

# Rapport de Projet

## Systeme de Gestion pour un Salon de Beauté

Présenté par :  
**Aya Aguilou**  
**Sabah Sayegh**

Encadrant :  
**Imane Daoudi**

**Année universitaire 2024-2025**

# Dédicace

*À nos familles,  
Pour leur amour inconditionnel, leur soutien et leurs encouragements constants.*

*À nos enseignants,  
Pour leur dévouement, leur patience et leur passion pour l'enseignement.*

*À nos amis,  
Pour leur aide précieuse et leur présence tout au long de ce parcours.*

*Merci à tous ceux qui ont cru en nous.*

# Remerciements

Nous tenons à exprimer notre profonde gratitude à toutes les personnes qui ont contribué de près ou de loin à la réalisation de ce projet.

Nous remercions tout particulièrement notre encadrante, **Mme Imane Daoudi**, pour ses conseils avisés, son suivi régulier, et sa disponibilité tout au long de ce travail.

Nous remercions également les membres de l'équipe pédagogique pour leur accompagnement durant toute notre formation.

Enfin, nous remercions nos familles et nos amis pour leur soutien moral et leur motivation constante tout au long de ce projet.

# Table des matières

<b>1</b>	<b>Introduction</b>	<b>4</b>
<b>2</b>	<b>Besoins du Système</b>	<b>5</b>
2.1	Besoins Fonctionnels . . . . .	5
2.2	Besoins Non Fonctionnels . . . . .	6
<b>3</b>	<b>Modélisation UML</b>	<b>13</b>
3.1	Diagramme des Fonctionnalités par Rôle . . . . .	13
3.2	Diagramme de Séquence : Prise de Rendez-vous . . . . .	14
3.3	Diagramme de Classes . . . . .	15
3.4	Diagramme d'État des Rendez-vous . . . . .	16
3.5	Diagramme de Flux : Algorithme de Réservation . . . . .	17
<b>4</b>	<b>Aperçu de l'Application</b>	<b>19</b>
4.1	Interface Client . . . . .	19
4.2	Architecture Technique . . . . .	22
<b>5</b>	<b>Conclusion</b>	<b>24</b>
<b>6</b>	<b>Perspectives d'amélioration et évolutions futures</b>	<b>25</b>

# 1. Introduction

Avec l'essor du numérique, de plus en plus de secteurs cherchent à optimiser leur gestion à l'aide de solutions logicielles. Les salons de beauté n'échappent pas à cette évolution : ils nécessitent une organisation rigoureuse pour assurer la satisfaction des clients, la gestion des ressources humaines, le suivi des prestations et la fluidité des rendez-vous.

Le présent rapport a pour objectif de décrire l'analyse et la conception d'un système de gestion dédié à un salon de beauté. Ce système vise à faciliter la gestion quotidienne en proposant une interface intuitive pour la prise de rendez-vous, le suivi des prestations, la facturation, et bien d'autres fonctionnalités indispensables à la bonne marche de l'activité.

Pour parvenir à une solution adaptée et évolutive, nous avons adopté une démarche basée sur la modélisation UML (Unified Modeling Language). Cette approche permet de visualiser et formaliser les besoins fonctionnels et techniques sous forme de diagrammes clairs et structurés.

Ce rapport présente :

- L'analyse des besoins fonctionnels et non fonctionnels,
- Une série de diagrammes UML (cas d'utilisation, séquence, classes, états, activités),
- L'architecture logique du système,
- Une synthèse des flux et des interactions entre les utilisateurs et le système.

Cette étape d'analyse est cruciale car elle pose les bases sur lesquelles reposera l'implémentation technique du logiciel. Elle permet également de détecter les contraintes, les dépendances, et les scénarios critiques afin d'anticiper les difficultés éventuelles.

Ainsi, ce document constitue un livrable fondamental pour orienter le développement, garantir la cohérence fonctionnelle et faciliter la collaboration entre les différents acteurs du projet (développeurs, designers, gestionnaires de projet, etc.).

## 2. Besoins du Système

Ce chapitre présente l'ensemble des besoins identifiés pour la conception du système de gestion d'un salon de beauté. Ces besoins sont divisés en deux catégories : les besoins fonctionnels (ce que le système doit faire) et les besoins non fonctionnels (comment le système doit se comporter).

### 2.1 Besoins Fonctionnels

Les besoins fonctionnels décrivent les fonctionnalités que le système doit offrir aux utilisateurs.

- **Gestion des rendez-vous clients** : Permettre aux clients de prendre, modifier ou annuler un rendez-vous via une interface simple. Le système doit proposer les créneaux disponibles en fonction de la prestation choisie et de la disponibilité des employés.
- **Enregistrement des informations clients** : Collecter et enregistrer les informations personnelles des clients (nom, prénom, numéro de téléphone, adresse e-mail, historique des rendez-vous) afin de faciliter le suivi et les communications.
- **Gestion des prestations** : Permettre l'ajout, la modification et la suppression des services proposés par le salon (soins du visage, coiffure, manucure, etc.), avec des détails comme la durée, le tarif et la catégorie.
- **Gestion des employés** : Suivre le planning des employés, leurs spécialités, leurs heures de travail, et les affecter aux prestations en fonction de leurs compétences et disponibilités.
- **Facturation des services** : Générer automatiquement une facture après la réalisation d'un rendez-vous, incluant les détails de la prestation, le prix, les réductions éventuelles, et le mode de paiement.
- **Suivi des disponibilités** : Mettre à jour en temps réel les disponibilités des salles de soins et des employés, en fonction des rendez-vous planifiés ou annulés.
- **Système de fidélité et de promotion** : Implémenter un programme de fidélité (points, réductions, offres spéciales) pour récompenser les clients réguliers, ainsi que la possibilité de créer des promotions ponctuelles.
- **Interface de prise de rendez-vous en ligne** : Offrir aux clients la possibilité de réserver un rendez-vous via un site web ou une application, sans devoir appeler ou se déplacer.

