

Para representar los productos se usan las clases Product, TimedProduct (Para los productos con fecha de caducidad) y BaseProduct (Para los productos básicos). En la clase abstracta Product se reúne la funcionalidad común a todos los productos (ver el id, modificar el precio, etc...).

Para gestionarlos usamos una clase Inventory, que almacena todos los productos y se encarga de las operaciones relativas a estos (crear, borrar, etc...), con el objetivo de reunir toda la funcionalidad que afecta a los productos en un mismo lugar.

Para los nuevos usuarios, usamos una estructura similar. La clase abstracta User reúne la funcionalidad común mientras que las clases Client y Cashier se encargan de lo específico de cada tipo de usuario.

Para gestionarlos usamos la clase UserManager que, de manera parecida a Inventory, se encarga de las operaciones relativas a los usuarios, añadiendo algunas funcionalidades como listar los tickets ya que estos se almacenan en los usuarios.

Para representar los tickets, usamos una clase Ticket que almacena objetos de la clase ProductInfo, que almacenan un producto e información de este exclusiva del ticket (personalizaciones y cantidad), que no tiene sentido almacenar en Product ya que varía en cada ocurrencia del producto en un ticket.

Para los comandos usamos una interfaz Command que implementan las clases de los comandos, de las que habrá una por cada tipo de comando (prod, ticket, etc...). Esto es para hacerlo más modular y, como consecuencia, más escalable. Además, simplifica el reparto de tareas.

La clase App se encarga de lo básico, leer el input del usuario, dividirlo en tokens y llamar al comando correspondiente para que se encargue de todo lo demás, además de algunas funciones de utilidad varia y los comandos simples (help y echo).