

SENA_APP

YEIMY TATIANA BAYPONA PÉREZ

SERVICIO NACIONAL DE APRENDIZAJE
CENTRO MINERO

ANALISIS Y DESARROLLO DE SOFTWARE (ADSO)

FICHA: 2993648

ANDRES FELIPE SANDOVAL HIGUERA

SOGAMOSO BOYACÁ
NOVIEMBRE
2025

Tabla de contenido

INTRODUCCIÓN	4
SENA APP.....	5
1. Configuración Inicial.....	5
• Crear un entorno virtual nuevo.....	5
• Activar el entorno virtual.....	5
• Instalar Django sobre en entorno virtual	5
• Listamos lo que se instalo	6
• Crear el proyecto	6
• Crear aplicación	6
• Configurar el archivo settings.py	7
2. Desarrollo del Modelo.....	7
• Implementar la clase Aprendiz con todos los campos solicitados	7
• Agregar métodos adicionales como: def __str__(self):	8
3. Ejecutar migraciones	8
• Crear la base de datos	8
• Base de datos creada en el proyecto	9
• Migramos la base de datos	10
• Verificar que se creó la base de datos en SQLite	10
4. Panel Administrativo.....	11
• Crear superusuario	11
• Ingresar al panel administrativo	11
• Panel administrativo Django	12
• Registrar el modelo en admin.py	12
• Mostrar aprendiz en panel administrativo	13
5. Vistas y Plantillas.....	13
• Registrar aprendices en la base de datos	13
• Crear las plantillas HTML en Templates	14
• Plantilla main.html.....	15

•	Plantilla master.html.....	15
6.	Implementación de Bootstrap	16
•	Instalar Bootstrap.....	16
•	Agregar en settings.py Bootstrap.....	16
•	Agregar Bootstrap a la plantilla master.html.....	17
•	Implementar la función aprendices en views.py	17
•	Configurar URLs correctamente	18
7.	Pruebas	19
•	Acceder al panel administrativo	19
•	Panel administrativo.....	19
•	Verificar que la lista se muestre correctamente	20
8.	Código de los Template	21
•	Template de aprendices_list	21
•	Template de detalle_aprendiz	21
•	Template main.html	22
•	Template master.html	22
9.	Pagina	23
•	Inicio	23
•	Aprendices	23
•	Detalle aprendiz.....	24
	Repositorio GitHub	24
	CONCLUSIONES	25

INTRODUCCIÓN

El desarrollo de **SENA APP** representa un ejercicio práctico de construcción de una aplicación web utilizando el framework **Django**, con el propósito de comprender y aplicar los conceptos fundamentales de la creación de proyectos estructurados bajo una arquitectura profesional. A lo largo de este trabajo se realiza el proceso completo de configuración del entorno de desarrollo, creación del proyecto y la aplicación, definición de modelos, migraciones, así como la implementación del panel administrativo integrado que ofrece **Django**.

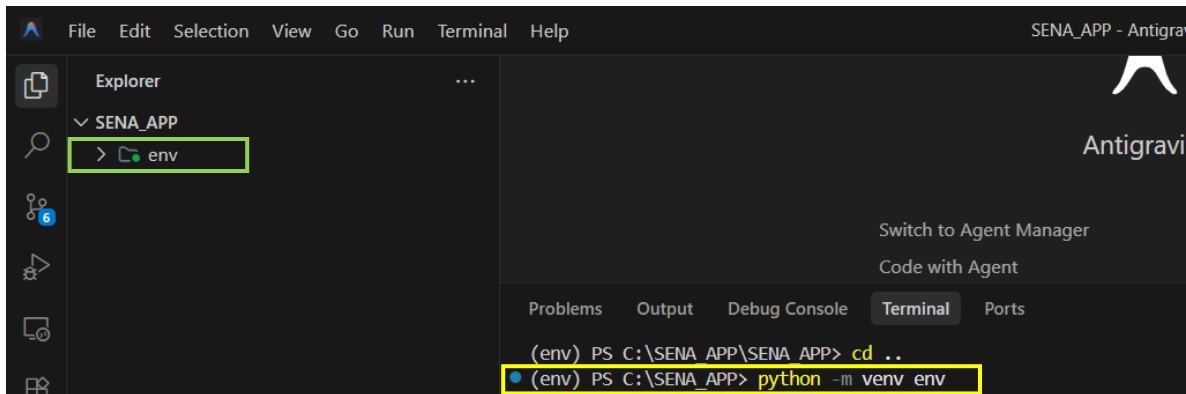
Este documento también detalla la configuración del archivo **settings.py** la creación de plantillas **HTML**, la integración de **Bootstrap** para mejorar la interfaz gráfica y la construcción de vistas funcionales que permiten mostrar información almacenada en la base de datos.

En conjunto este proyecto busca fortalecer las habilidades en el desarrollo de software con tecnologías modernas, fomentar el buen manejo de entornos virtuales y promover el uso de herramientas que facilitan la gestión de datos y la creación de interfaces dinámicas.

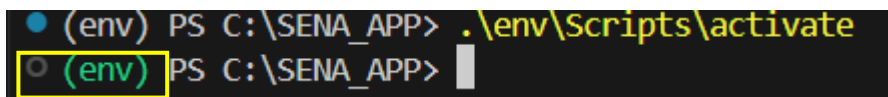
SENA APP

1. Configuración Inicial

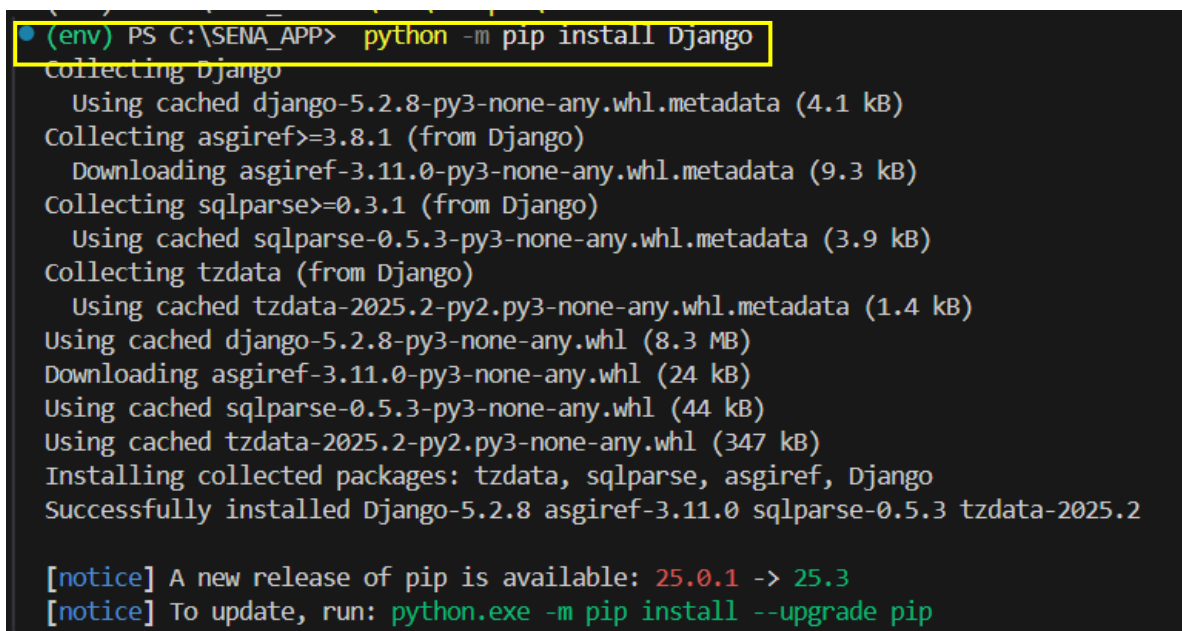
- Crear un entorno virtual nuevo



- Activar el entorno virtual



- Instalar Django sobre en entorno virtual

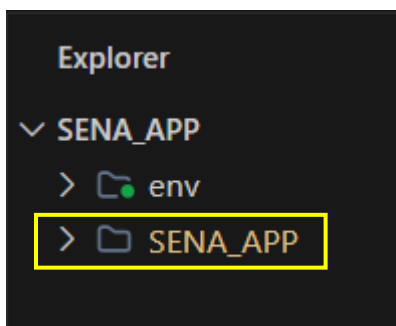


- Listamos lo que se instalo

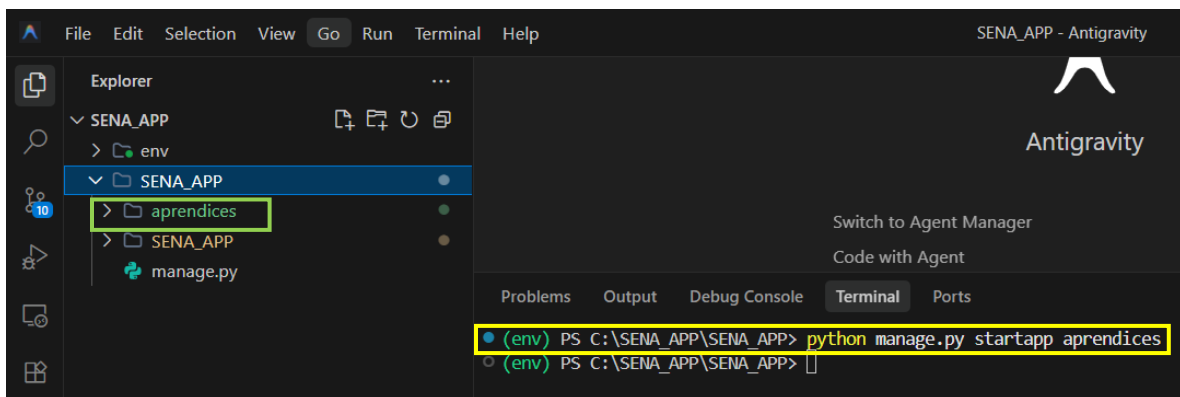
```
(env) PS C:\SENA_APP> pip list
Package Version
-----
asgiref 3.11.0
Django 5.2.8
pip 25.0.1
sqlparse 0.5.3
tzdata 2025.2
```

