

Dédicaces

Avec l'expression de mon reconnaissance, je tiens à dédier ce modeste projet
à :

A mes chers parents

Autant de phrases aussi expressives soient-elles ne sauraient montrer le degré
d'amour et d'affection que je éprouve pour eux.
Merci pour tout l'amour et la confiance et pour votre énorme support pendant la
réalisation de mon projet.

A mes chers amis

Spécialement Monaime, Hamza et Youssef, qui n'ont pas cessés de me conseiller,
encourager et soutenir tout au long ce projet.

A vous cher lecteur

Remerciement

Après avoir rendu ce rapport grâce à Dieu le Tout puissant et le Miséricordieux, je tiens à remercier vivement tous ceux qui de près ou de loin ont participé à la réalisation de ce projet de fin d'étude.

Je tiens à exprimer ma profonde gratitude à mon cher professeur et encadrant **Mr. MED.CHAOUKI.ABOUNAIMA**, enseignant à la Faculté des Sciences et Techniques Fès, pour sa disponibilité, son sens d'échange et d'écoute et pour son soutien que me a prodigué tout au long la période de projet.

J'adresse aussi mes vifs remerciements à mon encadrant au sein de l'agence de Marketing Digital(Webmarko), **Mr. Mohammed Rachidi**, pour le temps qu'il a consacré pendant la période de stage, pour son aide inestimable, pour toutes les informations précieuses qui ont mené à bien nourrir ce travail et à me accorder une expérience enrichissante.

Je ne laisserai pas passer cette occasion sans remercier tous nos professeurs de la Faculté des Sciences et Techniques qui ont contribué à notre formation pendant cette année.

Finalement, j'exprime mes remerciements et mes respects aux membres de jury, pour l'intérêt qu'ils ont apportés à mon projet de fin d'étude en acceptant d'examiner notre travail, et l'enrichir par leurs remarques.

Résumé

Dans le cadre de la bonne gestion du Service Informatique et au sein de **Agence de Marketing digital**, mon projet de fin d'étude réalisé pendant cette période de stage, consiste à développer une application web permettant la gestion des stocks pour un inventaire.

Cette application permet de faciliter le travail pour un inventaire qui se définit comme une opération comprenant : le comptage des produits d'une entreprise, puis leur valorisation à la clôture de l'exercice comptable .

En vue de structurer le travail et d'appliquer les acquis durant le cursus universitaire, J'ai tout d'abord élaboré une étude conceptuelle et technique de l'application avant de passer à son implémentation et son développement.

Le stage était une bonne occasion pour découvrir de nouvelles technologies. Principalement, La technologie JEE en utilisant le Framework « **Spring Boot** » pour le Backend, et le **SQL** pour la gestion de base de données.

Pour le Frontend, J'ai adopté plusieurs technologies : **HTML**, **CSS**, **JAVASCRIPT**, **BOOTSRAPE** et **JQUERY** pour donner à notre application un thème responsive bien stylisé.

Abstract

As part of the proper management of the IT Department and within the Digital Marketing Agency, my end-of-study project carried out during this internship period consists in developing a web application allowing the management of stocks for an inventory.

This application facilitates the work for an inventory which is defined as an operation comprising: the counting of a company's products, then their valuation at the end of the accounting year.

In order to structure the work and apply the acquired knowledge during the university courses, we first start with a conceptual and technical study of the application before moving on to its implementation and development.

The internship was a good opportunity to discover new technologies. Mainly, **JEE** technology using the Framework « **Spring Boot** » for the Backend, and **SQL** for database management.

For the Frontend, we have adopted several technologies : **HTML**, **CSS**, **JAVASCRIPT**, **BOOTSRAAP** and **JQUERY** to give our application a well styled responsive theme.

Table des matières

Introduction	12
Chapitre 1 : Contexte Général du Projet.....	13
1-Presentation de lieu de stage.....	14
1 .1.1 L’agence WebMarko.....	14
1.1.2 Ses Activités	14
2-Problématique	15
3. Description générale de la solution proposée	15
4-Description du cahier des charges.....	16
Chapitre 2 : Analyse et Conception	18
1- Analyse des besoins fonctionnels et techniques.....	19
1 .1. Outils d’analyse et de conception.....	19
1.1.1. Langage UML	19
1.1.2. modèle en cascade	19
1.1.3. Modèle MVC.....	20
1.2 – Les besoins fonctionnels.....	21
1-3. Les besoins non fonctionnels	21
1.4 architecture JEE.....	22
2. Conception	24
2 .1. Acteur identifié	24
2.2. Les diagrammes de cas d’utilisation	24
2.3. Analyse des cas d’utilisation et diagramme de séquence.....	25
2.3.1 Authentification	26
2.3.2 L’ajout d’un nouveau client.....	28
2.3.3 L’ajout d’un nouveau produit.....	30
2.4 Diagramme de classes	32
Chapitre 3 : Réalisation de L’Application	33



1 . Besoins techniques.....	34
2-Authentification.....	39
3-Table de bords	40
4-Gestion des clients	40
4 .1 Visualisation des magasins des Clients	40
4.2 Ajout de magasin de client	41
4.3 modification de magasin de client.....	41
4.4 Suppression de magasin de Client	42
5- Gestion des Produits	43
5-1 Visualisation	43
5-2 Ajout d'un produit	44
5-3 modification.....	44
5-4 Suppression d'un Produit	45
6-Gestion des inventaires.....	46
6-1 Visualisation	46
6-2 imprimer inventaire en pdf	47
7-profile	48
Conclusion et perspectives.....	49
Webographie	50

Liste des tables

Tableau 1 : tableau des acronymes	11
Tableau 2: Scénario de l'authentification	26
Tableau 3 : scénario d'un nouveau client	28
Tableau 4: scénario de l'ajout d'un produit	30

Liste des Figures

Figure 1 : Diagramme de Gant.....	17
Figure 2 : Modèle en Cascade	20
Figure 3:Modele MVC	21
Figure 4 : Achitecture de JEE	23
Figure 5 : Architecture de SpringBoot	24
Figure 6: Diagramme de cas d'utilisation.....	25
Figure 7 :digramme de séquence pour l'authentification	27
Figure 8 : digramme de séquence ajouter client.....	29
Figure 9 ; digramme de séquence ajouter produit.....	31
Figure 10: diagramme de classes.....	32
Figure 11 : : logo de JEE.....	34
Figure 12:Logo de Spring Boot	34
Figure 13 : Logo de Spring Data.....	34
Figure 14 : Logo de thymeleaf	35
Figure 15:Logo de Spring Security	35
Figure 16 : logo de hibernate	35
Figure 17 : Logo Apache Tomcat	35
Figure 18 : logo de eclipse	36
Figure 19 :Logo de Xamp.....	36
Figure 20 : Logo de PHP MyAdmin	36
Figure 21 : Logo de MySQL.....	37
Figure 22 : Logo de HTML.....	37
Figure 23 : Logo de CSS	37
Figure 24:Logo de Bootstrap	38
Figure 25 : Logo de JavaScript	38
Figure 26 : Logo de JQuery	38
Figure 27 : Logo de Entreprise Architect.....	38
Figure 28 : Interface de l'authentification	39
Figure 29 : Interface de table de bord	40
Figure 30 : Afficher les clients	41
Figure 31 : ajouter Client.....	41
Figure 32 : Modifier le client	42
Figure 33 : Avant La suppression d'un client.....	42
Figure 34 : Apres la suppression du client	43
Figure 35 : consulter produit.....	43
Figure 36 :Ajouter produit	44



Figure 37 : modifier produit	45
Figure 38 : Supprimer produit	46
Figure 39 : afficher inventaires.....	46
Figure 40 : pdf a imprime	47
Figure 41 : profile.....	48

Liste des Acronymes

Abréviation	Designation
PHP	Hypertext Preprocessor
CSS	Cascading Style Sheets
SQL	Structured Query Language
SGBD	Système de Gestion de Base de Données
HTML	Hypertext Markup Language
UML	Unified Modeling Language
MVC	Model View Controller
JEE	Java Enterprise Edition

Tableau 1 : tableau des acronymes

Introduction

Dans le cadre de mon formation en Licence Génie Informatique à la Faculté des Sciences et Techniques, j'ai effectué un stage de fin d'étude au sein du Agence Digitale Webmarko à Fès.

Comme son nom l'indique, un inventaire annuel doit être réalisé une fois par an avant la clôture de l'exercice comptable. Il s'agit d'une obligation légale pour toutes les entreprises (article 123.12 du Code de Commerce). Ce document détaille officiellement le stock de la société. Non seulement la quantité, mais également la valeur et l'état de chaque type de stock y sont précisés.

Au début, le travail d'un inventaire consiste à compter les produits restants manuellement pour un magazine à la fin d'année puis les valoriser, cette opération prend beaucoup de temps et il y a la possibilité de tromper pendant le calcul de prix total.

Dans le cadre de stage de fin d'études, j'ai été accueilli par l'agence **WebMarko** et la tâche qui m'a été confié c'est la réalisation d'une application web qui doit gérer les produits, les clients et leur stocks, ainsi que les inventaires. Certes, le bon fonctionnement de l'application et le respect du cahier de charges sont très importants. La sécurisation de cette dernière est d'une importance majeure. Cela m'a incité à penser à ce point important tout au long de la réalisation de cette application.

Dans ce PFE j'ai développé une application Web de gestion des stocks qui va offrir un service assez raffiné, adéquat et répondant particulièrement aux exigences demandées pour améliorer et faire évoluer les conditions de travail.

Ce rapport est organisé comme suit :

Le premier chapitre : décrit l'environnement de stage (WEBMARKO), la problématique, la solution proposée. Et le planning suivi.

Le deuxième chapitre : présente l'analyse des besoins fonctionnels et la conception adoptée.

Le troisième chapitre : illustre les interfaces graphiques de l'application.

Chapitre 1 : Contexte Général du Projet

1-Presentation de lieu de stage :

Dans cette partie, je présente le contexte général du projet :je commence par une présentation de l'organisme d'accueil **WebMarko**, ses clients ainsi que sa structure et ses domaines d'activités, ensuite je donne une brève présentation du projet de ce stage, à savoir l'application réalisée durant cette période.

1.1.1 L'agence WebMarko

WebMarko est une agence de communication digitale qui est située à la ville de Fès, l'agence webmarketing est spécialisée dans la création et le développement de sites web, ainsi que les applications web et mobiles. Cette agence est lancée par Mohamed Rachidi en 2012.

1.1.2 Ses Activités

Parmi ses activités dans le domaine de programmation et du développement :

- **Développement Web :**

Développer et créer des solutions sur mesure adaptées à vos besoins, en utilisant les meilleures fonctionnalités pour vous livrer un site professionnel, sécurisé, et optimal sur tous les supports IT.

- **Conception web design**

Accompagner votre stratégie digitale Webmarketing afin de vous assurer des résultats concrets à vos besoins. Analyser et mettre en place un plan stratégique pour suivre vos recommandations.

- **Développement application**

Créer un design unique et attractif pour vos supports de communications, je conçois une architecture de contenu qui reflète l'image de votre entreprise.

- **Référencement naturel**

Optimiser la visibilité de votre site internet, afin de le positionner et de le classer dans les premiers résultats des moteurs de recherche, en utilisant les meilleures techniques du référencement naturel.

- **Web Marketing**

Accompagner votre stratégie digitale Webmarketing afin de vous assurer des résultats concrets à vos besoins. Analyser et mettre en place un plan stratégique pour suivre vos recommandations.

- **Social media marketing**

Assurer la qualité d'un référencement naturel 100% efficace pour votre site web. Mettez votre site web dans la liste top numéro 1 !



2-Problématique :

Comme son nom l'indique, un inventaire annuel doit être réalisé une fois par an avant la clôture de l'exercice comptable. Il s'agit d'une obligation légale pour toutes les entreprises . Ce document détaille officiellement le stock de votre société. Non seulement la quantité, mais également la valeur et l'état de chaque type de stock y sont précisés. Un inventaire annuel possède plusieurs contraintes. En effet, il est chronophage et nécessite une importante charge de travail. Il faut prendre en compte le nombre d'employés indispensables pour le comptage des produits en stock. Si les marchandises sont nombreuses, l'inventaire peut prendre du temps. Afin d'assurer sa fiabilité, il est souvent nécessaire d'embaucher des personnes extérieures à l'entreprise formées dans ce domaine

3. Description générale de la solution proposée

Suite aux problèmes cités dans le paragraphe précédent, je propose la mise en place d'une application web Spring qui automatise le service de la gestion des stocks pour un responsable d'inventaire, accessible à le responsable d'Inventaire via une session personnelle, et qui permet d'achever les objectifs suivants :

- L'acteur de l'application peut consulter les données de ses travaux facilement même les données archivées en une sécurité totale, ce qui permet de rassembler sur une seule plateforme tous les éléments qui composent le projet.
- Charger les produits et les affecter à le client , puis calculer le prix totale pour valoriser son stock .
- Imprimer un inventaire contenant les informations personnelles du client et son stock .

