



**ECOLE MAROCAINE DES
SCIENCES DE L'INGENIEUR**
Membre de
HONORIS UNITED UNIVERSITIES



Rapport
4^{ème} année
Ingénierie Informatique et Réseaux

Sous le thème

**PLATEFORME DE GESTION DE PRODUITS
ET DE STOCK : SOLUTION INTEGREE
POUR LA GESTION DES TRANSACTIONS
ET DES CATEGORIES**

Réalisé par :

Ayoub Majjid
El Hilali Aymen
Ben Yamna Salma

Encadré par :

Mme. Kawtar Imgharene

DÉDICACE



Nous dédions ce travail à notre famille, dont le soutien inébranlable, l'amour et l'encouragement ont été notre boussole tout au long de ce projet. Leur présence à nos côtés a constitué une véritable source d'inspiration et de force. À nos parents, qui ont cru en nos capacités et nous ont toujours incités à viser l'excellence, nous exprimons notre plus profonde gratitude.

À nos amis et collègues, qui ont partagé avec nous les défis et les réussites de cette aventure, nous vous remercions pour votre camaraderie, votre patience et votre soutien indéfectible. Votre présence a rendu cette expérience plus enrichissante et mémorable.

Nous dédions également ce travail à nos professeurs et encadrants, dont les conseils avisés, les encouragements et le partage de connaissances ont joué un rôle clé dans la réalisation de ce projet. Leur dévouement et leur expertise ont été essentiels pour surmonter les défis et atteindre nos objectifs.

Enfin, à toutes les personnes qui, directement ou indirectement, ont contribué à cette réussite, nous exprimons notre gratitude la plus sincère. Ce projet est le reflet des efforts collectifs et des soutiens constants reçus tout au long de notre parcours.

REMERCIEMENTS

Nous tenons à exprimer notre gratitude la plus profonde envers toutes les personnes qui ont contribué à la réalisation de ce projet de **gestion de produits**.

Tout d'abord, nous remercions chaleureusement nos parents pour leur soutien inconditionnel. Leur confiance, leur patience et leurs encouragements constants ont été une source d'inspiration et de motivation tout au long de cette aventure. Merci d'avoir toujours cru en nous et de nous avoir soutenus dans nos ambitions.

Nous souhaitons également exprimer notre reconnaissance envers nos camarades et collègues pour leurs idées, leurs suggestions constructives et leur esprit de collaboration. Leur contribution a permis d'enrichir le développement de ce projet et d'en améliorer la qualité.

Enfin, nos remerciements les plus sincères vont à notre encadrant. Votre expertise, vos conseils avisés et votre disponibilité constante ont été d'une grande valeur pour le succès de ce projet. Merci pour votre encadrement rigoureux, vos retours pertinents et votre implication. Votre soutien a été essentiel pour mener ce projet à terme.

À tous, nous adressons nos remerciements les plus sincères pour votre aide précieuse.

RÉSUMÉ



Introduction :

La gestion efficace des produits et des stocks est un enjeu clé pour de nombreuses entreprises, mais les solutions disponibles sur le marché ne répondent pas toujours aux besoins spécifiques des structures au Maroc. Pour combler cette lacune, nous avons développé une application innovante de gestion de produits. Simple d'utilisation, cette application permet de gérer efficacement les produits, les ventes, les achats, les stocks et les catégories. Grâce à des fonctionnalités avancées comme le filtrage et une interface intuitive, elle offre une solution complète pour optimiser les processus de gestion. Notre application contribue à améliorer la productivité des entreprises en leur permettant de suivre leurs transactions en temps réel, de minimiser les erreurs, et de simplifier la prise de décisions stratégiques. Elle constitue ainsi un outil essentiel pour les entreprises souhaitant moderniser leurs systèmes de gestion.

.

Objectifs du Projet :

1. **Idée du Projet :** Le but de ce projet est de développer un système innovant permettant aux entreprises de gérer efficacement leurs produits, catégories, ventes, achats et stocks. Ce système vise à simplifier les opérations de gestion tout en offrant des fonctionnalités de filtrage et d'analyse pour une prise de décision optimale.
2. **Conception :** La conception du projet comprend plusieurs étapes clés, allant de l'analyse des besoins à la mise en œuvre finale. offrant une interface intuitive et performante. Les outils utilisés incluent des bases de données pour gérer les tables des produits, des ventes, des achats, des stocks et des catégories.
3. **Affichage des Pages :** quelques pages de l'application.

Résultats et Perspectives :

Ce projet m'a permis de combiner théorie et pratique, en développant des compétences essentielles en gestion des systèmes et en conception d'applications. Grâce à cette expérience, j'ai pu comprendre les enjeux liés à la gestion des produits, des stocks, des ventes, des achats et des catégories, tout en abordant les aspects techniques du développement de l'application.

Pour l'avenir, je souhaite continuer à faire évoluer cette plateforme en y intégrant des fonctionnalités supplémentaires. L'ajout d'outils de gestion plus personnalisés, de rapports détaillés et de mécanismes d'analyse prédictive permettra d'améliorer encore l'efficacité de la gestion des produits. Ces évolutions offriront une expérience plus enrichissante et intuitive pour les utilisateurs, tout en optimisant les processus internes.

ABSTRACT

Introduction:

Efficient management of products and inventory is crucial for many businesses, yet existing solutions often fail to meet the specific needs in Morocco. To address this gap, we have developed an innovative product management application. User-friendly and intuitive, it enables businesses to manage their products, sales, purchases, stock levels, and categories effectively. With powerful filtering tools and real-time tracking, our app simplifies business operations while optimizing decision-making. This solution is perfect for businesses seeking to modernize their product management processes and improve overall efficiency.

Project Objectives:

1. Project Idea: The goal of this project is to develop a system that allows businesses to efficiently manage their products, prices, stock levels, sales, and purchases. By providing real-time updates and automated tracking, the system aims to streamline operations, minimize errors, and optimize decision-making. This solution is designed to help businesses manage their inventory more effectively while offering features like advanced filtering to enhance the user experience and drive business growth.
2. Design: the various design steps, technological choices, and tools used.
3. Page Display: some pages of the application.

