## 2. Documento de Diseño UML

### 2.1 Arquitectura

### 2.2 Diseño

## Historia de usuario Mostrar Tienda

**En el primer sprint,** antes de tener la interfaz gráfica creada, simplemente creamos un método para mostrar los productos de nuestra tienda por pantalla, en formato textual. Dividimos los productos en páginas, que en el futuro serán páginas reales de la interfaz, y mostramos ese número de productos. Permitimos al usuario desplazar páginas a izquierda y derecha.

Diagrama de secuencia:

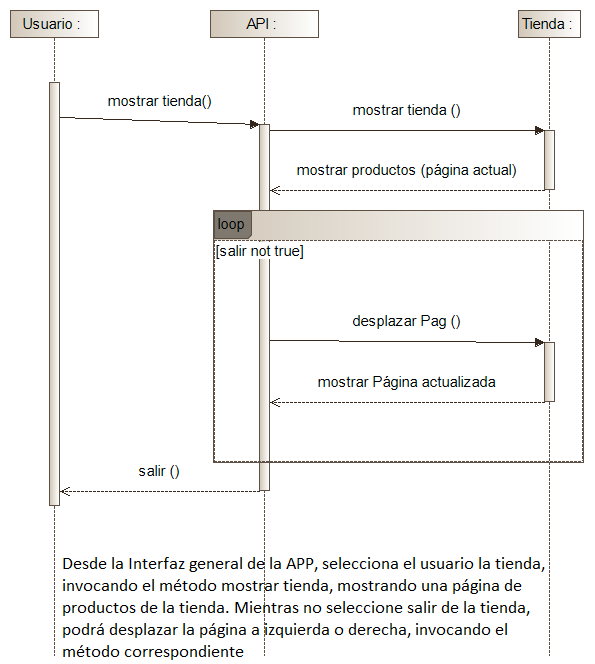
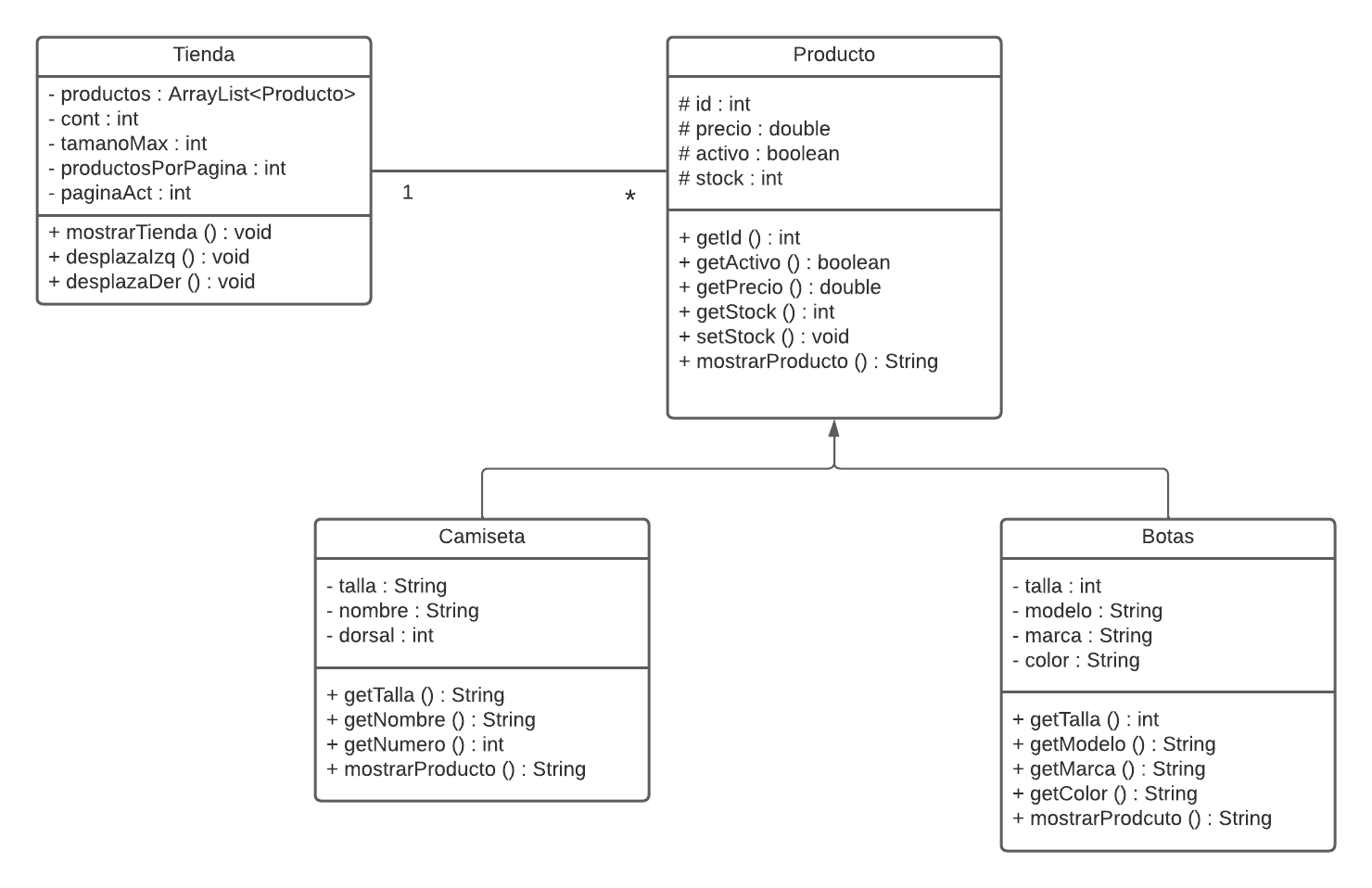


