

# **Rapport du projet de génie logiciel**

**Génie Logiciel : Projet 2023-2024**

**Développement guidé par UML d'une application  
(E-Shop )**

## **Membres du groupe:**

**Sabrina FARADJI**

**Dilia TIGHRINE**

**Lynda BEN BEKKOU**

**Saarah TABASSUM KHODADIN**

**Abdelalime MAZEGHRANE**

## **Introduction**

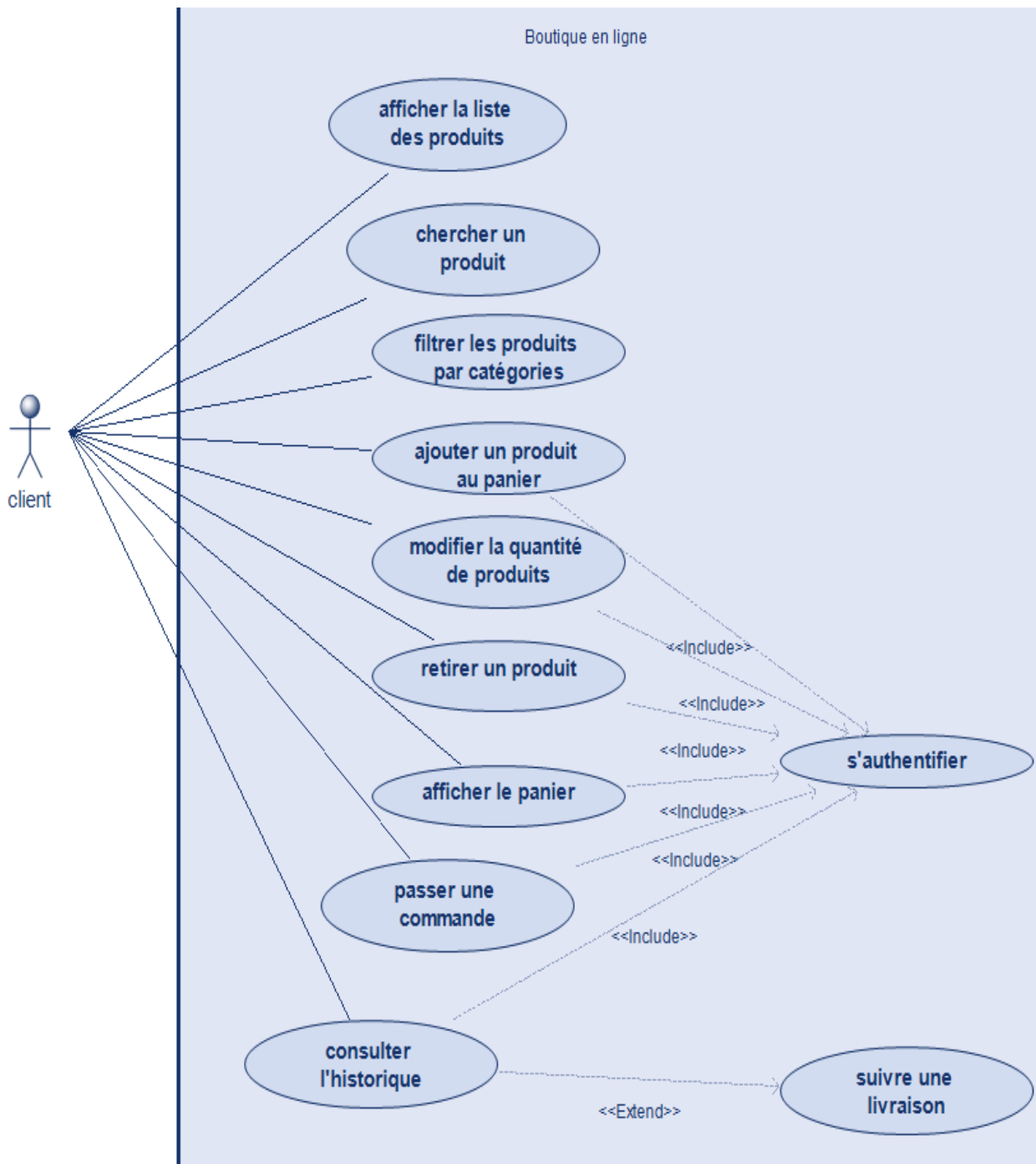
Le projet vise à modéliser et développer une application simplifiée d'un système de type E-Shop (boutique en ligne). L'objectif principal est de mettre en pratique les concepts de modélisation UML et d'acquérir de l'expérience dans le développement d'une application électronique basique en utilisant le framework Spring et JPA.

Dans ce rapport, nous présenterons en détail les modèles UML élaborés, allant des diagrammes de cas d'utilisation et de classes d'analyse aux séquences de conception. Nous explorerons également les fonctionnalités spécifiques mises en œuvre, telles que l'ajout de produits au panier, la modification de la quantité, le retrait d'articles, l'affichage du catalogue, et le processus de commande. De plus, nous expliciterons les décisions architecturales prises tout au long du projet, détaillant l'utilisation judicieuse de JPA et Spring, la structuration des classes, et le recours à des interfaces pour favoriser une conception modulaire.

Ce document offre ainsi une plongée approfondie dans le processus de conception et de développement de notre application E-Shop

