

République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique



Université 20 Aout 1955. Skikda

Faculté des Sciences

Département : d'informatique



Mémoire de fin d'étude en vue de l'obtention du diplôme de master

Filière : Informatique

Spécialité : Systèmes d'informatique

Thème

*Plate-forme SaaS pour La Prise des Rendez-Vous avec
l'Intégration de l'IA.*

Présenté par: **Bouhaouche Fatima**

Encadré par: **Boulehouache Soufiane**

Année universitaire : 2024-2025

Remerciement

Par la grâce d'Allah, j'ai pu mener à bien ce mémoire dans les meilleures conditions.

Je tiens à exprimer ma profonde gratitude à mon encadrant Dr. Boulehouache Soufiane, pour ses conseils précieux, sa patience et son accompagnement tout au long de ce travail. Son soutien a été d'une grande importance dans l'orientation et la réalisation de cette recherche.

Un grand merci à ma famille pour leur présence constante, leur soutien moral et leurs encouragements sans faille. Leur confiance en moi a été une source essentielle de motivation.

Dédicace

Je dédie ce mémoire,

À mes chers parents, pour leur amour infini, leurs sacrifices et leurs prières qui m'ont toujours portée et soutenue.

À mon unique et précieux frère, pour sa présence, ses encouragements et sa confiance silencieuse qui m'ont donné de la force.

À mon cher fiancé, mon héros, pour son soutien, sa patience, sa motivation constante et sa foi en moi, ont été un refuge dans les moments les plus incertains.

À toute ma famille, pour leur bienveillance, leur fierté et leurs mots qui ont souvent redonné du sens à mes efforts.

Merci à chacun de vous d'avoir été une lumière sur mon chemin.

Sommaire

Introduction Générale.....	1
Objectifs du mémoire.....	1
Méthodologie adoptée.....	1
Chapitre I : La gestion des rendez-vous : enjeux, domaines et évolution.....	3
I.1. Introduction.....	4
I.2. Domaines concernés par la prise de rendez-vous.....	4
I.2.1 Secteur médical.....	4
I.2.2 Services juridiques.....	4
I.2.3 Beauté et bien-être.....	4
I.2.4 Entreprises de services.....	5
I.2.5 Éducation et formation.....	5
I.2.6 Secteur public et administratif.....	5
I.3. Méthodes traditionnelles de gestion des rendez-vous.....	5
I.4. Avantages des systèmes numériques.....	6
I.5. Limites des solutions actuelles.....	7
I.5.1 Manque de personnalisation selon le secteur d'activité.....	7
I.5.2 Interfaces peu intuitives ou complexes.....	7
I.5.3 Faible intégration de l'intelligence artificielle.....	8
I.5.4 Coûts élevés ou formules d'abonnement inadaptées.....	8
I.5.5 Gestion difficile des profils multiples ou multi-prestataires.....	8
I.6. Vers une solution intelligente et universelle.....	8
I.6.1 Multisectorielle.....	8
I.6.2 Personnalisable.....	8
I.6.3 Intelligente.....	9
I.6.4 Basée sur le cloud (modèle SaaS).....	9
I.7. Conclusion.....	9
Chapitre II : Le modèle SaaS : principes, avantages et application à la gestion des rendez-vous.....	10
II.1. Introduction.....	11
II.2. Définition du modèle SaaS.....	11
II.3. Fonctionnement général.....	11
II.4. Caractéristiques principales.....	11
II.4.1 Modèle multi-tenant (mutualisation intelligente).....	12
II.4.2 Modèle d'abonnement souple et économique.....	12
II.4.3 Déploiement centralisé et maintenance simplifiée.....	12
II.5. Avantages du modèle SaaS.....	12
II.5.1 Accessibilité accrue.....	12
II.5.2 Réduction des coûts.....	12
II.5.3 Simplicité de maintenance.....	12
II.5.4 Évolutivité.....	13
II.5.5 Sécurité centralisée.....	13
II.6. Application du SaaS à la gestion des rendez-vous.....	13
II.6.1 Accès partagé et synchrone.....	13
II.6.2 Gestion des disponibilités.....	13
II.6.3 Notifications automatisées.....	13
II.6.4 Tableaux de bord et analyses.....	13
II.7. Exemples de plateformes SaaS existantes.....	13
II.7.1 Doctolib.....	13
II.7.2 Calendly.....	14
II.7.3 Booknetic.....	14
II.7.4 Cal.com (ex Calendso).....	14
II.7.5 Limites observées.....	14
II.8. Justification du choix du SaaS pour notre projet.....	14
II.8.1 Support multi-profil et multi-secteurs.....	14

II.8.2	Centralisation et simplification	14
II.8.3	Intégration de l'intelligence artificielle	14
II.8.4	Scalabilité et accessibilité	14
II.9.	Conclusion	15
Chapitre III :	Conception du Système	16
III.1.	Introduction	17
III.2.	Concepts de base de développement	17
III.2.1	Langage de modélisation UML	17
III.2.1.1	Définition d'UML	17
III.2.1.2	Les diagrammes d'UML	17
III.2.1.3	Caractéristiques d'UML	18
III.2.1.4	Les avantages d'UML	19
III.2.1.5	Inconvénients d'UML	19
III.3.	Les acteurs du système	19
III.4.	Analyse des besoins	20
III.4.1	Besoins fonctionnels	20
III.4.2	Besoins non fonctionnels	20
III.5.	Diagramme de cas d'utilisation	21
III.5.1	Descriptions textuelles détaillées	22
III.5.1.1	Créer un compte	22
III.5.1.2	Modifier un compte	22
III.5.1.3	Rechercher un prestataire	23
III.5.1.4	Réserver un RDV	23
III.5.1.5	Reprogrammer un RDV	24
III.5.1.6	Annuler un RDV	24
III.5.1.7	Lance AI (Assistant AI)	25
III.5.1.8	Gérer les demandes de RDV	25
III.5.1.9	Lancer le chat	26
III.5.1.10	Gérer client	26
III.6.	Diagramme de séquence :	26
III.7.	Diagramme de classe	32
III.8.	Diagramme d'activité	33
III.9.	Conclusion	33
Chapitre IV :	Implémentation	40
IV.1.	Introduction	41
IV.2.	Environnement de développement	41
IV.2.1	Plateforme de développement	41
IV.2.2	Les plateformes / outils utilisés	42
IV.3.	Intégration du module IA	43
IV.3.1	Architecture du module IA	43
IV.3.2	Fonctionnalités du module IA	43
IV.3.3	Communication entre les composants	44
IV.4.	Présentation de la plateforme :	44
IV.4.1	La page d'accueil:	44
IV.4.2	L'interface de prestataire de service :	44
IV.4.2.1	L'inscription	44
IV.4.2.2	Dashboard	45
IV.4.2.3	Appointments (Les RDV)	46
IV.4.2.4	Create Appointments (Créer un RDV) :	46
IV.4.2.5	Appointments requests (Les demandes de RDV) :	46
IV.4.2.6	Reschedule Appointments (Reprogrammer un RDV) :	47
IV.4.2.7	Chat :	47
IV.4.2.8	AI Assistant:	48
IV.4.3	L'interface Client:	48
IV.4.3.1	L'inscription :	48
IV.4.3.2	Dashboard :	49
IV.4.3.3	Search Business Owner (Prestataire de service) :	49

IV.4.3.4 Book appointment (Réserver un RDV) :	49
IV.4.3.5 Appointments (Les RDV) :	50
IV.4.3.6 Profile :	50
IV.4.3.7 Notifications:	50
IV.4.3.8 AI Assistant:	51
IV.5. Conclusion	51
Conclusion Générale	52
Perspectives	53
Bibliographie	54

Liste des figures

Figure III.1 : Diagramme de cas d'utilisation	21
Figure III.2 : Créer un compte	27
Figure III.3: Modifier un compte	27
Figure III.4: Recherche un prestataire	28
Figure III.5: Créer un RDV	28
Figure III.6: Reprogrammer un RDV	29
Figure III.7: Annuler un RDV	29
Figure III.8: Lancer AI	30
Figure III.9: Gérer utilisateurs	30
Figure III.10: Gérer Les demandes de RDV	31
Figure III.11: Lancer le chat	31
Figure III.12: Diagramme de classe	32
Figure III.13: Créer un compte	33
Figure III.14 : Modifier un compte	34
Figure III.15: Recherche un prestataire	34
Figure III.16: Créer un RDV	35
Figure III.17: Reprogrammer un RDV	35
Figure III.18: Annuler un RDV	36
Figure III.19: Lancer IA	36
Figure III.20: Gérer utilisateurs	37
Figure III.21: Lancer le chat	37
Figure III.22: Gérer les demandes de RDV	38

Liste des tableaux

Tableau III.1 : Les acteurs du système	19
Tableau III.2: Description textuelle de cas d'utilisation : Créer un compte	22
Tableau III.3: Description textuelle de cas d'utilisation : Modifier un compte	22
Tableau III.4: Description textuelle de cas d'utilisation : Rechercher un prestataire	23
Tableau III.5: Description textuelle de cas d'utilisation : Réserver un RDV	23
Tableau III.6: Description textuelle de cas d'utilisation : Reprogrammer un RDV	24
Tableau III.7: Description textuelle de cas d'utilisation : Annuler un RDV	24
Tableau III.8: Description textuelle de cas d'utilisation : Lancer AI	25
Tableau III.9: Description textuelle de cas d'utilisation : Gérer les demandes de RDV	25
Tableau III.10: Description textuelle de cas d'utilisation : Lancer le chat	26
Tableau III.11: Description textuelle de cas d'utilisation : Gérer Utilisateurs	26

Résumé

Ce mémoire présente la conception et le développement d'une plateforme SaaS intelligente de gestion de rendez-vous, destinée à plusieurs secteurs professionnels (santé, droit, esthétique, entreprises, etc.). Le projet est né d'un constat clair : les méthodes traditionnelles et certaines solutions numériques actuelles restent limitées en termes de personnalisation, d'intelligence et de flexibilité. La solution proposée repose sur une architecture modulaire comprenant un frontend React.js, un backend Node.js, une base de données MongoDB, et un module d'intelligence artificielle développé en Python. Ce module permet de recommander automatiquement des créneaux optimaux aux utilisateurs, en s'appuyant sur leurs habitudes et les disponibilités des prestataires. Le système offre une expérience fluide, sécurisée et évolutive, posant les bases d'un service numérique universel, intelligent et adaptable aux réalités de différents métiers.

Introduction

Générale

Introduction générale

Dans un monde de plus en plus digitalisé, la gestion des rendez-vous occupe une place centrale dans le bon fonctionnement des entreprises et des prestataires de services.

Que ce soit dans le domaine médical, juridique, esthétique ou entrepreneurial, la prise de rendez-vous est une tâche quotidienne qui, bien qu'en apparence simple, soulève plusieurs problématiques concrètes.

En effet, les systèmes traditionnels de réservation souffrent souvent de limitations importantes : indisponibilité des créneaux, doublons, retards dans les réponses, ou encore absence de personnalisation selon les profils clients. Ces obstacles peuvent générer une perte de temps, une baisse de la satisfaction client, voire un impact négatif sur le chiffre d'affaires.

Dans ce contexte, l'intégration des technologies émergentes – notamment l'intelligence artificielle – offre des perspectives prometteuses. Le modèle SaaS (Software as a Service), qui permet la mise à disposition d'une plateforme accessible en ligne, constitue une solution particulièrement adaptée. Il garantit non seulement une accessibilité continue, mais aussi une centralisation et une automatisation optimisées de la gestion des rendez-vous.

Le présent projet de fin d'études propose donc le développement d'une plateforme SaaS «Syncly» intelligente de gestion de rendez-vous, destinée à plusieurs catégories de professionnels (médecins, avocats, salons de beauté, entreprises, etc.). Cette plateforme vise à faciliter la prise de rendez-vous en ligne tout en intégrant un module d'intelligence artificielle capable de recommander les meilleurs créneaux selon la disponibilité du prestataire et les habitudes du client.

Objectifs du mémoire

Ce mémoire a pour objectif principal de concevoir et développer une solution SaaS modulable, intuitive et intelligente, qui permette :

- à tout prestataire de configurer facilement ses créneaux disponibles ;
- aux clients de rechercher et réserver des rendez-vous adaptés à leurs besoins ;
- à un assistant IA de proposer automatiquement les horaires optimaux.

Méthodologie adoptée

Pour répondre à cette problématique, nous avons suivi une démarche structurée allant de l'analyse du besoin jusqu'à l'implémentation technique du projet. Le mémoire est structuré comme suit :

- Chapitre 1 : Étude du domaine de la gestion des rendez-vous et de ses enjeux, avec des exemples concrets ;
- Chapitre 2 : Présentation du modèle SaaS et de ses caractéristiques techniques et fonctionnelles ;

Introduction générale

- Chapitre 3 : Conception de la solution (diagrammes UML, architecture, algo-rithmes IA) ;
- Chapitre 4 : Implémentation et démonstration pratique de la plateforme avec des captures d'écran.

Chapitre I : La gestion des rendez-vous : enjeux, domaines et évolution

I.1. Introduction

La gestion des rendez-vous constitue une activité transversale dans de nombreux secteurs d'activité. Elle consiste à organiser, planifier et suivre les rencontres entre un prestataire de services (professionnel ou entreprise) et ses clients. Historiquement effectuée de manière manuelle ou par téléphone, cette tâche a largement évolué avec la digitalisation des entreprises.

Aujourd'hui, une mauvaise gestion des rendez-vous peut avoir des conséquences néfastes : perte de clients, temps d'attente allongé, surcharge de travail pour le personnel administratif, et surtout une expérience utilisateur médiocre.

I.2. Domaines concernés par la prise de rendez-vous

La gestion de rendez-vous s'applique à une multitude de secteurs, chacun présentant des exigences, des contraintes et des pratiques spécifiques. Voici une analyse détaillée des principaux domaines concernés :

I.2.1 Secteur médical

Les professionnels de santé, tels que les médecins généralistes, spécialistes, cliniques ou laboratoires, doivent organiser efficacement les consultations tout en tenant compte des urgences, de la durée variable des soins et des contraintes sanitaires. Des plateformes comme Doctolib permettent aux patients de réserver des créneaux en ligne, mais exigent en parallèle une gestion rigoureuse des disponibilités et des types de consultations.

Contraintes spécifiques : confidentialité des données médicales, priorisation des urgences, durée variable selon le motif, intégration aux dossiers médicaux.

I.2.2 Services juridiques

Les avocats, notaires et médiateurs doivent planifier des rendez-vous souvent confidentiels et complexes. Ces rendez-vous nécessitent parfois une analyse préalable des dossiers et une préparation logistique.

Contraintes spécifiques : confidentialité élevée, nature souvent urgente ou légale, besoin de documents préparatoires.

I.2.3 Beauté et bien-être

Ce secteur inclut des services comme la coiffure, les soins esthétiques, le massage ou la manucure. Chaque service a une durée définie, et les périodes de forte demande (weekends, fêtes) nécessitent une organisation optimale.

Chapitre I : La gestion des rendez-vous : enjeux, domaines et évolution

Contraintes spécifiques : durées variables des prestations, gestion des pics de fréquentation, diversité des services.

I.2.4 Entreprises de services

Les agences immobilières, sociétés de conseil ou entreprises techniques (réparation, installation) ont souvent besoin de planifier des visites sur site. La gestion logistique, la synchronisation des équipes, ou encore la localisation géographique rendent la prise de rendez-vous plus complexe.

Contraintes spécifiques : mobilité des équipes, coordination des plannings, gestion géographique des missions.

I.2.5 Éducation et formation

Les établissements scolaires, formateurs indépendants ou coachs personnels proposent souvent des séances personnalisées. La flexibilité des horaires est un critère central pour répondre à la disponibilité des étudiants et enseignants.

Contraintes spécifiques : flexibilité horaire, multi-disponibilités, organisation récurrente ou ponctuelle.

I.2.6 Secteur public et administratif

La numérisation des démarches administratives impose de plus en plus la prise de rendez-vous en ligne pour des services tels que la demande de carte d'identité, de passeport ou de permis. Ces démarches nécessitent un strict respect des créneaux pour éviter les files d'attente.

Contraintes spécifiques : exactitude des horaires, volume élevé, respect des procédures réglementaires.

I.3. Méthodes traditionnelles de gestion des rendez-vous

Avant l'apparition des outils numériques spécialisés, la gestion des rendez-vous reposait sur des méthodes dites traditionnelles, largement utilisées dans les établissements médicaux, juridiques, administratifs ou encore artisanaux. Ces approches, bien qu'encore présentes dans certains contextes, révèlent aujourd'hui leurs nombreuses limites.

La première méthode repose sur la prise de rendez-vous par téléphone ou à travers un accueil physique, assurée généralement par un secrétaire ou un réceptionniste. Cette méthode a l'avantage d'instaurer un contact humain direct, mais elle est rapidement dépassée dans les contextes à forte affluence. Elle souffre notamment de contraintes horaires (accessibilité limitée aux heures de travail), de surcharge du personnel administratif, de délais de réponse longs, ainsi

Chapitre I : La gestion des rendez-vous : enjeux, domaines et évolution

que d'un risque d'erreurs dues à la saisie manuelle ou à une mauvaise compréhension au téléphone.

La seconde approche consiste à utiliser des agendas papier ou des fichiers Excel pour enregistrer les réservations. Cette méthode, simple à mettre en œuvre, est encore courante dans les petites structures. Toutefois, elle présente des limites importantes :

- les informations ne sont pas accessibles simultanément par plusieurs utilisateurs ;
- il n'existe pas d'historique automatisé des modifications ;
- les risques de perte, de détérioration ou d'erreur sont élevés ;
- et surtout, il n'est pas possible d'intégrer ces outils avec des rappels automatiques, des notifications ou d'autres services numériques.

Enfin, certaines organisations se sont dotées de plateformes internes de réservation, souvent développées sur mesure. Bien qu'elles permettent une informatisation partielle du processus, ces plateformes souffrent généralement :

- d'un manque d'ergonomie ;
- d'une interface peu intuitive, parfois obsolète ;
- d'une faible capacité d'adaptation aux différents métiers ;
- et d'une absence de compatibilité avec les appareils mobiles.

