

SENA\_APP CURSOS

YEIMY TATIANA BAYONA PÉREZ

SERVICIO NACIONAL DE APRENDIZAJE

CENTRO MINERO

ANALISIS Y DESARROLLO DE SOFTWARE(ADSO)

FICHA:2993648

ANDRES FELIPE SANDOVAL HIGUERA

SOGAMOSO BOYACÁ

DICIEMBRE

2025

# Tabla de contenido

<b>1. PANEL ADMINISTRATIVO.....</b>	<b>4</b>
• <b>Cursos.....</b>	<b>4</b>
• <b>Aprendices por curso .....</b>	<b>5</b>
• <b>Instructores por curso.....</b>	<b>6</b>
<b>2. BASE DE DATOS SQLITE.....</b>	<b>7</b>
• <b>Cursos.....</b>	<b>7</b>
• <b>Aprendices por curso .....</b>	<b>7</b>
• <b>Instructor por curso.....</b>	<b>8</b>
<b>3. MODELO.....</b>	<b>8</b>
<b>4. VISTA .....</b>	<b>9</b>
<b>5. URLs.....</b>	<b>10</b>
<b>6. TEMPLATE.....</b>	<b>10</b>
<b>7. NAVEGADOR DE LA APP CORRIENDO .....</b>	<b>11</b>
➤ <b>Lista cursos .....</b>	<b>11</b>
➤ <b>Detalle cursos .....</b>	<b>12</b>
<b>8. REPOSITORIO GITHUB .....</b>	<b>12</b>
<b>CONCLUSIONES .....</b>	<b>13</b>

## INTRODUCCIÓN

El desarrollo del módulo **SENA\_APP CURSOS** tiene como propósito gestionar de manera eficiente la información relacionada con los cursos, aprendices e instructores dentro de un entorno administrativo funcional. Mediante la implementación de un panel administrativo basado en Django, se facilita el registro, actualización y visualización de los datos almacenados en una base de datos SQLite, permitiendo así un flujo de trabajo organizado, seguro y confiable.

Este proyecto integra conceptos fundamentales del framework Django, como modelos, vistas, plantillas y URL, garantizando la interacción adecuada entre el backend y el frontend. Además, se desarrollaron interfaces amigables para la visualización de los cursos y su respectivo detalle, ofreciendo una experiencia clara y práctica para el usuario. La aplicación permite la administración completa de cada tabla, asegurando que el manejo de datos sea accesible incluso para usuarios sin conocimientos avanzados en bases de datos.

En conjunto, este sistema demuestra las capacidades de Django como herramienta para la construcción de aplicaciones web robustas y escalables, resaltando la importancia de una correcta separación de responsabilidades entre los componentes del proyecto.

## 1. PANEL ADMINISTRATIVO

Se registran los campos de las tablas desde el panel administrativo

- Cursos

The screenshot shows the Django administration interface at the URL [127.0.0.1:8000/admin/cursos/curso/](http://127.0.0.1:8000/admin/cursos/curso/). The title bar says "Select Curso to change | Django site". The main content area is titled "Django administration" and shows the navigation path "Home > Cursos > Cursos". On the left, there is a sidebar with a search bar and several model categories: APRENDICES, AUTHENTICATION AND AUTHORIZATION, CURSOS, INSTRUCTORES, and PROGRAMAS. Under the CURSOS category, the "Cursos" model is highlighted with a green border and a red box around it. In the center, a modal window titled "Select Curso to change" lists two selected items: "CURSO" and "2334323 - Cocina" and "2994648 - ADSO". A purple box highlights this modal window.

Se registran los datos de la tabla **cursos** cada una con sus parámetros que requiere el sistema

- **Aprendices por curso**

The screenshot shows the Django administration interface at the URL [127.0.0.1:8000/admin/cursos/aprendizcurso/](http://127.0.0.1:8000/admin/cursos/aprendizcurso/). The left sidebar is highlighted with a yellow box and lists several models: APRENDICES, CURSOS, INSTRUCTORES, and PROGRAMAS. The CURSOS section contains three items: Aprendices por Curso, Cursos, and Instructores por Curso. The Aprendices por Curso item is selected and highlighted with a red box. The main content area displays a list titled "Select Aprendiz por Curso to change". It shows two entries: "APRENDIZ POR CURSO" and "1002537360 Mayerly Cardenas - 2994648 - ADSO (DES)". Below the list, it says "2 Aprendices por Curso".

