

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

# Développement d'une application web de gestion commerciale et de facturation

Elaboré par :

Hosni Houssem

Encadré par :

Mr. Zekri Yessine

Effectué à : Globale Payment Gateway

Encadrant : Mr. Ben saber Ahmed

Année universitaire : 2021/2022

# **Dédicaces**

*Je dédie ce travail  
à Tous les membre de ma famille  
pour le très grand soutien moral qu'ils m'ont apporté ,en  
particulier:*

***ma mère***  
*pour avoir su me motiver quand le besoin se faisait  
sentir,*

***mon père***  
*pour ses précieux conseils qui m'ont permis de tenir le rythme ,*

## **À mes meilleurs amis**

*pour leurs soutien constant et  
leurs amour qui est le meilleur des  
encouragements*

*A tous ceux qui, de près ou de loin, ont contribué à la réalisation de  
ce travail*

*J'espère qu'ils trouveront dans ce travail tout ma reconnaissance*

*HOSNI HOUSSEM,*

## Remerciement

Ce projet, réalisé au sein du l'entreprise **globale payment gateway** s'inscrit dans le cadre du projet de Fin d'Études à l'institut supérieur des études technologiques de djerba.

Au terme de ce projet, Nous tenons à remercier dans un premier temps **Mr. Zekri Yassine**, mon encadrant académique pour l'aide et les conseils concernant les missions évoquées dans ce rapport, qu'il nous a apporté lors des différents suivis, pour le temps qu'il nous a consacré tout au long de la période du travail sachant répondre à toutes nos interrogations.

Nous tenons à remercier tout particulièrement et à témoigner toute notre reconnaissance à **Mr. Ben Saber Ahmed**,mon encadrant professionnel, pour la confiance qu'il nous a accordé dès notre arrivé à GPG ainsi que pour l'expérience enrichissante et pleine d'intérêt qu'il nous a fait vivre durant cette période sans oublier sa participation au cheminement de ce rapport. Ainsi que pour l'ensemble du personnel de GPG pour leur accueil sympathique et leur coopération professionnelle tout au long de ces quatre mois. Je tiens finalement à remercier vivement les membres du jury qui ont fait l'honneur de juger ce travail et j'espère qu'il sera à la hauteur de la confiance.

Avec beaucoup d'égard, nous ne manquerons pas d'exprimer notre grande reconnaissance à tous les enseignants et administrateurs de l'ISET et tous les membres de jury pour avoir accepté de juger ce modeste travail.

# Sommaire

Introduction générale .....	1
CHAPITRE I : ÉTUDE PRÉLIMINAIRE.....	2
Introduction .....	3
1.Contexte du projet .....	3
2.Présentation de l'organisme d'accueil .....	3
<b>2.1. Les activités assurées par le GPG .....</b>	<b>4</b>
<b>2.2. Sécurité .....</b>	<b>5</b>
3. Problématique .....	6
4. Étude et critique de l'existant.....	6
5.Solution proposée .....	7
6.Méthodologie et formalise adoptés.....	8
<b>6.1. Scrumssss .....</b>	<b>8</b>
<b>6.1.1. Planification du Projet .....</b>	<b>8</b>
<b>6.1.2. Equipes et Rôles .....</b>	<b>9</b>
Conclusion .....	9
CHAPITRE II : CAHIER DES CHARGES DU LOGICIEL .....	10
Introduction .....	11
1.Specification des besoins .....	11
<b>1.1. Besoins fonctionnels .....</b>	<b>11</b>
<b>1.2. Besoins non fonctionnels .....</b>	<b>12</b>
Conclusion .....	13
CHAPITRE III : Conception .....	14
Introduction .....	15
1.Conception de l'application .....	15
<b>1.1. Le modèle MVC.....</b>	<b>15</b>
<b>1.2. Architecture de l'application web.....</b>	<b>16</b>

2. Diagramme de cas d'utilisation.....	17
<b>2.1. Diagramme de cas d'utilisation .....</b>	<b>18</b>
<b>2.2. Diagrammes de cas d'utilisation détaillés.....</b>	<b>20</b>
<b>2.2.1. Cas d'utilisation « Gérer les Clients» .....</b>	<b>21</b>
<b>2.2.2. Cas d'utilisation «Consulter Produits-Services» .....</b>	<b>22</b>
<b>2.2.3.Cas d'utilisation « Gérer les commandes vente» .....</b>	<b>22</b>
<b>2.2.4.Cas d'utilisation « Gérer facture vente».....</b>	<b>23</b>
<b>2.2.5. Cas d'utilisation «Consulter Tableau de Bord vente» .....</b>	<b>24</b>
<b>2.2.6.Cas d'utilisation « Gérer le compte » .....</b>	<b>24</b>
<b>2.2.7. Cas d'utilisation « Consulter Balance Client » .....</b>	<b>25</b>
<b>2.2.8. Cas d'utilisation « Consulter Balance Fournisseur » .....</b>	<b>25</b>
<b>2.2.9. Diagramme de Cas d'utilisation « Consulter Tableau de Bord » .....</b>	<b>25</b>
<b>2.2.10.Cas d'utilisation «Gérer la Trésorier» .....</b>	<b>26</b>
3. Diagramme de classe .....	26
4.Diagramme des séquences scénario .....	28
<b>4.1. Diagramme de séquence du scénario du Gérer Clients.....</b>	<b>28</b>
<b>4.2. Diagramme de séquence du scénario Gérer Facture vente .....</b>	<b>30</b>
<b>4.3. Diagramme de séquence du scénario nominal effectuer paiement .....</b>	<b>32</b>
<b>4.4. Diagramme de séquence du scénario Gérer trésorerie .....</b>	<b>33</b>
<b>4.5. Diagramme de séquence du scénario nominal Consulter la liste des règlement.....</b>	<b>35</b>
<b>4.6. Diagramme de séquence du scénario S'authentifier.....</b>	<b>35</b>
5.Correspondance entre modèles objets et relationnels :.....	37
Conclusion.....	37
CHAPITRE VI : Etude de la partie reporting.....	38
Introduction .....	39
1.Phase d'ETL .....	39
<b>1.1. Extraction des données .....</b>	<b>39</b>
<b>1.2. Transformation des données.....</b>	<b>42</b>
<b>1.3. Chargement des données.....</b>	<b>42</b>
2.Interface «Rapport Vente».....	43

3.Interface «Rapport Achat» .....	44
4.Interface «Rapport Comptabilité» .....	45
Conclusion .....	45
<b>CHAPITRE V :RÉALISATION .....</b>	<b>46</b>
Introduction .....	47
1.Environnement de travail.....	47
2.Outils de réalisation .....	47
<b>2.1. Visual Studio Code.....</b>	<b>47</b>
<b>2.2. Trello .....</b>	<b>48</b>
<b>2.3. Postman .....</b>	<b>48</b>
<b>2.4. StarUML .....</b>	<b>49</b>
3.Les choix technologiques.....	49
<b>3.1 Laravel .....</b>	<b>49</b>
<b>3.2. Angular .....</b>	<b>49</b>
<b>3.3. PhpMyAdmin .....</b>	<b>50</b>
<b>3.4. Power BI Desktop .....</b>	<b>50</b>
<b>3.5. MySQL Workbench .....</b>	<b>51</b>
4.Présentation des interfaces de l'application.....	52
Conclusion .....	59
Conclusion générale .....	60

## Liste des figures

Figure 1 logo de la société .....	4
Figure 2 PCI DSS .....	5
Figure 3 Vue Globale de Scrum .....	8
Figure 4 Modèle MVC .....	16
Figure 5 Présentation de l'architecture 3-tiers.....	17
Figure 6 Diagramme de cas d'utilisation.....	20
Figure 7 – Diagramme de cas d'utilisation « Gérer les clients ».....	21
Figure 8 – Diagramme de cas d'utilisation « Consulter liste produits-services » .....	22
Figure 9 – Diagramme de cas d'utilisation « Gérer les commandes vente » .....	23
Figure 10 – Diagramme de cas d'utilisation « Gérer facture vente ».....	23
Figure 11 – Diagramme de Cas d'utilisation «Consulter Tableau de Bord vente» .....	24
Figure 12– Diagramme de Cas d'utilisation «Gérer le compte» .....	25
Figure 13 – Diagramme de cas d'utilisation « Consulter balance client » .....	25
Figure 14 – Diagramme de cas d'utilisation « Consulter balance Fournisseur » .....	25
Figure 15 – Diagramme de Cas d'utilisation «Consulter Tableau de Bord».....	26
Figure 16 – Diagramme de cas d'utilisation « Gérer la trésorier » .....	26
Figure 17 diagramme de classe .....	27
Figure 18 – Diagramme de séquence du scénario nominal Gérer client .....	29
Figure 19– Diagramme de séquence du scénario nominal Gérer facture vente .....	31
Figure 20– Diagramme de séquence du scénario nominal effectuer paiement .....	32
Figure 21 – Diagramme de séquence du scénario nominal Gérer trésorerie.....	34
Figure 22 – Diagramme de séquence du scénario nominal Consulter la liste des règlements .....	35
Figure 23 – Diagramme de séquence du scénario nominal S'authentifier.....	36

Figure 24 – ETL .....	39
Figure 25 – interface d'accueil MySQL Workbench.....	40
Figure 26 – les tables des données sources.....	40
Figure 27 – choix de mysql .....	41
Figure 28 – connexion mysql avec power bi .....	41
Figure 29 – Transformation des données avec power bi .....	42
Figure 30 – chargement des données avec power bi .....	43
Figure 31-Visual Studio Code .....	47
Figure 32– Trello .....	48
Figure 33 – Postman .....	48
Figure 34 – StarUML .....	49
Figure 35 –LARAVEL .....	49
Figure 36– Angular.....	50
Figure 37 – phpMyAdmin .....	50
Figure 38– power bi.....	51
Figure 39- MySQL Workbench.....	51
Figure 40 Interface «Tableau de bord vente».....	52
Figure 41 Interface «Clients» .....	52
Figure 42 Interface «Ajouter Client».....	53
Figure 43 Interface «Factures vente» .....	53
Figure 44 Interface «Ajouter factures vente» .....	54
Figure 45 Interface «paiement facture vente» .....	54
Figure 46 Interface «Tableau de bord comptabilité».....	55
Figure 47 Interface «Balance Client» .....	55
Figure 48 Interface «Balance fournisseur».....	56

Figure 49 Interface «Trésorerie» .....	56
Figure 50 Interface «Vue d'ensemble» .....	57
Figure 51 Interface «Ajouter opération» .....	57
Figure 52 Interface «Règlements».....	58
Figure 53 Interface «Gérer Profil» .....	58

## **Liste des Tableaux**

Tableau 1:table de comparaison .....	7
Tableau 2 :Equipe de Scrum.....	9
Tableau 3:Liste des fonctionnalités requises .....	11
Tableau 4:spécification du CU « ajouter client» .....	21
Tableau 5: spécification du CU «consulter liste des produit» .....	22
Tableau 6: spécification du CU «Ajouter facture» .....	23

## **Introduction générale**

Dans Le monde des affaires, les entreprises doivent adapter leurs stratégies pour être compétitifs et pour garantir leurs croissances.

Avec l'apparition des nouvelles technologies, les innovations dans le domaine de développement web sont multiples et évoluent sans cesse et les applications sont devenues de plus en plus populaires puisque ces entreprises sont ameneés à profiter des systèmes qui permettent d'optimiser les processus internes, d'accélérer la croissance et de gagner du temps et de l'argent.

C'est dans ce cadre que s'inscrit mon projet de fin d'études (PFE) au sein de la société « GPG» qui consiste à développer une application Web de gestion commercial et de paiement du facture. C'est un système efficace pour la gestion de vente, achat, stock et de comptabilité avec des tableaux de bords pour avoir un aperçu sur toutes les activités.

Le présent rapport synthétise tout le travail qui a été effectué. Il est élaboré en cinq chapitres. Le premier chapitre est dédié au contexte de projet, la présentation de l'organisme d'accueil et la problématique du projet. Le deuxième chapitre fera l'objet de l'analyse et de l'expression des besoins en présentant les besoins fonctionnels et non fonctionnels. Le troisième présentera la phase la plus importante à savoir la conception de la solution, son architecture générale. Le quatrième est dédié au partie reporting. Le dernier présentera la phase de réalisation du projet ainsi que les outils utilisés.

---

## **CHAPITRE I : ÉTUDE PRÉLIMINAIRE**

---

## Introduction

Le présent chapitre a pour but de définir le contexte général dont le but de situer le projet dans son environnement organisationnel. Il présente, en premier temps, l'organisme d'accueil, puis une description brève du sujet ainsi que la problématique à résoudre, nous allons aussi nous intéresser à l'étude de l'existant, les critiques de chacune des solutions envisageable et enfin, la solution que l'on propose dans le but de remédier à la problématique en question.

### 1.Contexte du projet

Ce stage s'inscrit dans le cadre d'un projet de fin d'études en vue de l'obtention du diplôme, spécialité développement web et multimédia. Ce stage est effectué au sein de la société « GPG » qui est spécialisée dans le domaine de développement des solutions web de paiement en ligne. Le projet consiste à concevoir et à développer une Application web de gestion commerciale et de suivi de comptabilité financière.

### 2.Présentation de l'organisme d'accueil

#### **GPG: Globale Payment Gateway**

Fondée en 2009 GLOBAL PAYMENT GATEWAY est une société anonyme au capital de 1 500.000 TND. Fournisseur majeur des solutions de paiement électronique nationale et internationale, GLOBAL PAYMENT GATEWAY est le premier prestataire de services paiements certifié PCI DSS de visa et MasterCard en Tunisie et au Moyen orient.

Etant donné l'expertise et le savoir-faire en matière de la monétique cette entreprise a pu développer un fort partenariat avec les banques, compagnies aériennes, institutions financières chose qui fait à ce que leurs solutions soient à la fois à la pointe de la technologie, innovantes et flexibles.

Leur enjeu est de traiter et créer des moyens de paiement dans un environnement conforme aux normes internationales et de simplifier leurs utilisations.

**Adresse :** No.59 – Pole technologique El-ghazela, Tunis

**TEL :** +216 71 85 72 85

**FAX :** +216 71 85 63 38

**Email :** [contact@gpgateway.com](mailto:contact@gpgateway.com)



Figure 1 logo de la société

## 2.1. Les activités assurées par le GPG

- **Assistance générale**

- |                                  |   |
|----------------------------------|---|
| ✓ Intégration du Kit de paiement | ✓ Personnalisation de l'url de paiement |
| ✓ Conseil                        | ✓ Suivi                                 |

- **Environnements stables et sécurisés**

- |                         |                               |
|-------------------------|-------------------------------|
| ✓ Environnement de test | ✓ Environnement de production |
|-------------------------|-------------------------------|

- **Moyens de paiement**

- ✓ CB
- ✓ VISA
- ✓ Mastercard
- ✓ Carte e-DINAR UNIVERSEL
- ✓ Carte DINARPOST VISA Electron
- ✓ Carte e-DINAR SMART
- ✓ Porte-monnaie électronique GP Pay

- **Dashboard complet**

- ✓ Liste des transactions
- ✓ Comptabilité
- ✓ Statistiques
- ✓ Graphique
- ✓ Reporting

## 2.2. Sécurité



Figure 2 PCI DSS

Global Payment Gateway est certifiée PCI DSS (Payment Card Industry Data Security Standard) pour garantir la sécurisation et la confidentialité des données des paiements en ligne.

PCI DSS est un standard de sécurité international qui garantit la sécurité des données sensibles des porteurs de carte.

La certification PCI DSS est accordée par un organisme indépendant qui effectue régulièrement des audits sur 259 points de contrôles sécuritaires.

En choisissant les services de paiements en ligne de Global Payment Gateway, vous bénéficiez des solutions de paiements sécurisés, parfaitement conformes aux règles PCI-DSS

de Visa MasterCard.

### 3. Problématique

Pour gérer et faire le suivi des activités de l'entreprise (ventes, achats et comptabilités), les responsables utilisent chacun des applications informatiques indépendants, Ceci génère:

- Une difficulté d'échange de données entre les différents responsables.
- Une incohérence de l'information interne.
- Le temps mis pour atteindre l'information est très long.
- Difficulté de connaître les origines des problèmes.
- Absence de suivi exacte de l'avancement de production.
- Chaque responsable se retrouve obligé d'effectuer lui-même un bon nombre des tâches.

