**CARRERA TECNOLIGIA SUPERIOR EN DESARROLLO DE SOFTWARE**

**MATERIA: PROGRAMACION VISUAL**

**NOMBRE: VICTOR MANUEL SAREZ SUAREZ**

**CURSO: 3 H**

**DOCENTE: CARLOS LUIS PAZMIÑO PALMA**

Imagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamente

**INDICE**

Pag 1) ¿Qué es Django?

Pag 1) ¿Qué es la máquina virtual en Django

Pag 2) ¿Qué es MVT en Django. ?

4) Crear un proyecto con la máquina virtual.

5) Descargar los instaladores de Django al proyecto

6) Crear un proyecto para programar en Django

7) Ejecutar el proyecto y el mensaje de felicitaciones.

8) Crear una Apps Core.

9) ¿Qué es la Carpeta Templetes?

10) ¿Qué es la Carpeta stactic?

11)Crear un archivo base HTML en la APPS Core

12)Como se llaman a los CSS desde el archivo base HTML.

13)Como consume un archivo hijo HTML al utilizar la herencia del archivo base HTML.

14)Crear un view que llame al HTML hijo

15)Crear la urls que llame al views.

16)Integrar la aplicación APPS Core al proyecto principal

17)Crear las tablas del sistema de usuarios para utilizar el panel de administración.

18)Crear un usuario para poder ingresar al Panel de Administración

19)Que es un modelo en Django

20)Crear un modelo en Django.

21)Migrar el Modelo a la base del Panel de Administración.

22)Integrar el Modelo al Panel de Administración.

23)Ingresar información al modelo por el Panel de Administración.

24)Realizar la consulta de todo lo ingresado en el modelo desde el views.

25)Mostrar los datos guardados en el modelo al HTML hijo.

1. **¿Qué es Django?**

Django es un framework de aplicaciones web gratuito y de código abierto (open source) escrito en Python. Un framework web es un conjunto de componentes que te ayudan a desarrollar sitios web más fácil y rápidamente.

Cuando construyes un sitio web, siempre necesitas un conjunto de componentes similares: una manera de manejar la autenticación de usuarios (registrarse, iniciar sesión, cerrar sesión), un panel de administración para tu sitio web, formularios, una forma de subir archivos, etc.

Por suerte para nosotros, hace tiempo que otros desarrolladores se dieron cuenta de que siempre se enfrentaban a los mismos problemas cuando construían sitios web, y por eso se unieron y crearon frameworks (Django es uno de ellos) con componentes listos para usarse.

Los frameworks sirven para que no tengamos que reinventar la rueda cada vez y que podamos avanzar más rápido al construir un nuevo sitio.

1. **¿Qué es la máquina virtual en Django?**

La máquina virtual es un entorno virtual de Python aislado creado para un proyecto. Este entorno tiene su propio intérprete de Python, bibliotecas y paquetes, lo que significa que el intérprete de Python y las dependencias instaladas pertenecen solo a este proyecto. Podemos tener diferentes versiones de Python y diversas bibliotecas para varios proyectos con entornos virtuales sin acumular las dependencias instaladas globalmente.

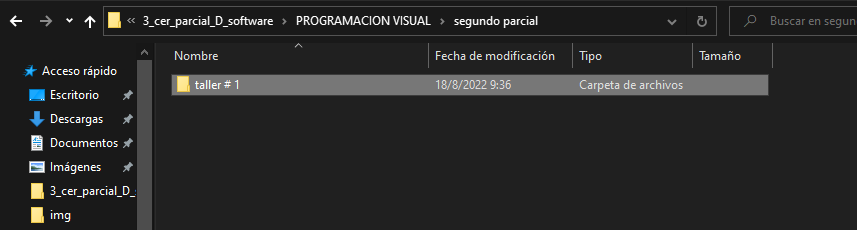
Al iniciar un nuevo proyecto, siempre se recomienda crear un entorno virtual y usarlo específicamente para ese proyecto.

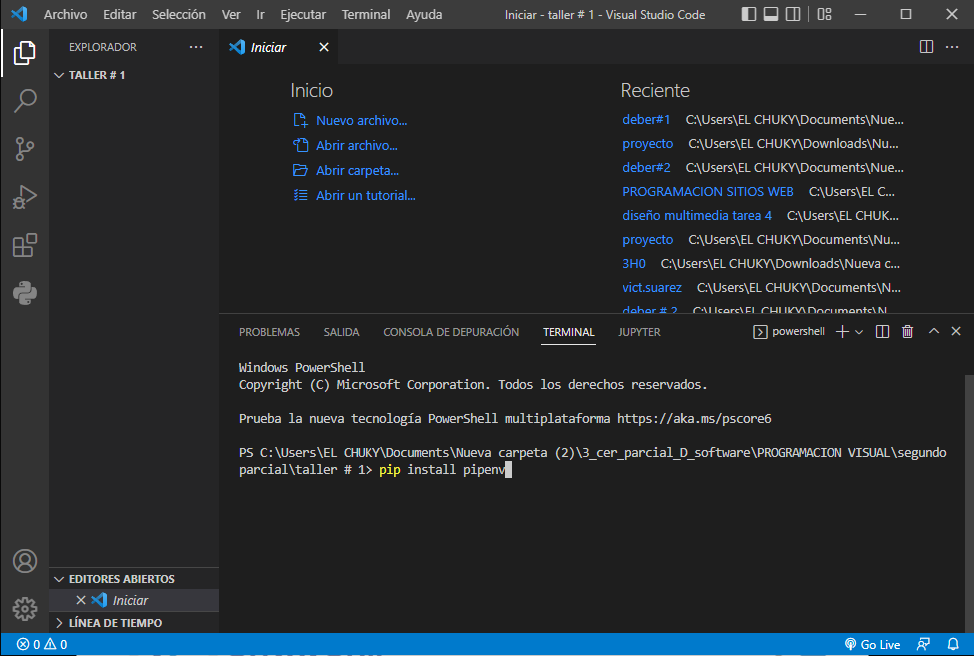
1. **¿Qué es MVT en Django ?**

Es un modelo de vistas templates.

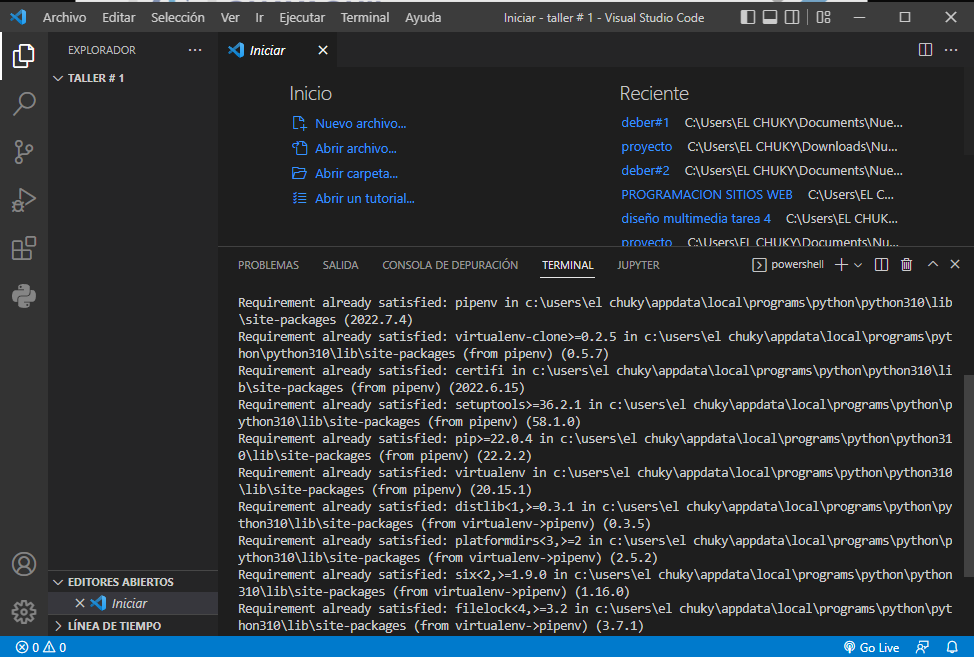
1. **Crear un proyecto con la máquina virtual.**

* Creamos una carpeta en nuestro ordenador.



