

Secure Job Application Form System

Using Django

1. Problem Statement

The job application process is often cumbersome for job seekers and employers. A centralized **Job Portal** can streamline this process by allowing users to submit applications and employers to review them efficiently. The goal of this project is to create a web-based secure **Job Portal** using Django that enables job seekers to apply for jobs through an intuitive interface.

2. Step-by-Step Process of Creating the Project

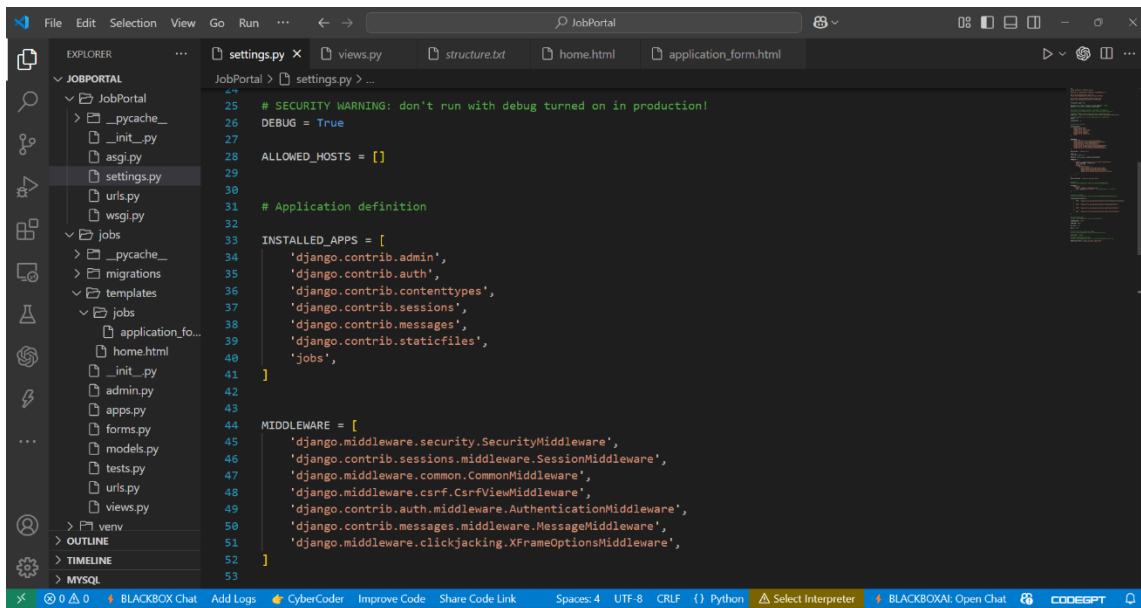
Step 1: Setting Up the Django Project

1. Install Django:
--- pip install django
2. Create a Django project named **JobPortal**:
---django-admin startproject JobPortal
3. Navigate into the project directory:
---cd JobPortal
4. Run the development server to verify setup:
---python manage.py runserver

