

Ministère de l'enseignement supérieur et de la recherche scientifique

Direction Générale des Etudes Technologiques

Institut Supérieur des Etudes Technologiques de Djerba

Département Technologies de l'informatique



PROXIWEB

RAPPORT DE STAGE DE FIN D'ETUDES

Code Projet	SFETI14
-------------	---------

Présenté en vue de l'obtention du diplôme de

**Licence en Technologies de
L'informatique**

Parcours : MDW

***Développement d'une plateforme pour La
réservation des Services
de maison***

Élaboré par :

Nermine Chehit

Asma Ben Mechlia

Encadré par : Mr. Anis Asses

Effectué à :

Entreprise : ProxiWeb

Encadreur : Mr. Kaies Ben Mimoun

Année universitaire : 2023/2024

Dédicaces

En témoignage de ma gratitude et de mon amour, je dédie ce travail:

À celle que j'ai de plus cher

À celle qui a toujours garni mes chemins avec force et lumière,

À la plus belle perle du monde

Aucun mot, aucune dédicace ne peut exprimer mon respect, ma considération et l'amour éternel pour les sacrifices que vous avez consentis pour mon instruction et mon bien être, pour son soutien durant ma vie et pour son amour et sa bienveillance jour et nuit

À ma très chère mère

À l'homme de ma vie

À mon homme admirable, pour être le bon exemple de père par son soutien et ses encouragements, je ne pourrais jamais oublier la tendresse et l'amour dévoués par lesquels vous m'avez toujours entourée depuis mon enfance.

À mes chers parents

Ma réussite n'est autre que la vôtre, que Dieu vous donne tout le bonheur de monde

À mes frères, À tous mes chers amis

Que Dieu nos garde si tendres et aimants les uns envers les autres

À toute ma famille

Asma ben Mechlia



Dédicaces

À tous ceux qui ont croisé mon chemin et contribué, de près ou de loin, à mon parcours académique, je vous adresse ma profonde gratitude.

À mes parents

Pour leur amour inconditionnel, leur soutien constant et leurs sacrifices qui ont rendu cette réalisation possible. Leur encouragement m'a guidé à travers les défis et m'a donné la force d'aller de l'avant.

À mes professeurs

Pour leur enseignement, leur guidance experte et leur patience infinie. Leurs conseils précieux ont illuminé mon chemin vers la connaissance et la réussite académique.

À mes amis

Pour leur soutien indéfectible, leur présence réconfortante et leur encouragement sans faille. Leur soutien moral m'a donné la motivation nécessaire pour surmonter les moments de doute et d'incertitude.

Ce projet est le fruit de nombreuses mains et de nombreux cœurs qui ont partagé leur savoir, leur expérience et leur affection.

Nermine Chehit



Remerciements

Nous tenons tout d'abord à exprimer notre gratitude envers Dieu pour nous avoir accordé la vie et la santé tout au long de nos études.

Nous tenons à exprimer notre profonde gratitude et notre immense respect à monsieur Kaies Ben Mimoun, notre encadreur au sein de proxiweb, pour sa disponibilité, ses avis éclaircis, sa sympathie et ses judicieux conseils.

Nous tenons à exprimer notre profonde gratitude et notre immense respect à monsieur Anis Asses, notre encadrant universitaire de l'Institut Supérieur des études technologique de Djerba pour avoir accepté d'encadrer ce projet, et qui a toujours été disponible pour assurer le suivi de ce travail avec beaucoup de patience et aussi pour ses conseils judicieux.

Nous tenons aussi à exprimer nos chaleureux remerciements à tous ceux qui ont participé de près ou de loin au bon déroulement de notre stage de fin d'études.

Enfin nous tenons à remercier tous les membres du jury pour l'acceptation d'évaluer ce travail et nous espérons que ce présent rapport refléter à tout le travail effectué et satisfera tout consultant. Nous adressons également nos sincères remerciements à tout le personnel enseignant de l'ISET de Djerba, qui n'a cessé de consacrer ses efforts et ses énergies pour mener à bien notre formation

Remerciements

Sommaire

Introduction générale-----	1
Chapitre 1 : Cadre général du projet-----	2
Introduction -----	2
I. Présentation de l'organisme d'accueil-----	2
1. Présentation de la société-----	2
2. Localisation de la société-----	2
3. Les services offerts de la part de la société -----	3
II. Présentation générale du projet-----	3
1. Contexte de projet-----	3
2. Etude de l'existant -----	4
3. Solution proposée -----	4
III. Etude préalable -----	4
1. Méthodologie de travail -----	4
1.1 Méthode agile-----	5
1.2 Présentation de la méthodologie Scrum -----	5
2. Équipe Scrum -----	5
3. Identification des acteurs -----	6
4. Backlog produit-----	6
5. Backlog sprint-----	8
IV. Cahier des charges du projet-----	9
1. Objectifs généraux -----	9
2. Identification des besoins -----	9
2.1 Besoins fonctionnels-----	9
2.2 Besoins non fonctionnels-----	10
Conclusion-----	11
Chapitre 2 : Etude conceptuelle-----	12
Introduction -----	12
I. Diagramme de cas d'utilisation global-----	12
II. Diagramme de classe global-----	14
III. Modélisation graphique-----	15
IV. Choix technologique-----	16
V. Architecture de la plateforme-----	17

Sommaire

1. Architecture physique de la plateforme-----	17
2. Architecture logique de la plateforme-----	18
3. Gestion de l'État de l'Application -----	18
Conclusion-----	20
Chapitre 3 : Sprint 1 : Gestion des utilisateurs et leurs comptes-----	21
Introduction -----	21
I. Etude fonctionnelle -----	21
1. Backlog du sprint -----	21
2. Diagramme de cas d'utilisation du sprint 1 -----	22
II. Etude conceptuelle-----	24
1. Diagramme de classe sprint 1 -----	24
2. Diagramme de séquence du sprint 1 -----	24
3. La modélisation graphique du sprint 1 -----	26
III. Réalisation-----	27
1. Interface de connexion-----	27
2. Interface inscription client -----	28
3. Interface inscription prestataire-----	29
4. Interfaces relatives à l'administrateur-----	29
1. Gestion des clients-----	29
2. Gestion des prestataires-----	30
5. Interface mise à jour profile -----	30
Conclusion-----	31
Chapitre 4 : Sprint 2 : Gestion des services-----	32
Introduction -----	32
I. Étude fonctionnelle -----	32
1. Backlog du sprint -----	32
2. Diagramme de cas d'utilisation du sprint 2 -----	32
II. Étude Conceptuelle -----	34
1. Diagramme de classe du sprint 2 -----	34
2. Diagramme de séquence du sprint 2-----	34
3 Réalisation.-----	35
1. Interface gestion des services -----	35
2. Interface d'ajout d'un service -----	36
3. Interface de modification d'un service-----	37

Sommaire

4. Interface des services-----	37
Conclusion-----	38
Chapitre 5 : Sprint 3 : Gestion des annonces et demandes-----	39
Introduction -----	39
I. Étude fonctionnelle -----	39
1. Backlog sprint-----	39
2. Diagramme de cas d'utilisation du sprint 3-----	40
II. Etude conceptuelle-----	43
1. Diagramme de classe du sprint 3-----	43
2. Diagramme de séquence du sprint 3-----*	43
III. Réalisation-----	44
1. Interfaces relatives aux prestataires-----	44
2. Interfaces relatives aux clients-----	45
3. Interfaces relatives à l'administrateur-----	47
Conclusion-----	47
Chapitre 6 : sprint 4 : Gestion des messages et notifications-----	48
Introduction -----	48
I. Etude fonctionnelle -----	48
1. Backlog du sprint -----	48
2. Diagramme de cas d'utilisation du sprint 4 -----	49
II. Etude conceptuelle-----	50
1. Diagramme de classe du sprint 4-----	50
2. Diagramme de séquence du sprint 4-----	50
III. Réalisation-----	51
1. Interfaces relatives aux prestataires-----	51
2. Interfaces relatives aux clients-----	52
3. Caractéristiques techniques-----	53
3.1 Infographie-----	53
3.2 Contrôle de données-----	54
Conclusion-----	54
Conclusion générale et perspectives-----	55
Bibliographie et Nétographie-----	56
Liste des abréviations-----	57

Liste des tableaux

<i>Tableau 1: Equipe du projet.....</i>	6
<i>Tableau 2: Backlog produit.....</i>	8
<i>Tableau 3: Choix technologique.....</i>	17
<i>Tableau 4: Backlog du sprint 1.....</i>	22
<i>Tableau 5: Description textuelle du cas d'utilisation "S'inscrire"</i>	23
<i>Tableau 6 : Description textuelle du cas d'utilisation "mise à jour profil"</i>	23
<i>Tableau 7: Backlog du sprint 2.....</i>	32
<i>Tableau 8: Description textuelle du cas d'utilisation "ajouter un service".....</i>	33
<i>Tableau 9: Description textuelle du cas d'utilisation "modifier un service"</i>	34
<i>Tableau 10: Backlog du sprint 3.....</i>	40
<i>Tableau 11: Description textuelle du cas d'utilisation "ajouter annonce"</i>	41
<i>Tableau 12: Description textuelle du cas d'utilisation "archiver une annonce"</i>	42
<i>Tableau 13: Description textuelle du cas d'utilisation "ajouter une demande"</i>	42
<i>Tableau 14: Description textuelle du cas d'utilisation "supprimer une demande"</i>	42
<i>Tableau 15: Backlog du sprint 4.....</i>	49
<i>Tableau 16:Description textuelle du cas d'utilisation "Envoyer un message"</i>	50

Liste des figures

<i>Figure 1: Logo de la société</i>	2
<i>Figure 2: Emplacement de la Société</i>	3
<i>Figure 3: Les différents services de la Société</i>	3
<i>Figure 4:Découpage du projet</i>	8
<i>Figure 5: Diagramme de cas d'utilisation global</i>	13
<i>Figure 6: Diagramme de classe global</i>	14
<i>Figure 7: Page d'accueil avec Figma</i>	16
<i>Figure 8: Architecture physique de la plateforme</i>	17
<i>Figure 9 : Architecture logique de la plateforme</i>	18
<i>Figure 10: Structure du Redux Store</i>	19
<i>Figure 11: Fonctionnement du Redux Store</i>	20
<i>Figure 12: Diagramme de cas d'utilisation du sprint 1</i>	22
<i>Figure 13: Diagramme de classe du sprint 1</i>	24
<i>Figure 14: Diagramme de séquence de cas d'utilisation s'inscrire</i>	25
<i>Figure 15: Diagramme de séquence de cas d'utilisation s'authentifier</i>	25
<i>Figure 16: Diagramme de séquence de cas d'utilisation mettre à jour profil</i>	26
<i>Figure 17: Inscription avec Figma</i>	26
<i>Figure 18: Authentification avec Figma</i>	27
<i>Figure 19: Mise à jour du profil avec Figma</i>	27
<i>Figure 20: Interface connexion</i>	28
<i>Figure 21: Interface inscription client</i>	28
<i>Figure 22: Interface inscription prestataire</i>	29
<i>Figure 23: Interface de gestion des clients</i>	29
<i>Figure 24: Interface de gestion des prestataires</i>	30
<i>Figure 25: Interface de mis à jour profil</i>	30
<i>Figure 26: Diagramme de cas d'utilisation sprint 2</i>	33
<i>Figure 27: Diagramme de classe sprint 2</i>	34
<i>Figure 28: Diagramme de séquence sprint 2</i>	35
<i>Figure 29: Interface de gestion des services</i>	36
<i>Figure 30: Interface d'ajout d'un service</i>	36
<i>Figure 31: Interface de modification d'un service</i>	37
<i>Figure 32: Interface des services</i>	37

Liste des figures

<i>Figure 33: Diagramme de cas d'utilisation sprint3 -----</i>	<i>41</i>
<i>Figure 34: Diagramme de classe sprint3-----</i>	<i>43</i>
<i>Figure 35: Diagramme de séquence sprint 3-----</i>	<i>43</i>
<i>Figure 36: Formulaire demande d'emploi -----</i>	<i>44</i>
<i>Figure 37: Annonce -----</i>	<i>44</i>
<i>Figure 38: Interface gestion des demandes -----</i>	<i>45</i>
<i>Figure 39: Formulaire annonce -----</i>	<i>45</i>
<i>Figure 40 : Demande d'emploi-----</i>	<i>46</i>
<i>Figure 41: Interface gestion des annonces -----</i>	<i>46</i>
<i>Figure 42: Interface admin de gestion des demandes -----</i>	<i>47</i>
<i>Figure 43: Interface admin de gestion des annonces-----</i>	<i>47</i>
<i>Figure 44: Diagramme de cas d'utilisation sprint 4-----</i>	<i>49</i>
<i>Figure 45 : Diagramme de classe sprint 4 -----</i>	<i>50</i>
<i>Figure 46: Diagramme de séquence sprint 4-----</i>	<i>50</i>
<i>Figure 47: Interface de notifications pour les prestataires-----</i>	<i>51</i>
<i>Figure 48: Interface de chat avec le client -----</i>	<i>52</i>
<i>Figure 49: Interface de notifications pour les clients -----</i>	<i>52</i>
<i>Figure 50: Interface de chat avec le prestataire -----</i>	<i>53</i>
<i>Figure 51: Logo de la plateforme -----</i>	<i>53</i>
<i>Figure 52: Palette de couleur utiliser-----</i>	<i>54</i>

Introduction générale

Dans le monde d'aujourd'hui, les familles et individus font face à de multiples défis lorsqu'il s'agit d'assurer le bien-être et l'entretien de leur domicile. De la maintenance régulière aux besoins urgents de réparation, il est souvent difficile de trouver des prestataires de services fiables et abordables pour répondre à ces besoins domestiques. De plus, les fournisseurs de services eux-mêmes rencontrent des difficultés à élargir leur clientèle et à développer leurs activités dans un marché concurrentiel.

Pour adresser ces défis, la création d'une plateforme web dédiée aux services à domicile peut s'avérer être une solution efficace. Cette plateforme pourrait servir de point de rencontre entre les propriétaires et les prestataires de services, en facilitant la recherche de professionnels compétents et dignes de confiance en fonction de leur localisation et des services requis. Elle offrirait également aux fournisseurs de services une vitrine pour présenter leurs compétences et gérer leurs agendas.

C'est dans cette perspective que s'inscrit notre projet de développement, réalisé au sein de la société PROXIWEB. Notre objectif est de concevoir une plateforme innovante dédiée aux soins à domicile, baptisée « HomeCare », afin de répondre aux besoins variés des propriétaires tout en offrant aux prestataires de services un moyen de développer leurs activités.

Le présent rapport est composé par 6 chapitres :

Dans le premier chapitre, nous présentons le cadre général du projet, en définissant l'organisme d'accueil, le cadre général du projet et la méthodologie de travail. Le deuxième chapitre explique en détail la conception globale du projet, incluant diagrammes et les choix technologiques. Les chapitres suivants sont ensuite dédiés à une analyse approfondie des différents sprints, décrivant les Backlog sprint, les réalisations et les résultats obtenus à chaque phase.

Chapitre 1 : Cadre général du projet

Introduction

Dans ce chapitre introductif, nous avons pour but de mettre notre travail dans son contexte général avant d'entamer toute étude concernant ce projet.

Tout d'abord, nous décrirons l'environnement du stage à travers une présentation de l'organisme d'accueil, puis une présentation générale du projet, Ensuite nous présentons l'étude préalable de l'existant et enfin, nous terminons par le cahier de charge.

I. Présentation de l'organisme d'accueil

1. Présentation de la société

Proxiweb est une entreprise qui se consacre entièrement aux services informatiques, se situe à 41 Avenue Hedi Chaker Lafayette Tunis et il y-a aussi une agence à Djerba dans laquelle nous avons passé notre stage.

Proxiweb met à la disposition de ses clients ses compétences dans les domaines d'informatiques, la création de sites internet, la résolution de programme et le marketing. [1]



Figure 1: Logo de la société

2. Localisation de la société

Le siège de la société Proxiweb est situé à 41 Avenue Hedi Chaker Lafayette Tunis et il y-a aussi une agence à Djerba dans laquelle On a passé notre stage.

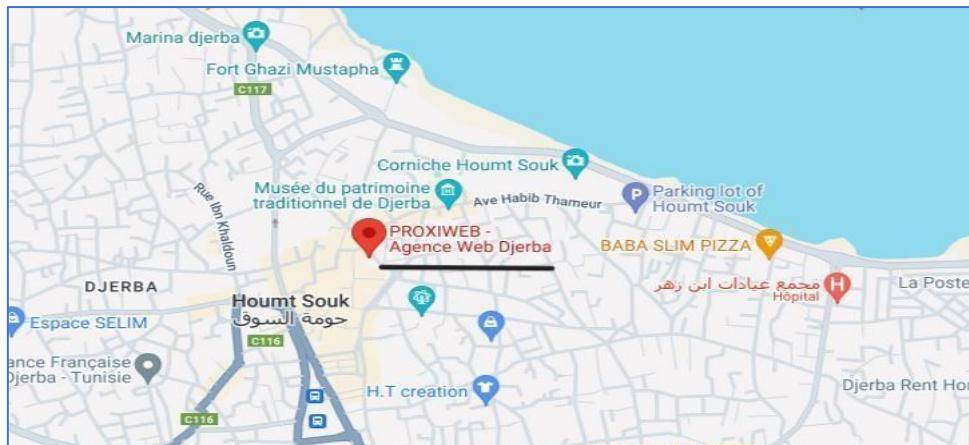


Figure 2: Emplacement de la Société

3. Les services offerts de la part de la société

L'activité principale de Proxiweb étant le conseil en informatique et la présentation des services dans le domaine de la programmation et le développement web. Proxiweb accompagne ses clients dans leur démarche sur le réseau.



Figure 3: Les différents services de la Société

II. Présentation générale du projet

1. Contexte de projet

HomeCare est une plateforme en ligne dédiée à la mise en relation entre les utilisateurs cherchant des services à domicile tels que femme de ménage, baby-sitting, aide-soignante, jardinage, et bien d'autres, et les prestataires de services qualifiés. Notre objectif est de faciliter la réservation et la prestation de ces services, offrant ainsi une solution pratique pour répondre aux besoins quotidiens des utilisateurs.

2. Etude de l'existant

Dans le domaine des travaux domestiques, trouver des professionnels compétents peut être difficile, tout comme pour les chercheurs d'emploi souhaitant offrir leurs services. Cela entraîne des retards et des complications pour les utilisateurs. De plus, obtenir un rendez-vous immédiat est rare, car les prestataires peuvent être débordés ou demander des tarifs élevés.

Notre plateforme vise à résoudre ces problèmes en facilitant la mise en relation des chercheurs d'emploi avec les employeurs. Nous offrons un espace où les chercheurs d'emploi peuvent présenter leurs compétences et leur disponibilité, et où les employeurs peuvent trouver des profils adaptés à leurs besoins. En simplifiant la recherche et la communication, notre plateforme aide à résoudre les problèmes domestiques et à trouver des opportunités professionnelles.

3. Solution proposée

Pour relever ces défis, nous avons développé une plateforme qui sert d'intermédiaire entre clients et prestataires de services. Les clients peuvent y publier des annonces, consulter les demandes d'emploi des prestataires, réserver des rendez-vous, demander des devis en fonction de leurs besoins et de leur localisation et mettre à jour leur profil.

Pour les prestataires, la plateforme permet de mettre à jour leur profil, de publier des demandes d'emploi, de consulter les annonces des clients et de gérer leur disponibilité. Cette approche facilite une gestion efficace de leur emploi du temps et une réponse rapide aux demandes des clients.

En misant sur la qualité, la transparence et la facilité d'utilisation, notre plateforme simplifie la recherche et la réservation de services domestiques, offrant une solution pratique pour tous.

III. Etude préalable

1. Méthodologie de travail

Pour mener à bien un projet complexe, une méthodologie de gestion de projet est essentielle. Elle fournit le cadre, les processus, et les techniques nécessaires pour gérer le personnel et le travail. Bien que diverses méthodologies existent, elles visent toutes à assurer le succès du projet. Pour notre projet, nous avons choisi la méthode agile, qui se concentre sur la valeur métier du livrable, améliore la qualité en continu et permet de réagir rapidement aux changements. Nous allons d'abord donner un aperçu des méthodologies agiles, puis nous concentrer sur la méthodologie Scrum, utilisée dans ce projet.

1.1 Méthode agile

Les besoins dans une équipe de développement logiciel qui travaille sur un projet interne sont parfois susceptibles d'être changés ou modifiés. De ce fait, l'équipe se trouve en train de développer une plateforme avec des spécifications parfois non précises, ce qui peut entraîner des retards dans le déploiement du projet. Ces problématiques ont poussé les ingénieurs à réinventer les méthodes de gestion de projet et de conception en introduisant ce que nous appelons la méthode Agile. « Une méthode agile est une approche itérative et incrémentale, qui est menée dans un esprit collaboratif avec juste ce qu'il faut de formalisme. Elle génère un produit de haute qualité tout en prenant en compte l'évolution des besoins des clients. C'est une façon de réduire le cycle de développement des projets informatiques qui sont gérés de manière adaptative, progressive et itérative » [2]

1.2 Présentation de la méthodologie Scrum

Scrum est la méthodologie suivie par la société Proxiweb pour la gestion de ses projets.

C'est une méthodologie agile itérative basée sur des itérations d'une durée de 2 à 4 semaines appelées Sprints. Durant chacune de ces itérations, une partie du produit nommée « Incrément » est réalisée en se basant sur les parties créées lors des itérations précédentes et livrées à la fin du Sprint. Elle offre les avantages suivants :

- Flexibilité.
- Pas de distinction des rôles au sein d'une équipe « Scrum ».
- Auto-confiance au sein de l'équipe en l'isolant durant le Sprint et en la rendant autogérée.
- Équipes soudées.