APRENDIZ POR CURSO
1002537360 Mayerly Cardenas - 2994648 - ADSO (DES)
1052312624 Jimy Angarita - 2334323 - Cocina (INS)

Se registran los datos de la tabla **aprendices por curso** cada una con sus parámetros que requiere el sistema.

- **Instructores por curso**

The screenshot shows the Django administration interface for the 'Instructores por curso' model. The left sidebar has a green highlight around the 'CURSOS' section, and the 'Instructores por Curso' item is highlighted with a red box. The main content area displays a list of two entries:

Action	INSTRUCTOR POR CURSO
<input type="checkbox"/>	Angela Medina - Chef Cocina - 2334323 - Cocina (Chef)
<input type="checkbox"/>	Diego Ojeda - Software - 2994648 - ADSO (Instructor)

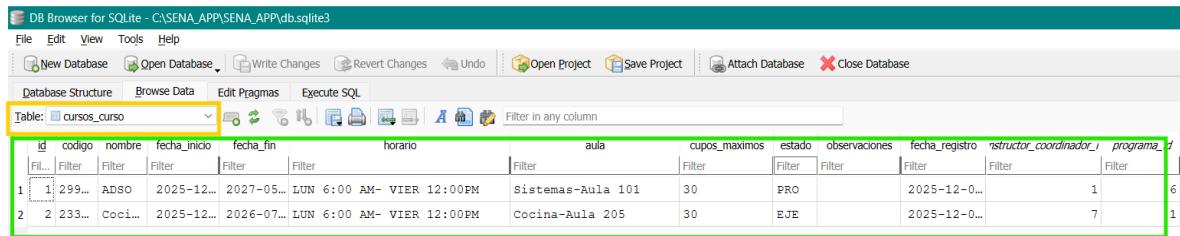
Below the list, it says '2 Instructores por Curso'.

Se registran los datos de la tabla **instructores por curso** cada una con sus parámetros que requiere el sistema.

## 2. BASE DE DATOS SQLITE

Se actualiza la tabla en la base de datos y cargan los registros que se hicieron en el panel administrativo.

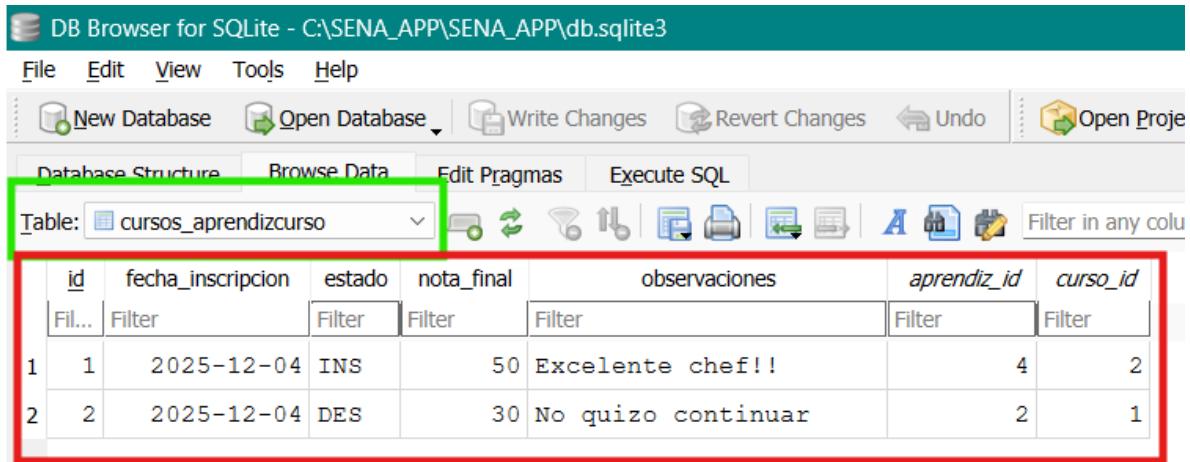
- **Cursos**



The screenshot shows the DB Browser for SQLite interface with the 'cursos\_curso' table selected. The table has columns: id, codigo, nombre, fecha\_inicio, fecha\_fin, horario, aula, cupos\_maximos, estado, observaciones, fecha\_registro, instructor\_coordinador\_id, and programa\_id. The data is as follows:

Table:	cursos_curso											
1	1	299...	ADSO	2025-12...	2027-05...	LUN 6:00 AM- VIER 12:00PM	Sistemas-Aula 101	30	PRO	2025-12-0...	1	6
2	2	233...	Coci...	2025-12...	2026-07...	LUN 6:00 AM- VIER 12:00PM	Cocina-Aula 205	30	EJE	2025-12-0...	7	1

- **Aprendices por curso**



The screenshot shows the DB Browser for SQLite interface with the 'cursos\_aprendizcurso' table selected. The table has columns: id, fecha\_inscripcion, estado, nota\_final, observaciones, aprendiz\_id, and curso\_id. The data is as follows:

Table:	cursos_aprendizcurso					
1	1	2025-12-04	INS	50	Excelente chef!!	4
2	2	2025-12-04	DES	30	No quizo continuar	2

- Instructor por curso

DB Browser for SQLite - C:\SENA\_APP\SENA\_APP\db.sqlite3

File Edit View Tools Help

New Database Open Database Write Changes Revert Changes

Database Structure Browse Data Edit Pragmas Execute SQL

Table: cursos\_instructorcurso

		id	rol	fecha_asignacion	curso_id	instructor_id
		Filter	Filter	Filter	Filter	Filter
1	1	Instructor		2025-12-04	1	1
2	2	Chef		2025-12-04	2	7

### 3. MODELO

File Edit Selection View Go Run Terminal Help

EXPLORER

SOURCE CONTROL: REPOSITORIES  
SOURCE CONTROL: CHANGES  
SOURCE CONTROL: GRAPH

SENA\_APP

SENA\_APP

cursos

models.py

views.py

Templates

detail\_curso.html

lista\_cursos.html

\_\_init\_\_.py

admin.py

apps.py

models.py

tests.py

urls.py

views.py

instructores

programas

SENA\_APP

\_\_init\_\_.py

asgi.py

settings.py

urls.py

wsgi.py

models.py

```

SENA_APP > cursos > models.py > Curso
1   from django.db import models
2
3   # Create your models here.
4
5   class Curso(models.Model):
6       ESTADO_CHOICES = [
7           ('PRO', 'Programado'),
8           ('INI', 'Iniciado'),
9           ('EJE', 'En Ejecución'),
10          ('FIN', 'Finalizado'),
11          ('CAN', 'Cancelado'),
12          ('SUS', 'Suspendido'),
13      ]
14
15      codigo = models.CharField(max_length=30, unique=True, verbose_name="Código del curso")
16      nombre = models.CharField(max_length=200, verbose_name="Nombre del Curso")
17      programa = models.ForeignKey('programas.Programa', on_delete=models.CASCADE, verbose_name="Programa")
18      instructor_coordinador = models.ForeignKey('instructores.Instructor', on_delete=models.CASCADE, verbose_name="Instructor Coordinador")
19      instructores = models.ManyToManyField('instructores.Instructor', through='InstructorCurso')
20      aprendices = models.ManyToManyField('aprendices.Aprendiz', through='AprendizCurso', related_name='cursos')
21      fecha_inicio = models.DateField(verbose_name="Fecha de Inicio")
22      fecha_fin = models.DateField(verbose_name="Fecha de Finalización")
23      horario = models.CharField(max_length=100, verbose_name="Horario")
24      aula = models.CharField(max_length=50, verbose_name="Aula/Ambiente")
25      cupos_maximos = models.PositiveIntegerField(verbose_name="Cupos Máximos")
26
27      estado = models.CharField(max_length=3, choices=ESTADO_CHOICES, default='PRO', verbose_name="Estado")
28      observaciones = models.TextField(blank=True, null=True, verbose_name="Observaciones")
29      fecha_registro = models.DateTimeField(auto_now_add=True, verbose_name="Fecha de Registro")
30
31      class Meta:
32          verbose_name = "Curso"
33          verbose_name_plural = "Cursos"
34          ordering = ['-fecha_inicio']

```

Se registran los parámetros que se van a guardar y cargar en la base de datos

## 4. VISTA

The screenshot shows a code editor interface with a green border around the main content area. On the left is an Explorer sidebar showing the project structure:

- SENA\_APP (selected)
- SENA\_APP (under SENA\_APP)
- aprendices (under SENA\_APP)
- urls.py (under aprendices)
- views.py (under aprendices)
- cursos (under SENA\_APP)
- migrations (under SENA\_APP)
- Templates (under SENA\_APP)
  - detalle\_curso.html
  - lista\_cursos.html
  - \_\_init\_\_.py
  - admin.py
  - apps.py
  - models.py
  - tests.py
  - urls.py
  - views.py (highlighted with a red box)
- instrutores (under SENA\_APP)
- programas (under SENA\_APP)
- SENA\_APP (under SENA\_APP)
- \_\_pycache\_\_ (under SENA\_APP)
- \_\_init\_\_.py (under SENA\_APP)
- asgi.py (under SENA\_APP)
- settings.py (under SENA\_APP)
- urls.py (under SENA\_APP)
- wsgi.py (under SENA\_APP)

The main code editor window displays the contents of the `views.py` file under the `cursos` directory:

```
from django.shortcuts import render
from django.template import loader
from .models import Curso
from django.http import HttpResponseRedirect

# Create your views here.
def lista_cursos(request):
    cursos = Curso.objects.all()
    template = loader.get_template("lista_cursos.html")

    context = {
        "lista_cursos": cursos,
        "total_cursos": cursos.count(),
    }

    return HttpResponseRedirect(template.render(context, request))

def detalle_curso(request, curso_id):
    curso = Curso.objects.get(id=curso_id)
    aprendices_curso = curso.aprendizcurso_set.all()
    instructores_curso = curso.instructorcurso_set.all()
    template = loader.get_template("detalle_curso.html")

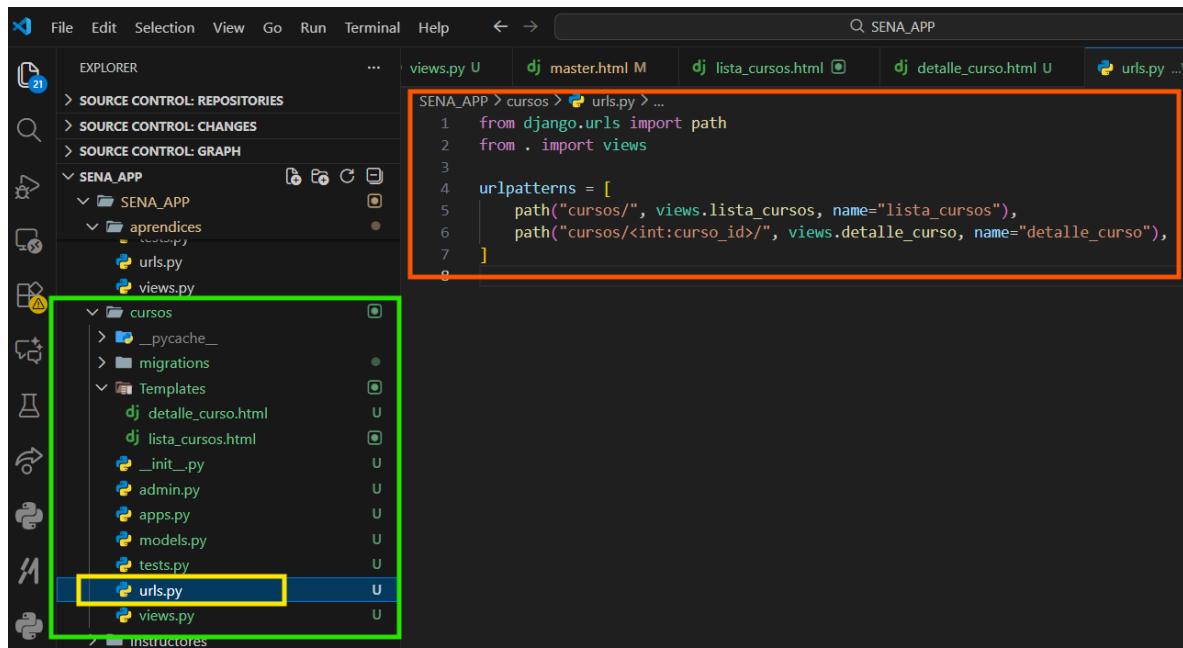
    context = {
        "curso": curso,
        "aprendices_curso": aprendices_curso,
        "instructores_curso": instructores_curso,
    }

    return HttpResponseRedirect(template.render(context, request))
```

