

République Algérienne Démocratique et Populaire  
Ministère de L'Enseignement Supérieur et de la Recherche Scientifique

Université Abderrahmane MIRA – Bejaia

Faculté Des Sciences Exactes

Département d'Informatique



Tasdawit n Bgayet  
Université de Bejaia

# Mémoire de fin de cycle

En vue de l'obtention du diplôme de Master En Informatique

Projet pour l'obtention d'un diplôme "Projet d'Entreprise Economique" dans le cadre de  
l'arrêté ministériel n°1275

Option : Génie Logiciel

## T H È M E

---

Conception et réalisation d'une Solution SaaS de  
Facturation destinée aux Petites et  
Moyennes Entreprises

---

### Réalisé par :

M. TOUAZI Sirem

M. ZAID Youcef

### Devant le jury composé de :

Président : Mme ELBOUHISSI Houda, M.C.A, U.A.M Bejaia

Examineur : M. SALHI Nadir, Docteur, U.A.M Bejaia

Encadrant : M. MOHAMMEDI Mohamed, M.C.A, U.A.M Bejaia

Année universitaire : 2024/2025

## REMERCIEMENTS

Nous souhaitons tout d'abord exprimer notre profonde gratitude envers Dieu, le Tout-Puissant, pour Sa guidance précieuse, ainsi que pour la santé, le sérieux et la volonté qu'Il nous a accordés. C'est grâce à cette force divine que nous avons pu surmonter les obstacles et poursuivre nos objectifs avec une détermination sans faille.

Nous tenons également à adresser nos remerciements les plus sincères aux membres du jury pour avoir accepté d'évaluer notre travail. Leur expertise et leurs remarques éclairées sont essentielles pour enrichir notre réflexion et nous orienter vers de nouveaux horizons. Leur engagement envers l'enseignement et la recherche constitue une véritable source d'inspiration pour nous tous.

Un remerciement tout particulier s'adresse à notre encadrant, Monsieur MOHAMMEDI Mohamed. Sa bienveillance, sa disponibilité constante et la pertinence de ses conseils ont été des piliers fondamentaux dans la réalisation de ce projet. Sa patience et son implication ont véritablement fait la différence, et nous avons énormément appris de son approche rigoureuse et passionnée.

Nous exprimons également notre gratitude à l'ensemble de nos enseignants, dont la qualité de l'enseignement et l'engagement ont grandement contribué à notre épanouissement académique. Leur passion pour la transmission des connaissances et leur soutien indéfectible nous ont permis d'enrichir notre compréhension et d'affiner notre esprit critique.

Nous tenons à remercier chaleureusement nos familles et amis, qui ont été une source constante de soutien moral. Leurs encouragements, leur compréhension et leur patience durant cette période exigeante ont été des éléments cruciaux dans notre réussite. Ils ont su nous motiver et nous rappeler l'importance de la persévérance face aux défis.

Enfin, nous souhaitons témoigner notre profonde reconnaissance à toutes les personnes qui, de près ou de loin, ont contribué à l'élaboration de ce mémoire. Que ce soit par un conseil, un encouragement ou un simple mot de soutien, chaque geste a été précieux et a renforcé notre détermination à mener à bien ce projet.

*Dédicaces*

*À mes chers parents, pour leur amour inconditionnel, leurs sacrifices  
et leur soutien permanent.*

*À ma famille, pour sa présence chaleureuse et ses encouragements  
constants.*

*À mes amis, pour les moments de joie partagés et leur appui  
indéfectible.*

SIREM

*Dédicaces*

*À mes chers parents, qui ont toujours été ma source de motivation et de persévérance.*

*À mes frères et sœurs, pour leur affection, leur soutien et leur patience.*

*À tous mes amis qui m'ont soutenu et encouragé tout au long de ce parcours.*

YOUCEF

# TABLE DES MATIÈRES

<b>Table des matières</b>	<b>i</b>
<b>Table des tableaux</b>	<b>iii</b>
<b>Table des figures</b>	<b>iv</b>
<b>Table des acronymes</b>	<b>v</b>
<b>Introduction générale</b>	<b>1</b>
<b>1 Cadre et enjeux de la gestion des petites et moyennes entreprises (PME)</b>	<b>3</b>
1.1 Introduction . . . . .	3
1.2 Contexte du Projet . . . . .	4
1.3 Problématique . . . . .	4
1.4 Objectifs du Projet . . . . .	5
1.5 Justification du Projet . . . . .	6
1.6 Méthodologie . . . . .	6
1.7 Conclusion . . . . .	9
<b>2 Préparation de projet</b>	<b>10</b>
2.1 Introduction . . . . .	10
2.2 Spécification des besoins . . . . .	10
2.2.1 Capture des besoins fonctionnels . . . . .	10
2.2.2 Capture des besoins non fonctionnels . . . . .	11
2.3 Modélisation des besoins . . . . .	12
2.3.1 Identification des acteurs du système . . . . .	13

2.3.2	Diagramme de contexte du systeme . . . . .	13
2.3.3	Diagramme de cas d'utilisation global . . . . .	14
2.4	Pilotage du projet avec Scrum . . . . .	16
2.4.1	Équipe et rôles . . . . .	16
2.4.2	Backlog du produit . . . . .	16
2.5	Environnement de travail . . . . .	18
2.6	Architecture du produit logiciel . . . . .	18
2.7	Conclusion . . . . .	20
<b>3</b>	<b>Implémentation</b>	<b>21</b>
3.1	Introduction . . . . .	21
3.2	Conception . . . . .	21
3.2.1	Diagramme de classes global . . . . .	21
3.2.2	Diagrammes de séquence clés . . . . .	22
	Authentification de l'utilisateur . . . . .	22
	Création d'une facture . . . . .	23
	Paieement en ligne . . . . .	24
3.3	Interfaces utilisateur . . . . .	25
3.4	Conclusion . . . . .	27
	<b>Conclusion générale et perspectives</b>	<b>28</b>
	<b>Bibliographie</b>	<b>30</b>

## TABLE DES TABLEAUX

2.1	Tableau des rôles . . . . .	16
2.2	Backlog du produit de l'application à réaliser. . . . .	17

## TABLE DES FIGURES

1.1	Schéma représentant le fonctionnement de Scrum [7]. . . . .	7
1.2	Classification des différents types de diagrammes UML [?]. . . . .	9
2.1	Diagramme de contexte de l'application à réaliser . . . . .	14
2.2	Diagramme de cas d'utilisation global . . . . .	15
2.3	Architecture logicielle MVC. . . . .	20
3.1	Diagramme de classes global de l'application à réaliser. . . . .	22
3.2	Diagramme de séquence pour l'authentification. . . . .	23
3.3	Diagramme de séquence pour la création d'une facture. . . . .	24
3.4	Diagramme de séquence pour le paiement en ligne. . . . .	25
3.5	Enregistrer un paiement. . . . .	26
3.6	Liste des factures. . . . .	26
3.7	Interface d'abonnement. . . . .	27

## TABLE DES ACRONYMES

<b>A</b>	AI	Article d'Imposition.
<b>C</b>	CIB	Carte Interbancaire.
	CRM	Customer Relationship Management (Gestion de la Relation Client).
	CSS	Cascading Style Sheets (Feuilles de Style en Cascade).
	CSV	Comma-Separated Values (Valeurs Séparées par des Virgules).
<b>D</b>	DA	Dinar Algérien.
	DZD	Dinar Algérien (Code ISO de la monnaie).
<b>E</b>	ERP	Enterprise Resource Planning (Progiciel de Gestion Intégré).
<b>H</b>	HTML	HyperText Markup Language (Langage de Balisage Hypertexte).
	HT	Hors Taxe.
<b>I</b>	ID	Identifiant.
<b>J</b>	JH	Jour-Homme.
<b>K</b>	KPI	Key Performance Indicator (Indicateur Clé de Performance).
<b>M</b>	MVC	Modèle-Vue-Contrôleur.

<b>N</b>	NIF	Numéro d'Identification Fiscale.
	NIS	Numéro d'Identification Statistique.
<b>O</b>	OMG	Object Management Group.
<b>P</b>	PDF	Portable Document Format (Format de Document Portable).
	PHP	Hypertext Preprocessor.
	PME	Petites et Moyennes Entreprises.
<b>R</b>	RC	Registre du Commerce.
<b>S</b>	SaaS	Software as a Service (Logiciel en tant que Service).
	SME	Small and Medium-sized Enterprises (équivalent anglais de PME).
<b>T</b>	TTC	Toutes Taxes Comprises.
	TVA	Taxe sur la Valeur Ajoutée.
<b>U</b>	UML	Unified Modeling Language (Langage de Modélisation Unifié).
	UUID	Universally Unique Identifier (Identifiant Unique Universel).

## *INTRODUCTION GÉNÉRALE*

La transformation numérique représente un levier de croissance et de compétitivité incontournable pour les petites et moyennes entreprises. Dans un contexte économique en constante mutation, l'optimisation des processus administratifs, notamment la gestion de la facturation, devient cruciale. Traditionnellement chronophage et sujette aux erreurs, la facturation manuelle ou via des outils inadaptés peut entraver la performance des PME (Petite ou moyenne entreprise).

Malgré la disponibilité de certains logiciels de facturation sur le marché algérien comme Sage, Dolibarr, Fatoura et InnovaCOM, nombre d'entre eux ne répondent pas adéquatement aux besoins spécifiques des PME locales. Ces solutions existantes souffrent souvent de complexité excessive, d'une accessibilité limitée (logiciels de bureau non connectés), d'une mauvaise adaptation à la législation algérienne en vigueur, et de coûts d'acquisition et de maintenance élevés. Ces lacunes entraînent une persistance des méthodes traditionnelles, sources d'erreurs fréquentes, de retards de paiement, et d'une vision financière imprécise, impactant négativement la trésorerie, les relations clients et la capacité des PME à prendre des décisions éclairées.