Dans l'ensemble, ces méthodes traditionnelles sont devenues insuffisantes pour répondre aux exigences actuelles en matière de rapidité, de disponibilité continue, de personnalisation et d'efficacité. Elles justifient pleinement le recours à des solutions plus modernes, automatisées, centralisées et accessibles, notamment à travers des plateformes SaaS (Software as a Service) intégrant l'intelligence artificielle.

I.4. Avantages des systèmes numériques

La transition vers des systèmes numériques a profondément transformé la manière dont les professionnels gèrent les prises de rendez-vous. Contrairement aux méthodes traditionnelles, les solutions digitales apportent une série d'avantages significatifs, tant pour les prestataires que pour les clients.

Tout d'abord, elles garantissent une disponibilité 24h/24 et 7j/7, grâce à des interfaces accessibles en ligne, permettant ainsi aux utilisateurs de réserver ou de modifier un rendez-vous à tout moment, indépendamment des horaires d'ouverture. Cette accessibilité permanente constitue un avantage concurrentiel majeur, notamment dans les secteurs où la réactivité est primordiale.

Chapitre I : La gestion des rendez-vous : enjeux, domaines et évolution

Ensuite, la digitalisation permet de réduire considérablement les tâches répétitives et manuelles, telles que la saisie des rendez-vous, la gestion des appels ou le suivi papier. Cela libère du temps pour les équipes, qui peuvent se concentrer sur des missions à plus forte valeur ajoutée.

Par ailleurs, ces systèmes offrent des fonctionnalités automatisées, comme l'envoi de rappels par SMS ou email, afin de limiter les oublis et les absences [4]. Ce type de rappel améliore significativement la ponctualité des clients et la fiabilité du planning.

Les plateformes numériques permettent également une centralisation des données et un suivi historique des réservations, ce qui facilite l'analyse, la traçabilité et la gestion administrative des rendez-vous passés et futurs. Cette consolidation des informations est essentielle pour une prise de décision éclairée.

De plus, l'expérience utilisateur est nettement améliorée : les interfaces sont généralement intuitives, le parcours est simplifié, et les clients se sentent plus autonomes dans leurs interactions [5]. Cela contribue directement à renforcer la fidélisation et la satisfaction.

Enfin, ces outils intègrent souvent des fonctionnalités analytiques, permettant aux prestataires d'accéder à des indicateurs clés tels que le taux d'occupation, les heures de forte affluence, ou encore les types de services les plus réservés [1]. Ces données sont précieuses pour optimiser l'organisation et anticiper la demande.

Ainsi, l'adoption des systèmes numériques dans la gestion des rendez-vous n'est pas simplement une tendance technologique, mais bien une réponse stratégique aux besoins actuels d'agilité, de fiabilité et de performance.

I.5. Limites des solutions actuelles

Malgré les nombreux bénéfices des plateformes numériques actuelles, plusieurs limites importantes persistent, notamment dans leur capacité à répondre aux besoins variés et évolutifs des professionnels. Parmi ces limites, on peut citer :

I.5.1 Manque de personnalisation selon le secteur d'activité

De nombreuses solutions restent généralistes et ne permettent pas d'adapter les champs, les formulaires ou les processus en fonction des spécificités d'un métier donné (médecin, avocat, esthéticienne, etc.). Cela limite leur pertinence dans des contextes spécialisés.

I.5.2 Interfaces peu intuitives ou complexes

Certaines plateformes souffrent d'un design obsolète ou d'un parcours utilisateur non ergonomique. Cela rend leur utilisation difficile, en particulier pour les utilisateurs peu familiers avec les outils numériques, réduisant ainsi leur adoption.

I.5.3 Faible intégration de l'intelligence artificielle

Bien que l'IA offre des opportunités considérables (recommandation de créneaux, analyse de comportement), elle est encore peu exploitée dans les solutions existantes [2]. La planification intelligente et l'optimisation automatique restent rares.

I.5.4 Coûts élevés ou formules d'abonnement inadaptées

Les modèles économiques proposés par de nombreuses plateformes ne sont pas toujours flexibles. Les abonnements sont parfois trop coûteux pour les micro-entreprises, freelances ou structures à budget limité, ce qui constitue une barrière à l'entrée.

I.5.5 Gestion difficile des profils multiples ou multi-prestataires

Dans les environnements où plusieurs professionnels travaillent sur une même plateforme (cabinet médical, agence juridique, centre esthétique), la gestion des accès, des plannings ou des profils reste souvent limitée. Cela complique la coordination et nuit à l'expérience globale.

Ces limites démontrent que, bien que les systèmes actuels aient largement amélioré la gestion des rendez-vous, ils restent perfectibles et appellent à des solutions plus flexibles, adaptables, et intelligentes.

I.6. Vers une solution intelligente et universelle

Face aux limites des systèmes actuels et à l'évolution rapide des attentes des utilisateurs, il devient impératif de concevoir une solution de gestion des rendez-vous à la fois moderne, intelligente et universelle. Cette solution doit répondre à des exigences fonctionnelles et technologiques de plus en plus précises, en intégrant les principes suivants :

I.6.1 Multisectorielle

La solution doit être capable de s'adapter à différents domaines d'activité tels que la médecine, le droit, la beauté, le conseil ou encore l'éducation. Une architecture modulaire permettrait ainsi d'ajuster les fonctionnalités selon les besoins propres à chaque secteur [3].

I.6.2 Personnalisable

Elle doit offrir des interfaces flexibles, configurables par métier ou par utilisateur. L'objectif est de garantir une expérience utilisateur optimale et adaptée, que ce soit pour un avocat, un coiffeur ou un technicien. Cette personnalisation inclut les champs de réservation, les notifications, les horaires ou encore les couleurs de l'interface.

I.6.3 Intelligente

L'intégration de l'intelligence artificielle est un facteur différenciant majeur. Elle permettrait d'automatiser la proposition de créneaux en fonction des disponibilités, des préférences client, et de l'historique des interactions [2]. Une telle intelligence améliorerait la fluidité de la planification et réduirait le nombre de rendez-vous non honorés.

I.6.4 Basée sur le cloud (modèle SaaS)

L'hébergement dans le cloud assure une accessibilité continue, une maintenance simplifiée, une montée en charge fluide, et une sécurité renforcée [6]. En adoptant le modèle SaaS, la solution devient accessible depuis n'importe quel appareil connecté, sans installation locale, avec une gestion centralisée des données.

En combinant ces quatre dimensions — adaptation sectorielle, personnalisation, intelligence artificielle et architecture SaaS —, on obtient une plateforme robuste, évolutive et capable de répondre aux besoins concrets des professionnels comme des clients. C'est dans cette optique que s'inscrit le projet de développement présenté dans ce mémoire.

I.7. Conclusion

Dans ce premier chapitre, nous avons étudié l'évolution de la gestion des rendez-vous, en analysant les méthodes traditionnelles ainsi que les solutions numériques actuelles. Il est apparu clairement que si la digitalisation a permis de franchir un cap important en matière d'organisation et d'efficacité, les systèmes en place présentent encore de nombreuses limites : manque de personnalisation, faible intégration de l'intelligence artificielle, coûts d'abonnement peu adaptés, et difficulté de gestion multi-profils.

Ces constats soulignent la nécessité de développer une solution nouvelle, plus intelligente et plus flexible. Celle-ci doit non seulement répondre aux exigences technologiques actuelles, mais aussi s'adapter aux spécificités de chaque domaine professionnel. Une plateforme multisectorielle, personnalisable, basée sur le cloud (modèle SaaS) et intégrant un assistant intelligent constitue une réponse pertinente et innovante à ces besoins.

Le chapitre suivant présentera ainsi les fondements techniques et fonctionnels du modèle SaaS, ses avantages, ses contraintes, ainsi que les raisons pour lesquelles il représente le choix idéal pour notre projet.

Chapitre II : Le modèle SaaS : principes, avantages et application à la gestion des rendez-vous

Chapitre II : Le modèle SaaS : principes, avantages et application à la gestion des rendez-vous

II.1. Introduction

L'évolution des technologies de l'information et du cloud computing a profondément transformé la manière dont les applications logicielles sont développées, distribuées et utilisées. Dans ce contexte, le modèle SaaS (Software as a Service) est devenu un standard incontournable pour la mise à disposition de services numériques à distance. Dans ce chapitre, nous explorerons en profondeur le modèle SaaS : ses principes fondamentaux, ses avantages stratégiques, ses limites potentielles, et enfin son application spécifique à la problématique de la gestion des rendez-vous.

II.2. Définition du modèle SaaS

Le modèle SaaS (Software as a Service) est une approche moderne de la distribution logicielle basée sur le cloud computing. Il permet aux utilisateurs d'accéder à des applications hébergées à distance via Internet, sans nécessiter d'installation locale, ni de gestion technique de l'infrastructure sous-jacente. Ce modèle a profondément transformé la façon dont les entreprises consomment les services numériques.[1]

II.3. Fonctionnement général

Dans un environnement SaaS, l'ensemble des ressources nécessaires à l'exécution de l'application est centralisé du côté du fournisseur. Concrètement :

- Les ressources logicielles sont hébergées sur des serveurs distants, généralement dans des centres de données hautement sécurisés, appartenant au fournisseur de services (comme AWS, Google Cloud ou Azure) ;
- L'accès s'effectue à travers un navigateur web, depuis n'importe quel appareil connecté à Internet (ordinateur, smartphone, tablette), ce qui garantit une accessibilité ubiquitaire et multiplateforme ;
- Les mises à jour, correctifs de sécurité et évolutions fonctionnelles sont assurés par le fournisseur, sans que l'utilisateur final n'ait besoin d'intervenir manuellement, ce qui assure une continuité de service et une homogénéité entre les utilisateurs.

II.4. Caractéristiques principales

Le modèle SaaS présente plusieurs caractéristiques distinctives qui en font une solution particulièrement avantageuse pour la gestion des services à distance :

Chapitre II : Le modèle SaaS : principes, avantages et application à la gestion des rendez-vous

II.4.1 Modèle multi-tenant (mutualisation intelligente)

Une même instance du logiciel est utilisée par plusieurs clients (ou "tenants"). Chaque utilisateur dispose de son propre environnement de travail isolé logiquement, bien que techniquement partagé. Cela permet une optimisation des ressources, une scalabilité fluide et une réduction des coûts d'exploitation.

II.4.2 Modèle d'abonnement souple et économique

Les utilisateurs accèdent aux services via un abonnement mensuel ou annuel, souvent en fonction du volume d'utilisation ou du nombre de fonctionnalités activées. Ce modèle évite les coûts initiaux élevés liés à l'achat de licences logicielles, et permet une meilleure prévisibilité des dépenses.

II.4.3 Déploiement centralisé et maintenance simplifiée

Le fournisseur contrôle l'ensemble du cycle de vie de l'application (déploiement, mises à jour, sauvegardes, sécurité). L'utilisateur n'a pas à se soucier de l'installation locale, ni de la compatibilité logicielle, ce qui diminue les coûts informatiques internes et améliore l'expérience globale.

II.5. Avantages du modèle SaaS

L'adoption du SaaS présente de nombreux avantages pour les entreprises, notamment les PME et les prestataires de services indépendants.

II.5.1 Accessibilité accrue

L'utilisateur peut accéder à l'application depuis n'importe où, n'importe quand, sur n'importe quel appareil (ordinateur, tablette, smartphone) à l'aide d'un simple navigateur web.

II.5.2 Réduction des coûts

Le SaaS permet de :

- éviter l'achat de serveurs coûteux ;
- réduire les frais de maintenance technique ;
- bénéficier d'un coût prévisible grâce à l'abonnement.

II.5.3 Simplicité de maintenance

Les mises à jour sont déployées automatiquement, ce qui garantit à tous les utilisateurs un accès à la dernière version sans interruption de service.

Chapitre II : Le modèle SaaS : principes, avantages et application à la gestion des rendez-vous

II.5.4 Évolutivité

Les fournisseurs SaaS peuvent adapter les ressources système en fonction du nombre d'utilisateurs, de la charge ou des nouvelles fonctionnalités, sans impact sur l'utilisateur final.

II.5.5 Sécurité centralisée

Les données sont stockées sur des serveurs professionnels avec des protocoles de sécurité (chiffrement, authentification forte, sauvegardes régulières).

II.6. Application du SaaS à la gestion des rendez-vous

La gestion des rendez-vous est une activité fortement dépendante de la disponibilité en temps réel, de la coordination et de la personnalisation. Le modèle SaaS y répond efficacement.

II.6.1 Accès partagé et synchrone

Clients et prestataires peuvent accéder simultanément à la plateforme, consulter les créneaux disponibles, réserver, modifier ou annuler leurs rendez-vous, et ce à tout moment.

II.6.2 Gestion des disponibilités

Le professionnel configure ses jours et heures de travail, ses jours fériés, ses pauses... Le système ajuste automatiquement les créneaux en fonction de ces paramètres.

II.6.3 Notifications automatisées

Des rappels peuvent être envoyés automatiquement par e-mail ou SMS aux clients avant le rendez-vous pour réduire les oublis [4].

II.6.4 Tableaux de bord et analyses

La plateforme peut générer des rapports statistiques : nombre de rendez-vous, taux d'occupation, heures de pointe, etc. [1].

II.7. Exemples de plateformes SaaS existantes

II.7.1 Doctolib

Spécialisée dans le domaine médical, elle permet la prise de rendez-vous, la téléconsultation, et l'accès sécurisé aux données de santé. Elle est très populaire en France.

Chapitre II : Le modèle SaaS : principes, avantages et application à la gestion des rendez-vous

II.7.2 Calendly

Solution générique pour planifier des réunions ou des appels. Elle permet la synchronisation avec des calendriers Google ou Outlook.

II.7.3 Booknetic

Plugin WordPress qui permet aux entreprises de toutes tailles d'intégrer un système de réservation en ligne configurable.

II.7.4 Cal.com (ex Calendso)

Plateforme open source très flexible permettant une personnalisation complète et une intégration facile avec d'autres outils [7].

II.7.5 Limites observées

Malgré leur efficacité, ces outils ont souvent :

- une spécialisation sectorielle limitée ;
- peu d'intégration d'intelligence artificielle pour la suggestion proactive [2] ;
- des coûts d'abonnement peu accessibles pour les freelances ou micro-entreprises.

II.8. Justification du choix du SaaS pour notre projet

Notre plateforme repose sur le modèle SaaS car il est le plus adapté aux besoins suivants:

II.8.1 Support multi-profils et multi-secteurs

Chaque type d'utilisateur (médecin, avocat, esthéticienne...) peut accéder à une interface personnalisée et adaptée à son activité [3].

II.8.2 Centralisation et simplification

Toutes les données (rendez-vous, profils, horaires, historique) sont stockées de manière centralisée, ce qui simplifie la gestion quotidienne.

II.8.3 Intégration de l'intelligence artificielle

Un module IA est prévu pour proposer automatiquement les créneaux optimaux selon l'historique, les habitudes du client et les contraintes du prestataire [2].

II.8.4 Scalabilité et accessibilité

Chapitre II : Le modèle SaaS : principes, avantages et application à la gestion des rendez-vous

Notre solution peut être utilisée par une startup comme par une entreprise de plus grande taille, avec une montée en charge fluide et sans installation [1].

II.9. Conclusion

Le modèle SaaS s'impose comme une réponse technologique moderne, souple et économique à la problématique de la gestion des rendez-vous. Il permet d'offrir un service personnalisable, accessible à distance et intégrant des technologies avancées telles que l'intelligence artificielle. Dans le chapitre suivant, nous détaillerons la conception technique de notre solution, à travers les schémas UML, l'architecture logicielle et la structure des modules.

Chapitre III :

Conception du

Systeme

III.1. Introduction

Afin d'établir le développement d'une application, et en répondant aux besoins de l'utilisateur, nous allons dévoiler dans ce chapitre une méthodologie « Spécification des besoins » pour exprimer nos besoins puis les analyser. L'objectif est de permettre la formalisation des étapes préliminaires du développement du projet, pour avoir un système bien construit et bien organisé [1]. Nous allons commencer par présenter notre projet et ses différents objectifs, ainsi que la description du contexte du système.

Ensuite nous allons faire une étude préliminaire qui mène à l'élaboration des différents diagrammes (les diagrammes obtenus par les différentes phases du processus de développement). Nous utiliserons la méthodologie d'analyse et de conception qui permet de décrire de manière non ambiguë, en utilisant une méthode de modélisation UML (Unified Modeling Language) et le processus de développement UP (Unified Process).