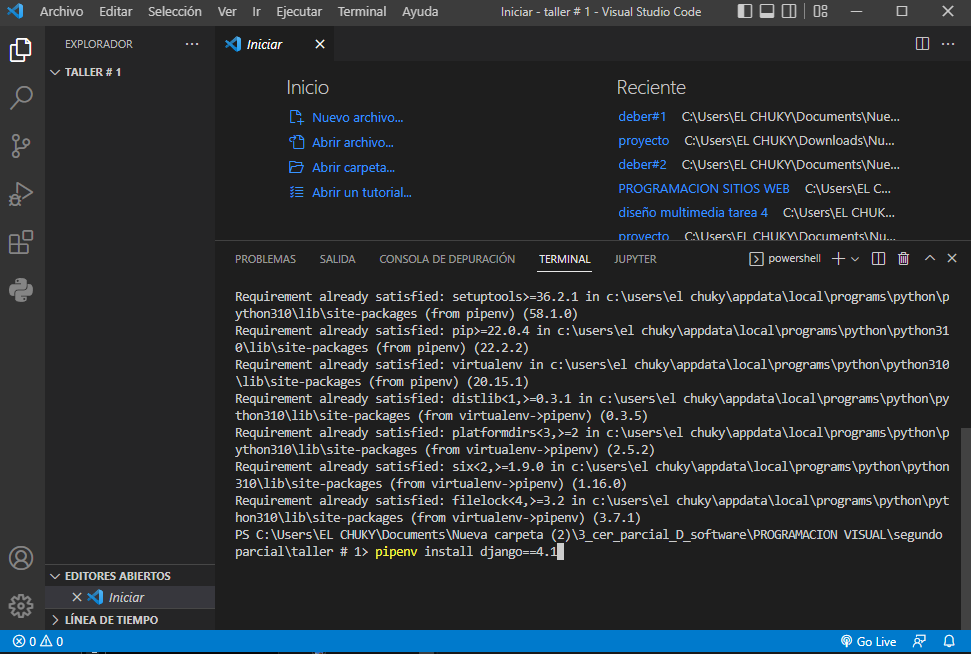
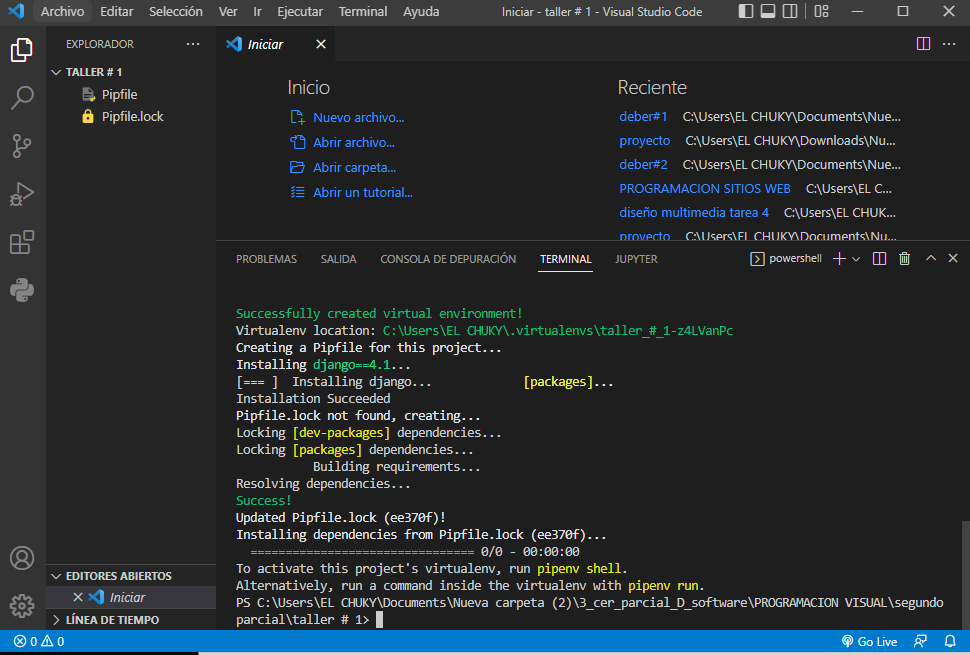
* luego para comenzar con nuestro proyecto comenzamos creando el directorio virtual con el siguiente comando en nuestro terminal: pip install pipenv

Damos enter y esperamos que se nos cree el entorno virtual.

Como podemos observar ya esta creado el entorno virtual.

5) Descargar los instaladores de Django al proyecto

* En la carpeta antes creada procedemos a instalar la última versión de Django con el comando: Pipenv install django==4.1

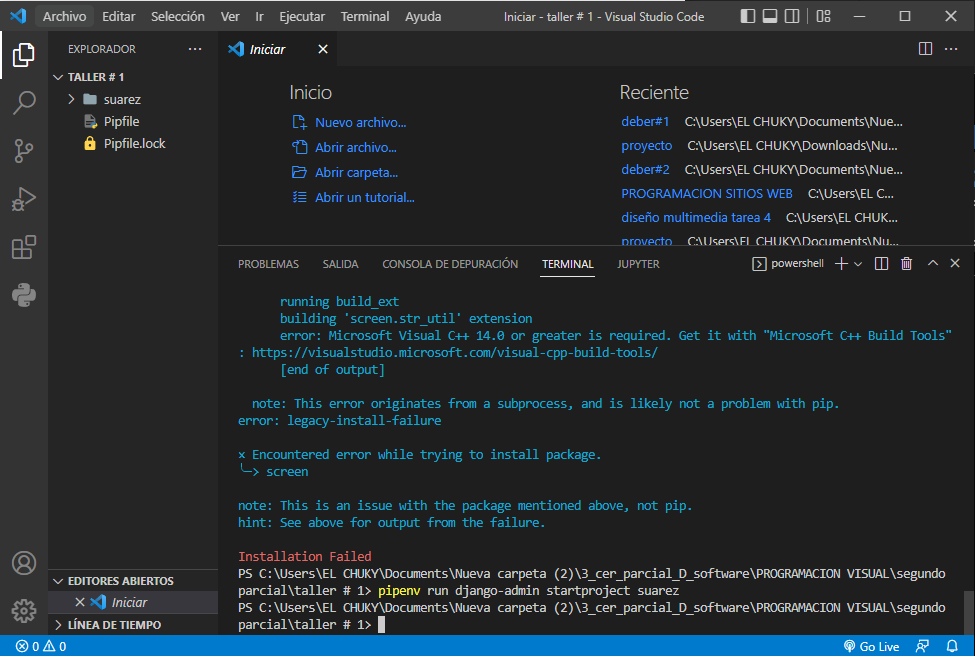
esperamos que se descargue Django, podemos observar que se descargó con éxito.

