Documento Técnico

Funcionalidad CRUD del Sistema

Plataforma web

“La Picá de la Chabelita”

Introducción

Este documento describe las funcionalidades básicas de un sistema basado en operaciones CRUD (Crear, Leer, Actualizar, Eliminar). Abarca los aspectos generales de cómo se gestionan las entidades dentro del sistema, proporcionando una visión detallada de las operaciones que se pueden realizar en cada módulo.

1. Entidades del Sistema

Las principales entidades del sistema incluyen:

* **Producto**
* **Categoría**
* **Usuario**
* **Pedido**
* **Pago**

Cada entidad tiene una serie de atributos y funcionalidades asociadas que permiten gestionar la información de manera eficaz.

2. Operaciones CRUD

**2.1. Crear**

**Descripción**: Permite la creación de nuevas instancias de las entidades del sistema.

**Operaciones**:

* **Producto**: Crear un nuevo producto con nombre, descripción, precio, stock, etc.
* **Categoría**: Crear una nueva categoría para clasificar productos.
* **Usuario**: Registrar un nuevo usuario en el sistema.
* **Pedido**: Crear un nuevo pedido con detalles de usuario y productos.
* **Pago**: Registrar un nuevo pago asociado a un pedido.

**Formularios y Vistas**:

* Formularios en HTML o Django Forms para ingresar datos.
* Vistas en Django que manejan la lógica de creación y validación de datos.

**2.2. Leer**

**Descripción**: Permite la visualización y consulta de la información almacenada en el sistema.

**Operaciones**:

* **Producto**: Consultar la lista de productos o los detalles de un producto específico.
* **Categoría**: Ver todas las categorías disponibles o los detalles de una categoría.
* **Usuario**: Consultar la información de un usuario o listar todos los usuarios registrados.
* **Pedido**: Ver los pedidos realizados por un usuario o el historial de pedidos.
* **Pago**: Consultar el estado de los pagos realizados.

**Vistas**:

* Listado de entidades y detalles individuales.
* Uso de vistas basadas en clase o funciones en Django para la presentación de datos.

**2.3. Actualizar**

**Descripción**: Permite la modificación de los datos existentes en el sistema.

**Operaciones**:

* **Producto**: Actualizar la información de un producto, como el precio o la cantidad en stock.
* **Categoría**: Modificar el nombre o la descripción de una categoría.
* **Usuario**: Actualizar los datos del perfil de un usuario.
* **Pedido**: Cambiar el estado de un pedido o actualizar los detalles de la dirección.
* **Pago**: Actualizar el estado de un pago.

**Formularios y Vistas**:

* Formularios pre-rellenos con los datos actuales de la entidad.
* Vistas en Django que manejan la actualización de datos y validaciones.

**2.4. Eliminar**

**Descripción**: Permite la eliminación de instancias de las entidades del sistema.

**Operaciones**:

* **Producto**: Eliminar un producto que ya no está disponible.
* **Categoría**: Eliminar una categoría (si no tiene productos asociados).
* **Usuario**: Eliminar un usuario del sistema.
* **Pedido**: Eliminar un pedido, si es necesario.
* **Pago**: Eliminar un registro de pago.

**Vistas**:

* Confirmación de eliminación antes de proceder.
* Vistas en Django para manejar la eliminación segura y verificar las dependencias.

3. Implementación en Django

**3.1. Modelos**

Los modelos en Django representan las entidades del sistema y definen sus atributos y relaciones. Cada modelo tiene un conjunto de métodos que permiten realizar operaciones CRUD sobre la base de datos.

**3.3. Formularios**

Django Forms se utilizan para la validación y procesamiento de datos de entrada del usuario. Los formularios permiten crear, actualizar y eliminar entidades a través de una interfaz amigable.

**3.4. URL. Vistas**

Las vistas en Django gestionan la lógica para mostrar y procesar datos. Se utilizan vistas basadas en clases (CBV) o vistas basadas en funciones (FBV) para manejar las operaciones CRUD.

Las URLs definen las rutas para acceder a las vistas CRUD. Cada vista CRUD tiene una URL específica que permite realizar las operaciones correspondientes.

**3.5. Plantillas**

Las plantillas en Django se utilizan para presentar la información al usuario. Incluyen formularios para crear y actualizar datos, así como vistas detalladas y listados.

Conclusión

El sistema implementa funcionalidades CRUD para gestionar las entidades principales: productos, categorías, usuarios, pedidos y pagos. Cada operación CRUD se realiza a través de formularios, vistas y plantillas en Django, asegurando una gestión efectiva de la información.

Este documento proporciona una visión general de cómo se manejan estas operaciones en el sistema, facilitando la comprensión y el mantenimiento del código.

Diccionario

CBV : Class-Based Views (Vistas Basadas en Clases)

FBV : Function-Based Views (Vistas Basadas en Funciones)