III.2. Concepts de base de développement

III.2.1 Langage de modélisation UML

III.2.1.1 Définition d'UML

UML, c'est l'acronyme anglais pour “Unified Modeling Language”. On le traduit par “Langage de modélisation unifié”. La notation UML est un langage visuel constitué d'un ensemble de schémas, appelés des diagrammes, qui donnent chacun une vision différente du projet à traiter. UML nous fournit donc des diagrammes pour représenter le logiciel à développer : son fonctionnement, sa mise en route, les actions susceptibles d'être effectuées par le logiciel, etc.[11]

III.2.1.2 Les diagrammes d'UML

- **Diagramme de cas d'utilisation** : est un diagramme UML qui représente une vision globale du comportement d'un système. Il montre les interactions fonctionnelles entre les acteurs et le système d'étude. **Acteur** : rôle joué par un utilisateur humain ou un autre système qui interagit directement avec le système d'étude. Un acteur participe à au moins un cas d'utilisation. **Cas d'utilisation (use case)** : ensemble d'actions réalisées par le système produisant un résultat observable intéressant pour un acteur particulier. Il peut être défini comme une collection de scénarios reliés par un objectif utilisateur commun.[11]

Chapitre III : Conception du Système

- **Diagramme de classe** : Le diagramme de classe est un schéma utilisé en génie logiciel pour présenter les classes et les interfaces des systèmes ainsi que les différentes relations entre celles-ci. Ce diagramme fait partie de la partie statique d'UML car il fait abstraction des aspects temporels et dynamiques.[11]
- **Diagramme d'activité** : Les diagrammes d'activité sont utilisés pour documenter le déroulement des opérations dans un système, du niveau commercial au niveau opérationnel (de haut en bas). Le diagramme d'activité est une variante du diagramme d'état où les "états" représentent des opérations, et les transitions représentent les activités qui se produisent quand l'opération est terminée.[11]
- **Diagramme de séquence** : Il permet d'étudier les interactions entre les objets et constitue une partie dynamique du système d'information. Il montre d'une façon séquentielle les envois des messages qui interviennent entre les objets, il peut également montrer les flux des données.[11]

III.2.1.3 Caractéristiques d'UML

- **UML est basé sur un méta-modèle** : UML est un moyen d'exprimer des modèles objet en faisant abstraction de leur implémentation, c'est-à-dire que le modèle fourni par UML est valable pour n'importe quel langage de programmation. UML est un langage qui s'appuie sur un métamodèle, un modèle de plus haut niveau qui définit les éléments d'UML (les concepts utilisables) et leur sémantique (leur signification et leur mode d'utilisation). Le méta-modèle permet de se placer à un niveau d'abstraction supérieur car il est étudié pour être plus générique que le modèle qu'il permet de construire. Le méta-modèle d'UML en fait un langage formel possédant les caractéristiques suivantes :
 - Un langage sans ambiguïtés
 - Un langage universel pouvant servir de support pour tout langage orienté objet
 - Un moyen de définir la structure d'un programme
 - Une représentation visuelle permettant la communication entre les acteurs d'un même projet
 - Une notation graphique simple, compréhensible même par des non informaticiens.Le méta-modèle permet de donner des bases solides et rigoureuses à ce langage graphique, dont les représentations graphiques ne sont là que pour véhiculer des concepts de réalisation.[12]
- **UML comme visualisation complète d'un système** : UML offre une manière élégante de représenter le système selon différentes vues complémentaires grâce aux diagrammes.

Chapitre III : Conception du Système

- **UML n'est pas une méthode ou processus** : Si l'on parle de méthode objet pour UML, c'est par abus de langage ! Une méthode propose aussi un processus, qui régit notamment l'enchaînement des activités de production d'une entreprise.
- **UML est un support de communication** : Sa notation graphique permet d'exprimer visuellement une solution objet.

III.2.1.4 Les avantages d'UML

- Gain de précision.
- Gage de stabilité.
- Beaucoup d'utilisation d'outils.
- UML est un support de communication performant.
- Il cadre l'analyse et facilite la compréhension de représentations abstraites complexes.

III.2.1.5 Inconvénients d'UML

- La mise en pratique d'UML nécessite un apprentissage et passe par une période d'adaptation.
- Le processus (non couvert par UML) est une autre clé de la réussite d'un projet.

III.3. Les acteurs du système

Un acteur est une personne extérieure au système en cours de modélisation. L'acteur peut consulter ou modifier l'état du système. Dans le tableau ci-dessous, nous identifions les différents acteurs de notre système.

Tableau III.1 : Les acteurs du système

Les acteurs	Les rôles dans le système
Client	<ul style="list-style-type: none">– Recherche un prestataire par domaine, ville ou disponibilité.– Consulte les créneaux horaires proposés.– Réserve, modifie ou annule un rendez-vous.– Reçoit des rappels automatiques.– Peut recevoir des suggestions de créneaux via l'assistant IA.
Prestataire de service (Service Provider)	<ul style="list-style-type: none">– Crée son compte et complète son profil professionnel.– Configure ses disponibilités : jours, heures, pauses, congés.– Reçoit les demandes de réservation et peut les accepter ou les refuser.– Accède à un tableau de bord affichant les rendez-vous à venir.– Peut consulter des statistiques (nombre de réservations, créneaux préférés...).

Chapitre III : Conception du Système

Assistant IA	<ul style="list-style-type: none">– Analyse les disponibilités des prestataires.– Suggère des créneaux optimisés selon la charge, les préférences, et les habitudes.
Administrateur	<ul style="list-style-type: none">- Créer, modifier, désactiver ou supprimer des comptes (clients, prestataires).

III.4. Analyse des besoins

III.4.1 Besoins fonctionnels

- L'utilisateur peut créer un compte et se connecter.
- Le client peut rechercher un prestataire par domaine, localisation ou disponibilité.
- Le client peut réserver, modifier ou annuler un rendez-vous.
- Le prestataire peut configurer ses jours et horaires de disponibilité.
- Le prestataire reçoit, accepte ou refuse les demandes de rendez-vous.
- Le système envoie des rappels automatiques (e-mail ou SMS) .
- Le module d'intelligence artificielle propose des créneaux de rendez-vous optimaux.

III.4.2 Besoins non fonctionnels

- **Sécurité** : Authentification sécurisée, confidentialité des données [6].
- **Accessibilité** : Interface responsive adaptée aux mobiles et tablettes.
- **Performance** : Temps de réponse rapide même avec un grand nombre d'utilisateurs.
- **Scalabilité** : Adaptation à la montée en charge progressive [1].

III.5. Diagramme de cas d'utilisation

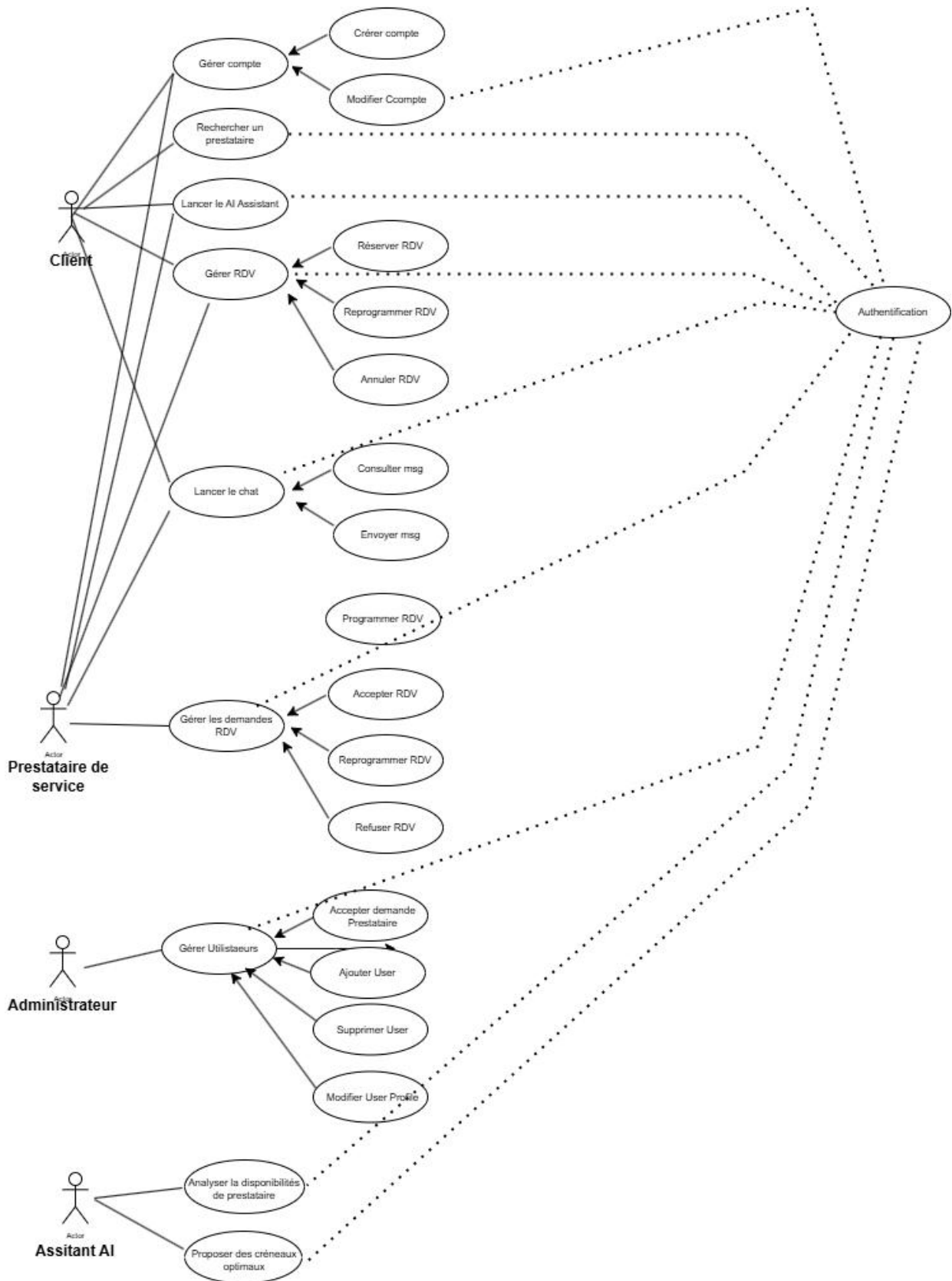


Figure III.1 : Diagramme de cas d'utilisation

Chapitre III : Conception du Système

III.5.1 Descriptions textuelles détaillées

III.5.1.1 Créer un compte

Tableau III.2: Description textuelle de cas d'utilisation : Créer un compte

Cas d'utilisation	Créer un compte
Type de cas	Primaire
Acteur principal	Client / Prestataire
Objectif	Permettre à l'utilisateur de s'inscrire à la plateforme
Pré-condition	Accès à la page d'inscription
Déroulement	1. Saisie des informations 2. Validation 3. Création du compte
Scénarios alternatifs	Email déjà utilisé → afficher une erreur
Exception	Connexion perdue avec le serveur → message d'échec

III.5.1.2 Modifier un compte

Tableau III.3: Description textuelle de cas d'utilisation : Modifier un compte

Cas d'utilisation	Modifier un compte
Type de cas	Primaire
Acteur principal	Client
Objectif	Mettre à jour les données personnelles (email, mot de passe, etc)
Pré-condition	L'utilisateur est authentifié
Déroulement	Accès à son profil → Modification → Enregistrement
Scénarios alternatifs	Données invalides → blocage de modification
Exception	Échec de sauvegarde → message d'erreur

Chapitre III : Conception du Système

III.5.1.3 Rechercher un prestataire

Tableau III.4: Description textuelle de cas d'utilisation : Rechercher un prestataire

Cas d'utilisation	Rechercher un prestataire
Type de cas	Primaire
Acteur principal	Client
Objectif	Trouver un prestataire par spécialité, p a y s , ou ville
Pré-condition	Être connecté et accéder à la page de recherche
Déroulement	1. Entrée des critères 2. Affichage des résultats 3. Sélection d'un profil
Scénarios alternatifs	Aucun résultat → message d'alerte
Exception	API de recherche hors service → afficher une erreur

III.5.1.4 Réserver un RDV

Tableau III.5: Description textuelle de cas d'utilisation : Réserver un RDV

Cas d'utilisation	Créer un RDV
Type de cas	Primaire
Acteur principal	Client / Prestataire
Objectif	Réserver un créneau auprès d'un prestataire
Pré-condition	Être connecté et avoir consulté le profil du prestataire
Déroulement	Choix du créneau → Réservation → Confirmation
Scénarios alternatifs	Créneau indisponible → mise à jour de la liste
Exception	Erreur de base de données → réservation non enregistrée

Chapitre III : Conception du Système

III.5.1.5 Reprogrammer un RDV

Tableau III.6: Description textuelle de cas d'utilisation : Reprogrammer un RDV

Cas d'utilisation	Reprogrammer un RDV
Type de cas	Primaire
Acteur principal	Client / Prestataire
Objectif	Modifier la date ou l'heure d'un rendez-vous existant
Pré-condition	RDV déjà confirmé
Déroulement	Choix d'un nouveau créneau → Validation
Scénarios alternatifs	Aucune autre disponibilité → alerte utilisateur
Exception	Surcharge du serveur → tentative échouée

III.5.1.6 Annuler un RDV

Tableau III.7: Description textuelle de cas d'utilisation : Annuler un RDV

Cas d'utilisation	Annuler un RDV
Type de cas	Primaire
Acteur principal	Client / Prestataire
Objectif	Supprimer une réservation existante
Pré-condition	Être connecté
Déroulement	Sélection du RDV → Annuler → Confirmation
Scénarios alternatifs	RDV déjà passé → non modifiable
Exception	Connexion instable → annulation non effectuée

Chapitre III : Conception du Système

III.5.1.7 Lancer AI (Assistant AI)

Tableau III.8: Description textuelle de cas d'utilisation : Lancer AI

Cas d'utilisation	Lancer AI
Type de cas	Primaire
Acteur principal	Client / Prestataire
Objectif	Proposer automatiquement des prestataire de services et des créneaux optimaux à l'utilisateur
Pré-condition	Demande explicite du client ou absence de choix de créneau
Déroulement	Analyse IA → Affichage des suggestions → Choix client
Scénarios alternatifs	Aucun créneau trouvé → message "Essayez un autre jour"
Exception	Module IA inactif → message technique

III.5.1.8 Gérer les demandes de RDV

Tableau III.9: Description textuelle de cas d'utilisation : Gérer les demandes de RDV

Cas d'utilisation	Gérer les demandes de RDV
Type de cas	Primaire
Acteur principal	Prestataire
Objectif	Traiter les nouvelles demandes de réservation
Pré-condition	Accès au tableau des demandes
Déroulement	Affichage des demandes → Action choisie (accepter/ refuser/ reporter)
Scénarios alternatifs	Client annule le RDV avant validation
Exception	Mauvaise synchronisation → statut du RDV incorrect

Chapitre III : Conception du Système

III.5.1.9 Lancer le chat

Tableau III.10: Description textuelle de cas d'utilisation : Lancer le chat

Cas d'utilisation	Lancer le chat
Type de cas	Primaire
Acteur principal	Client/ Prestataire
Objectif	Communiquer autour d'un rendez-vous ou d'un service
Pré-condition	Être connecté et avoir un RDV en cours ou passé
Déroulement	Accès à la messagerie → envoi ou consultation de Messages
Scénarios alternatifs	Aucune réponse de l'autre partie → rappel automatique
Exception	Serveur de messagerie hors ligne → erreur temporaire

III.5.1.10 Gérer Utilisateurs

Tableau III.11: Description textuelle de cas d'utilisation : Gérer Utilisateurs

Cas d'utilisation	Gérer Utilisateur
Type de cas	Primaire
Acteur principal	Administrateur
Objectif	Gérer les profils clients, prestataires
Pré-condition	Être connecté au tableau de bord
Déroulement	Ajouter ou modifier les données du client ou prestataires depuis l'espace « gestion des utilisateurs
Scénarios alternatifs	Profil déjà existant → demande de confirmation
Exception	Erreur d'écriture en base → opération échouée

III.6. Diagramme de séquence :

Le diagramme de séquence est la représentation graphique des interactions entre les acteurs et le système selon un ordre chronologique dans la formulation "Unified Modeling Language". Il permet de cacher les interactions d'objets dans le cadre d'un scénario d'un diagramme des cas d'utilisation. Dans un souci de simplification, on représente l'acteur principal à gauche du diagramme, et les acteurs secondaires éventuels à droite du système. Le but étant de décrire comment se déroulent les actions entre les acteurs ou objets.