**6) Crear un proyecto para programar en Django**

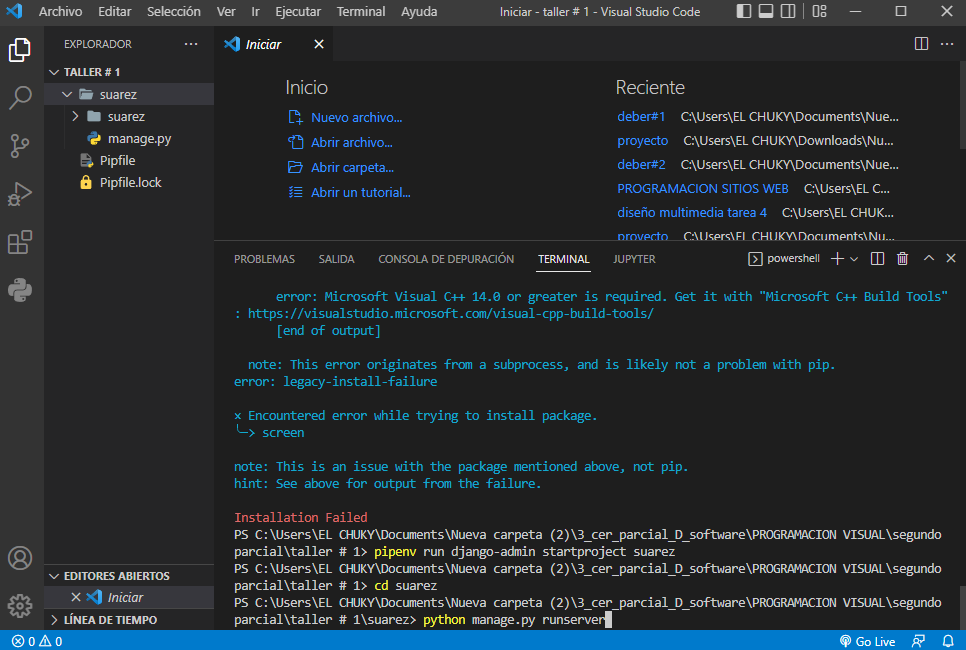
* Captura de pantalla de computadora

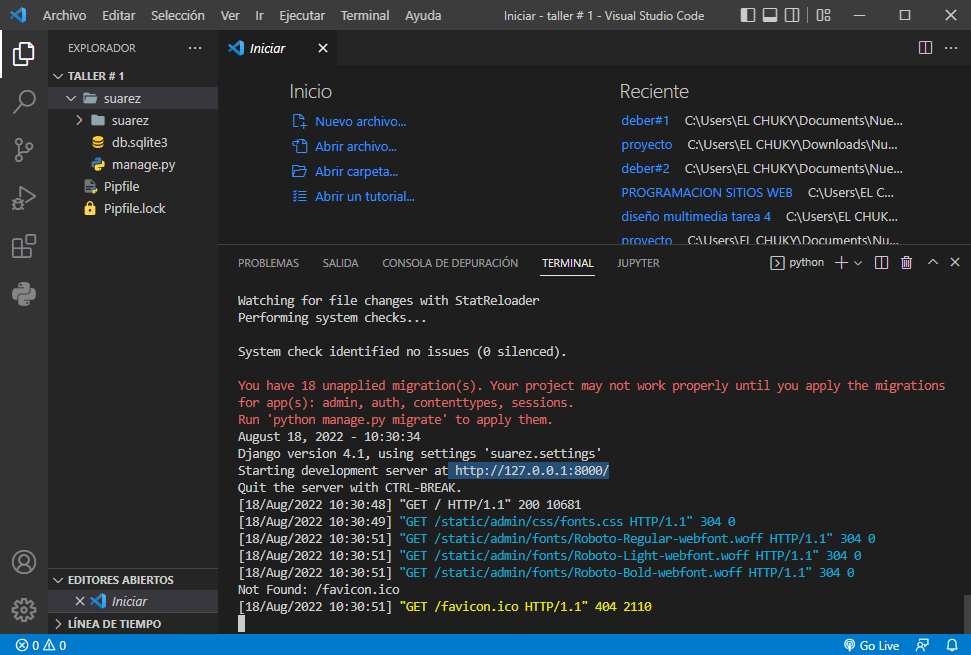
  Descripción generada automáticamenteLuego nos dirigimos a la carpeta creada donde descargamos Django y procedemos a crear nuestro proyecto con el comando: pipenv run django-admin startproject (nombre del proyecto)

Esperamos que se nos cree nuestro proyecto

Como podemos observar ya se creó nuestro proyecto

**7) Ejecutar el proyecto y el mensaje de felicitaciones.**

* Luego ingresamos a la carpera de nuestro proyecto donde se encuentra el manage.py y ejecutamos el proyecto en nuestro terminal con el comando: Python manage.py runserver

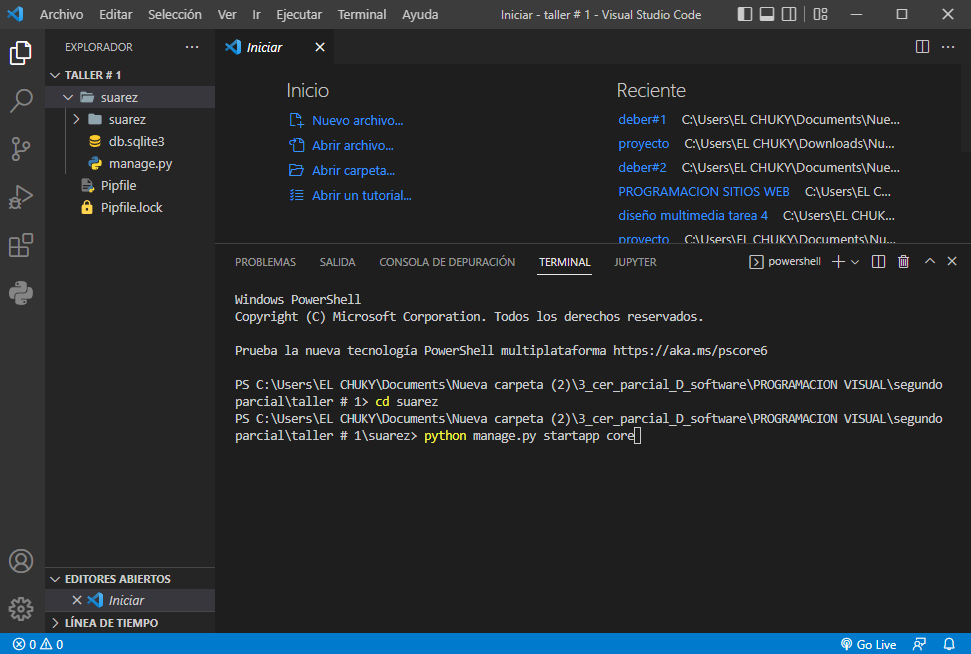
Como podemos observar ya se ejecuto nuestro proyecto copiamos y pegamos en nuestro navegador preferido y nos mostrara la página de felicitación de nuestro proyecto creado con éxito.

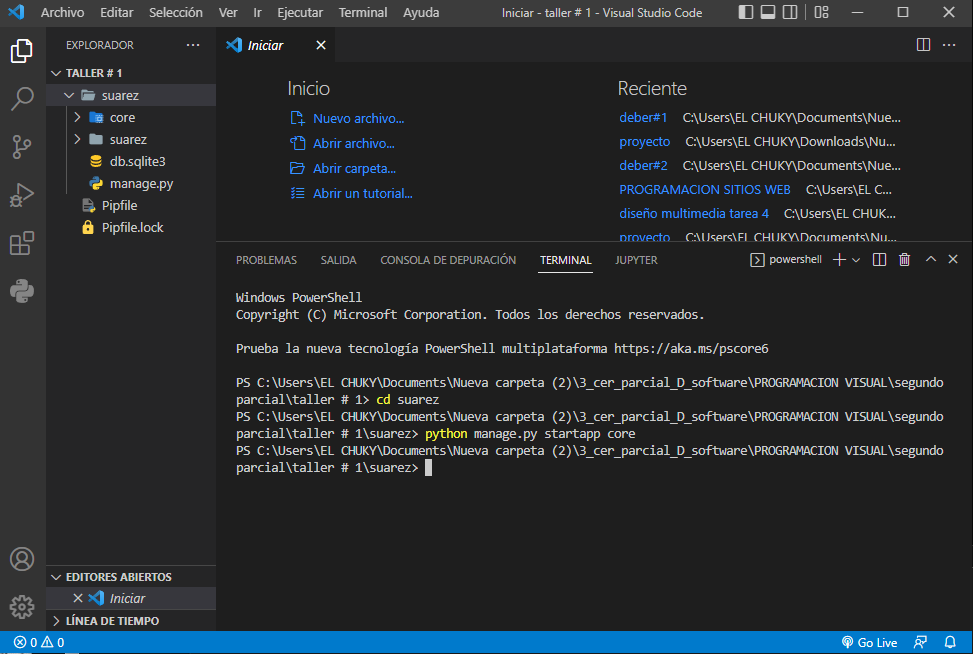
Interfaz de usuario gráfica, Texto, Sitio web

Descripción generada automáticamenteproyecto creado con éxito.

**8) Crear una Apps Core.**

* En nuestro terminal nos dirigimos a nuestra carpeta del proyecto creada y procedemos a crear nuestra Core con el comando: Python manage.py startapp Core



observamos que ya se encuentra creada nuestra Core.

**9) ¿Qué es la Carpeta Templates?**

Es una carpeta donde podemos crear subcarpeta en la cual podemos guardar nuestras plantillas que se muestren en el cuadro de dialogo crear un nuevo archivo. Podemos crear y guardar plantillas personalizadas.

**10) ¿Qué es la Carpeta stactic?**

Esta es una carpeta que contiene todos los archivos de estilos e imágenes .

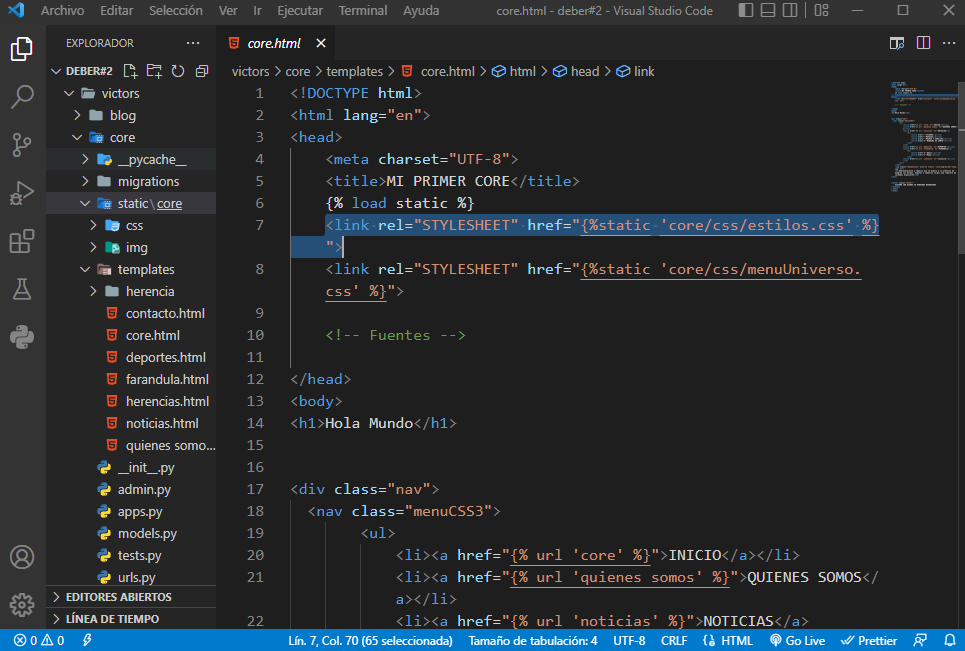
**11) Crear un archivo base HTML en la APPS Core**

* Texto

  Descripción generada automáticamenteEntramos a nuestra Core que creamos y creamos una carpeta témplate donde vamos a crear nuestro archivo HTML

**12) Como se llaman a los CSS desde el archivo base HTML.**

* En la Core creamos una carpeta llamada static y en ella creamos otra llamada Core donde pondremos nuestros archivos css e imágenes personalizados.
* Luego lo llamamos mediante un link desde el archivo HTML base que realizamos.