Ce mémoire s'inscrit dans cette thématique en abordant la conception et la réalisation d'une solution technologique moderne visant à simplifier et à fiabiliser ce processus essentiel, permettant ainsi aux entreprises de se concentrer sur leur cœur de métier et leur développement. Cette approche SaaS (Software as a Service) s'appuie sur les définitions et recommandations du NIST [1] concernant les services cloud.

L'objectif principal de ce projet est de développer une solution SaaS (Software as a Service) de facturation, performante, intuitive et spécifiquement adaptée aux réalités et aux exigences des PME algériennes, en assurant la conformité réglementaire locale, la simplicité d'utilisation, et l'accessibilité sur mobile avec un coût adapté au marché national. Cette solution vise à combler les insuffisances des outils existants en offrant une plateforme centralisée pour la gestion des clients, des produits et services, la création rapide de devis et de factures conformes, le suivi rigoureux des paiements, la génération de rapports financiers pertinents et l'assurance d'une conformité

avec la réglementation fiscale algérienne. L'ambition est de fournir un outil qui non seulement automatise les tâches, mais aide également les entrepreneurs à mieux piloter leur activité.

Pour atteindre cet objectif, ce mémoire détaille la conception et la réalisation de cette solution en trois chapitres. Le premier chapitre "**CADRE ET ENJEUX DE LA GESTION DES PETITES ET MOYENNES ENTREPRISES**" introduira le contexte général, la problématique et les objectifs du projet. Le deuxième chapitre "**PRÉPARATION DE PROJET**" sera consacré à la préparation du projet, incluant la spécification des besoins fonctionnels et non fonctionnels, la modélisation du système, la méthodologie de gestion de projet Scrum adoptée, ainsi que la description de l'environnement de développement et de l'architecture logicielle. Enfin le troisième chapitre "**IMPLÉMENTATION**", présente le développement de l'application, depuis la conception (diagrammes de classes et de séquence) jusqu'aux interfaces utilisateur, en illustrant les principales fonctionnalités réalisées.

Enfin, ce mémoire s'achève par une conclusion générale et une ouverture sur les perspectives d'évolution du projet, en mettant en lumière les limites identifiées, les améliorations envisageables, ainsi que les pistes futures pour enrichir et étendre la solution proposée.

# CHAPITRE 1

## CADRE ET ENJEUX DE LA GESTION DES PETITES ET MOYENNES ENTREPRISES (PME)

### 1.1 Introduction

Dans un monde en perpétuelle évolution, les entreprises ne doivent pas arrêter d'innover et de s'améliorer si elles veulent rester sur le marché et ne pas être évincées par la concurrence. Cela est encore plus vrai lorsqu'on parle des PME (petites et moyennes entreprises), souvent proches du précipice de la fermeture. La moindre erreur pourrait leur faire perdre le fil du marché et les engloutir face aux grandes entreprises, qui représentent un danger constant pour elles.

Souvent dotées d'une main-d'œuvre réduite, les petites entreprises comptent entre 10 et 49 employés, tandis que les moyennes entreprises en emploient environ 50 à 250. Les PME font donc face à divers problèmes, notamment en ce qui concerne la gestion administrative, qui est répétitive et sujette à l'erreur. Celle-ci représente un coût important pour les entreprises, tant en termes de temps que de ressources financières, sans oublier l'impact sur les ressources humaines, déjà limitées.

La gestion des factures est l'un des aspects de la gestion administrative souvent sujette à l'erreur humaine, surtout dans des pays tels que l'Algérie, où l'utilisation des machines reste très rudimentaire. Le développement de la technologie et des systèmes de facturation peut donc aider à l'amélioration des services de facturation, ce qui en fait une priorité.

Dans ce chapitre, nous allons présenter le contexte et la problématique du projet, ainsi qu'une introduction aux objectifs que nous cherchons à accomplir dans ce projet et les raisons qui font que notre projet est d'actualité et innovant. Nous terminerons par la méthodologie que nous allons suivre durant la réalisation de ce projet.

## 1.2 Contexte du Projet

Les PME (petites et moyennes entreprises) en Algérie sont des structures employant jusqu'à 250 salariés. Selon le Ministère de l'Industrie, le pays comptait 1 320 664 PME à la fin juin 2022, employant plus de 3,2 millions de personnes. Parmi elles, près de 98 % sont des très petites entreprises (TPE) de moins de 10 salariés [2].

Le bulletin officiel note aussi que 56,19 % des PME sont des personnes morales, tandis que 43,79 % sont des personnes physiques (dont 20,18 % exercent une profession libérale et 23,62 % relèvent de l'artisanat) [2].

Ces entreprises jouent un rôle essentiel dans la lutte contre le chômage et dans la dynamisation du tissu économique local.

## 1.3 Problématique

Plusieurs logiciels de facturation sont disponibles sur le marché algérien, mais la plupart ne sont pas adaptés aux besoins et aux réalités spécifiques des PME. Ces logiciels présentent plusieurs lacunes importantes.

Tout d'abord, la complexité est un problème majeur. Les solutions ERP (Enterprise Resource Planning) complètes sont souvent trop complexes pour les PME. Leur interface utilisateur peut être complexe et peu intuitive, ce qui les rend difficiles à utiliser pour les PME qui ne disposent pas nécessairement de compétences informatiques avancées. Un logiciel de facturation efficace pour les PME algériennes doit être simple à utiliser, même pour les utilisateurs non expérimentés.

Ensuite, l'accessibilité limitée est un autre inconvénient. Les logiciels de bureau présentaient des avantages significatifs en Algérie lorsque l'infrastructure internet était sous-développée, principalement grâce à leur fonctionnement sans connexion internet. Cependant, ils comportaient plusieurs limitations : les mises à jour doivent être effectuées manuellement, l'accès est restreint à un seul appareil, il existe un risque important de perte des données stockées localement et le processus d'installation est souvent contraignant.

De plus, l'adaptation à la législation algérienne est cruciale. Certaines solutions ne prennent pas en compte les spécificités de la législation algérienne en matière de facturation et de comptabilité. Le décret exécutif n° 05-468 du 10 décembre 2005 fixe les conditions et les modalités d'établissement de la facture en Algérie. Ces réglementations définissent les mentions obligatoires sur les factures, les règles de comptabilisation des transactions et les obligations fiscales des entreprises [3].

Enfin, les coûts élevés sont un obstacle pour de nombreuses PME. Les solutions de gestion intégrées dédiées aux entreprises peuvent être coûteuses, ce qui les rend inaccessibles pour certaines PME, qui n'ont pas besoin de 80% des fonctionnalités offertes par ces grands ERP.

Ces insuffisances expliquent le recours persistant aux méthodes traditionnelles, générant des

erreurs de facturation dans 23% des transactions selon une enquête de la Chambre de Commerce d'Alger.

Les risques juridiques liés à une mauvaise gestion administrative et financière ne doivent pas être sous-estimés. Selon le Bilan du contrôle économique et de la répression des fraudes 2024 du Ministère du Commerce, le montant du défaut de facturation s'élève à 551,46 millions de DA, avec 865 cas de défaut de facturation et 55 cas d'établissement de factures non conformes. Ces chiffres alarmants soulignent l'importance cruciale de la conformité aux réglementations en vigueur pour éviter des sanctions financières et juridiques [4].

Pour faire face à ces défis et inefficacités, ce projet vise à développer un logiciel de facturation avec les objectifs suivants.

## 1.4 Objectifs du Projet

Développer un logiciel de facturation performant, adapté aux besoins des PME algériennes et contribuant à l'amélioration de leur gestion administrative et financière. Cela va être accompli en améliorant les lacunes dans les logiciels existants avec des solutions innovantes, notamment :

- Le logiciel doit permettre d'enregistrer facilement les informations des clients (nom, adresse, contact, etc.) et des produits/services (description, prix, etc.) pour les réutiliser rapidement lors de la création de factures.
- Le logiciel doit permettre de créer des devis et des factures en trois clics, avec la possibilité de les personnaliser (logo, mentions légales, etc.) et de les envoyer par email aux clients.
- Le logiciel doit permettre de suivre les paiements des clients, d'identifier les factures impayées et d'envoyer automatiquement des rappels de paiement par email ou SMS.
- Le logiciel doit permettre de générer des rapports financiers (chiffre d'affaires, bénéfices, etc.) et des statistiques (ventes par produit/service, clients les plus rentables, etc.) pour aider les entrepreneurs à piloter leur activité.
- Le logiciel doit être conforme aux lois et réglementations fiscales en vigueur en Algérie, notamment en ce qui concerne la TVA, les obligations de facturation et la numérotation des factures. Il doit également permettre de générer les documents nécessaires pour les déclarations fiscales.

Pour réussir dans un environnement commercial dynamique, les PME algériennes doivent miser sur l'innovation et l'automatisation. L'automatisation améliore l'efficacité en libérant du temps pour les tâches à valeur ajoutée, comme le développement commercial, et réduit les coûts en optimisant les ressources. Ces technologies renforcent la compétitivité, améliorent la satisfaction client et permettent aux PME de se différencier. Les solutions cloud, avec le "self-service provisioning", offrent flexibilité, accessibilité et réduisent les coûts, donnant aux PME plus de contrôle sur leurs besoins logiciels. En résumé, ces outils sont essentiels pour la croissance et le succès des PME algériennes.

## 1.5 Justification du Projet

Dans le contexte actuel de numérisation croissante de l'économie algérienne, le développement d'un logiciel de facturation spécifiquement conçu pour les PME est absolument nécessaire. Le gouvernement algérien encourage activement la transition numérique des entreprises, et plusieurs initiatives ont été mises en place pour soutenir le développement des PME dans cette voie. Un logiciel de facturation adapté aux besoins de ces entreprises peut les aider à se conformer aux exigences légales en constante évolution en matière de facturation, assurant ainsi leur conformité et leur tranquillité d'esprit.

