** Ministère de l’enseignement supérieur et de la recherche scientifique**

**Direction générale des études technologiques**

**Institut supérieur des études technologiques de Djerba**

**Département technologie de l’informatique**

|  |  |
| --- | --- |
| **Code Projet** |  |

**Rapport de**

**STAGE DE FIN D’ETUDES**

**Présenté en vue de l’obtention du diplôme de**

**Licence Appliquée en Technologies de l’informatique**

**Parcours : MDW**

**Site de vente et achat en ligne**

**Elaboré par :**

Hfaiedh youssef

**Encadré par :**

M. Friaa Iskander

**Effectué à :** Online

**Encadrant** : Mme. Challakhi Souad

**Année** **universitaire** : 2020**/**2021

**Dédicace**

Je dédie ce travail à mes parents, aucun mot ne pourrait exprimer ma grande reconnaissance pour les sacrifices qu’ils ont déployés tout au long de ma scolarité. J’espère qu’ils trouveront dans ce travail le fruit de leurs efforts et l’expression de mon estime et de mon amour infini.

A mes amis et mes collègues pour leur encouragement et pour tous les bons moments qu’on a vécu ensemble.

Et enfin, à tous mes camarades de la promotion 2020/2021, je leur souhaite une carrière à la hauteur de leurs ambitions.

***Hfaiedh Youssef***

**Remerciements**

C’est avec un grand plaisir que je réserve cette page en signe de ma gratitude et de profonde reconnaissance à tous ceux qui m’ont aidé lors de la réalisation de ce travail.

J’offre premièrement de sincères et chaleureux remerciements à mon encadrant académique, **M. Friaa Iskander** qui m’a inspiré et guidé avec ses conseils précieux tout au long de ce projet.

De mêmes, je tiens à remercier mon encadrante de stage, **Mme. Challakhi Souad** pour son encadrement, son assistance, sa disponibilité, ses conseils et surtout son aide précieuse durant ce stage.

Mes remerciements s’adressent également à tous mes professeurs du département informatique pour la qualité de l’enseignement qu’ils ont bien voulu me prodiguer durant mes études afin de me fournir une formation efficiente.

Finalement, avec un grand honneur, j’adresse mes remerciements à tous les membres de jury, d’avoir accepté l’évaluation de ce modeste travail.

**Sommaire**

[Introduction générale 1](#_Toc74935020)

[Chapitre 1 : Présentation Générale 2](#_Toc74935021)

[I. Introduction 2](#_Toc74935022)

[II. Présentation de la société d’accueil 2](#_Toc74935023)

[III. Cadre général du travail 2](#_Toc74935024)

[IV. Présentation de l’application 3](#_Toc74935025)

[V. Méthodologie et formalise adoptés 3](#_Toc74935026)

[1. Scrum 3](#_Toc74935027)

[*a)* *Planification du Projet* 3](#_Toc74935028)

[*b)* *Equipes et Rôles* 4](#_Toc74935029)

[*2.* *UML* 5](#_Toc74935030)

[VI. Conclusion 5](#_Toc74935031)

[Chapitre 2 : Etude Préalable 6](#_Toc74935032)

[I. Introduction 6](#_Toc74935033)

[II. Etude de l’existant 6](#_Toc74935034)

[1. Analyse de l’existant 6](#_Toc74935035)

[2. Critique de l’existant 10](#_Toc74935036)

[3. Solution proposée 10](#_Toc74935037)

[III. Spécification des exigences 10](#_Toc74935038)

[1. Les spécifications des besoins 10](#_Toc74935039)

[a) Identification des acteurs 10](#_Toc74935040)

[b) Les besoins fonctionnels 11](#_Toc74935041)

[c) Les besoins non fonctionnels 11](#_Toc74935042)

[IV. Diagramme de cas d’utilisation 12](#_Toc74935043)

[1. Scénarios et des cas d’utilisation 12](#_Toc74935044)

[*a)* *Diagramme de cas d'utilisation global* 12](#_Toc74935045)

[*b)* *Diagramme de cas d'utilisation de l'authentification* 12](#_Toc74935046)

[c) Description textuelle du diagramme des cas d'utilisation 13](#_Toc74935047)

[V. Conclusion 14](#_Toc74935048)

[Chapitre 3 : Conception 15](#_Toc74935049)

[I. Introduction 15](#_Toc74935050)

[II. Architecture du système 15](#_Toc74935051)

[1. Architecture trois tiers 15](#_Toc74935052)

[2. Patron de conception MVVM (Modèle Vue Vue-Modèl) : 16](#_Toc74935053)

[III. Conception 17](#_Toc74935054)

[1. Modélisation conceptuelle 17](#_Toc74935055)

[2. Diagramme de classe 17](#_Toc74935056)

[3. Correspondance entre modèles objets et relationnels 18](#_Toc74935057)

[4. Diagramme de séquences 20](#_Toc74935058)

[a) Diagramme de séquence « Authentification » 20](#_Toc74935059)

[a) Diagramme de séquence « Inscription » 22](#_Toc74935060)

[IV. Conclusion 24](#_Toc74935061)

[Chapitre 4 : Réalisations 25](#_Toc74935062)

[I. Introduction 25](#_Toc74935063)

[II. Environnement de travail 25](#_Toc74935064)

[1. Environnement matériel 25](#_Toc74935065)

[2. Environnement logiciel 25](#_Toc74935066)

[III. Choix techniques 26](#_Toc74935067)

[1. Outils de développement 26](#_Toc74935068)

[2. Base de données 27](#_Toc74935069)

[IV. Conclusion 27](#_Toc74935070)

[Webographie 27](#_Toc74935071)

**Liste des figures**

[**Figure 1 : Logo de société « Online Djerba »** 2](#_Toc74935072)

[**Figure 2 : Logo de jumia** 6](#_Toc74935073)

[**Figure 3 : Logo du tayara** 6](#_Toc74935074)

[**Figure 4 : Logo du Dabchy** 6](#_Toc74935075)

[**Figure 5 : Interface d’accueil du site Jumia** 7](#_Toc74935076)

[**Figure 6 : Interface d’accueil du site tayara** 8](#_Toc74935077)

[**Figure 7 : Interface d’accueil du site Dabchy** 9](file:///C:\Users\user\OneDrive\Bureau\rapport\youssef-rapport.docx#_Toc74935078)

[**Figure 8 : Diagramme de cas d’utilisation globale** 12](#_Toc74935079)

[**Figure 9 : Raffinement de cas d’utilisation « Authentification »** 13](#_Toc74935080)

[**Figure 10 : L'architecteur 3-tiers** 15](#_Toc74935081)

**Liste des tableaux**

[**Tableau 1 : Equipe de Scrum** 4](#_Toc74935082)

[**Tableau 2 : Etude de cas de plateforme « Jumia »** 7](#_Toc74935083)

[**Tableau 3 : Etude de cas de plateforme « Tayara »** 8](#_Toc74935084)

[**Tableau 4 : Etude de cas de plateforme « Dabchy »** 9](#_Toc74935085)

[**Tableau 5 : Scénario de cas « S’authentifier »** 13](#_Toc74935086)

[**Tableau 6 : Scénario de cas « S’inscrire »** 14](#_Toc74935087)

# **Introduction générale**

Le commerce électronique (E-Commerce) a connu une croissance fulgurante ces dernières années. Les techniques et méthodes qui permettent de vendre de manière rapide et efficace se sont aussi développées. De grands sites tunisiens comme **Tayara, Jumia** et **Dabchy** sont aujourd’hui spécialisés dans les ventes et achats en ligne mais présentent quelques manques.

Avec l’avancée et l’émergence des technologies web, et suite aux constatations faites à l’issue de l'étude de la situation existante, j’ai décidé de profiter de ces technologies pour mettre en œuvre une application qui offre les fonctionnalités nécessaires du partage (achat et vente) des différents produits en ligne. Ainsi la plateforme permettra aux consommateurs d’acheter et vendre ce qu’ils souhaitent d’une manière facile et sécurisée.