- Crear el proyecto

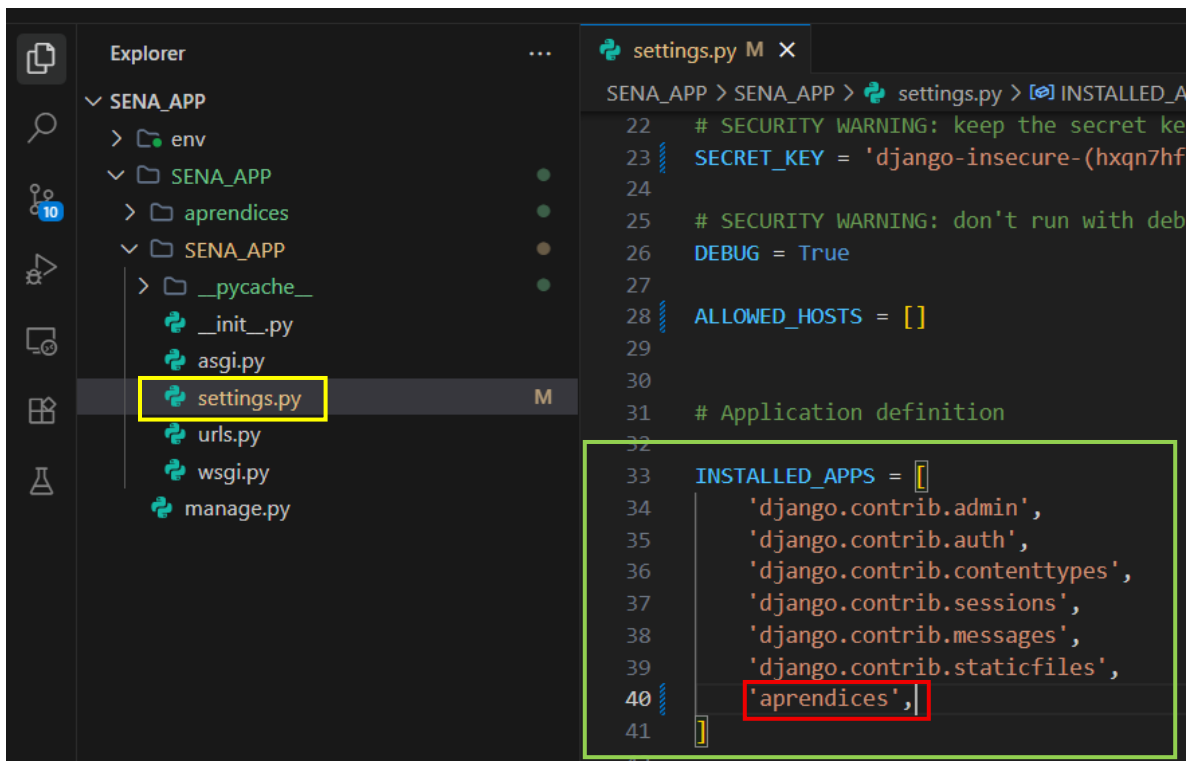
```
(env) PS C:\SENA_APP> django-admin startproject SENNA_APP
(env) PS C:\SENA_APP>
```



- Crear aplicación

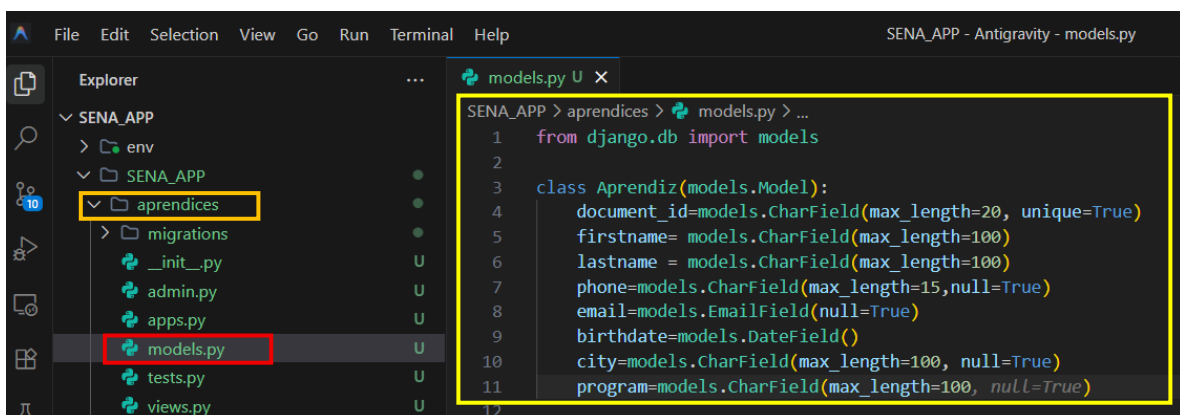


- Configurar el archivo settings.py

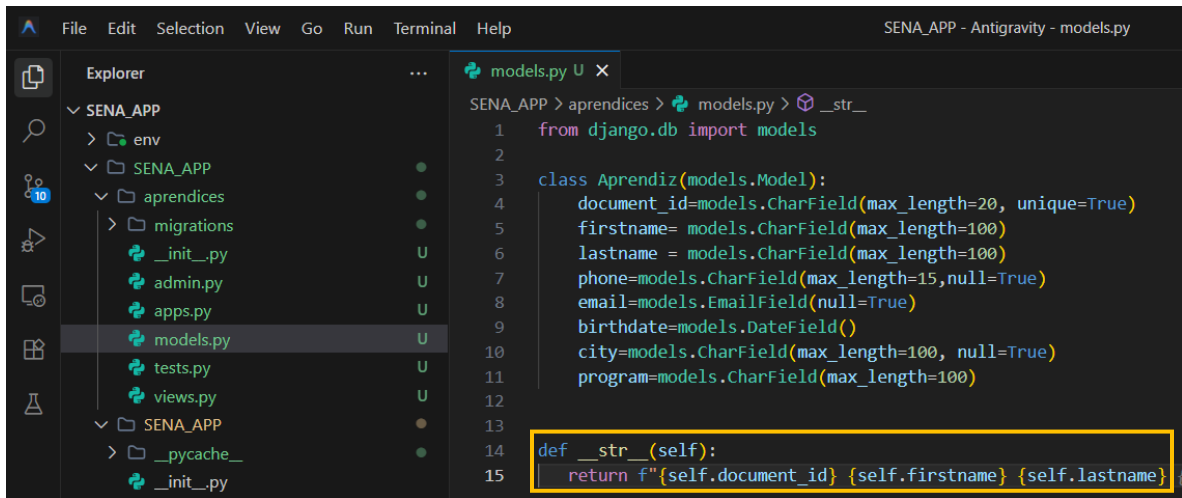


2. Desarrollo del Modelo

- Implementar la clase Aprendiz con todos los campos solicitados



- Agregar métodos adicionales como: `def __str__(self):`

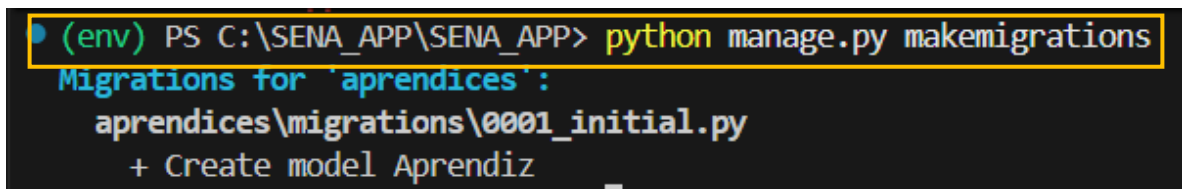


```
SENA_APP - AntigraVity - models.py

SENA_APP > aprendices > models.py > __str__
1  from django.db import models
2
3  class Aprendiz(models.Model):
4      document_id=models.CharField(max_length=20, unique=True)
5      firstname= models.CharField(max_length=100)
6      lastname = models.CharField(max_length=100)
7      phone=models.CharField(max_length=15,null=True)
8      email=models.EmailField(null=True)
9      birthdate=models.DateField()
10     city=models.CharField(max_length=100, null=True)
11     program=models.CharField(max_length=100)
12
13
14     def __str__(self):
15         return f"{self.document_id} {self.firstname} {self.lastname}"
```