Chapitre III : Conception du Système

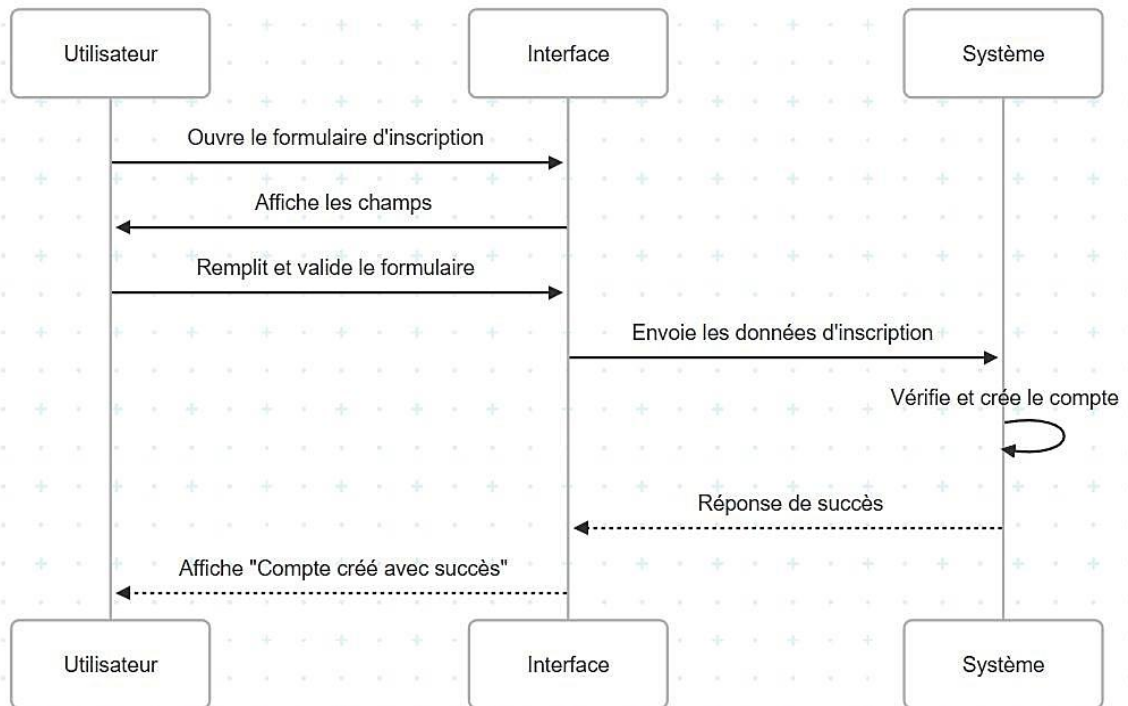


Figure III.2 : Créer un compte

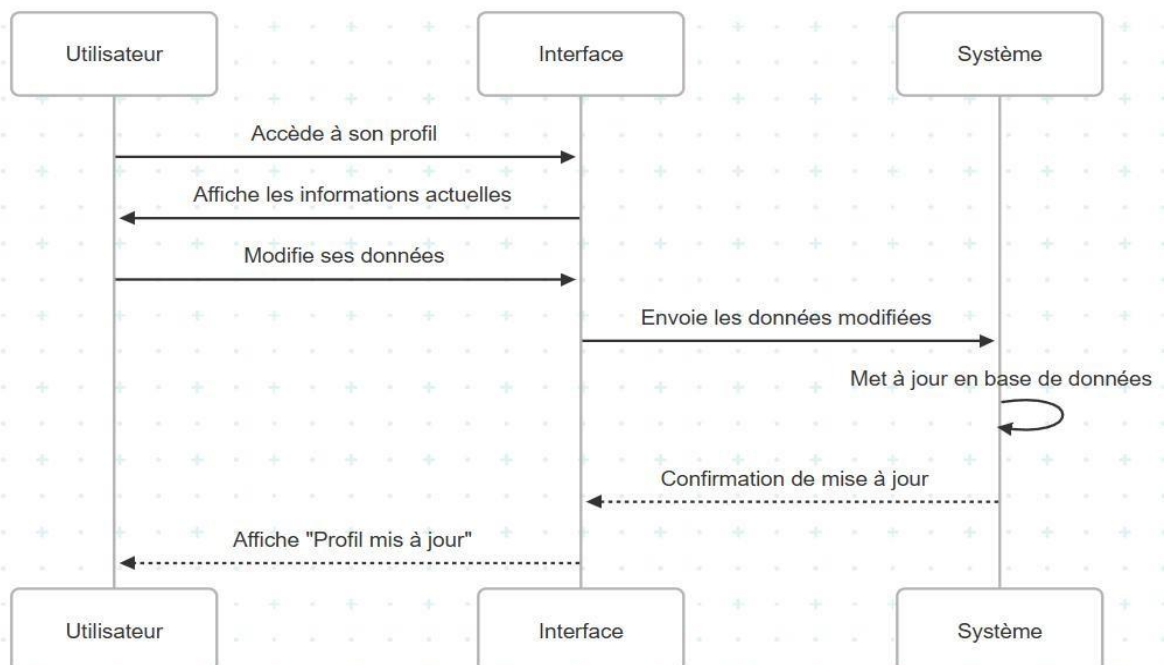


Figure III.3: Modifier un compte

Chapitre III : Conception du Système

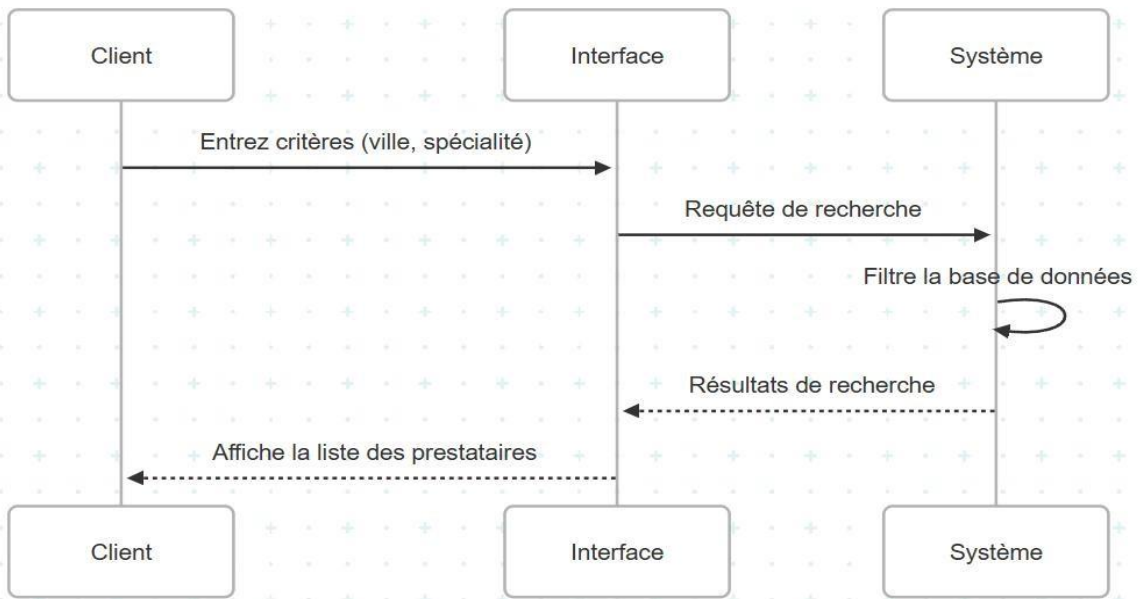


Figure III.4: Recherche un prestataire

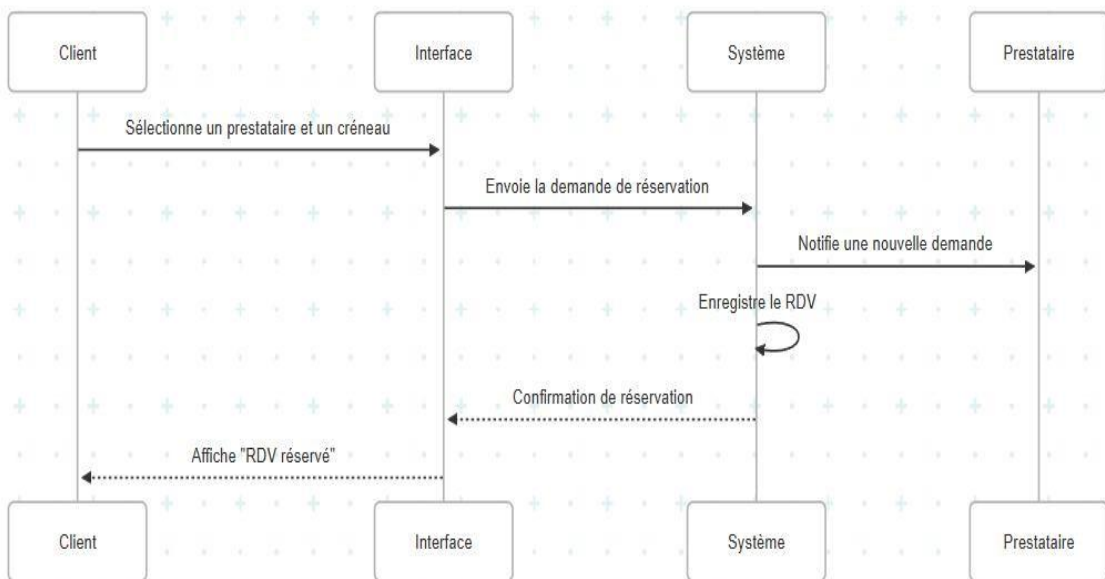


Figure III.5: Créer un RDV

Chapitre III : Conception du Système

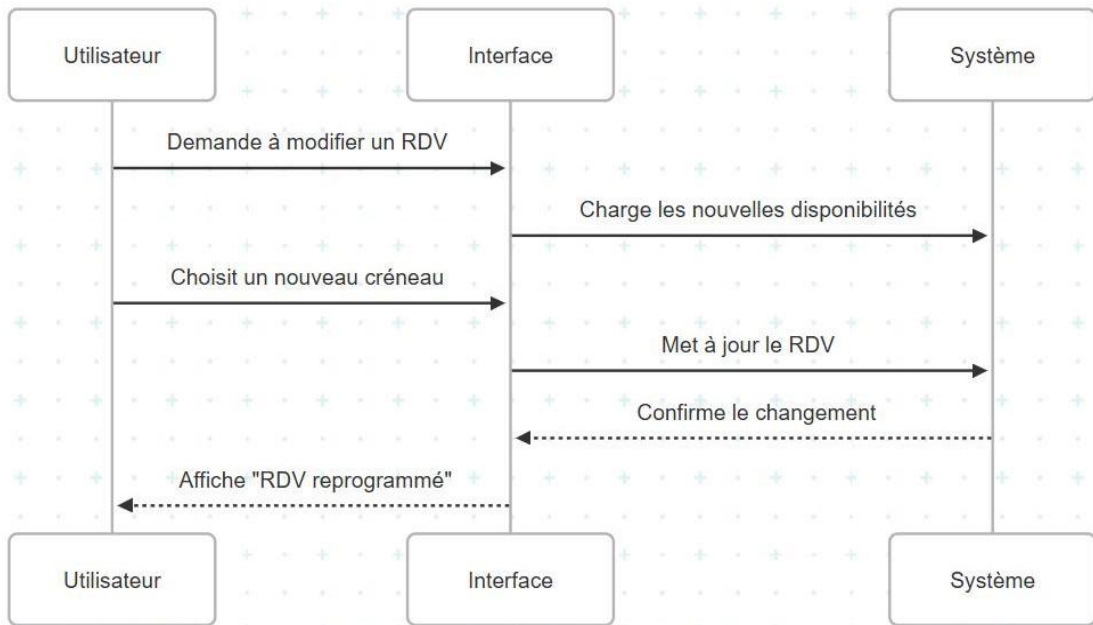


Figure III.6: Reprogrammer un RDV

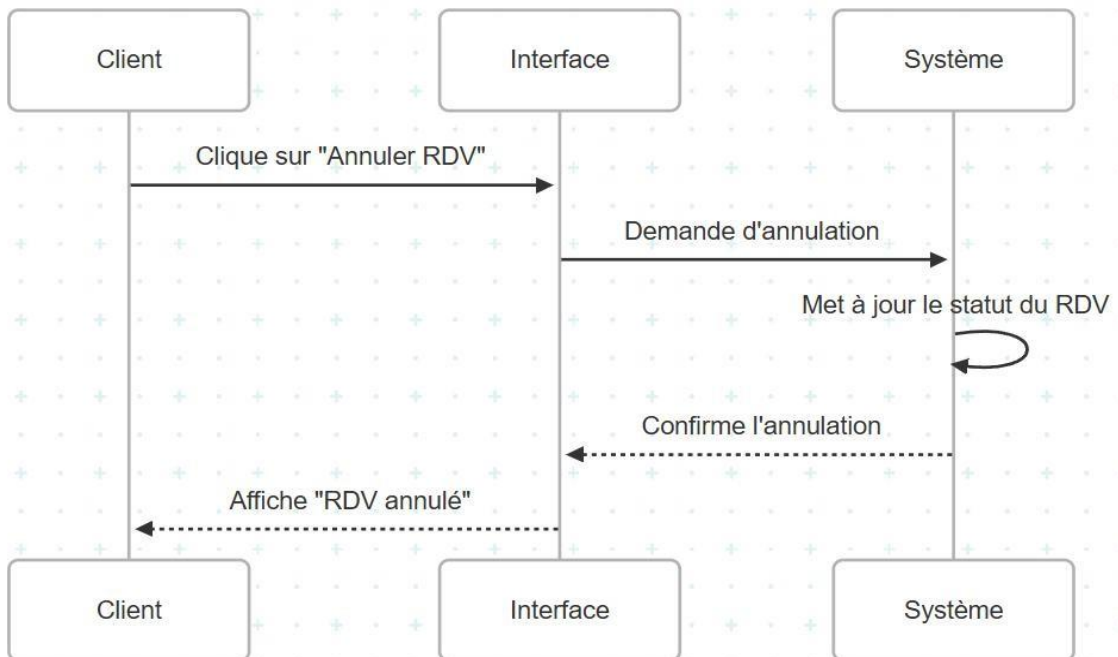


Figure III.7: Annuler un RDV

Chapitre III : Conception du Système

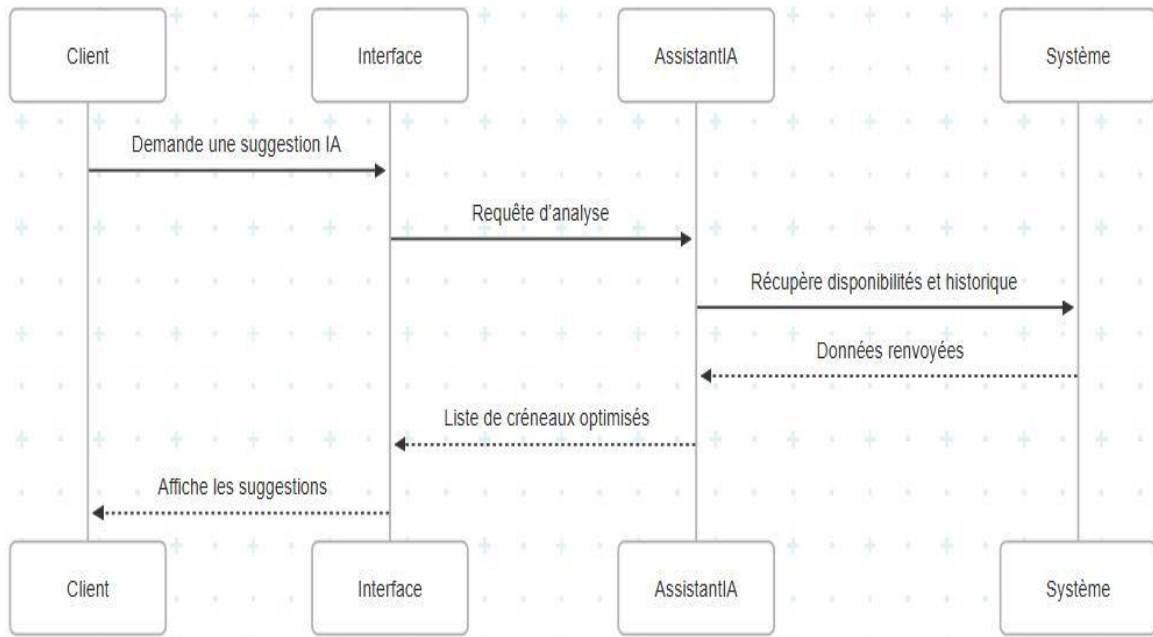


Figure III.8: Lancer AI

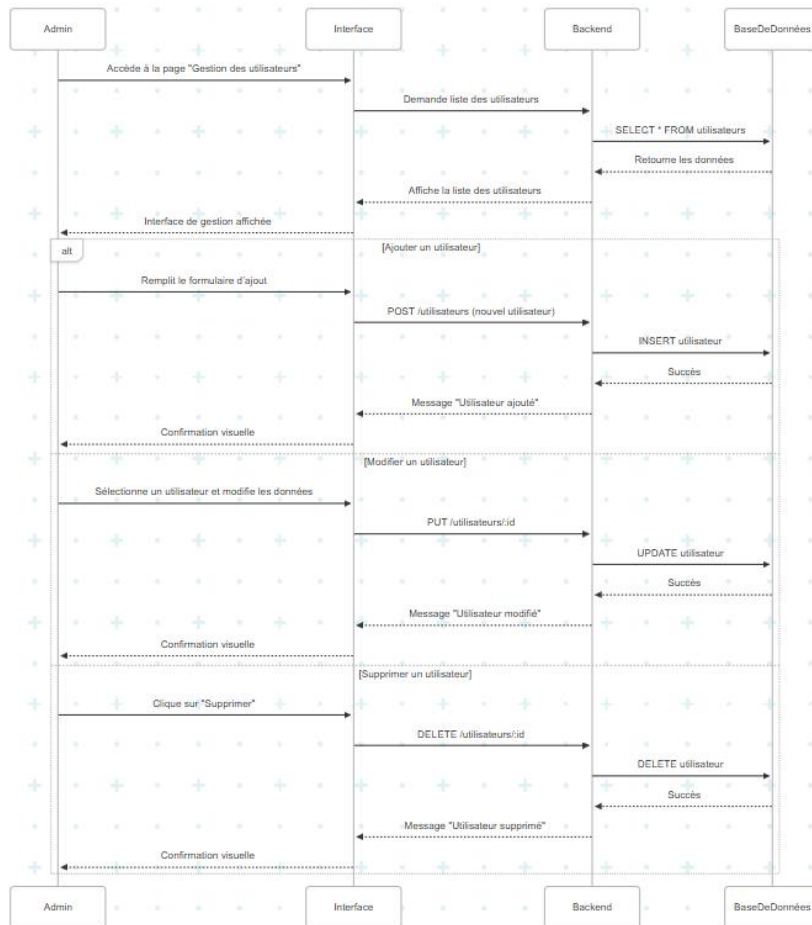


Figure III.9: Gérer utilisateurs

Chapitre III : Conception du Système

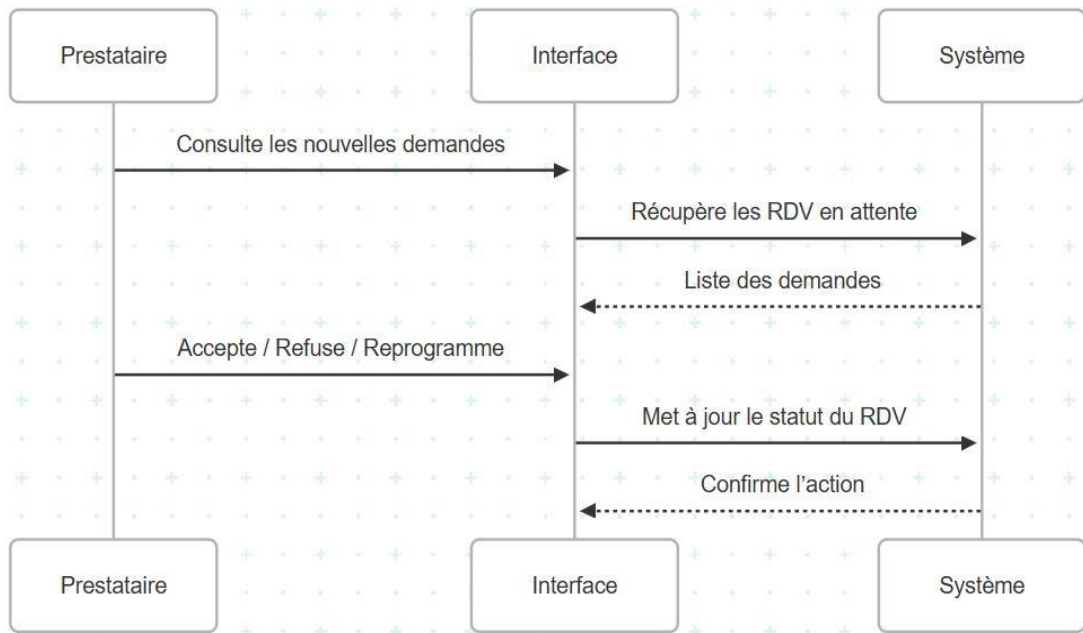


Figure III.10: Gérer Les demandes de RDV

