

REPUBLIQUE DU SENEGAL



UNIVERSITE CHEIKH ANTA DIOP DE DAKAR

Département de Mathématiques-Informatique

Section Informatique



**PROJET MICROSERVICES Master 1
SIR**

Sujet : microservices : Gestion d'un site e-commerce.

Présenté par :

Baye Serigne Seck

Superviseur :

Pr Samba Ndiaye

TABLE DES FIGURES

Figure 1 : logo de MySQL

Figure 2 : STS

Figure 3 : Logo de Visual Studio Code.

Figure 4 : Logo de Angular js.

Figure 5 : Logo de Spring Boot.

Figure 6 : Ajouter Un Client

Figure 7 : Liste des Clients

Figure 8 : Modifier Un Client

Figure 9 : Supprimer Un Client

Figure 10 : Ajouter Une Commande pour un Client.

Figure 11 : Liste des Commandes et les clients qui les ont commandés

Figure 12 : Modification d'une commande

Figure 13 : Supprimer Une Commande

Figure 14 : Ajouter Un produit

Figure 15 : Liste des Produits

Figure 16 : Modifier Un Produit

Figure 17 : Supprimer Un Produit

Figure 18 : Ajouter Details Commande

Figure 19 : Liste des Détails Commande

Figure 20 : Supprimer Détails Commande

Table des matières

Introduction.....	3
I. Le sujet	3
I.1 Qu'est-ce qu'un Microservice ?.....	3
I.2 Qu'est-ce qu'un API ?	3
I.3 différence entre une API et un microservice	4
I.4 Qu'est-ce Angular ?	4
II. Conception	5
II.1 Base de données.....	5
II.2 Environnement de développement	5
II.2.1 Spring Tools Suite 4	6
II.2.1 Visual Studio Code	6
II.3 Langage de programmation	7
II.3.1 Le Framework Angular js	7
II.3.2 Framework Spring Boot.....	8
III. Présentation des Solutions	8
III.1 Présentation de l'application	8
Conclusion Générale.....	17

Introduction

L'utilisation des technologies de l'information et de la communication (TIC) fait dorénavant partie de l'environnement de travail de tous les structures commerciales, aussi bien en établissement public qu'en pratique privée. Ce concept n'est pas nouveau mais le Sénégal n'est pas en avance dans ce domaine. Cependant, ces dernières années ont été marquées par la volonté politique d'adopter de nouvelles dispositions pour son développement.

Consciente de la nécessité d'utiliser les nouveaux moyens d'accès à l'information pour pouvoir créer et entretenir des relations profitables d'autant plus que certaines personnes ne peuvent se déplacer, ou habitent dans un désert commercial. On a décidé de mettre en place une plateforme de gestion d'un site e-commerce.

I. Le sujet

I.1 Qu'est-ce qu'un Microservice ?

Il existe un certain nombre de **définition des microservices**. La plus commune et généralisée reste celle de Martin Fowler qui dit :

« Le style architectural des microservices est une approche permettant de développer une application unique sous la forme d'une suite logicielle intégrant plusieurs services. Ces services sont construits autour des capacités de l'entreprise et peuvent être déployés de façon indépendante. »

Concrètement, les **microservices** sont une méthode développement logiciel utilisée pour concevoir une application comme un ensemble de services modulaires. Chaque module répond à un objectif métier spécifique et communique avec les autres modules.

Pour Fowler, les microservices doivent nécessiter le strict minimum en termes de gestion centralisée des services et peuvent être créés sous différents langages de programmations.

I.2 Qu'est-ce qu'un API ?

Le terme API est l'acronyme de Application Programming Interface qui signifie "Interface de programmation applicative". En termes simples, les APIs sont des "passerelles" qui permettent simplement à deux ou plusieurs applications de communiquer entre eux et de permettre et faciliter les échanges de données entre un client et un Serveur.

Une API définit exactement « les méthodes par lesquelles un programme peut interagir avec un autre ». Chaque fois que vous utilisez une application comme Facebook sur votre mobile, que vous affichez une Google Map dans votre navigateur web ou que vous consultez la météo sur votre smartphone, vous utilisez une API sans probablement le savoir ! Par exemple lorsque vous consultez une Google Maps (carte interactive) intégrée à un site Web, ce site Web utilise l'API Google Maps pour intégrer cette carte et afficher dynamiquement les données.

I.3 différence entre une API et un microservice

Les microservices et les API ont fait l'objet de beaucoup de communication à tel point qu'il existe aujourd'hui une certaine confusion sur la façon dont ils sont liés.

Tout d'abord, il faut insister sur le fait que les microservices et les APIs sont deux choses différentes. Un **microservice est une architecture pour les applications** qui sépare une application en plusieurs petits services web autonomes.

L'API (Application programming Interface) est, comme son nom l'indique une interface de programmation. Elle constitue le cadre à travers lequel un développeur peut interagir avec une application. Les APIs sont un ensemble de classes, méthodes, fonctions et constantes qui sert d'interface par laquelle un logiciel peut offrir ses services à d'autres logiciels. Elles servent concrètement à accéder aux données d'une application et à utiliser ses fonctionnalités.

La confusion se produit car il existe un chevauchement entre ces deux concepts informatiques. De **nombreux microservices utilisent en effet des API** pour communiquer entre eux.

I.4 Qu'est-ce Angular ?

Angular est un **Framework** de conception d'applications et une plate-forme de développement permettant de créer des applications **monopage** efficaces et sophistiquées.

Ces documents Angular vous aident à apprendre et à utiliser le **Framework** et la plate-forme de développement Angular, de votre première application à l'optimisation d'applications complexes d'une seule page pour les entreprises. Les tutoriels et les guides incluent des exemples téléchargeables pour accélérer vos projets.

II. Conception

Effectuée après l'analyse des besoins, la conception est un processus créatif et rigoureux qui vise à décrire de façon très précise et indépendamment de l'implémentation, le fonctionnement du futur système dans l'optique de faciliter sa réalisation. Elle répond à la question "Comment faire le système ?" et décompose de façon modulaire le système à mettre en place. Ainsi, il convient de définir l'architecture du système, des modèles détaillés qui pourront être implémentés sans la moindre retouche.

II.1 Base de données

Pour la conservation et la restitution des données de la plateforme, l'utilisation d'une base de données est inéluctable. De nos jours, plusieurs Systèmes de Gestion de Base de Données (SGBD) existent dont les plus fréquents sont MySQL, SQL Server, Oracle, PostgreSQL. En ce qui concerne cette plateforme, on a préféré MySQL pour les traitements effectués sur nos données et gérer les utilisateurs.

MySQL est un système de gestion de bases de données relationnelles (SGBDR). Il fait partie des logiciels de gestion de base de données les plus utilisés au monde, autant par le grand public (applications web principalement) que par des professionnels, en concurrence avec Oracle, PostgreSQL et Microsoft SQL Server .



Figure 1 : logo de MySQL

II.2 Environnement de développement

En programmation informatique, un environnement de développement est un ensemble d'outils qui permet d'augmenter la productivité des programmeurs qui développent des logiciels.

Un IDE se compose généralement d'un éditeur de code, d'un compilateur, d'un débogueur et d'un générateur d'interface graphique. Certains permettent également le travail à plusieurs dans le cadre de projet d'envergure et, dans ce cadre, la gestion des différentes versions d'un code.

