



**Direction Générale des Études Technologiques Institut Supérieur
des Études Technologiques de Béja
Département Technologies de l'informatique**

Rapport de Projet de Fin d'études

Présentée en vue de l'obtention du

Diplôme de Licence Appliquée en Technologies de l'Informatique

Spécialité : Développement Système d'Informations

SUJET

Conception et développement d'une application web

qui propose des services à domiciles

Réalisé par : Jouini oumayma

Au sein de :



Encadrant société : Mr Nafatti Ahmed

Encadrant ISET : Mme Melki Narjess

Année universitaire : 2021/2022

Dédicace

Avec l'expression de ma reconnaissance, je dédie ce modeste travail à ceux qui, quels que soient les termes embrassés, je n'arriverais jamais à leur exprimer mon amour sincère.

A la femme qui a souffert sans me laisser souffrir, qui n'a jamais dit non à mes exigences et qui n'a épargné aucun effort pour me rendre heureuse : mon adorable mère Loulaifa

Je t'aime beaucoup maman

A mon très cher père,

Pour le gout à l'effort qu'il a suscité en moi. Rien au monde ne vaut les efforts fournis jour

et nuit pour mon éducation et mon bien être.

A mon frère et ma sœur,

Qui m'ont toujours soutenu et encouragé durant ces années d'études.

A mes amis et mes camarades,

Pour tous les moments merveilleux dans mes années universitaires.

Merci pour leurs amours et leurs encouragements.

A toutes mes professeurs pendant mon parcours.

Remerciements

Au terme de ce travail je tiens tout d'abord à remercier :

Mon DIEU de m'avoir donné le courage, la force et la volonté pour achever ce modeste travail.

Mon encadrant de la société Monsieur Naffati Ahmed.....

Mon encadrante Mme Melki Narjess, enseignante à l'Institut Supérieur de Béja pour son encadrement, ses remarques constructives tout le long de mon travail, pour le temps qu'elle a bien voulu consacrer à l'encadrement et le suivi de ce travail, les conseils qu'elle a prodigué durant les réunions qui ont rythmées les différentes étapes de la rédaction de ce rapport.

Enfin je tiens à remercier également les membres du jury qui ont accepté d'évaluer mon travail.

Liste des figures

Figure 1: Logo société.....	2
Figure 2: Site Jemepropose Visité le 20/02/2022.....	3
Figure 3: Site Travaux.com Visité le 20/02/2022	4
Figure 4: Site Travaux.com Visité le 20/02/2022	4
Figure 5: Méthodologie de conception en cascade	5
Figure 6: Concept de la méthode Scrum	7
Figure 7: Diagramme de cas d'utilisation global	11
Figure 8 : Le Modèle MVC.....	18
Figure 9: Architecture physique	19
Figure 10: Diagramme de cas d'utilisation « Gérer son profil ».....	22
Figure 11:: Diagramme de cas d'utilisation « Gérer utilisateurs ».	22
Figure 12: Diagramme de classe du Sprint 1	23
Figure 13: Diagramme de séquence de cas « S'authentifier »	24
Figure 14: Diagramme de séquence de cas « Modifier son profil »	24
Figure 15: Diagramme de séquence du cas « Consulter liste des utilisateurs ».....	25
Figure 16: Diagramme de séquence du cas « Supprimer utilisateur »	25
Figure 17: interface d'inscription d'un utilisateur.....	26
Figure 18: Interface d'authentification.....	26
Figure 19: Interface d'accueil d'administrateur.	27
Figure 20: modifier son profil	27
Figure 21: Consulter liste utilisateurs.....	28
Figure 22: interface représentant la liste des utilisateurs bloqués	28
Figure 23: bouton « se déconnecter ».....	29
Figure 24: Interface d'accueil d'un client	29
Figure 25: Modifier son profil.....	30
Figure 26:: Diagramme de cas d'utilisation « Gérer ses annonces ».	32
Figure 27: Diagramme de cas d'utilisation « Gérer annonces ».	33
Figure 28: Diagramme de classe du sprint n°2	33
Figure 29: Diagramme de séquence du cas « Ajouter une annonce ».....	34
Figure 30: Ajout annonce	35
Figure 31: interface pour consulter ses annonces.....	35
Figure 32: Liste de ses annonces	36
Figure 33: interface de modifier son annonce	36
Figure 34: interface représentant la liste des annonces de clients	37
Figure 35: interface pour la suppression des annonces	37
Figure 36: Supprimer Annonce	38
Figure 37: Interface représentant la liste des autres annonces	38
Figure 38: Détails annonce.....	39
Figure 39: Cas d'utilisation « Gérer réclamation »	41
Figure 40: Diagramme de classe du sprint n°3	42
Figure 41: Diagramme de séquence du cas « Envoyer réclamation ».....	43

Figure 42:Diagramme de séquence du cas « Consulter liste des réclamations »	43
Figure 43:Diagramme de séquence du cas « Répondre au réclamation »	44
Figure 44:Formulaire de réclamation	44
Figure 45:Envoyer réclamation	45
Figure 46:Liste des réclamations.....	45
Figure 47:Envoyer Mail	46
Figure 48:Mail envoyé par l'administrateur	46
Figure 49: Interface Console Google Firebase.....	48
Figure 50 : Ajout d'un projet	48
Figure 51: interface ajout projet	48
Figure 52:Interface de Firebase	49
Figure 53: Interface de la création d'une base de données	49
Figure 54: Interface configuration de la base de données	49
Figure 55: Aperçu de la base.....	50
Figure 56: Interface de l'installation du module Firebase sur Angular	50
Figure 57: Importation de Firebase	50
Figure 58: Lien de la base de données	50
Figure 59:Initialisation du paramètre de configuration de Firebase.....	51
Figure 60:Liste des annonces des clients	51
Figure 61: Liste des annonces par catégorie	52
Figure 62: Détails annonce.....	52
Figure 63: chat pour professionnel.....	53
Figure 64:chat pour clients	53

Liste des Tableaux

Tableau 1: Backlog Product	11
Tableau 2: Tableau des langages de programmation utilisés	13
Tableau 3: Description des framework utilisés	14
Tableau 4: Description des bibliothèques utilisées	15
Tableau 5: Tableau des logiciels utilisés	15
Tableau 6 : Tableau des serveurs utilisés	17
Tableau 7: Backlog Sprint du Sprint n°1	20
Tableau 8 : Backlog Sprint du Sprint n°2	31
Tableau 9: Backlog Sprint du Sprint n°3	40
Tableau 10 : Backlog Sprint du Sprint n°4	47

Introduction Générale

Depuis quelques années, les innovations dans le domaine du développement web se multiplient et évoluent sans cesse, la révolution d'informatique a balayé tous les pays du monde, cela concerne également notre pays la Tunisie qui a connu un fort développement incontestable dans ce domaine. Pour cela l'informatique est devenue indispensable pour les grandes et les petites entreprises.

De nos jours, de nombreux artisans rencontrent de grandes difficultés pour trouver des offres d'emplois, d'autant que ces métiers ne disposent pas de plateforme pour les aider à trouver des offres comme les autres métiers.

Dans ce cadre nous sommes appelés durant notre projet de fin d'études à concevoir et à développer une solution informatique permettant de mettre en place une plateforme de services a domiciles Cette application leur permettra à ces derniers d'aider les professionnels à trouver des offres d'emploi plus facilement et de permettre aux gens de trouver des services proches de chez eux de manière simple, sans se déplacer pour chercher.