2. Equipe Scrum

Il existe différents rôles participants dans la mise en œuvre du projet selon la logique de Scrum :

- Product Owner : C'est un membre à part entière de l'équipe Scrum dont la responsabilité principale est de définir un produit qui apportera le maximum de valeur métier aux utilisateurs dans le temps et le budget. Il présente les besoins du client et propose de nouveaux objectifs. Il contrôle aussi la qualité et la date de délivrance de chaque sprint.
- Scrum Master : Il aide l'équipe à travailler de façon autonome et à s'améliorer constamment. Il est le garant de l'application du processus, Scrum en l'occurrence.
- Scrum Team : L'équipe de développement.

Le tableau ci-dessus présente l'équipe du projet qui a participé à la mise en œuvre de la solution.

Rôle	Nom et Prénom	Mission
Product Owner	Ben Mimoun Kaies	Présentation des caractéristiques et des fonctionnalités du produit à développer et approbation du produit à livrer.
Scrum Master	Asses Anis	Supervision de l'avancement du projet et des activités de l'équipe.
Scrum Team	Ben Mechlia Asma	Conception et développement.
	Chehit Nermine	

Tableau 1: Equipe du projet

3. Identification des acteurs

Un acteur présente une personne, une entité ou un système agissant d'une façon directe sur la plateforme. Les différents acteurs modélisant notre système sont :

- Client : personnes à la recherche de services pour leur maison ou leur entreprise.
- Prestataire : professionnels ou sociétés disponibles pour effectuer des services de nettoyage, jardinage, baby-sitting, petsitting, etc.
- Administrateur : est la personne la plus privilégiée. Il possède le droit de manipuler toutes les fonctionnalités offertes par notre plateforme.

4. Backlog produit

C'est une liste ordonnée de tout ce qui pourrait être requis dans le produit et elle est l'unique source des besoins pour tous les changements à effectuer sur le produit sous la responsabilité du Product Owner.

Le tableau ci-dessous présente le Backlog produit de notre plateforme où chaque user story est caractérisé par identifiant, un nom et une estimation.

ID	Taches	user story	Estim.
1	Gestion des comptes	1.1 En tant qu'utilisateur, je souhaite m'inscrire à la plateforme.	3
		1.2 En tant qu'utilisateur, je souhaite m'authentifier à la plateforme.	4
		1.3 En tant qu'utilisateur, je souhaite consulter mon profil.	1

Chapitre 1 : Cadre général du projet

		1.4 En tant qu'utilisateur, je souhaite mettre à jour mes données personnelles.	2
2	Gestion des utilisateurs	2.1 En tant qu'administrateur, je souhaite consulter la liste des utilisateurs de la plateforme.	2
		2.2 En tant qu'administrateur, je souhaite supprimer le compte d'un utilisateur.	2
		2.3 En tant qu'administrateur, je souhaite avertir un utilisateur.	2
3	Gestion des services	3.1 En tant qu'utilisateur, je souhaite consulter la liste des services.	2
		3.2 En tant qu'administrateur, je souhaite ajouter un service.	3
		3.3 En tant qu'administrateur, je souhaite modifier un service.	2
4	Gestion des annonces	4.1 En tant que client, je souhaite consulter la liste des demandes.	2
		4.2 En tant que client, je souhaite ajouter une annonce	2
		4.3 En tant que client, je souhaite modifier une annonce.	2
		4.4 En tant que client, je souhaite archiver une annonce.	2
		4.5 En tant que client, je souhaite supprimer une annonce.	2
		4.6 En tant qu'administrateur, je souhaite valider une annonce.	2
		4.7 En tant qu'administrateur, je souhaite supprimer une annonce.	2
5	Gestion des demandes	5.1 En tant que prestataire, je souhaite consulter la liste des annonces.	2
		5.2 En tant que prestataire, je souhaite ajouter une demande de travail.	2
		5.3 En tant que prestataire, je souhaite modifier une demande de travail.	2
		5.4 En tant que prestataire, je souhaite archiver une demande de travail.	2
		5.5 En tant que prestataire, je souhaite supprimer une demande de travail.	2
		5.6 En tant qu'administrateur, je souhaite valider une demande.	2
		5.7 En tant qu'administrateur, je souhaite supprimer une demande.	2
6	Gestion des notifications	6.1 En tant que prestataire, je souhaite postuler à une annonce.	3

		6.2 En tant que client, je souhaite m'intéresser à une demande.	2
7	Gestion des messages	7.1 En tant qu'utilisateur, je souhaite consulter la liste des messages.	2
		7.2 En tant qu'utilisateur, je souhaite ajouter un message.	4
		7.3 En tant que client, je souhaite consulter la liste des messages.	2
		7.4 En tant que client, je souhaite ajouter un message.	3

Tableau 2: Backlog produit

5. Backlog sprint

Le Backlog sprint est constitué à partir des thèmes du Backlog produit. Lors de la réunion de planification des sprints, l'équipe de développement choisit les éléments du Backlog produit qui seront réalisés. Le Backlog sprint est sous la responsabilité de l'équipe qui demeure la seule à pouvoir le modifier en cours d'itération.

La planification de notre projet est résumée par la figure ci-après :

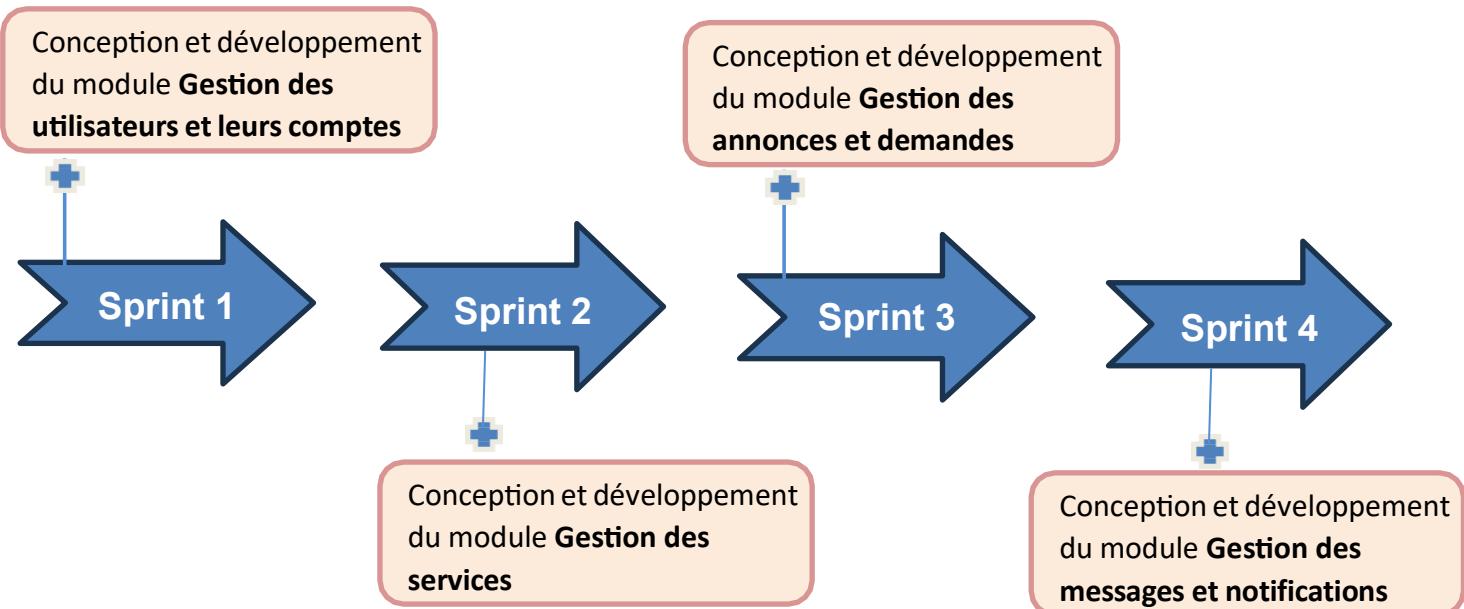


Figure 4:Découpage du projet

IV. Cahier des charges du projet

1. Objectifs généraux

Notre objectif principal consiste à développer une plateforme en ligne qui répond de manière efficace et pratique aux besoins des utilisateurs en termes de services domestiques. Nous avons pour ambition première de simplifier le processus de recherche et de réservation de services à domicile en proposant une solution centralisée et conviviale. Nous nous efforçons d'établir un lien transparent et de confiance entre les clients et les prestataires de services qualifiés, en introduisant un système d'évaluation par étoiles pour orienter les choix des utilisateurs. De plus, nous nous employons à optimiser la gestion du temps et de la disponibilité des prestataires de services afin d'assurer une réponse rapide et efficace aux demandes des clients. En adoptant une approche axée sur la qualité, la transparence et la facilité d'utilisation, notre objectif ultime est de rendre les services domestiques accessibles, fiables et pratiques pour tous les utilisateurs de notre plateforme.

2. Identification des besoins

Dans cette section, nous allons développer les besoins fonctionnels et non fonctionnels de la plateforme.

2.1 Besoins fonctionnels

Les besoins fonctionnels définissent les principales fonctionnalités de la plateforme, en mettant de côté les aspects de mise en œuvre.

Une analyse approfondie du système nous a permis d'identifier les principales exigences fonctionnelles des divers acteurs de notre application. Le système doit offrir à l'utilisateur la possibilité de créer un compte, de se connecter et de se déconnecter de manière sécurisée.

La plateforme doit permettre à l'**administrateur** de :

- ✓ Gérer les clients.
- ✓ Gérer les prestataires.
- ✓ Gérer les services.
- ✓ Gérer les annonces.
- ✓ Envoyer les avertissements vers les clients/prestataires.
- ✓ Gérer les demandes.

L'application doit permettre au **client** de :

- ✓ Consulter et modifier son profil.
- ✓ Consulter les demandes de travail par prestataires.
- ✓ Gérer les annonces.
- ✓ Publier une annonce.
- ✓ Gérer les messages.

L'application doit permettre au **prestataire** de :

- ✓ Consulter et modifier son profil.
- ✓ Consulter les annonces.
- ✓ Gérer les demandes de travail.
- ✓ Publier une demande de travail.
- ✓ Gérer les messages.

2.2 Besoins non fonctionnels

Les critères non-fonctionnels de notre application web, HomeCare, sont essentiels pour assurer son fonctionnement sans faille et un rendement optimal. Voici les exigences non-fonctionnelles clés que nous avons identifiées pour notre plateforme :

- ✓ Facilité d'utilisation et Rapidité : L'interface de HomeCare doit être intuitive et facile à naviguer pour tous les utilisateurs, offrant par la même occasion une expérience rapide et sans heurts.
- ✓ Compatibilité et Robustesse : HomeCare devrait être en mesure de fonctionner sur diverses plateformes de système d'exploitation et être à l'épreuve des pannes. L'application doit aussi gérer efficacement les erreurs pour assurer une expérience utilisateur fiable et sans interruption.
- ✓ Sécurité et Flexibilité : La protection des données et l'intégration de nouvelles fonctionnalités doivent être placées au premier plan pour répondre à l'évolution des exigences des utilisateurs et du marché.
- ✓ Extensibilité et Lisibilité du Code : HomeCare devrait pouvoir supporter une augmentation du nombre d'utilisateurs sans affecter sa performance. De même, son code devrait être bien structuré et documenté pour faciliter les mises à jour et les améliorations futures.

Conclusion

Dans ce chapitre, nous avons introduit notre projet en commençant par présenter l'organisme d'accueil. Puis, nous avons procédé à une analyse critique de l'existant dans le cadre de notre étude, avant de définir notre solution. Ensuite, nous avons exposé la méthodologie agile SCRUM que nous allons adopter pour la réalisation de notre projet. Enfin, nous avons conclu par la présentation du cahier de charge.

Maintenant que cette présentation est achevée, le chapitre suivant sera consacré à l'étude conceptuelle.

Chapitre 2 : Etude conceptuelle

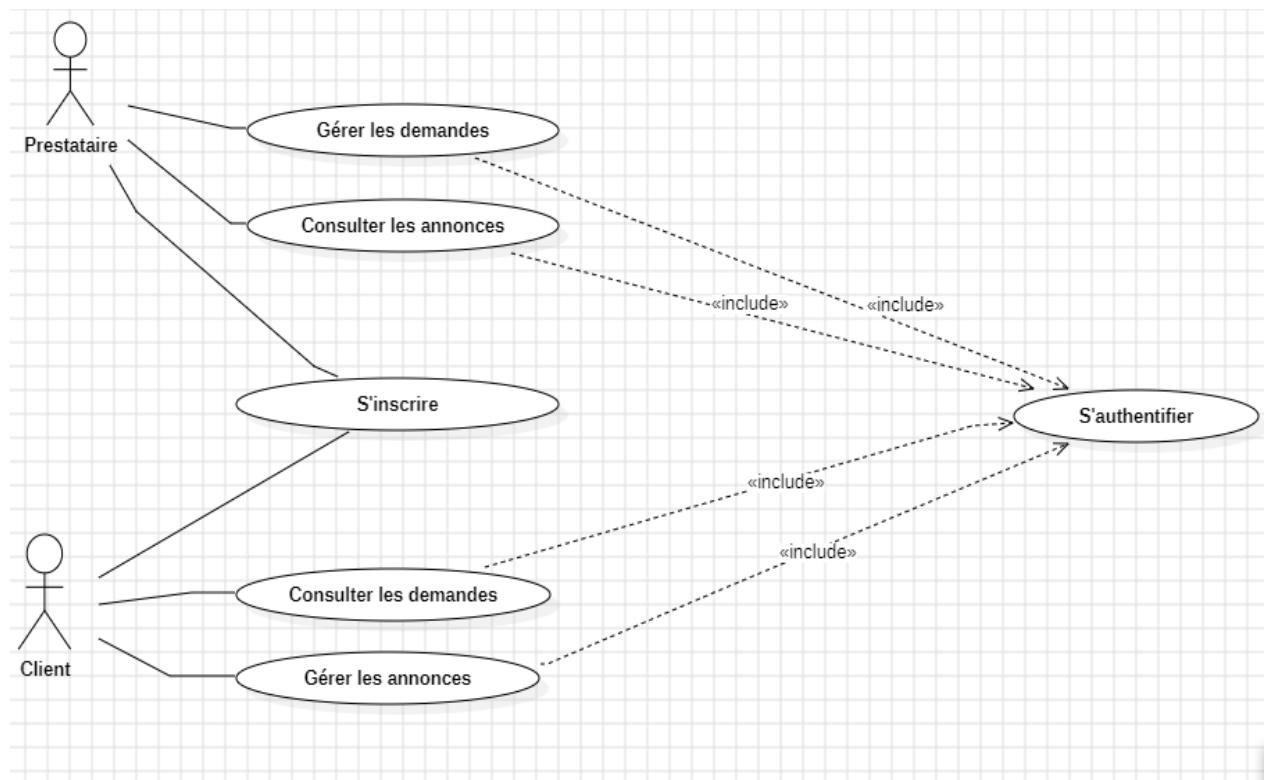
Introduction

Après avoir exposé le projet dans le chapitre précédent et avant d'entamer la phase de développement, nous nous attelons à présent à l'élaboration de la conception du projet.

Ce chapitre est consacré à la conception globale. D'abord, nous introduirons les diagrammes de cas d'utilisation et de classes qui forment le système, ainsi que leurs associations, pour structurer efficacement les différentes classes de notre application. Ensuite, nous proposerons une modélisation graphique de la plateforme, visant à concrétiser nos idées par des maquettes interactives de qualité. Enfin, nous présenterons les choix technologiques et l'architecture de la plateforme.

I. Diagramme de cas d'utilisation global

Le diagramme de cas d'utilisation est un moyen simple d'exprimer les exigences qui représente une vision globale sur les fonctionnalités fondamentales de l'application. Il correspond à un ensemble de transactions exécutées lors de l'interaction entre les acteurs et le système.



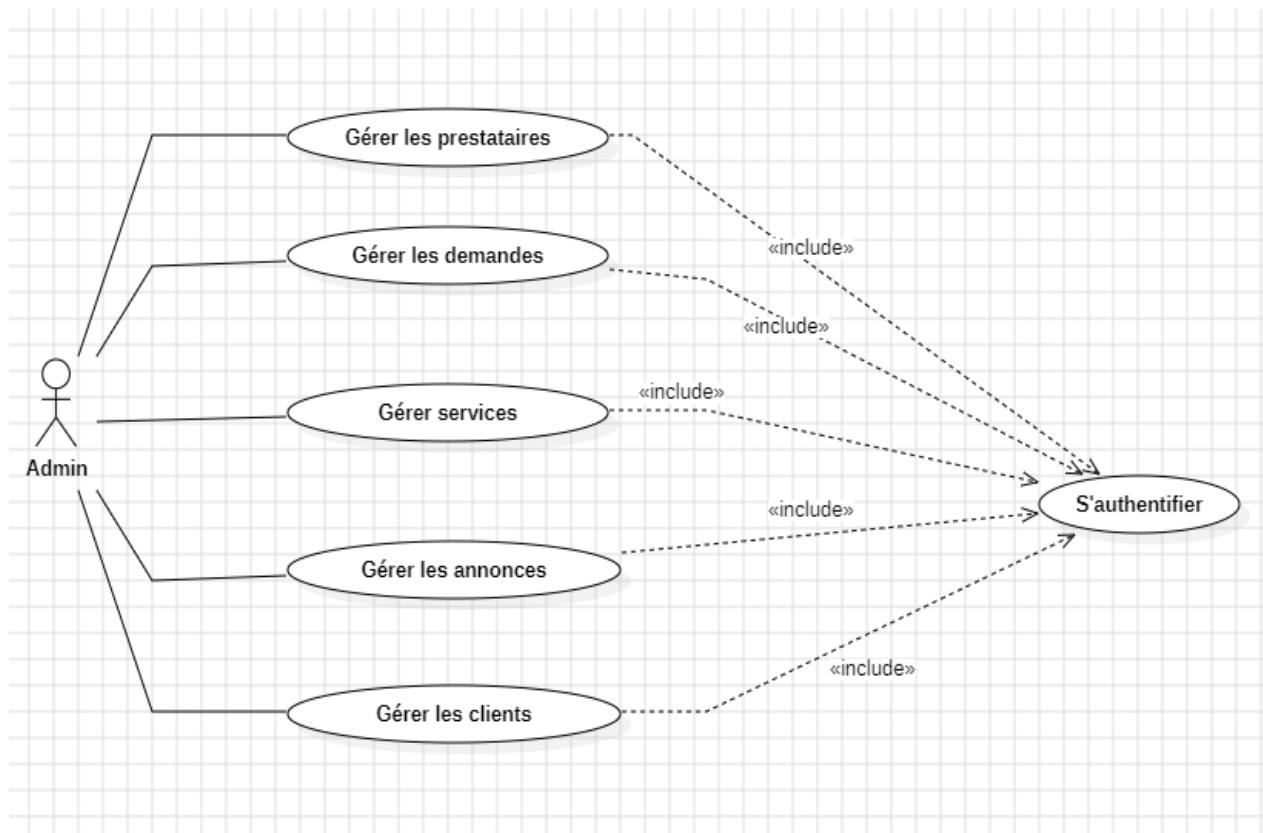


Figure 5: Diagramme de cas d'utilisation global

La figure ci-dessus présente l'ensemble de la situation d'utilisation de notre projet, définissant les acteurs clés et leurs principaux objectifs. Ce diagramme particulier offre un aperçu des relations entre ces acteurs et montre comment ils interagissent les uns avec les autres pour fournir différents services.

Au sommet de la hiérarchie, nous avons le prestataire. Il peut s'inscrire et se connecter à son compte, personnaliser les informations personnelles, effectuer des recherches sur les services présents dans le système, ainsi que de rechercher des offres d'emploi publiées par d'autres utilisateurs.

La deuxième partie prenante est le client, qui peut utiliser notre plateforme pour fixer des rendez-vous, demander des devis, publier des offres d'emploi, gérer les offres reçues à la suite de ces annonces, fournir des commentaires et des évaluations pour chaque transaction et éventuellement modifier toute partie de l'accord si nécessaire.

Le troisième participant, qui revêt le rôle crucial d'administrateur, assume une responsabilité primordiale dans la gestion holistique des clients, des prestataires de services, des annonces, ainsi que des divers services proposés.

Ce pilier central s'engage à orchestrer avec efficacité toutes les interactions et opérations au sein de la plateforme, garantissant ainsi une coordination efficace entre les clients, les prestataires de services.

II. Diagramme de classe global

Le diagramme de classe est un outil visuel essentiel pour les développeurs et les concepteurs de systèmes, permettant de comprendre la structure globale du système et de prendre des décisions de conception éclairées. Il offre une base solide pour le développement du code source et facilite la communication au sein de l'équipe de développement.

Ce diagramme, conforme à la norme UML, comme illustré dans la figure 6, représente les classes et leurs relations, offrant ainsi une vue logique d'un système informatique. Son but principal est de simplifier l'interaction entre les objets du système modélisé. Les diagrammes de classes expriment la structure statique du système en décrivant les classes et les relations entre elles. Chaque classe représente un ensemble d'objets, tandis que chaque association représente un ensemble de liens entre ces objets.

