

Despensa:

El modelo de negocio:

El proyecto es una despensa online, en la cual se tiene productos en stock y estos productos son vendidos. Los clientes realizan compras y pueden adquirir productos con un límite de 3 unidades por compra por día. Además de la funcionalidad de tienda, la despensa puede emitir determinados reportes estadísticos como las ventas por día, los montos totales de compra de cada cliente y determinar el producto más vendido.

Diseño de la aplicación:

Para el diseño del backend de la aplicación se utilizó Spring Boot, se desarrolló mediante el uso de JPA y el patrón repository. Se eligió este patrón porque permite aislar los objetos del dominio de los detalles del código de acceso a la base de datos, minimizando la dispersión y el duplicado del código de consulta.

Para diseñar la aplicación se realizó el siguiente Diagrama:

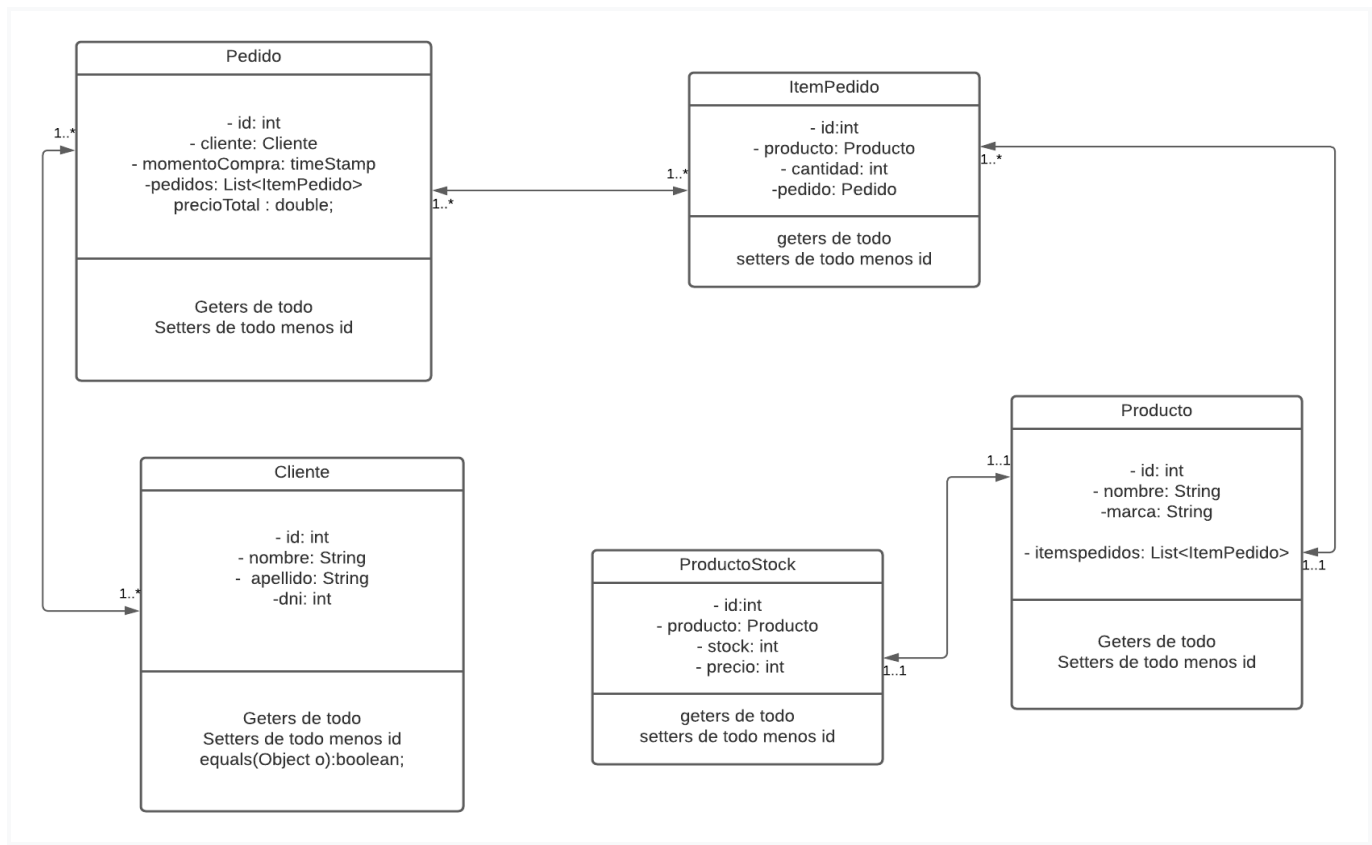


Imagen 1: Diagrama de clases

Así nuestra despensa online tendrá cinco entidades:

Producto: Determina las características de los productos de la empresa, teniendo un identificador, un nombre y una marca.