Ce rapport détaillera les différentes phases afin d'aboutir à une application fiable et satisfaisante, pour cela notre rapport sera composé de six chapitres organisés comme suit :

Le premier chapitre intitulé « Cadre général du projet et Méthodologies » est consacré à la présentation du contexte du projet et le choix de la méthodologie Scrum pour la phase de développement.

Le deuxième chapitre, « Sprint 0 : Préparation des bases du projet » s'articule autour de l'identification des acteurs et la description des besoins fonctionnels et non fonctionnels. Puis, nous élaborons le Backlog du produit et nous terminons par la présentation des architectures et l'environnement de développement de notre application.

Le troisième, le quatrième, le cinquième et le sixième chapitre « Réalisation du Sprint » mettent en œuvre les sprints Backlog associés à chaque sprint, les diagrammes de cas d'utilisations et par les diagrammes de séquences ainsi que la réalisation. Enfin, nous clôturons ce rapport par une conclusion générale ainsi que la proposition de quelques perspectives sur lesquels peut s'ouvrir le présent travail.

Chapitre 1 : Cadre général du projet et Méthodologies

Introduction

Avant d'entamer les différentes parties de ce rapport, il convient de présenter l'environnement dans lequel il a été mené. En effet, c'est de l'environnement que dépend, en grande partie, l'efficacité et la qualité d'un stage.

Ce chapitre présente tout d'abord l'organisme d'accueil, ensuite l'étude préalable qui nous mène à dégager la problématique et les objectifs du projet, puis la solution proposée, le choix de la méthode de conception. Enfin, nous clôturons ce chapitre par une conclusion.

I. Présentation de l'organisme de l'accueil

1. Description de l'organisme d'accueil (Beecoders)

Beecoders est une agence de développement web et mobile, spécialisée en création des sites internet, développement d'application mobile, web, et logiciels

Elle offre aussi pour les étudiants ou les personnes passionnées par les technologies de l'information, des formations pratiques en ligne sur leur plateforme 9antra.tn et leur permet d'obtenir des certificats. Elle contient des équipements informatiques dans différents domaines

- Web Design
- Web Development
- Product Management
- Marketing
- Graphic Design



Figure 1: Logo société BeeCoders

II. Etude de l'existant

L'étude de l'existant est la première étape pour la réalisation d'un projet. Pour évaluer une application, il faut tout d'abord analyser et critiquer l'existant pour dégager les points forts et les points faibles. Le résultat de cette section a pour objectif de trouver la meilleure solution aux problèmes rencontrés.

1. Description et critique de l'existant

À l'heure actuelle en Tunisie, il n'y a pas de plateforme qui fournit des services à domicile, c'est pourquoi nous avons eu l'idée de développer ce projet et de créer un site Web qui permet aux gens de trouver des services proches de chez eux de manière simple, sans se déplacer pour

chercher, en même temps notre plateforme aide les professionnels à trouver des offres d'emploi plus facilement.

Pour bien mener notre étude, nous avons consulté quelques sites à l'étranger présentant le même concept. Nous en citons :

- [Jemepropose.com](https://www.jemepropose.com) la plate-forme de jobbing française, d'emplois et de services pour particuliers et professionnels.

Elle permet aux utilisateurs de publier gratuitement leurs annonces afin que les professionnels sur le site puissent les contacter et proposer leurs services.

Plusieurs domaines sont présents sur la plateforme, à titre d'exemple nous citons : plomberie, ménage, babysitting, menuiserie, peinture, etc., la publication d'annonces est simple et rapide. La plate-forme contient un espace client et un espace professionnel.

L'attractivité du site est moyenne.



Figure 2: Site Jemepropose Visité le 20/02/2022

- [Travaux.com](https://www.travaux.com) est un site français mettant en relation les particuliers et professionnels de l'habitat. Il leur offre un moyen simple pour s'identifier, préparer et réaliser tous les projets de travaux.

Il permet aux particuliers de publier leur projet et de sélectionner les artisans qu'ils souhaitent, puis les artisans intéressés les contactent par mail.

L'attractivité du site est faible. Les fonctionnalités offertes aux particuliers sont limitées, ils ne peuvent que créer des projets, ils n'ont pas un espace privé, ne peuvent pas envoyer des messages directs aux professionnels. Ces derniers, peuvent créer des comptes et contacter les particuliers par mail.

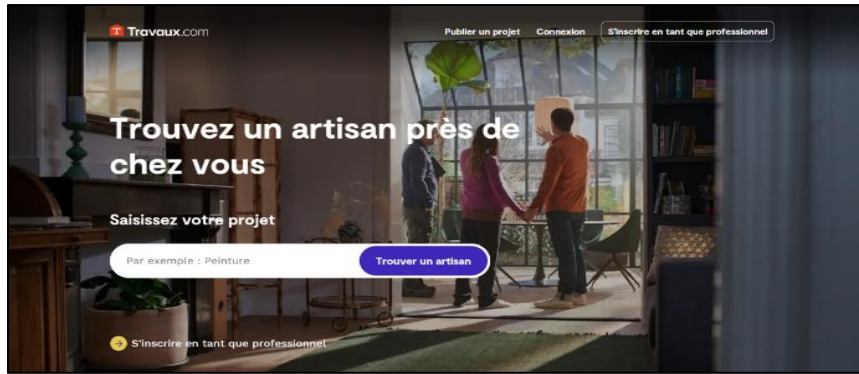


Figure 3: Site Travaux.com Visité le 20/02/2022



Figure 4: Site Travaux.com Visité le 20/02/2022

2. Solution proposée

Comme il n'existe pas de site en Tunisie dédié aux services à domiciles, et suite à l'étude effectuée dans la partie précédente, nous nous sommes proposés de développer un site web destinés aux professionnels dans les domaines de construction, de plomberie, etc. ainsi que les personnes ayant besoin de leurs services pour faciliter leur prise de contact et leur communication.

Notre application permet au professionnel de créer son profil et indiquer ses domaines d'expertise. Le client, en sa partie, peut publier des annonces pour les services dont il a besoin. Les deux peuvent se contacter et communiquer via notre plateforme.