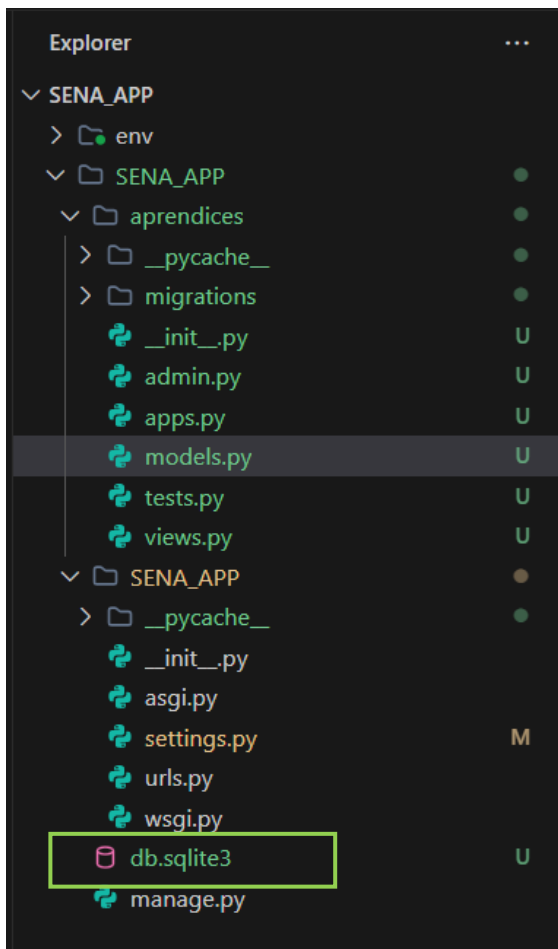
3. Ejecutar migraciones

- Crear la base de datos



```
(env) PS C:\SENA_APP\SENA_APP> python manage.py makemigrations
Migrations for 'aprendices':
  aprendices\migrations\0001_initial.py
  + Create model Aprendiz
```

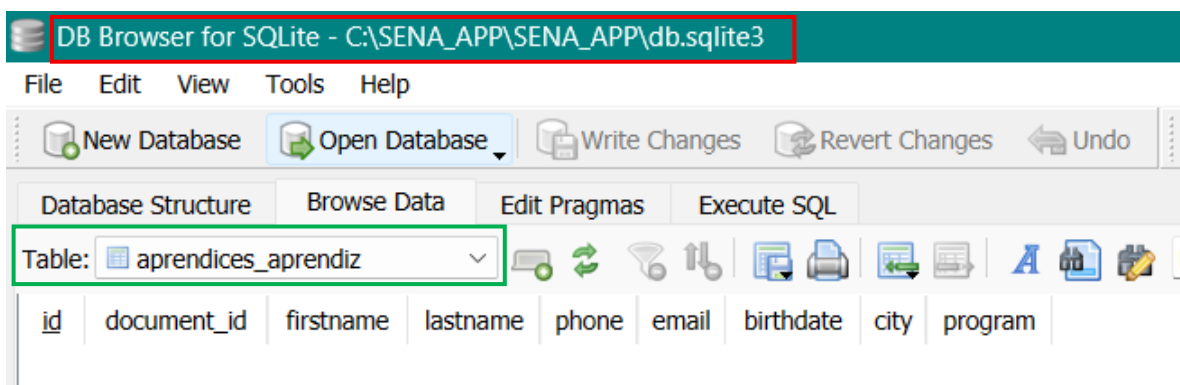
- Base de datos creada en el proyecto



- Migramos la base de datos

```
(env) PS C:\SENA_APP\SENA_APP> python manage.py migrate
Operations to perform:
  Apply all migrations: admin, aprendices, auth, contenttypes, sessions
Running migrations:
  Applying contenttypes.0001_initial... OK
  Applying auth.0001_initial... OK
  Applying admin.0001_initial... OK
  Applying admin.0002_logentry_remove_auto_add... OK
  Applying admin.0003_logentry_add_action_flag_choices... OK
  Applying aprendices.0001_initial... OK
  Applying contenttypes.0002_remove_content_type_name... OK
  Applying auth.0002_alter_permission_name_max_length... OK
  Applying auth.0003_alter_user_email_max_length... OK
  Applying auth.0004_alter_user_username_opts... OK
  Applying auth.0005_alter_user_last_login_null... OK
  Applying auth.0006_require_contenttypes_0002... OK
  Applying auth.0007_alter_validators_add_error_messages... OK
  Applying auth.0008_alter_user_username_max_length... OK
  Applying auth.0009_alter_user_last_name_max_length... OK
  Applying auth.0010_alter_group_name_max_length... OK
  Applying auth.0011_update_proxy_permissions... OK
  Applying auth.0012_alter_user_first_name_max_length... OK
  Applying sessions.0001_initial... OK
```

- Verificar que se creó la base de datos en SQLite

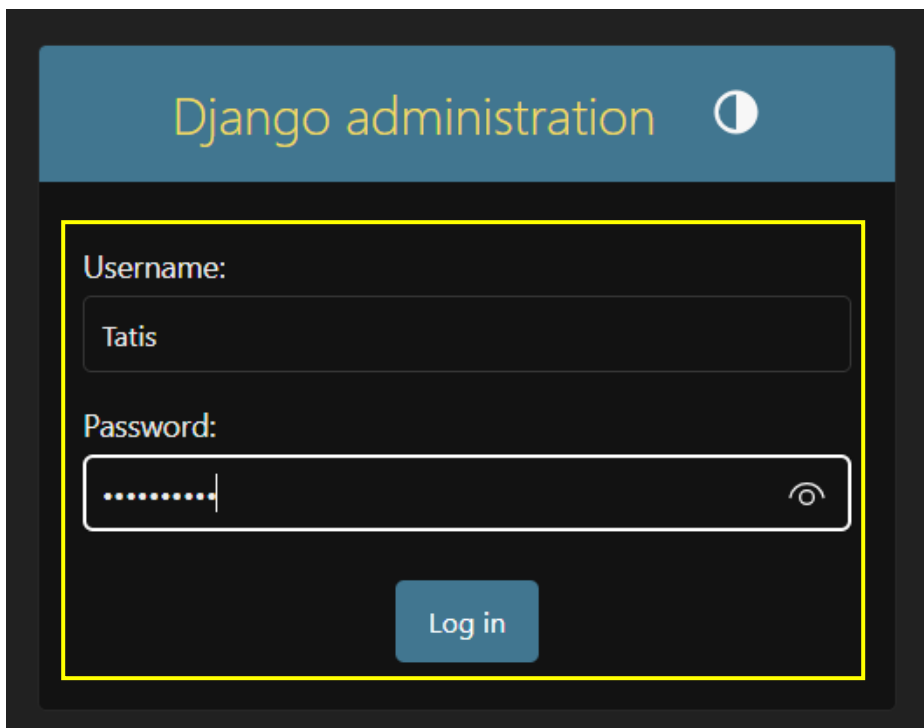


4. Panel Administrativo

- Crear superusuario

```
(env) PS C:\SENA_APP\SENA_APP> python .\manage.py createsuperuser  
Username (leave blank to use 'tatia'): Tatis  
Email address: tatis@gmail.com  
Password:  
Password (again):  
Superuser created successfully.
```

