



Université de la Manouba
École Nationale des Sciences de l'Informatique



RAPPORT DU PROJET DE CONCEPTION ET DE DÉVELOPPEMENT

Sujet : Portail de vente en ligne

Auteurs :

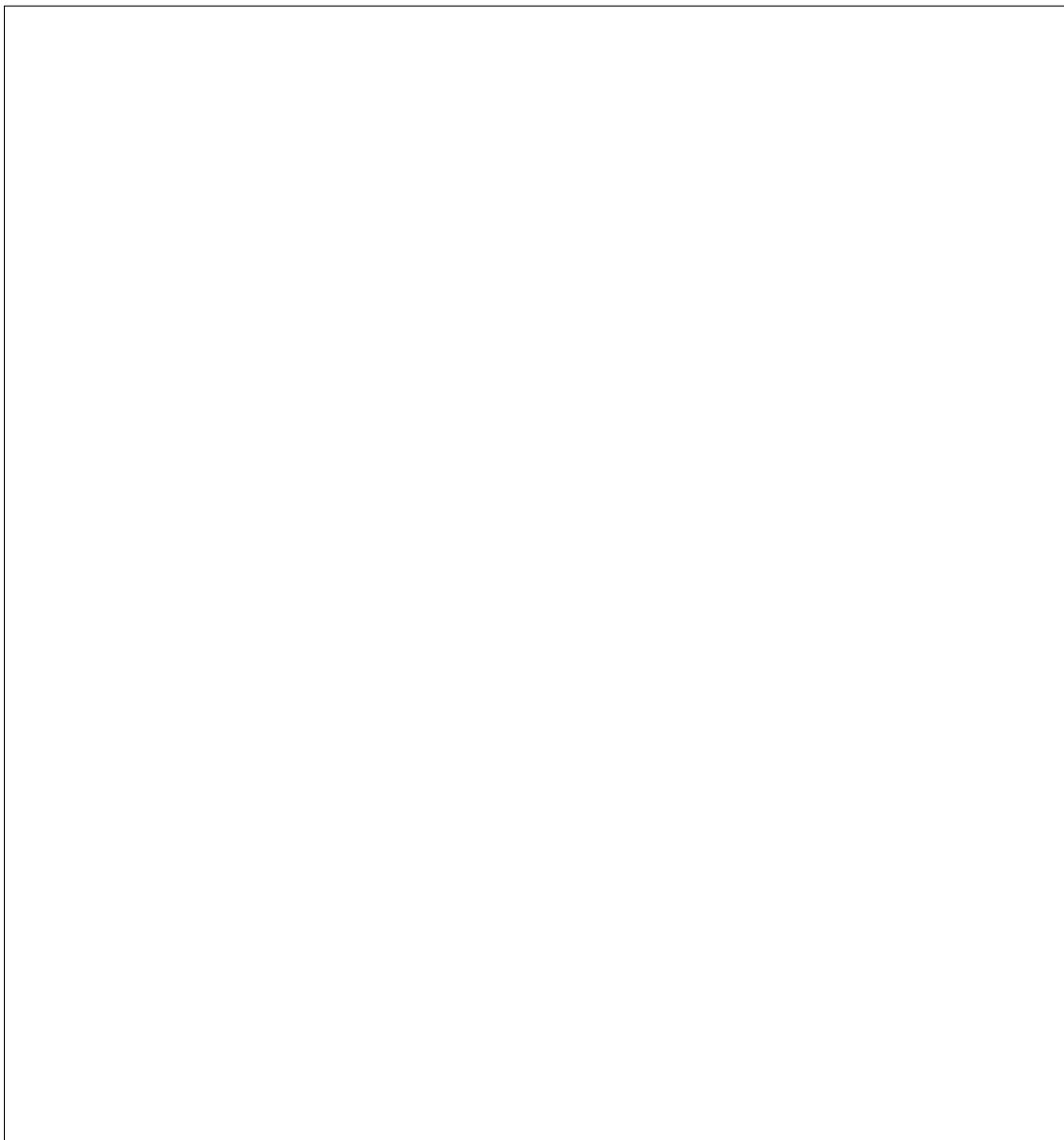
M. Firas JABEUR M^{elle} Chaima MESSADI M. Youssef CHRAIEF

Encadrant :

Dr. Wissem BEN SAÏD

Année Universitaire : 2022 /2023

Appréciations et signature de l'encadrant



Remerciements

Avant de commencer la présentation de notre travail, nous tenons à remercier tous ceux et celles qui nous ont apporté leur aide, leurs encouragements et leur soutien pour nous permettre de mener à bien ce projet, et nos remerciements s'adressent en particulier à Monsieur Wissem Ben Said pour son assistance, sa disponibilité et ses précieux conseils durant toute cette période.

Nous exprimons, également, tout notre respect et notre haute considération aux membres du jury qui ont accepté d'évaluer ce travail en espérant qu'ils trouvent notre rapport clair et exprimant toute la motivation qu'ils attendent.

Finalement, nos remerciements s'adressent à tous les enseignants de l'école nationale des sciences de l'informatique qui nous ont fait bénéficier de leurs connaissances et leur savoir-faire en sciences de l'informatique.

Table des matières

Introduction	1
1 Présentation générale du projet	2
1.1 Problématique	2
1.2 Travail demandé	2
1.3 Méthodologie adoptée	3
1.3.1 Choix de méthodologie	3
2 Etude de préalable	5
2.1 Étude de l'existant	5
2.1.1 Solutions existantes	5
2.1.1.1 Shopify	5
2.1.1.2 WooCommerce	6
2.1.1.3 MyEasyStore	7
2.1.2 Critiques de l'existant	7
3 Analyse et spécification des besoins	9
3.1 Analyse des besoins	9
3.1.1 Identification des acteurs	9
3.1.2 Besoins fonctionnels	9
3.1.3 Besoins non fonctionnels	10
3.1.4 Besoins du domaine	10
3.2 Spécification des besoins	11
3.2.1 Diagramme de cas d'utilisation global	11
3.2.2 Cas d'utilisation détaillés	13
3.2.2.1 Cas d'utilisation « S'inscrire client »	13
3.2.2.2 Cas d'utilisation « S'authentifier client »	13
3.2.2.3 Cas d'utilisation « Gérer son panier »	14
3.2.2.4 Cas d'utilisation « S'inscrire vendeur »	14
3.2.2.5 Cas d'utilisation « Gérer les produits »	15
3.2.3 Descriptions des scénarios	15

3.2.3.1	Interaction Administrateur-Système	15
3.2.3.2	Interaction Vendeur-Système	15
4 conception		19
4.1	Conception globale	19
4.1.1	Architecture physique	19
4.1.2	Architecture logique	20
4.2	Conception détaillés	21
4.2.1	Diagramme de classe	21
4.2.2	Diagrammes de séquence	23
4.2.2.1	Diagramme de séquence « Créer boutique »	23
4.2.2.2	Diagramme de séquence « Ajouter produit »	24
4.2.2.3	Diagramme de séquence « Authentification-Client »	25
4.2.2.4	Diagramme de séquence « Acheter Produit »	26
4.2.3	Diagrammes d'activité	27
4.2.3.1	Diagramme d'activité « Ajouter Produit »	27
4.2.3.2	Diagramme d'activité « Acheter Produit »	27
5 Réalisation et tests		29
5.1	Environnement de travail	29
5.1.1	Environnement matériel	29
5.1.2	Environnement logiciel	30
5.1.2.1	Technologies utilisées	30
5.1.2.2	Environnement de développement intégré	30
5.2	Aperçu des interfaces	31
5.2.1	Interface « Page d'accueil »	31
5.2.2	Interface « Créer compte »	32
5.2.3	Interface « Se connecter »	34
5.2.4	Interface « Choisir votre template »	35
5.2.5	Interface « Boutique »	36
5.2.6	Interface « Espace administrateur vendeur »	40
5.2.7	Interface « Détail du produit »	43
5.2.8	Interface « Comparer les produits »	43
5.2.9	Interface « Panier »	45
5.3	Chronogramme du projet	45
Conclusion et perspectives		47
Bibliographie		49

Table des figures

1.1	Modèle cycle en V.[1]	3
3.1	Diagramme de cas d'utilisation global.	12
3.2	Diagramme de cas d'utilisation « S'inscrire client ».	13
3.3	Diagramme de cas d'utilisation « S'authentifier client ».	13
3.4	Diagramme de cas d'utilisation « Gérer son panier ».	14
3.5	Diagramme de cas d'utilisation « S'inscrire vendeur ».	14
3.6	Diagramme de cas d'utilisation « Gérer les produits ».	15
3.7	Diagramme de séquence système « créer un magasin ».	16
3.8	Diagramme de séquence système « ajout d'un produit ».	16
3.9	Diagramme de séquence système « commander un produit ».	17
3.10	Diagramme de séquence système « comparaison produits ».	18
4.1	Les différentes interactions de l'architecture 3-tiers.	20
4.2	Les différentes interactions du modèle MVC.[2]	21
4.3	Diagramme de classe.	22
4.4	Diagramme de séquence « Créer boutique ».	23
4.5	Diagramme de séquence « Ajouter produit ».	24
4.6	Diagramme de séquence « Authentification-Client ».	25
4.7	Diagramme de séquence « Acheter Produit ».	26
4.8	Diagramme d'activité « Ajouter Produit ».	27
4.9	Diagramme d'activité « Acheter Produit ».	28
5.1	Page d'accueil.	32
5.2	Inscription client.	33
5.3	Inscription Vendeur.	34
5.4	S'authentifier.	35
5.5	Choisir un template.	35
5.6	Page d'accueil d'une boutique.	36
5.7	Exemple d'un template.	37
5.8	Ajout d'un produit.	38
5.9	Ajout d'une description.	38

5.10	Modifier un produit.	39
5.11	Une carte du produit.	39
5.12	Ajout d'une catégorie.	40
5.13	Profile d'un vendeur.	41
5.14	Dashboard.	42
5.15	Liste des commandes.	43
5.16	Détail d'un produit.	43
5.17	Comparer les produits.	44
5.18	Tableau de comparaison.	44
5.19	Panier d'un client.	45

Liste des tableaux

2.1 Comparaison des fonctionnalités de Shopify, WooCommerce, MyEasyStore et notre site	8
4.1 Description des différentes classes de l'application	21
5.1 Les caractéristiques des PCs utilisés	29
5.2 Chronogramme du projet	46

Introduction

De nos jours, la vente en ligne est devenue un canal de vente incontournable pour de nombreux commerçants. Les consommateurs sont de plus en plus nombreux à acheter des produits en ligne, que ce soit pour des raisons de commodité, de disponibilité ou de prix. Les avantages d'une plateforme de vente en ligne sont nombreux et peuvent permettre aux commerçants d'élargir leurs portés et de toucher un public plus large.

Dans ce contexte, ce présent rapport porte sur un projet de développement d'un portail de vente en ligne où les vendeurs peuvent créer leur propre espace de vente et gérer leurs produits, clients et commandes. Le système donne aux vendeurs la possibilité de personnaliser leur page de vente et effectue une recherche sémantique sur le web pour les produits similaires, ce qui permet au vendeur de connaître les prix des autres concurrents lors de l'ajout des produits. D'autre part, les clients peuvent parcourir les différentes boutiques du portail afin de découvrir les produits exposés et effectuer des achats.

Dans la suite, nous détaillerons toutes les phases de développement de cette plateforme, il s'étale sur cinq chapitres organisés comme suit :

- Le premier chapitre est réservé à la présentation générale du cadre du projet et une étude des méthodologies de travail et des langages de modélisation.
- Le deuxième chapitre présente une étude des plateformes Web existantes permettant de créer un espace de vente, afin de dégager les services offerts et de critiquer leurs points faibles.
- Quant au troisième chapitre, il est dédié à l'analyse et la spécification des besoins.
- Le quatrième chapitre est consacré à l'architecture physique et logique adoptées. Ensuite nous allons effectuer une étude statique et dynamique du système en schématisant les diagrammes de classes, de séquence et d'activité.
- Le dernier chapitre contient les choix techniques utilisés ainsi que des interfaces de l'application qui décrivent des scénarios utilisateur.
- À la fin, ce rapport sera clôturé par une conclusion générale qui résume le travail effectué en présentant quelques perspectives futures visant à améliorer notre application dans son état actuel.

Chapitre 1

Présentation générale du projet

Introduction

Ce chapitre sera réservé à la présentation du cadre de notre projet. Dans un premier temps, nous présenterons la problématique à résoudre suivie d'une description globale. Dans un deuxième temps, nous allons introduire la méthodologie de travail adoptée et le langage de modélisation.

1.1 Problématique

Le commerce en ligne est un secteur en constante évolution et la création d'un portail de vente en ligne est devenue une tendance importante dans le monde des affaires pour faciliter la commercialisation des produits. Les vendeurs peuvent créer leur propre espace de vente en ligne, ce qui présente de nombreux avantages pour eux. Pour les clients, cette approche offre un large choix de produits provenant de différents vendeurs, ce qui peut conduire à une expérience d'achat plus pratique et plus personnalisée.

Notre projet s'inscrit dans cette optique et consiste à concevoir et à développer un portail de vente en ligne performant permettant aux vendeurs de présenter leurs produits en ligne de manière flexible et efficace, tout en offrant une expérience d'achat en ligne pratique et fiable pour les clients.

1.2 Travail demandé

Le projet consiste à concevoir et à développer un portail de vente en ligne permettant aux vendeurs de créer leur propre espace de vente. Ces dernières peuvent gérer leurs produits, leurs clients et leurs commandes à travers leur espace d'administration. Le portail propose une fonctionnalité de recherche sémantique pour aider les vendeurs à aligner leurs prix avec ceux de leurs concurrents et à personnaliser leurs pages de produits afin de les

présenter de manière professionnelle. Les clients peuvent parcourir les pages de produits de chaque vendeur et effectuer des achats.

1.3 Méthodologie adoptée

Pour pouvoir entamer le développement du projet, il est primordial d'adopter une méthodologie de développement. Le choix entre une méthode et une autre dépend de la nature du projet et de sa taille. En effet, il s'agit plutôt de sélectionner la méthode la plus adaptée à notre projet.

1.3.1 Choix de méthodologie

Le modèle en V est un modèle de développement de logiciels qui découpe le processus de développement en deux parties principales, la partie gauche est une phase descendante concerne les spécifications et la conception et la partie droite est une phase ascendante travaille sur la mise en œuvre, les tests et l'installation.