4- Description du cahier des charges

Le projet porte sur la réalisation d'une application web pour la gestion de stocks, qui va être utilisable par responsable d'inventaire avec s'authentification pour garantir les droits d'accès .

Cette application va permettre de :

- Ajouter un produit
- Consulter les produits : modifications et suppression
- Ajouter un magasin de client
- Consulter un client : modifications et suppression
- Afficher ses produits
- Gérer un stock d'un magasin d'un client.
- Calculer la valeur de son stock.
- Etablir un inventaire: impression.
- Générer les statistiques de la base de données.

5- Déroulement de stage : diagramme de Gant

Ci-dessous le diagramme de Gantt illustrant le planning prévisionnel. Il a pour but de montrer les différentes étapes du stage et leur durée.

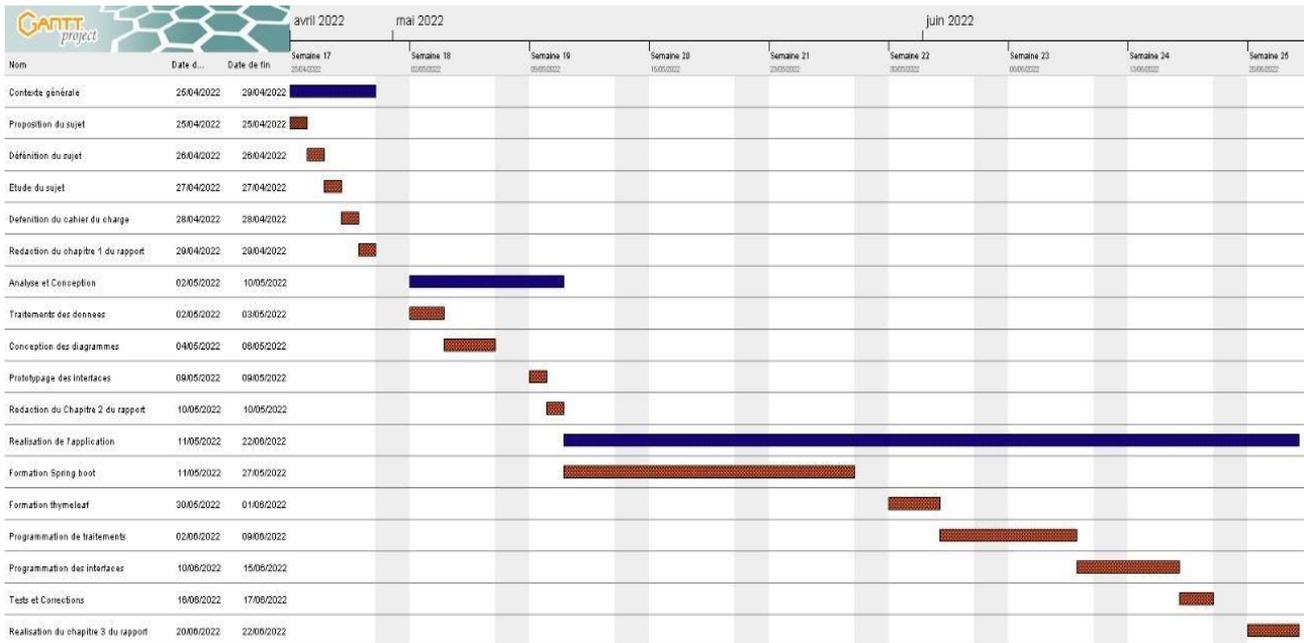


Figure 1 : Diagramme de Gant

Chapitre 2 : Analyse et Conception

1- Analyse des besoins fonctionnels et techniques :

Le but de mon projet est de satisfaire les besoins fonctionnels ; qui sont l'expression de ce que le produit ou le service délivré par le projet devrait être ou faire, aussi les exigences non fonctionnelles, qui représentent tout besoin qui spécifie quelque chose sur le système lui-même, et de quelle manière, il exécute ses fonctions.

1.1. Outils d'analyse et de conception

Afin de répondre aux besoins cités dans le cahier des charges, il est nécessaire de faire une étude conceptuelle pour les formuler sous forme de diagrammes à l'aide d'UML comme langage de modélisation.

Lors de l'étape de la conception j'ai utilisé les outils de modélisation suivants :

1.1.1. Langage UML

UML, c'est l'acronyme anglais pour « Unified Modeling Language ». Traduit par « Langage de modélisation unifié ». La notation UML est un langage visuel constitué d'un ensemble de schémas, appelés des diagrammes, qui donnent chacun une vision différente du projet à traiter. UML me fournit donc des diagrammes pour représenter le logiciel à développer :

son fonctionnement, sa mise en route, les actions susceptibles d'être effectuées par le logiciel.

1.1.2. modèle en cascade

Vu que les besoins auxquels doit répondre notre application sont stables et clairement identifiés dès le début, j'ai choisi comme modèle de cycle de mon projet le modèle en cascade .

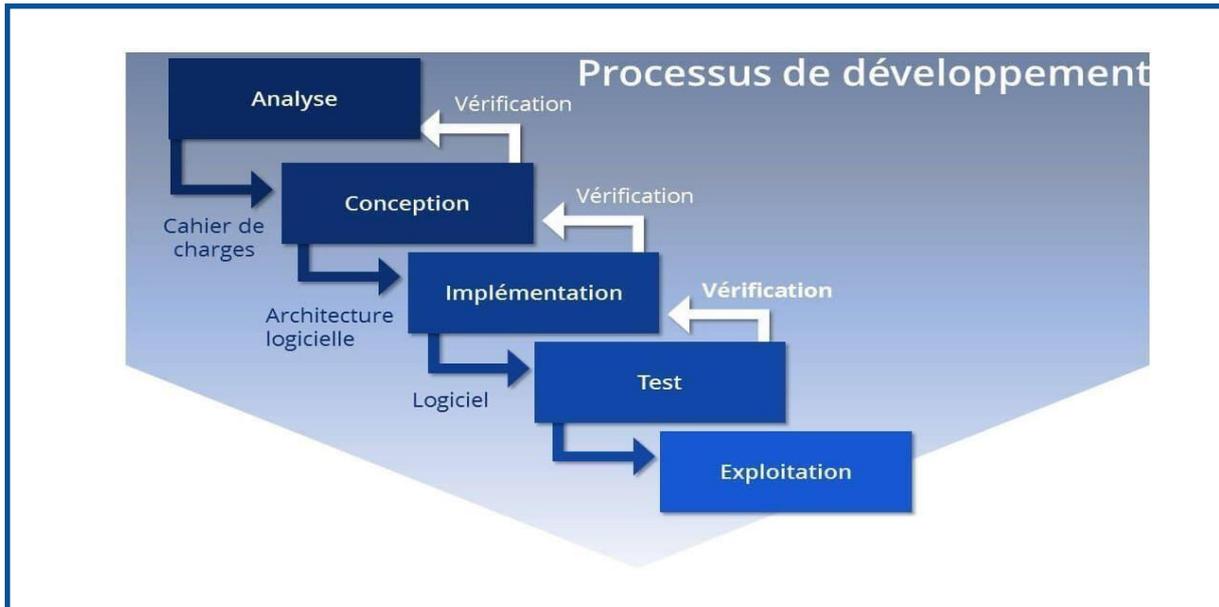


Figure 2 : Modèle en Cascade

Le principe du modèle en cascade est simple : on ne passe à la phase suivante que lorsque la précédente est validée. Cette méthode présente de nombreux avantages, notamment celui de sécuriser la planification du projet puisque l'on verrouille chacune des étapes les unes après les autres : on s'entend sur ce qu'on va faire (Analyse des besoins), le conçoit dans les grandes et les petites lignes (Conception) avant de le réaliser et de le produire (Codage et réalisation), en fin le tester (Intégration et tests).

1.1.3. Modèle MVC

L'architecture MVC (modèle, vue et contrôleur) est un concept très puissant qui intervient dans la réalisation d'une application. Il est un modèle destiné à répondre aux besoins des applications interactives en séparant les problématiques liées aux différents composants au sein de leur architecture respective, ce qui assure la clarté de l'architecture et simplifie la tâche du développeur responsable de la maintenance et de l'amélioration du projet.

- ❖ Un modèle contient les données à afficher.
- ❖ Une vue contient la présentation de l'interface graphique à l'utilisateur.
- ❖ Un contrôleur contient la logique concernant les actions effectuées par l'utilisateur.

Les différentes interactions entre le modèle, la vue et le contrôleur sont résumées par le schéma de

la figure suivante :

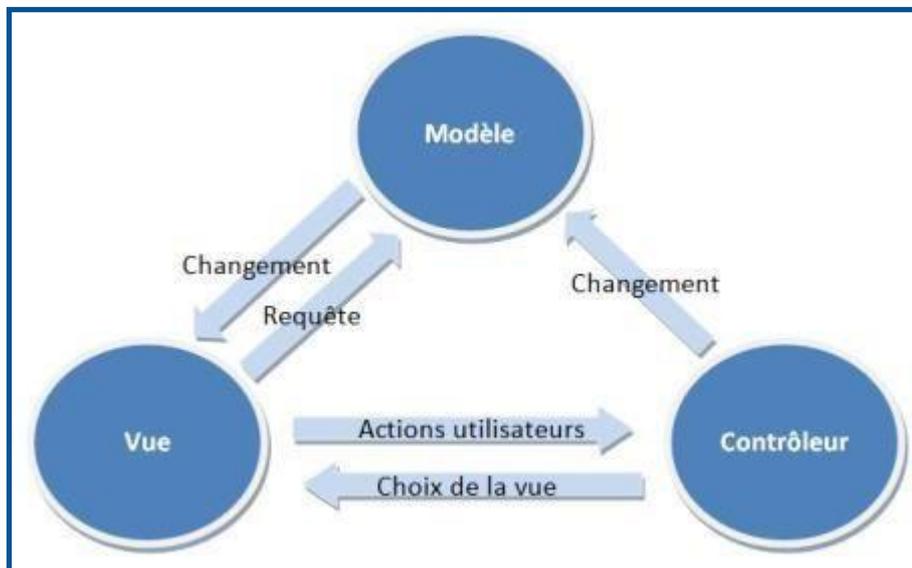


Figure 3:Modele MVC

1.2 – Les besoins fonctionnels

L'application envisagée doit satisfaire essentiellement les besoins suivants :

- **Gestion des clients :**

La gestion de stock est le mode de réalisation d'un projet, qui couvre l'ensemble des outils, techniques et méthodes qui permettent au responsable d'inventaire de saisir les informations personnelles des clients et afficher leurs stocks .

- **Gestion des produits:**

Cette phase consiste à ajouter les produits avec la possibilité de les éditer et les supprimer .

- **Gestion des inventaires :**

Etablir une inventaire contient les informations personnelles de client et la liste des produits ainsi le prix total .

1-3. Les besoins non fonctionnels :

Les besoins non fonctionnels décrivent toutes les contraintes auxquelles le système est soumis pour sa réalisation et son bon fonctionnement.

- Fiabilité de mon application doit fonctionner de façon cohérente sans erreurs.
- Ergonomie, souplesse et confort d'utilisation Pour faciliter l'utilisation, mon interface doit être unifiée, conviviale et ergonomique.

- Gain de temps , mon application doit optimiser les traitements pour avoir un temps de réponse minimal.

1.4 architecture JEE

- **Couche présentation :**

Elle correspond à la partie de l'application visible et interactive avec les Utilisateurs On parle alors d'Interaction Homme Machine. La couche présentation relaie les requêtes de l'utilisateur à destination de la couche métier, et en retour lui présente les informations renvoyées par les traitements de cette couche Il s'agit donc ici d'un assemblage de services métiers et applicatifs offerts par la couche inférieure.

- **Couche Web :**

Responsable de la logique applicative et cerveau de l'application : application informatique Servlets, JSF, framework Java EE ou autres.

- **Couche Métier :**

Elle correspond à la partie fonctionnelle de l'application, et qui décrit les opérations que l'application va appliquer sur les données en fonction des requêtes des utilisateurs, effectuées au travers de la couche présentation. Les différentes règles de gestion et de contrôle du système sont mises en œuvre dans cette couche. La couche métier offre des services applicatifs et métiers à la couche présentation.

- **Couche Accès aux données :**

Elle consiste en la partie gérant l'accès aux gisements donnés du système. Les données peuvent être stockées indifféremment dans de simples fichiers texte, ou extensible Markup Language (XML), ou encore dans une base de données.