Dans un premier temps, je vais présenter l’entreprise, le cadre du stage et décrire mon application et la méthodologie utilisé. Ensuite je vais passer à l’analyse des besoins en indiquant : les besoins fonctionnels et non fonctionnels, une étude de l’existant et des solutions pour la réalisation. Dans le troisième chapitre, je vais décrire la phase de conception, en présentant des diagrammes du langage UML comme : les diagrammes de cas d’utilisation en précisant les acteurs impliqués dans cette application, le diagramme de classes, les diagrammes des séquences et le diagramme de composants. Enfin, je vais décrire la phase de la réalisation et du codage de l’application en indiquant les outils informatiques utilisés tout au long du développement et présenter les interfaces inclues dans le site web et l’application mobile.

# **Chapitre 1 : Présentation Générale**

# **Introduction**

Dans ce chapitre, je vais vous présenter le cadre général du projet, en premier lieu, l’organisme d’accueil du stage, puis le cahier des charges du projet.

# **Présentation de la société d’accueil**

La société Online, située à Djerba, a été fondée en 2018. Online est une société de conseil en organisation et en ingénierie des systèmes d’information qui a pour mission de bâtir des solutions Internet de qualité, en offrant des conseils, des technologies et des services adéquats. Chez Online, chaque projet est analysé avec une attention particulière afin de développer des sites fiables qui répondent aux besoins des clients.



**Figure 1 : Logo de société « Online Djerba »**

# **Cadre général du travail**

Le projet entre dans le cadre du stage de fin d'études pour l'obtention de la licence appliquée en Multimédia Développement Web (MDW) de l'Institut Supérieur des Études Technologiques de Djerba. Ce projet a été effectué au sein de la société Online durant la période du 15/02/2021 jusqu’à début juin.

# **Présentation de l’application**

La communauté tunisienne souffre des prix élevés, c'est pour ça yourShop va créer un marché compétitif en donnant automatiquement des prix plus bases et des choix différents avec des clients satisfaits. Yourshop est un site e-commerce qui permet à l'utilisateur de vendre et acheter des produits (vêtements, électroniques, décoration ...) neufs et d'occasion d'une manière rapide, facile et efficace. Le business modèle de ce projet peut être B2C, C2B, B2B et B2C dont il est basé sur l'économie de partage (sharing Economy). Yourshop disposera d’une interface conviviale qui permettront aux tunisiens de commencer leur business en ligne en assurant la vente et le contact entre le vendeur et l'utilisateur.

# **Méthodologie et formalise adoptés**

## **Scrum**

Le cadre de travail Scrum est une méthode agile qui permet de produire un logiciel dans une courte durée. Ce processus s’appuie sur la planification en boite de temps, nommés sprints. La durée des sprints est comprise entre deux semaines et un mois.

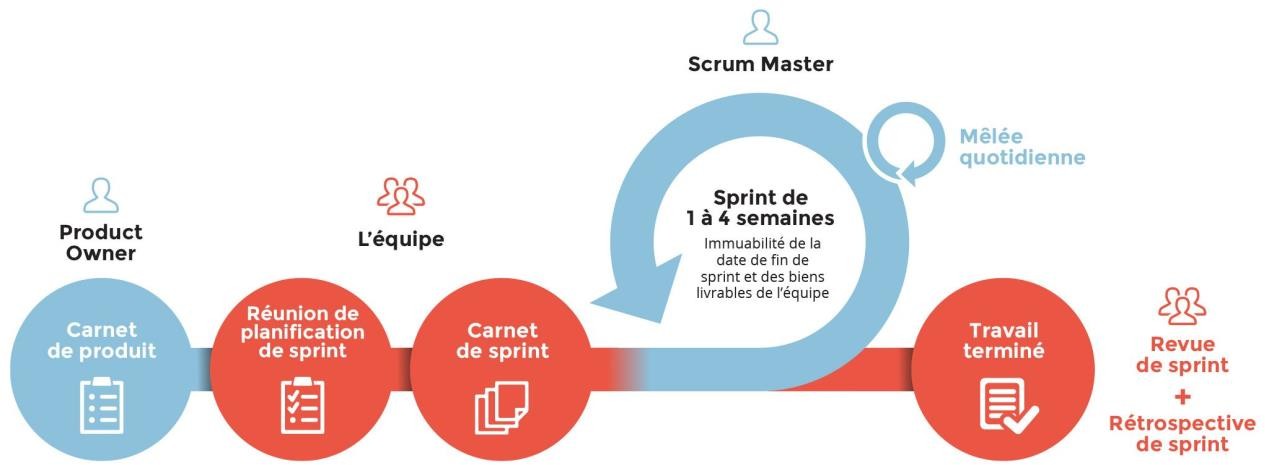


Figure 1.3 : cycle de vie de Scrum

## ***Planification du Projet***

La vie d'un projet Scrum est basée sur un ensemble de réunions définies et limitées dans le temps.

* **Les réunions quotidiennes :** chaque jour, toute l’équipe se réunit, généralement debout, pendant 15 minutes environ pour répondre aux trois questions suivantes :
  + Qu’ai-je fait hier ?
  + Que vais-je faire aujourd’hui ?
  + Y a-t-il un obstacle gênant aujourd’hui ?
* **Les réunions de planifications :** toute l’équipe se réunit pour décider des fonctionnalités qui vont composer le sprint suivant et mettre à jour la liste générale.
* **Les réunions de revue de travail :** Au cours de cette réunion qui a lieu à la fin du sprint, l'équipe de développement présente les fonctionnalités achevées au cours du sprint
* **Les réunions de rétrospective :** À chaque fin de sprint, l’équipe fait le point sur ce qui a bien fonctionné et sur ce qui a moins bien fonctionné.

## ***Equipes et Rôles***

La méthode SCRUM fait intervenir trois rôles principaux qui sont :

|  |  |  |
| --- | --- | --- |
| Acteurs | Tâches | Rôles |
| * Friaa Iskander | * Responsable des conseils * Responsable de l'approbation des tâches | Scrum master |
| * Challakhi Souad | * Tests et validation * Déploiement * Conception | Product Owner |
| * YOUSSEF HFAIEDH | * Développement de l'application | Développeur |

**Tableau 1 : Equipe de Scrum**

* **Product owner :** Porte la vision du produit à réaliser (représentant généralement le client). C’est lui qui va définir et prioriser la liste des fonctionnalités du produit.
* **Scrum master :** Facilitateur du projet et porte aussi une attention au respect des différentes étapes de SCRUM.
* **Équipe :** est constituée de trois à neuf personnes et a pour responsabilité de livrer à chaque fin d’itération une nouvelle version de l’application.

## ***UML***

UML, On le traduit par « Langage de modélisation unifié ». La notation UML est un **langage visuel** constitué d’un ensemble de schémas, appelés des **diagrammes**, qui donnent chacun une vision différente du projet à traiter. UML nous fournit donc des diagrammes pour **représenter** le logiciel à développer : son fonctionnement, sa mise en route, les actions susceptibles d’être effectuées par le logiciel, etc. [1]

# **Conclusion**

Au terme de ce chapitre, nous avons présenté le cadre de notre projet ainsi que la description générale de la société d’accueil aussi la méthodologie adoptée. Pour bien comprendre notre projet nous allons évoquer la phase d’étude de l’existant et la spécification des besoins de notre projet dans le chapitre suivant.

# **Chapitre 2 : Etude Préalable**

# **Introduction**

L’étude préalable est sans doute l’une des phases les plus importantes en matière de développement des sites web. Dans ce chapitre, je vais commencer par une partie d’analyse dans laquelle je vais essayer de dégager les spécifications fonctionnelles de mon projet. Enfin, je mets en relief ces besoins en se basant sur une représentation UML.

# **Etude de l’existant**

# **Analyse de l’existant**

Nous trouvons plusieurs applications qui répondent aux besoins des ventes et achats en ligne. Parmi ces applications, nous citons les applications étudiées ci-dessous.