Se llaman las funciones y URL de los template, cada uno con sus QUERYSETS

## 5. URLs

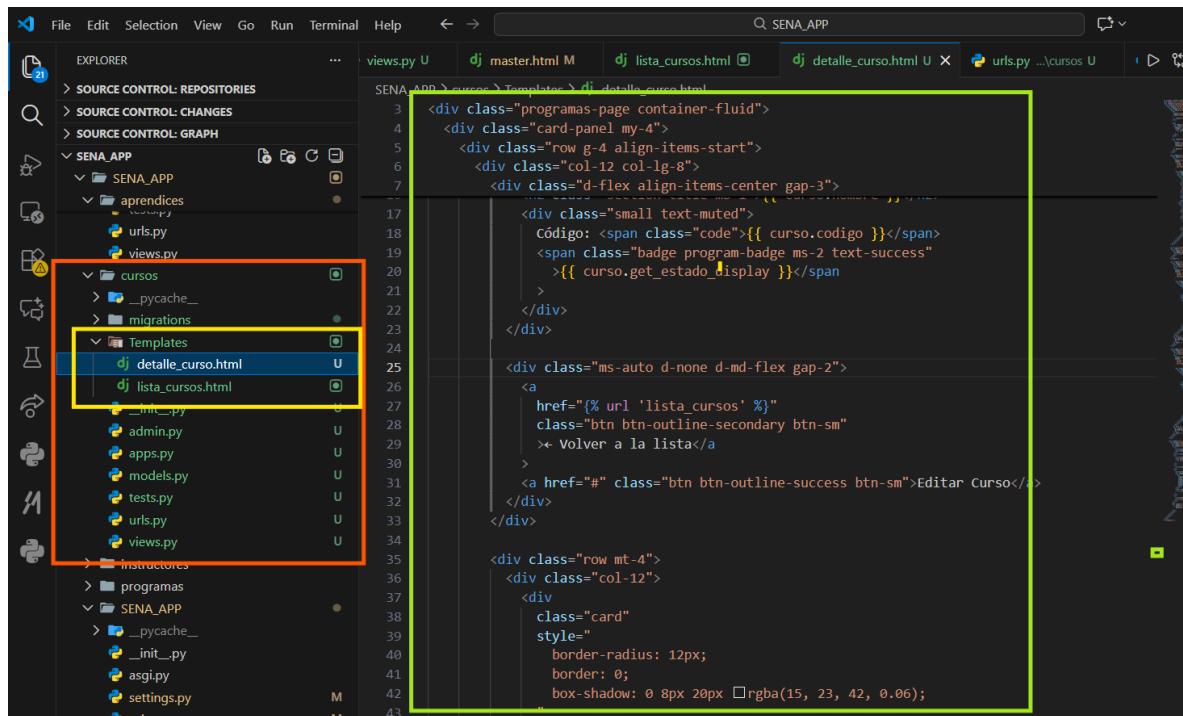
Se llaman las URLs de la vista



```
SENA_APP > cursos > urls.py > ...
1 from django.urls import path
2 from . import views
3
4 urlpatterns = [
5     path("cursos/", views.lista_cursos, name="lista_cursos"),
6     path("cursos/<int:curso_id>", views.detalle_curso, name="detalle_curso"),
7 ]
8
```