Notre entreprise cliente a besoin d'un système efficace contenant des espaces spécifiés pour chaque responsable afin d'organiser les tâches et de faciliter la suivie des activités commerciale et comptable.

### 4. Étude et critique de l'existant

La possession d'un système de gestion commercial pour une société qui est fournisseur de dizaines d'enseignes ou même des entreprises qui souhaitent optimiser leur processus, devient de plus en plus essentielle. C'est ici qu'intervient l'existant qui permet une compréhension approfondie du domaine d'études et d'élaborer un bilan critique de celui-ci. En d'autres termes, il permet de dégager les insuffisances et proposer des solutions dont le but est de les résoudre. Parmi ces solutions, nous pouvons citer 4 exemples qui permettent d'indiquer nos besoins et nos choix techniques :

- **Facture Tunisie** : est un logiciel commercial et un site dédié à la gestion en ligne de la facturation et la gestion des stocks et des inventaires.
- **BIROU** : c'est un logiciel de facturation Tunisien pour la gestion de la facturation des artisans.
- **BISON** : c'est un logiciel de facturation Tunisien pour la gestion de la facturation des artisans, commerçants et PME.

- **SWIVER** : C'est un Logiciel de facturation et gestion commerciale en ligne pour les TPE et PME en Tunisie

Le tableau ci-dessous effectue une comparaison en termes de fonctionnalités entre ces quatre solutions :

Tableau 1:table de comparaison

	Ergonomie	critère portail	Rapidité d'execusion	Reporting
<b>Facture Tunisie</b>			✓	
<b>BIROU</b>	✓		✓	✓
<b>BISON</b>	✓		✓	
<b>SWIVER</b>	✓		✓	✓
<b>Notre solution</b>	✓	✓	✓	✓

## 5.Solution proposée

Pour répondre aux problèmes énoncés, la solution consiste à réaliser une plateforme de suivi financière unifiée, qui permet à notre client de gérer ses activités d'achat, vente ainsi que de mettre en place un suivi de l'état financière et la comptabilité de l'entreprise grâce au "reporting" et à la visibilité en temps réel. On remarque qu'aucune des solutions proposées n'implémente aucunement les techniques BI afin de faciliter et d'améliorer l'usage de l'utilisateur. Pour faire face à ces limites, nous proposons la conception et l'implémentation d'un site web qui englobe les fonctionnalités requises.

Notre application offre les fonctionnalités principales suivantes :

- **Un plateforme de critère Portail:**

permet à nos acteurs venant de trois différents départements d'effectuer le suivi, chacun à travers son propre espace (espace Responsable de vente, d'achat, de comptabilité financière).

- **Un flux de reporting :**

permet de consulter et de suivre les différentes activités ainsi d'avoir un aperçu général sur les activités et la santé financière de l'entreprise.

## 6.Méthodologie et formalise adoptés

### 6.1. 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 boîte de temps, nommés sprints.

La durée des sprints est comprise entre deux semaines et un mois.

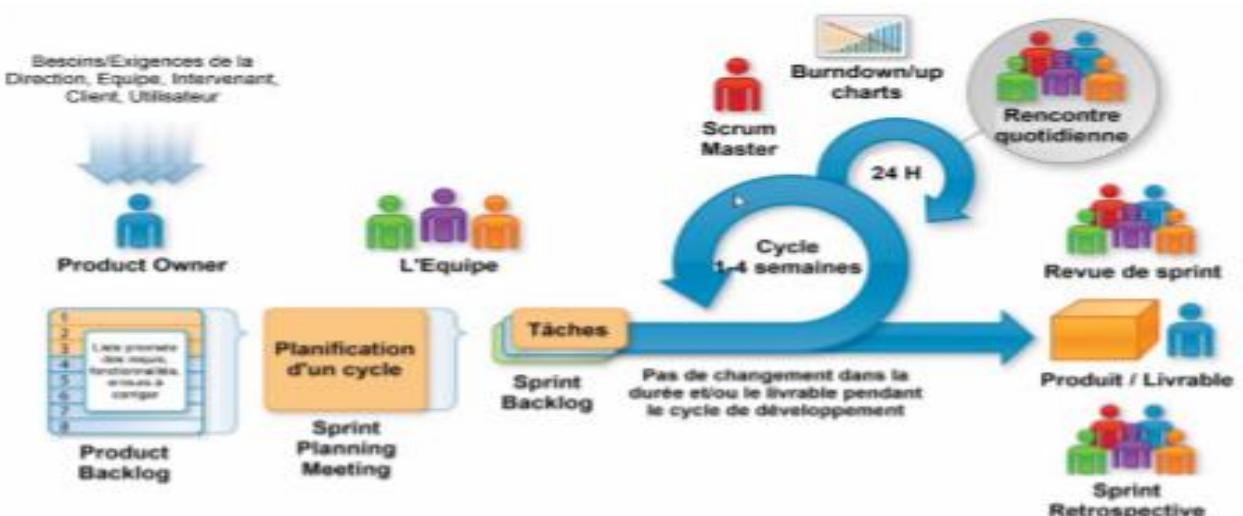


Figure 3 Vue Globale de Scrum

#### 6.1.1. 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.

### 6.1.2. Equipes et Rôles

La méthode SCRUM fait intervenir trois rôles principaux qui sont :

- **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.

Tableau 2 :Equipe de Scrum

Acteurs	Tâches	Rôles
• <b>Zekri Yessine</b>	<ul style="list-style-type: none"><li>• Responsable des conseils</li><li>• Responsable de l'approbation des tâches</li></ul>	<b>Scrum master</b>
• <b>Ahmed ben saber</b>	<ul style="list-style-type: none"><li>• Tests et validation</li><li>• Déploiement</li><li>• Conception</li></ul>	<b>Product Owner</b>
• <b>Hosni Houssem</b>	<ul style="list-style-type: none"><li>• Développement de l'application</li></ul>	<b>Développeur</b>

## Conclusion

Tout au long de ce chapitre, l'organisme d'accueil ainsi que ses services ont été présentés. Par la suite le contexte général du projet ainsi que la problématique a été détaillé. Pour finir, l'étude et la critique de l'existant ont été déterminés, ce qui a permis de cadrer l'objectif du projet et d'identifier les fonctionnalités à implémenter.

Pour bien comprendre le projet on va évoquer la phase de spécification des besoins du notre projet dans le chapitre suivant.

---

## **CHAPITRE II : CAHIER DES CHARGES DU LOGICIEL**

---

## Introduction

Dans ce chapitre, on va présenter les besoins des utilisateurs à travers une spécification des besoins fonctionnels et non-fonctionnels afin d'aboutir à une application performante et satisfaisante.

### 1. Spécification des besoins

L'application doit satisfaire les exigences de la totalité des utilisateurs. Nous exposerons ci-dessous les besoins fonctionnels et non fonctionnels de la solution.

#### 1.1. Besoins fonctionnels

Afin de mener à bien le projet, il est impératif de faire une analyse préliminaire des besoins ainsi qu'une liste des différentes fonctionnalités à mettre en œuvre. Les fonctionnalités requises sont présentées dans le tableau ci-dessous :

Tableau 3: Liste des fonctionnalités requises

Objectif	Fonctionnalité
Partie vente	<ul style="list-style-type: none"><li>— Consulter Tableau de Bord vente.</li><li>— Gérer client.</li><li>— Consulter liste des produits.</li><li>— Gérer commande client.</li><li>— Gérer facture.</li><li>— Se connecter ou modifier ses données.</li></ul>

Partie achat	<ul style="list-style-type: none"> <li>— Consulter Tableau de Bord achat.</li> <li>— Gérer fournisseurs.</li> <li>— Gérer produits.</li> <li>— Gérer commande achat.</li> <li>— Gérer facture achat.</li> <li>— Consulter le stock.</li> <li>— Se connecter ou modifier ses données.</li> </ul>
Partie comptabilité	<ul style="list-style-type: none"> <li>.</li> <li>— Consulter Tableau de Bord comptabilité.</li> <li>— Consulter la balance client.</li> <li>— Consulter la balance fournisseur.</li> <li>— Gérer trésorerie.</li> <li>— Se connecter ou modifier ses données.</li> </ul>

## 1.2. Besoins non fonctionnels

À part les besoins fonctionnels, notre système doit répondre aux critères suivants :

**Besoins de qualité** : Pour attirer un client sur une plateforme web et ensuite le fidéliser, il est important de répondre aux exigences de qualité suivantes :

- **Ergonomie** : le traitement d'une tâche sur la plateforme ne doit pas prendre beaucoup de temps. La mise en page des interfaces facilitera au maximum la

démarche à l'aide d'une présentation claire et intuitive.

- **Formulaire de facture simple :** La conception et la présentation de celui-ci seront particulièrement soignés pour ne pas rebouter le client.
- **Sécurité :** lors de la création d'un compte pour éviter que l'application ne soit victime des robots et des hackers, j'ai utilisé les « JSON Web Token » ou JWT qui 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 ou j'ai utilisé Le Lazy-Landing de modules qui est une fonctionnalité d'Angular permettant de charger du code applicatif lorsque celui-ci sera sollicité, plutôt que de charger toute l'application dès son démarrage.
- **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.

## Conclusion

Dans ce chapitre, j'ai présenté les besoins des utilisateurs à travers une spécification des besoins fonctionnels et non-fonctionnels. Le chapitre suivant sera réservé à l'étude conceptuelle de notre application.

---

## **CHAPITRE III : Conception**

---

## Introduction

La conception est une étape primordiale dans le cycle de vie d'un logiciel. Elle permet de décrire les solutions adaptées pour résoudre les problématiques d'une application ou d'un système. Dans ce chapitre, on présentera tout d'abord la conception générale du projet en détaillant son architecture n-tiers et sur le modèle MVC.

### 1. Conception de l'application

Afin d'avoir une visibilité parfaite sur l'architecture de l'application et pour une décomposition permettant une distinction des sous-modules pour des raisons de maintenance et de réutilisabilité, on va s'intéresser à l'architecture de la plateforme et ses différentes composantes.

#### 1.1. Le modèle MVC

Avant de se lancer dans le développement de notre application web, il est indispensable de choisir un motif de conception (pattern design). Présenté souvent par le Modèle-Vue-Contrôleur (MVC), ce modèle est maintenant très répandu et il est accepté sans contestation, notamment pour la réalisation des applications Web dynamiques, à une organisation satisfaisante le but recherché d'une organisation rigoureuse et logique du code. Le patron de conception Modèle-Vue-Contrôleur (MVC) est un patron architectural qui sépare les données (le modèle), l'interface homme-machine (la vue) et la logique de contrôle (le contrôleur), il est très populaire pour les applications web. Ce modèle de conception impose donc une séparation en 3 couches :

**Le modèle** : Il représente les données de l'application. Il définit aussi l'interaction avec la base de données et le traitement de ces données.

**La vue** : Elle représente l'interface utilisateur. Elle n'effectue aucun traitement, elle se contente simplement d'afficher les données que lui fournit le modèle. Il peut tout à fait y avoir plusieurs vues qui présentent les données d'un même modèle.

**Le contrôleur** : Il gère l'interface entre le modèle et le client. Il va interpréter la requête de ce dernier pour lui envoyer la vue correspondante. Il effectue la synchronisation entre le modèle et les vues.

La synchronisation entre la vue et le modèle se passe avec le pattern Observer. Il permet de générer des événements lors d'une modification du modèle et d'indiquer à la vue qu'il faut se mettre à jour. La figure qui suit présente le schéma d'interaction entre les composants de ce modèle.

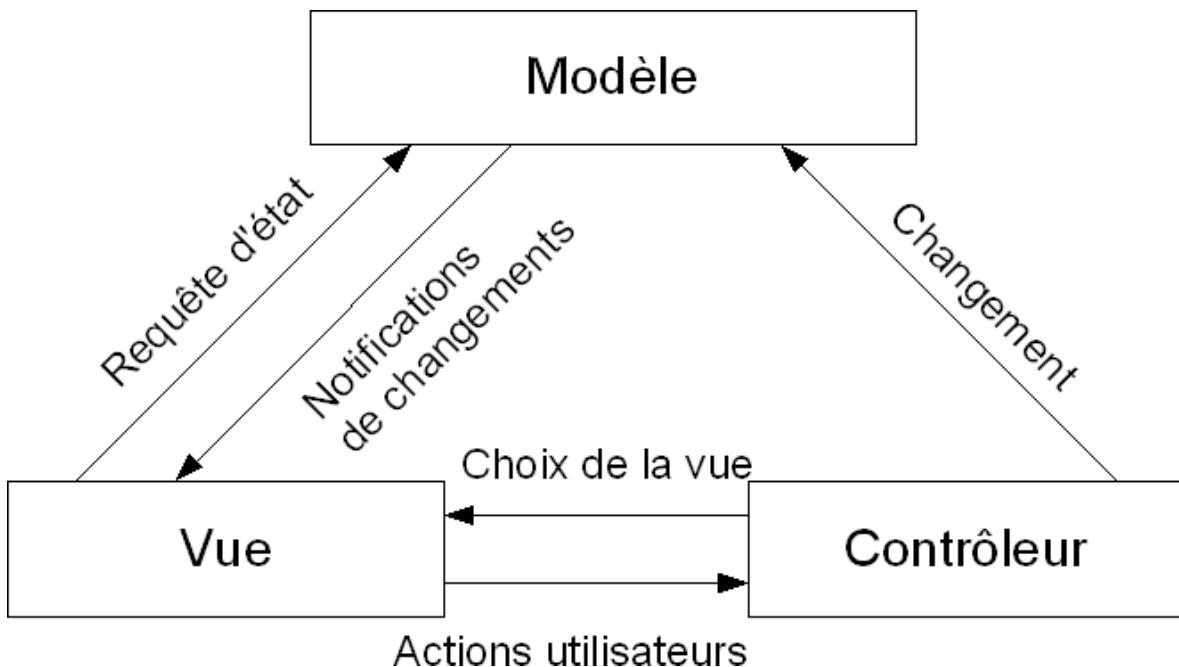


Figure 4 Modèle MVC

Avantages du modèle MVC : Le modèle MVC impose donc une séparation totale entre le traitement, l'interface et la communication entre ces deux parties. Cela permet d'avoir non seulement des objets réutilisables pour d'autres applications, mais aussi de pouvoir faire évoluer aisément son programme et faire alléger le poste de travail. Ainsi, si on souhaite modifier sa base de données il suffit de revoir son modèle et cela est valable pour le cas où on souhaite changer d'interface. Les 3 parties du modèle MVC sont réellement autonomes. Aucune d'elles n'occupe du fonctionnement de l'autre.

## 1.2. Architecture de l'application web

L'architecture n-tiers a été fait pour pallier les limitations des architectures simples telle que l'architecture 2-tiers, et concevoir des applications puissantes et simple à maintenir. Ce type d'architecture permet de distribuer plus librement la logique applicative, ce qui facilite

la répartition de la charge entre tous les niveaux. Cette évolution des architectures qualifie la distribution d'application entre multiples services et non la multiplication des niveaux de service. Cette distribution est facilitée par l'utilisation de composants « métier », spécialisés et indépendants, introduits par les concepts orientés objet (langages de programmation et middleware). Elle permet de tirer pleinement partie de la notion de composants métiers réutilisables.