De plus, l'adoption d'un tel outil contribue à améliorer leur image professionnelle auprès de leurs clients et partenaires, renforçant ainsi leur crédibilité et leur position sur le marché. Par ailleurs, la digitalisation et l'automatisation des processus, permises par un logiciel de facturation, peuvent considérablement augmenter la compétitivité des PME en leur permettant de gagner en efficacité, de réduire leurs coûts opérationnels et d'offrir des services de qualité supérieure.

L'adoption d'un logiciel de facturation adapté offre une multitude d'avantages significatifs pour les PME algériennes. En automatisant les tâches administratives, il permet de réaliser des gains de temps et d'argent considérables tout en réduisant les erreurs potentielles liées à la facturation et à la saisie de données. De plus, un tel logiciel améliore la gestion financière en facilitant le suivi des paiements, la gestion de la trésorerie et la génération de rapports financiers précis. Il assure également la conformité aux exigences légales en matière de facturation, renforce l'image professionnelle de l'entreprise et fournit des données financières fiables pour une prise de décision éclairée.

## 1.6 Méthodologie

Scrum est une méthode de conception agile utilisée dans la gestion des projets informatiques tels que la création de sites et d'applications web. Cette approche agile, dont les principes sont détaillés par Cohn [5], consiste en l'exécution de sprints, qui sont des cycles de travail itératifs et incrémentaux de 1 à 4 semaines, généralement sur lesquels on travaille sur des éléments du backlog, qui est une liste de fonctionnalités priorisées. Puis, nous avons le sprint planning, qui est une réunion pour définir les objectifs du sprint, le daily scrum, une réunion de 15 minutes journalières pour parler des progrès et des problèmes rencontrés lors des sprints, scrum review, réunion en fin de sprint pour présenter le travail accompli et avoir le retour des parties prenantes, et enfin le sprint rétrospective, une réunion en fin de sprint pour examiner ce qui a bien ou mal fonctionné et commencer à planifier le sprint suivant si ce dernier est nécessaire [6].

La figure 1.1 illustre le processus complet de la méthodologie Scrum que nous avons adoptée pour ce projet.

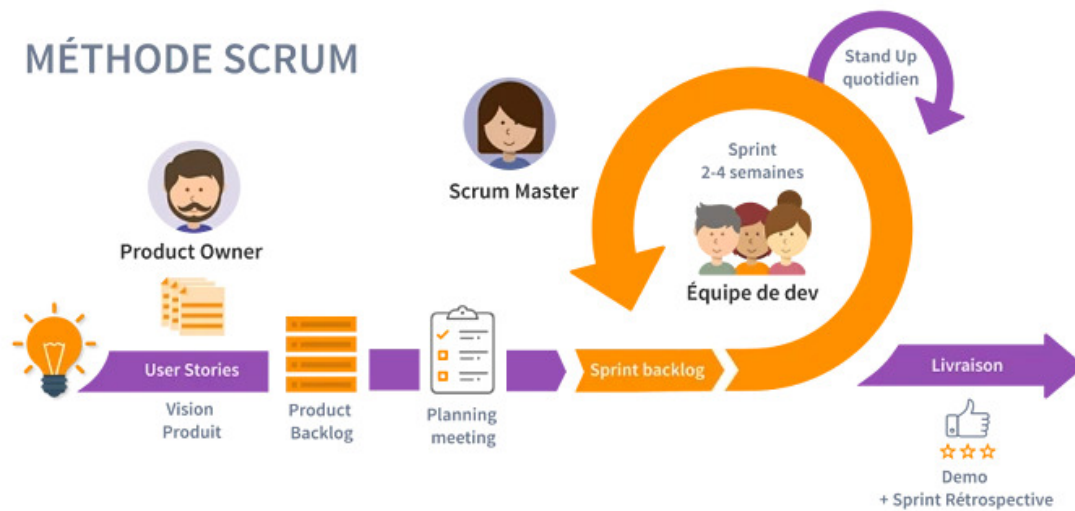


FIGURE 1.1 – Schéma représentant le fonctionnement de Scrum [7].

- **Product owner** est l'interlocuteur entre l'équipe de développement et le client. Il aide le client à définir les besoins de son produit avec un langage précis, compréhensible par l'équipe de développement. Il s'assure que cette dernière développe en priorité les fonctionnalités les plus importantes, car c'est le Product Owner qui définit le Product Backlog.
- **Product backlog** est une liste des fonctionnalités à développer, à améliorer ou à corriger. Elle contient toutes les tâches à effectuer pour aboutir à un produit fini et prêt à être livré au client après tests. Cette liste est gérée par le Product Owner, et elle est évolutive et priorisée.
- **Scrum Master** est l'un des acteurs clés qui assiste l'équipe de développement dans l'exécution de la méthode Scrum. Il veille au bon déroulement du travail de cette dernière afin d'éviter tout ralentissement
- **L'équipe de développement** est composée de développeurs, bien sûr, mais aussi de designers, d'interprètes et d'ingénieurs du son. On peut donc dire que l'équipe est formée en fonction des besoins du produit.
- **Planning Meeting** Une réunion pour définir les objectifs du sprint.
- **Sprint Backlog** est une liste établie au préalable qui détermine l'ensemble des tâches à accomplir pendant un Sprint.
- **Sprint** C'est une période de travail, généralement de 1 à 4 semaines, pendant laquelle l'équipe de développement développe et livre l'ensemble des fonctionnalités prédéfinies lors du planning meeting.
- **Stand-up quotidien** est une réunion quotidienne courte et rythmée entre le Scrum Master et l'équipe de développement pour connaître l'avancement de chacun et les problèmes

rencontrés durant le Sprint en cours, afin d'identifier rapidement les obstacles

- **Daily Scrum** Une réunion quotidienne de 15 minutes pour parler des progrès et des problèmes rencontrés lors des sprints.
- **Sprint Review** Une réunion en fin de sprint pour présenter le travail accompli et obtenir le retour des parties prenantes.
- **Sprint Retrospectives** Une réunion en fin de sprint pour examiner ce qui a bien ou mal fonctionné et commencer à planifier le sprint suivant, si nécessaire.

L'**UML** (*Unified Modeling Language*) est un langage de modélisation visuel standardisé, souvent utilisé en génie logiciel lors de la conception d'un logiciel, tout au long de son cycle de vie. Il offre différentes perspectives sur le projet afin de clarifier les exigences. Il permet également de structurer la pensée et de faciliter la collaboration entre les différents acteurs du projet [8]. La modélisation des cas d'utilisation, qui constitue une partie essentielle de notre approche, suit les meilleures pratiques décrites par Cockburn [9], tandis que l'analyse et la conception orientée objet s'appuient sur les principes établis par Larman [10].

La figure 1.2 illustre la classification complète des différents types de diagrammes UML. Cette classification divise les diagrammes en deux grandes catégories : les diagrammes de structure (diagrammes statiques) et les diagrammes de comportement (diagrammes dynamiques). Les diagrammes de structure incluent notamment les diagrammes de classes, de composants, de déploiement et d'objets, tandis que les diagrammes de comportement comprennent les diagrammes de cas d'utilisation, de séquence, d'activité et de communication. Cette taxonomie nous guide dans le choix des diagrammes appropriés pour modéliser les différents aspects de notre système de facturation.

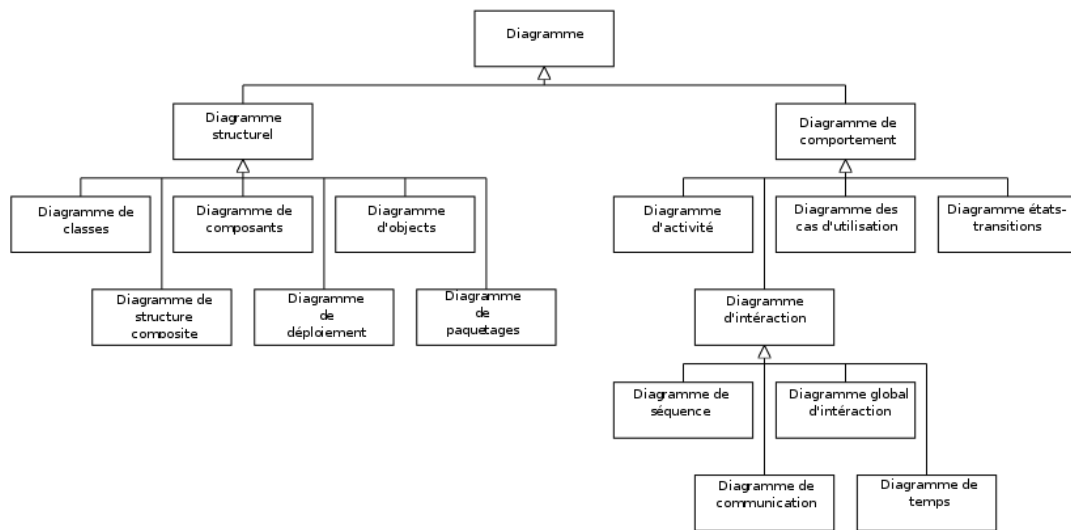


FIGURE 1.2 – Classification des différents types de diagrammes UML [?].

## 1.7 Conclusion

Ce chapitre a posé les bases de notre projet, spécifiant la vitalité et la nécessité de digitaliser le domaine de la facturation pour les PME et les administrations. Notre objectif est d'apporter une valeur ajoutée au travail des gestionnaires et de briser les chaînes qui limitent la croissance et la productivité.

Dans le chapitre suivant, nous allons effectuer la spécification des besoins fonctionnels et non fonctionnels, que nous modéliserons afin de mieux comprendre les attentes du client. L'identification des acteurs, ainsi que le plan de pilotage Scrum, nous aideront à structurer la méthode de travail des équipes de développement. Le but de ce chapitre est de structurer la méthode de conception et de délimiter notre champ d'action.