Un IDE peut être spécialement conçu pour un langage comme phpStorm ou RubyMine d'autres plus généralistes, mais dans tous les cas, il est rarement mono-langage presque toujours lié à un écosystème (Java, Développement Web etc.).

Parmi les IDE, nous pouvons citer entre autres Scilab, Visual Studio, Sublime Text, Eclipse, NetBeans, Zend Studio, Delphi...

II.2.1 Spring Tools Suite 4

Spring Tool Suite (STS) est un **IDE** étendu pour **Eclipse**. Il se spécialise dans le développement des applications **Spring**. SpringBoot est une extension du Spring Framework pour mettre en place rapidement des applications Java. Il nous permet de construire des projets java déjà préconstruit et disposant des configurations Spring de base.



Figure 2 : STS

II.2.1 Visual Studio Code

Visual Studio Code est un éditeur de code source léger mais puissant qui s'exécute sur votre bureau et est disponible pour Windows, macOS et Linux. Il est livré avec un support intégré pour JavaScript, TypeScript et Node.js et dispose d'un riche écosystème d'extensions pour d'autres langages et runtimes (tels que C ++, C #, Java, Python, PHP, Go, .NET).

Activez des langues, des thèmes, des débogueurs, des commandes, etc. supplémentaires. La communauté grandissante de VS Code partage sa sauce secrète pour améliorer votre flux de travail.

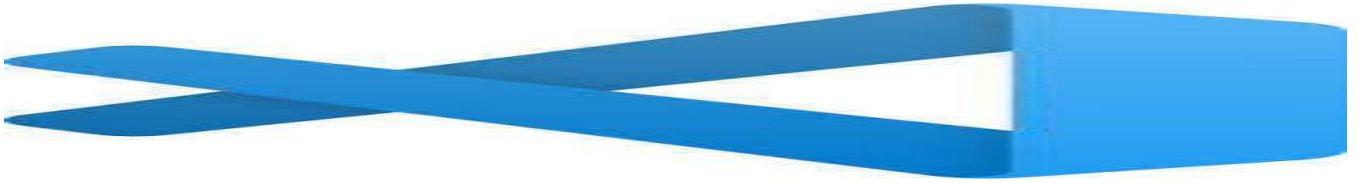


Figure 3 : Logo de Visual Studio Code.

II.3 Langage de programmation

Le choix d'un langage de programmation dépend de notre expérience en matière de langage et de la portée de l'application que nous réalisons. Alors que certaines applications sont souvent créées à l'aide d'un seul langage, il n'est pas rare d'utiliser plusieurs langages pour développer les grandes applications.

II.3.1 Le Framework Angular js

Angular est un framework Javascript qui permet de réaliser des applications WEB Cross Platform : Web, mobile et desktop.

Il est possible de développer sur Angular soit en Javascript natif, soit en Dart (un langage développé par Google), soit en TypeScript (un langage développé par Microsoft).

Dart et Typescript sont des surcouches de Javascript qui apportent notamment un typage fort, en gardant la syntaxe du langage. Ils permettent de faciliter le développement et le test des applications web.

Beaucoup plus rapide que la version 1, qui s'appelait AngularJS, Angular version 2 améliore particulièrement les performances, la structuration et la réutilisabilité du code source.

Enfin il est possible d'utiliser Angular quelle que soit son architecture serveur, à savoir avec JAVA, PHP, NodeJs, ou autres



Figure 4 : Logo de Angular js.

II.3.2 Framework Spring Boot

La bibliothèque java permet d'automatiser la génération de documentation API à l'aide de projets spring boot. Fonctionne en examinant une application au moment de l'exécution pour déduire la sémantique de l'API en fonction des configurations de ressort, de la structure de classe et de diverses annotations. Springdoc-openapi

Génère automatiquement de la documentation dans les API au format JSON/YAML et HTML. Cette documentation peut être complétée par des commentaires à l'aide d'annotations swagger-api.



Figure 5 : Logo de Spring Boot.

III. Présentation des Solutions

Nous allons dans cette section présenter à l'aide de certaines captures d'écran, les fonctionnalités les plus importantes.

