
MASTER : RESEAUX ET SYSTÈMES INFORMATIQUE

Gestion de Facturation

Réalisé par :

Mr. Hamza ELKADDARI

Mr. LAFRAOUZI Mouhssine

Encadré par :

Prof. MARZOUK Abderrahim

Contents

Abstrait.....	3
Introduction	4
Problématique	5
1. La méthode QQOCPQ	5
2. Problématique avec QQOCPQ.....	5
Solution	6
1. Cahier des charges.....	6
Gestion des clients.....	6
Gestion des produits.....	6
Gestion des fournisseurs	6
Gestion des commandes.....	6
Gestion des factures	6
Tableau de bord.....	7
Analyse et Conception	8
1. Introduction.....	8
2. Diagramme d'acteurs	8
3. Les exigences et les acteurs	9
4. Diagramme de cas d'utilisation.....	10
5. Diagramme de classes	11
Les langages Les outils	12
1. Conception	12
2. Collaboration.....	14
3. Développement	14
4. Système de gestion de base de données	16
5. Teste.....	Erreur ! Signet non défini.
Architecture du projet	17
Réalisation.....	18
Conclusion	25

Abstrait

Programme d'études Spring / Spring Boot
Nom de sujet Développement d'une Plateforme de Gestion des facturation
Superviseur MARZOUK ABDERRAHIM
<p>L'objectif de ce projet est de concevoir et développer une plateforme de gestion commerciale permettant à une entreprise de gérer efficacement ses clients, produits, fournisseurs, commandes, et factures. La solution offrira des fonctionnalités avancées pour automatiser les processus de gestion tout en assurant un suivi précis des opérations.</p> <p>Le projet sera développé en utilisant Java comme langage principal de programmation, Spring Boot pour la gestion du backend, Spring Data pour l'accès aux données, et MySQL comme système de gestion de base de données. La partie frontend reposera sur un framework web permettant une interface utilisateur fluide et intuitive.</p> <p>La plateforme assurera une gestion automatisée des opérations commerciales, incluant l'enregistrement et le suivi des commandes, la génération automatique des factures avec des références incrémentées et la gestion des paiements (en espèces ou par chèque). Elle permettra également de suivre l'état des factures et des stocks, d'associer des produits à plusieurs fournisseurs, et d'offrir un tableau de bord analytique avec des indicateurs clés tels que les produits en rupture, les factures impayées depuis un mois, la TVA collectée, et les produits les plus demandés.</p> <p>Le rapport couvrira une introduction sur la gestion commerciale, une analyse des besoins et fonctionnalités, une description des technologies utilisées (Spring Boot, Spring Data, MySQL), les étapes de conception et réalisation, ainsi que les résultats finaux avec captures d'écran.</p> <p>Ce projet vise à simplifier et automatiser les processus commerciaux tout en fournissant une solution robuste, personnalisable, et adaptée aux besoins spécifiques des entreprises.</p>

Introduction

La gestion commerciale est une discipline clé pour les entreprises, visant à organiser et coordonner les activités essentielles telles que la gestion des commandes, des factures, des produits, et des relations avec les fournisseurs. Les plateformes de gestion commerciale modernes facilitent ces processus en automatisant les tâches récurrentes et en optimisant la gestion des ressources grâce à des technologies avancées.

Dans le cadre de ce projet, l'utilisation de Spring Boot, Spring Data et MySQL permet de développer une solution robuste et performante, adaptée aux besoins spécifiques des entreprises. Ces technologies offrent une gestion efficace des opérations, notamment l'enregistrement et le suivi des commandes, la génération automatique de factures avec des références incrémentées, et le suivi des paiements.

Les fonctionnalités clés incluent la gestion des produits et fournisseurs, la gestion des états des factures et des commandes, ainsi qu'un tableau de bord analytique permettant de surveiller les produits en rupture de stock, les factures impayées depuis plus d'un mois, la TVA collectée, et les produits les plus demandés. Ces outils modernes améliorent non seulement l'efficacité opérationnelle mais également la prise de décision stratégique, offrant une solution complète pour répondre aux exigences des entreprises commerciales.

Ce projet vise non seulement à automatiser les tâches commerciales, mais également à offrir une expérience utilisateur fluide et intuitive. En intégrant des fonctionnalités avancées et en mettant l'accent sur la performance et la sécurité, cette plateforme répond aux défis actuels des entreprises tout en leur permettant de se concentrer sur leur croissance et leur compétitivité dans un environnement de plus en plus numérique.

Problématique

1. La méthode QQOCPQ

La méthode QQOCPQ est une approche analytique structurée utilisée pour comprendre et décrire une situation, un problème ou un processus en posant les questions suivantes : Quoi ? Qui ? Où ? Quand ? Comment ? Pourquoi ? Combien ? Ces questions guident une analyse approfondie en explorant différents aspects d'un sujet donné.

2. Problématique avec QQOCPQ

1. **Quoi ?**

La nécessité de gérer efficacement les opérations commerciales, y compris les commandes, les factures, les produits et les fournisseurs, tout en offrant un suivi précis des paiements et des stocks.

2. **Qui ?**

Les utilisateurs potentiels de cette plateforme sont les gestionnaires d'entreprise, les responsables des ventes, les comptables, et les employés en charge de la gestion des stocks et des fournisseurs.

3. **Où ?**

Dans un environnement commercial où les transactions et les opérations impliquent plusieurs acteurs, y compris les clients, les fournisseurs et les responsables internes, souvent répartis sur différents sites.

4. **Quand ?**

Tout au long du cycle de vie des opérations commerciales, depuis l'enregistrement des commandes jusqu'à la génération et le paiement des factures, avec des analyses périodiques pour surveiller les performances.

5. **Comment ?**

Grâce à l'automatisation des tâches répétitives, la génération de factures avec des références incrémentées, la gestion des paiements (espèces ou chèques), le suivi des états des commandes et des factures, ainsi que des outils d'analyse via un tableau de bord.

6. **Pourquoi ?**

Pour améliorer l'efficacité et la précision dans la gestion des opérations commerciales, réduire les erreurs humaines, optimiser les ressources, et offrir une visibilité claire sur les performances de l'entreprise.

Solution

1. Cahier des charges

Mettre en place un système de gestion pour une entreprise permettant d'organiser les clients, produits, fournisseurs, commandes, et factures. Ce système doit automatiser les tâches courantes, offrir une traçabilité et fournir des outils d'analyse via un tableau de bord.

Gestion des clients

- **Ajout et gestion des clients** : Création, modification et suppression des informations des clients (nom, adresse, contact, etc.).
- **Consultation des commandes et factures** associées à un client.

Gestion des produits

- Les produits sont de deux types : **articles** ou **services**.
- Un produit peut être fourni par plusieurs fournisseurs.
- **Gestion des stocks** : Mise à jour des quantités disponibles après chaque commande et Un alerte pour les produits en rupture de stock.
- **Informations produit** : Nom, description, prix unitaire, type (article ou service), et fournisseurs associés.

Gestion des fournisseurs

- Enregistrement et mise à jour des informations des fournisseurs.
- Consultation des produits fournis par un fournisseur.