III. Méthodologie de travail et langage de modélisation

1. Les méthodologies de travail traditionnelles VS Agiles

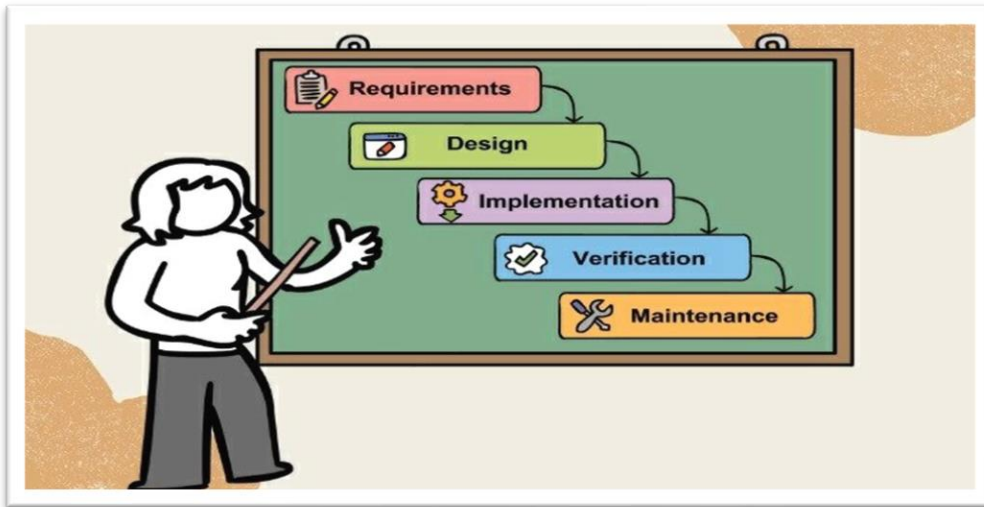


Figure 5: Méthodologie de conception en cascade

La méthode en Cascade est un modèle où les phases de développement suivent rigoureusement un ordre spécifique. La phase suivante ne peut commencer que lorsque la phase précédente a été conclue. D'une manière générale, l'approche en Cascade commence par une planification et un design étendu, suivis d'un codage et d'un test, et se termine par la publication. L'idée principale derrière la méthode en Cascade est que le processus de planification approfondi nie la nécessité d'apporter des ajustements importants lors du développement. À cette fin, la technique en Cascade tente de se préparer à tous les scénarii afin d'éviter des retards qui coûtent du temps et des ressources.

La solution pour remédier aux inconvénients des méthodes traditionnelles de gestion de projet est « la méthode Agile ».

Définition de la méthode Scrum

La méthode agile est une méthode de gestion de projet collaborative qui décompose les grands projets en petites tâches, que les équipes abordent en collaboration.

La méthode agile valorise la flexibilité et la collaboration entre équipes, ce qui libère les équipes de la structure par étapes des méthodes linéaires afin de s'adapter au projet au fur et à mesure de sa progression.

La gestion de projet agile fonctionne en itérations ou en sprints. Chaque itération est une partie du projet achevée, avec un court délai (généralement 1 à 3 semaines) sur laquelle une équipe se concentre et qu'elle mène à bout. Cette partie achevée est ensuite diffusée afin de recevoir un retour d'information. L'équipe étudie les commentaires qu'elle reçoit et adapte le projet au fur et à mesure qu'elle travaille sur chaque sprint.

La méthode Agile répond aux problèmes liés aux méthodes traditionnelles de gestion de projet en mettant l'accent sur deux aspects :

- Impliquer le client au maximum dans le projet
- Les développements itératifs et incrémentaux

2. Méthodologie choisie : SCRUM

Notre choix s'est porté sur la méthode Scrum de gestion de projet car c'est une gestion plus souple, plus intelligente du travail, avec suffisamment de liberté pour apporter des modifications à un projet, quel que soit son état d'avancement. Offrant plus de réactivité, elle est plus adaptée que les méthodes traditionnelles pour la gestion de projets web, tel que le développement logiciel, car elle traduit et organise les projets de façon simple, transparente et pragmatique.

a) Rôles SCRUM

La méthodologie SCRUM fait intervenir 3 rôles principaux qui sont :

Directeur du produit (Product Owner) : C'est le responsable de l'orientation du projet, la production et la mise à jour du carnet du produit.

C'est lui qui va définir et prioriser la liste des fonctionnalités du produit et choisir la date et le contenu de chaque sprint sur la base des valeurs (charges) qui lui sont communiquées par l'équipe

Chef de mêlée (Scrum Master) : c'est une personne chargée de résoudre les problèmes éventuels qui empêcherait l'avancement du projet pendant les différents sprints.

Equipe : l'équipe s'organise elle-même et elle reste inchangée pendant toute la durée d'un sprint. Le rôle est de transformer les besoins exprimés par le propriétaire du produit en fonctionnalités utilisables.

Dans notre projet, nous pouvons distinguer les rôles suivants :

- **Product Owner:** Mr Naffati Ahmed
- **Scrum Master:** Mme Melki Narjess
- **L'équipe de développeurs :** Jouini oumayma

b) Concepts SCRUM

Nous pouvons remarquer cette figure pour mettre en place la méthode SCRUM

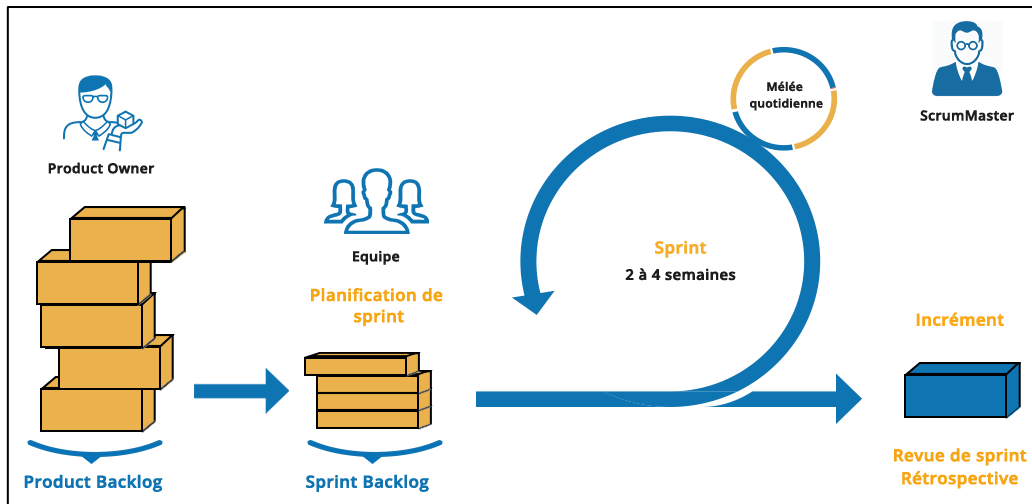


Figure 6: Concept de la méthode Scrum

- **Sprint** : Le Sprint est une période qui varie entre deux et quatre semaines au maximum. Durant cette période, l'équipe délivre un livrable du produit. La durée fixée reste constante pendant toute la durée du développement.
- **Product Backlog** (ou carnet de produit) : Contient un ou plusieurs éléments du Product Backlog à réaliser durant un sprint. L'équipe décompose chaque élément en liste de tâches de développement.
- **Sprint Backlog** : Contient un ou plusieurs éléments du product backlog à réaliser durant un sprint. L'équipe décompose chaque élément de là en liste de tâches de développement.
- **Scrum Meeting** : Le Scrum Meeting n'est qu'une réunion pendant laquelle nous cherchons à résoudre les problèmes, mais uniquement à les identifier et les exprimer.
- **Revue de Sprint** : À la fin du Sprint, l'Equipe, le Scrum Master et le Product Owner se réunissent pour effectuer la revue du sprint dont le but est de valider le livrable qui a été produit durant le Sprint. L'équipe commence par énoncer les user story du Backlog de produit qu'elle a réalisé.
- **Rétrospective de Sprint** : La rétrospective du sprint est faite en interne à l'équipe. L'objectif est de détecter les erreurs commises et de prendre des décisions pour les améliorer.

Scrum est :

- Léger.
- Simple à comprendre.
- Difficile à maîtriser.

Durant un développement d'un projet avec la méthode Scrum il y a plusieurs étapes à suivre avec une démarche spécifique et une interaction avec plusieurs intervenants.

c) *Langage de modélisation : UML (Unified Modeling Language)*

Nous avons besoin d'un langage de modélisation unifié pour la modélisation de notre projet. Pour concevoir notre système, nous avons choisi UML comme un langage de modélisation.