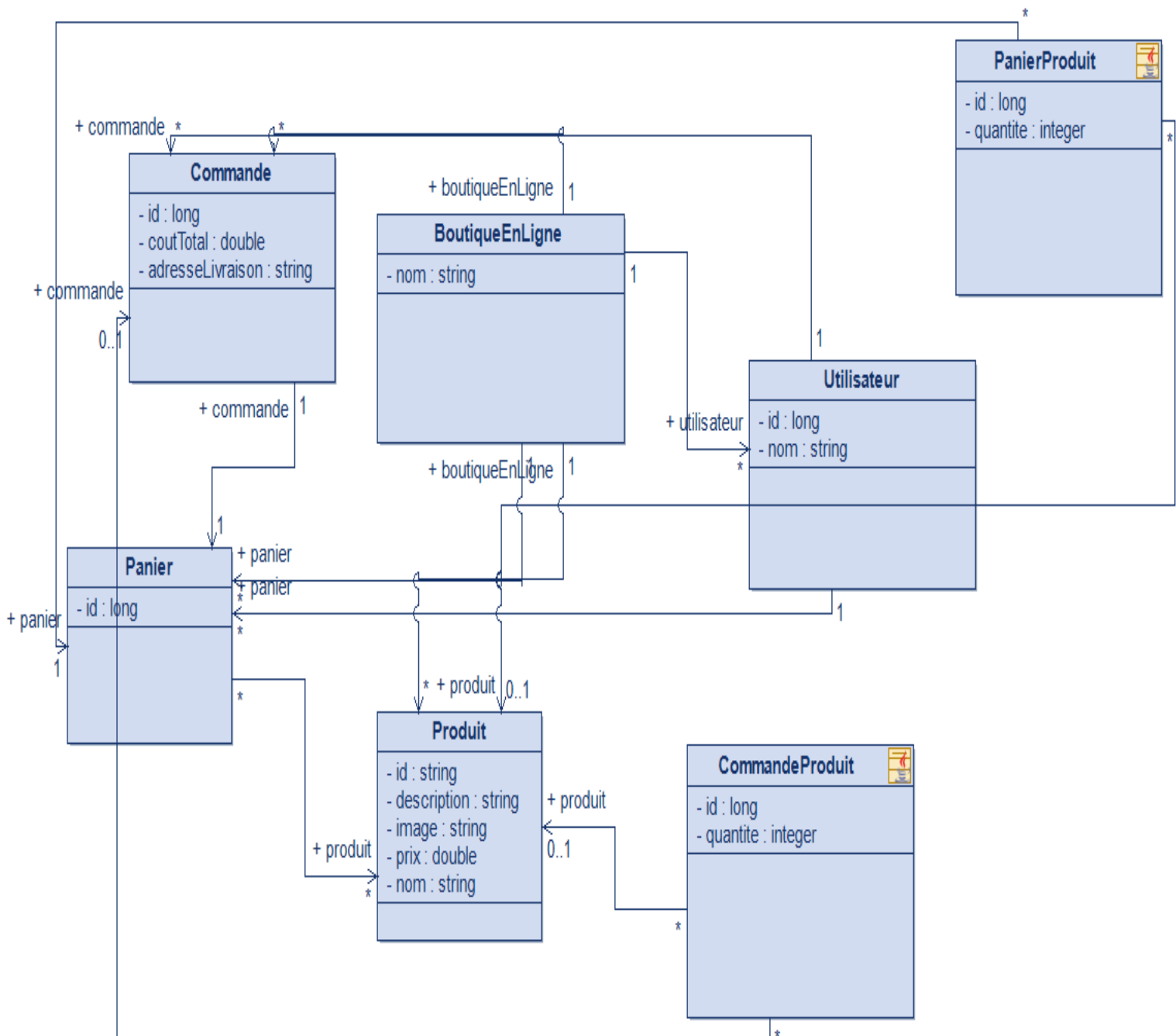
## Diagramme de use case

Représentation des interactions entre les utilisateurs et le système.



## Diagramme de Classes d'Analyse

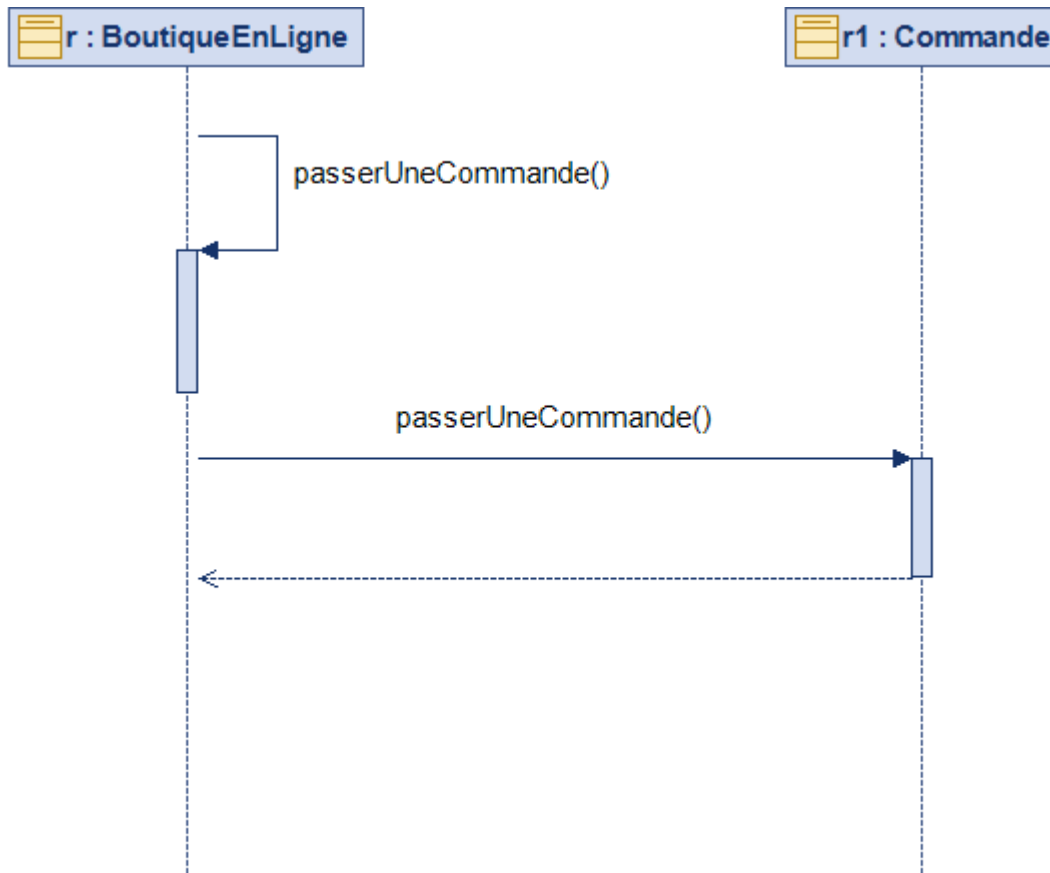
Modélisation des classes et de leurs relations.



