. Create the projects App (If Not Already Created)**

python manage.py startapp projects

**4. Register the projects App in settings.py**

Open settings.py and add "projects" to INSTALLED\_APPS:

INSTALLED\_APPS = [

...

"projects",

]

**5. Define the Project Model with User Relationship (models.py)**

Modify projects/models.py to link each project to a user:

from django.db import models

from django.contrib.auth.models import User

class Project(models.Model):

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

title = models.CharField(max\_length=200)

description = models.TextField(blank=True, null=True)

created\_at = models.DateTimeField(auto\_now\_add=True)

def \_\_str\_\_(self):

return self.title

**Run Migrations**

Apply model changes to the database:

python manage.py makemigrations projects

python manage.py migrate

**6. Register the Project Model in Django Admin (admin.py)**

Open projects/admin.py and register the model:

from django.contrib import admin

from .models import Project

admin.site.register(Project)

**7. Create Views for Projects (views.py)**

Modify projects/views.py:

from django.shortcuts import render, redirect

from django.contrib.auth.decorators import login\_required

from .models import Project

from .forms import ProjectForm

def project\_list(request):

projects = Project.objects.all()

return render(request, "projects/project\_list.html", {"projects": projects})

@login\_required

def project\_create(request):

if request.method == "POST":

form = ProjectForm(request.POST)

if form.is\_valid():

project = form.save(commit=False)

project.user = request.user # Assign logged-in user

project.save()

return redirect("project\_list")

else:

form = ProjectForm()

return render(request, "projects/project\_form.html", {"form": form})

@login\_required

def project\_edit(request, pk):

project = Project.objects.get(id=pk)

if project.user != request.user: # Prevent editing others' projects

return redirect("project\_list")

if request.method == "POST":

form = ProjectForm(request.POST, instance=project)

if form.is\_valid():

form.save()

return redirect("project\_list")

else:

form = ProjectForm(instance=project)

return render(request, "projects/project\_form.html", {"form": form})

@login\_required

def project\_delete(request, pk):

project = Project.objects.get(id=pk)

if project.user == request.user: # Allow only the owner to delete

project.delete()

return redirect("project\_list")

**8. Create the Project Form (forms.py)**

Inside projects, create a new file forms.py:

from django import forms

from .models import Project

class ProjectForm(forms.ModelForm):

class Meta:

model = Project

fields = ["title", "description"]

**9. Configure URLs for Projects (urls.py)**

Create projects/urls.py:

from django.urls import path

from . import views

urlpatterns = [

path("", views.project\_list, name="project\_list"),

path("new/", views.project\_create, name="project\_create"),

path("edit/<int:pk>/", views.project\_edit, name="project\_edit"),

path("delete/<int:pk>/", views.project\_delete, name="project\_delete"),

]

Then include this in the **main** urls.py:

from django.urls import path, include

urlpatterns = [

path("projects/", include("projects.urls")),

]

**10. Create Templates for Projects**

Inside projects/templates/projects/, create:

**project\_list.html (Project Listing)**

<h1>Projects</h1>

{% if user.is\_authenticated %}

<a href="{% url 'project\_create' %}">Create New Project</a>

{% endif %}

<ul>

{% for project in projects %}

<li>

{{ project.title }} by {{ project.user.username }}

{% if project.user == request.user %}

<a href="{% url 'project\_edit' project.id %}">Edit</a>

<a href="{% url 'project\_delete' project.id %}">Delete</a>

{% endif %}

</li>

{% empty %}

<li>No projects yet.</li>

{% endfor %}

</ul>

**project\_form.html (Create/Edit Project)**

<h1>{% if form.instance.pk %}Edit{% else %}Create{% endif %} Project</h1>

<form method="POST">

{% csrf\_token %}

{{ form.as\_p }}

<button type="submit">Save</button>

</form>

**PART 2: USER AUTHENTICATION**

**11. Create users App**

python manage.py startapp users

**12. Register users in settings.py**

INSTALLED\_APPS = [

...

"users",

]

**13. Create Authentication Views (views.py)**

Inside users/views.py:

from django.shortcuts import render, redirect

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

from django.contrib.auth.forms import UserCreationForm

def register(request):

if request.method == "POST":

form = UserCreationForm(request.POST)

if form.is\_valid():

user = form.save()

login(request, user)

return redirect("project\_list")

else:

form = UserCreationForm()

return render(request, "users/register.html", {"form": form})

def user\_login(request):

if request.method == "POST":

username = request.POST["username"]

password = request.POST["password"]

user = authenticate(request, username=username, password=password)

if user:

login(request, user)

return redirect("project\_list")

return render(request, "users/login.html")

def user\_logout(request):

logout(request)

return redirect("project\_list")

**14. Set Up URLs for Authentication (users/urls.py)**

from django.urls import path

from . import views

urlpatterns = [

path("register/", views.register, name="register"),

path("login/", views.user\_login, name="login"),

path("logout/", views.user\_logout, name="logout"),

]

Include this in the **main** urls.py:

urlpatterns += [

path("users/", include("users.urls")),

]

**15. Create Authentication Templates**

Inside users/templates/users/, create:

**register.html**

<h1>Register</h1>

<form method="POST">

{% csrf\_token %}

{{ form.as\_p }}

<button type="submit">Register</button>

</form>

**login.html**

<h1>Login</h1>

<form method="POST">

{% csrf\_token %}

<input type="text" name="username" placeholder="Username">

<input type="password" name="password" placeholder="Password">

<button type="submit">Login</button>

</form>

That’s it! 🚀 Your project now supports **user authentication** and **project management**, ensuring users can only manage their own projects while everyone can view all projects. Let me know if you need additional features! 😊