---

## 2.2 Besoins Non Fonctionnels

Les besoins non fonctionnels concernent les qualités attendues du système, notamment en termes de performance, de sécurité, et d'ergonomie.

- **Interface intuitive et responsive** : L'interface utilisateur doit être facile à comprendre et à utiliser, même pour les personnes peu familières avec la technologie. Elle doit également s'adapter à tous les types d'écrans (ordinateurs, tablettes, smartphones).
- **Sécurité des données personnelles (RGPD)** : Le système doit respecter le Règlement Général sur la Protection des Données, notamment en assurant la confidentialité, la traçabilité, et le consentement explicite des clients lors de la collecte de données personnelles.
- **Temps de réponse < 2 secondes** : Toutes les opérations (recherche de rendez-vous, affichage des services, enregistrement des données, etc.) doivent s'exécuter en moins de deux secondes pour garantir une expérience utilisateur fluide.
- **Sauvegarde automatique des données** : Le système doit effectuer des sauvegardes régulières afin d'éviter toute perte d'information en cas de panne ou d'erreur système.
- **Disponibilité du service à 99%** : L'application doit être accessible et fonctionnelle à tout moment, avec un minimum d'interruptions. Cela nécessite un hébergement fiable et une maintenance régulière.

# Glossaire

Ce glossaire regroupe les termes techniques et acronymes utilisés dans le présent rapport, afin de faciliter la compréhension pour tout lecteur.

- **UML (Unified Modeling Language)** : Langage de modélisation unifié. Il s'agit d'un langage standardisé utilisé pour la spécification, la visualisation, la conception et la documentation des systèmes logiciels. Il permet de représenter graphiquement les différents composants du système à travers des diagrammes (cas d'utilisation, séquence, classes, états, etc.).
- **RGPD (Règlement Général sur la Protection des Données)** : Règlement européen entré en vigueur en mai 2018, visant à protéger les données personnelles des citoyens de l'Union Européenne. Dans le cadre d'un système de gestion de salon de beauté, le RGPD impose des obligations concernant la collecte, le stockage, la sécurité et le consentement lié aux données clients.
- **CRUD (Create, Read, Update, Delete)** : Acronyme désignant les quatre opérations de base de la gestion des données dans une application. Ces opérations sont essentielles à toute application de gestion :
  - **Create** : créer un enregistrement (ex. : ajouter un nouveau client)
  - **Read** : lire ou consulter les données (ex. : afficher la liste des rendez-vous)
  - **Update** : mettre à jour les informations existantes (ex. : modifier l'horaire d'un rendez-vous)
  - **Delete** : supprimer un enregistrement (ex. : supprimer un employé ou une prestation)
- **Frontend** : Partie visible d'une application, c'est-à-dire l'interface utilisateur. C'est ce que l'utilisateur interagit avec, généralement via un navigateur web ou une application mobile.
- **Backend** : Partie invisible du système, située côté serveur. Elle gère la logique métier, l'accès aux données, la sécurité, les traitements et les règles internes du système.
- **Base de données** : Système de stockage structuré des informations (clients, rendez-vous, prestations, employés, factures, etc.). Elle permet la consultation, la modification et la sauvegarde fiable des données.
- **Framework** : Ensemble d'outils et de bibliothèques facilitant le développement d'applications. Par exemple, Django est un framework backend basé sur Python, qui structure et accélère la construction d'un site web.
- **ORM (Object-Relational Mapping)** : Technique permettant de manipuler les données d'une base relationnelle (comme MySQL) via des objets dans un langage de programmation orienté objet (comme Python). Cela simplifie le développement et améliore la lisibilité du code.



# Choix Technologiques et Outils Utilisés

Le développement d'un système de gestion performant nécessite une sélection rigoureuse des technologies adaptées aux besoins fonctionnels et non fonctionnels du projet. Le choix des outils s'est fait selon des critères tels que la facilité d'intégration, la performance, la courbe d'apprentissage, et la communauté de support. Voici les technologies et outils retenus :

## Interface Utilisateur (Frontend)

**React.js** : bibliothèque JavaScript développée par Facebook, utilisée pour construire des interfaces utilisateur dynamiques, rapides et réactives. Elle permet la création de composants réutilisables et offre une bonne gestion de l'état de l'application, ce qui est particulièrement utile pour gérer des formulaires complexes comme la prise de rendez-vous.

## Serveur Applicatif (Backend)

**Django** : framework web Python hautement sécurisé et rapide à mettre en œuvre. Il permet une gestion efficace de la base de données via son ORM (Object Relational Mapping), une architecture MVC robuste, et intègre par défaut des modules pour l'authentification, l'administration, et la gestion des utilisateurs. Il est donc bien adapté pour les besoins d'un salon de beauté en matière de sécurité, performance, et évolutivité.