Notre choix s'est basé sur le point fort de ce langage notamment sa standardisation et le divers diagramme qu'il propose. Aussi UML présente le meilleur outil pour Schématiser des systèmes complexes sous un format graphique et textuel simplifié et normalisé.

Dans notre projet, nous avons utilisé l'outil Lucidchart et Visual Paradigm pour schématiser nos diagrammes (diagramme du cas d'utilisation, diagramme de classe, diagramme de séquence).

Conclusion

Au cours de ce chapitre nous avons présenté le cadre général de notre travail. Aussi, Nous avons introduit, en premier lieu, l'organisme d'accueil, ensuite, nous avons cité les solutions existantes et critiqué l'existant, puis, nous avons proposé une solution et enfin nous avons mis l'accent sur la méthode du travail adoptée.

Chapitre 2 : Sprint 0 : Préparation des bases de projet

Introduction

Ce chapitre vise à capter les besoins ainsi qu'à identifier les rôles des utilisateurs qui utilisent l'application et préparer le plan de réalisation de notre projet. Dans un premier lieu, nous commençons par définir les acteurs, par la suite nous spécifions les besoins fonctionnels et non fonctionnels. Au niveau de la deuxième partie nous présentons le Backlog Product et l'architecture technique de notre application, ainsi que le diagramme de cas d'utilisation global.

Enfin, nous décrivons l'environnement de développement et les technologies utilisées.

I. Spécification des besoins

1. Identification des acteurs

L'application est destinée à 4 acteurs, qui sont :

- **Visiteur** : internaute qui peut consulter les annonces déposées des clients sans inscription.
- **Administrateur** : C'est la personne possédant le privilège de plus haut niveau. Il se charge de la gestion des utilisateurs, des annonces, des réclamations, etc.
- **Client** : C'est une personne qui a besoin d'un certain service et souhaite déposer des annonces. Il peut s'authentifier à notre site, déposer des annonces, faire des réclamations et finalement répondre aux messages.
- **Professionnel** : C'est un expert offrant ses services aux clients. Il peut s'authentifier à notre site, choisir l'annonce qui a retenu son attention, contacter le client et finalement il peut faire aussi des réclamations.

2. Descriptions des besoins fonctionnels

Dans ce qui suit, nous listons les besoins fonctionnels offerts par notre application en fonction de chaque acteur. Le système doit permettre :

A l'administrateur de :

- S'authentifier
- Gérer les annonces
 - Consulter la liste des annonces affichées
 - Supprimer une annonce
- Gérer les utilisateurs
 - Consulter la liste des utilisateurs
 - Consulter la liste des personnes bloquées
 - Supprimer un utilisateur
 - Bloquer/débloquer un utilisateur
- Gérer son profil
 - Consulter son profil
 - Modifier ses données
- Gérer les réclamations

- Consulter la liste des réclamations
- Supprimer une réclamation
- Répondre aux réclamations (par email)

Au professionnel de :

- S’inscrire
- S’authentifier
- Gérer son profil
 - Consulter son profil
 - Modifier ses données
- Faire des réclamations
- Gérer les annonces
 - Consulter la liste des annonces
 - Contacter le client

Au client de :

- S’inscrire
- S’authentifier
- Gérer son profil
 - Consulter son profil
 - Modifier ses données
- Faire des réclamations
- Consulter la liste des annonces déposées par les autres clients
- Gérer ses annonces
 - Publier une annonce
 - Consulter la liste de ses annonces
 - Supprimer son annonce
 - Modifier son annonce
- Gérer les messages

3. Description des besoins non-fonctionnels :

Les besoins non-fonctionnels de notre système se décrivent comme suit :

- **Besoins de sécurité :** tous les accès aux différentes espaces doivent être protégées par un mot de passe et un privilège d'accès. Ainsi, il faut s'assurer des cryptages des données au niveau de la base de données.
- **Besoins d'utilisation :** tous les standards d'ergonomies doivent être présents : interface utilisateur bien claire et simple dans l'utilisation.
- **L'ergonomie :** l'application doit être conforme aux principes de l'interface Homme-Machine.
- **La fiabilité :** l'application doit toujours donner des résultats corrects aux clients.

II. Diagramme de cas d'utilisation global

Les diagrammes de cas d'utilisation sont des diagrammes UML utilisés pour donner une vision globale du comportement fonctionnel d'un système logiciel.

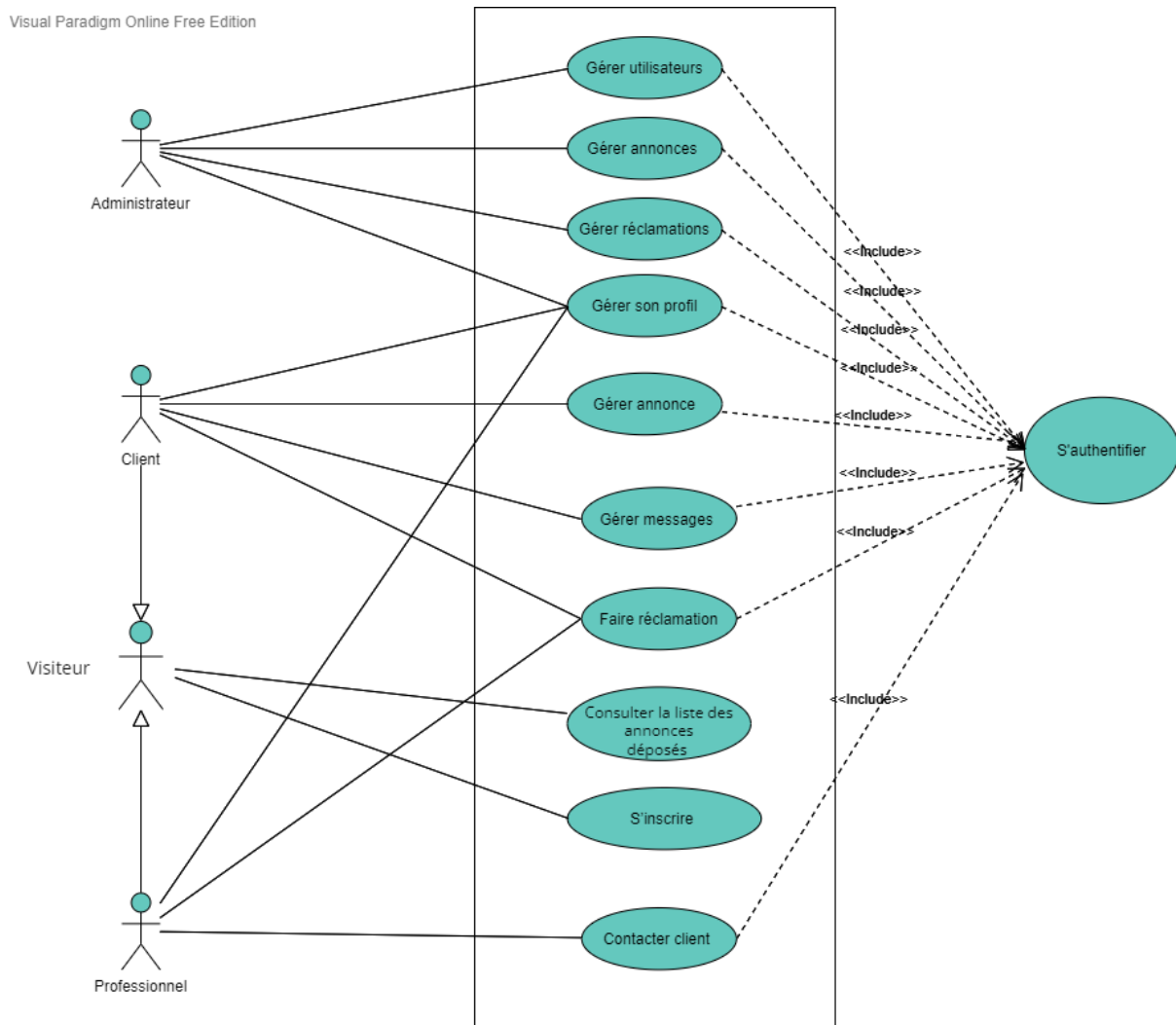


Figure 7:Diagramme de cas d'utilisation global

III. Backlog Product

En se basant sur la méthodologie Scrum et les besoins fonctionnels identifiés précédemment nous avons préparé notre Backlog Product :