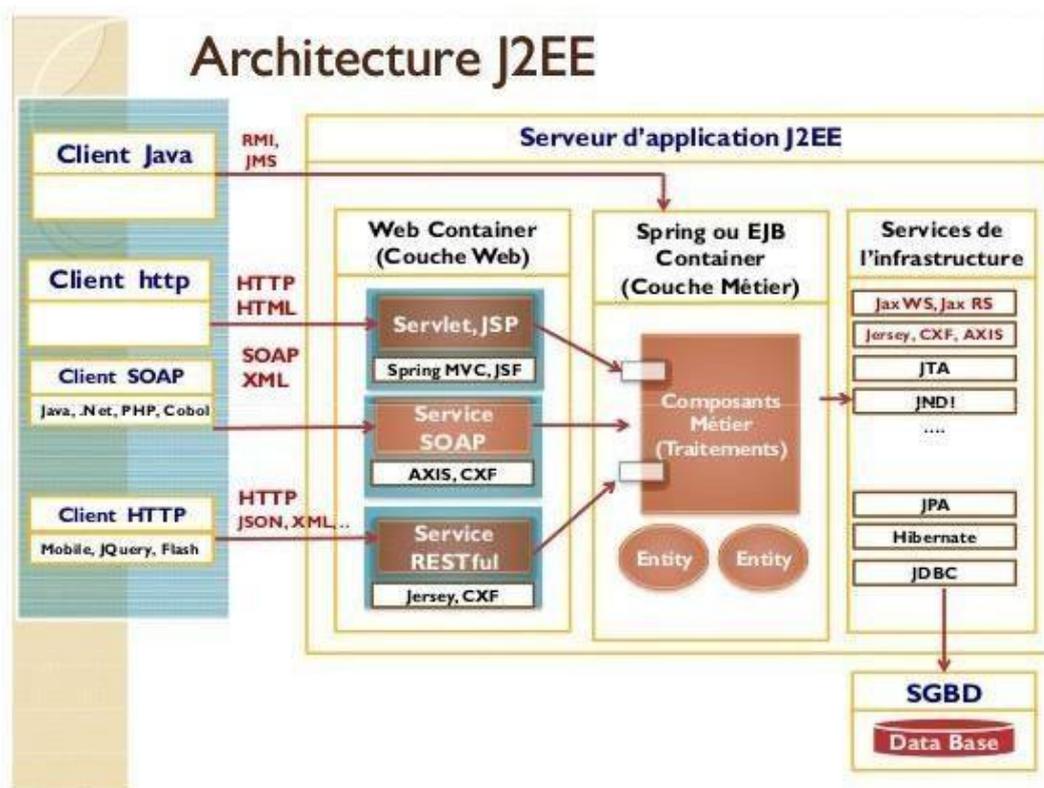


Figure 4 : Achitecture de JEE

Spring est né de l'idée de fournir une solution plus simple et plus légère que celle proposée par Java 2 EE. C'est pour cette raison que Spring a été initialement désigné comme un conteneur léger (lightweight container).

Il permet de faciliter une implémentation de bout en bout de l'application ; La couche haute utilisera Spring MVC. La couche sécurité utilisera Spring Security pour les rôles. Les autres couches utiliseront essentiellement la programmation orientée aspects (AOP), et

la couche d'accès aux données utilisera Spring Data.

Spring propose une très bonne intégration avec des frameworks open source (Struts, Hibernate, ...) ou des standards de Java (Servlets, JMS, JDO, ...)

Spring Boot flow architecture

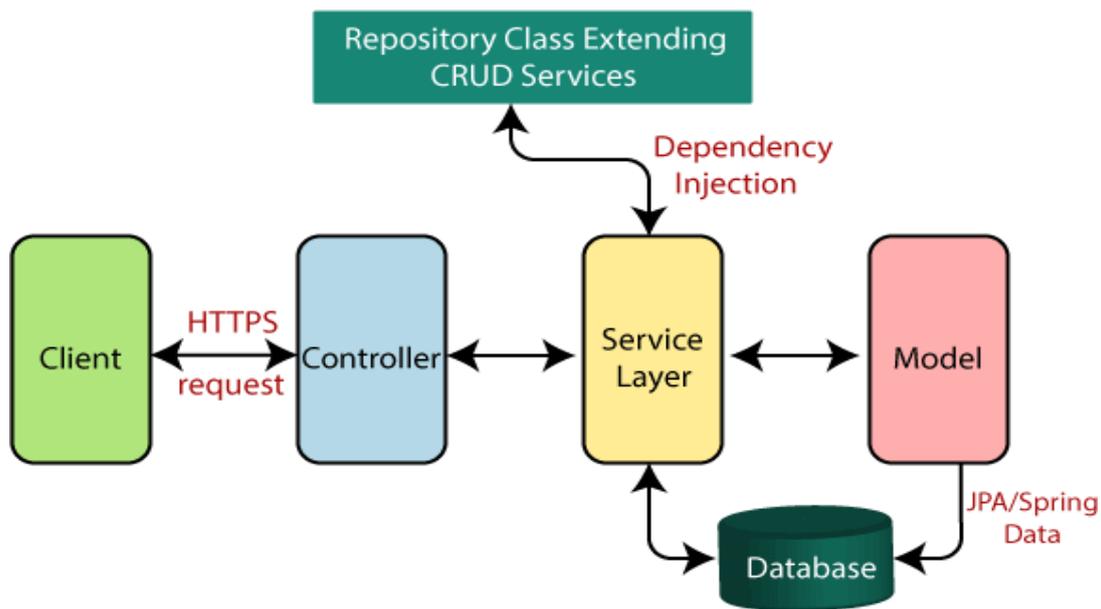


Figure 5 : Architecture de SpringBoot

2. Conception

2.1. Acteur identifié

C'est le responsable sur la gestion des stocks, les droits accordés à administrateur sont :

- **L'authentification**
- **L'ajout de magasin de client**
- **La modification des clients**
- **La suppression des clients**
- **L'ajout d'un produit**
- **La modification des produits**
- **La suppression des produits**
- **La gestion du stock d'un client**
- **La consultation des différentes listes :**
 - ✓ Les produits
 - ✓ Les clients
 - ✓ les inventaires

2.2. Les diagrammes de cas d'utilisation

C'est un diagramme fonctionnel. Il montre les interactions fonctionnelles des acteurs et du système d'étude. Il délimite précisément le système, décrit ce que fera le système sans spécifier comment. Il exprime les services (use cases) offerts par le système aux utilisateurs (acteurs).

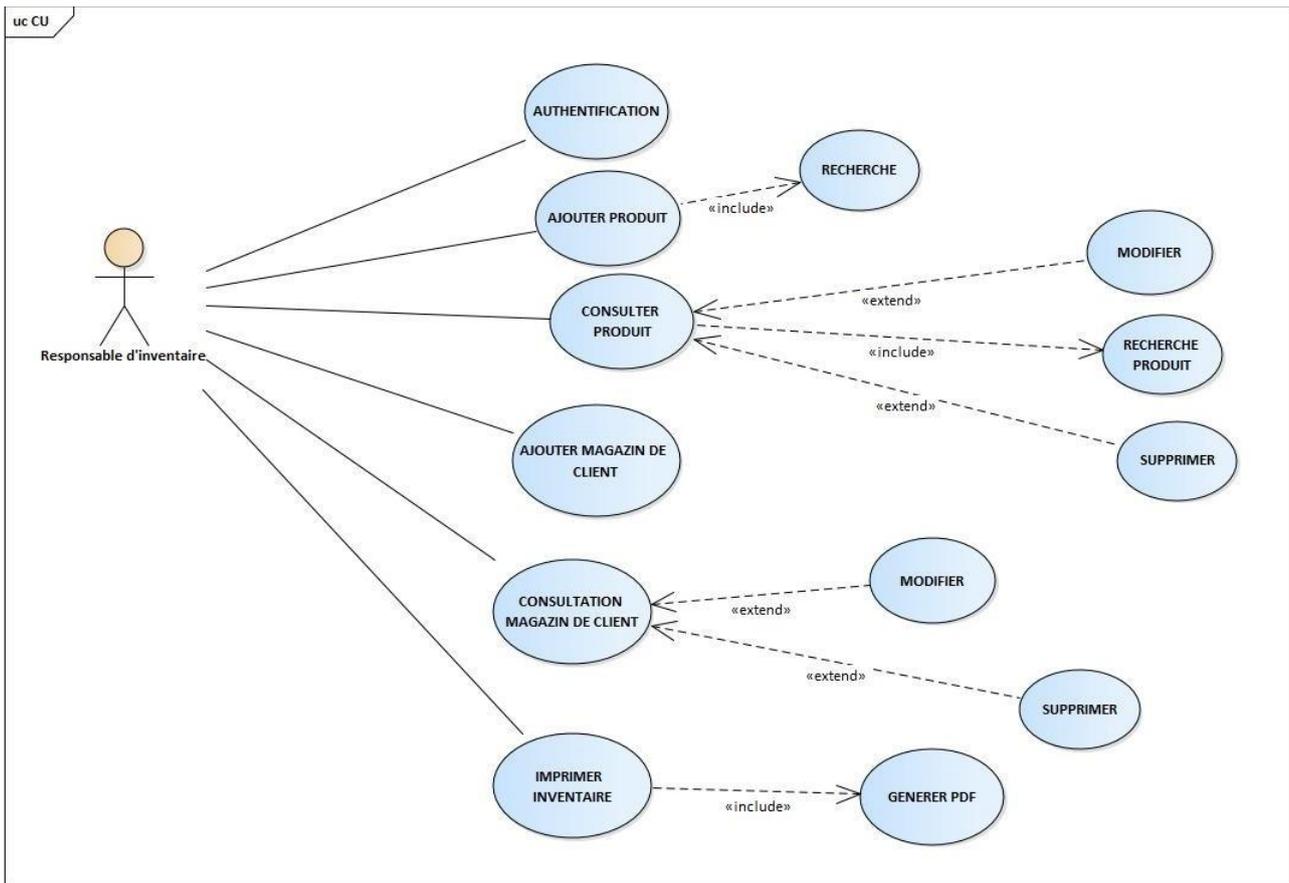


Figure 6: Diagramme de cas d'utilisation

2.3. Analyse des cas d'utilisation et diagramme de séquence

Après avoir établi les diagrammes des cas d'utilisations, J'ai décrit les cas d'utilisations les plus important à travers des scénarios, des diagrammes d'activités et des diagrammes de séquences:

2.3.1 Authentification

Cas d'utilisation	Authentification
Acteur	✓ Responsable d'inventaire
Description	Cette fonctionnalité permet à l'acteur l'accès à l'application
Scénario normal	
<ul style="list-style-type: none">• L'acteur remplit les champs et clique sur 'se connecter'.• Le système vérifie si l'acteur a bien rempli tous les champs.• Le système vérifie si les informations saisies sont correctes.• La session commence et la page d'accueil s'ouvre .	
Scénario alternatif 1	
<ul style="list-style-type: none">• L'acteur remplit les champs et clique sur 'se connecter'.• Le système détecte que l'acteur a oublié de remplir un champ ou plus.• Le système réaffiche le formulaire avec des alertes d'erreur à côté des champs non remplis.	
Scénario alternatif 2	
<ul style="list-style-type: none">• L'acteur remplit les champs et clique sur 'se connecter'.• Le système vérifie si l'acteur a rempli tous les champs.• Le système trouve que le mot de passe ou le username sont incorrectes.• Le système affiche une erreur indiquant que le mot de passe ou l'email sont incorrectes.	

