# 🧱 Paso 1: Crear el proyecto de Django

**📌 Objetivo**

Iniciar un nuevo proyecto Django con una app llamada **core**.

**🖥️ Comando en la terminal**

**django-admin startproject myproject**

**cd myproject**

**python manage.py startapp core**

🔹 Esto crea la estructura base de Django: myproject es el proyecto, y core será nuestra app donde irá toda la lógica de negocio.

# 🧾 Paso 2: Configurar settings.py

**📌 Objetivo**

Activar la app core para que Django la reconozca.

**📂 Archivo: myproject/settings.py**

Busca la lista INSTALLED\_APPS y añade 'core', al final.

**INSTALLED\_APPS = [**

**'django.contrib.admin',**

**'django.contrib.auth',**

**'django.contrib.contenttypes',**

**'django.contrib.sessions',**

**'django.contrib.messages',**

**'django.contrib.staticfiles',**

**# Nuestra aplicación personalizada**

**'core',**

]

# 🧾 Paso 3: Crear el modelo de usuarios

**📌 Objetivo**

Crear un modelo simple que represente un usuario con nombre, login y contraseña.

**📂 Archivo: core/models.py**

**from django.db import models**

**# Modelo que representa un usuario simple para nuestro ejercicio**

**class UserProfile(models.Model):**

**name = models.CharField(max\_length=100) # Campo para el nombre del usuario**

**login = models.CharField(max\_length=100, unique=True) # Campo único para el nombre de usuario**

**password = models.CharField(max\_length=100) # Contraseña (en texto plano solo para práctica)**

**def \_\_str\_\_(self):**

**return self.name # Cómo se mostrará el usuario en el panel de administración o en consultas**

# 🧾 Paso 4: Crear la base de datos y aplicar migraciones

**📌 Objetivo**

Django usa migraciones para crear las tablas en la base de datos a partir de los modelos.

**🖥️ Comandos en la terminal**

**python manage.py makemigrations**

**python manage.py migrate**

🔹 makemigrations detecta cambios en los modelos  
🔹 migrate aplica esos cambios y crea las tablas en db.sqlite3

# 🧾 Paso 5: Crear el formulario para el alta de usuarios

**📌 Objetivo**

Definir un formulario con los campos name, login y password, basándonos en el modelo UserProfile.

**📂 Archivo: core/forms.py**

**from django import forms**

**from .models import UserProfile**

**# Formulario basado en el modelo UserProfile**

**class UserProfileForm(forms.ModelForm):**

**class Meta:**

**model = UserProfile # Usamos el modelo que ya definimos**

**fields = ['name', 'login', 'password'] # Campos visibles en el formulario**

**# Usamos PasswordInput para ocultar el texto en el campo de contraseña**

**widgets = {**

**'password': forms.PasswordInput()**

**}**

# 🧾 Paso 6: Crear las vistas

**📌 Objetivo**

Definir las funciones que responden cuando se accede a cada URL del sitio: inicio, productos, contacto (formulario) y gestión de usuarios.