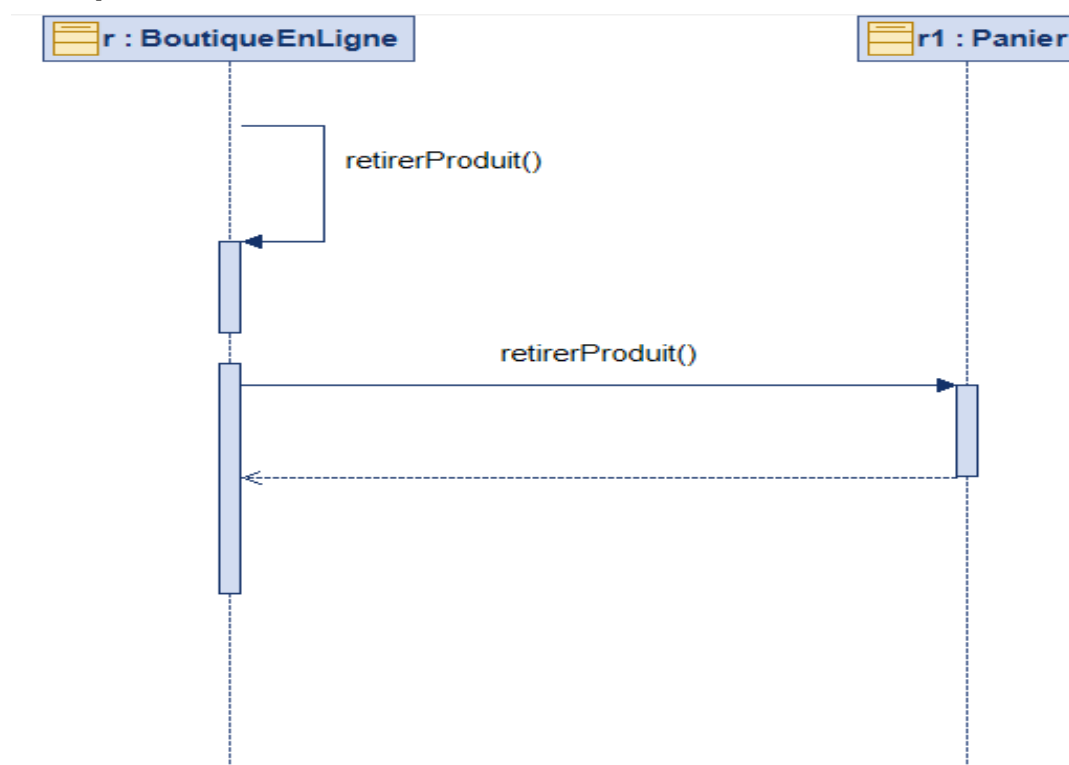
## Diagrammes de Séquence d'Analyse

Illustration des interactions entre les objets du système au fil du temps.

