

Módulo 6 – Desarrollo de Aplicaciones Web con Python Django

1) Propósito

Implementar un sistema de **autenticación de usuarios** en una aplicación web desarrollada con Django, incorporando páginas de inicio de sesión y cierre de sesión, así como vistas protegidas que permitan diferenciar el acceso a determinadas secciones del e-commerce.

2) Objetivos de aprendizaje

- Crear y configurar un proyecto Django.
- Implementar vistas y templates con contenido dinámico.
- Utilizar el sistema de autenticación de Django para login y logout.
- Implementar formularios para captura de datos de usuario.
- Proteger vistas para restringir el acceso a usuarios no autenticados.

3) Alcance del ejercicio (MVP)

3.1 Inicio de sesión y cierre de sesión

La aplicación debe permitir:

- **Inicio de sesión (login)** mediante un formulario.
- **Cierre de sesión (logout)** desde la aplicación.

Al iniciar sesión correctamente, el usuario debe ser redirigido a una sección interna del sistema.

3.2 Registro de usuario

Se debe implementar una página que permita **crear un nuevo usuario**, utilizando un formulario.

El registro debe considerar:

- Nombre de usuario.
- Contraseña.
- Confirmación de contraseña (si corresponde).

Una vez creado el usuario, este debe poder iniciar sesión en la aplicación.

3.3 Vistas protegidas

La aplicación debe incluir al menos una **vista protegida**, accesible solo para usuarios autenticados.

Ejemplos de vista protegida:

- Panel del usuario.
- Página de administración básica.
- Página de gestión de productos (solo visualización).

Si un usuario no autenticado intenta acceder a esta vista, debe ser redirigido al login.

3.4 Navegación y templates

- Implementar un **template base** (base.html) con estructura común.
- Utilizar herencia de templates para las distintas páginas.
- La navegación debe incluir:
 - Enlace a login cuando el usuario no ha iniciado sesión.
 - Enlace a logout cuando el usuario está autenticado.
 - Enlace a la vista protegida.

4) Requisitos técnicos

- Proyecto Django ejecutable con manage.py runserver.
- Rutas definidas en urls.py.
- Vistas que rendericen templates con contexto.
- Uso de formularios mediante Django Forms.
- Uso del sistema de autenticación de Django para login, logout y creación de usuarios.
- Protección de vistas mediante autenticación.

5) Entregables

El estudiante debe entregar:

- Proyecto Django comprimido o repositorio.
- Código fuente con estructura ordenada.
- Archivo README.md que incluya:
 - Pasos para ejecutar el proyecto.
 - Rutas principales (login, registro, vista protegida).
 - Usuario de prueba (si se utiliza uno).
 - Evidencia (capturas) de:
 - registro de usuario
 - inicio de sesión
 - acceso a vista protegida

Rúbrica de evaluación

Criterio	Excelente (3 pts)	Adecuado (2 pts)	Básico (1 pt)	Insuficiente (0 pts)
Login y logout	Inicio y cierre de sesión funcionan correctamente y sin errores.	Funciona con pequeños detalles.	Implementación parcial o inestable.	No funciona o no está implementado.
Registro de usuario	Registro funcional; permite crear usuarios y luego iniciar sesión.	Registro funciona con validaciones limitadas.	Registro incompleto o confuso.	No se implementa registro.
Vistas protegidas	Vista protegida correctamente restringida a usuarios autenticados.	Protección funcional con detalles menores.	Protección parcial o poco clara.	No hay vistas protegidas.
Templates y navegación	Uso correcto de template base y navegación clara según estado de sesión.	Templates correctos con pequeños problemas.	Navegación confusa o incompleta.	Páginas rotas o sin estructura.
Estructura del proyecto	Proyecto ordenado y fácil de comprender.	Organización general correcta.	Estructura poco clara.	Proyecto desordenado o no ejecuta.
Documentación	README claro y suficiente para ejecutar y probar el sistema.	README útil pero incompleto.	README poco claro.	No presenta README.