La figure 1.1 permet d'exprimer toutes les étapes nécessaires pour la réalisation d'un projet. Le cycle en V est subdivisé en 9 étapes :

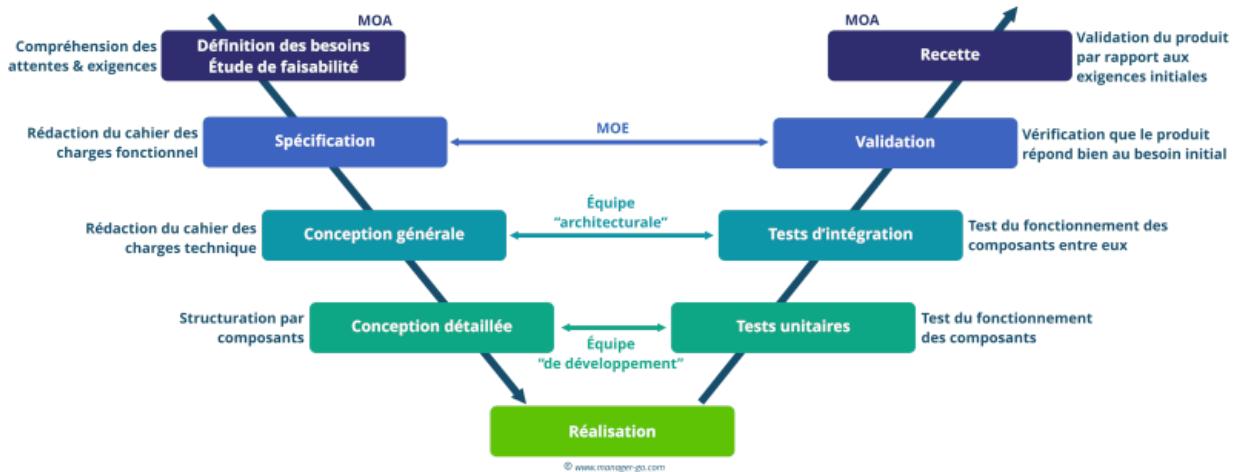


FIGURE 1.1 – Modèle cycle en V.[1]

- **Analyse des besoins** : l'expression des besoins du client, en décrivant les usages correspondant au produit fini.
- **Spécifications** : la définition du cahier de charge du produit final qui décrit les exigences sous forme de spécifications détaillées.

- **Conception générale** : la définition de l'architecture globale du système, l'identification des différents modules logiciels pour la réalisation des fonctionnalités du système, la détermination de la structure de base de données et l'organisation de code.
- **Conception détaillée** : la réalisation des modèles de données et les différents diagrammes en se basant sur des outils de modélisation UML.
- **Mise en œuvre** : la génération d'un code fonctionnel qui satisfait la conception et les spécifications déjà créées.
- **Tests unitaires** : tester chaque composant de code individuellement des autres.
- **Test d'intégration** : tester les différentes parties du système ensemble pour vérifier le bon fonctionnement.
- **Tests système** : ce sont des tests qui assurent que le système répond bien aux exigences spécifiées.
- **Test d'acceptation** : permet de vérifier la conformité du système aux normes et à la règlementation du client.

Au cours de ce projet, nous avons choisi le cycle en V comme méthodologie, car il permet de garantir une structuration claire et assure une gestion rigoureuse des tests. Aussi, ce modèle permet d'améliorer la qualité et la fiabilité de notre application.

Conclusion

Ce chapitre a servi à présenter le contexte et le cadre du projet et à expliquer la méthodologie du travail adoptée.

Chapitre 2

Etude de préalable

Introduction

Ce chapitre est dédié à la présentation de l'étude préalable de notre projet, qui consiste à examiner différentes plateformes qui permettent à des vendeurs de créer leur propre espace de vente en ligne. Cette étude préliminaire permettra d'analyser, d'évaluer et de critiquer le fonctionnement habituel de ces plateformes, en détaillant leurs avantages et leurs inconvénients. En effectuant une analyse des solutions existantes, nous pourrons cerner les objectifs à atteindre et développer un système de qualité pour notre propre plateforme.

2.1 Étude de l'existant

L'étape d'étude de l'existant est essentielle pour tout projet, car elle permet de détecter les insuffisances et les faiblesses des solutions actuelles, afin de les prendre en compte lors de la conception et de la réalisation d'un futur système. Dans cette section, nous présenterons une analyse des différentes plateformes existantes qui permettent à des vendeurs de créer leur propre espace de vente en ligne. Nous procéderons à une étude et une critique approfondies de ces plateformes, afin d'identifier leurs avantages et leurs inconvénients.

2.1.1 Solutions existantes

Dans cette section, nous allons présenter trois solutions existantes qui permettent aux vendeurs de créer leur propre espace de vente en ligne : Shopify, WooCommerce et MyEasyStore.

2.1.1.1 Shopify

Shopify est une plateforme de commerce électronique qui permet aux vendeurs de créer leur propre boutique en ligne. Elle propose de nombreuses fonctionnalités, notamment la

possibilité de personnaliser le design de la boutique, de gérer les commandes, les paiements et les livraisons, ainsi que de suivre les statistiques de vente en temps réel. Elle offre également une grande variété d'applications et de thèmes pour personnaliser davantage la boutique. Cependant, Shopify est une solution payante, avec des tarifs qui dépendent du plan choisi. [3]

- **Avantages :**

- Interface utilisateur intuitive et facile à utiliser
- Possibilité de personnaliser les designs de manière simple et rapide
- Large choix d'applications et de fonctionnalités supplémentaires disponibles
- Disponible en plusieurs langues

- **Inconvénients :**

- Les coûts peuvent être élevés pour les fonctionnalités avancées
- Limitations dans la personnalisation de certaines fonctionnalités
- La configuration de certains éléments peut nécessiter des compétences techniques avancées
- Les frais de transaction peuvent être élevés si vous n'utilisez pas la solution de paiement interne de Shopify

2.1.1.2 WooCommerce

WooCommerce est un plugin open source pour WordPress, qui permet de transformer un site WordPress en une boutique en ligne complète. Il offre une grande flexibilité en termes de personnalisation, avec de nombreux thèmes et extensions disponibles. WooCommerce est également gratuit, mais il nécessite un certain niveau de compétence technique pour l'installation et la configuration. [4]

- **Avantages :**

- Solution open source, gratuite et facile à installer
- Grande flexibilité et personnalisation possible grâce à un large choix de plugins
- Grande communauté d'utilisateurs, ce qui permet d'avoir accès à un grand nombre de tutoriels et de guides en ligne

- **Inconvénients :**

- Les coûts peuvent augmenter rapidement avec l'utilisation de plugins payants
- L'installation et la configuration peuvent être plus difficiles pour les utilisateurs non techniques
- L'assistance technique n'est pas incluse avec la solution
- Les mises à jour régulières peuvent causer des problèmes de compatibilité avec les plugins installés

2.1.1.3 MyEasyStore

MyEasyStore est une plateforme de commerce électronique qui permet de créer facilement une boutique en ligne. Elle offre des fonctionnalités de base gratuites et permet une personnalisation simple des designs et des fonctionnalités. Elle est disponible en plusieurs langues et peut être synchronisée avec d'autres plateformes de vente en ligne. Cependant, certaines fonctionnalités avancées ne sont disponibles qu'à travers des forfaits payants et la personnalisation avancée nécessite des compétences techniques. De plus, les coûts peuvent être élevés pour les fonctionnalités avancées et la disponibilité de l'assistance technique est limitée pour les utilisateurs gratuits. [5]

- **Avantages :**

- Interface utilisateur simple et facile à utiliser
- Des fonctionnalités de base sont disponibles gratuitement
- Possibilité de personnalisation des designs et des fonctionnalités
- Disponible en plusieurs langues

- **Inconvénients :**

- Certaines fonctionnalités ne sont disponibles qu'à travers des forfaits payants
- La personnalisation avancée nécessite des compétences techniques
- Les coûts peuvent être élevés pour les fonctionnalités avancées
- La disponibilité de l'assistance technique est limitée pour les utilisateurs gratuits

2.1.2 Critiques de l'existant

Pour faciliter la compréhension des différentes solutions présentées précédemment, nous avons dressé un tableau comparatif 2.1 mettant en évidence les différences au niveau des fonctionnalités offertes et des coûts associés. Bien que toutes ces solutions répondent à la problématique de la création d'une boutique en ligne, notre site se démarque en offrant la possibilité de recherche sémantique pour les vendeurs afin d'aligner les prix de leurs produits par rapport à ceux de la concurrence. De plus, notre site offre la possibilité aux visiteurs de comparer des produits provenant de tous les sites créés dans notre portail, ce qui constitue un avantage significatif pour l'expérience d'achat des clients potentiels.

TABLE 2.1 – Comparaison des fonctionnalités de Shopify, WooCommerce, MyEasyStore et notre site

Fonctionnalités	Shopify	WooCommerce	MyEasyStore	Notre site
Création de boutique en ligne	Oui	Oui	Oui	Oui
Interface utilisateur intuitive	Oui	Oui	Oui	Oui
Personnalisation des designs	Oui	Oui	Oui	Oui
Gestion des commandes	Oui	Oui	Oui	Oui
Gestion des paiements	Oui	Oui	Oui	Oui
Statistiques de vente	Oui	Oui	Oui	Oui
Recherche de produit sur le portail	Non	Non	Non	Oui
Comparaison des produits	Non	Non	Non	Oui
Recherche sémantique pour aligner les prix	Non	Non	Non	Oui

Conclusion

En conclusion de cette étude préalable, nous avons examiné les différentes solutions existantes pour la création de boutiques en ligne. Nous avons constaté que ces solutions partagent un certain nombre de fonctionnalités clés. Cependant, notre site offre également des fonctionnalités uniques. Nous sommes convaincus que ces fonctionnalités innovantes apporteront une valeur ajoutée significative à nos utilisateurs et les aideront à atteindre leurs objectifs de vente en ligne.

Chapitre 3

Analyse et spécification des besoins

Introduction

La phase d'analyse et spécification des besoins est une étape primordiale dans le cycle de développement d'un projet. En effet, elle permet de dégager les acteurs et de traduire les besoins de l'utilisateur. Pour ce faire, nous allons commencer par définir les acteurs ainsi que leurs rôles. Ensuite, nous détaillerons les besoins fonctionnels et non fonctionnels que le système devra assurer et enfin nous exposerons les différents cas d'utilisation et quelques scénarios et le pilotage du projet.

3.1 Analyse des besoins

3.1.1 Identification des acteurs

Administrateur : c'est la personne ou l'équipe chargée de la gestion du portail de vente qui peut gérer les comptes des vendeurs et les problèmes techniques.

Vendeur : ce sont les personnes qui s'inscrivent sur le portail de vente pour créer leur propre espace de vente et proposer leurs produits aux clients.

Client : cet acteur crée un compte sur notre site et il peut suivre le processus d'achat des produits.

3.1.2 Besoins fonctionnels

- Pour l'administrateur :

- Modifier et supprimer les comptes des vendeurs existants.
- Vérifier et valider les informations des vendeurs avant leur mise en ligne.

- Pour le vendeur :

- Créer un compte sur le portail.
- Ajouter, modifier et supprimer des produits dans son espace de vente.
- Voir les prix des produits similaires pour aligner ses prix.
- Présenter les produits sur une page propre à sa marque ou entreprise.
- Visualiser les clients et les commandes dans son espace d'administration.
- Accéder aux statistiques de vente pour son propre compte.

- Pour le client :

- Créer un compte sur le portail pour pouvoir passer des commandes.
- Rechercher des produits et des boutiques sur le portail.
- Ajouter des produits à son panier.
- Payer en ligne et recevoir une confirmation.
- Consulter les caractéristiques de chaque produit.
- Sélectionner deux produits et les afficher côté à côté pour une comparaison visuelle.

3.1.3 Besoins non fonctionnels

- **Sécurité** : Le système doit être sécurisé pour protéger les informations des clients, des vendeurs et des administrateurs, ainsi que les transactions financières. Les données sensibles doivent être cryptées et stockées de manière sécurisée.
- **Fiabilité** : Le système doit être fiable et disponible en tout temps. Les interruptions de service doivent être minimisées pour éviter de perdre des ventes ou des clients.
- **Performances** : Le système doit être performant et capable de gérer un grand nombre de transactions simultanément. Les temps de réponse doivent être rapides pour éviter que les clients ne s'impatientent et quittent le site.
- **Facilité d'utilisation** : Le système doit être facile à utiliser pour tous les acteurs du système, y compris les vendeurs, les clients et les administrateurs. Les interfaces utilisateur doivent être intuitives, conviviales et ergonomiques pour faciliter la navigation et les interactions avec le système.
- **Maintenance** : Le système doit être facile à maintenir pour permettre la correction rapide de bogues, la mise à jour des logiciels et la résolution de problèmes de performance.

3.1.4 Besoins du domaine

- **Suivi des stocks** : Le système doit être capable de suivre en temps réel les niveaux de stock des produits pour éviter les erreurs de commande.

- **Prise en charge des normes et des protocoles de sécurité :** Le système doit prendre en charge les normes et les protocoles de sécurité, tels que le SSL et le HTTPS, pour garantir la sécurité des données des utilisateurs.

3.2 Spécification des besoins

Après avoir précisé les acteurs du système et ses besoins, il est primordial de donner une vision globale sur le comportement fonctionnel de notre application. Nous procérons dans cette section à modéliser les fonctions et modules que doit fournir le système à l'aide du diagramme de cas d'utilisation.

3.2.1 Diagramme de cas d'utilisation global

Le diagramme de cas d'utilisation global du candidat et de l'administrateur est illustré dans La figure 3.1. Ce diagramme sera détaillé dans la section suivante.

