



# POJET FIN D'ÉTUDES

*présenté à*

**La Faculté des Sciences de Sfax**

*en vue de l'obtention du*

**DIPLÔME DE LICENCE EN SCIENCE DE L'INFORMATIQUE**

**Option :  
Génie Logiciel**

*par*

**Nadia Dhifi et Ahmed Chakroun**

**Plateforme de Cabinet de Comptabilité "Progescode"  
Côte Front End et Back End**

Soutenu le 12/06/2024, devant la commission d'examen :

Mr. Boulbeba Ben Ammar  
Mme. Amira Ectiouï  
Mme. Sirine Rebai Siala  
Mr. Yassine Elleuche

Président  
Examineur  
Encadrant Académique  
Encadrant Industriel



---

# Dédicaces

**C'**est grâce à Dieu que tout a commencé,

Et c'est à lui que je rends grâce.

Le reste n'est que dédicaces.

**À ma chère mère et mon cher père.**

Je te dédie ce travail en témoignage de mon profond amour.

Que DIEU vous garde et vous procure d'une bonne santé et longue vie.

**À mon cher frère et mon chère soeur.**

Je leur souhaite une vie pleine de joie, de bonheur et de succès.

**À** vous tous,

Je dédie ce travail.

*Nadia Dhifi*



---

## Dédicaces

**J'**aimerais dédier ce travail :

**A mes chers parents « Mourad » et « Sameh ».**

Si je suis là aujourd'hui, c'est grâce à vous. Merci pour votre aide apportée à moi tout au long de mes études. J'espère que vous êtes heureux et fiers de moi, ce travail ce n'est que le résultat de vos sacrifices. Que dieu vous bénissant.

**A mon chère frère «Ayoub» .**

Je te souhaite un avenir rempli de bonheur, de réussite et de sérénité.

**A ma chère sœur «Mariem» .**

Je te retrouve toujours à mes côtés m'écouter, me soutenant et suggère. Que la vie ne te donne que du succès.

Je t' aime !

**À** vous tous,

Je dédie ce travail.

**Ahmed Chakroun**



---

# REMERCIEMENTS

Au terme de ce projet nous tenons à remercier infiniment tous les enseignants de la faculté des sciences Sfax.

nous tenons à remercier dans un premier lieu, l'ensemble du personnel de La société **PrestaCode** pour le bon accueil et les facilités accordées pour l'accomplissement de notre stage dans les meilleures conditions.

Nous nous'exprimons nos sincères remerciements à Mr **Salmen Ajili** le directeur de la société **Prestacode** et notre encadrant pour l'intérêt manifesté à ce travail ainsi que pour leurs précieux conseils et leur suggestions

Nous adressons nos vifs remerciements à notre encadrante au sein de la faculté **Mme Sirine Rebai Siala** qui a gardé un oeil attentif sur le déroulement du projet en donnant des remarques constructives. nous la remercions pour sa disponibilité et son aide précieuse et cela a été un plaisir de travailler sous sa directive.

Enfin,nous adressons nos remerciements pour **les membres du jury** qui nous'avoir honoré en acceptant d'évaluer ce travail.

Merci pour tous ceux qui nous ont aidés de près ou de loin pour la réussite de ce modeste travail.



---

# TABLE DES MATIÈRES

## LISTE DES FIGURES

## LISTE DES ABRÉVIATIONS

## INTRODUCTION GÉNÉRALE

<b>1</b>	<b>ETUDE DE PROJET</b>	<b>2</b>
1.1	INTRODUCTION . . . . .	3
1.1.1	Cadre du projet : . . . . .	3
1.1.2	Présentation de l'organisme d'accueil : . . . . .	3
1.1.2.1	Information générale : . . . . .	3
1.1.2.2	Domaine d'activité : . . . . .	4
1.2	Présentation du projet . . . . .	4
1.2.1	Contexte du projet : . . . . .	4
1.2.2	Description du projet : . . . . .	4
1.2.3	Problématique : . . . . .	6
1.3	Étude et critique de l'existence : . . . . .	6
1.3.1	Étude de l'existence : . . . . .	7
1.3.2	Le logiciel Sage 100 : . . . . .	7
1.3.3	Le logiciel SAP : . . . . .	7
1.3.4	Le logiciel Ciel : . . . . .	8
1.3.5	Critique de l'existence : . . . . .	8
1.3.6	Solution proposée : . . . . .	9
1.4	Méthodologie et modélisation du travail . . . . .	9
1.5	Cahier de charge . . . . .	13
1.5.1	les besoins fonctionnels : . . . . .	13
1.5.2	Les besoins non fonctionnels : . . . . .	14
1.6	Conclusion . . . . .	15
<b>2</b>	<b>ANALYSE ET CONCEPTION</b>	<b>16</b>

2.1	INTRODUCTION . . . . .	17
2.2	Choix de méthodologie UML : . . . . .	17
2.3	Analyse et conception : . . . . .	18
2.3.1	Identification des acteurs : . . . . .	18
2.4	Diagrammes des cas d'utilisation : . . . . .	18
2.4.1	Diagramme de cas d'utilisation global : . . . . .	18
2.4.2	Diagrammes de cas d'utilisation par acteur détaillés : . . . . .	19
2.4.2.1	Diagramme de cas d'utilisation pour un acteur « <i>Chef cabinet</i> » : . . . . .	19
2.4.2.2	Diagramme de cas d'utilisation pour un acteur « <i>assistante</i> » : . . . . .	24
2.5	Diagramme de classe : . . . . .	27
2.6	Diagrammes de séquence : . . . . .	28
2.6.1	Diagrammes de séquences « Authentification » . . . . .	28
2.6.2	Diagramme de séquence « Gérer cabinet » : . . . . .	28
2.6.3	Diagramme de séquence « Ajouter cabinet » : . . . . .	29
2.6.4	Diagramme de séquence « supprimer cabinet » : . . . . .	30
2.7	Diagramme de séquence « Gérer client » : . . . . .	31
2.7.1	Diagramme de séquence « Ajouter client » : . . . . .	32
2.7.2	Diagramme de séquence « Modifier client » : . . . . .	32
2.7.3	Diagramme de séquence « Supprimer client » : . . . . .	33
2.8	Diagramme de séquence « ajouter facture » : . . . . .	34
2.9	Diagramme de séquence « ajouter débours » : . . . . .	35
2.10	Conclusion : . . . . .	36
<b>3</b>	<b>Réalisation et Implémentation</b>	<b>37</b>
3.1	INTRODUCTION . . . . .	38
3.2	Environnement du travail : . . . . .	38
3.2.1	Environnement matériel : . . . . .	38
3.3	Environnement logiciel : . . . . .	39
3.3.1	Langage de programmation : . . . . .	39
3.3.2	Les logiciels de programmation : . . . . .	39
3.4	Interfaces : . . . . .	40
3.4.1	Interfaces du site web : . . . . .	40
3.4.1.1	Page d'authentification : . . . . .	40
3.4.1.2	Page d'accueil : . . . . .	41
3.4.1.3	Page de Cabinet : . . . . .	43

## TABLE DES MATIÈRES

---

3.4.1.4	Page de client : . . . . .	45
3.4.1.5	Page de Facture : . . . . .	47
3.4.1.6	Page de Débour : . . . . .	48
3.4.1.7	Page d'avantage : . . . . .	50
3.4.1.8	Page de régime : . . . . .	51
3.4.1.9	Page de service : . . . . .	52
3.4.1.10	Page de Banque : . . . . .	54
3.5	Conclusion : . . . . .	55
<b>CONCLUSION GÉNÉRALE</b>		<b>56</b>
<b>BIBLIOGRAPHIE</b>		<b>58</b>

# LISTE DES FIGURES

1.1	Les événements de la méthode Scrum [1]	12
2.1	Logo UML [2]	17
2.2	Diagramme de cas d'utilisation globale	19
2.3	Diagramme de cas d'utilisation « <i>Gérer cabinet, Gérer clients, Gérer sous-traitants</i> » :	20
2.4	Diagramme de cas d'utilisation « <i>Gérer factures et Gérer Débours</i> »	24
2.5	Diagramme de classe	27
2.6	Diagramme de séquence « <i>Authentification</i> »	28
2.7	Diagramme de séquence « <i>Gérer cabinet</i> »	29
2.8	Diagramme de séquence « <i>Ajouter cabinet</i> »	30
2.9	Diagramme de séquence « <i>supprimer cabinet</i> »	31
2.10	Diagramme de séquence « <i>ajouter client</i> »	32
2.11	Diagramme de séquence « <i>Modifier client</i> »	33
2.12	Diagramme de séquence « <i>Supprimer client</i> »	34
2.13	Diagramme de séquence « <i>ajouter factures</i> »	35
2.14	Diagramme de séquence « <i>ajouter débours</i> »	36
3.1	Page d'authentification	41
3.2	Page d'accueil	42
3.3	Liste des Tâches	43
3.4	Page de création d'un cabinet	44
3.5	Page de gestion de cabinet	45
3.6	Page de création de client	46
3.7	Page de gestion des clients	46
3.8	Page de création de facture	47
3.9	Page de gestion des factures	48
3.10	Page de gestion des débours	49
3.11	Page de modification débours	49
3.12	Page de création avantage	50



3.13 Page de gestion des avantages . . . . .	51
3.14 Page de création de régime . . . . .	51
3.15 Page de gestion des régimes . . . . .	52
3.16 Page de création des services . . . . .	53
3.17 Page de gestion des services . . . . .	53
3.18 Page de création de banque . . . . .	54
3.19 Page de gestion des banques . . . . .	55



---

# LISTE DES ABRÉVIATIONS

**AAA** Aaa Aaa Aaa

**BB** Bbb Bbb

**CCC** Ccc Ccc Ccc



---

# INTRODUCTION GÉNÉRALE

L'utilisation de technologies innovantes et d'outils de pointe, tant matériels que logiciels, a permis le développement de sites web modernes de plus en plus sophistiqués. Ces sites web, intégrés dans divers secteurs, offrent des fonctionnalités avancées qui facilitent les tâches des utilisateurs et améliorent leur expérience. Cependant, leur conception et leur maintenance efficaces nécessitent des compétences et une expertise accrues dans la gestion des technologies disponibles. Dans ce contexte, les cabinets de comptabilité, qui jouent un rôle crucial en offrant des services essentiels d'ordre financier, juridique et stratégique, doivent tirer parti de ces technologies pour optimiser leurs opérations et améliorer leurs services clients.

Pour obtenir notre diplôme de licence en Sciences Informatiques, spécialité Génie Logiciel, nous avons entrepris un stage de trois mois au sein de PRESTACODE, une start-up spécialisée dans le développement informatique. Notre mission est de concevoir et de développer un site web complet pour un cabinet de comptabilité, en couvrant à la fois les aspects front-end et back-end.

Notre rapport s'articule autour de trois chapitres principaux :

## ***Étude de projet :***

Présentation de l'organisme d'accueil, PRESTACODE, et description générale du projet. Nous mettrons le projet en contexte, en soulignant les besoins spécifiques d'un cabinet de comptabilité et les avantages qu'un site web moderne peut apporter.

### ***Analyse et conception :***

Analyse approfondie des besoins du cabinet de comptabilité et des implications pour la conception du site web. Nous présenterons également la conception UML élaborée pour répondre aux besoins identifiés.

### ***Réalisation et implémentation :***

Description de l'environnement de travail utilisé pour le développement du site web, les interfaces développées, ainsi que les défis rencontrés et les solutions apportées.

Enfin, une conclusion générale synthétisera notre travail et proposera des perspectives pour l'avenir.

---

# ETUDE DE PROJET

## Sommaire

---

<b>1.1 INTRODUCTION</b>	<b>3</b>
1.1.1 Cadre du projet :	3
1.1.2 Présentation de l'organisme d'accueil :	3
<b>1.2 Présentation du projet</b>	<b>4</b>
1.2.1 Contexte du projet :	4
1.2.2 Description du projet :	4
1.2.3 Problématique :	6
<b>1.3 Étude et critique de l'existence :</b>	<b>6</b>
1.3.1 Étude de l'existence :	7
1.3.2 Le logiciel Sage 100 :	7
1.3.3 Le logiciel SAP :	7
1.3.4 Le logiciel Ciel :	8
1.3.5 Critique de l'existence :	8
1.3.6 Solution proposée :	9
<b>1.4 Méthodologie et modélisation du travail</b>	<b>9</b>
<b>1.5 Cahier de charge</b>	<b>13</b>
1.5.1 les besoins fonctionnels :	13
1.5.2 Les besoins non fonctionnels :	14
<b>1.6 Conclusion</b>	<b>15</b>

---

## 1.1 INTRODUCTION

Ce chapitre a pour objectif de contextualiser le projet. Dans une première partie, nous présentons l'environnement dans lequel se déroule le stage, en fournissant une présentation de la société d'accueil ainsi que du projet à réaliser. Dans une seconde partie, nous offrons un aperçu général des concepts sous-jacents. Enfin, nous présentons la solution proposée, la démarche à suivre ainsi que les besoins fonctionnels et non fonctionnels détaillés.

### 1.1.1 Cadre du projet :

Ce projet s'inscrit dans le cadre de la réalisation d'un projet de fin d'études en vue d'obtenir le diplôme de Licence de Science Informatique, spécialité Génie Logiciel, au sein de la Faculté des Sciences de Sfax (FSS).

### 1.1.2 Présentation de l'organisme d'accueil :

La société "Prestacode" est une entreprise multi-services spécialisée dans le domaine informatique. Elle a pour ambition d'offrir un service personnalisé basé sur la disponibilité, la compétence et la confiance.

#### 1.1.2.1 Information générale :

- Nom/Raison Sociale : Prestacode.
- Adresse : Appartement 103, Immeuble Fourat, Rue Sidi Lakhmi 3027, Sfax, Tunisie.
- Adresse mail : [contact@prestacode.io](mailto:contact@prestacode.io).
- Téléphone : 28833242.
- Site Web : [www.prestacode.io](http://www.prestacode.io).
- Page Facebook : <https://www.facebook.com/PrestaCodeio>.
- Date de création : 2021.
- Directeur : Ajili Salmen.

### **1.1.2.2 Domaine d'activité :**

L'entreprise propose plusieurs services d'activités, notamment :

- Développement de logiciels.
- Développement web.
- Développement mobile
- Fourniture de ressources qualifiées pour les comptes clients en France et au Canada.

## **1.2 Présentation du projet**

Dans cette section, nous allons explorer en détail le projet de développement d'un site web pour un cabinet de comptabilité. Ce projet, vise à créer une plateforme numérique moderne et fonctionnelle qui répond aux besoins spécifiques des cabinets comptables. Nous aborderons les différents aspects du projet, en commençant par le contexte dans lequel il s'inscrit, suivi d'une description des objectifs et des fonctionnalités prévues.

### **1.2.1 Contexte du projet :**

Le projet de Plateforme de Progescode a été réalisé dans le cadre d'un projet de fin d'études en vue de l'obtention d'une Licence en Science Informatique, spécialité Génie Logiciel. Ce projet a été développé en partenariat avec la société PRESTACODE et la faculté de science (FSS).

### **1.2.2 Description du projet :**

La Plateforme Progescode a été développée dans le but de fournir des services de gestion comptable spécialisés, destinés aux cabinets comptables. Elle vise à faciliter la collaboration entre le chef de cabinet, les assistants et les sous-traitants, en leur permettant de gérer efficacement les tâches liées à la comptabilité.

Pour atteindre cet objectif, la plateforme comprend plusieurs modules spécialisés, tels que la gestion des dossiers clients, la saisie comptable, la génération des déclarations mensuelles.

Notre équipe est composée d'experts comptable, comptable, commissaire au compte, prêts à fournir des conseils personnalisés et des solutions adaptées aux besoins spécifiques de nos clients, afin de les aider à atteindre leurs objectifs de manière efficace et sécurisée.

La plateforme telle que décrite dans le cahier des charges, contiendra les modules suivants :

- Un module de recherche avancée pour simplifier la recherche d'informations.
- Un module de gestion de cabinet pour traiter les informations relatives au cabinet comptable.
- Un module de gestion des clients pour rationaliser la gestion des informations relatives aux clients.
- Un module de gestion des factures pour rationaliser la création, la modification et le suivi des factures émises par le cabinet.
- Un module de gestion des factures pour rationaliser la création, la modification et le suivi des débours émises par le cabinet.
- Un module de gestion des avantages pour administrer les avantages offerts aux clients.
- Un module de gestion des regimes pour administrer et personnaliser les différents régimes fiscaux et comptables adaptés aux besoins spécifiques des clients.
- Un module de gestion des services pour organiser les différentes prestations proposées par le cabinet, permettant de définir, modifier et offrir divers services comptables et financiers aux clients.
- Un module de gestion des banques pour centraliser les informations sur les banques partenaires du cabinet, offrant une vue d'ensemble organisée pour faciliter la gestion des relations bancaires.

La plateforme se divise en trois parties distinctes :

.Gestion des données clients et cabinets : Cette section permet de gérer les informations relatives aux clients et aux cabinets partenaires.



.Facturation, débours et gestion des factures : Cette partie offre des fonctionnalités pour gérer la facturation, les débours et le suivi des factures émises par le cabinet.

.Paramètres et configurations : Cette dernière section regroupe les paramètres liés aux avantages, services, régimes fiscaux et bancaires, permettant une personnalisation et une gestion efficace de ces éléments.

### **1.2.3 Problématique :**

Le projet de la Plateforme Prolegal soulève des questions importantes qui peuvent être formulées sous forme de problématique. Voici une reformulation de ces questions pour créer une problématique claire :

1. Comment concevoir une plateforme numérique efficace pour répondre aux besoins spécifiques d'un cabinet d'expertise comptable ?
2. Quels sont les défis techniques et organisationnels rencontrés lors du développement d'une plateforme de gestion complète pour un cabinet comptable ?
3. Comment intégrer des fonctionnalités avancées telles que la gestion des factures, des débours ..ect, dans une plateforme unique, tout en assurant une expérience utilisateur optimale ?
4. Quels sont les enjeux liés à la sécurité des données clients et à la conformité réglementaire dans le cadre d'une plateforme de gestion comptable en ligne ?
5. Comment optimiser l'efficacité opérationnelle et la productivité des cabinets comptables grâce à l'automatisation et à la centralisation des processus de gestion ?