Results and Future Perspectives:

This project allowed me to combine theory and practice, enabling me to develop essential skills in system management and application development. I gained hands-on experience in managing products, sales, purchases, and stock levels. Looking ahead, I plan to continue evolving this platform by integrating new features, such as advanced reporting tools, automation for stock management, and predictive analytics. These improvements will optimize decision-making and streamline operations, offering businesses an even more efficient and intuitive product management solution.

TABLE DES MATIERES

Table of Contents

DÉDICACE	2
REMERCIEMENTS	3
RÉSUMÉ.....	4
ABSTRACT	6
TABLE DES MATIERES.....	7
TABLE DES FIGURES	9
LISTE DES ABREVIATIONS	10
INTRODUCTION	11
PROBLÉMATIQUE	13
.....	13
ORGANISATION DU MEMOIRE	11
.....	14
I- CHAPITRE 1:	16
ANALYSE ET CONCEPTION	16
.....	16
Introduction :	17
1 Périmètre du projet :	17
1.1 But	17
1.2 Missions.....	17
1.3 Réalisation	17
2 Etude de l'existant :	18
3 Etude fonctionnelle :	18

3.1	Objectifs fonctionnels	18
3.2	Besoins fonctionnels	18
4	Modélisation UML :	18
4.1	Cas d'utilisation :	19
4.2	Diagramme de séquence :	21
4.3	Conclusion :	22
5	Structure de la Base de Données	23
II-	CHAPITRE 2:	23
	RÉALISATION.....	24
3.1	Environnement du Travail :	24
3.2	Cycle de Développement :	27
3.3	Déroulement du Développement :	28
3.4	Documentation de l'API avec Postman :	29
III -	CHAPITRE 3 :	32
	INTERFACES DE L'APPLICATION	31-34
	CONCLUSION	35
1.	Réalisation Technique.....	35
2.	Cycle de Développement	35
3.	Interfaces de l'Application.....	35
4.	Documentation et Tests	35
5.	Perspectives.....	Error! Bookmark not defined.
	WEBOGRAPHIE.....	36

TABLE DES FIGURES

FIGURE 1: DIAGRAMME DE CAS D'UTILISATION	19
FIGURE 5: DIAGRAMME DE SEQUENCE "CONSULTER PAGE"	21
FIGURE 6: DIAGRAMME DE SEQUENCE "PAGE LOGIN "	22
FIGURE 10: LOGO DE MICROSOFT SQL SERVER REPRESENTANT LE SYSTEME DE GESTION DE BASE DE DONNEES RELATIONNELLE.	23
FIGURE 7:STRUCTURE DE BASE DE DONNES	23
FIGURE 8 :LOGO DE ANGULAR REPRESENTANT LA BIBLIOTHEQUE JAVASCRIPT.....	24
FIGURE 9: LOGO DE .NET, REPRESENTANT L'ENVIRONNEMENT D'EXECUTION	25
FIGURE 11:LOGO DE GITHUB, ILLUSTRANT LA PLATEFORME UTILISEE POUR LA GESTION De VERSION ET LA COLLABORATION SUR LE CODE.	26
FIGURE 12: LOGO DE POSTMAN, MONTRANT L'OUTIL UTILISE POUR LE TEST DES API.....	26
FIGURE 13: LOGO DE VISUAL STUDIO REPRESENTANT L'EDITEUR DE CODE UTILISE DANS LE PROJET	26
FIGURE 15: LOGO DE POSTMAN ILLUSTRANT L'INTERFACE DE CREATION ET D'ORGANISATION DES COLLECTIONS POUR LES REQUETES API.	26
FIGURE 16: HOME PAGE	31
FIGURE 17: STOCK PAGE.....	32
FIGURE 18: PURSHASE PAGE	33
FIGURE 19: SALES PAGE	34
FIGURE 20: CATEGORIES PAGES	34

LISTE DES ABREVIATIONS

Abréviation	Signification
EMSI	<u>École Marocaine des Sciences de l'Ingénieur</u>
IIR	Ingénierie Informatique et Réseaux
UML	Unified Modeling Language
API	Application Programming Interface
VSCode	Visual Studio Code
Git	Version Control System
GitHub	Platform for Version Control and Collaboration
Postman	API Testing Too
Figma	Cloud-Based UI Design Tool
.NET	Framework de développement multiplateforme.
Angular	JavaScript library for building user interfaces
Microsoft SQL Server	Pour stockage de la base de données

INTRODUCTION

L'industrie de la gestion de produits a connu une transformation significative ces dernières années, avec l'évolution des systèmes de gestion et l'intégration des outils numériques. Au Maroc, bien que de nombreuses entreprises aient progressivement adopté ces nouvelles technologies, plusieurs défis demeurent pour améliorer l'efficacité de la gestion des produits, des ventes et des stocks, ainsi que pour enrichir l'expérience utilisateur. La digitalisation des processus de gestion de produits permet aujourd'hui aux entreprises de gérer plus facilement les tarifs, les produits, les ventes et les achats, tout en bénéficiant d'une vue d'ensemble claire et organisée des stocks. Cependant, malgré ces avancées, plusieurs défis subsistent, notamment la nécessité d'une plateforme simple, intuitive et capable de gérer les produits, les catégories, les ventes et les achats en temps réel. La gestion des stocks, des promotions et des interactions entre les utilisateurs et les administrateurs reste une problématique clé pour garantir une expérience fluide et optimale.

Dans le cadre de mon projet, nous avons développé une application innovante dédiée à la gestion des produits et des processus associés, permettant de simplifier et améliorer l'expérience de gestion des produits, des ventes, des achats, des stocks et des catégories. Ce système, développé avec .NET, inclut des fonctionnalités de gestion avancées, telles que la création d'un compte administrateur (manuel) et un tableau de bord pour suivre et analyser les informations en temps réel.

Notre application est conçue pour faciliter la gestion des produits tout en offrant une expérience utilisateur optimisée. Elle aide les administrateurs et les utilisateurs à gérer efficacement leurs produits, ventes, achats, et stocks. Plusieurs fonctionnalités clés sont intégrées dans l'application, telles que :

Authentification : Assurer un accès sécurisé pour les administrateurs et les utilisateurs via un système de connexion sécurisé. L'authentification permet la gestion des rôles, comme la création d'un compte administrateur de manière manuelle.