Ce style d'architecture facilite la réutilisabilité au sein d'un système puisque les composants métiers correspondent à des concepts communs dans le cas présent. Dans ce contexte, nous avons divisé notre application en 3 couches :

**La présentation des données** : contient les différents types de clients.

**Le traitement métier des données** : correspondant à la logique applicative.

**L'accès aux données persistantes** : correspondant à la couche d'accès et de stockage des données.

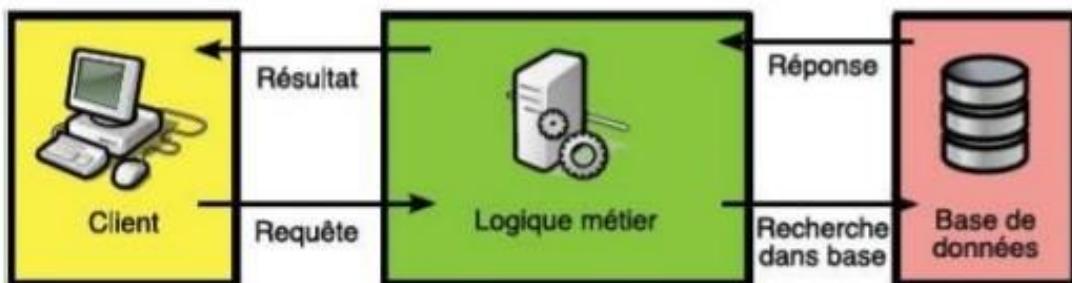


Figure 5 Présentation de l'architecture 3-tiers

## 2. Diagramme de cas d'utilisation

Le diagramme de cas d'utilisation est utilisés pour donner une vision globale du comportement fonctionnel d'un système. Un cas d'utilisation représente une unité discrète d'interaction entre un utilisateur (humain ou machine) et un système. C'est une unité significative de travail. Dans un diagramme de cas d'utilisation, les utilisateurs sont appelés acteurs, ils interagissent avec les cas d'utilisations (use cases).

## 2.1. Diagramme de cas d'utilisation

La figure donne une vue globale concernant le comportement fonctionnel du système. Ce diagramme permet de représenter les interactions entre les acteurs et les cas d'utilisation du système.

Les différents cas d'utilisation identifiés sont :

- **Consulter dashboard vente** : le responsable vente peut avoir un aperçu général sur les activités de vente de l'entreprise grâce aux statistiques contenues dans le tableau de bord.
- **Consulter la liste des produits-services** : le responsable vente peut consulter la liste et les détails des produits.
- **Gérer client** : le responsable vente peut faire la gestion des clients de l'entreprise.
- **Gérer commande client** : le responsable vente peut faire la gestion des commandes des clients.
- **Gérer facture vente** : le responsable vente peut faire la gestion des facture des ventes.
- **Consulter dashboard achat** : le responsable achat peut avoir un aperçu général sur les activités d'achat de l'entreprise grâce aux statistiques contenues dans le tableau de bord.
- **Gérer les fournisseurs** : le responsable achat peut faire la gestion des fournisseurs de l'entreprise.
- **Gérer les produits-services** : le responsable achat peut faire la gestion des produits-services.
- **Gérer commande achat** : le responsable achat peut faire la gestion des commandes achat.
- **Gérer facture achat** : le responsable achat peut faire la gestion des facture achat.
- **Consulter le stock** : le responsable achat peut consulter la liste des produits de stock ainsi que les détails et la quantité de chaque produit.
- **Consulter dashboard comptabilité** : le responsable comptabilité financière peut avoir un aperçu général sur la santé de l'entreprise grâce aux statistiques

contenues dans le tableau de bord.

- **Consulter balance client** :le responsable comptabilité financière peut consulter le cumul debit et credit pour avoir un aperçu sur le balance client.
- **Consulter balance fournisseur** :le responsable comptabilité financière peut consulter le cumul debit et credit pour avoir un aperçu sur le balance fournisseur.
- **s'authentifier** : Tout les acteurs doivent s'authentifier.
- **Gérer le profil** :Tout les acteurs peuvent modifier leurs mot de passe et/ou leurs nom d'utilisateur.
- **Gérer trésorie**:le responsable comptabilité financière peut ajouter une opération, un dépense et consulter la liste des règlements.

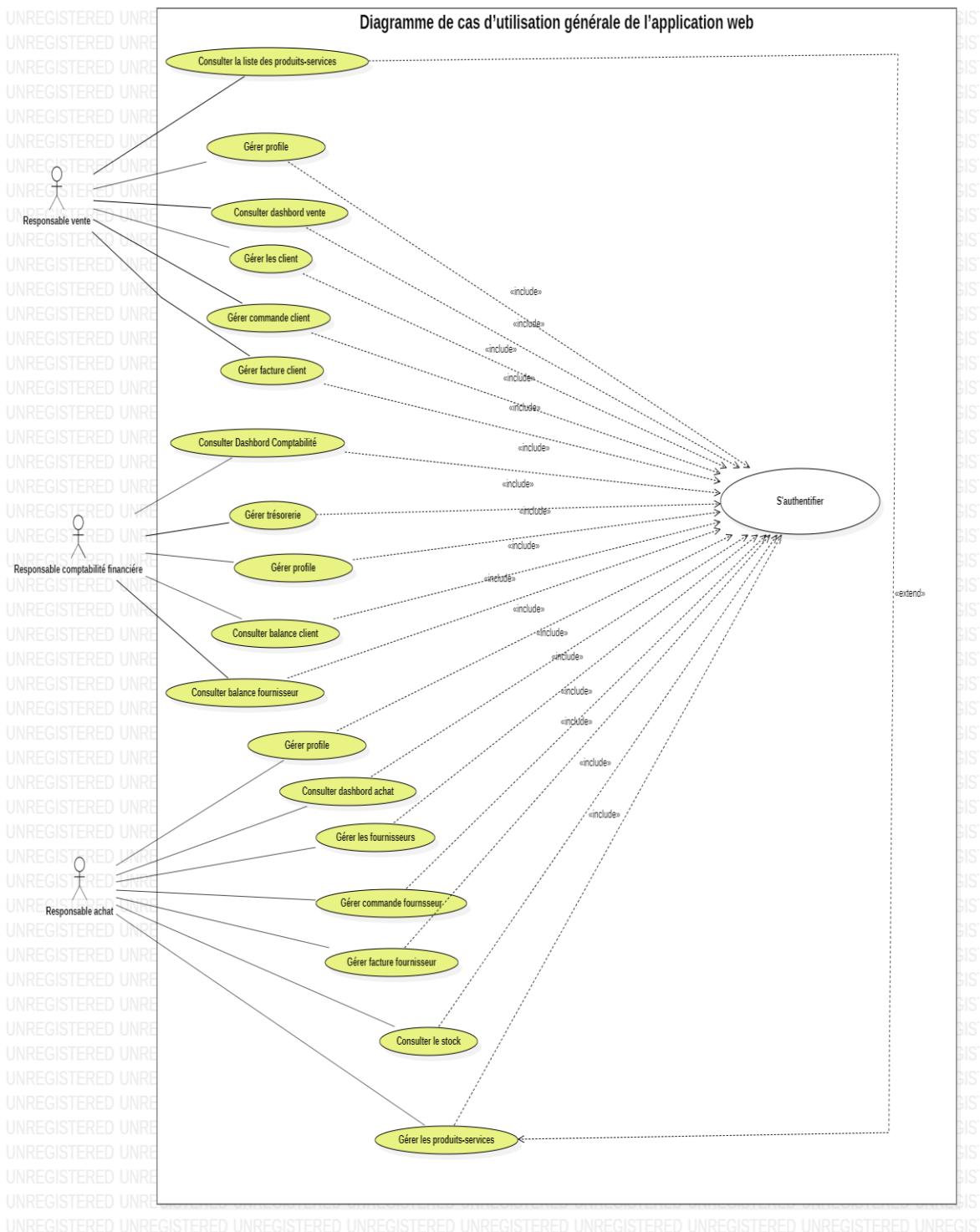


Figure 6 Diagramme de cas d'utilisation

## 2.2. Diagrammes de cas d'utilisation détaillés

Dans cette partie, nous allons présenter les diagrammes des cas d'utilisation qui détaillent les fonctionnalités qui sont offertes au responsable vente et au responsable comptabilité.

### 2.2.1. Cas d'utilisation « Gérer les Clients»

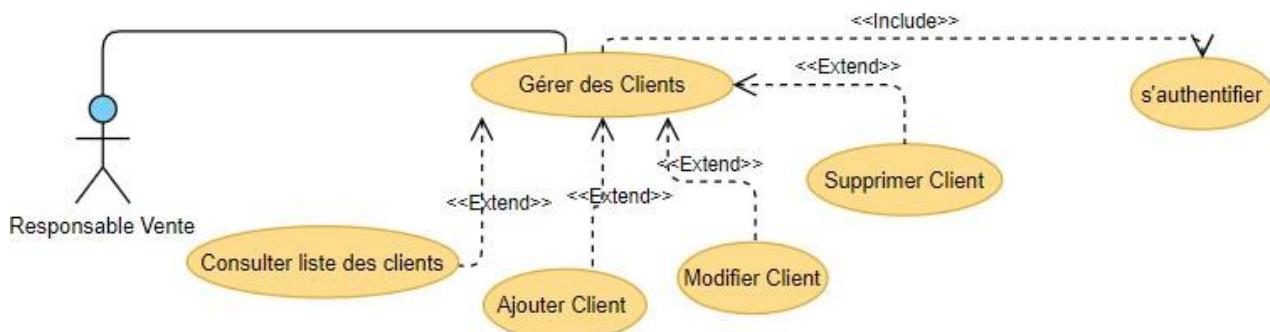


Figure 7 – Diagramme de cas d'utilisation « Gérer les clients »

Tableau 4:spécification du CU « ajouter client»

Nom de cas d'utilisation :	Ajouter client
Acteur :	— Responsable vente.
Pré-condition :	— Une authentification préalable de responsable vente. — Opération d'ajout.
Post-condition :	— client ajouté.
Scénario nominal :	<ul style="list-style-type: none"> <li>— 1. Le responsable demande d'ajouter un client.</li> <li>— 2. Le système affiche l'interfaces d'ajout.</li> <li>— 3. Le responsable remplit les champs.</li> <li>— 4. Le système vérifie les champs remplis.</li> <li>— 5. Le système informe le responsable du succès de l'opération et enregistre le client.</li> </ul>
Scénario alternatif :	<ul style="list-style-type: none"> <li>— 4. Le système affiche un message d'erreur d'ajout.</li> </ul>

### 2.2.2. Cas d'utilisation «Consulter Produits-Services»



Figure 8 – Diagramme de cas d'utilisation « Consulter liste produits-services »

Tableau 5: spécification du CU «consulter liste des produit»

Nom de cas d'utilisation :	Consulter liste des produits
Acteur :	— Responsable vente.
Pré-condition :	— Une authentification préalable de responsable vente.
Post-condition :	— Liste des produits affichée.
Scénario nominal :	<ul style="list-style-type: none"> <li>— 1. Le responsable accéder à l'espace produits.</li> <li>— 2. Le système affiche la liste des produits.</li> </ul>
Scénario alternatif :	— 1 . Aucun produit trouvé.

### 2.2.3.Cas d'utilisation « Gérer les commandes vente»

Le cas d'utilisation «Gérer les commandes vente» permet au responsable vente de créer une commande pour le client, de consulter la liste des commandes et de l'extraire en PDF.

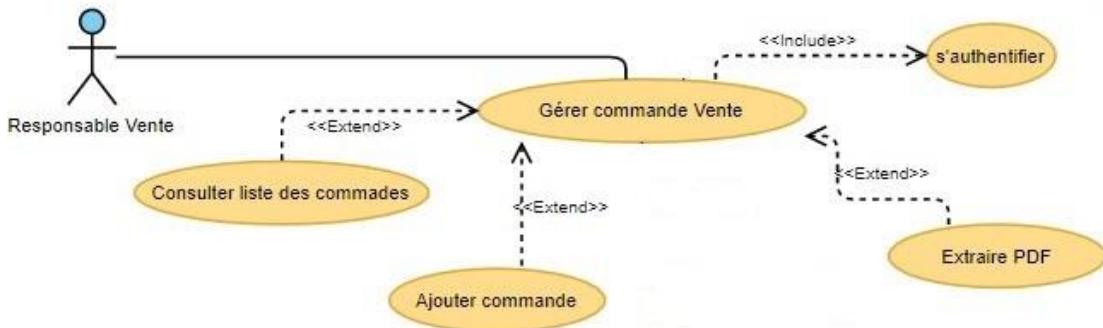


Figure 9 – Diagramme de cas d'utilisation « Gérer les commandes vente »

#### 2.2.4.Cas d'utilisation « Gérer facture vente»

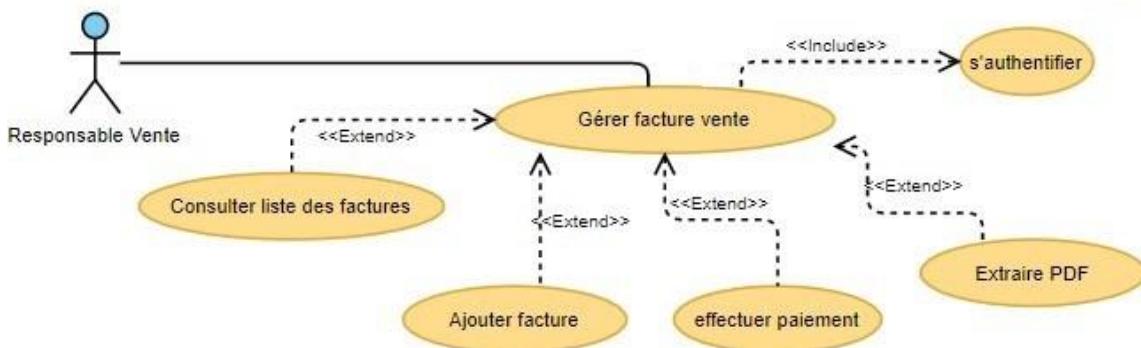


Figure 10 – Diagramme de cas d'utilisation « Gérer facture vente »

Tableau 6: spécification du CU «Ajouter facture»

Nom de cas d'utilisation :	Ajouter facture
Acteur :	— Responsable vente.
Pré-condition :	— Une authentification préalable de responsable vente. — Opération d'ajout facture.

Post-condition :	<ul style="list-style-type: none"> <li>— Facture ajoutée.</li> </ul>
Scénario nominal :	<ul style="list-style-type: none"> <li>— 1. Le responsable demande d'ajouter une facture.</li> <li>— 2. Le système affiche les interfaces de d'ajout.</li> <li>— 3. Le responsable selectionne un client.</li> <li>— 4. Le système affiche les détails du client.</li> <li>— 5. Le responsable selectionne la liste des produits.</li> <li>— 6. Le système affiche les Totals des prix.</li> <li>— 7. Le système vérifie les champs remplis.</li> <li>— 8. Le système informe le responsable de succès de l'opération et enregistre la facture.</li> </ul>
Scénario alternatif :	<ul style="list-style-type: none"> <li>— 3 , 5 , 7 Champ obligatoire vide ou aucun produit ou client.</li> </ul>

### 2.2.5. Cas d'utilisation «Consulter Tableau de Bord vente»

Le responsable vente peut suivre les activités de vente de l'entreprise grâce aux statistiques contenues dans le tableau de bord.

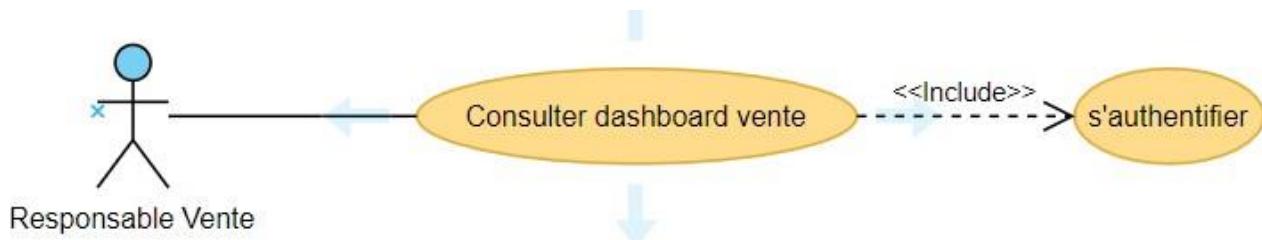


Figure 11 – Diagramme de Cas d'utilisation «Consulter Tableau de Bord vente»

### 2.2.6. Cas d'utilisation « Gérer le compte »

Le responsable achat peut accéder à son profil et modifier ses données (son nom

d'utilisateur ou son mot de passe), cette cas d'utilisation est pareil pour les autres responsables.