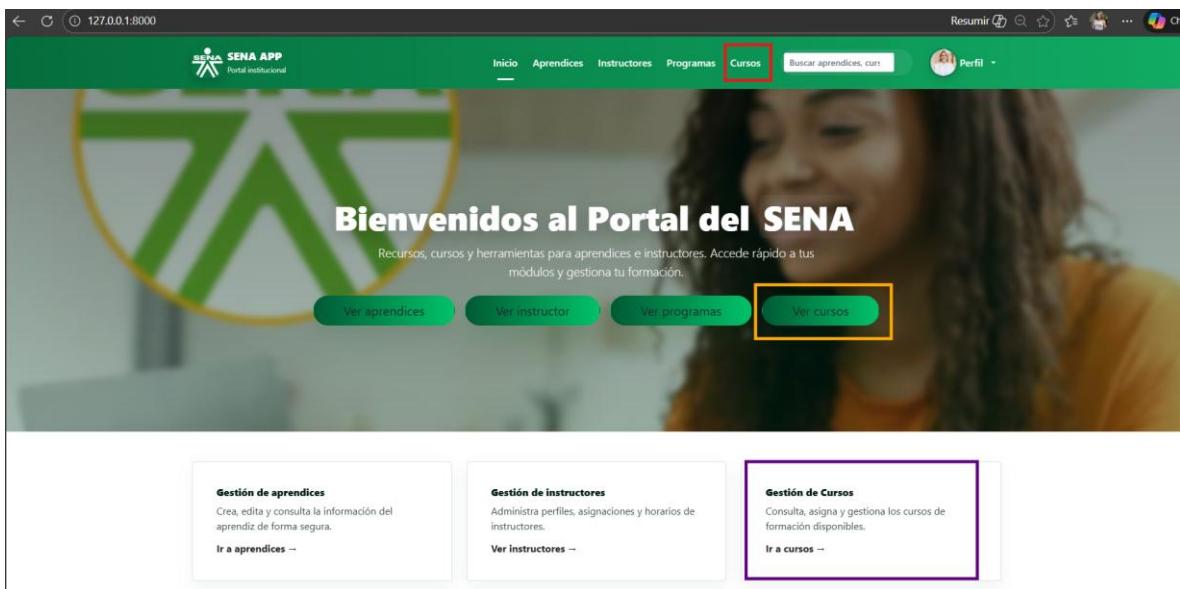
## 6. TEMPLATE

Se hacen los template de la lista de cursos y detalle curso



```
<div class="programas-page container-fluid">
  <div class="card-panel my-4">
    <div class="row g-4 align-items-start">
      <div class="col-12 col-lg-8">
        <div class="d-flex align-items-center gap-3">
          <div class="small text-muted">
            Código: <span class="code">{{ curso.codigo }}</span>
            <span class="badge program-badge ms-2 text-success">
              >{{ curso.get_estado_display }}</span>
            </div>
          </div>
        <div class="ms-auto d-none d-md-flex gap-2">
          <a href="{% url 'lista_cursos' %}" class="btn btn-outline-secondary btn-sm">
            <& Volver a la lista</a>
          >
          <a href="#" class="btn btn-outline-success btn-sm">Editar Curso</a>
        </div>
      </div>
    </div>
    <div class="row mt-4">
      <div class="col-12">
        <div class="card" style="border-radius: 12px; border: 0; box-shadow: 0 8px 20px rgba(15, 23, 42, 0.06);>
          <div>
```