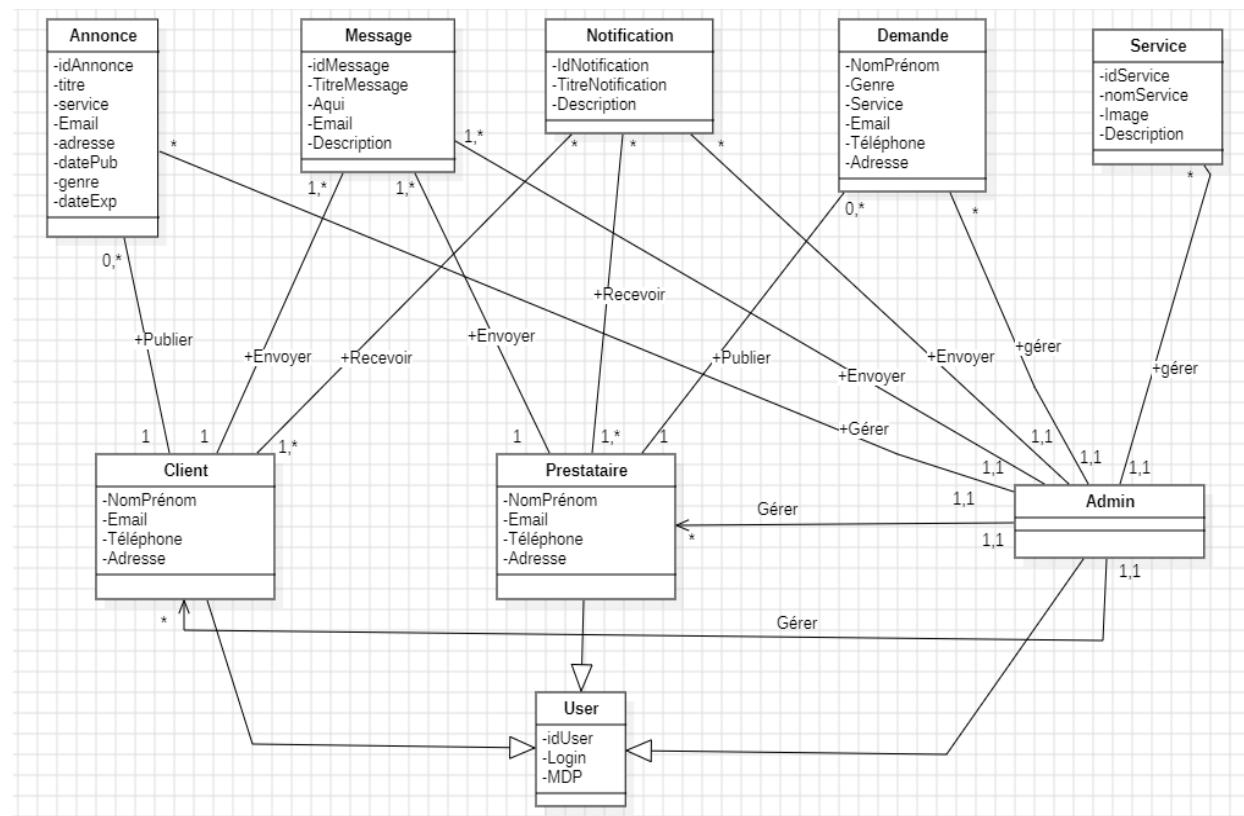
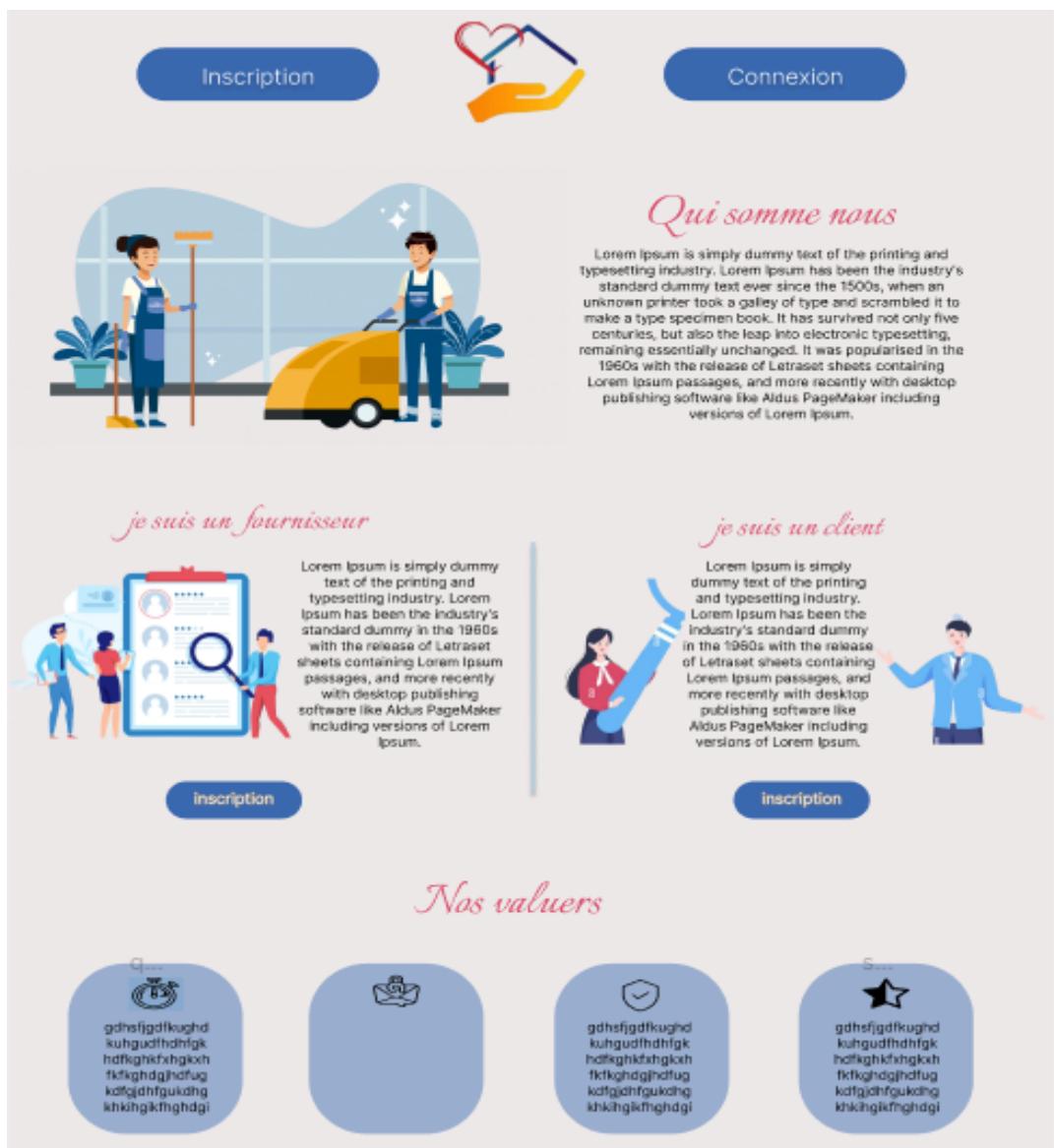


Figure 6: Diagramme de classe global

III. Modélisation graphique

Nous utilisons Figma pour réaliser la modélisation graphique de notre plateforme, comme illustre dans la figure 8 ci-dessous. Figma est essentiellement notre atelier de design en ligne, où nous transformons nos idées en superbes maquettes interactives. Avec Figma, nous pouvons concevoir chaque détail de l'expérience utilisateur, depuis les premiers écrans d'accueil jusqu'aux différents parcours de navigation. C'est un peu comme si nous peignions le tableau de l'expérience HomeCare, ajoutant chaque couleur et chaque forme pour créer quelque chose de vraiment spécial. En utilisant Figma, nous pouvons voir notre vision de HomeCare prendre forme sous nos yeux, nous rapprochant un peu plus chaque jour de notre objectif de rendre notre plateforme aussi conviviale que possible pour nos utilisateurs.



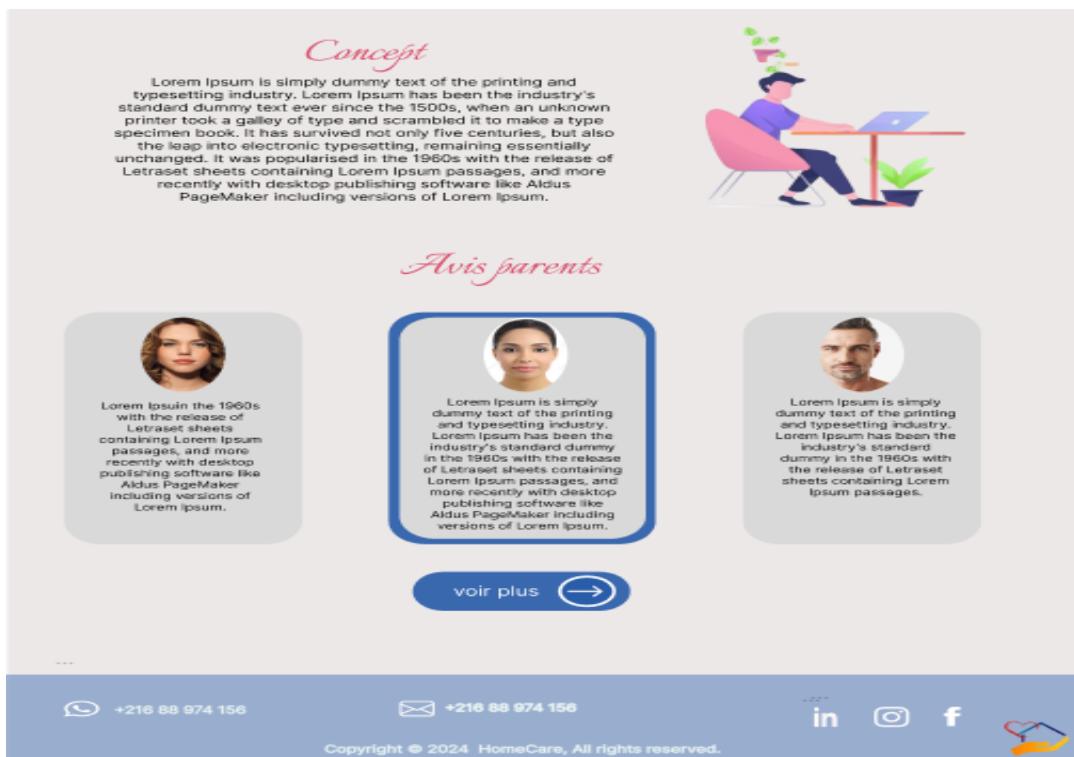


Figure 7: Page d'accueil avec Figma

IV. Choix technologique

La qualité de notre travail repose largement sur la sélection appropriée des technologies et des outils. Ainsi, nous allons examiner attentivement les différentes options pour élaborer notre projet et choisir l'environnement de travail qui convient le mieux. Ce choix est guidé aussi par l'existant et les exigences de l'entreprise.

Taches /Mission	Outil	logo	Description
Développement : Front End	React		C'est une bibliothèque JavaScript frontale open source qui permet de concevoir des interfaces utilisateur ou des composants d'interface utilisateur.[9]
Développement : Back End	Node		C'est un environnement d'exécution open-source, multi-plateforme et single-thread, conçu pour créer des applications serveur rapides et évolutives. [7]
Editeur du code	Visual Studio Code		C'est un éditeur de code extensible développé par Microsoft pour Windows, Linux et MacOs. [6]
Gestionnaire de base de données	Mongo DB		C'est un système orienté documents, offre une grande flexibilité pour associer et stocker différents types de données. [8]

Conception des diagrammes	StarUML		C'est un outil de génie logiciel dédié à la modélisation UML. Il est multiplateforme et fonctionne sous Windows, Linux et MacOs. [5]
Modélisation graphique	Figma		C'est un éditeur de graphiques vectoriels et un outil de prototypage. Il est principalement basé sur le web. [3]
Rédacteur du rapport	Microsoft Office Word		C'est un logiciel de traitement de texte publié par Microsoft. [10]

Tableau 3: Choix technologique

V. Architecture de la plateforme

1. Architecture physique de la plateforme

Dans notre projet, nous avons opté pour l'architecture 3-tiers afin d'assurer des performances optimales et de répondre aux exigences fonctionnelles du projet.

Cette architecture comprend 3 niveaux :

- ❖ Le niveau de présentation, qui correspond à la partie visible et interactive de la plateforme pour les utilisateurs.
- ❖ Le niveau de traitement métier des données, où sont implémentées les règles de gestion et la logique applicative.
- ❖ Le niveau de données, qui gère l'accès aux données en fournissant des méthodes pour interagir avec la base de données, notamment pour effectuer des opérations telles que les insertions, les mises à jour, les suppressions et l'extraction de données en fonction des entrées. [9]

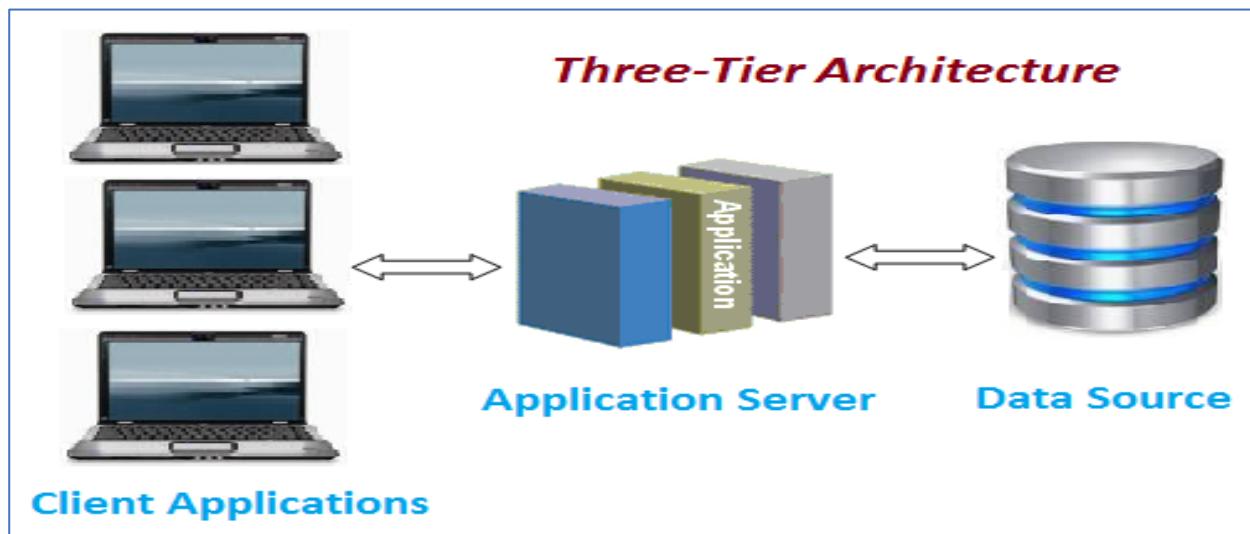


Figure 8: Architecture physique de la plateforme

2. Architecture logique de la plateforme

- L'utilisateur interagit avec l'application en effectuant une requête HTTP vers le serveur.
- La requête est reçue par le serveur Node.js, qui utilise Express.js pour gérer les routes entrantes.
- Express.js mappe la route correspondante et déclenche la logique appropriée pour traiter la demande.
- Dans le traitement de la requête, Node.js peut faire appel à des services backend (Mongoose) pour récupérer des données à partir de MongoDB.
- Les données récupérées sont transmises aux composants React du frontend.
- Les composants React peuvent manipuler ces données, les afficher dans l'interface utilisateur et réagir aux actions de l'utilisateur.

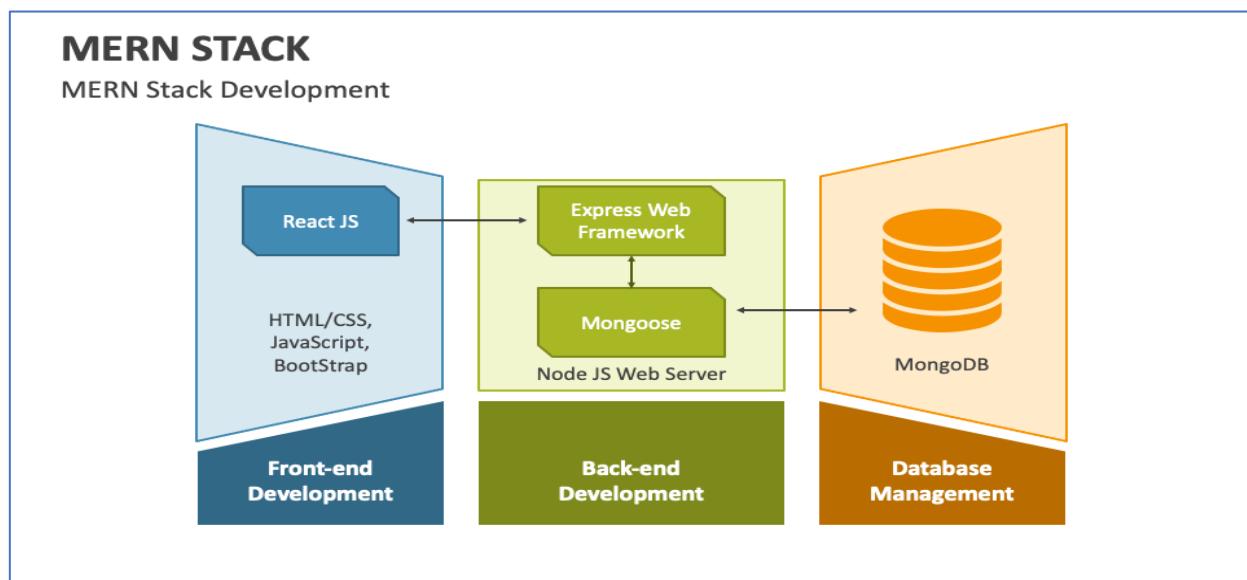


Figure 9 : Architecture logique de la plateforme

3. Gestion de l'État de l'Application

Afin de gérer notre plateforme, nous avons mis en place la structure du Redux Store.

➤ Structure du Redux Store

Le store Redux est un objet unique qui contient l'état de l'application et plusieurs méthodes pour interagir avec cet état. Voici les principaux composants :

État (State): C'est un objet JavaScript qui contient les données de l'application. Il représente l'état global de l'application.

Actions: Ce sont des objets qui décrivent les événements qui se produisent dans l'application. Une action doit avoir un type (type) et peut éventuellement contenir des données supplémentaires (payload).

Reducers : Ce sont des fonctions qui prennent l'état actuel et une action comme arguments, et retournent un nouvel état. Les reducers doivent être des fonctions pures, c'est-à-dire qu'elles ne modifient pas l'état d'origine mais retournent un nouvel état modifié.

Store : C'est un objet qui contient l'état de l'application, et plusieurs méthodes pour interagir avec cet état.

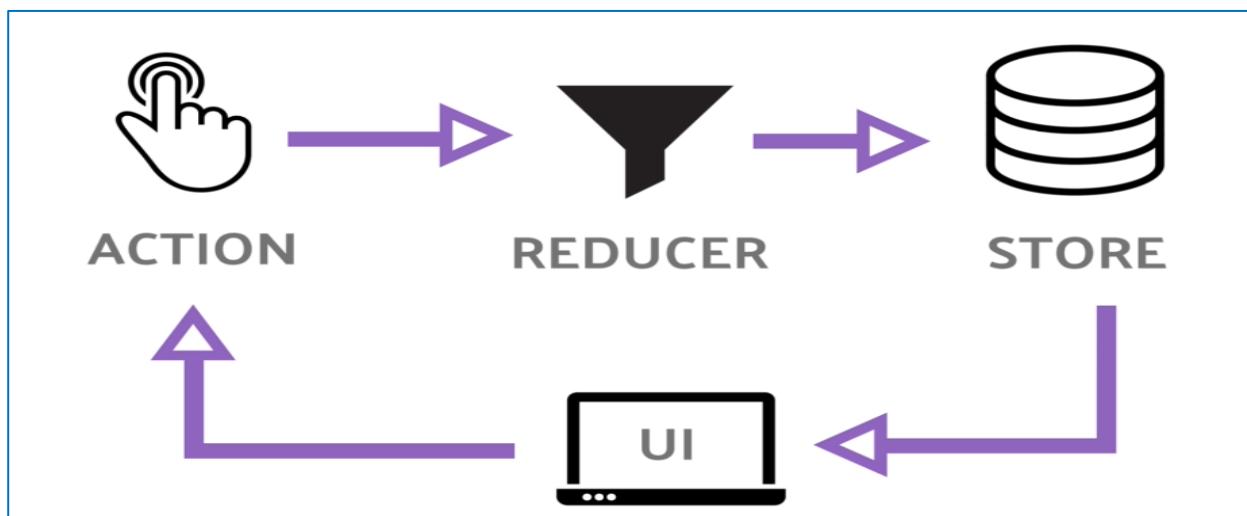


Figure 10: Structure du Redux Store

➤ Fonctionnement du Redux Store

Le fonctionnement du Redux Store repose sur trois principes fondamentaux :

Source unique de vérité (Single Source of Truth) : L'état de toute l'application est stocké dans un objet de state unique au sein du store. Cela facilite la gestion de l'état et le débogage.

L'état est en lecture seule (State is Read-Only) : La seule manière de modifier l'état est d'émettre une action, un objet décrivant ce qui s'est passé. Cela garantit que l'état ne peut être modifié de manière imprévisible.

Les modifications sont faites avec des fonctions pures (Changes are made with Pure Functions) : Pour spécifier comment l'état est transformé par les actions, vous écrivez des reducers, qui sont des fonctions pures. [12]

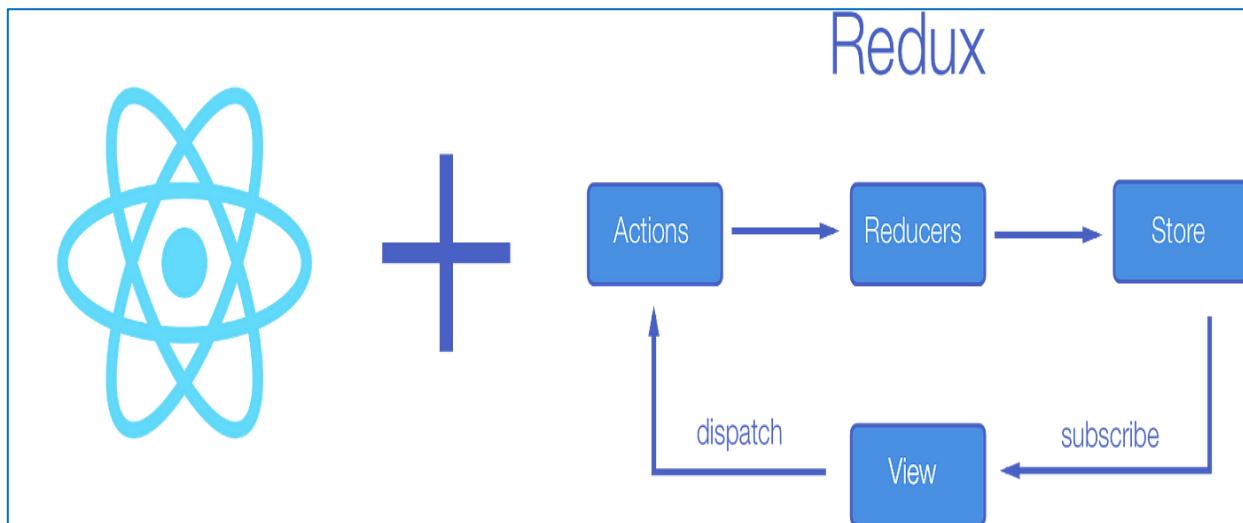


Figure 11: Fonctionnement du Redux Store

Conclusion

Tout au long de ce chapitre, nous avons exposé la conception globale du système et défini l'environnement de travail. Dans le chapitre suivant, nous entamerons la phase de réalisation des sprints, en commençant par le premier sprint dédié à la gestion des utilisateurs et de leurs comptes.