Tableau 2: Scénario de l'authentification

Cette fonctionnalité permet à l'acteur l'accès à l'application, et elle peut illustrée par le diagramme de séquence suivant :

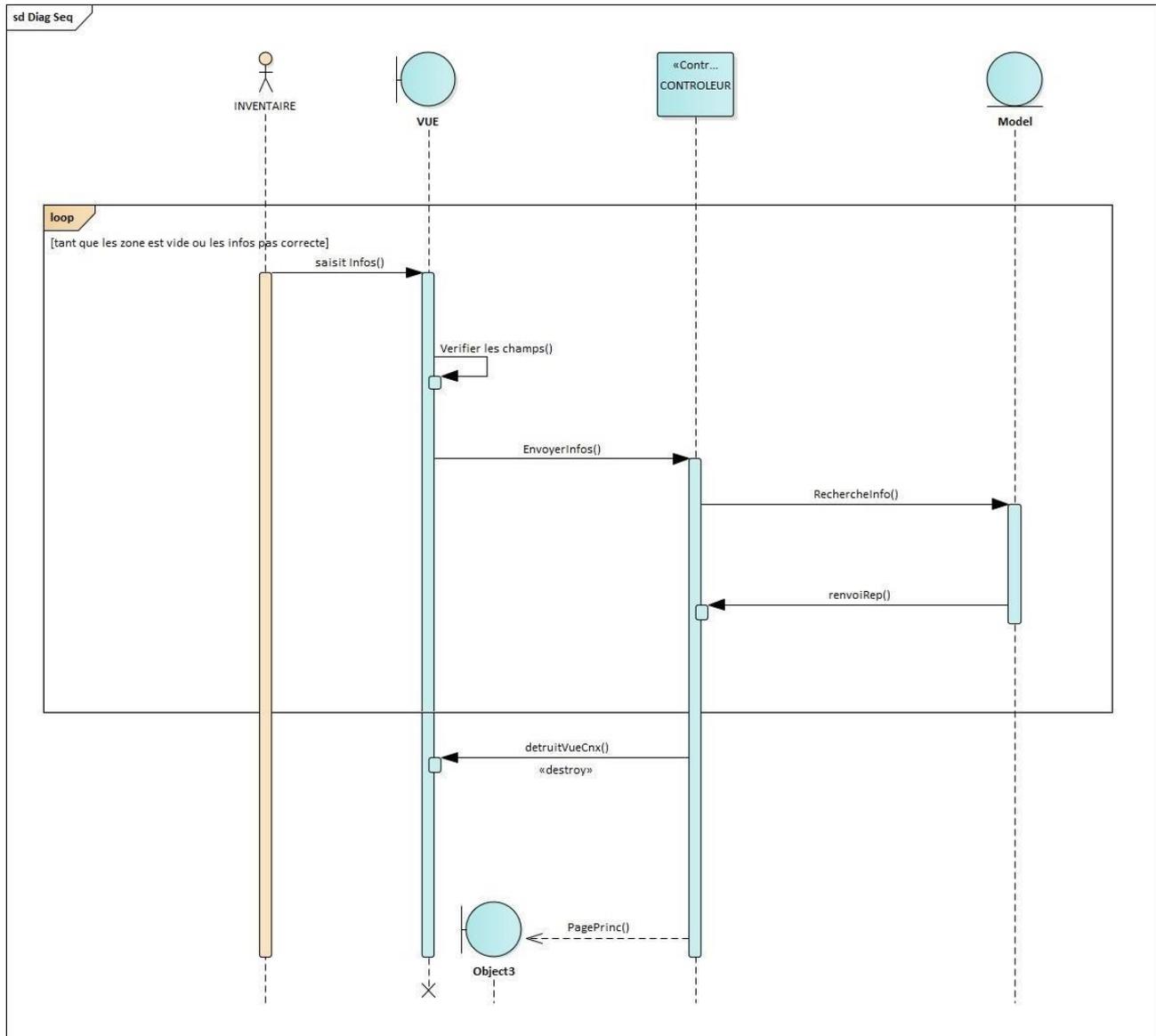


Figure 7 :diagramme de séquence pour l'authentification

2.3.2 L'ajout d'un nouveau client

Cas d'utilisation	Ajouter Client
Acteur	✓ Responsable d'inventaire
Description	Permet à l'acteur d'ajouter les informations sur Un client dans la base de données.
Scénario normal	
<ul style="list-style-type: none">• Le système génère une page d'ajout sous forme d'un formulaire à remplir.• L'administrateur insère toutes les informations du client.• Le système vérifie la validité des informations saisies.• Le système redirige l'acteur vers la vue de la table de clients .• Le système ajoute les informations du projet dans la base de données.	
Scénario alternatif	
<ul style="list-style-type: none">✓ Si un champ est invalide, le système indique dans la même vue une erreur des champs incomplets.	

Tableau 3 : scenario d'un nouveau client

Cette fonction permet à l'acteur d'ajouter les informations sur Un client dans la base de données, et elle peut illustrée par le diagramme de séquence suivant :

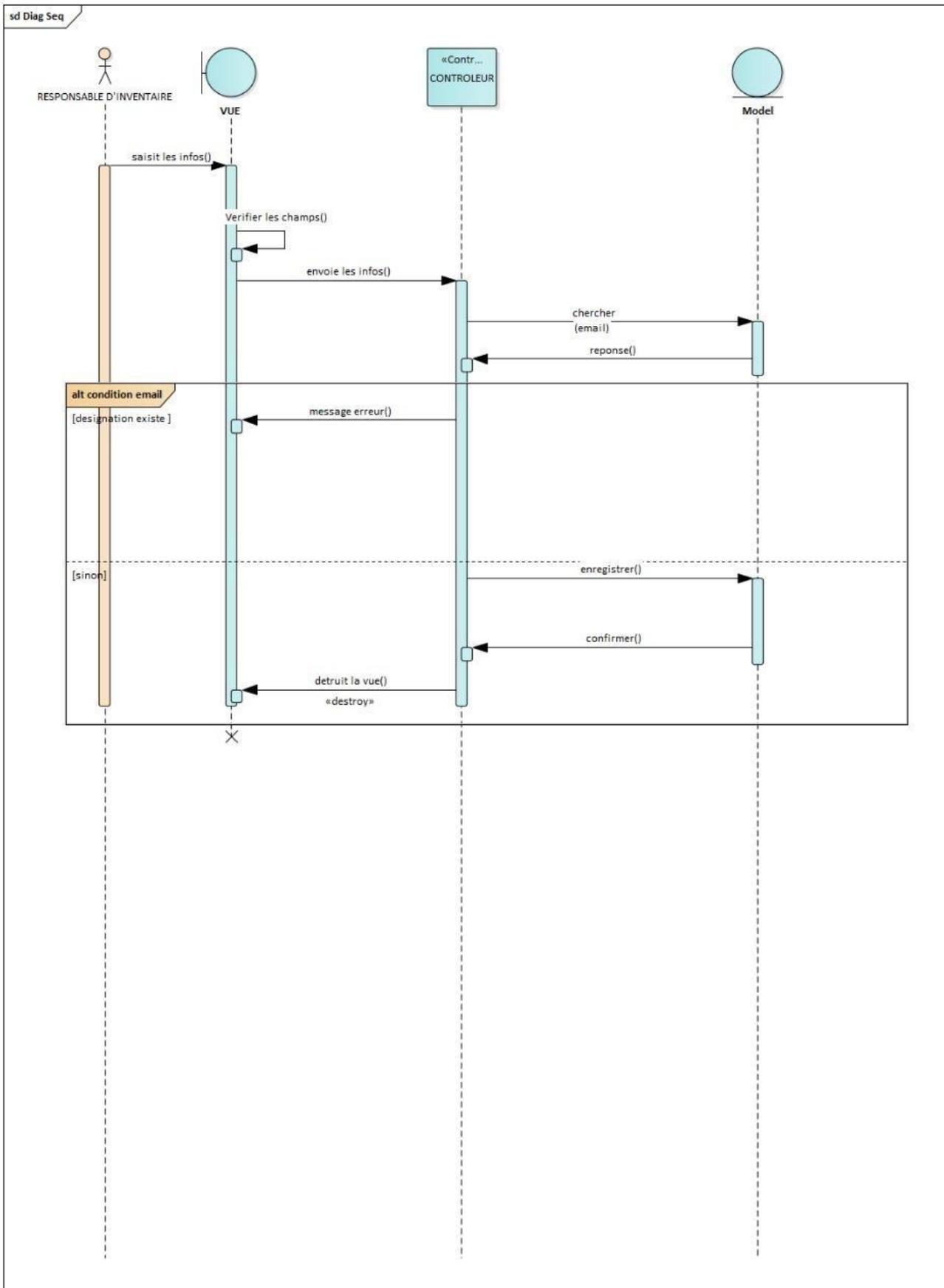


Figure 8 : digramme de séquence ajouter client

2.3.3 L'ajout d'un nouveau produit

Cas d'utilisation	Ajouter produit
Acteur	✓ Responsable d'inventaire
Description	Permet à l'acteur d'ajouter les informations sur Un produit dans la base de données
Scénario normal	
<ul style="list-style-type: none"> • Le système génère une page d'ajout sous forme d'un formulaire à remplir. • l'administrateur insère toutes les informations du produit. • Le système vérifie la validité des informations saisies. • Le système redirige l'acteur vers la vue de la table de clients . • Le système ajoute les informations de produit dans la base de données. 	
Scénario alternative	
<ul style="list-style-type: none"> ✓ Si un champ est invalide, le système indique dans la même vue une erreur des champs incomplets. 	

Tableau 4: scénario de l'ajout d'un produit

Cette fonction permet à l'acteur d'ajouter les informations sur Un produit dans la base de données, et elle peut illustrée par le diagramme de séquence suivant :

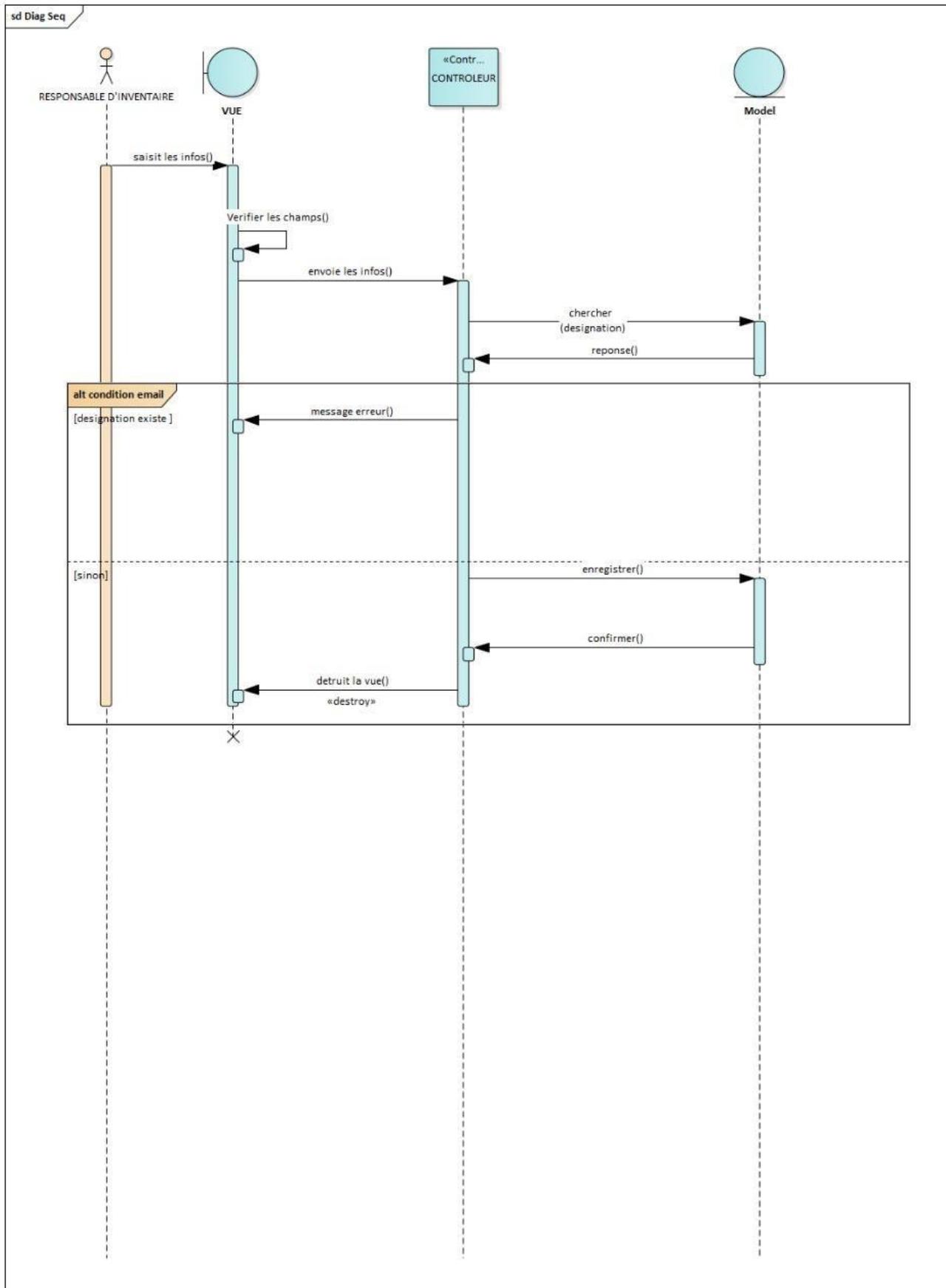


Figure 9 ; digramme de séquence ajouter produit

2.4 Diagramme de classes

Le diagramme de classes est un schéma utilisé en génie logiciel pour présenter les classes et les interfaces des systèmes ainsi que les différentes relations entre celles-ci.