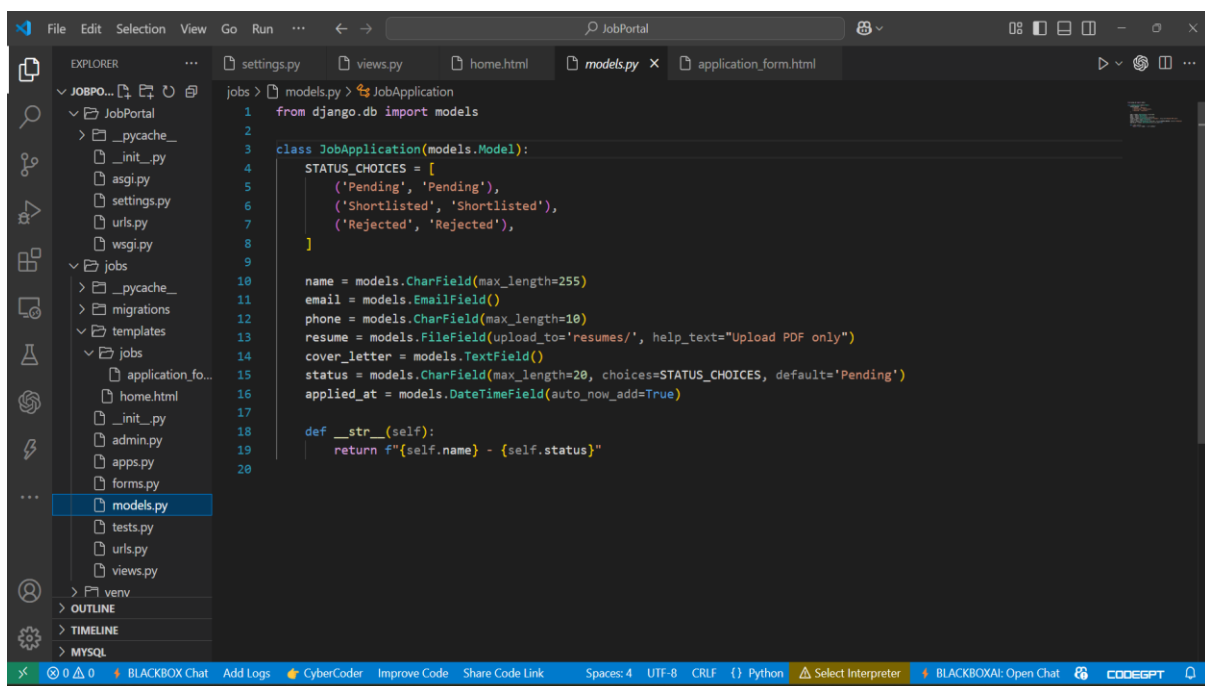
Step 2: Creating a Django App

1. Create an app named **jobs**:
---python manage.py startapp jobs

Add jobs to INSTALLED_APPS in JobPortal/settings.py:



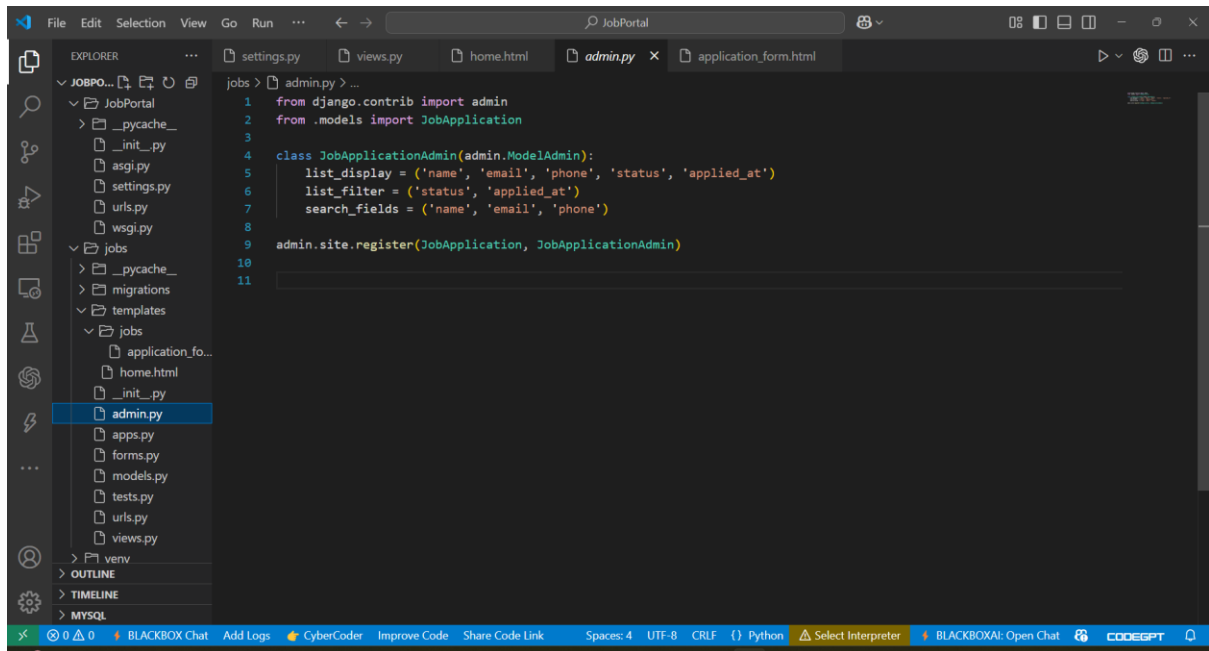
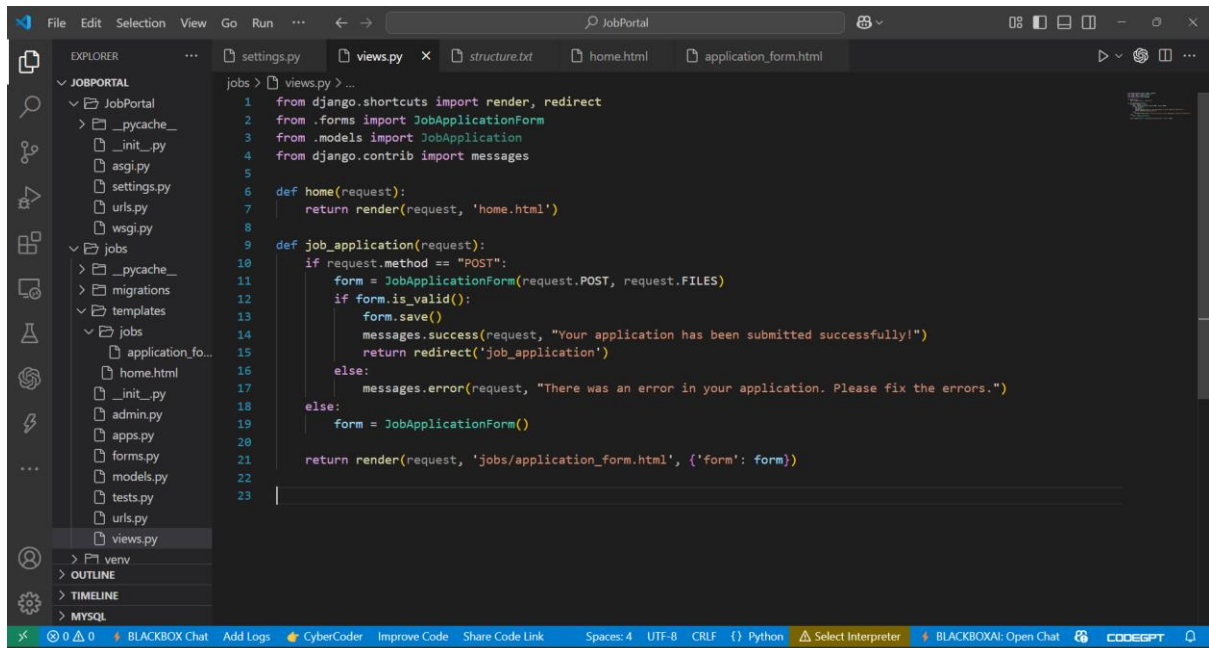
Step 3: Creating Models (jobs/models.py):