Cette partie consiste à introduire les 3 meilleures solutions existantes dans le marché tunisienne.

|  |  |  |
| --- | --- | --- |
| **Figure 2 : Logo de jumia** | **Figure 3 : Logo du tayara** | **Figure 4 : Logo du Dabchy** |

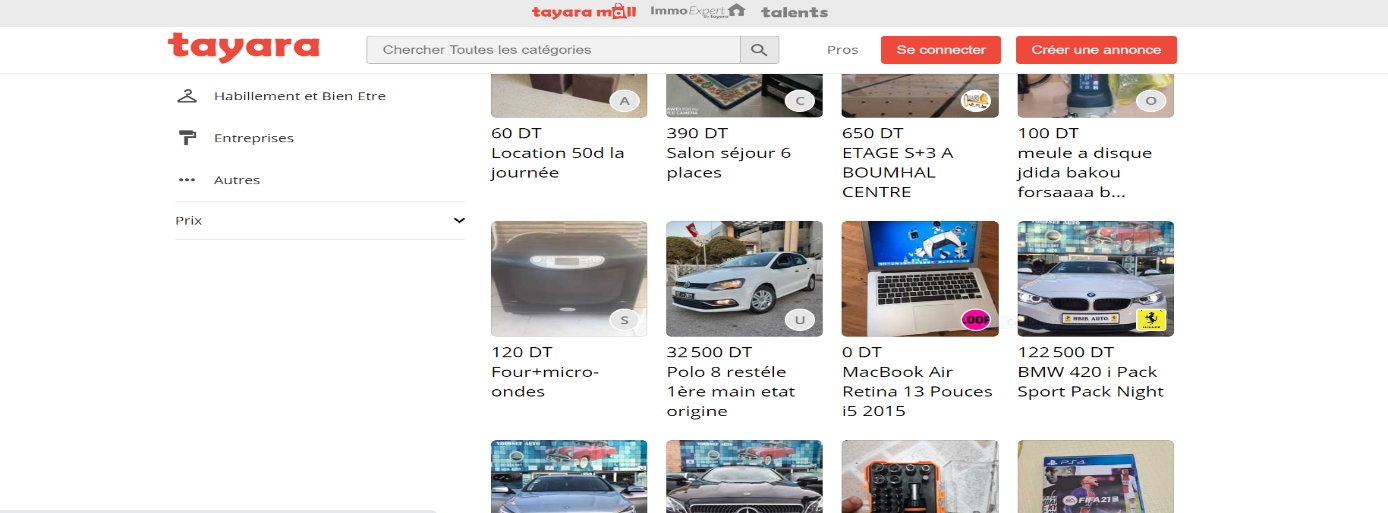
* **Jumia** :
* **Jumia :** La figure ci-dessous présente l’interface de la plateforme Jumia.tn.

**Figure 5 : Interface d’accueil du site Jumia**

|  |  |  |
| --- | --- | --- |
| **Description** | **Points faibles** | **Points forts** |
| * Jumia est une entreprise de commerce en ligne présente sur le marché africain et fondée en 2012. | * Il faut avoir une société pour vendre. * Vente des produits nouveaux seulement. | * A un système marketing très développer. |

**Tableau 2 : Etude de cas de plateforme « Jumia »**

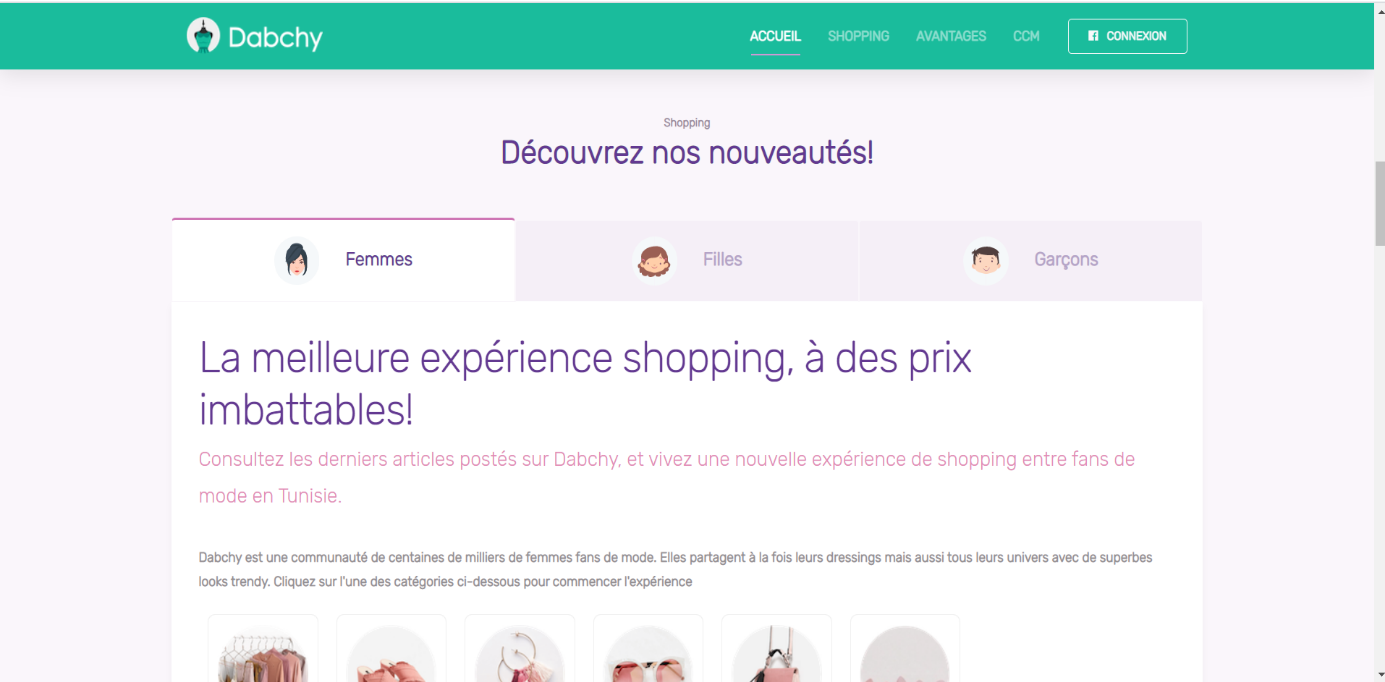
* **Tayara :**
* **Tayara :** La figure ci-dessous présente l’interface de la plateforme Tayara.tn.



**Figure 6 : Interface d’accueil du site tayara**

|  |  |  |
| --- | --- | --- |
| **Description** | **Points faibles** | **Points forts** |
| * Créé en 2012, tayara.tn est un site d'annonces pour vendre et acheter des biens d'occasion en ligne. | * Tayara.tn n'est pas responsable du processus d'achat et du contact entre le client et le vendeur. | * Tayara.tn a été classé en 2013 parmi les 100 sites les plus visités en Tunisie. Deux ans plus tard, le site compte environ 1,5 million de visites par mois, avec 100 millions de pages vues par mois, en faisant l'un des principaux sites du pays. |

**Tableau 3 : Etude de cas de plateforme « Tayara »**

* **Dabchy :**
* **Dabchy**: La figure ci-dessous présente l’interface de la plateforme Dabchy.com

**Figure 7 : Interface d’accueil du site Dabchy**

|  |  |  |
| --- | --- | --- |
| **Description** | **Points faibles** | **Points forts** |
| * Dabchy.com est vide-dressing qui permet aux femmes Tunisiennes d’acheter, vendre ou échanger en ligne des articles de mode neufs ou d’occasion. | * Manque de publicité * Cible limité (Vêtements de femme seulement) | * Interface simple et attirant. |

**Tableau 4 : Etude de cas de plateforme « Dabchy »**

# **Critique de l’existant**

En regardant d'autres sites de commerce électronique sur le marché́, j’ai constaté́ que leur utilisation est limitée. En prenant la plateforme Jumia comme exemple, j’ai remarqué́ que les utilisateurs ne peuvent s'inscrire qu'en tant que clients et gèrent la partie commerciale à travers des accords avec les marques et les entreprises. De plus, ce site e-commerce est destiné́ aux nouveaux produits. On retrouve aussi l'exemple de Dabchi, un site tunisien d'achat et de vente de vêtements en ligne. Le site Web de Dabchi ne traite que des catégories de vêtements. Le site Tayara.tn est également très connu sur le marché́ des produits d'occasion. Ce site permet aux internautes de poster leurs annonces et de visualiser des annonces selon la catégorie choisie. En revanche, Tayara.tn n'est pas responsable du processus d'achat entre le client et le vendeur.