## **1.3 Étude et critique de l'existence :**

Dans cette section, nous examinerons l'impact actuel de la Plateforme Progescode sur les processus de gestion des cabinets comptables et analyserons sa pertinence par rapport aux besoins du marché

### **1.3.1 Étude de l'existence :**

Avant de lancer un nouveau projet, il est crucial de réaliser une analyse approfondie des projets existants pour évaluer leurs atouts et appréhender le mode de fonctionnement des plateformes similaires. Dans cette perspective, nous examinerons le logiciel sage 100, le logiciel Ciel et le logiciel SAP

### **1.3.2 Le logiciel Sage 100 :**

Sage 100 est un logiciel de gestion d'entreprise conçu pour les petites et moyennes entreprises, offrant une intégration complète des différents modules, une personnalisation flexible et des outils d'analyse avancés. Parmi ses avantages, on trouve une gestion efficace des opérations commerciales, financières et comptables, ainsi que la possibilité de générer des rapports détaillés pour une prise de décision éclairée. Cependant, son coût initial élevé, sa complexité pour les utilisateurs novices et la dépendance au fournisseur pour les mises à jour et le support technique peuvent être des inconvénients pour certaines entreprises.

### **1.3.3 Le logiciel SAP :**

SAP est une suite de logiciels de gestion d'entreprise reconnue à l'échelle mondiale pour sa capacité à intégrer efficacement les processus métier, offrant ainsi une gestion complète des opérations commerciales, financières, des ressources humaines et de la chaîne d'approvisionnement. Ses avantages comprennent une intégration complète des processus métier, une analyse en temps réel pour une prise de décision rapide, et une flexibilité pour s'adapter aux besoins spécifiques de l'entreprise. Cependant, son coût initial élevé, sa complexité et sa dépendance au fournisseur pour les mises à jour et le support technique peuvent constituer des défis pour certaines entreprises.

### 1.3.4 Le logiciel Ciel :

Le logiciel CIEL est une solution de gestion comptable et commerciale largement utilisée par les petites et moyennes entreprises. Ses avantages incluent une interface conviviale, une grande souplesse d'utilisation et une adaptabilité aux besoins spécifiques de chaque entreprise. Grâce à ses fonctionnalités intégrées telles que la gestion des stocks, la facturation, la comptabilité et la gestion commerciale, CIEL permet aux entreprises de rationaliser leurs processus opérationnels. Cependant, son évolution parfois lente en termes de mises à jour et de nouvelles fonctionnalités, ainsi que son support technique parfois limité, peuvent constituer des inconvénients pour certaines entreprises, en particulier celles ayant des besoins plus complexes ou évolutifs.

### 1.3.5 Critique de l'existence :

**TABLE 1.1 – Tableau comparatif des sites**

Critères	Sage100	CIEL	SAP
Gestion des stocks	–	–	+
Facturation	–	–	+
Interface conviviale	+	–	–
Adaptabilité aux besoins	+	–	–
Évolution des mises à jour	+	+	–

### **1.3.6 Solution proposée :**

Progescode se positionne comme la solution optimale pour remédier aux lacunes constatées dans d'autres plateformes telles que CIEL, Sage 100 et SAP. En mettant l'accent sur une interface utilisateur conviviale, une souplesse d'utilisation et une adaptabilité aux besoins spécifiques de chaque entreprise, Progescode offre une réponse complète à une gamme variée de besoins opérationnels. En intégrant de manière transparente des fonctionnalités telles que la gestion des stocks, la facturation, la comptabilité et la gestion commerciale, Progescode élimine les obstacles rencontrés dans les autres solutions. Sa capacité à évoluer en parallèle avec les besoins évolutifs des entreprises, combinée à un support technique de qualité et des mises à jour régulières, en fait un choix stratégique pour les organisations cherchant à optimiser leur gestion opérationnelle de manière efficace et intégrée. En définitive, Progescode émerge comme un leader incontesté, offrant une solution professionnelle et complète pour les entreprises désireuses d'atteindre leurs objectifs avec succès.

## **1.4 Méthodologie et modélisation du travail**

Pour réaliser un projet, il est nécessaire de choisir une méthode de travail qui corresponde aux objectifs. Les méthodes traditionnelles déterminent les besoins du client dès le départ et ne les vérifient qu'à l'achèvement du projet. En revanche, les méthodes agiles permettent de faire évoluer les besoins tout au long du projet. Comme il est illustré dans Table 3.1, il existe de nombreuses méthodes agiles. Dans Table 3.1 nous décrivons quelques-unes pour découvrir l'approche la plus adaptée à notre travail.

**TABLE 1.2 – Tableau comparatif des méthodes agiles**

Méthodologie	Description	Avantages	Inconvénients
Kanban	<ul style="list-style-type: none"><li>- Impose un système à flux déclenché par la consommation du client</li><li>- Contrôle la visualisation du flux de travail avec l'utilisation d'un tableau</li><li>- Se base sur l'approche Lean</li></ul>	<ul style="list-style-type: none"><li>- Flexible</li><li>- Facilite la communication et la collaboration entre les équipes</li><li>- Améliore la productivité</li></ul>	<ul style="list-style-type: none"><li>- Peut causer des difficultés si les équipes ne sont pas bien formées aux règles de la méthode par étiquettes</li><li>- Ne supporte pas les demandes trop irrégulières</li><li>- Ne convient pas à tous les secteurs</li></ul>
ScrumBan	<ul style="list-style-type: none"><li>- Combine entre la méthode Scrum et Kanban - S'appuie sur un tableau visuel</li></ul>	<ul style="list-style-type: none"><li>- Permet d'effectuer des modifications légères au fil du temps</li></ul>	<ul style="list-style-type: none"><li>- S'exposer à la confusion en raison de l'absence de responsable.</li></ul>
Extreme programming (XP)	<ul style="list-style-type: none"><li>- Est une méthodologie de développement logiciel qui réponds aux besoins des clients</li></ul>	<ul style="list-style-type: none"><li>- La capacité d'apporter constamment des améliorations et des changements</li><li>- A développement itératif continu</li><li>- Code clair et simple</li></ul>	<ul style="list-style-type: none"><li>- Durée et coûts élevés</li><li>- Une forte participation du client</li></ul>
Scrum	<ul style="list-style-type: none"><li>- Se base sur une approche itérative et incrémentale</li></ul>	<ul style="list-style-type: none"><li>- Collaboration renforcée</li><li>- Efficacité dans l'atteinte des objectifs</li><li>- Agilité et flexibilité</li><li>- Participation du client</li></ul>	<ul style="list-style-type: none"><li>- Minimisation de documentation</li><li>- Dépendance aux rôles clés</li></ul>

### *Méthode de réalisation*

Suite à notre analyse de différentes méthodologies et à notre recherche d'une méthode appropriée et adaptée, nous avons décidé de nous appuyer sur la méthodologie Scrum, qui est la plus éprouvée. Comme il est montré dans Figure 1.1, Scrum est une approche de gestion de projet agile couramment employée dans le domaine du développement logiciel et d'autres secteurs. Elle est basée sur des principes et des valeurs fondamentales qui visent à transformer la façon dont les projets complexes sont traités et réalisés. La notion de flexibilité, d'adaptabilité et de collaboration est au cœur de cette approche. Elle est divisée par plusieurs équipes auto-organisées et pluridisciplinaires qui se compose de :

- Scrum Master : Le Scrum Master joue un rôle clé dans la progression du projet, garantissant le respect des principes et des valeurs du Scrum. Le coordinateur des équipes est responsable de garantir une communication optimale. Il augmente également la productivité et élimine les obstacles.

- Product Owner : Son rôle consiste à être le lien entre le client et l'équipe de développement. C'est lui qui donne les priorités aux fonctionnalités à développer ou à corriger, et qui accorde sa validation aux fonctionnalités achevées. À toutes les étapes du projet, ou sprint, il est responsable de présenter au client les travaux réalisés. Il étudie en collaboration avec l'équipe ses retours pour garantir que le produit ou le service développé répond parfaitement à ses attentes.

- Equipe de développement : Le produit est conçu par l'équipe de développement. Elle se structure de manière autonome, interdisciplinaire et collabore pour atteindre ses objectifs.

Scrum utilise trois principaux artefacts :

- Carnet de produits (Product Backlog) : La liste ordonnée des éléments de travail à réaliser, qui représente les exigences et les fonctionnalités attendues du produit.

- Carnet de sprint (Sprint Backlog) : La liste des éléments du carnet de produit sélectionnés pour être réalisés durant un sprint donné, avec un plan détaillé pour les réaliser.

- L'incrément produit : C'est la version améliorée et fonctionnelle du produit à la fin de chaque sprint.

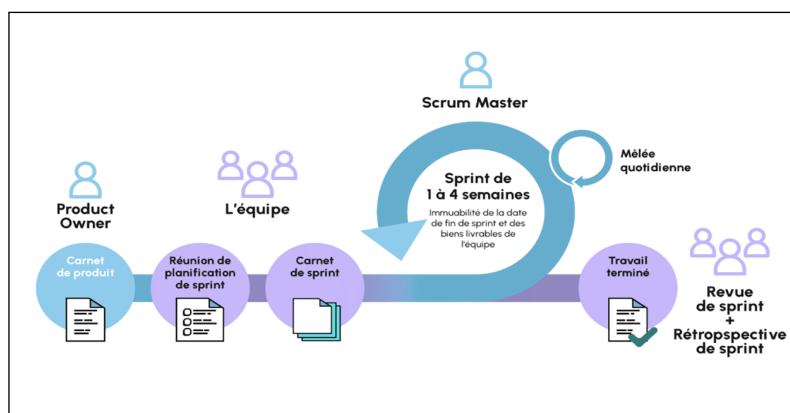
Scrum définit plusieurs événements clés qui sont :

- Planification du sprint : La planification permet de déterminer l'objectif du sprint. Et lors d'un sprint, l'équipe va travailler sur les tâches du backlog.

- Mêlée quotidienne : Une réunion quotidienne qui dure environ 15 minutes pour stimuler et motiver les équipes, mettre à jour les tâches de chacun, identifier les obstacles et mesurer leurs avancements.

- Revue de sprint : A la fin de chaque sprint, l'équipe de développement présente les fonctionnalités réalisées aux parties prenantes et collecte leurs feedbacks.

- Rétrospective de sprint : Une réunion au cours de laquelle l'équipe de développement ainsi que le Product Owner déterminent ce qui a bien fonctionné et discuter ce qu'il faut améliorer pour accroître l'efficacité du processus de travail.



**FIGURE 1.1 – Les événements de la méthode Scrum [1]**

## 1.5 Cahier de charge

Avant de commencer un projet, il est important de comprendre de manière précise les besoins des utilisateurs de l'application

### 1.5.1 les besoins fonctionnels :

Les besoins fonctionnels de notre système se décrivent comme suit :

#### *Au chef cabinet :*

Gestion de cabinet :

- S'authentifier en se connectant via un identifiant et un mot de passe.
- Ajouter un cabinet en remplissant le formulaire qui inclut les informations suivantes Nom, Prénom, Identifiant, Adresse, Matricule, CNSS ...
- Modifier ou supprimer une information dans fichier cabinet
- Gestion de client :
  - Ajouter un client en remplissant le formulaire qui inclut les informations des clients.
  - Modifier ou supprimer une information dans le formulaire du client

#### *A l'assistante :*

Gestion de facture :

- S'authentifier en se connectant via un identifiant et un mot de passe.
- Remplir le formulaire d'une facture d'un client.
- Modifier ou supprimer une information dans le formulaire facture.

Gestion débours :

- Remplir le tableau de débours des factures.
- Modifier ou supprimer une information dans tableau des débours.

Gestion avantages :

- Consulter la liste des avantages.
- Ajouter les avantages .
- Modifier ou supprimer une information dans le formulaire avantage.

Gestion regimes :



- Consulter la liste des regimes.
- Ajouter les regimes disponibles .
- Modifier ou supprimer une information dans le formulaire regime.

Gestion services :

- Consulter la liste des services.
- Ajouter les services disponibles .
- Modifier ou supprimer une information dans le formulaire service.

Gestion banques :

- Consulter la liste des banques.
- Ajouter les banques disponibles .
- Modifier ou supprimer une information dans le formulaire banque.

### **1.5.2 Les besoins non fonctionnels :**

Liés à l'application :

*Ergonomie :*

- Les interfaces doivent être agréables et faciles à utiliser afin d'aider les utilisateurs à formuler des requêtes correctes.
- Le logiciel doit avoir des interfaces homogènes et cohérentes.
- Le temps de réponse de l'application doit être rapide pour garantir de bonnes performances.

*Fiabilité :*

- Le système doit fournir des résultats précis et justes.

*Maintenance et réutilisabilité :*

- Le système doit être facilement modifiable pour s'adapter aux nouveaux besoins

*Compatibilité :*

- Le système doit être compatible avec différents navigateurs web et appareils, assurant ainsi une expérience utilisateur cohérente.

*Liés aux fichiers :*

### *Lisibilité ou intelligibilité :*

- Les fichiers lisibles implique que leur contenu soit clairement présenté et facilement compréhensible par les utilisateurs. Cela garantit une interprétation rapide et précise des informations, favorisant ainsi une prise de décision efficace et des actions appropriées.

### *Intégrité :*

- L'application doit garantir que les documents n'ont subi aucune modification et sont conformes aux fichiers originaux.

### *Pérennité :*

- Les fichiers doivent être archivés en utilisant des formats de conservation standardisés ou normalisés afin qu'ils puissent être relus pendant la période de conservation

## **1.6 Conclusion**

En conclusion, ce chapitre constitue le point de départ de notre projet, car il décrit l'objectif du stage, le fonctionnement de la plateforme Prolegal et la démarche à suivre pour présenter la spécification des besoins. Dans le chapitre suivant, nous allons présenter la phase de conception et d'analyse de notre projet.

---

# ANALYSE ET CONCEPTION

---

## Sommaire

---

<b>2.1</b>	<b>INTRODUCTION . . . . .</b>	<b>17</b>
<b>2.2</b>	<b>Choix de méthodologie UML : . . . . .</b>	<b>17</b>
<b>2.3</b>	<b>Analyse et conception : . . . . .</b>	<b>18</b>
2.3.1	Identification des acteurs : . . . . .	18
<b>2.4</b>	<b>Diagrammes des cas d'utilisation : . . . . .</b>	<b>18</b>
2.4.1	Diagramme de cas d'utilisation global : . . . . .	18
2.4.2	Diagrammes de cas d'utilisation par acteur détaillés : . . . . .	19
<b>2.5</b>	<b>Diagramme de classe : . . . . .</b>	<b>27</b>
<b>2.6</b>	<b>Diagrammes de séquence : . . . . .</b>	<b>28</b>
2.6.1	Diagrammes de séquences « Authentification » . . . . .	28
2.6.2	Diagramme de séquence « Gérer cabinet » : . . . . .	28
2.6.3	Diagramme de séquence « Ajouter cabinet » : . . . . .	29
2.6.4	Diagramme de séquence « supprimer cabinet » : . . . . .	30
<b>2.7</b>	<b>Diagramme de séquence « Gérer client » : . . . . .</b>	<b>31</b>
2.7.1	Diagramme de séquence « Ajouter client » : . . . . .	32
2.7.2	Diagramme de séquence « Modifier client » : . . . . .	32
2.7.3	Diagramme de séquence « Supprimer client » : . . . . .	33
<b>2.8</b>	<b>Diagramme de séquence « ajouter facture » : . . . . .</b>	<b>34</b>
<b>2.9</b>	<b>Diagramme de séquence « ajouter débours » : . . . . .</b>	<b>35</b>
<b>2.10</b>	<b>Conclusion : . . . . .</b>	<b>36</b>

---

## 2.1 INTRODUCTION

Après avoir mené une étude de l'existant et défini la conduite du projet en se référant à la démarche, il est temps de passer à la phase de conception. Cette phase permet à tous les membres de l'équipe de se mettre d'accord sur les idées, les processus et les résultats attendus. Dans le cycle de vie du projet, cette phase de préparation précède souvent l'élaboration du plan ou de la charte de projet. La phase de conception est consacrée à la création et la présentation des différents diagrammes UML qui permettront de mieux visualiser et comprendre le fonctionnement du système à développer.

## 2.2 Choix de méthodologie UML :

Le langage UML (Unified Modeling Language ou langage de modélisation unifié) est un langage graphique de représentation utilisé par les développeurs informatiques pour modéliser les objets, les états et les processus d'un logiciel ou d'un système. Les différents types de diagrammes intégrés dans UML permettent de visualiser les différents aspects d'un système de manière structurée et cohérente. Le langage de modélisation peut servir de modèle pour un projet et garantir une architecture d'information efficace. Il peut également aider les développeurs à présenter leur description d'un système de manière compréhensible pour les spécialistes internes et externes.



FIGURE 2.1 – Logo UML [2]

## **2.3 Analyse et conception :**

Dans ce qui suit, nous présentons la modélisation des principaux diagrammes UML qui décrivent la conception de notre projet.

### **2.3.1 Identification des acteurs :**

Un acteur représente une entité jouant un rôle défini par une ou plusieurs fonctions. Cette entité est externe au système, mais elle est en interaction avec celui-ci. Plusieurs acteurs peuvent intervenir au sein de la plateforme :

- Le chef de cabinet : le principal utilisateur et responsable du cabinet de comptabilité. Il a accès à toutes les fonctionnalités du site web de gestion.
- Les sous-traitants : sont des professionnels externes engagés par le cabinet pour des services spécialisés. Ils ont un accès limité au site web de gestion, spécifiquement aux parties pertinentes pour leurs missions
- L'assistante : est un membre du personnel du cabinet chargé de soutenir le chef de cabinet et les autres membres de l'équipe. Elle a un accès restreint au site web de gestion, centré sur les tâches administratives.

## **2.4 Diagrammes des cas d'utilisation :**

Le modèle des cas d'utilisation permet de décrire les besoins de l'utilisateur final de l'application.