Chapitre 3 : Sprint 1 : Gestion des utilisateurs et leurs comptes

Introduction

Dans ce chapitre, nous nous concentrerons sur la première étape du cycle de vie de notre projet. Nous commencerons par détailler les tâches à accomplir dans le Backlog du Sprint, puis nous aborderons les phases de conception. Enfin, nous explorerons la mise en œuvre de ce module.

I. Etude fonctionnelle

1. Backlog du sprint

ID	User Story	Description	Estim.
1.1	En tant qu'utilisateur, je souhaite m'inscrire à la plateforme.	Développer la vue d'inscription.	2
		Générer l'API de création d'un compte.	2
		Consommer l'API.	1
1.2	En tant qu'utilisateur, je souhaite m'authentifier à la plateforme.	Développer la vue d'authentification.	2
		Créer l'action d'interception de toutes les requêtes du front-end et d'affecter un jeton correspondant.	3
		Implémenter l'API d'authentification.	2
		Consommer l'API.	1
1.3	En tant qu'utilisateur, je souhaite consulter mon profil.	Développer la vue de consultation du profil.	2
		Générer l'API permettant de retourner des données d'un utilisateur.	3
		Consommer l'API.	1
1.4	En tant qu'utilisateur, je souhaite mettre à jour mes données personnelles.	Développer la vue de modification du profil.	2
		Créer l'API de modification du profil.	2
		Consommer l'API.	1
1.5	En tant qu'administrateur, je souhaite consulter la liste des utilisateurs de la plateforme.	Créer la "datatable" de sélection des utilisateurs.	3
		Implémenter l'API de sélection des utilisateurs.	2
		Consommer l'API.	1

1.6	En tant qu'administrateur, je souhaite supprimer le compte d'un utilisateur.	Implémenter l'API de suppression d'un utilisateur.	2
		Consommer l'API.	1
1.7	En tant qu'administrateur, je souhaite avertir un utilisateur.	Implémenter l'API de suppression d'un utilisateur.	2
		Consommer l'API.	1

Tableau 4: Backlog du sprint 1

2. Diagramme de cas d'utilisation du sprint 1

Le diagramme de cas d'utilisation du premier sprint présenté par la figure 8 pour objectif de déterminer les fonctionnalités de la première valeur métier.

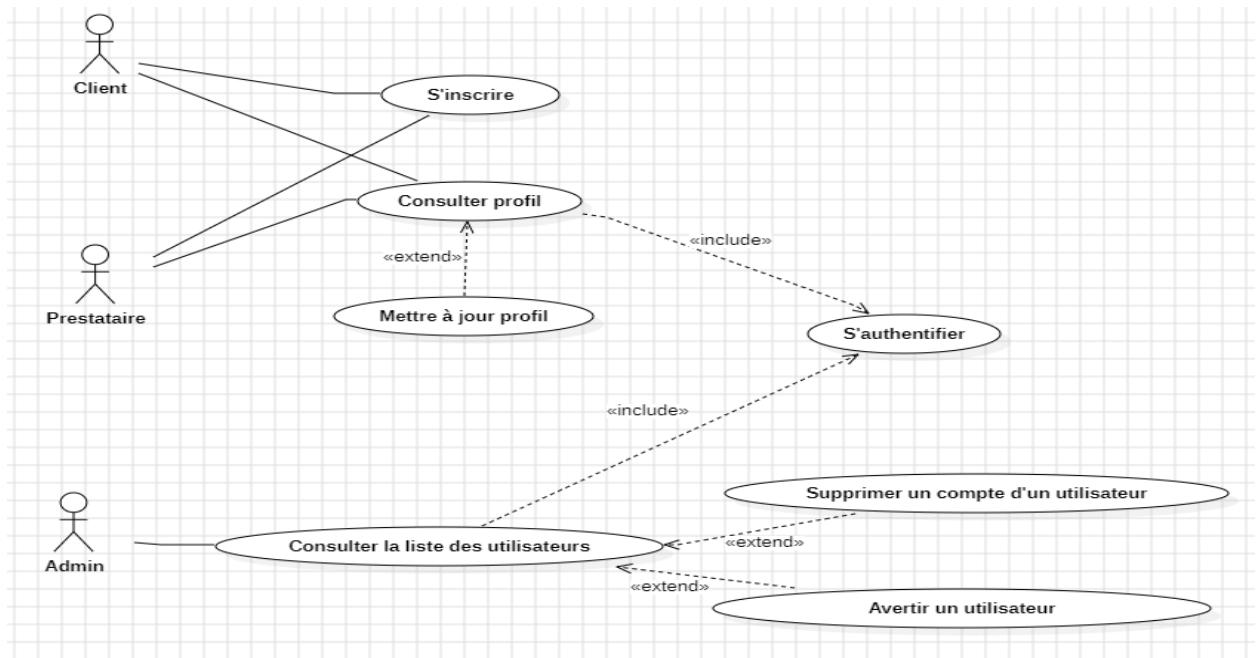


Figure 12: Diagramme de cas d'utilisation du sprint 1

Comme la montre la figure ci-dessus, ce sprint facilite l'accès des utilisateurs à la plateforme et la gestion de leurs données personnelles.

Si un utilisateur est nouveau sur la plateforme, il a la possibilité de créer un compte. Si les données saisies sont correctes, il sera dirigé vers la page d'authentification. Sinon, une erreur s'affichera pour signaler le problème. En cas d'authentification réussie, l'utilisateur accèdera à la page d'accueil ; sinon, il sera redirigé vers la page d'authentification.

Une fois authentifié, l'utilisateur peut accéder à son profil et mettre à jour ses données personnelles. Pendant cette période, l'administrateur a la capacité de gérer tous les utilisateurs enregistrés sur la plateforme. Il peut consulter les listes des utilisateurs, les avertir ou supprimer leurs comptes.

Dans cette partie, nous allons élaborer une explication textuelle des sous cas d'utilisation suivants: "Authentification", "Inscription" et "Mise à jour de profil".

- **Description textuelle du cas d'utilisation "S'inscrire"**

Acteur	Utilisateur
Description	S'inscrire au système.
Préconditions	Disponibilité d'accès.
Post-conditions	Utilisateur inscrit.
Scénario principal	L'utilisateur remplit le formulaire, puis confirme.
	Le système vérifie l'unicité de l'utilisateur.
	Le système ajoute l'utilisateur.
Scénario alternatif	Le système retourne un message d'erreur indiquant l'existence d'un utilisateur avec des coordonnées pareilles.

Tableau 5: Description textuelle du cas d'utilisation "S'inscrire"

- **Description textuelle du cas d'utilisation " Mettre à jour profil "**

Acteur	Utilisateur
Description	Mettre à jour profil.
Préconditions	Utilisateur authentifié.
Post-conditions	Profil mis à jour.
Scénario principal	L'utilisateur visite son profil dans le Dashboard.
	Un formulaire s'affiche.
	L'utilisateur remplit le formulaire et confirme.
	Le système met à jour le profil.
	Le système affiche le profil modifié.
Scénario alternatif	L'utilisateur laisse un champ vide, ou saisit des données invalides.
	Le système répond avec des messages d'erreurs.

Tableau 6 : Description textuelle du cas d'utilisation "mise à jour profil"

II. Etude conceptuelle

1. Diagramme de classe sprint 1

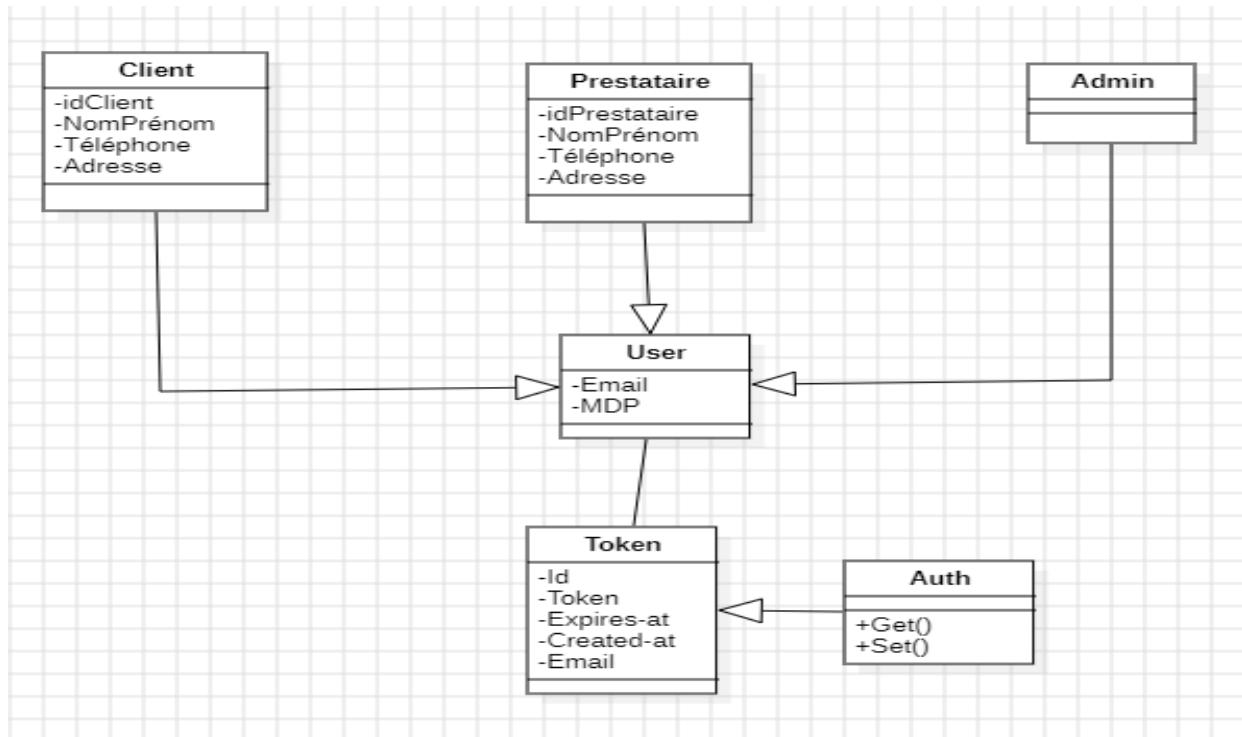
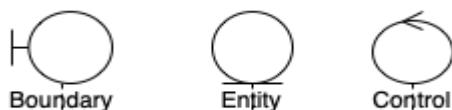


Figure 13: Diagramme de classe du sprint 1

Le diagramme de classe illustre les interactions dynamiques entre les objets, offrant une perspective temporelle sur les collaborations dans le système. Dans ce qui suit, nous présenterons le diagramme de séquence pour le cas d'utilisation "S'authentifier".

2. Diagramme de séquence du sprint 1

Lors de ce premier sprint, nous avons employé des diagrammes de séquence pour illustrer le comportement du système. Ces diagrammes montrent les interactions entre les objets et les acteurs du système au fil du temps, en se focalisant sur un cas d'utilisation particulier, également appelé histoire d'acteur. Dans la section suivante, nous présenterons les diagrammes de séquence correspondant aux principaux cas d'utilisation de notre sprint 1.



Dans notre contexte, le terme "Boundary" désigne une interface graphique ou une autre interface homme-machine. La classe "Entity," en revanche, est utilisée pour modéliser des informations durables et souvent persistantes. Enfin, la classe "Control" sert à représenter la coordination, la gestion et le contrôle des autres objets.

La figure ci-dessous détaille l'enchaînement des séquences nécessaire à la réalisation du cas d'utilisation s'inscrire.

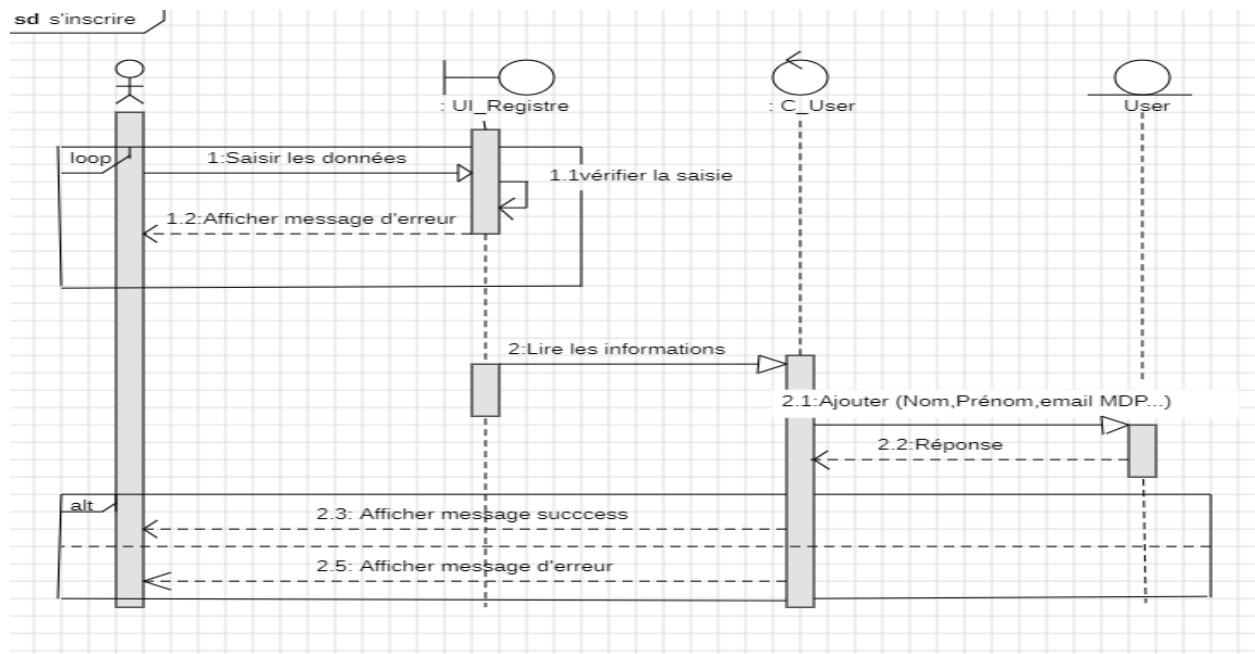


Figure 14: Diagramme de séquence de cas d'utilisation s'inscrire

La figure 13 montre les étapes successives requises pour réaliser le cas d'utilisation d'authentification.

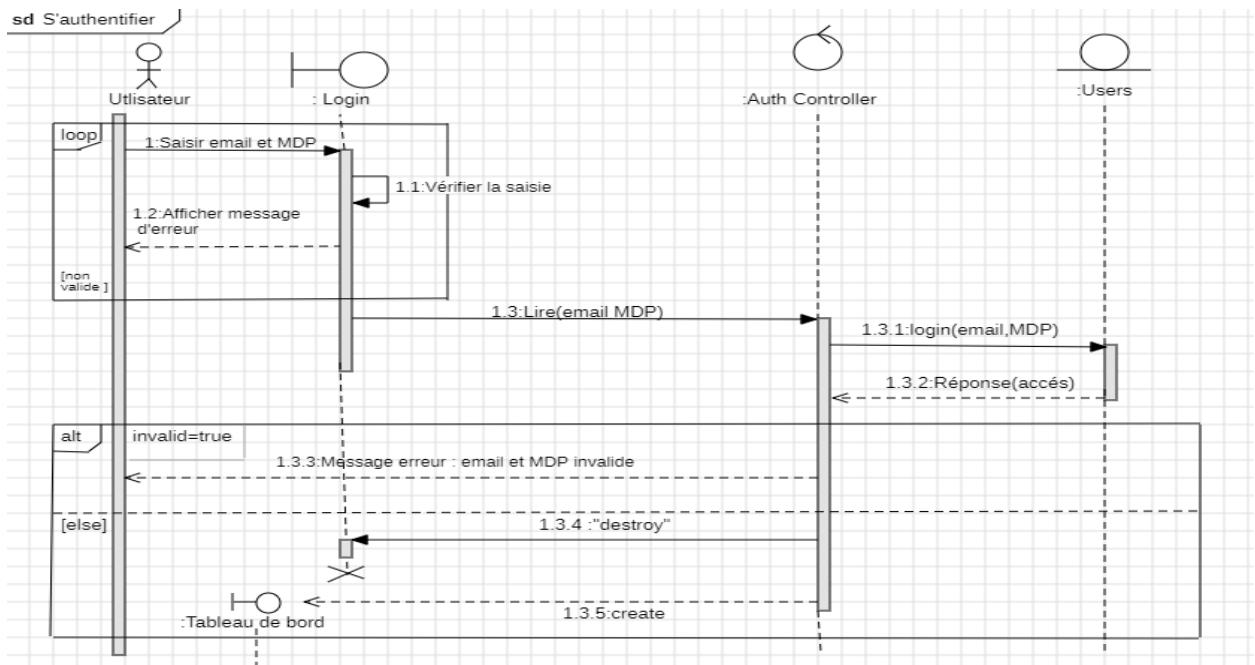


Figure 15: Diagramme de séquence de cas d'utilisation s'authentifier

La figure 14 illustre les différentes étapes nécessaires pour effectuer le cas d'utilisation de mise à jour.

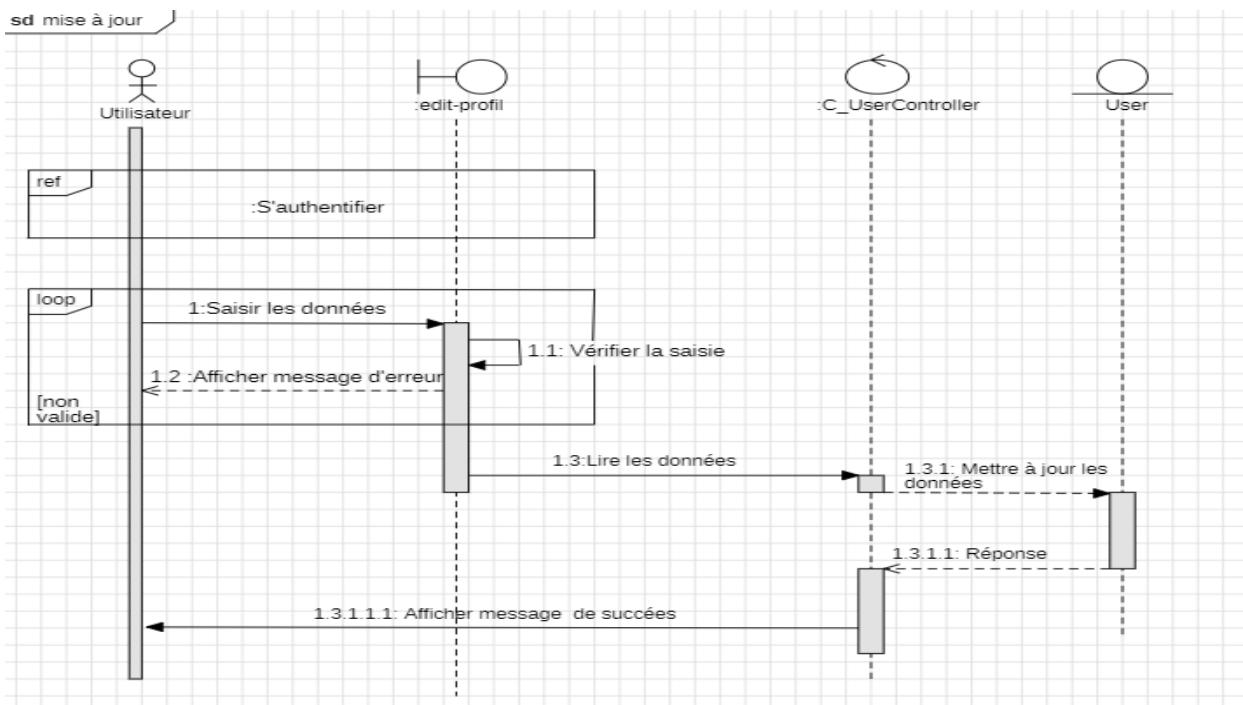


Figure 16: Diagramme de séquence de cas d'utilisation mettre à jour profil

3. La modélisation graphique du sprint 1

Figma est utilisé pour créer la représentation visuelle de notre plateforme. Cette section présente nos concepts concernant « l'inscription », « l'authentification » et « la mise à jour du profil » conformément à notre vision pour HomeCare.