Step 4: Applying Migrations

---python manage.py makemigrations

---python manage.py migrate

Step 5: Creating Forms (jobs/forms.py):**Step 6: Creating Views (jobs/views.py):**

Step 7: Creating Templates

Home Page (templates/home.html):

```

1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Welcome to Job Portal</title>
7   <style>
8     body {
9       background: linear-gradient(to bottom, #e6ffe6, beige, skyblue);
10      font-family: Arial, sans-serif; height: 100vh; display: flex; align-items: center;
11      justify-content: center; margin: 0;
12    }
13    .container {
14      max-width: 400px; padding: 15px; text-align: center;
15      border-radius: 8px; background: #f9f9f9; box-shadow: 0 0 8px rgba(0, 128, 0, 0.2);
16    }
17    h2 { color: green; }
18    a {
19      display: inline-block; text-decoration: none;
20      background: green; color: white; padding: 8px 15px; border-radius: 5px; font-size: 14px;
21    }
22  </style>
23 </head>
24 <body>
25   <div class="container">
26     <h2>Welcome to the Job Portal</h2>
27     <p><a href="{% url 'job_application' %}">Apply for a Job</a></p>
28   </div>
29 </body>
30 </html>

```

Job Application Form (templates/jobs/application_form.html):

```

1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Job Application</title>
7   <style>
8     body {
9       background: linear-gradient(to bottom, #e6ffe6, beige, skyblue);
10      font-family: Arial, sans-serif; height: 100vh; display: flex;
11      align-items: center; justify-content: center; margin: 0;
12    }
13    .container {
14      width: 350px; padding: 15px; text-align: center;
15      border-radius: 8px; background: #f9f9f9; box-shadow: 0 0 8px rgba(0, 128, 0, 0.2);
16    }
17    .msg { padding: 8px; border-radius: 5px; color: white; }
18    .success { background: green; }
19    .error { background: red; }
20    form { display: flex; flex-direction: column; gap: 10px; }
21    button {
22      background: green; color: white; padding: 8px;
23      border: none; border-radius: 5px; cursor: pointer;
24    }
25  </style>
26 </head>

```

```

28 <body>
29 <div class="container">
30 <h2 style="color: green;">Apply Now</h2>
31 {% if messages %}
32 {% for message in messages %}
33 <p class="msg" {% if message.tags == 'success' %}success{% else %}error{% endif %}>{{ message }}</p>
34 {% endfor %}
35 {% endif %}
36 <form method="POST" enctype="multipart/form-data">
37 <input type="checkbox" name="csrf_token" value="{{ csrf_token }}">
38 <input type="text" name="form.as_p" value="{{ form.as_p }}">
39 <button type="submit">Submit</button>
40 </form>
41 </div>
42 </body>
43 </html>