<link rel="STYLESHEET" href="{%static 'core/css/estilos.css' %}">

**13) Como consume un archivo hijo HTML al utilizar la herencia del**

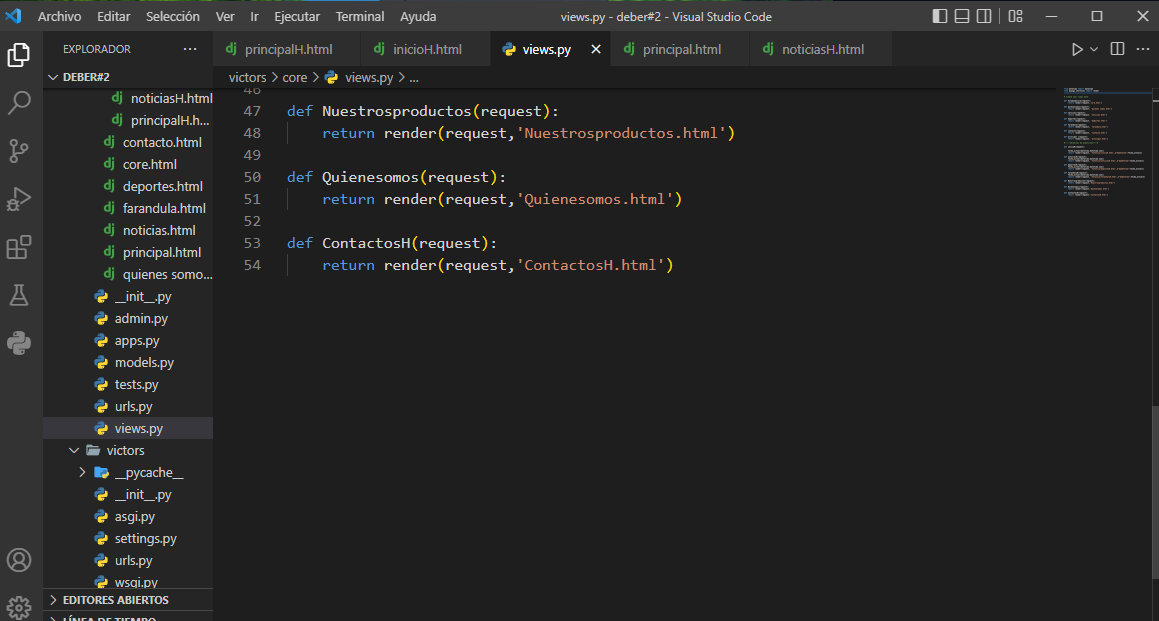
**archivo base HTML.**

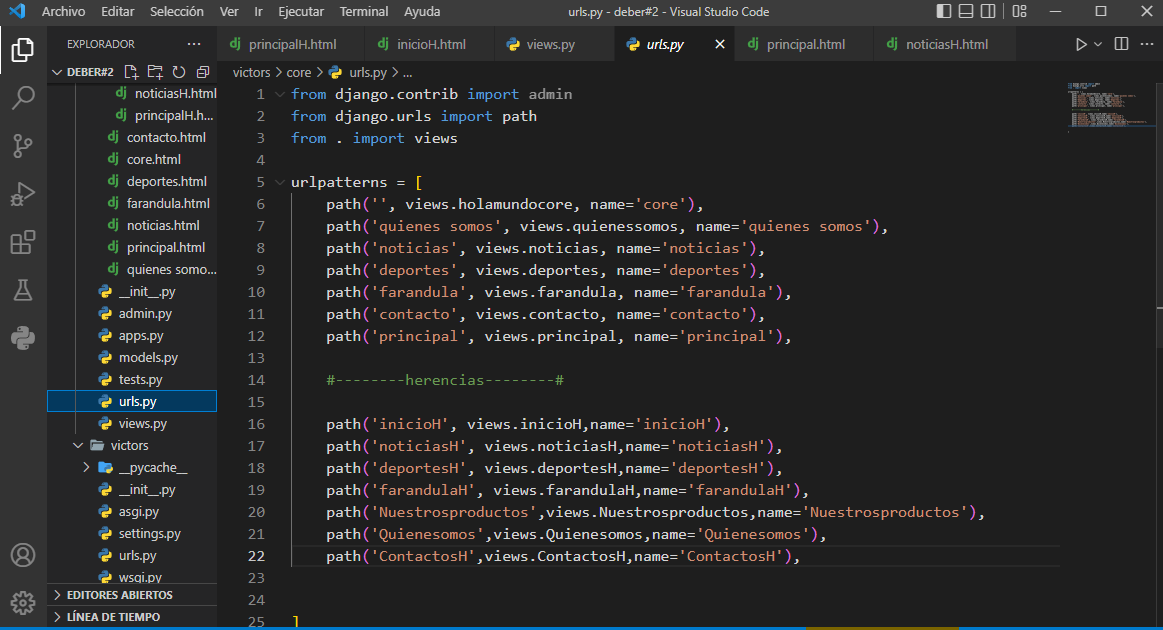
El archivo base es donde va a contener todos los datos comunes de nuestra página,

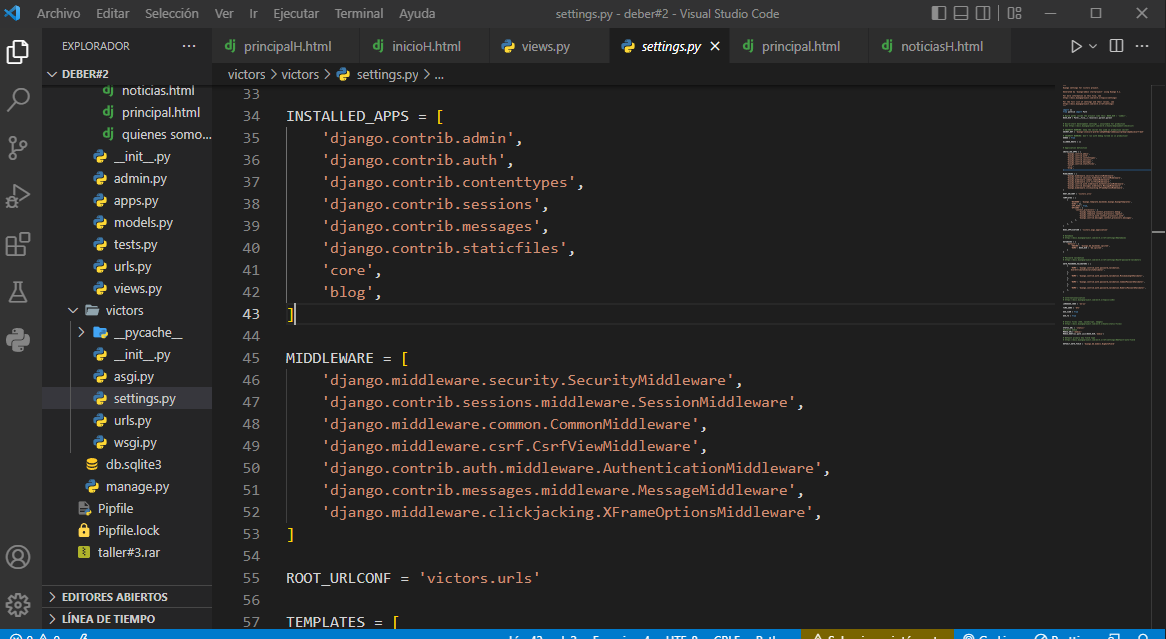
Captura de pantalla de computadora

Descripción generada automáticamenteDonde definimos bloques de contenido para los archivos hijos para cuando lo hereden.