Gestion des commandes

- Création et gestion des commandes :
 - Association des commandes à un client.
 - Sélection des produits ou services dans les commandes.
- État des commandes :
 - **Confirmée** : Commande validée et facturée.
 - **En attente de confirmation** : Commande créée mais non encore validée.

Gestion des factures

- **Génération automatique** des factures après la confirmation de commande.
- **Références continues** :
 - Numérotation automatique et incrémentale des factures.
 - Conservation de la liaison avec la facture originale en cas de modification.
- Une facture générée ne peut pas être supprimée.
- **Téléchargement des factures au format PDF**.

- **Paiement des factures :**
 - Modes de paiement acceptés :
 - **Espèce** : Montant maximum de 5000 DH.
 - **Chèque** : Nécessite les informations suivantes :
 - Intitulé de la banque.
 - Numéro du chèque.
 - Propriétaire du chèque.
- **Gestion de l'état de la facture :**
 - **Payée en totalité.**
 - **Partiellement payée** (affichage des montants réglés et restants).
 - **Non payée.**

Tableau de bord

- Indicateurs clés pour la gestion et le suivi :
 1. **Produits en rupture de stock.**
 2. **Factures impayées depuis plus d'un mois.**
 3. **Montant de la TVA collectée sur les trois derniers mois.**
 4. **Produits les plus demandés.**

Ce qui donne finalement les besoins :

➤ *Fonctionnels*

1. Gestion des clients.
2. Gestion des produits (articles et services).
3. Gestion des fournisseurs.
4. Gestion des commandes et de leur état.
5. Gestion des factures avec génération, modification et téléchargement en PDF.
6. Gestion des paiements des factures avec contrôle des modes de paiement.
7. Tableau de bord interactif pour le suivi des indicateurs clés.

➤ *Non fonctionnels*

1. **Disponibilité** : Application accessible en continu.
2. **Performance** : Temps de réponse rapide, y compris pour les opérations complexes (calculs TVA, génération PDF).
3. **Traçabilité** : Historique des modifications des factures et des commandes.
4. **Simplicité d'utilisation** : Interface intuitive, adaptée à tous les utilisateurs.
5. **Confidentialité** : Sécurisation des données des clients, commandes, et paiements.
6. **Modularité** : Possibilité d'étendre les fonctionnalités à l'avenir.

Analyse et Conception

1. Introduction

Cette première étape, nous permettra de délimiter le périmètre de notre projet, elle consiste à effectuer un premier repérage des besoins fonctionnels et non fonctionnels du système, ainsi que l'identification des acteurs qui interagissent avec le système et les messages échangés entre eux.

2. Diagramme d'acteurs

But : les acteurs qui gravitent autour de notre application.

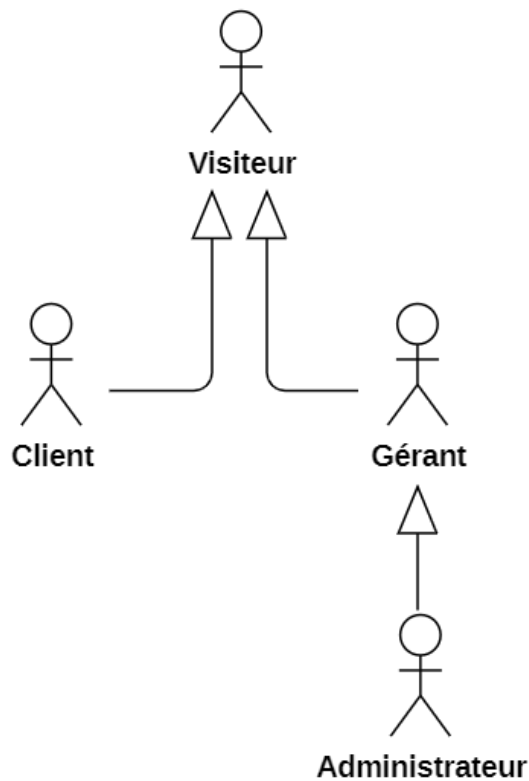


Figure 1 : Diagramme d'acteurs

Explication :

Nous avons 3 acteurs principaux qui interagissent directement avec le système : Invité, Gérant et Client.

3. Les exigences et les acteurs

Les exigences que nous allons découvrir sont les exigences fonctionnelles. A partir du problème posé, c'est à dire de tout ce qui est écrit, plus tout ce qui ne l'est pas, nous allons lister l'ensemble des fonctions qui pourront être réalisées par le logiciel. Ces exigences seront numérotées pour pouvoir les tracer dans les intentions d'acteur puis dans les uses cases.

1. Invité

Fonction	Module 00
S'inscrire	S'Authentifier
Se connecter	

Table 1 : les exigences vue invité

2. Client

Fonction	Module 00	Module 01	Module 02
Ajouter un produit au panier	Gérer son panier	-----	Gérer sa commande
Supprimer un produit du panier	Gérer son panier	-----	
Modifier la quantité d'un produit dans le panier	Gérer son panier	-----	
Confirmer le panier	Gérer son panier	-----	
Mentionner le mode de paiement (Espèces ou chèque)	Gérer son paiement	-----	
Ajouter l'intitulé de la banque	Gérer son paiement	-----	
Modifier le mode de paiement	Gérer son paiement	Manipuler sa facture	
Modifier l'intitulé de la banque	Gérer son paiement		
Télécharger la facture	-----		
Modifier son profile	Gérer son profile	-----	

Table 2 : les exigences vue client

3. Gérant

Fonction	Module 00	Module 01
Confirmer la commande	Gérer les commandes	-----
Refuser la commande		
Ajouter un produit	Gérer les produits	
Modifier un produit		
Supprimer un produit		
Ajouter et valider la livraison de fournisseur	Gérer le stock	
Avancer l'état des factures	Gérer les factures	

Table 3 : les exigences vue gérant

4. Administrateur

Fonction	Module 00	Module 01
Ajouter Un gérant	Gérer les gérants	-----
Supprimer un gérant		
Modifier les informations d'un gérant		