# **Solution proposée**

Pour faire face aux limites citées dans la section précédente nous avons pensé à une application nommée «YourShop » qui permet au client de choisir et acheter parmi nos différentes catégories et garantir un paiement sécurisé et livraison rapide. Ainsi que de vendre de nouveaux articles ou bien des articles usés qui sont en bonne qualité ce qui permet aux consommateurs de gagner de l’argent.

# **Spécification des exigences**

# **Les spécifications des besoins**

## **Identification des acteurs**

* **Administrateur :**

C’est le seul acteur qui peut avoir l’accès complèt au système de l’application. Son principal rôle est la gestion des utilisateurs, transporteur, des catégories, des produits et recevoir les réclamations.

* **Client*:***

Cet utilisateur à la possibilité de vendre et acheté des produits, modifier son propre compte.

* **Transporteur*:***

Cet acteur est responsable a la livraison de commande et la récupération des produits de vendeur*.*

Ce type d’acteur permet la transaction du produit vendu entre les clients.

## **Les besoins fonctionnels**

Ce projet doit satisfaire certains besoins et doit préparer l’application contenant les modules nécessaires pour l’utilisateur final. Ces tâches à effectuer sont définies comme suit :

* S’authentifier : cette fonctionnalité est obligatoire pour l’utilisateur afin d’avoir accès au contenu de l’application.
* S’inscrire : Pour profiter des services offerts par notre application l’utilisateur doit créer un compte.
* Modifier compte : L’utilisateur peut modifier les données relatives de son compte.
* Vendre produits : Pour que l’utilisateur peut ajouter des produits à vendre, à travers l’application, il doit indiquer la description, la taille, la couleur, l’état, les catégories et des photos du produit.
* Acheter produit : Pour que l’utilisateur confirmer l’achat, il doit ajouter le produit au panier puis passer la commande.
* Envoyer réclamation : l’utilisateur de notre application a la permission d’envoie une réclamation.
* Gérer utilisateur : L’administrateur à la possibilité de supprimer l’utilisateur (Client, Transporteur).
* Gérer les réclamations : l’administrateur peut répondre aux réclamations envoyer par les clients.
* Gérer produit : L’administrateur à la possibilité de supprimer les produits.

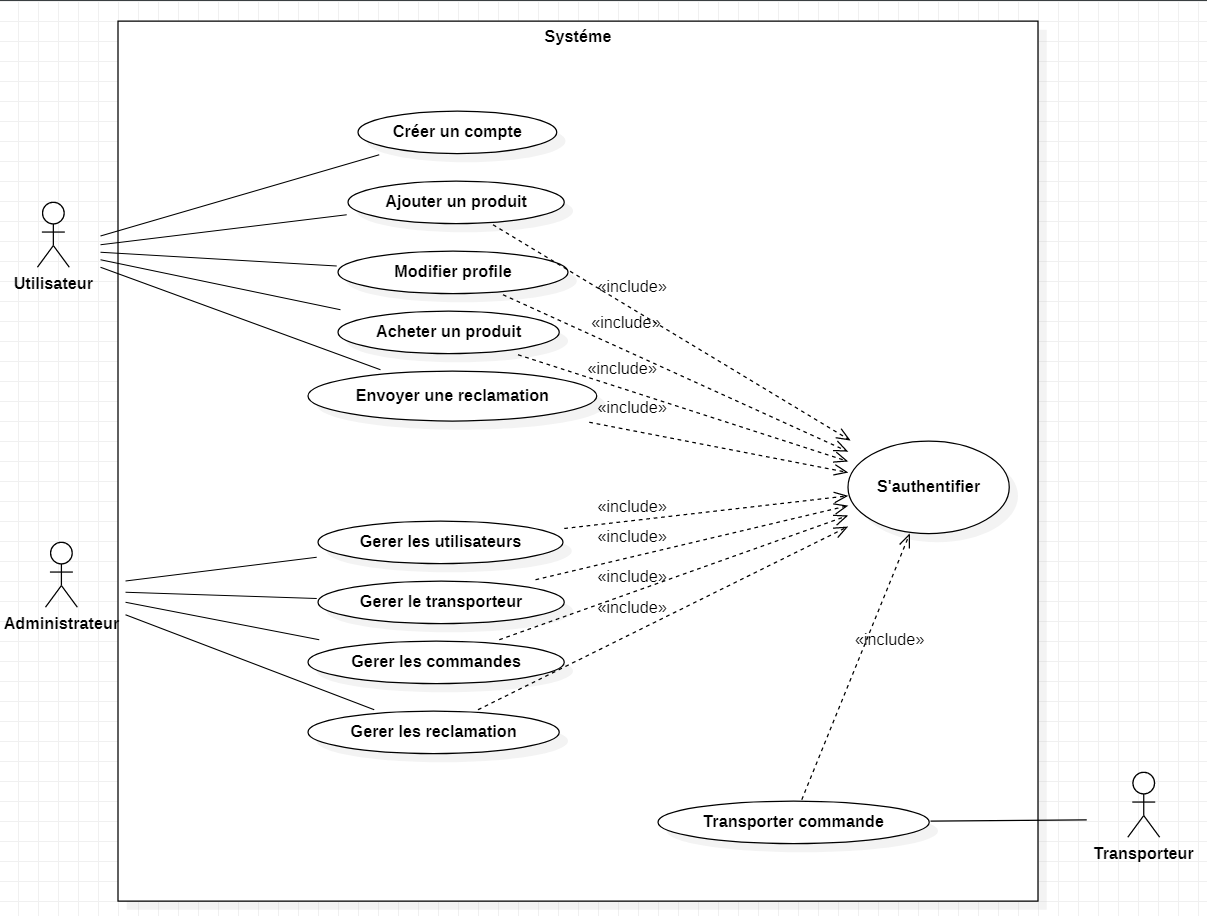
## **Les besoins non fonctionnels**

* Assurer une cohérence et une crédibilité aux informations stockées dans la base de données.
* L’ergonomie des interfaces de l’application mobile et web.
* Sécurité : lors de la création d’un compte pour éviter que l’application soit une victime des robots et des hackers, j’ai utilisé les « JSON Web Token » ou JWT sont des jetons générés par un serveur lors de l’authentification d’un utilisateur sur une application Web, et qui sont ensuite transmis au client.
* Temps de réponse : Le temps de réponse doit être le plus court possible.
* La portabilité : l’application doit s’exécuter sans problème sur tous les navigateurs. Il est ainsi primordial qu’elle soit compatible avec les différents appareils Android.

# **Diagramme de cas d’utilisation**

# **Scénarios et des cas d’utilisation**

Le diagramme de cas d’utilisation illustre d’une manière globale les différentes actions que peut faire l’utilisateur c’est-à-dire l’interaction qui existe entre l’utilisateur et l’application.

