Creating a Simple Resume Page in Django

Step 1: Set up Django Project

1. Create a new Django project:

```
django-admin startproject resume_project

cd resume_project
```

2. Create a new Django app:

```
python manage.py startapp resume
```

3. Add the new app to INSTALLED_APPS in resume_project/settings.py:

```
INSTALLED_APPS = [

    ...

    'resume',

]
```

Step 2: Create Models

In resume/models.py, define models for different sections of the resume, such as Profile, Experience, Education, and Skills.

```
from django.db import models


class Profile(models.Model):
    name = models.CharField(max_length=100)
    email = models.EmailField()
    phone = models.CharField(max_length=15)
```

```python
    address = models.TextField()

    summary = models.TextField()


class Experience(models.Model):

    profile = models.ForeignKey(Profile, on_delete=models.CASCADE, related_name='experiences')

    job_title = models.CharField(max_length=100)

    company = models.CharField(max_length=100)

    start_date = models.DateField()

    end_date = models.DateField(blank=True, null=True)

    description = models.TextField()


class Education(models.Model):

    profile = models.ForeignKey(Profile, on_delete=models.CASCADE, related_name='educations')

    degree = models.CharField(max_length=100)

    institution = models.CharField(max_length=100)

    start_date = models.DateField()

    end_date = models.DateField(blank=True, null=True)


class Skill(models.Model):

    profile = models.ForeignKey(Profile, on_delete=models.CASCADE, related_name='skills')

    name = models.CharField(max_length=100)
```

Step 3: Create Views

In resume/views.py, create views to render the resume page. For simplicity, we'll create a view that displays a single profile and its related sections.

```python
from django.shortcuts import render

from .models import Profile


def resume_view(request):

    profile = Profile.objects.first()  # For simplicity, we'll just use the first profile

    return render(request, 'resume/resume.html', {'profile': profile})
```

Step 4: Create URLs

In resume/urls.py, define the URL pattern for the resume view.

```python
from django.urls import path

from .views import resume_view


urlpatterns = [

    path('', resume_view, name='resume'),

]
```

Include the resume app URLs in the main resume_project/urls.py file.

```python
from django.contrib import admin

from django.urls import path, include


urlpatterns = [

    path('admin/', admin.site.urls),

    path('resume/', include('resume.urls')),

]
```

Step 5: Create Templates

Create a directory named templates/resume in the resume app directory, then create a resume.html file inside it.

```html
<!DOCTYPE html>
<html lang='en'>
<head>
    <meta charset='UTF-8'>
    <title>{{ profile.name }}'s Resume</title>
</head>
<body>
    <h1>{{ profile.name }}</h1>
    <p>Email: {{ profile.email }}</p>
    <p>Phone: {{ profile.phone }}</p>
    <p>Address: {{ profile.address }}</p>
    <h2>Summary</h2>
    <p>{{ profile.summary }}</p>

    <h2>Experience</h2>
    <ul>
    {% for experience in profile.experiences.all %}
        <li>
            <h3>{{ experience.job_title }} at {{ experience.company }}</h3>
            <p>{{ experience.start_date }} - {{ experience.end_date }}</p>
            <p>{{ experience.description }}</p>
```

```html
        </li>
    {% endfor %}
    </ul>


    <h2>Education</h2>

    <ul>

    {% for education in profile.educations.all %}

        <li>

            <h3>{{ education.degree }} at {{ education.institution }}</h3>

            <p>{{ education.start_date }} - {{ education.end_date }}</p>

        </li>
    {% endfor %}

    </ul>


    <h2>Skills</h2>

    <ul>

    {% for skill in profile.skills.all %}

        <li>{{ skill.name }}</li>
    {% endfor %}

    </ul>
</body>
</html>
```

Step 6: Migrate and Run the Server

1. Create and apply migrations:

```
python manage.py makemigrations

python manage.py migrate
```

2. Run the development server:

```
python manage.py runserver
```

3. Create a superuser to access the admin interface:

```
python manage.py createsuperuser
```

Step 7: Add Data via Admin Interface

Navigate to /admin, log in with your superuser account, and add a Profile with related Experience, Education, and Skill entries.

Step 8: View the Resume Page

Navigate to /resume to see the resume page rendered with the data you entered.