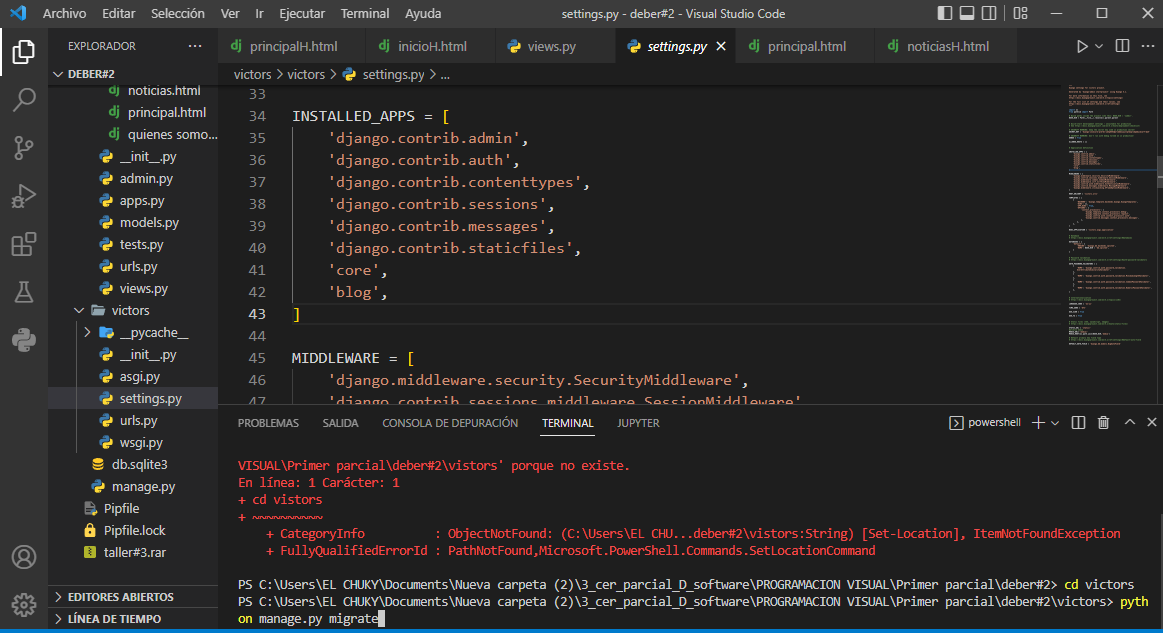
**14)Crear un views que llame al HTML hijo**

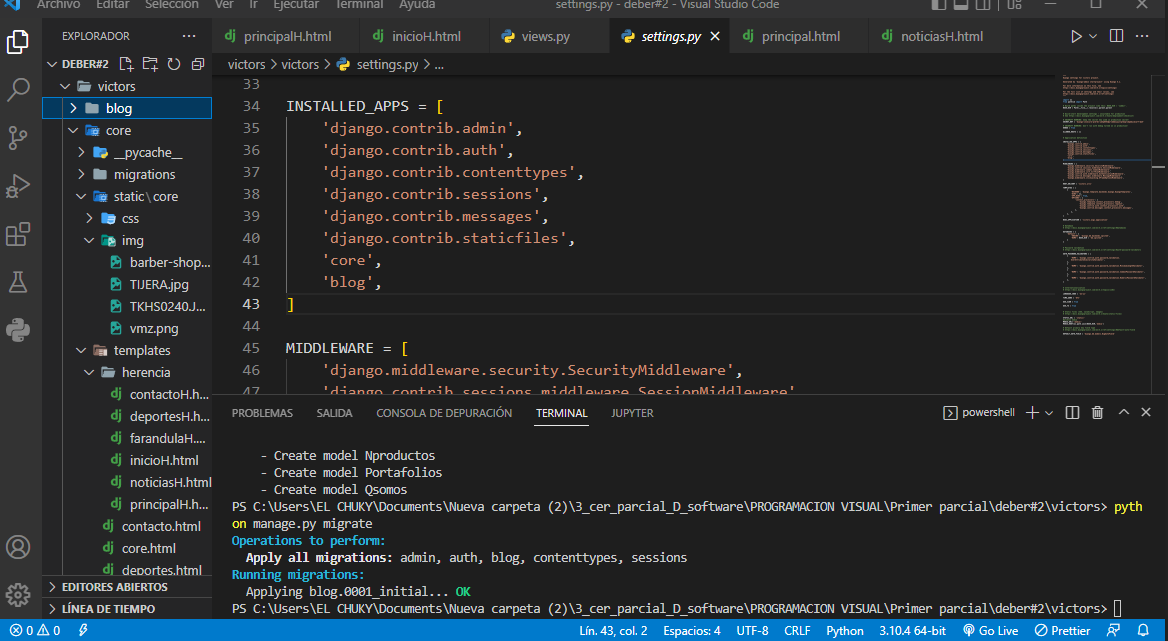


**15)Crear la urls que llame al views.**

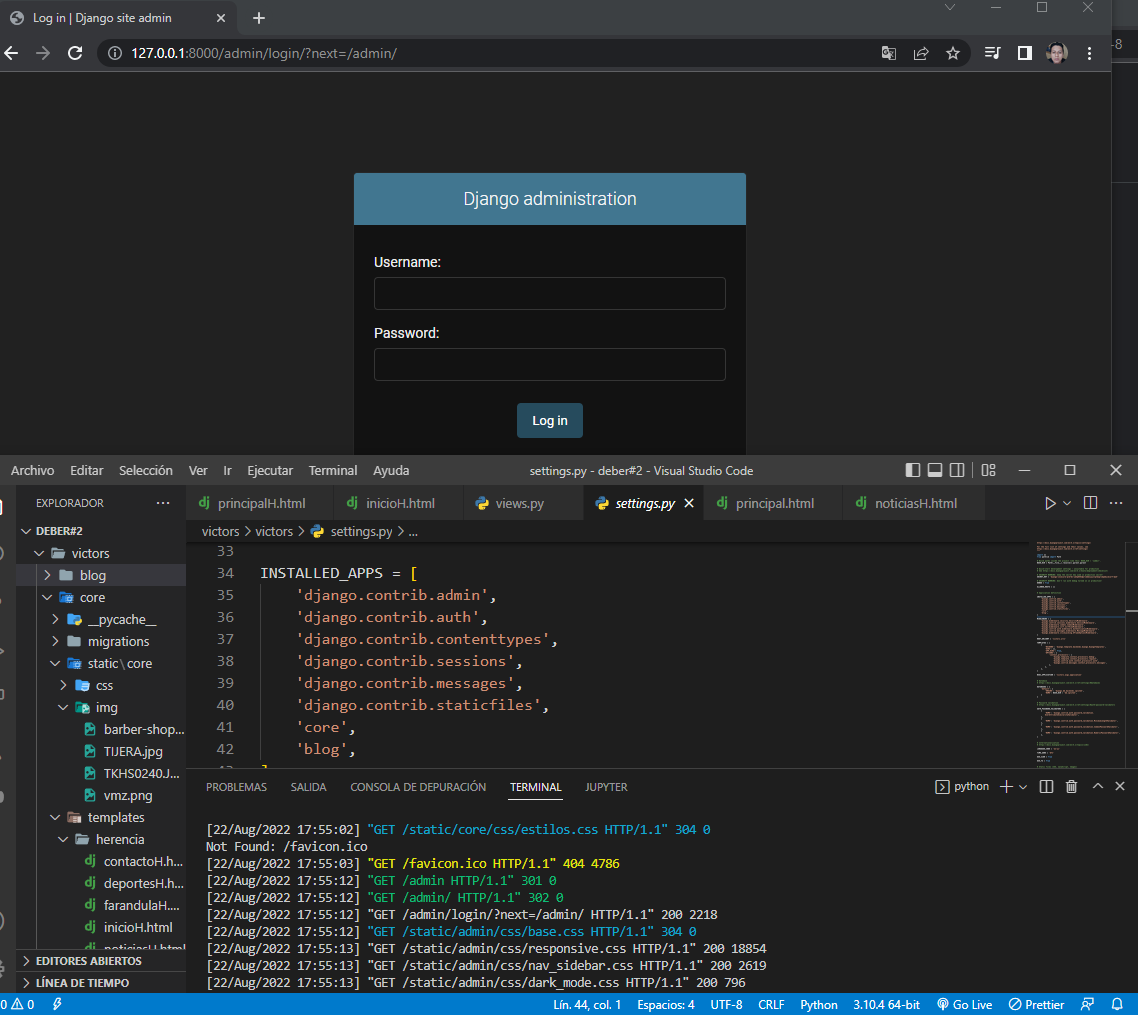
**16) Integrar la aplicación APPS Core al proyecto principal**