- Ingresar al panel administrativo



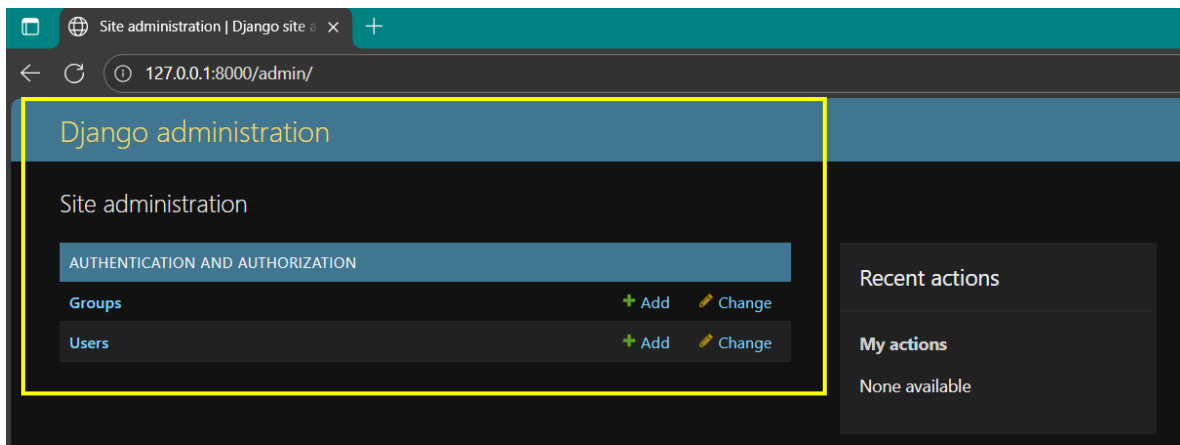
Django administration

Username:
Tatis

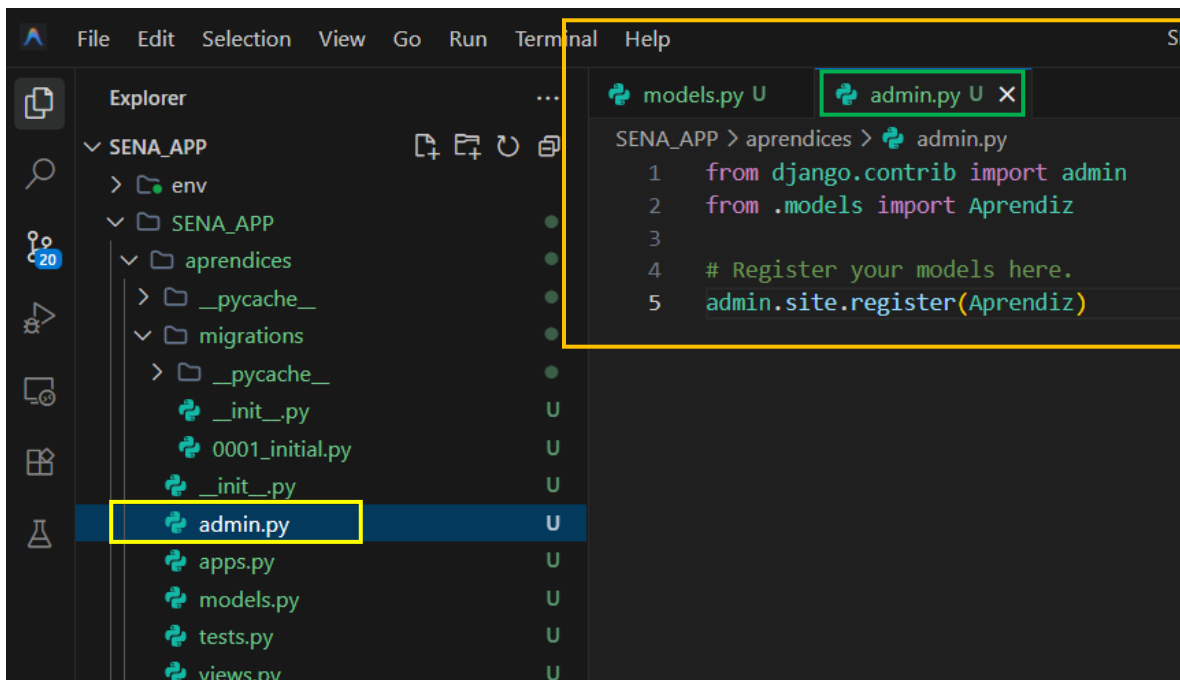
Password:
.....

Log in

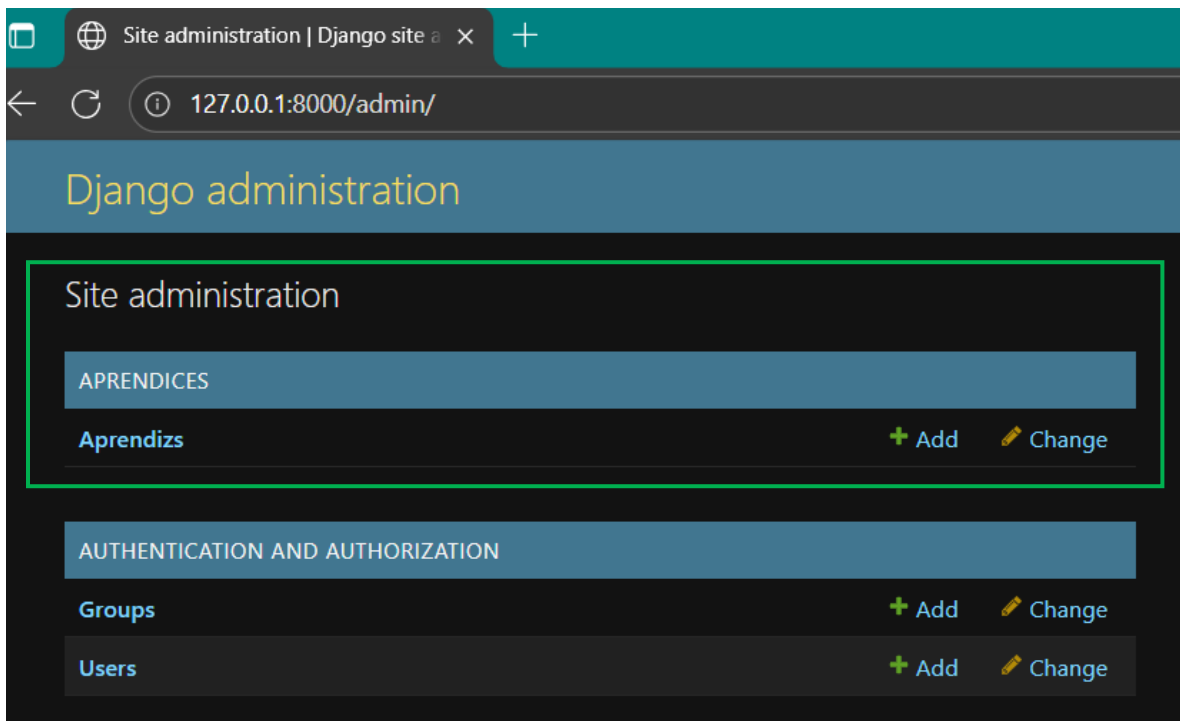
- **Panel administrativo Django**



- **Registrar el modelo en admin.py**



- **Mostrar aprendiz en panel administrativo**



5. Vistas y Plantillas

- **Registrar aprendices en la base de datos**

DB Browser for SQLite - C:\SENA_APP\SENA_APP\db.sqlite3

File Edit View Tools Help

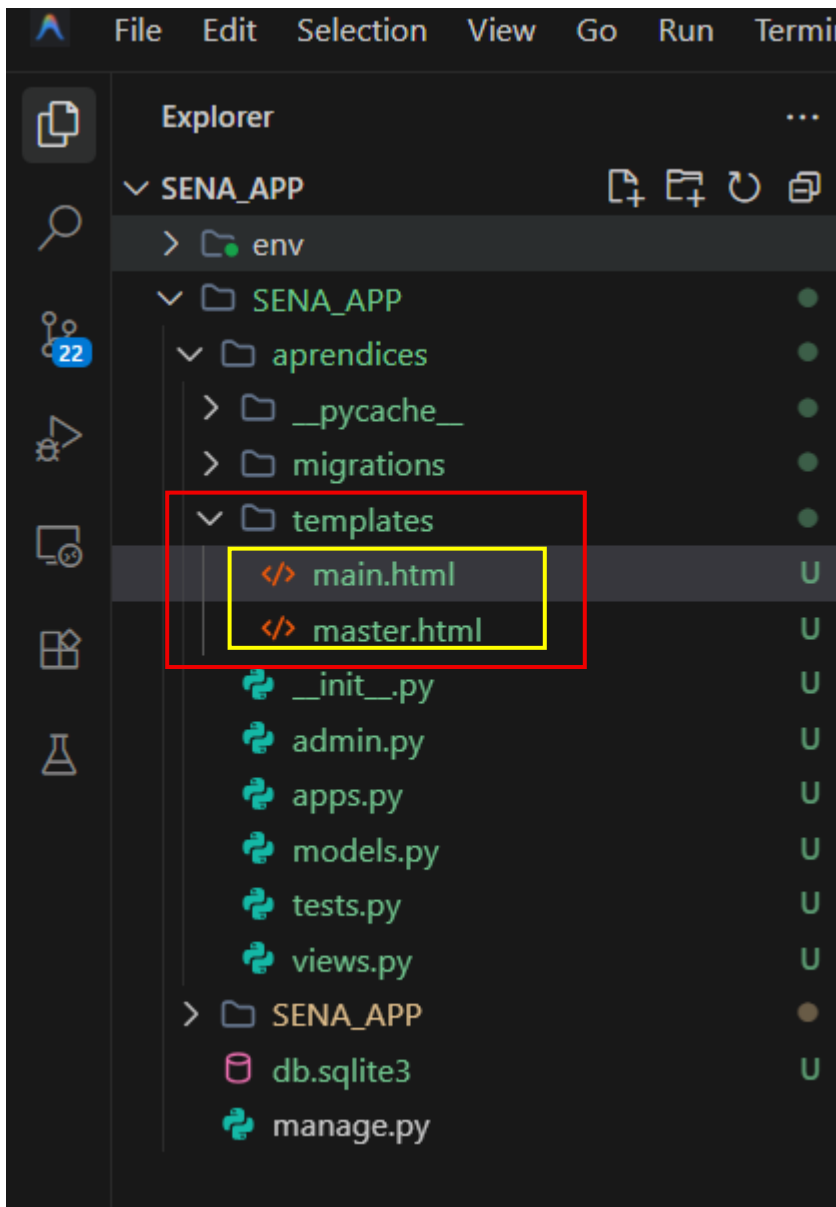
New Database Open Database Write Changes Revert Changes Undo Open Project Save Project Attach Database Close Database