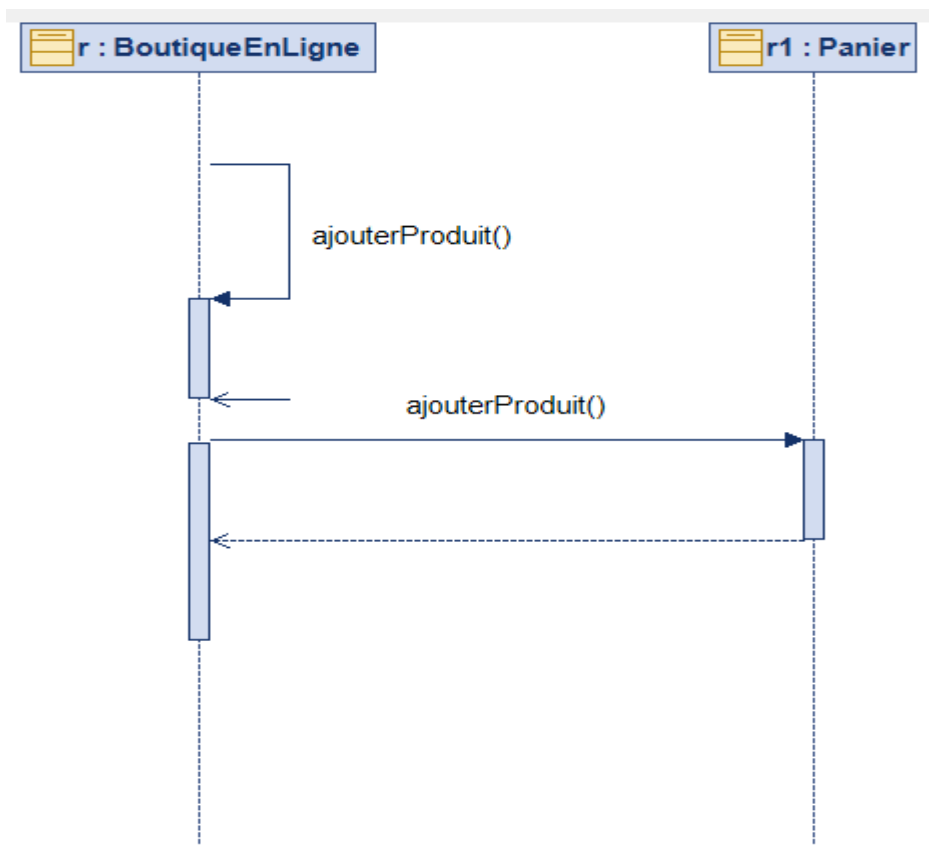
### Passer une commande



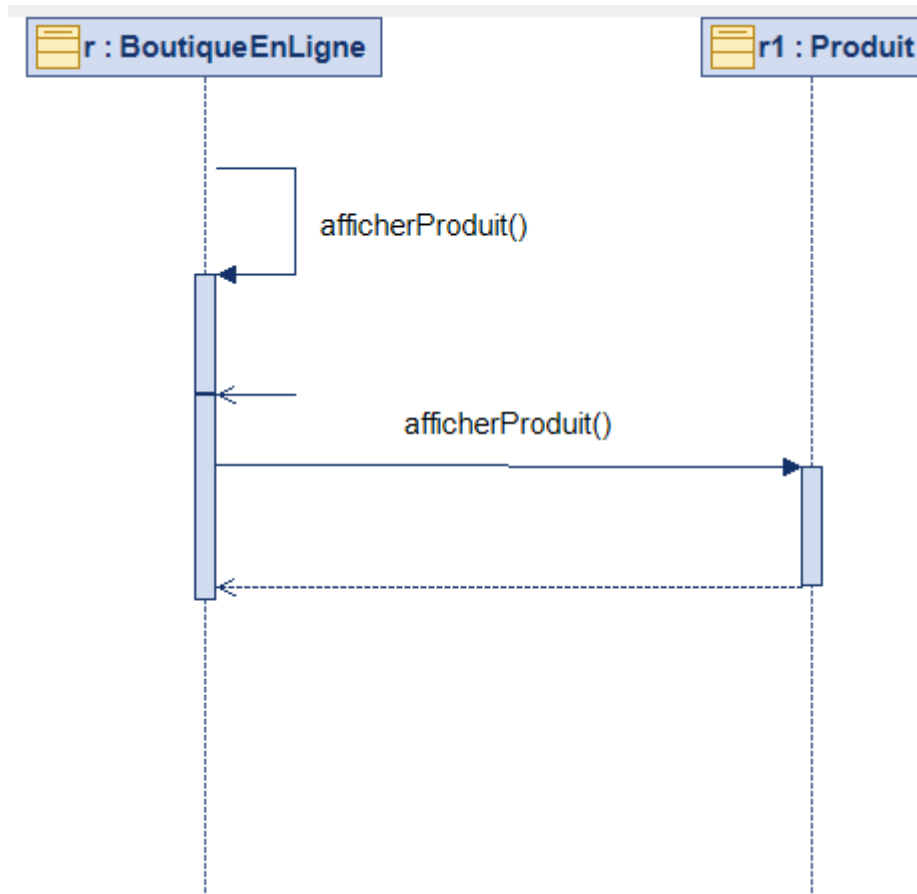
### Retirer un produit



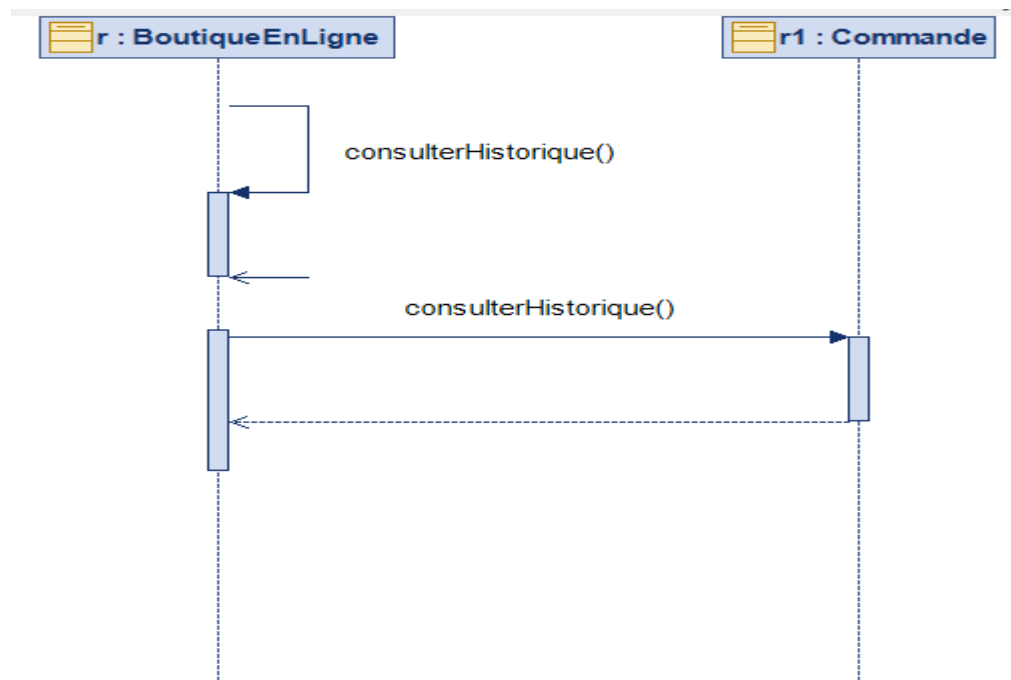