```

Step 8: Configuring URLs

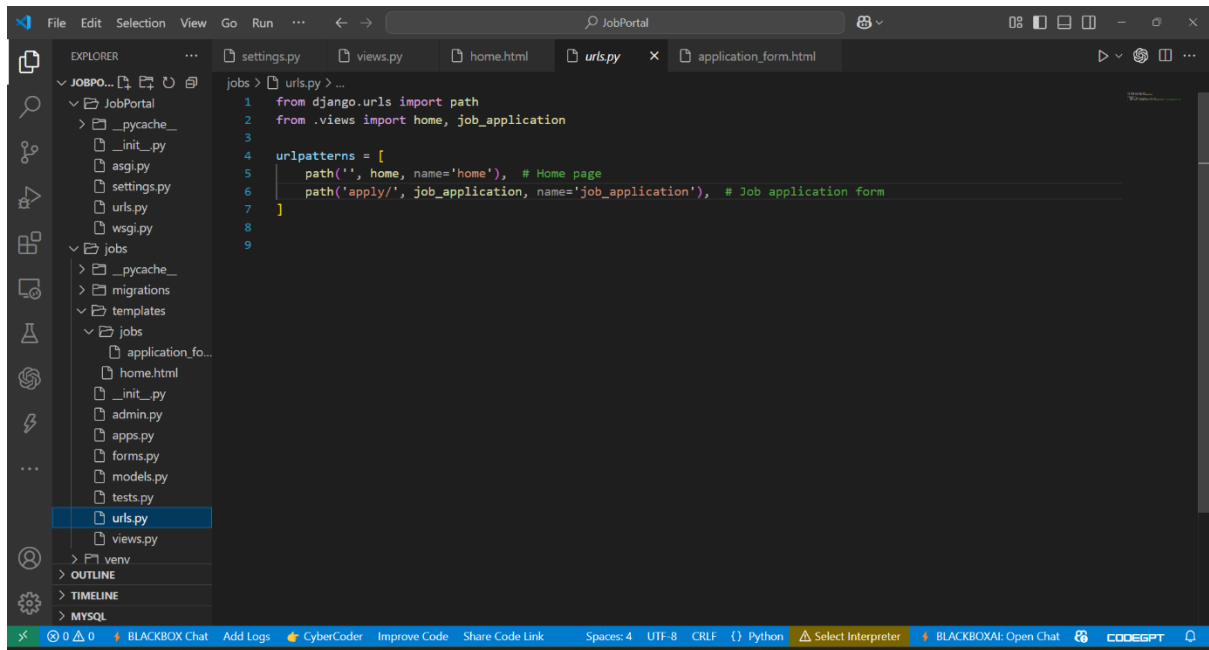
- Project URLs (JobPortal/urls.py):

```

1 """
2 URL configuration for JobPortal project.
3
4 The 'urlpatterns' list routes URLs to views. For more information please see:
5 https://docs.djangoproject.com/en/5.1/topics/http/urls/
6 Examples:
7 Function views
8 1. Add an import: from my_app import views
9 2. Add a URL to urlpatterns: path('', views.home, name='home')
10 Class-based views
11 1. Add an import: from other_app.views import Home
12 2. Add a URL to urlpatterns: path('', Home.as_view(), name='home')
13 Including another URLconf
14 1. Import the include() function: from django.urls import include, path
15 2. Add a URL to urlpatterns: path('blog/', include('blog.urls'))
16 """
17 from django.contrib import admin
18 from django.urls import path, include
19
20 urlpatterns = [
21     path('admin/', admin.site.urls),
22     path('', include('jobs.urls')),
23 ]
24
25
26