### **2.4.1 Diagramme de cas d'utilisation global :**

Ce diagramme de cas d'utilisation préliminaire présente les différents acteurs impliqués dans notre application ainsi que les différents cas d'utilisations pour chacun des acteurs, comme illustré dans la figure suivante :

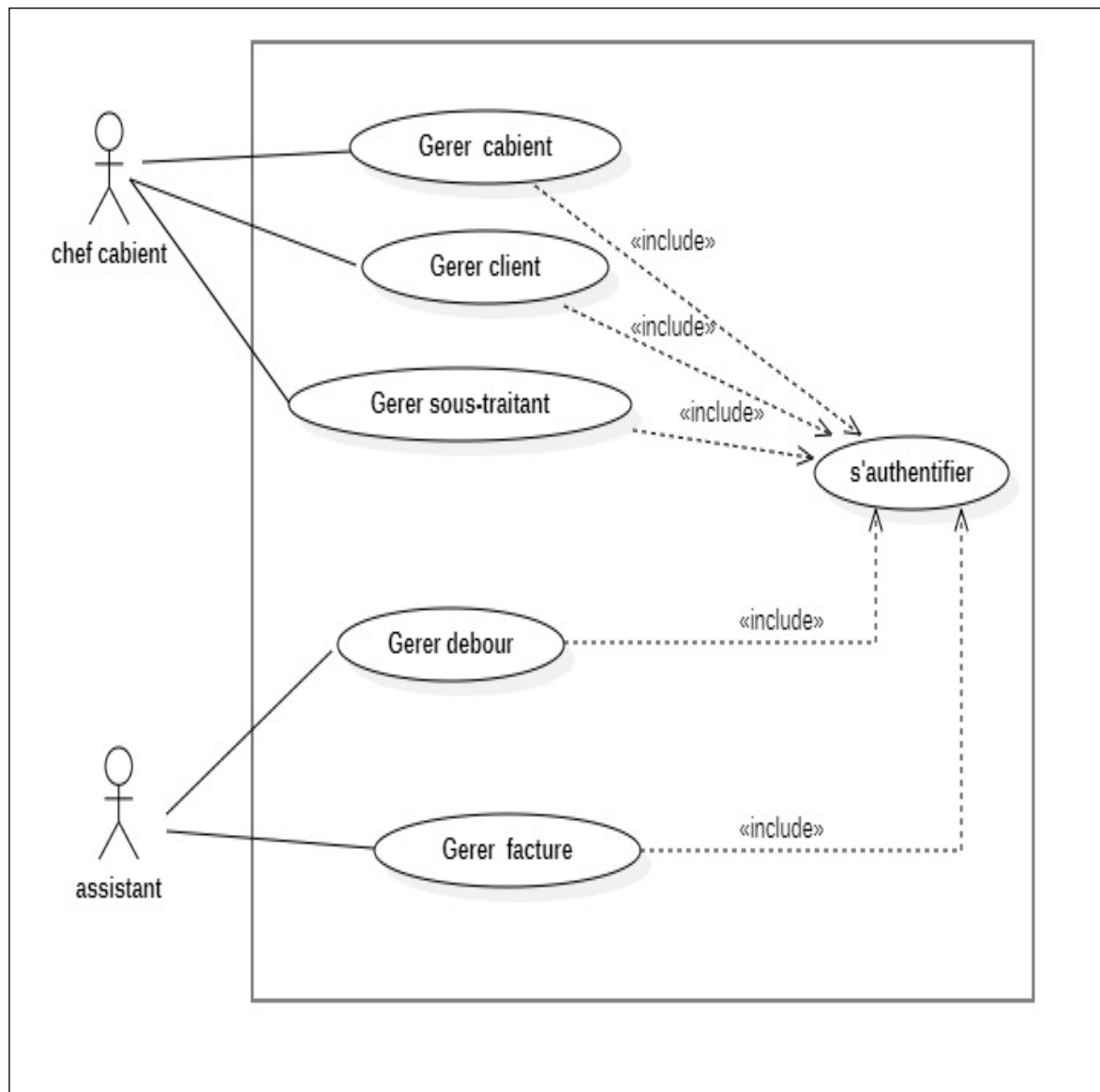


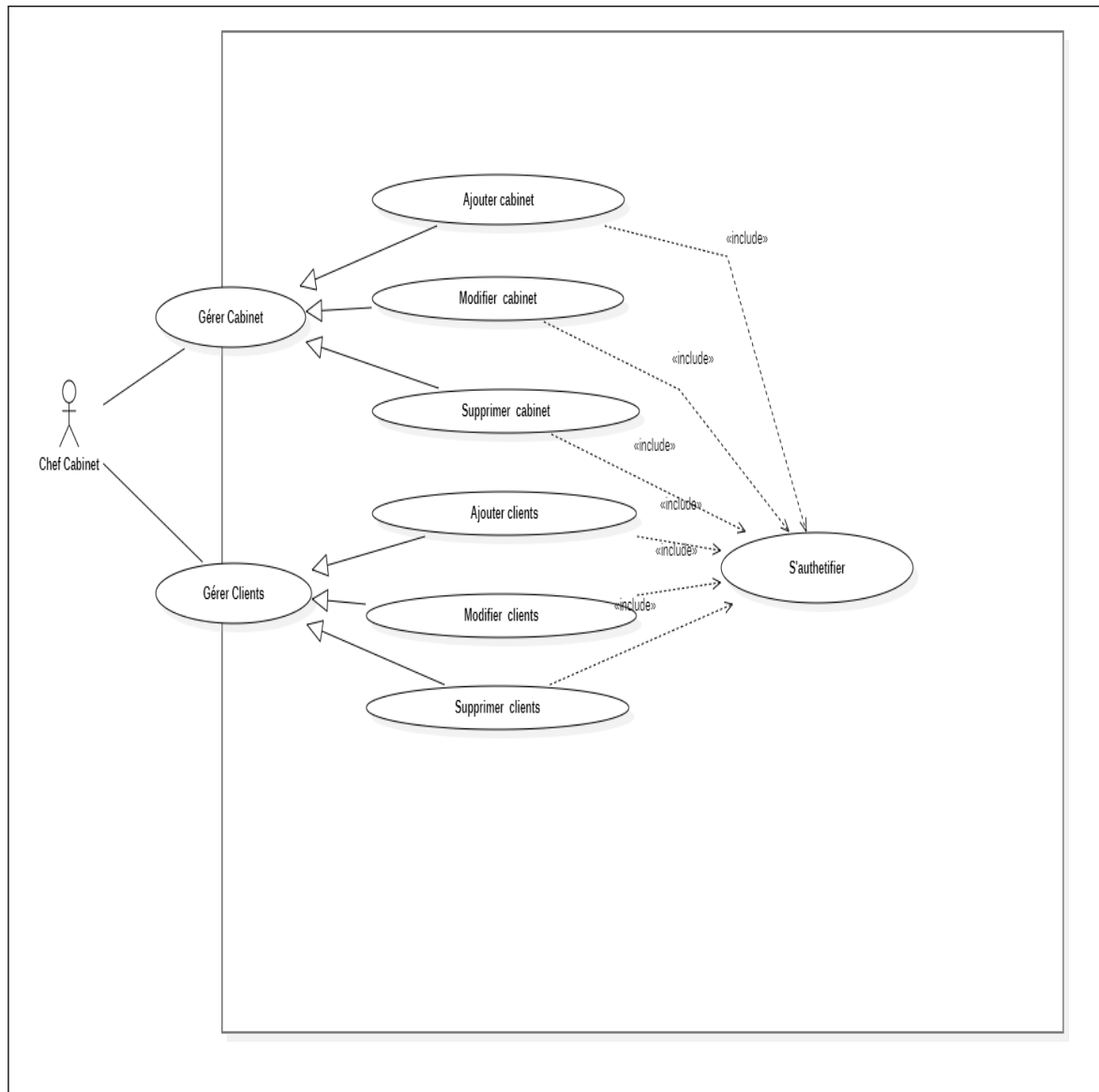
FIGURE 2.2 – Diagramme de cas d'utilisation globale

## 2.4.2 Diagrammes de cas d'utilisation par acteur détaillés :

### 2.4.2.1 Diagramme de cas d'utilisation pour un acteur « Chef cabinet » :

#### • Diagramme de cas d'utilisation :

Le chef de cabinet dispose de fonctionnalités essentielles telles que la gestion du cabinet, gestion du clients, gestion du sous-traitant. Le sous-traitants bénéficie également de fonctionnalités similaires, avec certaines restrictions, comme gestion des clients, ce qui montre l'illustration ci-dessous :



**FIGURE 2.3 – Diagramme de cas d'utilisation « Gérer cabinet, Gérer clients, Gérer sous-traitants » :**

**Description textuelle d'un cas d'utilisation :** • **Description textuelle d'un cas d'utilisation « Gérer cabinet » :**

TABLE 2.1 – Description textuelle d'un cas d'utilisation « *Gérer cabinet* »

Acteur :	Chef cabinet
Cas d'utilisation :	Ajouter Cabinet
Description :	-Permettre au chef de cabinet d'assurer une gestion efficace des activités, des ressources et des opérations au sein du cabinet de comptabilité.
Préconditions :	- Le système de gestion du cabinet de comptabilité est opérationnel. - . Le chef de cabinet dispose d'un accès autorisé au système avec des identifiants valides.
Postconditions :	- Les opérations du cabinet sont gérées de manière efficace et organisée. - Les informations et les ressources du cabinet sont mises à jour et précises.
Scénario principal :	<ol style="list-style-type: none"> <li>1. Le Chef de Cabinet s'authentifie au système de gestion de l'organisation à l'aide de ses identifiants.</li> <li>2. le Chef de Cabinet sélectionne l'option "Ajouter cabinet" dans le menu.</li> <li>3. Le système affiche une page ou un formulaire dédié à l'ajout d'un nouveau cabinet de comptabilité.</li> <li>4. Le Chef de Cabinet entre les informations nécessaires pour le nouveau cabinet, telles que le nom du cabinet, l'adresse, les coordonnées de contact, le domaine d'activité, etc.</li> <li>5. Après avoir rempli toutes les informations requises, le Chef de Cabinet soumet la demande en cliquant sur le bouton "Ajouter"</li> <li>6. Le système traite la demande et enregistre les détails du nouveau cabinet de comptabilité dans la base de données, ainsi que l'affectation des membres de l'équipe.</li> <li>7. Une notification de succès est affichée pour confirmer que le cabinet de comptabilité a été ajouté avec succès.</li> </ol>



Scénario d'exception :	<ul style="list-style-type: none"><li>• Si le système rencontre des problèmes techniques, le chef de cabinet peut contacter le support informatique pour résoudre les problèmes.</li><li>• Si des informations sensibles sont compromises, le chef de cabinet prend des mesures immédiates pour assurer la sécurité des données et informer les parties concernées.</li></ul>
------------------------	---

• **Description textuelle d'un cas d'utilisation « Gérer client » :**

**TABLE 2.2 – Description textuelle d'un cas d'utilisation « Gérer client »**

Acteur :	Chef de Cabinet, Sous-traitants
Cas d'utilisation :	Ajouter client
Description :	-Le chef de cabinet permet d'effectuer diverses actions liées à la gestion des clients du cabinet de comptabilité, y compris l'ajout, la modification, la consultation et la suppression des informations client.
Préconditions :	- Le système de gestion du cabinet de comptabilité est opérationnel. - . Le chef de cabinet dispose d'un accès autorisé au système avec des identifiants valides.
Postconditions :	-• Les informations des clients sont mises à jour dans le système. • Les interactions avec les clients sont enregistrées pour un suivi efficace. • Les modifications apportées aux clients sont sauvegardées correctement dans la base de données. .

Scénario principal :	<ol style="list-style-type: none"><li>1. Le Chef de Cabinet ou le sous-traitant s'authentifie au système de gestion de l'organisation à l'aide de ses identifiants.</li><li>2. Le Chef de Cabinet ou le sous-traitant sélectionne l'option "Ajouter client" dans le menu.</li><li>3. Le système affiche un formulaire dédié à l'ajout d'un nouveau client de comptabilité.</li><li>4. Le Chef de Cabinet ou le sous-traitant entre les informations nécessaires pour le nouveau client, telles que le nom, l'adresse, les coordonnées de contact, le domaine d'activité, le statut, etc.</li><li>5. Après avoir rempli toutes les informations requises, le Chef de Cabinet ou le sous-traitant soumet la demande en cliquant sur le bouton "Enregistrer".</li><li>6. Le système traite la demande et enregistre les détails du nouveau client de comptabilité dans la base de données.</li><li>7. Une notification de succès est affichée pour confirmer que le client de comptabilité a été ajouté avec succès.</li></ol>
Scénario d'exception :	<ul style="list-style-type: none"><li>• Si le chef de cabinet rencontre des problèmes techniques lors de l'accès au système, il contacte l'équipe de support informatique pour résoudre le problème.</li><li>• Si les informations fournies pour ajouter un nouveau client sont incomplètes ou incorrectes, le système affiche un message d'erreur et demande au chef de cabinet de les corriger.</li></ul>

### 2.4.2.2 Diagramme de cas d'utilisation pour un acteur « assistante » :

Diagramme de cas d'utilisation « Gérer factures et Gérer Débours» Le client bénéficie d'un large éventail de fonctionnalités pour gérer son profil et interagir avec le système. Il peut gérer les factures et gérer les débours , comme illustré dans la figure suivante :

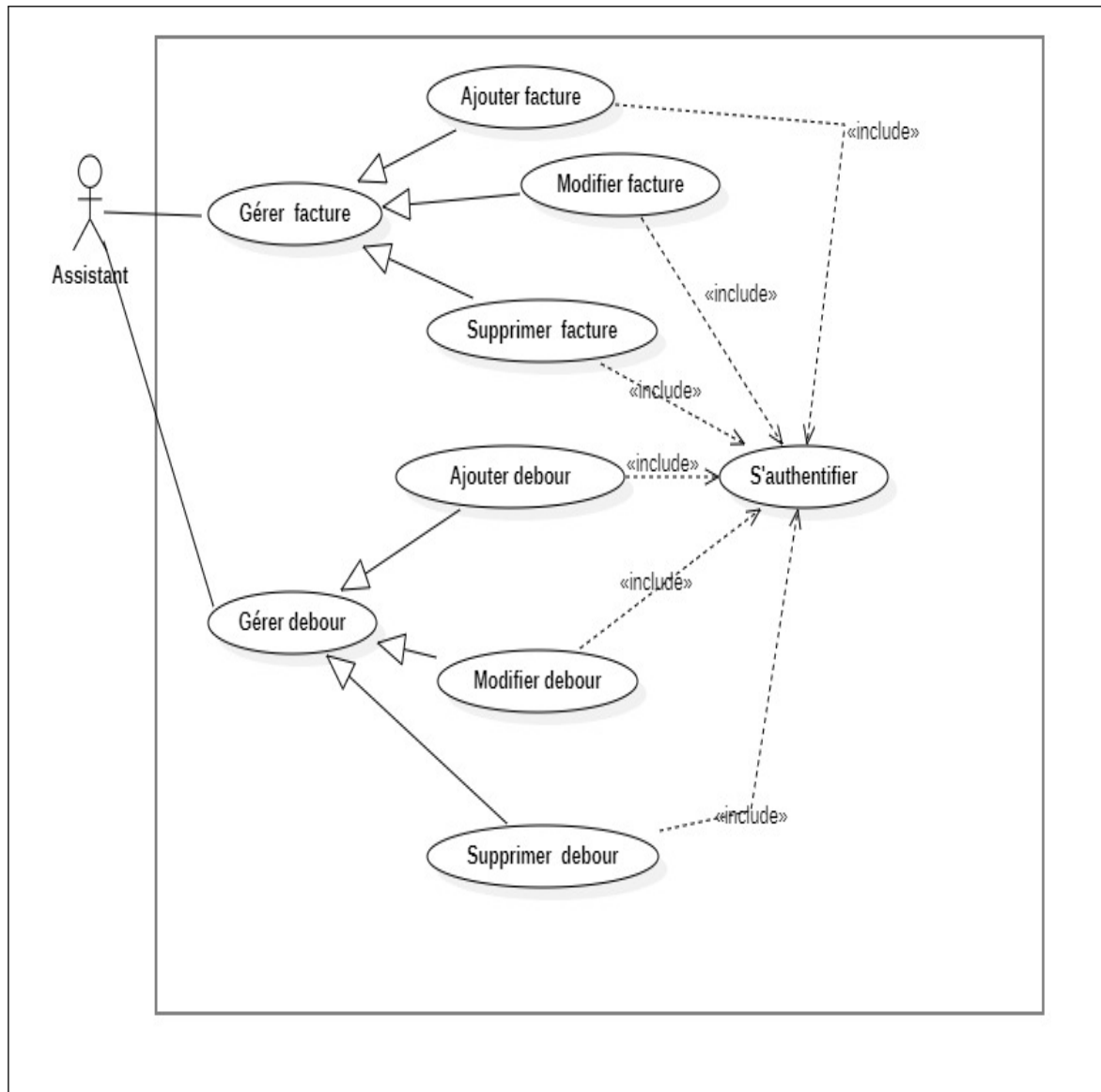


FIGURE 2.4 – Diagramme de cas d'utilisation « Gérer factures et Gérer Débours»

**TABLE 2.3 – Description textuelle d'un cas d'utilisation « Gérer facture »**

Acteur :	Assistante
Cas d'utilisation :	Créer d'une facture
Description :	- L'assistante crée une nouvelle facture pour un client.
Préconditions :	<ul style="list-style-type: none"><li>• L'assistante est connectée au système de gestion du cabinet de comptabilité.</li><li>• Le client pour lequel la facture est créée est déjà enregistré dans le système.</li></ul>
Postconditions :	<ul style="list-style-type: none"><li>• Une nouvelle facture est générée et enregistrée dans le système.</li></ul>
Scénario principal :	<ol style="list-style-type: none"><li>1. L'assistante s'authentifie au système de gestion du cabinet de comptabilité.</li><li>2. L'assistante accède à la fonction de création de facture.</li><li>3. L'assistante sélectionne le client pour lequel la facture est émise.</li><li>4. L'assistante saisit les détails de la facture, y compris les services rendus, les tarifs, les quantités et autres informations pertinentes.</li><li>5. L'assistante valide les informations saisies.</li><li>6. Le système génère automatiquement un numéro de facture et la date d'émission.</li><li>7. La facture est enregistrée dans le système.</li></ol>
Scénario d'exception :	<ul style="list-style-type: none"><li>• Si les informations fournies par l'assistante sont incomplètes ou incorrectes, le système affiche un message d'erreur et demande à l'utilisateur de corriger les informations.</li><li>• Si le système rencontre une erreur technique lors de la génération de la facture, un message d'erreur est affiché, et l'assistante est invitée à contacter le support technique pour résoudre le problème.</li></ul>

TABLE 2.4 – Description textuelle d'un cas d'utilisation « *Gérer débours* »

Acteur :	Assistante
Cas d'utilisation :	Modifier débours
Description :	- L'assistante modifie les débours associés à un client dans le système de gestion
Préconditions :	<ul style="list-style-type: none"> <li>• L'assistante est s'authenticée au système de gestion du cabinet de comptabilité. Le débours à modifier est déjà enregistré dans le système .</li> </ul>
Postconditions :	<ul style="list-style-type: none"> <li>• Les modifications apportées aux débours sont enregistrées dans le système.</li> </ul>
Scénario principal :	<ol style="list-style-type: none"> <li>1. L'assistante s'authentifie au système..</li> <li>2. Elle accède à la fonction de modification des débours..</li> <li>3. L'assistante recherche le client pour lequel les débours doivent être modifiés..</li> <li>4. Elle sélectionne les débours à modifier parmi la liste des débours associés au client.</li> <li>5. L'assistante effectue les modifications nécessaires, telles que la correction des montants, des descriptions ou d'autres détails.</li> <li>6. Elle valide les modifications apportées.</li> <li>7. Si l'assistante souhaite annuler en cours de traitement, il clique sur le bouton "Annuler" .</li> <li>8. Le système enregistre les modifications et met à jour les débours dans la base de données .</li> </ol>
Scénario d'exception :	<ul style="list-style-type: none"> <li>• Si les informations fournies par l'assistante ou sous-traitant sont incomplètes ou incorrectes, le système affiche un message d'erreur et demande à l'utilisateur de corriger les informations.</li> <li>• Si le système rencontre une erreur technique lors de la génération de la facture, un message d'erreur est affiché, et l'assistante ou sous-traitant est invitée à contacter le support technique pour résoudre le problème.</li> </ul>

## 2.5 Diagramme de classe :

Le modèle ou diagramme des classes d'UML, c'est le plus important des modèles d'UML. Il saisit la structure statique d'un système en montrant les objets dans le système, les relations entre les objets, les attributs et les opérations qui caractérisent chaque classe d'objets.