Database Structure Browse Data Edit Pragmas Execute SQL

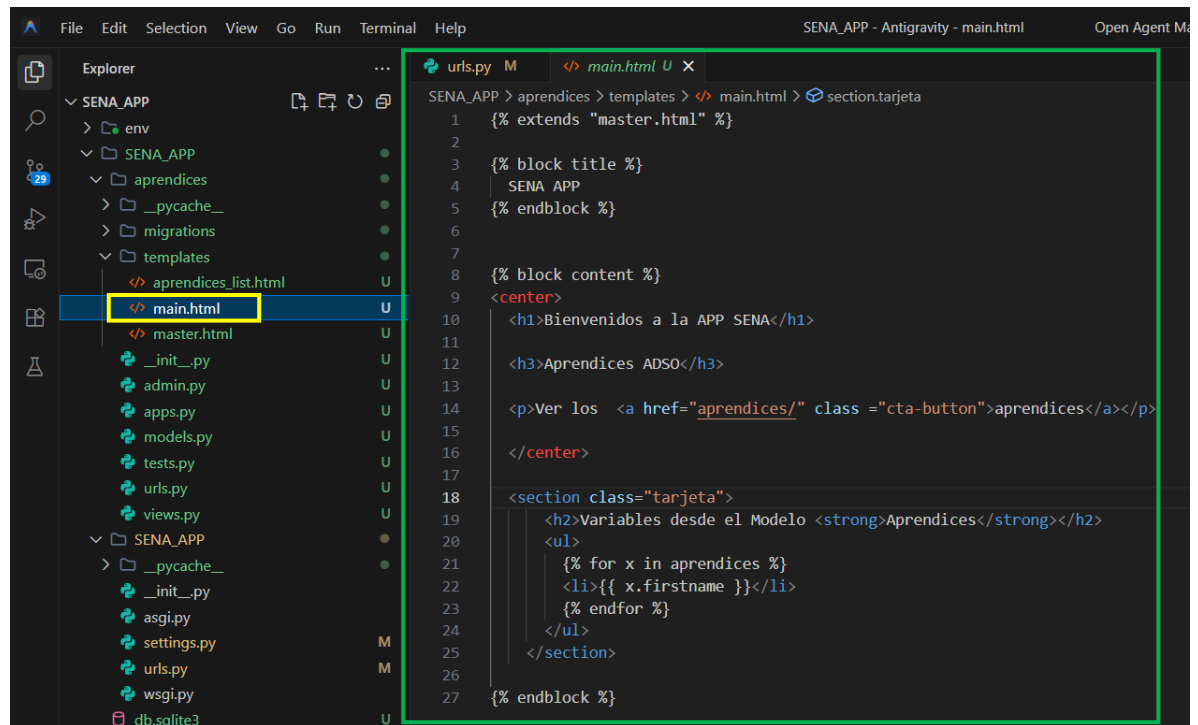
Table: aprendices_aprendiz

id	document_id	firstname	lastname	phone	email	birthdate	city	program
1	1056928954	Andres	Ardila	3223394432	andres@gmail.com	2006-12-25	Toca	ADSO
2	1002537360	Mayerly	Cardenas	3202915130	Mayerly@gmail.com	2003-02-03	Nobsa	ADSO
3	1098635221	Kamila	Vargas	3251556266	kamila@gmail.com	2006-05-16	Duitama	ADSO
4	1052312624	Jimmy	Angarita	3134872036	jimmy@gmail.com	2006-01-11	Belen	ADSO
5	1057579055	Yamith	Rosas	3223726878	yamith@gmail.com	2006-08-27	Sogamoso	ADSO

- Crear las plantillas HTML en Templates

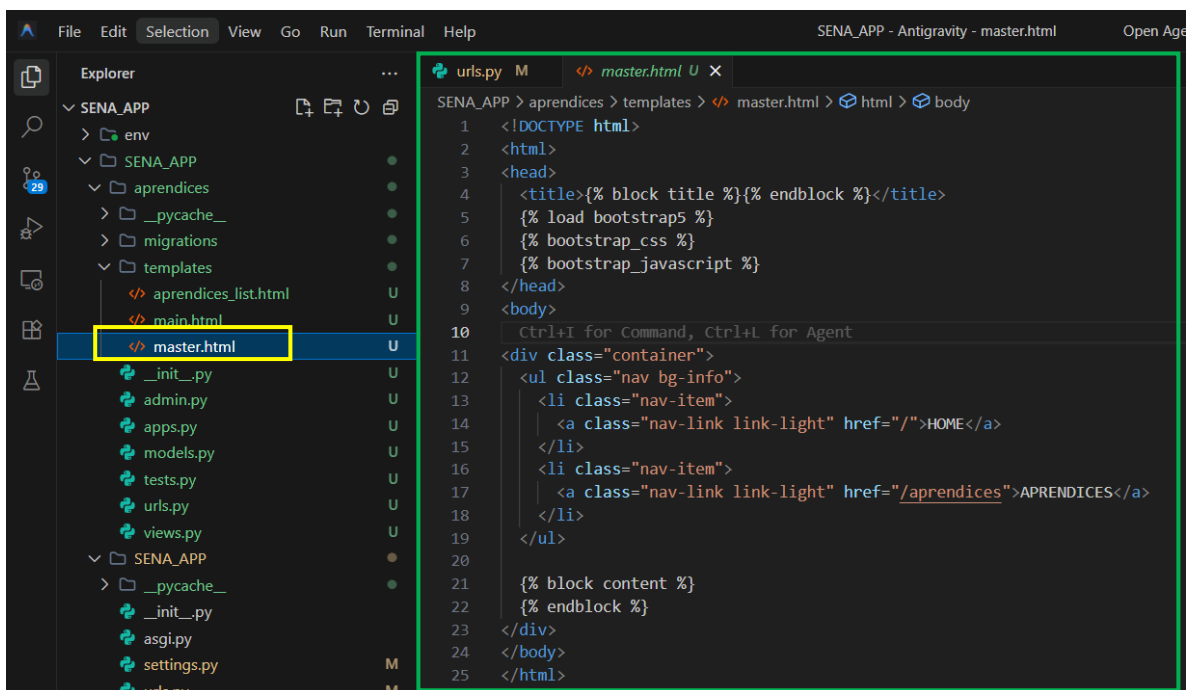


- Plantilla main.html



```
SENA_APP > aprendices > templates > main.html > section.tarjeta
1 {% extends "master.html" %}
2
3 {% block title %}
4     SENA APP
5 {% endblock %}
6
7
8 {% block content %}
9     <center>
10         <h1>Bienvenidos a la APP SENA</h1>
11         <h3>Aprendices ADSO</h3>
12         <p>Ver los <a href="/aprendices/" class="cta-button">aprendices</a></p>
13     </center>
14
15     <section class="tarjeta">
16         <h2>Variables desde el Modelo <strong>Aprendices</strong></h2>
17         <ul>
18             {% for x in aprendices %}
19                 <li>{{ x.firstname }}</li>
20             {% endfor %}
21         </ul>
22     </section>
23
24 {% endblock %}
```

- Plantilla master.html



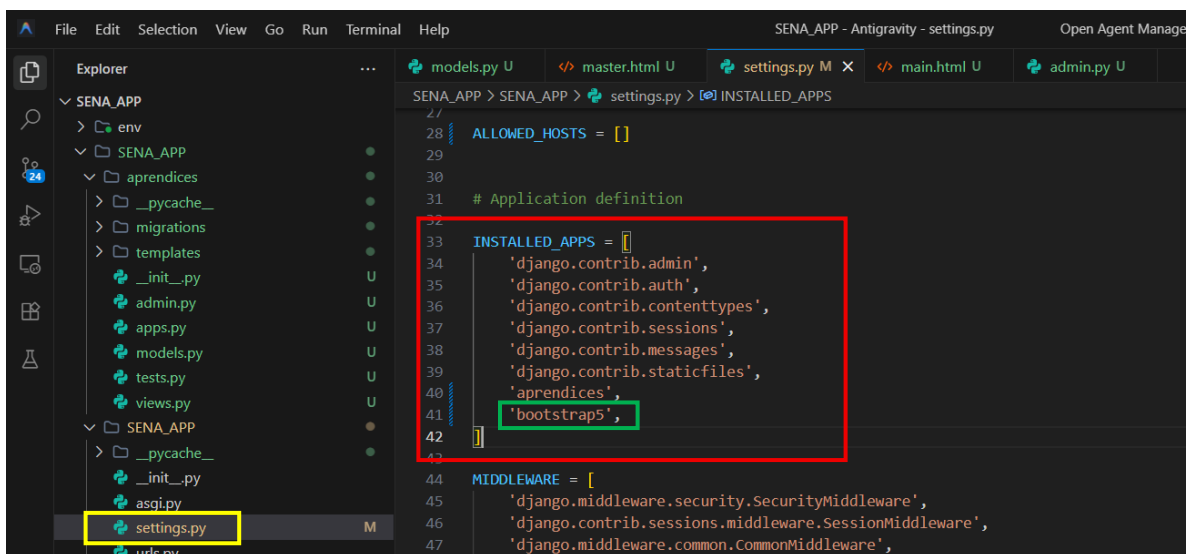
```
SENA_APP > aprendices > templates > master.html > html > body
1 <!DOCTYPE html>
2 <html>
3 <head>
4     <title>{% block title %}{% endblock %}</title>
5     {% load bootstrap5 %}
6     {% bootstrap_css %}
7     {% bootstrap_javascript %}
8 </head>
9 <body>
10     Ctrl+I for Command, Ctrl+L for Agent
11 <div class="container">
12     <ul class="nav bg-info">
13         <li class="nav-item">
14             <a class="nav-link link-light" href="/">HOME</a>
15         </li>
16         <li class="nav-item">
17             <a class="nav-link link-light" href="/aprendices">APRENDICES</a>
18         </li>
19     </ul>
20
21     {% block content %}
22     {% endblock %}
23 </div>
24 </body>
25 </html>
```