III.1 Présentation de l'application

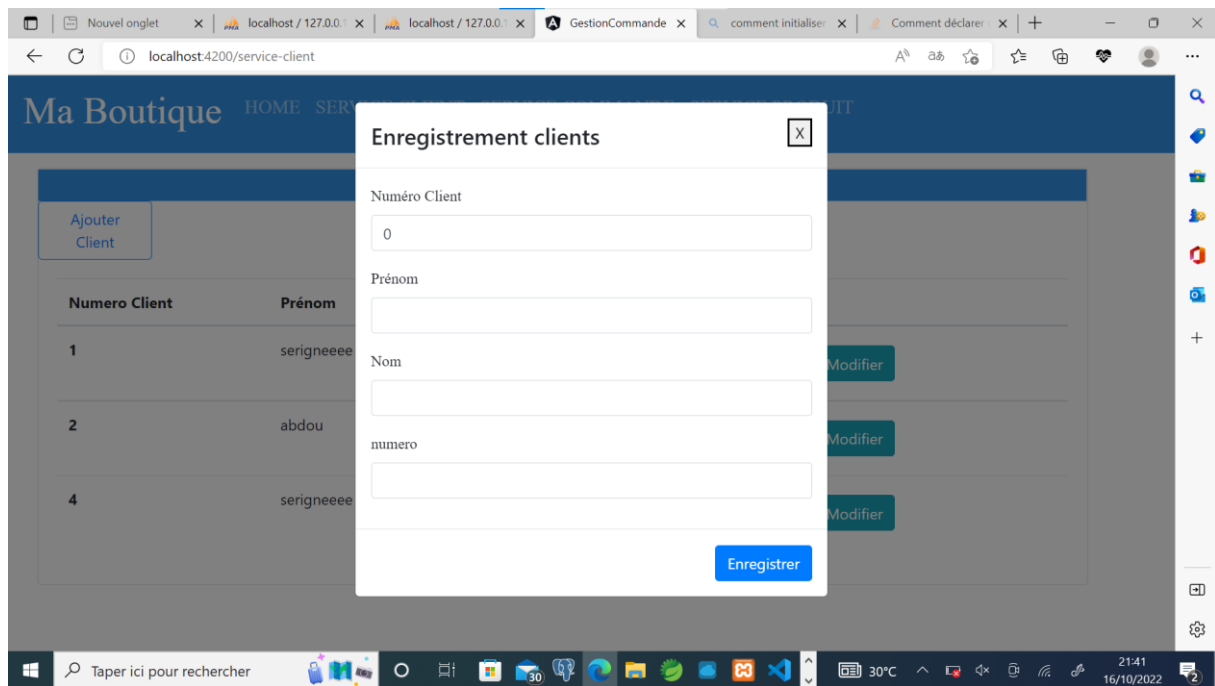


Figure 6 : Ajouter Un Client

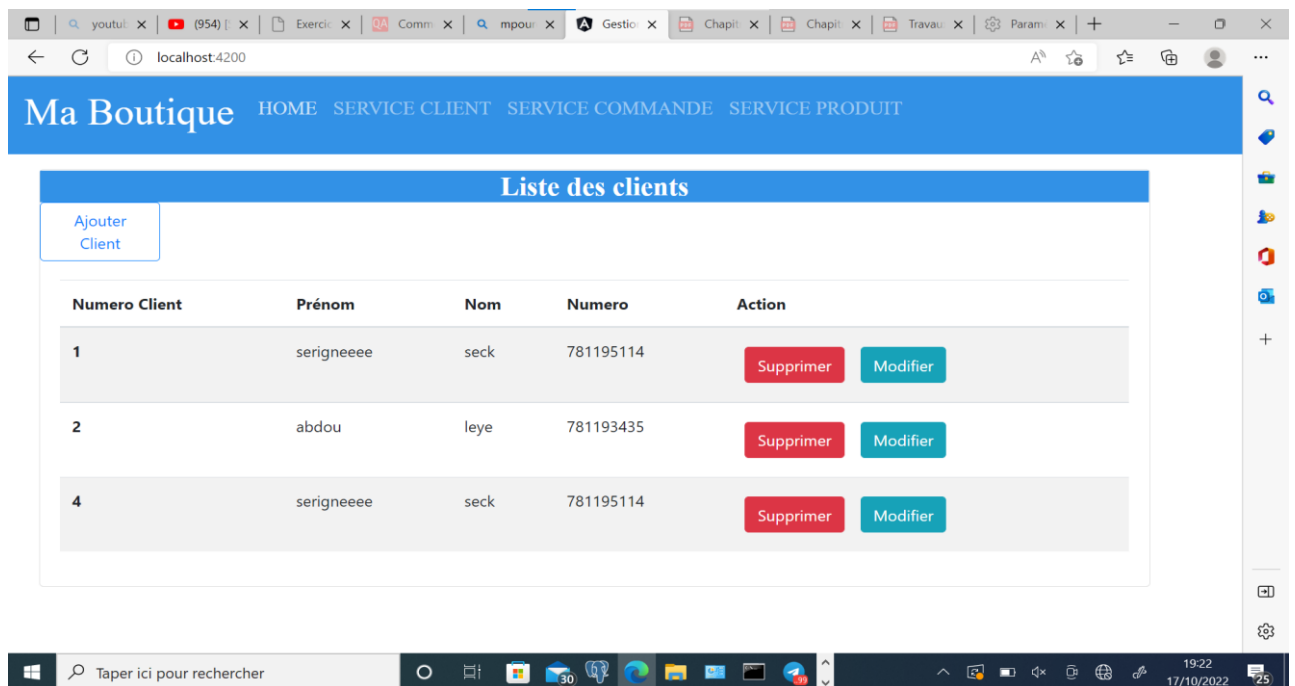


Figure 7 : Liste des Clients

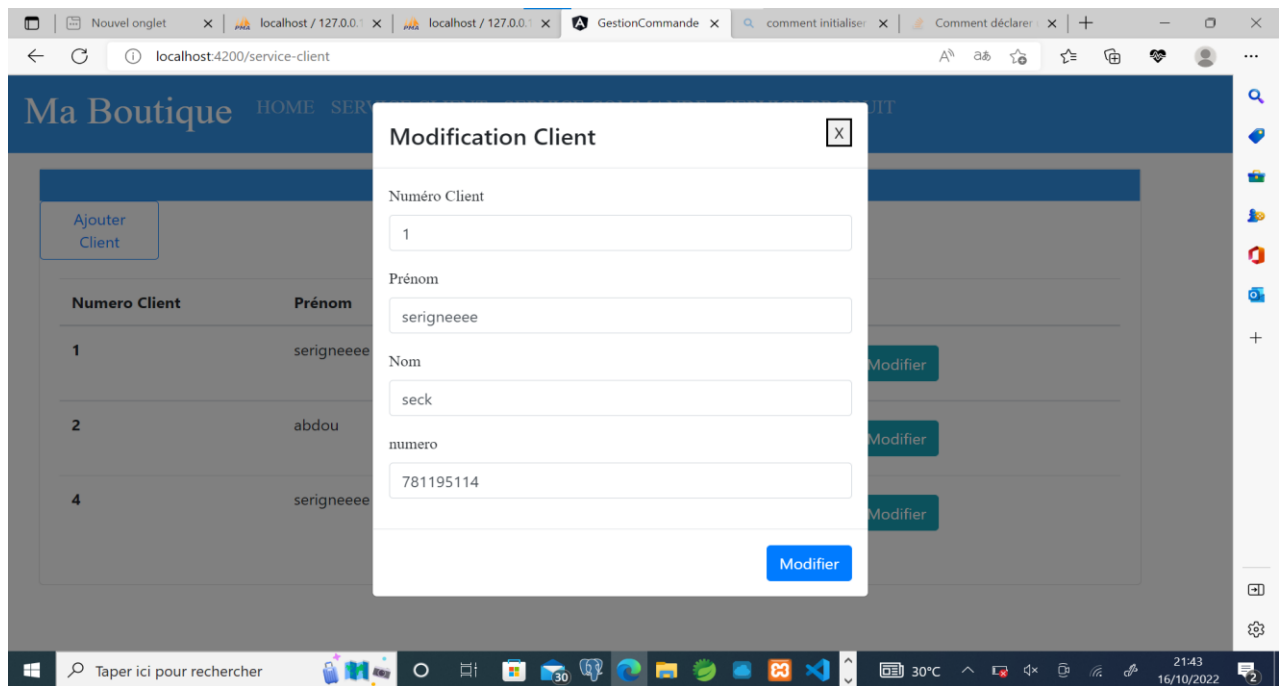


Figure 8 : Modifier Un Client

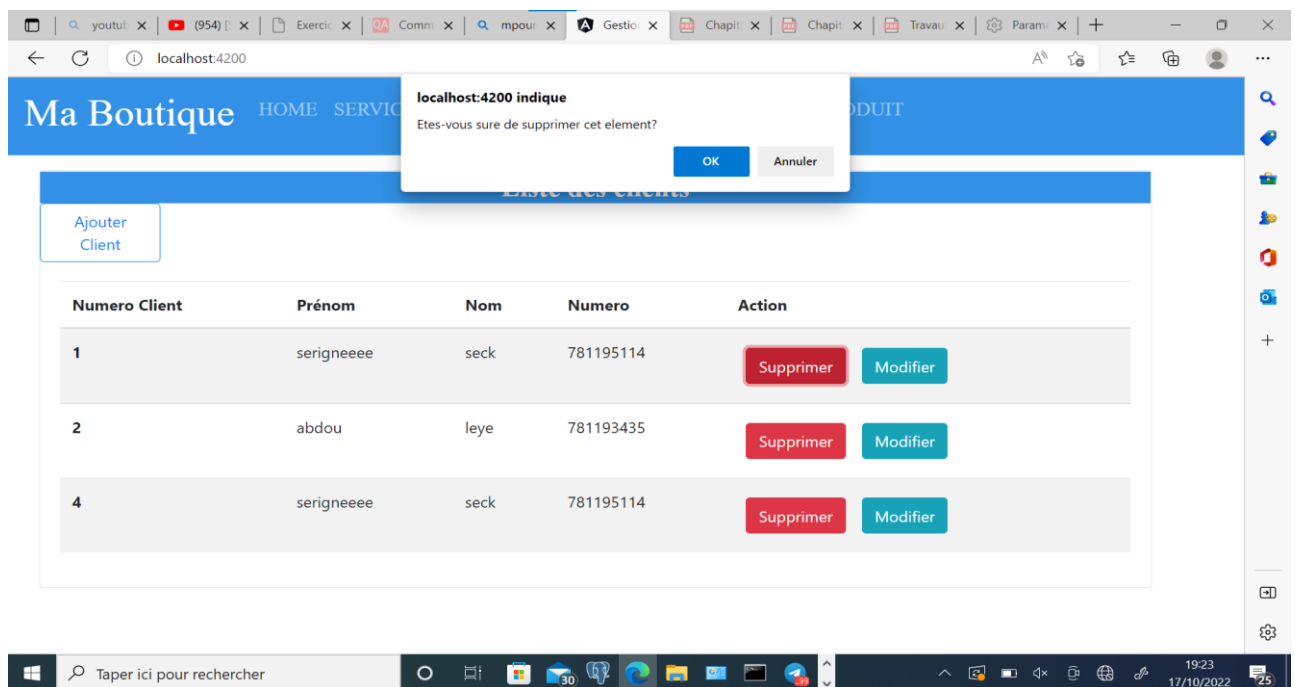


Figure 9 : Supprimer Un Client.