1. **Diagramme de cas d'utilisation global

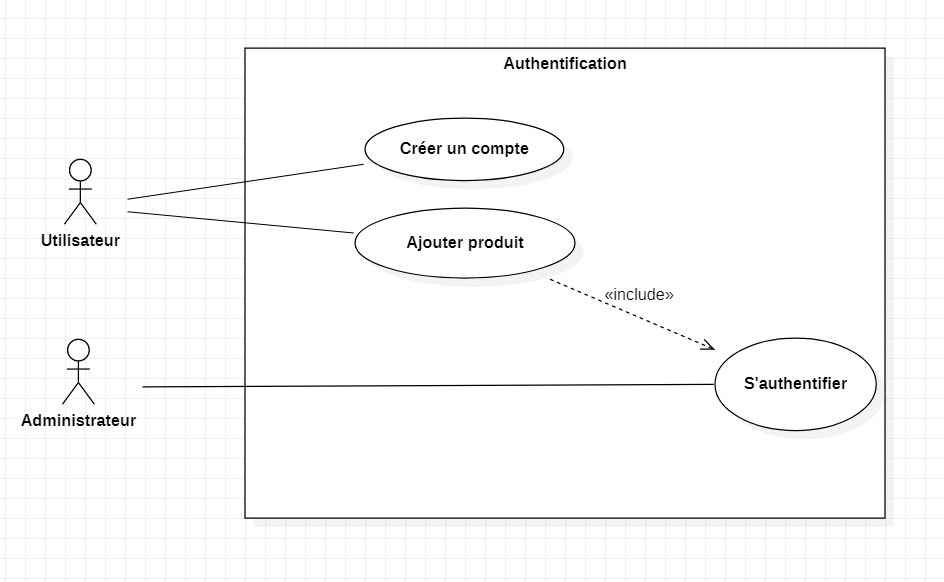
**Figure 8 : Diagramme de cas d’utilisation globale**

## Diagramme de cas d'utilisation de l'authentification

Description textuelle du cas d'utilisation « Authentification » :

|  |  |
| --- | --- |
| Cas d’utilisation | S’authentifier |
| Acteur | Administrateur, utilisateur, transporteur |
| Description | L’acteur s’authentifie en utilisant son login et son mot de passe pour accéder aux différentes fonctionnalités de l’application selon son rôle. |
| Pré condition | L’acteur dispose d’un compte valide. |
| Post condition | Authentification effectuée. |
| Scénario de base | 1. L’acteur demande la page authentification.  2. Le système affiche la page d’authentification.  3. L’acteur saisit son login et son mot de passe.  4. Le système vérifie les données de l’acteur.  5. Le système redirige l’acteur vers la page d’accueil selon son rôle. |
| Scénario de l’exception | 4. Données erronées (Acteur inexistant ou mot de passe ou login invalide.) :  - Le système affiche un message d’erreur.  - Reprise de l’étape 2 du scénario de base. |

**Tableau 5 : Scénario de cas « S’authentifier »**



**Figure 9 : Raffinement de cas d’utilisation « Authentification »**

## Description textuelle du diagramme des cas d'utilisation

Description textuelle du cas d'utilisation « S’inscrire » :

|  |  |
| --- | --- |
| Cas d’utilisation | Créer un compte |
| Acteur | Utilisateur |
| Description | L’utilisateur crée un compte en introduisant ses données pour accéder aux différentes fonctionnalités de l’application. |
| Pré condition | Opération d’inscription choisie. |
| Post condition | Utilisateur enregistré. |
| Scénario de base | 1. Choisit la page d’inscription  2. L’utilisateur saisie ses données personnelles  3. l’utilisateur valide l’inscription.  4. Le système effectue l’opération.  5. Le système envoie l’utilisateur à la page d’accueil |
| Scénario de l’exception | 4. données non valides :  - Un message d’erreur s’affiche  - Retourne à l’étape 2 du scénario de base. |

**Tableau 6 : Scénario de cas « S’inscrire »**

# **Conclusion**

L’analyse des besoins nous a permis d’avoir une vision claire du sujet et une compréhension plus profonde des tâches à réaliser. En effet, dans ce deuxième chapitre nous avons identifié les différents acteurs impliqués dans notre application ainsi que leurs besoins fonctionnels et non fonctionnels. Ensuite, nous avons présenté les différents diagrammes de cas d’utilisation ayant servi à la modélisation des besoins. Le chapitre suivant sera réservé à l’étude conceptuelle de notre application.

# Chapitre 3 : Conception

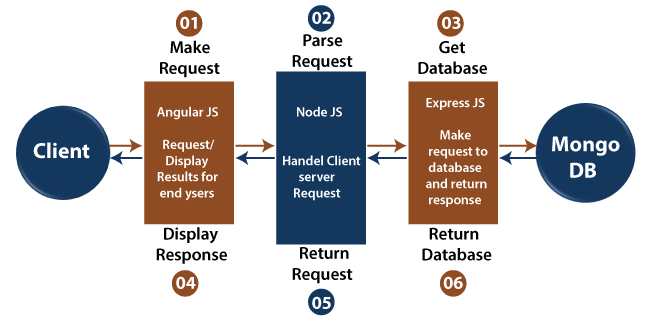
# **Introduction**

La conception est un processus créatif. C’est la phase la plus importante dans le cycle de développement d’une application. Elle permet de décrire d’une manière non ambigüe le fonctionnement d’un système afin de simplifier sa réalisation. C’est pour cette raison nous présentons dans ce chapitre l’architecture utilisée. Par la suite, nous détaillons la conception de l’application en établissant les diagrammes de séquences, le diagramme de classe et le diagramme d’activité.

# **Architecture du système**

# **Architecture trois tiers**

L'architecture trois tiers, aussi appelée architecture à trois niveaux ou architecture à trois couches, est l'application du modèle plus général qu'est le multi-tiers. C'est une architecture basée sur l'environnement client-serveur. [2]



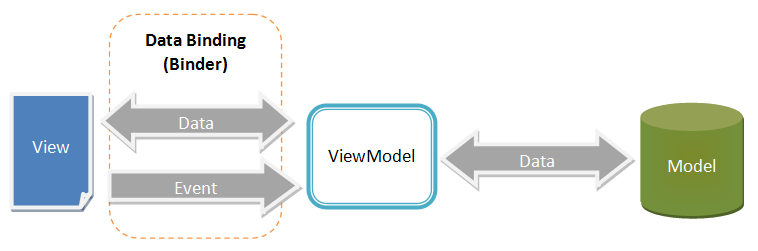
**Figure 10 : L'architecteur 3-tiers**

Comme montrer la figure ci-dessus cette architecture trois tiers, sépare l'application en 3 niveaux de services distincts, conformes au principe précédent :

* AngularJS est un langage côté client écrit en JavaScript. Ainsi, tout d'abord, la demande du client est traitée par lui.
* Après cela, la requête entre dans le serveur (Node.js), c'est-à-dire la phase 2. Node.js est un langage côté serveur écrit en JavaScript.
* Après cela, ExpressJS fait la demande à la base de données, et elle est traitée comme la phase 3.
* Après avoir reçu une demande, MongoDB récupère les données et renvoie la réponse à ExpressJS.
* Une réponse est renvoyée à Node.js depuis Express.js, puis transmise à AngularJS par Node.js pour afficher le résultat.