Gestion des Produits : Permettre aux utilisateurs de gérer les produits en ajoutant, modifiant ou supprimant des articles, en associant chaque produit à une catégorie spécifique pour faciliter le tri et l'affichage.

Gestion des Ventes : Offrir un système de suivi des ventes, permettant aux utilisateurs de consulter les ventes passées et d'analyser les performances en temps réel.

Gestion des Achats : Permettre aux administrateurs de suivre les achats de produits, avec un tableau de bord pour gérer les commandes et les retours de manière efficace.

Gestion des Stocks : Assurer une gestion optimisée des stocks en temps réel, permettant de suivre les niveaux de produits, les alertes de réapprovisionnement et la gestion des inventaires.

Filtrage Avancé : Offrir aux utilisateurs la possibilité de filtrer les produits selon diverses catégories, prix, disponibilité et autres critères, afin de faciliter la recherche.

PROBLÉMATIQUE



Dans le cadre de ce projet, la problématique centrale est la suivante :

Comment concevoir une plateforme de gestion de produits simple, intuitive et sécurisée, permettant aux utilisateurs (administrateurs, hôteliers) de gérer les produits, ventes, achats et stocks tout en garantissant une expérience utilisateur fluide ?

Il s'agit également de permettre aux administrateurs et hôteliers de gérer efficacement les prix, la disponibilité des produits, les achats et les promotions, tout en offrant aux utilisateurs un accès rapide et sécurisé aux fonctionnalités essentielles. Une gestion précise des informations (prix, produits, stocks) et des droits d'accès adaptés pour chaque catégorie d'utilisateur (administrateur, hôtelier, client) constitue un défi majeur pour garantir une plateforme performante et fiable.

ORGANISATION DU MEMOIRE

Chapitre 1 : analyse et Conception

Dans cette partie, nous détaillons la phase de conception du projet, comprenant les diagrammes UML et la conception de la base de données. Ces éléments sont cruciaux pour structurer et organiser le projet avant son implémentation.

2.1 Diagrammes UML

Les diagrammes UML (Unified Modeling Language) permettent de visualiser et de planifier les différentes interactions et fonctionnalités de la plateforme. Voici les principaux diagrammes utilisés :

2.2 Conception de la base de données

La conception de la base de données a été réalisée en prenant en compte les besoins Fonctionnels du système.

Chapitre 2 : Réalisation

Ce chapitre documente les étapes clés de la réalisation du projet, couvrant tout le processus depuis la création initiale jusqu'au déploiement final. Il est structuré comme suit :

1. Environnement du Travail :

- **Angular** : Bibliothèque JavaScript utilisée pour construire l'interface utilisateur dynamique et interactive de l'application.
- **.NET**: Environnement d'exécution JavaScript utilisé pour développer le backend, gérer les requêtes HTTP et interagir avec la base de données.
- **Microsoft SQL Server** : Système de gestion de base de données relationnelle utilisé pour stocker et gérer les données de l'application.
- **Git/GitHub** : Outils pour la gestion du code source, le suivi des versions, et la collaboration entre les développeurs
- **Postman** : Utilisé pour tester les API et vérifier le bon fonctionnement des endpoints.

- **VSCode** : Éditeur de code principal pour le développement du projet, offrant des fonctionnalités telles que l'auto-complétion et le débogage.

2. Cycle de Développement :

- **Analyse du Projet** : Identification des besoins et définition des objectifs.
- **Recherche et Conception** : Étude sur les plateformes MOOC et conception à l'aide de diagrammes UML pour structurer le système.
- **Initialisation du Référentiel** : Création du projet sur GitHub et sélection des technologies appropriées (Angular pour le front, .NET pour le back-end, Microsoft SQL Server pour la base de données).
-

3. Déroulement du Développement :

- **Création des Endpoints API** : Développement des endpoints pour les fonctionnalités du projet.
- **Tests avec Postman** : Vérification du bon fonctionnement des endpoints.
- **Implémentation du Design** : Intégration du design en Angular.
- **Connexion Front-End et Back-End** : Mise en place de la communication entre le front-end et le back-end.
- **Tests Finaux** : Validation de toutes les fonctionnalités pour garantir la conformité avec les attentes.

4. Documentation de l'API avec Postman :

- **Création et Organisation des Collections** : Documentation des endpoints dans Postman, facilitant la gestion et l'utilisation des API.
-



ECOLE MAROCAINE DES
SCIENCES DE L'INGENIEUR

Membre de

HONORIS UNITED UNIVERSITIES



I- CHAPITRE 1:

ANALYSE ET CONCEPTION

Introduction :

Ce chapitre présente l'analyse et la conception du projet de gestion de produits pour la plateforme de gestion commerciale, incluant la modélisation des rôles des utilisateurs, la gestion des produits, des ventes, des achats et des stocks, ainsi que la structure de la base de données. Ces éléments sont essentiels pour garantir une organisation efficace avant l'implémentation, permettant ainsi de proposer une gestion optimisée des produits et des opérations commerciales.

1 Périmètre du projet :

1.1 But

L'objectif de ce projet est d'étudier, concevoir et réaliser un système de gestion des produits et des opérations commerciales pour une plateforme de gestion, permettant de simplifier et optimiser la gestion des produits, des ventes, des achats, des stocks et des catégories.

1.2 Missions

- Etude technique
- Etude fonctionnelle
- Conception
- Réalisation
- Déploiement
- Documentation

1.3 Réalisation

1.3.1 Contrainte technique

- Développement d'une application .NET utilisant Microsoft SQL Server comme base de données.
- Exploitation des API de Angular pour le frontend.

1.3.2 Contrainte de Sécurité

- Cryptage des données sensibles.
- Hachage des mots de passe.
- Utilisation de JWT pour l'authentification.

1.3.3 Contrainte de délais

Durée de 2 mois, incluant la gestion des risques.