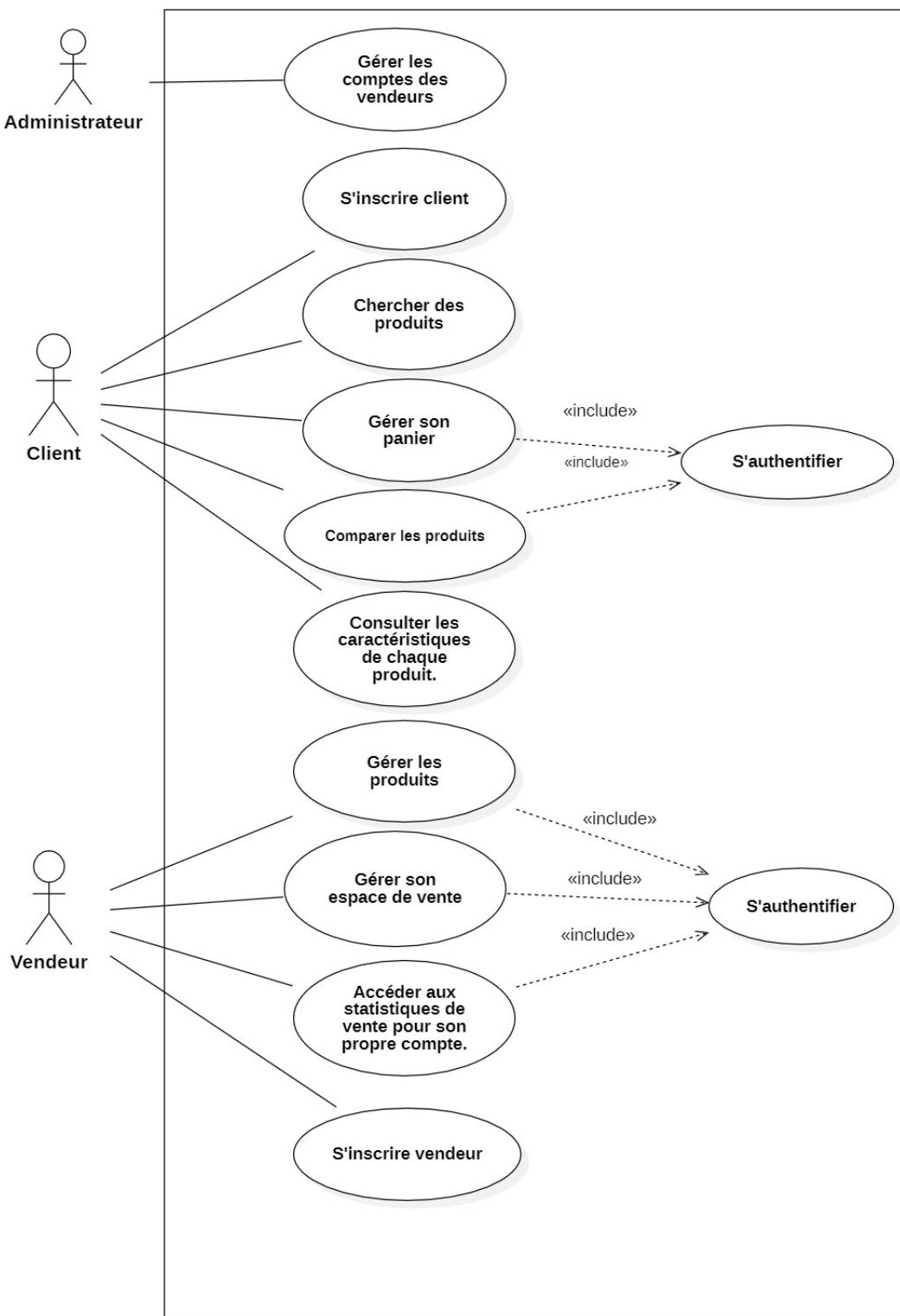


FIGURE 3.1 – Diagramme de cas d'utilisation global.

3.2.2 Cas d'utilisation détaillés

3.2.2.1 Cas d'utilisation « S'inscrire client »

La figure 3.2 ci-dessous permet à un utilisateur de créer un compte client sur la plateforme.

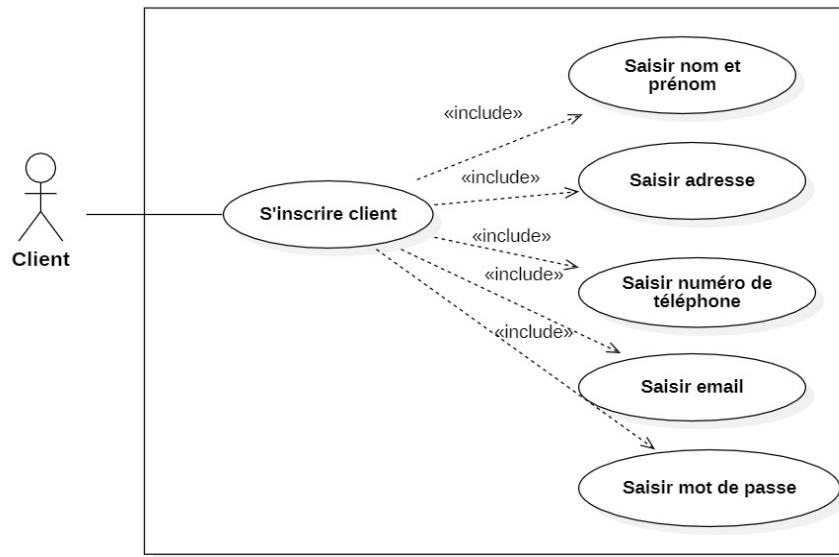


FIGURE 3.2 – Diagramme de cas d'utilisation « S'inscrire client ».

3.2.2.2 Cas d'utilisation « S'authentifier client »

La figure 3.3 décrit le processus d'authentification du client.

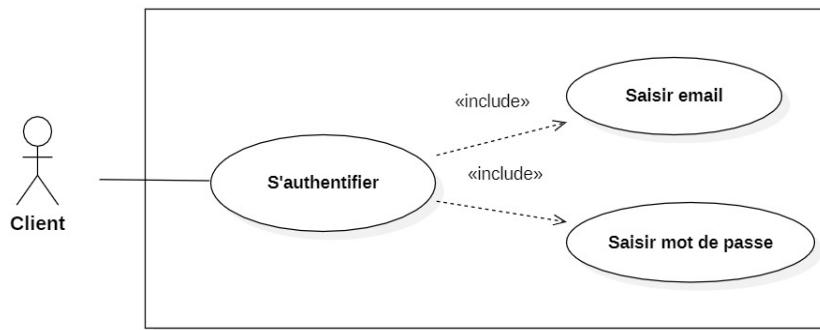


FIGURE 3.3 – Diagramme de cas d'utilisation « S'authentifier client ».

3.2.2.3 Cas d'utilisation « Gérer son panier »

La figure 3.4 représente la gestion du contenu du panier par l'utilisateur, avec la possibilité d'ajouter ou de supprimer des produits.

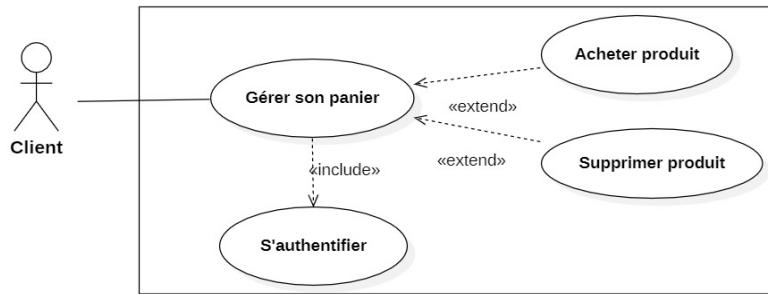


FIGURE 3.4 – Diagramme de cas d'utilisation « Gérer son panier ».

3.2.2.4 Cas d'utilisation « S'inscrire vendeur »

La figure 3.5 offre la possibilité à un utilisateur de mettre en place sa propre boutique sur la plateforme.

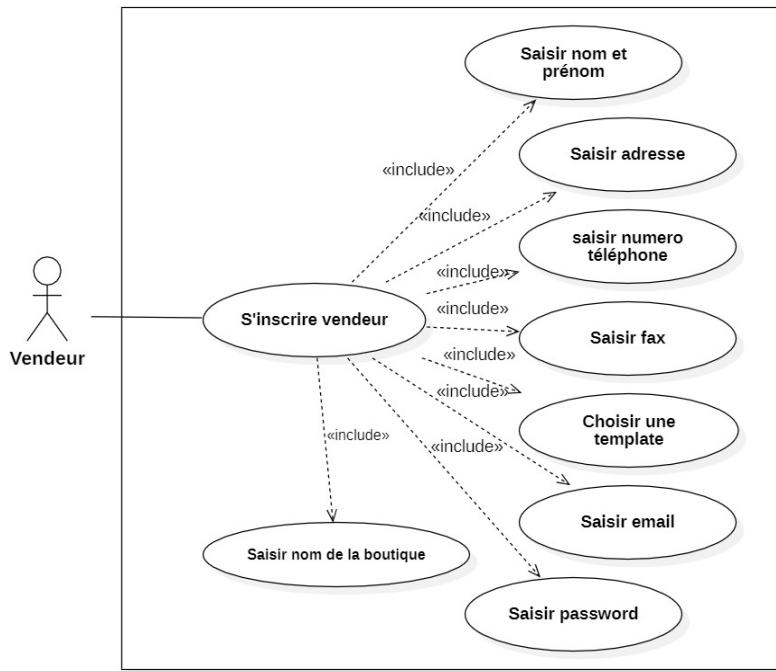


FIGURE 3.5 – Diagramme de cas d'utilisation « S'inscrire vendeur ».

3.2.2.5 Cas d'utilisation « Gérer les produits »

La figure 3.6 permet à un vendeur de gérer ses produits, en ajoutant, modifiant ou supprimant des produits de sa boutique.

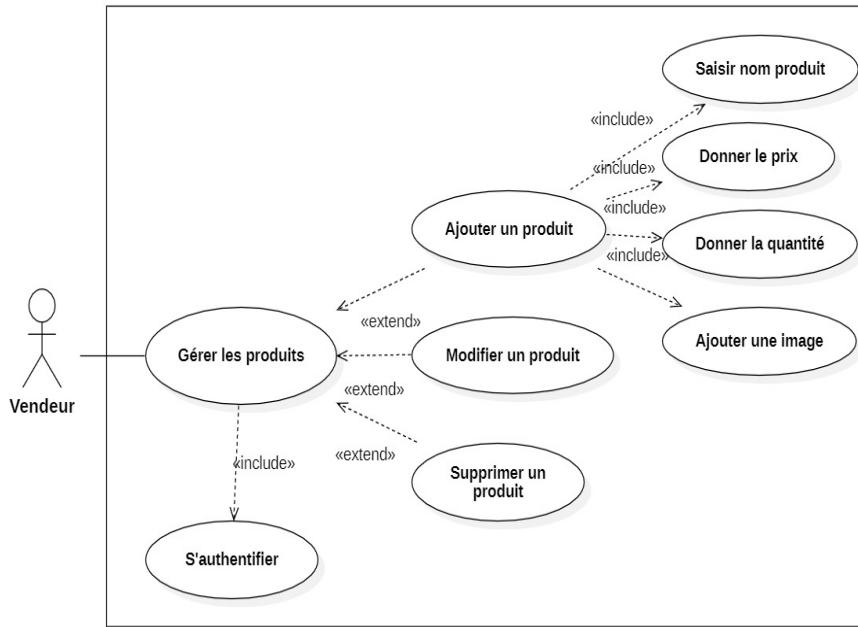


FIGURE 3.6 – Diagramme de cas d'utilisation « Gérer les produits ».

3.2.3 Descriptions des scénarios

Ci-dessous sont présentés des scénarios d’interaction entre les utilisateurs et le système.

3.2.3.1 Interaction Administrateur-Système

L’administrateur est responsable à la gestion globale de la plateforme. Il peut gérer les comptes utilisateurs.

3.2.3.2 Interaction Vendeur-Système

Le diagramme de séquence système « créer un magasin » permet à un vendeur de créer sa boutique sur la plateforme.

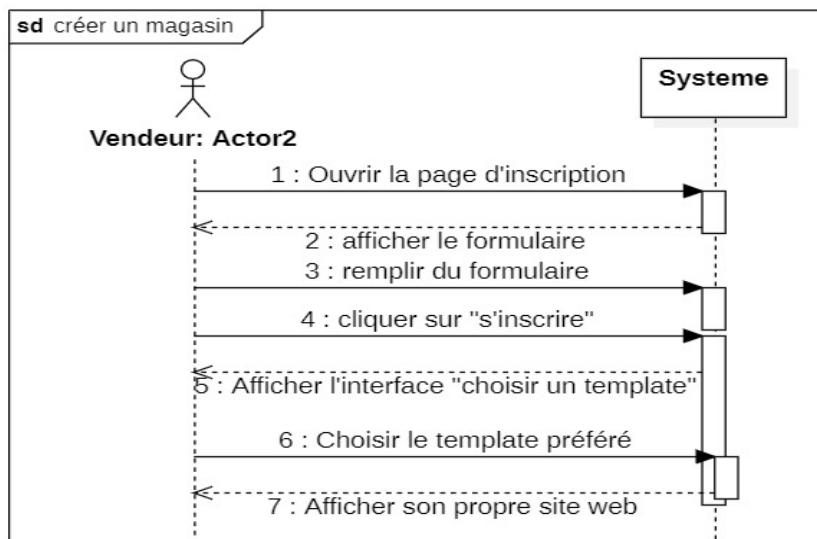


FIGURE 3.7 – Diagramme de séquence système « créer un magasin ».

La figure 3.8 permet à un vendeur d'ajouter un produit à sa boutique.

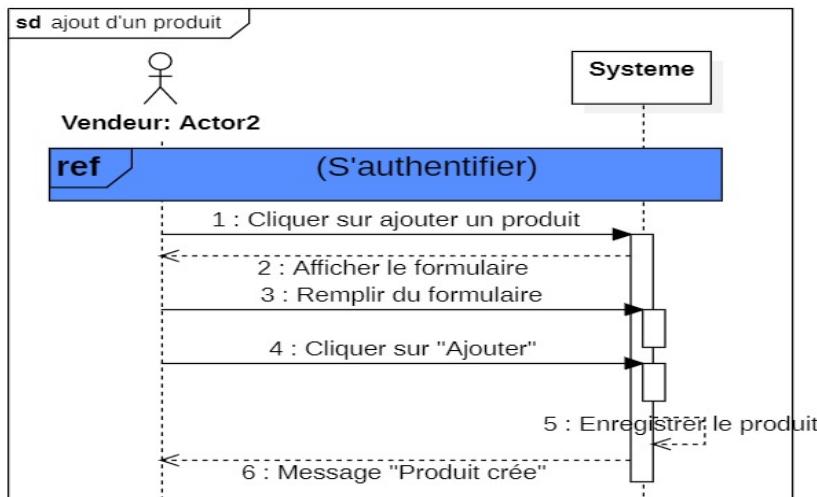


FIGURE 3.8 – Diagramme de séquence système « ajout d'un produit ».

La figure 3.9 détaille les étapes à suivre pour qu'un client puisse passer une commande.

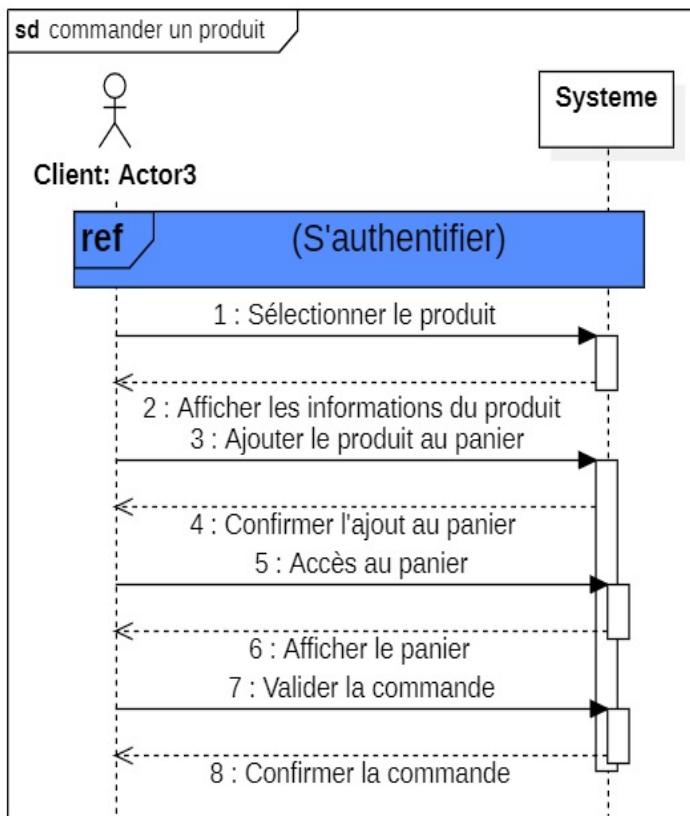


FIGURE 3.9 – Diagramme de séquence système « commander un produit ».