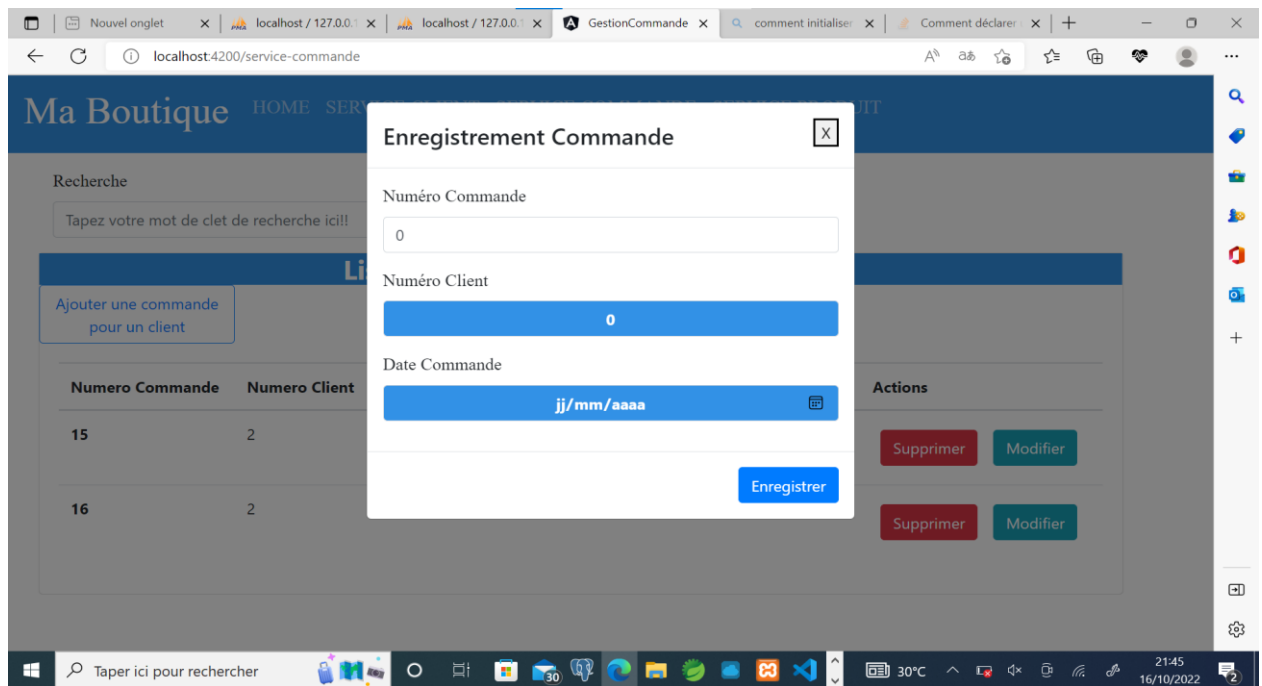


Figure 10 : Ajouter Une Commande pour un Client.

La figure 10 ci-dessus représente le formulaire d'ajout de commande, pour ajouter la commande on saisie le numéro de commande et le numéro du client, si le client existe dans la table cliente du microservice client, les autres informations qui concernent le client vont être récupéré dans la table cliente à l'aide de la méthode **Feign**.

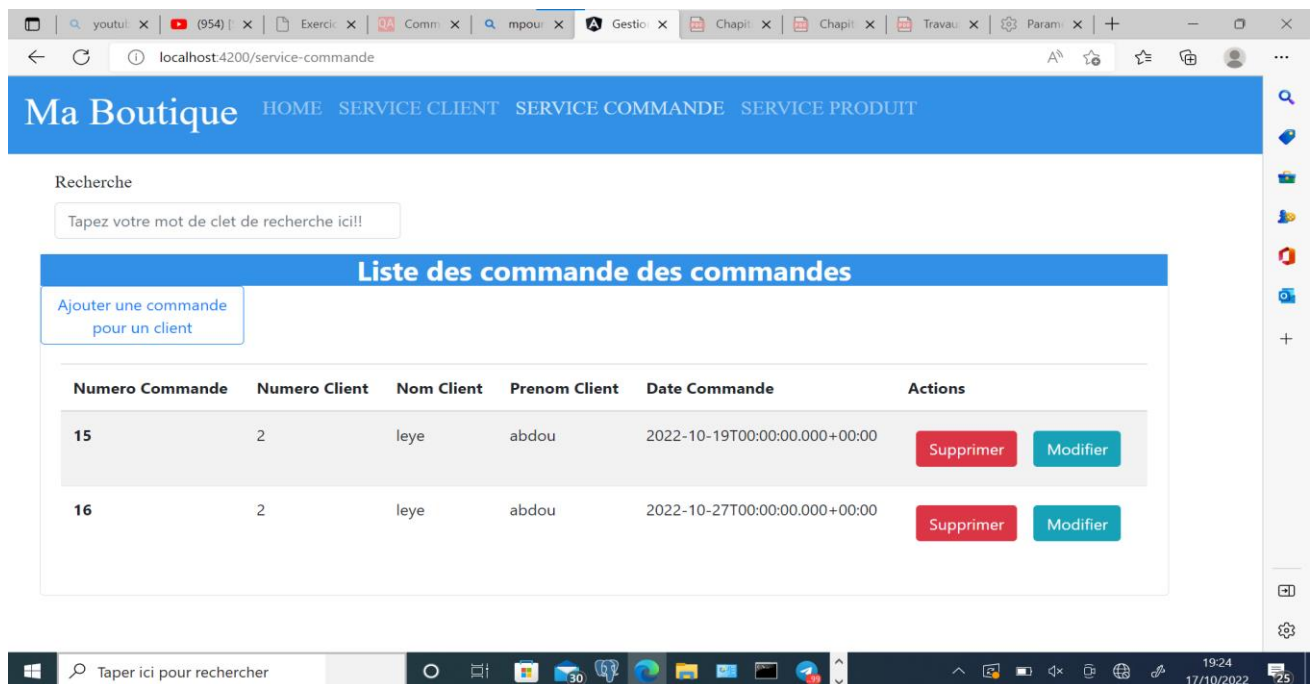


Figure 11 : Liste des Commandes et les clients qui les ont commandés.