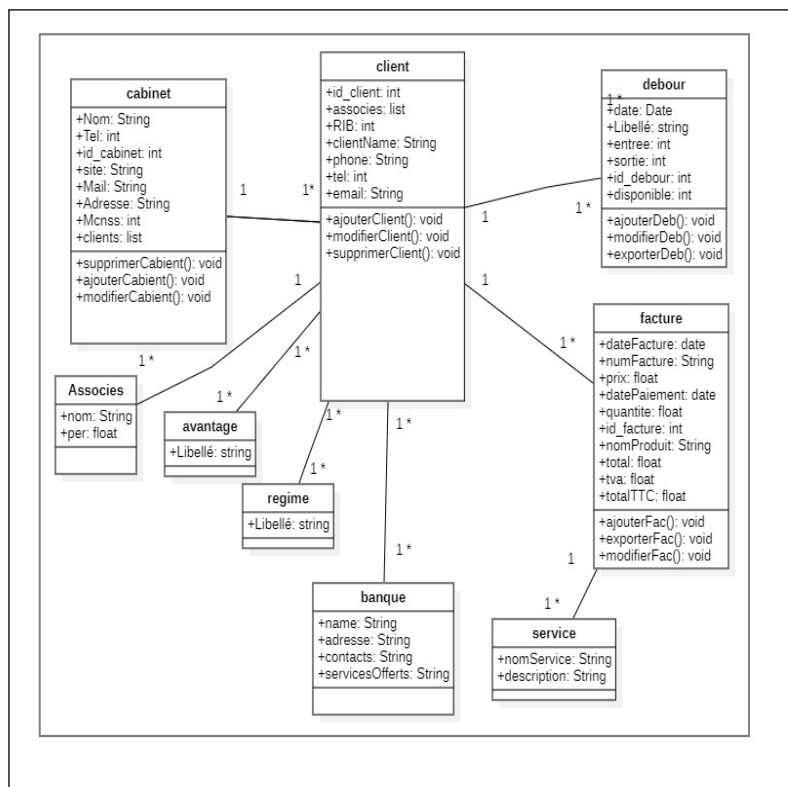


FIGURE 2.5 – Diagramme de classe

Le modèle physique des données peut être élaboré directement à partir du modèle conceptuel de données en suivant un certain nombre de règles prédéfinies :

- Une entité devient une table.
- Une propriété d'entité devient une colonne.
- L'identifiant d'une entité devient une clé primaire.
- Pour une association portant un lien identifiant (1, \*), la clé étrangère migrée fait partie de la clé primaire de la table dans laquelle elle migre.

## 2.6 Diagrammes de séquence :

Les diagrammes de séquences permettent de décrire comment les éléments du système interagissent entre eux et avec les acteurs. Ils présentent la coopération dans le temps entre les différents objets.

### 2.6.1 Diagrammes de séquences « Authentification »

Le diagramme de séquence "Authentification" illustre le processus de connexion dans l'application. Il met en évidence les interactions entre les acteurs et les composants impliqués dans cette authentification, comme présenté dans l'illustration ci-dessous.

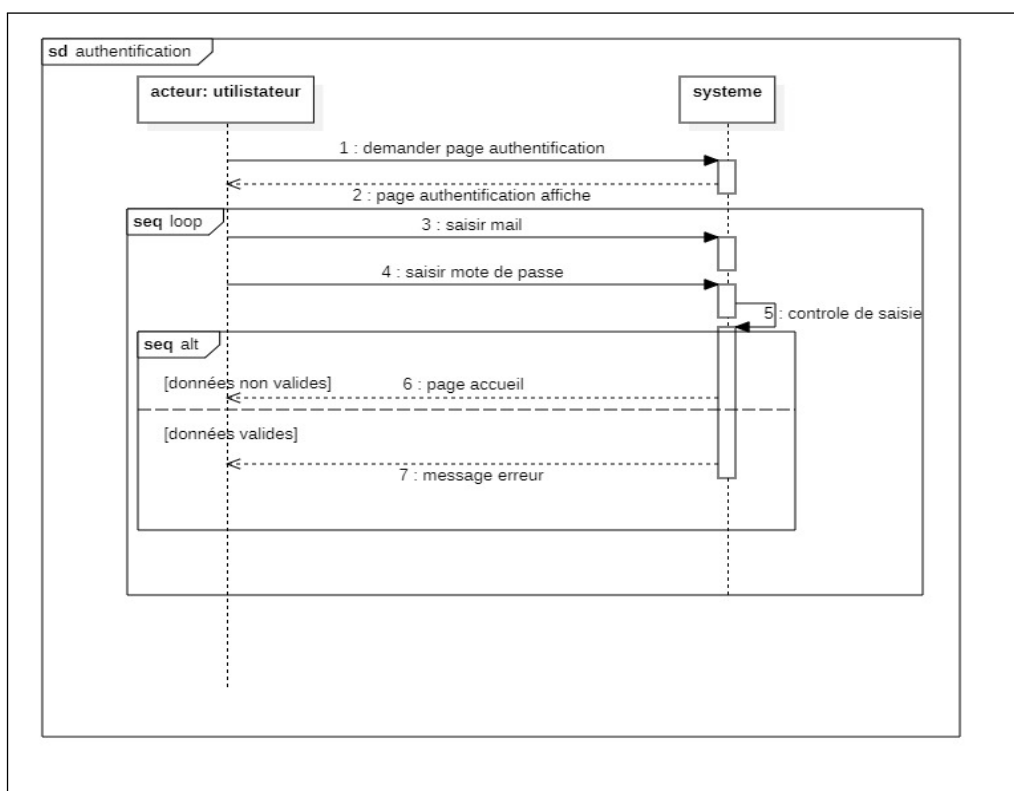


FIGURE 2.6 – Diagramme de séquence « Authentification »

### 2.6.2 Diagramme de séquence « Gérer cabinet » :

Le diagramme de séquence "Gérer cabinet" offre une vue détaillée et chronologique des interactions entre les différents éléments impliqués dans la gestion d'un cabinet, ce qui permet

de mieux comprendre le fonctionnement et le déroulement des processus liés à cette activité.,  
comme présenté dans l'illustration ci-dessous :

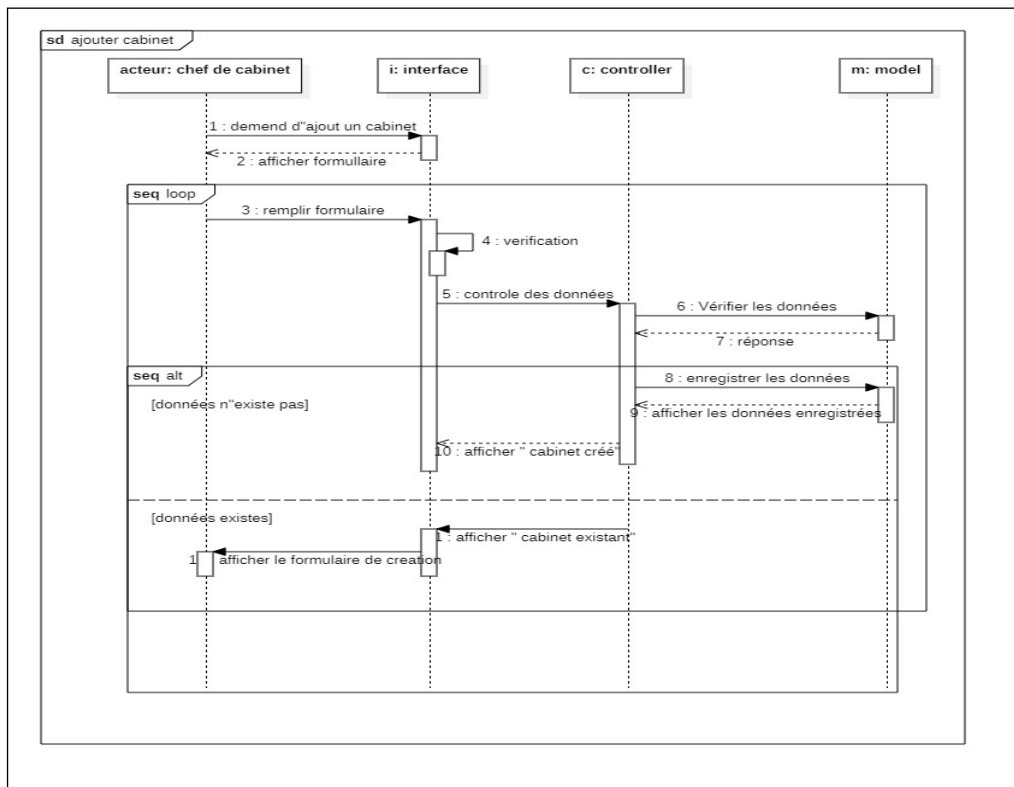
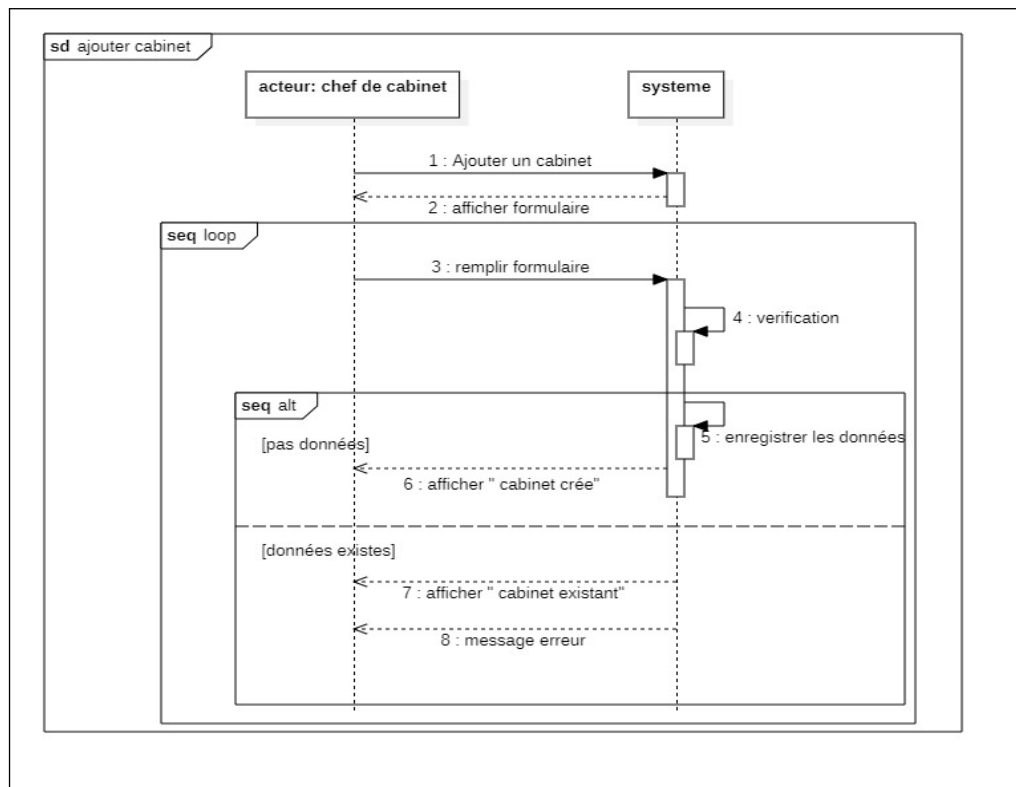


FIGURE 2.7 – Diagramme de séquence « *Gérer cabinet* »

### 2.6.3 Diagramme de séquence « Ajouter cabinet » :

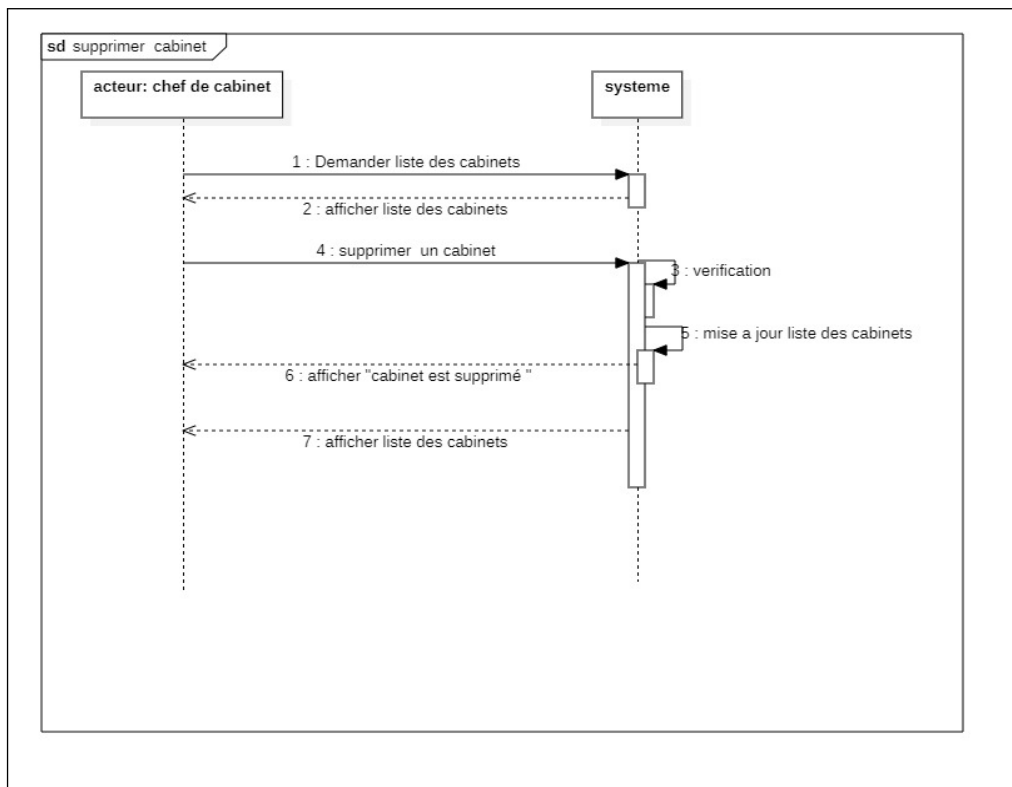
Le diagramme de séquence "Ajouter cabinet" illustre le processus d'ajout d'un nouveau cabinet dans l'application. Il met en évidence les interactions entre les acteurs et les composants impliqués dans cette opération, comme présenté dans l'illustration ci-dessous :



FIGURE 2.8 – Diagramme de séquence « *Ajouter cabinet* »

#### 2.6.4 Diagramme de séquence « supprimer cabinet » :

Le diagramme de séquence "Supprimer cabinet" illustre le processus de suppression d'un cabinet existant dans l'application. Il met en évidence les interactions entre les acteurs et les composants impliqués dans cette opération, comme présenté dans l'illustration ci-dessous :

FIGURE 2.9 – Diagramme de séquence « *supprimer cabinet* »

## 2.7 Diagramme de séquence « Gérer client » :

Le diagramme de séquence « Gérer client », Ajout, Modification, Suppression illustre de manière séquentielle les interactions entre les objets et composants impliqués dans la gestion des clients d'une entreprise. Il met en lumière les étapes nécessaires pour ajouter, modifier ou supprimer des informations client, en montrant la séquence des actions à effectuer, les messages échangés entre les objets, et les validations nécessaires. Ce diagramme permet de visualiser clairement le processus de gestion des clients, facilitant ainsi la compréhension du fonctionnement du système et pouvant servir de guide pour son développement ou son amélioration.