6. Implementación de Bootstrap

- Instalar Bootstrap

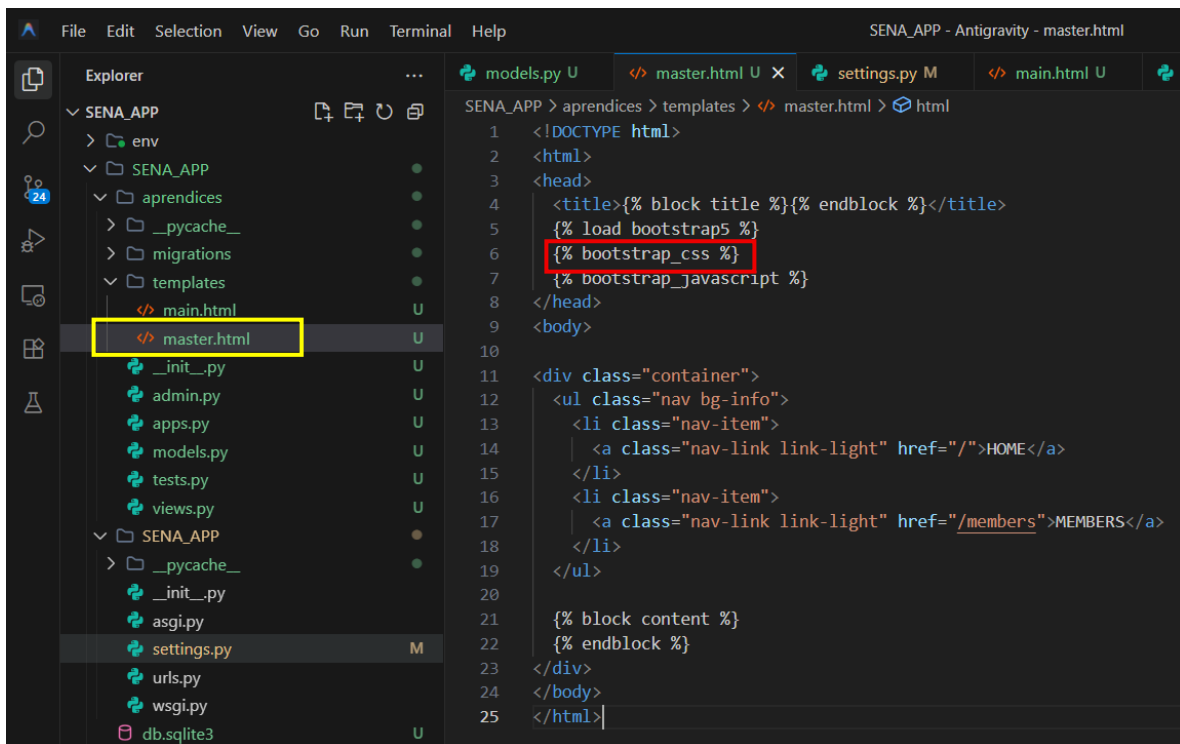
```
(env) PS C:\SENA_APP\SENA_APP> pip install django-bootstrap-v5
Collecting django-bootstrap-v5
  Downloading django_bootstrap_v5-1.0.11-py3-none-any.whl.metadata (4.2 kB)
Collecting beautifulsoup4<5.0.0,>=4.8.0 (from django-bootstrap-v5)
  Downloading beautifulsoup4-4.14.2-py3-none-any.whl.metadata (3.8 kB)
Collecting django<5.0,>=2.2 (from django-bootstrap-v5)
  Downloading django-4.2.26-py3-none-any.whl.metadata (4.2 kB)
Collecting soupsieve>1.2 (from beautifulsoup4<5.0.0,>=4.8.0->django-bootstrap-v5)
  Downloading soupsieve-2.8-py3-none-any.whl.metadata (4.6 kB)
Collecting typing-extensions>=4.0.0 (from beautifulsoup4<5.0.0,>=4.8.0->django-bootstrap-v5)
  Using cached typing_extensions-4.15.0-py3-none-any.whl.metadata (3.3 kB)
Requirement already satisfied: asgiref<4,>=3.6.0 in c:\sena_app\env\lib\site-packages (from django-bootstrap-v5) (3.11.0)
Requirement already satisfied: sqlparse>=0.3.1 in c:\sena_app\env\lib\site-packages (from django-bootstrap-v5) (0.5.3)
Requirement already satisfied: tzdata in c:\sena_app\env\lib\site-packages (from django-bootstrap-v5) (2025.2)
Downloading django_bootstrap_v5-1.0.11-py3-none-any.whl (24 kB)
Downloading beautifulsoup4-4.14.2-py3-none-any.whl (106 kB)
Downloading django-4.2.26-py3-none-any.whl (8.0 MB)
8.0/8.0 MB 5.8 MB/s eta 0:00:00
Downloading soupsieve-2.8-py3-none-any.whl (36 kB)
```

- Agregar en settings.py Bootstrap



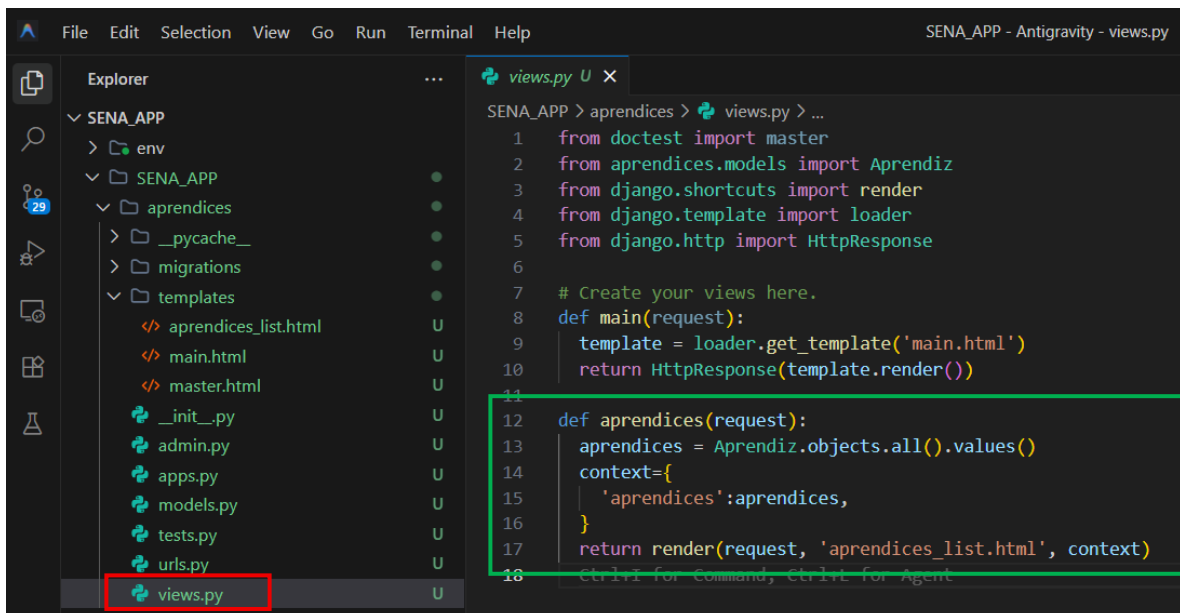
```
SENA_APP > SENA_APP > settings.py > INSTALLED_APPS
28 ALLOWED_HOSTS = []
29
30
31 # Application definition
32
33 INSTALLED_APPS = [
34     'django.contrib.admin',
35     'django.contrib.auth',
36     'django.contrib.contenttypes',
37     'django.contrib.sessions',
38     'django.contrib.messages',
39     'django.contrib.staticfiles',
40     'aprendices',
41     'bootstrap5',
42 ]
43
44 MIDDLEWARE = [
45     'django.middleware.security.SecurityMiddleware',
46     'django.contrib.sessions.middleware.SessionMiddleware',
47     'django.middleware.common.CommonMiddleware',
```

- Agregar Bootstrap a la plantilla master.html



```
1 <!DOCTYPE html>
2 <html>
3 <head>
4     <title>{% block title %}{% endblock %}</title>
5     {% load bootstrap5 %}
6     {% bootstrap_css %}
7     {% bootstrap_javascript %}
8 </head>
9 <body>
10
11 <div class="container">
12     <ul class="nav bg-info">
13         <li class="nav-item">
14             <a class="nav-link link-light" href="/">HOME</a>
15         </li>
16         <li class="nav-item">
17             <a class="nav-link link-light" href="/members">MEMBERS</a>
18         </li>
19     </ul>
20
21     {% block content %}
22     {% endblock %}
23 </div>
24 </body>
25 </html>
```