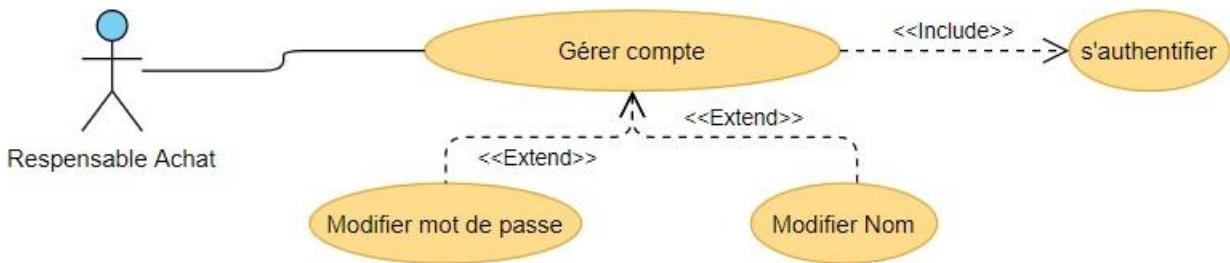


Figure 12– Diagramme de Cas d'utilisation «Gérer le compte»

### 2.2.7. Cas d'utilisation « Consulter Balance Client »

Le responsable de la comptabilité financière peut suivre l'accumulation des crédits et des débits pour avoir une vue d'ensemble du solde de chaque client de l'entreprise.



Figure 13 – Diagramme de cas d'utilisation « Consulter balance client »

### 2.2.8. Cas d'utilisation « Consulter Balance Fournisseur »

Le responsable de la comptabilité financière peut suivre l'accumulation des crédits et des débits pour avoir une vue d'ensemble du solde de chaque fournisseur de l'entreprise.



Figure 14 – Diagramme de cas d'utilisation « Consulter balance Fournisseur »

### 2.2.9. Diagramme de Cas d'utilisation « Consulter Tableau de Bord »

Depuis le tableau de bord comptable, le responsable de la comptabilité financière peut consulter des statistiques et des données pour avoir un aperçu de la trésorerie et de la santé de l'entreprise en termes de ventes et d'achats



Figure 15 – Diagramme de Cas d'utilisation «Consulter Tableau de Bord»

### 2.2.10.Cas d'utilisation «Gérer la Trésorier»

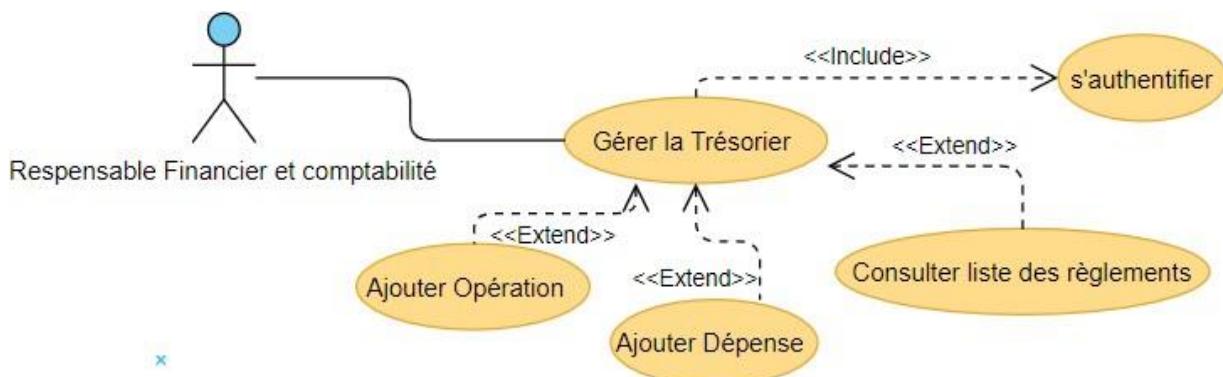


Figure 16 – Diagramme de cas d'utilisation « Gérer la trésorier »

## 3. Diagramme de classe

Le diagramme de classe est une représentation statique des éléments du système et de leurs relations. Chaque application qui va mettre en œuvre le système sera une instance des différentes classes qui le compose. À ce titre il faudra bien garder à l'esprit qu'une classe est un modèle et l'objet de sa réalisation . Dans ce qui suit, on présente les principales classes et les relations entre elles.

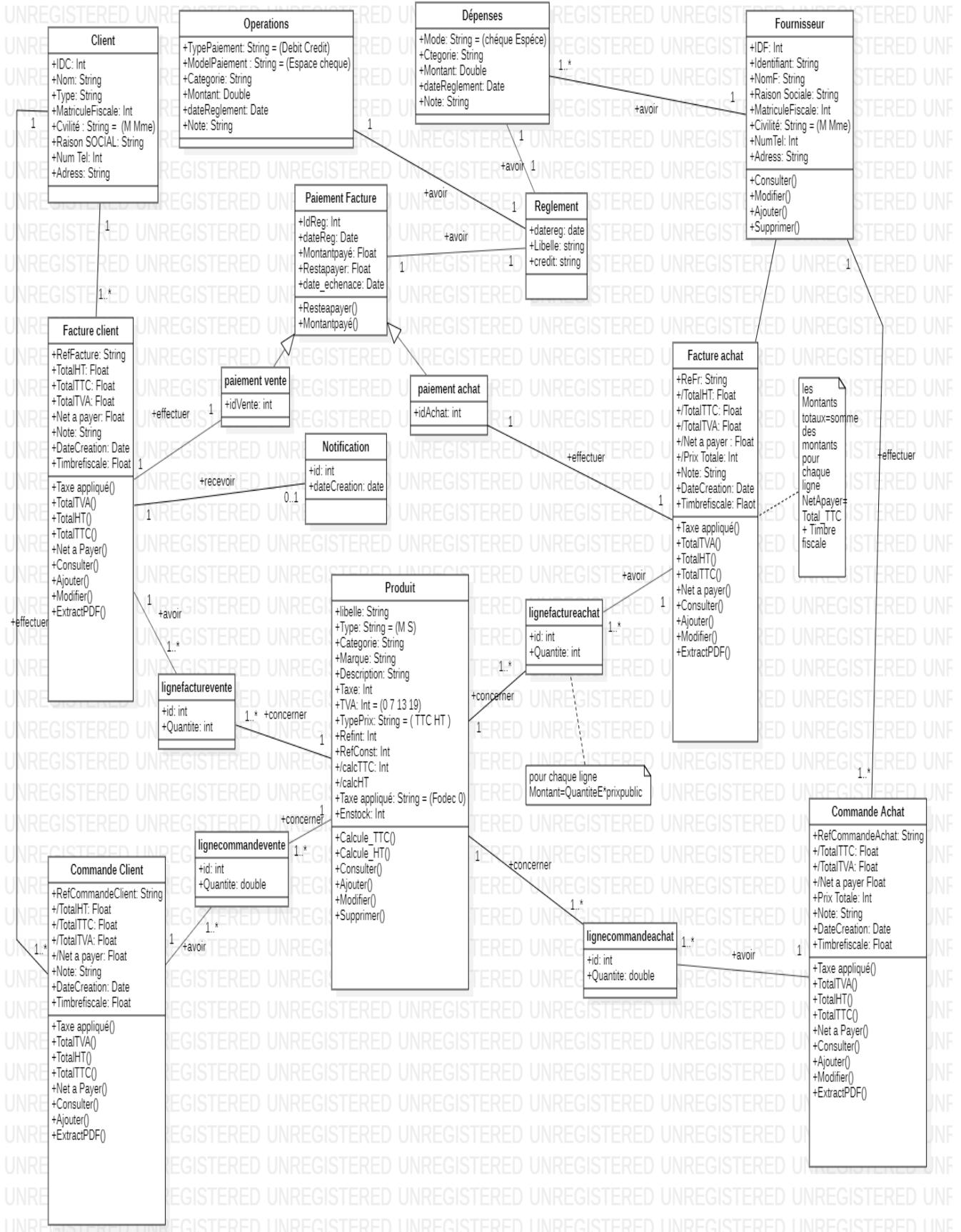


Figure 17 diagramme de classe

## 4. Diagramme des séquences scénario

Les diagrammes de séquences (DS) permettent de décrire en détail la façon avec laquelle les éléments du système interagissent entre eux en précisant les messages échangés suivant un ordre chronologique. On va donc utiliser ce type de diagramme pour décrire chaque scénario du cas d'utilisation

### 4.1. Diagramme de séquence du scénario du Gérer Clients

Lors de la création d'un Client, l'acteur est invité à s'authentifier et puis à saisir les informations requises. Le système vérifie ensuite si les champs sont bien remplis. Si c'est le cas, le bouton « Créer » devient fonctionnel et en appuyant dessus, le « Controller client » reçoit une requête HTTP de type « POST » contenant dans son corps les informations du produit à créer , ce dernier ajoute le client dans la base de données, crée une nouvelle ligne de client et informe l'acteur de la réussite de l'opération avec une requête HTTP OK , ainsi le système affichera tous les fournisseurs pour la mise à jour d'un client , après la sélection d'un dernier à modifier, le « Controller client » reçoit une requête HTTP de type « GET-ById » contenant dans son corps les informations de la client, après le client sélectionné sera affiché,l'acteur fait les modifications souhaitées, ensuite une requête HTTPUPDATE est envoyée pour l'enregistrement d'une nouvelle modification.

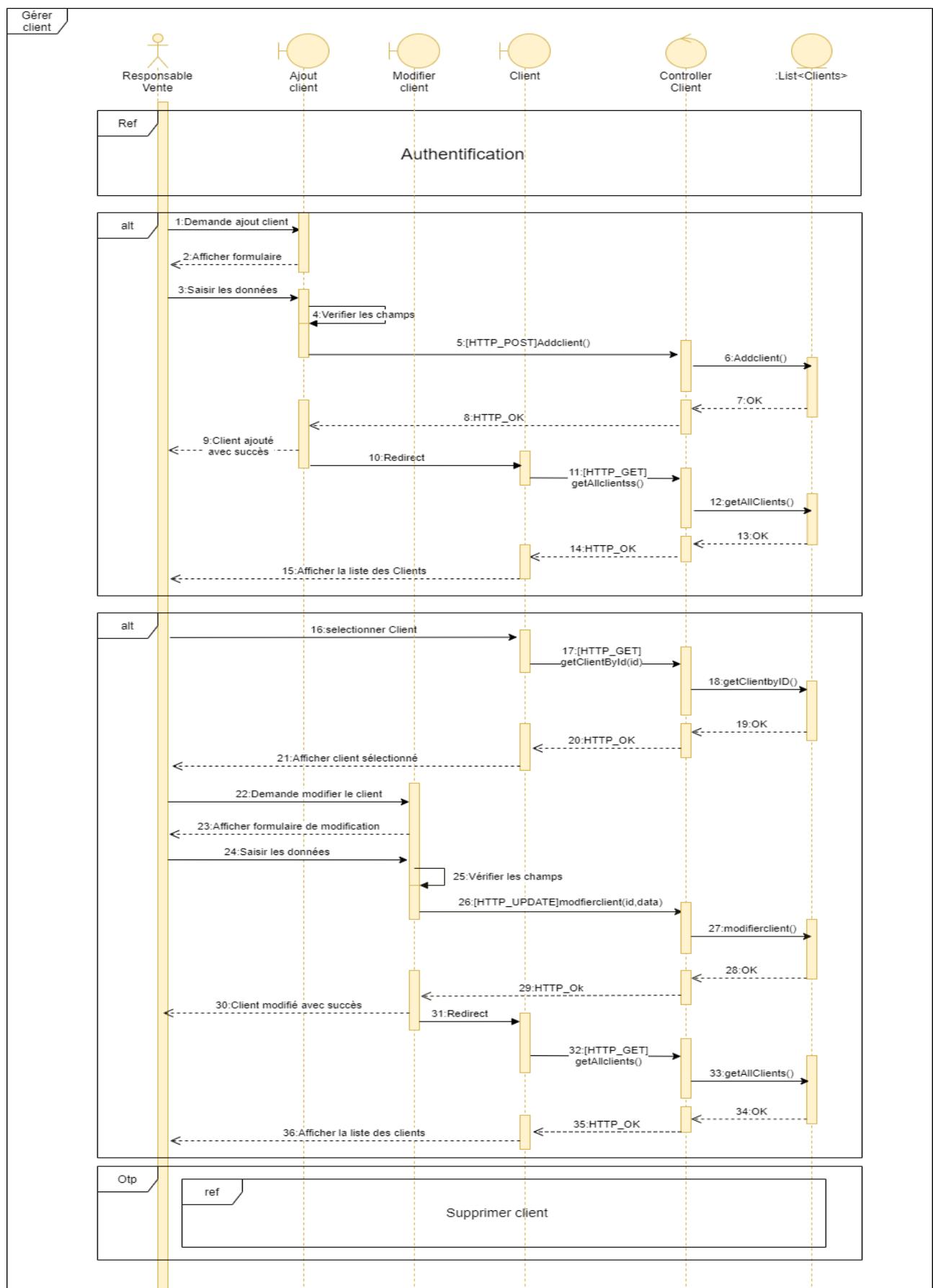


Figure 18 – Diagramme de séquence du scénario nominal Gérer client

## 4.2. Diagramme de séquence du scénario Gérer Facture vente

Au moment de la création d'une facture, un formulaire va s'afficher. On va sélectionner un client ainsi qu'un ou des produits à partir de la base de donnée, par la suite on va s'occuper du remplissage des autres champs du formulaire. Le système vérifie ensuite si les champs sont bien remplis, cette opération donnera lieu au calcul de la facture (TVA,TTC,HT) . Des lors, le bouton « Ajouter» devient Fonctionnel et en appuyant dessus, le« Controller Facture» reçoit une requête HTTP de type « POST »contenant dans son corps les informations de la factureà créer. Celui-ci ajoutera alors la Facture dans la base de données et va créer une nouvelle ligne de Facture et va informer l'acteur de la réussite de l'opération avec une requête HTTP OK. Pour le paiement d'une facture, l'acteur sélectionne la facture à payer, une requette HTTP GET est envoyée suite à laquelle les informations de cette facture seront affichée, il clique sur payer, le formulaire du paiement est affiché, l'acteur remplit le formulaire et clique sur "payer" le paiement suite a ça une vérification est effectuée et une requette HTTP POST est envoyée et notre facture est payée.

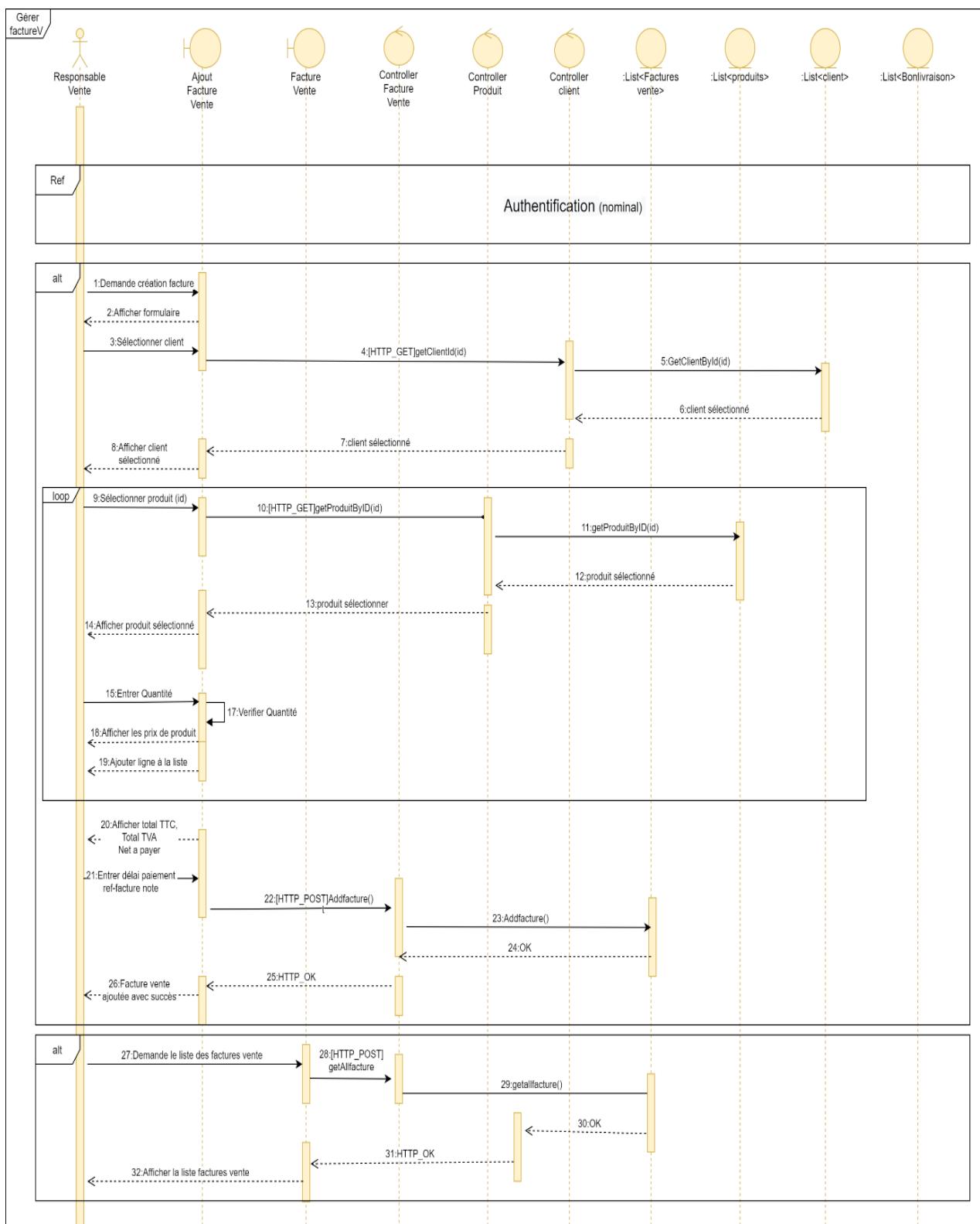


Figure 19— Diagramme de séquence du scénario nominal Gérer facture vente

### 4.3. Diagramme de séquence du scénario nominal effectuer paiement

Pour le paiement d'une facture, l'acteur sélectionne la facture à payer, une requête HTTP GET est envoyée suite à laquelle les informations de cette facture seront affichées. En cliquant sur payer, le formulaire du paiement est affiché, l'acteur rempli le formulaire et clique sur "payer" le paiement suite à ça, une vérification est effectuée et une requête HTTP POST est envoyée et la facture est payée.