# **Patron de conception MVVM (Modèle Vue Vue-Modèl) :**

Notre application fait recours à l’architecture MVVM. Le principe de base de ce modèle est tel le modèle MVC (Modèle Vue Contrôleur), de séparer la vue de la logique et de l’accès aux données en accentuant les principes de liaison et d’évènement. Pour plus de clarté, nous détaillons les composants de ce modèle dans ce qui suit :



**Figure 11: le patron MVVM**

* Modèle : Le modèle contient les données. Généralement, ces données provi- ennent d’une base de données ou d’un service externe comme une API.
* Vue : La vue correspond à ce qui est affiché. La vue contient les différents composants graphiques (boutons, liens, listes) ainsi que le texte.
* Vue-Modèle : Ce composant fait le lien entre le modèle et la vue. Il s’occupe de gérer les liaisons de données et les éventuelles conversions. C’est ici qu’intervient le [binding.[5]](#_bookmark178)

# **Conception**

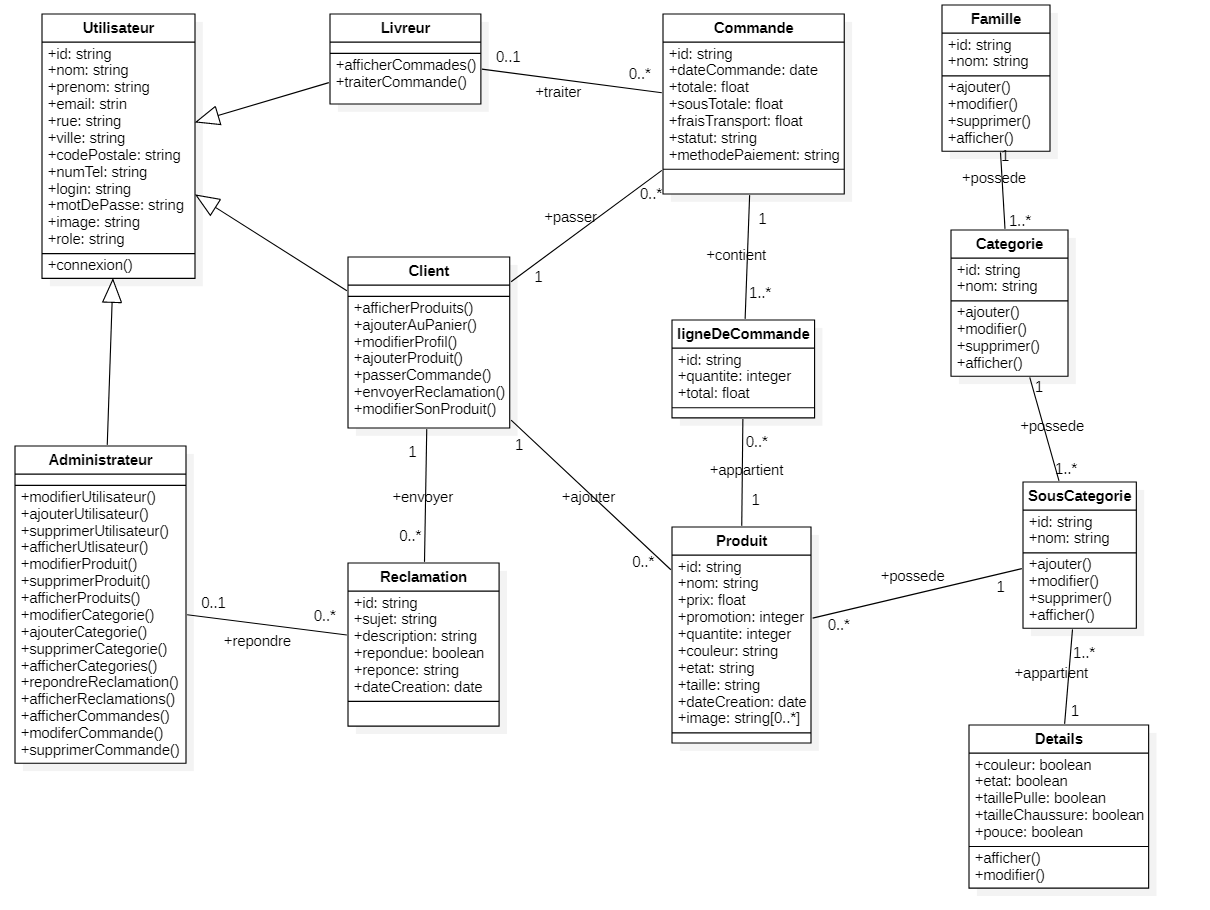
# **Modélisation conceptuelle**

La modélisation conceptuelle est une approche intuitive pour concevoir une base de données. Elle est adaptée dès qu’on doit structurer un ensemble de données un peu complexe (à chacun de juger quand un ensemble de données atteint ce seuil). Une autre approche, la théorie de la normalisation, est plus formelle, plus rigoureuse et offre des moyens d’analyse très fins pour régler des problèmes sur des sous-ensembles précis des données [3].

# **Diagramme de classe**

Les diagrammes de classes sont l'un des types de diagrammes UML les plus utiles, car ils décrivent clairement la structure d’un système particulier en modélisant ses classes, ses attributs, ses opérations et les relations entre ses objets [4].

Le diagramme ci-dessous est le diagramme de classes de mon application :

****

# **Correspondance entre modèles objets et relationnels**

|  |  |  |
| --- | --- | --- |
| **Classe :** Produit | | |
| **Attribut** | **Type** | **Description** |
| id | String | Il représente l’identifiant de produit |
| nom | String | Il représente le nom du produit |
| Description | String | Il représente la description du produit |
| Prix | Float | Il représente le prix du produit |
| Taille | String | Il représente le taille du produit |
| Couleur | String | Il représente le couleur du produit |
| imageUser | String | Il représente l’url d’image d’utilisateur |
| idUser | String | Il représente l’id d’utlisateur |
| date | date | Il représente le date du produit |

|  |  |  |
| --- | --- | --- |
| **Classe :** famille | | |
| **Attribut** | **Type** | **Description** |
| id | String | Il représente l’identifiant de Categorie famille |
| nom | String | Il représente le nom de Categorie famille |

**Tableau 7 : Description relative aux attributs de la classe Famille**

**Tableau 7 : Description relative aux attributs de la classe Produit**

|  |  |  |
| --- | --- | --- |
| **Classe :** Categorie | | |
| **Attribut** | **Type** | **Description** |
| id | String | Il représente l’identifiant de Categorie |
| nom | String | Il représente le nom de Categorie |
| idFamille | String | Il représente l’identifiant de Categorie famille |

**Tableau 8 : Description relative aux attributs de la classe Catégorie**

|  |  |  |
| --- | --- | --- |
| **Classe :** sousCategorie | | |
| **Attribut** | **Type** | **Description** |
| id | String | Il représente l’identifiant de sousCategorie |
| nom | String | Il représente l’identifiant de sousCategorie |
| idCategorie | String | Il représente l’identifiant de categorie |

**Tableau 9 : Description relative aux attributs de la classe sousCategorie**

|  |  |  |
| --- | --- | --- |
| **Classe :** Utilisateur | | |
| **Attribut** | **Type** | **Description** |
| id | String | Il représente l’identifiant d’utilisateur |
| nom | String | Il représente le nom d’utilisateur |
| prenom | String | Il représente le prenom d’utilisateur |
| Image | String | Il représente l’image d’utilisateur |
| Email | String | Il représente l’email d’utilisateur |
| numTel | String | Il représente le numero du telephone d’utilisateur |
| Rue | String | Il représente le rue d’utilisateur |
| Ville | String | Il représente le ville d’utilisateur |
| codePostale | String | Il représente le code postale d’utilisateur |
| idPanier | String | Il représente l’identifiant de panier d’utilisateur |
| Login | String | Il représente le login d’utilisateur |
| password | String | Il représente le mot de passe d’utilisateur |
| role | String | Il représente le role d’utilisateur |

**Tableau 10 : Description relative aux attributs de la classe Utilisateur**

|  |  |  |
| --- | --- | --- |
| **Classe :** Reclamation | | |
| **Attribut** | **Type** | **Description** |
| id | String | Il représente l’identifiant de reclamation |
| sujet | String | Il représente le sujet de reclamation |
| description | String | Il représente la description de reclamation |
| client | String | Il représente le client qui fait le reclamation |
| repondue | Boolean | Il représente l’etat de reclamation |
| reponse | String | Il représente le reponse pour le reclamation |
| dateCreation | date | Il représente le date de creation du reclamation |

**Tableau 11 : Description relative aux attributs de la classe Réclamation**

|  |  |  |
| --- | --- | --- |
| **Classe:** Rating | | |
| **Attribut** | **Type** | **Description** |
| id | String | Il représente l’identifiant de rating |
| idUser | String | Il représente l’identifiant d’utlisateur qui fait le note |
| idProduit | String | Il représente l’identifiant de produit qui |
| value | integer | Il représente le valeur de note |