## Base de Données

**MySQL** : système de gestion de base de données relationnelle fiable, performant, et largement adopté. Il permet de stocker efficacement les informations des clients, employés, prestations, rendez-vous et factures.

**XAMPP** : environnement de développement local incluant Apache, MySQL, PHP et Perl. Il est utilisé ici pour faciliter l'hébergement et la gestion locale de la base de données MySQL durant la phase de développement et de test.

## Outils de Modélisation

**Draw.io** : outil en ligne gratuit pour la création de diagrammes UML (cas d'utilisation, séquence, états, etc.). Il permet une modélisation graphique rapide et collaborative.

---

**StarUML** : outil professionnel de modélisation UML permettant la conception détaillée du système via différents types de diagrammes. Il est utilisé pour structurer l'architecture du projet avant le développement.

# Contraintes et Limites

- Le système ne gère pas encore les paiements en ligne.
- Aucune intégration avec des outils externes (Google Calendar, SMS).
- L'interface mobile n'est pas encore optimisée.

# Perspectives d'Évolution

- Une application mobile pour clients et employés.
- Un système de statistiques de performance.
- Une intégration avec des plateformes de paiement en ligne.

# Planning de Réalisation

Tâche	Description	Durée estimée
Analyse des besoins	Identifier les besoins fonctionnels et non fonctionnels	1 semaine
Modélisation UML	Création des diagrammes UML (cas d'utilisation, classes, etc.)	2 semaines
Prototypage	Esquisse de l'interface utilisateur	1 semaine
Rédaction du rapport	Compilation et mise en page du document final	1 semaine

## 3. Modélisation UML

### 3.1 Diagramme des Fonctionnalités par Rôle



FIGURE 3.1 – Diagramme des fonctionnalités par rôle

#### Description

Ce diagramme illustre les fonctionnalités accessibles selon le type d'utilisateur (Client, Employé, Administrateur).

#### Fonctionnalités par Profil

##### **Client**

- Prendre un rendez-vous
- Consulter l'historique client

##### **Employé**

- Gérer les prestations
- Gérer les horaires
- Gérer les employés

##### **Administrateur**

- Générer une facture

#### Objectif

Clarifier les permissions et organiser l'interface selon le profil utilisateur.

## 3.2 Diagramme de Séquence : Prise de Rendez-vous

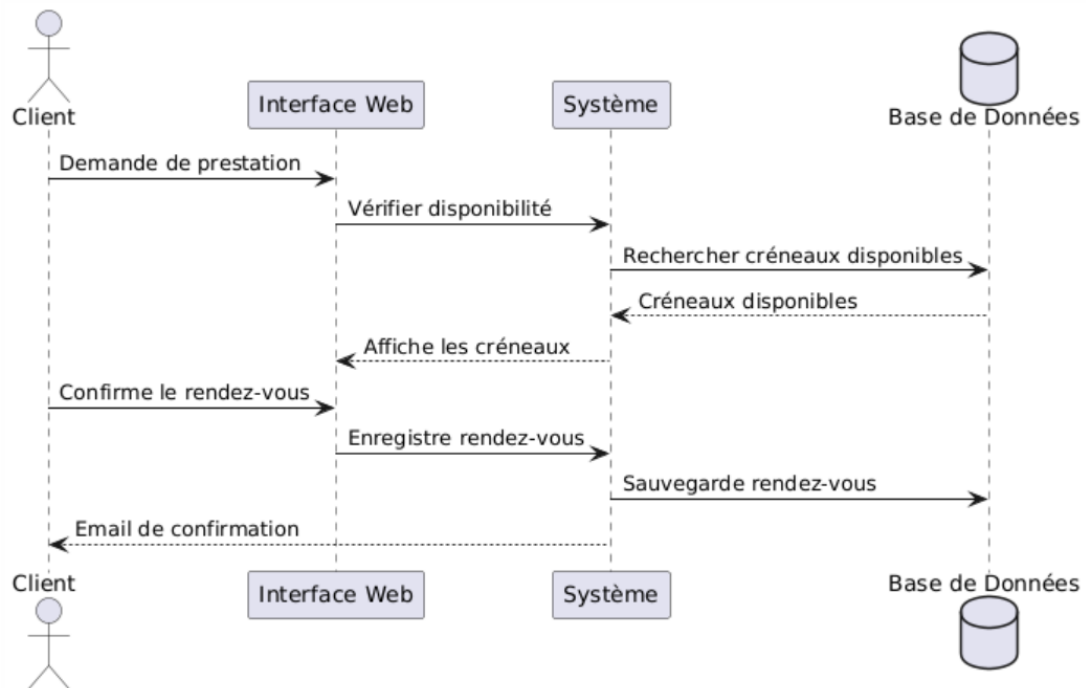


FIGURE 3.2 – Diagramme de séquence pour la prise de rendez-vous

### Description

Ce diagramme décrit les interactions entre le client, l'interface web, le système, et la base de données.