Table 3 : les exigences vue administrateur

4. Diagramme de cas d'utilisation

1. Visiteur

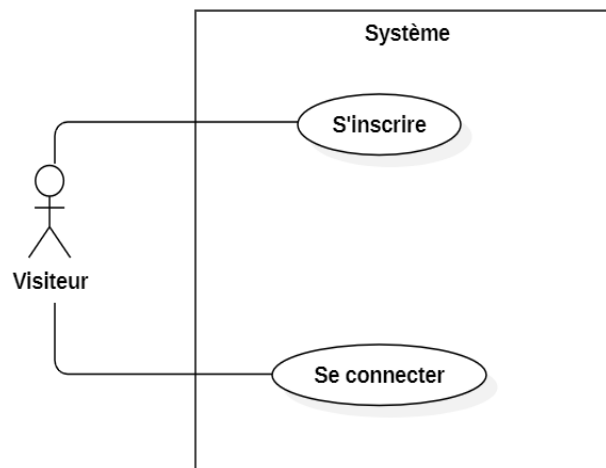


Figure 2 : Diagramme de cas d'utilisation vue visiteur

2. Gérant

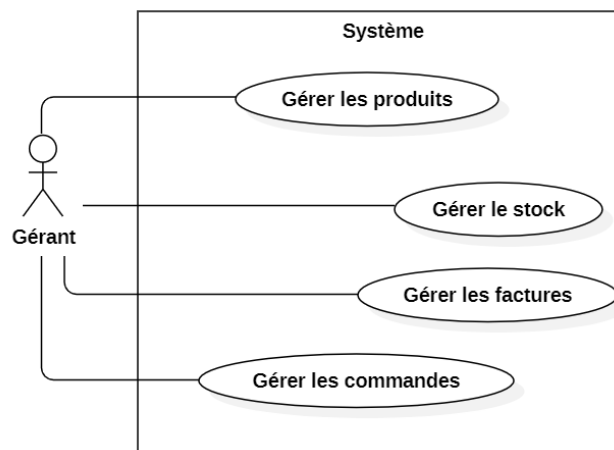


Figure 3 : Diagramme de cas d'utilisation vue gérant

3. Client

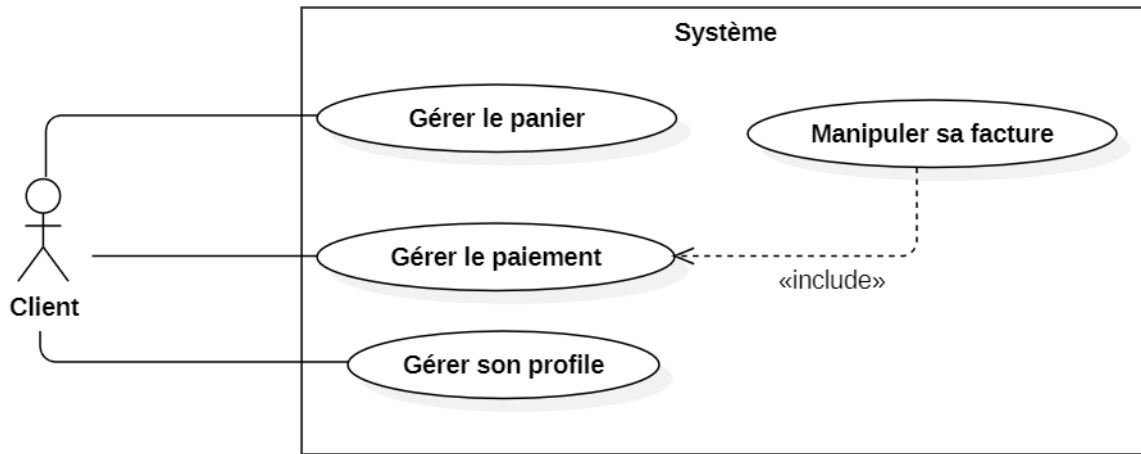


Figure 4 : Diagramme de cas d'utilisation vue client

4. Administrateur

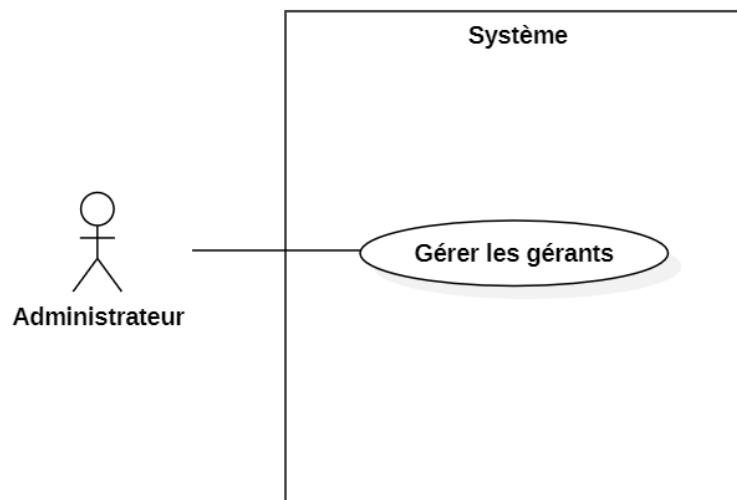


Figure 5 : Diagramme de cas d'utilisation vue Administrateur

5. Diagramme de classes

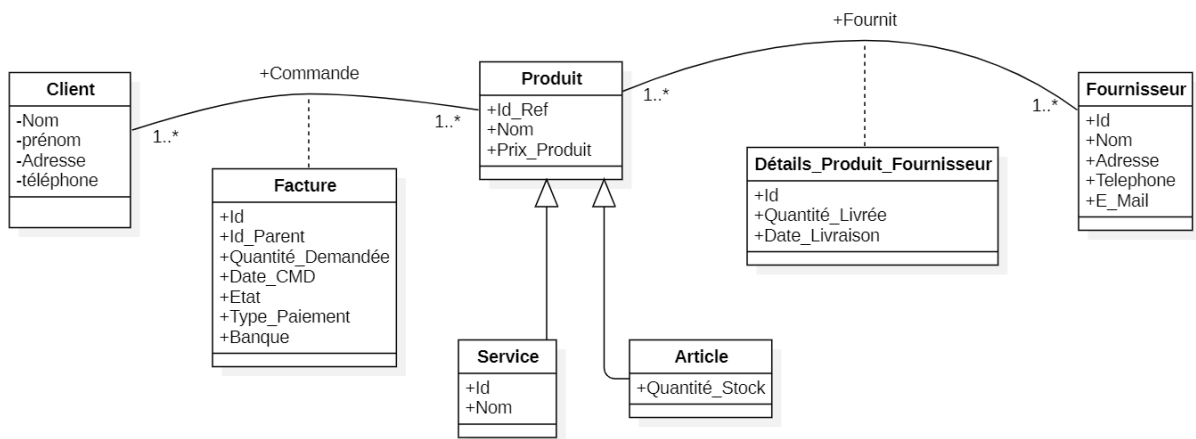


Figure 24 : Diagramme de classes du système

Les langages Les outils

1. Conception

1.1 Modélisation

Langage UML

UML, ou Unified Modeling Language (Langage de Modélisation Unifié en français), est un langage de modélisation standard utilisé dans le domaine du génie logiciel pour visualiser, spécifier, construire et documenter les systèmes logiciels. UML fournit un ensemble de notations graphiques et de règles de modélisation normalisées, ce qui permet aux développeurs et aux concepteurs de communiquer efficacement sur la conception et la structure d'un système logiciel.



Figure 25 : Logo UML

Outil StarUML

StarUML est un outil de modélisation UML (Unified Modeling Language) qui permet aux développeurs et aux concepteurs de créer des diagrammes UML pour représenter visuellement la conception de logiciels. Il offre une interface graphique conviviale pour créer différents types de diagrammes UML, tels que des diagrammes de classes, de séquence, d'activité, de cas d'utilisation ...



Figure 26 : Logo StarUML

1.2 Graphique

FIGMA

Figma est une plateforme de conception d'interface utilisateur (UI) et d'expérience utilisateur (UX) basée sur le cloud. C'est un outil de conception collaboratif qui permet aux équipes de travailler ensemble en temps réel sur des projets de conception, peu importe leur emplacement géographique.