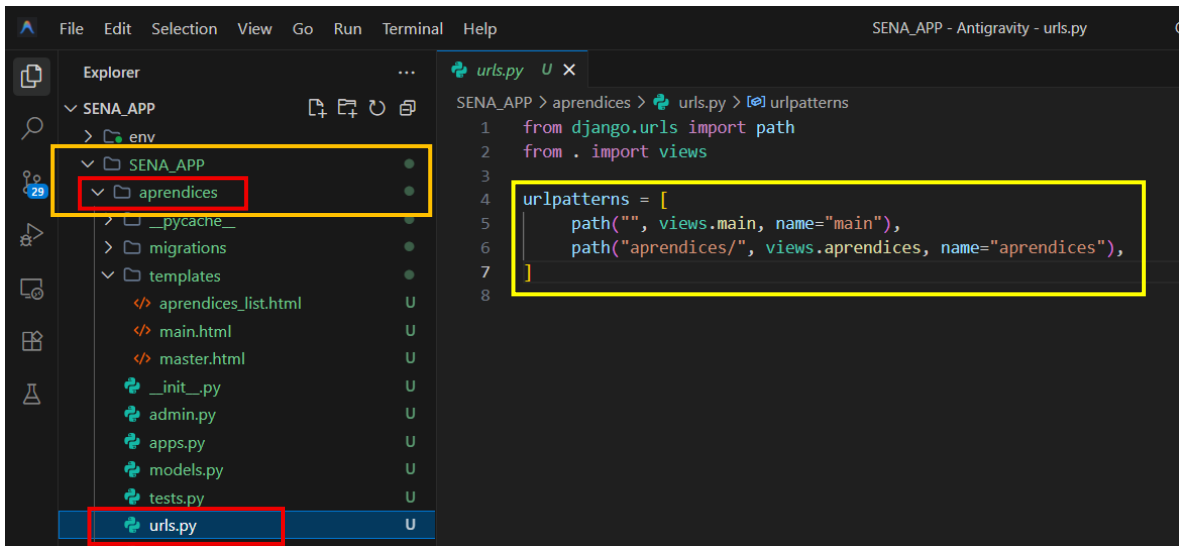
- Implementar la función aprendices en views.py



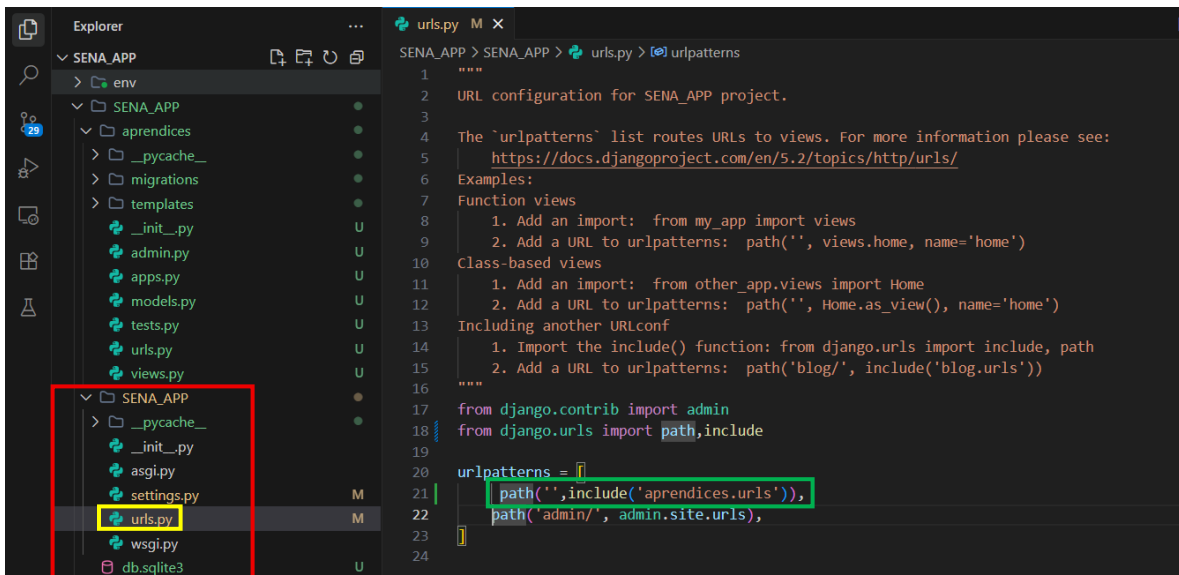
```
1 from doctest import master
2 from aprendices.models import Aprendiz
3 from django.shortcuts import render
4 from django.template import loader
5 from django.http import HttpResponse
6
7 # Create your views here.
8 def main(request):
9     template = loader.get_template('main.html')
10    return HttpResponse(template.render())
11
12 def aprendices(request):
13     aprendices = Aprendiz.objects.all().values()
14     context={
15         'aprendices':aprendices,
16     }
17     return render(request, 'aprendices_list.html', context)
```

- Configurar URLs correctamente

➤ *URL De la aplicación*

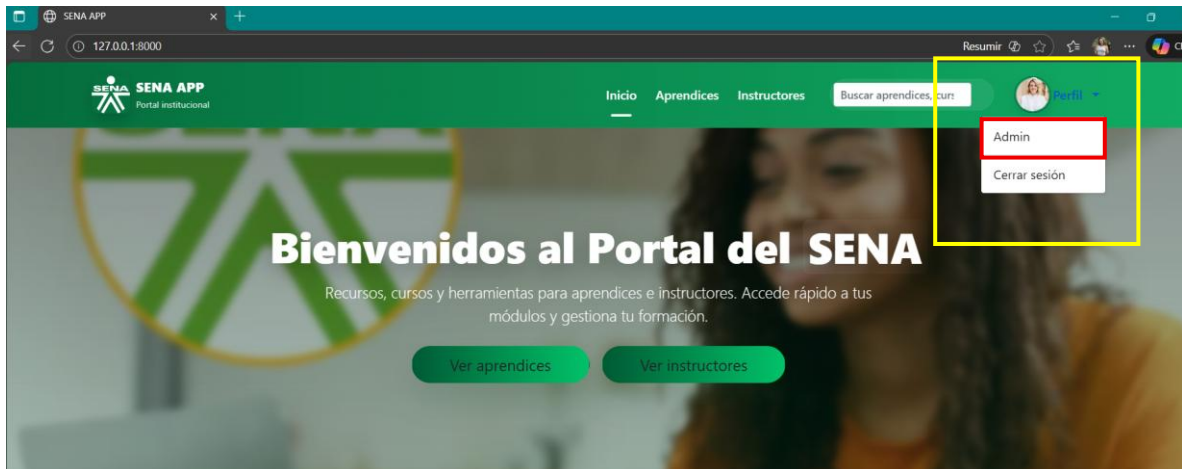


➤ *URL Del proyecto*

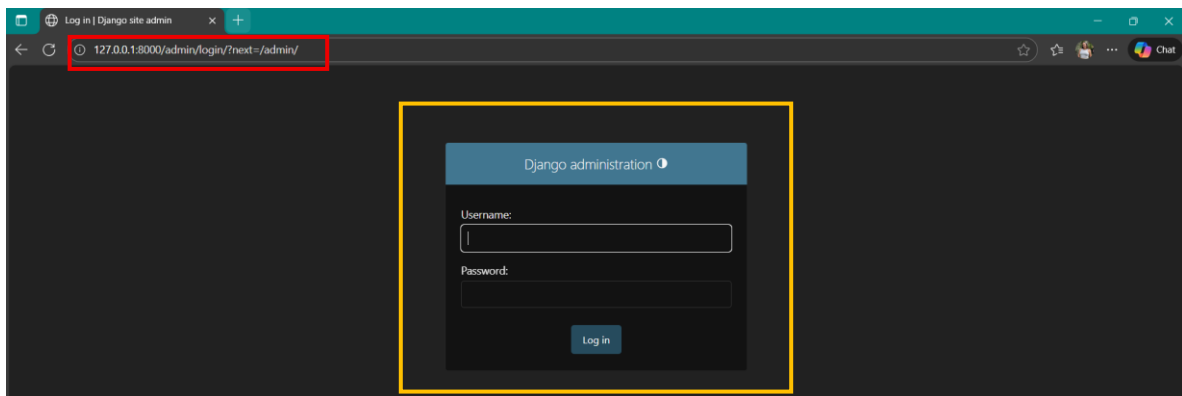


7. Pruebas


- Acceder al panel administrativo




- Panel administrativo



- Verificar que la lista se muestre correctamente

SENA APP
Portal institucional

[Inicio](#) [Aprendices](#) [Instructores](#)

 [Perfil](#)

Lista de Aprendices

Consulta rápida y organizada de los aprendices registrados.

DOCUMENTO	NOMBRE	APELLIDO	PROGRAMA
1056928954	Andres	Ardila	ADSO
1002537360	Mayerly	Cardenas	ADSO
1098635221	Kamila	Vargas	ADSO
1052312624	Jimmy	Angarita	ADSO
1057579055	Yamith	Rosas	ADSO



Plataforma institucional donde aprendices e instructores pueden consultar convocatorias, cursos, estadísticas, etc.

Institucional

- Oferta académica
- Sedes
- Convocatorias
- Noticias

Servicios

- Soporte técnico
- Calendario académico
- Reportes y estadísticas
- Preguntas frecuentes

Contacto

Email: contacto@sena.edu.co

Teléfono: +57 310 000 0000

Dirección: Cra. 50 #26-56, Bogotá D.C.

8. Código de los Template

- Template de aprendices_list

The screenshot shows the Visual Studio Code editor with the Django project 'SENA_APP' open. The Explorer sidebar on the left shows the file structure, with the 'templates' directory expanded and 'dj aprendices_list.html' selected. The main editor displays the content of this template. The code is a Django template that extends 'master.html' and renders a list of apprentices. It includes a table with columns for Document, Name, Surname, and Program. The table is generated using a for loop over the 'aprendices' context variable.