**17)Crear las tablas del sistema de usuarios para utilizar el panel de administración.**

* Nos dirigimos al proyecto y procedemos a crear las tablas con el comando: Python manage.py mígrate
* Esperamos que se ejecute el comando



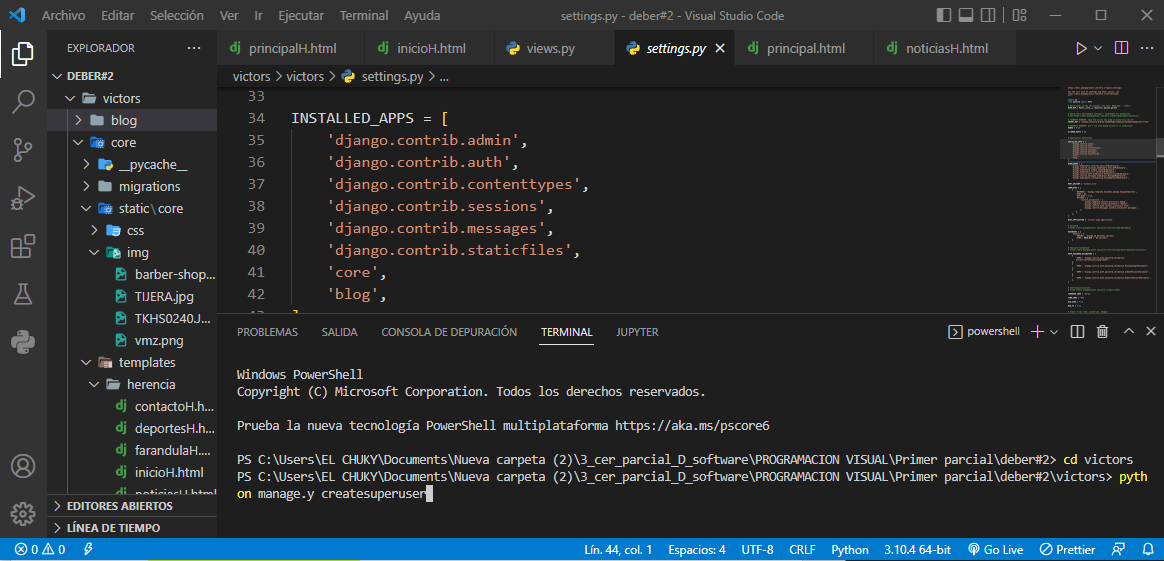
* Luego ejecutamos el proyecto y comprobamos que ya nos carga el admin.

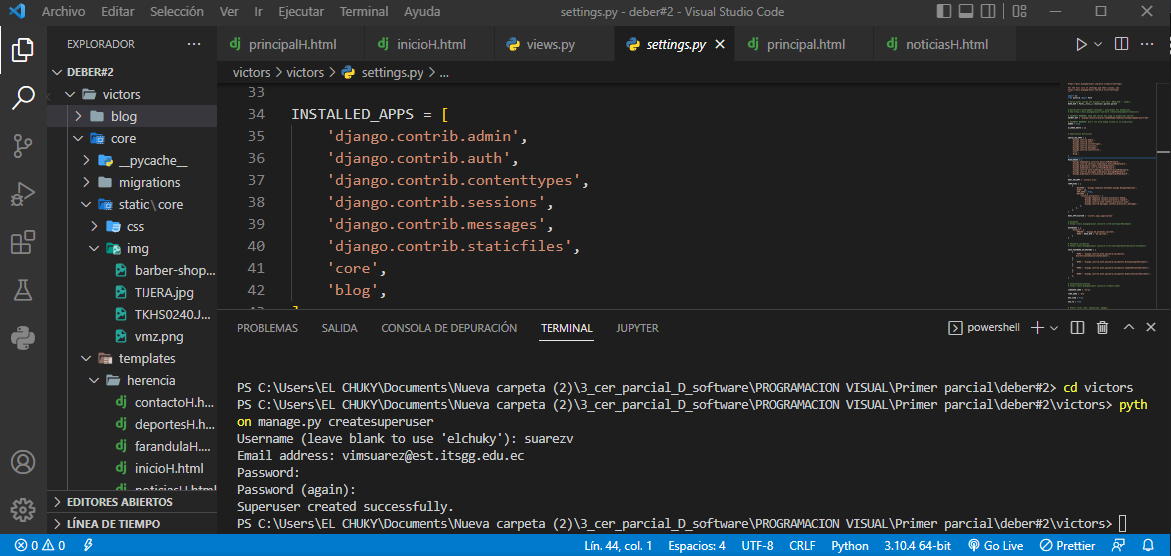


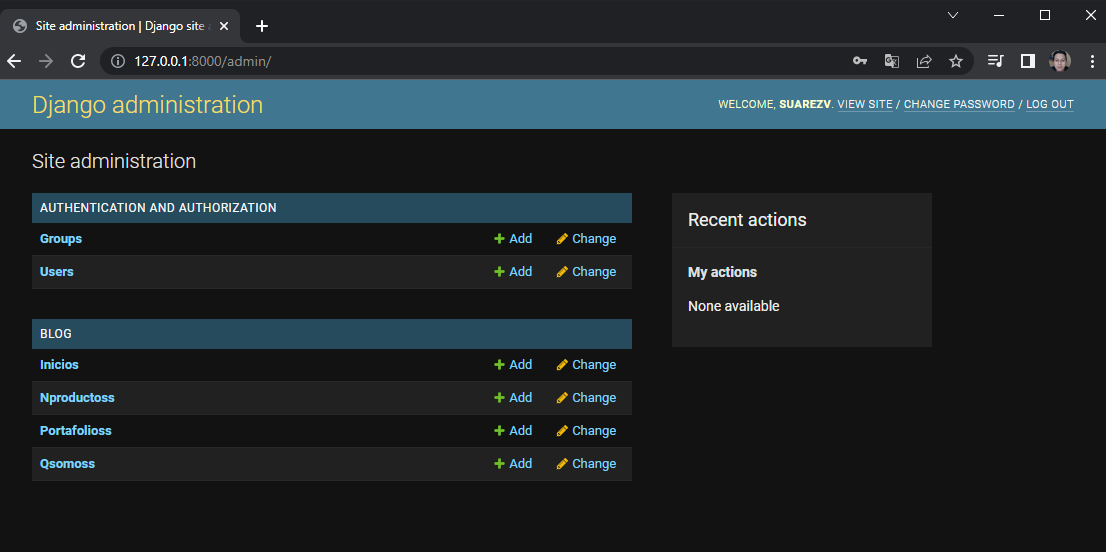
**18)Crear un usuario para poder ingresar al Panel de Administración**

* Creamos el usuario para poder ingresar en el admin con el comando:

Python manage.py createsuperuser

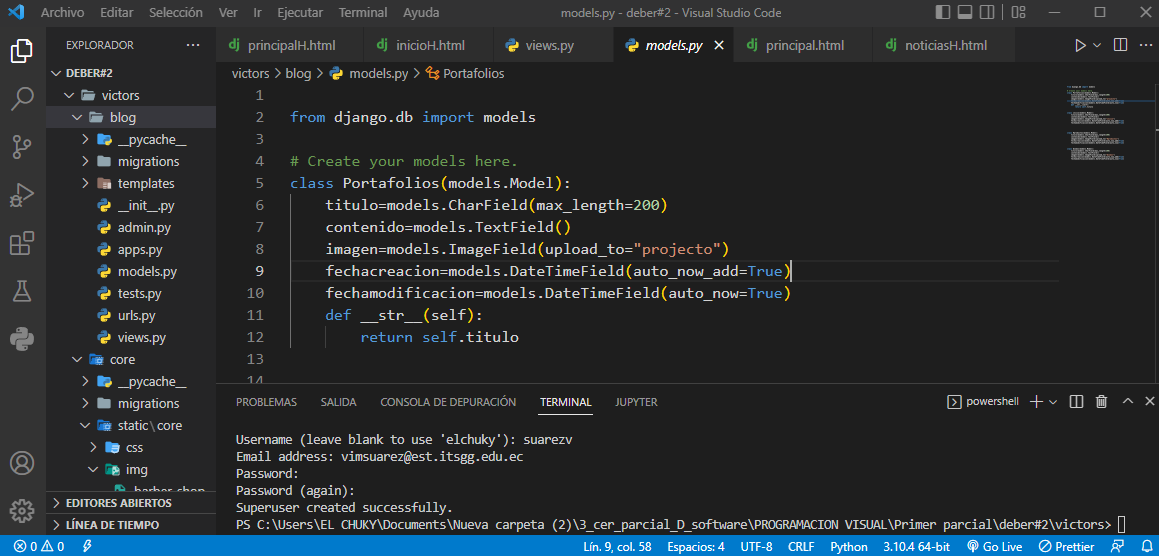


* Ejecutamos el comando y esperamos que nos cargue para crear el usuario.
* Luego ingresamos el usuario con el cual nos logiaremos en el admin.
* Después nos pide el correo por lo que usaremos el institucional.
* Luego nos pide un password (contraseña).
* Y la confirmación del password (contraseña).
* y luego se procede a crearse el usuario para poder ingresar en el admin.
* Ingresamos al admin para verificar si esta conectado con nuestro proyecto