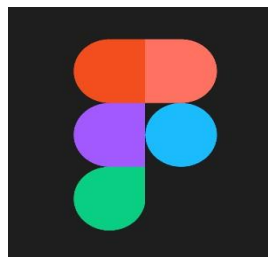


Figure 27 : Logo FIGMA

2. Collaboration

1. Github

GitHub est une plateforme de développement collaboratif basée sur le système de gestion de versions Git. Elle offre un ensemble de services et d'outils qui facilitent la collaboration entre les développeurs, le suivi des versions des projets logiciels, la gestion du code source, et la coordination des efforts au sein des équipes de développement.



Figure 28 : Logo GitHub

3. Développement

3.1 Front-end

Tailwindcss

Tailwind CSS est un framework CSS basé sur des classes utilitaires pré-définies qui facilitent le développement et la conception d'interfaces utilisateur. Contrairement à d'autres frameworks CSS tels que Bootstrap ou Foundation, Tailwind n'impose pas de styles prédéfinis. Au lieu de cela, il fournit un ensemble de classes basées sur des utilitaires qui peuvent être directement appliquées dans le code HTML pour styliser les éléments.



Figure 29 : Logo Tailwindcss

Thymeleaf

Thymeleaf est un moteur de template Java open source utilisé pour construire des interfaces utilisateur dynamiques. Conçu pour s'intégrer facilement aux applications Spring, il permet de générer des pages HTML enrichies en données. Thymeleaf est particulièrement adapté au développement d'applications web côté serveur et offre une syntaxe intuitive qui permet de lier efficacement les données de l'application à l'interface utilisateur



Figure 30 : Logo Thymeleaf

3.2 Back-end

Spring boot

Spring Boot, complément essentiel du framework Spring, simplifie le développement d'applications Java en fournissant une autoconfiguration et un démarrage rapide. Il intègre des fonctionnalités clés telles que l'inversion de contrôle, la persistance des données via Spring Data, et la sécurité avec Spring Security. Ainsi, les développeurs peuvent créer rapidement des applications robustes et sécurisées sans nécessiter une configuration manuelle complexe.



Figure 31 : Logo Spring boot

4. Système de gestion de base de données

PostgreSQL

Est un système de gestion de base de données relationnelle (SGBDR) open source, puissant et extensible. Il est souvent abrégé en "Postgres". Contrairement à d'autres systèmes de gestion de base de données, PostgreSQL est réputé pour sa conformité aux standards SQL, sa robustesse, sa capacité à gérer des charges de travail complexes, et sa prise en charge d'extensions et de fonctions avancées.

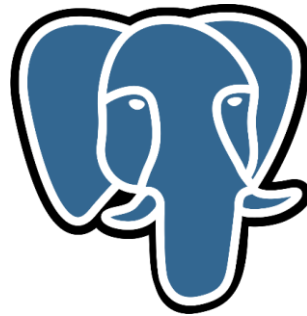


Figure 32 : Logo PostgreSQL

Architecture du projet

L'architecture de notre projet est basée sur le modèle MVC (Model View Controller) pour séparer les préoccupations de présentation, de logique métier et de données.

L'utilisateur interagit avec l'application via une interface utilisateur thymeleaf.

Le contrôleur est responsable de la distribution des requêtes aux services appropriés. Les services sont écrits en Spring Boot et utilisent le framework JPA pour accéder aux données.

Les services sont protégés par la sécurité Spring Security.

Spring MVC Web Application

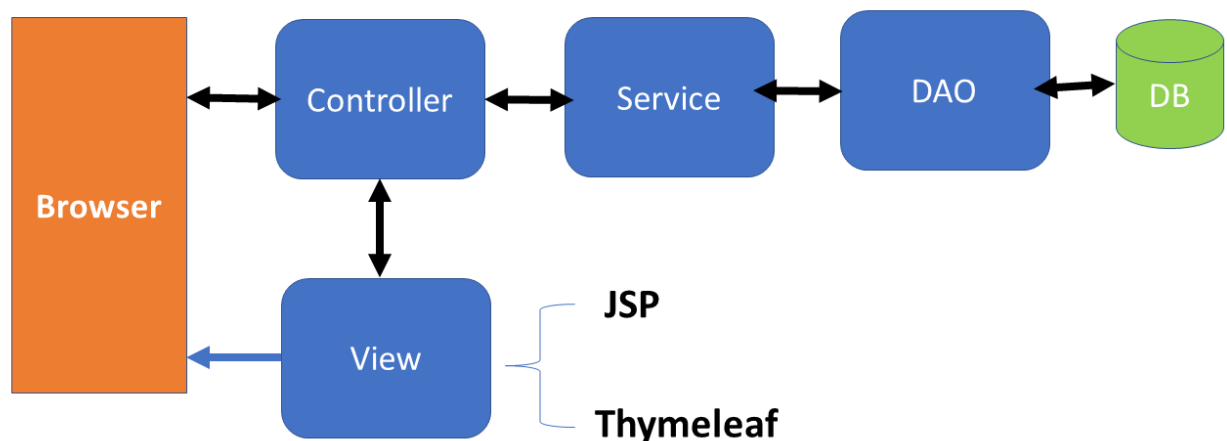


Figure 34 : architecture du projet

L'architecture présente les avantages suivants :

Réutilisabilité : Les services sont des composants indépendants qui peuvent être réutilisés dans différentes applications.

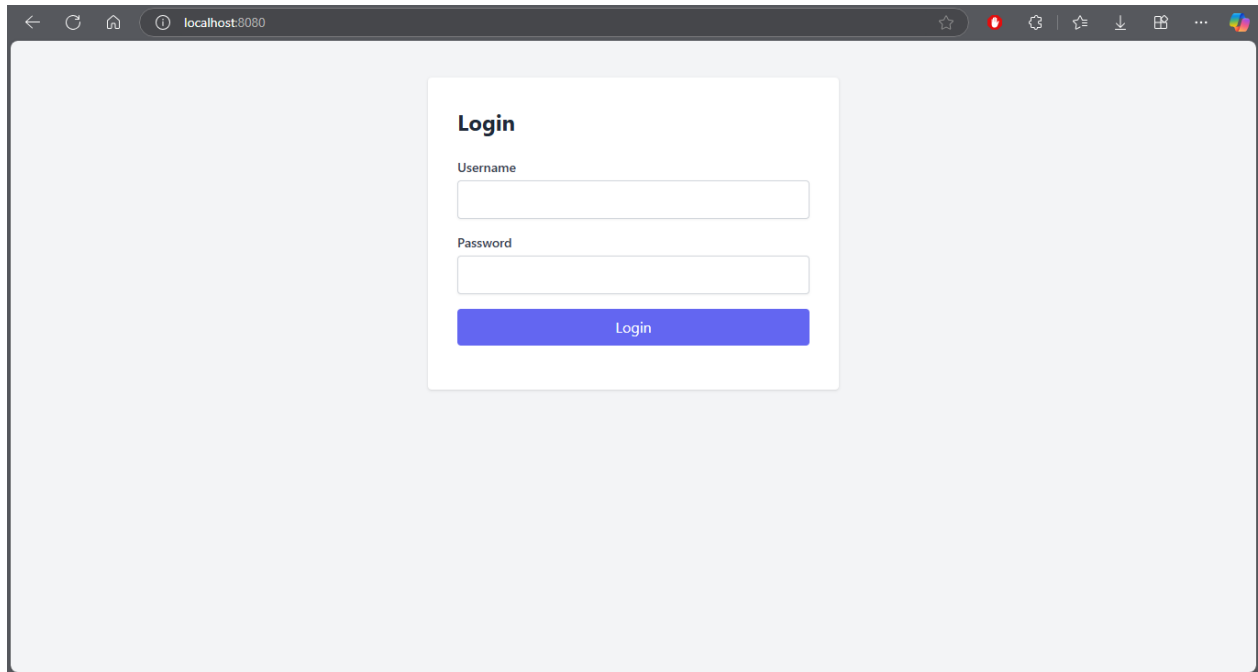
Scalabilité : Les services peuvent être mis à l'échelle de manière indépendante, ce qui permet à l'application de gérer des charges de travail élevées.