**📂 Archivo: core/views.py**

**from django.shortcuts import render, redirect, get\_object\_or\_404**

**from .models import UserProfile**

**from .forms import UserProfileForm**

**# Página de inicio**

**def home(request):**

**return render(request, 'home.html') # Devuelve la plantilla de inicio**

**# Página de productos (por ahora estática)**

**def products(request):**

**return render(request, 'products.html') # Devuelve la plantilla de productos**

**# Página de contacto con formulario para registrar nuevos usuarios**

**def contact(request):**

**if request.method == 'POST':**

**# Si el formulario ha sido enviado, se procesa**

**form = UserProfileForm(request.POST)**

**if form.is\_valid():**

**form.save() # Guarda el nuevo usuario en la base de datos**

**return redirect('users') # Redirige a la lista de usuarios**

**else:**

**# Si es una petición GET, se muestra el formulario vacío**

**form = UserProfileForm()**

**return render(request, 'contact.html', {'form': form}) # Muestra la plantilla con el formulario**

**# Página para ver todos los usuarios registrados**

**def users(request):**

**user\_list = UserProfile.objects.all() # Consulta todos los usuarios**

**return render(request, 'users.html', {'users': user\_list})**

**# Página para editar un usuario existente**

**def user\_edit(request, pk):**

**user = get\_object\_or\_404(UserProfile, pk=pk) # Busca el usuario por su ID o devuelve error 404**

**form = UserProfileForm(request.POST or None, instance=user) # Formulario con los datos del usuario**

**if form.is\_valid():**

**form.save() # Guarda los cambios**

**return redirect('users')**

**return render(request, 'contact.html', {'form': form}) # Reutilizamos la plantilla de contacto**

**# Acción para eliminar un usuario**

**def user\_delete(request, pk):**

**user = get\_object\_or\_404(UserProfile, pk=pk)**

**user.delete() # Borra el usuario**

**return redirect('users')**

# 🧾 Paso 7: Definir las URLs de la app

**📌 Objetivo**

Conectar las rutas (como /, /contact/, etc.) con las vistas correspondientes.

**📂 Archivo: core/urls.py**

**from django.urls import path**

**from . import views**

**urlpatterns = [**

**path('', views.home, name='home'), # Ruta para la página de inicio**

**path('products/', views.products, name='products'), # Ruta para la página de productos**

**path('contact/', views.contact, name='contact'), # Ruta para el formulario de contacto**

**path('users/', views.users, name='users'), # Ruta para ver usuarios**

**path('edit/<int:pk>/', views.user\_edit, name='edit\_user'), # Ruta para editar**

**path('delete/<int:pk>/', views.user\_delete, name='delete\_user'), # Ruta para eliminar**

**]**

**📂 Archivo: myproject/urls.py**

Modifica este archivo para incluir las URLs de core:

**from django.contrib import admin**

**from django.urls import path, include**

**urlpatterns = [**

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

**path('', include('core.urls')), # Incluye las rutas de la app 'core'**

**]**

Vamos ahora a crear las **plantillas HTML** que forman las páginas de nuestra aplicación Django.

Incluiremos una plantilla base con un menú de navegación, y luego crearemos las plantillas específicas de cada vista: **inicio**, **productos**, **formulario de contacto**, y **usuarios registrados**.

# 🧾 Paso 8: Crear la plantilla base base.html

**📌 Objetivo**

Definir la estructura común (menú y diseño) que usaremos en todas las páginas.

**📂 Ruta: core/templates/base.html**

**<!DOCTYPE html>**

**<html lang="es">**

**<head>**

**<meta charset="UTF-8">**

**<title>Mi Tienda</title>**

**</head>**

**<body>**

**<!-- Menú de navegación principal -->**

**<nav>**

**<a href="{% url 'home' %}">Inicio</a> |**

**<a href="{% url 'products' %}">Productos</a> |**

**<a href="{% url 'contact' %}">Contacto</a> |**

**<a href="{% url 'users' %}">Usuarios</a>**

**</nav>**

**<hr>**

**<!-- Aquí se mostrará el contenido de cada página específica -->**

**{% block content %}**

**{% endblock %}**

**</body>**

**</html>**

# 🧾 Paso 9: Crear la página de inicio

**📂 Ruta: core/templates/home.html**

**{% extends "base.html" %}**

**{% block content %}**

**<h2>Bienvenido a la página de inicio</h2>**

**<p>Este es un sitio de práctica para aprender Django paso a paso.</p>**

**{% endblock %}**

# 🧾 Paso 10: Crear la página de productos

**📂 Ruta: core/templates/products.html**

**{% extends "base.html" %}**

**{% block content %}**

**<h2>Productos disponibles</h2>**

**<p>Aquí podrías listar productos si lo deseas más adelante.</p>**

**{% endblock %}**

# 🧾 Paso 11: Crear el formulario de contacto

**📂 Ruta: core/templates/contact.html**

**{% extends "base.html" %}**

**{% block content %}**

**<h2>Formulario de alta de usuario</h2>**

**<!-- Formulario que envía datos por POST -->**

**<form method="post">**

**{% csrf\_token %}**

**{{ form.as\_p }} <!-- Muestra el formulario como párrafos -->**

**<button type="submit">Guardar</button>**

**</form>**

**{% endblock %}**

# 🧾 Paso 12: Crear la página de usuarios registrados

**📂 Ruta: core/templates/users.html**

**{% extends "base.html" %}**

**{% block content %}**

**<h2>Usuarios registrados</h2>**

**<ul>**

**{% for user in users %}**

**<li>**

**{{ user.name }} ({{ user.login }})**

**[<a href="{% url 'edit\_user' user.pk %}">Editar</a>]**

**[<a href="{% url 'delete\_user' user.pk %}">Eliminar</a>]**

**</li>**

**{% empty %}**

**<li>No hay usuarios registrados aún.</li>**

**{% endfor %}**

**</ul>**

**{% endblock %}**

**✅ Resumen hasta ahora:**

Ya tienes:

1. Proyecto y app creados.
2. Modelo definido.
3. Formulario con validación.
4. Vistas que conectan la lógica con la base de datos.
5. URLs bien configuradas.
6. Plantillas HTML básicas y funcionales.

# Puesta en marcha:

**🧾 Paso 13: Ejecutar el servidor y probar la app**

**✅ Asegúrate de estar en la carpeta raíz del proyecto, donde está manage.py.**

**🖥️ 1. Crear y activar un entorno virtual (opcional, pero recomendado)**

**python -m venv venv**

**venv\Scripts\activate # En Windows**

**# source venv/bin/activate # En Linux/macOS**

**🖥️ 2. Instalar Django**

**pip install django**

**🖥️ 3. Crear las migraciones y aplicar los cambios**

**python manage.py makemigrations**

**python manage.py migrate**

✅ Esto creará la base de datos db.sqlite3 con la tabla core\_userprofile.

**🖥️ 4. Ejecutar el servidor de desarrollo**

**python manage.py runserver**

**🌐 5. Acceder a la aplicación en tu navegador**

Abre esta URL en el navegador:

[**http://127.0.0.1:8000/**](http://127.0.0.1:8000/)

Ahora puedes probar cada ruta:

| **Página** | **URL** |
| --- | --- |
| Inicio | http://127.0.0.1:8000/ |
| Productos | http://127.0.0.1:8000/products/ |
| Contacto | http://127.0.0.1:8000/contact/ |
| Usuarios | http://127.0.0.1:8000/users/ |

# 🧾 Paso 14: Añadir Bootstrap a las plantillas

**📌 Objetivo**

Incluir Bootstrap desde un CDN y adaptar el formulario para que use sus clases.

**🛠️ 1. Editar base.html para incluir Bootstrap**

📂 Archivo: core/templates/base.html

**<!DOCTYPE html>**

**<html lang="es">**

**<head>**

**<meta charset="UTF-8">**

**<title>Mi Tienda</title>**

**<!-- Bootstrap 5 desde CDN -->**

**<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet">**

**</head>**

**<body class="bg-light">**

**<div class="container py-4">**

**<!-- Menú de navegación principal -->**

**<nav class="mb-4">**

**<a href="{% url 'home' %}" class="btn btn-outline-primary me-2">Inicio</a>**

**<a href="{% url 'products' %}" class="btn btn-outline-primary me-2">Productos</a>**

**<a href="{% url 'contact' %}" class="btn btn-outline-primary me-2">Contacto</a>**

**<a href="{% url 'users' %}" class="btn btn-outline-primary">Usuarios</a>**

**</nav>**

**<!-- Aquí se mostrará el contenido de cada página específica -->**

**{% block content %}**

**{% endblock %}**

**</div>**

**</body>**

**</html>**

# 🛠️ Modificar contact.html para que el formulario use Bootstrap

📂 Archivo: core/templates/contact.html

**{% extends "base.html" %}**

**{% block content %}**

**<h2 class="mb-4">Formulario de alta de usuario</h2>**

**<form method="post" class="border p-4 bg-white rounded shadow-sm">**

**{% csrf\_token %}**

**<!-- Aplicamos clases Bootstrap a cada campo manualmente -->**

**{% for field in form %}**

**<div class="mb-3">**

**<label class="form-label">{{ field.label }}</label>**

**{{ field }}**

**{% if field.errors %}**

**<div class="text-danger small">{{ field.errors|striptags }}</div>**

**{% endif %}**

**</div>**

**{% endfor %}**

**<button type="submit" class="btn btn-primary">Guardar</button>**

**</form>**

**{% endblock %}**

Vamos a llevar el diseño al siguiente nivel añadiendo:

* ✅ Un **header fijo profesional** con branding.
* ✅ Un **footer con información corporativa y redes sociales**.
* ✅ Un poco de **CSS personalizado** para refinar estilos.
* ✅ Bootstrap seguirá siendo la base del diseño.