Tableau 1: Backlog Product

ID	User Story	Priorité	Estimation	Risque 0 ->10	Sprint
1	En tant qu'administrateur, je dois pouvoir m'authentifier.	Haute	24 Heures	2	Sprint N°1

2	En tant qu'administrateur, je dois pouvoir gérer mon profil.	Haute	9 Heures	2	Sprint N°1
3	En tant qu'administrateur, je dois pouvoir gérer les utilisateurs.	Haute	5 Heures	3	Sprint N°1
4	En Tant qu'Administrateur, je dois pouvoir gérer les annonces déposés	Haute	12 Heures	2	Sprint N°2
5	En Tant qu'Administrateur, je dois pouvoir consulter le tableau de bord	Haute	9 Heures	2	Sprint N°3
6	En tant qu'administrateur je dois consulter la liste des réclamations des utilisateurs	Haute	12 Heures	2	Sprint N°3
7	En tant qu'administrateur je dois répondre aux réclamations des utilisateurs	Haute	24 Heures	5	Sprint N°3
8	En tant qu'administrateur je dois pouvoir me déconnecter	Haute	1 heure	1	Sprint N°1
9	En Tant que Professionnel/client, je peux m'inscrire	Haute	12 Heures	2	Sprint N°1
10	En Tant que Professionnel/client, je peux m'authentifier	Haute	12 Heures	3	Sprint N°1
11	En Tant que Professionnel /client, je dois pouvoir gérer mon profil	Moyenne	9 Heures	2	Sprint N°1
12	En tant que Professionnel/client, je dois faire des réclamations	Moyenne	5 Heures	2	Sprint N°3
13	En tant que Professionnel, je dois consulter la liste des annonces déposés	Moyenne	12 Heures	2	Sprint N°4
14	En tant que Professionnel, je contacter les clients	Moyenne	48 Heures	6	Sprint N°4

15	En Tant que Client, je dois pouvoir gérer mes annonces	Faible	24 Heures	5	Sprint N°2
16	En Tant que Client, je dois consulter la liste des autres annonces	Faible	24 Heures	2	Sprint N°2
17	En Tant que Professionnel/client, je dois gérer les messages	Faible	24 Heures	5	Sprint N°4
18	En tant que Client/Professionnel, je dois pouvoir se déconnecter.	Faible	1 heure	1	Sprint N°1

IV. Environnement de travail

Dans cette partie, nous présenterons l'environnement matériel et logiciel relatif à la réalisation de l'application.

1. Environnement matériel

Pour réaliser notre travail, nous avons utilisé comme environnement matériel un ordinateur portable de marque « Lenovo » avec un système d'exploitation Windows 10 pro 64 bits, ayant un processeur Intel® Core(TM) i5-1135G7 @ 2.40GHz 2.42 GHz, doté d'une Ram de 16,0 Go




2. Environnements logiciels

Dans cette partie, nous présentons l'environnement logiciel de notre application de point de vue langages, Framework, Logiciels et technologies.

a) Langages

Dans le tableau suivant, Il y'a une description des langages utilisés lors du développement de l'application.


Tableau 2: Tableau des langages de programmation utilisés


Langages	Description
HTML 	Hypertext Markup Language (HTML) , langage de balisage utilisé pour la création de pages web, permettant notamment de définir des liens hypertextes. [B3]
CSS 	Les feuilles de style en cascade généralement appelées CSS de l'anglais <i>Cascading Style Sheets</i> . Il est employé pour déterminer le style de l'application. Le CSS (ou feuille de style), est un document au travers duquel j'ai défini le choix de couleurs, type de police, les positions... [B3]
	JSON (JavaScript Object Notation) est un format léger d'échange de données. Il est facile à lire ou à écrire pour des humains. Il est aisément analysable ou généré par des machines. Il est basé sur un sous-ensemble du langage de programmation JavaScript (JavaScript Programming Language, Standard ECMA-262 3rd Edition - Décembre 1999). [B3]

b) *Framework et technologies utilisées*

Dans le tableau suivant, Il y'a une description de frameworks utilisés lors du développement de l'application.

Tableau 3: Description des framework utilisés


Framework/Technologies	Description
	Spring Boot est un framework avancée qui simplifie le démarrage et le développement des nouvelles applications Java EE. Les configurations sont atténuées avec Spring Boot, qui soutient des conteneurs embarqués. Cela permet à des applications web de s'exécuter indépendamment et sans déploiement sur le serveur web. [W5]

	<p>Angular est un Framework basé sur TypeScript coté client qui est utilisé pour créer des applications Web dynamiques. Il est très similaire à ses versions précédentes, à l'exception de quelques fonctions étendues. [B3]</p>
---	---

c) Bibliothèques

Dans le tableau suivant, Il y'a une description des bibliothèques utilisés lors du développement de l'application.


Tableau 4: Description des bibliothèques utilisées


Bibliothèque	Description
	<p>Angular Material est une bibliothèque de composants d'interface utilisateur pour les développeurs Angular JS. Les composants Angular Material aident à créer des pages Web et des applications Web attrayantes, cohérentes et fonctionnelles tout en adhérant aux principes de conception Web modernes tels que la portabilité du navigateur, l'indépendance de l'appareil et la dégradation gracieuse.[W4]</p>
	<p>Bootstrap est une compilation de plusieurs éléments et fonctions web design personnalisables, le tout emballé dans un seul et même outil. Ces éléments sont une combinaison de HTML, CSS et JavaScript. C'est l'un des projets les plus populaires sur la plateforme de gestion de développement GitHub. [B3]</p>

d) Logiciels

Dans le tableau suivant, Il y'a une description des logiciels utilisés lors du développement de l'application.

