

Procedimiento de Implementación de Software

1. Introducción

La implementación de software es un proceso crítico que implica la planificación, instalación, configuración, pruebas, y despliegue de una aplicación o sistema de software. Este documento describe el procedimiento de implementación del proyecto "Tienda Virtual Construmole", una plataforma de comercio electrónico desarrollada con Django.

2. Planificación de la Implementación

2.1. Definición de Objetivos

Objetivo General: Implementar la plataforma de comercio electrónico "Construmole" de manera eficiente y sin interrupciones, garantizando que todas las funcionalidades estén operativas y cumplan con los requisitos del cliente.

Objetivos Específicos:

- Asegurar que todos los componentes del sistema estén correctamente configurados.
- Validar la funcionalidad de la plataforma mediante pruebas exhaustivas.
- Capacitar a los usuarios finales en el uso del sistema.

2.2 Levantamiento de requerimientos

Entrevistas

La recolección de requisitos es una etapa crítica donde se recopila toda la información necesaria para desarrollar el proyecto "Construmole". Este proceso incluye varias reuniones con el propietario de la tienda y otros stakeholders clave.

Actividades Clave de esta etapa:

- **Entrevistas con el cliente:** Se realizan entrevistas detalladas con el propietario de la tienda para comprender sus necesidades específicas, tales como las funcionalidades necesarias para la gestión de inventario y la visualización de pedidos.
- **Documentación de casos de uso:** Se documentan casos de uso que describen cómo interactuarán los usuarios con la plataforma. Por ejemplo, cómo un cliente agregará productos al carrito, cómo un administrador gestionará el inventario, o cómo se visualizarán los pedidos.

2.2. Análisis de Requerimientos

Se analizarán los requisitos técnicos y funcionales para asegurar que la implementación se realice conforme a las especificaciones del proyecto.

Para asegurar que la implementación del software cumpla con las especificaciones y necesidades del cliente, se realizó un análisis de los requerimientos. A continuación, se detalla la información obtenida de la Propuesta Técnica de Software para Tienda Virtual y las Historias de Usuario.

2.2.1 Identificación de los Stakeholders

En esta fase, se identifican los principales stakeholders del proyecto. Para "Construmole," los stakeholders incluyen:

- **Propietario de la Tienda:** El cliente principal que solicitó la creación de la plataforma. Su rol es fundamental en la toma de decisiones, especialmente en la aprobación de requisitos y la validación de las funcionalidades implementadas.
- **Usuarios Administrativos:** Empleados o administradores que gestionarán el inventario, las órdenes y la información de los clientes a través de la interfaz de administración.
- **Clientes:** Usuarios finales que utilizarán la tienda virtual para realizar compras. Es importante entender sus necesidades para diseñar una experiencia de usuario efectiva

2.2.2. Requerimientos Funcionales

Documentación

Todos los requisitos, tanto funcionales como no funcionales, se tienen que documentar. Esto incluye:

- **Requisitos Funcionales:** Como el registro de usuarios, la capacidad de iniciar sesión, la gestión del carrito de compras, y la eliminación de cuentas de usuario.
- **Requisitos No Funcionales:** Incluyendo aspectos como la seguridad, la escalabilidad del sistema, y la facilidad de uso.

Requisitos funcionales identificados:

- **Registrar usuario**
Los usuarios pueden registrarse en la plataforma completando un formulario con los campos de nombre de usuario y contraseña.
- **Permitir iniciar sesión a usuario registrado**
Los usuarios registrados pueden iniciar sesión en la plataforma utilizando su nombre de usuario y contraseña.
- **Carrito de compras**

Los usuarios pueden añadir productos a su carrito de compras desde la página del producto seleccionado.

- **Eliminar cuenta de usuario**
Los usuarios pueden eliminar su cuenta desde la página de configuración de cuenta.
- **Añadir un nuevo producto**
El administrador puede añadir nuevos productos a la tienda desde la consola de Django admin, proporcionando información como categoría, nombre, descripción, precio, imagen, stock, etc.
- **Eliminar un producto**
El administrador puede eliminar productos, tanto individualmente como en grupo, desde la consola de Django admin.
- **Mostrar productos a la venta**
La plataforma muestra los productos disponibles a los usuarios en una lista filtrada por categorías y con la opción de ver detalles.
- **Ver pedidos del usuario**
Los usuarios pueden ver un historial de sus pedidos desde su cuenta, incluyendo detalles del pedido y su estado actual.
- **Pago de productos**
Los usuarios pueden seleccionar una opción de pago (pago al momento de la entrega o al recoger el producto en la tienda) al finalizar su compra.
- **Creación y eliminación de cuentas de usuario por el administrador**
El administrador puede crear o eliminar cuentas de usuario directamente desde la consola de Django admin.

2.2.2. Implementación del Software

Basado en la sección de Implementación del Software, se identificaron los siguientes requerimientos técnicos:

- **Base de Datos:**
Utilizar una base de datos relacional como MySQL para almacenar la información de usuarios, productos, órdenes y transacciones.
- **Backend:**
Implementar la lógica del negocio utilizando Django, asegurando la modularidad y reutilización del código.
 - Crear APIs RESTful.

- **Frontend:**
Utilizar tecnologías como HTML, CSS y JavaScript para desarrollar una interfaz de usuario responsiva y amigable.
- **Despliegue:**
Configurar servidores de producción.

3. Documentación de Diseño de Software para la Tienda Virtual

3.1. Arquitectura del Sistema

La aplicación tiene una arquitectura monolítica en la que todos los componentes del sistema se implementan como una única aplicación. Esto incluye tanto el frontend como el backend, así como la gestión de datos y la lógica de negocio.

El sistema está compuesto por dos aplicaciones principales:

- **Store:** Gestiona productos, órdenes, y el carrito de compras aplicación
- **accounts:** Se encarga del manejo de usuarios y autenticación.

Django sigue el patrón MVC (Model-View-Controller), en el que los Modelos manejan la lógica de datos, las Vistas gestionan la presentación y los Controladores dirigen las interacciones de usuario.