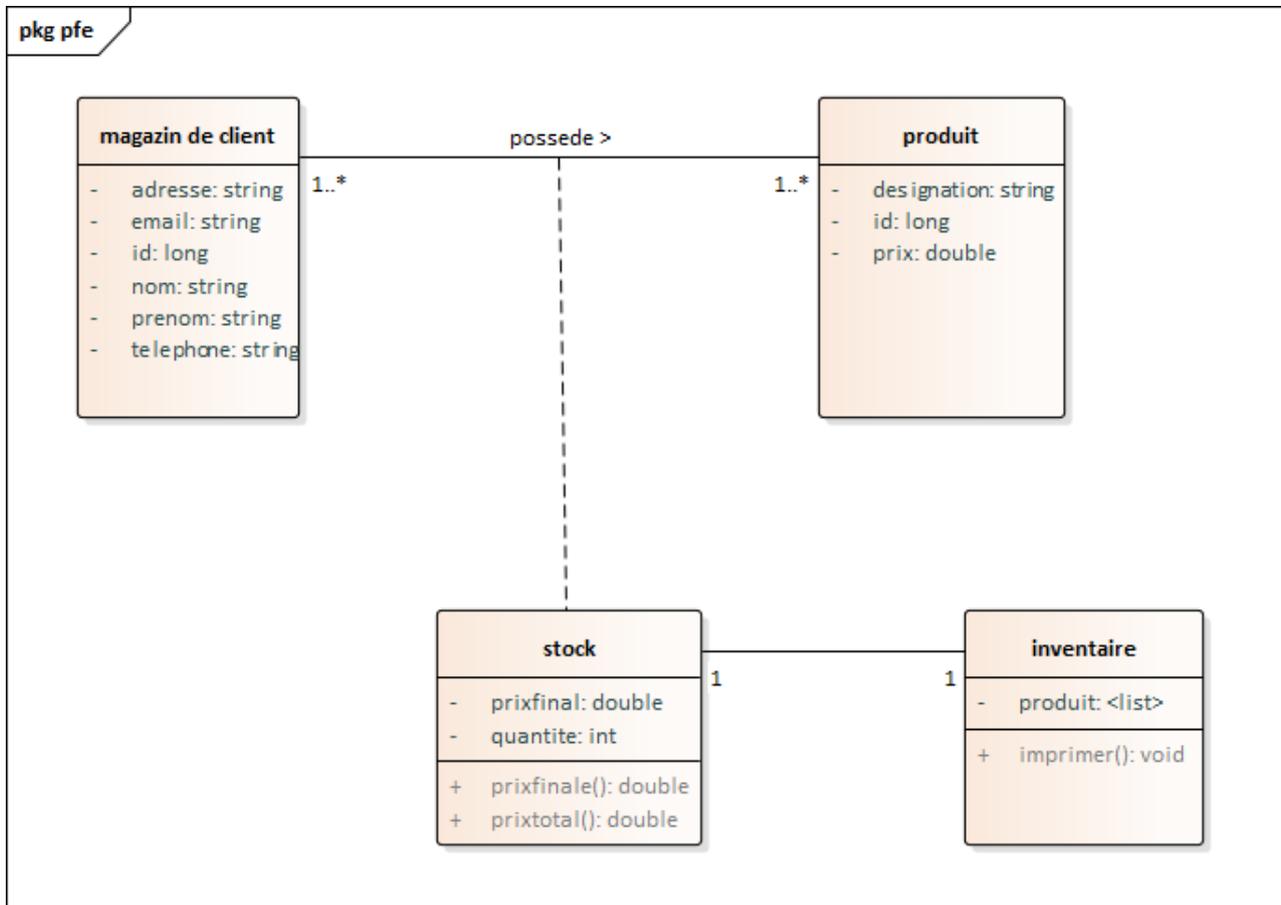


Figure 10: diagramme de classes

Chapitre 3 : Réalisation de L'Application

1 . Besoins techniques

➤ JEE



Figure 11 : : logo de JEE

C'est la technologie adoptée, le terme Java JEE signifie Java Enterprise Edition, et était anciennement raccourci en « JEE ». Il fait quand à lui référence à une extension de la plate- forme standard. Autrement dit, Java EE construite le langage et la plate-forme Java SE, et elle ajoute un grand nombre de bibliothèque remplissant tous un tas de fonctionnalités que plate-forme standard ne remplit pas d'origine. L'objectif majeur de Java EE est de faciliter le développement d'application web exécutée sur un serveur d'application.

➤ Spring Boot



Spring Boot

Figure 12:Logo de Spring Boot

Spring est un Framework open source pour construire et définir l'infrastructure d'une application Java3, C'est une déclinaison du Framework classique de Spring qui permet de mettre en œuvre les autres composants (Spring Data, Spring Security) avec facilité, notamment grâce aux starters de dépendances et à l'auto-configuration.

➤ Spring Data



Figure 13 : Logo de Spring Data

Ce composant permet de communiquer avec de nombreux types de bases de données. Par exemple, il offre la capacité de communiquer avec une base de données en implémentant uniquement des interfaces.

➤ Thymeleaf



Figure 14 : Logo de thymeleaf

Thymeleaf est un moteur de template, sous licence Apache 2.0, écrit en Java pouvant générer du XML/XHTML/HTML5 Thymeleaf peut être utilisé dans un environnement web (utilisant l'API Servlet) ou non web. Son but principal est d'être utilisé dans un environnement web pour la génération de vue pour les applications web basées sur le modèle MVC.

➤ Spring Security



Figure 15:Logo de Spring Security

Spring Security est un Framework de sécurité léger qui fournit une authentification et un support d'autorisation afin de sécuriser les applications Spring. Il est livré avec des implémentations d'algorithmes de sécurité populaires.

➤ Hibernate



Figure 16 : logo de hibernate

C'est un Framework gérant la persistance des objets en base de données rationnelle. Hibernate apporte une solution aux problèmes d'adaptation entre le paradigme objet et les SGBD en remplaçant les accès à la base de données par des appels à des méthodes objet de haut.

➤ Apache Tomcat



Figure 17 : Logo Apache Tomcat

Apache Tomcat est un conteneur libre web de servlets et JSP. Issu du projet Jakarta. Il implémente les spécifications des servlets et des JSP du Java Community process est paramétrable par des fichiers XML et des propriétés, et inclut des outils pour la configuration et la gestion. Il comporte également un serveur HTTP.

➤ Eclipse IDE



Figure 18 : logo de eclipse

Eclipse Mars Release (4.5.0) est un environnement de développement intégré aidant à écrire, compiler, déboguer et déployer des programmes, S'appuyant principalement sur Java.

➤ Xamp



Figure 19 :Logo de Xamp

WampServer est une plateforme de développement Web de type WAMP, permettant de faire fonctionner localement (sans avoir à se connecter à un serveur externe) des scripts PHP.

WampServer n'est pas en soi un logiciel, mais un environnement comprenant :

- Trois serveurs (Apache, MySQL et MariaDB)
- Un interpréteur de script (PHP).
- PhpMyAdmin pour l'administration Web des bases MySQL.

➤ PHP MyAdmin



Figure 20 : Logo de PHP MyAdmin

Il s'agit de l'une des plus célèbres interfaces pour gérer une base de données MySQL sur un serveur PHP.

➤ MySQL



Figure 21 : Logo de MySQL

C'est un Système de gestion de base de données (SGBD). Issu du monde libre, il est l'un des logiciels de gestion de base de données le plus utilisé au monde. MySQL est un serveur de base de données relationnelles SQL, il est multithread (peut exécuter plusieurs processus en même temps) et multi-Utilisateur.

➤ HTML



Figure 22 : Logo de HTML

L'HyperText Markup Language, généralement abrégé HTML, C'est un langage permettant d'écrire de l'hypertexte, et c'est le langage de balisage conçu pour représenter les pages web.

➤ CSS



Figure 23 : Logo de CSS

Cascading Style Sheets, les feuilles de style en cascade forment un langage informatique qui décrit la présentation des documents HTML et XML.

Les standards définissant CSS sont publiés par le World Wide Web Consortium (W3C).

➤ Bootstrap



Figure 24: Logo de Bootstrap

C'est un Framework de CSS destiné aux applications web. Développé par Twitter et distribué sous licence Apache 2, c'est un outil à considérer lors du développement d'applications web.

➤ JavaScript



Figure 25 : Logo de JavaScript

JavaScript est un langage de programmation de scripts principalement employé dans les pages web interactives mais aussi pour les serveurs.

➤ JQuery



Figure 26 : Logo de JQuery

est une bibliothèque JavaScript libre et multiplateforme créée pour faciliter l'écriture de scripts côté client dans le code HTML des pages web2 . La première version est lancée en janvier 2006 par John Resig.

➤ Enterprise Architect



Figure 27 : Logo de Enterprise Architect

C'est un logiciel de modélisation et de conception UML, édité par la société australienne Sparx Systems. Couvrant, par ses fonctionnalités, l'ensemble des étapes du cycle de la conception d'application, il est l'un des logiciels de conception et de modélisation les plus reconnus

2-Authentification

L'interface d'authentification est la première interface affichée au lancement de l'application. Elle a pour but de vérifier l'identité de responsable d'inventaire. L'accès sera autorisé seulement, si les informations sont correctes, sinon la vue sera rafraichir.

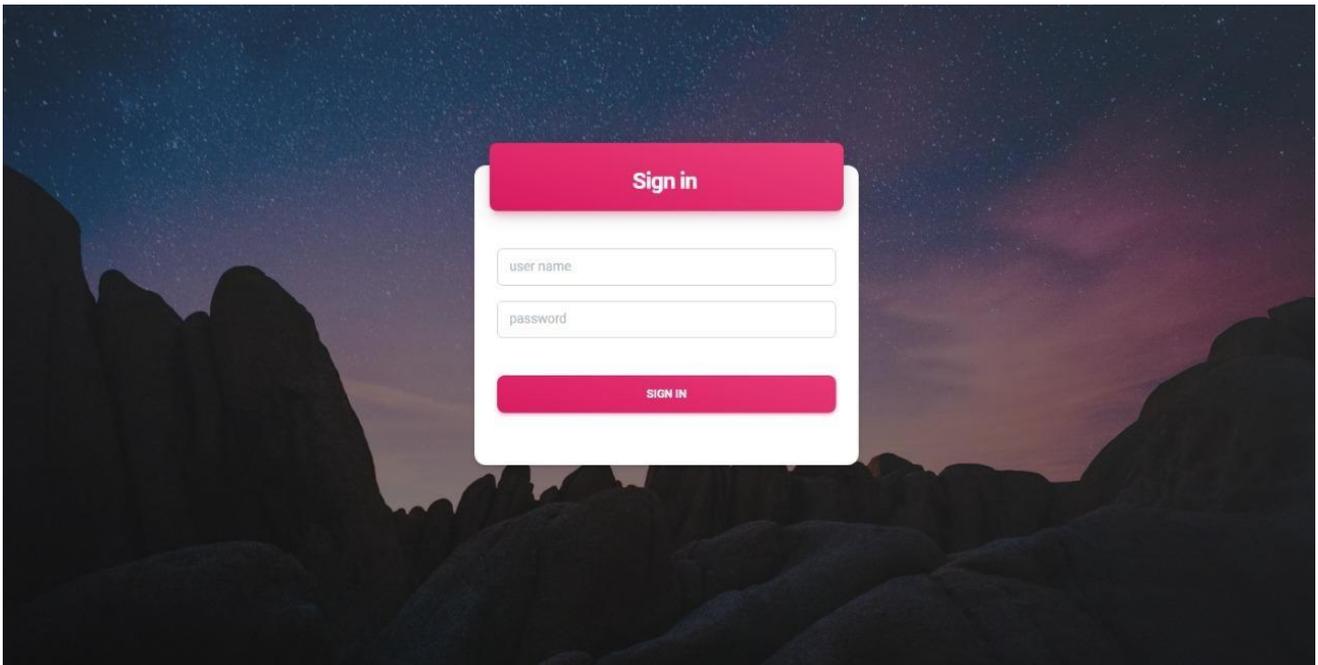


Figure 28 : Interface de l'authentification

3-Table de bords

Le corps de cette page contient trois cellules qui représente les nombres des clients, des produits et des factures. Et un menu « Sidebar » qui permet un accès rapide aux différents tâches de l'application