## 2.1 Introduction

Ce chapitre est consacré à la spécification et la modélisation des besoins, l'organisation du projet selon la méthodologie Scrum, la définition de l'environnement de travail et enfin, une première approche de l'architecture logicielle envisagée. Cette phase de préparation est essentielle pour garantir le bon déroulement du développement et la réussite du projet.

## 2.2 Spécification des besoins

La spécification des besoins est une étape fondamentale qui consiste à identifier et à documenter de manière précise les attentes des utilisateurs et les exigences du système. Ces besoins se divisent en deux catégories principales : les besoins fonctionnels et les besoins non fonctionnels.

### 2.2.1 Capture des besoins fonctionnels

Les besoins fonctionnels décrivent les fonctionnalités que le logiciel doit offrir aux utilisateurs. Ils répondent à la question : "Que doit faire le système?". Basés sur l'analyse des problématiques des PME et les objectifs du projet définis au chapitre précédent, les principaux besoins fonctionnels sont les suivants :

- **Gestion des Clients** : Créer, modifier et consulter des fiches clients. Enregistrer les informations essentielles des clients : nom, adresse, informations de contact (téléphone, email), informations de facturation, etc. Classer et rechercher facilement les clients. Conserver l'historique des achats et des interactions avec chaque client.

- **Gestion des Produits et Services** : Créer, modifier et consulter un catalogue de produits et services. Définir des informations détaillées pour chaque produit/service : nom, description, prix unitaire, taux de TVA, unité de mesure, code produit, etc. Classer et rechercher facilement les produits et services. Gérer les stocks pour les produits (quantité en stock, alertes de stock bas).
- **Création de Devis et Factures** : Créer des devis professionnels personnalisables. Transformer un devis en facture en un clic. Créer des factures professionnelles personnalisables avec des modèles dynamiques (intégration du logo de l'entreprise, informations légales, etc.). Générer automatiquement des numéros de facture séquentiels et conformes aux normes algériennes. Calculer automatiquement les montants totaux, la TVA et les remises. Gérer différentes devises. Permettre l'ajout de mentions légales et de conditions générales de vente personnalisées.
- **Envoi et Suivi des Factures** : Envoyer les factures directement par email aux clients au format PDF. Suivre le statut des factures (envoyée, consultée, payée, impayée). Envoyer des rappels de paiement automatiques par email pour les factures en retard.
- **Gestion des Paiements** : Enregistrer les paiements reçus (type de paiement, date, montant, facture concernée). Gérer les paiements partiels et les acomptes. Identifier rapidement les factures impayées et les retards de paiement. Générer des rapports sur les paiements reçus et en attente.

En plus de ces besoins fondamentaux, d'autres fonctionnalités avancées peuvent avoir un atout majeur pour les utilisateurs et compléter les besoins fondamentaux, notamment :

- **Analyse Financière et Rapports** : Fournir un tableau de bord personnalisable avec des indicateurs clés de performance (KPI) : chiffre d'affaires, montants dus, paiements en retard, etc. Générer des graphiques visuels pour l'analyse des recettes et des dépenses. Générer automatiquement des rapports de ventes périodiques (journaliers, hebdomadaires, mensuels, annuels). Permettre la personnalisation et l'exportation des rapports (formats PDF, CSV, etc.). Fournir des analyses de performance : produits/services les plus vendus, clients les plus rentables, etc.
- **Intégrations du E-paiement** : Intégration avec des passerelles de paiement en ligne populaires en Algérie (CIB et e-paiement d'Algérie Poste) pour faciliter les paiements en ligne avec l'outil Chargili Pay.

### 2.2.2 Capture des besoins non fonctionnels

Les besoins non fonctionnels définissent les qualités et les contraintes du système. Ils répondent à la question : "Comment le système doit-il être?". Ces besoins sont tout aussi importants que les besoins fonctionnels car ils influencent directement l'expérience utilisateur, la performance et la fiabilité du logiciel. Les principaux besoins non fonctionnels sont :

- **Conformité Légale et Réglementaire** : Le logiciel doit être conforme à la législation algérienne en matière de facturation et de comptabilité (décret exécutif n° 05-468 du 10 décembre 2005, lois fiscales, etc.). Il doit respecter les mentions obligatoires sur les factures, assurer la gestion de la TVA selon les règles algériennes, et proposer une numérotation des factures conforme.
- **Ergonomie et Facilité d’Utilisation** : L’interface utilisateur doit être intuitive, simple et facile à apprendre et à utiliser, même pour des utilisateurs non experts en informatique. Elle doit offrir une navigation claire et logique, un design épuré et professionnel. Le logiciel doit également s’adapter à différents types d’écrans pour une utilisation sur ordinateurs, tablettes et smartphones (même si ce n’est pas pertinent pour ce genre de logiciel, le marché algérien préfère d’avoir un accès rapide depuis un smartphone). Ces principes d’ergonomie et d’utilisabilité suivent les recommandations de Nielsen [11].
- **Performance** : Le logiciel doit être rapide et réactif, avec des temps de réponse courts pour toutes les actions (création de factures, recherche, génération de rapports, etc.). Le système doit supporter un nombre d’utilisateurs simultanés et un volume de données importants, adaptés aux besoins des PME.
- **Sécurité** : La sécurité des données est primordiale. Le système doit garantir la confidentialité, l’intégrité et la disponibilité des données des entreprises clientes. Il doit inclure des mécanismes d’authentification robustes (gestion des mots de passe, protection contre les attaques courantes) et une protection contre les accès non autorisés. Des sauvegardes régulières doivent être mises en place pour prévenir les pertes. Ces exigences de sécurité s’alignent sur les meilleures pratiques du génie logiciel moderne décrites par Sommerville [12].
- **Fiabilité et Disponibilité** : Le logiciel doit être fiable et fonctionner sans erreurs ou interruptions fréquentes. Il doit présenter une disponibilité maximale du service. Une gestion des erreurs et une reprise en cas de problème doivent être prévues.
- **Scalabilité** : Le système doit être conçu pour être scalable, c’est-à-dire capable de s’adapter à l’augmentation du nombre d’utilisateurs, du volume de données et des besoins futurs des PME.

## 2.3 Modélisation des besoins

La modélisation des besoins consiste à représenter graphiquement les interactions entre le système et son environnement. Cette approche permet de visualiser clairement les acteurs impliqués et leurs interactions avec le système, facilitant ainsi la compréhension des exigences fonctionnelles. Cette section présente les acteurs du système, le diagramme de contexte et le diagramme de cas d’utilisation global.

### 2.3.1 Identification des acteurs du système

Les acteurs sont des entités externes qui interagissent avec le système. Dans le cadre de notre application, trois acteurs principaux ont été identifiés :

- **Utilisateurs finals** : Ce sont les PME algériennes et leurs employés qui utiliseront l'application pour la gestion de leurs factures, devis, clients et paiements. Ces utilisateurs ont besoin d'une interface intuitive pour créer des devis, générer des factures, suivre les paiements et analyser leurs données financières.
- **Passerelle de paiement en ligne** : Il s'agit des services externes permettant l'intégration des solutions de paiement électronique, notamment CIB et e-paiement d'Algérie Poste. Cette passerelle communique avec notre système pour traiter les transactions financières et confirmer les paiements.
- **Service de messagerie** : Ce composant permet l'envoi automatique d'e-mails transactionnels, tels que les factures, les devis, les confirmations de paiement ou les relances clients. Il assure une communication fiable entre le système et les utilisateurs finaux.

### 2.3.2 Diagramme de contexte du système

Le diagramme de contexte définit les frontières du système et illustre les interactions de haut niveau entre le système et les acteurs externes. Il offre une vue d'ensemble simple mais complète des flux d'informations entre l'application et son environnement.

Comme illustré dans la Figure 2.1, notre système interagit avec les trois acteurs principaux à travers différents flux d'informations :

- Les **Utilisateurs finals** saisissent des données clients, produits et services, créent des devis et des factures, et reçoivent en retour des documents générés, des tableaux de bord et des analyses.
- La **Passerelle de paiement en ligne** reçoit des demandes de traitement de paiement et renvoie des confirmations de transaction et des statuts de paiement.
- Le **Service de messagerie** reçoit des événements système et envoie en retour des e-mails transactionnels tels que les factures, devis, confirmations ou relances clients.

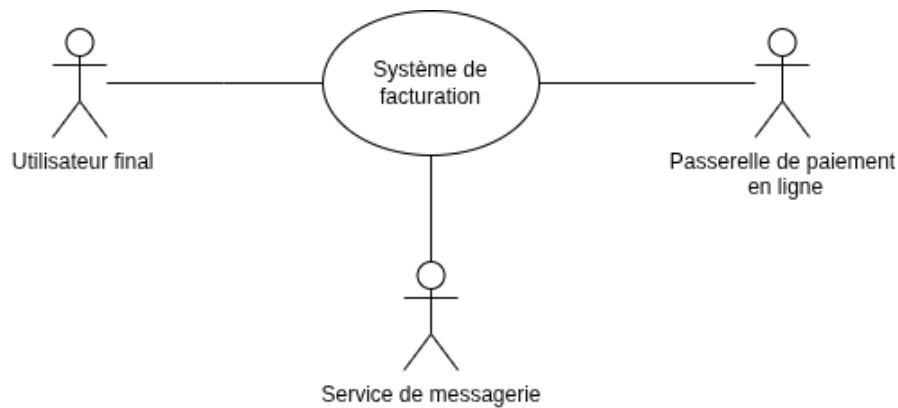


FIGURE 2.1 – Diagramme de contexte de l'application à réaliser

### 2.3.3 Diagramme de cas d'utilisation global

Le diagramme de cas d'utilisation global présente une vue d'ensemble des fonctionnalités offertes par le système et de leur relation avec les acteurs. Il constitue une représentation visuelle des besoins fonctionnels identifiés précédemment.

