

Gestión de Login y Sesiones en Flask

Módulo: 11_LoginSession

1 11_LoginSession

1.1 Descripción General

El módulo **11_LoginSession** implementa un sistema de autenticación de usuarios en una aplicación web desarrollada con **Flask**. Este módulo permite validar credenciales almacenadas en una base de datos MySQL y gestionar la información del usuario mediante sesiones.

El sistema incluye páginas de inicio, login, validación, acceso al contenido y manejo de errores, utilizando **Bootstrap** para mejorar la presentación visual.

1.2 Objetivo

Implementar un sistema de autenticación que permita:

- Validar usuarios contra una base de datos MySQL.
- Crear y mantener una sesión activa después del login.
- Mostrar información personalizada al usuario autenticado.
- Controlar el acceso mediante variables de sesión.
- Mostrar un mensaje de error en caso de credenciales incorrectas.

1.3 Arquitectura del Sistema

El sistema está compuesto por:

- **app.py**: controlador principal que gestiona rutas y sesiones.
- **Base de datos MySQL**: base *sesionesbd* con la tabla *usuarios*.
- **Templates HTML**:
 - index.html (pantalla principal)
 - home.html (pantalla de selección)
 - login.html (formulario de autenticación)
 - error.html (mensaje de error)

1.4 Validación de Usuario

El sistema utiliza la función:

```
validar_usuario_mysql(usuario, clave)
```

Esta función:

- Se conecta a MySQL.
- Consulta la tabla *usuarios*.
- Verifica si existe el usuario con la contraseña proporcionada.
- Retorna verdadero si el usuario es válido.

1.5 Proceso de Login

El flujo de autenticación es el siguiente:

1. El usuario accede al formulario de login.
2. Ingresa usuario y contraseña.
3. Se envía la información mediante POST a la ruta */validar*.
4. Si las credenciales son correctas:
 - Se crea la sesión:

```
session["usuario"] = usuario
session["login"] = True
```
 - Se muestra el contenido personalizado.
5. Si las credenciales son incorrectas:
 - Se muestra la página de error.

1.6 Uso de Sesiones

Además del usuario autenticado, el sistema almacena en sesión:

- Lista de notebooks disponibles.
- Marca seleccionada.
- Estado de autenticación.

Las variables de sesión se muestran en pantalla para fines de verificación y aprendizaje.

1.7 Contenido Después del Login

Una vez autenticado, el sistema:

- Muestra un mensaje de bienvenida.
- Presenta información del notebook seleccionado.
- Muestra el precio correspondiente.
- Permite continuar hacia la página principal.

1.8 Seguridad de Sesión

Para el manejo de sesiones, la aplicación define una clave secreta:

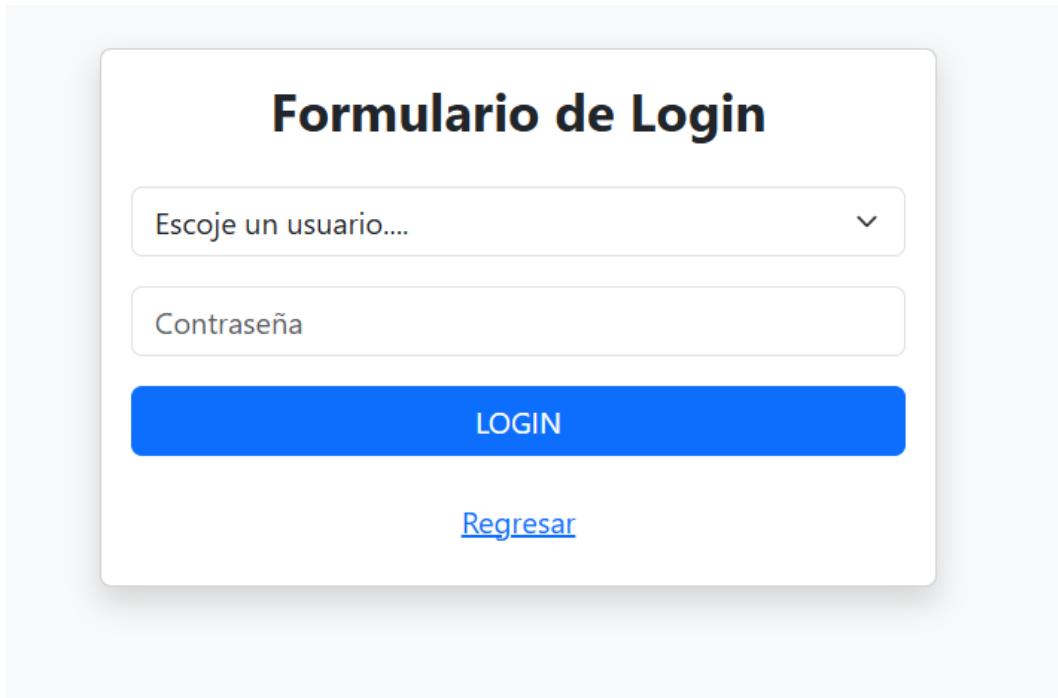
```
app.secret_key
```

Esto permite firmar las cookies y proteger la integridad de los datos de sesión.

1.9 Resultados Esperados

Al ejecutar el sistema, el usuario podrá:

- Acceder a la página principal.
- Ingresar credenciales mediante el formulario de login.
- Ser autenticado contra la base de datos.
- Visualizar contenido personalizado.
- Ver las variables almacenadas en sesión.
- Recibir un mensaje de error si las credenciales son incorrectas.



The image shows a login form titled "Formulario de Login". It contains two input fields: a dropdown menu labeled "Escoje un usuario...." and a password field labeled "Contraseña". Below the fields is a blue "LOGIN" button. At the bottom right is a link labeled "Regresar".

Figura 1: Formulario de Login

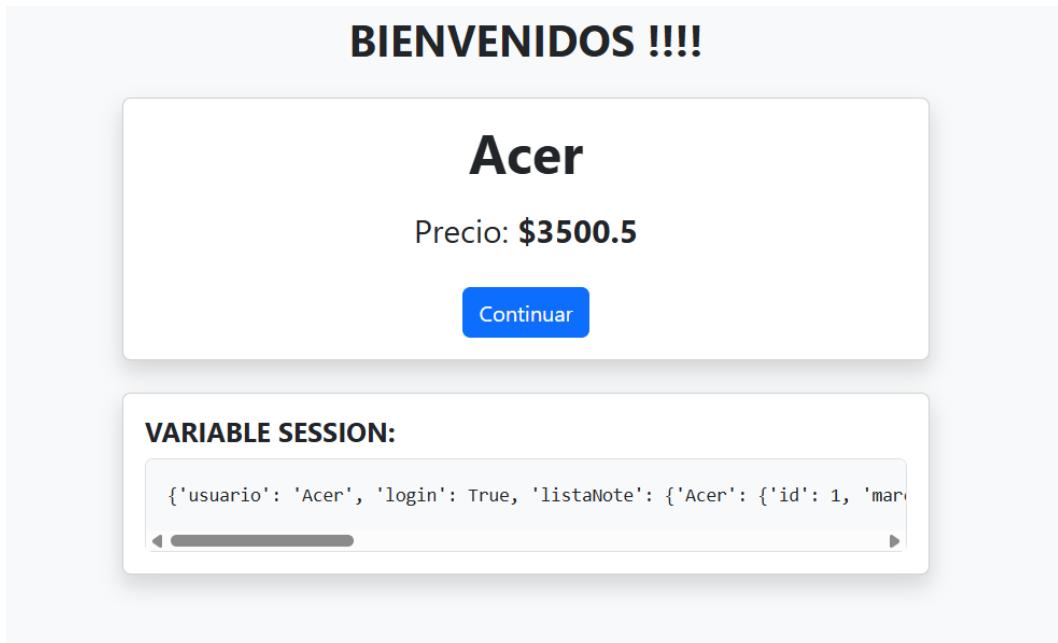


Figura 2: Pantalla después de autenticación

1.10 Importancia del Módulo

Este módulo introduce el concepto de autenticación en aplicaciones web, permitiendo controlar el acceso a los recursos del sistema. El uso de sesiones es fundamental para sistemas reales como plataformas educativas, sistemas administrativos, carritos

de compra y aplicaciones empresariales, ya que permite mantener el estado del usuario durante su navegación.