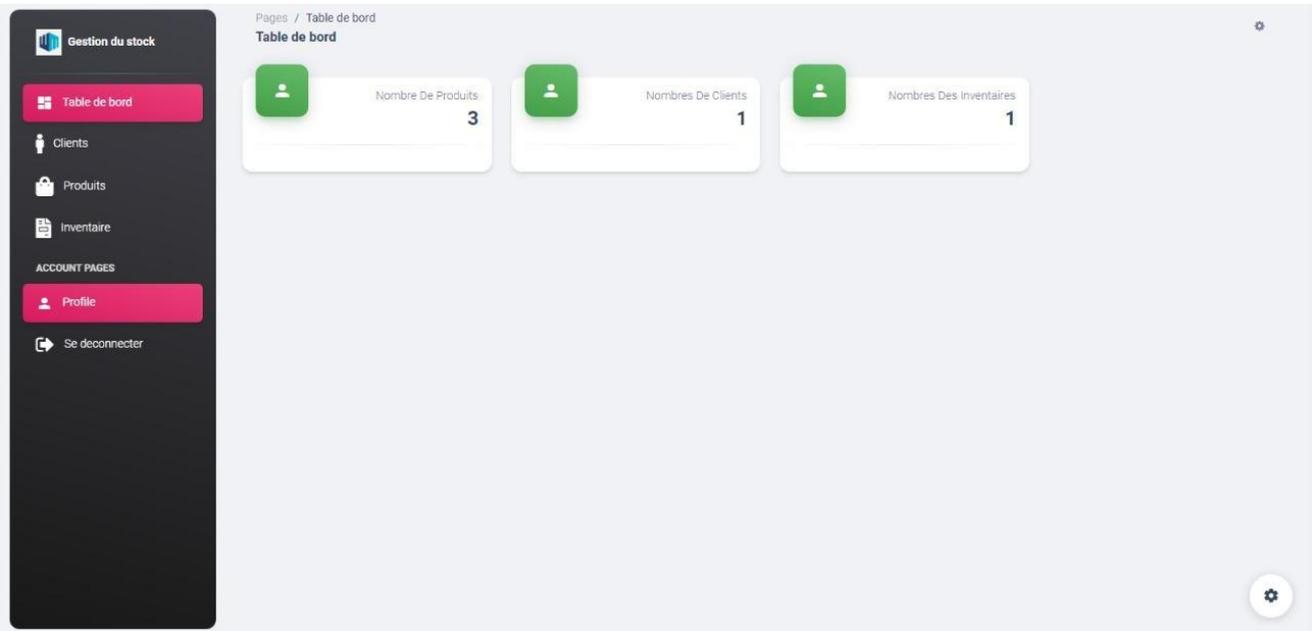


Figure 29 : Interface de table de bord

4-Gestion des clients

4.1 Visualisation des magasins des Clients

Cette interface permet de consulter la liste des clients avec ses informations personnelles.

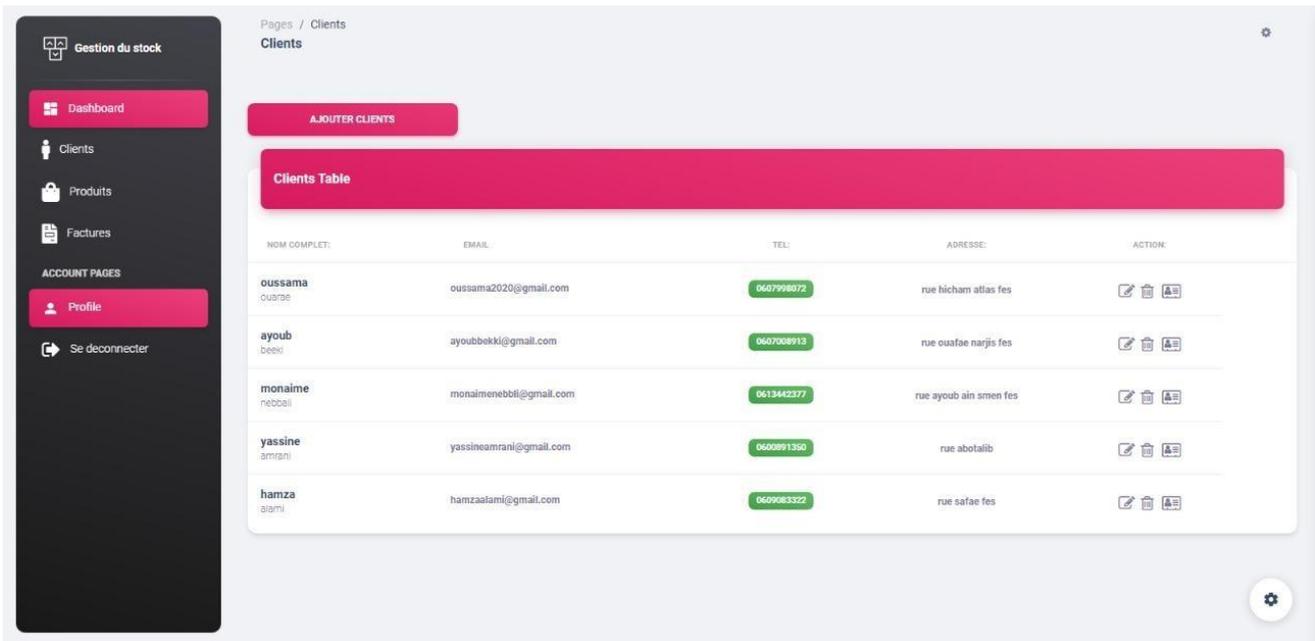


Figure 30 : Afficher les clients

4.2 Ajout de magasin de client

Cette interface permet le responsable d'inventaire de saisir les informations personnelles d'un nouveau client.

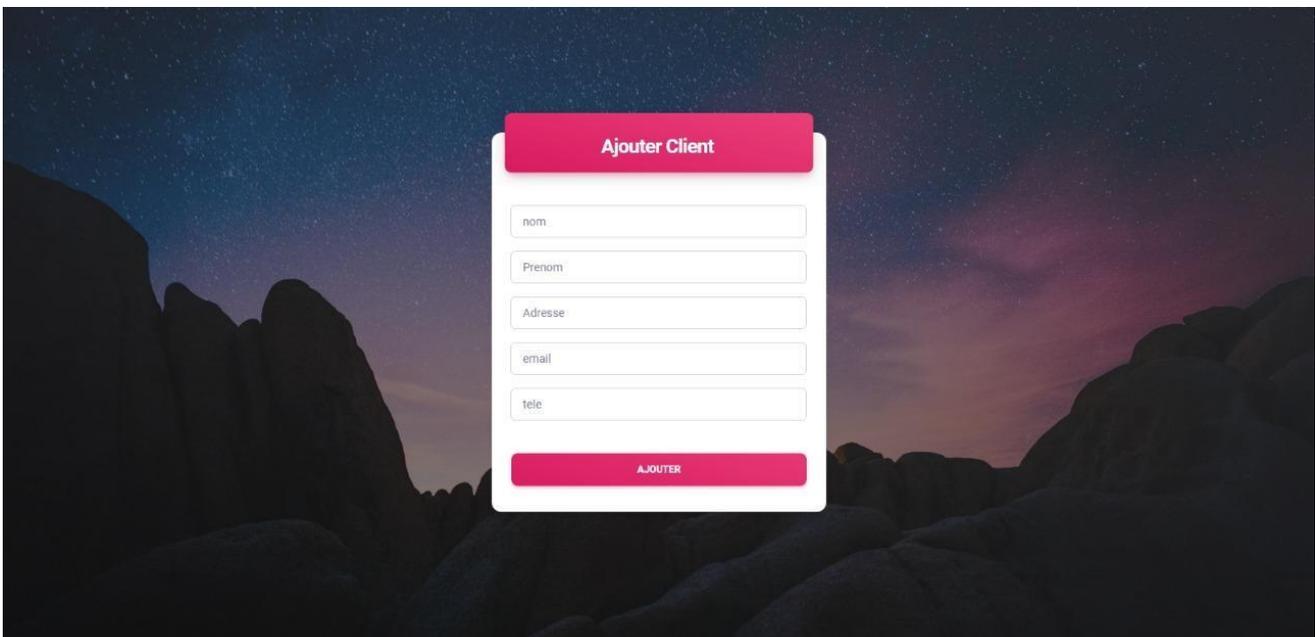


Figure 31 : ajouter Client

4.3 modification de magasin de client

Cette interface permet le responsable d'inventaire de modifier les informations de client sélectionné.

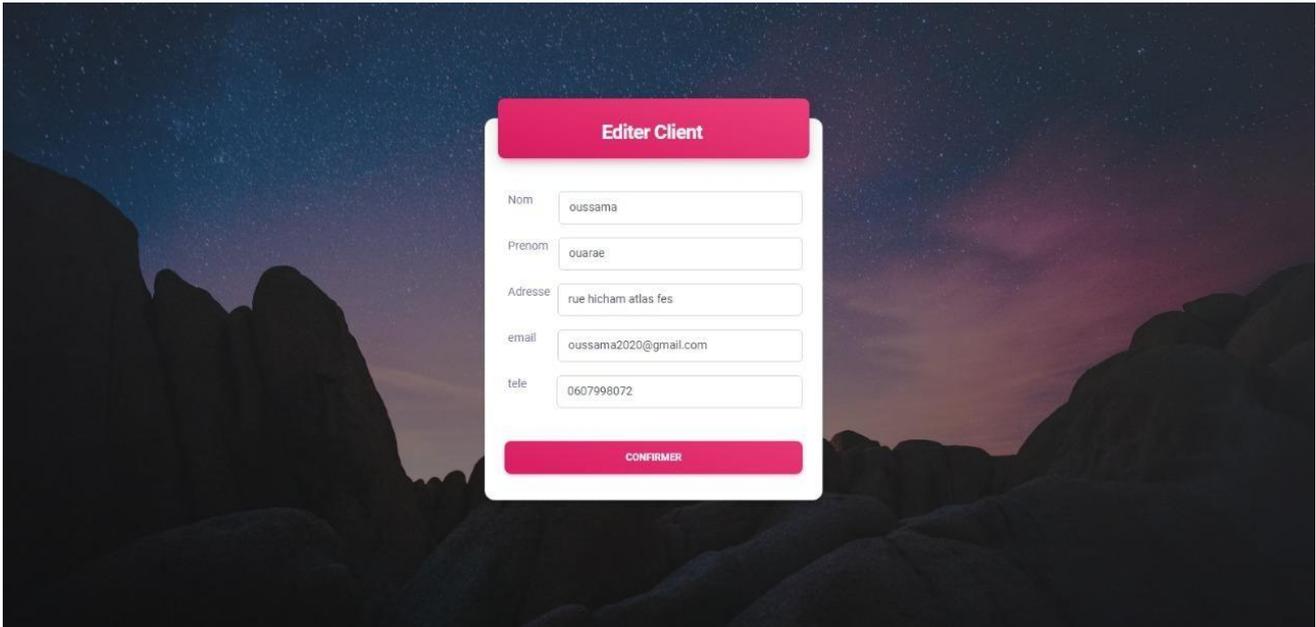


Figure 32 : Modifier le client

4.4 Suppression de magasin de Client

Avec l'icône de button <supprimer> me permet de supprimer un client .