# 🧾 Paso 15: Mejorar base.html con header, footer y estilos

**📂 Archivo: core/templates/base.html**

**<!DOCTYPE html>**

**<html lang="es">**

**<head>**

**<meta charset="UTF-8">**

**<title>Mi Tienda</title>**

**<!-- Bootstrap 5 desde CDN -->**

**<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet">**

**<!-- Iconos Bootstrap -->**

**<link href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.10.5/font/bootstrap-icons.css" rel="stylesheet">**

**<style>**

**body {**

**padding-top: 70px;**

**background-color: #f8f9fa;**

**}**

**footer {**

**background-color: #212529;**

**color: white;**

**padding: 2rem 0;**

**margin-top: 4rem;**

**}**

**footer a {**

**color: #adb5bd;**

**text-decoration: none;**

**}**

**footer a:hover {**

**color: white;**

**}**

**</style>**

**</head>**

**<body>**

**<!-- HEADER -->**

**<nav class="navbar navbar-expand-lg navbar-dark bg-dark fixed-top shadow-sm">**

**<div class="container">**

**<a class="navbar-brand" href="{% url 'home' %}">⚽ Mi Tienda</a>**

**<div>**

**<a href="{% url 'products' %}" class="btn btn-outline-light btn-sm me-2">Productos</a>**

**<a href="{% url 'contact' %}" class="btn btn-outline-light btn-sm me-2">Contacto</a>**

**<a href="{% url 'users' %}" class="btn btn-outline-light btn-sm">Usuarios</a>**

**</div>**

**</div>**

**</nav>**

**<!-- CONTENIDO -->**

**<main class="container mt-4">**

**{% block content %}**

**{% endblock %}**

**</main>**

**<!-- FOOTER -->**

**<footer class="mt-auto">**

**<div class="container text-center">**

**<div class="row">**

**<div class="col-md-4 mb-3 mb-md-0">**

**<p class="mb-0">© 2025 Mi Tienda</p>**

**</div>**

**<div class="col-md-4 mb-3 mb-md-0">**

**<a href="#">Preguntas frecuentes</a> |**

**<a href="#">Dónde estamos</a>**

**</div>**

**<div class="col-md-4">**

**<a href="#"><i class="bi bi-facebook me-2"></i></a>**

**<a href="#"><i class="bi bi-twitter me-2"></i></a>**

**<a href="#"><i class="bi bi-instagram"></i></a>**

**</div>**

**</div>**

**</div>**

**</footer>**

**</body>**

**</html>**

# 🧾 Paso 16: Enlace a una localización ficticia en Google Maps

**📌 Objetivo**

Al hacer clic en "Dónde estamos", se abrirá una nueva pestaña con un mapa de Madrid, por ejemplo:

**"Calle Falsa 123, Madrid"**

**🔗 Enlace a usar:**

**https://www.google.com/maps?q=Calle+Falsa+123,+Madrid**

**📂 Modifica el bloque de footer en base.html**

Reemplaza esta parte:

**<a href="#">Dónde estamos</a>**

Por esto:

**<a href="https://www.google.com/maps?q=Calle+Falsa+123,+Madrid" target="\_blank" rel="noopener noreferrer">Dónde estamos</a>**

✅ target="\_blank" abre en una nueva pestaña.  
✅ rel="noopener noreferrer" es una buena práctica de seguridad.

**🔄 Footer completo actualizado:**

**<footer class="mt-auto">**

**<div class="container text-center">**

**<div class="row">**

**<div class="col-md-4 mb-3 mb-md-0">**

**<p class="mb-0">© 2025 Mi Tienda</p>**

**</div>**

**<div class="col-md-4 mb-3 mb-md-0">**

**<a href="#">Preguntas frecuentes</a> |**

**<a href="https://www.google.com/maps?q=Calle+Falsa+123,+Madrid" target="\_blank" rel="noopener noreferrer">Dónde estamos</a>**

**</div>**

**<div class="col-md-4">**

**<a href="#"><i class="bi bi-facebook me-2"></i></a>**

**<a href="#"><i class="bi bi-twitter me-2"></i></a>**

**<a href="#"><i class="bi bi-instagram"></i></a>**

**</div>**

**</div>**

**</div>**

**</footer>**