Le diagramme présenté dans la Figure 2.2 illustre les principales fonctionnalités du système et leurs relations avec les acteurs. Les cas d'utilisation sont organisés selon les thèmes identifiés dans le backlog du produit :

- **Gestion des clients, produits et services** : Permet aux utilisateurs de créer, modifier et consulter les informations relatives aux clients, produits et services.
- **Création et gestion de devis et factures** : Inclut la création de devis, leur transformation en factures, et l'envoi automatique des documents aux clients.
- **Gestion des paiements** : Comprend l'enregistrement des paiements, le suivi des statuts des factures et l'envoi de rappels automatiques.
- **Analyse financière et rapports** : Offre la génération de tableaux de bord, de rapports financiers et d'analyses de performance.
- **Intégration du E-paiement** : Permet le traitement des paiements en ligne via les passerelles de paiement.

La passerelle de paiement en ligne est impliquée dans les cas d'utilisation relatifs à l'intégration du E-paiement. Les utilisateurs finals, quant à eux, interagissent avec l'ensemble des fonctionnalités du système.



FIGURE 2.2 – Diagramme de cas d'utilisation global

## 2.4 Pilotage du projet avec Scrum

La méthodologie Scrum a été adoptée pour le pilotage de ce projet afin de garantir une approche itérative et flexible du développement. Cette méthode agile permet une meilleure adaptation aux changements d'exigences et favorise la collaboration entre les membres de l'équipe. L'adoption de Scrum facilite également la livraison régulière de fonctionnalités opérationnelles et assure une meilleure visibilité sur l'avancement du projet à travers des sprints de développement structurés. Cette approche itérative s'inspire des meilleures pratiques du développement logiciel décrites par Kroll et Kruchten [13].

### 2.4.1 Équipe et rôles

La réussite d'un projet Scrum repose sur une définition claire des rôles et responsabilités de chaque membre de l'équipe. Dans le cadre de ce projet, l'équipe a été organisée selon les rôles standards de Scrum, chacun ayant des responsabilités spécifiques qui contribuent à l'atteinte des objectifs du projet. Cette organisation, détaillée dans le tableau 2.1, permet d'assurer une coordination efficace et une répartition équilibrée des tâches tout au long du cycle de développement.

Rôle Scrum	Personne affectée	Tâches
Product Owner	MOHAMMEDI Mohamed	Il doit s'assurer de la bonne réalisation des fonctionnalités essentielles du produit.
Scrum Master	MOHAMMEDI Mohamed	Son rôle est de s'assurer que la méthodologie Scrum est bien appliquée dans le processus de développement de notre produit.
Équipe de développement	TOUAZI Sirem et ZAID Youcef	Développer les différentes fonctionnalités définies dans le backlog.

TABLE 2.1 – Tableau des rôles

### 2.4.2 Backlog du produit

Dans un premier temps, nous allons définir les différentes fonctionnalités à réaliser dans un laps de temps donné, ainsi que le temps de réalisation de chaque tâche, afin de permettre un travail efficace et coordonné au sein de notre équipe de développement. Le backlog détaillé est présenté dans le tableau 2.2.

TABLE 2.2 – Backlog du produit de l'application à réaliser.

<b>Product Backlog</b>				
<b>Thème</b>	<b>Description</b>	<b>Priorité</b>	<b>Risque</b>	<b>Estimation</b>
<b>Authentification</b>	L'utilisateur doit être capable de s'identifier et de se connecter à son compte via un identifiant et un mot de passe.	Haute	Haut	4JH
<b>Gestion des clients, des services et/ou des produits</b>	En tant qu'utilisateur, je dois être capable de voir les informations de mes clients ainsi que leurs différents achats (factures déjà distribuées).	Haute	Moyen	4JH
	En tant qu'utilisateur, je dois être capable d'ajouter ou de modifier ma liste de clientèles.	Haute	Moyen	4JH
<b>Création et gestion de devis et factures</b>	L'utilisateur doit être capable de transformer son devis en facture avec toutes les caractéristiques de cette dernière.	Haute	Moyen	6JH
	Le système de facturation doit générer automatiquement un numéro de facture et l'envoyer par mail au propriétaire.	Haute	Moyen	5JH
<b>Gestion des paiements</b>	L'utilisateur doit pouvoir accéder à un tableau de bord affichant toutes les factures ainsi que leur état (payée ou non payée).	Moyenne	Moyen	6JH
	L'utilisateur veut envoyer des rappels de paiement automatiquement par e-mail avec des paramètres prédéfinis.	Moyenne	Haut	5JH
<b>Analyse financière et rapports</b>	Un tableau de bord permettra de suivre les statistiques comme le nombre de factures distribuées et celles non payées.	Basse	Moyen	7JH
Suite sur la page suivante				

Table 2.2 – suite de la page précédente

Thème	Description	Priorité	Risque	Estimation
	Générer automatiquement des rapports de vente (quotidiens, hebdomadaires, mensuels, annuels) avec analyse de performance.	Basse	Haut	8JH
<b>Intégration du E-paiement</b>	L'utilisateur pourra utiliser différents moyens de paiement en ligne connus en Algérie, tels que la carte CIB et le e-paiement d'Algérie Poste.	Basse	Haut	10JH

## 2.5 Environnement de travail

Le développement du projet s'appuie sur un ensemble d'outils, de technologies et de plateformes sélectionnés pour garantir la productivité, la qualité du code, ainsi que la maintenabilité du système. L'environnement de travail retenu est le suivant :

## 2.6 Architecture du produit logiciel

Nous utilisons l'architecture MVC (Modèle-Vue-Contrôleur), une approche éprouvée pour les applications web qui maintient le code organisé en séparant les préoccupations. Cette architecture, dont les principes sont détaillés par Fowler [14], crée un logiciel plus clair et plus facile à maintenir, qui sera plus simple à mettre à jour sur le long terme. L'implémentation de cette architecture suit également les patterns de conception établis par Gamma et al. [15].

Le Modèle représente la couche de données et la logique métier de l'application. Il encapsule les structures de données, les règles de validation et les interactions avec la base de données. Ainsi, les entités telles que les clients, les factures et les produits sont modélisées à travers des objets du Modèle, chacun assurant la gestion de ses données associées. La conception de la base de données relationnelle suit les principes fondamentaux décrits par Elmasri et Navathe [16].

La Vue constitue l'interface utilisateur de l'application. Elle est chargée de présenter les données à l'utilisateur de manière conviviale et interactive. Les technologies de rendu telles que HTML, CSS et JavaScript sont employées pour la création des vues, lesquelles sont alimentées par les données fournies par le Contrôleur.

Le Contrôleur agit comme un intermédiaire entre le Modèle et la Vue. Il reçoit les requêtes de l'utilisateur, sollicite les données du Modèle et les transmet à la Vue pour affichage. Il assure ainsi la coordination des actions de l'utilisateur et la mise à jour des données de l'application.

L'adoption de l'architecture MVC offre plusieurs avantages significatifs. Elle favorise une séparation claire des responsabilités, facilitant la maintenance et la mise à jour du code. Elle permet la réutilisation des composants, réduisant ainsi les efforts de développement. Elle améliore la testabilité de l'application, chaque composant pouvant être testé indépendamment. Enfin, elle offre une base solide pour l'évolution future de l'application.

Dans le cadre de notre solution, le framework PHP Laravel, qui implémente le modèle MVC, sera utilisé pour faciliter le développement. Les entités de l'application seront modélisées à travers des modèles, les interactions avec l'utilisateur seront gérées par des contrôleurs, et les données seront présentées à travers des vues dynamiques utilisant le moteur de templates Blade de Laravel.

Cette approche architecturale permettra de construire une application robuste et évolutive. La figure 2.3 illustre l'architecture MVC (Modèle-Vue-Contrôleur) adoptée pour notre application de facturation. Cette représentation schématique montre clairement la séparation des responsabilités entre les trois couches : le Modèle qui gère les données et la logique métier, la Vue qui s'occupe de l'affichage et de l'interaction utilisateur, et le Contrôleur qui coordonne les échanges entre les deux autres composants. Cette architecture garantit une organisation claire du code, facilite la maintenance et permet une évolutivité optimale de notre solution SaaS.

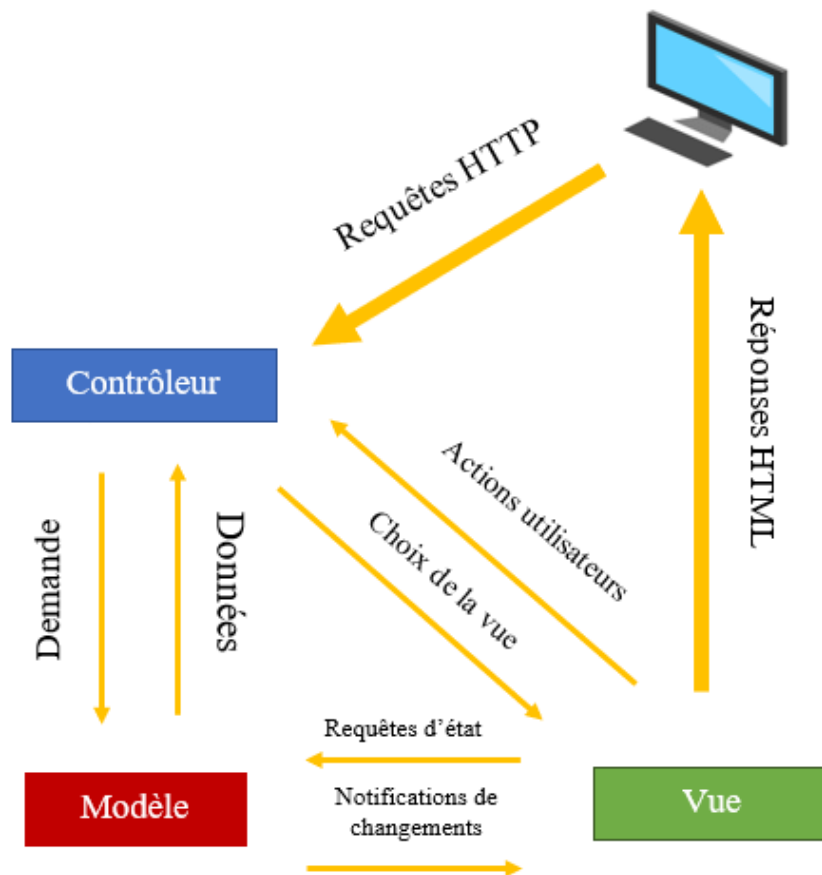


FIGURE 2.3 – Architecture logicielle MVC.