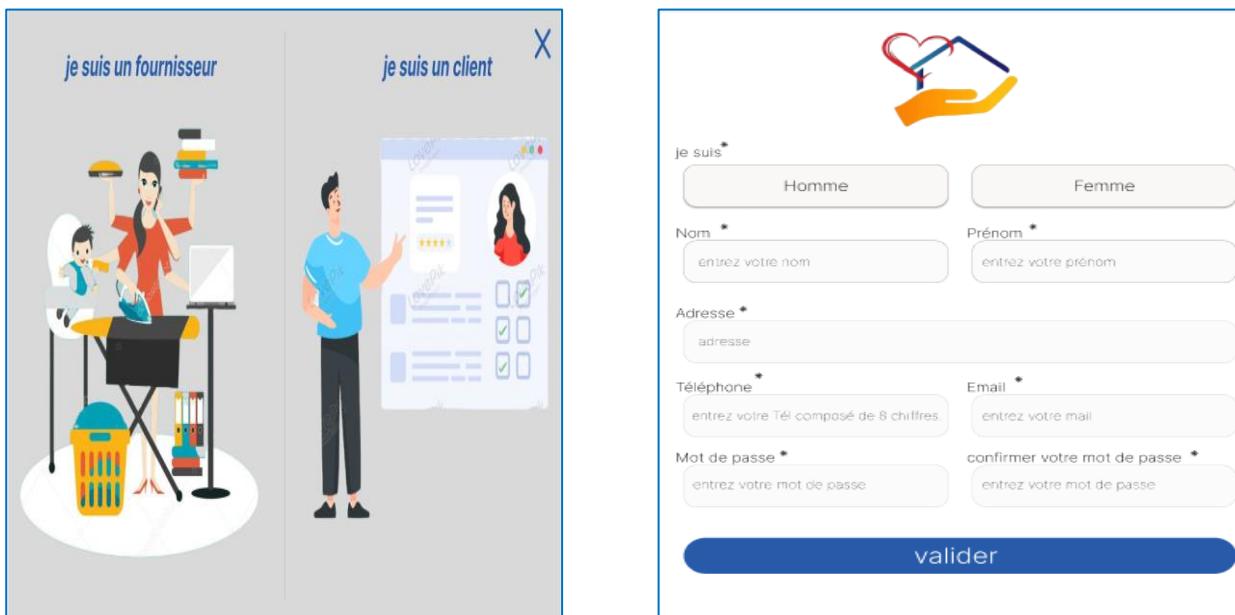


Figure 17: Inscription avec Figma

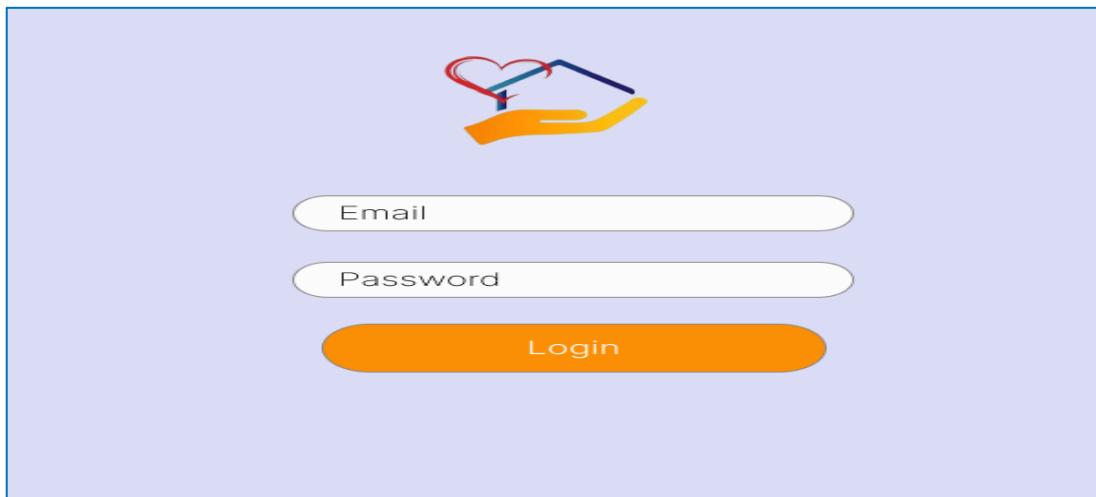


Figure 18: Authentification avec Figma



Figure 19: Mise à jour du profil avec Figma

III. Réalisation

Nous montrons le résultat du sprint courant, dans la partie qui suit, à travers des imprimés écran.

1. Interface de connexion

La Figure 20 expose l'interface de connexion de notre plateforme.

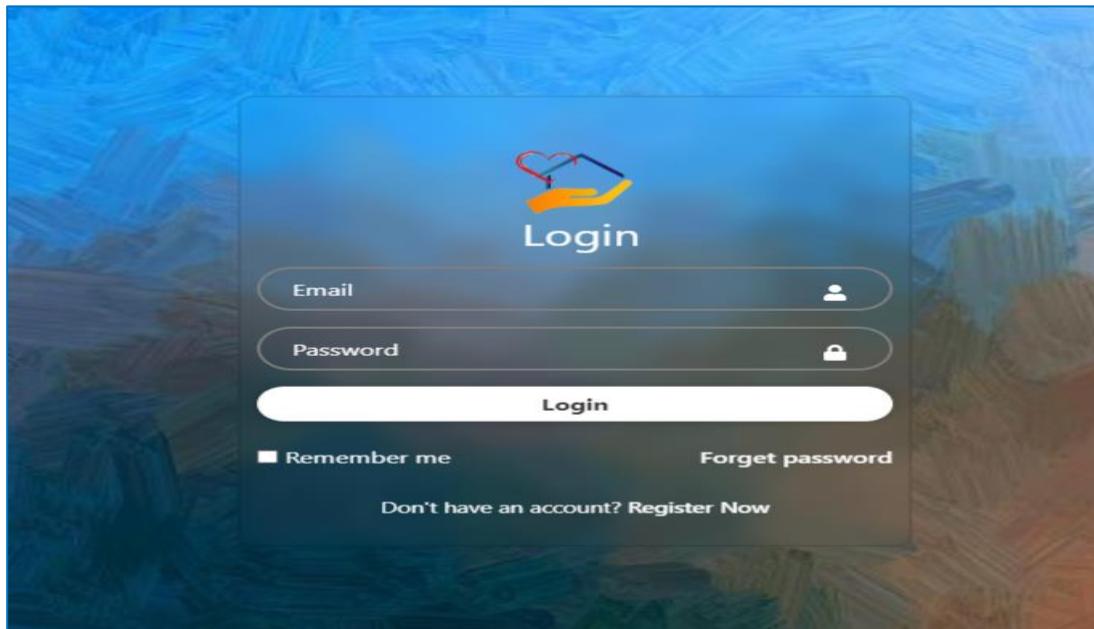


Figure 20: Interface connexion

2. Interface inscription client

Pour ceux qui n'ont pas encore de compte, ils peuvent s'inscrire soit comme client en utilisant l'interface décrite dans la Figure 21.

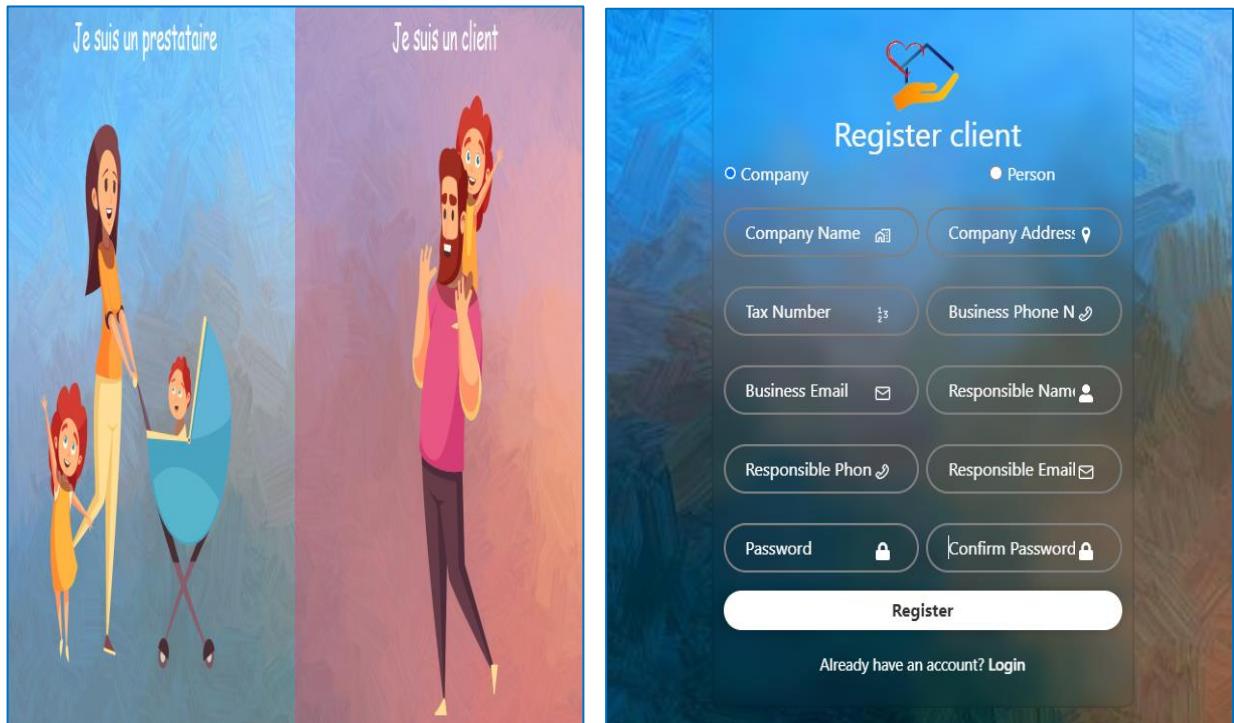


Figure 21: Interface inscription client

3. Interface inscription prestataire

La Figure 20 présente l'interface d'inscription pour les prestataires.

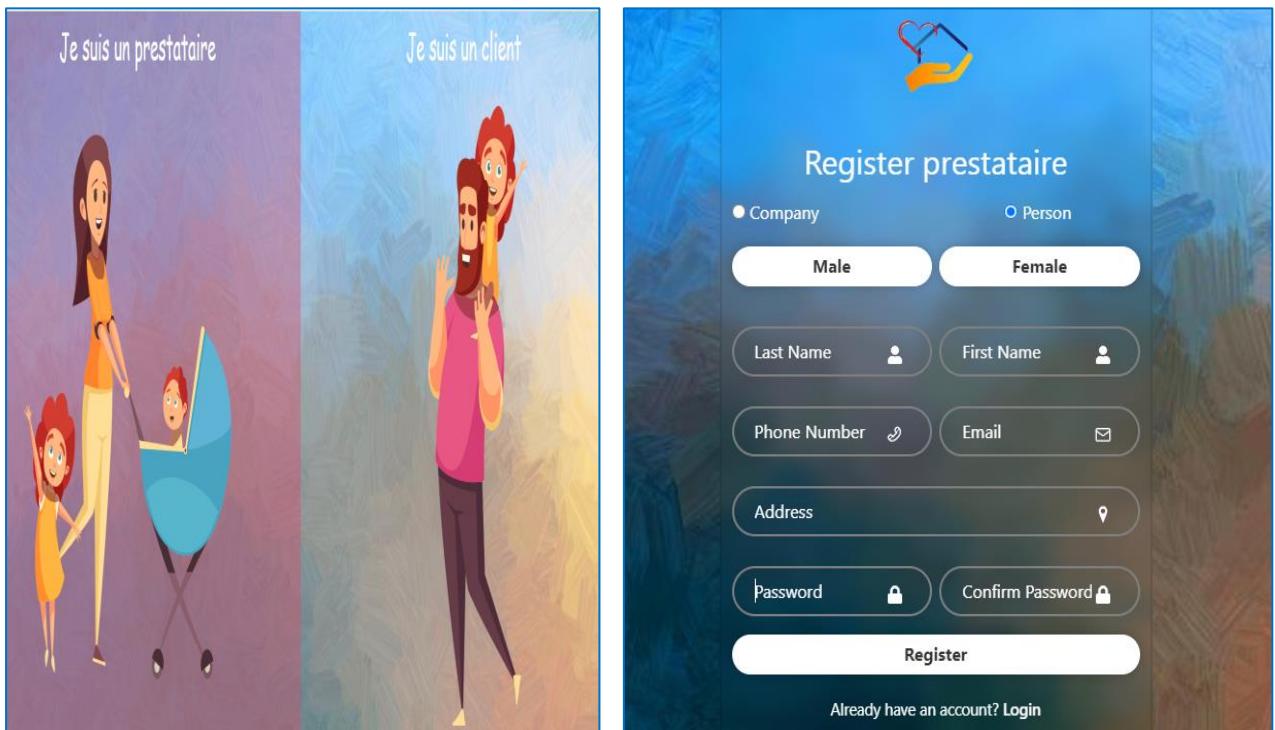


Figure 22: Interface inscription prestataire

4. Interfaces relatives à l'administrateur

1. Gestion des clients

La figure 23 représente l'interface de gestion des clients, grâce à laquelle l'administrateur pourra avertir ou supprimer un client.

Nom et Prénom	Email	Téléphone	Type d'utilisateur	Actions
ben foulen foulene	foulene@gmail.com	99876451	person	
sghaier mejdaa	mejda@gmail.com	21949765	person	

Figure 23: Interface de gestion des clients

2. Gestion des prestataires

La figure 23 représente l'interface de gestion des prestataires, grâce à laquelle l'administrateur pourra avertir ou supprimer un prestataire.

Nom et Prénom	Email	Téléphone	Type d'utilisateur	Actions
salha ben saleh	n@gmail.com	22333666	person	

Figure 24: Interface de gestion des prestataires

5. Interface mise à jour profile

La Figure 25 illustre l'interface de gestion du profil, où l'utilisateur peut modifier ses informations personnelles.

Figure 25: Interface de mis à jour profil

Conclusion

Dans ce chapitre dédié à la gestion des utilisateurs et de leurs comptes, nous avons réussi à créer un incrément avec une valeur ajoutée significative. Nous avons d'abord présenté une étude fonctionnelle, suivie d'une étude conceptuelle incluant les diagrammes de classe et de séquence ainsi que la modélisation graphique. Enfin, nous avons capturé les interfaces de notre projet.

Dans le prochain chapitre, nous nous concentrerons sur la création d'une nouvelle valeur métier en développant les fonctionnalités de gestion des services.

Chapitre 4 : Sprint 2 : Gestion des services

Introduction

Ce chapitre vise à introduire la deuxième itération du cycle de vie de notre projet, consacrée à la gestion des services. Nous commencerons par définir les tâches à accomplir dans le backlog du sprint, avant de passer à la phase de conception. Enfin, nous décrirons la phase de mise en œuvre de ce module.

I. Étude fonctionnelle

1. Backlog du sprint

Le tableau ci-dessous présente le Backlog du sprint, qui répertorie les travaux à accomplir pour chaque user story pendant ce sprint.

ID	User Story	Description	Estim.
3.1	<i>En tant qu'administrateur, je souhaite ajouter un service.</i>	Créer le modal de l'ajout d'un service.	2
		Créer l'API de l'ajout d'un service.	1
		Consommer l'API.	1
3.2	En tant qu'administrateur, je souhaite modifier un service.	Créer le modal de modification d'un service.	2
		Créer l'API de modification d'un service.	1
		Consommer l'API.	1
3.3	En tant qu'utilisateur, je souhaite consulter la liste des services.	Créer la vue de la liste des services.	2
		Consommer l'API de sélection des services.	1

Tableau 7: Backlog du sprint 2

2. Diagramme de cas d'utilisation du sprint 2

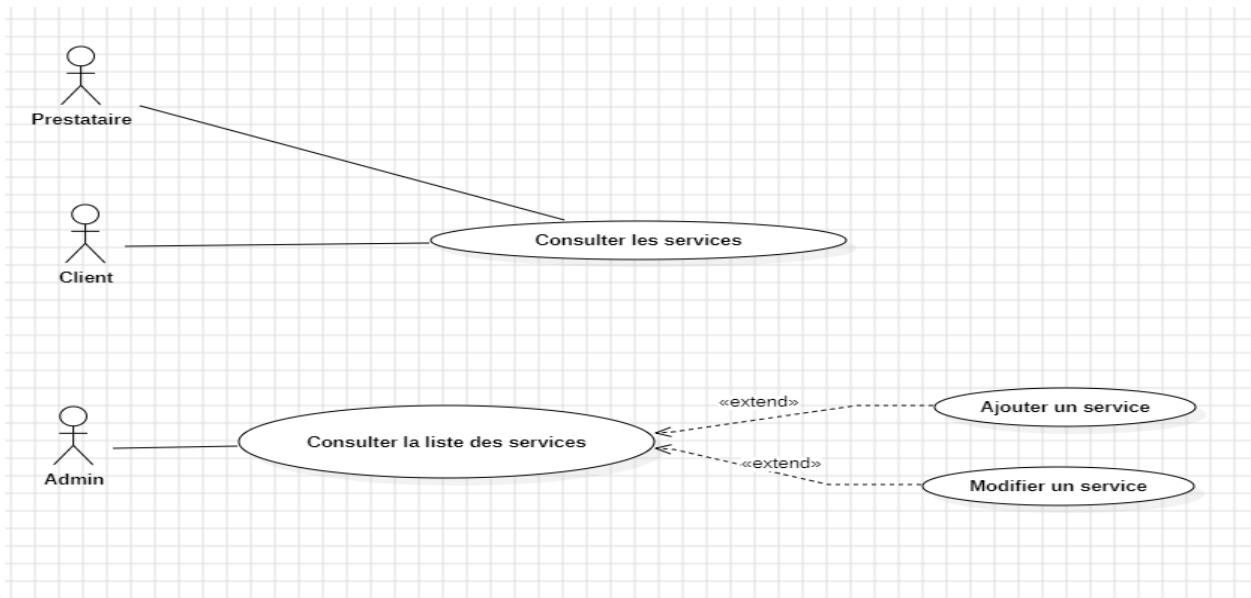


Figure 26: Diagramme de cas d'utilisation sprint 2

Ce sprint offre une gamme étendue de fonctionnalités aux divers utilisateurs de notre système. En premier lieu, l'administrateur joue un rôle central en ayant la capacité de consulter, ajouter ou modifier un service. Cette souplesse lui permet de maintenir une liste de services constamment à jour, en accord avec les besoins et les évolutions du système.

De plus, les clients et les prestataires bénéficient de la facilité de consulter la liste des services depuis la page d'accueil. Cela leur permet d'explorer les différentes options disponibles et de prendre des décisions éclairées en fonction de leurs besoins spécifiques.

Pour mieux comprendre les cas d'utilisation de ce sprint, nous allons fournir, dans cette section, une description textuelle cas d'utilisation "Ajouter/Supprimer un service".

- Description textuelle du cas d'utilisation : "Ajouter un service"**

Acteur	Administrateur
Description	Ajouter un nouveau service au système.
Préconditions	Administrateur authentifié.
Post-conditions	Service ajouté.
Scénario principal	L'administrateur remplit les champs relatifs à l'ajout d'un nouveau service puis valide.
	Le système vérifie les données.
	Le système ajoute le service.
	Le système affiche la liste des services mis à jour.
Scénario alternatif	Le système retourne un message d'erreur.

Tableau 8: Description textuelle du cas d'utilisation "ajouter un service"

- **Description textuelle du cas d'utilisation : "Modifier un service "**

Acteur	Administrateur
Description	Modifier un service.
Préconditions	Administrateur authentifié.
Post-conditions	Service modifié.
Scénario principal	L'administrateur click sur le bouton modifier.
	Un modal s'affichera.
	L'administrateur remplit le modèle et confirme.
	Le système met à jour le service.
	Le système affiche la liste des services mis à jour.
Scénario alternatif	L'administrateur click sur le bouton annuler.
	La liste des services s'affichera sans modification.