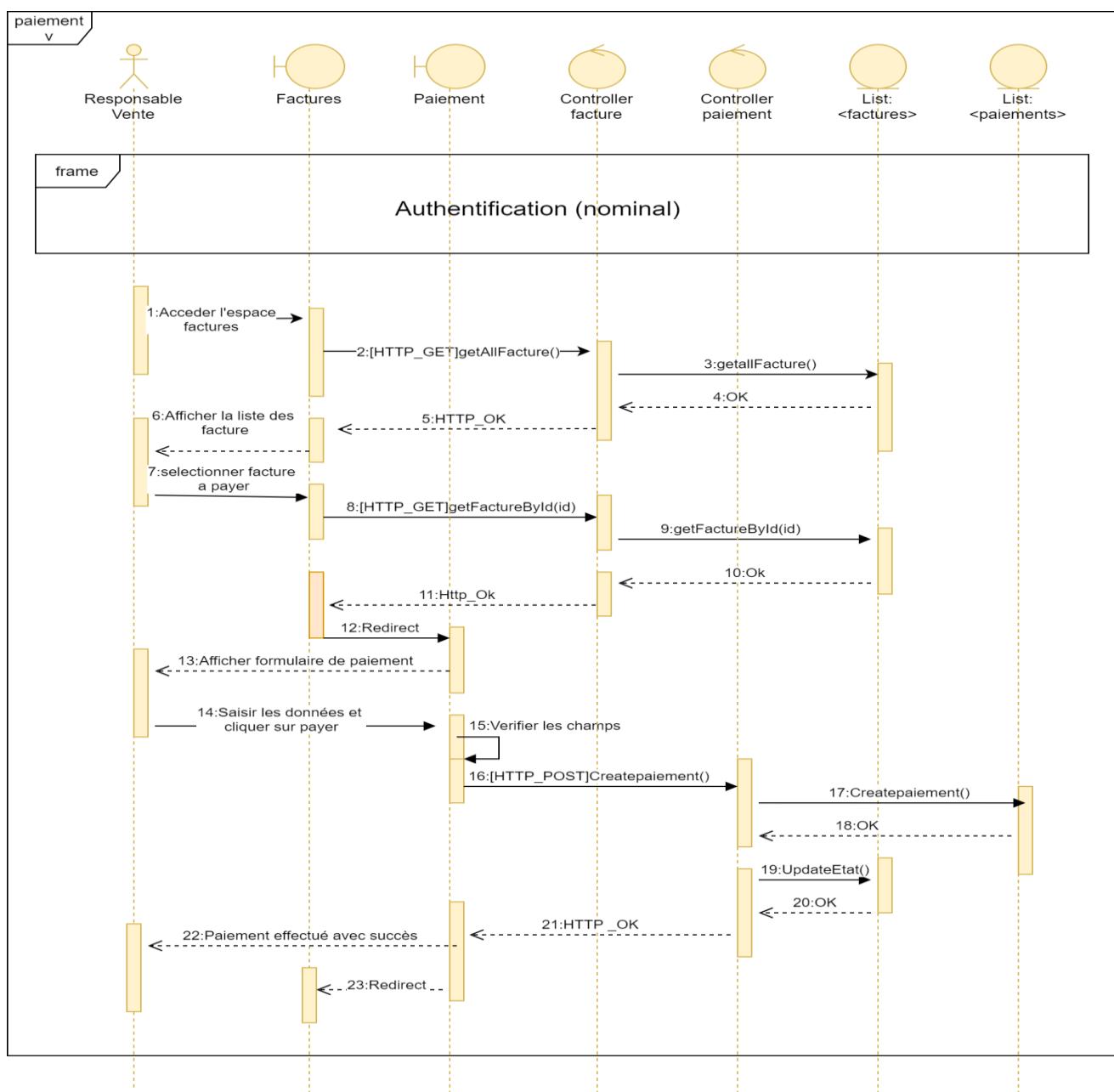


Figure 20– Diagramme de séquence du scénario nominal effectuer paiement

#### **4.4. Diagramme de séquence du scénario Gérer trésorerie**

Lors de la création d'une dépense, l'acteur est invité à s'authentifier et par la suite un formulaire d'ajout est affiché, puis il sélectionne un fournisseur, par la suite il s'occupe du remplissage des autres champs du formulaire. Le système vérifie ensuite si les champs sont bien remplis. Si c'est le cas, en appuyant le bouton « Créer » dessus, le « Controller dépense » reçoit une requête HTTP de type « POST » contenant dans son corps les informations du dépense à créer. Ce dernier ajoute le dépense dans la base de données, crée une nouvelle ligne et informe l'acteur de la réussite de l'opération avec une requête HTTP OK. Lors de la création d'une opération, l'acteur saisit les informations requises. Le système vérifie ensuite si les champs sont bien remplis. Si c'est le cas, le bouton « Créer » devient cliquable et en appuyant dessus le « Controller Opération » reçoit une requête HTTP de type « POST » contenant dans son corps les informations d'opération à créer. Ce dernier ajoute l'opération dans la base de données, crée une nouvelle ligne et informe l'acteur de la réussite de l'opération avec une requête HTTP OK.

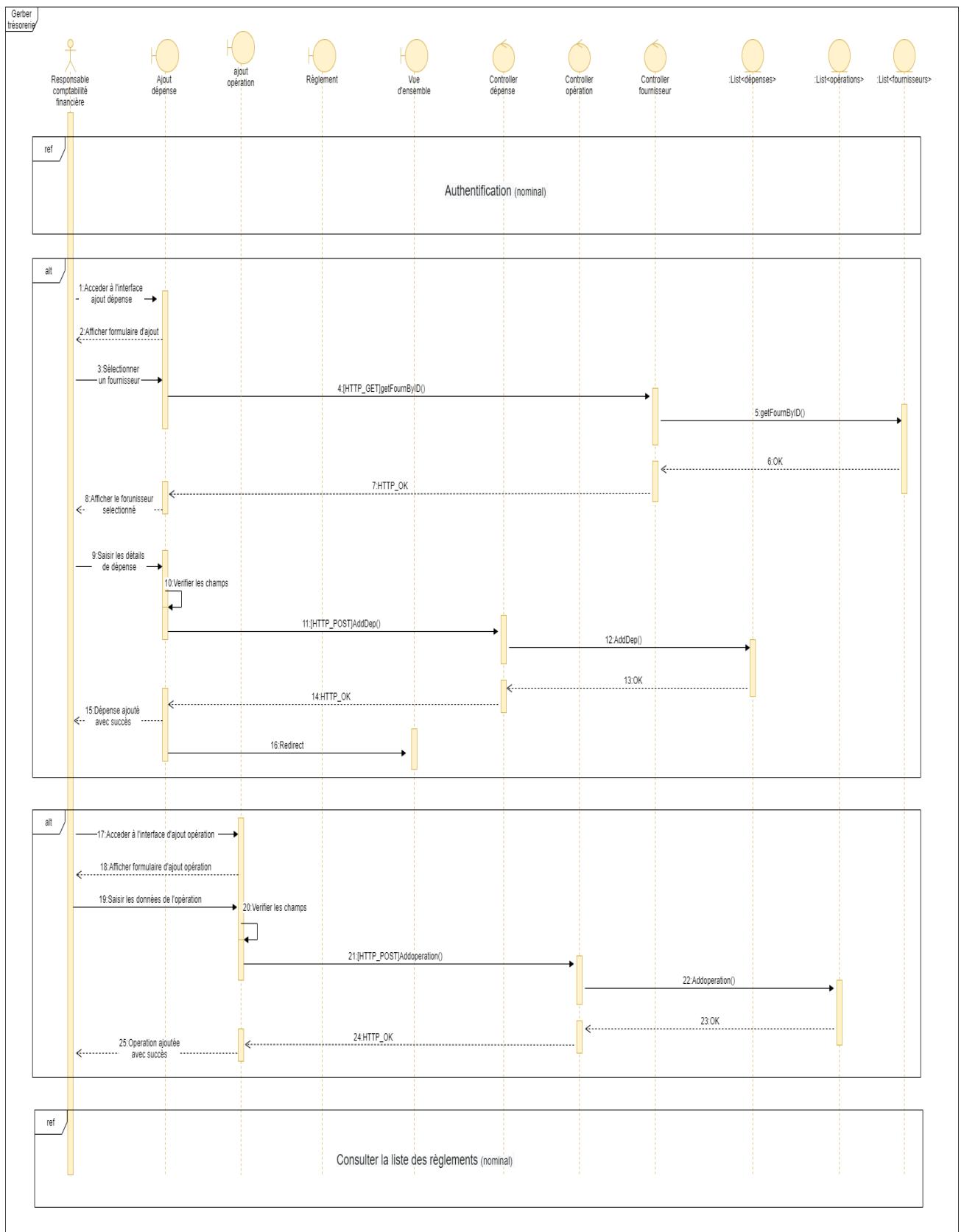


Figure 21 – Diagramme de séquence du scénario nominal Gérer trésorerie

## 4.5. Diagramme de séquence du scénario nominal Consulter la liste des règlement

Le Responsable financier et comptabilité peut accéder à l'interface Règlement .il permet également de consulter la liste des règlements qui sera apportent à toutes les dépenses, opérations et paiements des factures

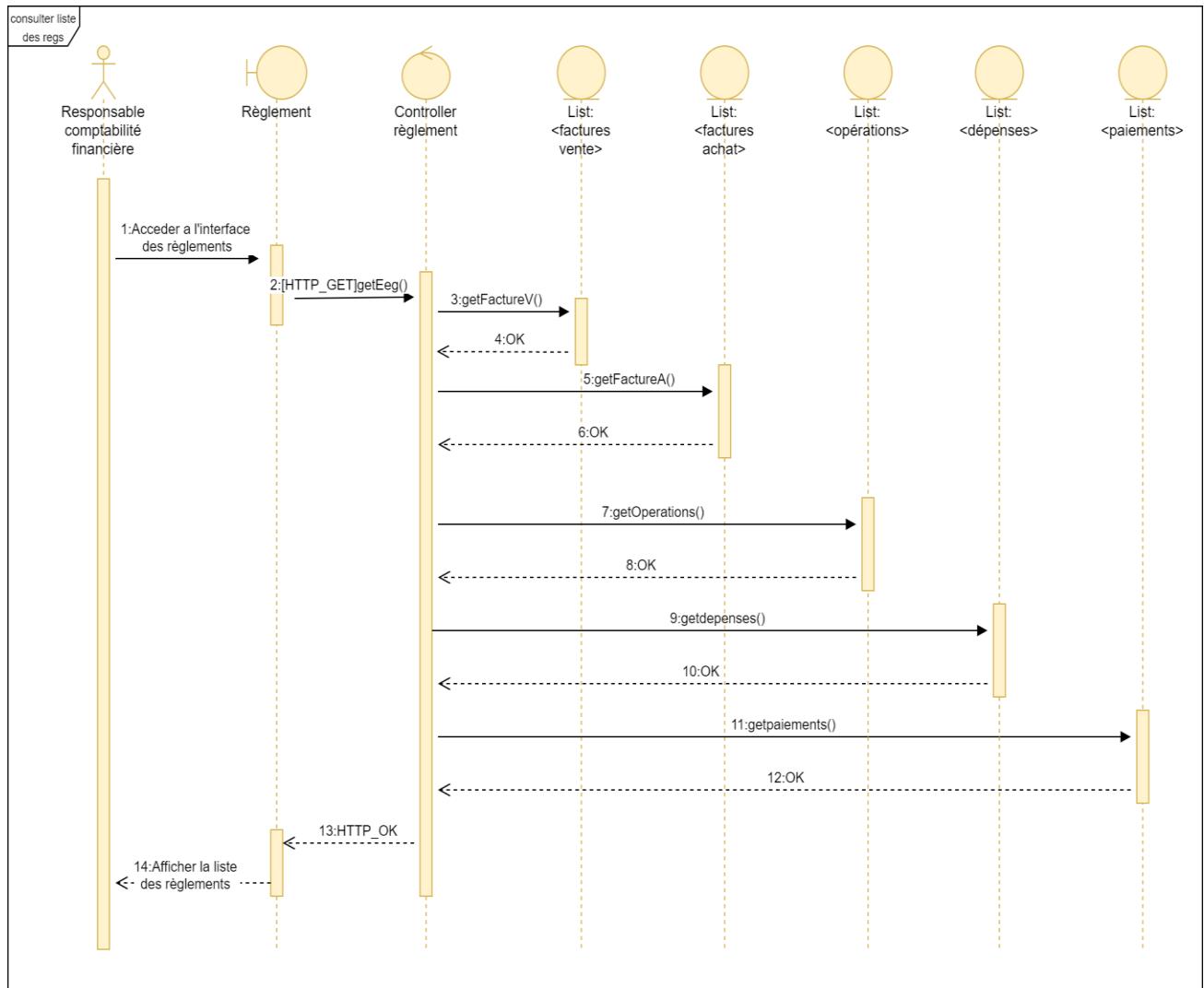


Figure 22 – Diagramme de séquence du scénario nominal Consulter la liste des règlements

## 4.6. Diagramme de séquence du scénario S'authentifier

Pour pouvoir utiliser l'application, l'utilisateur est invité à saisir correctement son email et son mot de passe dans l'interface « Connexion ». Une requête HTTP de type « POST » est transmise au contrôle « ControllerConnexion » contenant dans son corps l'email et le mot de passe saisis. Le système va vérifier si les informations correspondent à un compte existant

Si ce n'est pas le cas, une réponse HTTP UNAUTHORIZED est renvoyée à l'interface. Dans le cas où tout est en ordre, le contrôle récupère l'id de l'utilisateur depuis l'entité adéquate puis chiffre ce dernier avec le rôle de l'utilisateur dans un jeton d'authentification et le remet à l'interface.