**Tableau 12 : Description relative aux attributs de la classe Rating**

|  |  |  |
| --- | --- | --- |
| **Classe:** Facture | | |
| **Attribut** | **Type** | **Description** |
| id | String | Il représente l’identifiant de rating |
| nomClient | String | Il représente l’identifiant d’utlisateur qui fait le note |
| panier | String | Il représente l’identifiant de produit qui |
| adresse | String | Il représente le valeur de note |
| ville | String |  |
| codePostale | String |  |
| numTel | String |  |
| dateCommande | String |  |

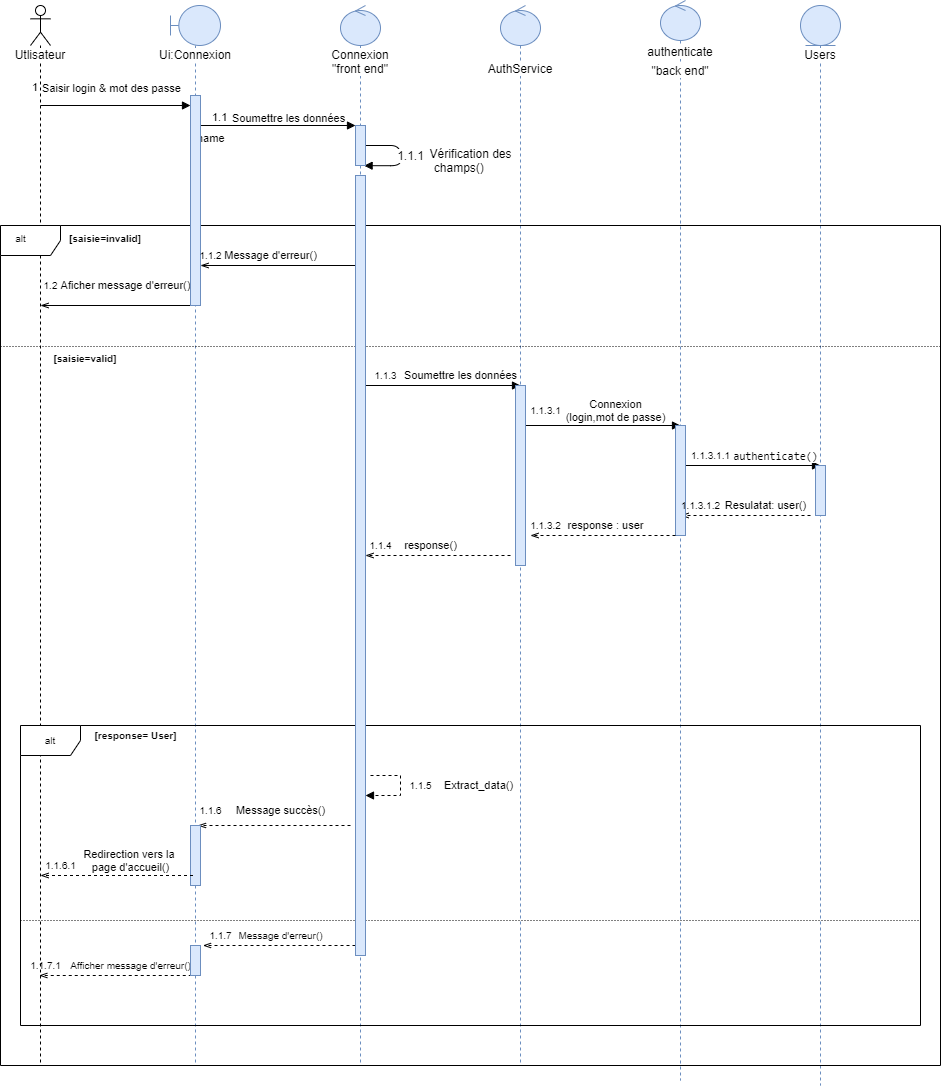
**Tableau 13 : Description relative aux attributs de la classe Facture**

# **Diagramme de séquences**

Le diagramme de séquence représente la succession chronologique des opérations réalisées par un acteur. Il indique les objets que l’acteur va manipuler et les opérations qui font passer d’un objet à l’autre.

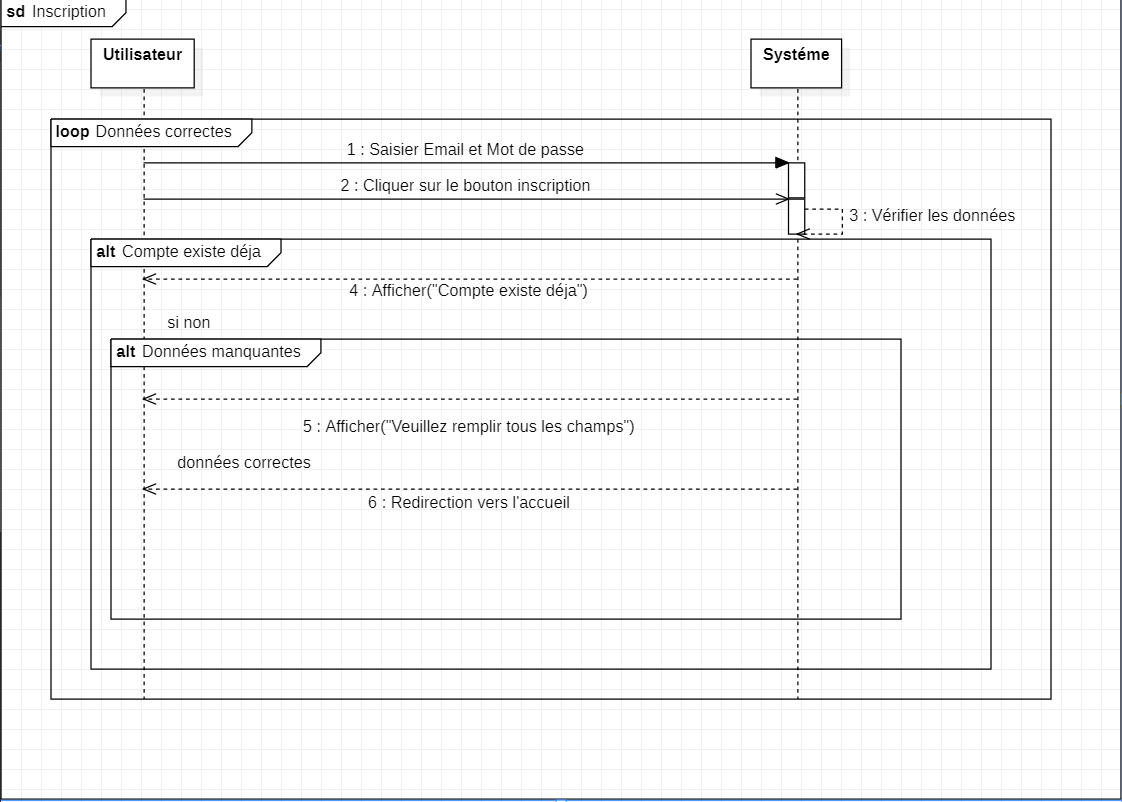
## **Diagramme de séquence « Authentification »**

Ce diagramme décrit le cas de l’authentification dans l’application web, l’utilisateur doit indiquer son login et mot de passe. Une fois vérifié le système charge l’interface d’accueil si non un message d’erreur survenue sera affiché.