### 2.7.1 Diagramme de séquence « Ajouter client » :

Le diagramme de séquence "Ajouter client" illustre le processus d'ajout d'un client existant dans l'application. Il met en évidence les interactions entre les acteurs et les composants impliqués dans cette opération, comme présenté dans l'illustration ci-dessous :

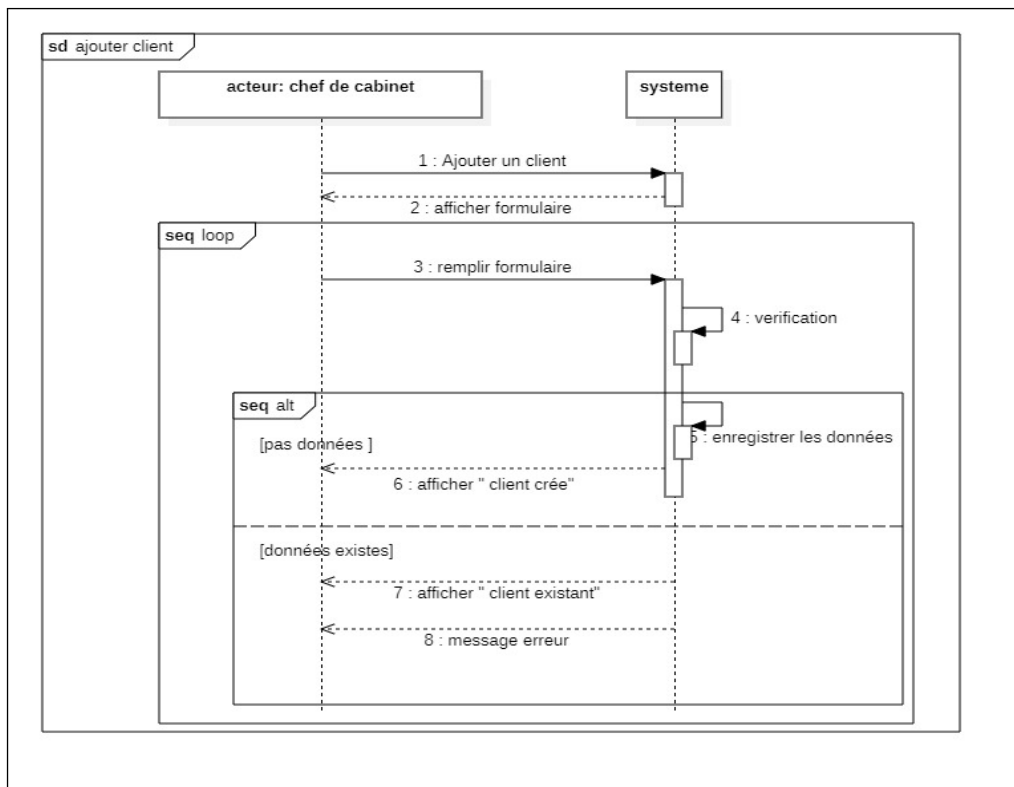
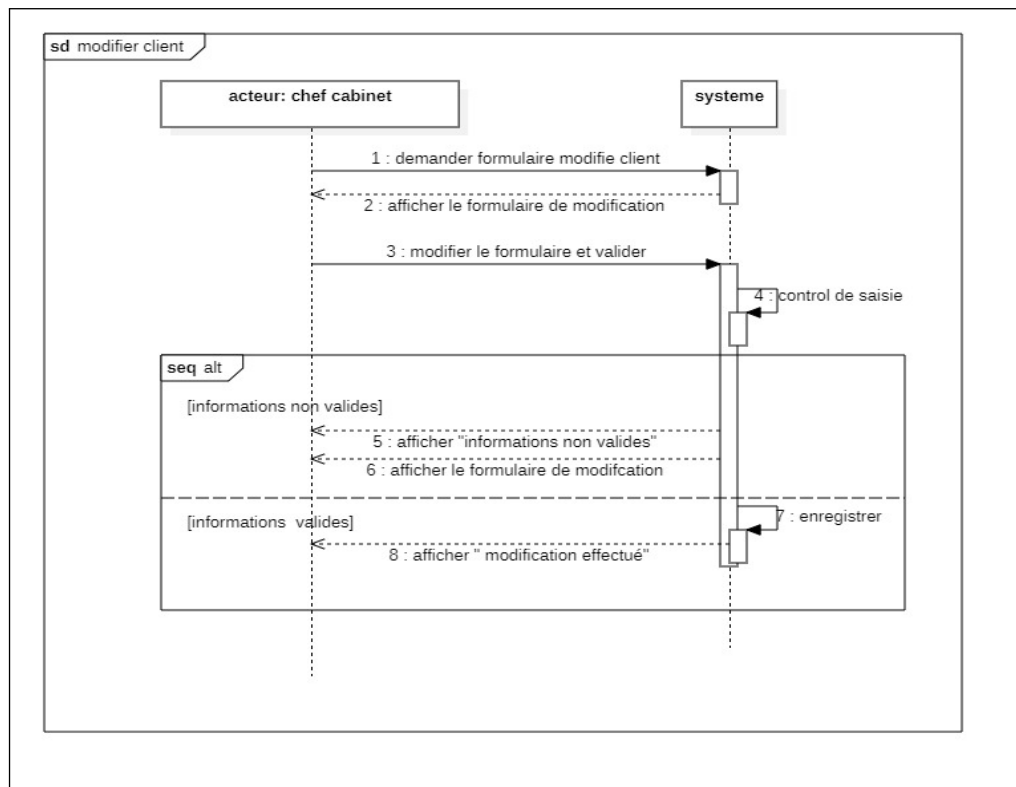


FIGURE 2.10 – Diagramme de séquence « ajouter client »

### 2.7.2 Diagramme de séquence « Modifier client » :

Le diagramme de séquence "Modifier client" illustre le processus de modification d'un client existant dans l'application. Il met en évidence les interactions entre les acteurs et les composants impliqués dans cette opération, comme présenté dans l'illustration ci-dessous :

FIGURE 2.11 – Diagramme de séquence «*Modifier client*»

### 2.7.3 Diagramme de séquence « Supprimer client » :

Le diagramme de séquence "Supprimer client" illustre le processus de suppression d'un client existant dans l'application. Il met en évidence les interactions entre les acteurs et les composants impliqués dans cette opération, comme présenté dans l'illustration ci-dessous :

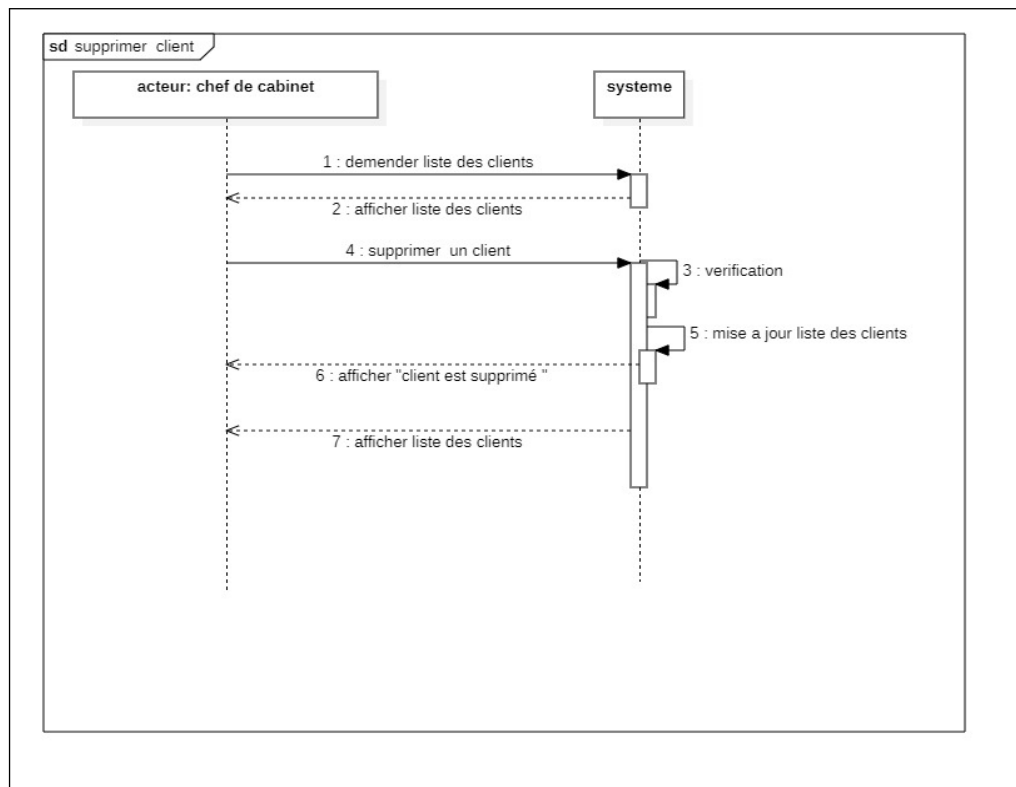


FIGURE 2.12 – Diagramme de séquence «*Supprimer client* »

## 2.8 Diagramme de séquence « ajouter facture » :

Le diagramme de séquence "ajouter factures" représente le processus de gestion des partenaires dans une application. Il met en évidence les interactions entre les acteurs et les composants impliqués dans cette gestion, comme présenté dans l'illustration ci-dessous :

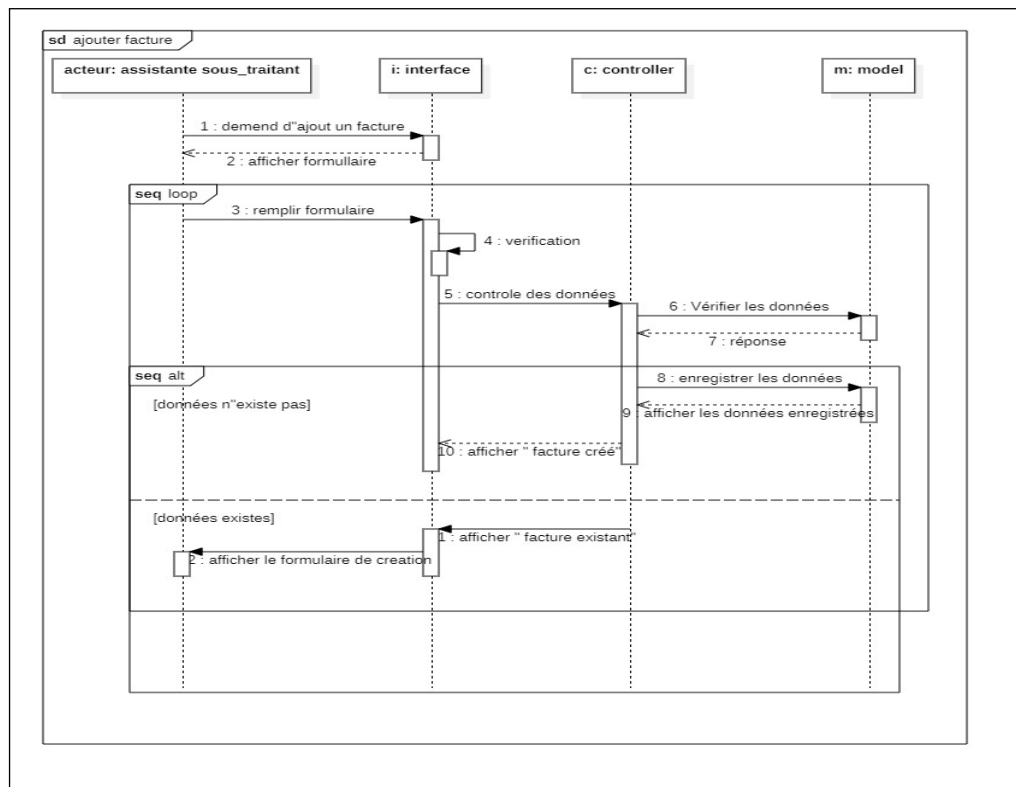


FIGURE 2.13 – Diagramme de séquence «ajouter factures »

## 2.9 Diagramme de séquence « ajouter débours » :

Le diagramme de séquence "ajouter débours" représente le processus de gestion des partenaires dans une application. Il met en évidence les interactions entre les acteurs et les composants impliqués dans cette gestion, comme présenté dans l'illustration ci-dessous :

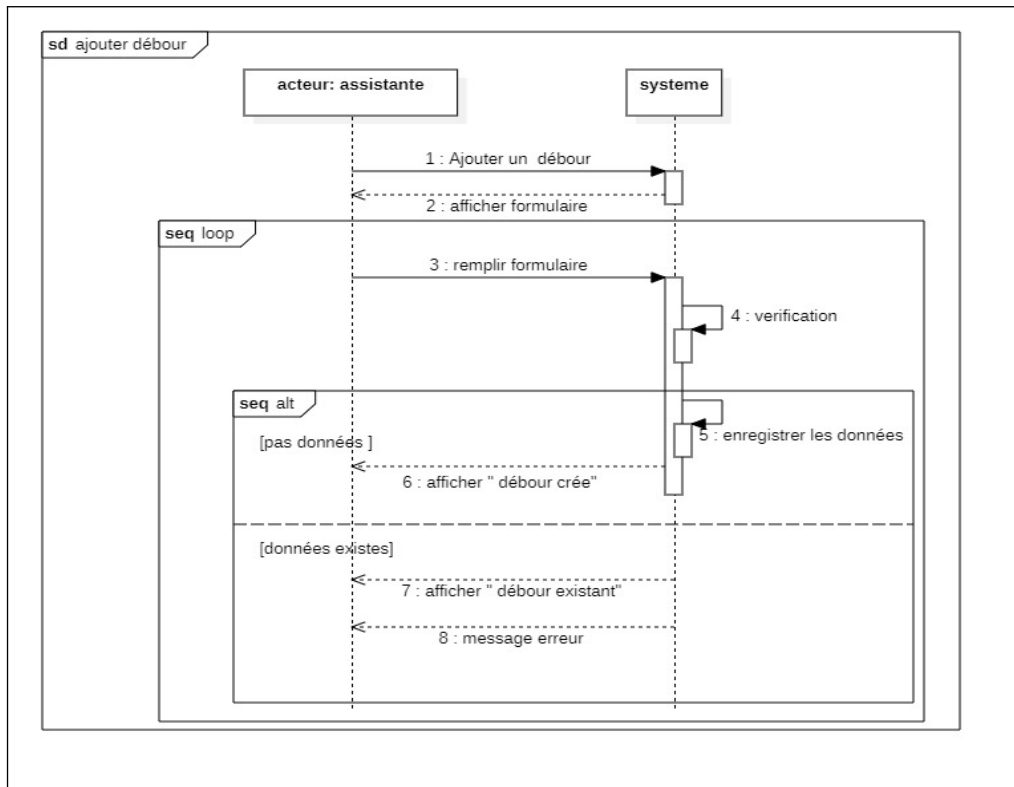


FIGURE 2.14 – Diagramme de séquence «ajouter débours »

## 2.10 Conclusion :

Dans ce chapitre, nous avons abordé la conception organisationnelle de notre système en expliquant les rôles de chaque acteur à travers différents diagrammes et scénarios. Dans le chapitre suivant, nous allons examiner les environnements utilisés pour la réalisation de notre projet, ainsi que les applications qui ont été développées.

---

# Réalisation et Implémentation

## Sommaire

---

<b>3.1</b>	<b>INTRODUCTION . . . . .</b>	<b>38</b>
<b>3.2</b>	<b>Environnement du travail : . . . . .</b>	<b>38</b>
3.2.1	Environnement matériel : . . . . .	38
<b>3.3</b>	<b>Environnement logiciel : . . . . .</b>	<b>39</b>
3.3.1	Langage de programmation : . . . . .	39
3.3.2	Les logiciels de programmation : . . . . .	39
<b>3.4</b>	<b>Interfaces : . . . . .</b>	<b>40</b>
3.4.1	Interfaces du site web : . . . . .	40
<b>3.5</b>	<b>Conclusion : . . . . .</b>	<b>55</b>

---



## 3.1 INTRODUCTION

Ce chapitre marque la conclusion du projet, avec la phase de réalisation et la présentation finale. Nous aborderons tout d'abord l'environnement de travail utilisé, suivi par une description des différentes interfaces de l'application développée.

## 3.2 Environnement du travail :

### 3.2.1 Environnement matériel :

Afin d'obtenir des temps de réponse et d'exécution optimaux pour notre application, il est essentiel d'utiliser un équipement performant doté d'un processeur et d'une mémoire adéquats. Pour le développement de notre application, nous avons utilisé un ordinateur portable DELL et LENOVO avec la configuration suivante :

**TABLE 3.1 – Caractéristique de l'ordinateur**

Marque :	- DELL - LENOVO
Processeur :	- Intel de 7ème génération. Selon le modèle, cela peut inclure des processeurs Core i5 - Intel de 7ème génération. Selon le modèle, cela peut inclure des processeurs Core i3
RAM :	-8GO -8GO
Disque dur :	-1 To de stockage SSD. -1 To de stockage SSD.
Système d'exploitation :	- Windows 11 - Windows 11

### 3.3 Environnement logiciel :

Dans cette partie, nous allons présenter les outils logiciels et l'environnement de développement que nous avons utilisés pour l'implémentation de notre application. Ces outils sont essentiels pour développer, tester et déployer l'application de manière efficace et cohérente.

#### 3.3.1 Langage de programmation :

- **HTML** : Html signifie « HyperText Markup Language » qu'on peut traduire par « langage de balises pour l'hypertexte ». Il est utilisé afin de créer et de représenter le contenu d'une page web et sa structure. [3]
- **CSS** : Les feuilles de style en cascade, généralement appelées CSS de l'anglais Cascading Style Sheets, forment un langage informatique qui décrit la présentation des documents HTML et XML. Les standards définissant CSS sont publiés par le World Wide Web Consortium (W3C).[4]
- **Bootstrap** : Bootstrap est une collection d'outils utiles à la création du design (graphisme, animation et interactions avec la page dans le navigateur, etc.) de sites et d'applications web.[5]
- **Angular** : Angular (communément appelé « Angular 2+ » ou « Angular v2 et plus »)2,3 est un framework pour clients, open source, basé sur TypeScript et codirigé par l'équipe du projet « Angular » chez Google ainsi que par une communauté de particuliers et de sociétés.[6]

#### 3.3.2 Les logiciels de programmation :

- **Visual Studio Code** : Visual Studio Code est un éditeur de code extensible développé par Microsoft pour Windows, Linux et macOS3.[7]
- **JSON** :JavaScript Object Notation (JSON) est un format de données textuel dérivé de la notation des objets du langage JavaScript. Il concurrence XML pour la représentation et la transmission d'information structurée1.[8]

- **GIT** : Git est un logiciel de gestion de versions décentralisé. C'est un logiciel libre et gratuit, créé en 2005 par Linus Torvalds, auteur du noyau Linux, et distribué selon les termes de la licence publique générale GNU version 2. [9]
- **IntelliJ IDEA** : également appelé « IDEA » est un environnement de développement destiné au développement de logiciels informatiques reposant sur la technologie Java. Il est développé par JetBrains et disponible en deux versions, l'une communautaire, open source, sous licence Apache 2 et l'autre propriétaire, protégée par une licence commerciale. Tous deux supportent les langages de programmation Java, Kotlin, Groovy et Scala.
- **MySQL** : en tant que système de gestion de base de données et mentionne sa dernière version, MySQL 8.0, soulignant qu'il offre une solution fiable et puissante pour gérer les données.