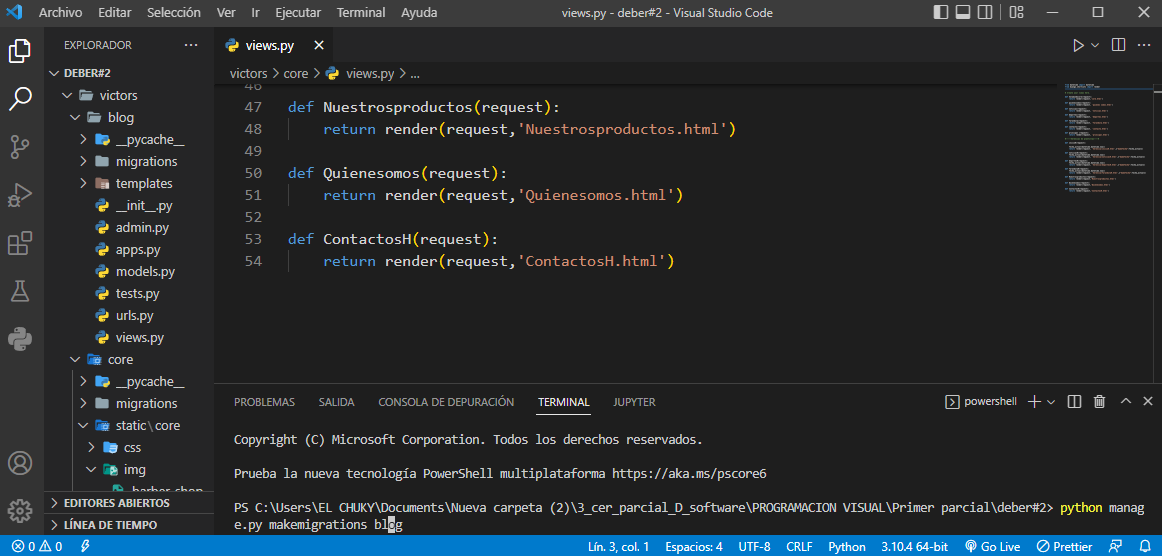


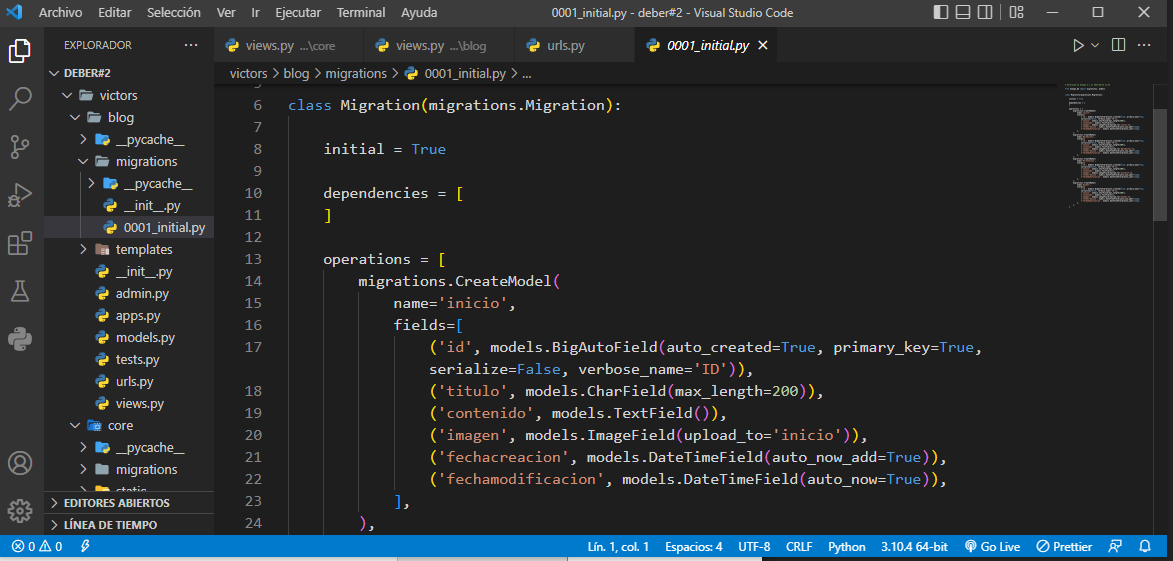
**19)Que es un modelo en Django**

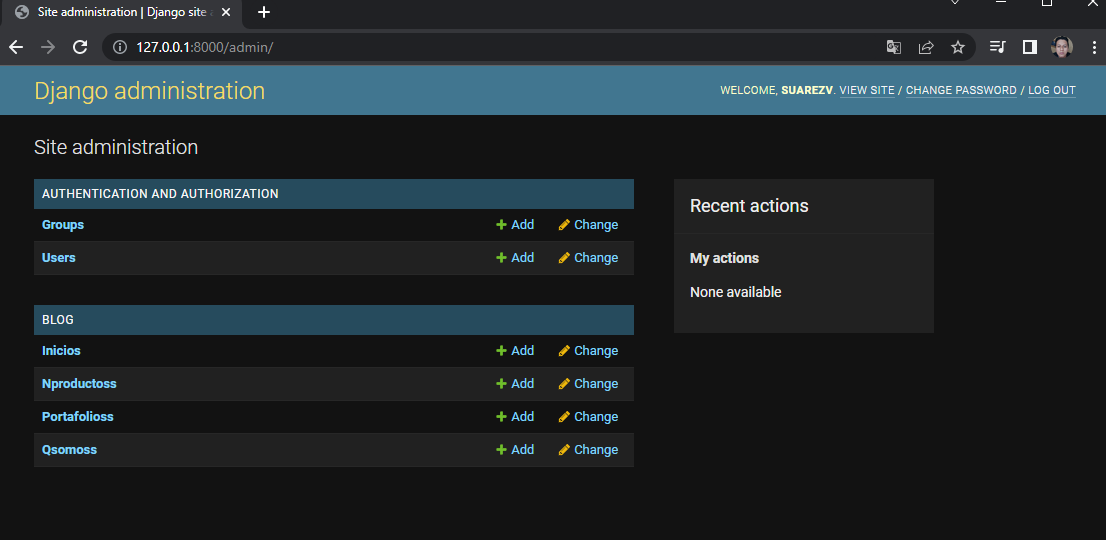
Un modelo en Django es un tipo especial de objeto que se guarda en la base de datos. Una base de datos es una colección de datos. Es un lugar en el cual almacenarás la información sobre usuarios, tus entradas de blog, etc.

**20)Crear un modelo en Django.**

21)Migrar el Modelo a la base del Panel de Administración.

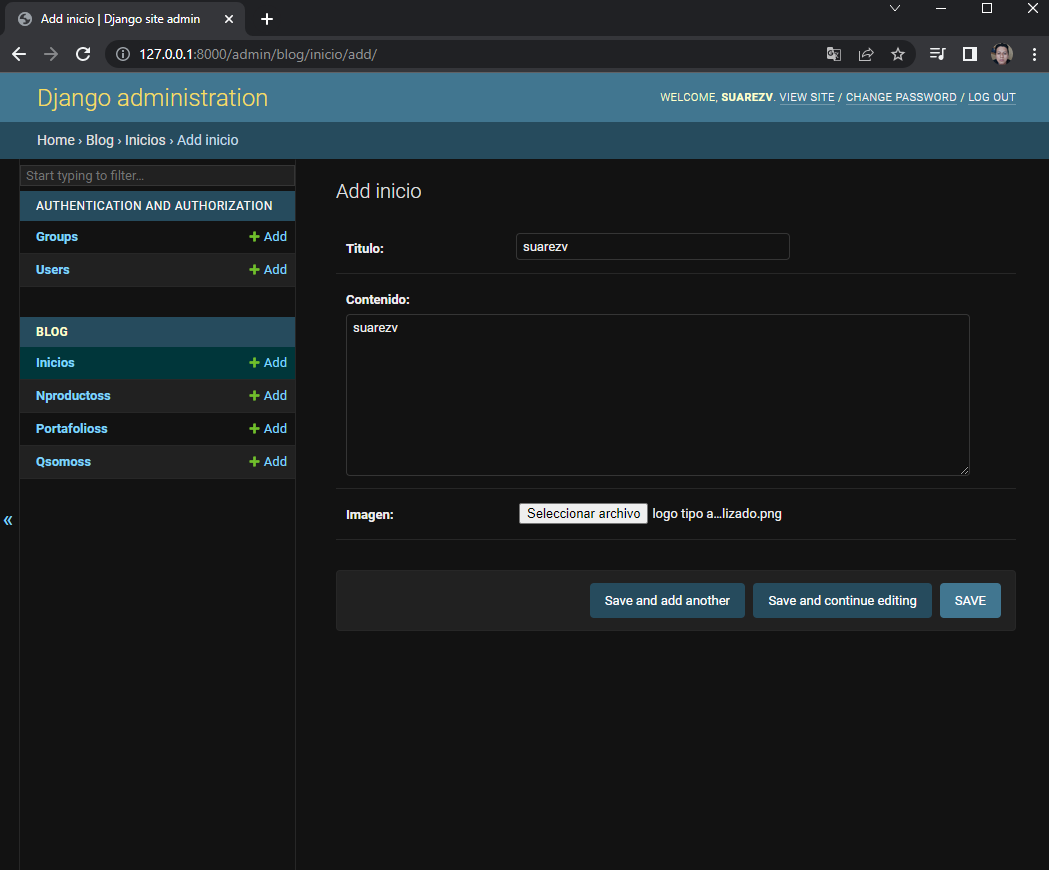
* Migramos con el comando: Python manage.py makemigrations blog
* Revisamos y vemos que ya se realizó la migración

