

Rapport de Projet de fin d'études

Pour l'Obtention du Diplôme

LICENCE SCIENCES ET TECHNIQUES

Spécialité : Génie Informatique

**Titre Titre Titre Titre Titre Titre Titre Titre
Titre Titre Titre Titre Titre**

Réalisé par :

Prénom et NOM

Sous la direction de :

M. Prénom et NOM, professeur à la FST de Settat

M. Prénom et NOM, Société

Soutenu le : ... juin 2025

JURY

M. Prénom et NOM, professeur à la FST de Settat

M. Prénom et NOM, professeur à

Année Universitaire : 2024-2025

Présentation de l'organisme d'accueil:

Dans le cadre de mon Projet de Fin d'Études, j'ai décidé de développer un projet personnel, en autonomie, sans organisme d'accueil. Il s'agit d'un dashboard e-commerce permettant à un administrateur de gérer les commandes, les produits et les clients d'une boutique en ligne.

Organigramme :

Le projet étant réalisé de manière individuelle, je suis intervenu en tant que :

- Analyste des besoins
- Développeur backend
- Développeur frontend
- Designer de l'interface

Métiers / Activité / Secteur d'activité :

Le projet s'inscrit dans le domaine du **commerce électronique (e-commerce)**.

Il permet de simuler la gestion d'une boutique en ligne du côté administrateur, avec des fonctionnalités clés utiles aux :

- Gestionnaires de produits
- Responsables de la relation client
- Responsables logistique (suivi des commandes)

Contexte du projet

◆ Critique de l'existant

Beaucoup de dashboards existants sont complexes ou payants, et peu adaptés à un usage pédagogique ou pour de petites boutiques. L'idée était donc de créer une solution simple, moderne et évolutive.

◆ Problématique

Comment concevoir une interface d'administration complète pour une boutique e-commerce qui permette de gérer efficacement les commandes, les clients et les produits ?

◆ Objectif(s) du projet

- Créer un tableau de bord d'administration clair et ergonomique
- Gérer les commandes (filtrage, statut, détails)
- Gérer les clients (liste, historique d'achats)
- Gérer les produits (CRUD, images, stock)
- Implémenter des filtres, recherches, statistiques

◆ Solution(s) proposée(s)

J'ai conçu une application full-stack avec :

- **Backend** : PHP, Laravel, Mysql.
- **Frontend** : Typescript, React.js, Tailwind CSS, avec ShadCN/UI.
- **Authentication** : Session
- **Dashboard** : Statistiques, filtres, recherches

Conclusion

Ce projet m'a permis de mobiliser l'ensemble des compétences acquises au cours de ma formation en génie informatique, notamment dans :

- Le développement web full-stack
- La conception d'interfaces modernes et réactives
- La structuration d'une base de données relationnelle
- La gestion d'un projet technique de bout en bout

Il pourrait être utilisé comme base pour une vraie boutique en ligne ou un projet professionnel futur.

Chapitre II : Analyse et Conception

1. Besoins fonctionnels

A. Gestion des Produits:

- Création de nouveaux produits
- Modification des produits existants
- Suppression de produits
- Gestion des images de produits
- Catégorisation des produits
- Filtrage et recherche de produits

○ B. Gestion des Catégories

- Création de catégories
- Modification des catégories
- Suppression de catégories
- Association produits-catégories

○ C. Gestion des Commandes

- Visualisation des commandes
- Détails des commandes
- Mise à jour du statut des commandes
- Filtrage des commandes

○ D. Gestion des Clients

- Visualisation des clients
- Détails des clients
- Modification des informations clients
- Historique des commandes par client

○ E. Tableau de Bord

- Statistiques de ventes
- Produits les plus vendus
- Commandes récentes
- Performances des ventes

○ F. Gestion du Profil

- Modification des informations personnelles
- Gestion de la photo de profil
- Mise à jour de la localisation