La figure 3.10 montre comment un client peut comparer plusieurs produits pour faciliter sa décision d'achat.

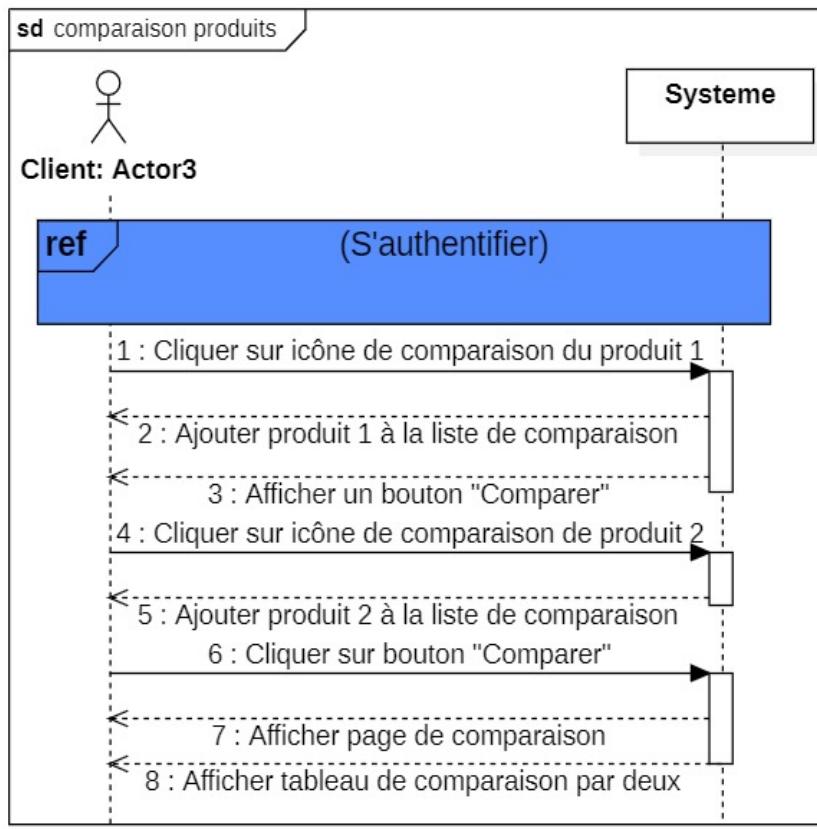


FIGURE 3.10 – Diagramme de séquence système « comparaison produits ».

Conclusion

Dans ce chapitre, nous avons identifié les acteurs de notre système, détaillé les besoins fonctionnels et les besoins non fonctionnels. Aussi, nous avons présenté les différentes interactions entre les acteurs et le système.

Le chapitre suivant sera consacré à la phase de conception de l'application

Chapitre 4

conception

Introduction

Durant ce chapitre, nous allons présenter la conception de notre application. Pour ce faire nous commençons par le choix de notre architecture physique et logique. Puis nous détaillerons le diagramme de classe ainsi que les diagrammes de séquence et d'activité.

4.1 Conception globale

Dans cette partie, il est important de mentionner le choix de notre architecture physique et logique.

4.1.1 Architecture physique

Dans cette section, nous allons aborder l'architecture 3-tiers qui sépare les différentes couches de l'application en trois niveaux distincts (couche de présentation, couche d'application, couche de données).

- La couche de présentation est responsable de l'interface utilisateur. Elle permet d'envoyer le contenu de l'application sous forme de HTML, CSS, JavaScript. Elle s'appuie sur des frameworks comme React et Angular.
- La couche d'application permet de traiter la logique métier et d'effectuer toutes les requêtes des utilisateurs.
- La couche de données présente un système de gestion de base de données tel que MySQL et MongoDB qui assure l'accès à la base de données.

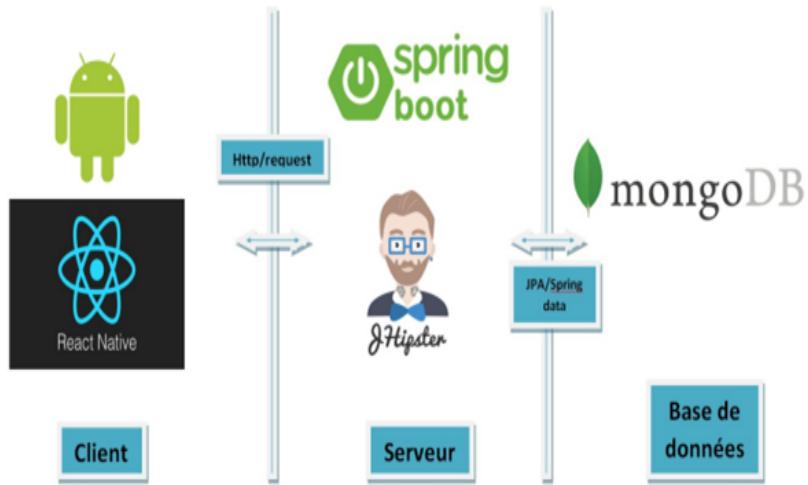


FIGURE 4.1 – Les différentes interactions de l'architecture 3-tiers.

4.1.2 Architecture logique

Nous avons choisi le modèle MVC (Modèle, Vue, Contrôleur) pour notre application. Ce modèle se compose de trois parties suivantes :

- **Le Modèle :** représente la couche de données de l'application. Il contient des objets qui stockent les informations à afficher ou à modifier.
- **La Vue :** c'est l'interface de l'application que l'utilisateur voit et interagit avec. Cette interface permet d'afficher les données du modèle et permet à l'utilisateur de les modifier.
- **Le Contrôleur :** est responsable de la gestion des actions de l'utilisateur et assure la coordination entre le Modèle et la Vue. Il reçoit les requêtes de l'utilisateur et met à jour le Modèle en conséquence.

La figure 4.2 permet d'illustrer clairement le modèle MVC.

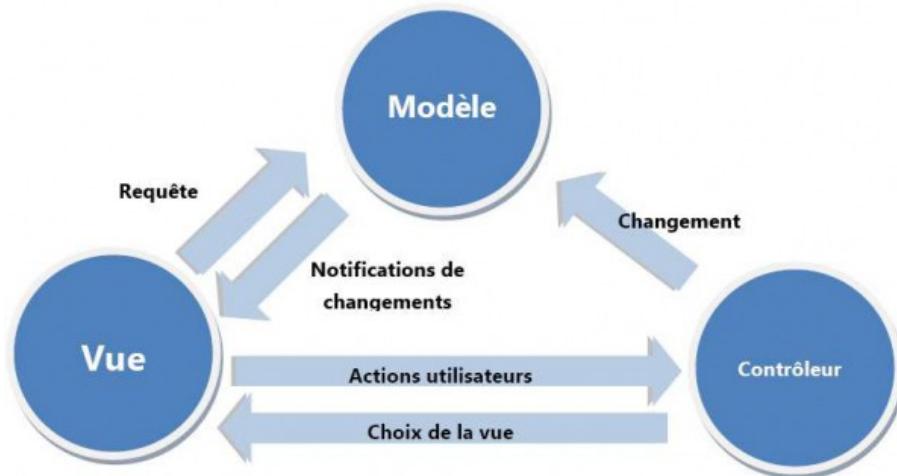


FIGURE 4.2 – Les différentes interactions du modèle MVC.[2]

4.2 Conception détaillés

4.2.1 Diagramme de classe

Le diagramme de classe est une étape cruciale dans le processus de conception. Il fournit une vue statique et abstraite des objets et contient les différentes classes avec leurs attributs, leurs méthodes ainsi que les relations qui les relient.

TABLE 4.1 – Description des différentes classes de l'application

Classe	Description
Utilisateur	C'est une classe mère pour les deux classes Vendeur et Client
Vendeur	C'est une classe fille de la classe Utilisateur. Elle contient les informations des vendeurs
Client	C'est une classe fille de la classe Utilisateur. Elle contient les informations des clients
Produit	C'est une classe qui définit l'ensemble des produits
Catégorie	C'est une classe qui définit l'ensemble des catégories
Commande	C'est une classe qui définit l'ensemble des commandes
Achats	C'est une classe qui définit l'ensemble des produits vendus
Description	C'est une classe qui définit la description pour chaque produit
Template content	C'est une classe qui est responsable à la modification du contenu de la page d'accueil de la boutique, incluant les textes et images.

La figure 4.3 représente le diagramme de classe de notre application.

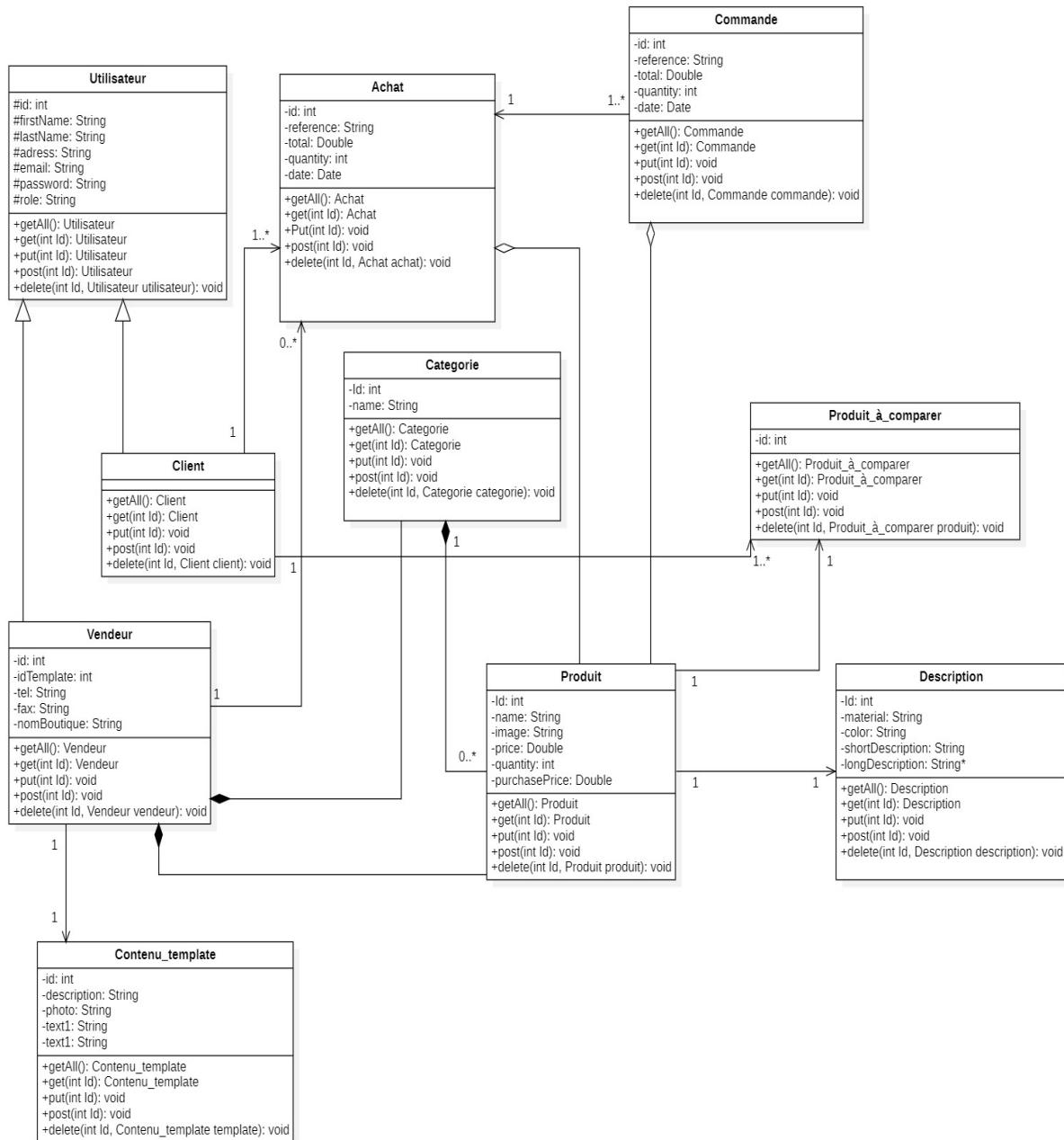


FIGURE 4.3 – Diagramme de classe.

4.2.2 Diagrammes de séquence

Les diagrammes de séquence font partie des diagrammes dynamiques. Ils présentent les interactions entre les différents objets et permettent d'illustrer un scénario d'utilisation du système.

4.2.2.1 Diagramme de séquence « Crée boutique »

La figure 4.4 présente l'ensemble des étapes pour la création d'une nouvelle boutique. Tout d'abord, le vendeur doit remplir un formulaire d'inscription. Si tous les champs sont valides, il y aura enregistrement de ses informations puis le vendeur effectue le choix de son template et il y aura enregistrement de cette information dans la base.