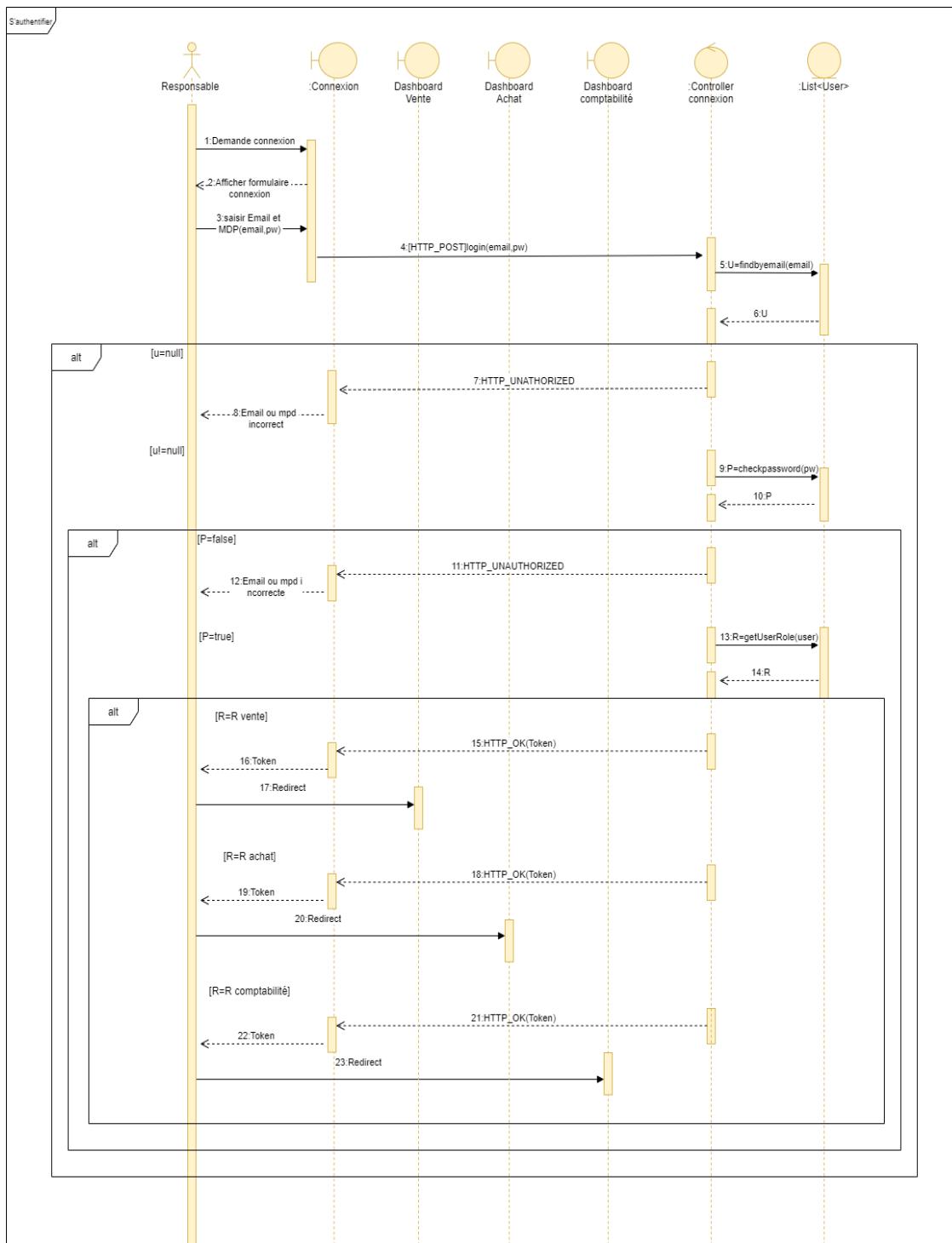


Figure 23 – Diagramme de séquence du scénario nominal S'authentifier

## 5. Correspondance entre modèles objets et relationnels :

Client(id , name,type,raisonsociale, matfiscale,civilite,email ,adresse,ntelephone)

Fournisseur(id,nom,civilite,raisonsociale,email,adresse,ntelphone,identifiant,matfiscale)

Product(id,name,category,marque,type,typep,typetaxe,pricev,priceht,pricetc;refconst,refint ,tva,enstock, desc)

Paiements(id ,reste,paye,fournisseur,reffact,date\_reglement,date\_echenace,#id\_facture)

Paiementscs(id ,reste,paye,client,reffact,date\_reglement,date\_echenace,#id\_facture)

Operations(id ,mode,typep,categorie,montant,datereg(note)

Liste\_product\_achats(id,quantite, #id\_product,libelle,#id\_facture)

Liste\_productventes(id,quantite,#id\_product,libelle,#id\_facture)

Factures(id,montant\_TTC,montant\_TVA,Total\_Ht,id\_client,etat,ref\_facture,date\_echeance ,Nom\_client, note, timbre\_fiscale)

Facturefs(id,montant\_TTC,montant\_TVA,Total\_Ht,#id\_fournisseur,etat,ref\_facture, date\_echeance, Nom\_fournisseur,note,timbre\_fiscale)

Depenses(id,#id\_fournisseur,type,fournisseur,montant,categorie,datereg(note)

Commandeclient(id,montant\_TTC,montant\_TVA,Total\_Ht,#id\_client,etat,ref\_facture, date\_echeance, Nom\_client,note,timbre\_fiscale)

Commandeachat(id,montant\_TTC,montant\_TVA,Total\_Ht,#id\_fournisseur,etat,ref\_facture ,date\_echeance,Nom\_fournisseur,note,timbre\_fiscale)

Users( id ,name,email,password,# role\_id)

Role(role\_id ,key,name)

## Conclusion

À l'issu de ce chapitre, j'ai mis en place la partie analyse et conception de mon application, en ayant recours aux différents diagrammes d'uml. Le chapitre suivant sera réservé à la partie reporting.

---

## **CHAPITRE VI : Etude de la partie reporting**

---

## Introduction

Dans ce chapitre, je veux présenter, le processus ETL avec ses différents phases et illustrés quelques rapports réalisés avec Power Bi.

### 1. Phase d'ETL

L'ELT (Extract Transform Load) est le processus par lequel les données brutes sont extraites de leurs sources original (ERP, CRM..), puis chargées dans les sources cibles, généralement des entrepôts de données.

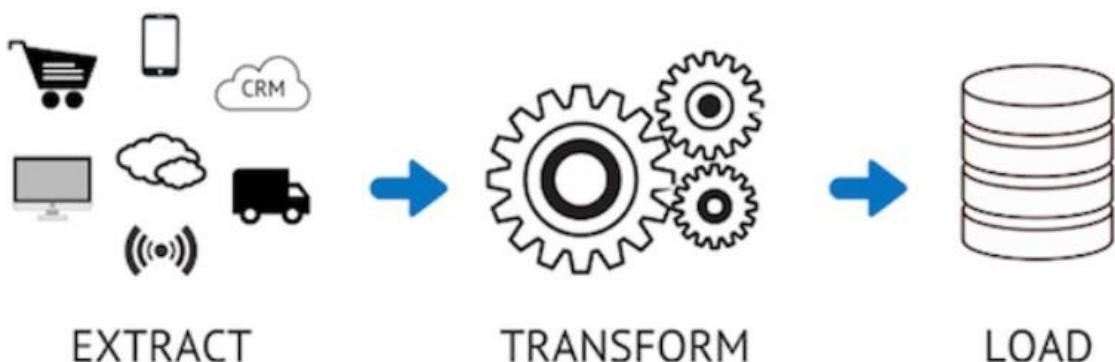


Figure 24 – ETL

#### 1.1. Extraction des données

On va extraire les données source de notre application stockées dans MySql Workbench.

## Chapitre 4 : Etude de la reporting

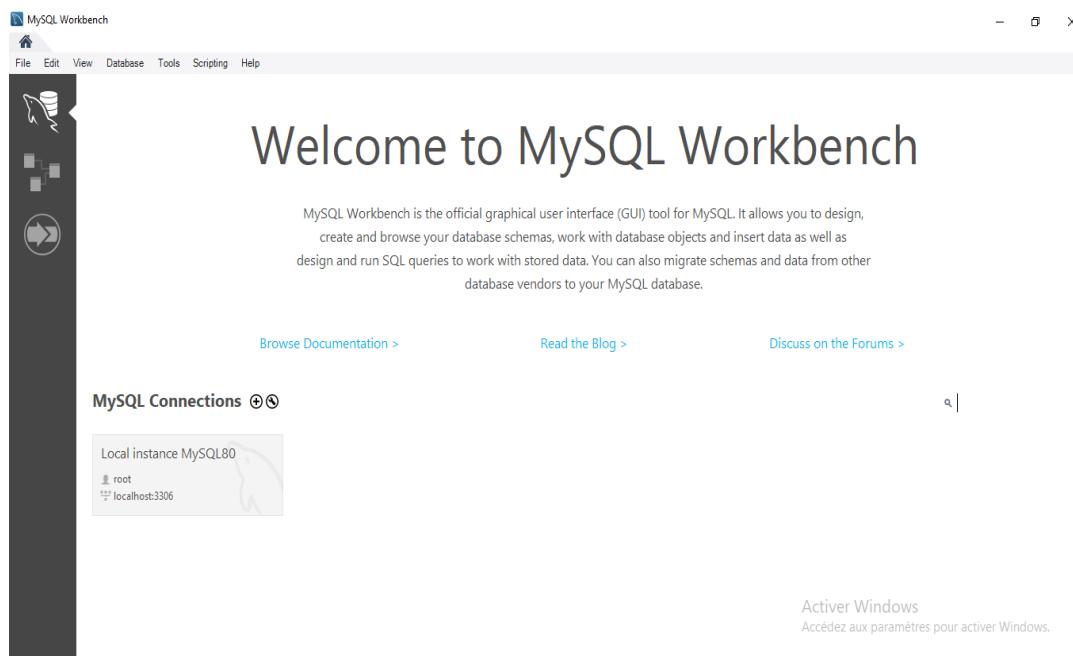


Figure 25 – interface d'accueil MySQL Workbench

Comme l'indique la figure précédente la connexion avec mysql workbench est effectuée on peut visualiser les tables de notre base et les relations entre eux.

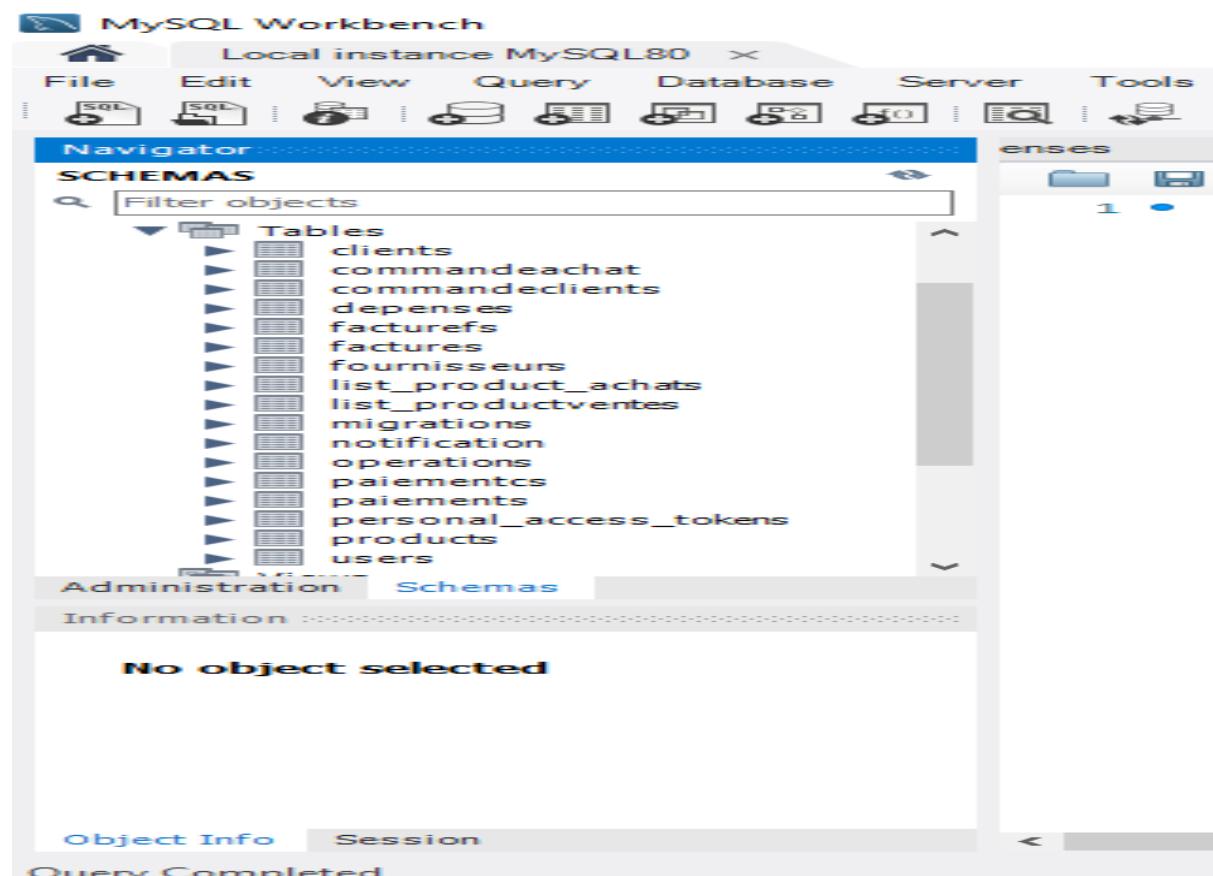


Figure 26 – les tables des données sources

## Chapitre 4 : Etude de la reporting

Après s'être connecté et avoir obtenu les tables de la base de données, nous connecterons Power bi avec MySQL Workbench pour assurer les étapes de transformation et chargement des données.