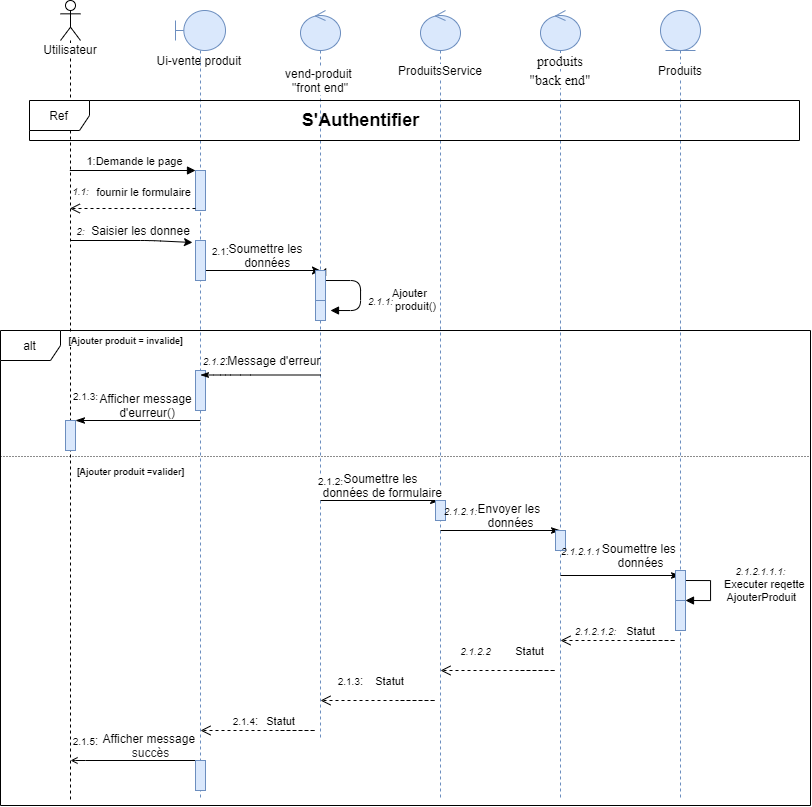


# **Diagramme de séquence « Inscription »**

Ce diagramme décrit le cas de l’inscription dans l’application, l’utilisateur doit saisir toutes les informations nécessaires. Une fois vérifiée le système charge l’interface d’accueil si non un message d’erreur survenue sera affiché.



1. **Diagramme de séquence «Ajouter produit  »**

****

1. **Diagramme de séquence «  »**

# Conclusion

À l’issu de ce chapitre, j’ai mis en place la partie d’analyse et la conception de mon application, en ayant recours aux différents diagrammes d’uml. Ainsi, je peux passer à l’étape finale de mon projet, celle de l’implémentation de l’application, qui sera présentée dans le chapitre qui suit avec la spécification de mon choix technologique.

# **Chapitre 4 : Réalisations**

# **Introduction**

# **Environnement de travail**

# **Environnement matériel**

Pendant le processus de développement, j’ai utilisé mon ordinateur personnel. Ils disposent de configuration suivante :

❖ ASUS ROG GL552VW:

* Intel R Core i7-6700HQ.
* CPU @2.60GHz
* Ram : 16, 00 Go
* Système d’exploitation : Windows 10 professionnel.

# **Environnement logiciel**

|  |  |  |
| --- | --- | --- |
| **Icone** | **Logiciel** | **Description** |
|  | **Visual Studio Code** | Visual Studio Code (VSC par la suite) est un éditeur de code open-source, gratuit et multi-plateforme (Windows, Mac et Linux), développé par Microsoft. Il est livré avec un support intégré pour JavaScript, Type Script et Node.js et possède un riche écosystème d'extensions pour d'autres langages (tels que C ++, C #, Java, Python, PHP, Go) et des runtimes (tels que .NET et Unity). |
|  | **StarUML** | StarUML est un logiciel de modélisation UML, open source gère la plupart des diagrammes spécifiés dans la norme UML. |

# **Choix techniques**

# **Outils de développement**

|  |  |  |
| --- | --- | --- |
|  | **Angular** | Angular est une plate-forme et un cadre pour la création d'applications clientes d'une seule page à l'aide de HTML et de Type Script. Angular est écrit en Type Script. Il implémente les fonctionnalités de base et facultatives en tant qu'ensemble de bibliothèques Type Script que vous importez dans vos applications. |
|  | **Node JS** | Node JS est une plateforme construite sur le moteur JavaScript V8 de Chrome qui permet de développer des applications en utilisant du JavaScript. Il se distingue des autres plateformes grâce à une approche non bloquante permettant d'effectuer des entrées/sorties (I/O) de manière asynchrone. |
|  | **Express** | Express est un framework pour construire des applications web basées sur Node.js. C'est le framework standard pour le développement de serveur en Node.js. |
|  | **HTML 5** | HyperText Markup Language, généralement abrégé HTML, est le format de données conçu pour représenter les pages web. C’est un langage de balisage permettant d’écrire de l’hypertexte, d’où son nom. |
|  | **CSS 3** | Les feuilles de styles (en anglais "Cascading Style Sheets", abrégé CSS) sont un langage qui permet de gérer la présentation d'une page Web. Le langage CSS est une recommandation du World Wide Web Consortium (W3C), au même titre que HTML ou XML. |
|  | **Bootstrap** | Bootstrap est un Framework front-end (HTML5, CSS et JavaScript) spécialement conçu pour le développement d'application web "responsive", c'est-à-dire qui s’adapte automatiquement à différents dispositifs et tailles d'écran (desktop, tablettes, smartphones, ...). |

# **Base de données**

|  |  |  |
| --- | --- | --- |
|  | **MongoDB** | MongoDB est un système de gestion de base de données orienté documents, répartissable sur un nombre quelconque d'ordinateurs et ne nécessitant pas de schéma prédéfini des données. Il est écrit en C++. Le serveur et les outils sont distribués sous licence SSPL, les pilotes sous licence Apache et la documentation sous licence Creative Commons. Il fait partie de la mouvance NoSQL. |

# ***Presentation de l’application***

# Les interfaces de l’application web

## ***Interface d’authentification***

permet aux utilisateurs de s’authentifier à travers le login et mot de passe .

## ***Interface d’inscription***

Pour accéder à l’application, l’utilisateur doit avoir un compte dont il peut le créer facilement il suffit juste de remplir le formulaire ci-dessus.

## ***Interface d’accueil***

Cette page est la page principale du site. Tout le monde peut y accéder. Elle contient tous les produits disponibles ainsi que leurs descriptions, couleurs, prix…et avis.

## ***Interface réclamation***

On peut exprimer ici notre réclamation en mettant le sujet et la description.

## ***Interface modifier profile***

L’utilisateur peut changer ses informations grâce à cette page .

## ***Interface vends un article***

L’utilisateur peut déposer des articles à vendre en mettant les photos correspondantes ainsi que la catégorie et ma description

## ***Interface Panier***

Cette page correspond au panier qui contient tous les articles à acheter ainsi que le prix à payer.

## ***Interface gérer utilisateurs***

L’administrateur peut ajouter /modifier ou bien supprimer des utilisateurs.

## ***Interface gérer catégories***

L’administrateur peut modifier les catégories affichés ou bien d’ajouter de nouvelles catégories.

## ***Interface gérer produits***

L’administrateur peut ajouter , modifier ou bien supprimer des produits

## ***Interface gérer réclamation***

L’administrateur peut répondre aux réclamations ou supprimer celles qui n’ont aucun sens.

## ***Interface de livreur***

Cette interface est destinée au transporteur qui peut choisir quelle commande il veut livrer la première.

# **Conclusion**

Ce chapitre était la phase de planification du projet. Nous avons présenté Les besoins de spécification, l’approche de management du projet avec SCRUM, l’environnement de travail, et la méthodologie de conception.

Le chapitre prochain présentera le premier sprint de notre projet qui est le site officiel.

# Webographie

**[1]** https://www.lucidchart.com/pages/fr/langage-uml (visité le 21/05/2021)

**[2]** https://www.techno-science.net/definition/5266.html (visité le 21/05/2021)

**[3]** http://formations.imt-atlantique.fr/bd\_ihm/fr/bd/modelisation/modelisation/ (visité le 21/05/2021)

**[4]** https://www.lucidchart.com/pages/fr/diagramme-de-classes-uml (visité le 21/05/2021)

**[5]** https://www.dotnetdojo.com/mvvm/ (visité le 21/05/2021)