### 3.4 Interfaces :

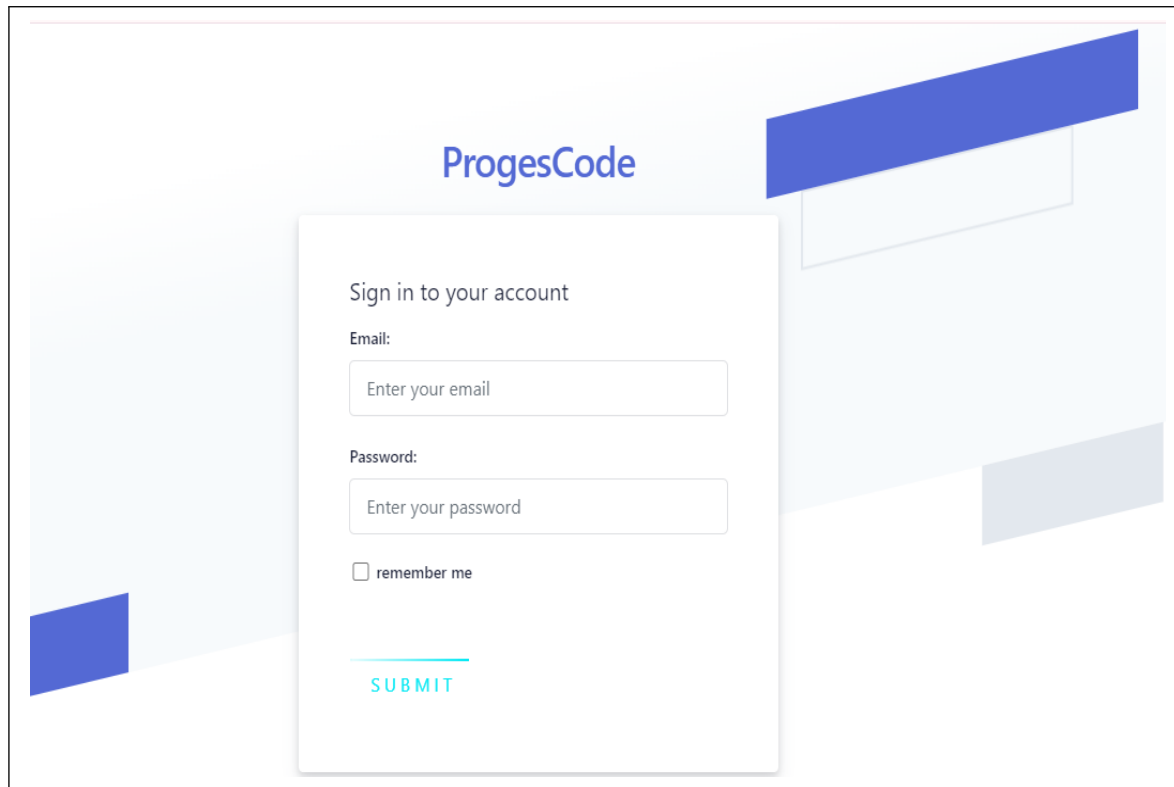
Dans la partie suivante, nous allons donner un aperçu des interfaces de l'application développée, illustrant les différents cas d'utilisation qui ont été abordés dans le chapitre d'analyse et de conception. Ces interfaces offrent aux utilisateurs une expérience visuelle pour interagir avec l'application et effectuer les tâches prévues.

#### 3.4.1 Interfaces du site web :

Le site web de notre système offre une gamme d'interfaces bien conçues pour garantir une expérience utilisateur intuitive et complète.

##### 3.4.1.1 Page d'authentification :

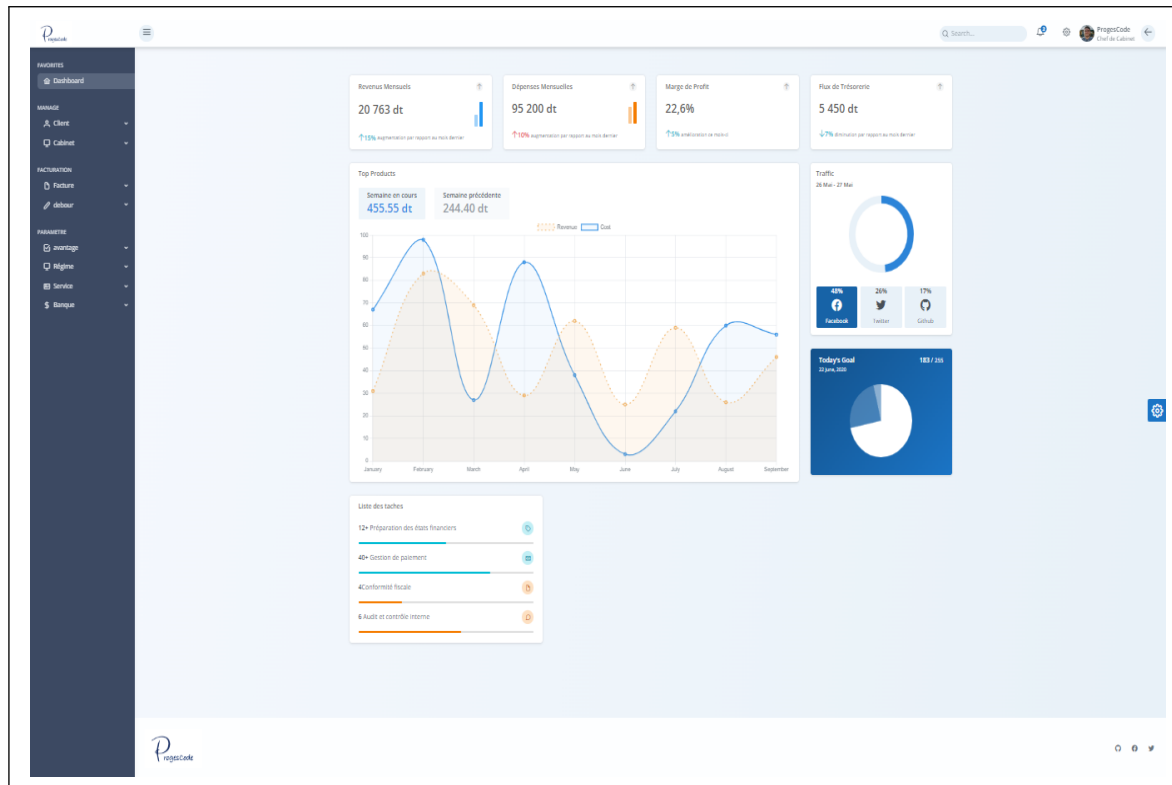
L'interface d'authentification de notre application est représentée dans la figure ci-dessous. Pour accéder à votre espace personnel, veuillez entrer votre email et votre mot de passe. Notre application propose par défaut trois types d'utilisateurs différents. Après une authentification réussie, l'utilisateur est automatiquement redirigé vers l'espace qui correspond à son type d'utilisateur



**FIGURE 3.1 – Page d’authentification**

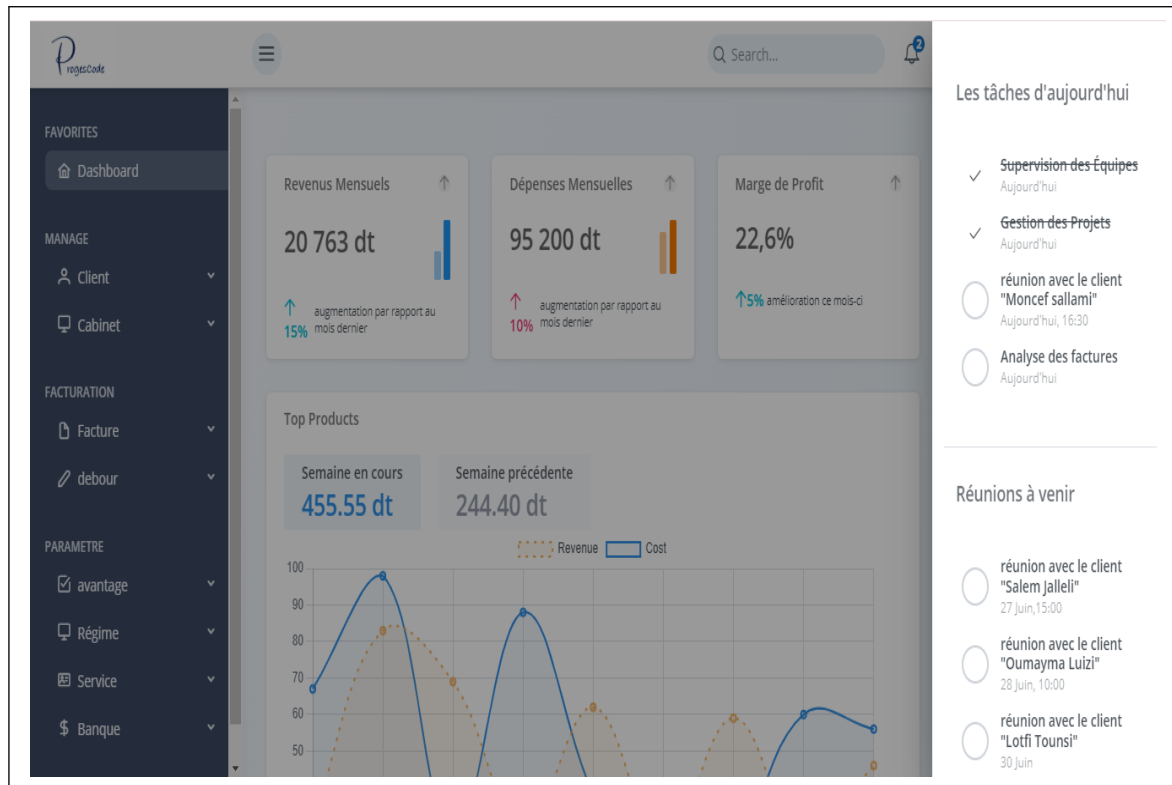
### **3.4.1.2 Page d’accueil :**

La figure ci-dessous représente l’interface principale de notre application. L’interface d’accueil de notre application web, illustré dans la figure , donne une présentation générale sur l’application et aide les utilisateurs à connaître l’ensemble des fonctionnalités et des services offerts par notre plateforme.



**FIGURE 3.2 – Page d'accueil**

Nous avons intégré dans le tableau de bord une fonctionnalité permettant au chef de cabinet de gérer efficacement ses tâches et rendez-vous. Cette liste des tâches offre une vue d'ensemble détaillée des responsabilités quotidiennes. Chaque tâche est présentée de manière claire et ordonnée, facilitant ainsi la priorisation et le suivi des actions à accomplir.



**FIGURE 3.3 – Liste des Tâches**

### 3.4.1.3 Page de Cabinet :

Cette page est conçue pour faciliter la gestion efficace des cabinets, permettant des modifications rapides, des suppressions sécurisées et la création facile de nouveaux cabinets.

#### • Page de création d'un cabinet

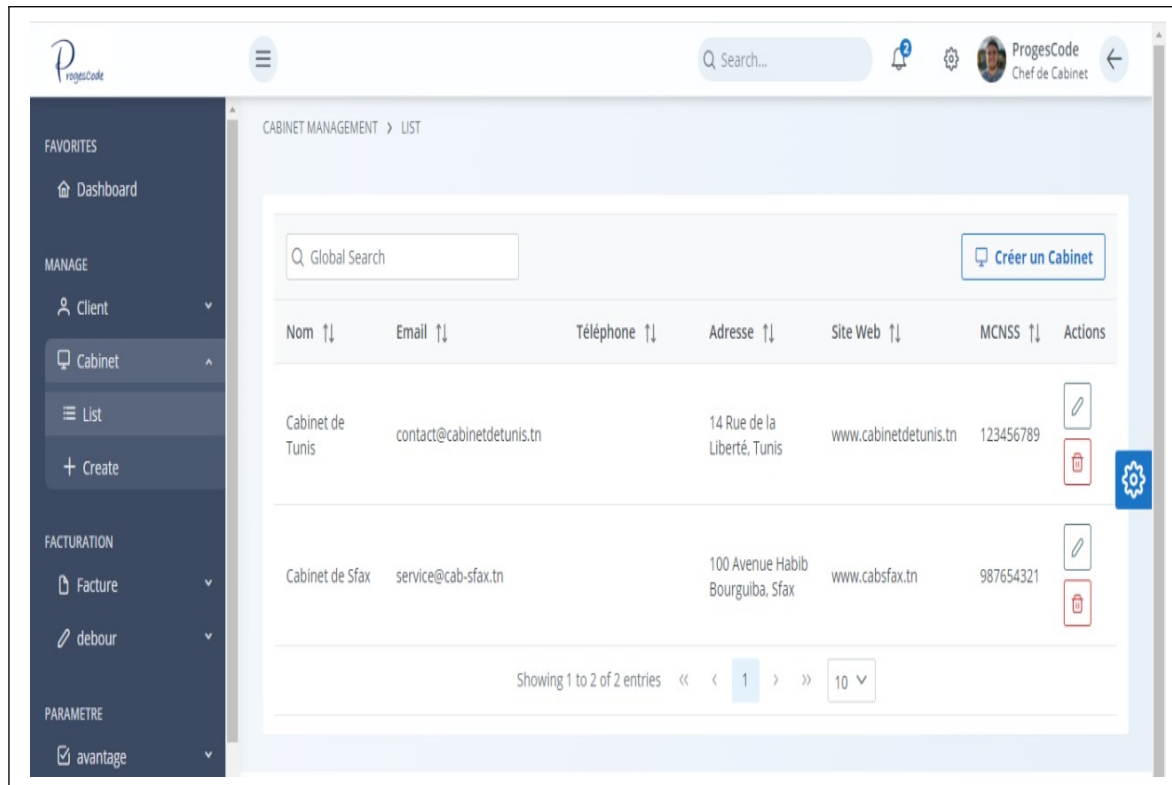
La page de création d'un cabinet présentée dans la figure, permet aux administrateurs de notre plateforme d'ajouter facilement de nouveaux cabinets partenaires. Cette page comprend un formulaire intuitif et complet où les administrateurs peuvent saisir toutes les informations nécessaires ce qui garantit une gestion efficace et une mise à jour en temps réel des informations sur les cabinets partenaires.

The screenshot displays the 'Créer un Cabinet' (Create a Cabinet) interface. On the left, a dark sidebar lists navigation categories: MANAGE (with sub-items Client, Cabinet, List, and a Create button), FACTURATION (with Facture and debour), and PARAMETRE (with avantage, Régime, Service, and Banque). The main content area is titled 'Créer un Cabinet' and contains several text input fields: 'Nom' (filled with 'Cabinet de Tunis'), 'Email' (filled with 'contact@cabinetdetunis.tn'), 'Téléphone' (filled with '+216 71 234 567'), 'Adresse' (filled with '14 Rue de la Liberté, Tunis'), 'Site Web' (filled with 'www.cabinetdetunis.tn'), and 'MCNSS' (filled with '123456789'). A blue 'Créer Cabinet' button is positioned at the bottom of the form. The top of the page features a header with the ProgesCode logo, a search bar, and a user profile for 'ProgesCode Chef de Cabinet'.

**FIGURE 3.4 – Page de création d’un cabinet**

- *Page de gestion de cabinet*

La page de gestion des cabinets présentée dans la figure, est conçue pour être informative et conviviale, permettant aux utilisateurs de découvrir facilement les différents cabinets partenaires et de trouver rapidement celui qui correspond le mieux à leurs besoins.



**FIGURE 3.5 – Page de gestion de cabinet**

### 3.4.1.4 Page de client :

La page de client permet une administration efficace des informations clients sur la plateforme, permettant des modifications rapides, des suppressions sécurisées et la création facile de nouvelles calients.

- *Page de création de client*

La page de création de client offre un formulaire intuitif pour saisir les informations d'un nouveau client.



**Créer un Client**

nomcabinet  
Cabinet de Tunis

clientName  
ahmad

Email  
ahmadchakroun904@gmail.com

Phone  
56752287

Adresse  
ROUTE SFAX km 1

Avantage  
Expertise ca... Consults Poca...

Regime  
REGIME FISCAL RETENUE A L...

Banque  
Banque de T...

RIB  
55

Nom Associé  
nadia

Pourcentage  
45%

**Créer Client**

**FIGURE 3.6 – Page de création de client**

- **Page de gestion de client**

La page de gestion des clients offre une interface pour gérer les clients enregistrés sur la plateforme.

**CLIENT MANAGEMENT > LIST**

Global Search Créer un Client

Nom ↑↓	Email ↑↓	Téléphone ↑↓	Adresse ↑↓	rib ↑↓	Actions
ahmad	ROUTE SFAX	ahmadchakroun904@gmail.com	+2166543211	55	
nadia	ROUTE SFAX	nadiadhifi14@gmail.com	+21626162259	99	
				0	

**FIGURE 3.7 – Page de gestion des clients**

### 3.4.1.5 Page de Facture :

La page de facturation fournit une interface centrale pour gérer les factures générées par la plateforme, permettant des modifications rapides, des suppressions sécurisées et la création facile de nouvelles factures.

- *Page de création de facture*

La page de création de facture offre un formulaire convivial pour générer de nouvelles factures dans la plateforme.

The screenshot displays the 'Créer une Facture' (Create Invoice) interface. On the left is a dark sidebar with a menu including 'MAINTENANT', 'MAINTENANT', 'CLIENT', 'CABINET', 'FACTURATION', 'Liste', '+ create', 'déclarer', 'PARAMÈTRES', 'inventaire', 'régime', 'Service', and 'Emploi'. The main content area is titled 'Créer une Facture' and contains the following form fields:

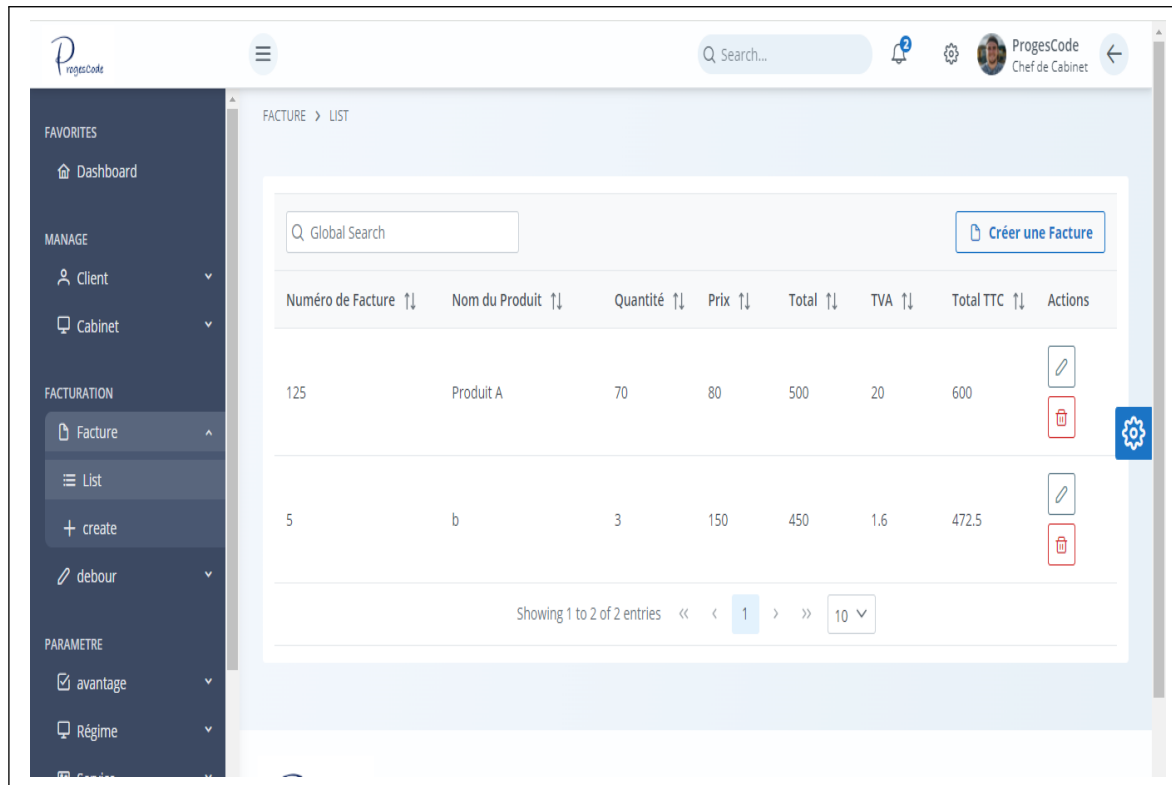
- Client Name: nadia
- Service: Déclarations fiscales
- Numéro de Facture: 5
- Date de Facture: 21/05/2024
- Date de Paiement: 21/05/2024
- Nom du Produit: b
- Quantité: 3
- Prix: 150
- Total: 450
- TVA: 1.6
- Total TTC: 472.5

A blue button labeled 'Créer une Facture' is positioned at the bottom of the form. The footer of the page includes the ProgestCode logo and social media icons.