1.3.4 Contrainte de maintenance

- Fournir une maintenance complète et assurer une bonne scalabilité.

2 Etude de l'existant :

2.1. Concept métier

- Gestion des produits.
- Gestion des ventes .
- Gestion des achats.
- Facturation de stock.
- Gestion des permissions.

2.2. Logiciels similaires

- Shopify
- Magento

3 Etude fonctionnelle :

3.1 Objectifs fonctionnels

- Simplifier la gestion des produits et des stocks pour les administrateurs .
- Optimiser la gestion des ventes et des achats.
- Assurer une gestion efficace des stocks
- Améliorer l'expérience utilisateur avec des filtres avancés.

3.2 Besoins fonctionnels

3.2.1. *Bloc fonctionnel : Gestion des utilisateurs :*

En tant qu'Admin :

- Lister les utilisateurs.
- Filtrer les utilisateurs.
- Mettre à jour un utilisateur.
- Supprimer un utilisateur.

3.2.2. *Bloc fonctionnel : gestion des compte :*

En tant qu'Admin :

- Valider une demande de création d'un compte.
- Supprimer un compte.
- Mettre à jour un compte.
- Lister les compte.
- Filtrer les compte.

3.2.3. *Bloc fonctionnel : Gestion des catégories:*

En tant qu'Admin :

- Lister les catégories.
- Supprimer une catégorie.
- Filtrer les catégories.

En tant que Gestionnaire:

- Lister ses propres catégories.
- Ajouter, mettre à jour ou supprimer ses propres catégories.

En tant qu'Utilisateur :

- Lister les catégories disponibles.
- Filtrer les catégories..

3.2.4. Bloc fonctionnel : Gestion des produits:

En tant qu'Admin :

- Lister les produits.
- Mettre à jour un produit.
- Supprimer un produit.
- Filtrer les produits

En tant que Gestionnaire:

- Lister ses propres produits..
- Ajouter, mettre à jour ou supprimer ses produits .
- Filtrer ses produits .

En tant qu'Utilisateur :

- Lister et filtrer les produits.
- Filtrer les produits.

3.2.5. Bloc fonctionnel Gestion des commandes :

En tant qu'Admin :

- Lister les commandes.
- Valider ou annuler une commande.
- Filtrer les commandes.
- Générer une facture.

En tant que gestionnaire :

- Lister ses propres commandes.
- Filtrer ces commandes.

En tant que gestionnaire :

- lister et filtrer ses commandes.
- Annule une commande.
- Imprimer une facture

3.2.6. Bloc fonctionnel : Gestion des paiements :

En tant qu'Admin :

- Lister les paiements mensuels des produits.
- Générer une facture .
- Filtrer les paiements.
-

En tant que Gestionnaire :

- Lister ses paiements mensuels.
- Générer et payer une facture.

3.2.7. Bloc fonctionnel : Gestion des stocks :

En tant qu'Admin :

- Lister les stocks.
- Mettre à jour le stock.
- Supprimer un produit du stock.

En tant que Gestionnaire:

- Lister ses stocks.
- Mettre à jour ou supprimer un produit du stock.

3.2.8. Bloc fonctionnel : Gestion des profils :

En tant qu'Admin :

- Lister le profil
- Mettre à jour le profil

En tant qu'Utilisateur :

- Lister le profil
- Mettre à jour le profil

3.2.9. Bloc fonctionnel : gestion d'authentification :

En tant qu'Admin :

- Login
- Sign-out

En tant qu'Hotel :

- Register
- Login
- Sign-out

En tant qu'Utilisateur :

- Register
- Sign-out
- Login

4 Modélisation UML :

Les diagrammes UML (Unified Modeling Language) permettent de visualiser les interactions entre les différents éléments de l'application. Les principaux diagrammes utilisés incluent les cas d'utilisation, les diagrammes de séquence, et le diagramme de classes. Ces diagrammes décrivent les fonctionnalités telles que la gestion des produits, des ventes, des achats, du stock et des catégories, ainsi que les relations entre les tables Produits, Sales, Purchases, Stock et Categories .

Vue Fonctionnelle

La vue fonctionnelle utilisera des diagrammes de cas d'utilisation pour représenter les interactions entre les différents acteurs (principalement l'administrateur) et les fonctionnalités offertes par l'application, telles que la gestion des produits, des ventes, des achats, du stock et des categories .

Vue Dynamique

Les diagrammes de séquence seront utilisés pour illustrer le flux d'actions dans l'application, montrant comment l'administrateur interagit avec le système pour gérer les produits, enregistrer les ventes, suivre les achats et mettre à jour le stock.

4.1 Cas d'utilisation :

Le diagramme de cas d'utilisation illustre les principales fonctionnalités et les interactions entre l'administrateur et le système. Voici les cas d'utilisation clés : gestion des produits, gestion des ventes, gestion des achats, gestion du stock et gestion des categories :