○

1. Identification et Description des Acteurs :

A. Administrateur

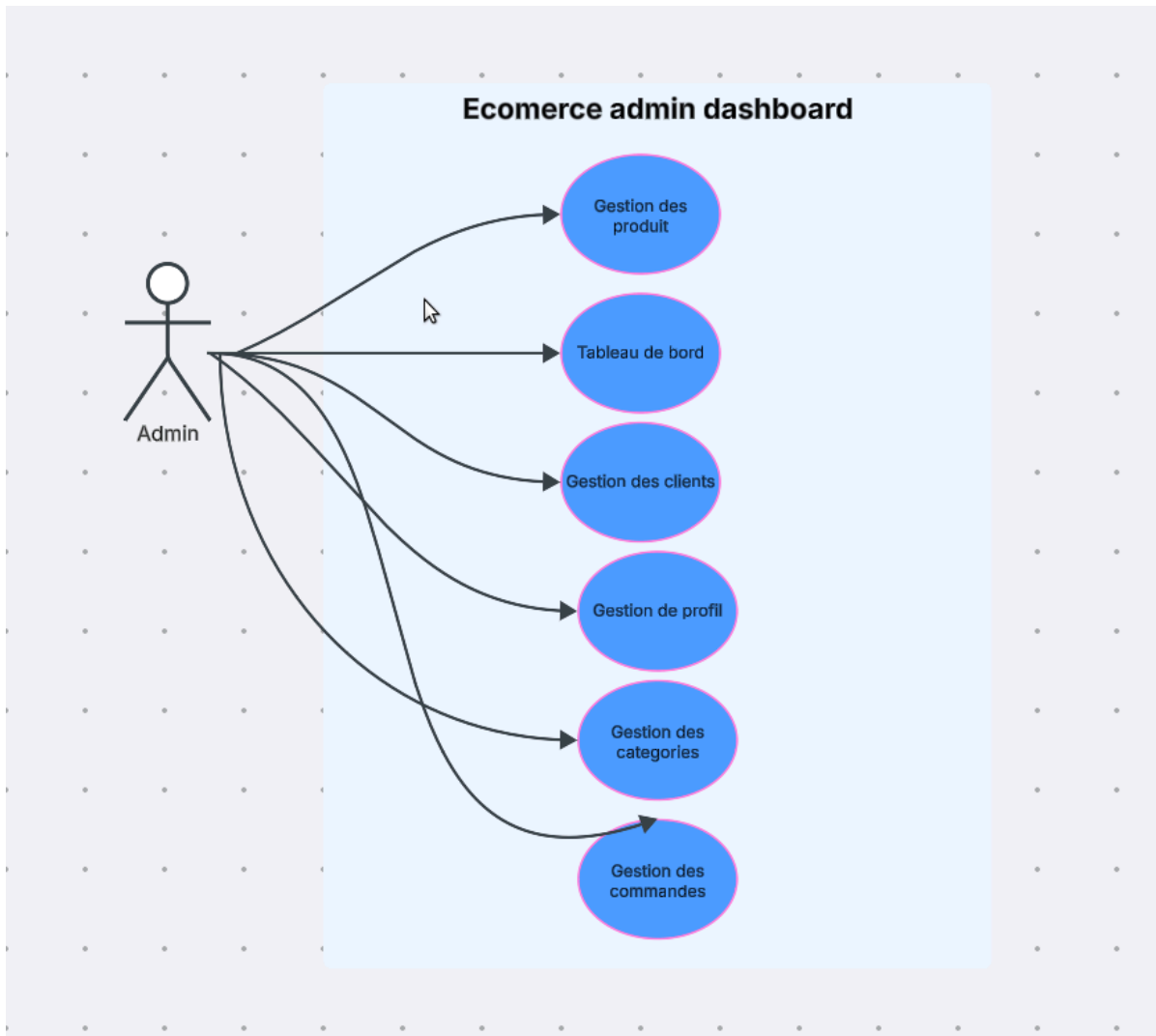
- Description: Utilisateur principal du système
- Responsabilités:
- Gestion des produits
- Gestion des catégories
- Gestion des commandes
- Gestion des clients
- Accès au tableau de bord
- Gestion du profil

B. Système :

- Description: Entité technique gérant les opérations automatiques
- Responsabilités:
- Gestion des notifications
- Gestion des données
- Validation des opérations
- Gestion de la sécurité