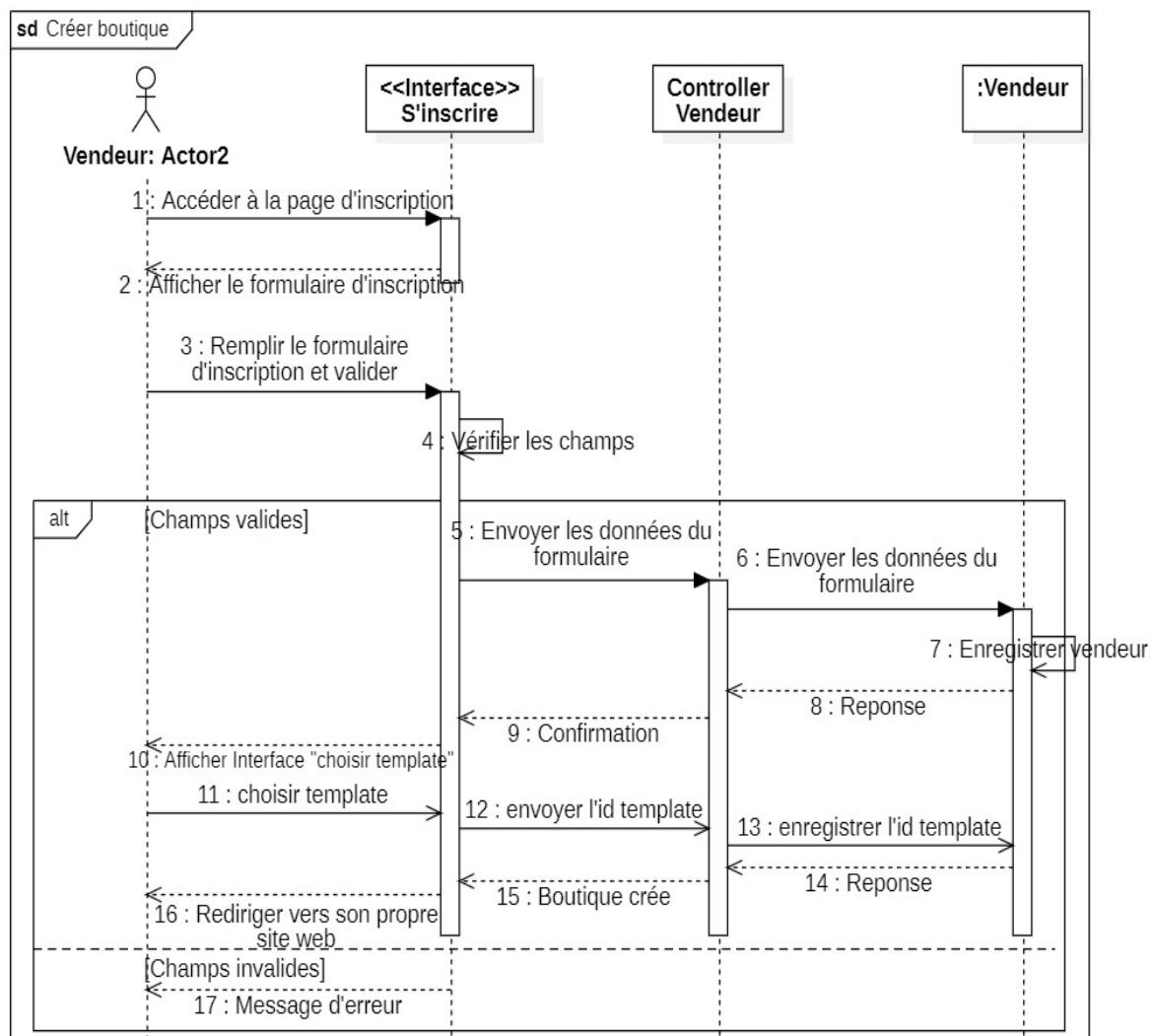


FIGURE 4.4 – Diagramme de séquence « Crée boutique ».

4.2.2.2 Diagramme de séquence « Ajouter produit »

La figure 4.5 illustre le processus de l'ajout d'un nouveau produit. Le vendeur doit remplir les champs nécessaires pour effectuer l'ajout.

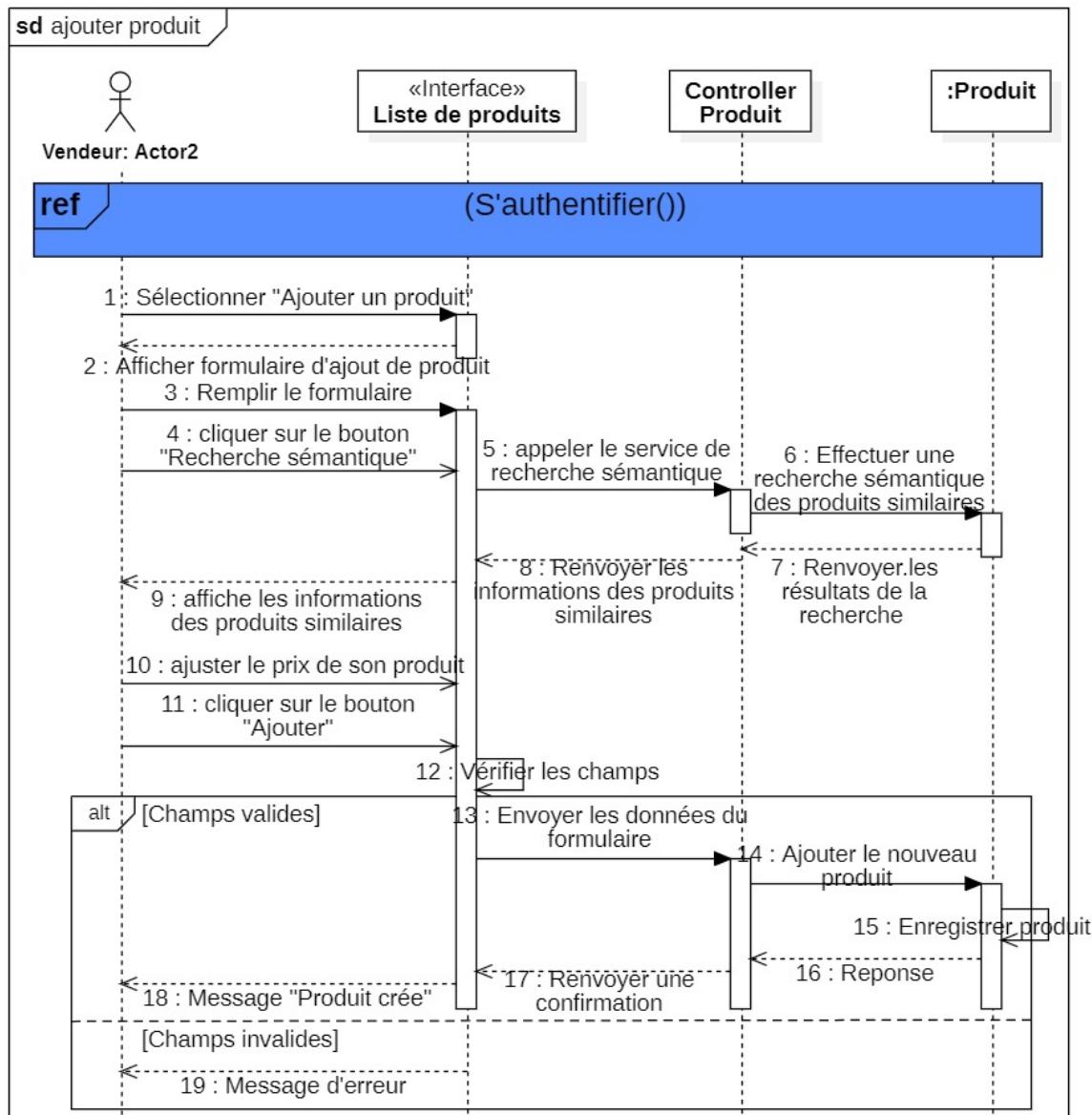


FIGURE 4.5 – Diagramme de séquence « Ajouter produit ».

4.2.2.3 Diagramme de séquence « Authentification-Client »

La figure 4.6 présente le scénario d'authentification du client. Le client saisit son email et son mot de passe, le « Controller » vérifie l'existence de cet utilisateur et en cas de validité le client sera redirigé vers la liste de produit.

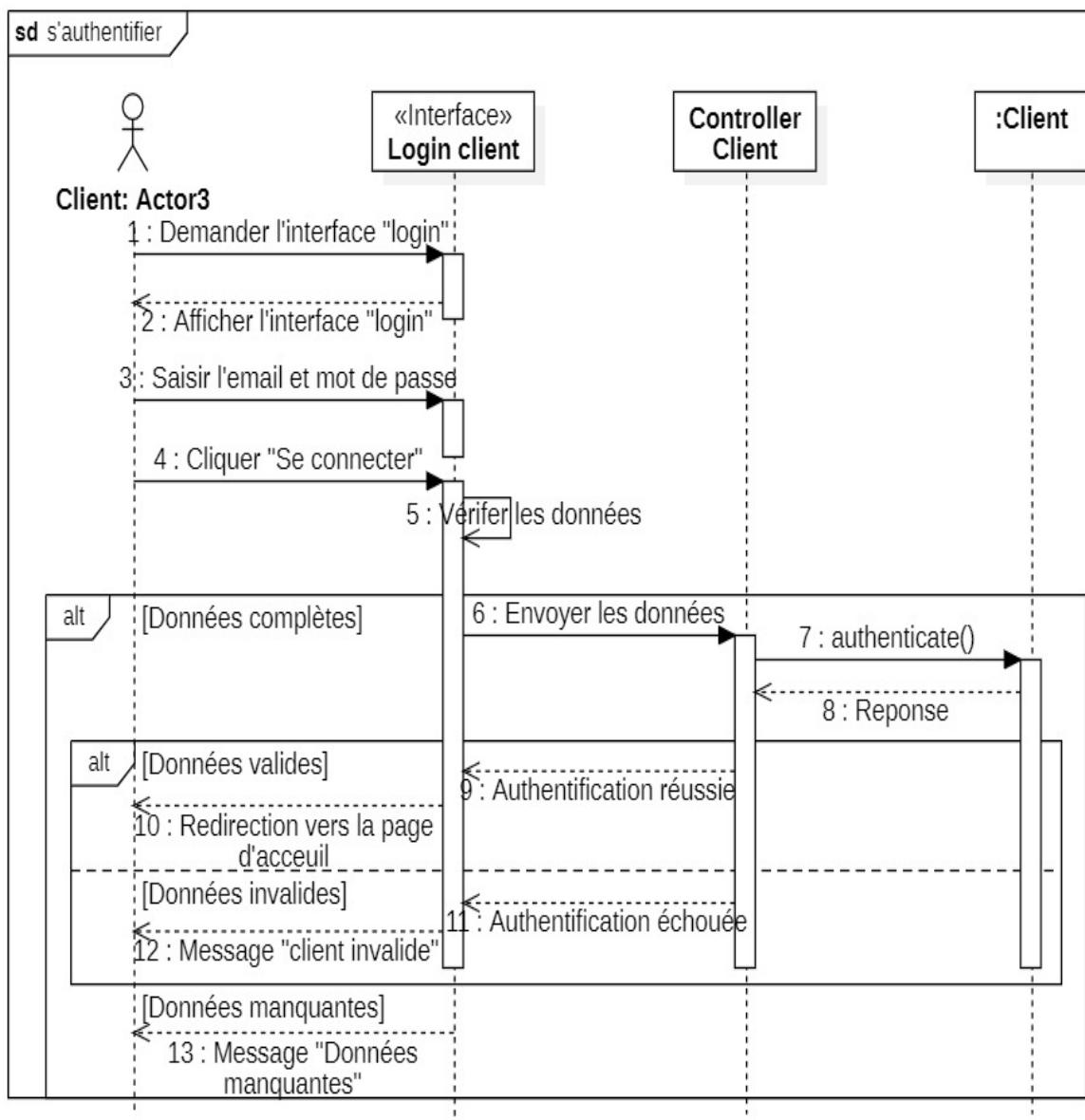


FIGURE 4.6 – Diagramme de séquence « Authentification-Client ».

4.2.2.4 Diagramme de séquence « Acheter Produit »

La figure 4.7 illustre le scénario de l'achat d'un produit. Tout d'abord, le client sélectionne un produit pour l'ajouter à son panier puis il doit saisir ses informations bancaires pour effectuer l'achat.

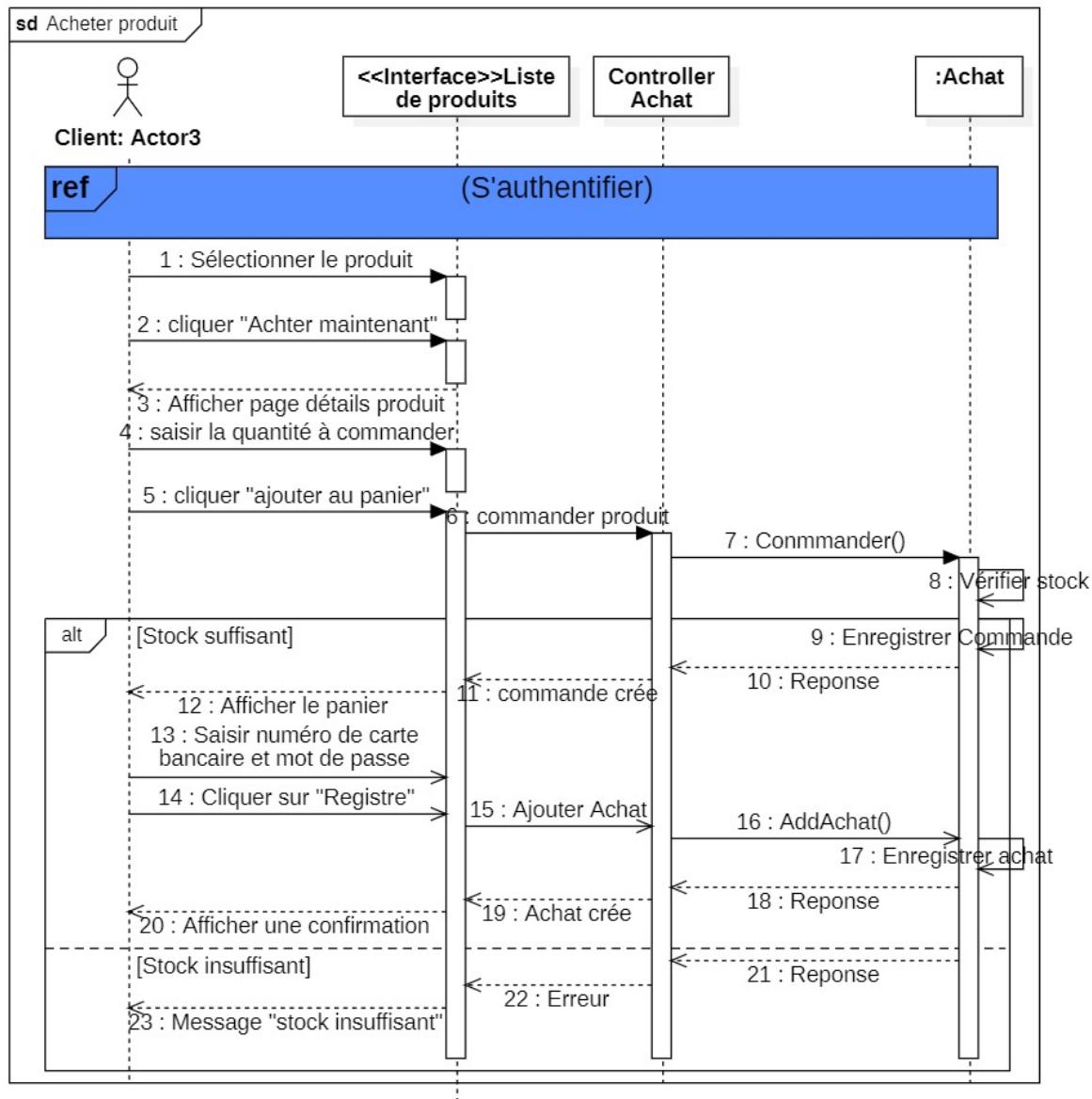


FIGURE 4.7 – Diagramme de séquence « Acheter Produit ».

4.2.3 Diagrammes d'activité

4.2.3.1 Diagramme d'activité « Ajouter Produit »

La figure 4.8 montre l'ensemble d'activité à réaliser lors de l'ajout d'un produit. Après l'authentification correcte, le vendeur doit saisir les données du produit. Puis validé l'ajout du produit.