## Ajouter un produit



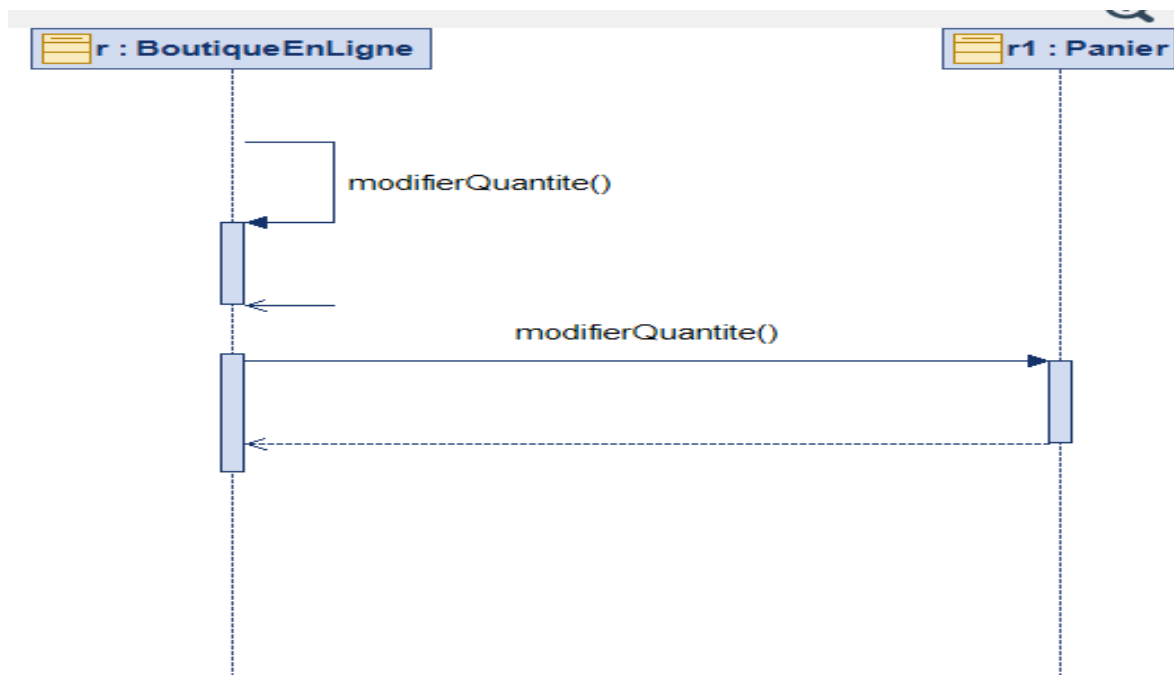
## Afficher un produit



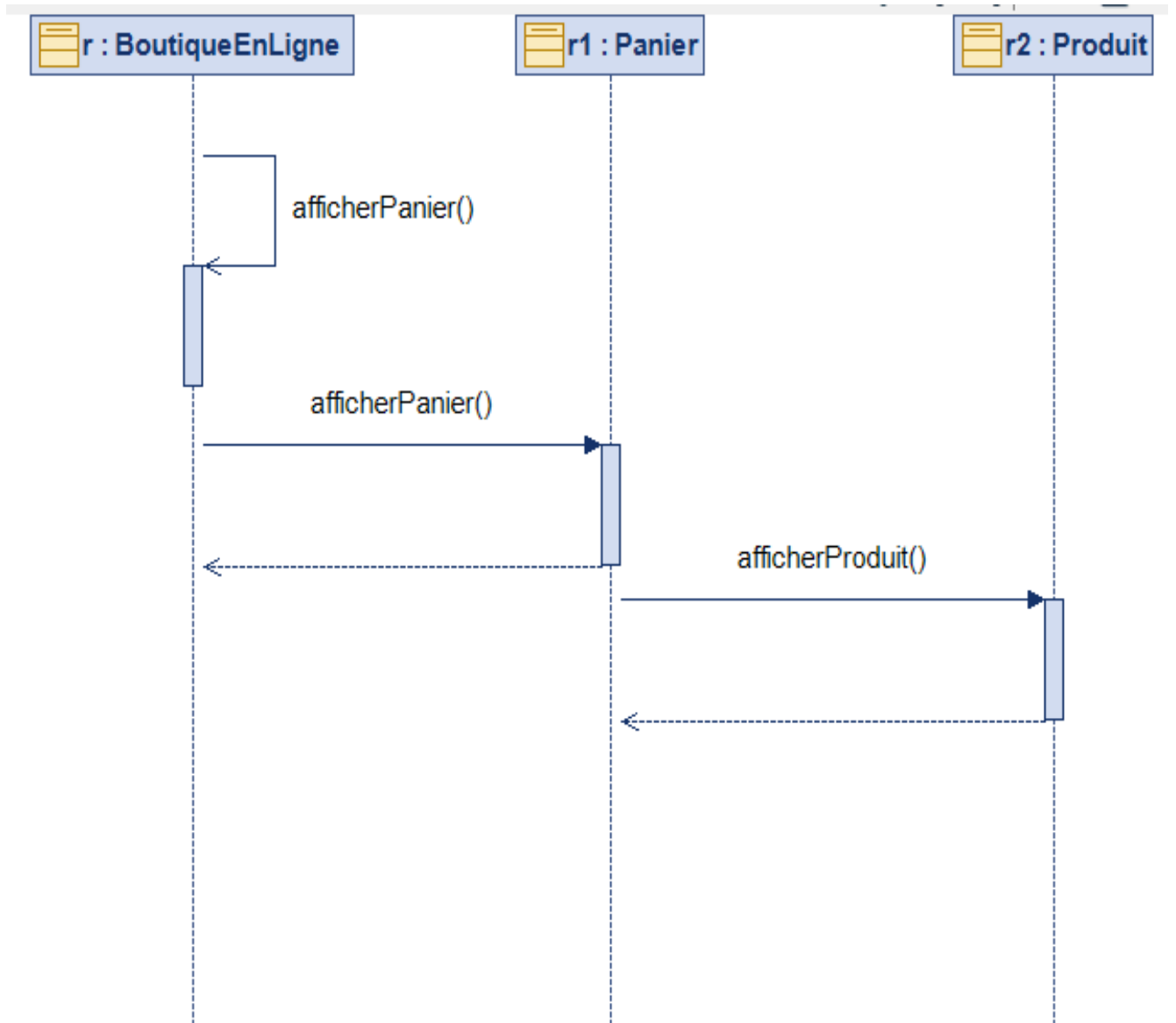
## Consulter historique



## Modifier la quantité d'un produit



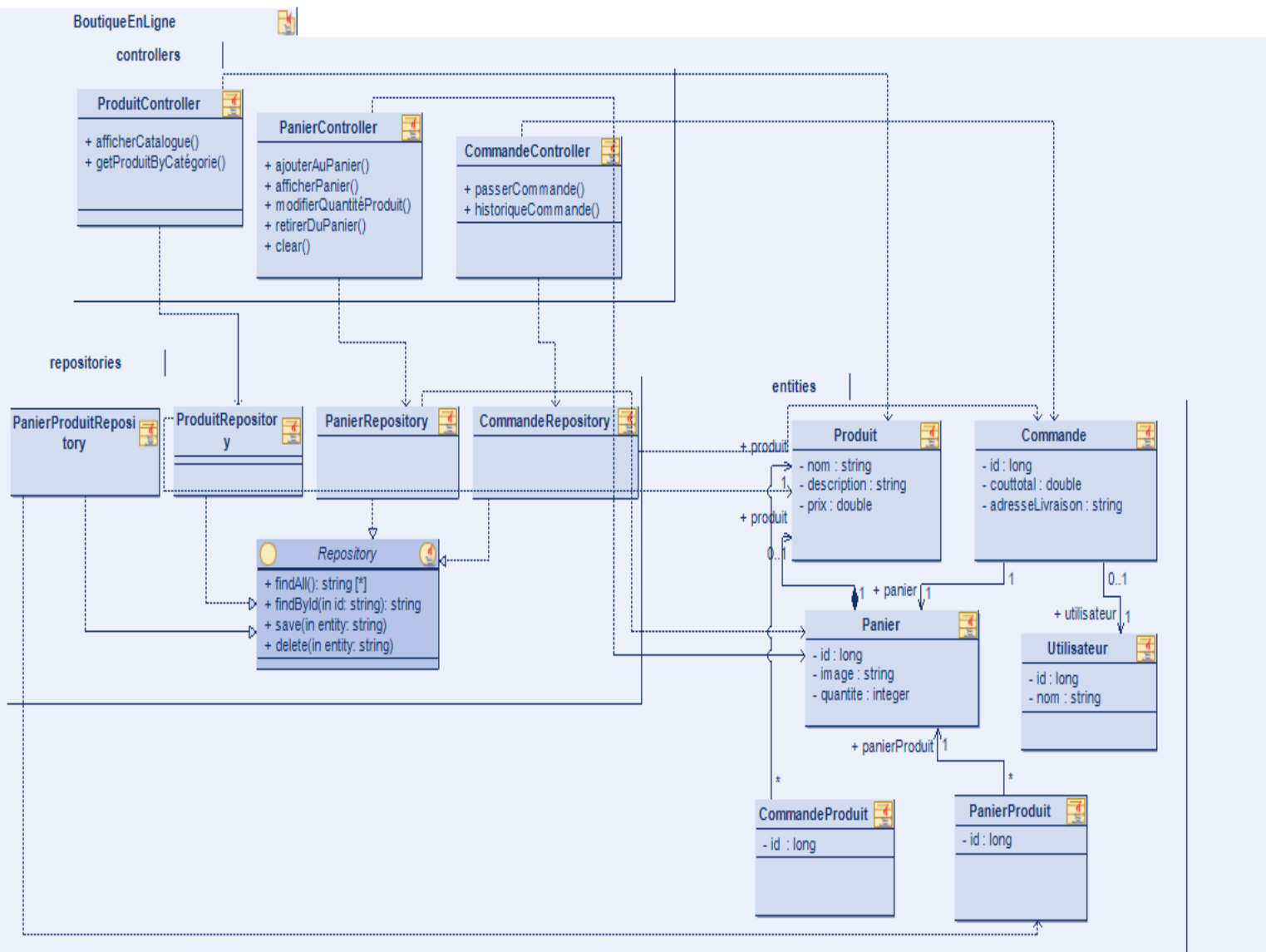