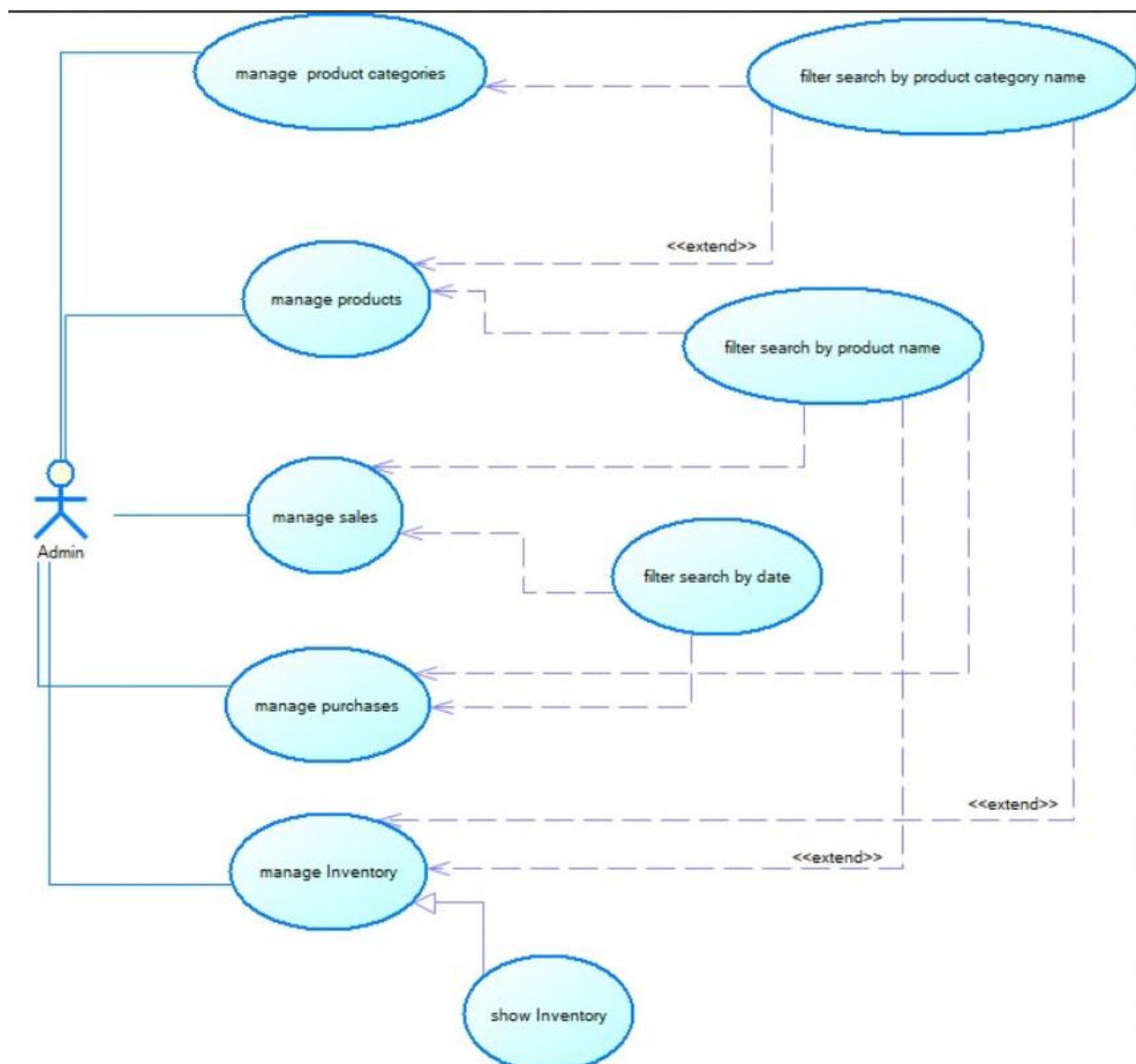


Figure 1: Diagramme de cas d'utilisation

Les figures ci-dessus illustrent les différents cas d'utilisation du système à travers les interactions entre l'administrateur et les fonctionnalités principales de l'application. Voici une analyse détaillée basée sur le diagramme et le rôle de l'administrateur dans la gestion des produits, des ventes, des achats, du stock et des catégories.

Gestion des utilisateurs :

- **Admin** : Les administrateurs à le rôle le plus élevé dans la hiérarchie des utilisateurs. C'est le seul acteur qui possède tous les droits de la gestion. Il peut :
 - >Gérer les produits, les ventes, les achats, le stock et les catégories.
 - >Ajouter, modifier ou supprimer des données dans les tables principales
- **Utilisateur normal** : Les utilisateur standards (hotels & guests) interagissent principalement avec le contenu de l'application. Ils ont la possibilité de :
 - Visualiser les produits disponibles.
 - Consulter les détails des produits (prix, catégorie, stock).
 - Utiliser les fonctionnalités de filtrage pour trouver des produits spécifiques.

Conclusion :

Le diagramme des cas d'utilisation montre clairement les différentes interactions et permissions des acteurs principaux (Admin et utilisateur normal) sur l'application. Il met en évidence la gestion hiérarchique des données, en distinguant bien les actions possibles selon les rôles. Les admins gèrent les données, tandis que les utilisateurs interagissent principalement avec les fonctionnalités de consultation.

4.2 Diagramme de séquence :

Le diagramme de séquence illustre le flux d'interactions entre les différents acteurs et les composants du système pour diverses actions.

