

Diseño de la Arquitectura y de la Base de Datos

Contenido

- Planteamiento de necesidades
- Especificación de requerimientos
- Diagrama de paquetes de la Arquitectura de la Iteración
- Definición del ambiente de implementación

Planteamiento de necesidades

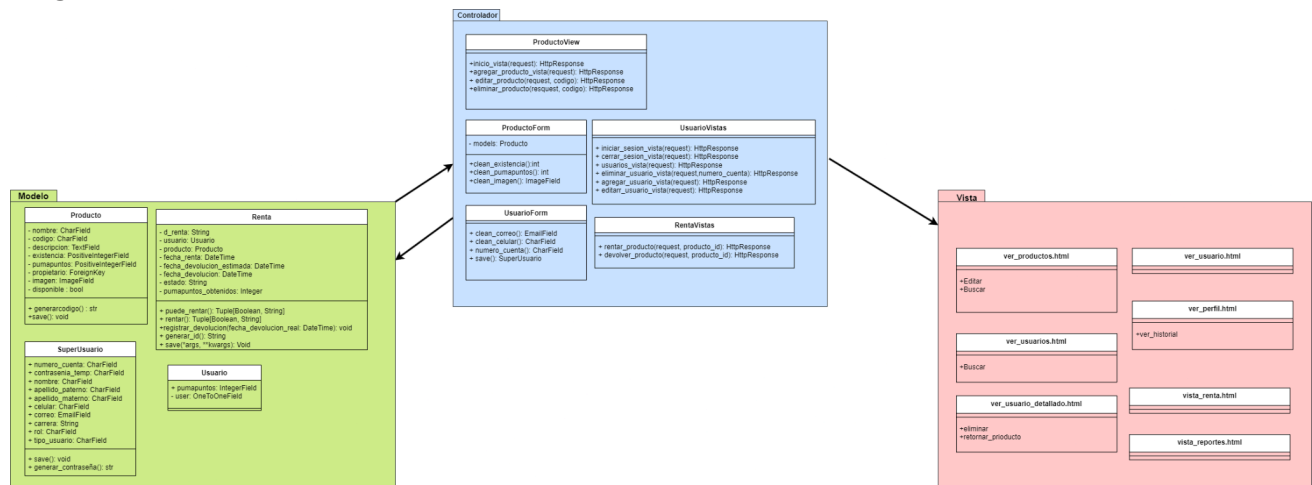
[Liga al Planteamiento de necesidades del proyecto](#)

Especificación de requerimientos

[Liga al documento de la Especificación de requerimientos de todos los casos de uso de la iteración](#)

Diagrama de paquetes de la Arquitectura de la Iteración

Diagrama de paquetes General



Definición del ambiente de implementación

Especificación de los siguientes elementos del ambiente de implementación con su nombre y versión:

- **Lenguaje (o lenguajes) de Programación:** Python 3.10.12, Javascript y SQL (PostgreSQL 14.13)
- **IDE (Editor de desarrollo):** Visual Studio Code 1.93.1
- **Gestor de dependencias:** Pip 22.0.2
- **Framework de frontend:** Javascript
- **Framework de backend:** Django 5.1.1
- **Herramienta para Control de versiones:** Github
- **Repositorio de bibliotecas:** Pip 22.0.2
- **Herramienta para Control de calidad del código:** Github
- **Diagramador UML:** draw.io
- **Herramienta para Prototipado de interfaz de usuario:** Figma
- **Manejador de bases de datos:** DBeaver community
- **Visualizador de la BD**
- **Sistema operativo :** Windows 10 en adelante

Uso del Proyecto con Docker

El proyecto utiliza **Docker** y **Docker Compose** para su ejecución.

1. **Clonar el repositorio:** Descarga el código desde GitHub y accede al directorio.
2. **Levantar los contenedores:** Ejecuta `docker-compose up --build`.
3. **Migrar la base de datos:** Usa `docker-compose exec web python manage.py migrate`.
4. **Crear superusuario (opcional):** Ejecuta `docker-compose exec web python manage.py createsuperuser`.
5. **Acceso a la aplicación:** Abre <http://localhost:8000> en tu navegador.

Con esto, el sistema estará listo para usar.