Tableau 9: Description textuelle du cas d'utilisation "modifier un service"

II. Étude Conceptuelle

1. Diagramme de classe du sprint 2

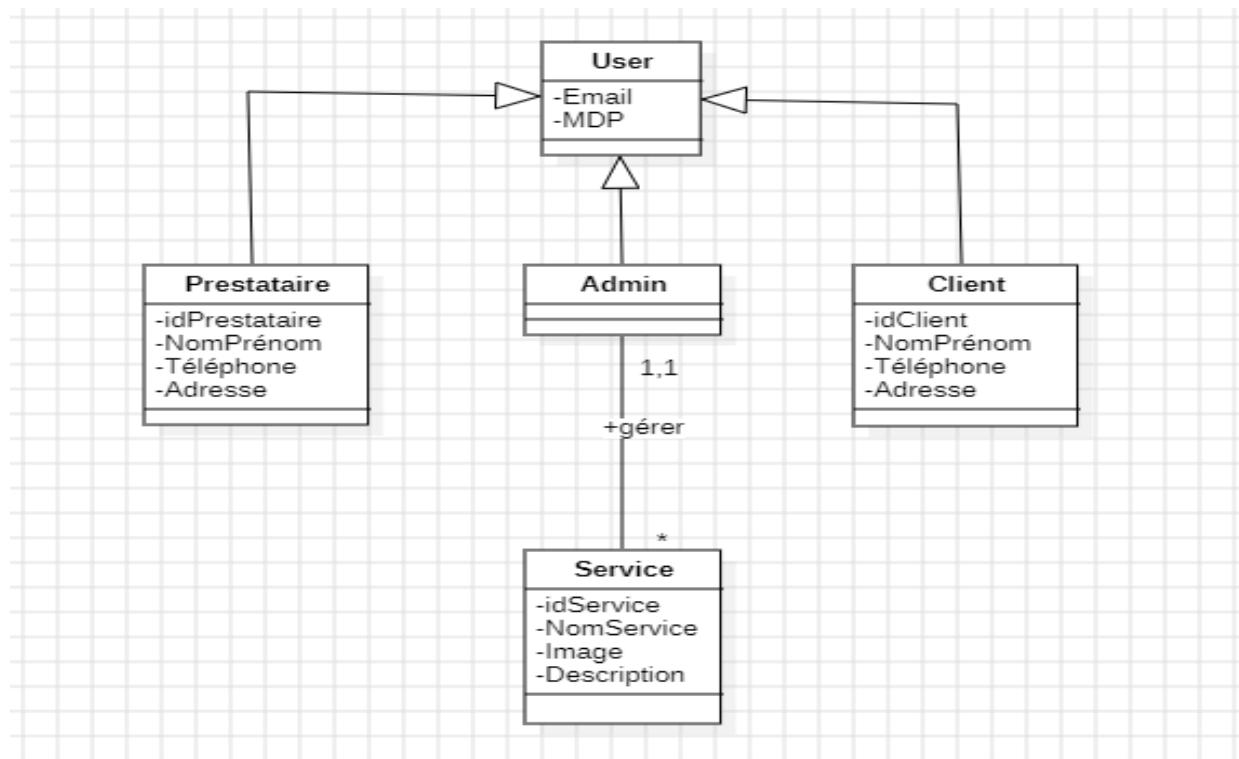


Figure 27: Diagramme de classe sprint 2

2. Diagramme de séquence du sprint 2

Dans cette partie, nous allons modéliser la façon de mettre à jour d'un service à travers le diagramme de séquence « Mettre à jour un service »

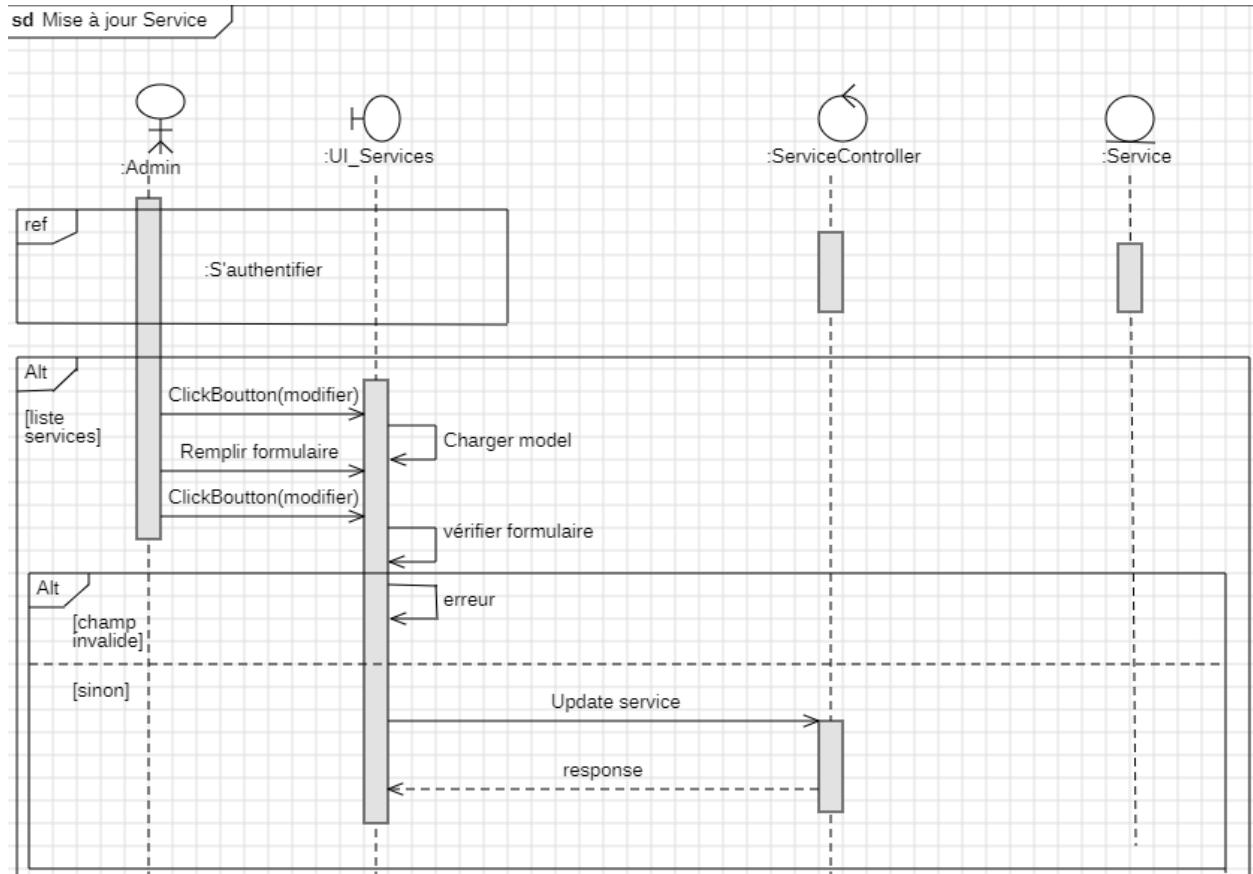


Figure 28: Diagramme de séquence sprint 2

Le diagramme de la figure 28 décrit l'opération de modification d'un service. Tout d'abord, l'administrateur choisit un service à modifier depuis la liste des services et appuis sur modifier qui déclenche un modal continent les informations du service, après avoir modifié le formulaire, l'administrateur envoie les données au serveur. Une fois le formulaire validé, les données seront mises à jour.

3. Réalisation

Passons maintenant à la réalisation, où nous allons capturer et concevoir les interfaces clés dans le cadre de ce deuxième sprint.

1. Interface gestion des services

La figure 29 présente l'interface de gestion des services.

The screenshot shows the 'HOME CARE' service management interface. On the left, a sidebar menu includes 'Accueil', 'Prestataire', 'Client', 'Services', 'Liste Annonces', 'Liste Demandes', 'Messages', and 'Nous Contacter'. The main area is titled 'Admin' and features a button '+ Ajouter un service'. A table lists existing services:

Service	Image	Description	Actions
babysitting		aaaaaaaaaa	<input type="checkbox"/>

Figure 29: Interface de gestion des services

2. Interface d'ajout d'un service

Pour ajouter un service, l'administrateur doit remplir un formulaire en entrant le nom, l'image et la description du service, comme illustré dans la figure 30.

The screenshot shows the 'Ajouter un service' (Add service) dialog box. It contains fields for 'Nom du service' (Service name), 'Image du service' (Service image) with a file selection button ('Choisir un fichier') and a message ('Aucun fichier n...été sélectionné'), and 'Description du service' (Service description). At the bottom are 'Ajouter' (Add) and 'Fermer' (Close) buttons. In the background, the main service management interface is visible, showing a list of services with columns 'Description' and 'Actions'.

Figure 30: Interface d'ajout d'un service

3. Interface de modification d'un service

En cas de modification, une fenêtre modale affichera les informations du service, permettant à l'administrateur de les modifier, comme montré dans la figure 31.

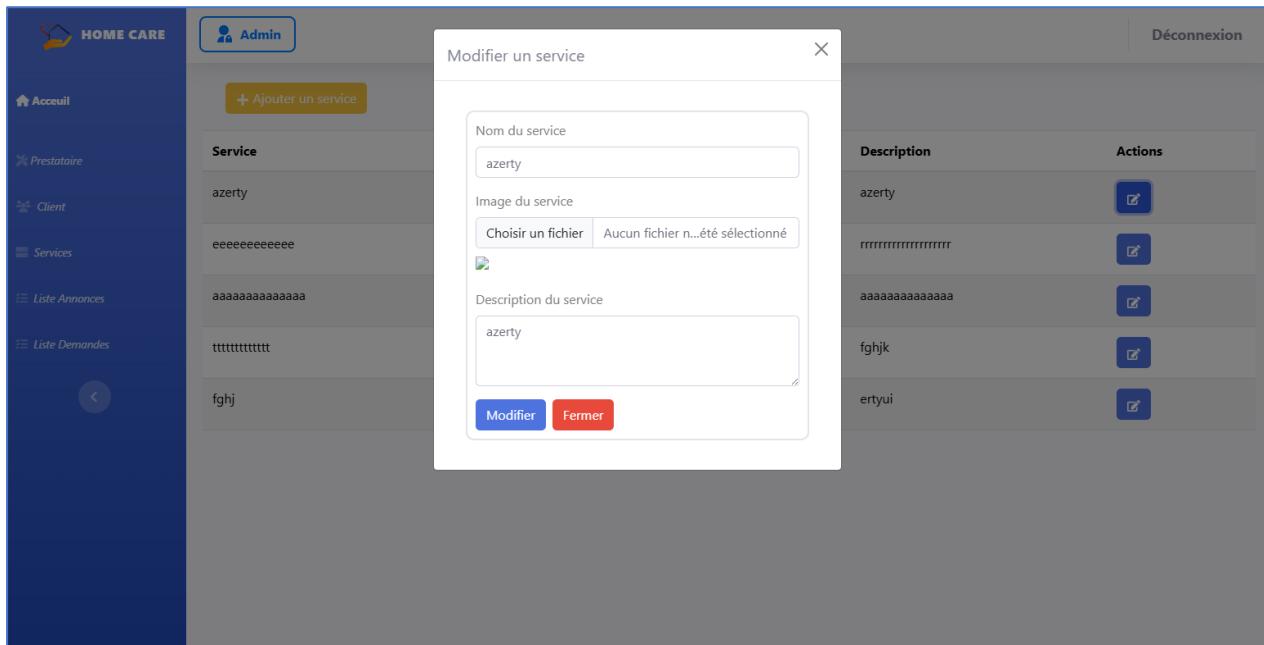


Figure 31: Interface de modification d'un service

4. Interface des services

La figure 32 montre l'interface des services disponibles sur notre plateforme, que les clients et les prestataires peuvent consulter.

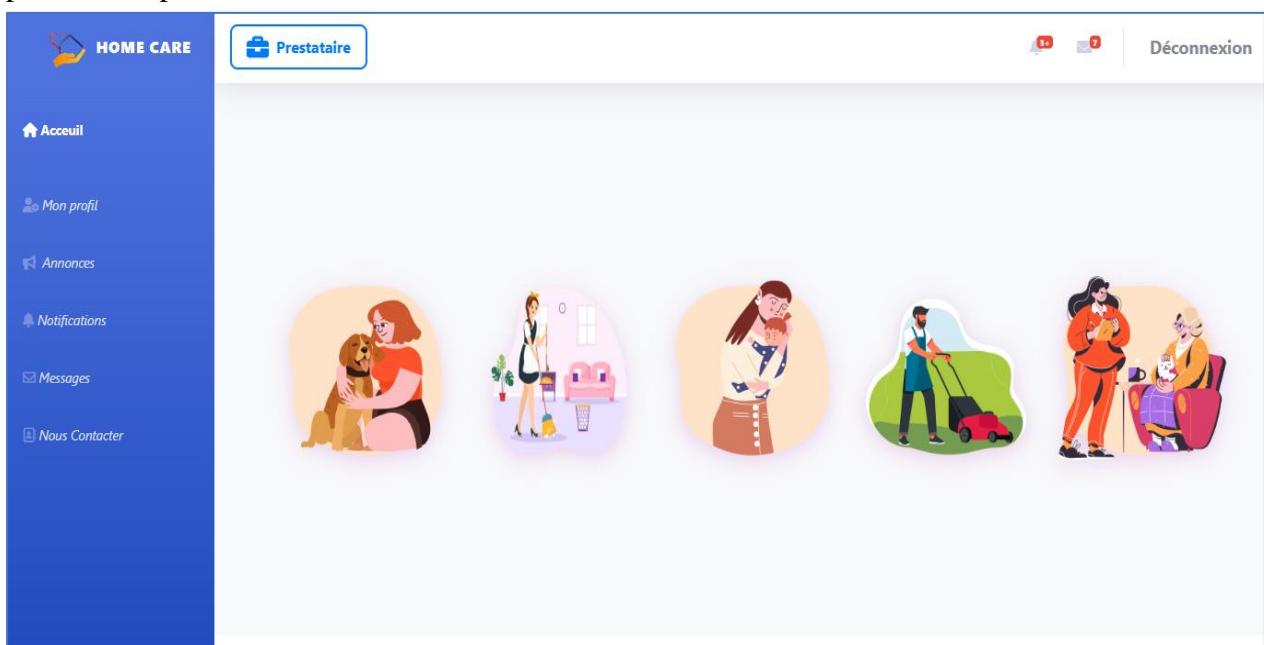


Figure 32: Interface des services

Conclusion

Lors de ce sprint, nous avons précisé les fonctionnalités essentielles de la gestion des services.

Dans le prochain chapitre, nous appliquerons cette même méthodologie pour traiter l'aspect le plus crucial de notre plateforme : la gestion des annonces et demandes.

Chapitre 5 : Sprint 3 : Gestion des annonces et demandes

Introduction

Dans ce chapitre, nous nous concentrerons sur le développement des fonctionnalités du troisième sprint, dédié à la gestion des annonces et des demandes, éléments clés de notre projet. Nous commencerons par identifier les tâches à inclure dans le backlog du sprint, puis nous passerons aux phases de conception. Enfin, nous décrirons la phase de réalisation, en capturant les interfaces principales de ce module.

I. Étude fonctionnelle

1. Backlog sprint

ID	User Story	Description	Estim.
4.1	<i>En tant que client, je souhaite consulter la liste des annonces.</i>	Créer l'API de sélection des annonces.	2
		Consommer l'API.	1
4.2	En tant que client, je souhaite ajouter une annonce.	Créer le modal de l'ajout d'une annonce.	1
		Créer l'API de l'ajout d'une annonce.	1
		Consommer l'API.	1
4.3	En tant que client, je souhaite modifier une annonce.	Créer le modal de la modification d'une annonce.	2
		Créer l'API de la modification d'une annonce.	2
		Consommer l'API.	1
4.4	En tant que client, je souhaite archiver une annonce.	Créer le modal de l'archivage d'une annonce.	2
		Créer l'API de l'archivage d'une annonce.	2
		Consommer l'API.	1
4.5	En tant que client, je souhaite supprimer une annonce.	Créer le modal de la suppression d'une annonce.	2
		Créer l'API de la suppression d'une annonce.	2
		Consommer l'API.	1

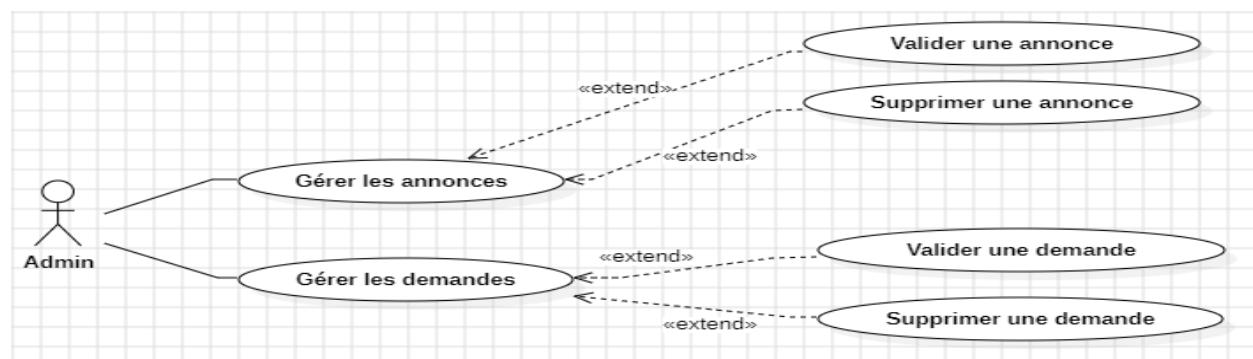
5.1	En tant que prestataire, je souhaite consulter la liste des demandes.	Créer l'API de sélection des demandes.	2
		Consommer l'API.	1
5.2	En tant que prestataire, je souhaite ajouter une demande de travail.	Créer le modal de l'ajout d'une demande.	2
		Créer l'API de l'ajout d'une demande.	2
		Consommer l'API.	1
5.3	En tant que prestataire, je souhaite modifier une demande de travail.	Créer le modal de la modification d'une demande.	2
		Créer l'API de la modification d'une demande.	2
		Consommer l'API.	1
5.4	En tant que prestataire, je souhaite archiver une demande de travail.	Créer le modal de l'archivage d'une demande.	2
		Créer l'API de l'archivage d'une demande.	2
		Consommer l'API.	1
5.5	En tant que prestataire, je souhaite supprimer une demande de travail.	Créer le modal de la suppression d'une demande.	2
		Créer l'API de la suppression d'une demande.	2
		Consommer l'API.	1

Tableau 10: Backlog du sprint 3

2. Diagramme de cas d'utilisation

Le diagramme de cas d'utilisation exposé par la figure 32 a pour but d'établir les fonctionnalités rattachées à la troisième valeur métier.

Tous les cas d'utilisation de ce sprint sont précédés par une opération d'authentification.



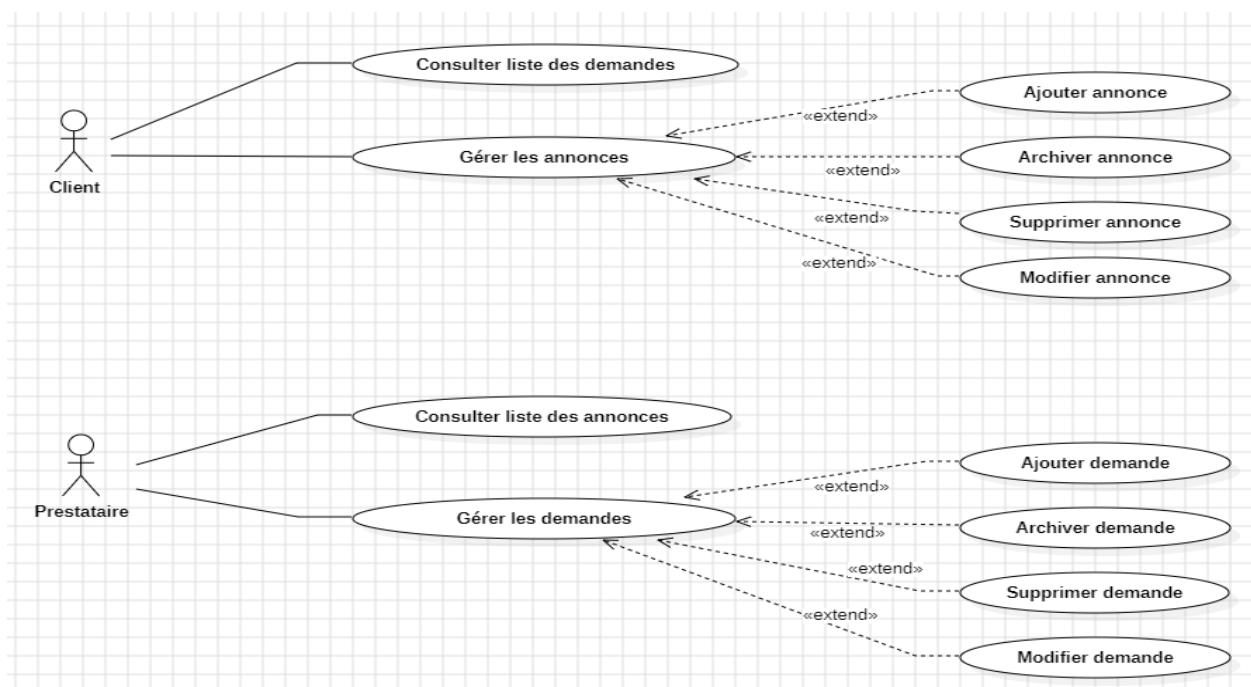


Figure 33: Diagramme de cas d'utilisation sprint3

- Description textuelle du cas d'utilisation : " ajouter une annonce "**

Acteur	Client
Description	Ajouter une annonce.
Préconditions	Client authentifié.
Post-conditions	Annonce publiée.
Scénario principal	Le client remplit le formulaire. Le système ajoute une annonce. Le système affiche la liste des annonces mis à jour.
Scénario alternatif	Le système retourne un message d'erreur.

Tableau 11: Description textuelle du cas d'utilisation "ajouter annonce"

- Description textuelle du cas d'utilisation : " Archiver une annonce "**

Acteur	Client
Description	Archiver une annonce.
Préconditions	Client authentifié.
Post-conditions	Annonce archivée.
Scénario principal	Le client click sur le bouton archiver. Un toast de confirmation s'affichera.

	Le système archive l'annonce.
	Le système affiche la liste des annonces mis à jour.
Scénario alternatif	Le système retourne un message d'erreur.

Tableau 12: Description textuelle du cas d'utilisation "archiver une annonce"

- **Description textuelle du cas d'utilisation : " ajouter une demande "**

Acteur	Prestataire
Description	Ajouter une demande.
Préconditions	Prestataire authentifié.
Post-conditions	Demande ajoutée.
Scénario principal	Le prestataire remplit le formulaire.
	Le système ajoute une demande.
	Le système affiche la liste des demandes mis à jour.
Scénario alternatif	Le système retourne un message d'erreur.

Tableau 13: Description textuelle du cas d'utilisation "ajouter une demande"

- **Description textuelle du cas d'utilisation : " supprimer une demande "**

Acteur	Prestataire
Description	Supprimer une demande.
Préconditions	Prestataire authentifié.
Post-conditions	Demande supprimée.
Scénario principal	Le prestataire click sur le bouton supprimer.
	Un toast de confirmation s'affichera.
	Le système supprime la demande.
	Le système affiche la liste des demandes mis à jour.
Scénario alternatif	Le système retourne un message d'erreur.