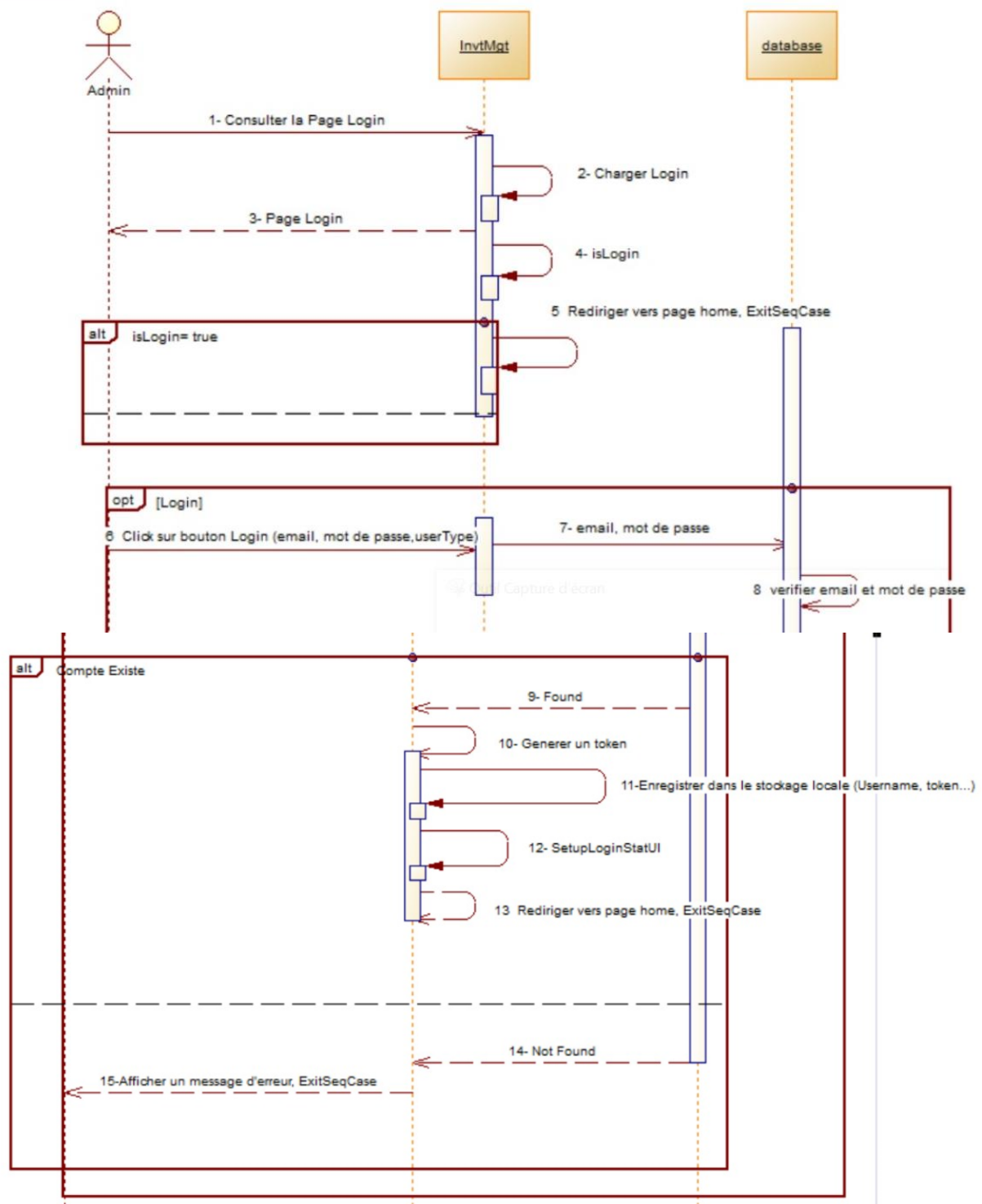


Figure 2: Diagramme de séquence "Page Login "

Ce diagramme illustre le processus de connexion d'un utilisateur en saisissant ses identifiants et en accédant à son compte.

4.3 Conclusion :

Ces exemples de diagrammes de séquence illustrent une variété d'interactions entre l'administrateur et les fonctionnalités principales du système, telles que la gestion des produits, des ventes, des achats, du stock et des catégories.

5 Structure de la Base de Données

Pour la gestion de l'application, nous avons utilisé Microsoft SQL Server, un système de gestion de base de données relationnelle (RDBMS), pour stocker et gérer les informations relatives aux produits, aux ventes, aux achats, au stock et aux catégories. La structure suivante présente les tables principales utilisées dans la base de données :



Figure 7: Structure de la base de données



ECOLE MAROCAINE DES
SCIENCES DE L'INGENIEUR
Membre de
HONORIS UNITED UNIVERSITIES



II- CHAPITRE 2:

RÉALISATION

Ce chapitre décrit les étapes principales de la réalisation du projet, depuis la création initiale jusqu'à son déploiement.

3.1 Environnement du Travail :

ReactJS



Figure 4 :Logo de Angular représentant la bibliothèque JavaScript

Utilisé pour construire des interfaces utilisateur dynamiques.

- **Description** : Angular est une bibliothèque JavaScript utilisée pour créer des interfaces utilisateurs interactives et dynamiques. Elle permet de construire des composants réutilisables et de gérer efficacement l'état de l'application.
- **Rôle dans le Projet** : Angular a permis de développer une interface utilisateur fluide et réactive, facilitant l'interaction avec les utilisateurs et améliorant l'expérience globale de navigation sur l'application.

Node.Js



Figure 5: Logo de .net , représentant le framework de développement

- **Description** .NET est un framework de développement polyvalent et performant, permettant de créer des applications web, desktop et mobiles. Il est particulièrement adapté pour gérer les requêtes HTTP, la logique métier et les interactions avec les bases de données.
- **Rôle dans le Projet** NET a permis de créer le backend de l'application, gérer les requêtes de l'administrateur, traiter la logique métier (gestion des produits, ventes, achats, stock) et interagir avec la base de données.

Microsoft SQL Server



Figure 6: Logo de Microsoft SQL Server représentant le système de gestion de base de données relationnelle.

- **Description** : Microsoft SQL Server est un système de gestion de base de données relationnelle (SGBDR) qui permet de stocker, organiser et interroger des données. Il utilise le langage SQL pour la gestion des données et des transactions.
- **Rôle dans le Projet** Microsoft SQL Server a été utilisé pour gérer et stocker les informations relatives aux produits, aux ventes, aux achats, au stock et aux catégories, garantissant une gestion sécurisée et efficace des données.

Git/GitHub



Figure 7: Logo de GitHub, illustrant la plateforme utilisée pour la gestion De version et la collaboration sur le code.

- **Description** : Git est un système de contrôle de version qui permet de suivre les modifications du code source. GitHub est une plateforme de collaboration basée sur Git.

- **Rôle dans le Projet** : Git et GitHub ont été utilisés pour gérer le code source, collaborer avec les membres de l'équipe, et suivre les versions du projet.

Postman



Figure 8: Logo de Postman, montrant l'outil utilisé pour Le test des API.

- **Description** : Postman est un outil utilisé pour tester les API en envoyant des requêtes HTTP et en inspectant les réponses.
- **Rôle dans le Projet** : Postman a été utilisé pour tester les endpoints de l'API, s'assurer qu'ils fonctionnent correctement, et vérifier les réponses aux requêtes.

VSCode



Figure 9: Logo de Visual Studio représentant L'éditeur de code utilisé dans le projet

- **Description** : Visual Studio Code est un éditeur de code source développé par Microsoft. Il offre une gamme d'extensions et de fonctionnalités pour le développement web.
- **Rôle dans le Projet** : VSCode a été utilisé comme principal éditeur de code, offrant des fonctionnalités telles que l'auto-complétion, le débogage, et l'intégration avec Git.