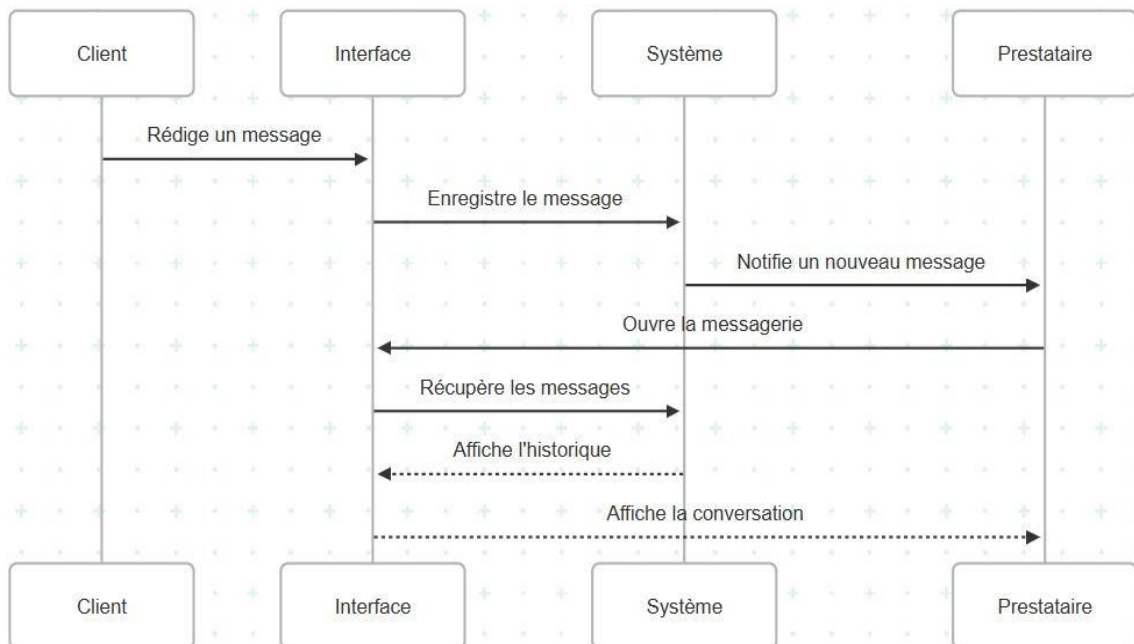


Figure III.11: Lancer le chat

Chapitre III : Conception du Système

III.7. Diagramme de classe

Le diagramme de classes est le point central dans le développement de notre projet. Ce diagramme représente la description statique du système en intégrant dans chaque classe la partie dédiée aux données et celle consacrée aux traitements. C'est le diagramme pivot de l'ensemble de la modélisation d'un système.

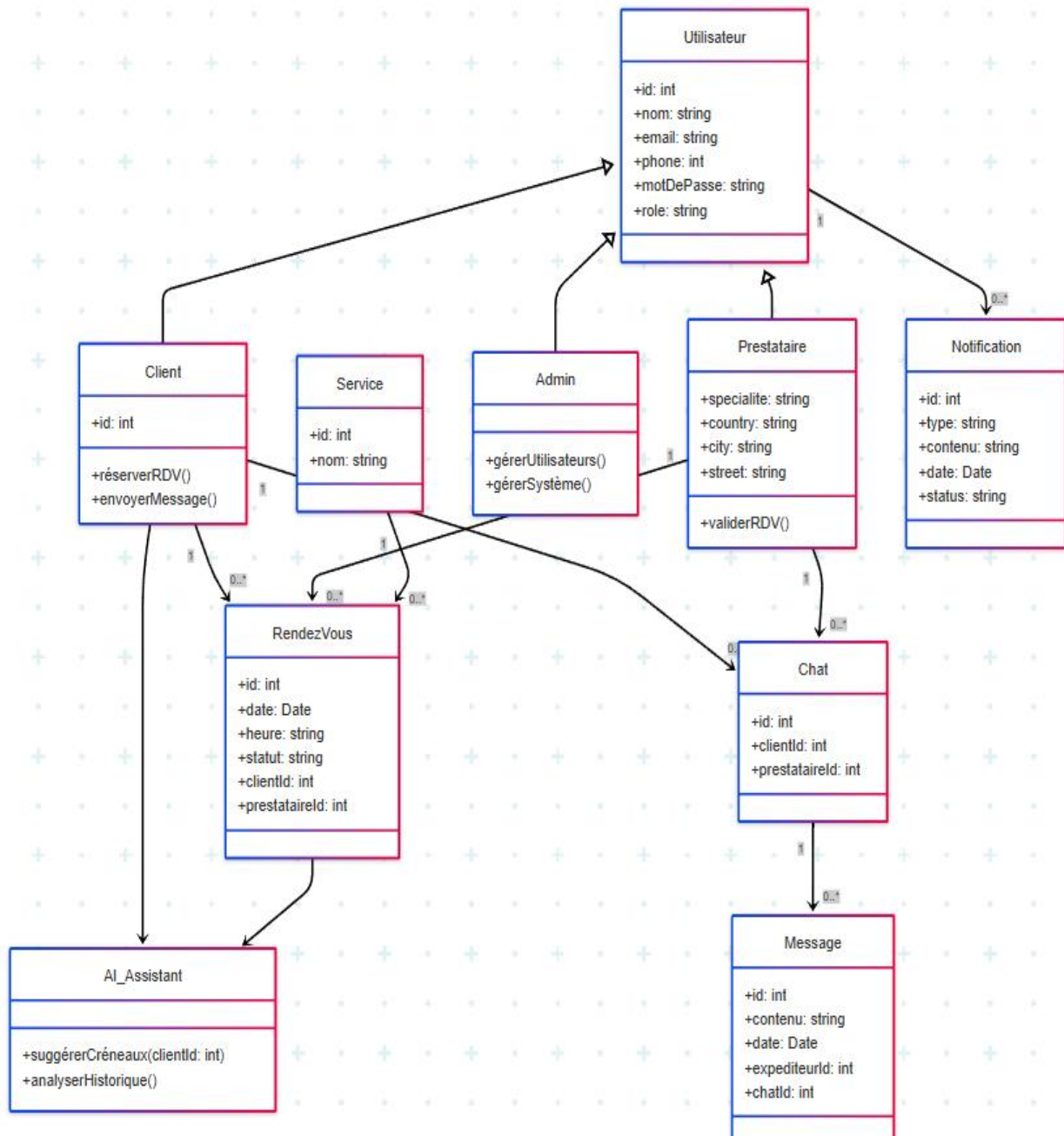


Figure III.12: Diagramme de classe

III.8. Diagramme d'activité

Les diagrammes d'activités permettent de mettre l'accent sur les traitements. Ils sont donc particulièrement adaptés à la modélisation du cheminement de flots de contrôle et de flots de données. Ils permettent ainsi de représenter graphiquement le comportement d'une méthode ou le déroulement d'un cas d'utilisation.