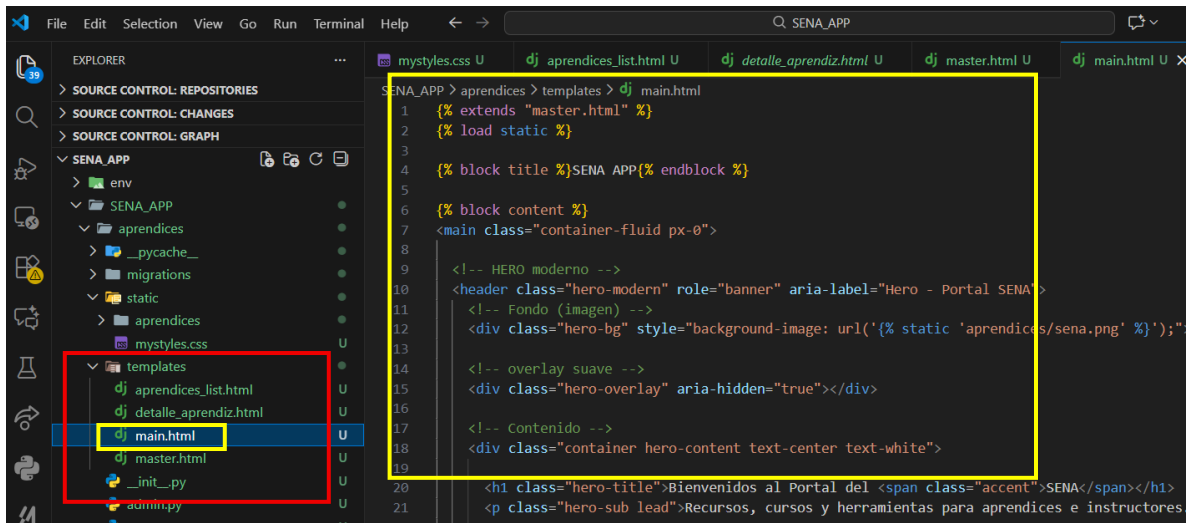
```
1 {% extends 'master.html' %} {% block title %}Lista de Aprendices{% endblock %}
2 {% block content %}
3 {% load static %}
4 <div class="row mt-4">
5   <div class="col-12">
6     <link rel="stylesheet" href="{% static 'mystyles.css' %}" />
7     <!-- Título moderno -->
8     <div class="section-header mb-4">
9       <h2 class="section-title">
10         <i class="bi bi-people-fill me-2"></i> Lista de Aprendices
11       </h2>
12       <p class="section-sub">
13         Consulta rápida y organizada de los aprendices registrados.
14       </p>
15     </div>
16
17     <!-- TABLA (pantallas grandes) -->
18     <div class="table-responsive d-none d-md-block">
19       <table class="table table-modern table-hover align-middle">
20         <thead>
21           <tr>
22             <th>Documento</th>
23             <th>Nombre</th>
24             <th>Apellido</th>
25             <th>Programa</th>
26           </tr>
27         </thead>
28         <tbody>
29           {% for aprendiz in aprendices %}
30             <tr>
31               <td>
32                 <a
33                   class="table-link"
34                   href="{% url 'detalle_aprendiz' aprendiz.id %}"
```

- Template de detalle_aprendiz

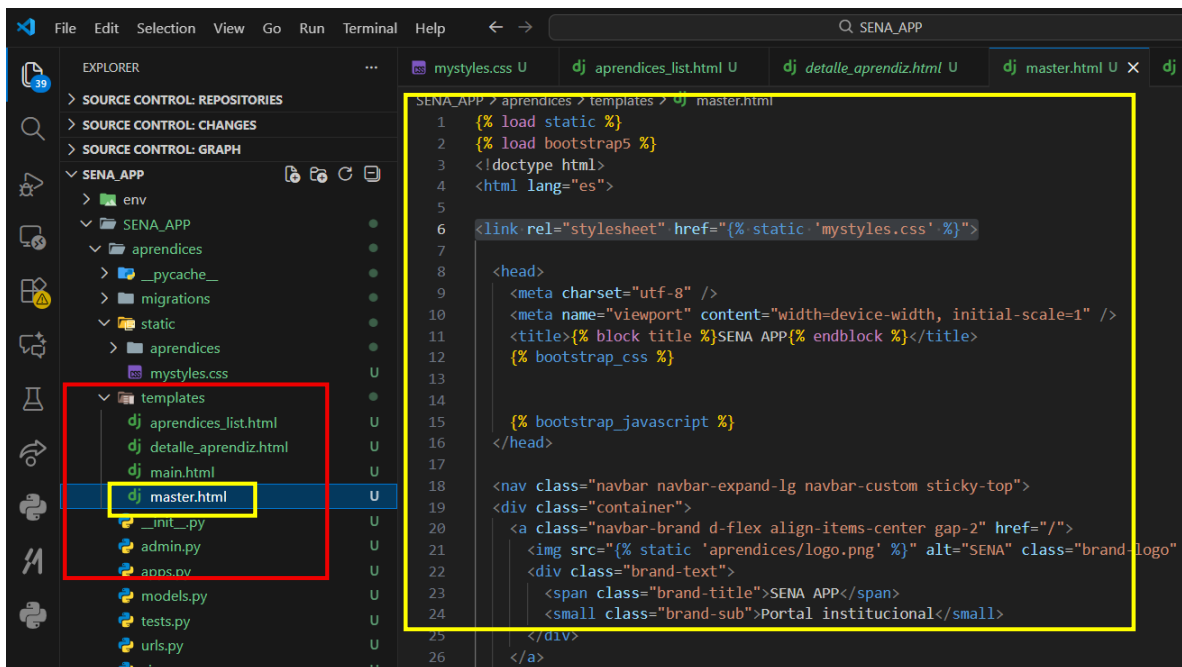
The screenshot shows the Visual Studio Code editor with the Django project 'SENA_APP' open. The Explorer sidebar on the left shows the file structure, with the 'templates' directory expanded and 'dj detalle_aprendiz.html' selected. The main editor displays the content of this template. The code is a Django template that extends 'master.html' and renders the details of a specific apprentice. It includes a card with the apprentice's name and program, and a section for the program details.

```
1 {% extends "master.html" %}
2 {% block title %}Detalle del Aprendiz - {{ aprendiz_detalle.firstname }} {{ aprendiz_detalle.lastname }}{% endblock %}
3
4 {% block content %}
5 <div class="card shadow-sm my-4 detalle-aprendiz-card">
6   <div class="card-body p-4">
7     <div class="row g-3 align-items-center">
8       <!-- Avatar con iniciales -->
9       <div class="col-auto">
10         <div class="avatar rounded-3 d-flex align-items-center justify-content-center">
11           {{ aprendiz_detalle.firstname|slice:"1" }} {{ aprendiz_detalle.lastname|slice:"1" }}
12         </div>
13       </div>
14
15       <!-- Nombre y Programa -->
16       <div class="col">
17         <h2 class="h4 mb-1">{{ aprendiz_detalle.firstname }} {{ aprendiz_detalle.lastname }}</h2>
18         <div class="small text-muted">
19           Programa:
```

- Template main.html



- Template master.html



9. Pagina

- Inicio



Inicio

Aprendices

Instructores

Buscar aprendices, curs...

 Perfil



Bienvenidos al Portal del SENA

Recursos, cursos y herramientas para aprendices e instructores. Accede rápido a tus módulos y gestiona tu formación.

Ver aprendices

Ver instructores

Gestión de aprendices

Crea, edita y consulta la información del aprendiz de forma segura.

[Ver aprendices](#)

Gestión de instructores

Administra perfiles, asignaciones y horarios de instructores.

[Ver instructores](#)

Agenda y reportes

Consulta eventos, talleres y reportes de desempeño.

[Ver agenda](#)

- Aprendices



Inicio

Aprendices

Instructores

Buscar aprendices, cursos...

 Perfil

Lista de Aprendices

Consulta rápida y organizada de los aprendices registrados.

DOCUMENTO	NOMBRE	APELLIDO	PROGRAMA
1056928954	Andres	Ardila	ADSO
1002537360	Mayerly	Cardenas	ADSO
1098635221	Kamila	Vargas	ADSO
1052312624	Jimmy	Angarita	ADSO
1057579055	Yamith	Rosas	ADSO

- Detalle aprendiz

DOCUMENTO	NOMBRE
1056928954	Andres
1002537360	Mayerly
1098635221	Kamila
1052312624	Jimmy
1057579055	Yamith

AA **Andres Ardila**

Programa:

Documento: 1056928954

Ciudad: Toca

Fecha de nacimiento: 25 de December 2006

Email: andres@gmail.com

Teléfono: [3223394432](tel:3223394432)

Enviar correo

Ver lista

Ver historial

Repositorio GitHub

- https://github.com/TattysB/SENA_APP.git

CONCLUSIONES

- La creación de **SENA APP** permitió comprender de manera práctica el funcionamiento de framework **Django**, evidenciando su potencia para gestionar proyectos estructurados, escalables y con funcionalidades como el panel administrativo.
- La implementación del entorno virtual, la instalación de dependencias y la configuración del proyecto fortalecieron las competencias en buenas prácticas de desarrollo, resaltando la importancia del orden y la modularidad en el manejo de aplicaciones web.
- La creación de plantillas HTML, el uso de **master.html**, **main.html** y otras vistas, junto con la inclusión de Bootstrap, permitió diseñar una interfaz clara y funcional, resaltando el papel fundamental de la capa de presentación en la experiencia del usuario.
- Finalmente, las pruebas realizadas confirmaron la correcta interacción entre modelos, vistas, URLs y plantillas, validando el funcionamiento global de la aplicación y dejando bases sólidas para futuras ampliaciones del proyecto.