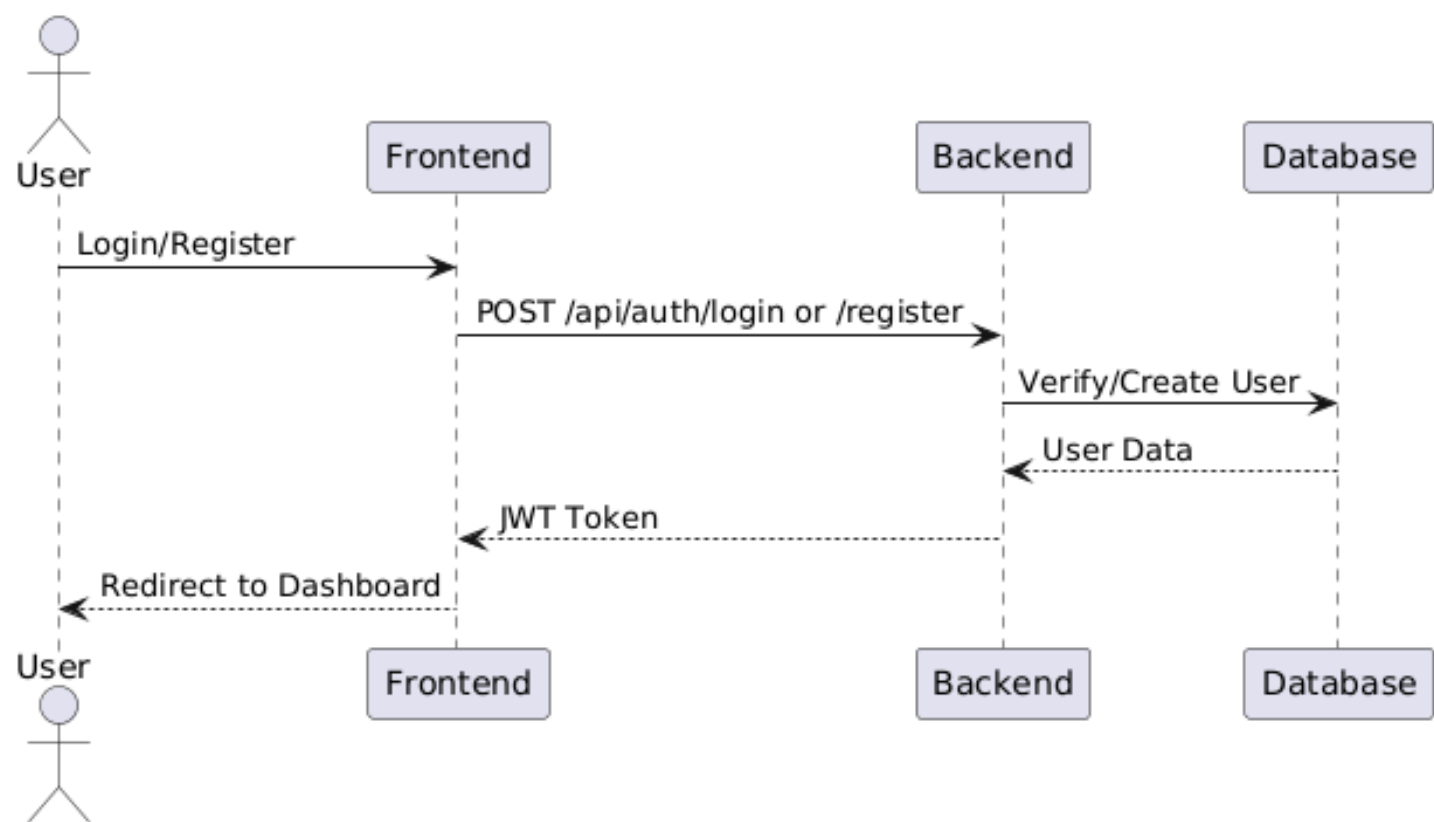
○

3 . Cas d'Utilisation/Fonctionnalités Attendues