Tableau 14: Description textuelle du cas d'utilisation "supprimer une demande"

II. Etude conceptuelle

1. Diagramme de classe du sprint 3

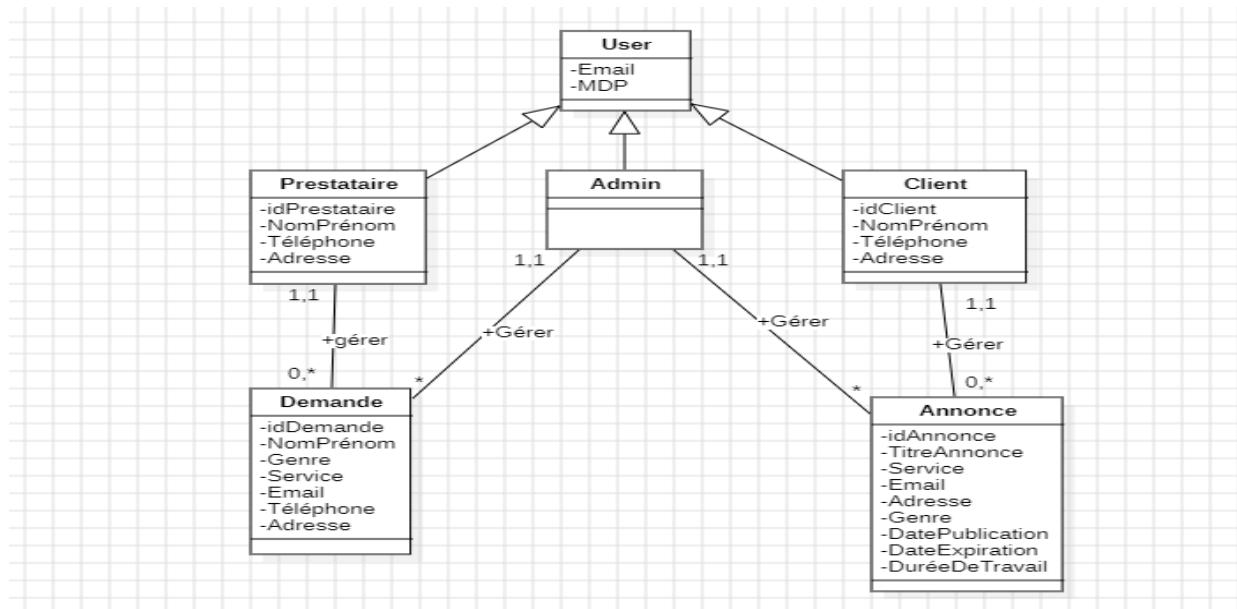


Figure 34: Diagramme de classe sprint3

2. Diagramme de séquence du sprint 3

Dans cette section, nous allons modéliser le processus de modification d'une annonce à l'aide du diagramme de séquence intitulé « Modifier annonce ».

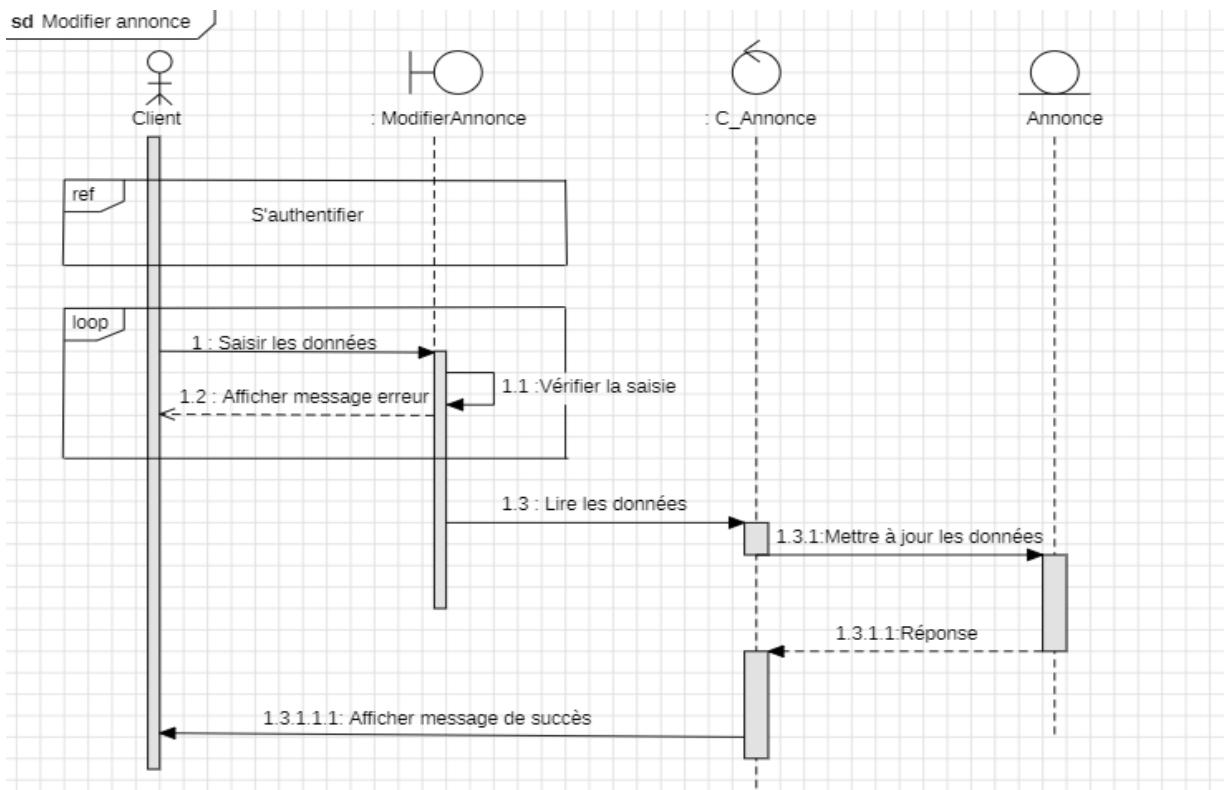


Figure 35: Diagramme de séquence sprint 3

III. Réalisation

Passons maintenant à la phase de réalisation, où nous allons capturer et concevoir les interfaces principales dans le cadre de ce troisième sprint.

1. Interfaces relatives aux prestataires

Pour ajouter une demande, le prestataire doit remplir un formulaire en saisissant le nom, le prénom, l'email, le téléphone, et l'adresse, puis sélectionner le genre et le service, comme illustré dans la figure 36.

Figure 36: Formulaire demande d'emploi

La figure 37 montre l'interface des annonces des clients, où le prestataire peut rechercher une annonce par localisation, genre ou date spécifique.

Figure 37: Annonce

La figure 38 illustre l'interface de gestion des demandes, où le prestataire peut modifier, supprimer ou archiver une demande.

The screenshot shows the 'Prestataire' (Provider) section of the 'HOME CARE' application. On the left is a sidebar with navigation links: Accueil, Mon profil, Demande, Mes Demandes, Notifications, Messages, and Nous Contacter. The main area is titled 'Prestataire' and displays a table of requests:

Nom et Prénom	Service	Adresse	Détails	Actions
sonia borji	Femme de ménage	rue hannibal zaouiet kontech	Voir plus	
foulen foulen	Babysitting	moknin	Voir plus	
sonia borji	Papysitting	rue hannibal zaouiet kontech	Voir plus	

Figure 38: Interface gestion des demandes

2. Interfaces relatives aux clients

Pour publier une annonce, le client doit remplir un formulaire en entrant le titre de l'annonce, son adresse email, la durée du travail et l'adresse. Ensuite, il doit sélectionner le genre, le service, la date de publication et la date d'expiration, comme illustré dans la figure 39.

The screenshot shows the 'Client' section of the 'HOME CARE' application. On the left is a sidebar with navigation links: Accueil, Mon profil, Annonces, Mes Annonces, Notifications, Messages, and Nous Contacter. The main area is titled 'Client' and displays a form titled 'Créer une annonce':

Créer une annonce

Titre de l'annonce

Service

Email

Adresse

Date de publication Genre

Date d'expiration Duree de Travail

Publier votre annonce

Figure 39: Formulaire annonce

La figure 40 montre l'interface des demandes d'emploi publiées par les prestataires.

The screenshot shows the 'Demandes pour Pet-Sitting' (Pet-Sitting Requests) section. On the left, there's a sidebar with navigation links: Accueil, Mon profil, Annonces, Mes Annonces, Notifications, Messages, and Nous Contacter. The main area displays a request from a user named 'salah salah'. The request details are as follows:

- Adresse : sahline
- Email : salahbensalah@gmail.com
- Téléphone : 55058987
- Publier le : 18/05/2024

At the bottom of the request card are two buttons: 'Contacter' (Contact) and 'Intéressé' (Interested).

Figure 40 : Demande d'emploi

La figure 41 illustre l'interface de gestion des annonces, où le client peut modifier, supprimer ou archiver une annonce.

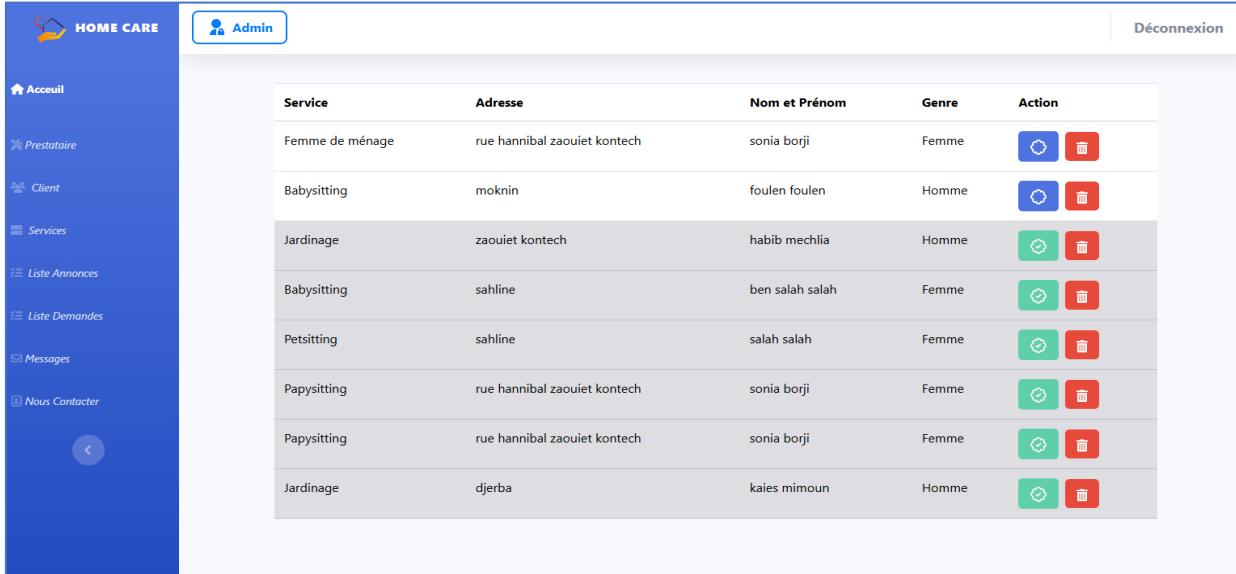
The screenshot shows the 'Interface gestion des annonces' (Advertisement Management Interface). The left sidebar is identical to Figure 40. The main area displays a list of advertisements:

Titre	Service	Adresse	Détail	Action
Je besoins de petsitting	Petsitting	jerjis	Voir plus	
baby sitting	Babysitting	moknin	Voir plus	

Figure 41: Interface gestion des annonces

3. Interfaces relatives à l'administrateur

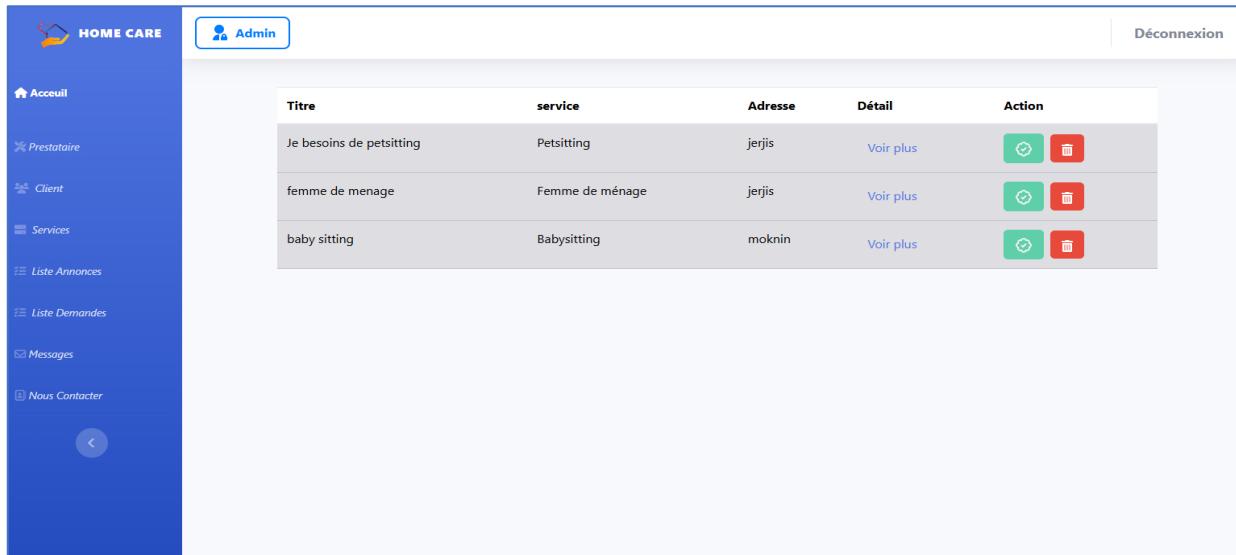
La figure 42 illustre l'interface de gestion des demandes, où l'administrateur peut valider la demande publiée par le prestataire ou la supprimer.



Service	Adresse	Nom et Prénom	Genre	Action
Femme de ménage	rue hannibal zaouiet kontech	sonia borji	Femme	
Babysitting	moknin	foulen foulen	Homme	
Jardinage	zaouiet kontech	habib mechlia	Homme	
Babysitting	sahline	ben salah salah	Femme	
Petsitting	sahline	salah salah	Femme	
Papysitting	rue hannibal zaouiet kontech	sonia borji	Femme	
Papysitting	rue hannibal zaouiet kontech	sonia borji	Femme	
Jardinage	djerba	kaies mimoun	Homme	

Figure 42: Interface admin de gestion des demandes

La figure 43 illustre l'interface de gestion des annonces, où l'administrateur peut valider ou supprimer une annonce publiée par un client.



Titre	service	Adresse	Détail	Action
Je besoins de petsitting	Petsitting	jerjis	Voir plus	
femme de menage	Femme de ménage	jerjis	Voir plus	
baby sitting	Babysitting	moknin	Voir plus	

Figure 43: Interface admin de gestion des annonces

Conclusion

Dans le chapitre dédié au sprint 3, nous avons concentré nos efforts sur la gestion des annonces et des demandes. Ce sprint a permis d'instaurer une gestion efficace, améliorant ainsi l'expérience des clients et des prestataires.

Chapitre 6 : sprint 4 : Gestion des messages et notifications

Introduction

Dans ce chapitre consacré à la gestion des messages et notifications, nous commencerons par détailler les tâches à accomplir dans le Backlog du sprint, puis nous aborderons les phases de conception. Enfin, nous explorerons la mise en œuvre de ce module.

I. Etude fonctionnelle

1. Backlog du sprint

Le tableau ci-dessous présente le Backlog du sprint, répertoriant les tâches à accomplir pour chaque user story au cours de ce sprint.

ID	User story	Description	Estim.
6.1	En tant que prestataire, je souhaite postuler à une annonce.	Développer la vue de postulation.	2
		Générer l'API.	2
		Consommer l'API.	1
6.2	En tant que client, je souhaite m'intéresser à une demande.	Développer la vue d'intérêt.	2
		Générer l'API.	2
		Consommer l'API.	1
7.1	En tant que client, je souhaite consulter la liste des messages.	Développer la vue de consultation des messages.	2
		Générer l'API permettant de retourner la liste des messages.	2
		Consommer l'API.	1
7.2	En tant que client, je souhaite envoyer un message.	Créer le modal d'envoi d'un message.	2
		Créer l'API d'envoi d'un message.	2
		Consommer l'API.	1
7.3	En tant que prestataire, je souhaite consulter la liste des messages.	Développer la vue de consultation des messages.	2
		Générer l'API permettant de retourner la liste des messages.	2
		Consommer l'API.	1

7.4	En tant que client, je souhaite envoyer un message.	Créer le modal d'envoi d'un message.	2
		Créer l'API d'envoi d'un message.	2
		Consommer l'API.	1

Tableau 15: Backlog du sprint 4

2. Diagramme de cas d'utilisation du sprint 4

Le diagramme de cas d'utilisation exposé par la figure 44 a pour but d'établir les fonctionnalités rattachées à la quatrième valeur métier.

Tous les cas d'utilisation de ce sprint sont précédés par une opération d'authentification.

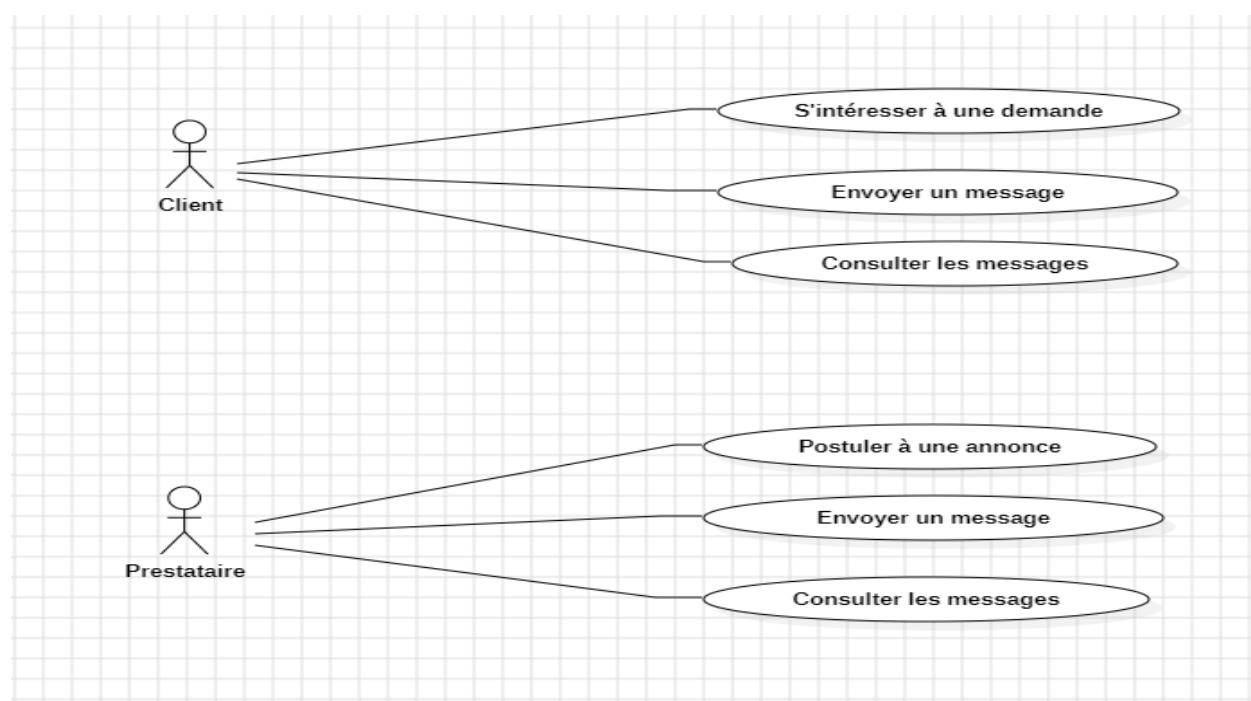


Figure 44: Diagramme de cas d'utilisation sprint 4

- **Description textuelle du cas d'utilisation : " Envoyer un message "**

Acteur	Client
Description	Envoyer un message.
Préconditions	Client authentifié.
Post-conditions	Message envoyé.
Scénario principal	Le client click sur le bouton contacter. Un chat avec le prestataire s'ouvre. Le client écrit son message. Le client clique sur le bouton envoyer.

Scénario alternatif

Le système retourne un message d'erreur.

Tableau 16: Description textuelle du cas d'utilisation "Envoyer un message"

II. Etude conceptuelle

1. Diagramme de classe du sprint 4

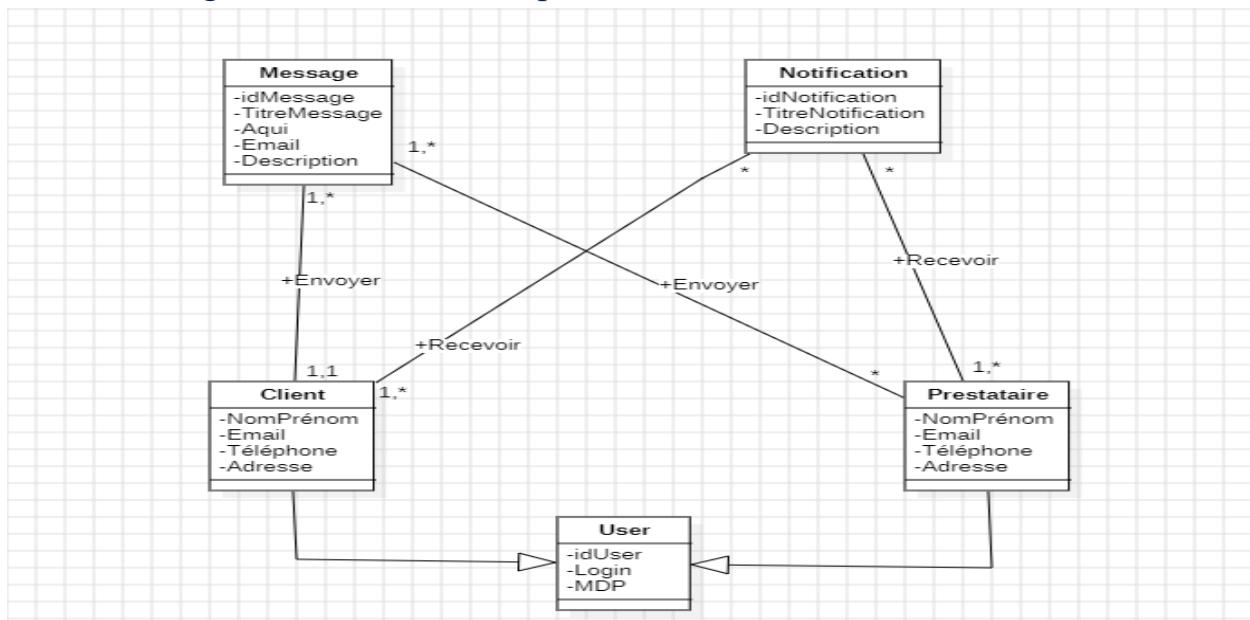


Figure 45 : Diagramme de classe sprint 4

2. Diagramme de séquence du sprint 4

Dans cette section, nous allons utiliser le diagramme de séquence intitulé « envoyer message » pour modéliser le processus d'ajout d'un message.