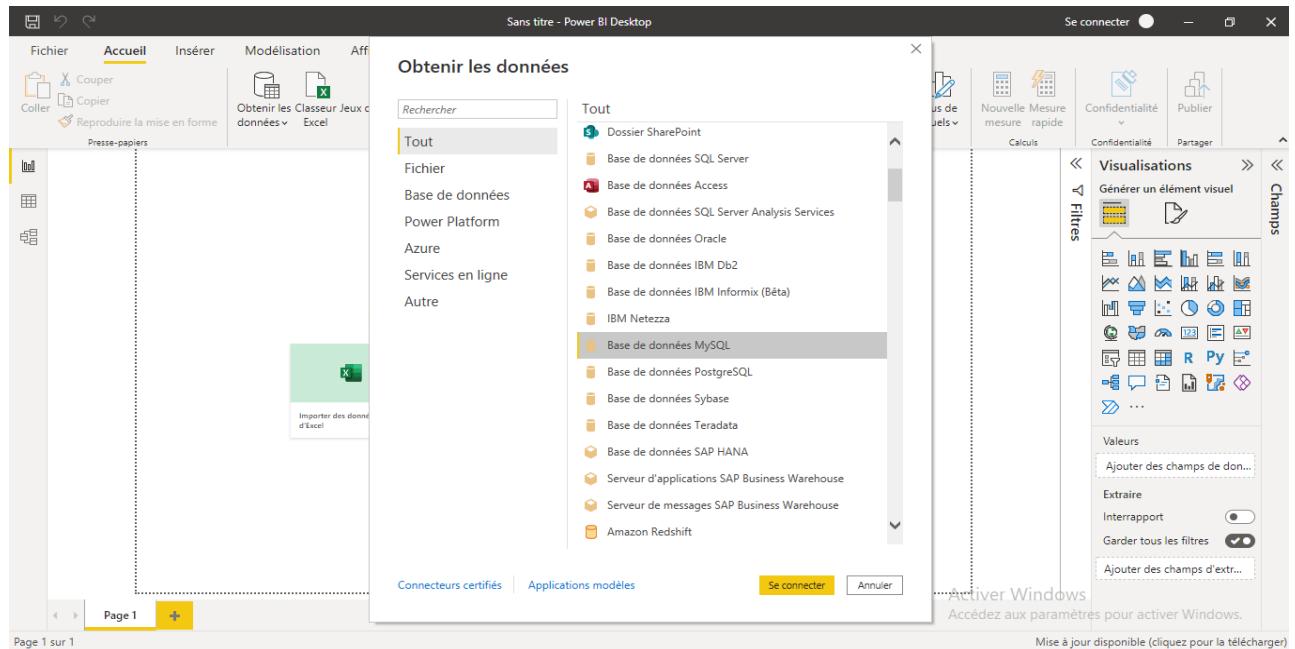


Figure 27 – choix de mysql

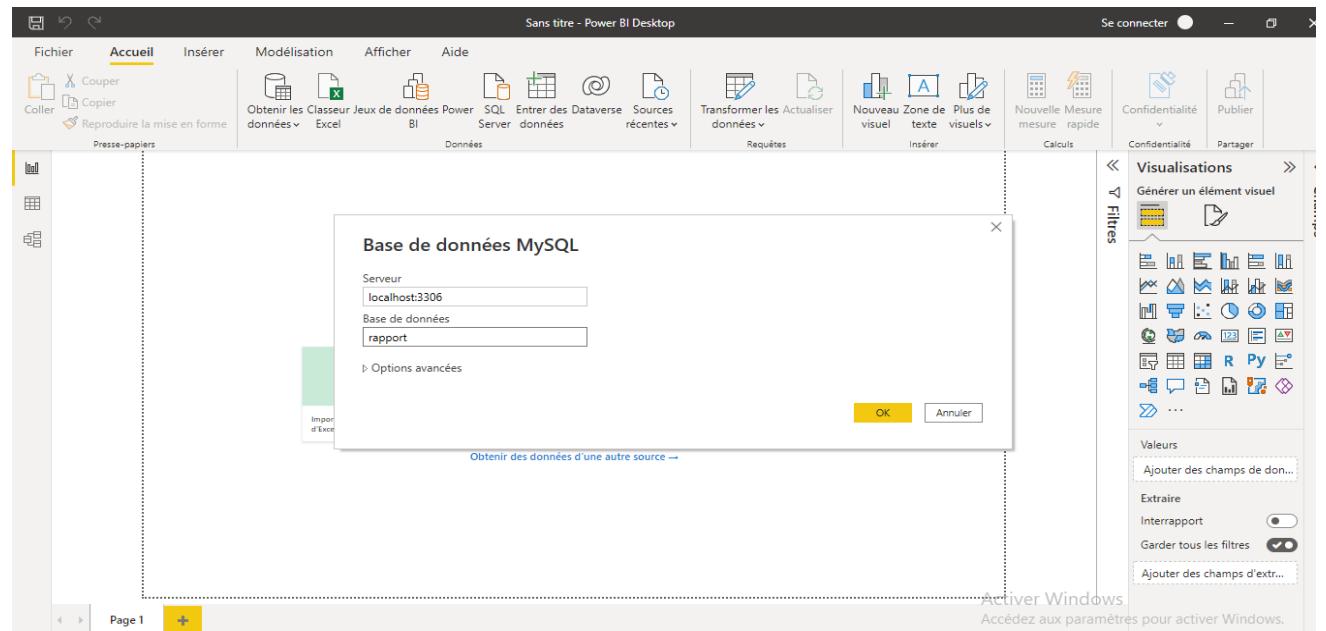


Figure 28 – connexion mysql avec power bi

## 1.2. Transformation des données

Nous avons appliqué quelques transformations pour les dimensions dont nous avons éliminé quelques données redondantes.

The screenshot shows the Power BI Data Transformations interface. On the left, there's a navigation pane titled "Navigateur" with a search bar and a tree view of datasets. The "localhost:3306: rapport [17]" node is expanded, showing various tables like "rapport.clients", "rapport.commandeachat", etc., with "rapport.products" selected and highlighted with a gray border. The main area displays the "rapport.products" table with the following schema and data:

id	name	category	marque	type	typep	typetaxe	pricev	priceht	pricettc	refconst	refint
1	disque dur 250 ssd	Cat	marque	materiel	TTC	fodec	1000	930	1000	0	0
2	carte graphique msi geforce	Cat	marque	materiel	TTC	fodec	500	465	500	0	0
3	pc portable i5 10ème gen	Cat	marque	materiel	TTC	fodec	3000	2790	3000	0	0
4	cable hdmi	Cat	marque	materiel	TTC	fodec	20	19	20	0	0
5	housses hosni	Cat	marque	service	TTC	fodec	3	3	3	0	0

At the bottom right, there are buttons for "Activer Windows", "Sélectionner les tables associées", "Charger", "Transformer les données", and "Annuler".

Figure 29 – Transformation des données avec power bi

## 1.3. Chargement des données

La dernière étape du processus ETL standard consiste à charger les données extraites et transformées dans leur nouvel emplacement.

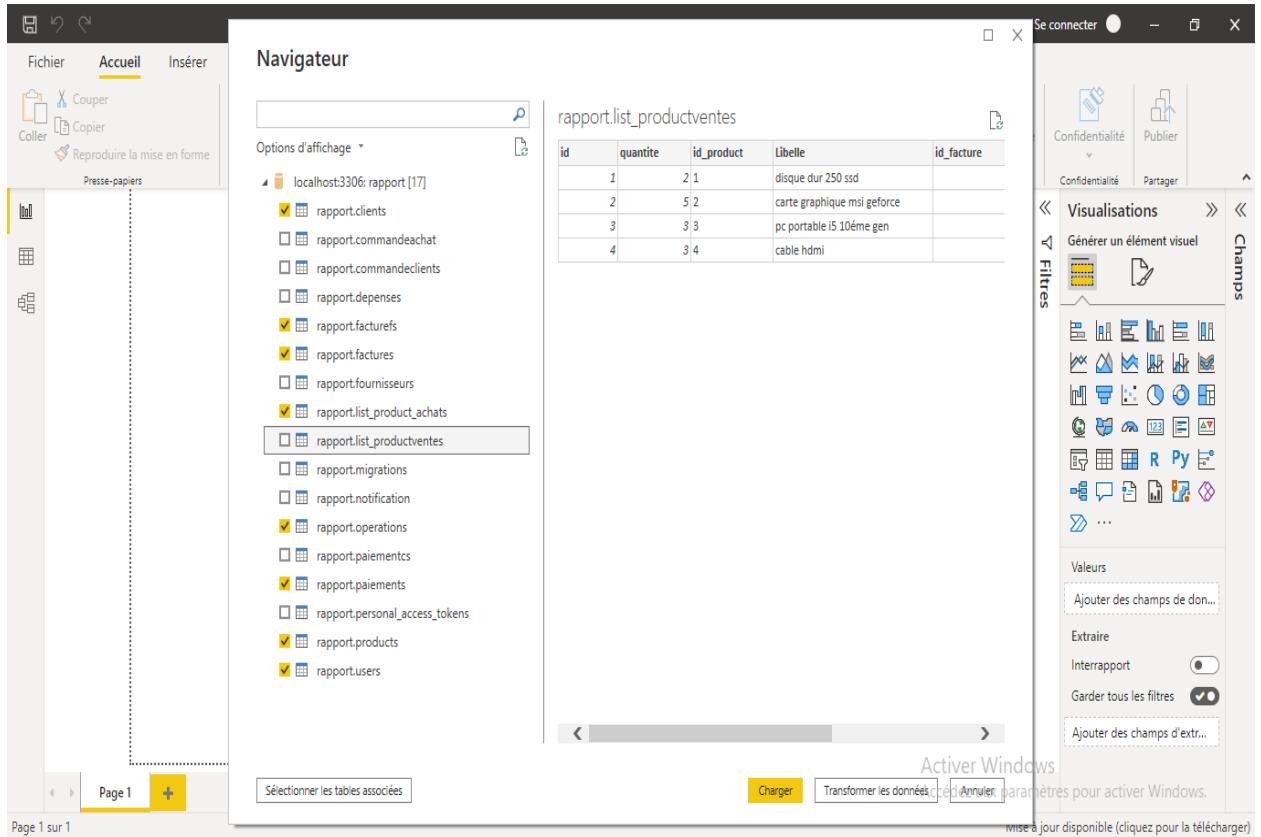
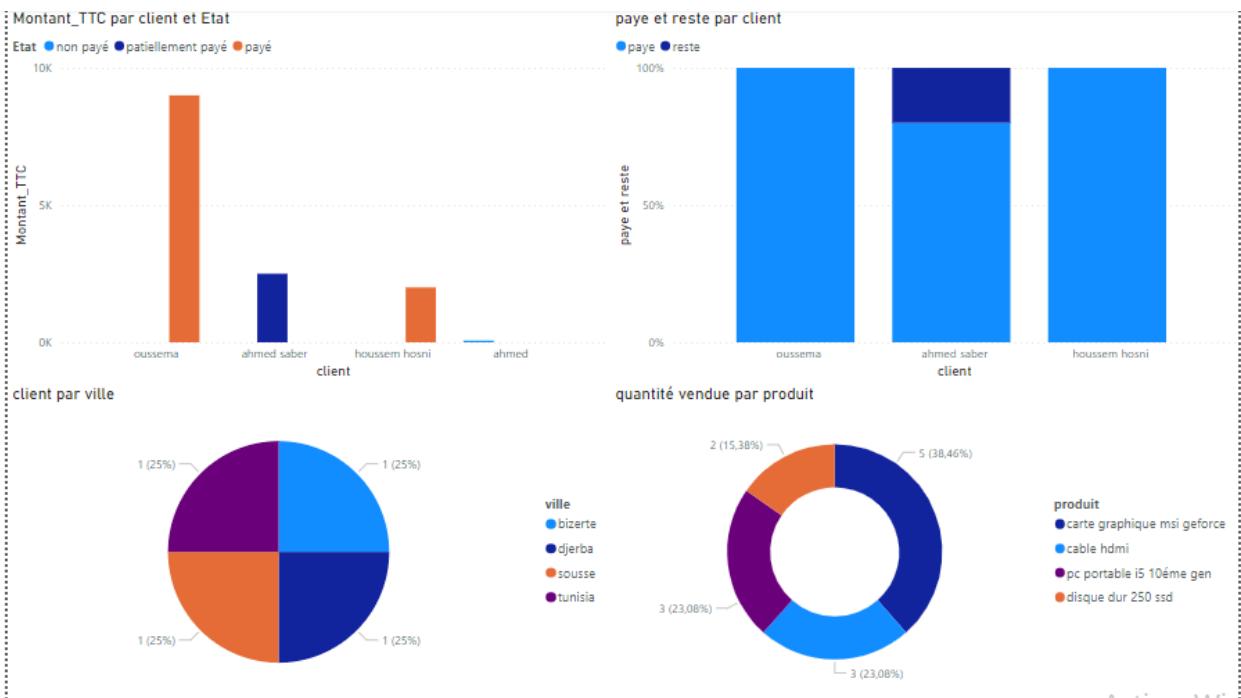


Figure 30 – chargement des données avec power bi

## 2. Interface «Rapport Vente»

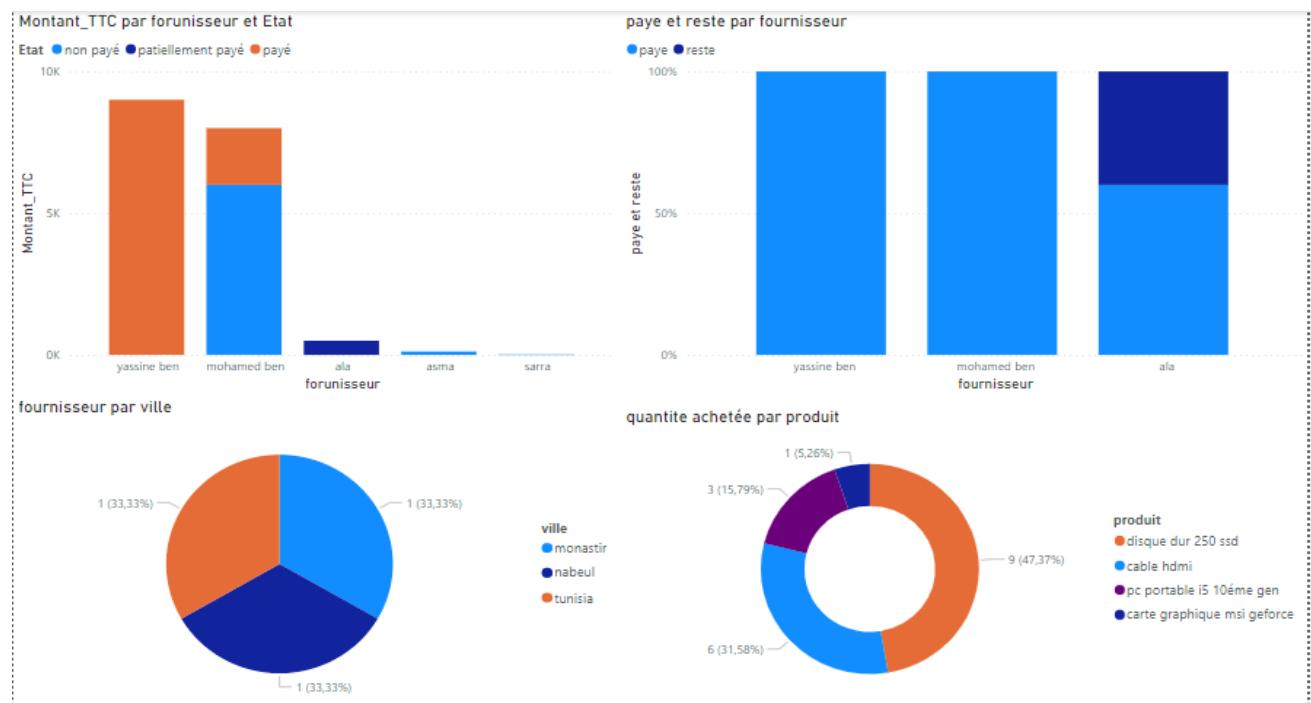
Le rapport des ventes permet au gestionnaire de suivre les activités de vente à partir de courbes et de statistiques contenant par exemple les montants par clients, les produits les plus vendus et les totaux clients.

## Chapitre 5 : Réalisation



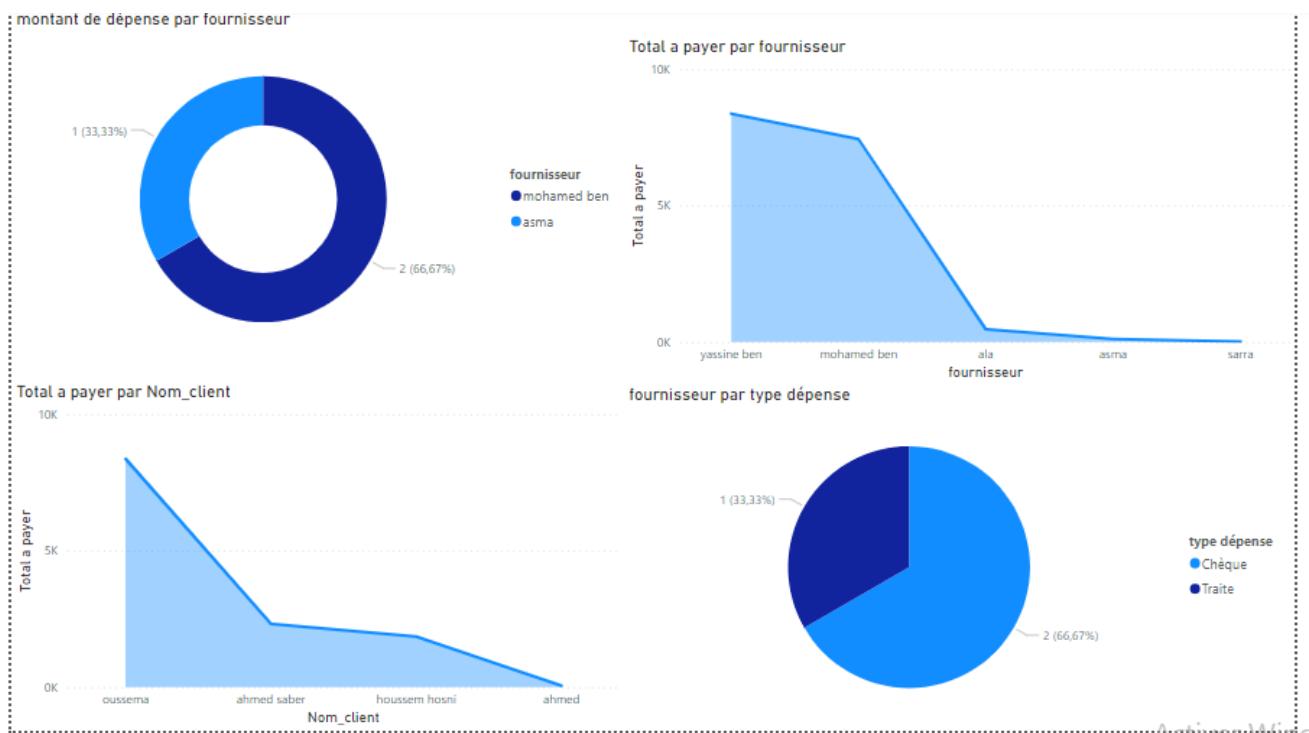
## 3. Interface «Rapport Achat»

Le rapport des ventes permet au gestionnaire de suivre les activités d'achat à partir de courbes et de statistiques contenant par exemple les montants par fournisseurs, les produits les plus achetés et les totaux fournisseurs.