3.2 Cycle de Développement :

1. **Analyse du Projet** : La première étape a consisté à analyser les besoins du projet et à définir les objectifs.
2. **Conception en Utilisant UML** : La conception du projet a été réalisée à l'aide de diagrammes UML pour une meilleure structuration et une compréhension claire du système.
 - **2.1 Diagramme de Cas d'Utilisation** : Permet de décrire les interactions entre les utilisateurs et le système.
 - **2.2 Diagramme de Séquence** : Détaille les interactions entre les différents composants du système dans le temps.
 - **2.3 Conception de la Base de Données** : Conception des schémas de la base de données et définition des structures nécessaires pour stocker les données.
3. **Initialisation d'un Nouveau Référentiel GitHub** : Un nouveau projet a été créé sur GitHub pour gérer le code source. L'éditeur de texte VSCode a été utilisé pour le développement.
4. **Choix des Technologies** : Les technologies suivantes ont été sélectionnées :
 - **Front-End** : Angular pour la création de l'interface utilisateur dynamique et interactive.
 - **Back-End** : .net pour le développement du backend, la gestion des requêtes serveur et l'intégration avec la base de données.
 - **Base de Données** : Microsoft SQL Server pour la gestion des données relationnelles et structurées.
5. **Utilisation de Figma pour le Design** : Figma a été utilisé pour concevoir le design des interfaces, facilitant la création de maquettes et de prototypes.

3.3 Déroulement du Développement :

- **Création des Endpoints** : Une série d'endpoints API a été développée pour les différentes pages ou sections du projet.
- **Tests des Endpoints avec Postman** : Chaque endpoint a été testé pour vérifier son bon fonctionnement et s'assurer qu'il répond aux exigences définies.
- **Implémentation du Design ReactJS**: Après avoir validé les endpoints, le design a été intégré en utilisant ReactJS.
- **Connexion Front-End et Back-End avec JS** : L'interaction entre le front-end et le back-end a été établie à l'aide de JavaScript pour assurer une communication fluide.
- **Tests Finaux** : Des tests approfondis ont été réalisés pour garantir que toutes les fonctionnalités sont opérationnelles et que le système répond aux attentes.

Ce cycle de développement a été répété plusieurs fois pour affiner et améliorer le système, en itérant sur les fonctionnalités et en résolvant les problèmes rencontrés tout au long du processus.

3.4 Documentation de l'API avec Postman :

La documentation de l'API est essentielle pour assurer une compréhension claire et une utilisation correcte des endpoints par les développeurs et les intégrateurs. Postman, un outil populaire pour tester les API, a été utilisé pour documenter les endpoints du projet. Cette documentation aide à la fois les développeurs à tester les API et les utilisateurs finaux à comprendre comment interagir avec le système.

3.4.1 Création et Organisation des Collections

- **Collections Postman** : Les endpoints de l'API ont été organisés en collections dans Postman. Une collection est un ensemble de requêtes API regroupées pour faciliter leur gestion et leur utilisation.



Figure 10: Logo de Postman illustrant l'interface de création et d'organisation des collections pour les requêtes API.

- **Structure des Collections** : Chaque collection contient des requêtes classées par fonctionnalité ou module de l'application, telles que l'authentification, la gestion des utilisateurs, et les opérations sur les formations.

3.4.2 Documentation des Endpoints

- **Endpoints Décrits** : Chaque requête dans Postman a été documentée avec des détails clés, incluant :
 - **URL de l'Endpoint** : Le chemin de l'API pour accéder à la ressource.
 - **Méthode HTTP** : La méthode utilisée (GET, POST, PUT, DELETE, etc.).
 - **Paramètres** : Les paramètres requis et optionnels pour chaque requête
 - **Corps de la Requête** : La structure du corps de la requête pour les méthodes qui nécessitent des données envoyées.
 - **Réponses** : Les exemples de réponses attendues, y compris les codes de statut HTTP et les structures de données retournées.

Voici le lien pour accéder à la documentation de notre projet :

<https://documenter.getpostman.com/view/34697773/2sAYJ9Ae1u>



**ECOLE MAROCAINE DES
SCIENCES DE L'INGENIEUR**
Membre de
HONORIS UNITED UNIVERSITIES



III - CHAPITRE 3 :

INTERFACES DE L'APPLICATION

Cette section présente les différentes interfaces de l'application, illustrant comment chaque page est conçue et intégrée dans le projet. Les captures d'écran fournies montrent le design et les fonctionnalités de chaque page.