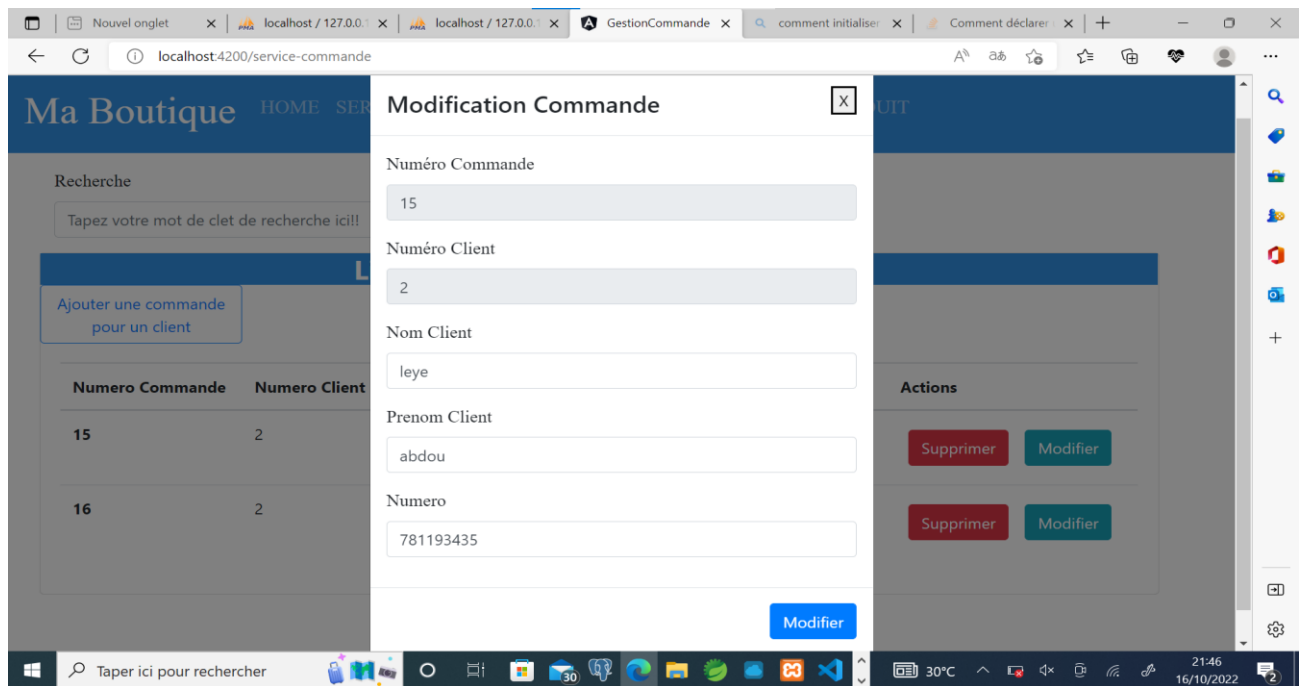


Figure 12 : Modification d'une commande

La figure 12 ci-dessus représente le formulaire de modification d'une commande client, ce pendant les champs Numéro Client et Numéro Commande ne sont pas modifiables.