Diagrama de clases:



En esta historia de usuario usamos 4 clases. Tenemos una tienda, que puede tener 0 o varios productos. Estos pueden ser camisetas o botas.

## 

## 

## 

## 

## Historia de usuario Mostrar Inventario

**En el primer sprint,** creamos un método para mostrar los productos del inventario por pantalla, en formato textual. Se muestran en lista todos los productos con la cantidad disponible de cada uno y su información. Esta lista en un futuro, cuando tengamos la interfaz gráfica estará más compacta y categorizada.

Diagrama de secuencias:

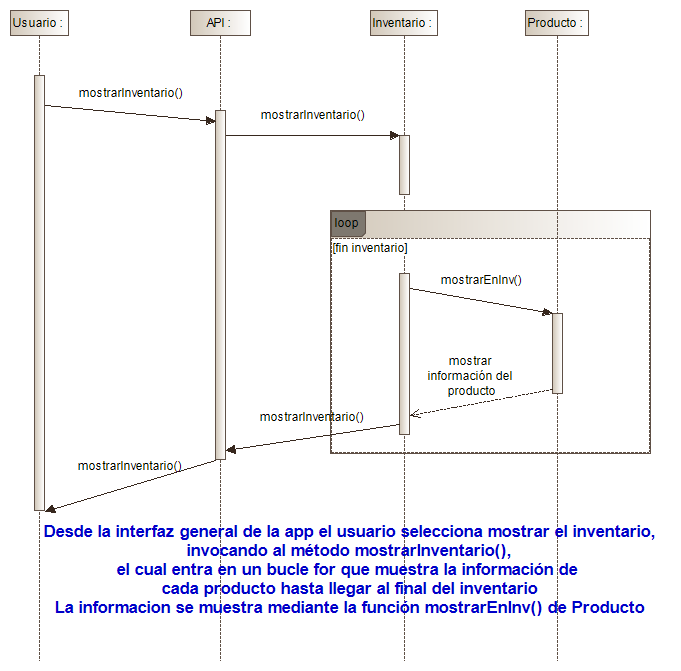
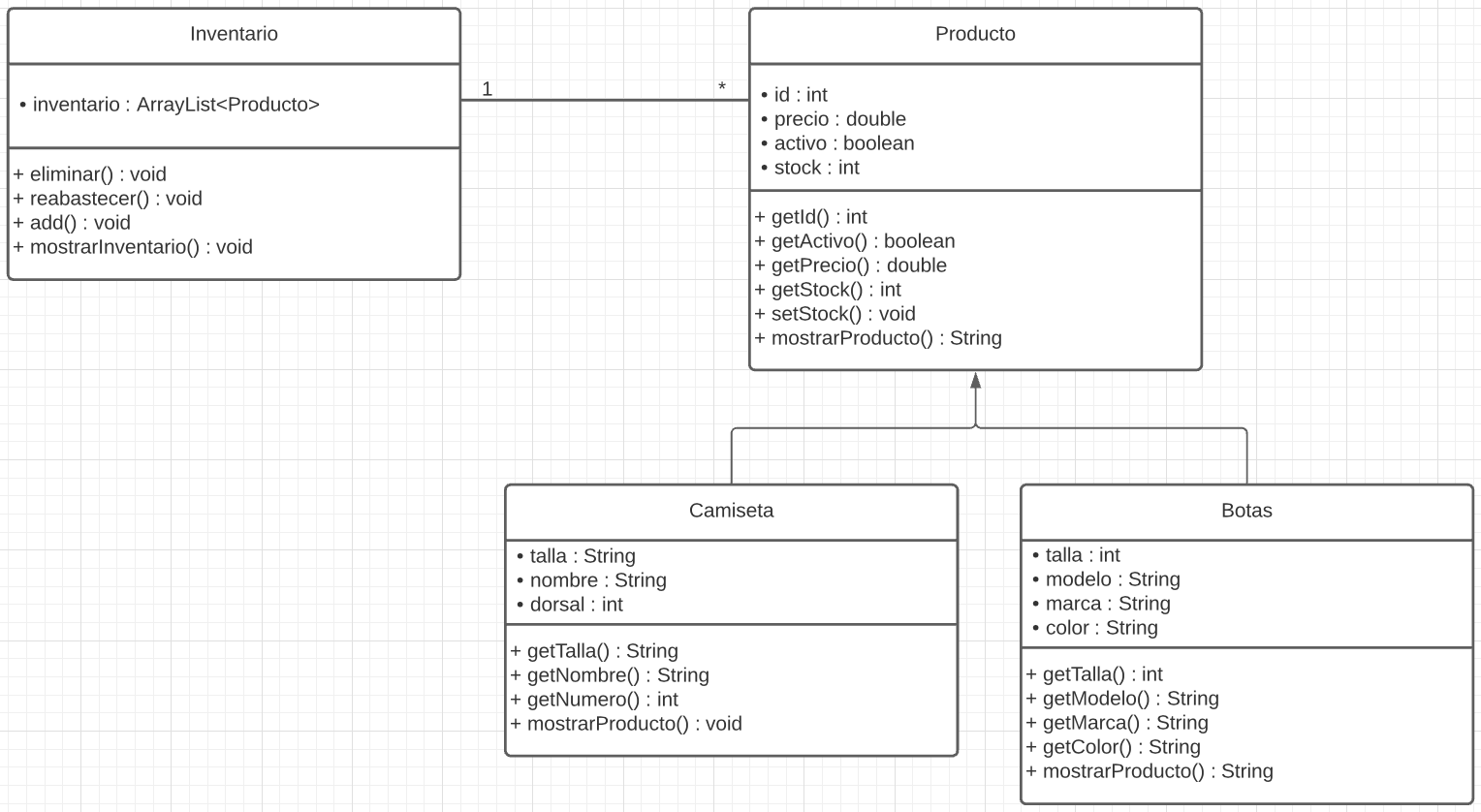


Diagrama de clases:



## 

## Historia de usuario Eliminar Producto

**En el primer sprint** hemos creado la clase Inventario donde metemos todos los productos existentes, tanto si se muestran en la tienda o no.

Añadimos diversos métodos, entre los cuales se encuentra eliminar(), donde pasándole la lista de productos de la tienda y el elemento a eliminar, eliminamos el producto tanto del inventario como de la tienda.