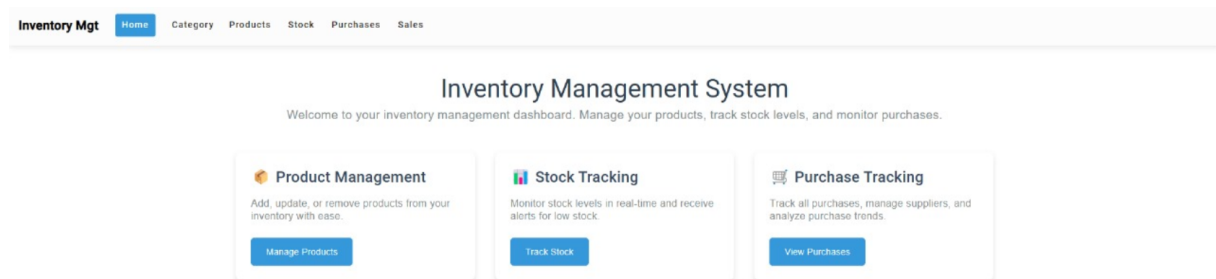


Figure 11: Home page

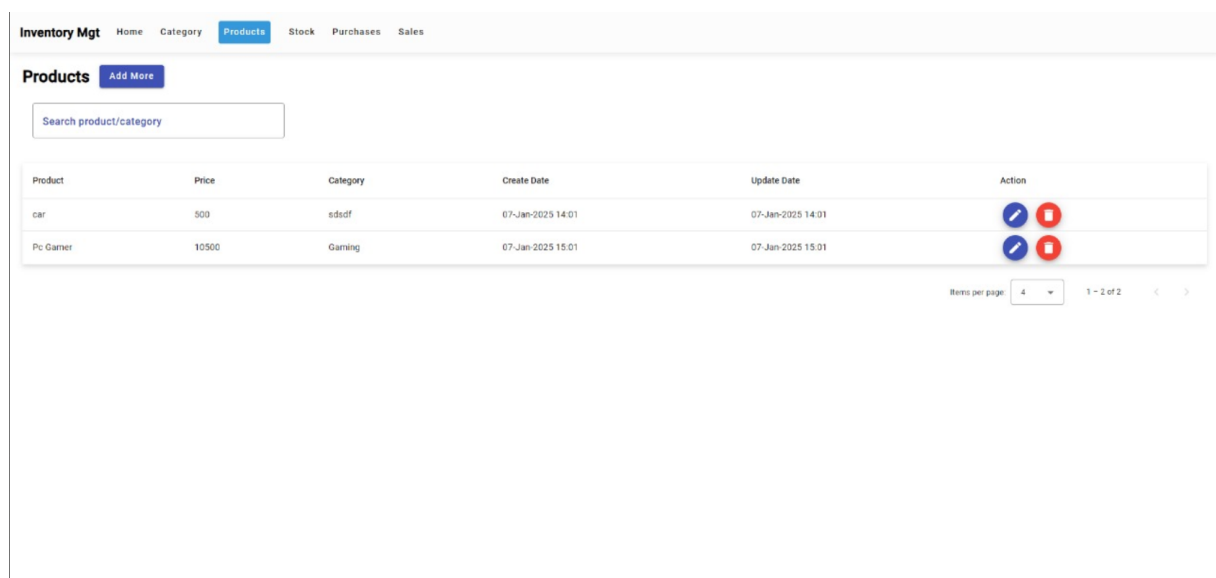


Figure 17: Home Products

Inventory Mgt	Home	Category	Products	Stock	Purchases	Sales
---------------	------	----------	----------	-------	-----------	-------

Stock <input type="text" value="Search prouduct/category"/>		
Product	Category	Quantity
car	sdsdf	10
Pc Gamer	Gaming	4

Items per page:
 1 - 2 of 2

Figure 18: page stock

- Description :** La page de gestion du stock offre une vue d'ensemble des produits disponibles, des quantités en stock et des mouvements récents (ventes et achats). Elle permet également de filtrer et de rechercher des produits spécifiques.
- Rôle dans l'Application :** Elle sert d'interface centrale pour l'administrateur, facilitant la gestion et le suivi du stock en temps réel.







Inventory Mgt Home Category Products Stock Purchases Sales						
Purchases Add More						
<div> <input type="text" value="Product Name"/> <input type="text" value="Start date - End date"/> <input type="button" value="Clear"/> </div>						
Purchase Date(dd-MM-yyyy)	ProductName	Price	Quantity	Description	TotalPrice	Action
07-01-2025 00:01	car	500	10	sdsds	5000	 
08-01-2025 00:01	car	500	10		5000	 
07-01-2025 00:01	Pc Gamer	10500	5	A laptop for game lovers	52500	 
<div> Items per page: <input type="text" value="5"/> 1 - 3 of 3 </div>						

Figure 19: Purchases page

- **Description** : La page de gestion des achats permet à l'administrateur d'enregistrer de nouveaux achats, de suivre les commandes fournisseurs et de mettre à jour les stocks en conséquence.
- **Rôle dans l'Application** : Elle est essentielle pour la gestion des achats, garantissant un suivi précis des commandes et une mise à jour automatique du stock.









Inventory Mgt Home Category Products Stock Purchases Sales						
Sales Add Sale						
<div> <input type="text" value="Product Name"/> <input type="text" value="Start date - End date"/> <input type="button" value="Clear"/> </div>						
Selling Date (dd-MM-yyyy)	Product Name	Price	Quantity	Total Price	Description	Action
14-01-2025 00:01	dsfsdf	500	1	500	dd	 
07-01-2025 00:01	car	500	1	500		 
07-01-2025 00:01	car	500	9	4500		 
07-01-2025 00:01	Pc Gamer	10500	1	10500		 
<div> Items per page: <input type="text" value="5"/> 1 - 4 of 4 </div>						

Figure 20: sales page

- **Description** : La page de gestion des ventes permet à l'administrateur d'enregistrer les transactions de vente, de suivre les produits vendus et de mettre à jour les stocks en temps réel..

Inventory Mgt
Home
Category
Products
Stock
Purchases
Sales

Categories

Parent Category
▼

Save
Reset









Category	Parent Category	Create Date	Update Date	Action
cat2		06-Jan-2025 20:01	07-Jan-2025 14:01	 
sdsdf		06-Jan-2025 20:01	06-Jan-2025 20:01	 
cat3	sdsdf	07-Jan-2025 14:01	07-Jan-2025 14:01	 
Gaming		07-Jan-2025 15:01	07-Jan-2025 15:01	 

Figure 20: Categories page

CONCLUSION

Ce projet a été une aventure enrichissante, intégrant diverses technologies et méthodologies pour développer une plateforme numérique. Depuis les phases initiales de conception et d'analyse jusqu'à la réalisation et le déploiement, chaque étape a été soigneusement orchestrée pour répondre aux besoins définis tout en assurant une cohérence et une efficacité opérationnelle.

1. Réalisation Technique

Le projet a bénéficié d'un environnement de travail bien défini, utilisant des outils et technologies modernes tels que Angular pour le front-end, et .NET avec Microsoft SQL Server pour le back-end. Chaque composant a joué un rôle crucial dans la création de cette application. L'utilisation de Visual Studio pour le développement, et Postman pour les tests d'API ont assuré une qualité élevée et une intégration fluide entre les différentes parties du système.

2. Cycle de Développement

Le cycle de développement a suivi une approche méthodique, incluant l'analyse des besoins, la recherche, la conception en UML, et le développement.

3. Interfaces de l'Application

Les différentes interfaces de l'application, allant de la page d'accueil à la gestion des profils utilisateurs, ont été conçues pour offrir une expérience utilisateur fluide et intuitive.

4. Documentation et Tests

La documentation complète des API et des processus de développement a été un aspect crucial pour assurer la transparence et la facilité d'utilisation pour les développeurs et les utilisateurs finaux.