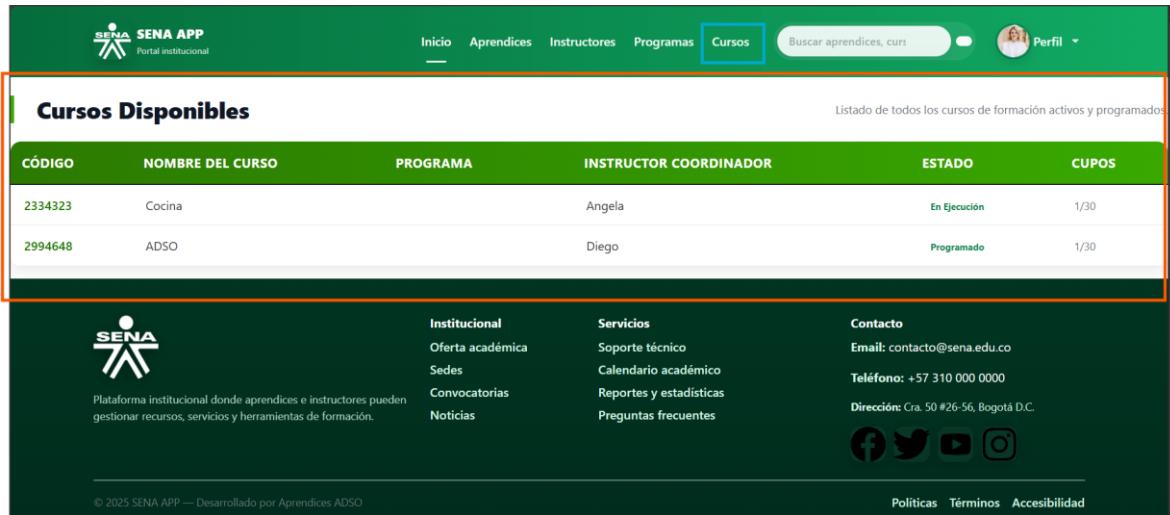
## 7. NAVEGADOR DE LA APP CORRIENDO



The screenshot shows the SENA App homepage. At the top, there is a navigation bar with links for Inicio, Aprendices, Instructores, Programas, and Cursos. The 'Cursos' link is highlighted with a red box. Below the navigation bar is a large banner with the text 'Bienvenidos al Portal del SENA' and a subtext about resources for apprentices and instructors. Below the banner are four buttons: 'Ver aprendices', 'Ver instructor', 'Ver programas', and 'Ver cursos'. The 'Ver cursos' button is highlighted with a yellow box. Below these buttons are three boxes labeled 'Gestión de aprendices', 'Gestión de instructores', and 'Gestión de Cursos'. The 'Gestión de Cursos' box is highlighted with a purple border.

En la página de inicio, se añaden accesos directos para acceder más fácilmente al módulo de cursos y los pueda listar.

### ➤ Lista cursos



The screenshot shows the 'Cursos Disponibles' section of the SENA App. The section title is 'Cursos Disponibles' and it includes a subtitle 'Listado de todos los cursos de formación activos y programados'. Below the title is a table with columns: CÓDIGO, NOMBRE DEL CURSO, PROGRAMA, INSTRUCTOR COORDINADOR, ESTADO, and CUPOS. Two rows of data are shown:

CÓDIGO	NOMBRE DEL CURSO	PROGRAMA	INSTRUCTOR COORDINADOR	ESTADO	CUPOS
2334323	Cocina		Angela	En Ejecución	1/30
2994648	ADSO		Diego	Programado	1/30

At the bottom of the page, there is a footer with the SENA logo, a brief description of the platform, links to institutional services like Oferta académica and Soporte técnico, contact information (Email: contacto@sena.edu.co, Teléfono: +57 310 000 0000, Dirección: Cra. 50 #26-56, Bogotá D.C.), social media icons, and links to Políticas, Términos, and Accesibilidad.

Este es el apartado de la lista de cursos teniendo la visualización completa de los que están registrados.

## ➤ Detalle cursos

The screenshot shows a web application interface for the SENA SENAPP Portal institutional. At the top, there is a green header bar with the logo 'SENA SENAPP' and 'Portal institucional'. Below the header, the main content area has a white background. On the left side, there is a sidebar with sections for 'Detalles Generales', 'Instructores del Curso', and 'Aprendices Inscritos'. The main content area displays detailed information about a course named 'Cocina' with the code '2334323' and status 'En Ejecución'. It includes fields for 'Fecha Inicio' (24 de December 2025), 'Fecha Finalización' (10 de July 2026), 'Estado' (En Ejecución), 'Cupos disponibles' (29), and 'Ocupación' (3%). There are buttons for 'Volver a la lista', 'Editar Curso', and 'Exportar Reporte'. A yellow border highlights the entire content area.

Mostramos toda la información de los cursos que se registraron en la base de datos.

## 8. REPOSITORIO GITHUB

[https://github.com/TattyB/SENA\\_APP.git](https://github.com/TattyB/SENA_APP.git)

## CONCLUSIONES

La construcción del módulo **SENA\_APP CURSOS** permitió comprender y aplicar los fundamentos esenciales para la creación de aplicaciones web dinámicas utilizando Django. A través del desarrollo del panel administrativo y su conexión con la base de datos SQLite, se garantizó un manejo adecuado y estructurado de la información, cumpliendo con los requerimientos de gestión de cursos, aprendices e instructores.

Asimismo, el uso de modelos, vistas, plantillas y URL evidenció la importancia de una arquitectura organizada para asegurar el funcionamiento correcto de la aplicación. La integración de los listados y detalles de cada curso consolidó una interfaz intuitiva y funcional para los usuarios finales.

Finalmente, este proyecto permitió afianzar competencias en desarrollo web, diseño de interfaces y manejo de bases de datos, demostrando que Django es una herramienta poderosa para crear soluciones ágiles y completas en entornos educativos y administrativos.