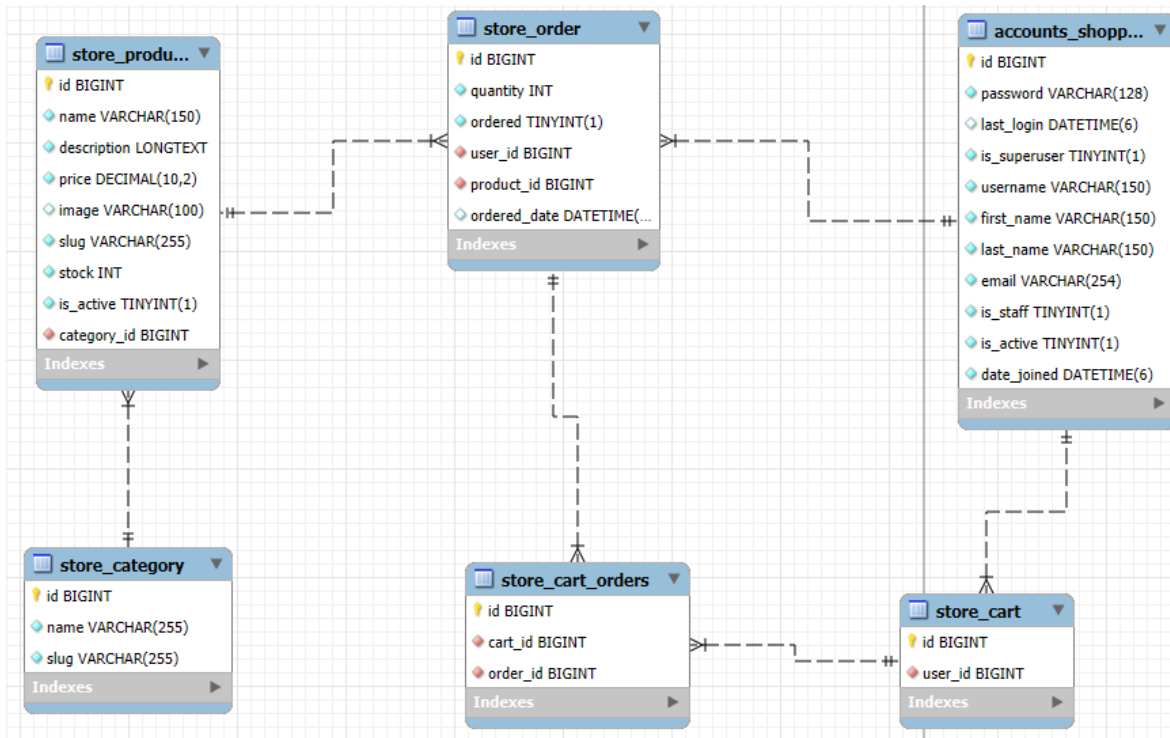
3.2. Componentes Principales

El sistema está basado en la arquitectura de Django, dividido en varias aplicaciones:

- **Aplicación root:** Gestiona la configuración general del proyecto.
- **Aplicación store:** Maneja la lógica de productos, categorías, carritos de compra y órdenes.
- **Aplicación accounts:** Gestiona la autenticación de usuarios y el perfil de usuarios

3.3 Diseño de la base de datos

Modelo Relacional: Se implementa un modelo relacional para la base de datos, donde las entidades principales incluyen Productos, Categorías, Órdenes, Usuarios, y Carrito de Compras. Las relaciones entre estas tablas se establecen a través de claves primarias y foráneas, lo que asegura la consistencia referencial.



3.4 Interfaz de Usuario (UI) y Experiencia de Usuario (UX)

El diseño de la interfaz de usuario (UI) y la experiencia de usuario (UX) son componentes críticos que se abordan en esta etapa.

- **Prototipos y Mockups:** Se crean prototipos y mockups para visualizar cómo se verá la tienda virtual. Estos prototipos se utilizan para hacer ajustes tempranos y recibir retroalimentación del cliente antes de proceder con el desarrollo completo.
- **Navegación Intuitiva:** El diseño de la UI se enfoca en la facilidad de navegación. Los clientes deben poder buscar productos, agregar artículos al carrito, y completar compras con el mínimo esfuerzo.

4. Desarrollo

Esta etapa implica la codificación de la tienda virtual “Construmole” según los diseños y la arquitectura establecidos.

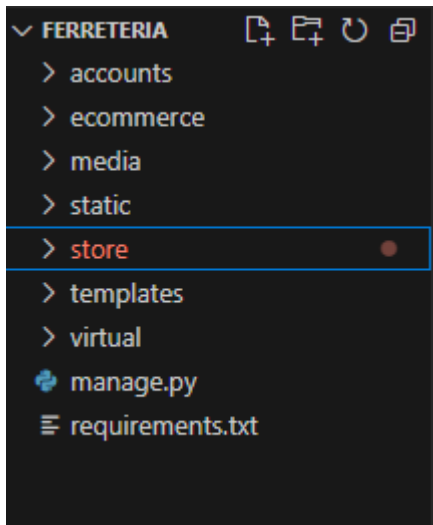
4.1 Desarrollo Frontend y Backend

- **Frontend:** El desarrollo frontend se centra en crear una interfaz atractiva y funcional. Se utilizan tecnologías como HTML, CSS, y JavaScript para construir la interfaz de usuario, asegurando que sea responsiva y accesible desde dispositivos móviles.
- **Backend:** El desarrollo backend incluye la implementación de la lógica de negocio y la gestión de la base de datos. Django, el framework utilizado, permite una

rápida implementación de funcionalidades clave como la autenticación de usuarios, la gestión de sesiones, y la manipulación de datos.

4.2 Estructura del proyecto

La estructura del proyecto incluye varias aplicaciones de Django. Cada una tiene su propio conjunto de archivos y directorios necesarios para su funcionamiento.



4.2.1 Modificaciones en el Código

Las modificaciones del código deben realizarse en las aplicaciones correspondientes. Por ejemplo, para modificar la lógica de las cuentas, se debe trabajar en la aplicación `accounts`.

Cada aplicación contiene:

- **settings.py**: Configuración global del proyecto.
- **urls.py**: Configuración de rutas.
- **models.py**: Definición de las entidades de datos.
- **views.py**: Lógica de manejo de vistas y controladores.
- **templates/**: Archivos HTML que definen la interfaz de usuario.