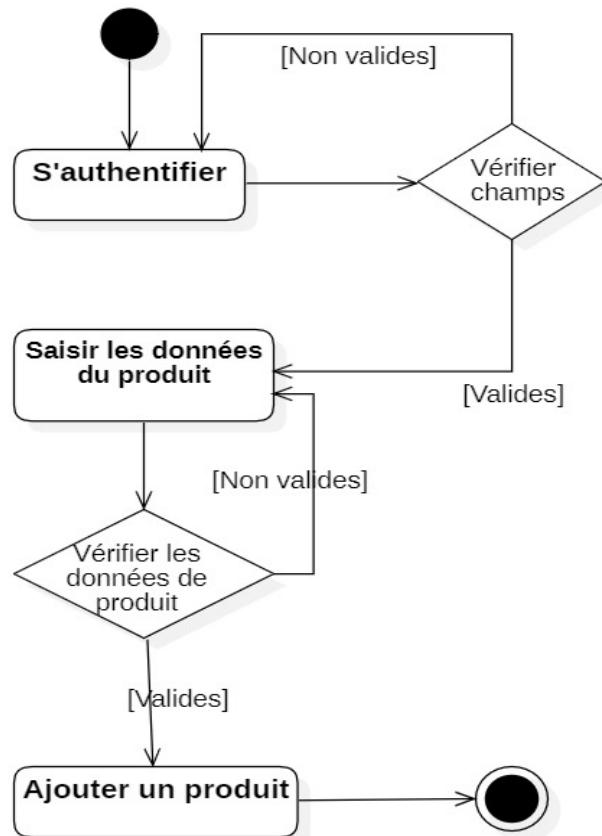


FIGURE 4.8 – Diagramme d'activité « Ajouter Produit ».

4.2.3.2 Diagramme d'activité « Acheter Produit »

La figure 4.9 présente la séquence des activités enchainées lors de l'achat d'un produit. Tout d'abord, le client doit s'authentifier puis il cherche le produit désiré et l'ajoute au panier. En fin il effectue son achat.

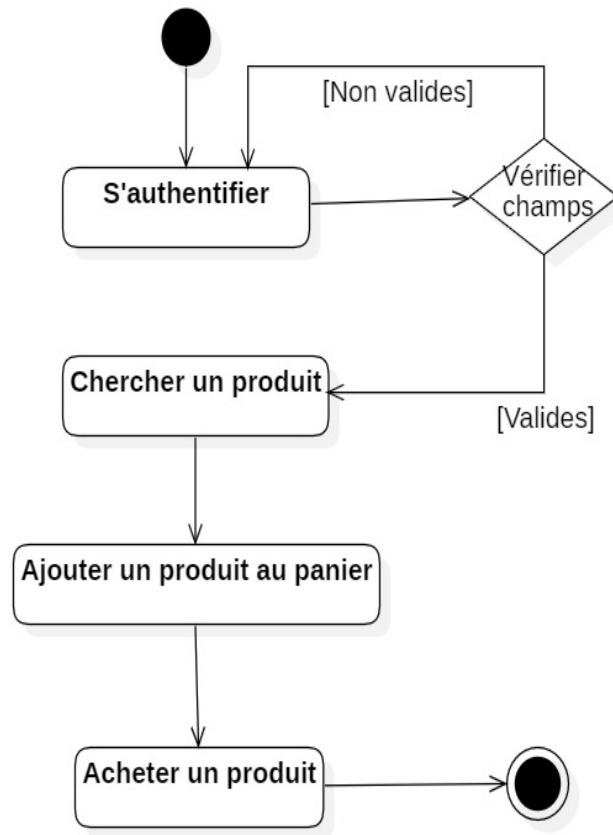


FIGURE 4.9 – Diagramme d'activité « Acheter Produit ».

Conclusion

Au cours de ce chapitre, nous avons présenté l'architecture de notre projet et les aspects statiques et dynamiques grâce aux diagrammes de classe, séquence et activités. Le chapitre suivant sera consacré à l'étape de réalisation et tests.

Chapitre 5

Réalisation et tests

Introduction

Ce chapitre est une partie cruciale dans le développement de notre projet pour cela il est important de noter en premier lieu les différentes technologies et les outils que nous avons utilisés. Ensuite, la deuxième partie sera consacrée à la mise en œuvre de l'application via les interfaces.

5.1 Environnement de travail

Cette partie est dédiée à la présentation des environnements matériels et logiciels qui nous permet de réaliser notre projet.

5.1.1 Environnement matériel

Le tableau 5.1 montre l'environnement matériel utilisé par chacun des étudiants.

Réf PC	Processeur	Disque Dur	Système d'exploitation
Hp X541U	Intel Core i7-7500U	HDD 500 SSD	Microsoft Windows 10 - 64 bits
Msi X556U	Intel Core i7-6500U	HDD : 1 To	Microsoft Windows 10 - 64 bits
Lenovo Ideapad	Rizen 7-3700	HDD : 1 To + 128 SDD	Microsoft Windows 10 - 64 bits

TABLE 5.1 – Les caractéristiques des PCs utilisés

5.1.2 Environnement logiciel

5.1.2.1 Technologies utilisées

Afin d'atteindre les finalités de notre projet, nous avons choisi de travailler avec les outils suivants :

Angular

Angular 15 est un Framework frontend qui permet de créer des pages web en utilisant HTML, SCSS et Type Script. Il implémente les fonctionnalités de base et facultatives sous la forme d'un ensemble de bibliothèques Type Script que vous importez dans vos applications [6]

Bootstrap 5

Bootstrap 5 est un Framework utilisé pour le design de la partie frontale d'un site web qui permet de faciliter le développement. Il possède des modèles de conception HTML et CSS pour les formulaires, boutons, tableaux ... [7]

Python

Python est un langage de programmation open source et interprété. Ce langage est utile surtout dans l'analyse de données, la gestion d'infrastructure et le développement de logiciels. Le développement avec Python est reconnu surtout par sa rapidité et facilité. [8]

Spring boot

Spring boot est un Framework backend open-source pour le développement d'applications basé sur le Framework Spring. Il permet de simplifier la création rapide des applications Spring avec peu de configuration en utilisant des annotations. Aussi, il offre également plusieurs dépendances pour une fonctionnalité courante telle que Spring Security, Spring Web ... [9]

MySQL

MySQL est un système de gestion de bases de données relationnelles. Il est distribué sous une double licence GPL et propriétaire. Il fait partie des logiciels de gestion de base de données les plus utilisés au monde, autant par le grand public (applications web principalement) que par des professionnels, en concurrence avec Oracle, Informix et Microsoft SQL Server. [10]

5.1.2.2 Environnement de développement intégré

Visual Studio Code

Visual Studio Code est un éditeur de code extensible développé par Microsoft. Il intègre des extensions pour plusieurs langages de programmation, notamment Java, JavaScript et C++ et des environnements d'exécution tels que .Net et Unity. [11]

Postman

Postman est une plateforme collaborative de développement des API. Il permet d'envoyer des requêtes REST, SOAP pour tester vos API et surveiller leurs performances. [12]

IntelliJ Idea

IntelliJ Idea est un environnement de développement intégré (IDE) pour le développement de logiciels en Java, Kotlin surtout. Il offre l'intégration avec des outils de construction tels que Maven et Gradle. Aussi, il prend en charge l'intégration des frameworks de développement tels que Spring, Hibernate et Struts. [13]

Star UML

Star UML est un logiciel qui permet de dessiner des diagrammes ou des organigrammes. Il vous propose de concevoir toutes sortes de diagrammes, de dessins vectoriels, de les enregistrer sur plusieurs formats XML, PNG, JPG puis de les exporter. [14]

PyCharm

PyCharm est un environnement de développement intégré (IDE) pour le développement de logiciels en Python. Il offre l'intégration avec des outils de construction tels que pip et setuptools. Aussi, il prend en charge l'intégration des frameworks de développement tels que Django et Flask. [15]

5.2 Aperçu des interfaces

Dans cette partie, nous allons dévoiler les interfaces de l'application.

5.2.1 Interface « Page d'accueil »

L'interface présentée dans la figure 5.1 est la première page affichée lors du démarrage de l'application par le vendeur ou le client.



FIGURE 5.1 – Page d'accueil.

5.2.2 Interface « Créeer compte »

La figure 5.2 présente l'interface d'inscription pour un client, ce dernier doit remplir les champs spécifiques : (Nom, Prénom, Adresse, Numéro De Téléphone, Email, Mot De Passe). Une fois inscrit, le client se dirige vers la page d'authentification.

The form is titled "S'inscrire" at the top center. It consists of six input fields arranged vertically, each with a placeholder text: "Nom", "Prenom", "Adresse", "Numero Telephone", "E-mail", and "Mot de passe". Below these fields is a large blue rectangular button with the text "S'inscrire" in white.

FIGURE 5.2 – Inscription client.

La figure 5.3 est l’interface d’inscription pour un vendeur, ce dernier doit remplir les champs spécifiques : (Nom De Boutique, Prénom, nom, Adresse, Fax, Numéro De Téléphone, Email, Mot De Passe). Une fois inscrit, le vendeur se dirige vers la page « choisir votre template » pour qu'il puisse choisir le template voulu pour sa boutique.

The image shows a registration form titled "S'inscrire". It consists of eight input fields arranged vertically, each with a placeholder text: "Nom", "Prenom", "Nomboutique", "Adresse", "Numero Telephone", "E-mail", "Mot de passe", and "Fax". Below these fields is a large blue rectangular button with the text "S'inscrire" in white.

FIGURE 5.3 – Inscription Vendeur.

5.2.3 Interface « Se connecter »

La figure 5.4 présente l'interface de connexion. Cette interface est utilisée par le client pour accéder à son compte ou par le vendeur pour accéder à son propre site de commerce.

Les utilisateurs doivent saisir correctement leurs adresses email et leurs mots de passe puis cliquer sur le bouton se connecter. Si un alerte « vous n'avez pas de compte veuillez s'inscrire » l'utilisateur doit cliquer sur le bouton s'inscrire pour qu'il soit amener à la page d'inscription pour créer un nouveau compte.

The image shows a user interface for logging in. At the top center is the title "Se connecter". Below it are two input fields: the first for "E-mail" and the second for "Mot de passe". To the left of the password field is a blue link labeled "S'inscrire". To the right of the password field is a blue button with a white checkmark and the text "Se connecter".

FIGURE 5.4 – S'authentifier.

5.2.4 Interface « Choisir votre template »

Comme vous pouvez remarquer dans la figure 5.5, Chaque template présenté offre un design spécifique avec des structures différentes, mais propose les mêmes fonctionnalités de base. Ainsi, les vendeurs peuvent choisir le modèle qui correspond le mieux à leurs produits. En explorant les différents templates disponibles.

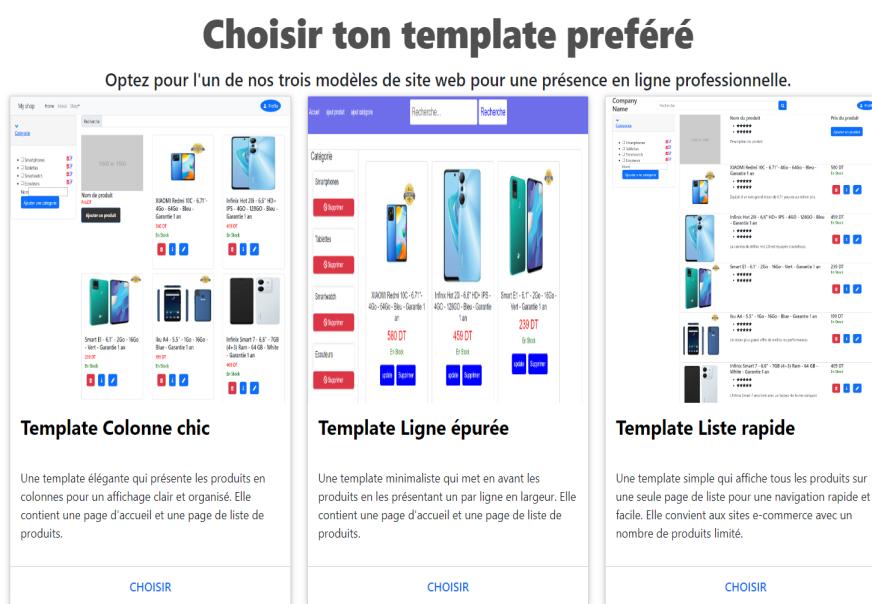


FIGURE 5.5 – Choisir un template.

5.2.5 Interface « Boutique »

Cette interface est composée de deux pages : une page d'accueil et une page de liste de produits. La page d'accueil (figure 5.6) permet aux vendeurs de modifier le contenu de leur boutique et de changer la photo de couverture en cliquant sur l'icône "modifier".



FIGURE 5.6 – Page d'accueil d'une boutique.

Dans la page de liste de produits, le vendeur peut gérer les fonctionnalités principales de gestion des catégories et des produits à savoir l'ajout, la modification et la suppression. Dans la figure 5.7 pour ajouter un nouveau produit, il suffit de cliquer sur le bouton "Ajouter un produit".

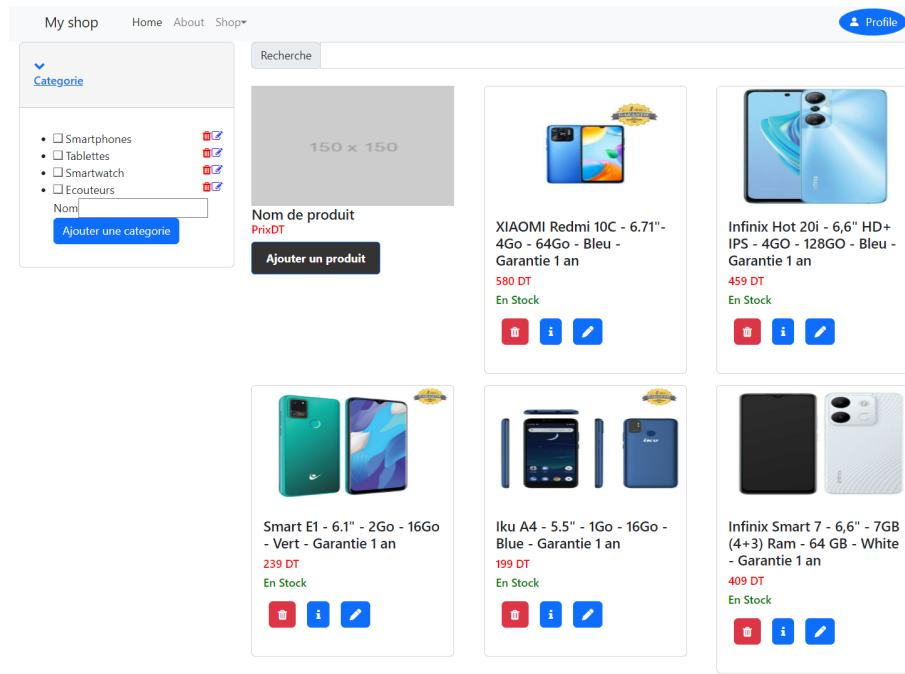


FIGURE 5.7 – Exemple d'un template.