### Étapes du Processus

1. Demande de prestation
2. Vérification des disponibilités
3. Affichage des créneaux disponibles
4. Confirmation
5. Enregistrement
6. Envoi d'email

### 3.3 Diagramme de Classes

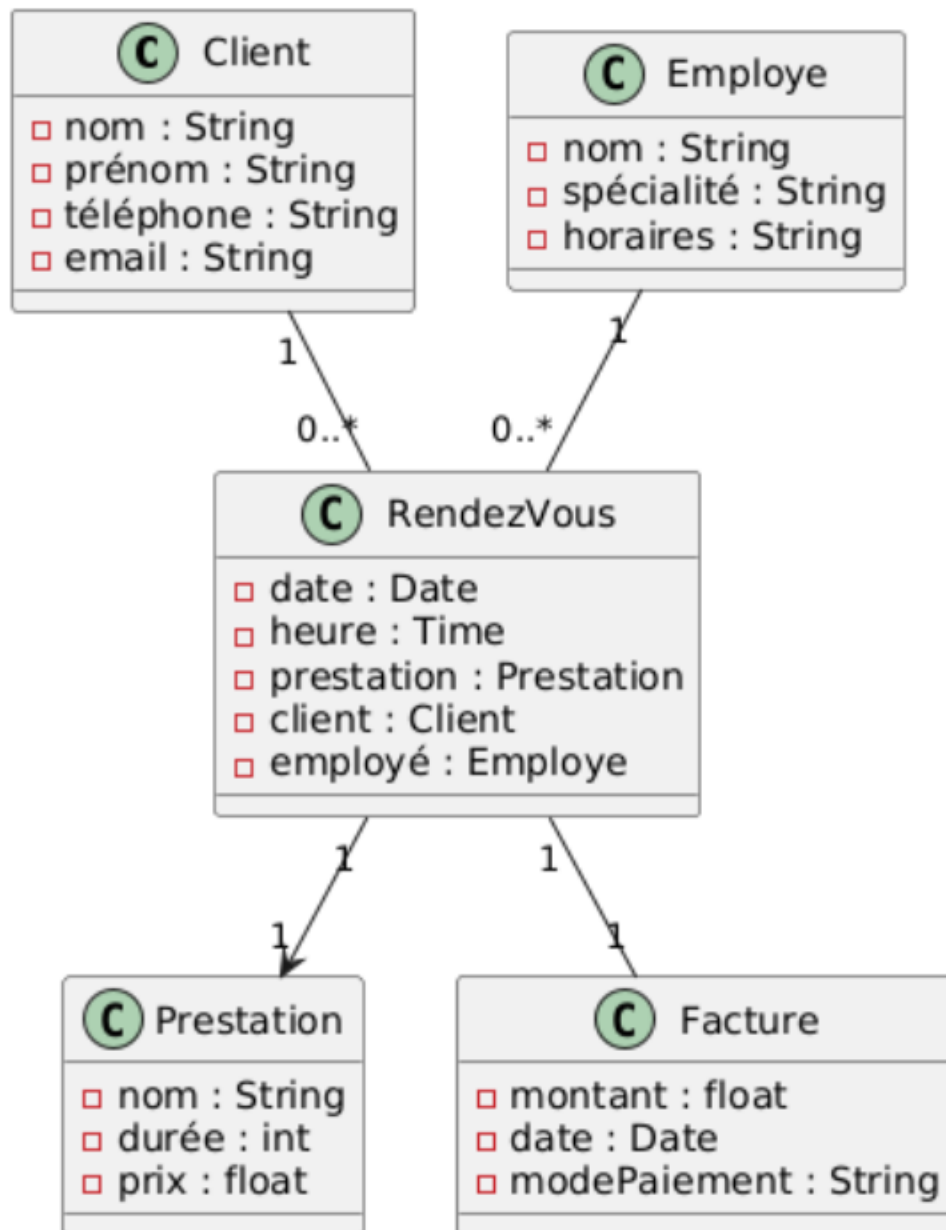


FIGURE 3.3 – Diagramme de classes du système

#### Description

Ce diagramme représente la structure du système avec les entités principales.

#### Table des classes



Classe	Attributs	Relations
Client	nom, prénom, téléphone, email	1 $\rightarrow$ 0..* RendezVous
Employé	nom, spécialité, horaires	0..* $\rightarrow$ 1 RendezVous
RendezVous	date, heure, prestation	Lié à Client, Employé, Prestation
Prestation	nom, durée, prix	1 $\rightarrow$ 1 RendezVous
Facture	montant, date, modePaiement	Générée après RendezVous

### 3.4 Diagramme d'État des Rendez-vous

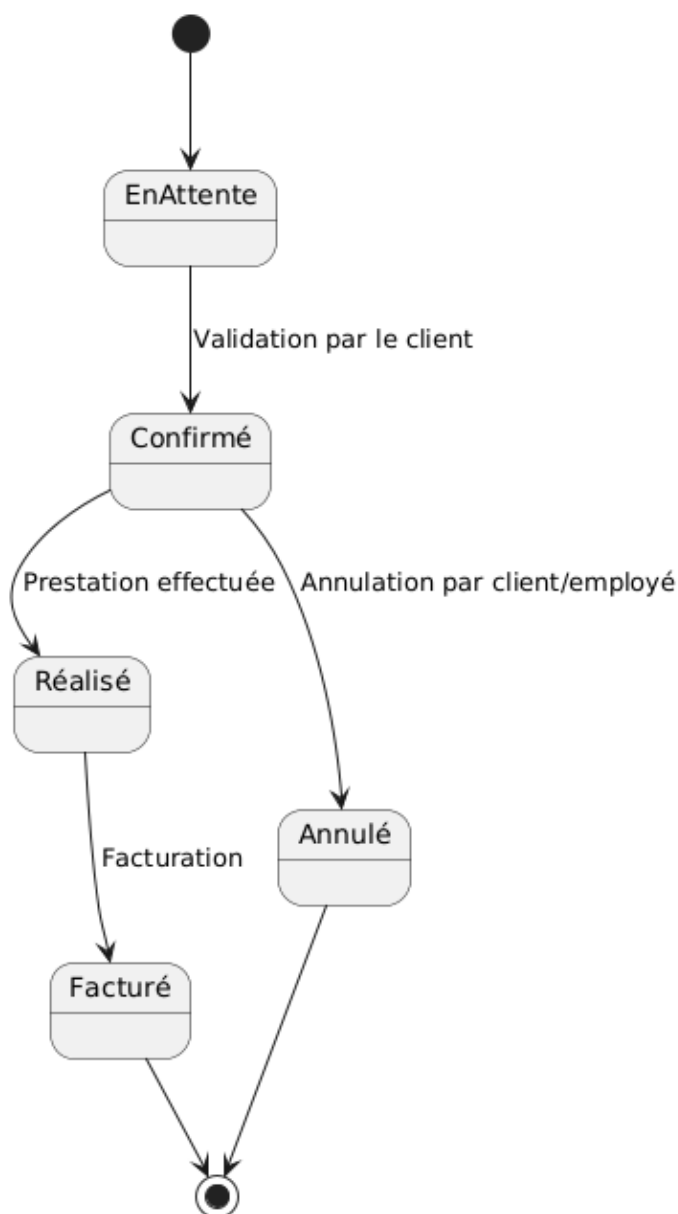


FIGURE 3.4 – Diagramme d'états-transitions des rendez-vous

## États

- EnAttente
- Confirmé
- Annulé
- Réalisé
- Facturé

## Transitions

- EnAttente → Confirmé
- Confirmé → Annulé
- Confirmé → Réalisé
- Réalisé → Facturé

### 3.5 Diagramme de Flux : Algorithme de Réservation

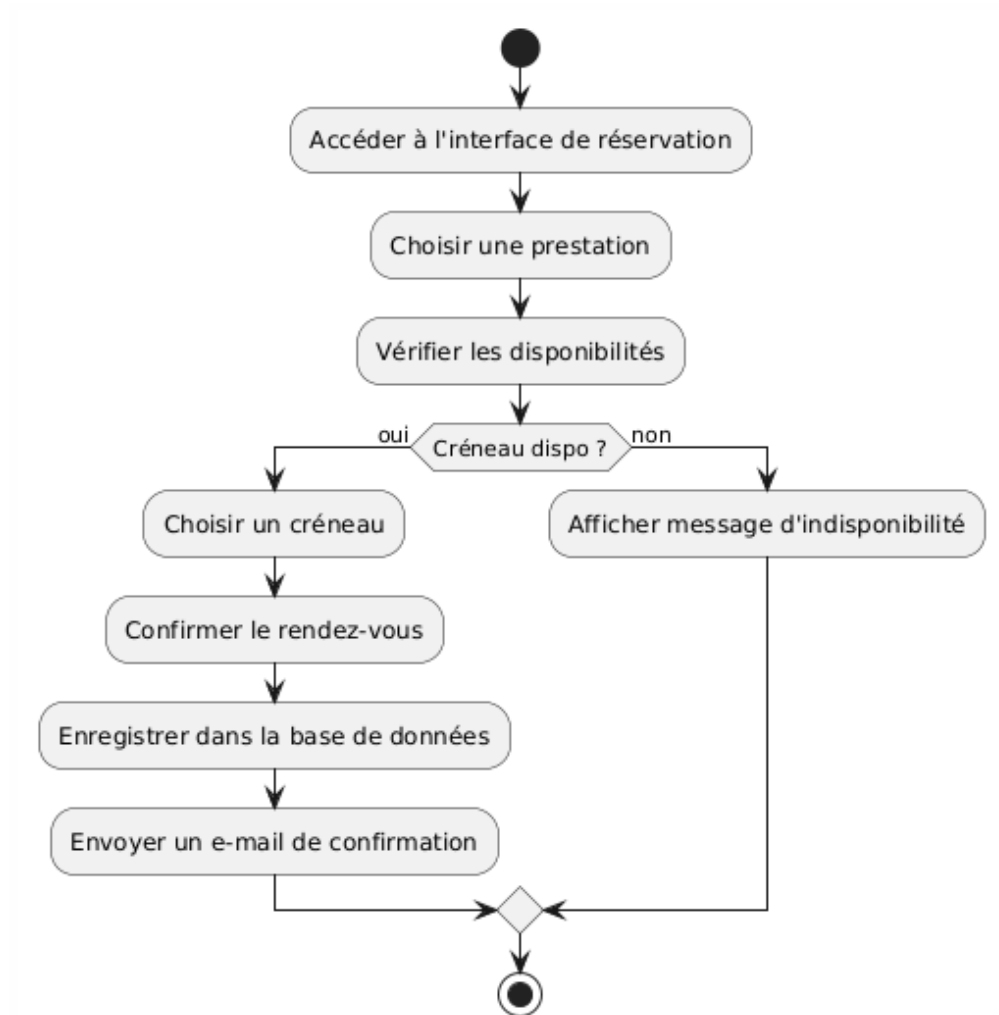


FIGURE 3.5 – Diagramme de flux de l'algorithme de réservation

---

## Étapes

1. Accès à l'interface
2. Choix d'une prestation
3. Vérification des disponibilités
4. Si disponible : Affichage  $\rightarrow$  Confirmation
5. Sinon : Message d'indisponibilité
6. Enregistrement
7. Email de confirmation

[a4paper, 12pt]report [utf8]inputenc [T1]fontenc [french]babel graphicx float caption hyperref  
[margin=2.5cm]geometry

## 4. Aperçu de l'Application

### 4.1 Interface Client

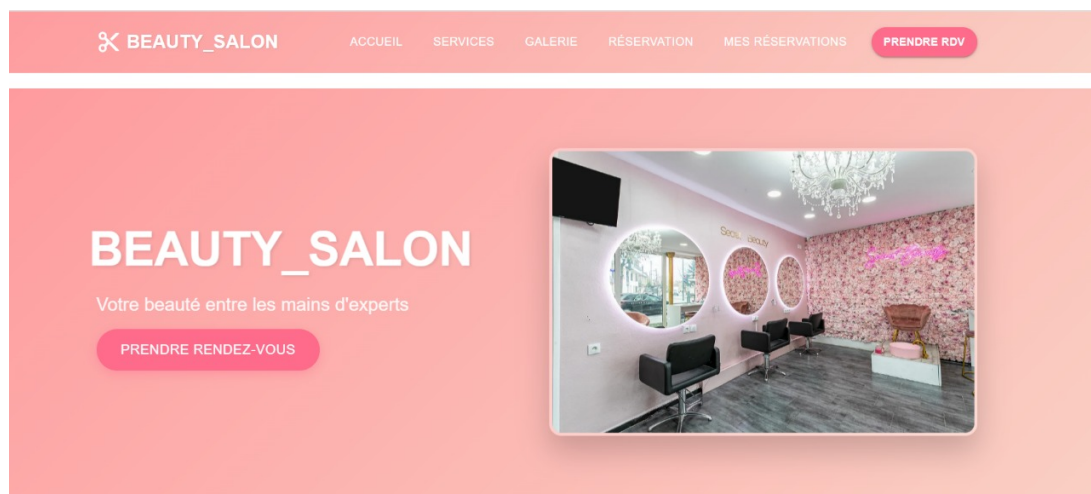
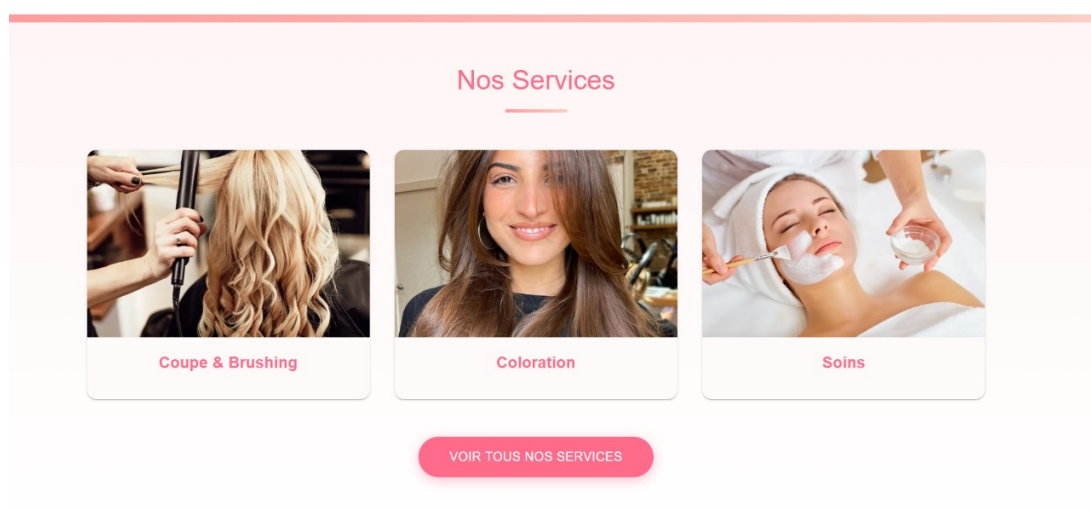


FIGURE 4.1 – Page d'accueil du salon avec menu de navigation



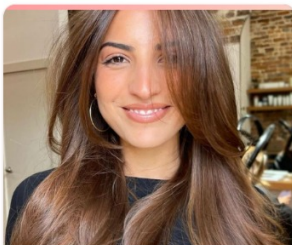
## Notre Galerie

Découvrez nos plus belles réalisations



### Coupe & Brushing

Une coupe moderne et stylée.



### Coloration

Des couleurs vives et naturelles.



### Soin Capillaire

Prenez soin de vos cheveux.

## Nos Services

Découvrez notre gamme complète de services de beauté

### Coupe Femme

Coupe, brushing et mise en forme pour femme

45.00€

Durée: 60 min

RÉSERVER

### Coupe Homme

Coupe et coiffage pour homme

25.00€

Durée: 30 min

RÉSERVER

### Coloration

Coloration professionnelle avec produits de qualité

65.00€

Durée: 90 min

RÉSERVER

### Mèches

Mèches ou balayage pour un effet naturel

75.00€

Durée: 120 min

### Brushing

Brushing et mise en forme

30.00€

Durée: 45 min

### Soin Profond

Soin profond revitalisant pour cheveux

35.00€

Durée: 45 min

FIGURE 4.2 – Liste des prestations disponibles avec prix et durée

<b>Coupe Femme</b> Coupe, brushing et mise en forme pour femme <b>45.00€</b> <small>Durée: 60 min</small> RÉSERVER	<b>Coupe Homme</b> Coupe et coiffage pour homme <b>25.00€</b> <small>Durée: 30 min</small> RÉSERVER	<b>Coloration</b> Coloration professionnelle avec produits de qualité <b>65.00€</b> <small>Durée: 90 min</small> RÉSERVER
<b>Mèches</b> Mèches ou balayage pour un effet naturel <b>75.00€</b> <small>Durée: 120 min</small> RÉSERVER	<b>Brushing</b> Brushing et mise en forme <b>30.00€</b> <small>Durée: 45 min</small> RÉSERVER	<b>Soin Profond</b> Soin profond revitalisant pour cheveux <b>35.00€</b> <small>Durée: 45 min</small> RÉSERVER
<b>Coupe Femme</b> Coupe, brushing et mise en forme pour femme <b>45.00€</b> <small>Durée: 60 min</small> RÉSERVER	<b>Coupe Homme</b> Coupe et coiffage pour homme <b>25.00€</b> <small>Durée: 30 min</small> RÉSERVER	<b>Coloration</b> Coloration professionnelle avec produits de qualité <b>65.00€</b> <small>Durée: 90 min</small> RÉSERVER

## Réservez votre moment détente

Complétez ce formulaire pour réserver votre prestation

Nom complet \*

Email \*

Téléphone \*

Choisissez un service \*

Date de rendez-vous

Heure de rendez-vous

Commentaires ou demandes spéciales

FIGURE 4.3 – Formulaire de réservation avec sélection de créneau

BEAUTY\_SALON

ACCUEILSERVICESGALERIERÉSERVATIONMES RÉSERVATIONS

PRENDRE RDV

Mes Réservations

Votre email \*

RECHERCHER MES RÉSERVATIONS

FIGURE 4.4 – Interface de consultation des réservations par email

## 4.2 Architecture Technique

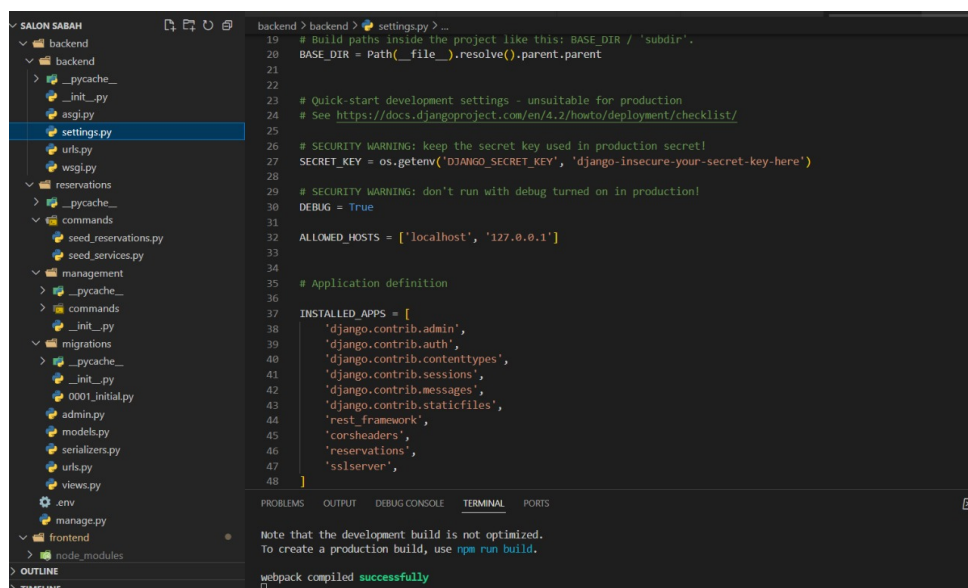


FIGURE 4.5 – Arborescence des fichiers du projet (frontend/backend)