4.3 Testing y Revisión Continua

Durante todo el proceso de desarrollo, se llevan a cabo pruebas unitarias y de integración para asegurar que cada componente funcione correctamente y que no haya errores al combinar diferentes módulos.

5. Pruebas del Sistema

Las pruebas son una fase crítica del desarrollo de software, ya que aseguran que todos los componentes de la tienda virtual "Construmole" funcionen según lo previsto. La fase de pruebas se divide en varias sub-etapas que abordan diferentes aspectos del sistema.

- **Pruebas Unitarias:** Estas pruebas se realizan durante la fase de desarrollo y se enfocan en verificar que cada unidad de código (como funciones, métodos, y clases) funcione correctamente. En el contexto del proyecto, las pruebas unitarias podrían incluir la verificación de la funcionalidad de los modelos Django, como la creación de un producto, la gestión del carrito de compras, y la autenticación de usuarios.
- **Pruebas de Interfaz de Usuario (UI):** En esta etapa, se realizan pruebas para asegurar que la interfaz de usuario sea intuitiva, fácil de navegar, y funcione correctamente en diferentes dispositivos y navegadores. Esto incluye la verificación de que los formularios de registro, inicio de sesión, y el proceso de compra se comporten como se espera.

6. Capacitación y Soporte

Manual Técnico

La creación de un manual técnico es esencial para asegurar que los desarrolladores y administradores puedan gestionar, mantener, y actualizar el sistema de manera eficiente.

- **Contenido del Manual Técnico:** El manual técnico incluye detalles sobre la arquitectura del sistema, la estructura de la base de datos, y los componentes clave de la aplicación. También cubre instrucciones sobre cómo desplegar y actualizar la aplicación, cómo realizar copias de seguridad de la base de datos, y cómo solucionar problemas comunes.

Manual de Usuario

El manual de usuario se crea para guiar a los usuarios finales, tanto administradores como clientes, en el uso de la tienda virtual.

- **Contenido del Manual de Usuario:** El manual incluye instrucciones paso a paso sobre cómo registrarse, iniciar sesión, navegar por la tienda, agregar productos al carrito, realizar compras, y gestionar cuentas. También cubre cómo los administradores pueden añadir, modificar, y eliminar productos, así como gestionar órdenes y usuarios.

7. Lanzamiento

Lanzamiento

El lanzamiento de la tienda virtual "Construmole" marca la transición del proyecto desde el entorno de desarrollo y pruebas hacia un entorno de producción en vivo, donde los usuarios finales podrán interactuar con la plataforma. Esta fase es crucial, ya que cualquier problema no detectado anteriormente podría impactar la experiencia del usuario y, en última instancia, la reputación de la tienda.

Actividades Clave del Lanzamiento:

1. Preparación del Entorno de Producción:

- **Configuración de la Infraestructura:** Asegurar que todos los servidores, bases de datos y servicios estén correctamente configurados en el entorno de producción, ya sea en AWS o PythonAnywhere. Esto incluye la optimización de los recursos para manejar la carga esperada de usuarios.
- **Migración de Datos:** Si es necesario, migrar datos desde un entorno de desarrollo o prueba a producción. Esto podría incluir información sobre productos, categorías, usuarios registrados, y cualquier dato relevante para el funcionamiento del sistema.

2. Despliegue del Código:

- **Verificación de Versiones:** Asegurarse de que la versión del código que se va a desplegar sea la más reciente y estable. Esto incluye revisar que todas las dependencias y módulos estén correctamente actualizados.

3. Pruebas Finales en Producción:

- **Pruebas de Funcionalidad:** Realizar un último conjunto de pruebas para verificar que todas las funcionalidades críticas, como el proceso de compra, la gestión de inventario, y la administración de cuentas, funcionen correctamente en el entorno de producción.

4. Activación de la Plataforma:

- **Habilitación de la URL Pública:** Una vez que se haya verificado que todo está funcionando correctamente, la tienda virtual será activada para que esté accesible al público a través de su URL oficial.