```

- **App URLs (jobs/urls.py)**

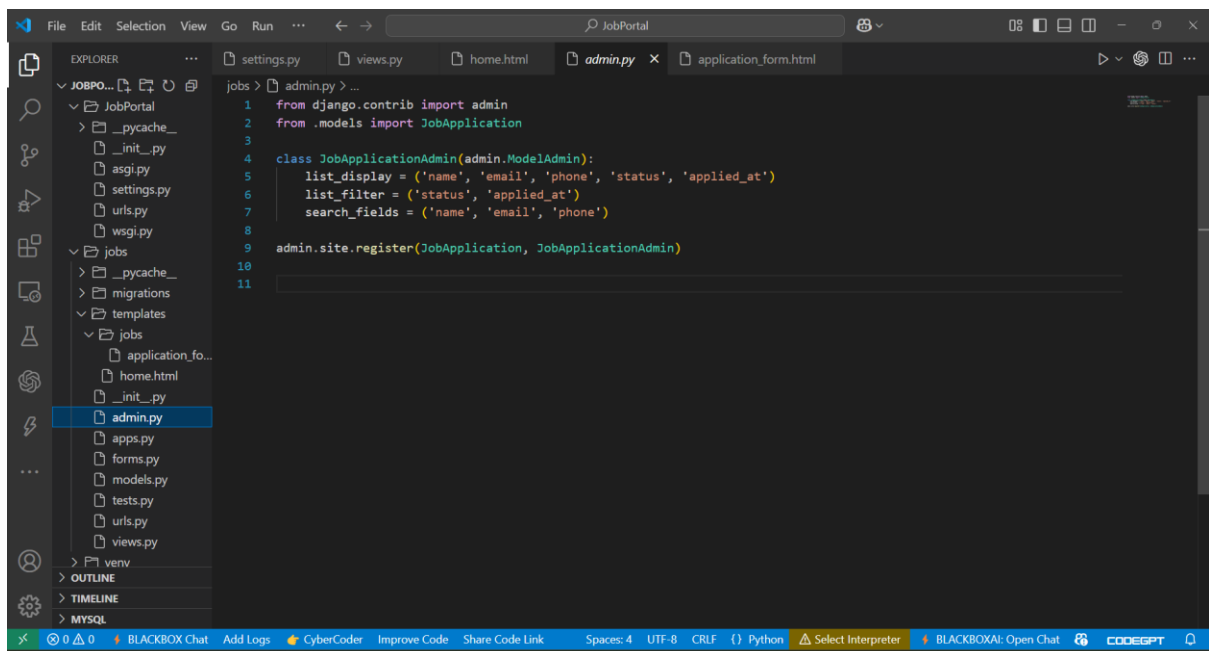


```

1 from django.urls import path
2 from .views import home, job_application
3
4 urlpatterns = [
5     path('', home, name='home'), # Home page
6     path('apply/', job_application, name='job_application'), # Job application form
7 ]
8
9

```

Step 9: Setting Up Admin Panel (jobs/admin.py):



```

1 from django.contrib import admin
2 from .models import JobApplication
3
4 class JobApplicationAdmin(admin.ModelAdmin):
5     list_display = ('name', 'email', 'phone', 'status', 'applied_at')
6     list_filter = ('status', 'applied_at')
7     search_fields = ('name', 'email', 'phone')
8
9 admin.site.register(JobApplication, JobApplicationAdmin)
10
11

```

Step 10: Creating a Superuser

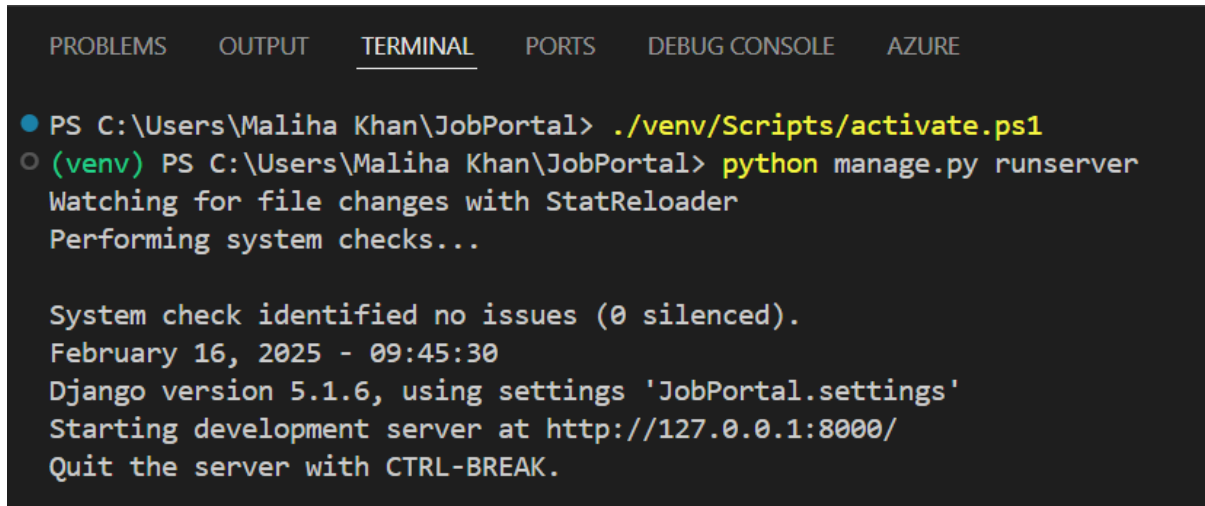
---python manage.py createsuperuser

Provide details like username, email, and password when prompted.