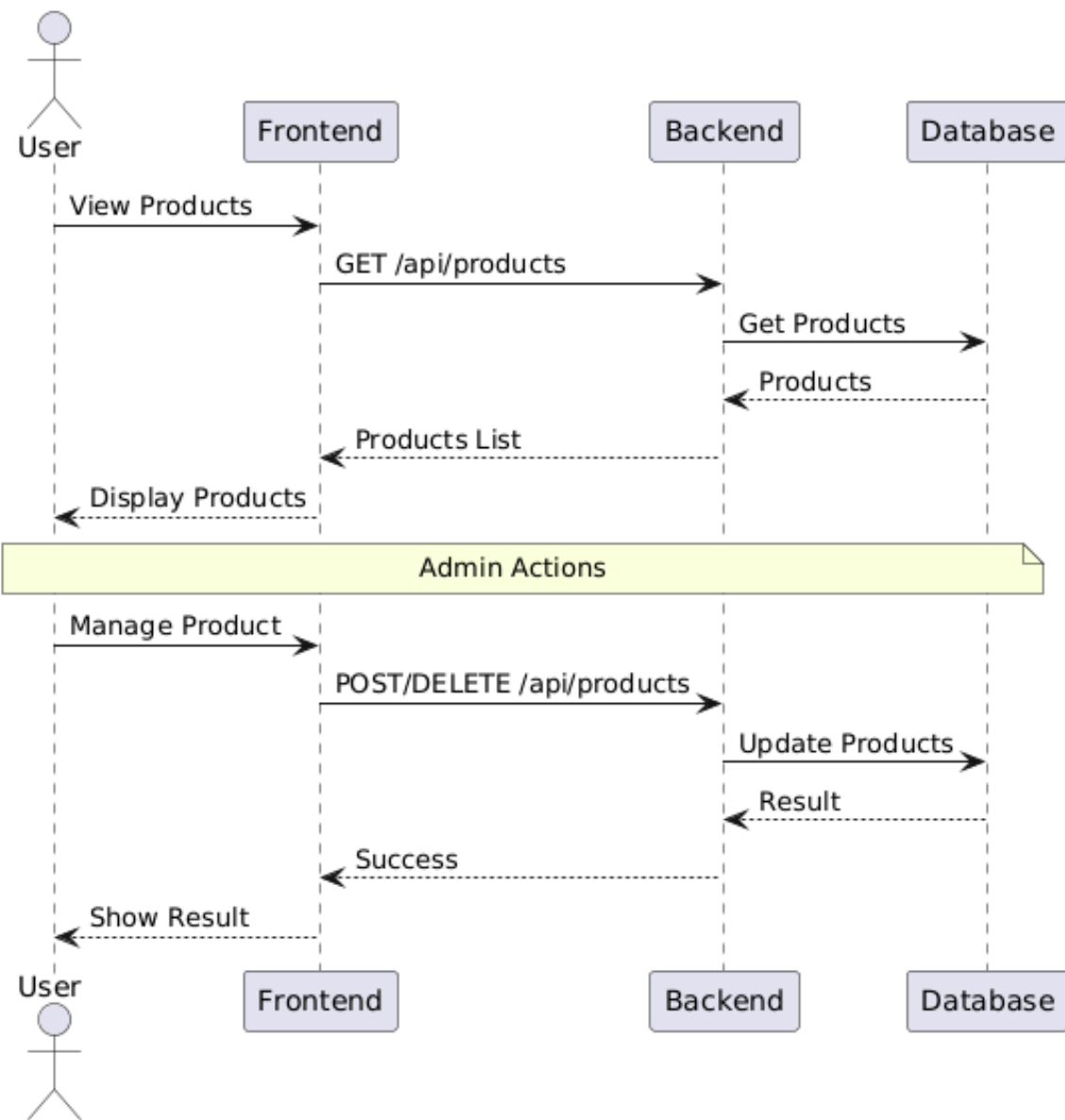


4 . Diagramme de Séquence (Exemple pour la Gestion des Produit, Authentifaction...)