## Afficher le panier





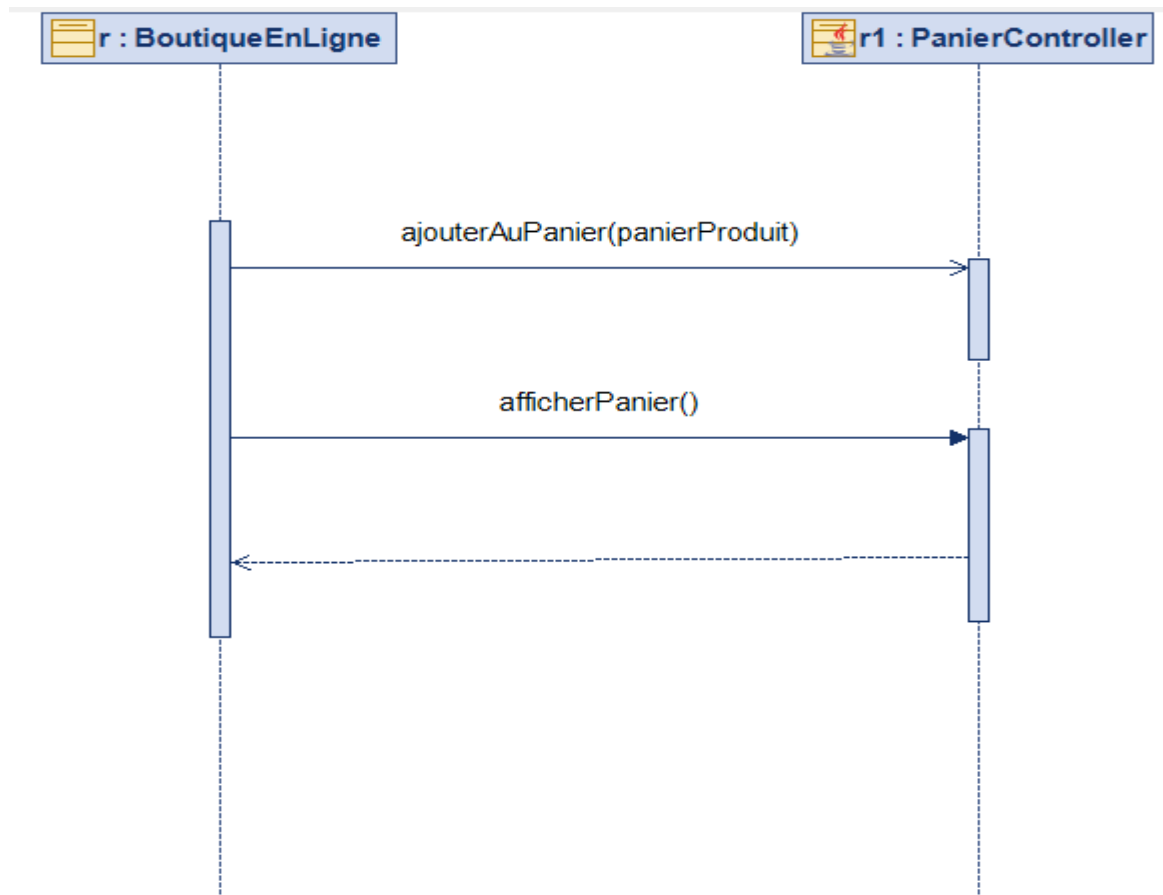
## Diagrammes de Classes de Conception

Schématisation des classes de conception avec leurs attributs et méthodes.