Flexibilité : L'architecture est flexible et peut être adaptée aux besoins spécifiques de l'application.

Réalisation

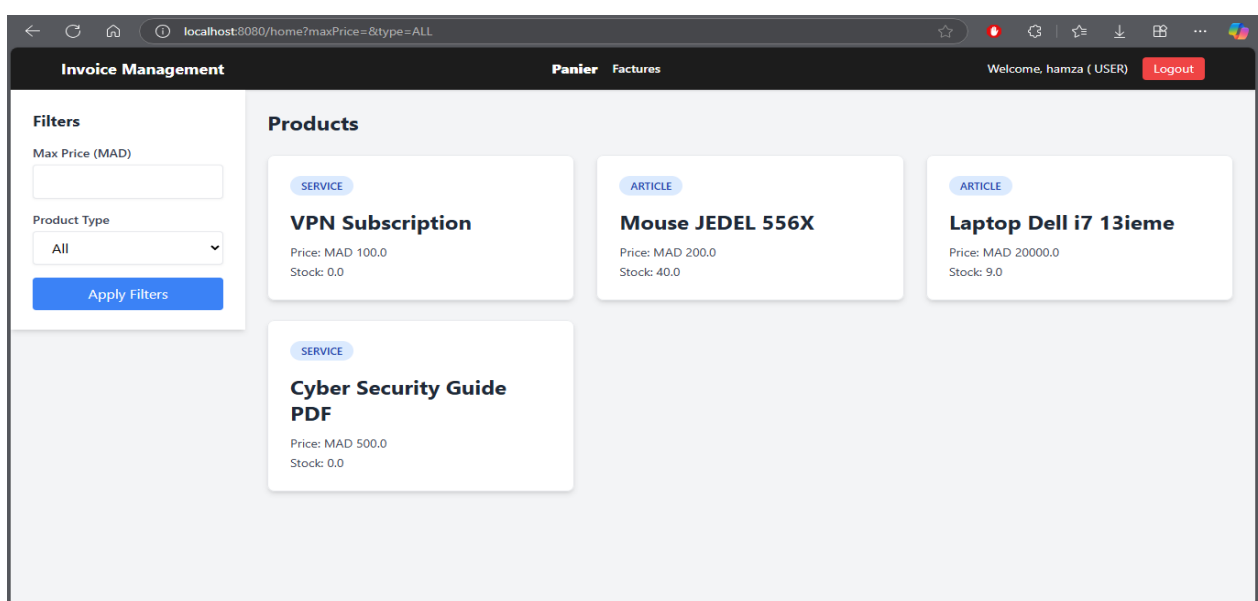
Au sein de cette section dédiée, nous vous proposerons une présentation les différents pages de notre plateforme. Explorez-en davantage pour approfondir votre compréhension et découvrir les multiples facettes de notre service.

- **Login**



The screenshot shows a web browser window with the address bar displaying 'localhost:8080'. The main content area features a centered white box with the title 'Login'. Inside this box, there are two input fields: 'Username' and 'Password'. Below these fields is a blue button labeled 'Login'.

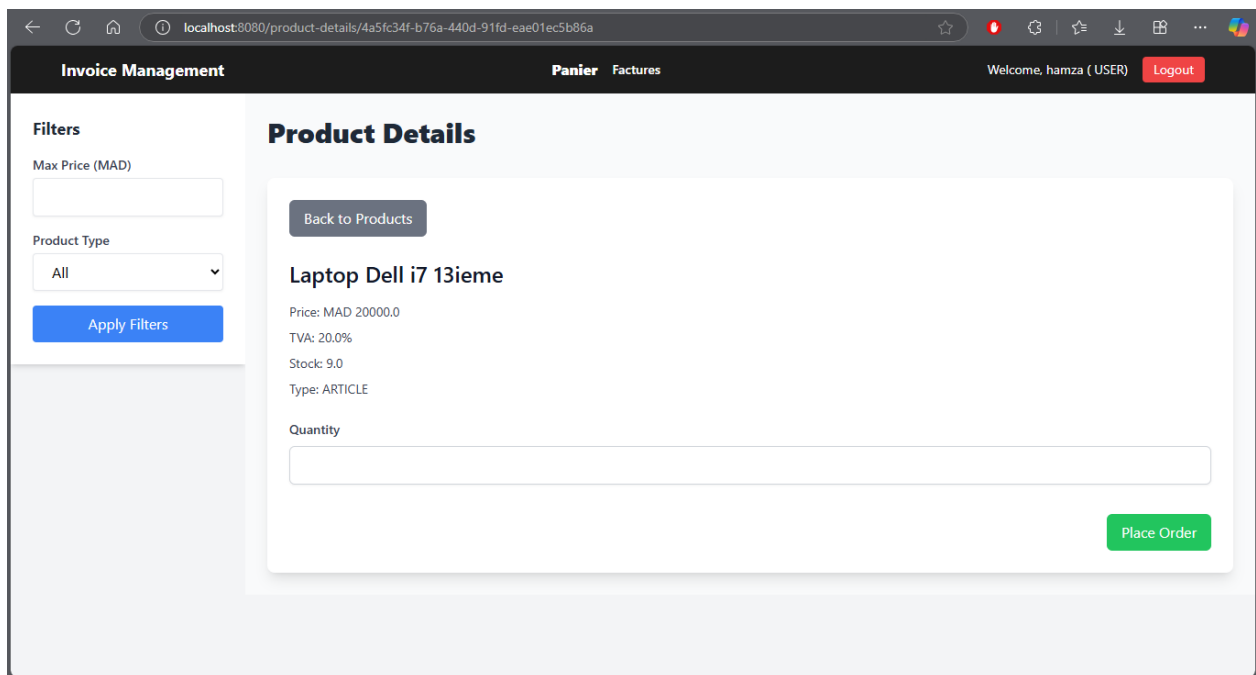
Authentification de simple user nommé hamza