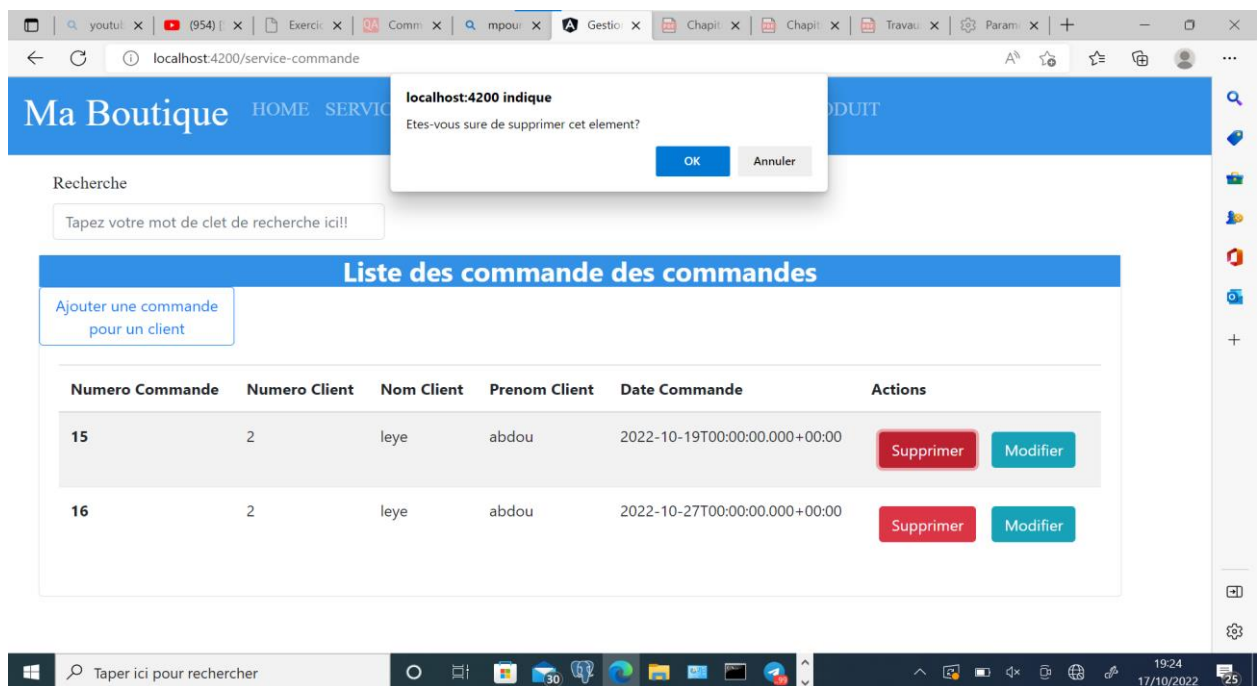


Figure 13 : Supprimer Une Commande

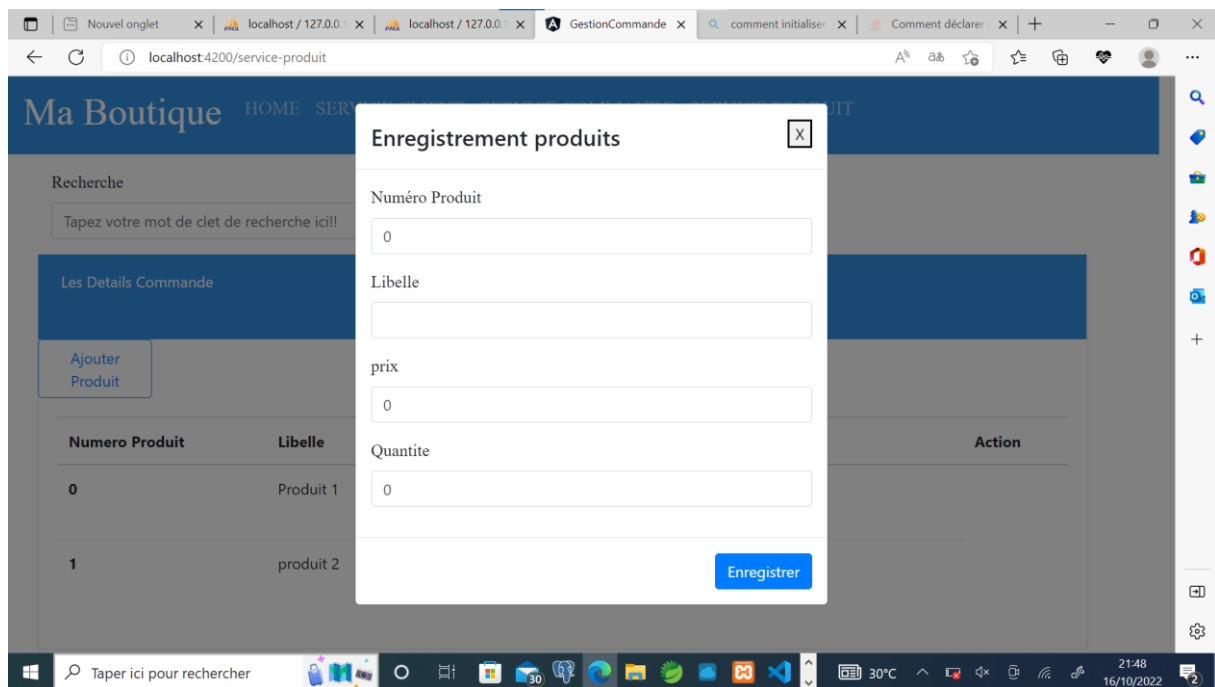


Figure 14 : Ajouter Un produit

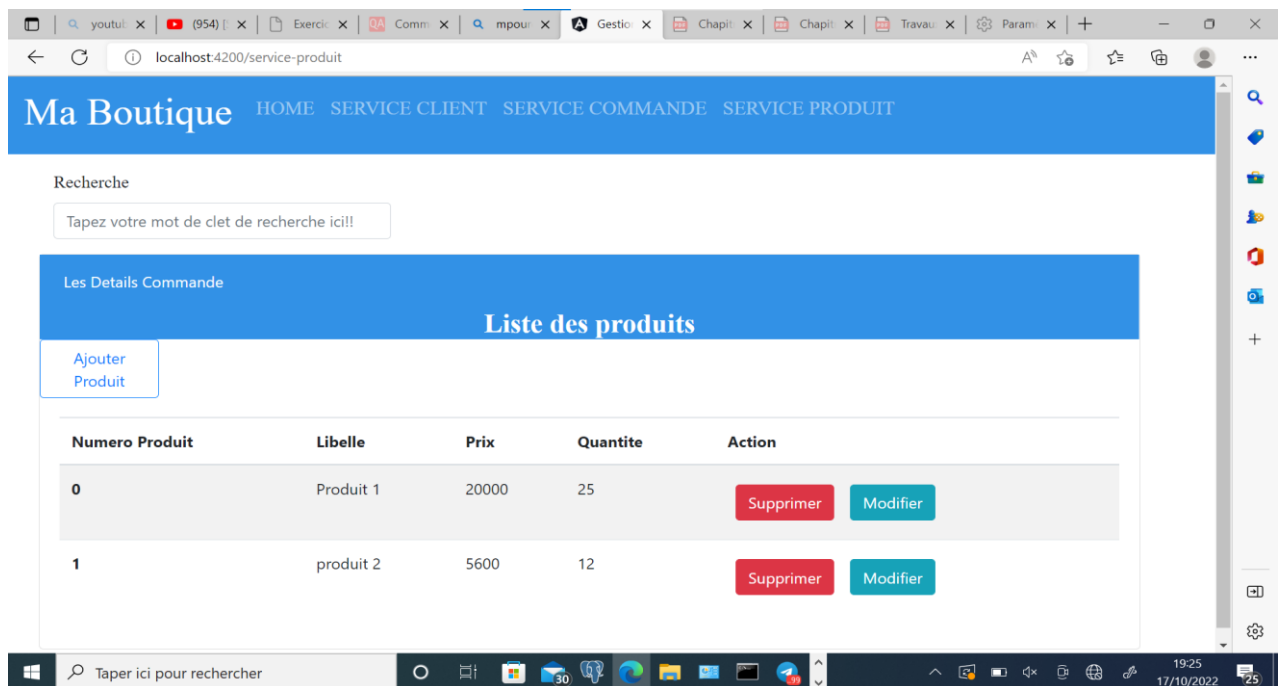


Figure 15 : Liste des Produits

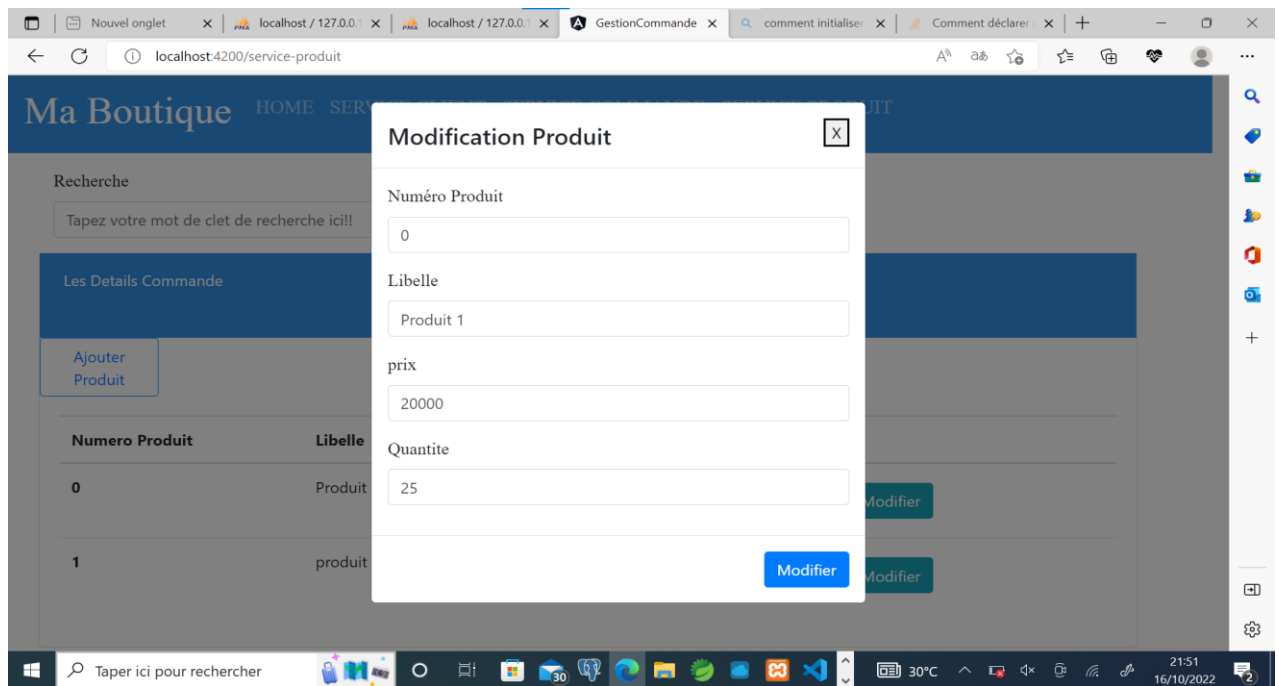


Figure 16 : Modifier Un Produit

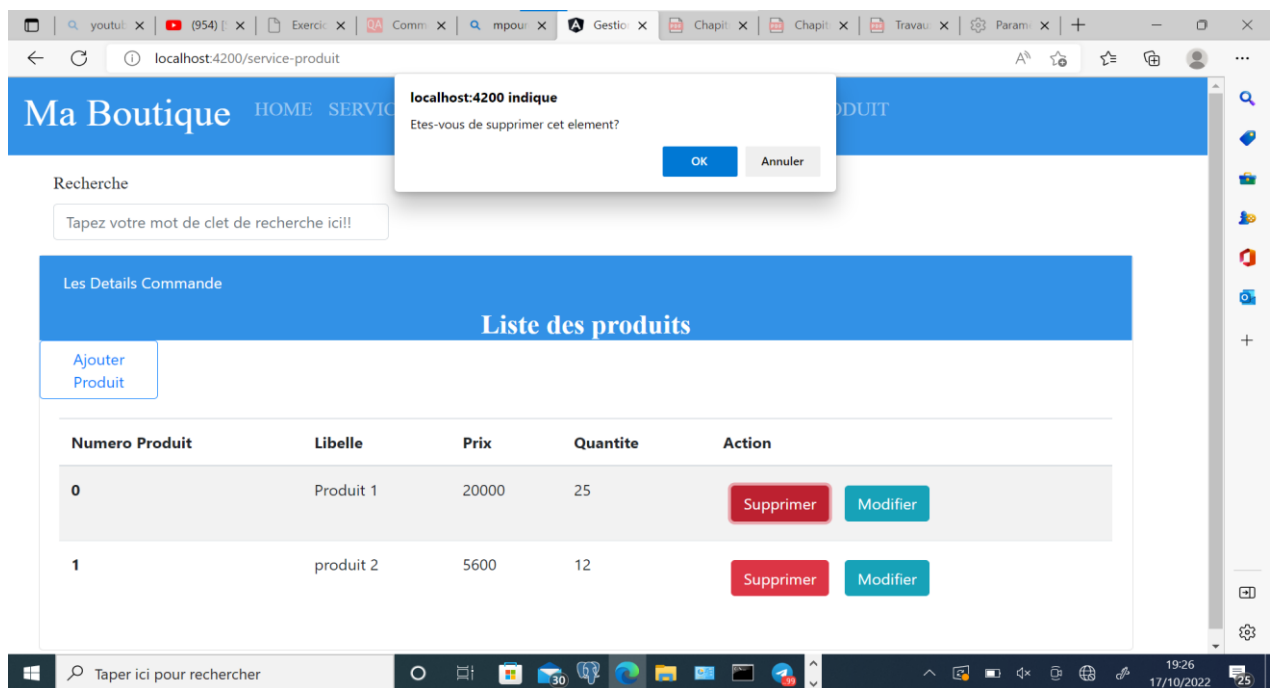


Figure 17 : Supprimer Un Produit

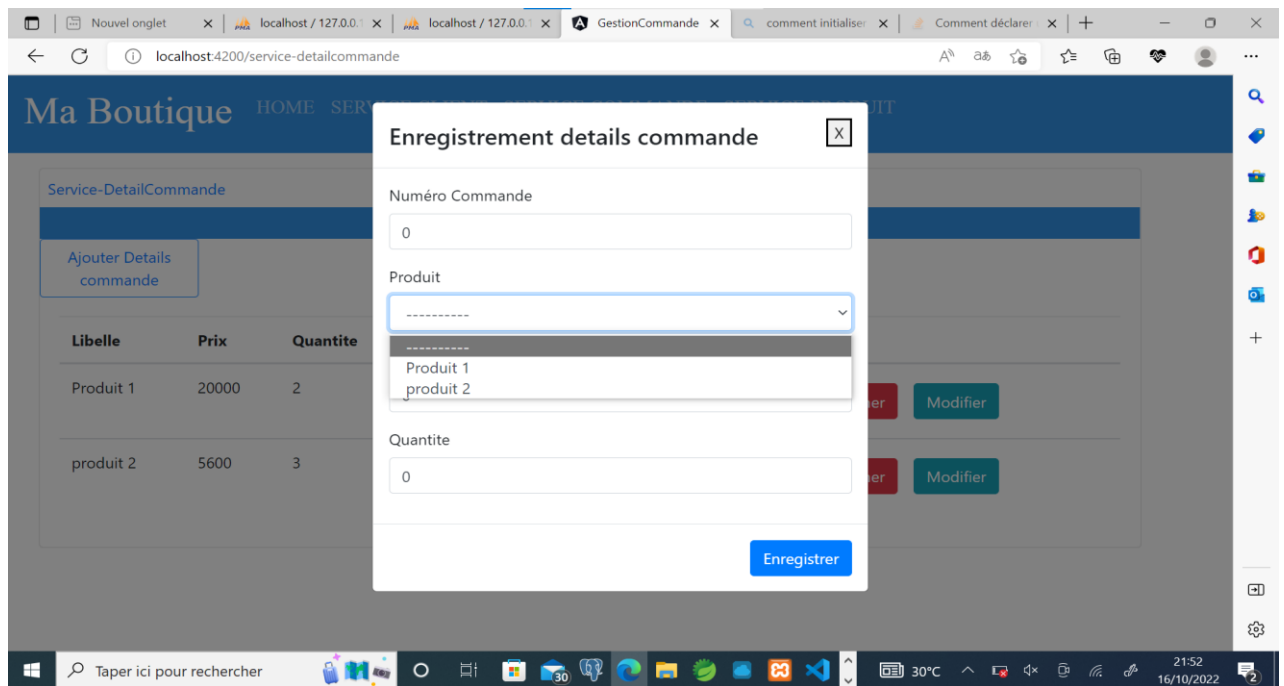


Figure 18 : Ajouter Details Commande

La figure 18 ci-dessus représente le formulaire d'ajout des détails commande, pour saisir les détails commande on entre le numéro de commande puis on sélectionne le produit qu'on a commandé parmi les produits existant dans la table produit puis on met la quantité, en suite le prix du produit, une fois les détails saisis on interroge le microservice commande pour vérifier l'existence du numéro de commande.

Si le numéro commande existe on enregistre les détails commande sinon un message d'erreur est signalé.