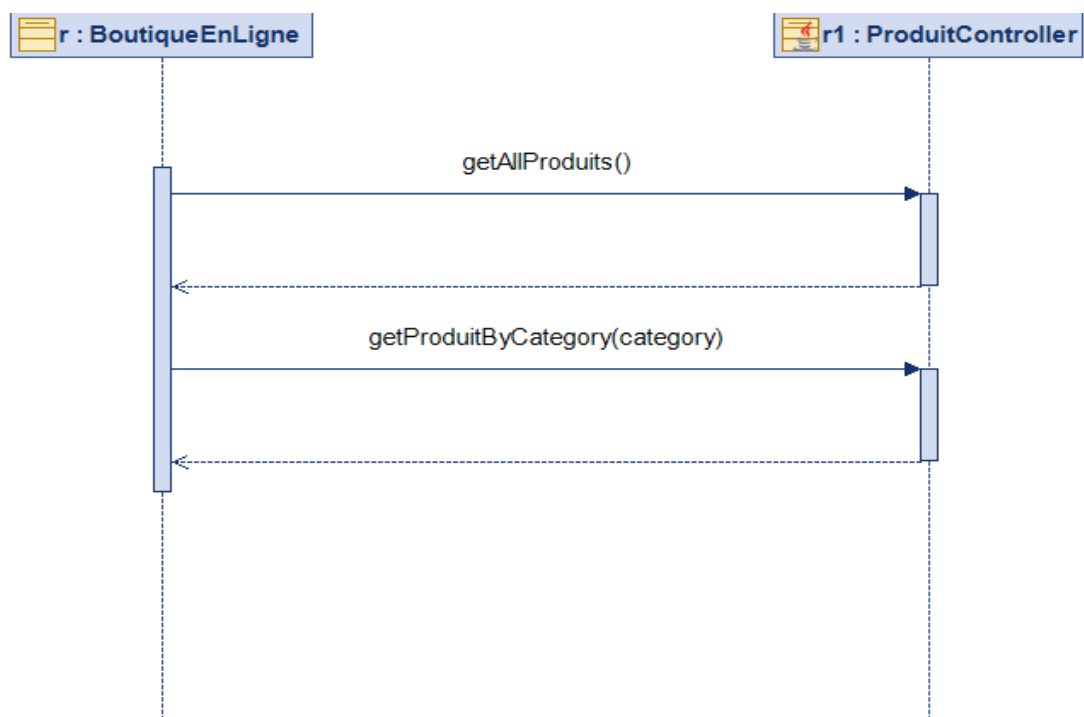


## Diagrammes de séquence de conception

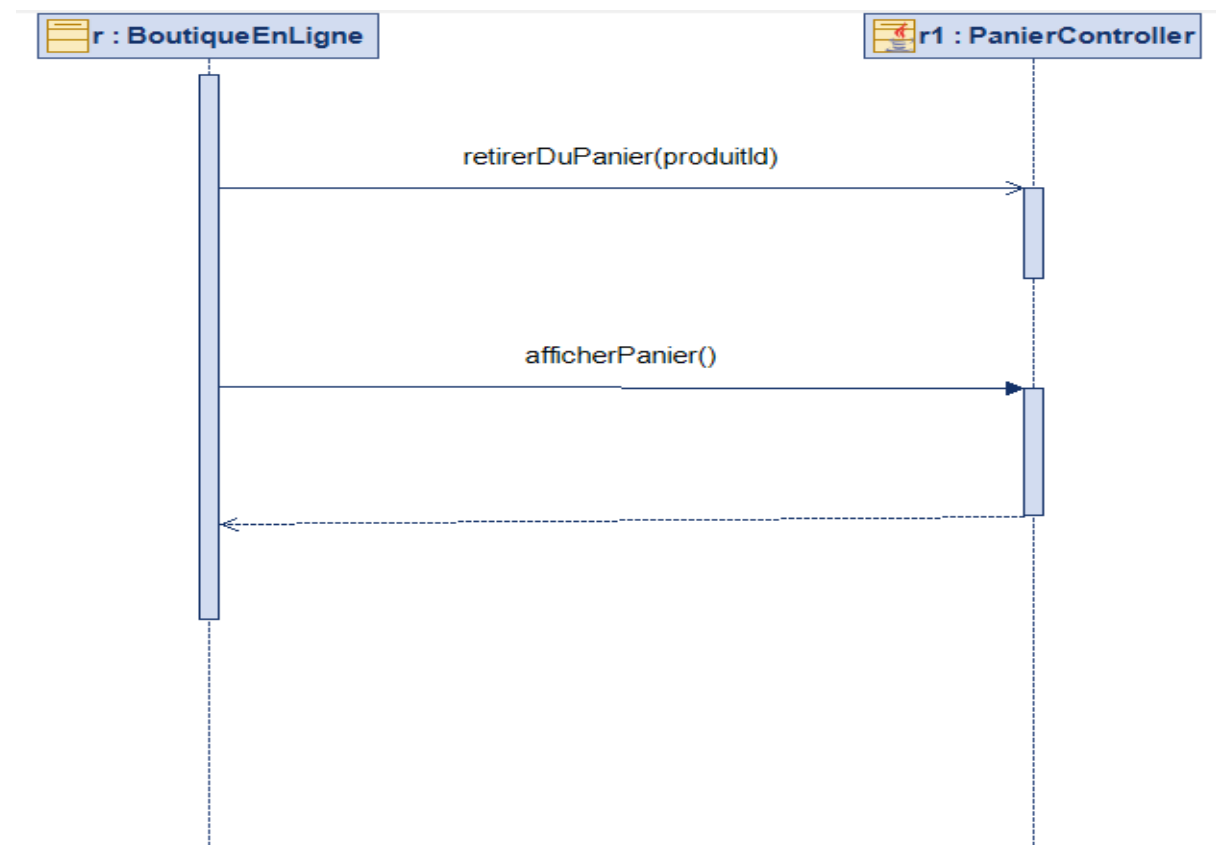
### Ajout au panier:



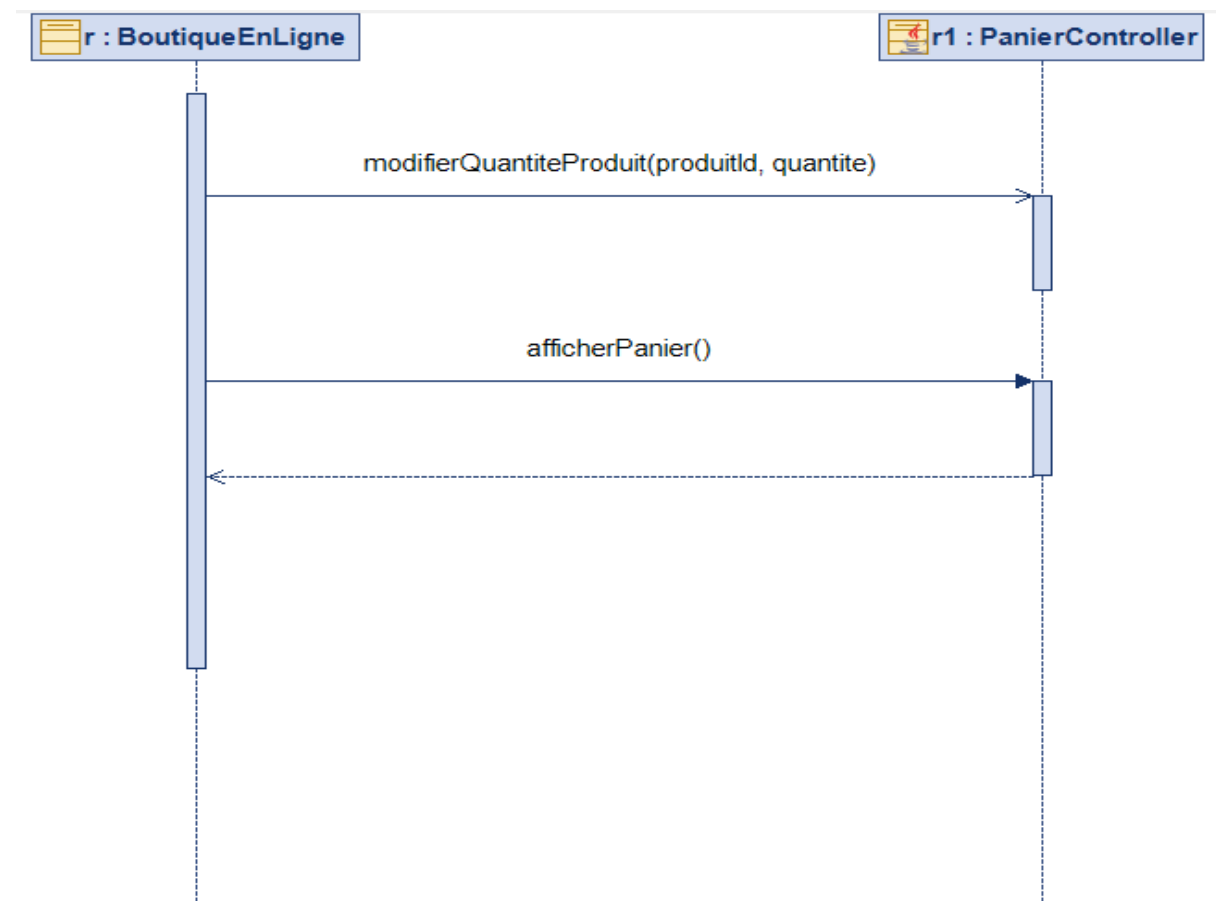
### Affichage des produits:



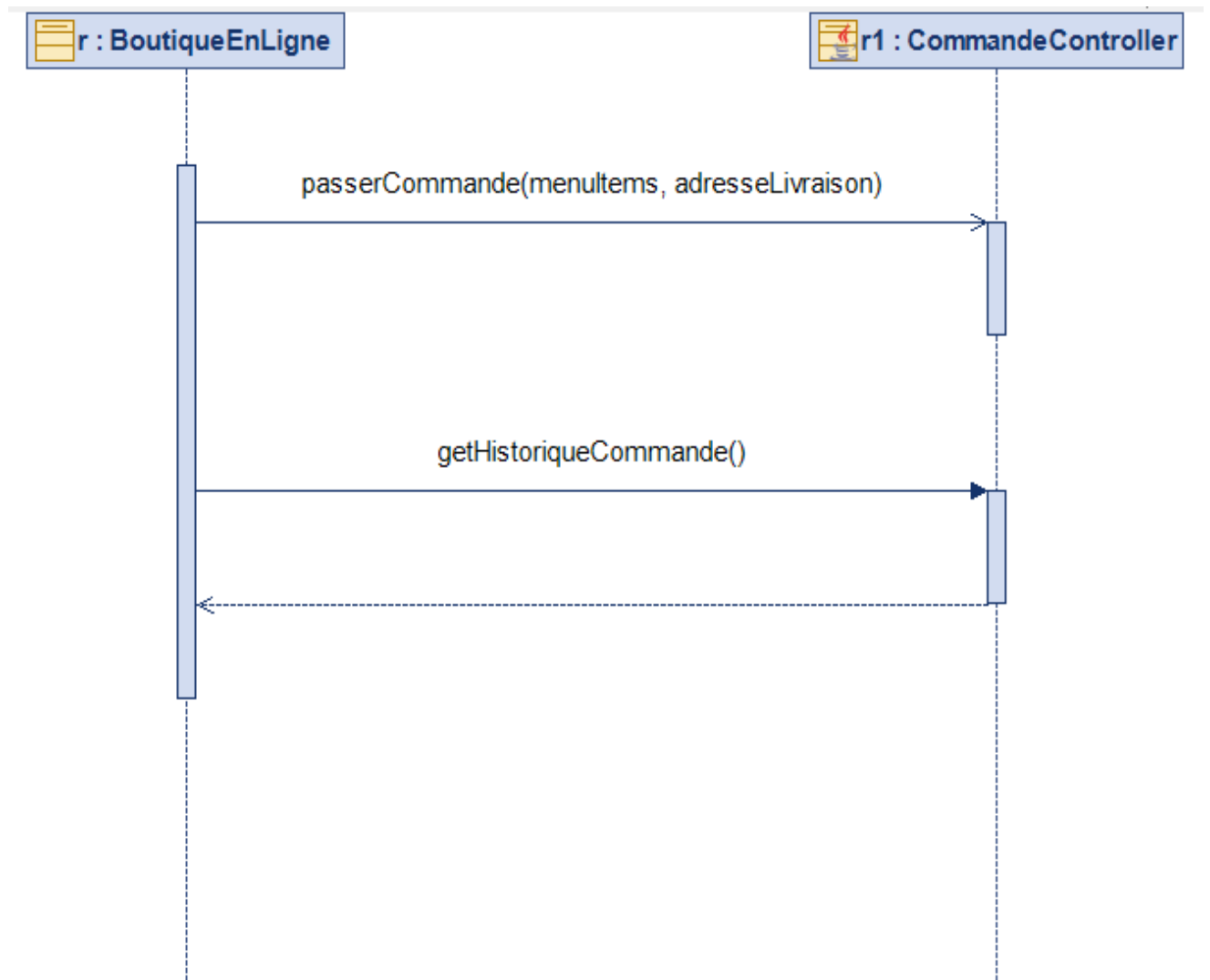
### Retirer un produit:



### Modifier la quantité d'un produit:



## Passer une commande:



## **Explications des fonctionnalités implémentées:**

### **Ajouter un Produit au Panier:**

- Les utilisateurs peuvent ajouter des produits à leur panier depuis le catalogue.
- Le diagramme de séquence illustre les étapes, y compris la validation de la disponibilité du produit et la mise à jour du panier.

### **Modifier la Quantité d'un Produit:**

- Les utilisateurs peuvent ajuster la quantité d'un produit dans leur panier.
- Le diagramme de séquence détaille les interactions entre le panier et l'utilisateur lors de cette modification.

### **Retirer un Produit du Panier:**

- La fonctionnalité permet aux utilisateurs de retirer un produit spécifique de leur panier.
- La séquence explique les étapes, telles que la mise à jour du panier après la suppression du produit.

### **Afficher les Produits:**

- Les utilisateurs peuvent visualiser la liste de produits disponible à l'achat.
- Cette fonctionnalité est liée au cas d'utilisation "Naviguer dans le catalogue".

### **Passer une Commande:**

- Les utilisateurs peuvent passer une commande à partir du contenu de leur panier.
- Le diagramme de séquence décompose le processus, montrant les étapes de création de la commande et de mise à jour des stocks.

## **Décisions d'architectures**

### **Utilisation de JPA pour la persistance des données:**

JPA a été choisi comme solution de gestion de la persistance des données. Cela permet une intégration transparente avec la base de données, facilitant la création, la modification et la récupération des entités sans dépendre d'un langage de requête spécifique.

### **Adoption de Spring MVC pour la gestion des contrôleurs :**

Raison: L'utilisation du framework Spring pour la gestion des contrôleurs offre une structure MVC robuste. Les contrôleurs Spring simplifient la gestion des requêtes HTTP, permettant une séparation claire entre la logique métier et la gestion des vues.

### **Utilisation d'annotations pour la configuration Spring et JPA:**

Les annotations ont été largement utilisées pour configurer les composants Spring tels que les contrôleurs et les services. Pour la persistance des données, les annotations JPA ont été intégrées dans les entités pour définir les relations et les propriétés des objets persistants.

En résumé, ces décisions d'architecture ont été prises dans le but de créer une application E-Shop robuste, flexible et extensible. L'utilisation judicieuse de JPA et Spring, la structuration des classes, et l'intégration d'interfaces ont contribué à une conception modulaire, facilitant la maintenance et l'évolution continue de l'application.