ProductoStock: define el stock de la tienda conteniendo un identificador, un producto y la cantidad de unidades disponibles, además del precio de venta de los mismos.

Cliente: Almacena las características que permiten identificar y diferenciar a cada cliente, un identificador, su nombre, apellido y su DNI

ItemPedido: encapsula cada producto del pedido junto con la cantidad a ordenar y un identificador único.

Pedido: Es la entidad que va a definir las ventas, cuenta con un identificador, un cliente y una lista de items.

De esta forma para realizar una compra en la tienda se creará un pedido que intentará agregarse a nuestra aplicación, si el pedido cumple con las condiciones de la despensa el pedido será agregado a la base de datos y se reducirá del stock de cada producto las cantidades adquiridas en el pedido en cuestión.

Los informes son transportados por DTOs (Data Transfer Object), y la base de datos se llena automáticamente con un filler al iniciar el proyecto, este filler genera datos aleatorios para 20 clientes.

Para el consumo de la aplicación se desarrolló una aplicación web que consume los servicios de nuestro backend. El cliente se comunica mediante peticiones HTTP(GET, POST, PUT and DELETE)

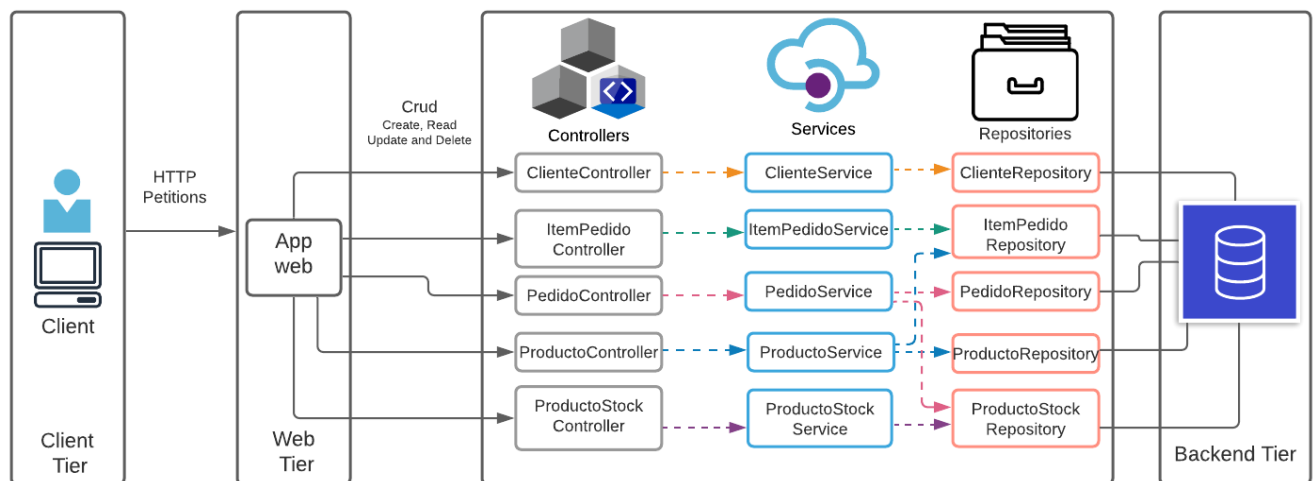


Imagen 2: Vista de componentes y conectores

Nivel del Cliente: Un usuario que se conecta con un equipo mediante internet.

Nivel Web: Incluye la aplicación Web desarrollada para conectarse con la Despensa

Aplicación Despensa Online: Aquí se encuentra la lógica de negocio, que verifica los datos recibidos y evalúa las reglas correspondientes para que las operaciones se acepten y sean comunicadas a la base de datos o se rechacen.

Nivel de Backend: Incluye la DB que nos permitirá persistir los datos