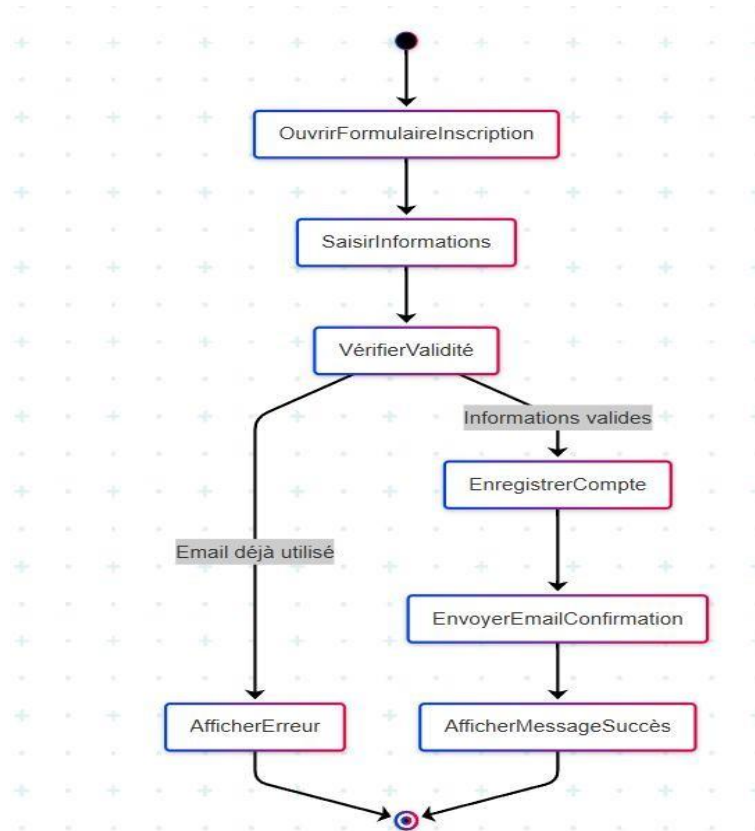


Figure III.13: Créer un compte

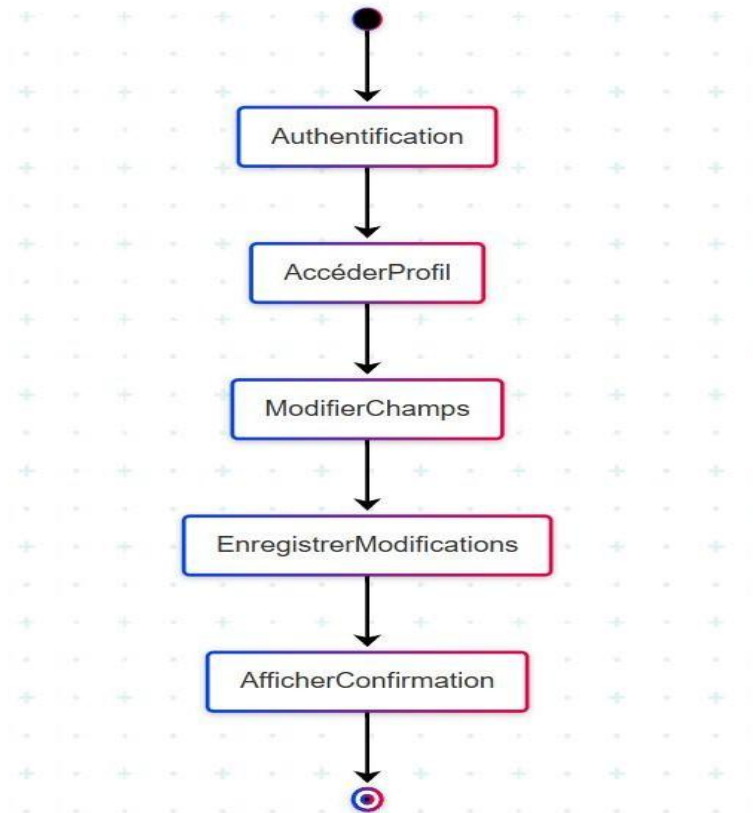


Figure III.14 : Modifier un compte

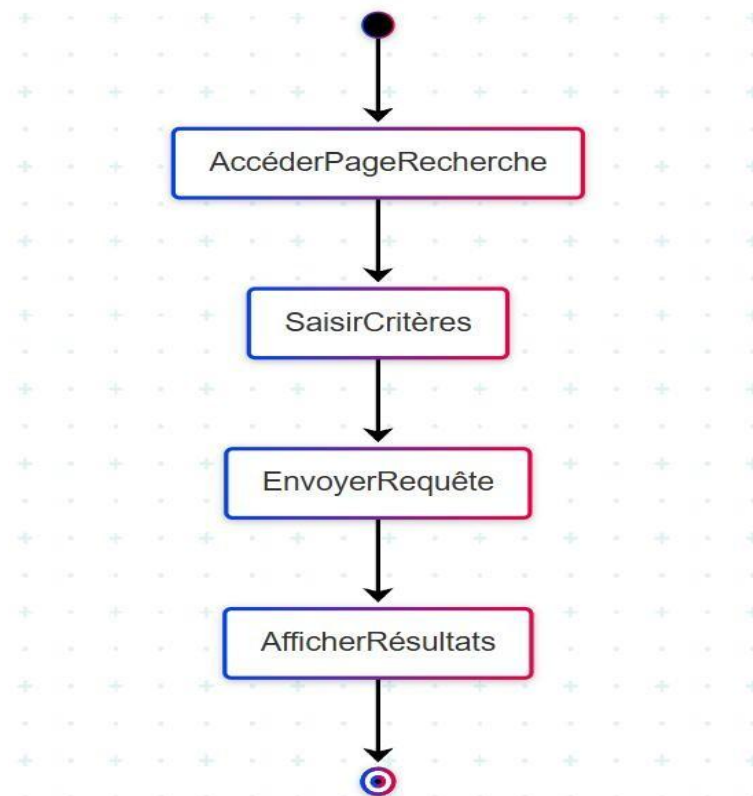


Figure III.15: Recherche un prestataire

Chapitre III : Conception du Système

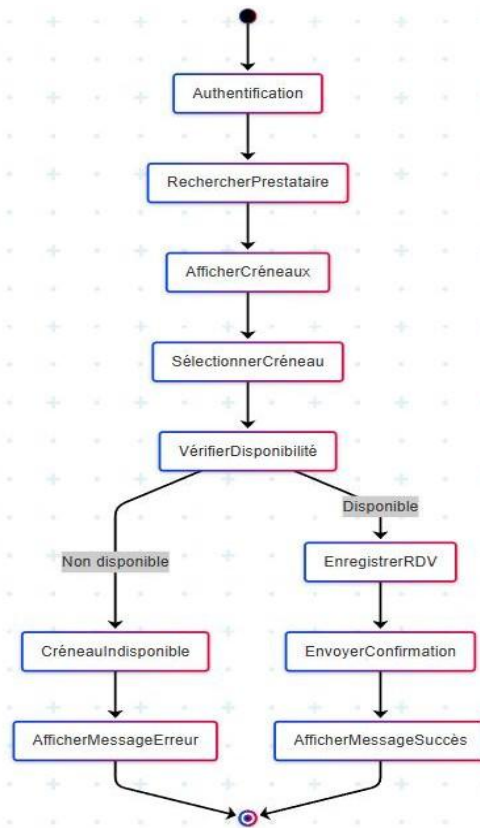


Figure III.16: Créer un RDV

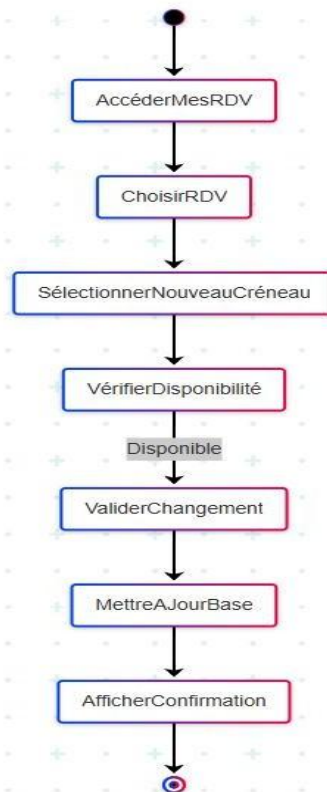


Figure III.17: Reprogrammer un RDV

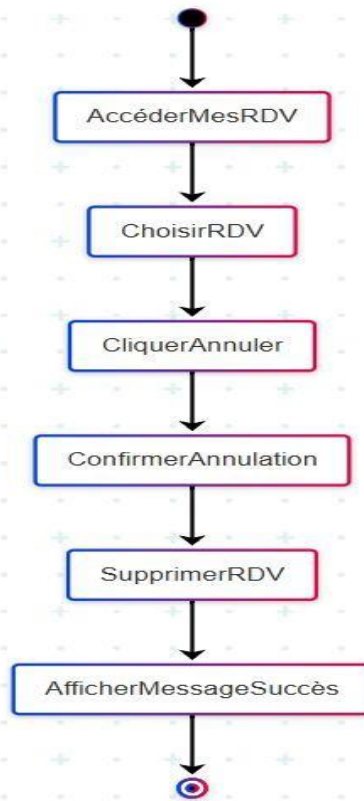


Figure III.18: Annuler un RDV

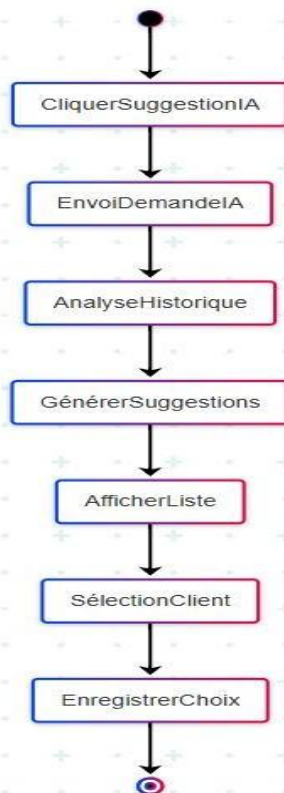


Figure III.19: Lancer IA

Chapitre III : Conception du Système

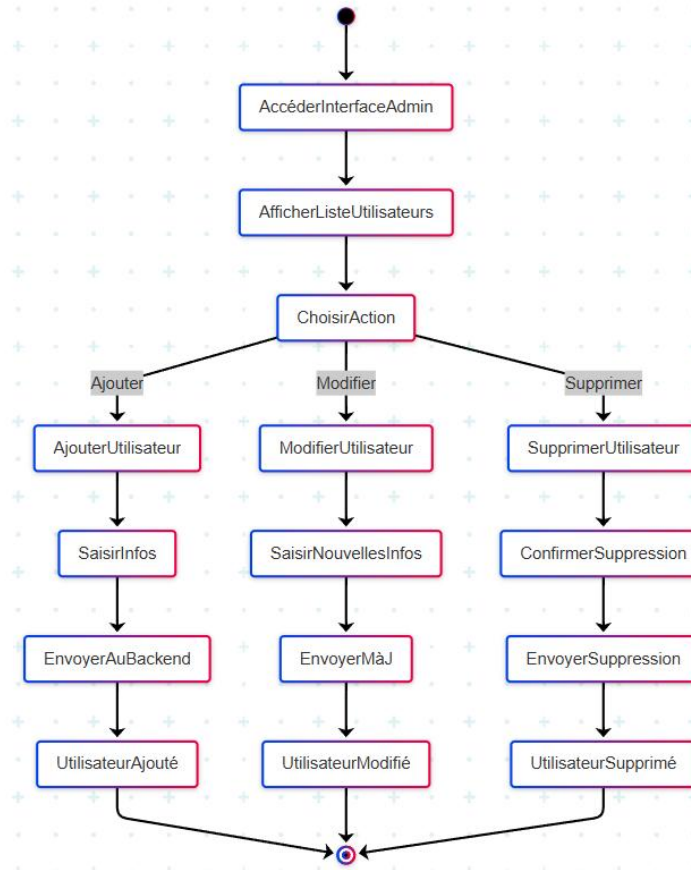


Figure III.20: Gérer utilisateur

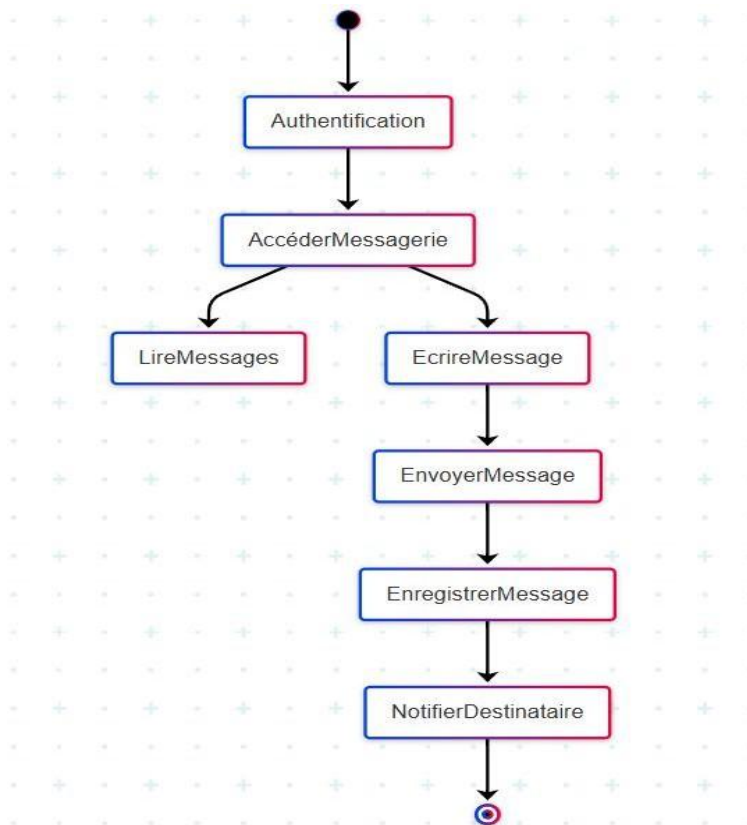


Figure III.21: Lancer le chat

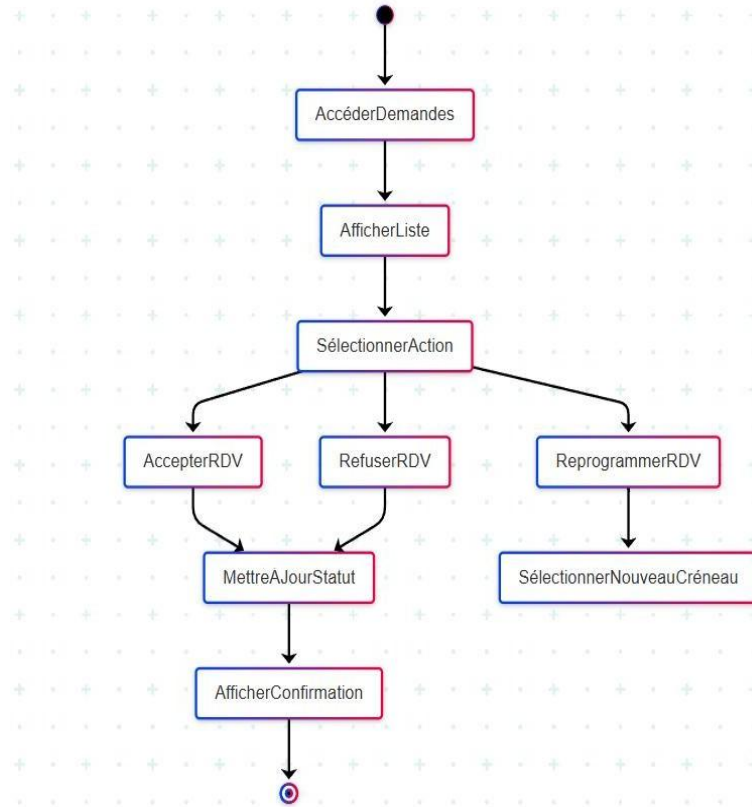


Figure III.22: Gérer les demandes de RDV

III.9. Conclusion

Ce chapitre a permis de définir l'architecture fonctionnelle et technique de la plateforme SaaS de gestion de rendez-vous. À partir de l'analyse des besoins exprimés par les utilisateurs cibles (clients, prestataires, administrateur, Assistant AI), nous avons construit une série de modèles UML cohérents, tels que le diagramme de cas d'utilisation, les diagrammes de séquence, les diagrammes de classes et les diagrammes d'activités.

Ces représentations graphiques ont facilité la formalisation des interactions, des traitements et des entités principales du système. La modélisation orientée objet a permis d'assurer une bonne structuration des composants, tout en préparant l'implémentation sur des bases solides et réutilisables.

Le modèle relationnel de la base de données a également été conçu de manière à refléter fidèlement la structure métier du système, en tenant compte des relations entre utilisateurs, rendez-vous, messages, services et suggestions intelligentes.

Cette phase de conception fournit donc une vue claire, rigoureuse et détaillée du fonctionnement du système, servant de fondement direct au développement technique abordé dans le chapitre suivant.

Chapitre IV :

Implémentation

Chapitre IV : Implémentation

IV.1. Introduction

Ce chapitre décrit l'implémentation concrète de la solution de gestion des rendez-vous développée dans le cadre de ce projet de fin d'études. Après avoir défini les besoins fonctionnels et conçu l'architecture du système dans les chapitres précédents, l'objectif ici est de présenter les différentes étapes techniques ayant conduit à la mise en œuvre de la plateforme.

L'accent sera mis sur l'environnement de développement utilisé, les technologies choisies, la structure technique de l'application ainsi que les interfaces principales destinées aux utilisateurs.

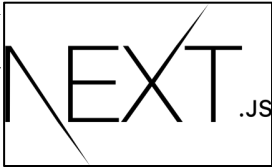
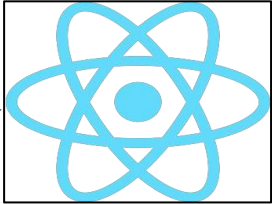
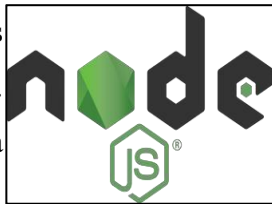
Un aperçu du fonctionnement du module d'intelligence artificielle sera également fourni, en plus des modalités de déploiement de la solution et des tests réalisés.

IV.2. Environnement de développement

L'environnement de développement constitue l'ensemble des outils logiciels, plateformes et configurations nécessaires à la réalisation du projet. Pour développer une solution SaaS de gestion intelligente de rendez-vous, il a été essentiel de mettre en place une infrastructure moderne, modulaire et adaptée au développement web **full-stack**.

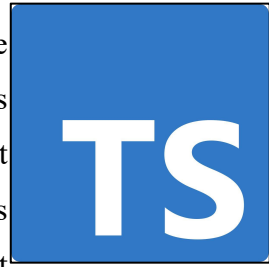
IV.2.1 Plateforme de développement

Le projet a été développé principalement sur un environnement local à l'aide des outils suivants:

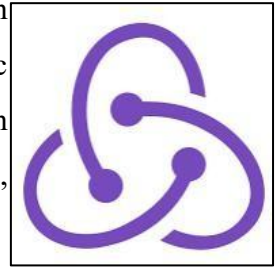
- **Next.js** : Next.js est un cadre (framework) gratuit et open source s'appuyant sur la librairie JavaScript React et sur la technologie Node.js. Nous avons utilisé la version 8.5.0.[11] 
- **React** : React (aussi appelé React.js ou ReactJS) est une bibliothèque JavaScript libre développée par Facebook depuis 2013. Le but principal de cette bibliothèque est de faciliter la création d'application web monopage, via la création de composants dépendant d'un état générant une page (ou portion) HTML à chaque changement d'état. Nous avons utilisé la version 18.1.0.[11] 
- **Node.js** : Node.js est une plateforme logicielle libre en JavaScript, orientée vers les applications réseau événementielles hautement concurrentes qui doivent pouvoir monter en charge. Elle utilise la machine virtuelle V8, la librairie libuv pour sa boucle d'événements, et implémente sous licence. [11] 

Chapitre IV : Implémentation

- **TypeScript** : TypeScript est un langage de programmation open source développé par Microsoft. Il s'agit d'une surcouche de JavaScript qui ajoute un système de typage statique et des fonctionnalités orientées objet avancées. Le code TypeScript est transcompilé en JavaScript, ce qui permet son exécution dans n'importe quel navigateur ou environnement Node.js. Il est largement utilisé pour le développement d'applications web robustes et évolutives.[11]



- **Redux** : Redux est une bibliothèque open-source JavaScript de gestion d'état pour applications web. Elle est plus couramment utilisée avec des bibliothèques comme React ou Angular pour la construction d'interfaces utilisateur. Semblable à (et inspirée de) l'architecture Flux, elle a été créée par Dan Abramov et Andrew Clark. [11]



- **MongoDB** : MongoDB est un système de gestion de base de données orienté documents, répartisable sur un nombre quelconque d'ordinateurs et ne nécessitant pas de schéma prédéfini des données. Il est écrit en C++. Le serveur et les outils sont distribués sous licence SSPL, les pilotes sous licence Apache et la documentation sous licence Creative Commons. Il fait partie de la mouvance NoSQL.[11]

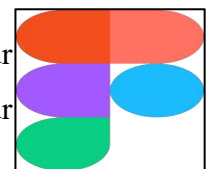


IV.2.2 Les plateformes / outils utilisés

- **Mermaid Charts** : Mermaid est un outil de diagramming et de cartographie basé sur JavaScript qui utilise une syntaxe inspirée de Markdown pour créer des diagrammes dynamiques. Il permet de visualiser des données et des processus complexes sous forme de graphiques, schémas, diagrammes de flux, diagrammes de séquence, etc.[11]



- **Figma** : Figma est un outil de conception d'interface (UI/UX) basé sur le cloud, qui permet de créer, prototyper et collaborer en temps réel sur des maquettes graphiques, sans installation locale.[11]

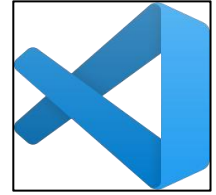


- **Word** : Microsoft Word est un logiciel de traitement de texte publié par Microsoft. Word a été intégré en tant qu'élément de la suite Microsoft Office depuis 1993, même s'il était également vendu seul ou inclus dans la suite Microsoft Works. Nous avons utilisé la version 2010.[11]



Chapitre IV : Implémentation

- **Microsoft Visual Studio Code** : Code est un éditeur de code extensible développé par Microsoft pour Windows, Linux et macOS. Les fonctionnalités incluent la prise en charge du débogage, la mise en évidence de la syntaxe, la complétion intelligente du code, les snippets, la refactorisation du code et Git intégré. Les utilisateurs peuvent modifier le thème, les raccourcis clavier, les préférences et installer des extensions qui ajoutent des fonctionnalités supplémentaires.[11]



IV.3. Intégration du module IA

L'intelligence artificielle joue un rôle clé dans notre plateforme SaaS, en apportant une couche de recommandation intelligente des créneaux de rendez-vous. Le module IA a été conçu comme un composant indépendant, afin de garantir sa maintenabilité, sa réutilisabilité et sa scalabilité.

IV.3.1 Architecture du module IA

Le module IA a été développé en Python, en utilisant le framework léger et performant **FastAPI**, qui permet de créer rapidement des API RESTful asynchrones. Ce service fonctionne comme une API séparée, que le backend principal peut interroger à tout moment.

Ce découplage entre la logique métier (Node.js) et la logique IA (Python) facilite:

- la maintenance du code,
- l'évolution des modèles d'intelligence artificielle indépendamment du reste de l'application [2],
- et le déploiement sur des environnements spécifiques aux besoins de l'IA (GPU, containers, etc.) [1].

IV.3.2 Fonctionnalités du module IA

Le module d'IA est responsable des opérations suivantes:

- **Analyse de l'historique des rendez-vous** : Il collecte les données des rendez-vous passés pour identifier les comportements fréquents, comme les horaires préférés d'un client, les taux d'annulation ou les plages sous-utilisées.
- **Identification des créneaux optimaux** : En croisant les disponibilités du prestataire et les habitudes du client, le module IA sélectionne les créneaux les plus adaptés, tout en évitant les surcharges et en équilibrant les périodes creuses.
- **Retour de suggestions classées** : Les créneaux générés sont classés par pertinence (score) selon plusieurs critères : disponibilité réelle, correspondance avec les préférences,

Chapitre IV : Implémentation

historique, etc. Ces suggestions sont renvoyées vers le backend qui les transmet à l'interface utilisateur.

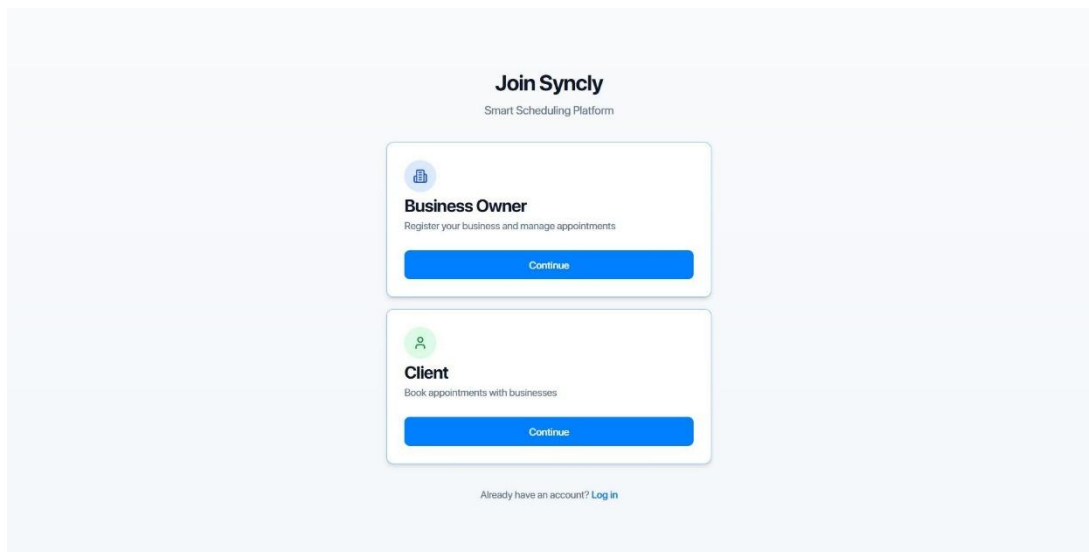
IV.3.3 Communication entre les composants

L'échange de données entre les différents modules suit une architecture basée sur des API REST :

- **Frontend** → **Backend** : Le frontend (React.js) communique avec le backend (Node.js) via des requêtes HTTP standard, en utilisant la bibliothèque Axios. Toutes les actions utilisateur passent par cette couche intermédiaire qui applique la logique métier.
- **Backend** → **Module IA** : Lorsqu'un utilisateur demande une suggestion intelligente, le backend envoie une requête HTTP au service IA FastAPI. Cette requête contient les données nécessaires (ID du client, disponibilité du prestataire, etc.). Le service IA répond ensuite avec une liste de créneaux classés.