Step 11: Running the Development Server

```
---python manage.py runserver
```

Access the project at <http://127.0.0.1:8000/>



```
PROBLEMS  OUTPUT  TERMINAL  PORTS  DEBUG CONSOLE  AZURE

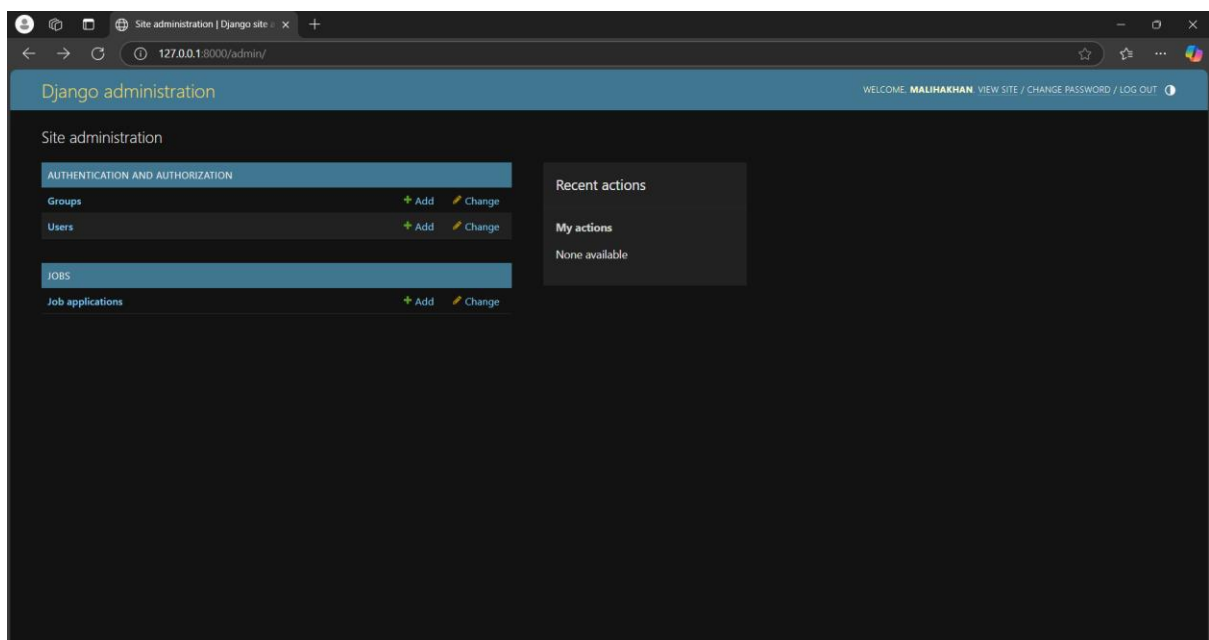
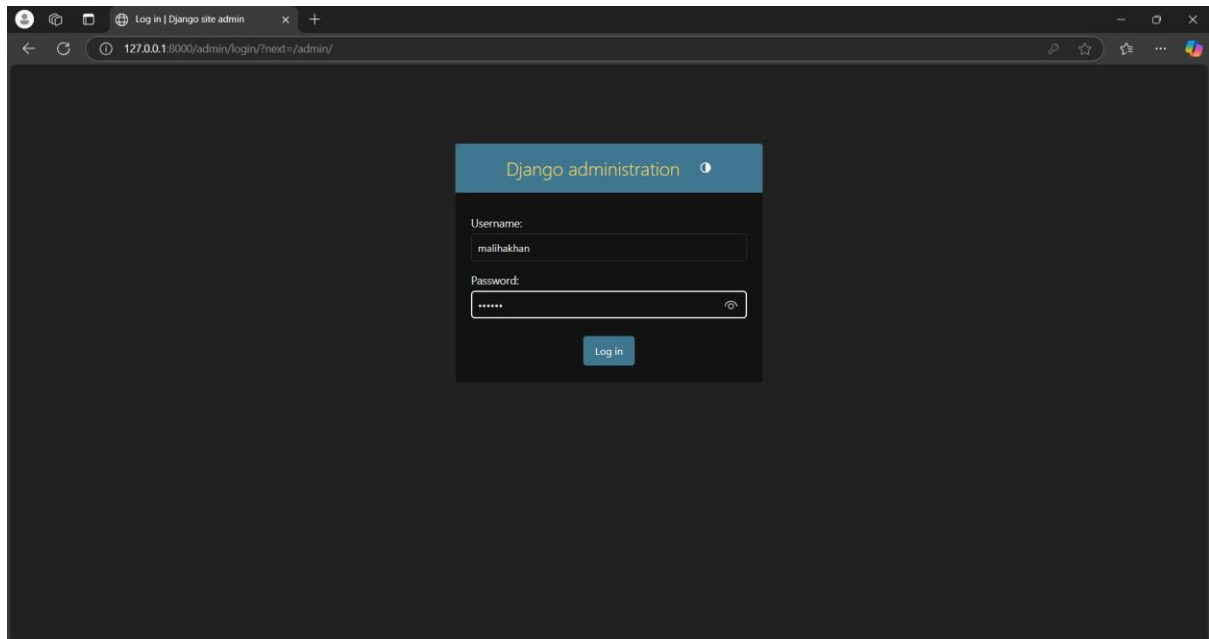
● PS C:\Users\Maliha Khan\JobPortal> ./venv/Scripts/activate.ps1
○ (venv) PS C:\Users\Maliha Khan\JobPortal> python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

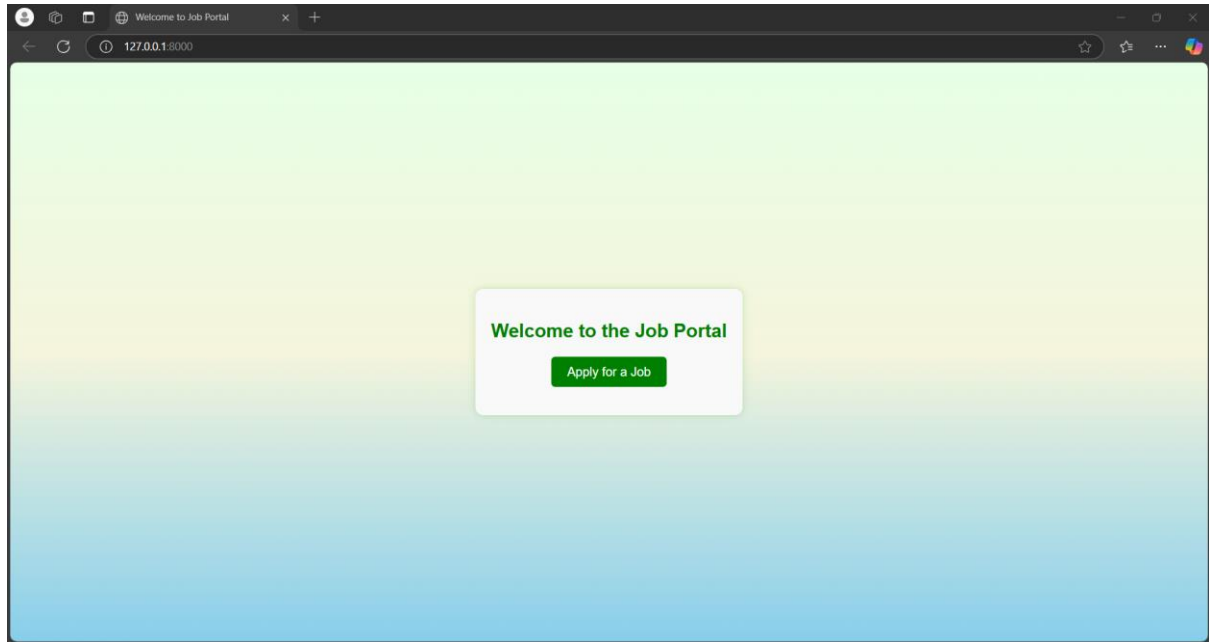
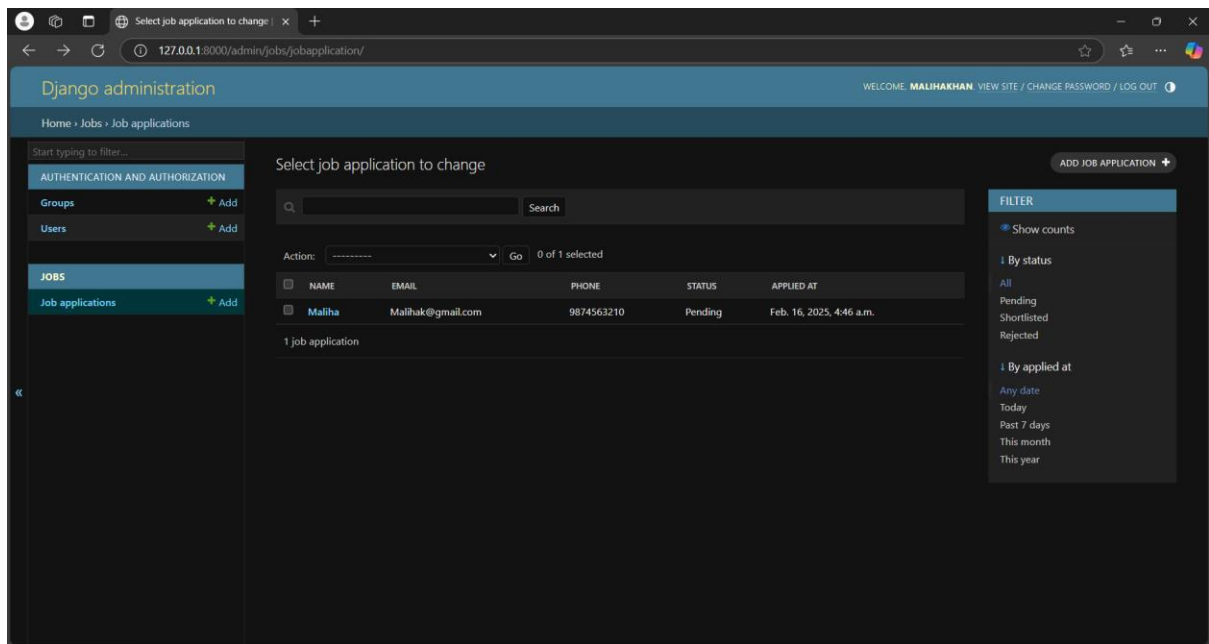
System check identified no issues (0 silenced).