Tableau 5: Tableau des logiciels utilisés


Logiciels	Description
 Visual Studio Code	Visual Studio Code est un éditeur de code extensible développé par Microsoft par Windows, Linux et MacOS [6].
 Lucidchart	Lucidchart est une plateforme de collaboration en ligne, basée sur le cloud, permettant la création de diagrammes et la visualisation de données et autres schémas conceptuels.
 POSTMAN	Postman est un outil de développement logiciel, il permet aux utilisateurs de tester les appels aux API, les utilisateurs du facteur saisissent des données, les données sont envoyées à une adresse de serveur Web spéciale [7].
	Github est une entreprise de développement et services logiciels sise aux Etats-Unis. Github développe notamment la plateforme Github, l'éditeur de texte Atom ou encore la structure Electron.
	Google chrome : C'est un navigateur internet développé par Google. Ce navigateur est sur le marché depuis 2008 et fonctionne sur différentes plateformes (PC, tablettes, smartphone ...) et sous différents OS (Windows, mac, Linux, Android). [1]
 Visual Paradigm	Visual Paradigm est un logiciel de création de diagrammes dans le cadre de développement de logiciels.
	Microsoft teams est une plateforme collaborative personnalisable qui intègre la visioconférence, le stockage et le transfert de fichiers avec SharePoint, un bloc de prise de note avec OneNote. La messagerie instantanée permettant d'échanger en équipe ou par groupe de travail tout en conservant une trace de tous les échanges. [W4]

	<p>Git Bash une application pour les environnements Microsoft Windows qui fournit une couche d'émulation pour une expérience de ligne de commande Git. Bash est un acronyme pour Bourne Again Shell. Un shell est une application de terminal utilisée pour s'interfacer avec un système d'exploitation via des commandes écrites. [B3]</p>
---	--

e) Serveur SGBD

Dans le tableau suivant, Il y'a une description des serveurs (serveurs SGBD, serveurs de communication en temps réel) utilisés lors du développement de l'application.

Tableau 6 : Tableau des serveurs utilisés

Serveurs	Description
	<p>MySQL est un serveur de bases de données relationnelles SQL développées avec un souci de performances élevées en lecture, ce qui signifie qu'il est davantage orienté vers le service de données déjà en place que vers celui de mise à jour fréquentes et fortement sécurisées Il est multi-thread et multi-utilisateur. [W4]</p>
	<p>Firebase est un ensemble de services d'hébergement pour n'importe quel type d'application. Il propose d'héberger en NoSQL et en temps réel des bases de données, du contenu, de l'authentification sociale, et des notifications, ou encore des services, tel que par exemple un serveur de communication temps réel. [W9]</p>
	<p>Firebase Realtime Database est une base de données NOSQL hébergée sur le cloud. Les données sont stockées au format JSON et synchronisées en temps réel avec les utilisateurs.</p>

V. Architecture du système

1. Architecture logique :

Notre application a été développée sous l'architecture MVC (Model View Controller) :

C'est un modèle d'architecture qui cherche à séparer nettement les couches de présentation

(UI : Utilisateur Interface), métier (BLL : Business Logic Layer) et d'accès aux données

(DAL : Data Access Layer). Le but étant d'avoir une dépendance minimale entre les différentes couches de l'application ; ainsi les modifications effectuées sur n'importe quelle couche de l'application n'affectent pas les autres couches.

- **Model** : c'est le noyau de l'application qui gère les données, permet de récupérer les informations dans la base de données, de les organiser pour qu'elles puissent ensuite être traitées par le contrôleur.

- **View** : c'est un composant graphique de l'interface qui permet de présenter les données du modèle à l'utilisateur.

- **Contrôleur** : c'est un composant responsable des prises de décision, gère la logique du code qui prend des décisions, il est l'intermédiaire entre le modèle et la vue.

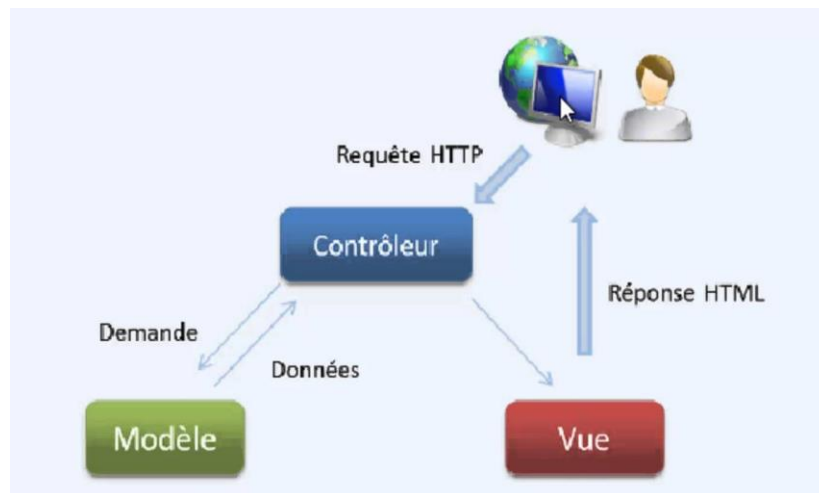


Figure 8 : Le Modèle MVC

2. Architecture physique :

Afin de réaliser notre projet, nous avons opté pour une architecture 3-tiers. Il s'agit d'un modèle logique d'architecture applicative qui vise à modéliser une application comme un empilement de trois couches logicielles dont le rôle est clairement défini :

- La présentation des données : correspondant à l'affichage, la restitution sur le poste de travail, le dialogue avec l'utilisateur.
- Le traitement métier des données : correspondant à la mise en œuvre de l'ensemble des règles de gestion et de la logique applicative.
- L'accès aux données persistantes : correspondant aux données qui sont destinées à être conservées sur la durée, voire de manière définitive.

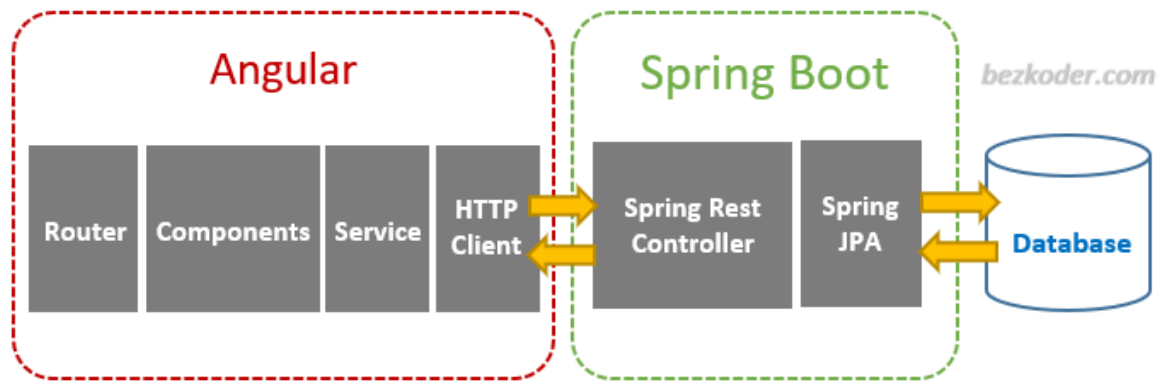


Figure 9: Architecture physique

Conclusion

Dans ce chapitre, nous avons planifié notre travail et identifié les besoins fonctionnels et non fonctionnels de notre application, les rôles des utilisateurs. Par la suite nous avons présenté le Backlog de notre système, ainsi nous avons détaillé la phase de planification des sprints. Nous enchainons à présent avec notre premier release dans le chapitre qui suit.

Chapitre 3 : Sprint 1 – Gestion des comptes des utilisateurs

Introduction

Dans ce chapitre nous présentons l'analyse, la conception et la réalisation du premier sprint de notre application. Ce sprint a pour objectif de permettre à l'administrateur et à l'utilisateur de s'authentifier et de pouvoir accéder à leurs fonctionnalités, enfin on décrit notre application avec des captures d'écran.