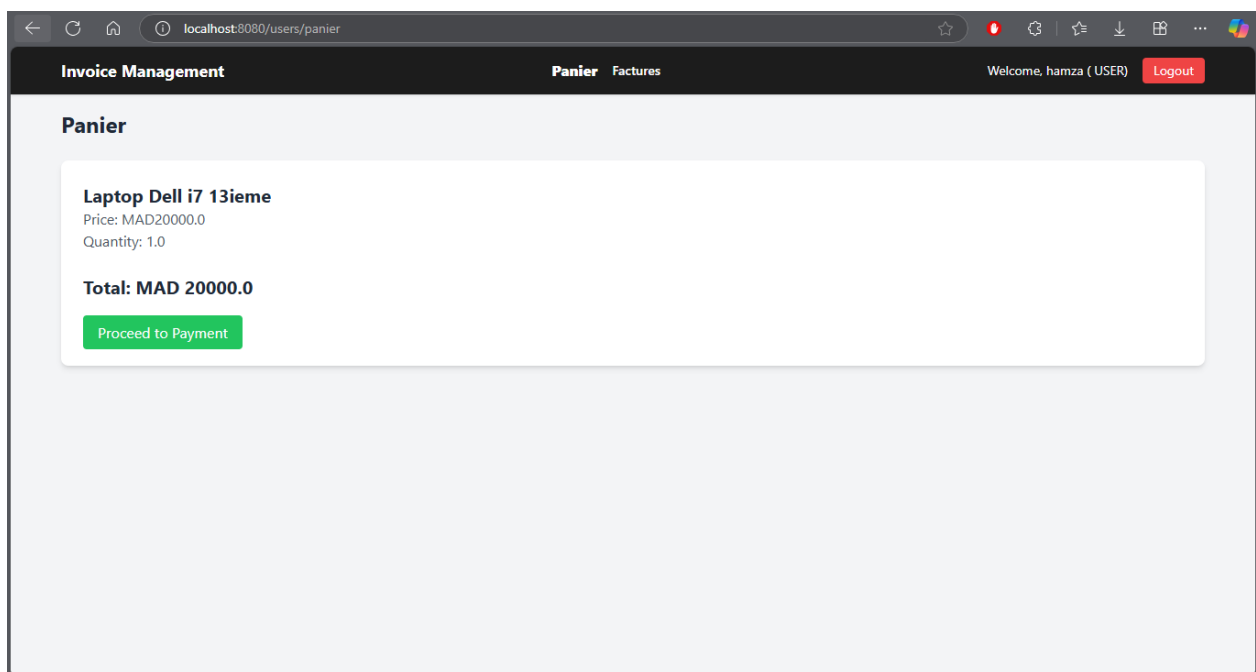
The screenshot shows a web application dashboard. The top navigation bar includes 'Invoice Management', 'Panier', and 'Factures'. The user is logged in as 'hamza (USER)' with a 'Logout' button. The main content area is divided into a 'Filters' sidebar and a 'Products' section. The 'Filters' sidebar has a 'Max Price (MAD)' input field and a 'Product Type' dropdown menu set to 'All'. The 'Products' section displays four product cards:

- VPN Subscription** (SERVICE): Price: MAD 100.0, Stock: 0.0
- Mouse JEDEL 556X** (ARTICLE): Price: MAD 200.0, Stock: 40.0
- Laptop Dell i7 13ieme** (ARTICLE): Price: MAD 20000.0, Stock: 9.0
- Cyber Security Guide PDF** (SERVICE): Price: MAD 500.0, Stock: 0.0

- Selection d'un produit pour avoir le détails (ex : Laptop Dell i7 13ieme)