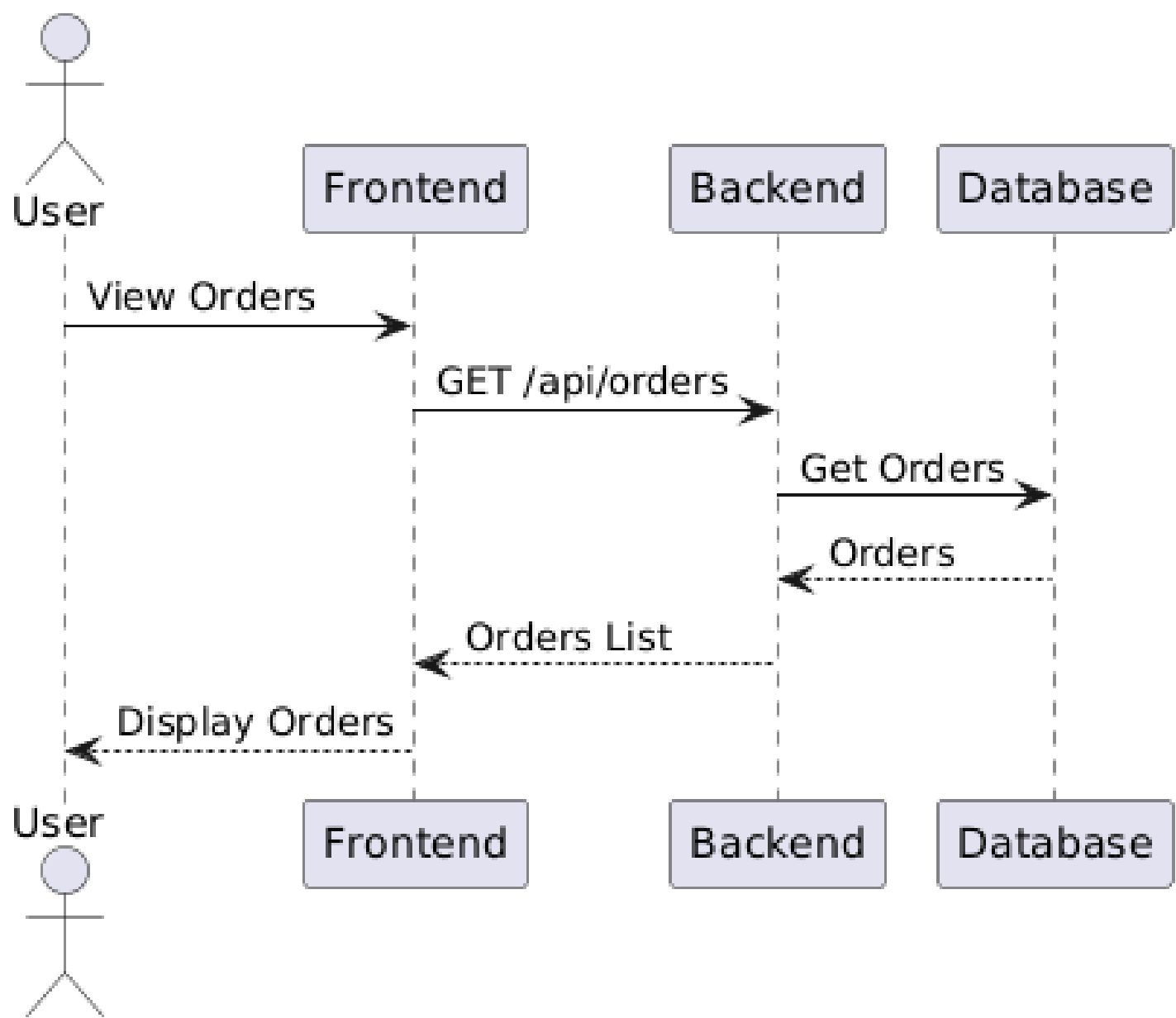
Authentication Flow:



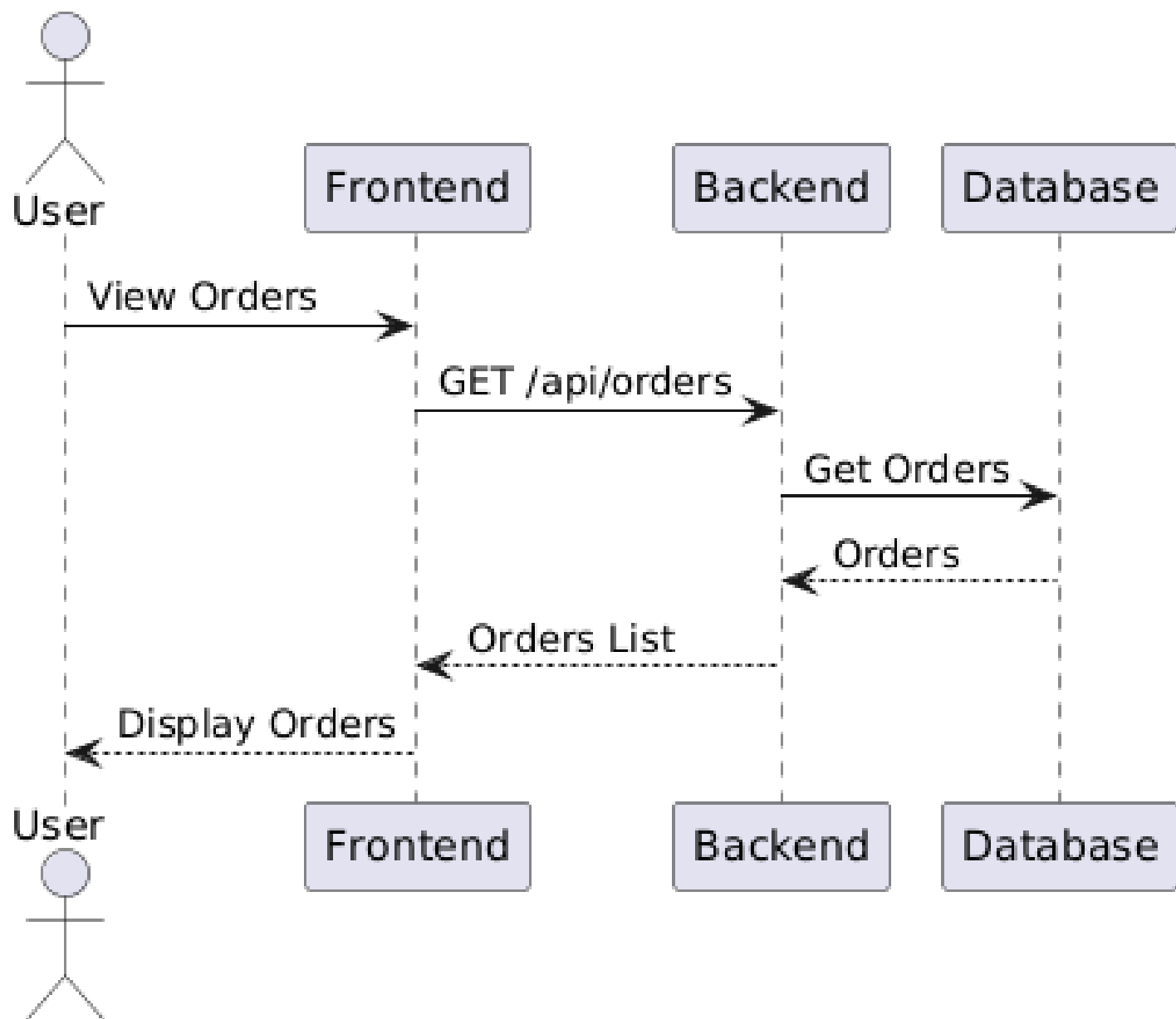
Product Flow:



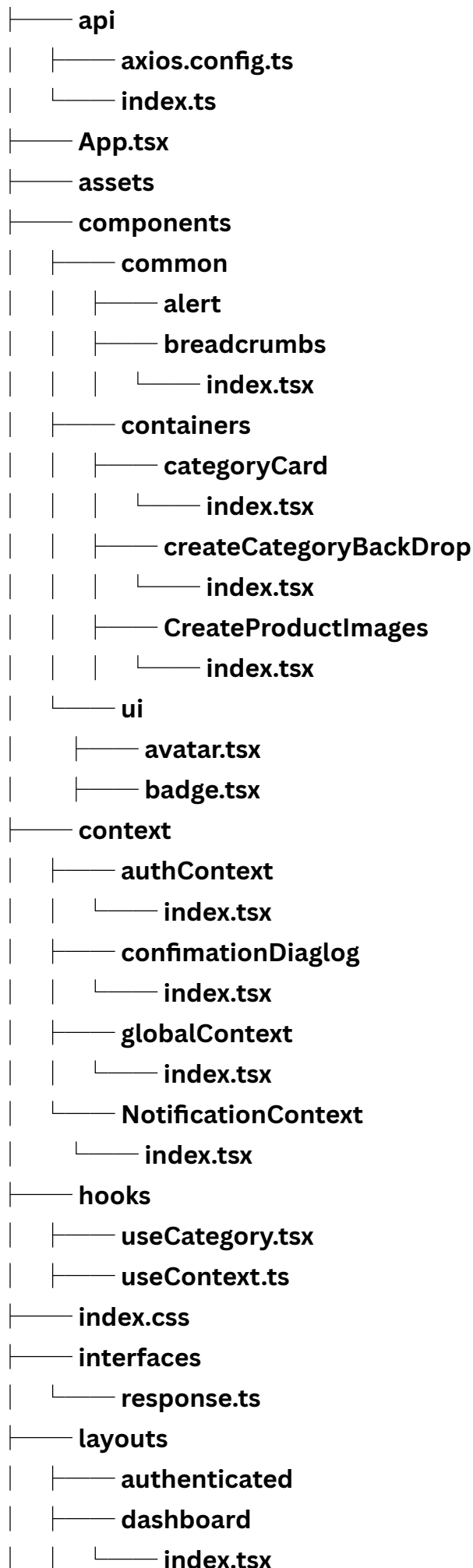
5 . Commandes Flow



6 . Categories Flow:



src



```
|   |   | DashboardLayout.tsx
|   |   |   | defaultLayout
|   |   |   |   | index.tsx
|   |   | lib
|   |   |   | utils.ts
|   |   |   | Validator.ts
|   |   | main.tsx
|   |   | pages
|   |   |   | Analytics.tsx
|   |   |   | category
|   |   |   |   | components
|   |   |   |   |   | categoriesTable
|   |   |   |   |   |   | index.tsx
|   |   |   |   |   |   | filterCategory
|   |   |   |   |   |   |   | index.tsx
|   |   |   |   |   |   |   | index.ts
|   |   |   |   |   |   | index.tsx
|   |   |   |   | categoryDetail
|   |   |   |   |   | components
|   |   |   |   |   |   | categoryInfo
|   |   |   |   |   |   |   | index.tsx
|   |   |   |   |   |   |   | categoryMedia
|   |   |   |   |   |   |   |   | index.tsx
|   |   |   |   |   | hooks
|   |   |   |   |   |   | index.tsx
|   |   |   |   |   |   | useCategoryDetail
|   |   |   |   |   |   |   | index.tsx
|   |   |   |   |   | index.tsx
|   |   |   |   | createCategory
|   |   |   |   |   | index.tsx
|   |   |   |   | createProduct
|   |   |   |   |   | index.tsx
|   |   |   |   | customerDetail
|   |   |   |   |   | components
|   |   |   |   |   |   | customerGeneralInfomation
|   |   |   |   |   |   |   | index.tsx
|   |   |   |   |   | hooks
|   |   |   |   |   |   | useCustomerDetail.tsx
|   |   |   |   |   |   | index.tsx
|   |   |   |   | customers
```

- | | | — components
- | | | | — customersFilter
- | | | | | — index.tsx
- | | | | — index.ts
- | | | — hooks
- | | | | — index.tsx
- | | | — index.tsx
- | | — dashboard
- | | | — index.tsx
- | | — login
- | | | — index.tsx
- | | — orderDetail
- | | | — components
- | | | — hooks
- | | | | — useOrderDetail.tsx
- | | | — index.tsx
- | — routes
- | | — guestRoute.tsx
- | | — index.tsx
- | | — protectedRoutes.tsx
- | — services
- | | — localStorageService.ts
- | | — localStorage.ts
- | — types
- | | — api.ts
- | | — auth
- | | | — index.ts
- | | — category.ts
- | — utils
- | | — index.ts
- | — vite-env.d.ts