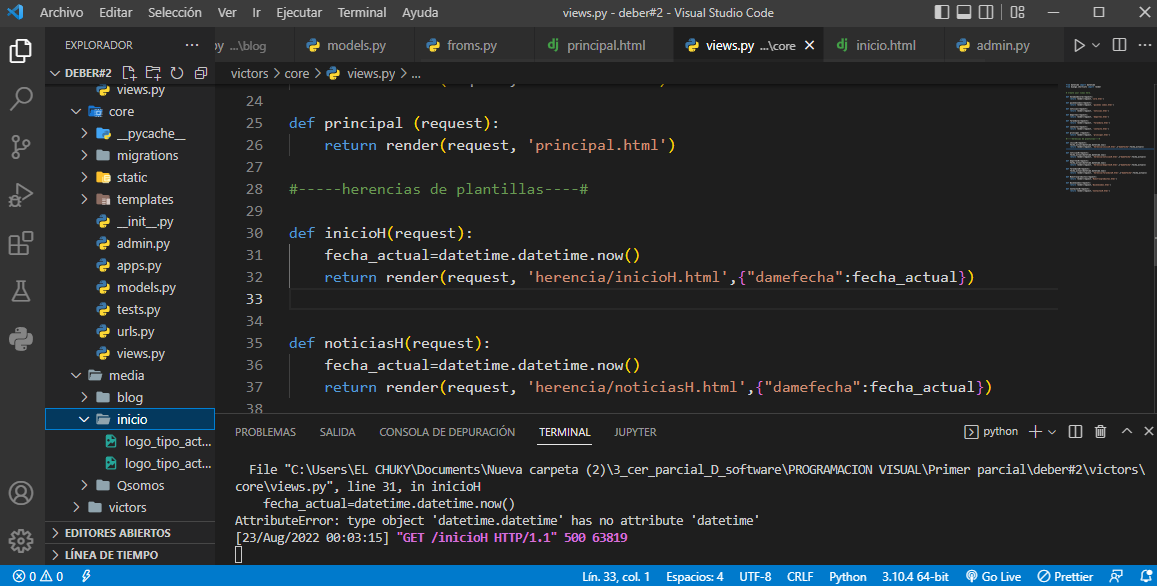


**22)Integrar el Modelo al Panel de Administración.**

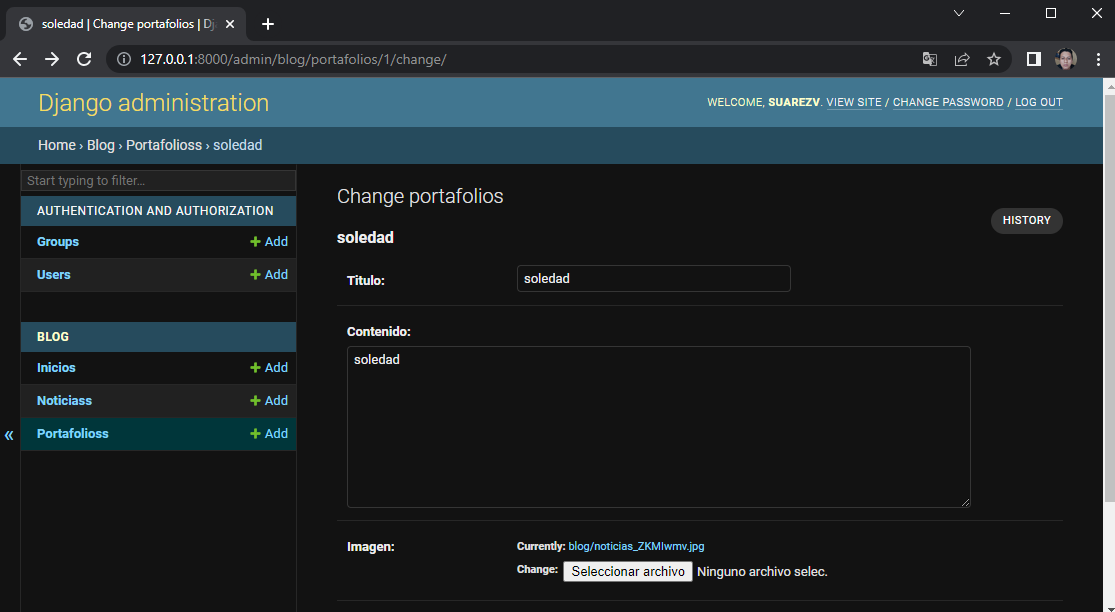
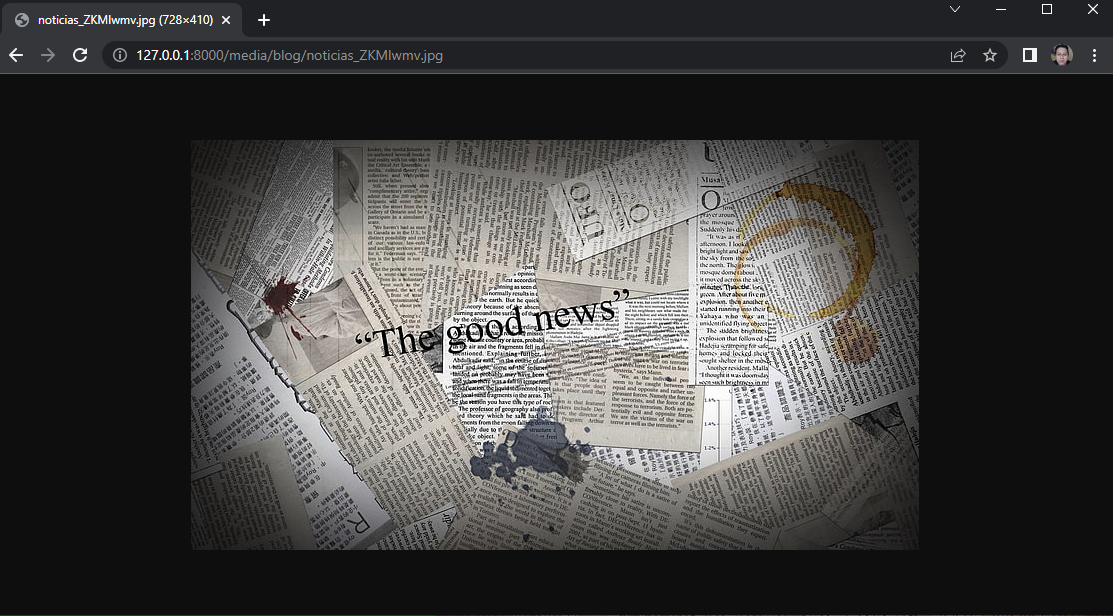
**23)Ingresar información al modelo por el Panel de Administración.**

* Nos dirigimos a administración
* Ponemos un título, comentario
* Y cargar una imagen
* Y damos clic en el botón guardar
* Y se cargara la información a nuestro proyecto



**24)Realizar la consulta de todo lo ingresado en el modelo desde el views.**

25)Mostrar los datos guardados en el modelo al HTML hijo

* revisamos para constatar que la imagen ingresada desde el admin al HTML hijo
* como observamos se guardo correctamente.
* Comprobamos que la pagina no se nos cae y nos permita visualizar la imagen guardada desde el admin.
* Ahora revisamos desde nuestro proyecto.