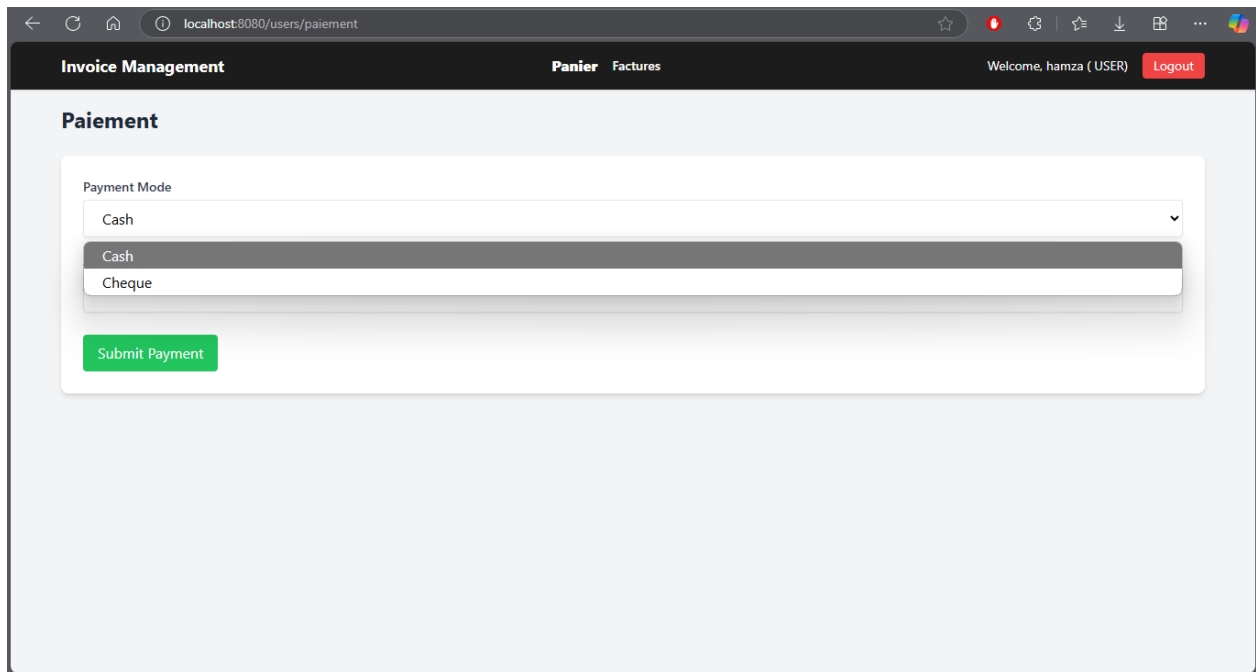


- Ajout au panier



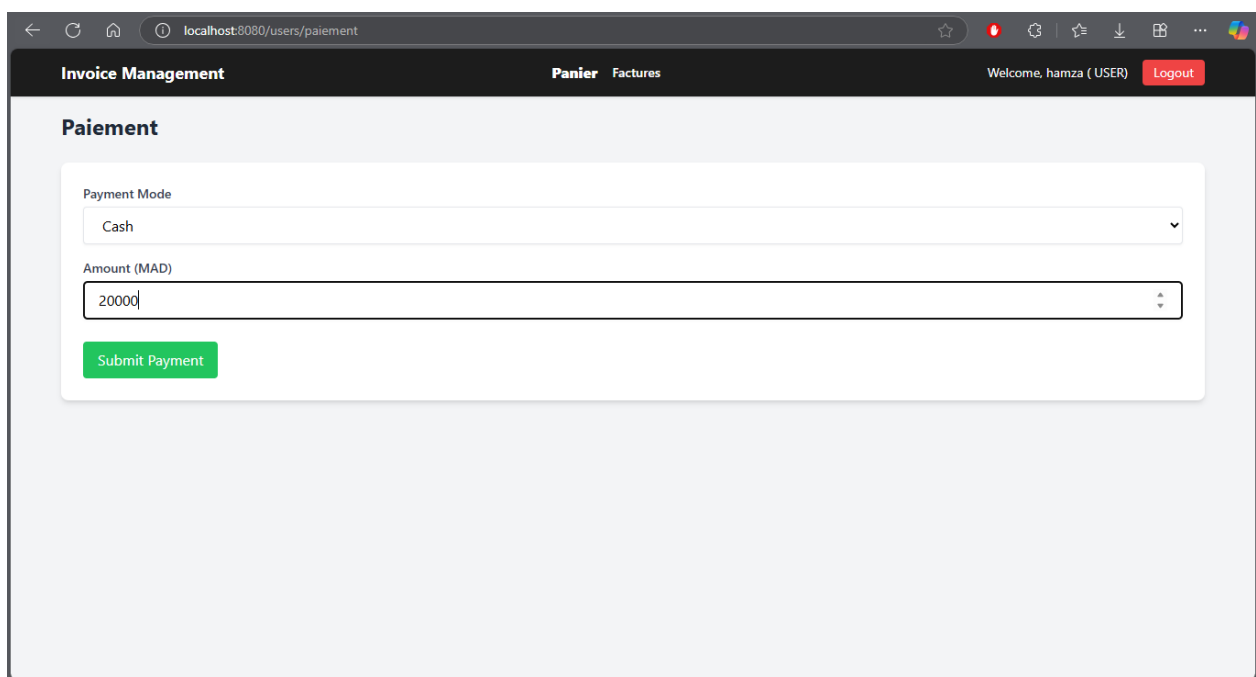
- **Paielement pour demander la commande**

1- Choix de mode de règlement :



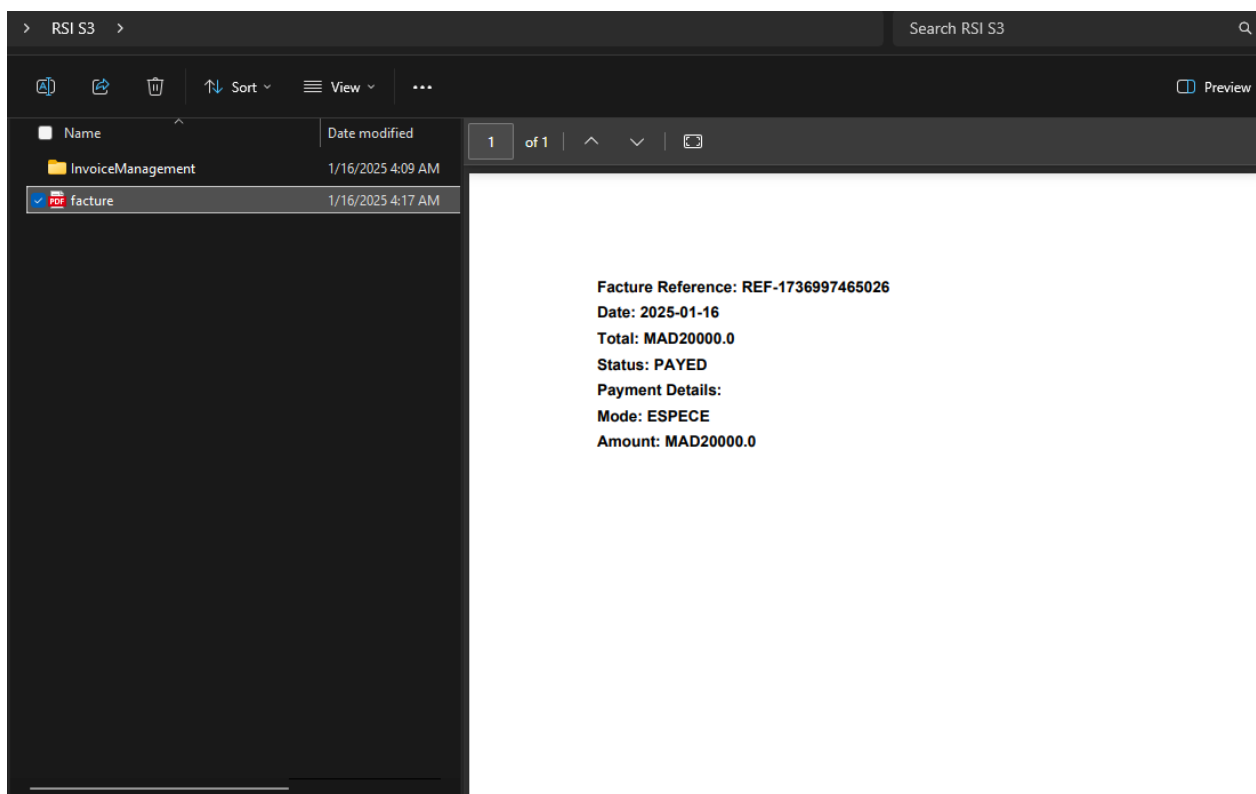
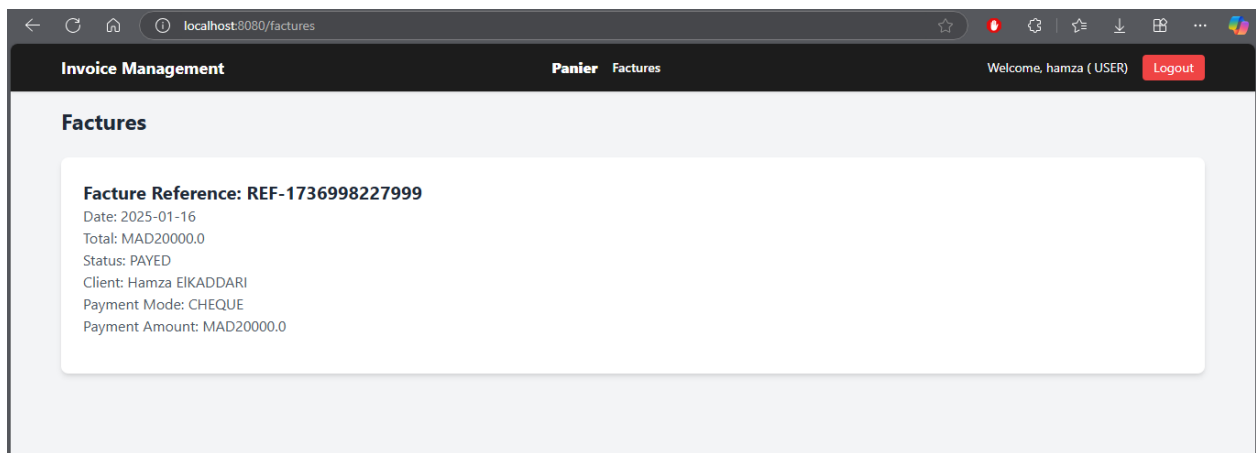
The screenshot shows a web browser window with the URL `localhost:8080/users/paiement`. The page has a dark header with 'Invoice Management' on the left, 'Panier Factures' in the center, and 'Welcome, hamza (USER) Logout' on the right. The main content area is titled 'Paielement'. It contains a form with a 'Payment Mode' dropdown menu. The dropdown is open, showing 'Cash' and 'Cheque' as options. Below the dropdown is a green 'Submit Payment' button.