**FIGURE 3.8 – Page de création de facture**

- *Page de gestion des factures*

La page de gestion des factures propose une interface centralisée pour administrer les factures enregistrées sur la plateforme.



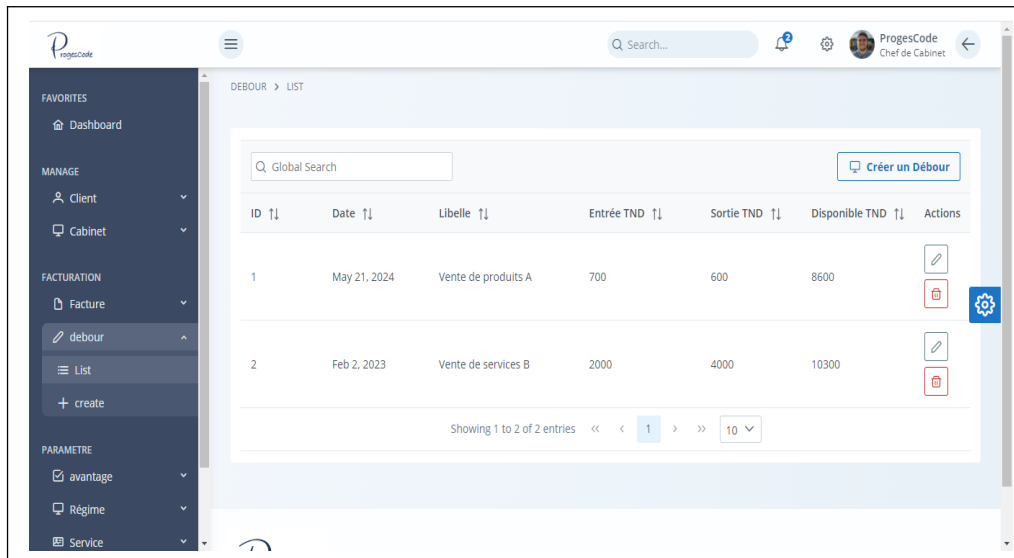
**FIGURE 3.9 – Page de gestion des factures**

### 3.4.1.6 Page de Débours :

La page de débours assure une gestion complète et efficace des dépenses engagées par le cabinet, garantissant une traçabilité et une organisation optimale. permettant des modifications rapides, des suppressions sécurisées et la création facile de nouvelles débours.

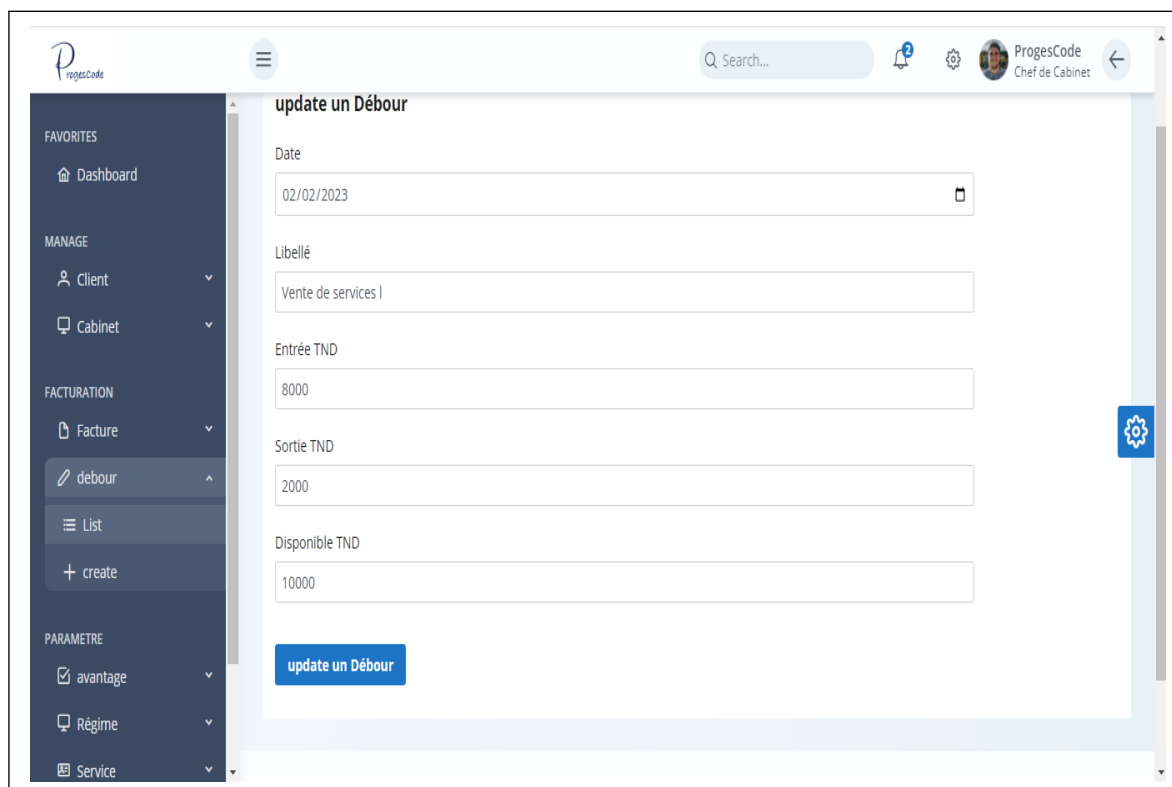
- *Page de gestion des débours*

La page de gestion des débours propose une interface centralisée pour administrer les débours enregistrées sur la plateforme.



**FIGURE 3.10 – Page de gestion des débours**

La page de modification de débours offre une interface pour mettre à jour les informations d'un débours déjà enregistré dans la plateforme.



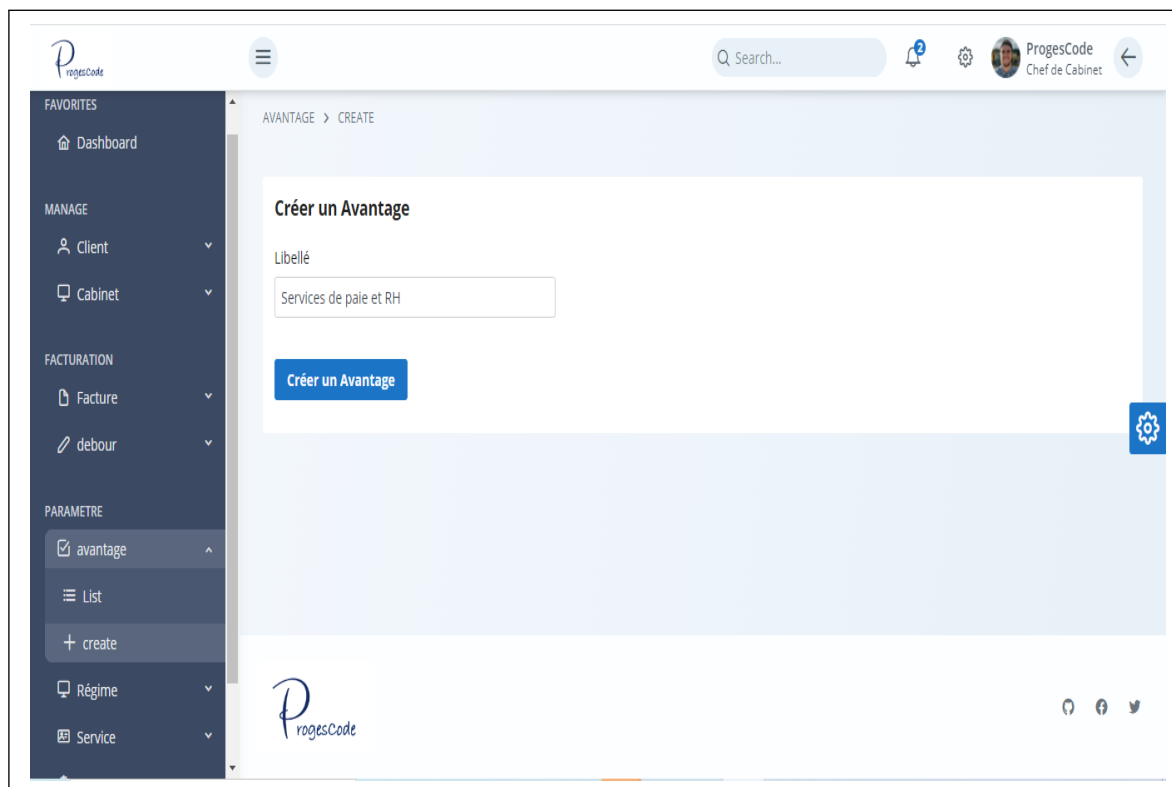
**FIGURE 3.11 – Page de modification débours**

### 3.4.1.7 Page d'avantage :

La page d'avantages offre une vue complète des avantages disponibles pour les clients du cabinet.

- *Page de création avantage*

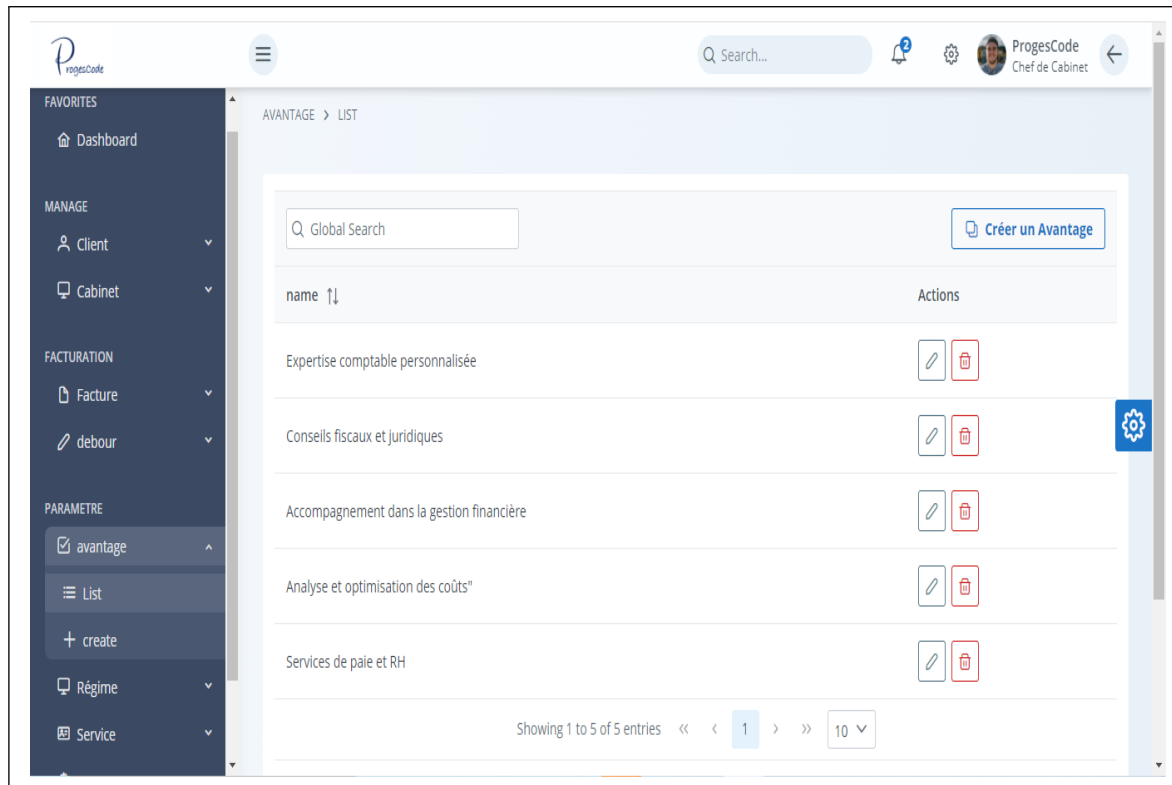
La page de création d'avantage fournit un formulaire convivial pour ajouter de nouveaux avantages à la plateforme.



**FIGURE 3.12 – Page de création avantage**

- *Page de gestion des avantages*

La page de gestion des avantages offre une interface pour administrer les avantages disponibles sur la plateforme.

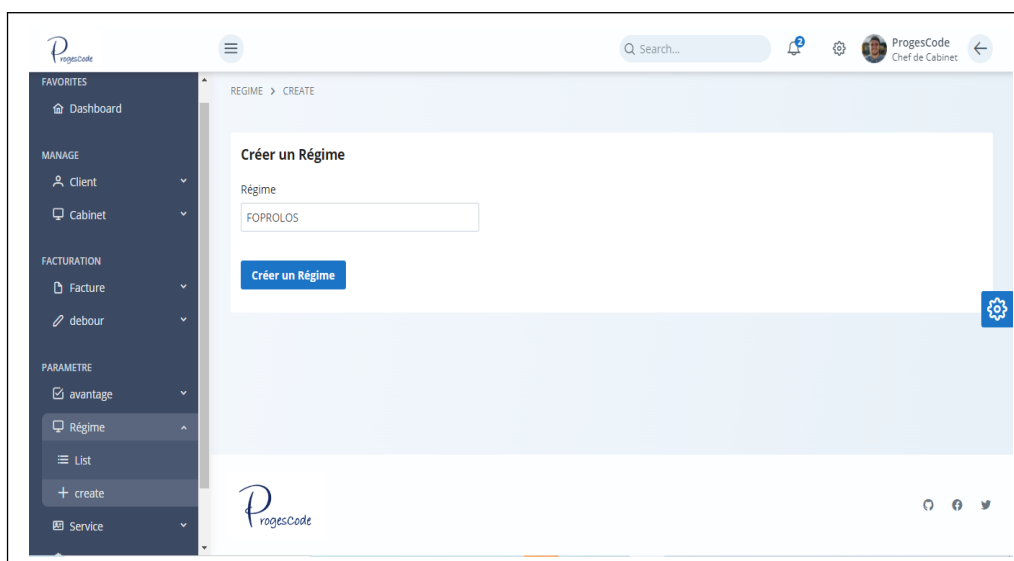


**FIGURE 3.13 – Page de gestion des avantages**

### 3.4.1.8 Page de régime :

- *Page de création de régime*

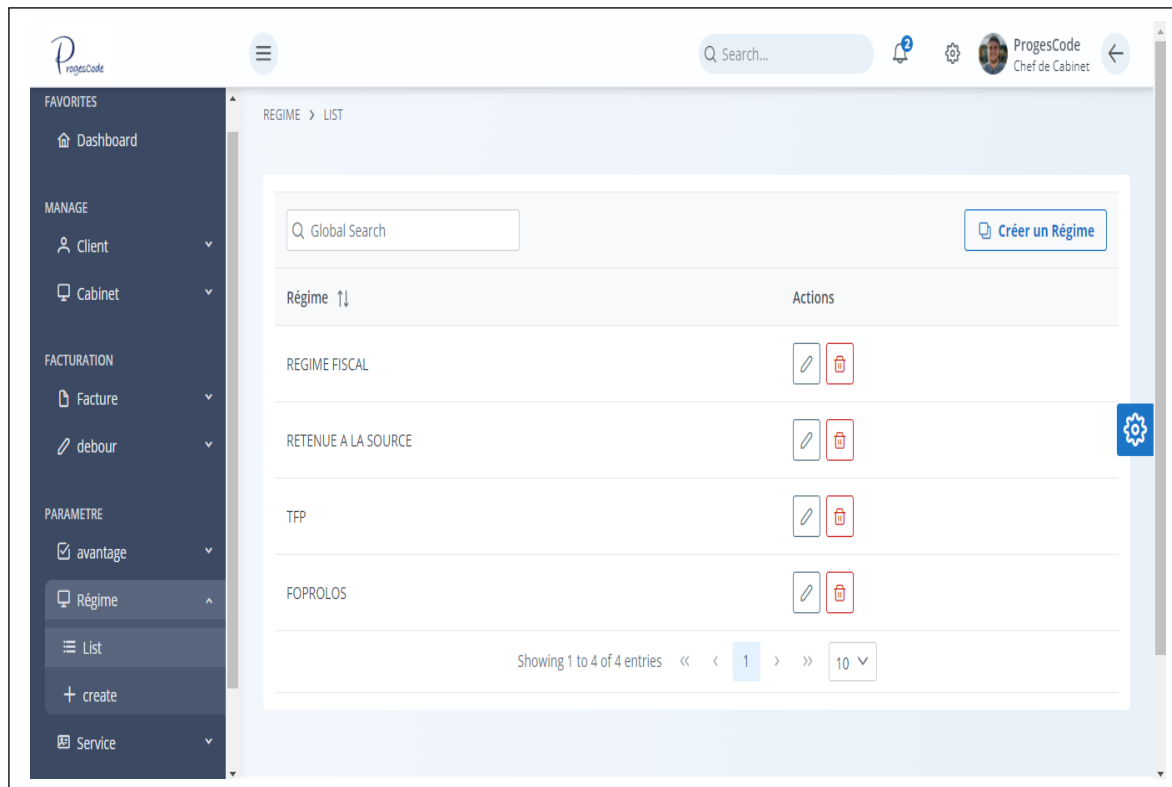
La page de création de régime vise à faciliter l'ajout de régimes spécifiques.