Diagrama de secuencia:

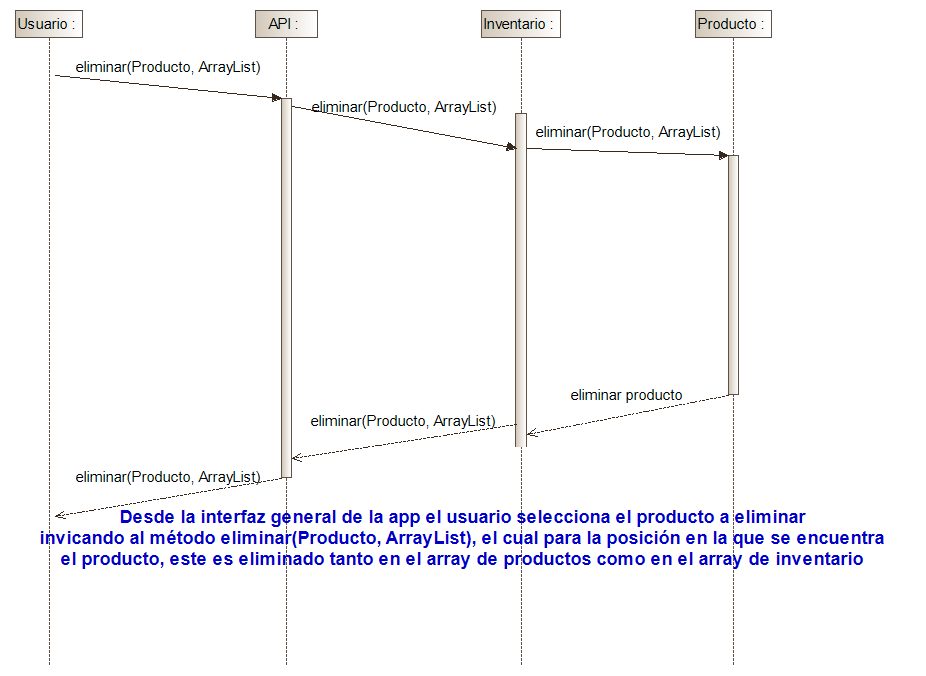
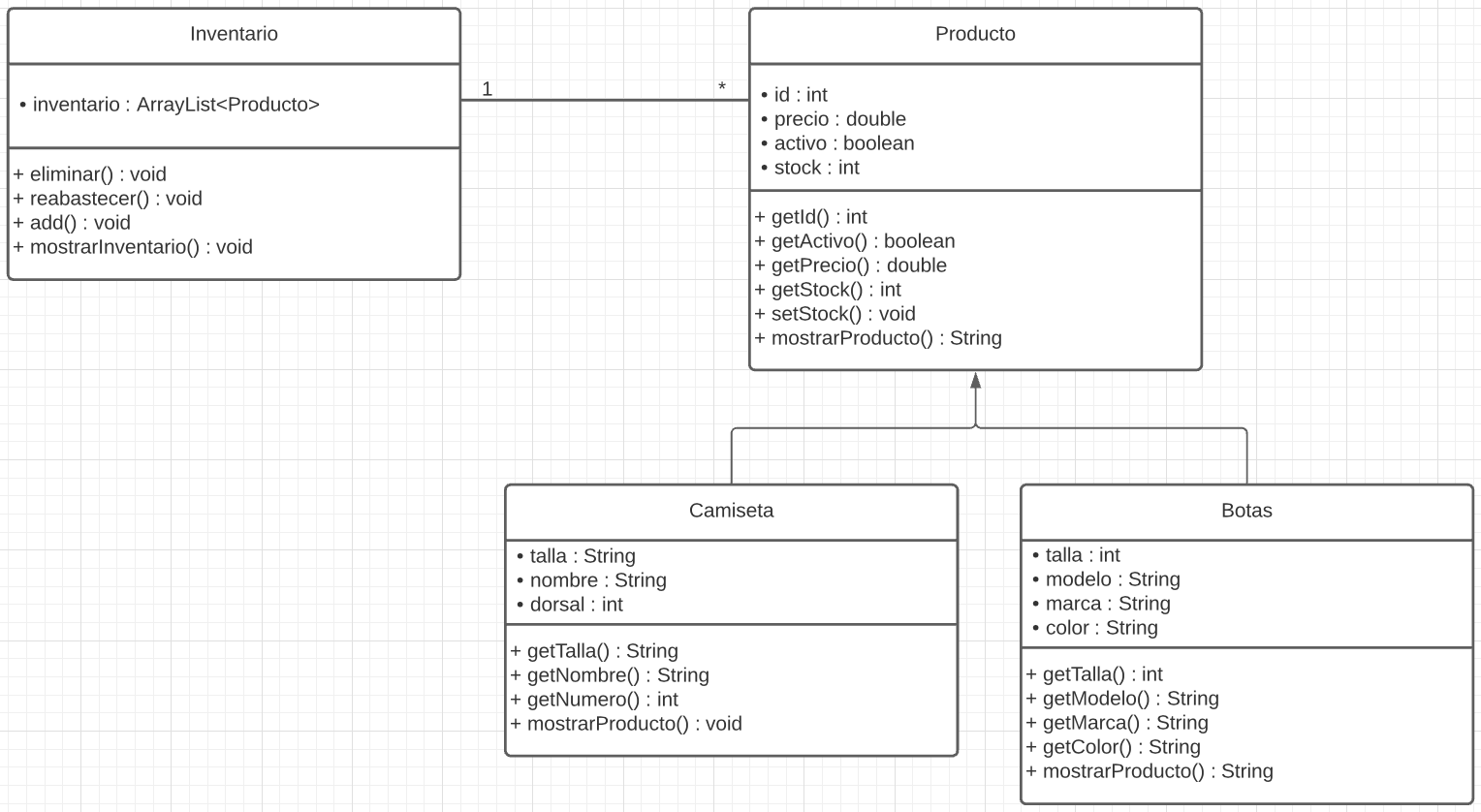