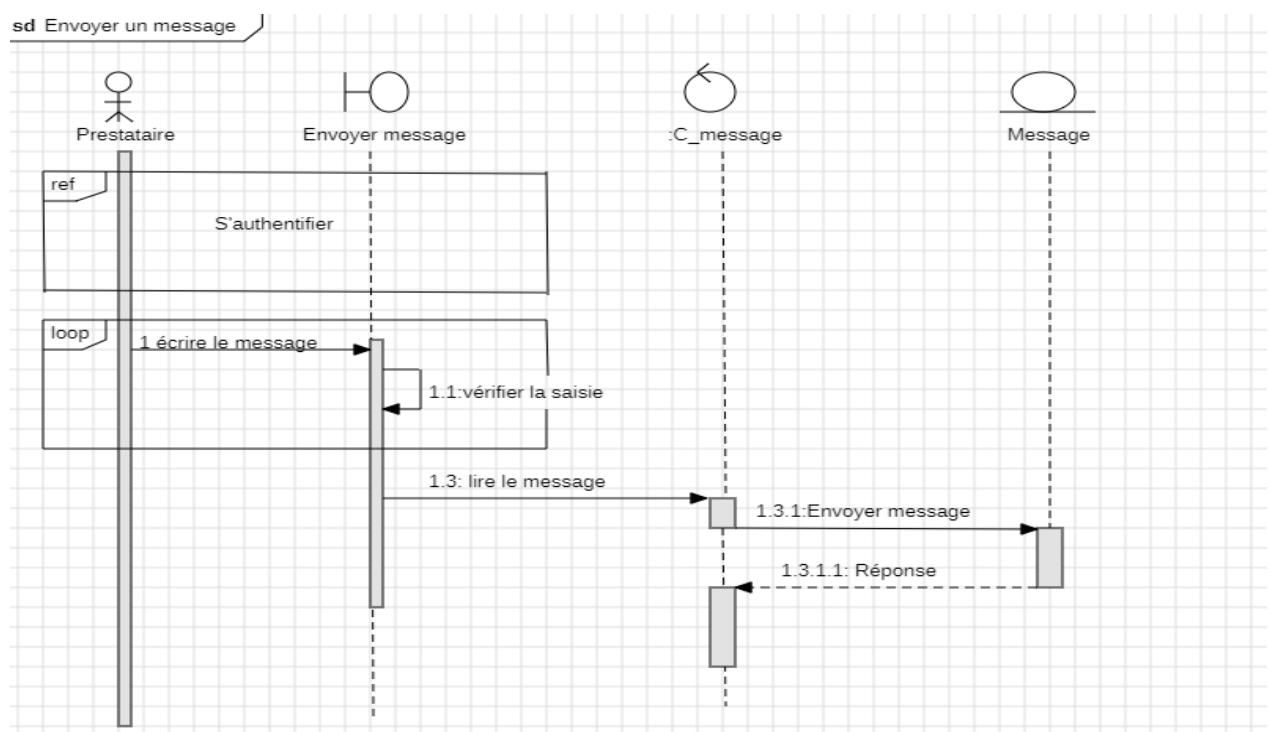


Figure 46: Diagramme de séquence sprint 4

III. Réalisation

Passons maintenant à la phase de réalisation, au cours de laquelle nous allons capturer et concevoir les interfaces principales pour ce quatrième sprint.

1. Interfaces relatives aux prestataires

La figure 47 présente l'interface des notifications, où le prestataire peut voir les clients intéressés par sa demande d'emploi. Il a également la possibilité de marquer les notifications comme lues ou non lues.

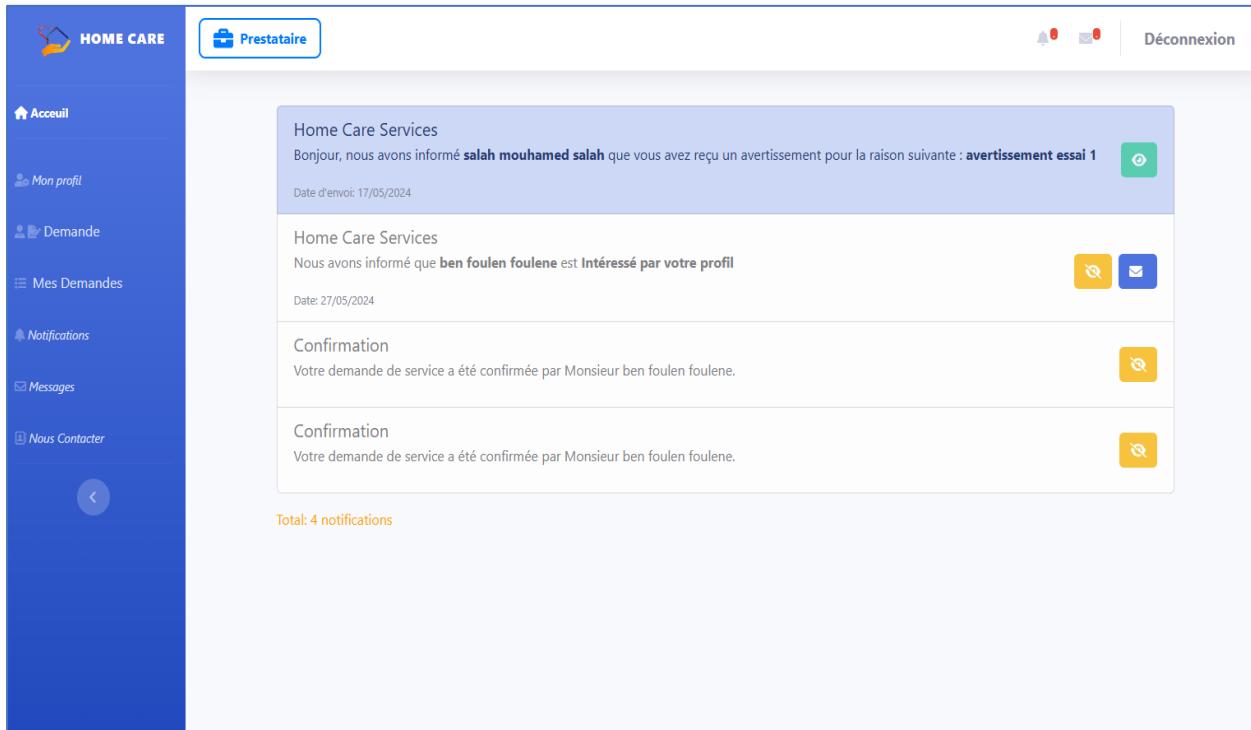


Figure 47: Interface de notifications pour les prestataires

Lorsque le prestataire reçoit une notification indiquant qu'un client est intéressé par son offre, il peut communiquer avec ce client ou répondre à son message en cliquant sur le bouton message, comme illustré dans la figure ci-dessous.

Chapitre 6 : Sprint 4: Gestion des messages et notifications

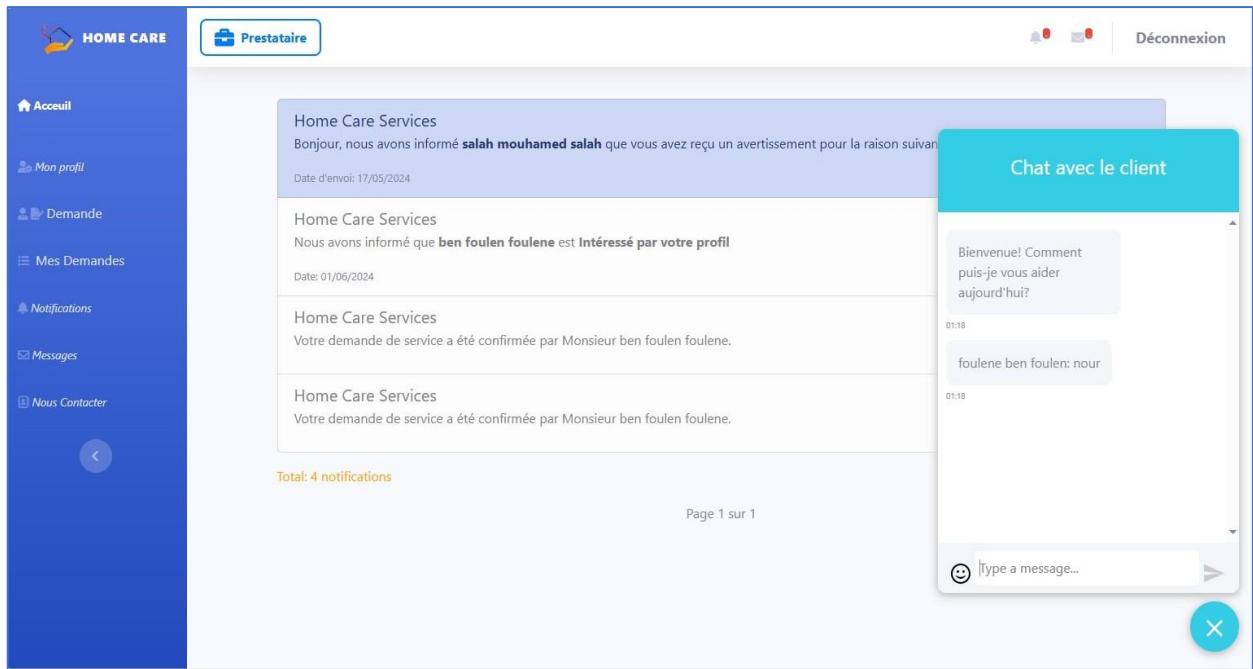


Figure 48: Interface de chat avec le client

2. Interfaces relatives aux clients

La figure 49 montre l'interface des notifications, où le client peut voir les prestataires ayant postulé à son annonce et valider leur candidature en cliquant sur le bouton valider.

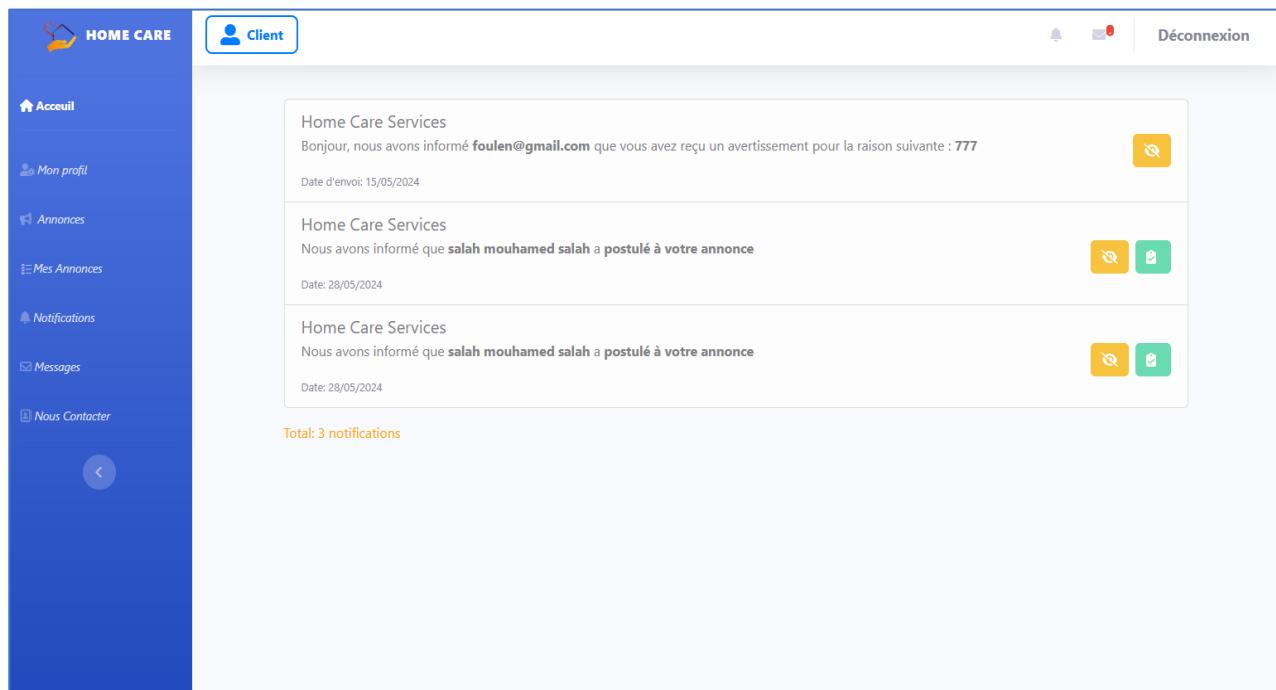


Figure 49: Interface de notifications pour les clients

Si le client est intéressé par une candidature, il peut également contacter le prestataire comme le montre la figure 50.

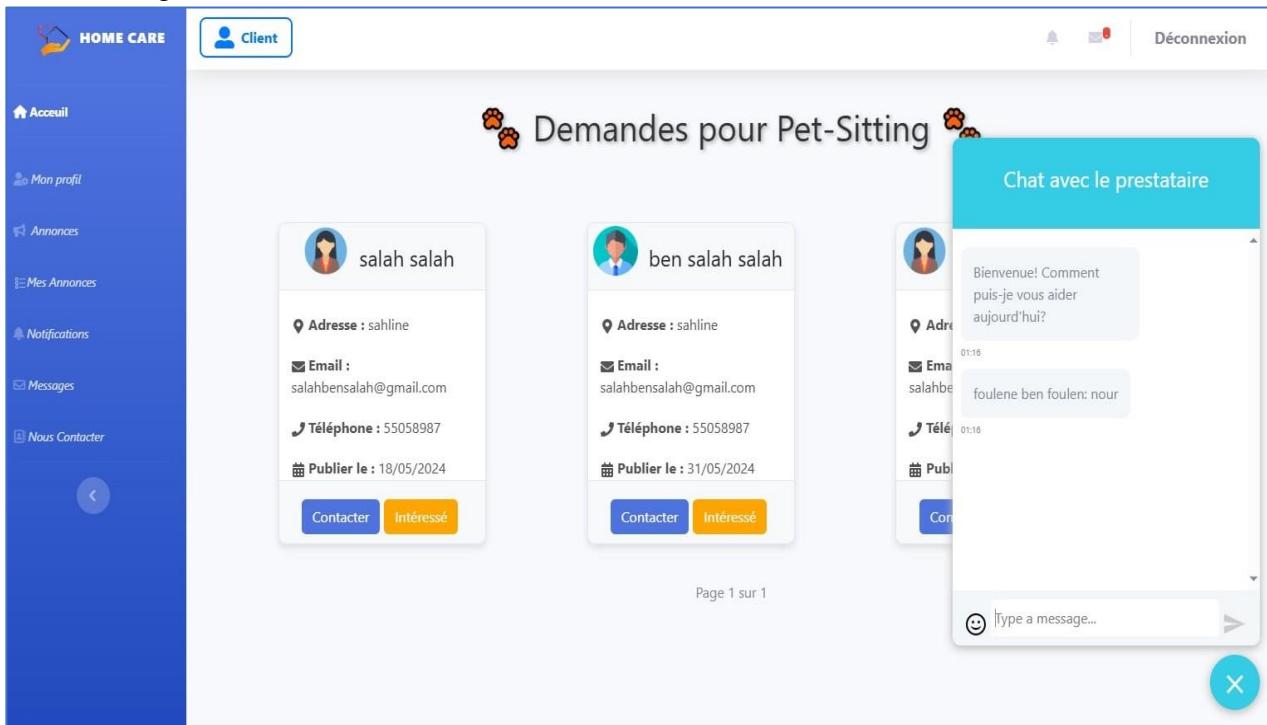


Figure 50: Interface de chat avec le prestataire

3. Caractéristiques techniques

3.1 Infographie

- **Logo de la plateforme**

Pour le logo, nous avons sélectionné une image trouvée sur Google et apporté quelques modifications pour qu'elle corresponde à nos attentes.



Figure 51: Logo de la plateforme

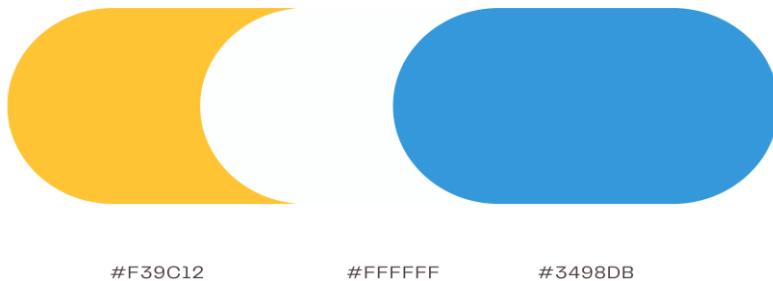
- **Feuille de style**

Nous avons choisi un template bootstrap [13] comme base pour notre projet. Cependant, afin de mieux répondre à nos exigences spécifiques, nous avons apporté plusieurs modifications. Le header a été ajusté pour refléter notre identité visuelle unique, la sidebar a été réorganisée pour une navigation plus intuitive, et le footer a été personnalisé pour inclure des informations et des liens pertinents à notre plateforme.

Ces ajustements ont permis de créer une interface qui est non seulement fonctionnelle, mais aussi distinctement adaptée à nos besoins.

- Choix des couleurs

Pour notre plateforme, nous avons choisi une palette de couleurs dominée par le bleu et le blanc, en harmonie avec notre logo. Pour ajouter du contraste et rendre l'ensemble plus dynamique, nous avons intégré des touches d'orange, de rouge et de vert pour les boutons. Cette combinaison crée une identité visuelle forte et professionnelle, tout en offrant une expérience utilisateur agréable et équilibrée grâce à une esthétique soignée.



#F39C12 #FFFFFF #3498DB

Figure 52: Palette de couleur utiliser

3.2 Contrôle de données

Pour enrichir notre projet, nous avons intégré Bootstrap, ce qui nous a permis d'ajouter facilement des composants modernes et réactifs. En particulier, nous avons utilisé Bootstrap pour implémenter des toasts[14], qui offrent une manière élégante de fournir des notifications temporaires aux utilisateurs, ainsi que des cards[15], qui structurent visuellement le contenu de manière claire et attrayante. Ces ajouts ont considérablement amélioré l'interface utilisateur et l'expérience globale de notre plateforme.

Conclusion

Au cours de ce sprint, nous avons détaillé les principales fonctionnalités liées à la gestion des messages et des notifications. Nous terminons notre rapport par une conclusion générale et les différentes perspectives identifiées.

Conclusion générale et perspectives

En conclusion, notre projet HomeCare a été conçu pour répondre aux besoins croissants du secteur des services à domicile en simplifiant les processus administratifs. Grâce à notre application, nous avons créé une plateforme centralisée facilitant une collaboration efficace entre administrateurs, clients et prestataires. L'adoption d'une approche agile nous a permis de livrer des fonctionnalités de manière itérative, en intégrant les retours des utilisateurs tout au long du développement.

Nous avons particulièrement soigné la conception des interfaces utilisateur pour garantir une expérience fluide et intuitive. Notre projet met également l'accent sur le suivi individuel des clients avec des fonctionnalités telles que la gestion des annonces, offrant ainsi aux clients et aux prestataires des outils précieux pour évaluer et améliorer les performances de réservation des services à domicile.

En implémentant notre application HomeCare, nous visons à moderniser et optimiser le processus de réservation tout en améliorant la qualité des services. Nous sommes convaincus que notre projet facilitera le travail des prestataires et offrira une expérience plus enrichissante aux clients.

En somme, HomeCare représente une avancée significative dans l'utilisation de la technologie pour améliorer le système de réservation des services à domicile. Nous espérons que notre application continuera à évoluer et à s'adapter aux besoins changeants du secteur, créant ainsi des opportunités favorables pour les prestataires et les clients.

Bibliographie et Nétographie

- [1] présentation de la société ProxiWeb : <https://proxiweb.tn/> [consulté le 15/02/2024].
- [2] "Les méthodes agiles" : https://fr.wikipedia.org/wiki/M%C3%A9thode_agile [consulté le 29/03/2024].
- [3] Florent Lothon "Guide scrum" : <https://agiliste.fr/guide-de-demarrage-scrum/> [consulté le 01/04/2024].
- [4] Figma : <https://www.figma.com/> [consulté le 05/02/2024].
- [5] StarUML : <https://staruml.io/> [consulté le 10/02/2024].
- [6] Visual Studio Code : <https://code.visualstudio.com/> [consulté le 01/03/2024].
- [7] Node.js : <https://nodejs.org/en> [consulté le 01/03/2024].
- [8] MongoDB : <https://www.mongodb.com> [consulté le 01/03/2024].
- [9] React : <https://fr.react.dev/> [consulté le 01/03/2024].
- [10] Microsoft Word : https://fr.wikipedia.org/wiki/Microsoft_Word/ [consulté le 01/03/2024].
- [11] "Architecture 3-tiers" : <https://www.ibm.com/fr-fr/topics/three-tier-architecture> [consulté le 06/04/2024].
- [12] "Redux Store" : <https://redux.js.org/api/store/> [consulté le 15/05/2024].
- [13] "Template bootstrap" : <https://startbootstrap.com/previews/sb-admin-2/> [consulté le 08/02/2024].
- [14] "Toasts bootstrap" : <https://getbootstrap.com/docs/5.3/components/toasts/> [consulté le 25/02/2024].
- [15] "Card bootstrap" : <https://getbootstrap.com/docs/5.3/components/card/> [consulté le 13/02/2024].

Liste des abréviations

UML : Unified Modeling Language

API : Application Programming Interface

HTTP : Hypertext Transfer Protocol