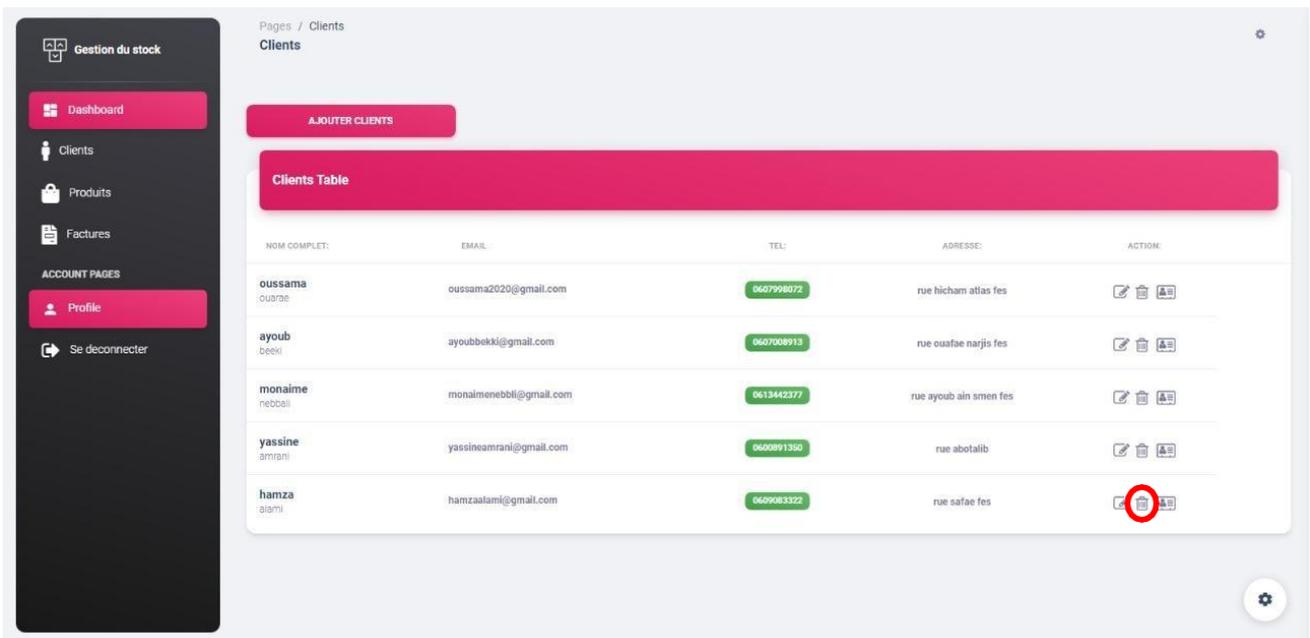


Figure 33 : Avant La suppression d'un client

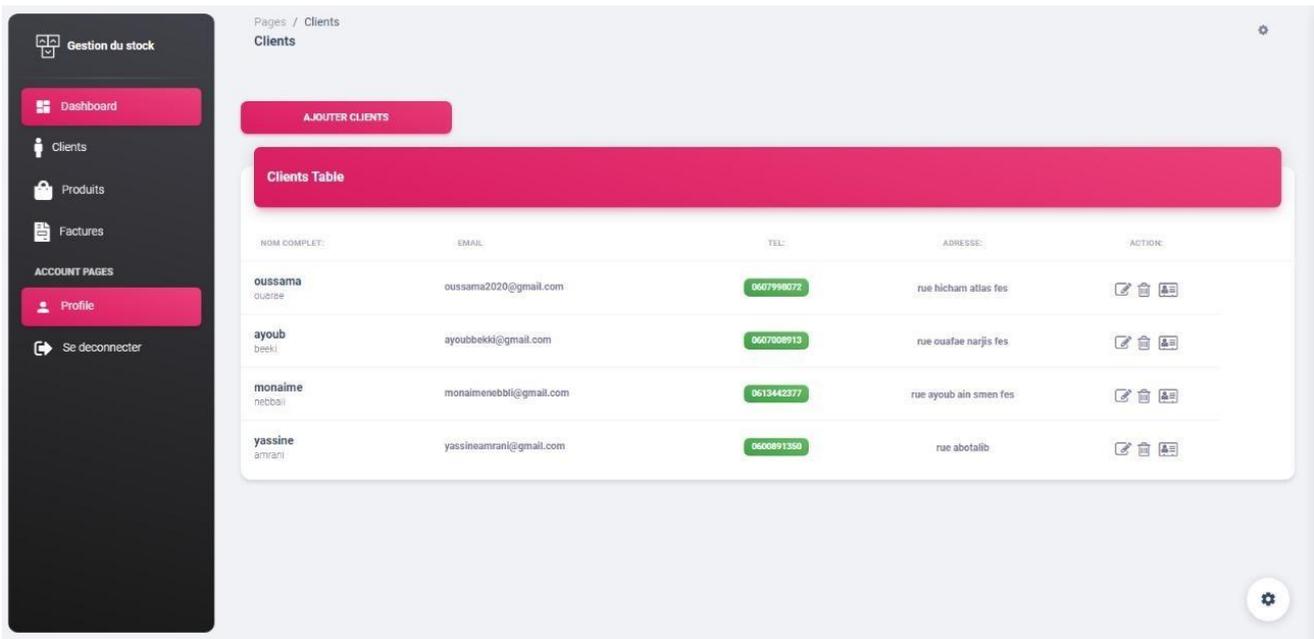


Figure 34 : Apres la suppression du client

5- Gestion des Produits

5-1 Visualisation

Cette interface permet de consulter la liste des produits.

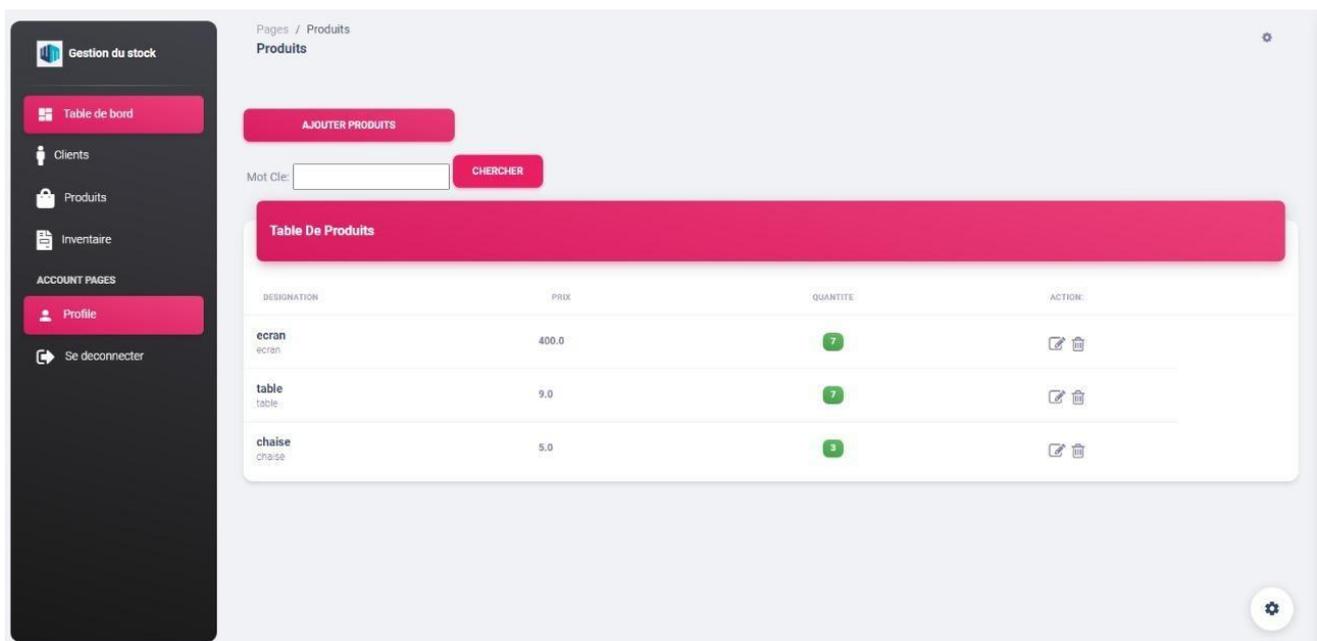
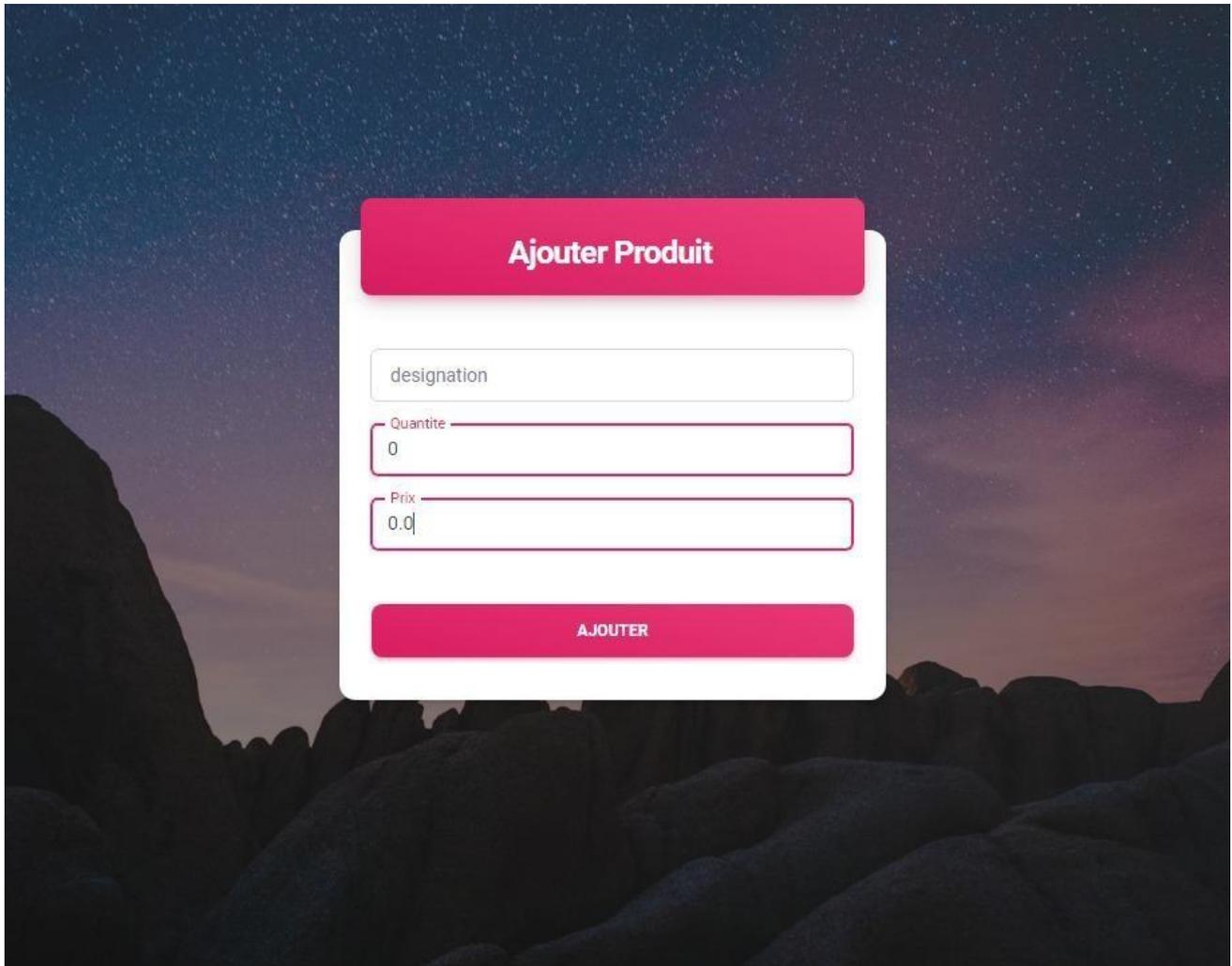


Figure 35 : consulter produit

5-2 Ajout d'un produit

Cette interface permet le responsable d'inventaire de saisir les informations personnelles d'un nouveau produit.

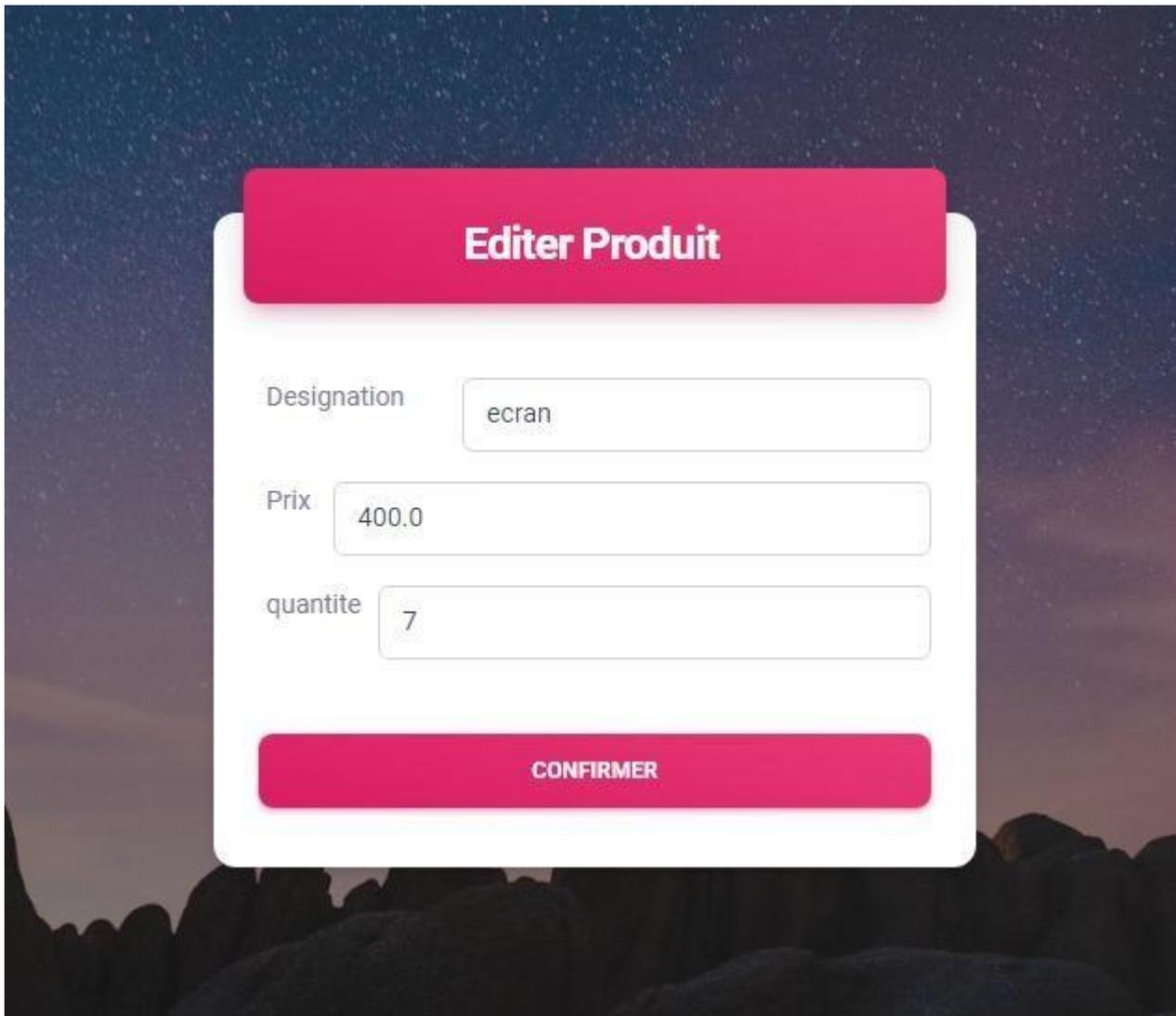


The image shows a mobile application interface for adding a product. It features a white modal box with a pink header titled "Ajouter Produit". Below the header are three input fields: "designation" (containing the text "designation"), "Quantite" (containing "0"), and "Prix" (containing "0.0"). At the bottom of the modal is a pink button labeled "AJOUTER". The background is a dark, starry night sky over a rocky landscape.