February 16, 2025 - 09:45:30
Django version 5.1.6, using settings 'JobPortal.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

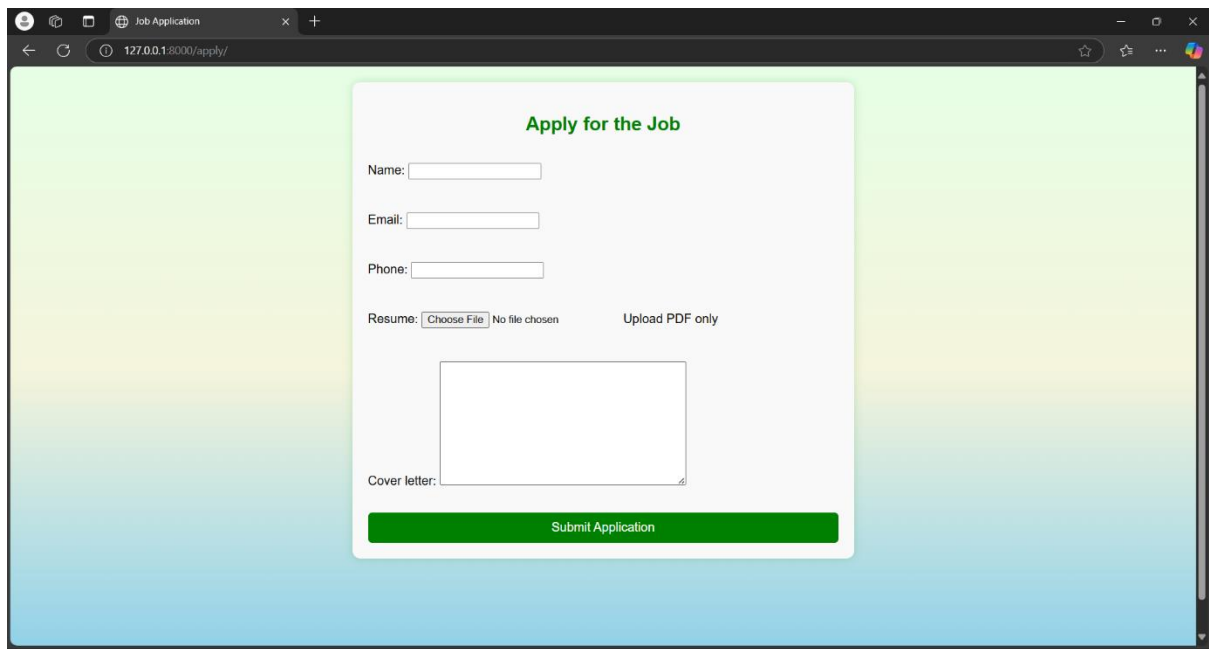
Step 12: Testing the Application

1. Open the **Home Page** at <http://127.0.0.1:8000/>.
2. Click **Apply for a Job** and fill out the application.
3. Submit the form and verify the stored data in **Admin Panel** (<http://127.0.0.1:8000/admin/>).

Output:







The screenshot shows a web browser window with a single tab titled 'Job Application'. The address bar displays '127.0.0.1:8000/apply/'. The main content area features a light green gradient background. Centered on this background is a white rectangular form titled 'Apply for the Job' in green text. The form contains the following fields and elements:

- Name:** A text input field.
- Email:** A text input field.
- Phone:** A text input field.
- Resume:** A file upload section with a 'Choose File' button, the text 'No file chosen', and the instruction 'Upload PDF only'.
- Cover letter:** A large text area for writing the cover letter.
- Submit Application:** A prominent green button at the bottom of the form.