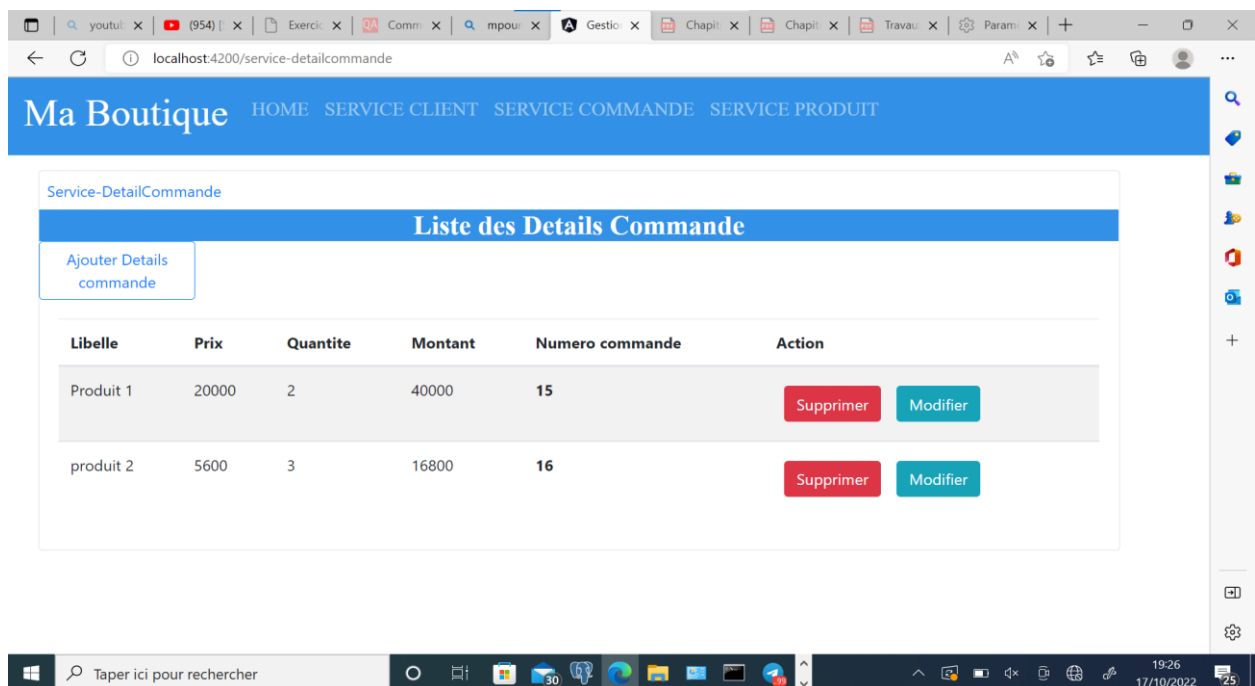


Figure 19 : Liste des Détails Commande

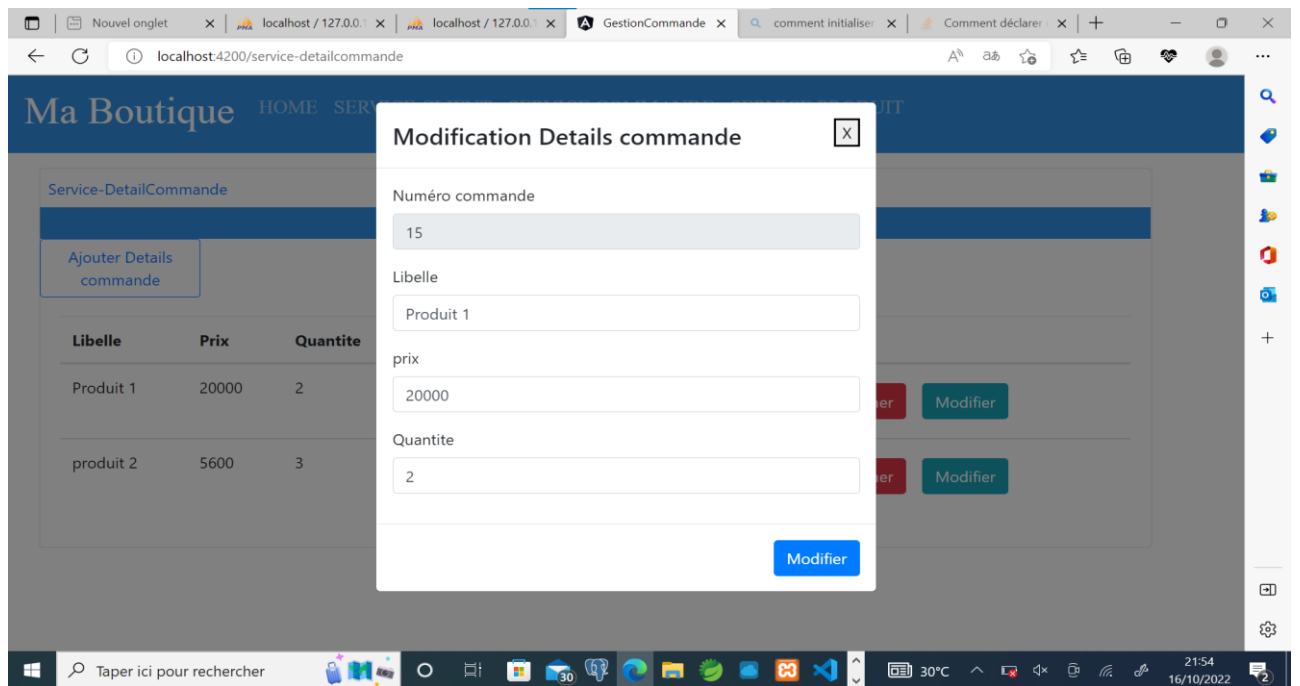


Figure 20 : Modifier Détails Commande

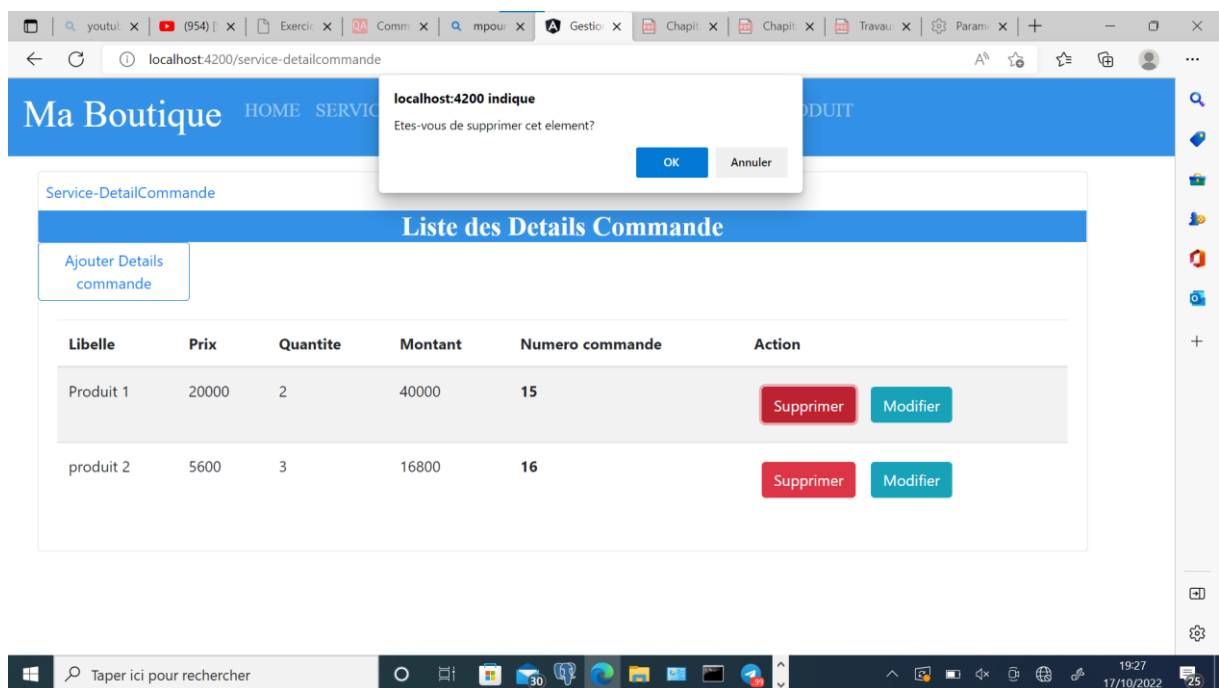


Figure 20 : Supprimer Détails Commande

Conclusion Générale

Ainsi dans le cadre de notre travail nous avons eu à réaliser une application de Gestion d'un site e-commerce composée de trois microservices. Chaque microservice est indépendant avec sa propre base de données. Les microservices communiquent entre eux au moyen de la méthode « Feign ».

Bien vrai que nous disposons d'applications fonctionnelles qui implémente les parties prioritaires à savoir la gestion d'un site e-commerce.

Nous pouvons dégager une perspective qui consiste à implémenter un outil permettant de traiter toutes les opérations d'un site e-commerce.