Figure 36 :Ajouter produit

5-3 modification

Cette interface permet l'administrateur de modifier les informations du produit sélectionné.



Editer Produit

Designation

Prix

quantite

CONFIRMER

Figure 37 : modifier produit

5-4 Suppression d'un Produit

Avec l'icône de bouton <supprimer> me permet de supprimer un produit .

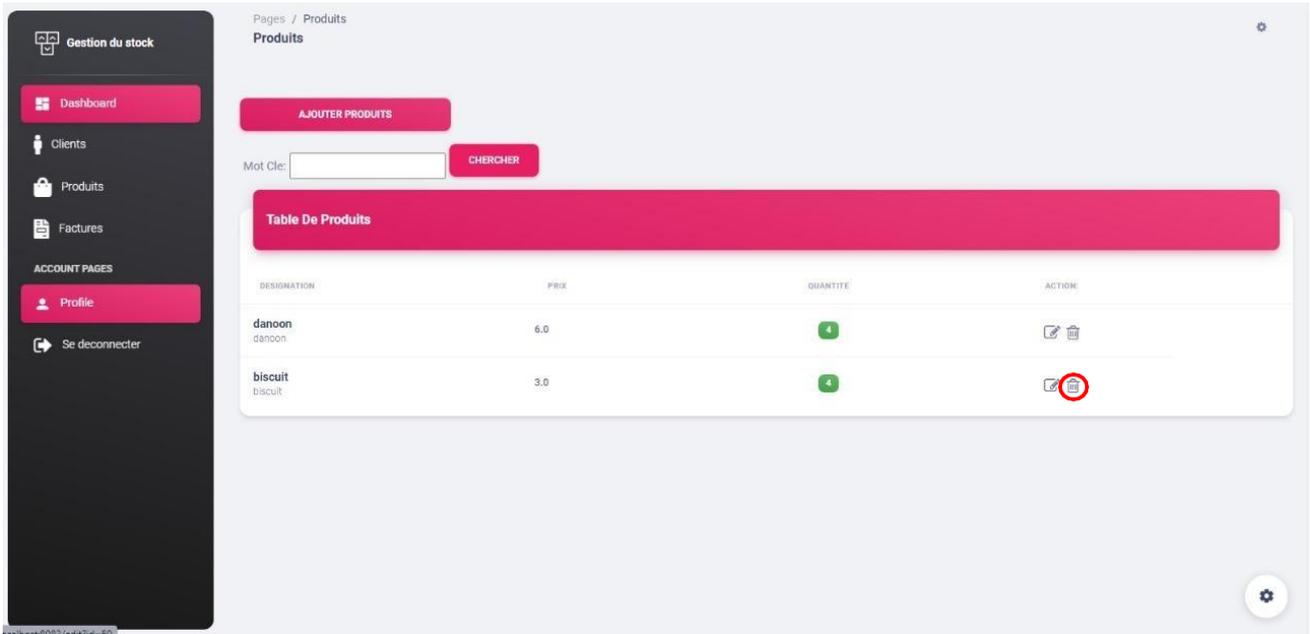


Figure 38 : Supprimer produit

6- Gestion des inventaires

6-1 Visualisation

Cette interface permet de consulter la liste des inventaires.

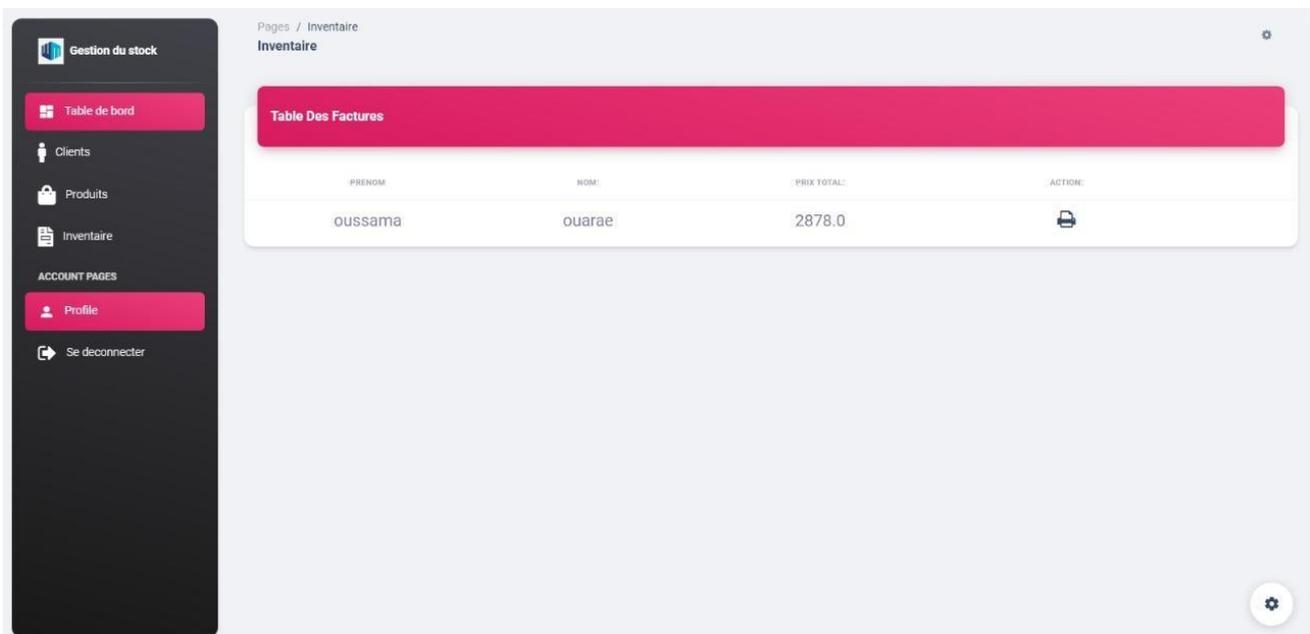


Figure 39 : afficher inventaires

6-2 imprimer inventaire en pdf

Comme une étape finale, ce fichier permet d'afficher les informations du client ainsi que le prix de ses produits.

Inventaire :

Nom : oussama
Prenom : ouarae
Email : oussama2020@gmail.com
Telephone : 0607998072

Designation	Quantité	Prix
ecran	7	400.0
table	7	9.0
chaise	3	5.0

Prix Total : 2878.0

Figure 40 : pdf a imprime

7-profile

Cette interface permet d'afficher les informations personnelles du responsable d'inventaire.

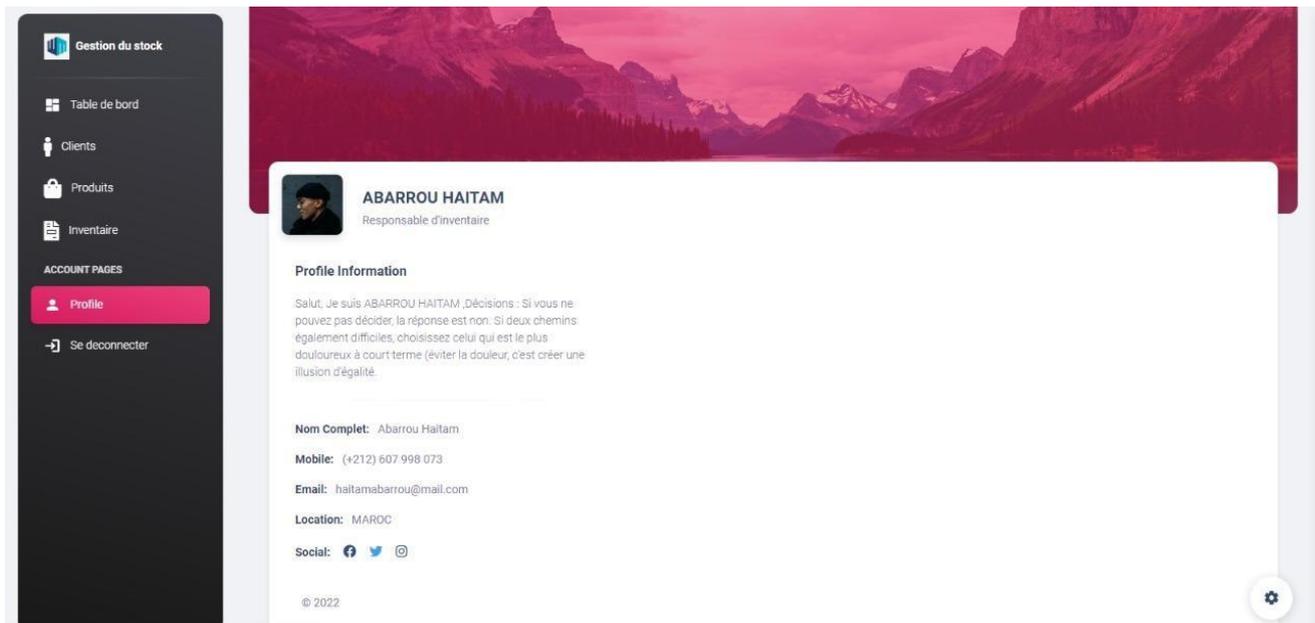


Figure 41 : profile

Conclusion et perspectives

Mon travail durant ce stage consistait à étudier et à développer une application web de gestion de stock, dans le but de répondre aux besoins de l'agence **WebMarko** afin de trouver une solution de la problématique et de faciliter le travail des personnels de l'agence

Ce rapport de synthèse présente mon projet intitulé « Application web de gestion des stocks pour responsable d'inventaire » qui a pour but, la mise en place d'une solution de suivi des stocks et la gestion de leurs inventaires, réalisé durant la période de stage au sein d'Agence WebMarko, service informatique, dans lequel, j'ai appris à évoluer en répondant aux exigences et aux contraintes techniques, et en inculquant des notions et des habitudes du professionnalisme.

Le présent projet réalisé au cours de mon stage de fin d'étude a été très bénéfique tant au niveau personnel qu'au niveau académique. Ainsi, il m'a permis d'approfondir mes connaissances en termes d'outils de programmation et de conception . Il a été une opportunité pour développer de nouvelles compétences dans la programmation avec Spring Boot qui est devenue très utilisée dans le développement informatique. Ce stage m'a aussi offert la chance de découvrir et travailler dans un contexte réel et affronter de vraies problématiques. Ces dernières m'ont aidé à acquérir une expérience très intéressante et très riche.

Ce travail répond aux besoins préalablement fixés mais il pourra évidemment être amélioré et optimisé par l'ajout de nouvelles fonctionnalités comme la gestion des erreurs et des bugs; c'est-à-dire, avoir un suivi du projet réalisé même après sa validation.

A long terme, afin de mieux développer l'application réalisée il est prévu comme perspectives :

- ✓ serveur/client : client peut accéder à l'application pour visualiser son stock via une session privé .
- ✓ Un outil pour analyser les statistiques des stocks .

Webographie

♣ STACKOVERFLOW :

<https://stackoverflow.com/> dernière consultation 2022/06/24

♣ BOOTSTRAP :

<https://bootstrap.com/> dernière consultation 2022/05/30

♣ SPRING BOOT:

<https://springboot.io/> dernière consultation 2022/06/19

♣ THYMELEAF :

[https://thymeleaf.org /](https://thymeleaf.org/) dernière consultation 2022/06/20

♣ GITHUB :

<https://github.com/> dernière consultation 2022/06/10

♣ CODEJAVA :

<https://codejava.net/> dernière consultation 2022/06/22

♣ chaine youtube :

<https://www.youtube.com/c/CodeJava> dernière consultation 2022/06/20