Le vendeur doit donner les informations nécessaires pour un produit (nom, prix, quantité, la catégorie à laquelle appartient, la couleur, la matière et la description du produit). Après la saisie du nom de produit, il peut cliquer sur le bouton "Recherche sémantique" pour afficher les prix des produits similaires, les figures suivantes 5.8 et 5.9 permettent d'illustrer ceci

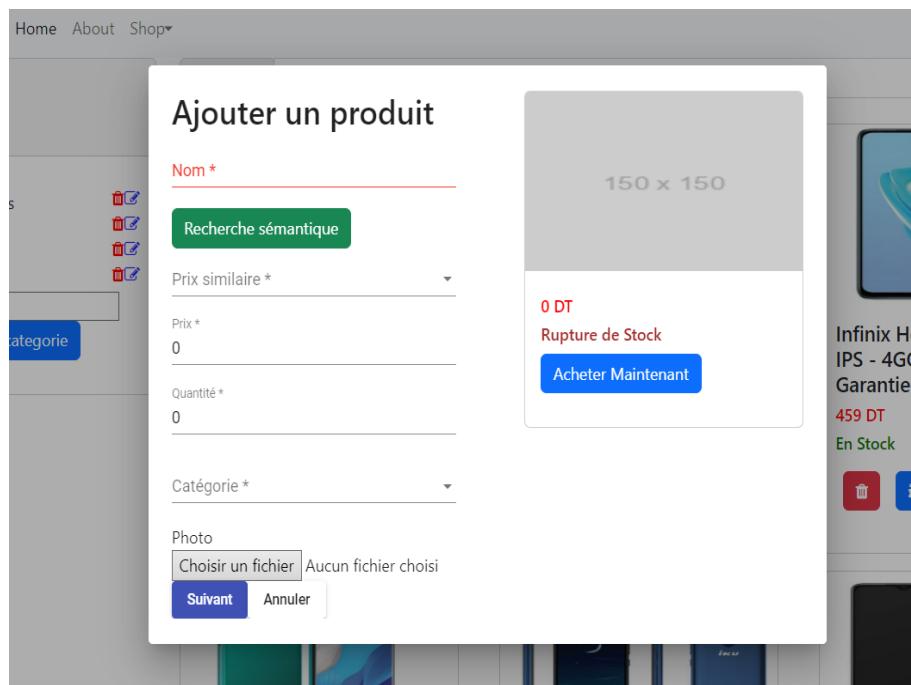


FIGURE 5.8 – Ajout d'un produit.

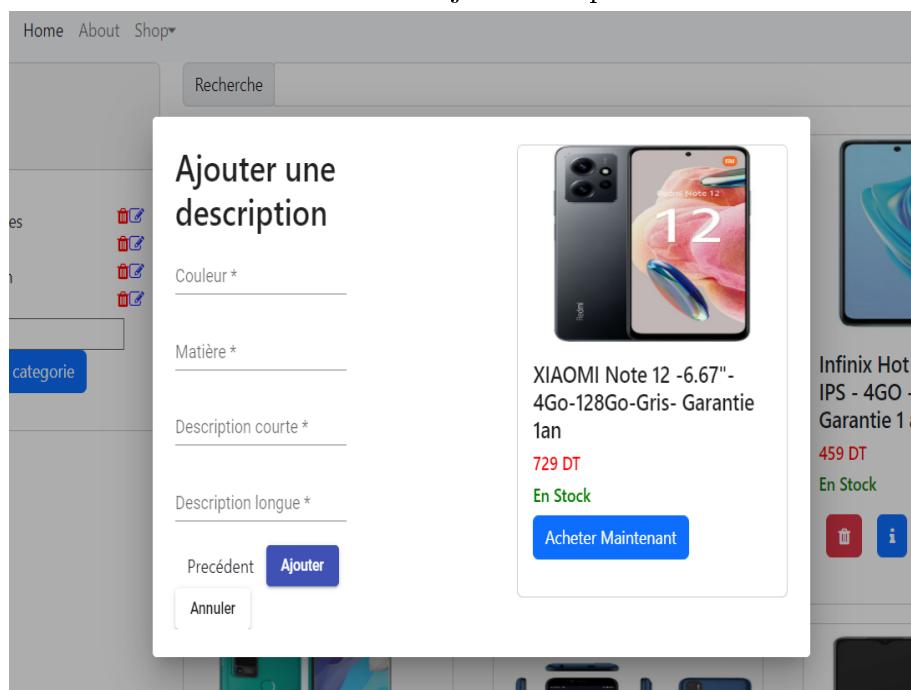


FIGURE 5.9 – Ajout d'une description.

Dans la figure 5.10, lorsque le vendeur clique sur l'icône de modification, un formulaire prérempli avec les informations du produit s'affiche, lui permettant de les modifier. Une fois les modifications effectuées, le vendeur peut cliquer sur le bouton "Enregistrer".

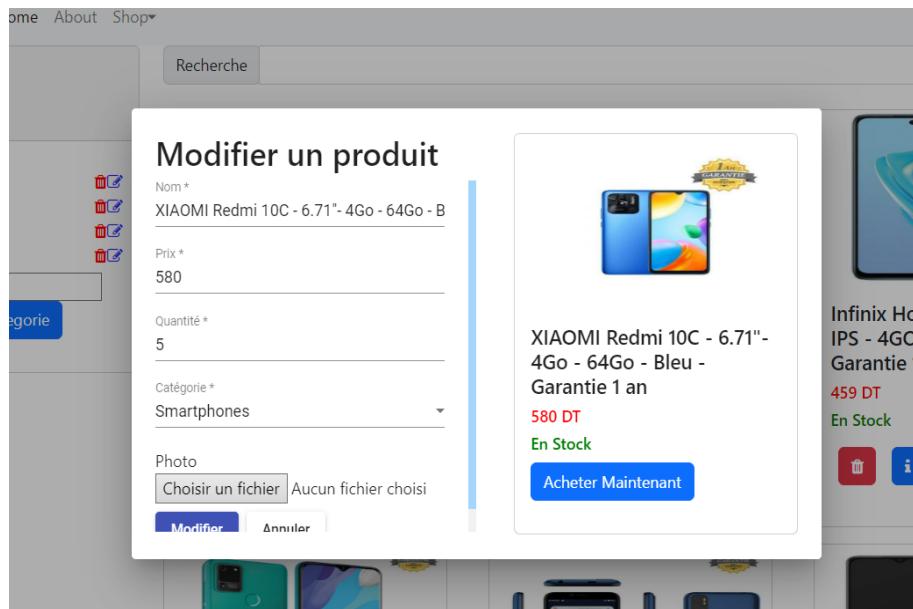


FIGURE 5.10 – Modifier un produit.

En cliquant, sur le bouton supprimer (icône corbeille) le produit sera supprimé comme vous pouvez le remarquer dans la figure 5.11.



FIGURE 5.11 – Une carte du produit.

La figure 5.12 présente la modification, la suppression et l'ajout d'une catégorie.

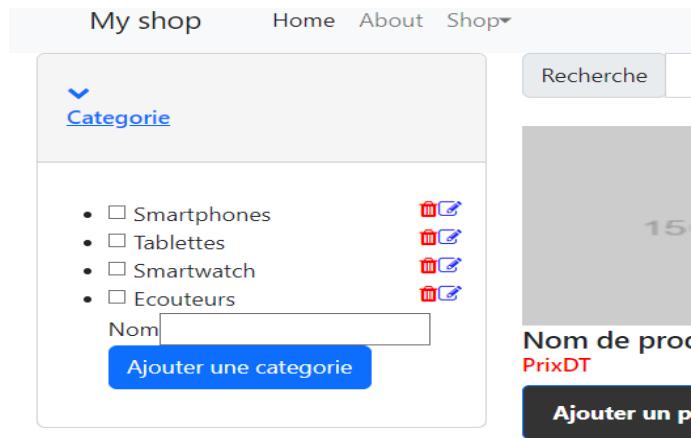


FIGURE 5.12 – Ajout d'une catégorie.

5.2.6 Interface « Espace administrateur vendeur »

L'interface Espace administrateur vendeur est conçue pour permettre aux vendeurs de gérer leur boutique en ligne. Elle se compose de trois pages principales : la page "Profile", la page "Dashboard" et la page "Commandes".

La page "Profile" (figure 5.13) permet aux vendeurs d'éditer leurs informations en cliquant sur le bouton "Enregistrer", telles que le nom de la boutique, le nom, le prénom, l'adresse e-mail, le numéro de téléphone, le numéro de fax et l'adresse de la boutique.

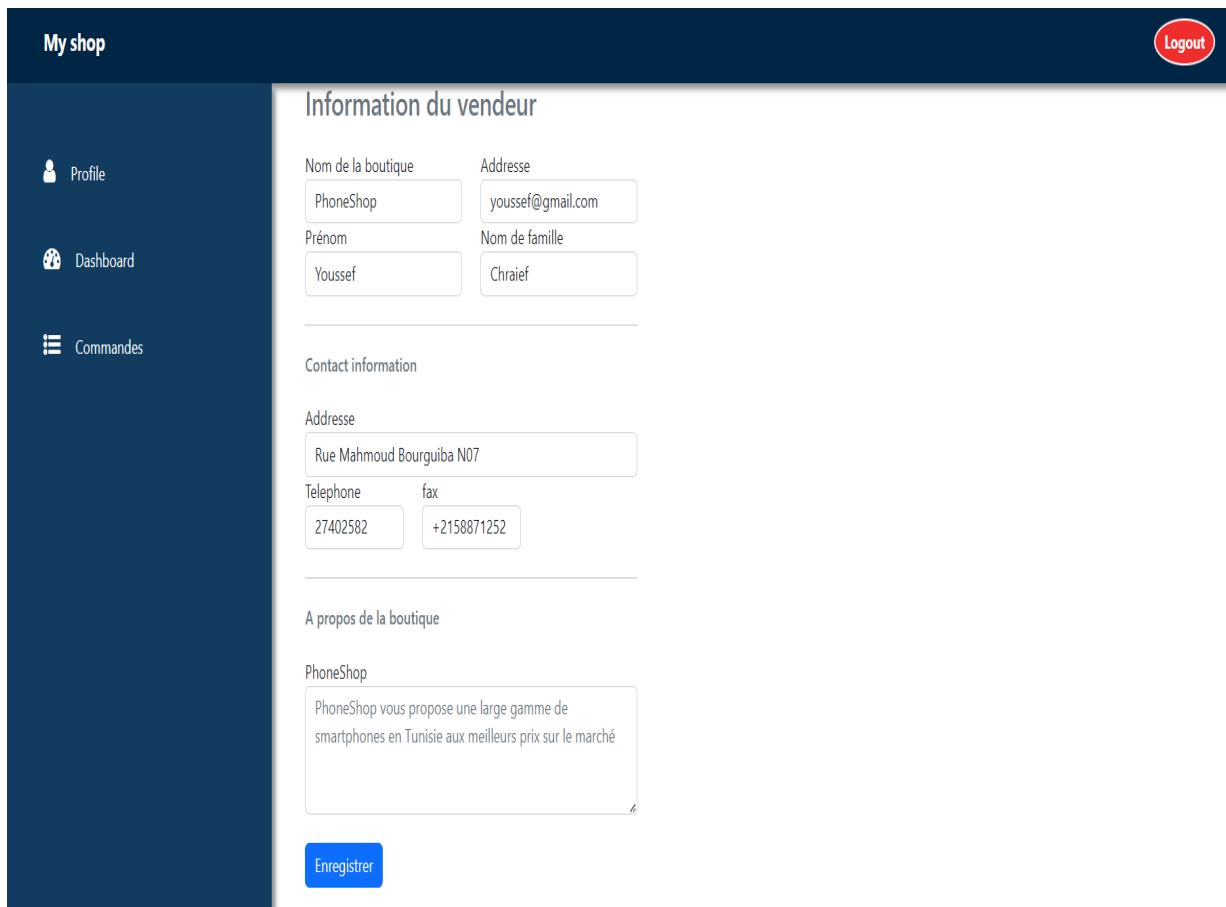


FIGURE 5.13 – Profile d'un vendeur.

La page "Dashboard" (figure 5.14) illustre la présentation des statistiques de revenu de chaque boutique. Nous utilisons différents types de graphiques (barres, courbes, circulaires) pour permettre une visualisation complète de ces informations.

Ses graphes sont présentés à l'aide d'une bibliothèque JavaScript « Chart.js » qui facilite la création des graphiques interactifs et assure une bonne visualisation de données.

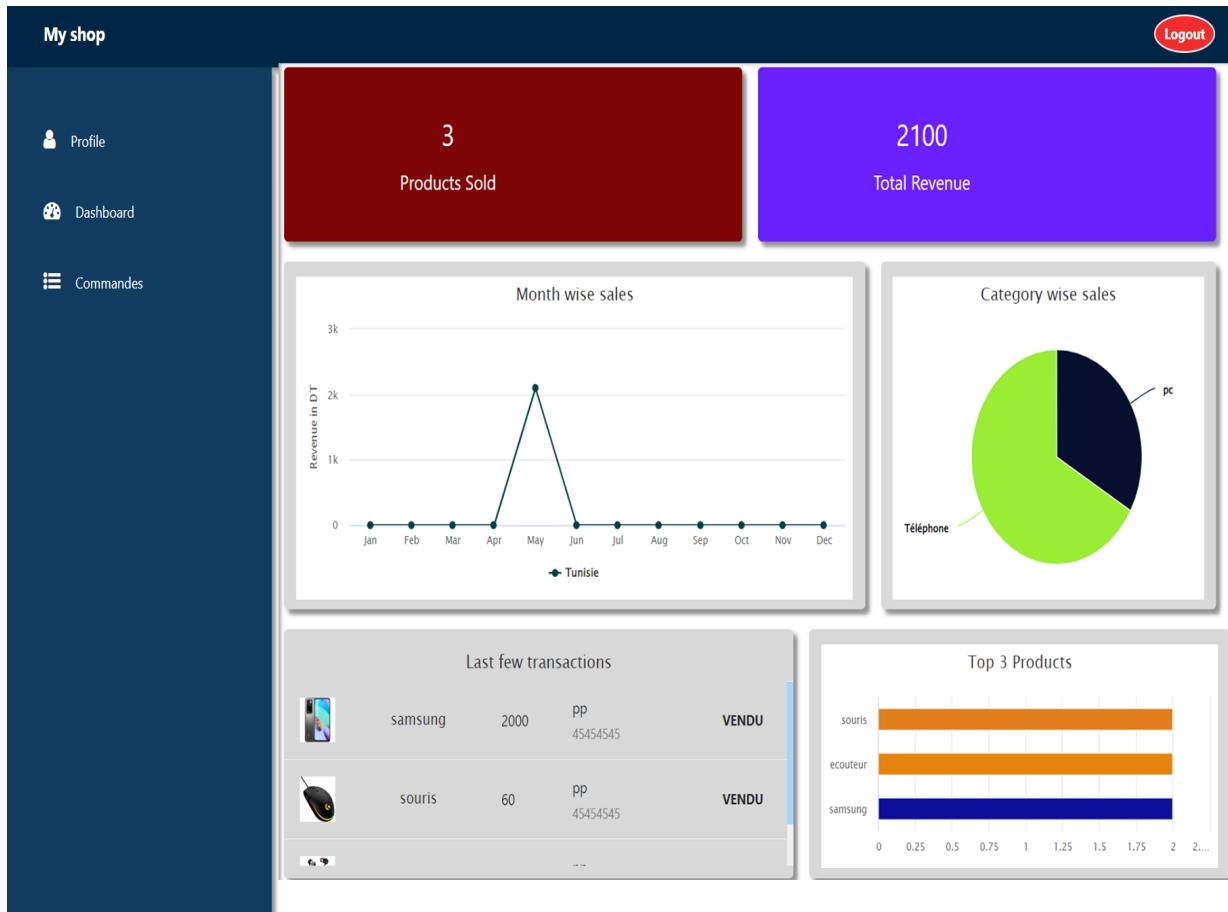


FIGURE 5.14 – Dashboard.