2- Paiement :



The screenshot shows the same web browser window as before. The 'Payment Mode' dropdown is now closed and set to 'Cash'. Below it, the 'Amount (MAD)' input field is visible, containing the text '20000'. The green 'Submit Payment' button remains at the bottom of the form.

- **Creation d'une commande qui est en attente avec la generation d'une facutre pdf :**



- **Admin :**

Add user → automatiquement add client

Invoice Management **Dashboard** Users Clients Produits Factures Welcome, admin (ADMIN) Logout

Liste des Utilisateurs

[+ Ajouter](#)

ID	ID_CLIENT	NOM D'UTILISATEUR	RÔLE	ACTIONS
1		admin	ADMIN	
2	c26716d3-18fb-4d7c-bb64-a82aca768b82	hamza	USER	
3	47eb35a3-9e90-4680-b3aa-78a8fa8bb06e	mouhssine	USER	

Dashboard Users Clients Produits Factures Welcome, admin (ADMIN) Logout

User Register

Username

Email

Password

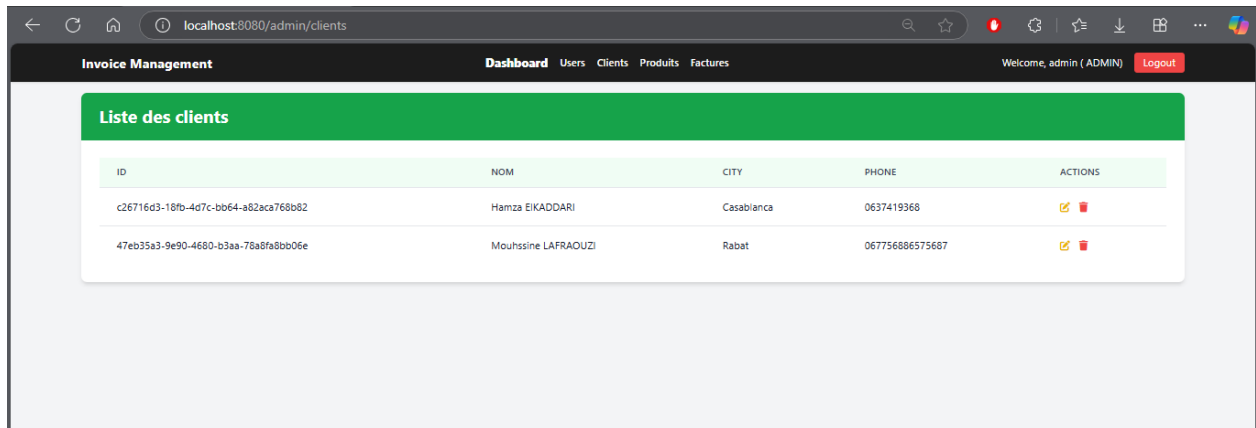
Role
 User ▼

Name

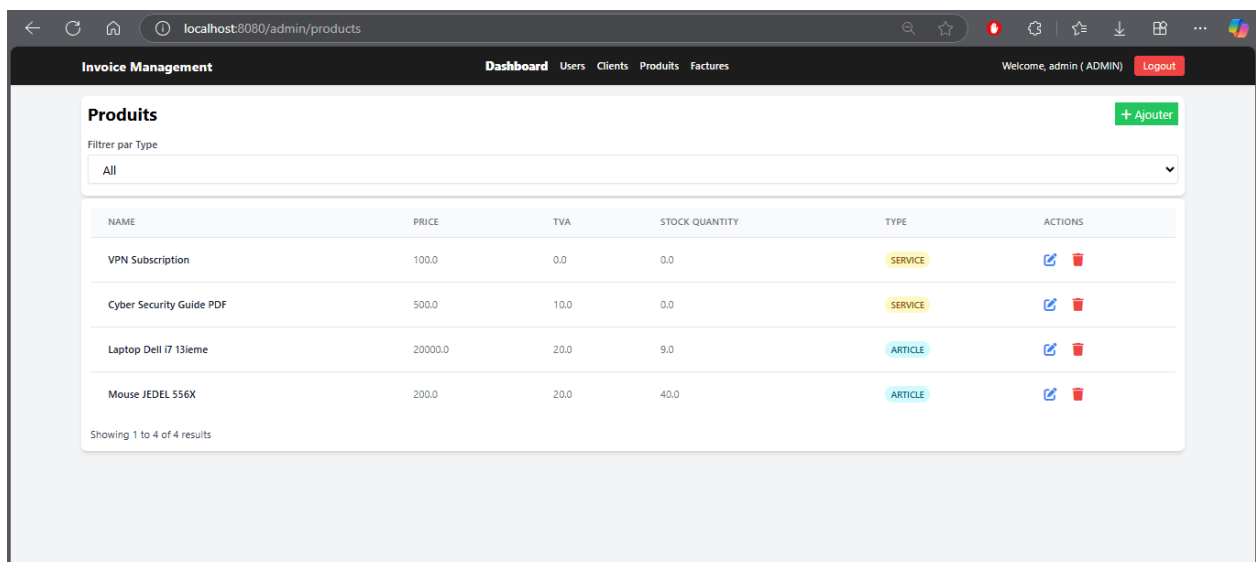
City

Phone

[Register](#)



- **Gestion des produits :**



The screenshot shows a web browser at the URL `localhost:8080/admin/products/new`. The application header includes 'Invoice Management', a navigation menu with 'Dashboard', 'Users', 'Clients', 'Produits', and 'Factures', and a user greeting 'Welcome, admin (ADMIN)' with a 'Logout' button. The main content area features a modal titled 'Add Product' with the following fields: 'Name' (text input with placeholder 'Enter product name'), 'Price' (text input with value '0.0'), 'TVA' (text input with value '0.0'), 'Stock Quantity' (text input with value '0.0'), and 'Type' (dropdown menu currently showing 'SERVICE'). A 'Save' button is located at the bottom right of the modal.

- Les factures :

The screenshot shows the 'Factures' page in the application. The header is identical to the previous screenshot. The main content area is titled 'Factures' and contains a white box with the following information: 'Facture Reference: REF-1736998227999', 'Date: 2025-01-16', 'Total: MAD20000.0', 'Status: PAYED', 'Client: Hamza EIKADDARI', 'Payment Mode: CHEQUE', and 'Payment Amount: MAD20000.0'.

Conclusion

Le développement d'une plateforme de gestion de la facturation constitue un levier essentiel pour améliorer l'efficacité financière et la transparence au sein d'une organisation. En intégrant des fonctionnalités telles que l'automatisation des processus de facturation, le suivi en temps réel des paiements et la gestion des comptes clients, cette solution permet de réduire les erreurs, d'accélérer les délais de traitement et de renforcer la satisfaction des clients. En outre, elle offre une meilleure visibilité sur les flux financiers, facilitant ainsi la prise de décisions stratégiques. Investir dans une telle plateforme permet aux entreprises de rationaliser leurs processus, d'améliorer leur rentabilité et de se concentrer davantage sur leurs objectifs stratégiques.