Diagrama de clases:



## 

## Historia de usuario Reabastecer Producto

En el primer sprint, igual que pasaba con el método eliminar() de la clase Inventario, hemos creado un método llamado reabastecer(), el cual se encarga de añadir una cantidad “n” a un elemento concreto del inventario. La modificación de dicho elemento queda reflejado tanto en el inventario como en la lista de productos de la tienda.

Diagrama de secuencia:

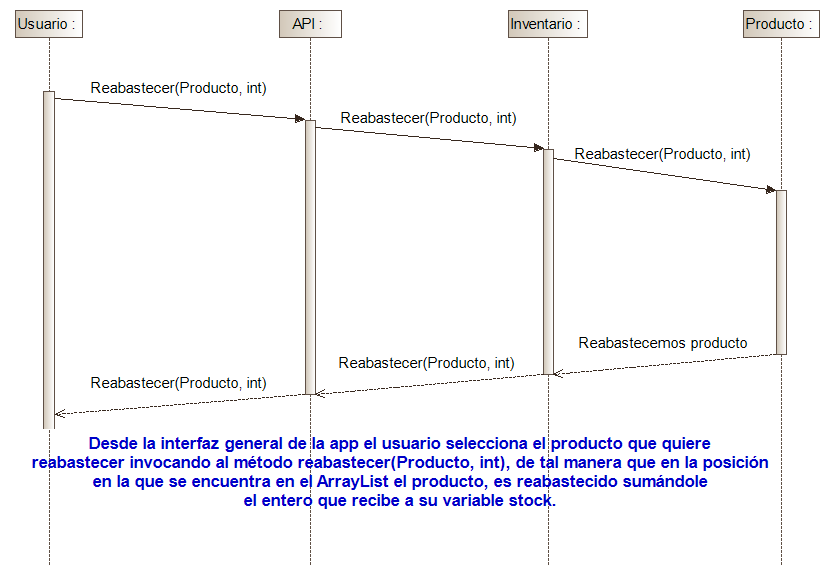


Diagrama de clases:

