Université Abou Bekr Belkaid
Tlemcen Algérie
جامعة أبي بكر بلقايد
تلمسان الجزائر

République Algérienne Démocratique et Populaire
Université Abou Bakr Belkaid– Tlemcen
Faculté des Sciences
Département d'Informatique

Mémoire de fin d'études

Pour l'obtention du diplôme de Master en Informatique

Option : *Génie Logiciel (G.L)*

Thème

Conception et développement d'un système de gestion des rendez-vous au sein de l'administration Algérienne

Réalisé par :

- GADIRI Nadjat
- GUENNOUN Ouissem

Présenté le 26 Juin 2023 devant le jury composé de MM.

- *Mr CHIKH Azzedine* *(Président)*
- *Mme BENMAHDI Meriem Bouchra* *(Examinateur)*
- *Mr MATALLAH Houcine* *(Encadrant)*
- *Mr SEMMOUD Abderrazak* *(Co-encadrant)*

Année universitaire: 2022-2023

Remerciement

En premier lieu, nous souhaitons exprimer notre profonde gratitude à Allah pour nous avoir donner l'aide, la santé, la patience et la volonté pour l'accomplissement de ce travail. C'est grâce à LUI que nous sommes parvenues à atteindre la fin de notre cursus avec succès.

Nos remerciements les plus sincères vont à nos encadrants, en particulier à Monsieur MATALLAH HOUCINE pour ses conseils avisés, sa confiance, son soutien inconditionnel et ses orientations éclairées. Nous exprimons également notre gratitude envers Monsieur SEMMOUD ABDERRAZAK pour son suivi attentif et le temps précieux qu'il a consacré à nos réunions tout au long de notre stage au sein de l'entreprise DATA MASTER.

Nous tenons à remercier chaleureusement les membres du jury qui ont généreusement accepté d'évaluer et d'examiner notre travail avec diligence et expertise.

Nous remercions tous les enseignants de notre département d'informatique pour leurs efforts fournis durant notre cursus.

Enfin, nous souhaitons exprimer notre profonde reconnaissance à nos très chers parents, nos proches et nos amis qui ont été d'un soutien indéfectible tout au long de notre parcours.

Nous espérons sincèrement que ces quelques lignes témoignent de notre reconnaissance profonde et sincère envers tous ceux qui ont contribué à notre réussite.

Dédicaces

En mémoire de mes chers parents, que Dieu leurs fasse miséricorde, je dédie ce travail reconnaissance de leurs efforts et de leur soutien inconditionnel tout au long de ma vie. Leurs enseignements, leur amour et leur présence résonnent encore dans mon cœur, et je suis profondément reconnaissant pour les valeurs qu'ils m'ont inculquées et les leçons de vie qu'ils m'ont transmises, même par leur absence physique.

A mon cher frère que Dieu le protège et sa famille, pour tout le soutien que tu m'as offert je te dis merci.

À ma chère belle-mère, je souhaite une longue vie remplie de bonheur. Je vous suis reconnaissante pour tout l'amour et l'attention que vous m'avez témoignés.

À mes trois chères sœurs, Dalila, Salima et Nabila, ainsi qu'à tous vos enfants que j'aime tant, je souhaite exprimer mon amour et ma gratitude. Votre présence dans ma vie a été une source constante de joie et de soutien.

À mes deux encadrants, je tiens à vous remercier du fond du cœur pour votre confiance et votre soutien, votre expertise, votre guidage et vos encouragements et pour le temps précieux que vous avez consacré à m'accompagner dans ce projet.

À tous les enseignants et enseignantes qui m'ont aidé à atteindre ce succès depuis le début de mes études.

À tous mes chers amis, je souhaite exprimer ma gratitude pour votre amitié sincère et votre soutien indéfectible.

À Guennoun Ouissem, mon binôme et très chère amie, je veux te remercier pour les merveilleux moments que nous avons partagés tout au long de notre cursus universitaire. Ta présence à mes côtés a rendu cette expérience enrichissante et inoubliable. Tu as été une source d'inspiration et de motivation, et je suis reconnaissante d'avoir eu la chance de te connaître et de partager cette amitié avec toi.

Enfin, à toute ma famille et à tous ceux qui m'aiment et que j'aime.

Gadiri Nadjat

Dédicaces

À mes chers parents, je tiens à dédier ce travail qui est le fruit de mon parcours académique et professionnel, à vous qui avez été mes plus grands soutiens. Votre amour inconditionnel, votre soutien constant et votre présence inébranlable ont constitué la base de mon succès. Dans les moments de bonheur et de réussite, vous étiez toujours là pour partager ma joie. Votre dévouement et votre engagement envers moi n'ont jamais connu de limites. Votre confiance en moi a été un puissant moteur qui m'a poussé à me surpasser et à atteindre de nouveaux sommets.

Je souhaite également dédier ce travail à mes frères, qui ont toujours été là pour me soutenir et me protéger. Votre amour fraternel et votre attention ont été des cadeaux précieux qui ont enrichi ma vie et m'ont permis de grandir en tant que personne.

À mes encadrants et professeurs, je tiens à exprimer ma profonde reconnaissance pour votre guidance et votre soutien tout au long de mon parcours académique. Je me suis laissé inspirer par votre expertise, votre passion et votre dévouement. Vous m'avez guidé avec patience et bienveillance, m'encourageant à repousser mes limites et à viser l'excellence.

Et à mon binôme de travail, nous avons partagé des moments de travail intenses, relevé des défis stimulants et passé des heures à réfléchir ensemble. Chaque étape de ce parcours a été rendue plus facile et plus enrichissante grâce à notre étroite collaboration.

Ta perspicacité, ton engagement et ta passion ont été une source d'inspiration pour moi.

Ensemble, nous avons surmonté les obstacles, trouvé des solutions innovantes et partagé nos connaissances. Cette expérience restera avec moi en tant que temps de croissance personnelle et de succès partagé.

Enfin, à mes chers amis, je souhaite vous exprimer ma gratitude pour les moments merveilleux que nous avons partagés. Votre amitié a été un pilier essentiel dans ma vie, apportant bonheur, soutien et des souvenirs inoubliables. Chaque instant passé ensemble a été une source de joie et d'épanouissement.

Ainsi, à vous mes chers parents, à mes frères, à mes encadrants, professeurs et à mon binôme de travail, je vous dédie cette dédicace en signe de gratitude éternelle. Votre amour, votre soutien et votre confiance resteront gravés dans mon cœur tout au long de ma vie.

Avec tout mon amour et ma reconnaissance.

Guennoun Ouissem

Résumé

Les citoyens algériens sont fréquemment confrontés à des difficultés lorsqu'ils souhaitent obtenir des documents administratifs. Ils sont souvent contraints de se déplacer vers les établissements concernés, d'attendre en file pendant de longues heures, et ce malgré ces efforts, ils ne parviennent pas toujours à atteindre leurs objectifs en raison d'une mauvaise gestion des rendez-vous.

Dans ce contexte notre projet de fin d'étude et notre stage au sein de l'entreprise DATA MASTER propose la conception et le développement d'un système de gestion des rendez-vous pour les administrations algériennes. Notre objectif principal du système PlanIt est d'optimiser le processus de prise de rendez-vous, en réduisant les files d'attente et en améliorant l'efficacité du service public.

PlanIt permet aux citoyens de prendre des rendez-vous avec différentes administrations en utilisant une application mobile développer avec Flutter et facilite la gestion des rendez-vous des administrations par l'application web développé avec les Framework ReactJs et NodeJs.

Table de matières

Table de figures

Liste des tableaux

Liste des acronymes :

UML	Unified Modeling Language
API	Application Programming Interface
RDV	Rendez-vous
UP	Unified Process
DBMS	Database Management System
MVC	Modèle Vue Controlleur
UI	User Interface
UX	User Experience
IDE	Integrated Development Environment
XML	eXtensible Markup Language
HTTP	Hypertext Transfer Protocol
SQL	Structured Query Language
ORM	Object-Relational Mapping
PHP	Hypertext Processor

Introduction générale

Introduction générale

1. Contexte

Au cours des dernières décennies, le monde a connu un développement technologique dans divers secteurs, et ces avancées ont eu un impact significatif sur les administrations Algériennes.

Face à la croissance démographique, les administrations ont besoin de moyens de gestion et d'organisation, notamment pour le processus de gestion des rendez-vous. Ce processus est un élément important pour toute entreprise souhaitant offrir un service de qualité à ses clients et améliorer sa performance publique.

La gestion des rendez-vous dans les administrations consiste à planifier et organiser des réunions, des rencontres ou des rendez-vous avec des clients ou des partenaires commerciaux. Elle inclut la confirmation des rendez-vous, la gestion des annulations, des changements et les rappels.

Avec la popularisation d'Internet et l'augmentation significative de l'utilisation d'applications web et mobiles, de nombreuses entreprises optent pour l'utilisation de systèmes de gestion des rendez-vous. Ces systèmes facilitent la prise des rendez-vous pour les clients tout en simplifiant la gestion des rendez-vous pour le personnel de l'établissement.

C'est dans ce contexte que notre projet de fin d'études, a été mené au sein de l'entreprise DATA MASTER, vise à concevoir et développer un système de gestion des rendez-vous adapté aux besoins de l'administration algérienne.

2. Problématique

L'Algérie connaît une croissance démographique rapide, et la majorité des citoyens algériens rencontrent des difficultés lorsqu'ils doivent signaler des problèmes aux services communaux ou lorsqu'ils ont besoin de documents administratifs. Ils sont souvent contraints de se déplacer vers les établissements concernés, de faire la queue et d'attendre pendant de longues heures, ce qui entraîne une perte de temps et d'énergie. Malheureusement, en fin de compte, ils ne parviennent pas toujours à obtenir ce dont ils ont besoin en raison d'une mauvaise gestion des rendez-vous.

Cette mauvaise gestion des rendez-vous a des conséquences significatives sur la vie quotidienne des citoyens. Les retards dans l'obtention des services administratifs entraînent des frustrations et une insatisfaction générale. Les personnes sont obligées de se présenter à plusieurs reprises dans les établissements, espérant obtenir des réponses à leurs préoccupations, mais elles sont souvent confrontées à un manque de suivi et à des retards supplémentaires. Ces inefficacités dans la gestion des rendez-vous entraînent une perte de confiance des citoyens envers les administrations et créent un sentiment d'injustice dans l'accès aux services publics.

Le problème de la gestion des rendez-vous est exacerbé par l'ampleur de la demande de services administratifs en Algérie. Les infrastructures et les ressources humaines ne sont souvent pas suffisamment équipées pour répondre à cette demande croissante. De plus, les administrations peuvent souffrir d'un manque de coordination entre les différents services, ce qui entraîne une confusion pour les citoyens dans la prise de rendez-vous et la compréhension des procédures requises.

Les citoyens sont confrontés à des inconvénients spécifiques, tels que les longues files d'attente et le manque d'informations claires sur les procédures à suivre et les documents requis. Ils ont du mal à obtenir des rendez-vous à des moments qui leur conviennent, ce qui complique davantage leur accès aux services administratifs. Cette situation entraîne non seulement des frustrations individuelles, mais aussi un impact négatif sur l'efficacité et la productivité des administrations.

3. Objectifs

Notre projet de fin d’études a pour objectif de concevoir et développer un système de gestion des rendez-vous au sein de l’administration algérienne pour l’entreprise DATA MASTER.

Le système réalisé PlanIt conçu pour simplifier et optimiser le processus de planification et de suivi des rendez-vous comporte 3 composants principaux :

- Une application mobile pour les clients finaux permettant à chacun d’entre eux de réserver facilement leurs rendez-vous à tout moment et de n'importe où juste en quelques clics directement depuis leurs téléphones avec une communication bidirectionnelle entre l'application et le calendrier Google de l'utilisateur.

- Un site web dédié aux administrations Algériennes pour définir les fonctionnalités principales offertes par PlanIt.
- Une application web dédiée et mise à disposition du personnel professionnel pour gérer leurs rendez-vous et leurs plannings, gérer ses réservations, ses services et ses clients pour assurant la performance globale de l'établissement et l'efficacité opérationnelle et en réduisant les erreurs de plannings.

4. Environnement de stage

Dans le cadre de projet de fin d'étude, nous avons fait notre stage au sein de la boîte de développement informatique "Eurl Data Master". Ce stage est une collaboration entre notre faculté des sciences de l'université Abou Bekr BELKAID Tlemcen et l'entreprise Data Master pour la période du 15 février 2023 jusqu'à fin Juin 2023.

Eurl DATA MASTER a été créé en 2015, elle se trouve à Nassima résidence, Kiffane-Tlemcen. Ses principales fonctionnalités sont :

- Activités d´études et de conseils en matière de systèmes informatiques (matériels et logiciels)
- Activités de développement, production, fourniture et documentation de logiciels standards, (progiciels, utilitaires d´application, etc. ...), ainsi que leur édition
- Conseil et suivi des applications
- Assistance à l´installation ou à la mise en place d´équipements et matériels informatiques.

5. Organisation de mémoire

Notre manuscrit décrit l'essentiel du travail réalisé lors de ce projet. Il est organisé en trois chapitres, une introduction et une conclusion :

- Le mémoire est entamé par une introduction générale dans laquelle le contexte du sujet, la problématique, les objectifs à atteindre ainsi que l'environnement du stage sont présentés.
- Le premier chapitre consiste à décrire la description et critique de l'existant, et les besoins fonctionnels et non fonctionnels du système.
- Dans le deuxième chapitre, nous présentons la conception et les différents diagrammes qui modélise notre système.

- Le dernier chapitre est consacré à la présentation de système PlanIt réalisée ainsi que les outils de développement utilisés et les tests.
- Enfin, nous terminons ce mémoire par une conclusion dans laquelle nous présentons objectifs atteints et les perspectives à développer dans le futur.

Chapitre I :

Analyse et étude des besoins

I. Chapitre I : Analyse et étude de besoins

I .1. Introduction

Le marché des applications web et mobiles connait une forte croissance, l'utilisation de ses derniers devient très intéressante pour la majorité des citoyens algériens et utiles pour les administrations algériennes pour la gestion et l'organisation.

Dans ce chapitre nous allons définir qu'est-ce qu'un système de gestion des rendez-vous, faire une étude de l'existant, identifier les acteurs, présenter les besoins fonctionnels et non fonctionnels de notre système.

I.2. Définition d'un système de gestion de rendez-vous

Un système de gestion des rendez-vous est un ensemble de procédures, de technologies et d'outils qui permettent de planifier, organiser et suivre les rendez-vous entre différentes parties prenantes, tels que les clients, les patients, les employés ou les fournisseurs. Il vise à optimiser l'efficacité et la productivité de la gestion des rendez-vous, en automatisant les tâches liées à la planification, à la réservation, à la modification et à l'annulation des rendez-vous.

I.3. Etude de l'existant

Il existe de nombreuses applications mobiles de gestions des rendez-vous pour diverses entreprises :

I.3.1. TABIBE

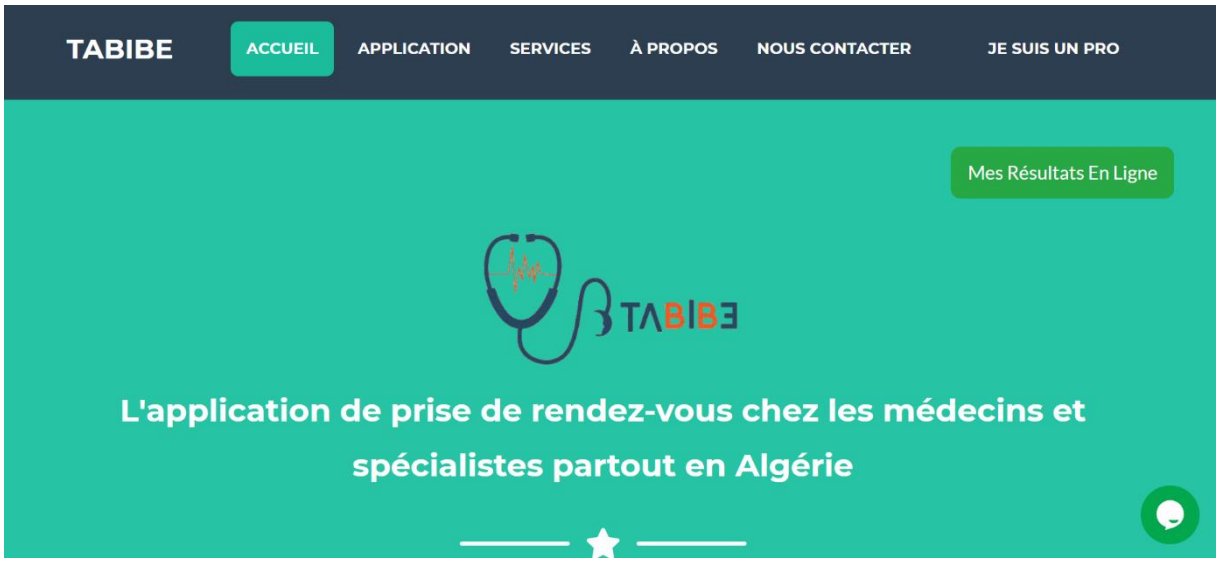


Figure 1: Plateforme "www.tabibe.fr"

TABIBE.fr : est une plateforme algérienne qui permet aux utilisateurs de prise de rendez-vous chez les médecins et spécialistes partout en Algérie

I.3.2. BLS INTERNATIONAL

Figure 2: Plateforme "https://algeria.blsspainvisa.com/"

algeria.blsspainvisa.com : est une plateforme qui permet aux utilisateurs de demande de visa pour l'Espagne au Algérie. Il fournit des informations sur la façon de demander selon des étapes à suivre :

- Connaissez vos besoins de visa
- Réserver votre rendez-vous
- Visitez le centre

1.3.3. eSiha

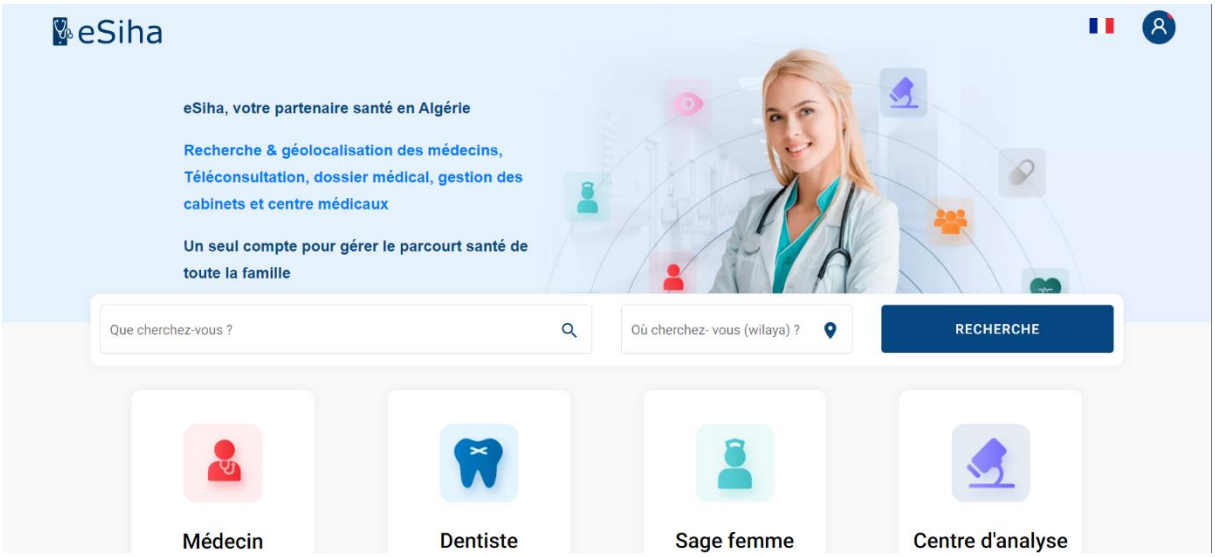


Figure 3 : Plateforme https://esiha.net/

eSiha.net : est une plateforme algérienne qui permet aux utilisateurs de :

- Recherchez et géo localiser votre professionnel de santé (toutes catégories et spécialités) - Prenez un RDV en ligne.

- Dossier et suivi médical

- Téléconsultation audiovisuelle

I.4. Identification des acteurs de systèmes

Client potentiel : un citoyen algérien qui ne dispose pas encore de compte pour bénéficier de l'application et réserver des rendez-vous, il peut juste accéder à l'application pour rechercher des établissements et pour l'inscription pour avoir un compte.

Client final : un citoyen algérien qui possède un compte normal pour la réservation des rendez-vous.

Super admin : un citoyen algérien qui est responsable de la gestion du système et la validation des comptes, il a aussi les fonctionnalités d'un client final.

Admin de l'établissement : un citoyen algérien qui possède une entreprise ou responsable d'un établissement, son rôle est de la création des services de l'établissement

et la création des comptes de ses employés et les chef service, il a tous les fonctionnalités d'un chef service, d'un employé et d'un client final

Chef service : c'est celui qui est responsable de service et ses employés, il a les fonctionnalités d'un employé et d'un client final.

Employé : c'est celui qui responsable des rendez-vous qui se passe au bureau et il responsable de la gestion les clients, il a aussi les fonctionnalités d'un client final.

I.5. Spécification des besoins

I.5.1. Besoins fonctionnels

- **Authentification :** chaque utilisateur de système doit avoir un compte pour accéder à son compte personnel.

1.5.1.1. Profile client potentiel

- **Chercher une entreprise :** l'application mobile permet au client potentiel de chercher une entreprise dont lequel il besoin. La recherche est multicritère : par nom de l'entreprise, sa localisation, …
- **L'inscription**
  - (Pour avoir un compte normal) : après la recherche d'une entreprise et l'accès aux détails de cette dernière, l'application mobile oblige le client potentiel à s'inscrire pour avoir un compte normal pour qu'il puisse consulter les offres de l'entreprise et réserver des rendez-vous. Donc il doit préalablement demander l'inscription en remplissant un formulaire contenant son nom, son prénom, son adresse email, son adresse personnelle, son id national et une photo avec sa carte d'identité.
  - (Pour avoir un compte professionnel) : les administrations souhaitant utiliser notre système de gestion des rendez doit disposer d'un compte professionnel, donc l'admin de l'entreprise doit s'inscrire en replissant un formulaire contenant son nom, prénom, le nom de l'entreprise, l'adresse de l'entreprise, l'email de l'entreprise et son type. Les informations seront validées par le super admin de l'application par un rendez-vous avec un admin de l'entreprise.

1.5.1.2. Profile client final

- **Gestion de profile :** Permet à l'utilisateur de modifier son mot de passe, son login, sa photo de profile, son adresse ou d'autre coordonnées ou de les consulter en détails.
- **Gestion de notifications :** Permet de consulter les rappels et les messages ou de les supprimer ou répondre, d'accepter les notifications des rendez-vous ou les refuser.
- **La recherche d'une solution à un problème :** L'application permet au client final de chercher une solution à son problème parmi la liste des solutions proposées par les instituts, il peut aussi consulter les solutions en détail.
  Il peut enregistrer la solution dans la liste des favoris, demander le service associé à cette solution en réservant un rendez-vous.
- **Réservation des rendez-vous :** L'application permet au client final de prendre un rendez-vous avec un établissement précis en vérifiant la disponibilité du client avec son compte Google Calendar. Elle permet aussi au client de demander un rendez-vous selon sa disponibilité et aussi la disponibilité de l'établissement auquel il souhaite réserver un rendez-vous.
- **Gestion des rendez-vous :** Offre la possibilité de modifier ou annuler un rendez-vous en informant l'autre participant à ce rendez-vous ainsi que la possibilité de consulter en détail un rendez-vous et de le chercher par date.

1.5.1.3. Profile super admin

- **Validation de l'inscription :** Il valide l'inscription des entreprises pour avoir un compte professionnel en réservant un rendez-vous avec un admin de l'entreprise en validant l'identité de l'entreprise.

1.5.1.4. Profile admin de l'établissement

- **Gestion des services :** Permet à l'admin de l'établissement d'ajouter un service en précisant leur chef, de modifier le service ou de le supprimer.
- **Création des comptes des employés et les autres admin.**

1.5.1.5. Profile chef service

- **Gestion de bureau :** Permet au chef de service d'ajouter un bureau, de le modifier ou le supprimer ou d'ajouter des employés à ce bureau ou de les retirer.

- **Gestion de service :** Permet au chef service d'ajouter un employé à son service, de le retirer ou de préciser la visibilité de service et sa disponibilité.
- **Gestion des solutions :** Permet au chef service d'ajouter des solutions à des problèmes posés par les citoyens, de modifier ses solutions ou les supprimer.
- **Gestion des propriétés des rendez-vous :** Permet de préciser la durée de rendez-vous, et les jours disponibles pour prendre ou demander un rendez-vous.

1.5.1.6. Profile employé

- **Gestion des rendez-vous de bureau :** Donne la possibilité de reporter un rendez-vous en informant les participants, des modifier les paramètres de rendez-vous, de consulter la liste des rendez-vous.
- **Gestion des demandes :** Permet à l'employé d'affecter le rendez-vous au client approprié, ou de refuser les demandes.

I.5.1. Besoins non fonctionnels

- **Fiabilité** : le site, l'application mobile et l'application web fonctionnent sans erreurs.
- **Sécurité** :

- L'utilisation de **JSON Web Token :** Un JSON Web Token est un accès token (jeton d'accès) aux normes RFC 7519 qui permet un échange sécurisé de donnée entre deux parties. Il contient toutes les informations importantes sur une entité, ce qui rend la consultation d'une base de données superflue et la session n'a pas besoin d'être stockée sur le serveur. [1]
- Mot de passe crypté avec la possibilité de les réinitialiser.
- L'utilisation de l'API Google Sign-In pour la gestion de flux d'authentification et le cycle de vie des jetons.

- **Confidentialité :** les employés et les gérants ont accès qu'aux services et fonctionnalités qui leurs appartient.
- **Portabilité :** le système est multiplateforme : il fonctionne sur les différents navigateurs, sur différents systèmes d'exploitation et sur différents appareils.
- **Utilisabilité :** une interface facile à utiliser avec un design clair.

I.6. Conclusion

La gestion des rendez-vous est un élément essentiel de la gestion du temps et de la planification pour les administrations. Notre système a pour but de rassembler les entreprises qui veulent faciliter le processus de gestion des rendez-vous.

Dans ce chapitre nous avons introduit les différents besoins de notre système et les systèmes concurrents.

Dans le chapitre suivant nous allons introduire la conception de notre système PlanIt avec les différents diagrammes utilisés.

Chapitre II :

Conception du système PlanIt

II. Chapitre 2 : Conception du système

II.1. Introduction

Dans ce chapitre, nous allons présenter la conception de notre système en utilisant le langage de modélisation UML (Unified Modeling Language) et en suivant la méthode de développement UP (Unified Process).

II.2. Modélisation conceptuelle

II.2.1. Définition du langage UML

L'UML *(Unified Modeling Language* ou Langage de modélisation unifiée en français) est un langage graphique de modélisation informatique. Ce langage est désormais la référence en modélisation objet, ou programmation orientée objet. Cette dernière consiste à modéliser des éléments du monde réel (immeuble, ingrédients, personne, logos, organes du corps...) ou virtuel (temps, prix, compétence...) en un ensemble d'entités informatiques appelées « objet ». [2]

II.2.2. Définition de la méthode de développement UP

La méthode du Processus Unifié (UP pour Unified Process) est un processus de développement qui possède les caractéristiques suivantes :

- Itératif et incrémental
- Pilotés par les cas d'utilisation
- Centré sur l'architecture.

Le projet informatique est découpé en plusieurs phases très courtes à l'issue desquelles un incrément est livré.

La méthode du processus unifié s'appuie principalement sur la modélisation et les diagrammes UML pour la description de l'architecture physique du logiciel (plateforme, serveur, postes etc…) et la mise au point de cas d'utilisation permettant d'identifier et retranscrire les besoins des utilisateurs. [3]

II.2.3. Les diagrammes UML :

Pour commencer, nous avons utilisé des diagrammes de cas d'utilisation pour identifier les différentes fonctionnalités du système et les interactions entre les utilisateurs et le système lui-même. Ces diagrammes permettent de visualiser les acteurs impliqués, les cas d'utilisation et les flux d'exécution des fonctionnalités.

Ensuite, nous avons créé des diagrammes de séquence pour représenter l'ordre chronologique des messages échangés entre les objets du système lors de l'exécution des cas d'utilisation.

En complément, nous avons élaboré des diagrammes de classe pour représenter la structure statique du système, y compris les classes, les attributs et les relations entre les classes.

Enfin, nous avons utilisé le diagramme de Gantt pour illustrer la planification du projet et la gestion du temps.

II.2.3.1. Diagrammes de cas d'utilisation :

II.2.3.1.1. Diagramme de cas d'utilisation « profile Client Potentiel »

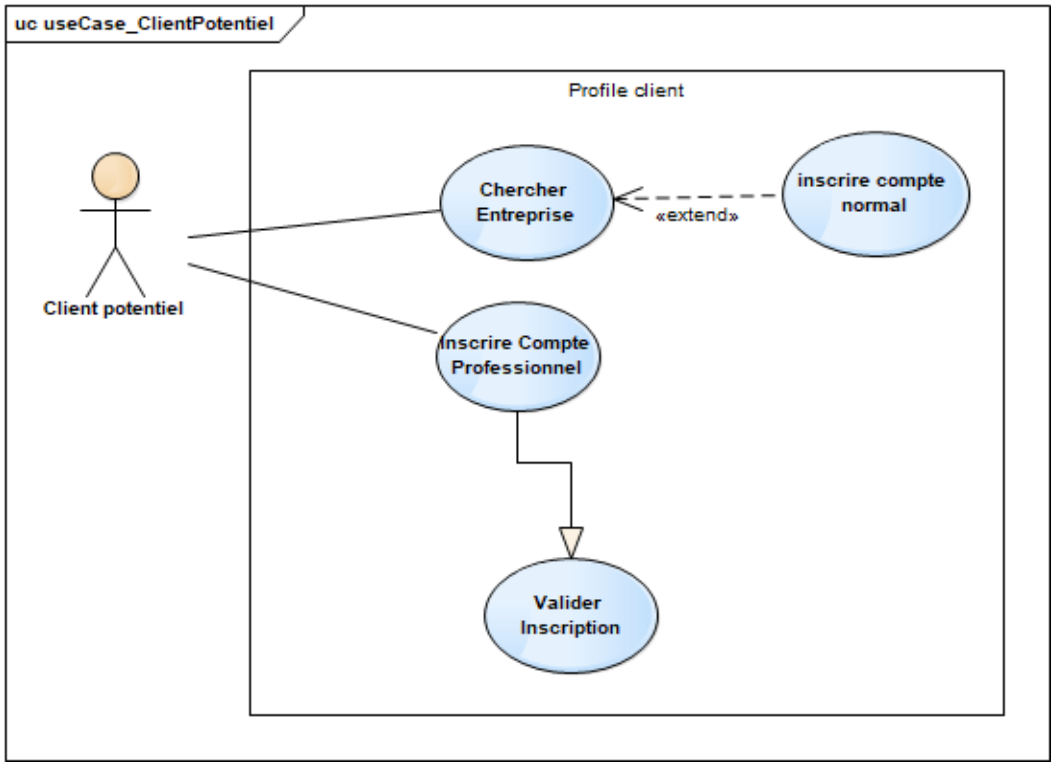


Figure 4 : diagramme de cas d'utilisation « profile client potentiel »

Description textuelle

Cas d'utilisation	Inscrire compte professionnel
Acteur	Client potentiel
Scénarios	1. Le client potentiel demande la page d'inscription sur le site web de PlanIt. 2. Le système affiche le formulaire d'inscription. 3. Le client potentiel saisit les données de l'administration. 4. Le système envoie les données de l'entreprise au super admin de système PlanIt pour les vérifier.
Post-conditions	Il permet le client potentiel d'avoir un compte professionnel pour une administration .

Tableau 1 : cas d'utilisation inscrire compte professionnel

II.2.3.1.2. Diagramme de cas d'utilisation « profile Client final »

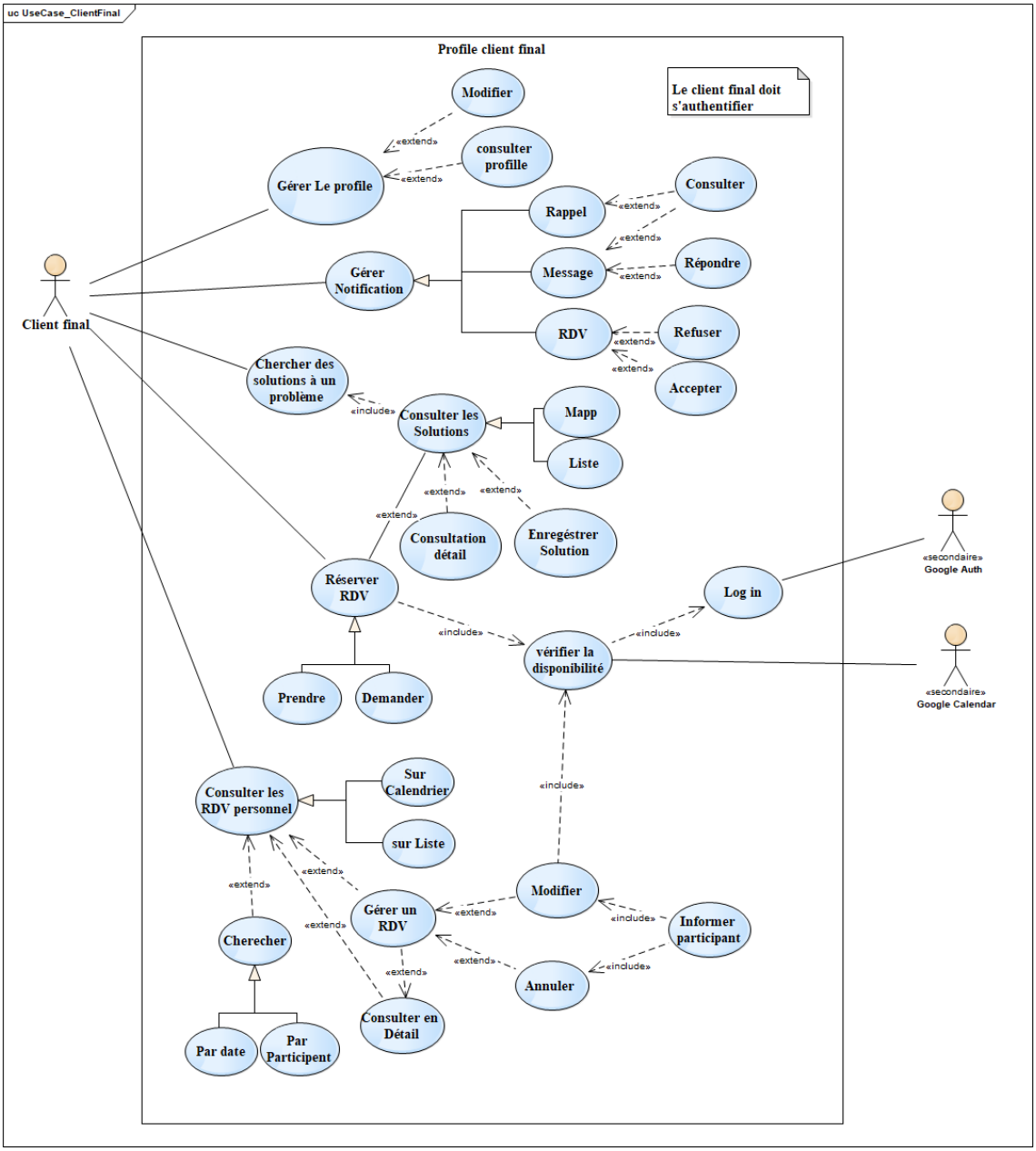


Figure 5: diagramme de cas d'utilisation « profile client potentiel »

Description textuelle

Cas d'utilisation	Chercher des solutions à un problème
Acteur	Client final
Pré-condition	Le client final est authentifié
Scénarios	1. Le client final demande la page des solutions. 2. Le système affiche la page des solutions. 3. Le client final cherche une solution à son problème en entrant le nom de solution. 4. Le système affiche le résultat de recherche. 5. Le client final examine les résultats de recherche pour trouver une solution à son problème et le consulter.
Post-conditions	Il permet le client final de trouver une solution satisfaite pour son problème.

Tableau 2 : cas d'utilisation chercher des solutions à un problème

Cas d'utilisation	Réserver RDV
Acteur	Client final
Pré-condition	Le client final est authentifié
Scénarios	1. Le client accède à la page des solutions. 2. Le système affiche la page des solutions. 3. Le client final choisi une solution selon son besoin. 4. Le système affiche les détails de cette solution. 5. Le client final clique sur ajouter un rendez-vous avec l'administration qui offre cette solution. 6. Il effectue sa réservation selon la date et l'heure disponible. 7. Il confirme sa réservation en cliquant sur confirmer ou l'annuler. 8. Le système confirme sa réservation en envoyant un message de succès.
Post-conditions	Il permet le client final de réserver un rendez-vous avec l'administration qui veut avec la date et l'heure qui lui convient.

Tableau 3 : cas d'utilisation réserver RDV.

Cas d'utilisation	Gérer notification
Acteur	Client final
Pré-condition	Le client final est authentifié
Scénarios	1. Le client accède à la page des notifications. 2. Le système affiche la page des notifications. 3. Le client final consulte les rappels et répond aux messages.

Tableau 4 : cas d'utilisation gérer notifications.

II.2.3.1.3. Diagramme de cas d'utilisation « profile Chef Service »

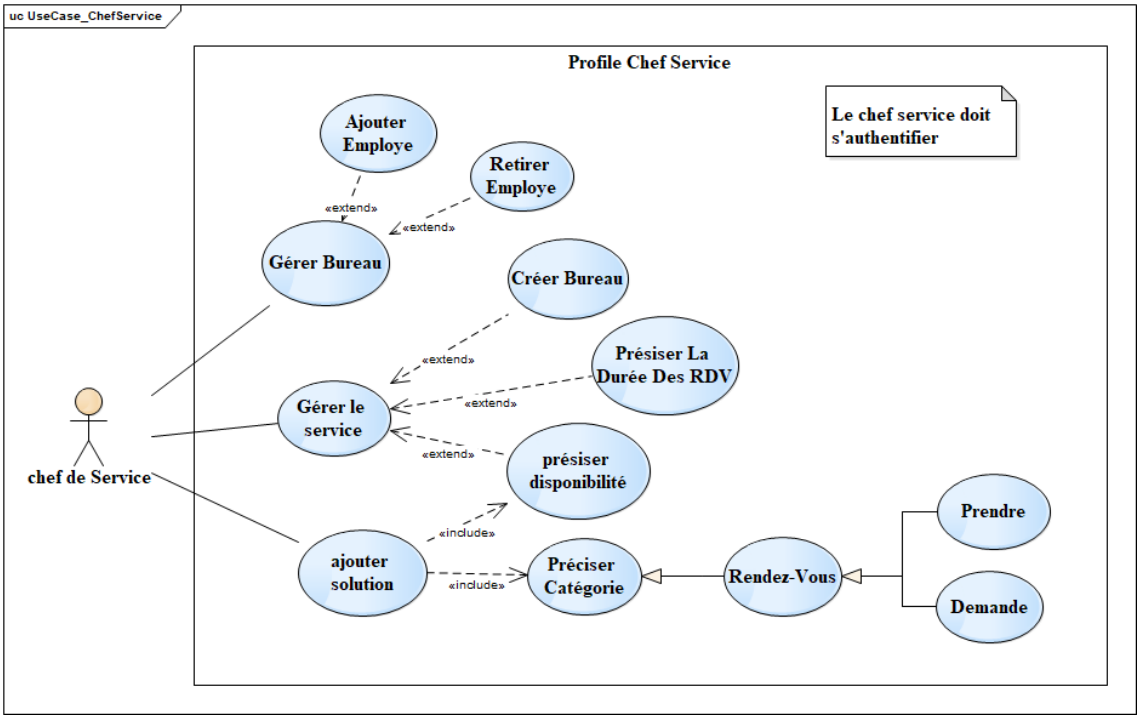


Figure 6: diagramme de cas d'utilisation « profile Chef Service »

Description textuelle

Cas d'utilisation	Ajouter employé
Acteur	Chef de service
Pré-condition	Le chef de service est authentifié
Scénarios	1. Le chef de service demande la page d'ajouter employé. 2. Le système affiche la page souhaitée. 3. Le chef de service entre le nom d'employé avec ses données pour vérifier s'il existe un utilisateur avec ses données sinon il le crée un nouveau compte. 4. Le chef de service confirme l'insertion en cliquant sur ajouter. 5. Le système valide son ajout en affichant un message de succès.
Post-conditions	Il permet le chef de service d'ajouter des employés pour son service.

Tableau 5 : cas d'utilisation ajouter employé

Cas d'utilisation	Ajouter solution
Acteur	Chef de service
Pré-condition	Le chef de service est authentifié
Scénarios	1. Le chef de service demande la page d'ajout de solutions. 2. Le système affiche la page souhaitée. 3. Le chef de service entre le nom de solution avec sa description et sa catégorie 4. Le chef de service confirme l'insertion en cliquant sur ajouter. 5. Le système valide son ajout en affichant un message de succès.

Tableau 6 : cas d'utilisation ajouter solution

II.2.3.1.4. Diagramme de cas d'utilisation « profile Employé »

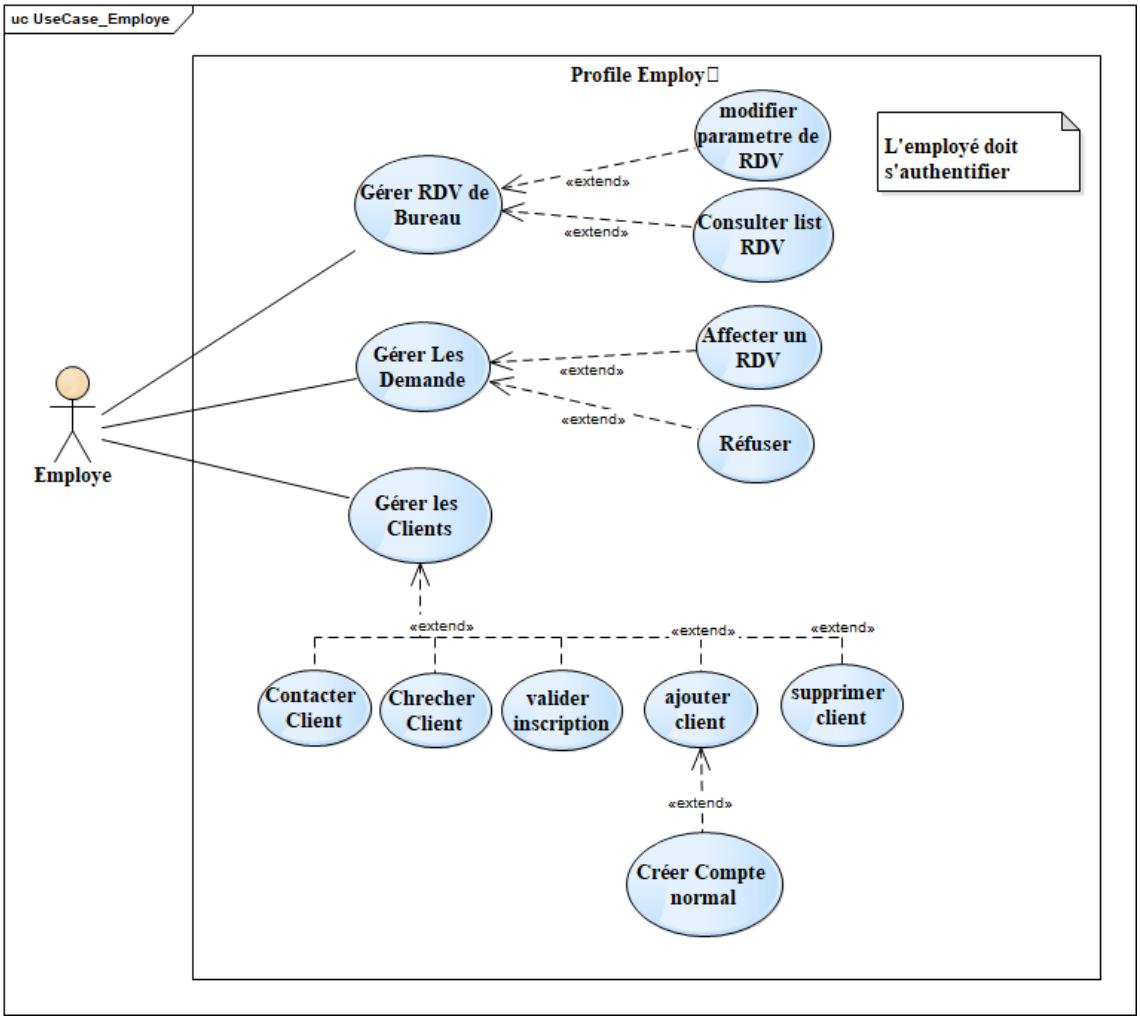


Figure 7 : diagramme de cas d'utilisation « profile Employé »

Description textuelle

Cas d'utilisation	Gérer les clients
Acteur	Employé
Pré-condition	L'employé est authentifié
Scénarios	1. L'employé demande la page des clients 2. Le système affiche la page souhaitée. 3. L'employé demande le formulaire pour ajouter un client, le chercher, le supprimer, le modifier. 4. Le système affiche le formulaire demandé par l'employé, ou demande la confirmation de suppression un client. 5. L'employé saisit les données de client qui souhaite l'ajouter ou confirmer ou annuler la demande de suppression. 6. Le système valide les données saisis.

Tableau 7 : cas d'utilisation Gérer les clients

II.2.3.1.5. Diagramme de cas d'utilisation « profile Admin de l'établissement »

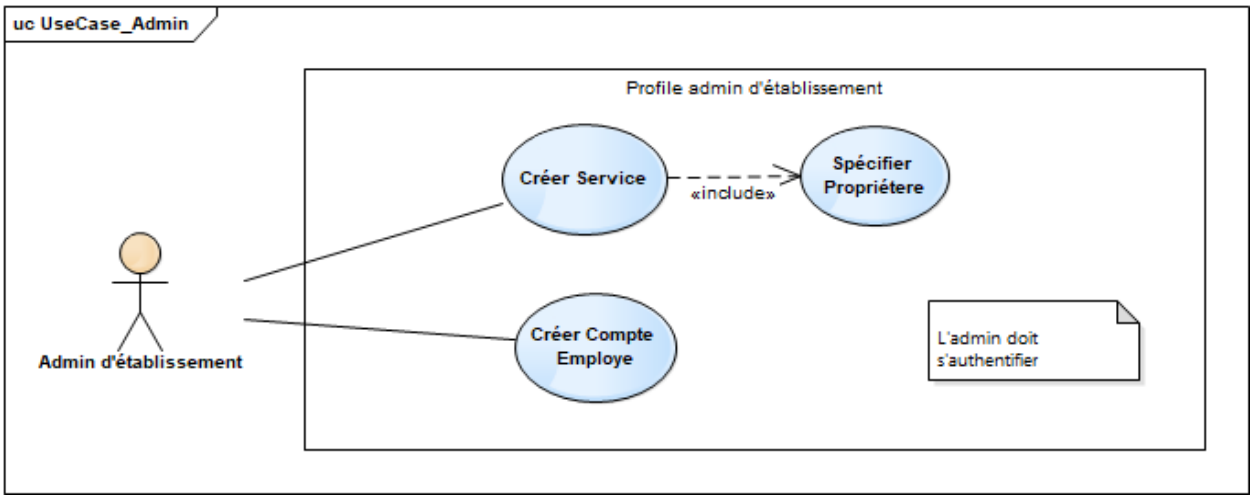


Figure 8: diagramme de cas d'utilisation « profile Admin de l'établissement »

Description textuelle

Cas d'utilisation	Créer service
Acteur	Admin d'établissement
Pré-condition	L'admin d'établissement est authentifié
Scénarios	1. L'admin d'établissement demande la page de création des services. 2. Le système affiche la page souhaitée. 3. L'admin d'établissement entre le nom de service. 4. Le système demande au admin d'établissement de spécifier le propriétaire. 5. L'admin d'établissement sélectionne le nom de propriétaire pour le service. 6. L'admin conforme la création en cliquant sur le bouton créer ou annuler la création.

Tableau 8 : cas d'utilisation créer service

II.2.3.1.6. Diagramme de cas d'utilisation « profile Super Admin »

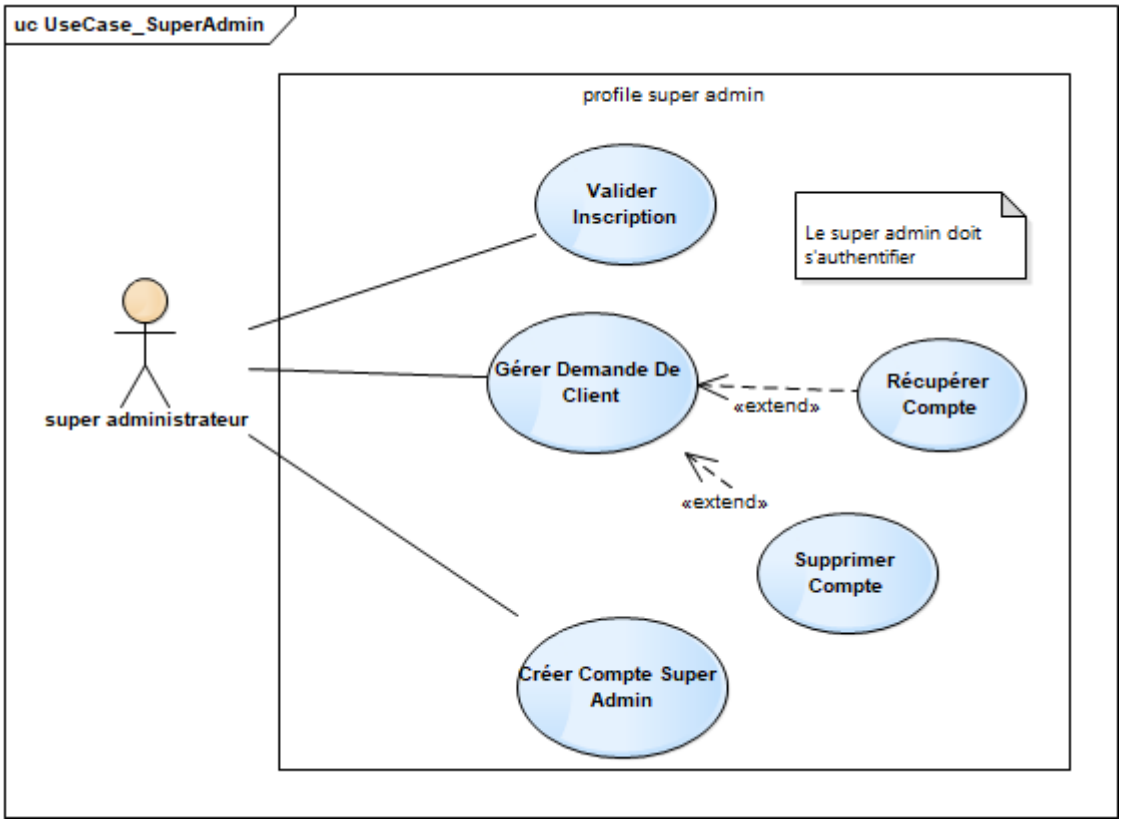


Figure 9 :diagramme de cas d'utilisation « profile Super Admin »

Description textuelle

Cas d'utilisation	Valider l'inscription
Acteur	Super admin
Pré-condition	Le Super admin est authentifié
Scénarios	1. Le super admin demande la page des demandes d'inscription des clients. 2. Le système affiche la page. 3. Le super admin consulte les demandes un par un et les valide ou les refuser.
Post-conditions	Il permet l'obtention des comptes validés.

Tableau 9: cas d'utilisation valider l'inscription

II.2.3.2. Diagrammes de séquences

II.2.3.2.1. Diagramme de séquence « Authentification »

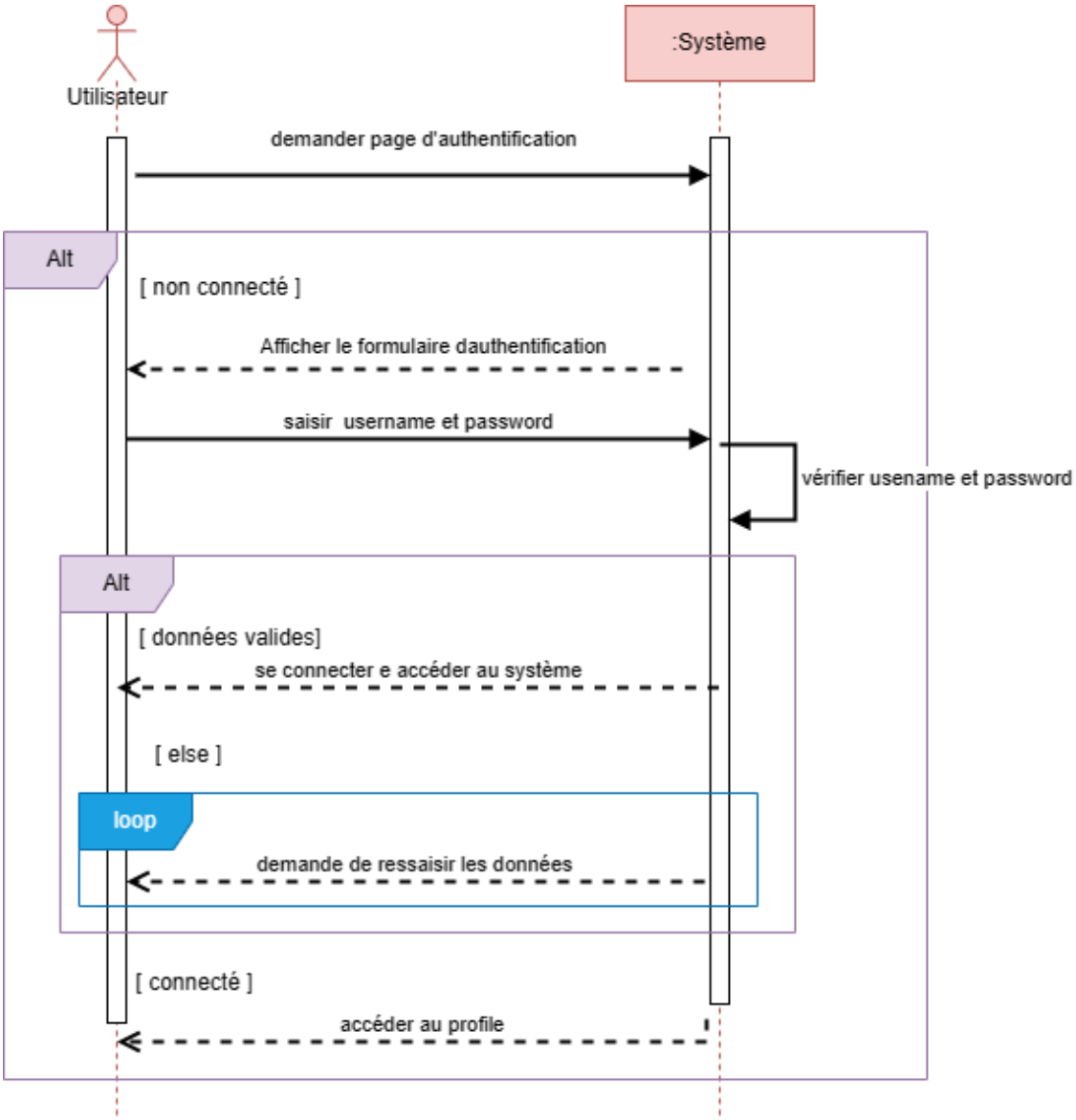


Figure 10: diagramme de séquence « Authentification »

II.2.3.2.2. Diagramme de séquence « Création de solution »

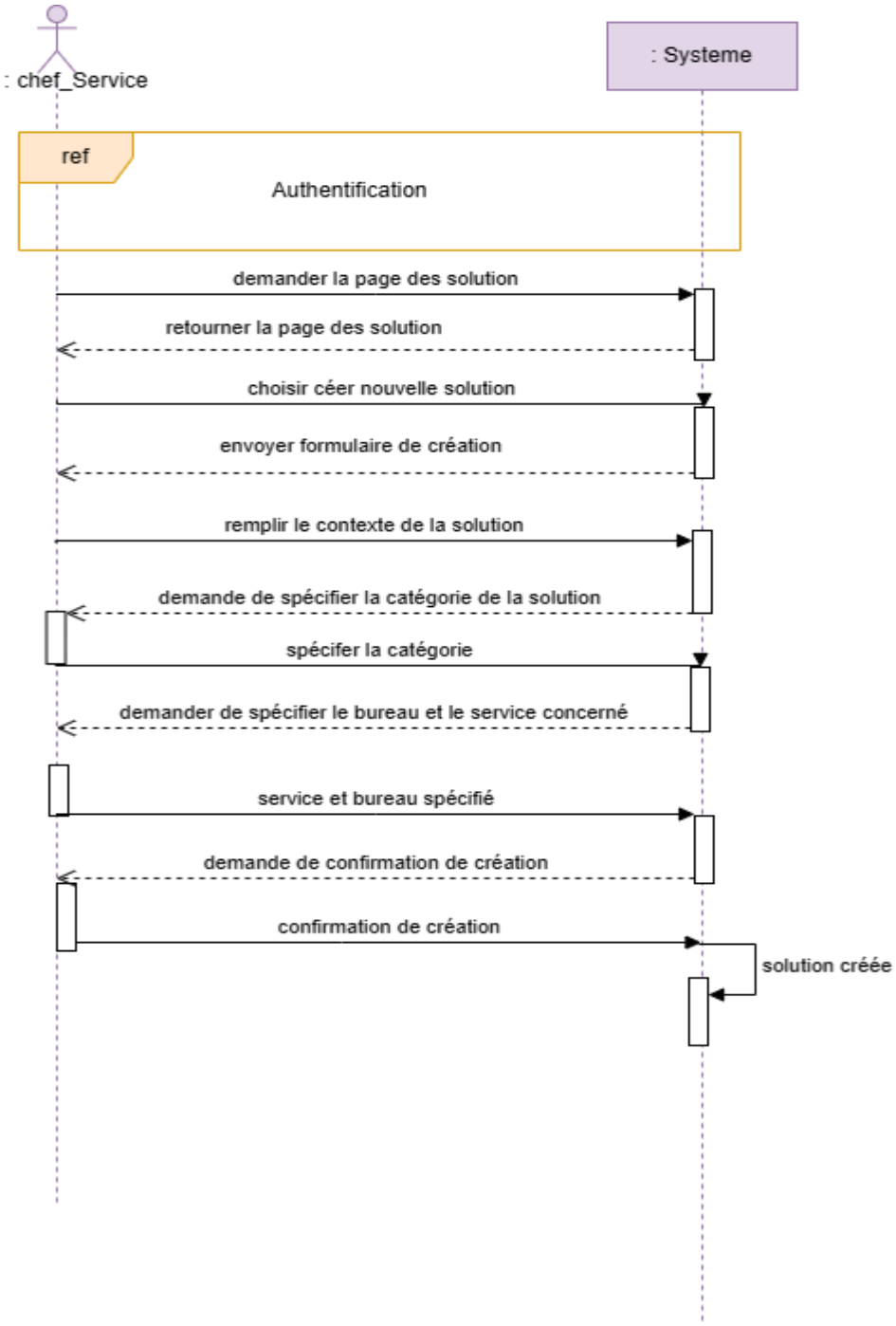


Figure 11 : diagramme de séquence « Création de solution »

II.2.3.2.3. Diagramme de séquence « réserver rendez-vous »

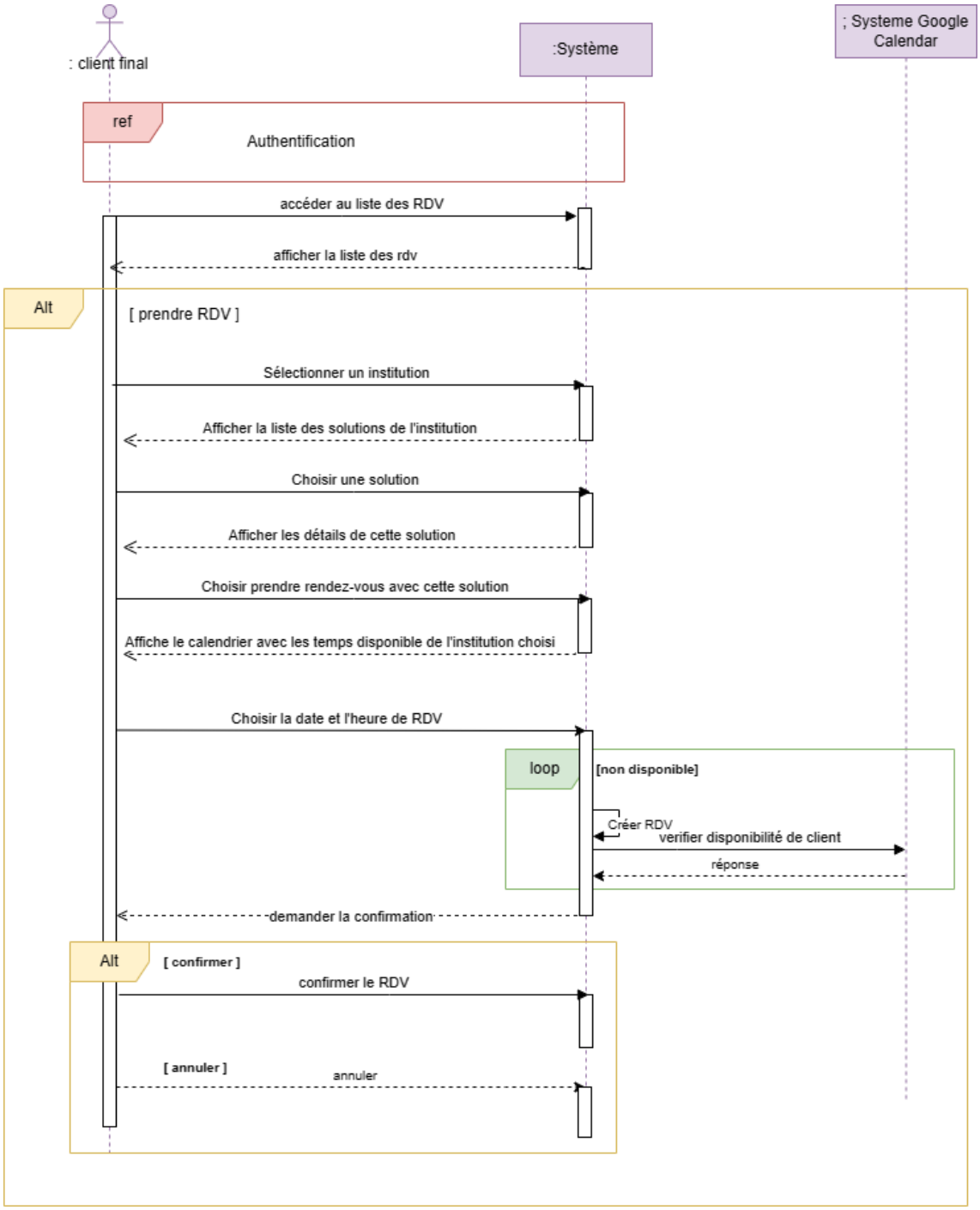


Figure 12: diagramme de séquence « Réserver rendez-vous »

II.2.3.2.4. Diagramme de séquence « Gestion es demandes »

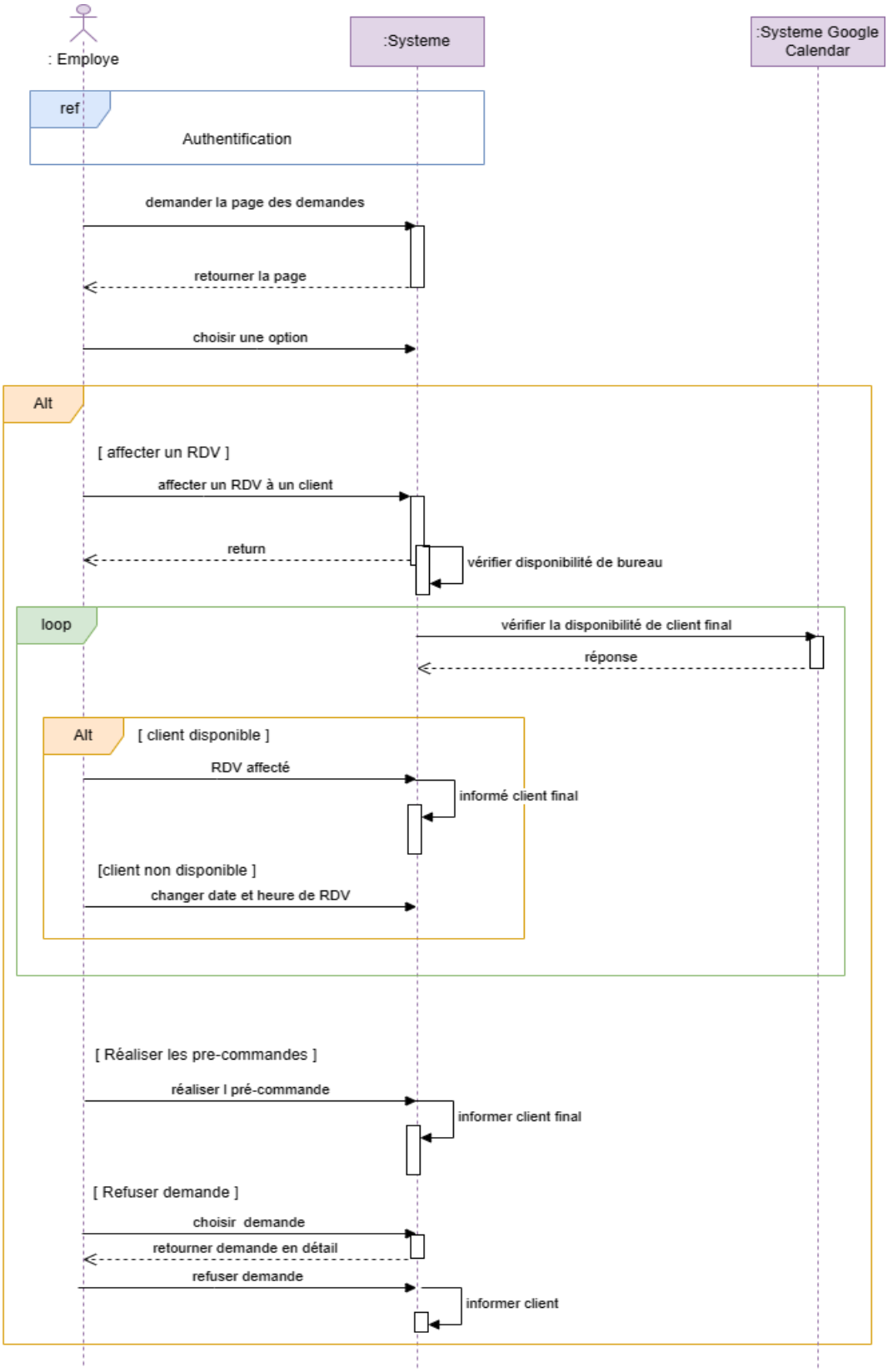


Figure 13: diagramme de séquence « Gestion des demandes »

II.2.3.3. Diagramme de classes

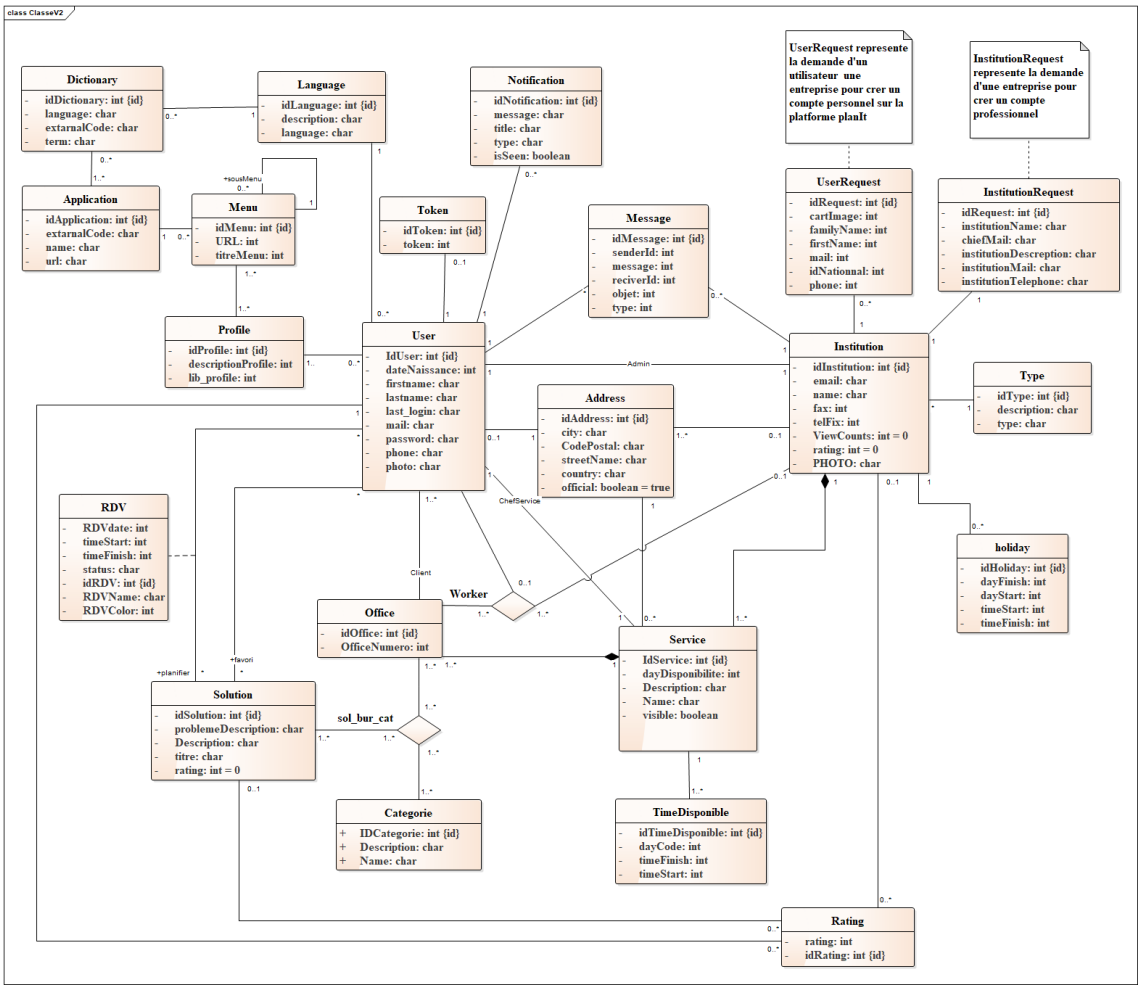


Figure 14: diagramme de classes

II.2.3.4. Diagramme de Gantt

PlanIt Project

	Nom de la tâche	Début	Fin
1	1er Réunion avec Mr Semmoud Abderrazak pour l'explication de thème	23/01/23	23/01/23
2	Faire des recherche concernant le thème gestion des rendez-vous	23/01/23	30/01/23
3	Extraire les besoins et discuter les outils et les technologies qui seront utilisés	30/01/23	30/01/23
4	Suivre les formations sur Flutter, ReactJs et NodeJs	30/01/23	14/02/23
5	Installation des environnement et outils de travail	14/02/23	14/02/23
6	Premier jour de stage dans Data Master	15/02/23	15/02/23
7	Réalisation de la partie introduction générale du rapport	15/02/23	15/02/23
8	Réalisation de diagramme de cas d'utilisation et de séquence	16/02/23	19/02/23
9	Réalisation du chapitre 1 du rapport	19/02/23	20/02/23
10	Réalisation de diagramme de classe	20/02/23	21/02/23
11	Réalisation du prototype de l'application mobile PlanIt	22/02/23	15/03/23
12	Création de la base de données	23/02/23	24/02/23
13	Réalisation du chapitre 2 du rapport	01/03/23	03/03/23
14	Réalisation de l'interface de l'application mobile	02/03/23	20/03/23
15	Création de backend application mobile	04/03/23	01/05/23
16	Intégration du backend avec flutter	10/03/23	20/05/23
17	Création du site web	30/03/23	04/04/23
18	Création de l'interface d l'application web	10/04/23	20/04/23
19	Intégration du backend avec l'application web	30/04/23	10/05/23
20	L'ajout de la partie profile et gestion des utilisateurs dans l'application web	15/05/23	20/05/23
21	Réalisation du chapitre 3 du rapport	25/05/23	27/05/23
22	l'ajout de la partie test	01/06/23	01/06/23
23	Réalisation de la conclusion générale du projet	03/06/23	03/06/23
24	Test et correction des bugs et conflits	05/06/23	08/06/23
25	Rassembler les chapitres dans un seul document	10/06/23	12/06/23
26	Correction des fautes	18/06/23	19/06/23
27	Préparation du PPT de soutenance	17/06/23	20/06/23
28	Préparation du soutenance	20/06/23	25/06/23

Figure 15: Diagramme de Gantt

II.2.4. Prototype et design d'interface d'utilisateur

Dans cette partie nous abordons le prototype de notre système ainsi le design de l'interface utilisateur en mettant en évidence l'importance de l'expérience utilisateur (UX) dans la conception du prototype, en mettant l'accent sur la facilité d'utilisation, l'ergonomie et l'esthétique des interfaces.

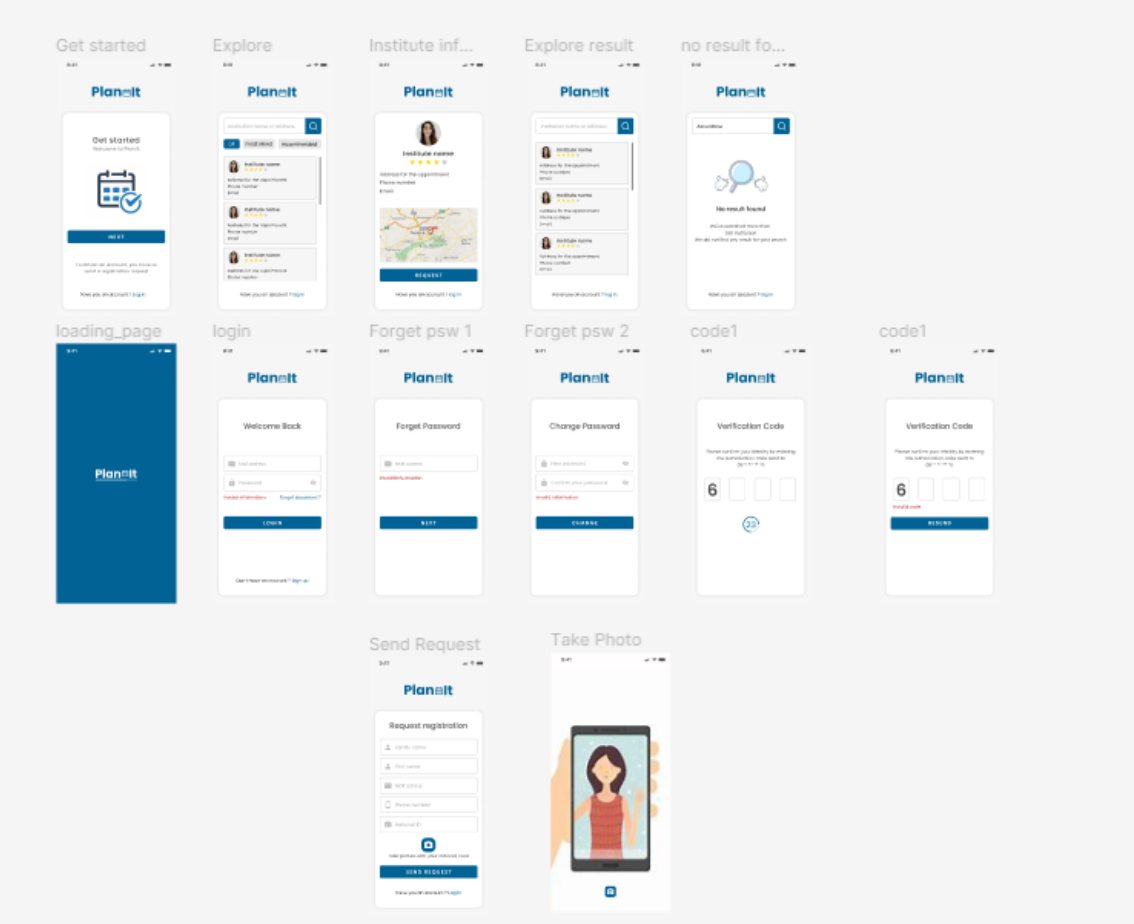


Figure 16: Design d'une partie d'interface utilisateur pour l'application mobile

II.3. Conclusion

Dans ce chapitre nous avons réalisé la conception de notre système PlanIt avec les trois diagrammes UML : diagramme de cas d'utilisation, diagramme de séquence, diagramme de classe.

Chapitre III :

Développement du système PlanIt

III. Chapitre III : Réalisation et développement

III.1. Introduction

Dans ce chapitre de ce mémoire, nous allons aborder la réalisation et le développement de notre application web et mobile. Pour ce faire, nous allons nous intéresser aux différentes API et technologies utilisées, ainsi qu'aux outils qui ont été nécessaires pour mener à bien ce projet.

III.2. Architecture du système « PlanIt »

III.2.1. Présentation de l'architecture client/serveur à trois niveaux

L'architecture client/serveur à trois niveaux, également appelée architecture 3-tiers, est une approche de conception de logiciel qui sépare la présentation des données, la logique métier et la gestion des données en trois couches distinctes. Cette architecture permet de séparer clairement les différentes responsabilités de l'application, ce qui facilite le développement, la maintenance et l'évolutivité du système. [4]

Dans cette présentation, nous allons explorer les différentes couches d'une architecture à trois niveaux, en expliquant ce que chacune fait et comment elles communiquent entre elles pour créer une application efficace et fonctionnelle.

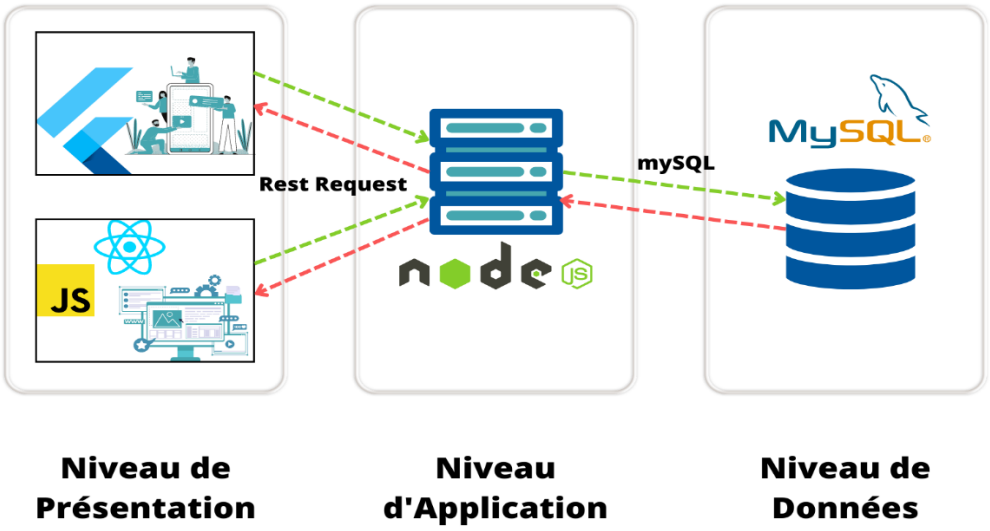


Figure 17: Architecture du système “PlanIt”

- **Le niveau de présentation** : est l'interface utilisateur et la couche de communication de l'application, où l'utilisateur final interagit avec l'application. Son objectif principal est d'afficher des informations et de collecter des informations auprès de l'utilisateur. Dans notre cas on utilise les technologies ReactJs et flutter [4]

- **Le niveau Application (**niveau logique ou niveau intermédiaire**)** est : le cœur de l'application. Dans ce niveau, les informations collectées dans le niveau Présentation sont traitées. Il peut également ajouter, supprimer ou modifier des données dans le niveau Données. [4]

- **Le niveau de données (**niveau de base de données, niveau d'accès aux données**)** est : l'endroit où les informations traitées par l'application sont stockées et gérées. Il s'agit de la couche inférieure de notre application qui stocke les données dans une base de données. [4]

Le niveau Présentation utilise une API REST pour communiquer avec le niveau Application, permettant ainsi une communication efficace et sécurisée entre les deux couches. Le niveau Application, quant à lui, communique avec le niveau de données via un système de gestion de base de données (DBMS) qui permet de stocker, récupérer et gérer les données de manière organisée et efficace.

III.2.2. Les avantages de l'architecteur à 3 niveaux

- **Séparation des préoccupations** : Les différentes couches de l'architecture sont responsables de tâches spécifiques, ce qui permet une séparation claire des préoccupations. Cela facilite la maintenance et l'évolution de l'application.

- **Modularité** : Les différentes couches peuvent être développées indépendamment les unes des autres, ce qui facilite la gestion des équipes de développement et accélère le développement de l'application.

- **Sécurité** : L'utilisation de couches distinctes pour la présentation, l'application et les données permet de renforcer la sécurité de l'application, en limitant l'accès aux données sensibles.

- **Réutilisabilité** : Les différents composants de l'application peuvent être réutilisés dans d'autres projets, ce qui facilite la création d'applications similaires.

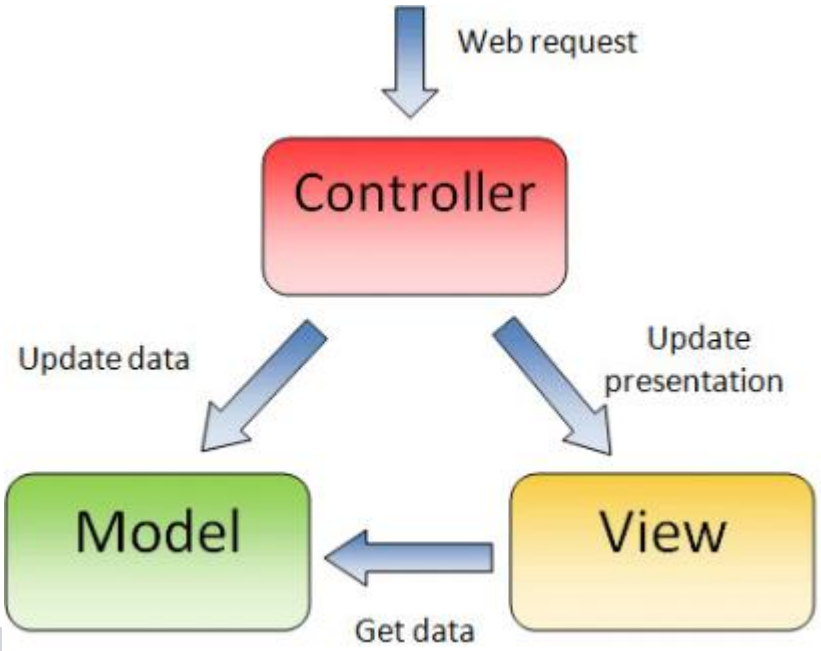
III.2.3. Design Pattern MVC

L'architecture *Modèle/Vue/Contrôleur* (MVC) est une façon d'organiser une interface graphique d'un programme. Elle consiste à distinguer trois entités distinctes qui sont, le *modèle*, la *vue* et le *contrôleur* ayant chacun un rôle précis dans l'interface. [5]

Le modèle représente les données et la logique métier de l'application. Il est responsable de la gestion et de la mise à jour des données de l'application, ainsi que de la manipulation de la logique métier.

La vue représente la présentation de l'interface utilisateur de l'application. Elle est responsable de l'affichage des données et de la réponse aux événements utilisateur.

Le contrôleur est responsable de la gestion des événements utilisateur et de la communication entre le modèle et la vue. Il reçoit les demandes des utilisateurs, interagit avec le modèle pour traiter les données et met à jour la vue pour refléter les changements



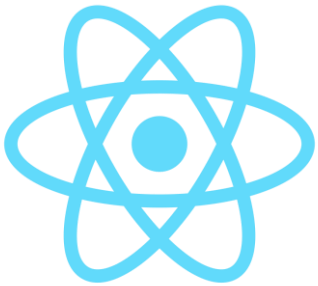
[6]

Figure 18: Modèle MVC

III.3. Technologies et Outils utilisés

III.3.1. Technologies

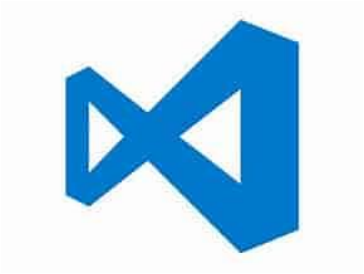
Logo	Description
	**Flutter :** Framework de développement d'applications multiplateforme, conçu par Google, dont la première version a été publiée sous forme de projet open source à la fin de l'année 2018. Flutter met à disposition une grande variété de bibliothèques d'éléments d'IU standard pour Android et iOS. [7]
	**Dart :** Langage de programmation, qui a été développé principalement par Google. Il est standardisé par Ecma (Ecma est une organisation internationale de normalisation des systèmes d'information et de communication ainsi que des appareils électroniques grand public). [8]
	**Node JS :** Sert à faire du Javascript server side, peut être utilisé dans des applications de bases de données, la plus populaire étant MySQL. Node.js est souvent confondu avec ce dernier, car c'est sa base : créer des applications en temps réel, où le serveur a la possibilité de transmettre de l'information au client. [9]

	**JavaScript :** Langage de programmation dynamique complet qui, appliqué à un document HTML, peut fournir une interactivité dynamique sur les sites Web. [10]
	**React :** Bibliothèque JavaScript déclarative, efficace et flexible pour construire des interfaces utilisateurs (UI). Elle vous permet de composer des UI complexes à partir de petits morceaux de code isolés appelés « composants ». [11]
	**Nodemailer :** Bibliothèque facile à utiliser qui permet d'envoyer des e-mails à l'aide de Node.js. Il fournit une API simple et puissante pour l'envoi d'e-mails à l'aide de différentes méthodes de transport, telles que SMTP, Sendmail et Amazon SES. Il prend en charge plusieurs services de messagerie, notamment Gmail, Yahoo, Outlook, etc. Il offre une interface conviviale et fournit un ensemble flexible d'API qui peuvent être utilisées pour configurer les messages électroniques. [12]
	**Sequelize :** ORM TypeScript et Node.js moderne pour Oracle, Postgres, MySQL, MariaDB, SQLite et SQL Server, et plus encore. Avec une prise en charge solide des transactions, des relations, un chargement rapide et paresseux, une réplication en lecture et plus encore. [13]

	**Google Firebase** : Plate-forme de développement d'applications qui permet aux développeurs de créer des applications iOS, Android et Web. [14]

Tableau 10: Technologies utilisées

III.3.2. Outils

Logo	Description
	**Visual studio code :** Editeur de code open-source développé par Microsoft supportant un très grand nombre de langages grâce à des extensions. Il supporte l'auto-complétion, la coloration syntaxique, le débogage, et les commandes git. [15]
	**Enterprise architecte :** Appelé EA, est un atelier de modélisation UML flexible, complet et puissant, développé par la société Sparx. Fournissant un avantage concurrentiel pour le développement, la gestion de projet et l'analyse métier, Enterprise Architect est un outil CASE couvrant le cycle de vie complet d'un projet informatique. [16]

	**Android studio :** L'environnement de développement intégré de la plate-forme Android de Google. Les versions d'Android Studio sont compatibles avec certains systèmes d'exploitation Apple, Windows et Linux. Avec la prise en charge de Google Cloud Platform et de l'intégration d'applications Google, Android Studio offre aux développeurs une boîte à outils bien fournie pour créer des applications Android ou d'autres projets, et fait partie intégrante du développement Android depuis 2013. [17]
	**Draw.io :** Application gratuite en ligne, accessible via son navigateur (protocole https) qui permet de dessiner des diagrammes ou des organigrammes. Cet outil vous propose de concevoir toutes sortes de diagrammes, de dessins vectoriels, de les enregistrer au format XML puis de les exporter. Draw.io est un véritable couteau suisse de la frise chronologique, de la carte mentale et des diagrammes de tout genre. [18]

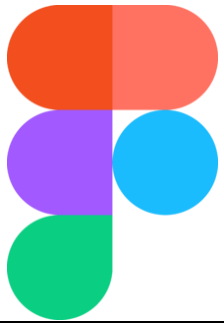
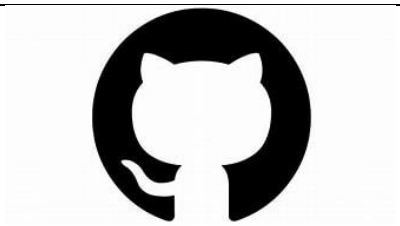
	**Figma :** Plateforme collaborative pour éditer des graphiques vectoriels et faire du prototypage. Elle permet de concevoir des design systems pour faciliter la création de sites web et d'applications mobiles. C'est une solution à destination des UI et UX designers et des développeurs. [19]
	**Postman :** Plate-forme API pour la création et l'utilisation d'API. Postman simplifie chaque étape du cycle de vie des API et rationalise la collaboration afin que vous puissiez créer de meilleures API plus rapidement. [20]
	**GitHub :** Service d'hébergement Open-Source, permettant aux programmeurs et aux développeurs de partager le code informatique de leurs projets afin de travailler dessus de façon collaborative. On peut le considérer **comme un Cloud** dédié au code informatique. [21]

Tableau 11: Outils utilisés

III.3.3. API

Google Calendar : L'API Google Agenda est une API RESTful accessible via des appels HTTP explicites ou via les bibliothèques clientes Google. L'API présente la plupart des fonctionnalités disponibles dans l'interface Web de Google Agenda. [22]

Google Sing-In : gère le flux OAuth 2.0 et le cycle de vie des jetons, ce qui simplifie votre intégration aux API Google. Un utilisateur a toujours la possibilité de révoquer l'accès à une application à tout moment. [23]

III.4. Présentation graphique du système « PlanIt »

Dans ce projet nous avons créé une application mobile, un site web et une application web. L'application mobile fonctionne en mode en-ligne destinée aux citoyens pour la réservation des rendez-vous. Le site web présente les fonctionnalités principales du système avec une interface de connexion et inscription pour les administrations souhaitant avoir un compte professionnel et une autre application web pour les employés des administrations.

Dans cette partie nous vous présentons les principales interfaces de notre système.

III.4.1. Interfaces de l'application mobile

III.4.1.1. Interface explorer

La figure 19 correspond à la page "Explorer" qui répertorie tous les instituts proposant le service de prise de rendez-vous. Si un citoyen n'a pas de compte, il doit demander son inscription en choisissant un institut afin de valider son identité.

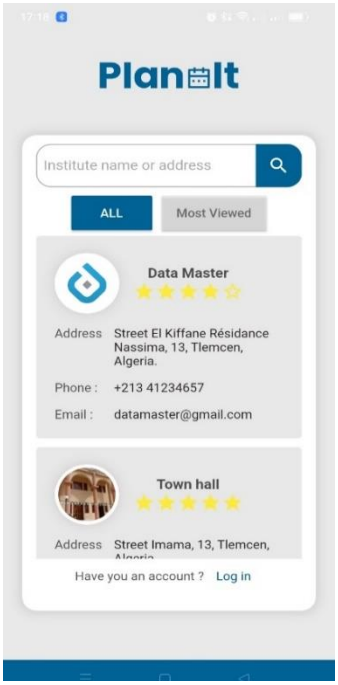


Figure 19:Page explorer

La figure 20 correspond à la page détails d'institut qu'il a choisi contenant les coordonnées de l'institut avec son adresse et bouton Request pour demander l'inscription.

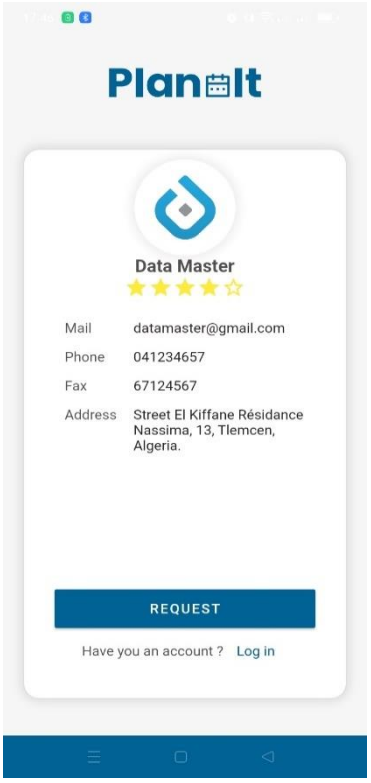


Figure 20: Détails institution dans la partie explorer

III.4.1.2. Interface demander l'inscription

La figure 21 représente un formulaire pour renseigner les informations personnelles d'utilisateur pour demander l'inscription.

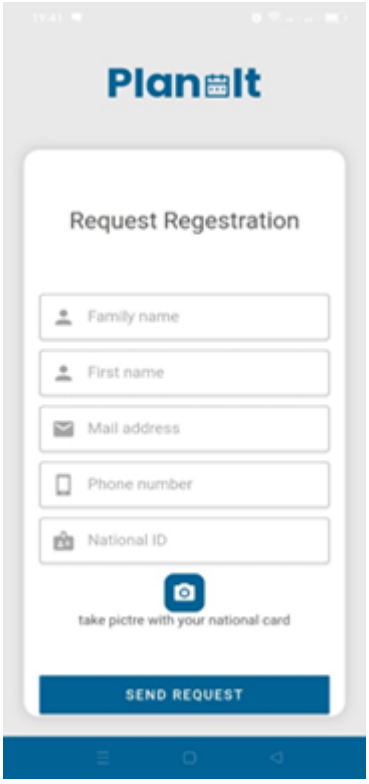


Figure 21:formulaire demander l'inscription

III.4.1.3. Interface d'authentification

Un citoyen qui possède un compte il pourra directement utiliser l'application en saisissant leur email et mot de passe.

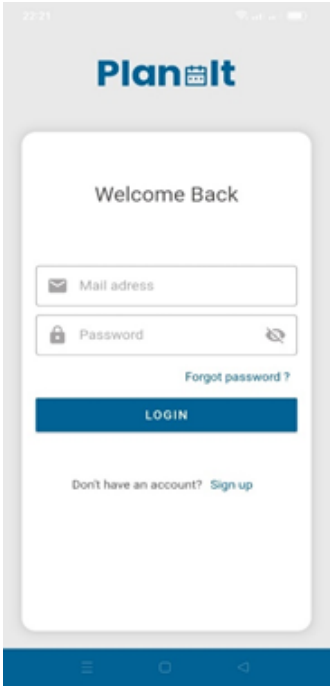


Figure 22: formulaire d'authentification

III.4.1.4. Interface liste des rendez-vous

La figure 23 représente la liste des rendez-vous réservés avec le numéro de client dans le file d'attente et les informations sur le rendez-vous comme son adresse, sa date et sa durée.

Ainsi les détails d'un rendez-vous contenant la durée, la description...

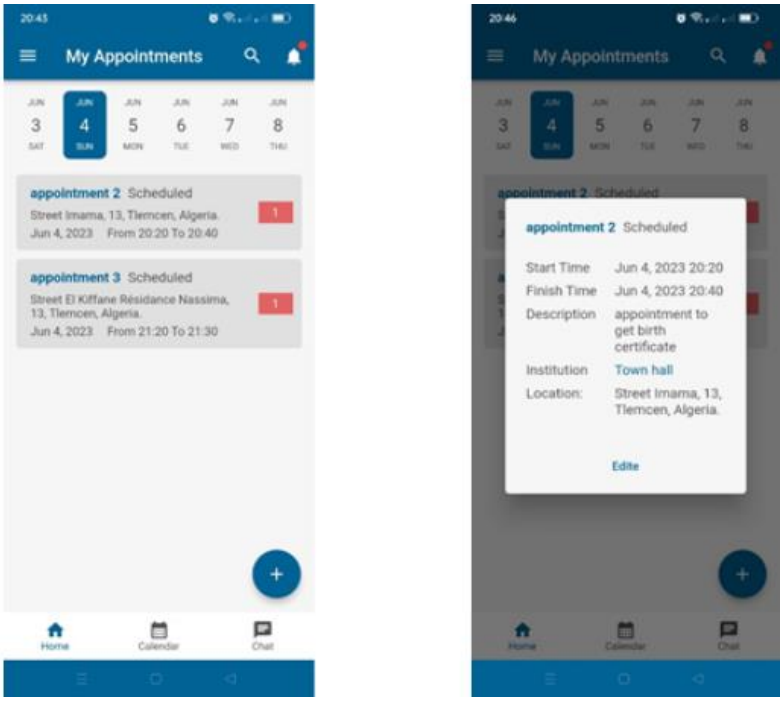


Figure 23:la liste des rendez-vous par jour et détails d'un rendez-vous

III.4.1.5. Interface liste des institutions

Interface représente la liste des institutions qui offre le service des rendez-vous, avec les détails d'une institution avec son adresse et la possibilité d'ajouter une évaluation pour l'institution

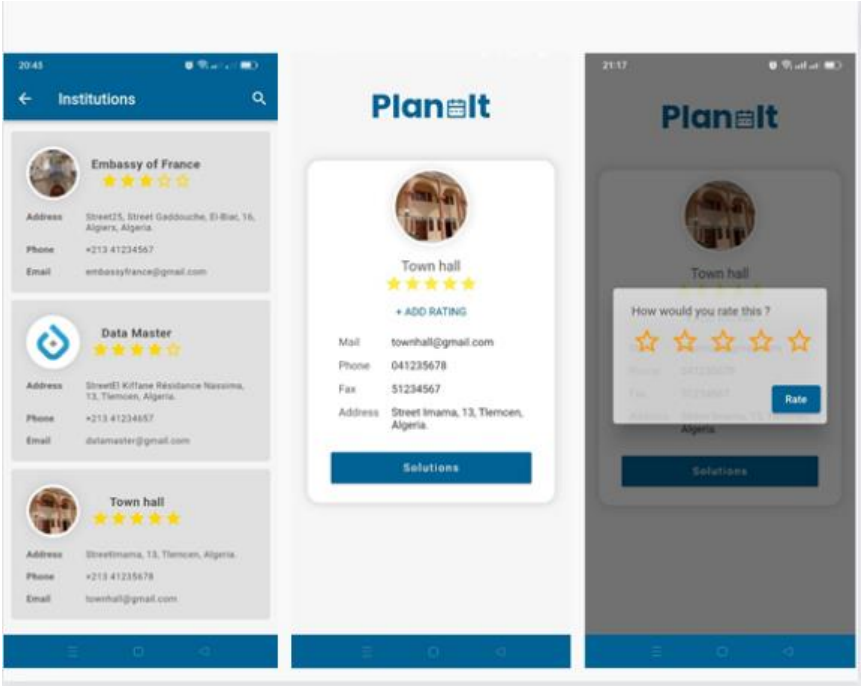


Figure 24: Liste des institutions et leurs détails

III.4.1.6. Interface liste des solutions

Cette figure représente les solutions qui intéressent les citoyens pour résoudre ces problèmes, ils sont ajoutés par les institutions qui offrent le service de rendez-vous, avec la possibilité d'ajouter une solution à la liste des favoris.

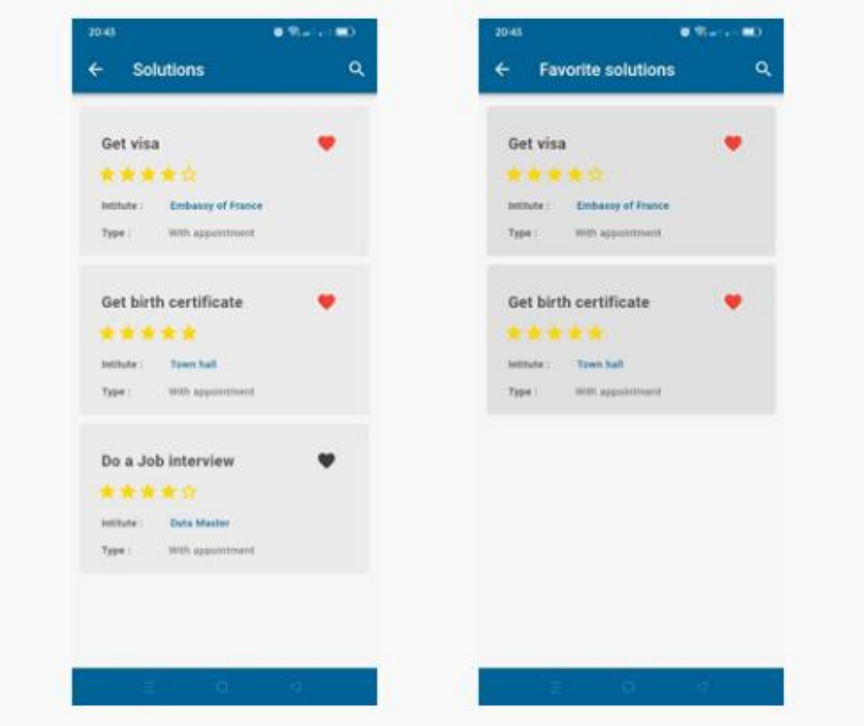


Figure 25: la liste des solutions et les favoris

III.4.1.7. Interface de calendrier

Cette figure représente le calendrier de l'application avec les dates et les heures et l'option de changer le mode de vue de calendrier

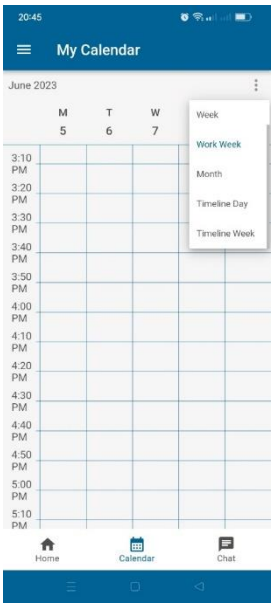


Figure 26: interface calendrier

III.4.1.8. Interface de profile utilisateur client final

Cette figure affiche les données personnelles du client final avec l'icône de modification

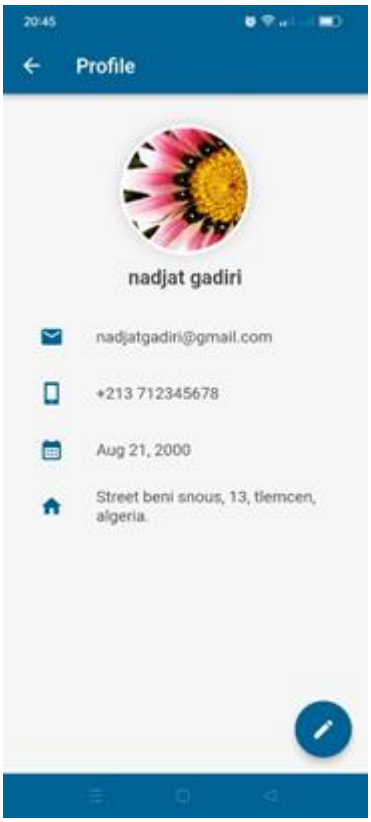


Figure 27: Interface profile client final

III.4.1.9. Interface liste des paramètres de l'application

Cette interface permet à l'utilisateur d'accéder à son profile, de changer son mot de passe, de changer sa localisation, de changer le thème ou switcher vers le thème sombre, de changer le langage de l'application ou de déconnecter.

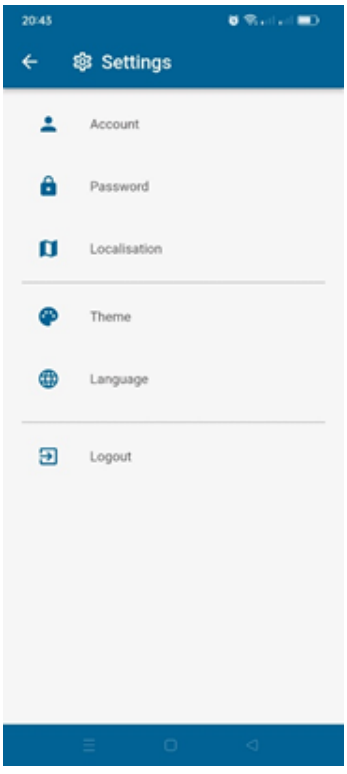


Figure 28 : Interface liste des paramètres

III.4.2. Interfaces du site web

Nous avons créé un site web vitrine dédié à notre système PlanIt de gestion des rendez-vous professionnels.

III.4.2.1. Interface de la page d'accueil

La figure 29 représente la page d'accueil de site web ou il situ le menu de site et deux boutons pour se connecter et s'inscrire.

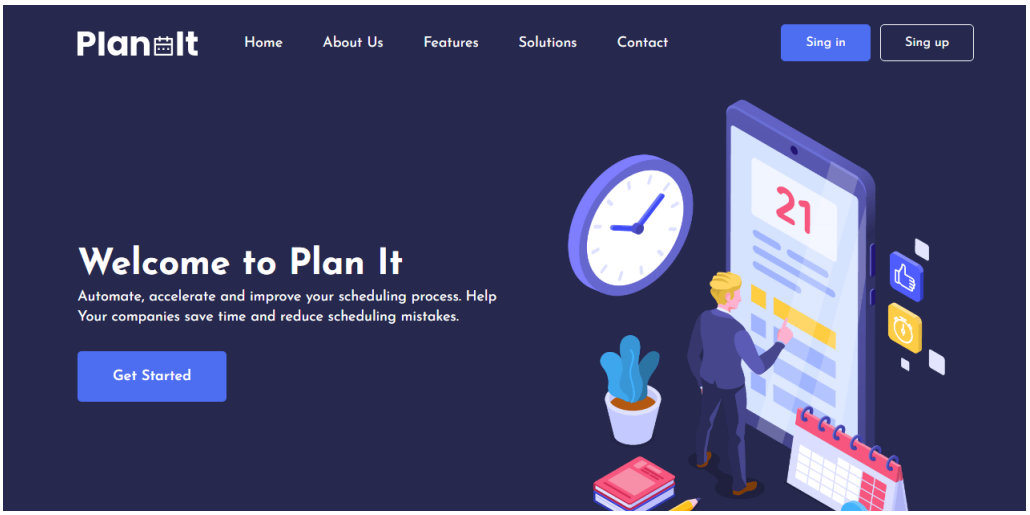


Figure 29: Interface de la page d'accueil

III.4.2.2. Interface de la page à propos

La figure 30 représente la page à propos qui permet de mieux comprendre la mission de PlanIt.

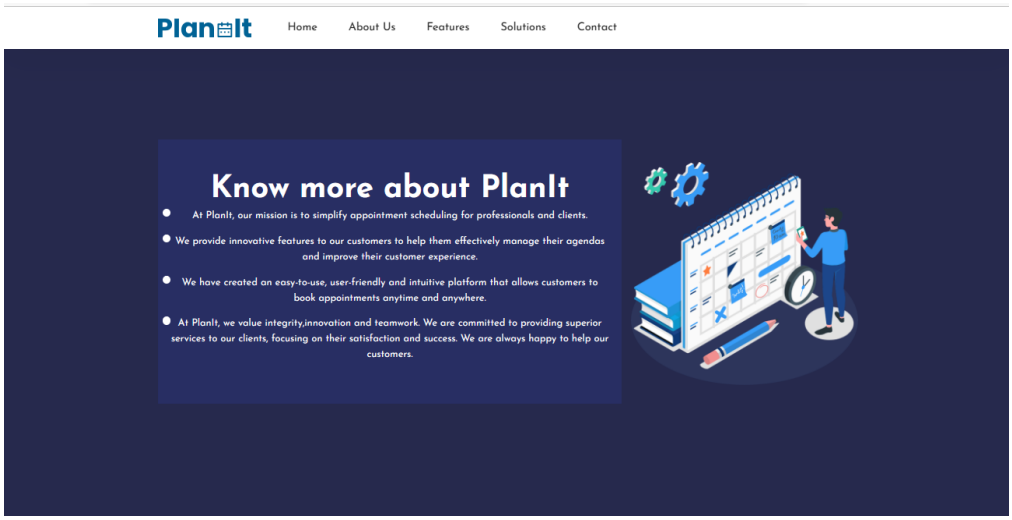


Figure 30 : Page à propos

III.4.2.3. Interface de la page fonctionnalités

La figure 31 représente les fonctionnalités majeures de système PlanIt.

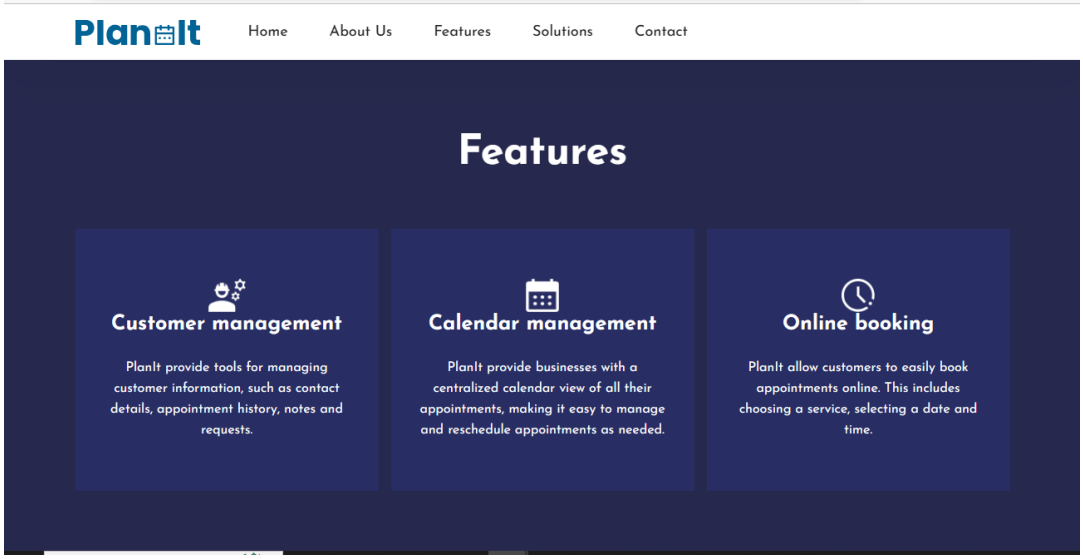


Figure 31: Page de fonctionnalités

III.4.2.4. Interface de la page solutions

La figure 32 est destiné pour les solutions offerte par PlanIt.

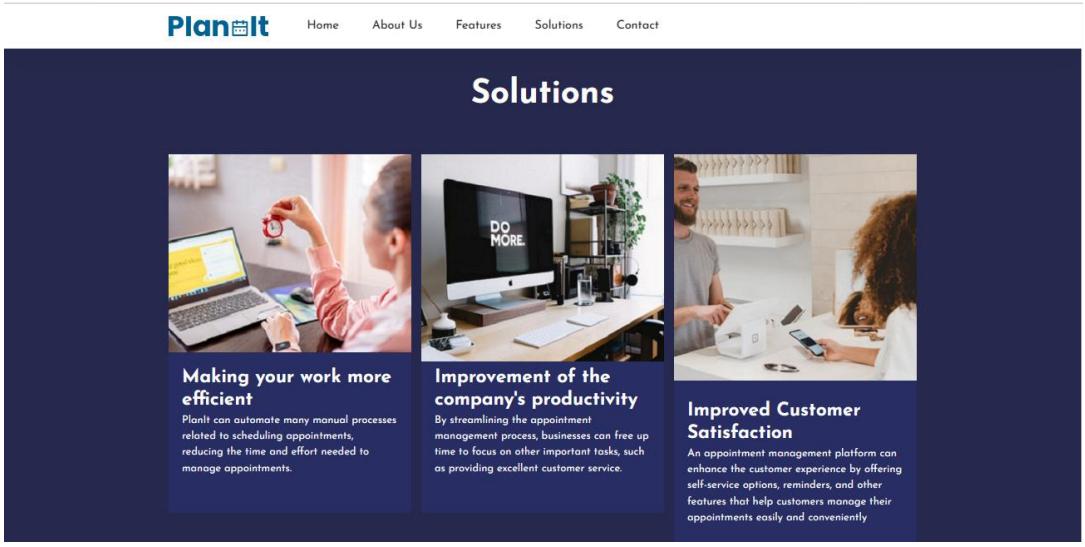


Figure 32: Page des solution offertes par PlanIt

III.4.2.5. Interface de la page se connecter

La figure est affichée lorsque le super Admin, admin de l'institution, chef de service ou un employé normal souhaitent se connecter pour accéder à leurs tableaux de bord,

Pour se connecter l'utilisateur doit saisir son email et son mot de passe suivie par un clic sur le bouton « LOGIN ».

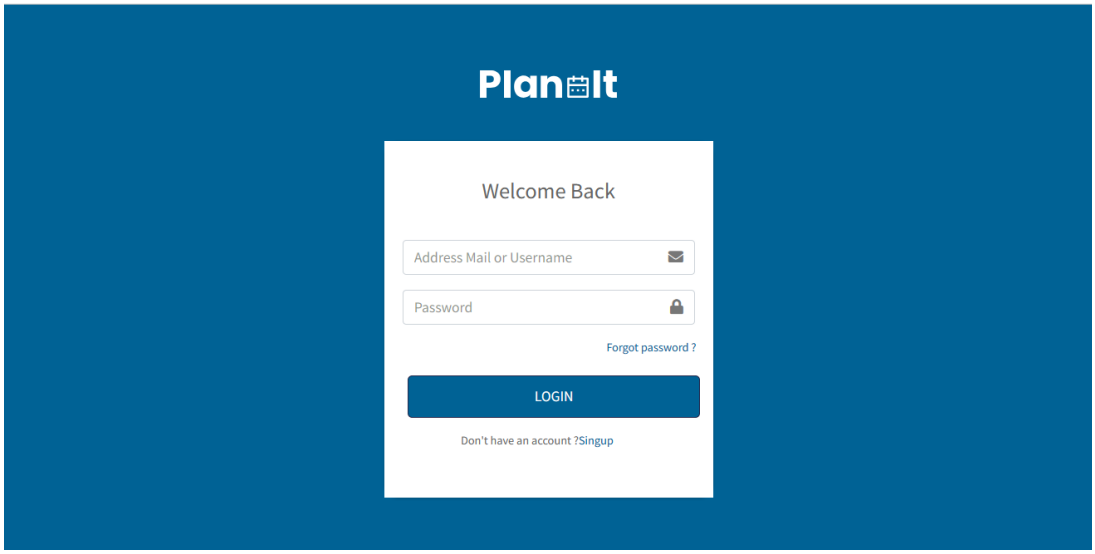


Figure 33: Interface se connecter

III.4.2.6. Interface de la page s'inscrire

La figure est affichée lorsqu'un admin souhaite créer un compte son entreprise.

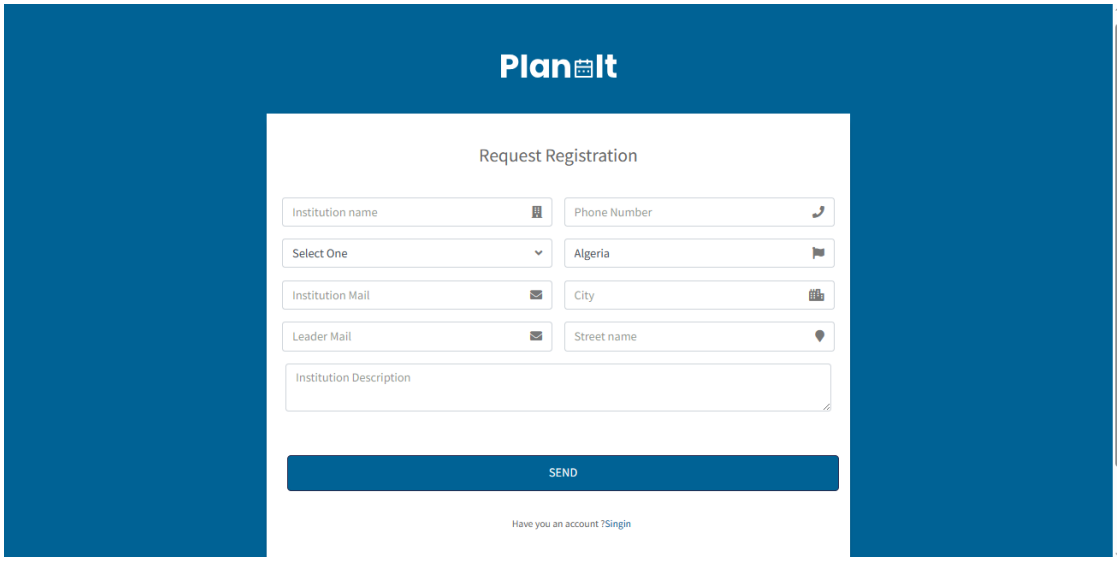


Figure 34: Interface s'inscrire

III.4.3. Interfaces de l’application mobile

III.4.3.1. Interfaces du profile super admin

III.4.3.1.1. Menu vertical

Cette partie change selon le profil d’utilisateur et ses droits d’accès.

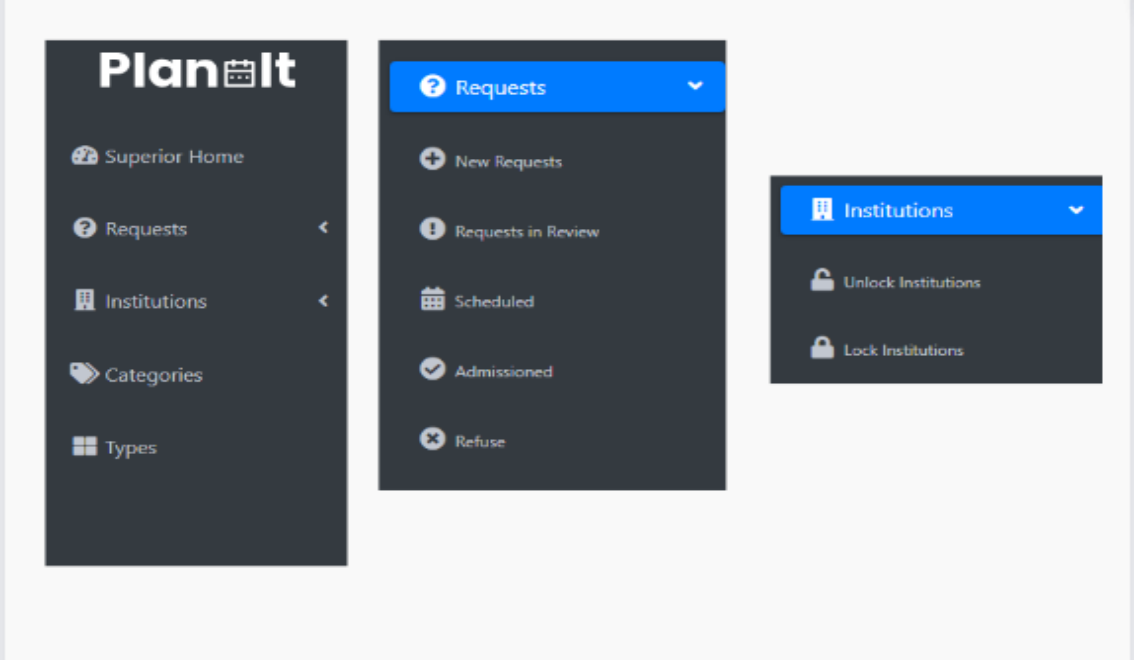


Figure 35: Interface du menu vertical pour le profile super admin

III.4.3.1.2. Dashboard

Lorsque le super admin est authentifié la page Dashboard s'affiche contenant le nombre d'institutions qui offrent le service de rendez-vous et des statistiques.

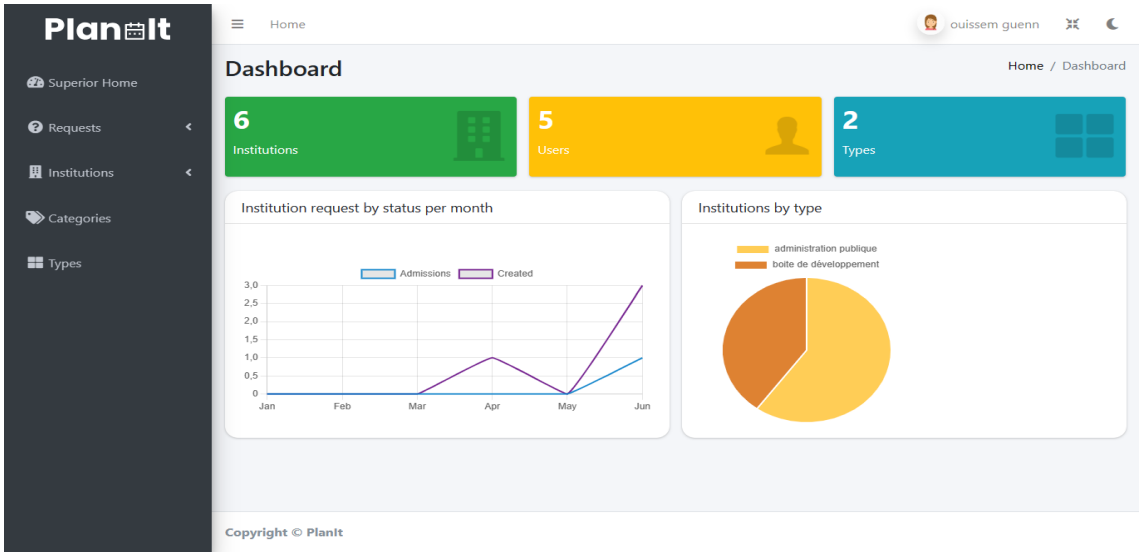


Figure 36:interface du Dashboard du super admin

III.4.3.1.3. Liste des demandes d'inscriptions pour les institutions

Elle se divise en 5 catégories

- **Les nouvelles demandes** : sont des demandes ne sont pas encore visualisés par le super admin de PlanIt

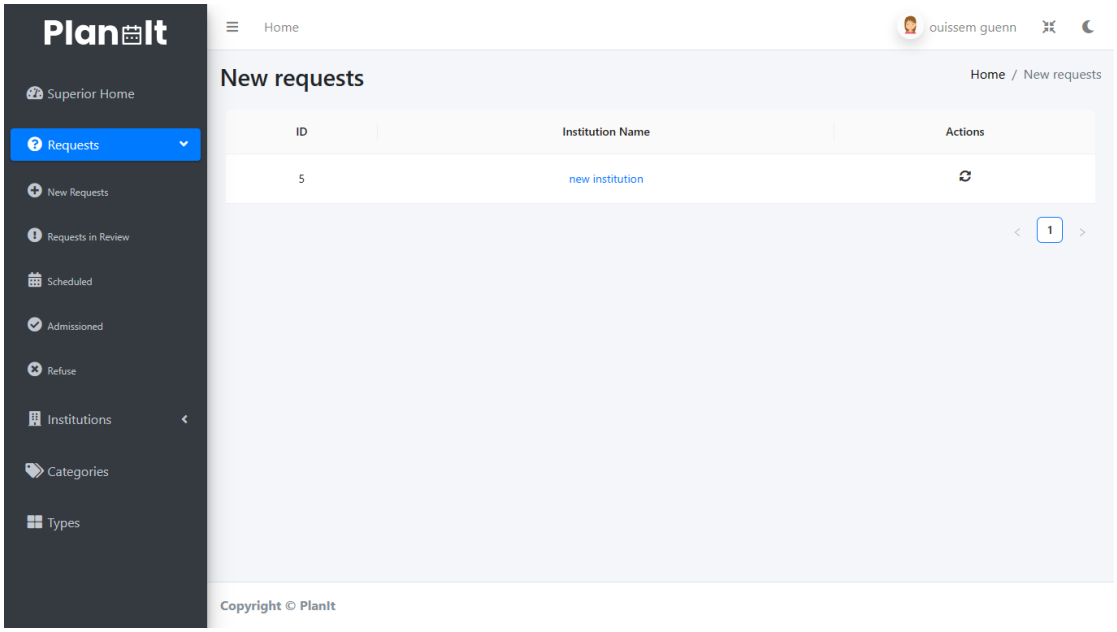


Figure 37:liste des nouvelles demandes d'inscription pour les institutions

L'icône permet de changer l'état de demande

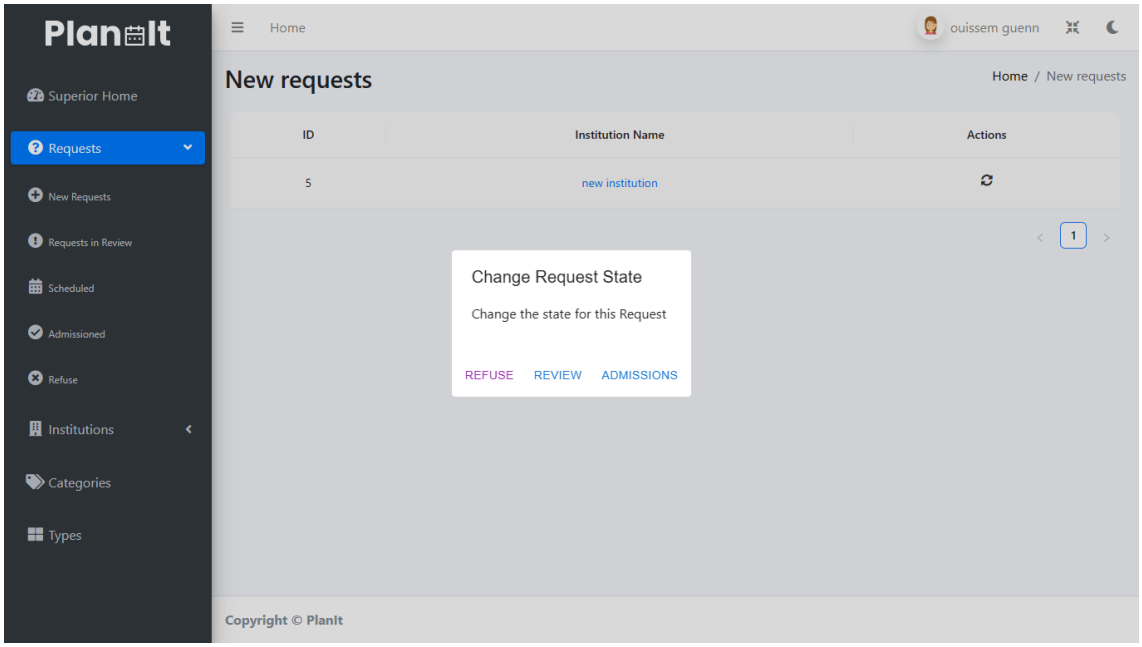


Figure 38:interface de changement d'état de demande d'inscription

- **Les demandes en révision :** sont encore en phase d'étude de validité, en examinant la demande pour vérifier si elle remplit les critères nécessaires pour être prise en considération.

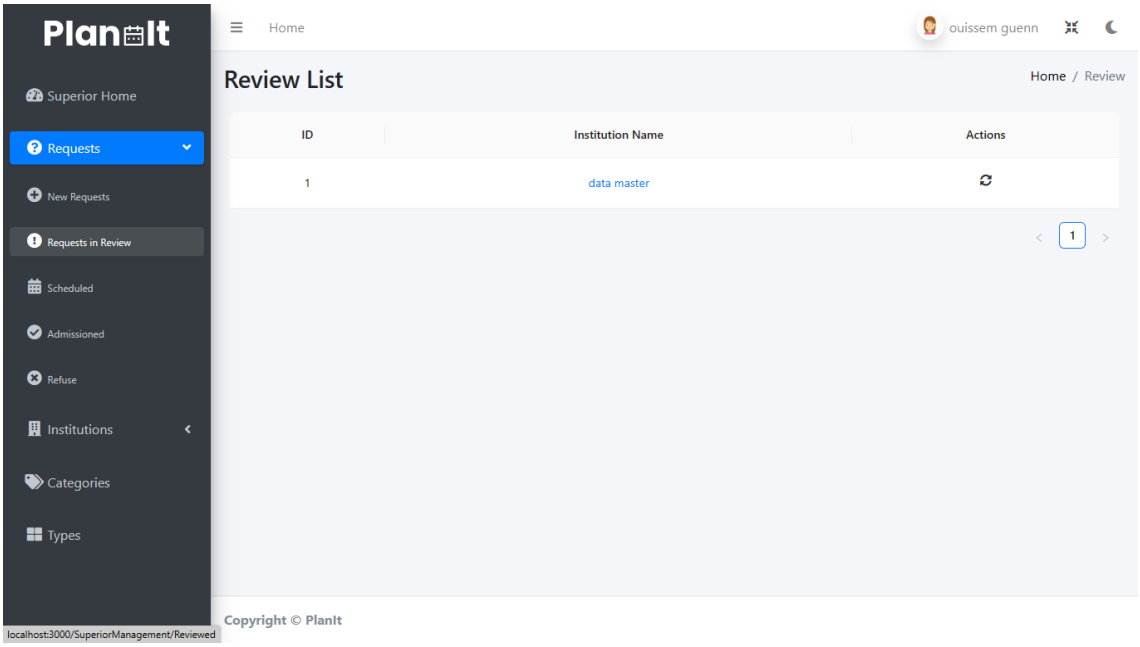


Figure 39: Interface des demandes en révision

- **Les demandes programmées avec rendez-vous :** sont des demandes ou le super admin programme des rendez-vous avec l'admin de l'institution qui envoie une demande d'inscription sur le site de PlanIt, le rendez-vous est programmé pour valider l'institution.

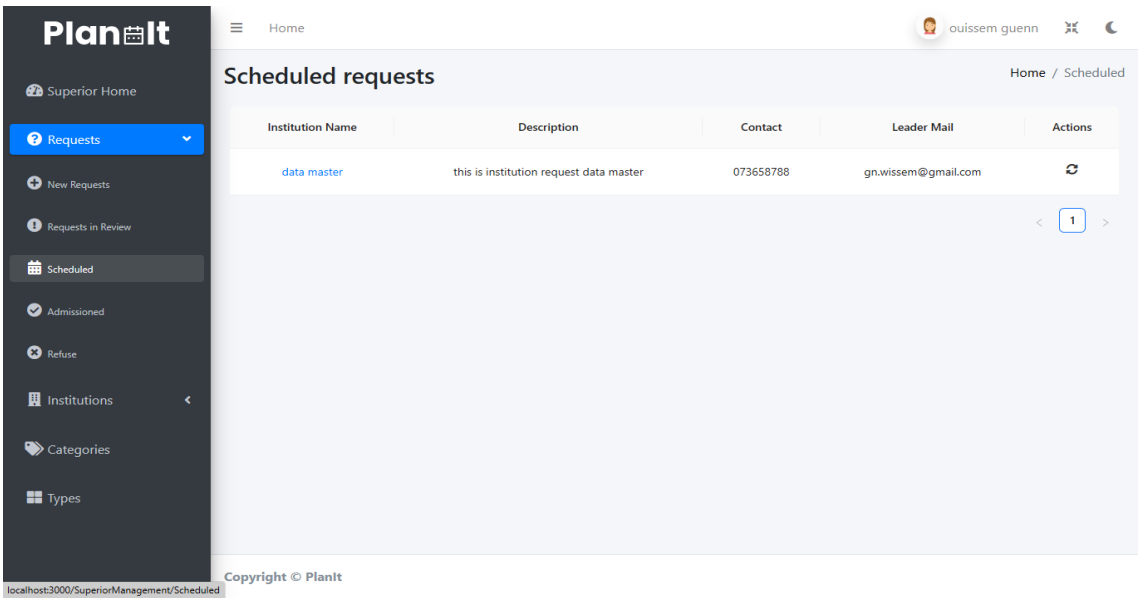


Figure 40: Interface des demandes programmés avec rendez-vous

- **Les demandes admis :** sont les demandes qui sont déjà validés par le super admin de PlanIt.

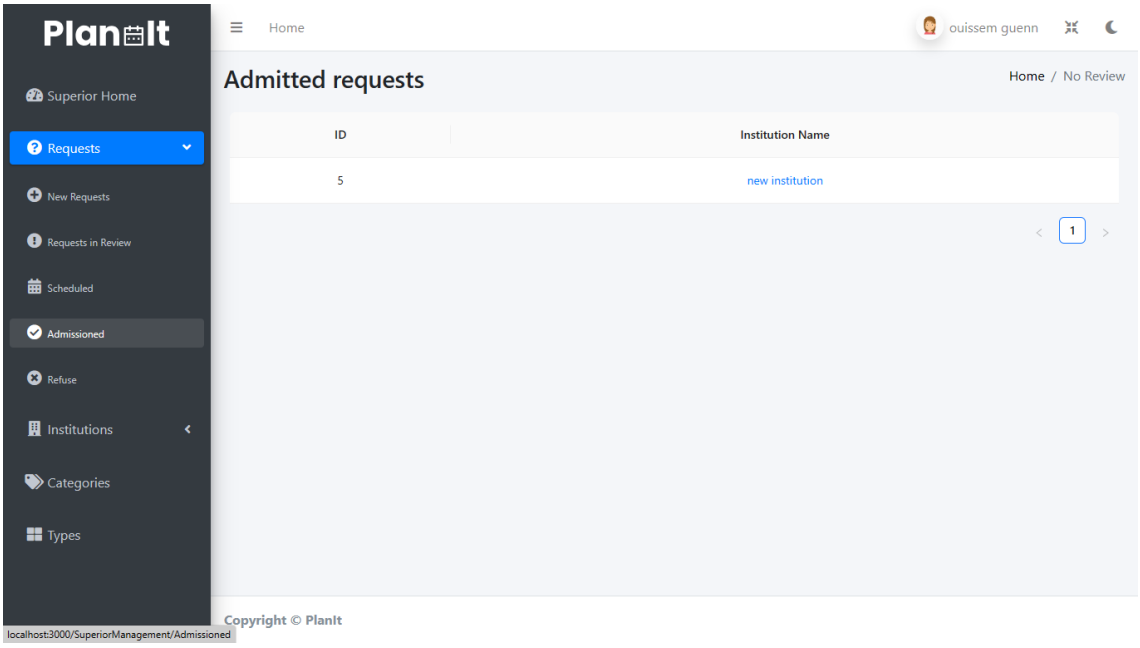


Figure 41: Interface des demandes admis

- **Les demandes refusées :** les demandes qui ne sont pas validés par le super admin à cause de manque d'informations sur l'institution ou l'institution n'est pas légal.

III.4.3.1.4. Liste des institutions

Se compose en deux catégories :

- **Les institutions verrouillées :** sont les institutions qui n'offrent plus les services des rendez-vous ou le super admin a bloqué leurs comptes.

- **Les institutions déverrouillées :** les institutions qui sont légaux et qui offrent le service de rendez-vous qui sont affichés dans la liste des institutions dans l'application mobile de PlanIt.

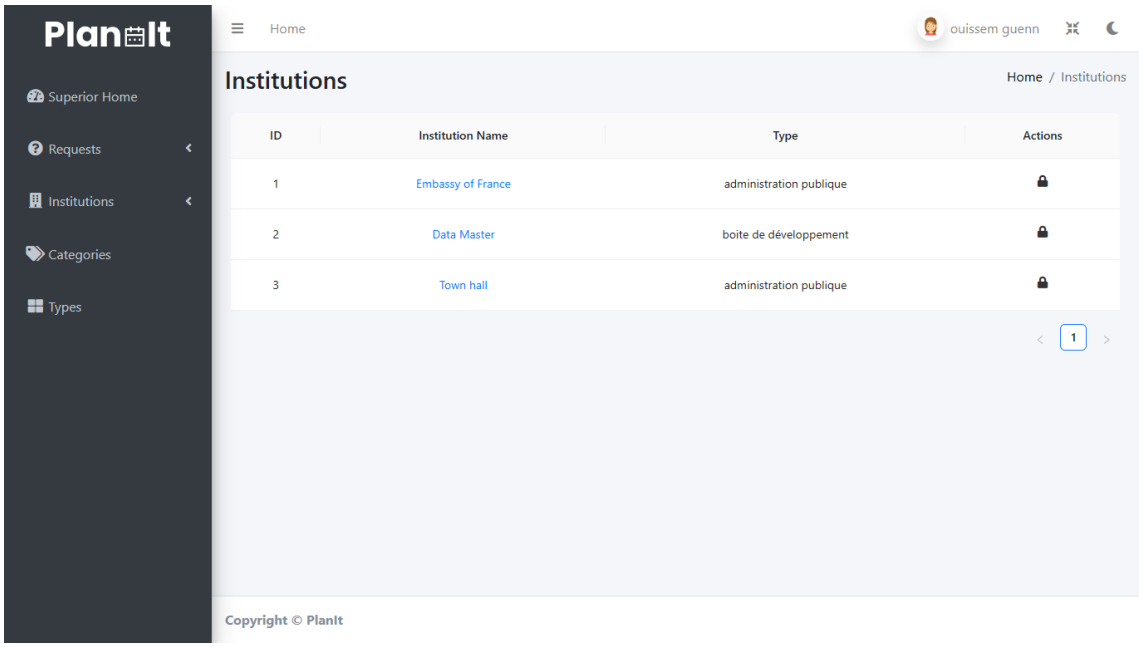


Figure 42: Liste des institutions déverrouillés

III.4.3.1.5. Liste des catégories

C'est la catégorie de service par exemple un service avec rendez-vous

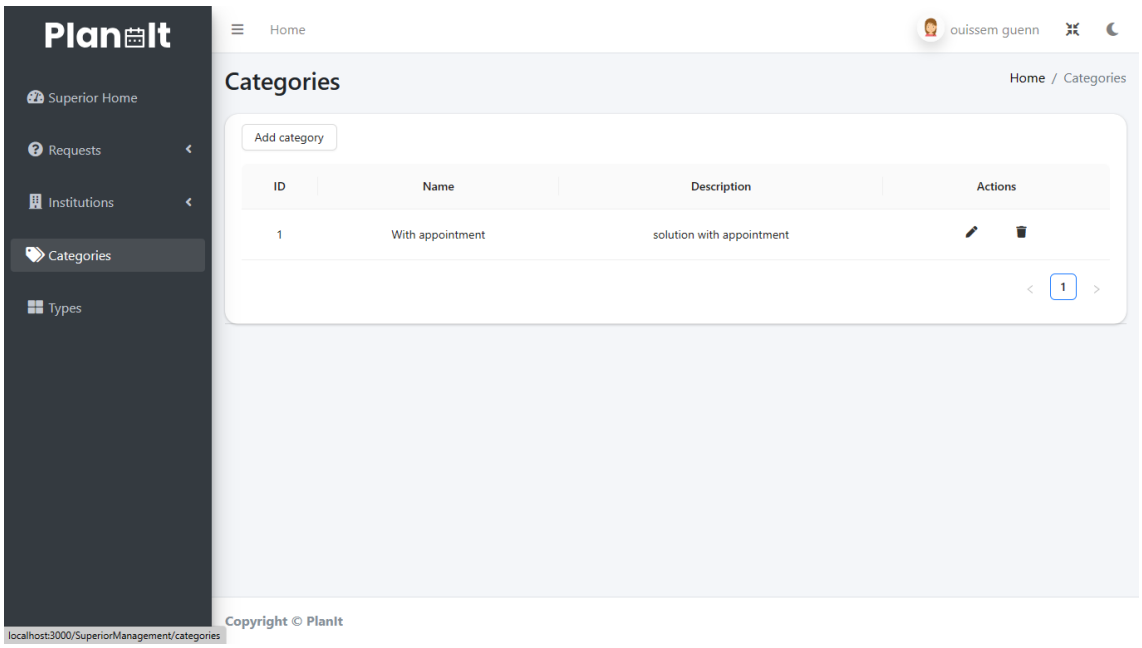


Figure 43:liste des catégories

III.4.3.1.6. Liste des types

Sont les types d'institutions offrant le service de rendez-vous.

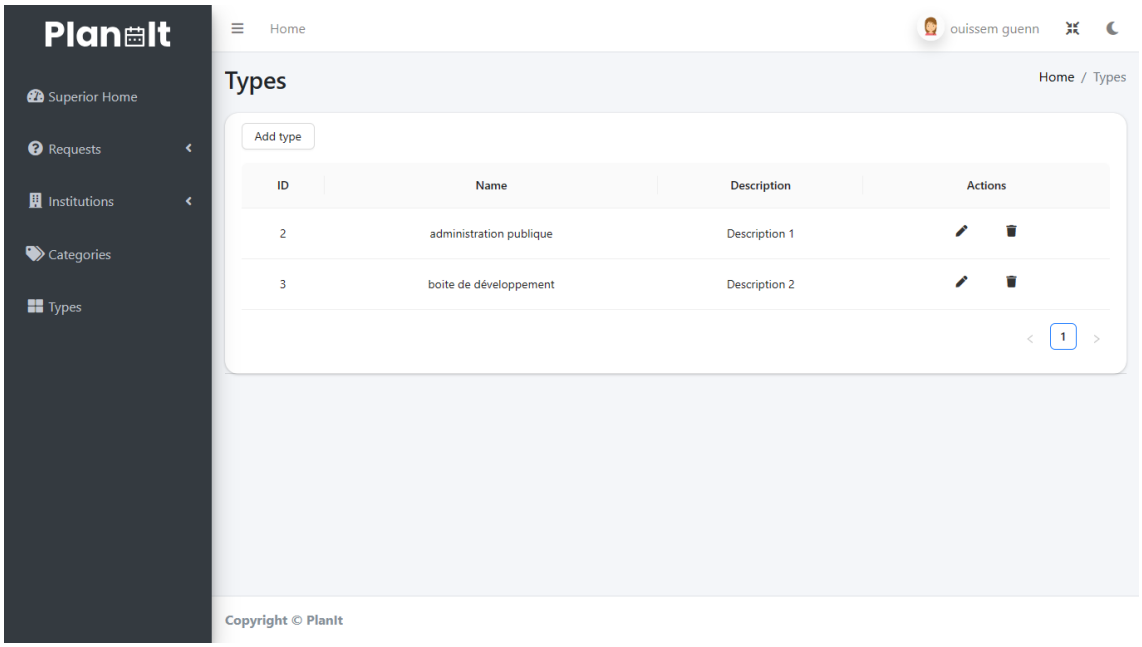


Figure 44:liste des types des institutions

III.4.3.2. Interfaces du profile admin de l'institution

III.4.3.2.1. Menu vertical

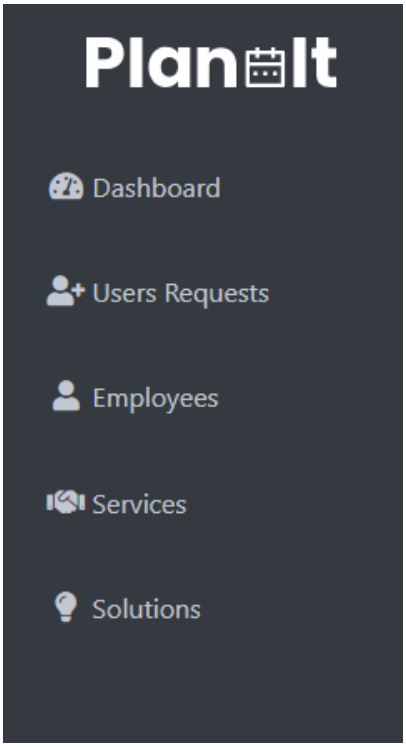


Figure 45: Menu vertical pour le profile admin

III.4.3.2.2. Liste des employées

La figure suivante affiche la liste des employées avec la possibilité de supprimer un employé, de visualiser son profil où d'ajouter un nouvel employé.

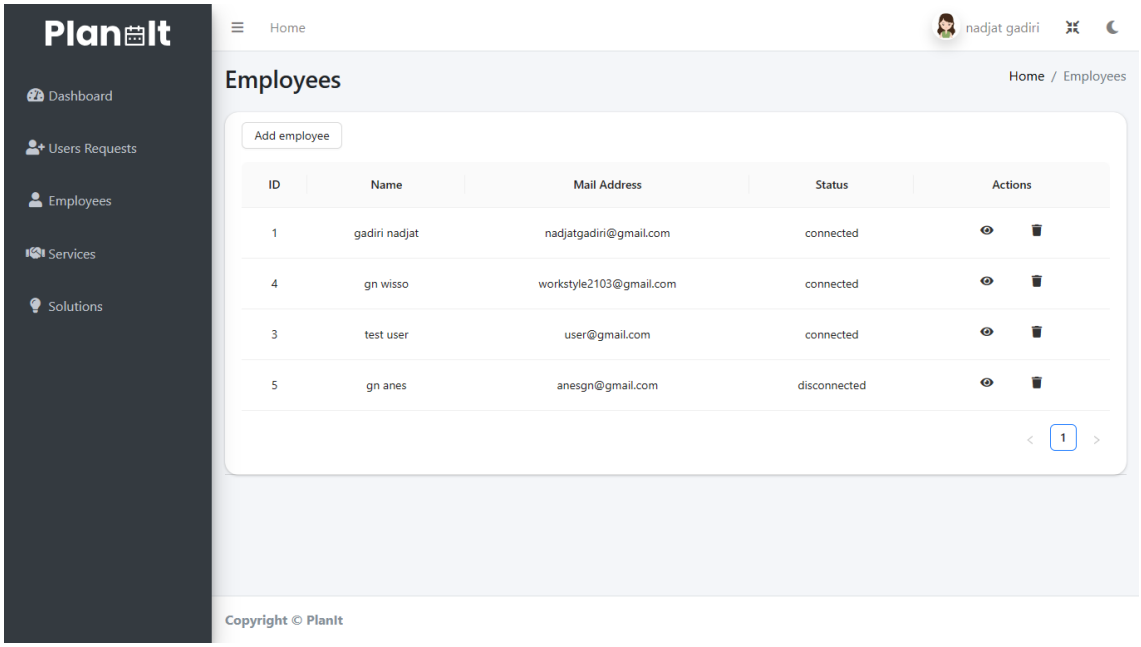


Figure 46: Liste des employés

III.4.3.2.3. Liste des services

La figure suivante représente la liste des services ou l'admin de l'institution pourra ajouter un nouveau service en précisant leur chef, modifier un service ou le supprimer, visualiser les détails d'un service et ses offices.

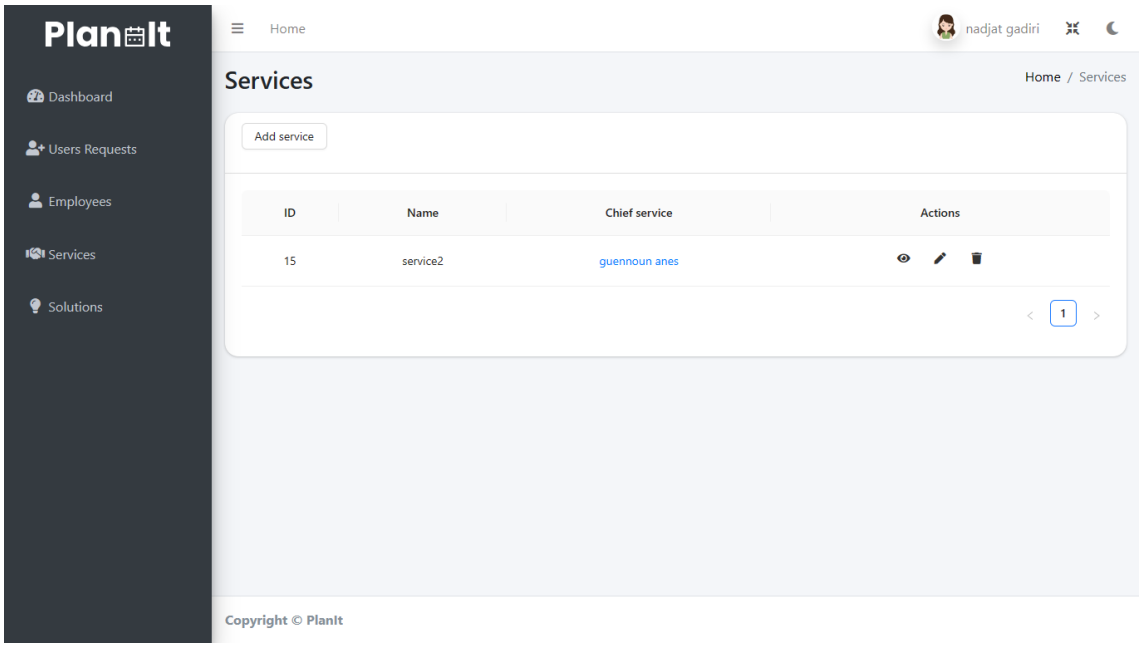


Figure 47: Liste des services

III.4.3.3. Interfaces du profile employée

III.4.3.3.1. Menu vertical

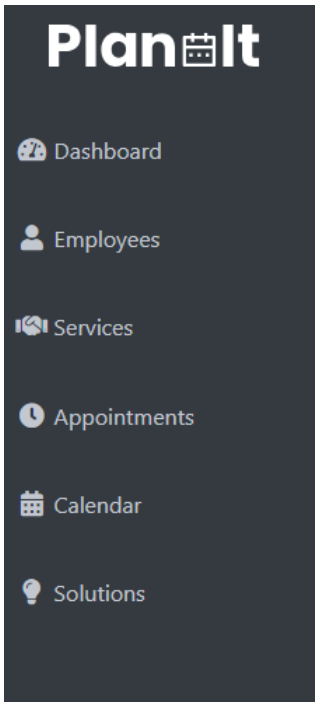


Figure 48: Menu vertical du profil employé

III.4.3.3.1. Calendrier

La figure suivante représente le calendrier pour le profil employé ou il peut visualiser les rendez-vous de son bureau, de changer le mode de vue de calendrier et de changer l'état de rendez-vous.

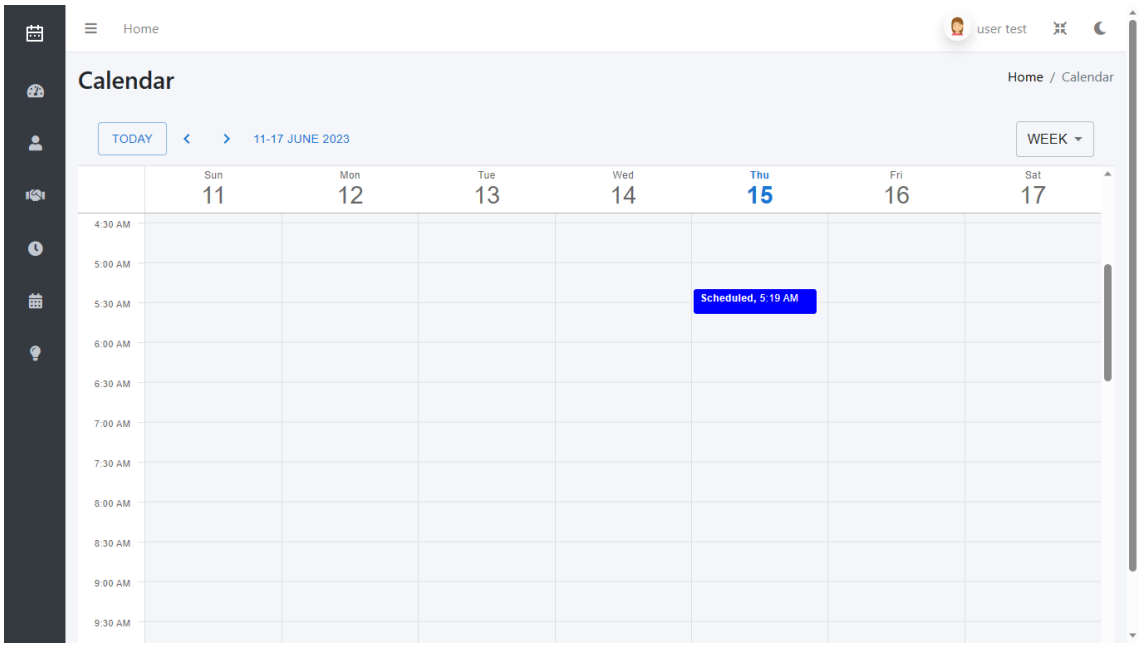


Figure 49: Interface du calendrier pour l'employé

III.5. Tests et validation

Dans cette partie nous abordons la phase des tests, cette phase joue un rôle fondamental dans le génie logiciel en garantissant la qualité, la fiabilité et la performance des systèmes logiciels. Ils constituent une étape critique permettant de vérifier le bon fonctionnement de l'application dans son environnement réel et de détecter les erreurs, les anomalies et les dysfonctionnements avant sa mise en production.

III.5.1. Tests unitaires

Dans la partie des tests unitaires, on se concentre sur la vérification du bon fonctionnement des unités individuelles de code, des fonctions ou des méthodes, de manière isolée.

Le test unitaire est un type de test de logiciel dans lequel des unités ou des composants individuels sont testés. L'objectif principal des tests unitaires est de valider chaque unité du code logiciel et de déterminer si elle fonctionne conformément aux attentes. [24]

Pour exécuter les cas de tests nous avons utilisé le Framework Mocha, qui fournit des assertions pour vérifier si les résultats réels correspondent aux résultats attendus.

III.5.1.1. Test unitaire pour les institutions visibles

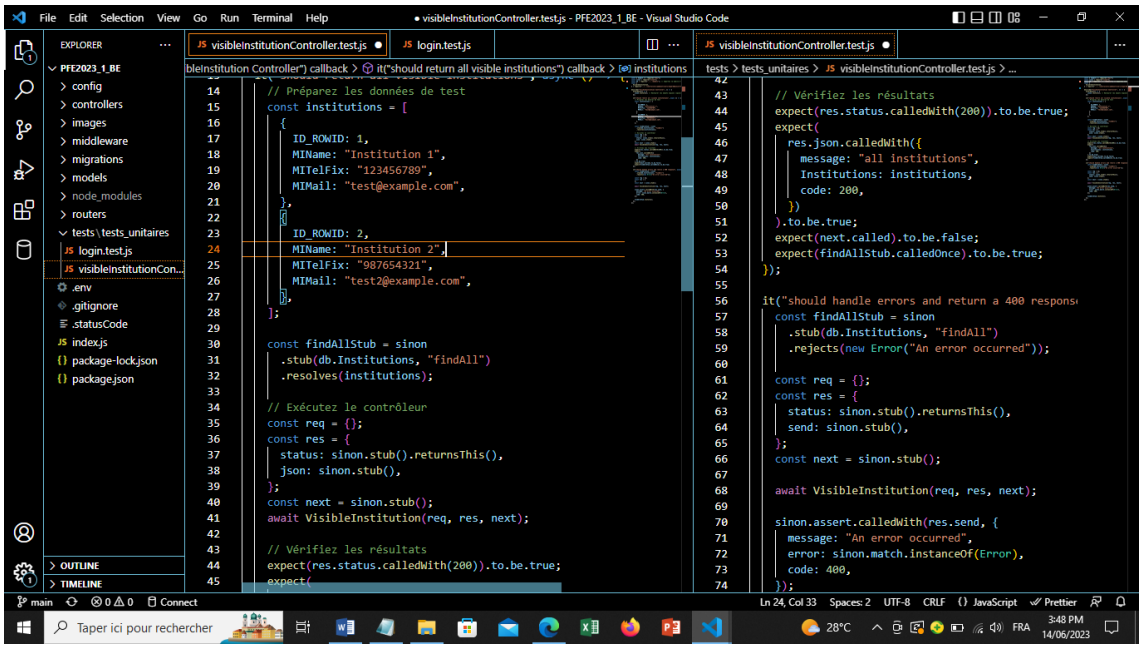


Figure 50:Tests unitaires pour les institutions visibles

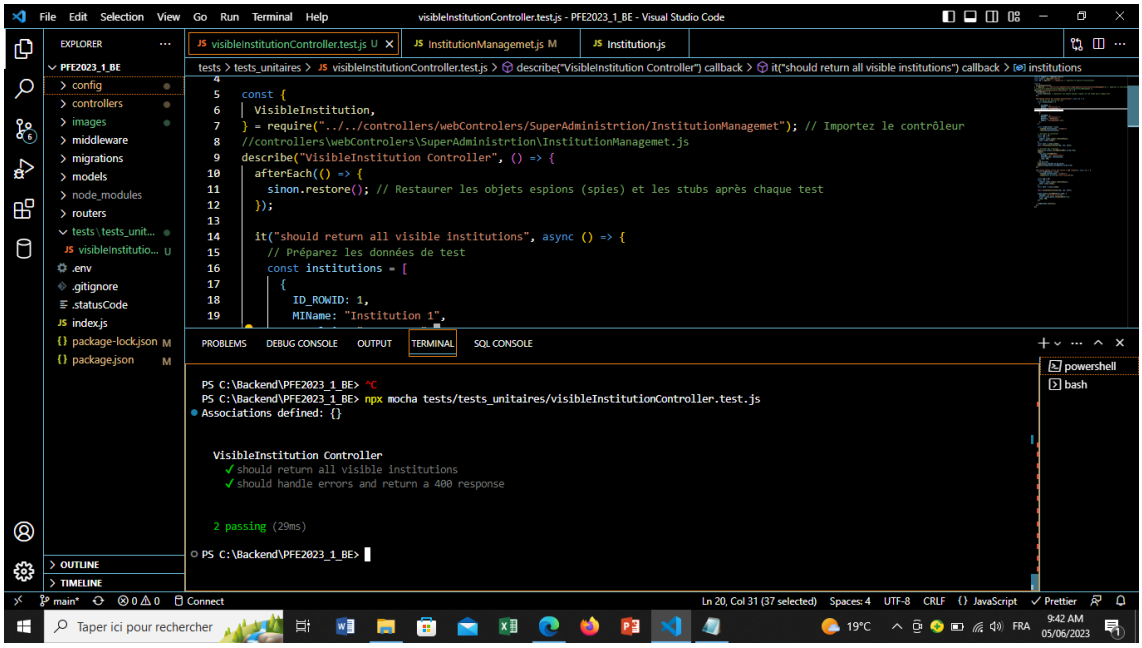


Figure 51:Résultats de tests unitaires pour les institutions visibles

III.5.1.2. Test unitaire pour la connexion au comptes

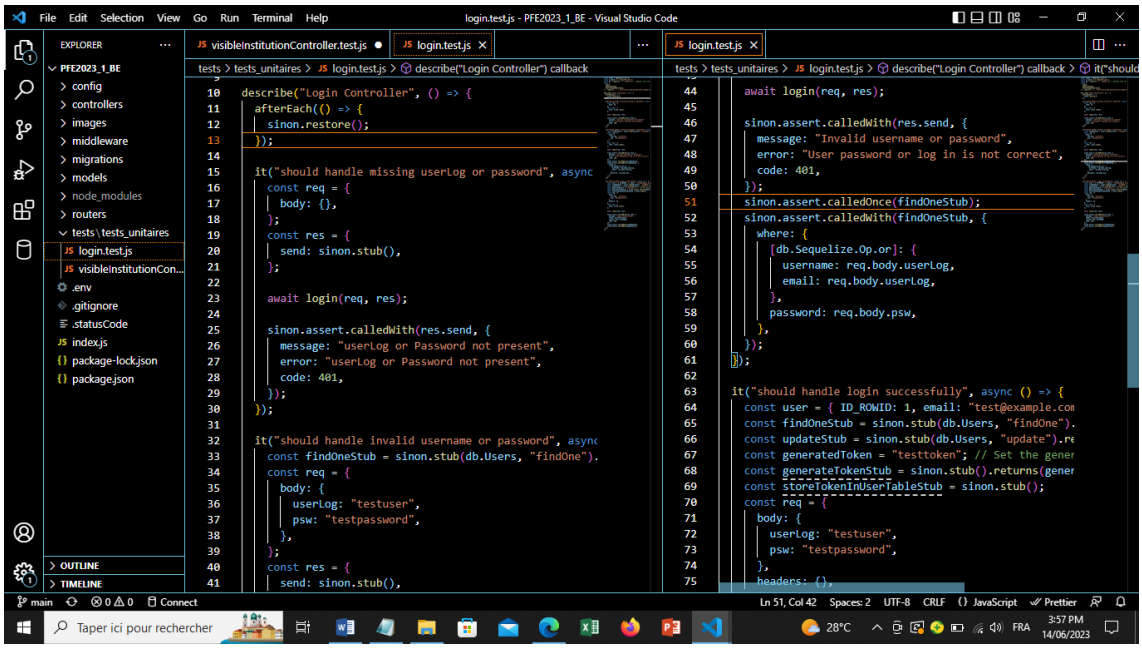


Figure 52:Tests unitaire pour la connexion aux comptes

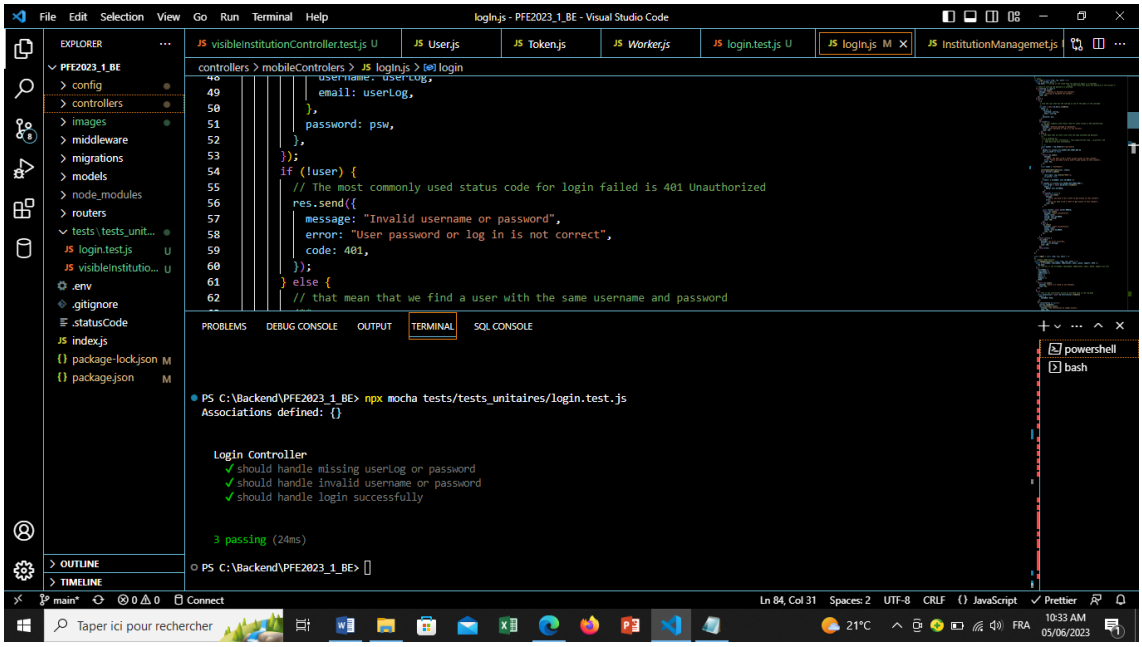


Figure 53:Résultats de tests unitaires pour la connexion aux comptes

III.6. Conclusion

Dans ce dernier chapitre nous avons présenté certaines interfaces principales du système PlanIt tout en présentant l'ensemble des outils, technologies et API utilisés.

Conclusion générale

1. Conclusion

Notre projet de fin d'études a fait objet d'un stage au sein de l'entreprise "Eurl Data Master”. Le projet avait pour objectif la conception et le développement d'un système de gestion des rendez-vous au sein des administrations Algérienne.

Le but principal de notre système de gestion des rendez-vous PlanIt est de faciliter et d'optimiser le processus de planification et de réservation des rendez-vous entre les clients et les fournisseurs de services dans les administrations. Il vise à simplifier la prise de rendez-vous, à réduire les temps d'attente, à éviter les conflits de planning. Il fournit une interface conviviale qui présente les différentes fonctionnalités de système telles que la réservation des rendez-vous avec les administrations, la gestion des disponibilités et la gestion des rendez-vous existants.

Le système de gestion des rendez-vous vise à améliorer l'expérience des utilisateurs, à optimiser les ressources et améliorer l'efficacité opérationnelle de l'administration.

Le développement du système a été fait au sein de l'entreprise "DATA MASTER" qui nous a accompagnée tout au long de la réalisation de notre projet durant le stage effectué.

Nous avons entamé notre travail par une analyse de besoins du client DATA MASTER, puis on a enchainé avec la partie conception et modélisation de notre système Plant en se basant sur le langage de modélisation UML où nous avons créé notre modèle de données.

Par la suite, la solution conçue a été concrétisée dans la phase de réalisation de notre système PlanIt en utilisant divers langages de programmation tels que Flutter, Node.js et React.js.

De plus, nous avons exploité différentes API et technologies comme GOOGLE CALENDAR pour améliorer les fonctionnalités de notre système.

Ce projet a été une occasion exceptionnelle qui nous a permis d'enrichir nos connaissances, il nous a permis aussi d'acquérir une expérience personnelle et professionnelle très riche qui va certainement nous aider à s’intégrer dans le milieu socioprofessionnel. De plus, il nous a offert l'opportunité de découvrir de nouvelles technologies et langages de programmation d’actualité et très riche dans de le monde de

développement, qui vont s’ajouter aux différentes compétences acquises durant notre cursus universitaire.

2. Perspectives

Ce projet ouvre de nombreuses perspectives en ce qui concerne le développement des systèmes de gestion des rendez-vous.

Dans un avenir proche, nous envisageons plusieurs améliorations et extensions possibles pour le système PlanIt afin de répondre aux besoins changeants des utilisateurs et des administrations. Parmi ces pistes de développement ouvertes, nous prévoyons :

- Intégration de fonctionnalités de rappel et de notifications : Nous souhaitons ajouter des fonctionnalités de rappel automatique pour les utilisateurs afin de leur rappeler les rendez-vous à venir.
- Intégration de la géolocalisation : Nous envisageons d'incorporer la géolocalisation dans le système pour permettre aux utilisateurs de trouver facilement les administrations les plus proches de leur emplacement. Literait la prise de rendez-vous en proposant des options à proximité.
- Exploitation des techniques d'intelligence artificielle pour améliorer le processus de validation des personnes et des entreprises.
- Intégration de paiements en ligne : Nous envisageons d'intégrer des fonctionnalités de paiement en ligne pour les services qui le nécessitent.

Cela permettrait aux utilisateurs de régler leurs frais directement via l'application.

Bibliographie

[1] «Digital Guide IONOS,» [En ligne]. Available: https://www.ionos.fr/digitalguide/sites-internet/developpement-web/json-web-token-jwt/. [Accès le 4 Juin 2023].

[2] C. Deluzarche, «informatique-uml-3979,» 22 Aout 2O21. [En ligne]. Available: https://www.futura-sciences.com/tech/definitions/informatique-uml-3979/. [Accès le 01 Avril 2023].

[3] «UP vs XP,» 7 Avril 2015. [En ligne]. Available: https://www.petite-entreprise.net/P-2503-83-G1-up-vs-xp.html#:~:text=La%20m%C3%A9thode%20du%20Processus%20Unifi%C3%A9%20%28UP%20pour%20Unified,par%20les%20cas%20d%E2%80%99utilisation%203%20Centr%C3%A9%20sur%20l%E2%80%99architecture.. [Accès le 4 Juin 2023].

[4] «three-tier-architecture,» [En ligne]. Available: https://www.ibm.com/fr-fr/topics/three-tier-architecture. [Accès le 01 Avril 2023].

[5] «Architecture Modèle/Vue/Contrôleur,» [En ligne]. Available: https://www.irif.fr/~carton/Enseignement/InterfacesGraphiques/Cours/Swing/mvc.html. [Accès le 01 Avril 2023].

[6] «MVC Interview Questions and Answers For Graduates Part-1,» TheBlogReaders.com, [En ligne]. Available: https://theblogreaders.com/mvc-interview-questions-and-answers-for-graduates-part-1/. [Accès le 29 Mai 2023].

[7] «présentation du framework de développement d'applications multiplateforme,» 09 Octobre 2020. [En ligne]. Available: https://www.ionos.fr/digitalguide/sites-internet/developpement-web/flutter-cest-quoi/. [Accès le 1 Avril 2023].

[8] «Dart : présentation du langage de programmation,» 15 Octobre 2020. [En ligne]. Available: https://www.ionos.fr/digitalguide/sites-internet/developpement-web/le-langage-de-programmation-dart/. [Accès le 01 Avril 2023].

[9] «Node.js : définition simple et utilisation pratique,» Journal du net, 16 Septembre 2019. [En ligne]. Available: https://www.journaldunet.fr/web-tech/dictionnaire-du-webmastering/1445306-node-js-definition-simple-et-utilisation-pratique/. [Accès le 01 Avril 2023].

[10] «Les bases de JavaScript,» MDN Web Docs, [En ligne]. Available: https://developer.mozilla.org/fr/docs/Learn/Getting_started_with_the_web/JavaScript_basics. [Accès le 1 Avril 2023].

[11] «Tutoriel : intro à React,» [En ligne]. Available: https://fr.reactjs.org/tutorial/tutorial.html. [Accès le 2 Avril 2023].

[12] «NODEJS: SEND EMAIL WITH NODEMAILER,» onjsdev, 10 Mars 2023. [En ligne]. Available: https://onjsdev.com/article/nodejs-send-email-with-nodemailer. [Accès le 2 Avril 2023].

[13] «Sequelize,» [En ligne]. Available: https://sequelize.org/. [Accès le 2 Avril 2023].

[14] I. OMISOLA, «Qu'est-ce que Google Firebase et pourquoi devriez-vous l'utiliser ?,» [En ligne]. Available: https://www.savoirdanslavie.com/what-is-google-firebase-why-use-it/. [Accès le 4 Juin 2023].

[15] «Visual Studio Code,» Framalibre, [En ligne]. Available: https://framalibre.org/content/visual-studio-code. [Accès le 2 Avril 2023].

[16] «Enterprise Architect,» EXPERT-IT, [En ligne]. Available: https://www.expert-it.com/index.php/fr/enterprise-architect-sparx. [Accès le 2 Avril 2023].

[17] «Qu'est-ce que Android Studio? - définition de techopedia,» [En ligne]. Available: https://fr.theastrologypage.com/android-studio. [Accès le 2 Avril 2023].

[18] «Draw.io : un outil pour dessiner des diagrammes en ligne,» tice-education, 14 Janvier 2014. [En ligne]. Available: https://www.tice-education.fr/tous-les-articles-er-ressources/articles-internet/819-draw-io-un-outil-pour-dessiner-des-diagrammes-en-ligne. [Accès le 2 Avril 2023].

[19] «Découvrir Figma,» [En ligne]. Available: https://www.blogdumoderateur.com/tools/figma/. [Accès le 2 Avril 2023].

[20] «What is Postman?,» [En ligne]. Available: https://www.postman.com/. [Accès le 2 Avril 2023].

[21] «GitHub : qu’est-ce que c’est et comment apprendre à l’utiliser ?,» 7 Janvier 2023. [En ligne]. Available: https://datascientest.com/github-tout-savoir. [Accès le 4 Juin 2023].

[22] «Présentation de l'API Google Calendar,» [En ligne]. Available: https://developers.google.com/calendar/api/guides/overview?hl=fr. [Accès le 4 Juin 2023].

[23] «Intégrer Google Sign-In à votre application Web,» [En ligne]. Available: https://developers.google.com/identity/sign-in/web/sign-in?hl=fr. [Accès le 4 Juin 2023].

[24] «Tests unitaires vs Tests fonctionnels,» mobiskill, [En ligne]. Available: https://mobiskill.fr/blog/conseils-emploi-tech/tests-unitaires-vs-tests-fonctionnels-quelles-differences. [Accès le 14 6 2023].

Résumé

Les citoyens algériens sont fréquemment confrontés à des difficultés lorsqu'ils souhaitent obtenir des documents administratifs. Ils sont souvent contraints de se déplacer vers les établissements concernés, d'attendre en file pendant de longues heures, et ce malgré ces efforts, ils ne parviennent pas toujours à atteindre leurs objectifs en raison d'une mauvaise gestion des rendez-vous.

Dans ce contexte notre projet de fin d'étude et notre stage au sein de l'entreprise DATA MASTER propose la conception et le développement d'un système de gestion des rendez-vous pour les administrations algériennes. Notre objectif principal du système PlanIt est d'optimiser le processus de prise de rendez-vous, en réduisant les files d'attente et en améliorant l'efficacité du service public.

PlanIt permet aux citoyens de prendre des rendez-vous avec différentes administrations en utilisant une application mobile développer avec Flutter et facilite la gestion des rendez-vous des administrations par l'application web développé avec les Framework ReactJs et NodeJs.

Mots clés : Rendez-vous, PlanIt , Flutter, ReactJs, NodeJs, Framework.

Abstract

Algerian citizens are frequently faced with difficulties when they want to obtain administrative documents. They are often forced to travel to the establishments concerned, to wait in line for long hours, and despite these efforts, they do not always manage to obtain their objectives due to poor management of appointments.

In this context, our end-of-study project and our internship within the company DATA MASTER proposes the design and development of an appointment management system for the Algerian administrations. Our main objective of PlanIt system is to optimize the appointment booking process, reducing queues and improving the efficiency of the public service.

PlanIt which allows citizens to make appointments with different administrations using a mobile application developed with Flutter, and facilitates the management of administration appointments by the web application developed with the ReactJs and NodeJs Frameworks.

Keywords: Appointments, PlanIt, Flutter, ReactJs, NodeJs, Framework.

ملخص

كثيرا ما يواجه المواطنون الجزائريون صعوبات عندما يريدون الحصول على وثائق إدارية. غالبًا ما يضطرون للسفر إلى المؤسسات المعنية، والانتظار في طابور لساعات طويلة، وعلى الرغم من هذه الجهود، لا يتمكنون دائمًا من تحقيق أهدافهم بسبب سوء إدارة المواعيد.

في هذا السياق، يقترح مشروع نهاية الدراسة وتدريبنا الداخلي داخل شركة DATA MASTER تصميم وتطوير نظام إدارة المواعيد للإدارات الجزائرية. هدفنا الرئيسي من نظام PlanIt هو تحسين عملية حجز المواعيد وتقليل قوائم الانتظار وتحسين كفاءة الخدمة العامة.

PlanIt الذي يسمح للمواطنين بإجراء مواعيد مع إدارات مختلفة باستخدام تطبيق جوال تم تطويره باستخدام Flutter ، ويسهل إدارة المواعيد الإدارية بواسطة تطبيق الويب الذي تم تطويره باستخدام ReactJs وNodeJs Frameworks

الكلمات المفتاحية: موعد، Flutter، PlanIt، NodeJs ، ReactJs، إطار العمل.