## 2.7 Conclusion

Dans ce chapitre, nous avons posé les bases solides du développement de notre projet, portant sur Conception et réalisation d'une Solution SaaS de Facturation destinée aux petites et moyennes entreprises (PME). De manière structurée, nous avons abordé le plan de conception à suivre pour les prochaines étapes. L'adoption de la méthodologie Scrum nous garantit une gestion agile et itérative du développement, favorisant la réactivité de l'équipe et l'amélioration continue du produit.

Le chapitre suivant portera sur la première release, constituée des sprints 1 et 2. Nous présenterons la conception du diagramme de cas d'utilisation de chaque sprint, suivie de sa description textuelle ainsi que de son diagramme de séquence. Chaque sprint se terminera par quelques captures d'écran de l'interface de notre application.

### 3.1 Introduction

Ce chapitre présente la concrétisation de notre projet de développement d'une solution SaaS (Software as a Service) de facturation pour les PME (Petite ou moyenne entreprise) algériennes. Le processus de développement a suivi une méthodologie agile Scrum, permettant une construction itérative et incrémentale de l'application. En partant d'une version initiale axée sur les fonctionnalités de base, le produit a été progressivement enrichi pour aboutir à une solution complète. Ce chapitre consolide les phases de conception et de réalisation, en se concentrant sur l'architecture, les fonctionnalités et l'interface utilisateur du produit final. L'objectif est de fournir une vue d'ensemble de l'application telle qu'elle a été livrée, en mettant en évidence ses capacités et sa structure technique.

### 3.2 Conception

La conception technique détaille la structure interne du système, y compris la structure de la base de données et la logique des interactions entre les différents composants.

#### 3.2.1 Diagramme de classes global

Le diagramme de classes de la figure 3.1 représente la structure statique du système. Il montre les principales classes de l'application et leurs relations, offrant une vue d'ensemble des entités métier et de la manière dont elles sont connectées.

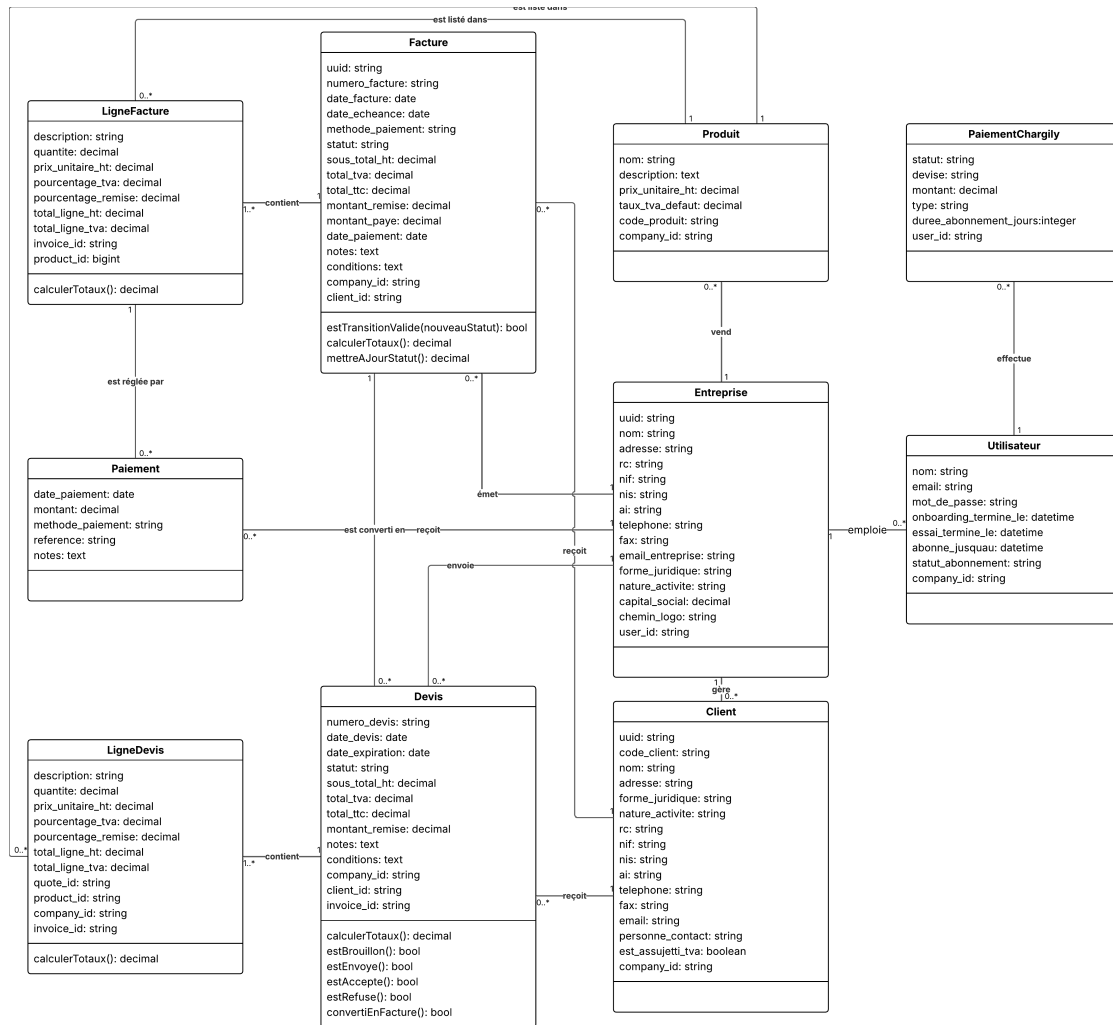


FIGURE 3.1 – Diagramme de classes global de l’application à réaliser.

### 3.2.2 Diagrammes de séquence clés

Les diagrammes de séquence illustrent les interactions entre les objets du système pour réaliser une fonctionnalité spécifique. Nous présentons ici quelques diagrammes pour les processus les plus importants.

#### Authentification de l’utilisateur

La figure 3.2 montre le processus d’authentification, depuis la saisie des identifiants par l’utilisateur jusqu’à la vérification par le système et l’accès à l’application.

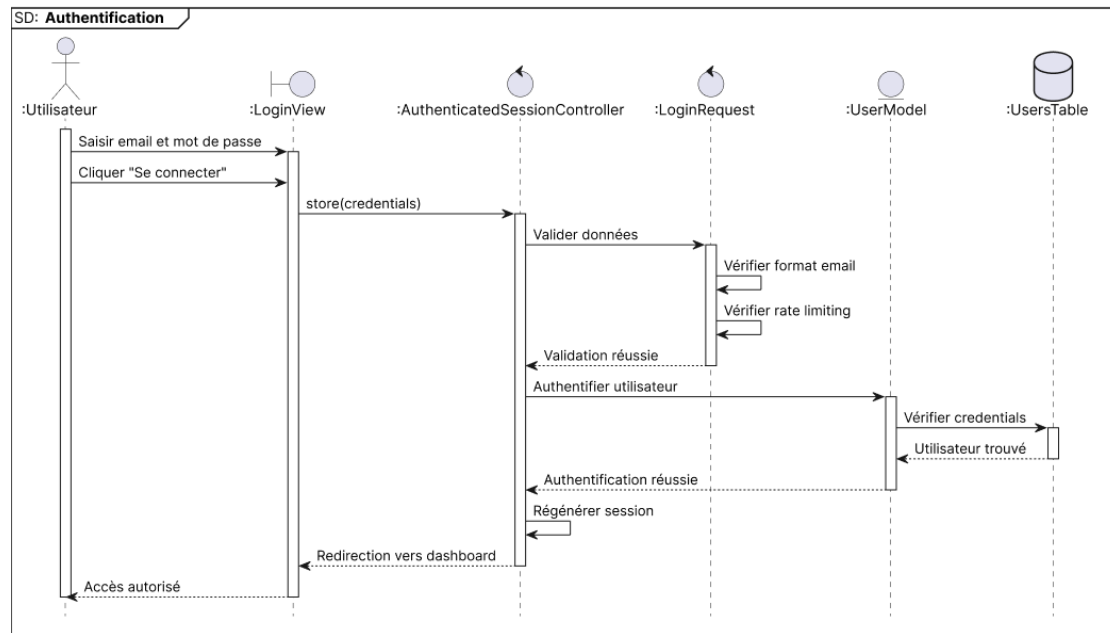


FIGURE 3.2 – Diagramme de séquence pour l'authentification.

### Création d'une facture

La création d'une facture, illustrée par la figure 3.3, est un processus central. Le diagramme montre comment l'interface utilisateur, le contrôleur et le modèle collaborent pour créer et stocker une nouvelle facture.