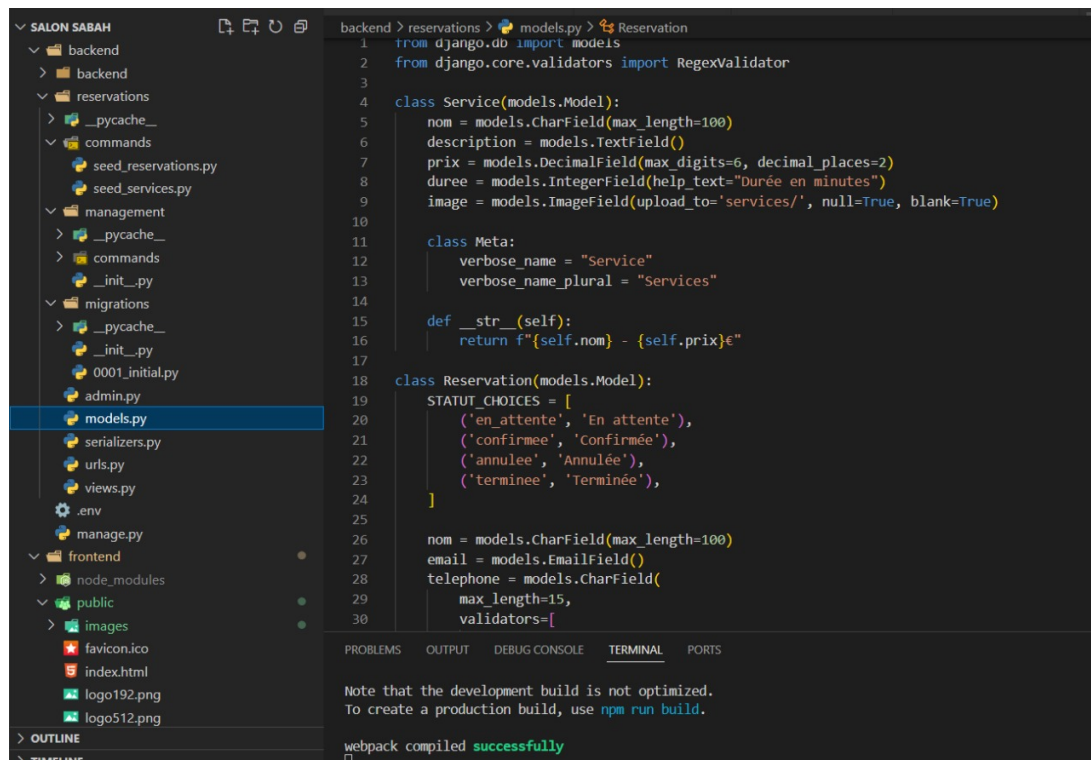


FIGURE 4.6 – Diagramme des modèles Django (Services, Réservations)

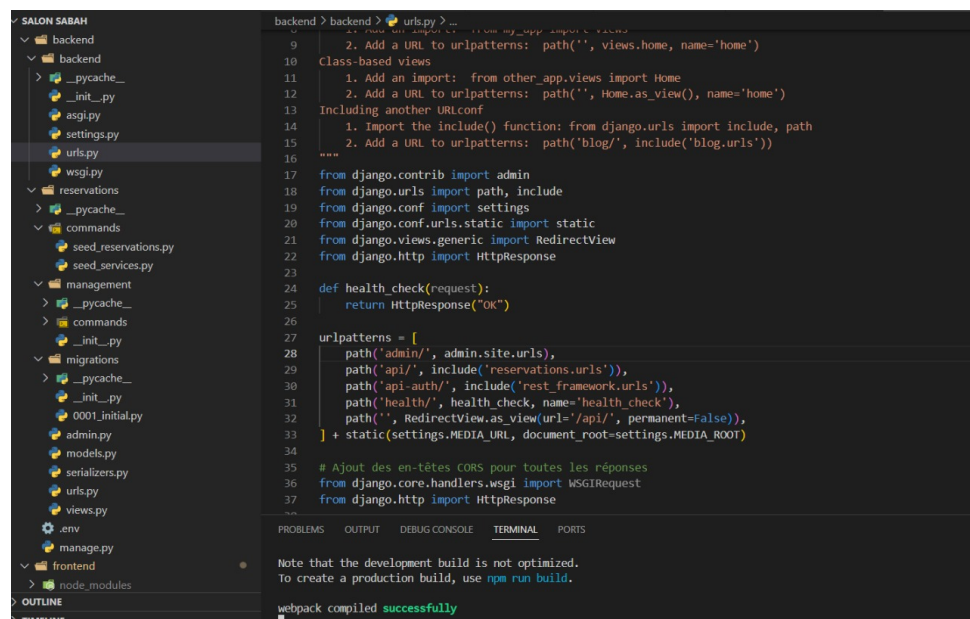


FIGURE 4.7 – Exemple de endpoints API pour les réservations



## 5. Conclusion

Conclusion améliorée sur la modélisation UML La modélisation UML réalisée dans le cadre de ce projet constitue une base solide pour le développement technique du logiciel de gestion du salon de beauté. Les différents diagrammes UML (cas d'utilisation, séquence, classes, activité, états-transitions) offrent une représentation claire et structurée des besoins fonctionnels et non fonctionnels du système.

Ces représentations facilitent la compréhension globale du système, aussi bien pour les développeurs que pour les autres parties prenantes du projet (designers, chefs de projet, testeurs). Elles jouent un rôle essentiel dans :

- la communication entre les membres de l'équipe,
- la détection précoce des incohérences ou oublis,
- et la planification du travail de développement.

Grâce à cette modélisation, chaque fonctionnalité est clairement définie, chaque interaction est anticipée, et chaque composant logiciel peut être développé en conformité avec les exigences métiers.

## 6. Perspectives d'amélioration et évolutions futures

Malgré la richesse de la modélisation actuelle, certaines améliorations pourraient être envisagées à l'avenir, notamment :

Ajout de diagrammes de déploiement : pour visualiser l'architecture physique du système une fois déployé (serveurs, clients, base de données, réseau).

Affinement des diagrammes de séquence : en intégrant davantage de scénarios spécifiques (gestion des erreurs, processus de réservation avancée, notifications, etc.).

Validation des modèles avec les utilisateurs finaux : organiser des ateliers de validation pour s'assurer que les cas d'utilisation couvrent bien tous les besoins métiers réels.

Mise à jour continue de la documentation UML : tout au long du cycle de développement, pour refléter les changements ou évolutions du logiciel.

Utilisation d'un outil collaboratif UML en ligne : tel que Lucidchart, Draw.io, ou StarUML avec Git, pour permettre aux équipes de maintenir les diagrammes à jour en temps réel.