IV.4. Présentation de la plateforme :

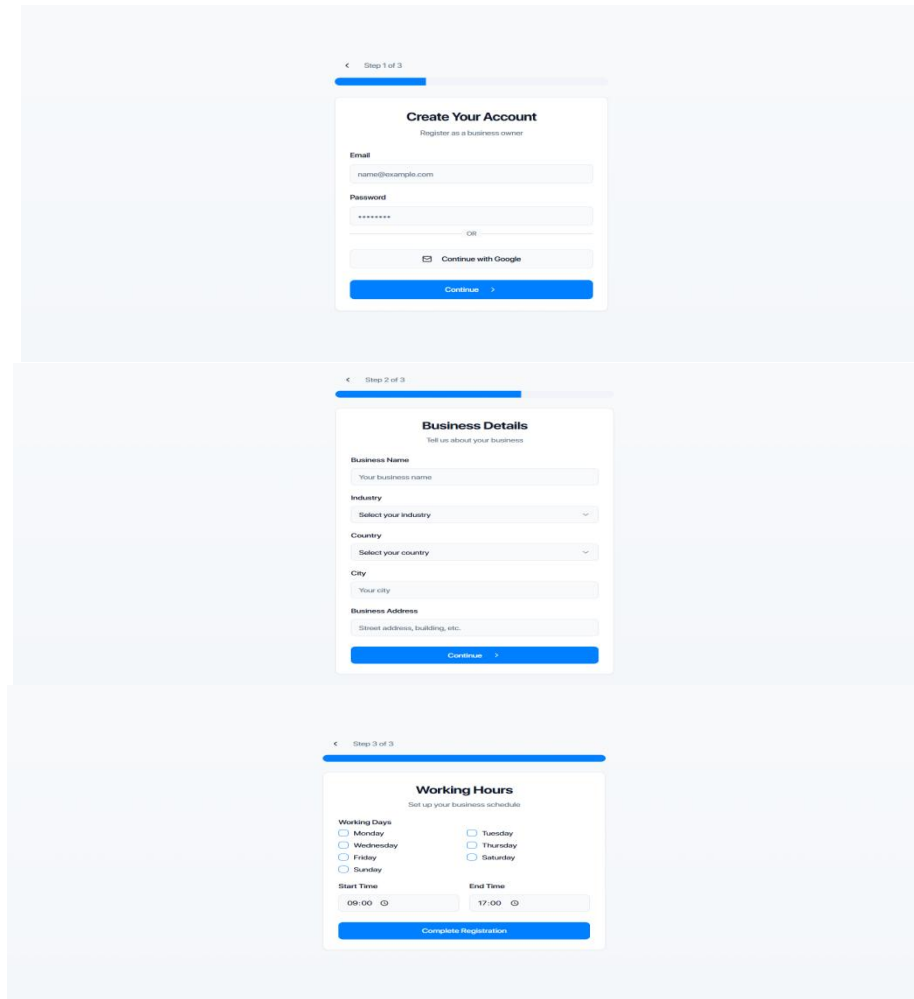
IV.4.1 La page d'accueil:



IV.4.2 L'interface de prestataire de service :

IV.4.2.1 L'inscription

Chapitre IV : Implémentation



IV.4.2.2 Dashboard

The dashboard for Synclly, a Smart Scheduling Platform, provides a comprehensive overview of a user's scheduling data. The interface includes a search bar, user profile (John Doe), and a sidebar with navigation options like Dashboard, Appointments (12), Appointment Requests (3), Clients, Messages (5), AI Assistant (Now), Integrations, and Settings.

Dashboard Summary:

- Total Appointments:** 248 (All scheduled appointments, +12% from last month)
- Clients:** 86 (Active clients, +8% from last month)
- Average Duration:** 42m (Per appointment, 5% from last month)

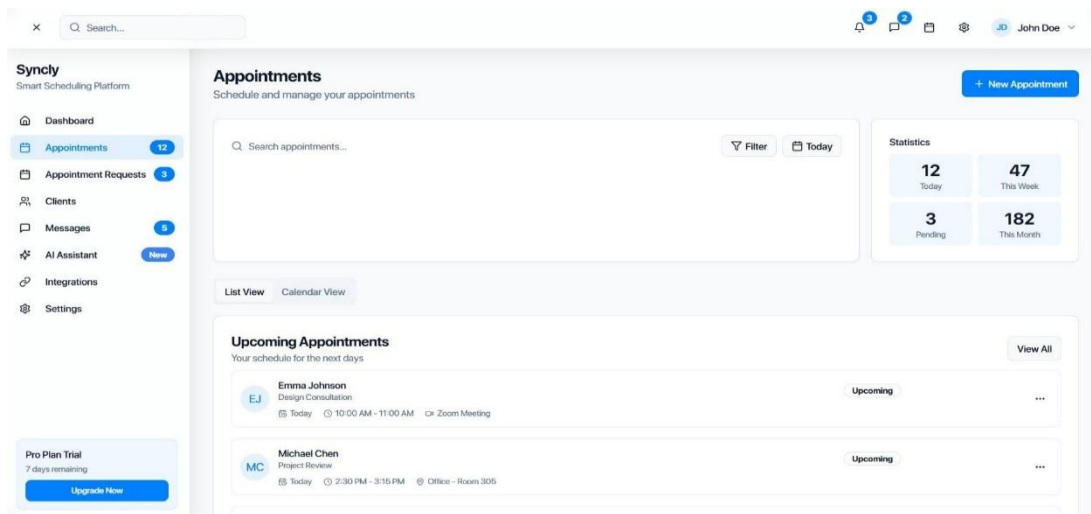
Upcoming Appointments:

- Emma Johnson (EJ):** Design Consultation, Today, 10:00 AM - 11:00 AM, Zoom Meeting.
- Michael Chen (MC):** Project Review, Today, 2:30 PM - 3:15 PM, Office - Room 305.
- Sophia Williams (SW):** Strategy Meeting, Tomorrow, 9:00 AM - 10:30 AM, Zoom Meeting.

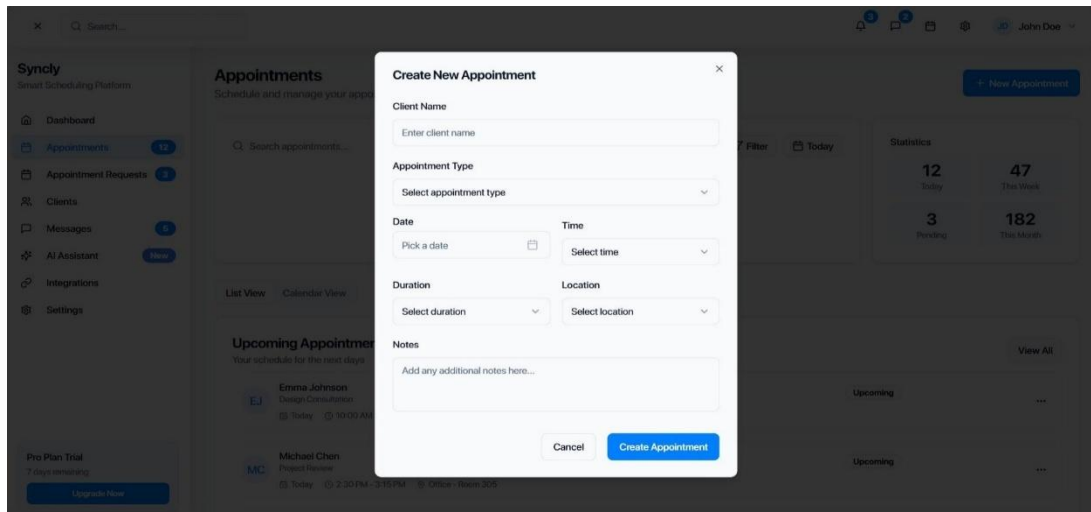
Calendar: A monthly calendar for June 2025 showing appointment slots with colored dots indicating scheduled events.

Chapitre IV : Implémentation

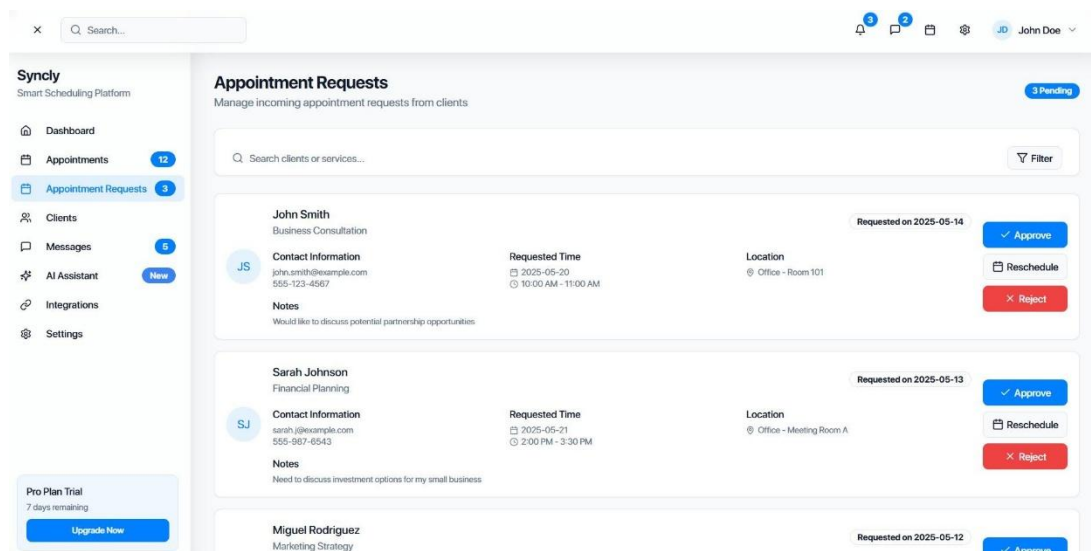
IV.4.2.3 Appointments (Les RDV)



IV.4.2.4 Create Appointments (Créer un RDV) :

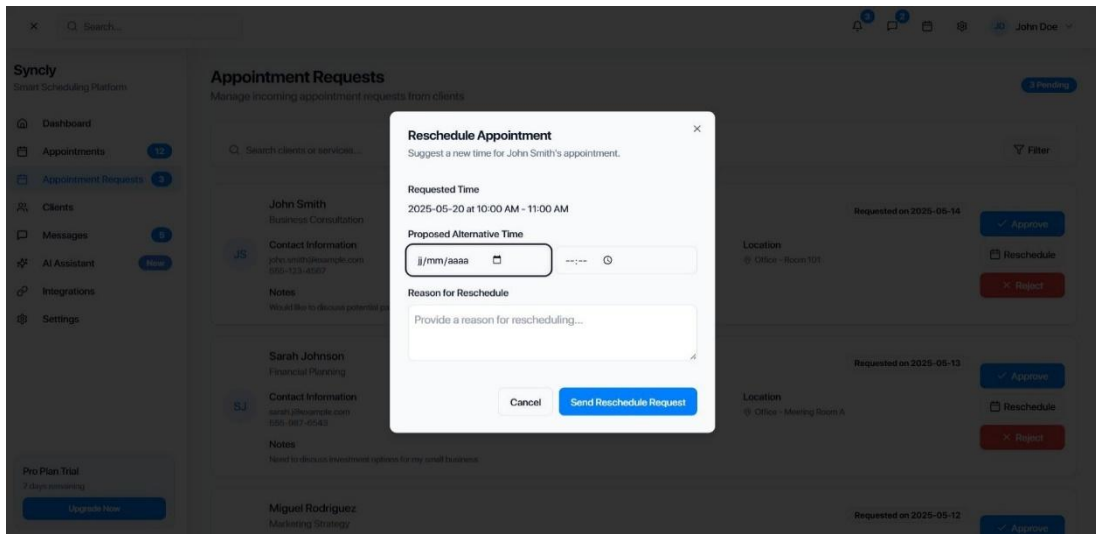


IV.4.2.5 Appointments requests (Les demandes de RDV) :

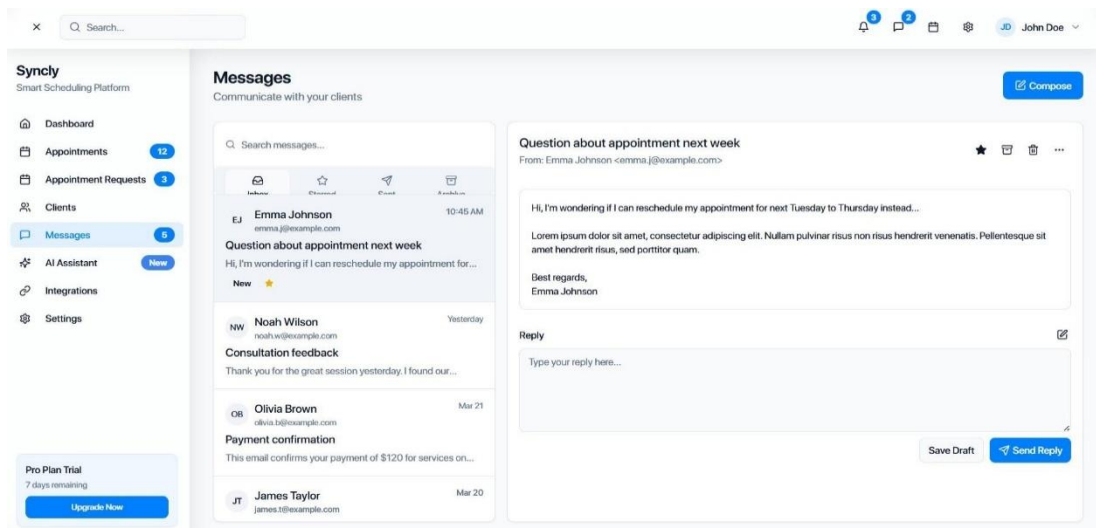


Chapitre IV : Implémentation

IV.4.2.6 Reschedule Appointments (Reprogrammer un RDV) :

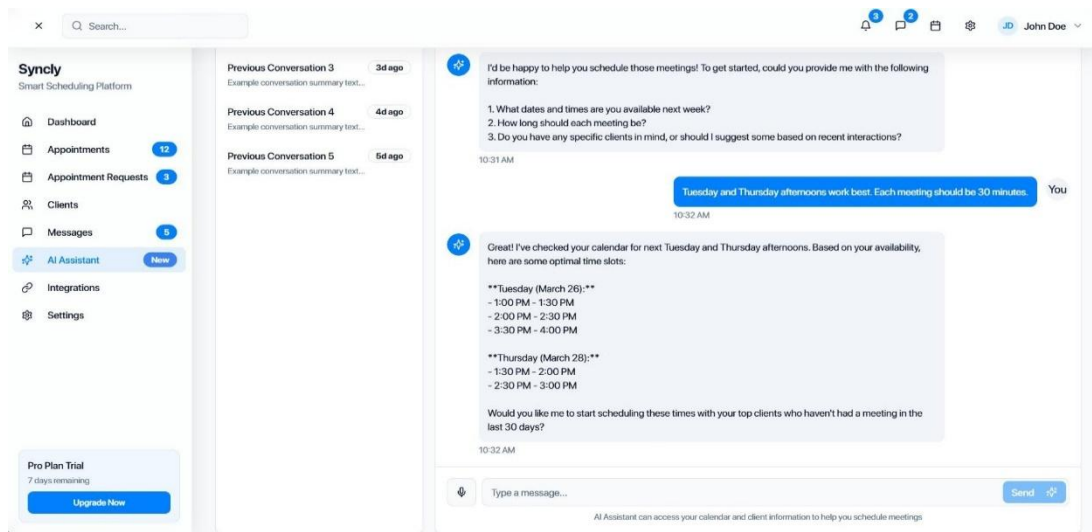


IV.4.2.7 Chat :



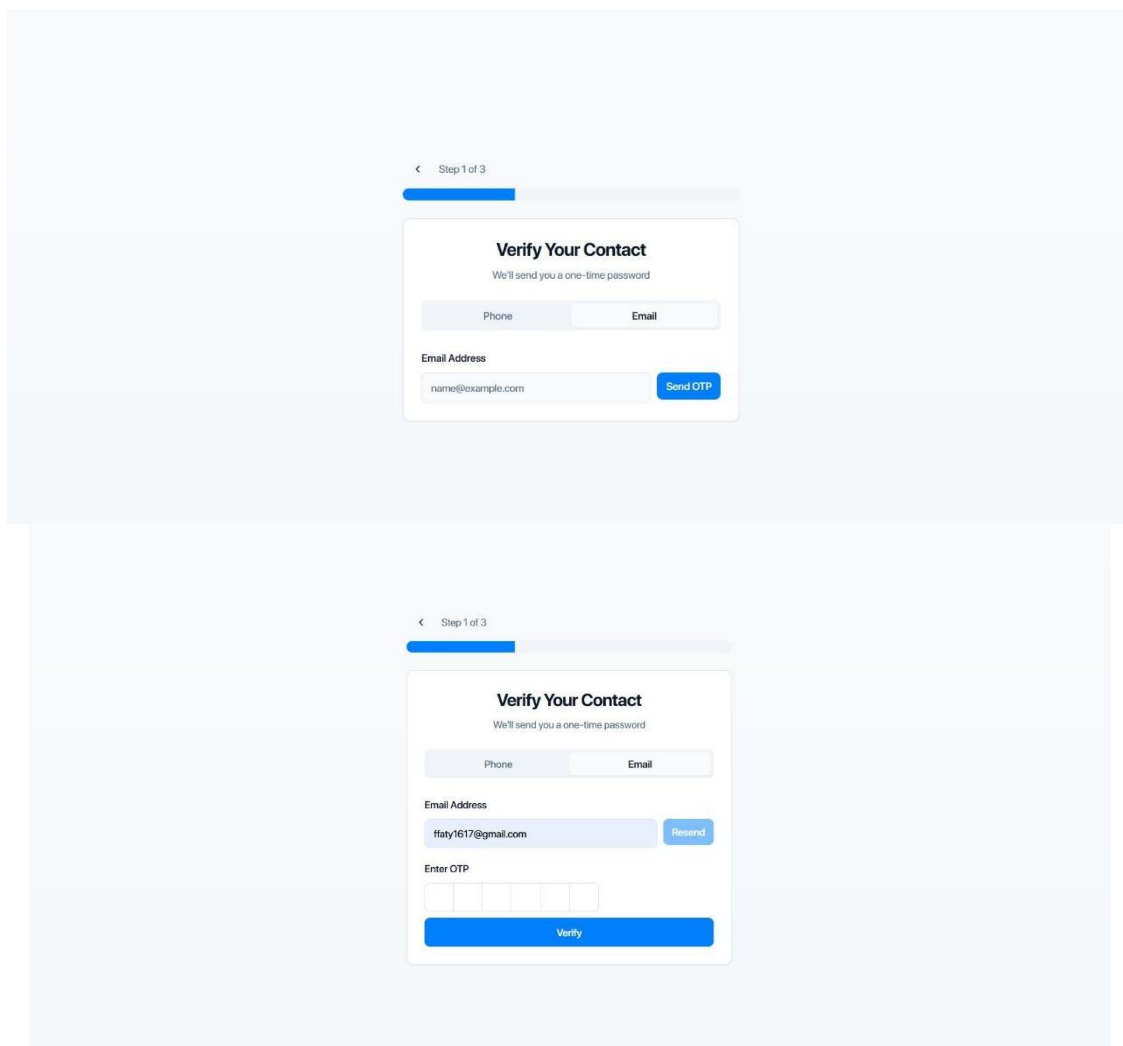
Chapitre IV : Implémentation

IV.4.2.8 AI Assistant:



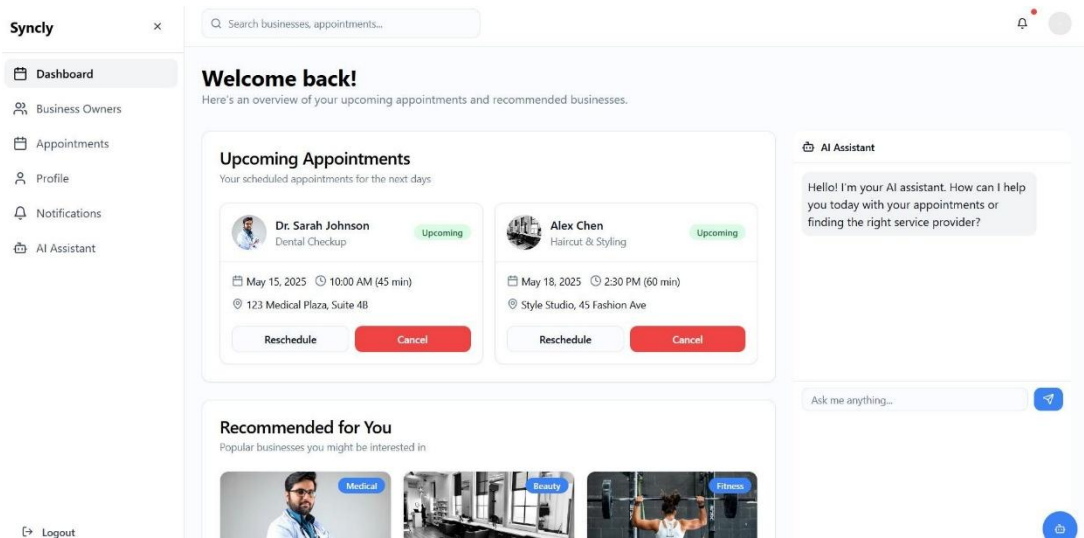
IV.4.3 L'interface Client:

IV.4.3.1 L'inscription :

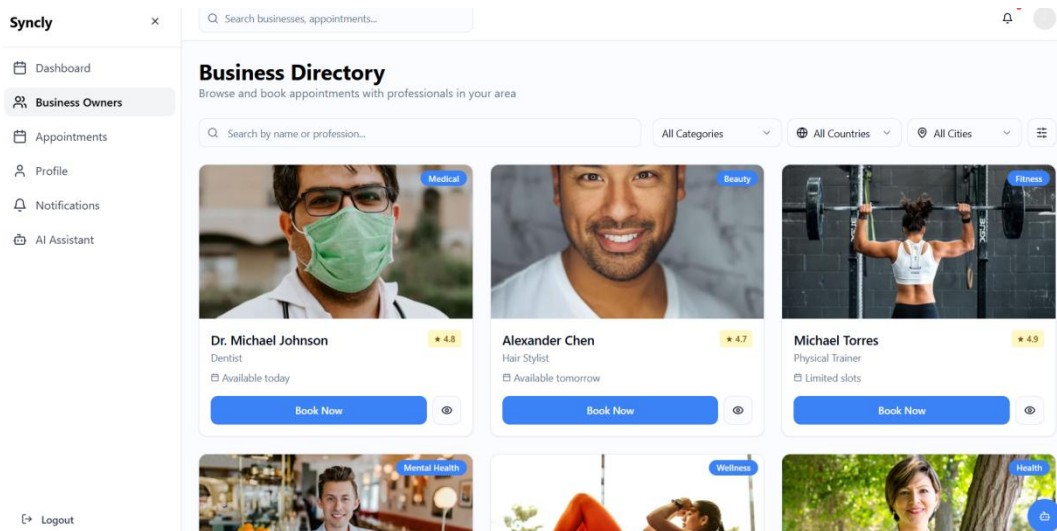


Chapitre IV : Implémentation

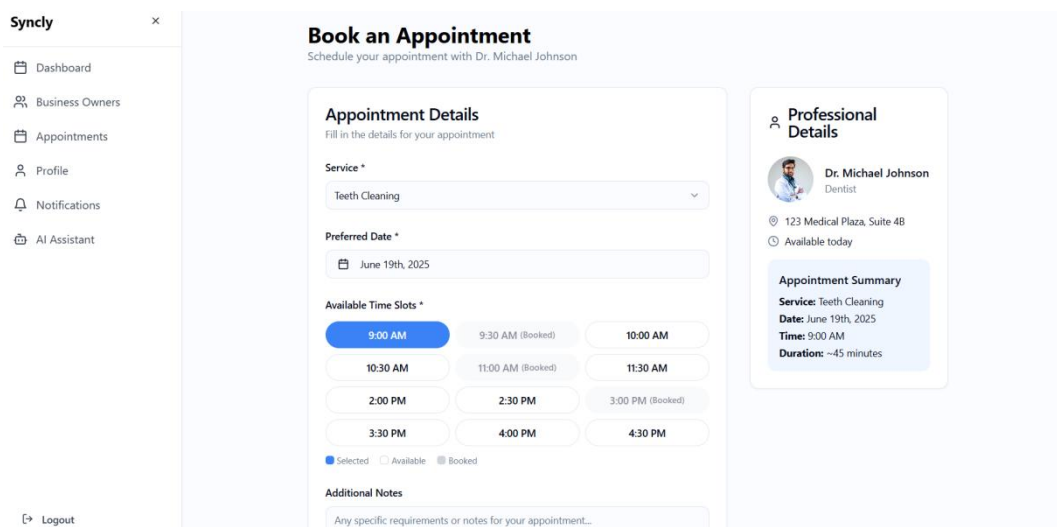
IV.4.3.2 Dashboard :



IV.4.3.3 Search Business Owner (Prestataire de service) :

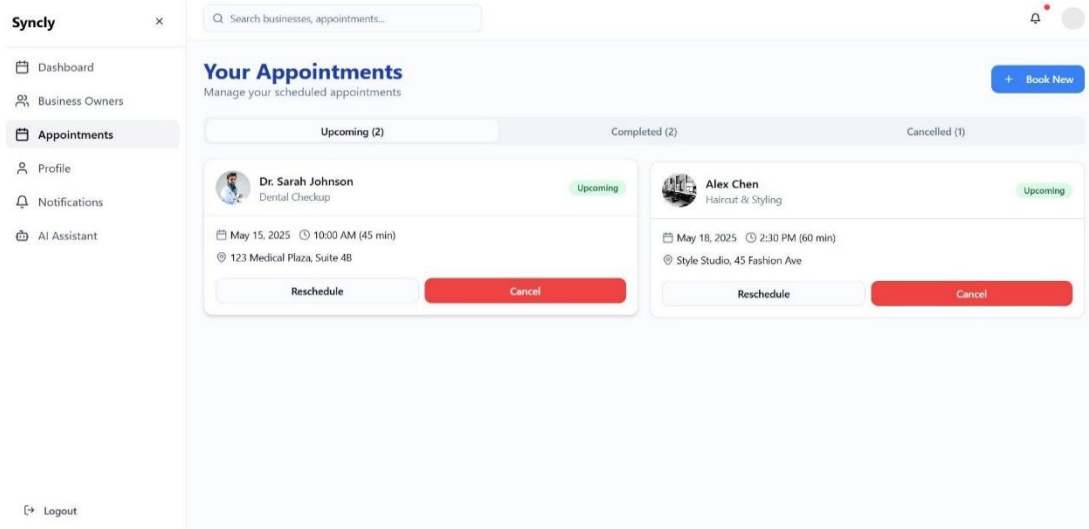


IV.4.3.4 Book appointment (Réserver un RDV) :

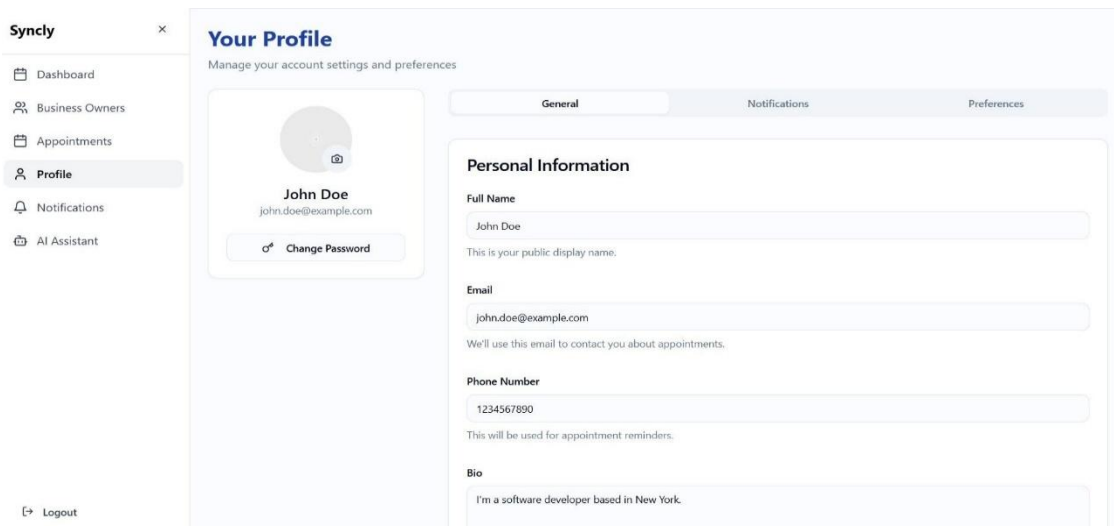


Chapitre IV : Implémentation

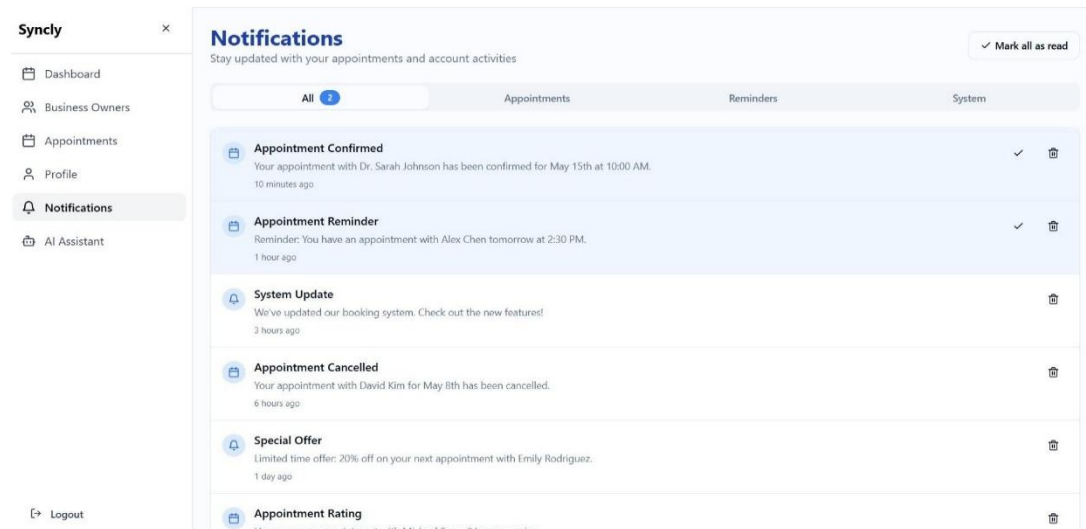
IV.4.3.5 Appointments (Les RDV) :



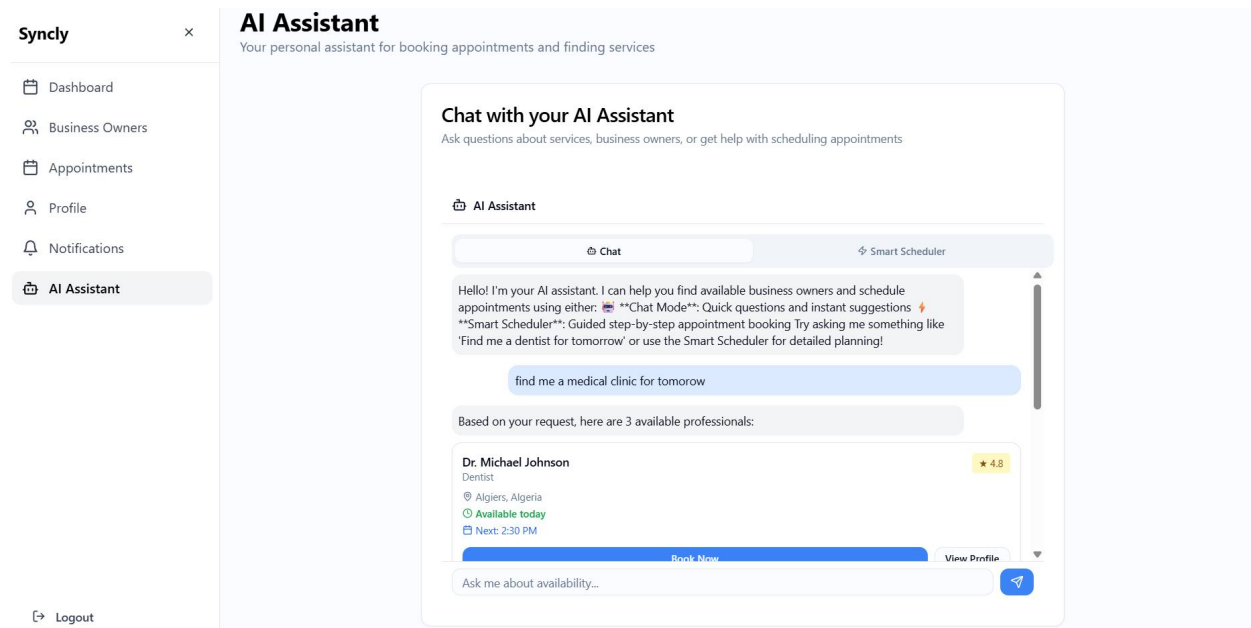
IV.4.3.6 Profile :



IV.4.3.7 Notifications:



IV.4.3.8 AI Assistant:



IV.5. Conclusion

Ce chapitre a présenté en détail le processus d'implémentation de la plateforme SaaS « Synclly » développée dans le cadre de ce projet de fin d'études. À travers une approche modulaire et structurée, nous avons mis en place une solution technique complète intégrant à la fois un frontend réactif, une API backend robuste, et un module d'intelligence artificielle autonome.

Nous avons exposé l'environnement de développement, les technologies utilisées, ainsi que l'architecture technique adoptée pour garantir la scalabilité, la sécurité et la maintenabilité du système. L'intégration du module IA, élément innovant du projet, permet de proposer aux utilisateurs des créneaux optimisés, améliorant ainsi significativement l'expérience de prise de rendez-vous.

Enfin, à travers des interfaces ergonomiques et des communications fluides entre les composants, la solution répond aux objectifs fixés : automatiser, personnaliser et optimiser la gestion des rendez-vous dans un cadre multi-sectoriel.

Conclusion

Générale

Conclusion générale

Dans un contexte où la transformation numérique s'impose comme un levier essentiel de performance, ce projet de fin d'études a permis de concevoir, développer et expérimenter une solution SaaS intelligente pour la gestion des rendez-vous, à destination de multiples secteurs professionnels.

Partant d'une analyse approfondie des méthodes traditionnelles et des limites des solutions existantes, nous avons formulé une réponse innovante, intégrant les bénéfices du cloud computing, de l'automatisation, et surtout, de l'intelligence artificielle. L'approche adoptée s'est appuyée sur une architecture modulaire, avec séparation claire des responsabilités (frontend, backend, IA), garantissant ainsi flexibilité, maintenabilité et évolutivité.

Le module IA développé apporte une véritable valeur ajoutée à la plateforme en optimisant la planification selon des critères dynamiques (habitudes du client, charge du prestataire, historique). Cette intelligence opérationnelle renforce l'expérience utilisateur, tout en fluidifiant la gestion du temps pour les prestataires.

En somme, cette plateforme répond efficacement aux besoins identifiés en amont, et pose les bases d'un système capable d'évoluer vers des usages encore plus riches, au service de la digitalisation des services.

Perspectives

Bien que la solution actuelle soit fonctionnelle et réponde aux besoins identifiés, plusieurs pistes d'amélioration peuvent être envisagées pour renforcer sa performance et étendre son champ d'application :

- Intégration d'un système de **paiement en ligne sécurisé** ;
- Développement d'une **application mobile native** pour Android et iOS ;
- Renforcement du **module d'IA** avec des modèles d'apprentissage automatique plus avancés ;
- Mise en place de **notifications en temps réel** via WebSockets ;
- Création d'un **portail administrateur centralisé** pour la gestion globale de la plateforme ;
- Support du **multilinguisme** afin de viser un public international.

Ces évolutions permettront à la plateforme de gagner en maturité, en adaptabilité et en compétitivité sur le marché des solutions SaaS intelligentes.

Bibliographie

Référence :

- [1]. Ian Sommerville. Software Engineering (10^e édition). Pearson, 2016.
- [2]. Stuart Russell & Peter Norvig. Artificial Intelligence: A Modern Approach (4^e édition). Pearson, 2021.
- [3]. Robert C. Martin. Clean Architecture: A Craftsman's Guide to Software Structure and Design. Prentice Hall, 2017.
- [4]. IBM Cloud. AI for scheduling and appointment management. WhitePaper, 2023.
- [5]. Gartner. Market Guide for Appointment Scheduling Software, 2022.
- [6]. OWASP. Principes de sécurité des API REST.
- [7]. GitHub – divers dépôts open source en SaaS/booking systems pour inspiration structurelle.
- [8]. <https://www.typescriptlang.org>
- [9]. <https://reactjs.org>
- [10] <https://nodejs.org> et <https://expressjs.com>
- [11] <https://fr.wikipedia.org>
- [12] OMG – UML 2.5 Specification. <https://www.omg.org/spec/UML>