**FIGURE 3.14 – Page de création de régime**

### • *Page de gestion des régimes*

La page de gestion des régimes offre une interface pour administrer les différents régimes fiscaux et comptables disponibles sur la plateforme.

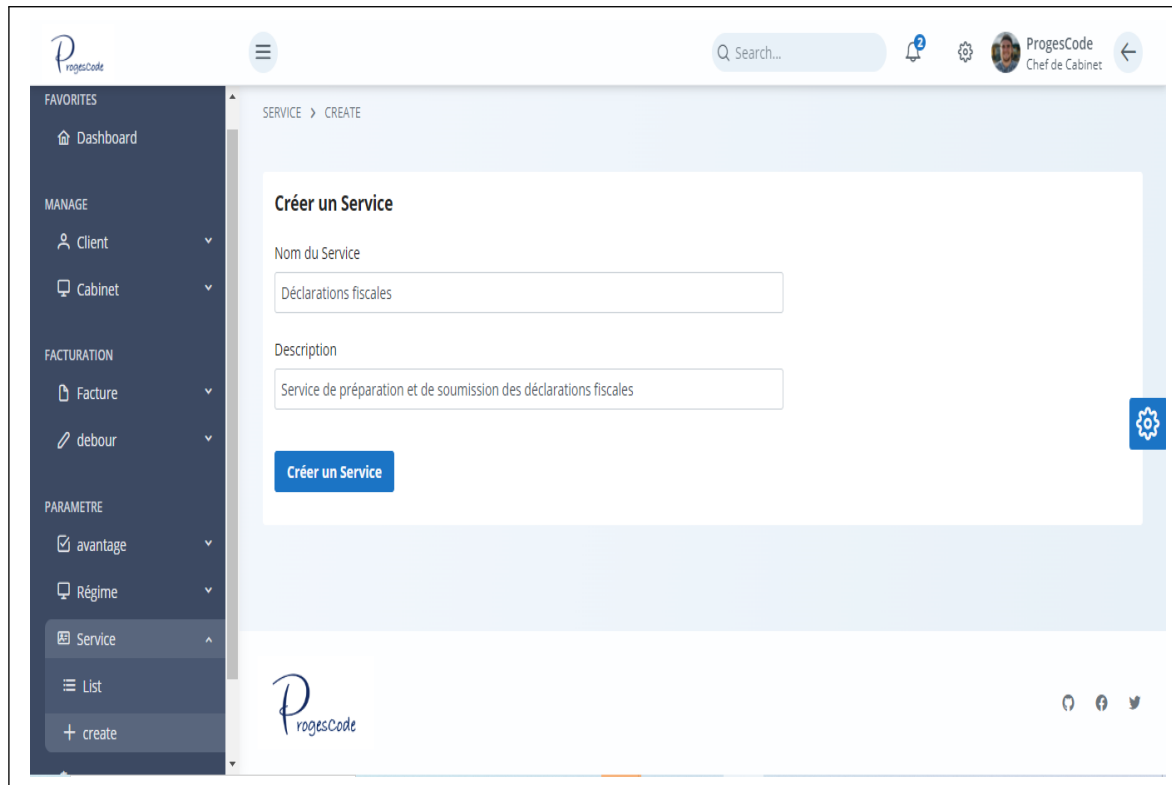


**FIGURE 3.15 – Page de gestion des régimes**

### 3.4.1.9 Page de service :

#### • *Page de création des services*

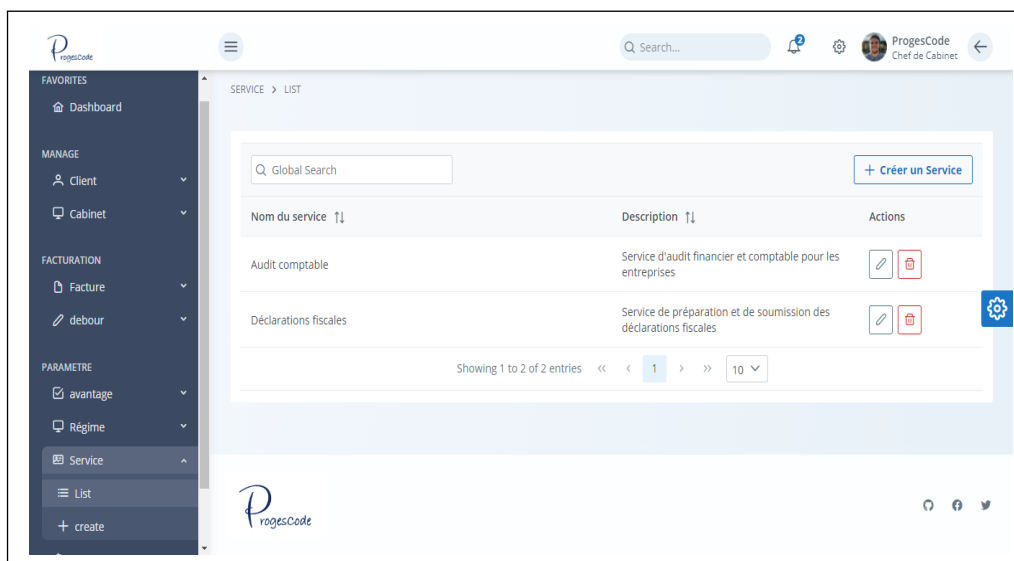
La page de création des services permet d'ajouter de nouveaux services comptables et financiers à la plateforme.



**FIGURE 3.16 – Page de création des services**

- *Page de gestion des services*

La page de gestion des services offre une interface complète pour administrer et surveiller les services proposés par le cabinet comptable.



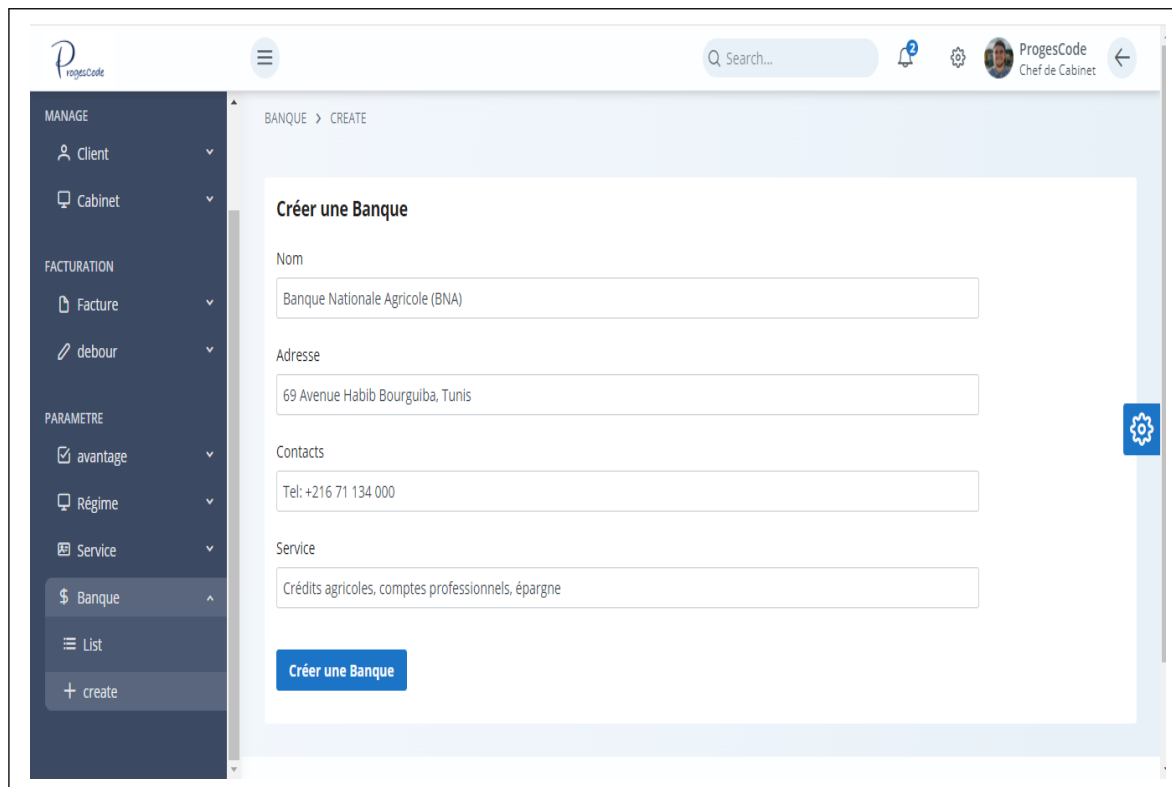
**FIGURE 3.17 – Page de gestion des services**



### 3.4.1.10 Page de Banque :

- *Page de création de banque*

La page de création de banque permet d'ajouter de nouvelles banques partenaires au système de gestion du cabinet comptable.

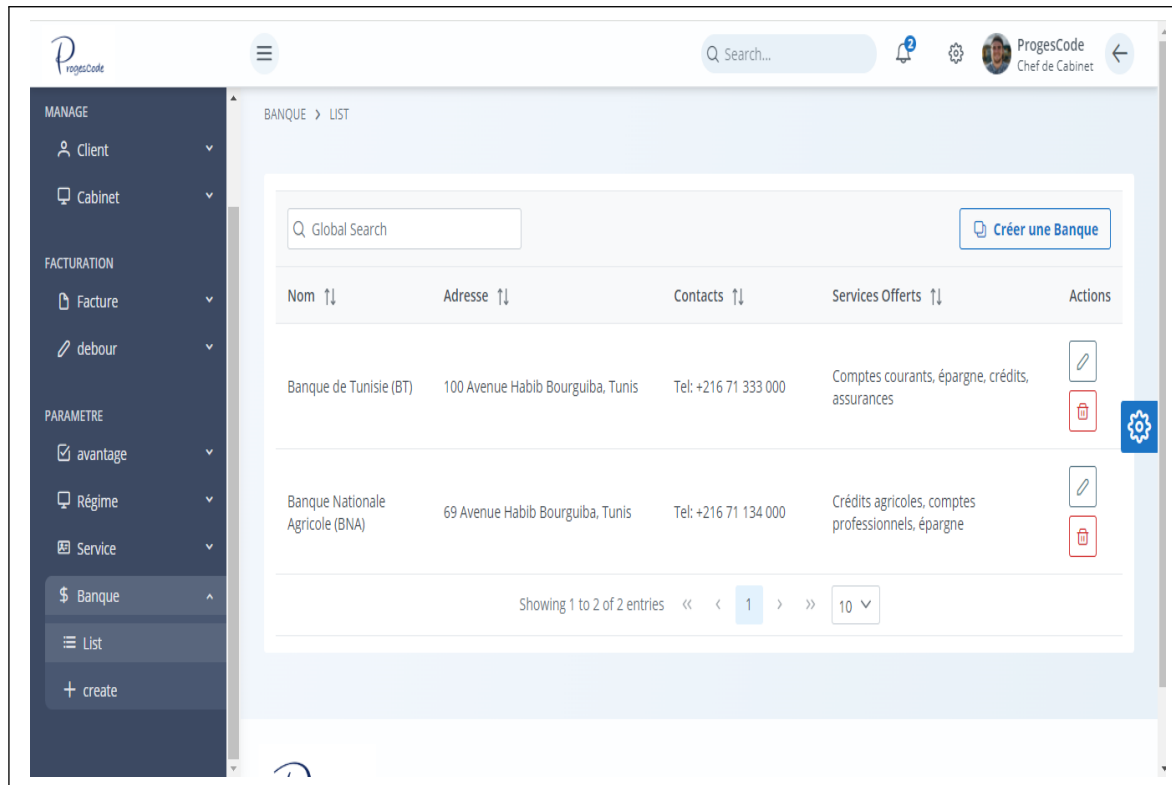


The screenshot displays the 'Créer une Banque' (Create a Bank) page within the ProgesCode application. The interface features a dark blue sidebar on the left with a menu containing 'MANAGE' (Client, Cabinet), 'FACTURATION' (Facture, debour), and 'PARAMETRE' (avantage, Régime, Service, Banque, List, + create). The 'Banque' option is currently selected. The main content area is titled 'BANQUE > CREATE' and contains a form with the following fields: 'Nom' (filled with 'Banque Nationale Agricole (BNA)'), 'Adresse' (filled with '69 Avenue Habib Bourguiba, Tunis'), 'Contacts' (filled with 'Tel: +216 71 134 000'), and 'Service' (filled with 'Crédits agricoles, comptes professionnels, épargne'). A blue 'Créer une Banque' button is positioned at the bottom of the form. The top of the page includes a search bar, notification bell, settings gear, and a user profile for 'ProgesCode Chef de Cabinet'.

**FIGURE 3.18 – Page de création de banque**

- *Page de gestion des banques*

Cette page est conçue pour offrir une gestion complète et facile des banques partenaires.



**FIGURE 3.19 – Page de gestion des banques**

## 3.5 Conclusion :

Dans ce dernier chapitre, nous avons présenté les outils et les éléments de l'application que nous avons utilisés pour le développement. Cependant, il est important de souligner que la réalisation de l'application n'est pas la dernière étape du processus de développement. Il est essentiel de continuer à surveiller et à superviser l'application dès sa mise en exploitation avec des utilisateurs. Cela permet de détecter d'éventuels bugs et anomalies et de les corriger, garantissant ainsi la stabilité et la fiabilité du système.



---

# CONCLUSION GÉNÉRALE

En conclusion, ce projet de fin d'étude a été dédié à la conception et à la réalisation d'une plateforme web pour un cabinet de comptabilité, visant à optimiser la gestion des services comptables et financiers. Tout au long du développement de cette application, nous avons entrepris une analyse minutieuse du contexte et des besoins des utilisateurs pour garantir une solution adaptée et efficace.

L'utilisation d'UML nous a permis de structurer les besoins de manière claire et de faciliter la communication entre les membres de l'équipe de développement. En intégrant des outils comme Visual Studio, JSON-SERVER et Angular, nous avons pu concevoir une application performante et évolutive.

Cette expérience de stage nous a offert l'opportunité de mettre en pratique nos connaissances académiques dans un contexte professionnel, en nous exposant à toutes les étapes d'un projet de développement logiciel, de l'analyse initiale à la phase de tests et de déploiement. Nous avons également renforcé nos compétences en gestion de projet et en travail d'équipe, essentielles dans le domaine informatique.

Nous sommes fiers de constater que notre plateforme répond aux attentes de nos utilisateurs, en offrant une solution de gestion complète et intuitive pour les cabinets de comptabilité. La validation de notre travail par les utilisateurs est une preuve de la qualité et de l'utilité de notre solution.

Regardant vers l'avenir, nous envisageons d'enrichir encore davantage l'expérience utilisateur en développant une version mobile de notre application. Cette extension permettra d'offrir un accès pratique et optimisé aux services comptables sur les appareils mobiles, répondant ainsi aux besoins croissants de mobilité et de flexibilité de nos utilisateurs.

En conclusion, ce projet nous a permis de développer des compétences techniques et professionnelles précieuses, et nous sommes déterminés à continuer d'innover pour fournir des solutions de qualité supérieure, tout en restant à l'écoute des besoins de nos utilisateurs. Nous sommes convaincus que notre plateforme Progescode est la solution idéale pour combler les lacunes et répondre aux défis des cabinets comptables, en leur offrant un outil puissant et flexible pour gérer efficacement leurs services comptables et financiers.



---

# BIBLIOGRAPHIE

[1] Mis à jour en le 5 Avril 2024 . Disponible sur :

<https://www.lucidchart.com/pages/fr/langage-uml>

[2] Mis à jour en le 15 Avril 2024 . Disponible sur :

<https://fr.m.wikipedia.org/wiki/Fichier:UMLlogo.svg>

[3] Mis à jour en le 20 Avril 2024 . Disponible sur :

<https://developer.mozilla.org/fr/docs/Web/HTML>

[4] Mis à jour en le 20 Avril 2024 . Disponible sur :

[https://fr.wikipedia.org/wiki/Feuilles\\_de\\_style\\_nascade](https://fr.wikipedia.org/wiki/Feuilles_de_style_nascade)

[5] Mis à jour en le 20 Avril 2024 . Disponible sur :

[https://fr.wikipedia.org/wiki/Bootstrap\\_framework](https://fr.wikipedia.org/wiki/Bootstrap_framework)

[6] Mis à jour en le 20 Avril 2024 . Disponible sur :

<https://fr.wikipedia.org/wiki/Angular>

[7] Mis à jour en le 20 Avril 2024 . Disponible sur :

<https://fr.wikipedia.org/wiki/VisualStudioCode>

[8] Mis à jour en le 20 Avril 2024 . Disponible sur :

[https://fr.wikipedia.org/wiki/JavaScript\\_object\\_notation](https://fr.wikipedia.org/wiki/JavaScript_object_notation)

[9] Mis à jour en le 20 Avril 2024 . Disponible sur :

<https://fr.wikipedia.org/wiki/Git>

[10] Mis à jour en le 20 Avril 2024 . Disponible sur :

[https://en.wikipedia.org/wiki/IntelliJ\\_IDEA](https://en.wikipedia.org/wiki/IntelliJ_IDEA)

[11] Mis à jour en le 20 Avril 2024 . Disponible sur :

<https://fr.wikipedia.org/wiki/MySQL>

## **Plateforme de Cabinet de Comptabilité "Progescode"**

### **Côte Front End et Back End**

---

---

**Nadia DHIFI et Ahmed CHAKROUN**

---

---

#### **Résumé :**

Progescode est une plateforme innovante de gestion comptable, conçue pour optimiser les opérations des cabinets de comptabilité. Sur le plan front-end, elle offre une interface utilisateur intuitive et réactive, permettant une navigation fluide et une expérience utilisateur améliorée. Le back-end, robuste et sécurisé, est développé pour assurer un traitement efficace des données financières, incluant la comptabilité générale, la gestion des paies, et les déclarations fiscales. L'intégration de technologies avancées permet une synchronisation en temps réel et une conformité réglementaire automatique, garantissant aux cabinets une gestion comptable précise et fiable.

**Mots clés :** Progescode, plateforme, comptable ,sécurisé.

#### **Abstract :**

Progescode is an innovative accounting management platform designed to optimize the operations of accounting firms. On the front end, it offers an intuitive and responsive user interface, ensuring smooth navigation and an enhanced user experience. The back end is robust and secure, developed to efficiently process financial data, including general accounting, payroll management, and tax filings. The integration of advanced technologies allows for real-time synchronization and automatic regulatory compliance, ensuring precise and reliable accounting management for firms.

**Key-words :** Progescode, Secure, Accounting, Platform.