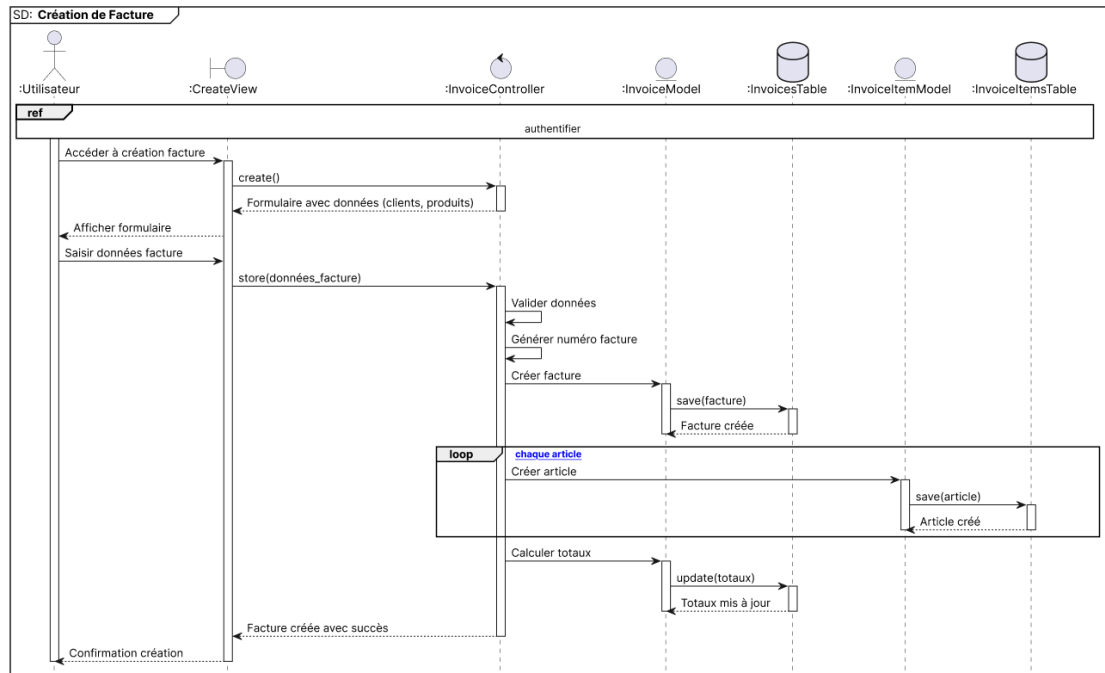


FIGURE 3.3 – Diagramme de séquence pour la création d’une facture.

### Paiement en ligne

La figure 3.4 détaille le processus de paiement en ligne pour un abonnement. Il met en évidence les interactions entre l'utilisateur, notre système et la passerelle de paiement externe.

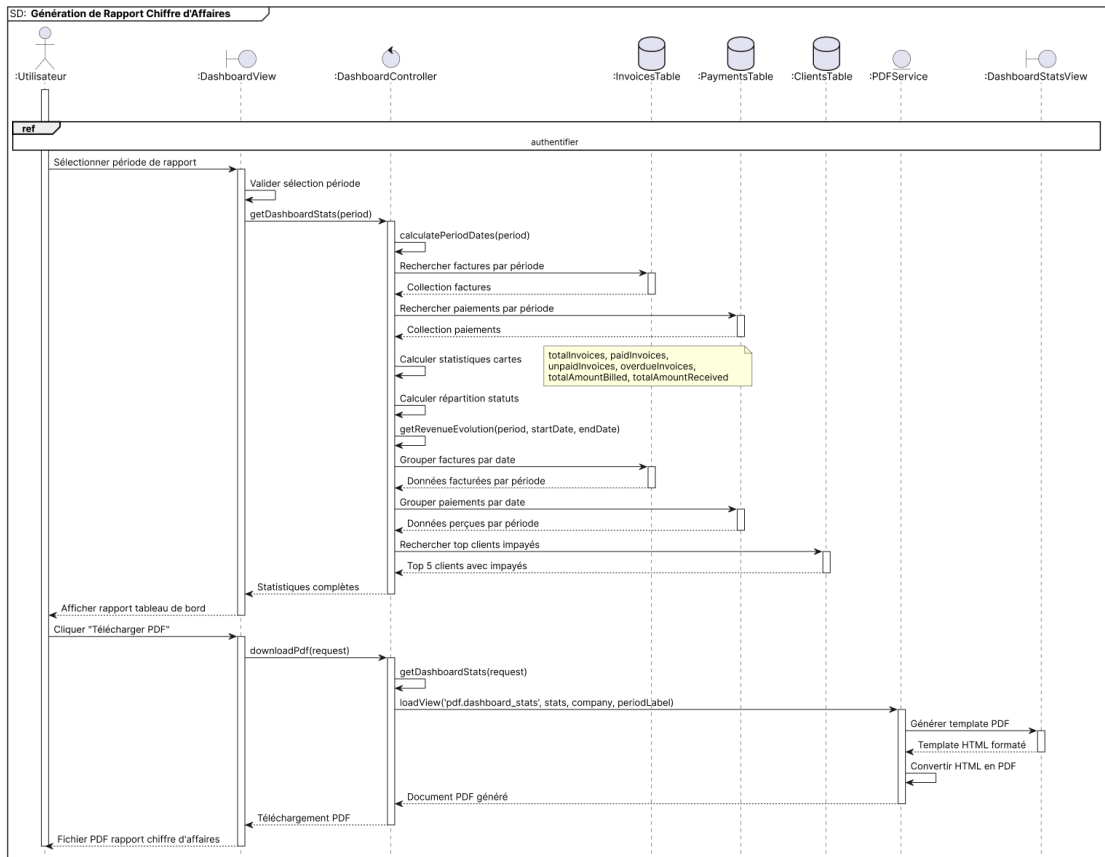


FIGURE 3.4 – Diagramme de séquence pour le paiement en ligne.

### 3.3 Interfaces utilisateur

Les interfaces utilisateur ont été conçues pour être intuitives et efficaces. Les captures d'écran suivantes (figures 3.5, 3.6 et 3.7) présentent quelques-unes des vues principales de l'application finale.

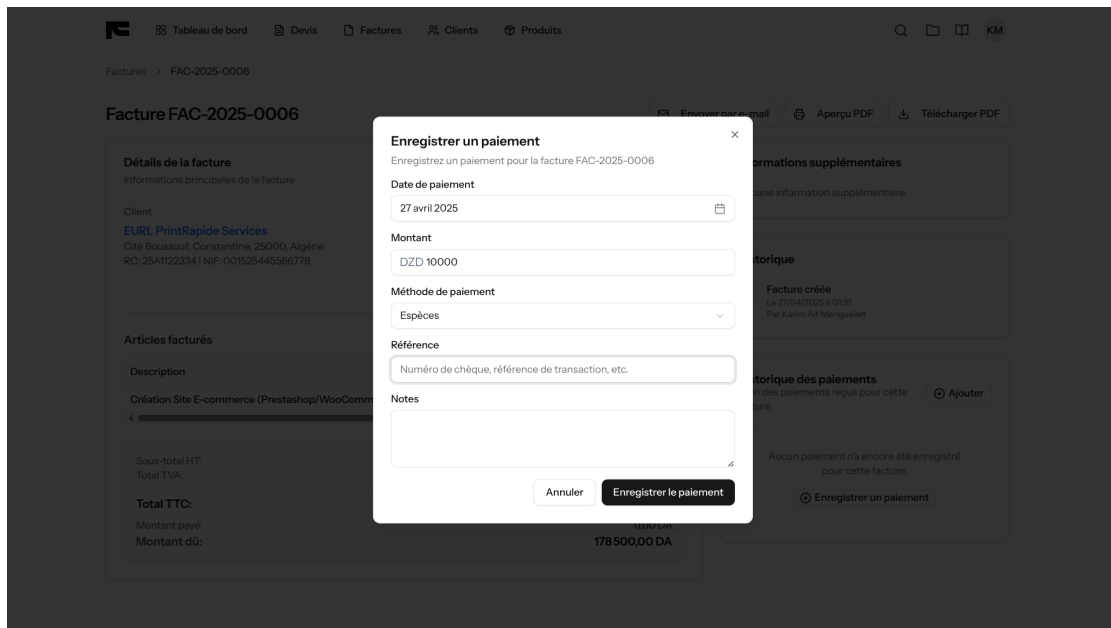


FIGURE 3.5 – Enregistrer un paiement.

Numéro	Client	Date	Montant TTC	Montant Payé	État	Actions
FAC-2025-0001	SARL El Bahia Import Export RC: 1689876543	27/04/2025	29750,00 DA	0,00 DA	Annulée	⋮
FAC-2025-0002	EURL PrintRapide Services RC: 25A1122334	27/04/2025	21420,00 DA	0,00 DA	Impayée	⋮
FAC-2025-0004	Freelancer Web Karim	27/04/2025	95200,00 DA	0,00 DA	Brouillon	⋮
FAC-2025-0005	SNC Frères Boulanger RC: 13D5566778	27/04/2025	142800,00 DA	142800,00 DA	Payée	⋮
FAC-2025-0006	SPA AgroFood Annaba RC: 23C9888776	27/04/2025	21420,00 DA	11420,00 DA	Part. Payée	⋮

FIGURE 3.6 – Liste des factures.