I. Backlog Sprint

L'objectif du Sprint (1) est de développer les cas d'utilisations représentés par le tableau suivant :

Tableau 7: : Backlog Sprint du Sprint n°1

ID	User Story	Tâches
1	En tant qu'administrateur, je dois pouvoir m'authentifier.	<ol style="list-style-type: none">1. Installation de logiciels2. Création de la base de données3. Création du projet Angular4. Création du serveur5. Réalisation des diagrammes de séquence du cas « authentification »6. Création de l'entité « User »7. Code pour s'authentifier à l'application (Back-end).8. Code pour s'authentifier à l'application (Front-end)9. Création de l'interface « Login »10. Test du cas « Authentification »
2	En tant qu'administrateur, je dois pouvoir gérer mon profil.	<ol style="list-style-type: none">11. Réalisation des diagrammes de cas d'utilisation, et de séquence détaillée du cas « gérer profil ».12. Création du formulaire pour la modification de son propre profil.13. Création du « mat-dialog » avec lequel, on va afficher le formulaire.14. Code du modification profil (Back-end).15. Code du modification profil (Front-end).16. Tester la modification.
3	En tant qu'administrateur, je dois pouvoir gérer les utilisateurs.	<ol style="list-style-type: none">17. Réalisation des diagrammes de cas d'utilisation, de séquence et de classe détaillés du cas « Gérer utilisateurs ».18. Création de l'interface dans laquelle, on va afficher la liste des utilisateurs nouveaux-inscrits.19. Code relatif à la liste des utilisateurs (Back-end).

		20. Code relatif à la liste des utilisateurs (Front-end). 21. Création des boutons supprimer et bloquer. 22. Code de la suppression (Back-end). 23. Code de la suppression (Front-end). 24. Création de l'interface dans laquelle, on va afficher les utilisateurs bloqués. 25. Code du blocage/ déblocage (Back-end). 26. Code du blocage/ déblocage (Front-end). 27. Tester la liste de tous les utilisateurs. 28. Tester la suppression d'un utilisateur. 29. Tester le blocage d'un utilisateur. 30. Tester le déblocage d'un utilisateur.
4	En tant qu'administrateur, je dois pouvoir me déconnecter.	31. Création du bouton log out. 32. Code relatif à la déconnection (Front-end). 33. Test du cas « se déconnecter ».
5	En tant que client, je dois pouvoir m'inscrire et m'authentifier à la plateforme.	34. Création de l'interface d'authentification (Front-end). 35. Code pour s'authentifier à l'application (Front-end). 36. Code pour s'authentifier à l'application (Back-end). 37. Création de l'interface d'inscription (Front-end). 38. Code pour s'inscrire à l'application (Front-end). 39. Code pour s'inscrire à l'application (Back-end). 40. Tester l'inscription 41. Tester l'authentification
6	En tant que professionnel je dois pouvoir m'inscrire et m'authentifier à la plateforme.	42. Création de l'interface d'authentification (Front-end). 43. Code pour s'authentifier à l'application (Front-end). 44. Code pour s'authentifier à l'application (Back-end). 45. Création de l'interface d'inscription (Front-end). 46. Code pour s'inscrire à l'application (Front-end). 47. Code pour s'inscrire à l'application (Back-end). 48. Tester l'inscription 49. Tester l'authentification
7	En tant que Professionnel/Client, je dois pouvoir gérer mon profil.	50. Réalisation des diagrammes de cas d'utilisation, et de séquence détaillée du cas « gérer profil ». 51. Création du formulaire pour la modification de son propre profil. 52. Création du « mat-dialog » avec lequel, on va afficher le formulaire. 53. Code du modification profil (Back-end). 54. Code du modification profil (Front-end).

		55. Tester la modification.
8	En tant que, professionnel /Client je dois pouvoir me déconnecter.	56. Création du bouton log out. 57. Code relatif à la déconnection (Front-end). 58. Test du cas « se déconnecter ».

II. Diagrammes de cas d'utilisation détaillés du Sprint n°1

Nous présentons ci-dessous le diagramme de cas d'utilisation du sprint 1

1. Cas d'utilisation « Gérer son profil »

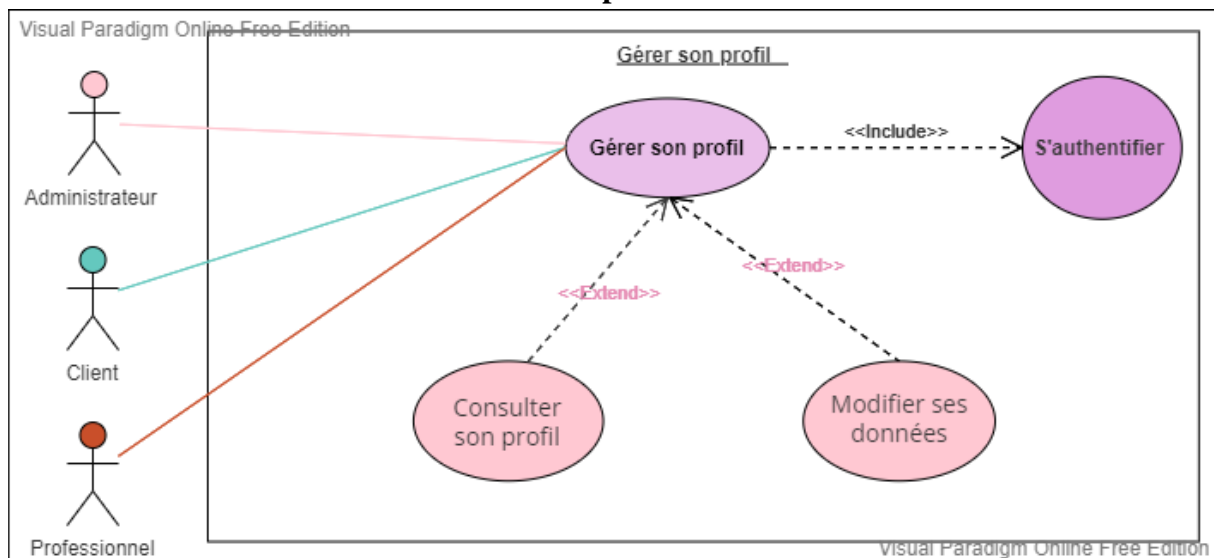


Figure 10: Diagramme de cas d'utilisation « Gérer son profil ».