## 4. Interface «Rapport Comptabilité»

Le rapport comptabilité permet au gestionnaire de suivre les activités d'achat vente et de suivre les états financiers de l'entreprise à partir de courbes et de statistiques contenant, par exemple, les montants par client/fournisseur, les totaux par client/fournisseur, les opérations et les dépenses.



## Conclusion

À l'issu de ce chapitre, j'ai mis en place la partie reporting de mon application, en ayant recours aux différentes phases ETL et quelques interfaces du rapports réalisés . 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 V :RÉALISATION**

---

## Introduction

Après avoir clôturé la phase de conception , la solution étant déjà choisie et bien étudiée, je présente dans ce chapitre quelques détails de réalisation. Tout d'abord, je présenterai un aperçu sur les technologies et les outils utilisés pour réaliser mon application. Finalement, je présenterai les différentes interfaces créées.

### 1. Environnement de travail

Les caractéristiques de la machine utilisée sont :

**Nom :**HP

**Processeur :** Intel Core I3-7020U CPU (2.30GHz 2.30 GHz)

**RAM :** 8GO

**Disque Dur :** 1 TO

**Type système :** Windows 64 bits

### 2. Outils de réalisation

Au niveau du système d'exploitation et des logiciels utilisés.

#### 2.1. Visual Studio Code

Visual Studio Code est un éditeur de code open source, gratuit et multiplateforme (Windows, Mac et Linux), développé par Microsoft. Principalement conçu pour le développement d'application avec JavaScript, TypeScript et Node.js, l'éditeur peut s'adapter à d'autres types de langages grâce à un système d'extension bien fourni.



Figure 31-Visual Studio Code

## 2.2. Trello

Trello est une application de gestion de projet gratuite qui permet d'organiser les projets sous forme de tableaux, eux-mêmes composés de listes en colonnes, qui répertorient des tâches sous formes de cartes. Inspiré de la méthode agile Kanban, Trello est un outil incontournable, destiné aux équipes qui cherchent une plateforme de référence pour leur gestion de projet. Trello est très intuitif et scalable, quelle que soit la taille de votre entreprise.



Figure 32 – Trello

## 2.3. Postman

Postman est un logiciel qui se focalise sur les tests des API. Il est devenu très populaire pour tester les Microservices, notamment grâce à sa simplicité et ses fonctionnalités très spécialisées.



Figure 33 – Postman

## 2.4.StarUML

StarUML est un logiciel qui sert à la création des diagrammes UML.



Figure 34 – StarUML

## 3.Les choix technologiques

### 3.1Laravel

Laravel est un framework web libre PHP, destiné au développement d'applications web suivant le modèle architectural (MVC) et basé sur Symfony. Certaines des caractéristiques de Laravel sont un système d'emballage modulaire avec un gestionnaire de dépendance dédié, différentes façons d'accéder aux bases de données relationnelles, des utilitaires qui aident au déploiement et à la maintenance des applications.



Figure 35 –LARAVEL

### 3.2. Angular

Angular est un Framework permettant de créer des applications clientes d'une seule page a l'aide de HTML et de Type Script. Angular est écrit en Type Script. Il implémente les fonctionnalités principales et facultatives sous la forme d'un ensemble de bibliothèques

Type\_Script.



Figure 36– Angular

### 3.3. PhpMyAdmin

C'est une application web de gestion pour les systèmes de gestion de base de données MySQL réalisé en PHP et distribué sous licence GNU GPL. Il s'agit l'une des plus célèbres interfaces pour gérer une base de données MySQL sur un serveur PHP. La figure suivante montre le logo PhpMyAdmin.



Figure 37 – phpMyAdmin

### 3.4. Power BI Desktop

Power BI est une solution d'analyse de données de Microsoft. Il permet de créer des visualisations de données personnalisées et interactives avec une interface suffisamment simple pour que les utilisateurs finaux créent leurs propres rapports et tableaux de bord.

Power BI est un ensemble de services logiciels, d'applications et de connecteurs qui fonctionnent ensemble pour transformer différentes sources de données en informations

visuelles, immersives et interactives. Plusieurs sources de données peuvent être utilisées telles que des fichiers Excel, des sources SQL, ou des entrepôts de données hybrides locaux ou sur le cloud. Les données sont personnalisées et interactives avec une interface suffisamment simple pour que les utilisateurs finaux créent leurs propres rapports et tableaux de bord. L'objectif est de faciliter la création des tableaux de bord afin d'améliorer les moyens de communications et de collaboration proposés par Microsoft. Il permet donc de collecter, construire et exposer les données au travers d'indicateurs. Son ergonomie permet par la suite d'animer des présentations interactives qui aideront à la prise de décision.



Figure 38 – power bi

### 3.5. MySQL Workbench

MySQL Workbench (anciennement MySQL administrator) est un logiciel de gestion et d'administration de bases de données MySQL créé en 2004. Via une interface graphique intuitive, il permet, entre autres, de créer, modifier ou supprimer des tables, des comptes utilisateurs, et d'effectuer toutes les opérations inhérentes à la gestion d'une base de données. Pour ce faire, il doit être connecté à un serveur MySQL.



Figure 39- MySQL Workbench

## 4. Présentation des interfaces de l'application

The screenshot shows the ASKUISINES application's dashboard. On the left is a dark sidebar with navigation links: Dashboard, Vente (selected), Commande Client, Facture, Clients (selected), and Produits. The main content area has a header "Accueil". It features four blue summary cards: "Clients" (4), "Produits" (4), "Commandes Vente" (0), and "Factures Vente" (4). Below these are two tables: "CLIENTS PAR CHIFFRE D'AFFAIRES" and "CLIENTS PAR ETAT / DELAI PAIEMENT".

Client	HT EUR	TTC EUR
houssem hosni	1860	2000
ahmed saber	2325	2500
oussema	8370	9000
ahmed	57	60

Client	Etat	TTC EUR
houssem hosni	payé	2022-06-04
ahmed saber	patiemment payé	2022-06-10
oussema	payé	2022-06-09
ahmed	non payé	2022-06-04

Figure 40 Interface «Tableau de bord vente»

The screenshot shows the ASKUISINES application's clients management interface. The sidebar navigation is identical to Figure 40. The main content area has a header "Clients" and a search bar "Recherche client". It features a table with columns: #, Nom, Type, N°Téléphone, Email, and Action. There are four client entries:

#	Nom	Type	N°Téléphone	Email	Action
1	houssem hosni	particulier	25924715	houssemhosni2018@gmail.com	
2	ahmed saber	particulier	1454224	bensaber@gmail.com	
3	oussema	professionel	45422	emxple@example.com	
4	ahmed	particulier	55157	emxple@example.com	

Figure 41 Interface «Clients»

## Chapitre 5 : Réalisation

Ajouter Client

Type du Client : particulier Civilité : M

Nom Complet : Email : example@example.com

Adress : N°Téléphone :

Raison Sociale : Matricule fiscale : 0 0

Annuler Enregistrer

Figure 42 Interface «Ajouter Client»

TOTAL	RÉGLÉES	À RÉGLER	EN RETARD
13560 EUR TTC	13000 EUR TTC	500 EUR TTC	0 EUR TTC
12612 EUR HT			

+ Creer Facture

RÉFÉRENCE	Date	Client	Montant	Etat	Action
fac-	2022-06-03	houssem hosni	2000	payé	\$ 🖍️ 🌐
fac-	2022-06-03	ahmed saber	2500	partiellement payé	\$ 🖍️ 🌐
fac-	2022-06-03	oussema	9000	payé	\$ 🖍️ 🌐
fac-f	2022-06-03	ahmed	60	non payé	\$ 🖍️ 🌐

Figure 43 Interface «Factures vente»

## Chapitre 5 : Réalisation

Figure 44 Interface «Ajouter factures vente»

Figure 45 Interface «paiement facture vente»

## Chapitre 5 : Réalisation

The screenshot shows the ASKUISINES accounting dashboard. On the left, a sidebar menu includes 'Dashboard' (selected), 'Trésorerie', 'Compte Bancaire', 'Balance Client', and 'Balance Fournisseur'. The main area displays several key figures: 4 Clients, 4 Produits, 0 Commandes Vente, 4 Factures Vente, 5 Fournisseurs, 0 Commandes Achat, 5 Factures Achat, and a total of 1080 EUR in Trésorerie. It also shows current sales and purchases for the day, week, month, and year. Below these are two tables: 'FOURNISSEURS PAR CHIFFRE D'AFFAIRES' and 'FOURNISSEURS PAR ETAT / DELAI PAIEMENT', each listing five suppliers with their names, HT EUR, TTC EUR, and payment status. There are also sections for 'CLIENTS PAR CHIFFRE D'AFFAIRES' and 'CLIENTS PAR ETAT / DELAI PAIEMENT', each listing five clients with their names, HT EUR, TTC EUR, and payment status.

Figure 46 Interface «Tableau de bord comptabilité»

The screenshot shows the ASKUISINES interface focusing on the 'Balance Client' section. The sidebar menu is identical to Figure 46. The main content area is titled 'Balance Client' and displays a table with columns: RÉFÉRENCE, CLIENT, CUMUL DEBIT, CUMUL CRÉDIT, and SOLDE DEBIT. The table contains three rows for clients 'houssem hosni', 'ahmed saber', and 'oussema', showing their respective balance details.

Figure 47 Interface «Balance Client»

## Chapitre 5 : Réalisation

The screenshot shows the ASKUISINES application interface. The left sidebar contains navigation links: 'comp', 'Dashboard', 'Trésorerie', 'Compte Bancaire', 'Balance Client', and 'Balance Fournisseur'. The 'Balance Fournisseur' link is highlighted with a blue background. The main content area is titled 'Balance Fournisseur' and displays a table with the following data:

RÉFÉRENCE	FOURNISSEUR	CUMUL DEBIT	CUMUL CRÉDIT	SOLDE CRÉDIT
Fac-	mohamed ben	0	2000	2000
Fac-	ala	200	300	500
Fac-	yassine ben	0	9000	9000

Figure 48 Interface «Balance fournisseur»

The screenshot shows the ASKUISINES application interface. The left sidebar contains navigation links: 'comp', 'Dashboard', 'Trésorerie', 'Compte Bancaire', 'Balance Client', and 'Balance Fournisseur'. The 'Trésorerie' link is highlighted with a blue background. The main content area is titled 'Trésorerie' and displays two boxes: 'Solde primaire 1080 EUR TTC' and 'Solde secondaire 0,000 EUR TTC', each with a 'Details' button.

Figure 49 Interface «Trésorerie»

## Chapitre 5 : Réalisation

The screenshot shows the ASKUISINES software interface. On the left is a dark sidebar with a logo and several menu items: 'comp', 'Dashboard', 'Trésorerie' (which is highlighted in blue), 'Compte Bancaire', 'Balance Client', and 'Balance Fournisseur'. The main area has a header with 'Accueil', 'Trésorerie', 'Recherche', and a search bar. Below the header is a title 'Solde primaire' with tabs 'VUE D'ENSEMBLE' (selected), 'RÈGLEMENT', and 'ENREGISTRER L'OPÉRATION'. A table lists transactions: #1 Chèque (500), #2 Chèque (50), and #3 Traite (135). A total of 745 EUR is shown at the bottom. A blue button on the right says '+ Ajouter Dépense'.

Figure 50 Interface «Vue d'ensemble»

This screenshot shows the 'Ajouter opération' (Add operation) screen. It has a similar layout to Figure 50, with the ASKUISINES logo and sidebar on the left. The main area is titled 'Solde primaire' with tabs 'VUE D'ENSEMBLE', 'RÈGLEMENT' (selected), and 'ENREGISTRER L'OPÉRATION'. It includes fields for 'Type de paiement' (Débit), 'Mode de paiement' (Espèces), 'Catégorie' (with input 'Cat'), 'Montant' (0), 'Date de règlement' (04/06/2022), and a 'Note' field ('pas de note'). At the bottom are 'Annuler' and 'Enregistrer' buttons.

Figure 51 Interface «Ajouter opération»

## Chapitre 5 : Réalisation

Figure 52 Interface «Règlements»

Figure 53 Interface «Gérer Profil»

## Conclusion

Dans ce chapitre, j'ai présenté l'environnement de développement de l'application et les différentes technologies utilisées. À la fin du chapitre, j'ai présenté les interfaces de l'application.

## Conclusion générale

Ce rapport a présenté le travail réalisé au sein de la société Globale Payment Gateway dans le cadre du projet de fin d'études. Il s'agit d'une conception et réalisation d'une plateforme web ayant pour finalité la gestion efficace des ventes, achat et comptabilité avec un partie de reporting.

En effet, nombreuses sociétés demeurent jusqu'à ce jour démunis d'un système informatisé capable de réaliser le traitement de gestion commerciale et facturation et restent sur des moyens traditionnels où rien n'est centralisé et par conséquent l'exécution de certaines opérations de gestion censée être simple devient de plus en plus compliquée sans oublier l'incapacité d'avoir une vue d'ensemble sur la santé de la société. Ces défaillances ainsi que les points faibles de certaines plateformes disponibles sur le marché nous ont conduit à développer une solution proposant à ses utilisateurs une expérience de qualité grâce à des fonctionnalités évoluées et une utilisation simplifiée. Le présent rapport détaille les différentes phases de réalisation du projet. J'ai décrit, tout d'abord, le cadre général en présentant une panoplie de solutions existantes. Ensuite, j'ai concentré sur la description des différentes fonctionnalités qu'ils peuvent exploiter. Dans la troisième partie, avec les diagrammes de classes et séquences nous a permis de mieux appréhender la structure et le comportement de notre système. Finalement nous avons mentionné les technologies dont nous avons fait recours et présenté les principales interfaces graphiques.

Cette solution développée a permis de valider nos objectifs initiaux et peut donner suite à une éventuelle évolution enrichie par de nouvelles fonctionnalités telles que :

**Créer une application mobile :** Pour permettre aux responsables de suivre les activités de l'entreprise en cas de déplacement

**Messagerie instantanée :** la création d'une messagerie instantanée facilitera la communication entre les différents responsables de l'entreprise.

En dernier lieu, ce projet a été une expérience enrichissante en terme d'apprentissage, mise en pratique des connaissances théoriques en utilisant des frameworks populaires et puissants sans oublier l'amélioration des aptitudes de communication qui représenteront sans doute des avantages lors de l'intégration du monde professionnel.

# **Webographie**

- <https://online.visual-paradigm.com/w/nmjpuzqb/drive/#diagramlist:proj=0&new>.
- <https://www.astera.com/fr/type/blog/data-warehouse-concepts/>.
- [https://fr.wikipedia.org/wiki/Diagramme\\_de\\_s%C3%A9quence](https://fr.wikipedia.org/wiki/Diagramme_de_s%C3%A9quence)
- <http://www.lebigdata.fr/datamart-definition>.
- <https://baptiste-wicht.developpez.com/tutoriels/conception/mvc/>.
- [https://mrproof.blogspot.com/2011/03/larchitecture-client-serveur.html/](https://mrproof.blogspot.com/2011/03/larchitecture-client-serveur.html).
- <http://laurentpiechocki.developpez.com/uml/tutoriel/lp/cours/>.
- <https://www.uml-sysml.org/diagrammes-uml-et-sysml/>
- [diagramme-uml/diagrammede-classe](https://www.uml-sysml.org/diagramme-uml/diagrammede-classe).
- <https://www.talend.com/fr/resources/guide-etl/>.
- <https://www.postman.com>.
- <https://trello.com/fr>.
- <https://code.visualstudio.com>.
- <https://powerbi.microsoft.com/fr-be/>.
- <https://www.phpmyadmin.net>.
- <https://swiver.io>.
- <https://birou.tn/fr>.
- <https://www.mysql.com/fr/products/workbench/>.
- <https://laravel.com>.
- <https://angular.io>.