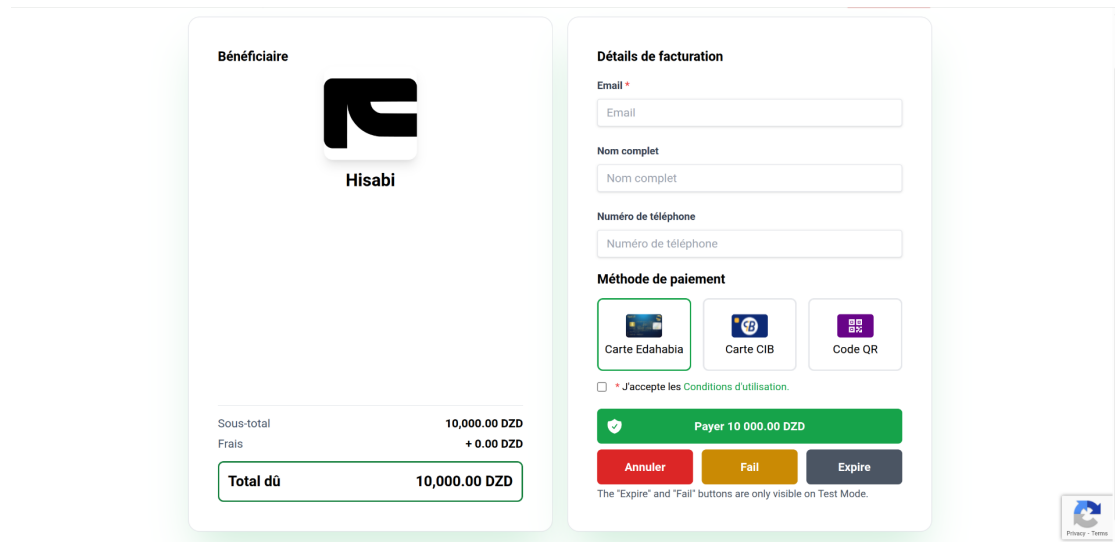


FIGURE 3.7 – Interface d’abonnement.

### 3.4 Conclusion

Ce chapitre a présenté une synthèse du travail d’implémentation de notre solution de facturation. En nous basant sur une architecture MVC (Modèle-Vue-Contrôleur) et une approche de développement agile, nous avons construit une application robuste et fonctionnelle. La conception fonctionnelle et technique a permis de mettre en place un système qui répond aux besoins identifiés des PME, avec des interfaces utilisateur soignées pour une expérience optimale.

## *CONCLUSION GÉNÉRALE ET PERSPECTIVES*

Au terme de ce projet, nous avons conçu et réalisé une solution SaaS (Software as a Service) de facturation complète, spécifiquement pensée pour répondre aux défis rencontrés par les Petites et Moyennes Entreprises en Algérie. Le développement itératif, mené selon la méthodologie Scrum, a permis de construire progressivement une application robuste, depuis les fonctionnalités fondamentales d'authentification et de gestion des données de base, jusqu'à des modules avancés incluant la gestion des paiements, le suivi détaillé des factures, les notifications automatiques, la génération de rapports financiers et l'intégration de systèmes de paiement électronique. Cette approche de développement respecte les principes du génie logiciel moderne décrits par Pressman et Maxim [17]. Le résultat est un outil intuitif et performant, qui vise à simplifier la gestion administrative, à améliorer le suivi financier et à assurer la conformité légale, offrant ainsi aux PME (Petite ou moyenne entreprise) un avantage compétitif significatif.

Les perspectives d'évolution de cette solution sont nombreuses et prometteuses. À court terme, l'accent pourrait être mis sur la collecte des retours d'utilisateurs suite à un déploiement pilote afin d'affiner les fonctionnalités existantes et d'identifier de nouveaux besoins. À moyen terme, des améliorations pourraient inclure l'intégration d'analyses prédictives plus poussées basées sur l'intelligence artificielle pour anticiper les comportements de paiement ou optimiser la gestion des stocks. Le développement d'une application mobile dédiée offrirait également une flexibilité accrue aux utilisateurs. Enfin, des intégrations avec d'autres outils de gestion d'entreprise, tels que des logiciels comptables ou des plateformes CRM (Customer Relationship Management), ainsi que l'adaptation continue aux évolutions réglementaires et technologiques, assureront la survie et la pertinence de la solution sur le long terme.

En somme, ce projet a constitué une véritable mise en situation professionnelle, en nous confrontant à toutes les étapes de la conception d'un produit numérique, de l'analyse des besoins jusqu'au déploiement fonctionnel. Il a également été l'occasion d'appréhender les contraintes réelles du marché local, souvent sous-estimées dans les approches purement théoriques. L'ex-

périence acquise nous a permis de développer un esprit critique, une rigueur technique et une meilleure capacité à travailler en équipe dans un cadre agile.

Pour conclure, ce travail ne représente pas une fin en soi, mais plutôt le point de départ d'une dynamique d'innovation plus large, ancrée dans les besoins réels du tissu économique algérien. Il ouvre la voie à des projets futurs plus ambitieux, porteurs de valeur et susceptibles de contribuer concrètement à la transformation numérique des PME.

## BIBLIOGRAPHIE

- [1] Peter Mell and Timothy Grance. *The NIST Definition of Cloud Computing*. NIST Special Publication 800-145, 2011. Référence pour les concepts de cloud computing et SaaS.
- [2] Ministère de l'Industrie Algérie. Le parc national des pme à la fin juin 2022, 2022. Consulté en juin 2025.
- [3] Journal officiel de la république algérienne démocratique et populaire, n° 80, décembre 2005. Consulté le 24 février 2025.
- [4] Ministère du Commerce Algérien. Bilan de contrôle économique et de la répression des fraudes durant la deuxième décade du mois de ramadhan 2024, 2024. Consulté le 24 février 2025.
- [5] Mike Cohn. *Agile Estimating and Planning*. Prentice Hall, 2005. Référence pour les méthodes agiles et la planification de projets.
- [6] [scrum.org/resources/what-scrum module](https://scrum.org/resources/what-scrum-module). What is scrum ?, 2025.
- [7] <https://www.bocasay.com.methode-scrum-benefices-developpements-web>, 2025.
- [8] Object Management Group (OMG). Unified modeling language (uml) resource page, 2025.
- [9] Alistair Cockburn. *Writing Effective Use Cases*. Addison-Wesley, 2001. Référence pour la modélisation des cas d'utilisation.
- [10] Craig Larman. *Applying UML and Patterns : An Introduction to Object-Oriented Analysis and Design and Iterative Development*. Prentice Hall, 3rd edition, 2005. Référence pour l'analyse et conception orientée objet avec UML.
- [11] Jakob Nielsen. *Designing Web Usability*. New Riders, 2000. Référence pour l'ergonomie et l'utilisabilité des interfaces web.
- [12] Ian Sommerville. *Software Engineering*. Pearson, 10th edition, 2016. Référence pour les concepts de génie logiciel et de développement d'applications.

## Bibliographie

---

- [13] Per Kroll and Philippe Kruchten. *The Rational Unified Process Made Easy : A Practitioner's Guide to the RUP*. Addison-Wesley, 2003. Référence pour les processus de développement logiciel itératif.
- [14] Martin Fowler. *Patterns of Enterprise Application Architecture*. Addison-Wesley, 2003. Référence pour l'architecture MVC et les patterns d'applications d'entreprise.
- [15] Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides. *Design Patterns : Elements of Reusable Object-Oriented Software*. Addison-Wesley, 1994. Référence fondamentale pour les patterns de conception logicielle.
- [16] Ramez Elmasri and Shamkant B. Navathe. *Fundamentals of Database Systems*. Pearson, 7th edition, 2017. Référence pour les concepts de bases de données relationnelles.
- [17] Roger S. Pressman and Bruce R. Maxim. *Software Engineering : A Practitioner's Approach*. McGraw-Hill, 8th edition, 2014. Référence pour les pratiques du génie logiciel.

## RÉSUMÉ

L'objectif principal de ce projet de fin de cycle était de développer une solution SaaS (Software as a Service) de facturation, performante, intuitive et spécifiquement adaptée aux réalités et exigences des petites et moyennes entreprises (PME) algériennes. Cette solution vise à combler les insuffisances des outils existants en offrant une plateforme centralisée pour la gestion des clients, des produits et services, la création rapide de devis et de factures conformes, le suivi rigoureux des paiements, et la génération de rapports financiers pertinents. L'ambition est de fournir un outil qui non seulement automatise les tâches, mais aide également les entrepreneurs à mieux piloter leur activité et à garantir leur conformité avec la réglementation fiscale algérienne. Le développement, mené selon la méthodologie Scrum, a abouti à une application robuste intégrant des fonctionnalités avancées telles que la gestion complète des paiements, le suivi des statuts de factures, les rappels automatiques, l'analyse financière et l'intégration de solutions de e-paiement. Dans le cadre de la conception technique, un soin particulier a été accordé à la modularité de l'architecture et à la maintenabilité du code. L'application repose sur le framework Laravel, reconnu pour sa sécurité et sa scalabilité, couplé à une base de données relationnelle performante permettant une gestion fiable des transactions. L'interface utilisateur, réalisée avec des technologies web modernes, garantit une expérience fluide, adaptée tant aux environnements de bureau qu'aux dispositifs mobiles.

**Mots-clés :** SaaS, Facturation, PME, e-paiement, Méthodologie Scrum, Transformation numérique.

## ABSTRACT

The main objective of this final project was to develop a high-performance, intuitive SaaS (Software as a Service) billing solution specifically adapted to the realities and requirements of Algerian small and medium enterprises (SMEs). This solution aims to address the shortcomings of existing tools by offering a centralized platform for client management, products and services, rapid creation of compliant quotes and invoices, rigorous payment tracking, and generation of relevant financial reports. The ambition is to provide a tool that not only automates tasks but also helps entrepreneurs better manage their business and ensure compliance with Algerian tax regulations. The development, conducted using Scrum methodology, resulted in a robust application integrating advanced features such as comprehensive payment management, invoice status tracking, automatic reminders, financial analysis, and e-payment solution integration. In the technical design framework, particular attention was given to architectural modularity and code maintainability. The application is built on the Laravel framework, recognized for its security and scalability, coupled with a high-performance relational database enabling reliable transaction management. The user interface, developed with modern web technologies, ensures a smooth experience adapted to both desktop environments and mobile devices.

**Keywords :** SaaS, Billing, SME, e-payment, Scrum Methodology, Digital Transformation.