Tout d'abord, on peut visualiser le nombre total de produit vendu ainsi que le revenu total pour une boutique spécifique. La courbe à gauche permet de visualiser tous les produits vendus par chaque mois. Le diagramme circulaire présente le pourcentage des ventes par catégorie. Au-dessous à droite, le graphique en barre présente les trois premiers produits les plus vendus en termes de quantité. Au-dessous à gauche, on remarque toutes les dernières transactions de la boutique.

La page "Commandes" (figure 5.15) est l'endroit où les vendeurs peuvent voir toutes les commandes passées par les clients. Ils peuvent voir les détails de chaque commande, y compris les produits commandés, les quantités, les prix, les dates des achats et les catégories.

My shop		
		Logout
Profile	 <p>nom produit : souris quantité : 2 prix : 60 date : Sat May 06 2023 14:21:58 GMT+0100 (UTC+01:00)</p>	nom de la catégorie pc
Dashboard	 <p>nom produit : écouteur quantité : 2 prix : 40 date : Sat May 06 2023 14:22:30 GMT+0100 (UTC+01:00)</p>	nom de la catégorie Téléphone
Commandes	 <p>nom produit : samsung quantité : 2 prix : 2000 date : Sat May 06 2023 14:22:31 GMT+0100 (UTC+01:00)</p>	nom de la catégorie Téléphone

FIGURE 5.15 – Liste des commandes.

5.2.7 Interface « Détail du produit »

La figure 5.16 montre la description du produit, où le client a la possibilité de saisir une quantité pour passer une commande.



FIGURE 5.16 – Détail d'un produit.

5.2.8 Interface « Comparer les produits »

Dans l'interface présentée dans la figure 5.17, lorsque le client choisit un produit pour le comparer en cliquant sur l'icône "2-colonnes", ce produit est ajouté à une liste de compa-

raison. Un bouton "Comparer" s'affiche, ce qui permet au client d'ajouter d'autres produits à la liste pour la comparaison.

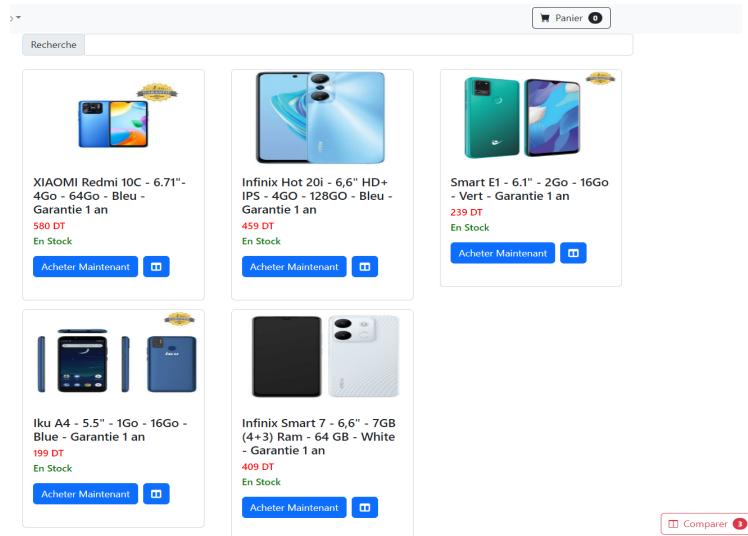


FIGURE 5.17 – Comparer les produits.

En appuyant sur le bouton "Comparer", la figure 5.18 est affichée, qui compare les produits sélectionnés par paires, offrant ainsi au client une vue facile des différences entre chaque produit.

The interface shows three selected products from the previous screen:

- XIAOMI Redmi 10C**: 6.71" - 4Go - 64Go - Bleu - Garantie 1 an. Priced at 580 DT, in stock. Button: Select.
- Infinix Hot 20i**: 6.6" HD+ IPS - 4GO - 128GO - Bleu - Garantie 1 an. Priced at 459 DT, in stock. Button: Select.
- Infinix Smart 7**: 6.6" - 7GB (4+3) Ram - 64 GB - White - Garantie 1 an. Priced at 409 DT, in stock. Button: Select.

A blue 'Compare' button is located above the products.

Product Results

Product	XIAOMI Redmi 10C - 6.71" - 4Go - 64Go - Bleu - Garantie 1 an	Infinix Hot 20i - 6.6" HD+ IPS - 4GO - 128GO - Bleu - Garantie 1 an
Couleur	bleu	bleu
Description courte	La caméra du Infinix Hot 20i est équipée d'autofocus.	Équipé d'un rare grand écran de 6.71 pouces au même prix.
Description longue	La caméra du Infinix Hot 20i est équipée d'autofocus. L'autofocus est une fonction de caméra qui règle avec précision le focus de la caméra, c'est une fonction appréciable de ce smartphone.	Équipé d'un rare grand écran de 6.71 pouces au même prix, le monde bouleversant du cinéma vous attend. Redmi 10C prend également en charge Netflix HD et Prime Video HD pour une bonne expérience de streaming.

FIGURE 5.18 – Tableau de comparaison.

5.2.9 Interface « Panier »

La figure 5.19 affiche tous les produits ajouter au panier par le client. Aussi le client peut gérer ses achats à partir de son panier en effet, il peut supprimer un produit ou bien confirmer son achat en cliquant sur le bouton registre.

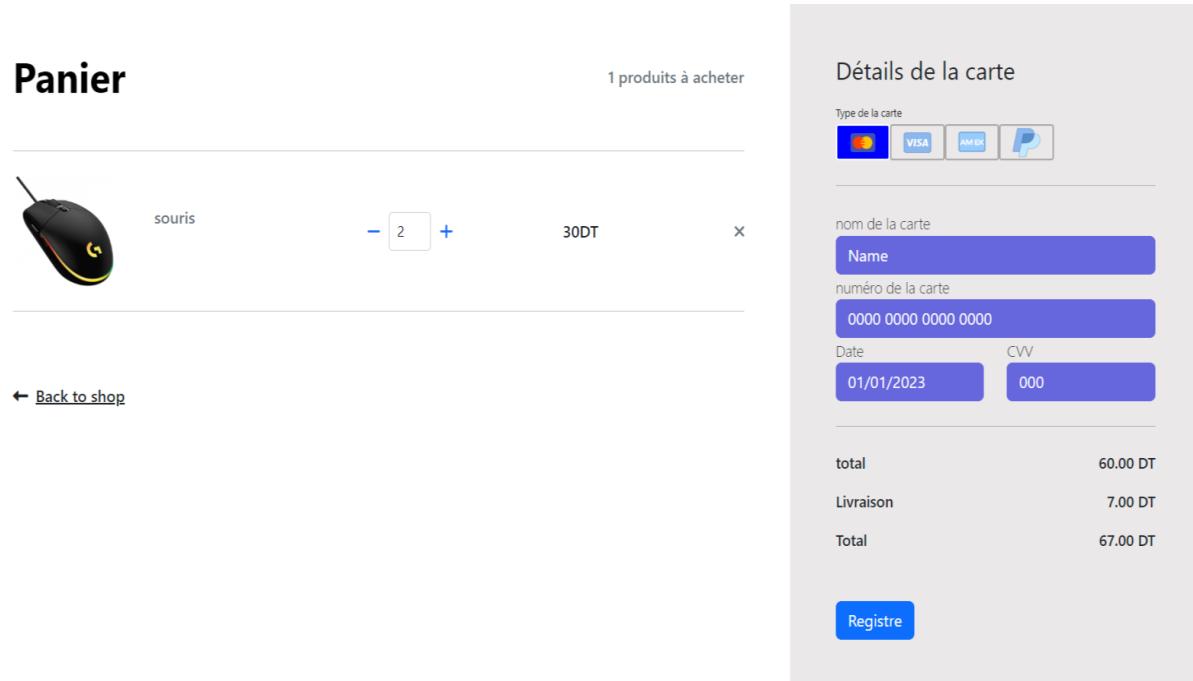


FIGURE 5.19 – Panier d'un client.

5.3 Chronogramme du projet

Le projet a eu lieu pendant une période de 3 mois, s'étalant du 08/02/2023 au 15/05/2023. Le tableau 5.2 ci-dessous présente un planning détaillant les principales étapes qui ont permis de parvenir à une solution fonctionnelle répondant aux critères énoncés dans le chapitre 2.

TABLE 5.2 – Chronogramme du projet

	Février				Mars				Avril				Mai	
	1	2	3	4	1	2	3	4	1	2	3	4	1	2
Étude théorique et documentation	■		■											
Analyse et spécification des besoins			■	■										
Conception					■	■								
Implémentation							■	■	■	■	■	■		
Tests et intégration							■	■	■	■	■	■		
Rédaction du rapport			■	■	■	■	■	■	■	■	■	■		

Conclusion

Dans ce chapitre, nous avons présenté les différents éléments techniques de notre projet. Nous avons commencé par une étude des langages de programmation et bibliothèques utilisés dans notre environnement de développement. Ensuite, nous avons décrit les fonctionnalités offertes par notre application à travers des captures d'écran. Enfin, nous avons inclus le chronogramme de notre projet pour donner une vision globale de l'avancement de notre travail.

Conclusion et perspectives

Notre projet de conception et de développement consiste à concevoir et à implémenter un portail de vente en ligne. Cette application doit permettre d'aider toute personne à créer son propre boutique en ligne et de le gérer de façon personnalisée.

Dans ce rapport nous avons détaillé les différentes étapes par lesquelles nous avons passé pour arriver au résultat attendu.

Nous avons commencé tout d'abord par la présentation de la problématique à résoudre et les objectifs à atteindre suivie d'une étude préalable des solutions existantes pour nous aider à spécifier les besoins demandés de notre système et d'analyser toutes les fonctionnalités à livrer. Ensuite, nous avons passé à la phase de conception, cette phase nous a permis de bien structurer notre projet et faciliter l'implémentation. Finalement, nous avons présenté la phase de réalisation au cours de laquelle nous avons mentionné les technologies utilisées et les environnements de travailler pour le développement de notre projet ainsi des aperçus de captures d'écran pour mieux expliqué le fonctionnement des diverses fonctionnalités. Ce projet nous a permis de développer nos compétences théoriques, pratiques et de maîtriser les outils de développement web les frameworks Angular et Spring boot. Malgré les difficultés rencontrées, nous avons réussi à atteindre notre objectif.

Notre portail est ouvert à plusieurs améliorations. En effet, nous pouvons ajouter des fonctionnalités pour enrichir notre site. Parmi lesquelles :

- Assurer une interface de communication entre le vendeur et le client pour que ce dernier puisse avoir plus d'information sur les produits existants ou bien pour commander un produit.
- Permettre au client de donner des commentaires et des avis sur la boutique et sur la qualité des produits.
- Intégrer des moyens de paiement aux clients lors de leurs achats.

Bibliographie

- [1] Cycle en V en gestion de projet : définition et méthode. https://www.manager-go.com/gestion-de-projet/cycle-en-v.htm?fbclid=IwAR0ffGcw8B9RVgGXLLB_PkXa03ndPRZVaM7dBebX51VqqKvaCT4mRmtIutY/, [consulté le 15/02/2023].
- [2] Le pattern MVC. <https://mouradev.wordpress.com/2015/10/29/le-pattern-mvc/>, [consulté le 18/02/2023].
- [3] Qu'est-ce que Shopify et comment ça marche ? (2023). <https://www.shopify.com/blog/what-is-shopify>, [consulté le 22/02/2023].
- [4] QU'EST-CE QUE WOOCOMMERCE ? UN GUIDE DU COMMERCE ÉLECTRONIQUE WORDPRESS. <https://multilingualpress.org/what-is-woocommerce/>, [consulté le 22/02/2023].
- [5] E-COMMERCE MULTICANAL Vendez en ligne où que se trouvent vos clients. <https://www.myeasy-store.in/about>, [consulté le 22/02/2023].
- [6] Introduction aux concepts angulaires. <https://angular.io/guide/architecture/>, [consulté le 01/04/2023].
- [7] Bootstrap : définition, tutoriels, astuces, pratiques. <https://www.journaldunet.com/web-tech/developpeur/1159810-bootstrap-definition-tutoriels-astuces-pratiques/>, [consulté le 01/04/2023].
- [8] Python : définition et utilisation de ce langage informatique. <https://www.journaldunet.fr/web-tech/dictionnaire-du-webmastering/1445304-python-definition-et-utilisation-de-ce-langage-informatique/>, [consulté le 02/05/2023].
- [9] Qu'est-ce que Spring Boot ? <https://stackify.com/what-is-spring-boot/>, [consulté le 01/04/2023].
- [10] Qu'est-ce que MySQL ? Tout ce que tu as besoin de savoir. <https://www.talend.com/resources/what-is-mysql/>, [consulté le 01/04/2023].
- [11] Qu'est-ce que Visual Studio Code ? L'éditeur de code extensible de Microsoft. <https://www.infoworld.com/article/3666488/what-is-visual-studio-code-microsofts-extensible-code-editor.html>, [consulté le 01/04/2023].

- [12] Qu'est-ce que le facteur ? <https://www.postman.com/product/what-is-postman/>, [consulté le 01/04/2023].
- [13] Présentation d'IntelliJ IDEA. <https://www.jetbrains.com/help/idea/discover-intellij-idea.html>, [consulté le 01/04/2023].
- [14] StarUML Présentation du logiciel. <https://dictionnaire.sensagent.com/STARUML/fr-fr/>, [consulté le 01/04/2023].
- [15] Pycharm - Présentation. https://www.tutorialspoint.com/pycharm/pycharm_introduction.htm, [consulté le 02/05/2023].