2. Cas d'utilisation « Gérer utilisateurs »

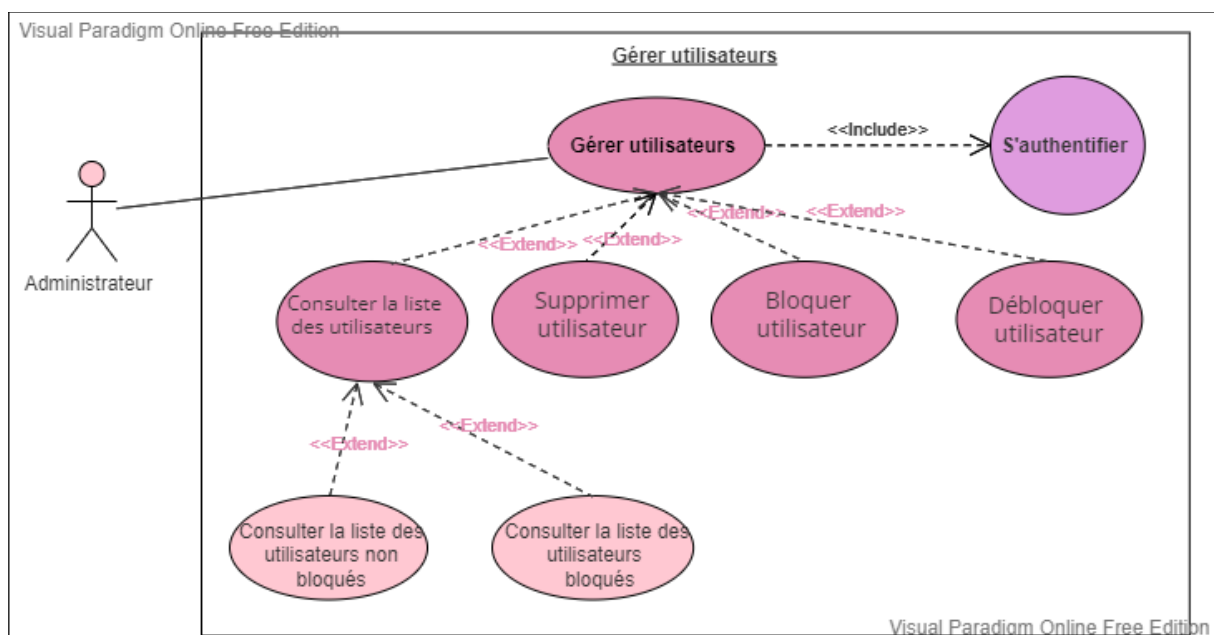


Figure 11: Diagramme de cas d'utilisation « Gérer utilisateurs ».

III. Diagramme de classe du Sprint n°1

Le diagramme de classe est considéré comme le plus important de la modélisation orientée objet, c'est le seul diagramme obligatoire lors d'une telle modélisation. Il permet de définir la structure interne du système.

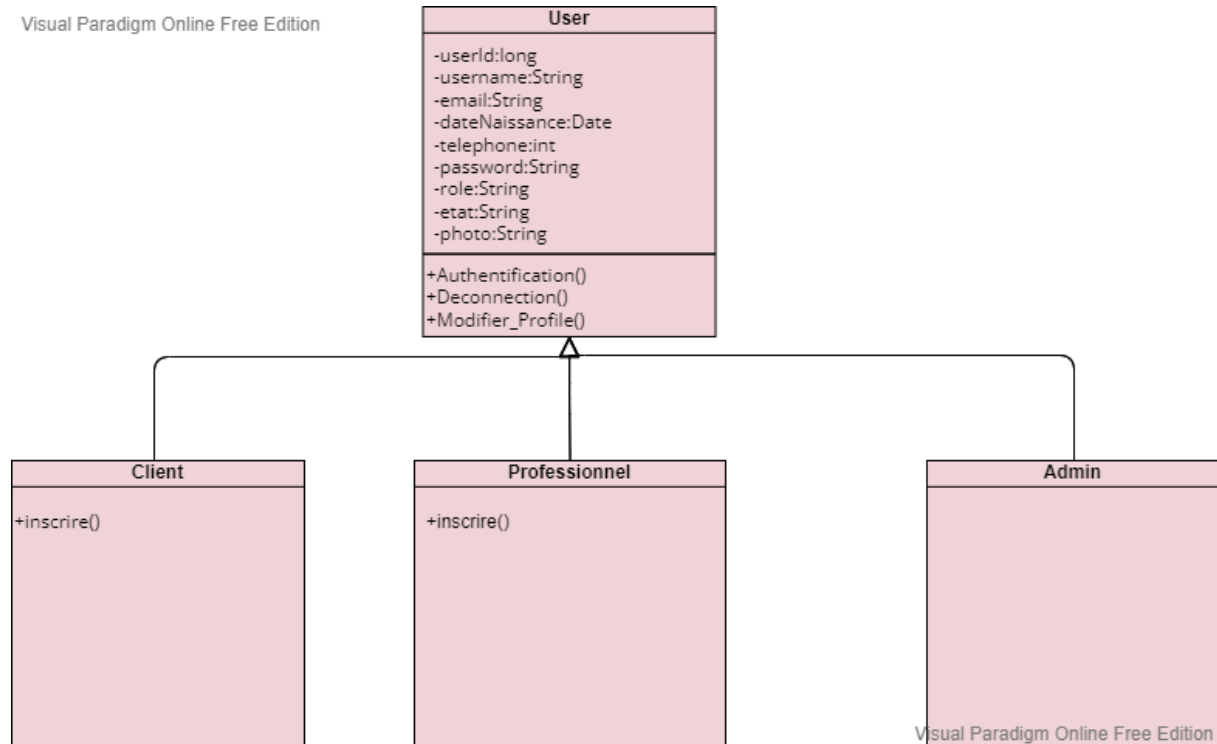


Figure 12:Diagramme de classe du Sprint 1

IV. Diagrammes de séquence du Sprint n°1

Le diagramme de séquence système est une modélisation de la vue dynamique du système. Ce diagramme reflète les interactions et les échanges entre un acteur et le système par des messages dans leurs ordres chronologiques.

1. Diagramme de séquence du cas « S'authentifier »

Le diagramme de séquence du cas d'utilisation s'authentifier est présenté par la figure suivante :

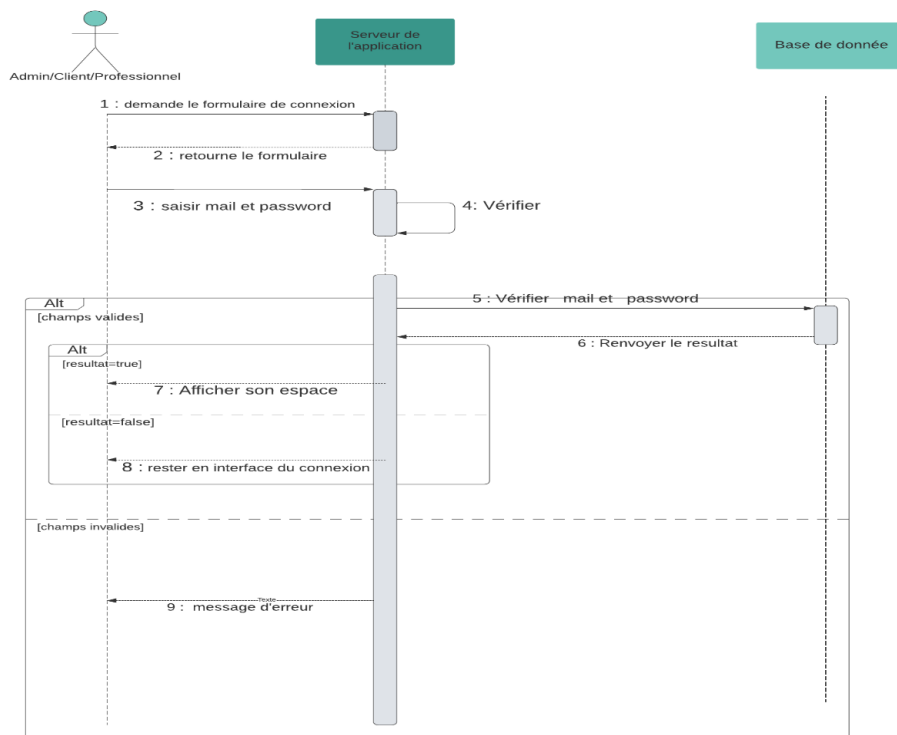


Figure 13:Diagramme de séquence de cas « S'authentifier »

2. Diagramme de séquence du cas « Modifier son profil ».

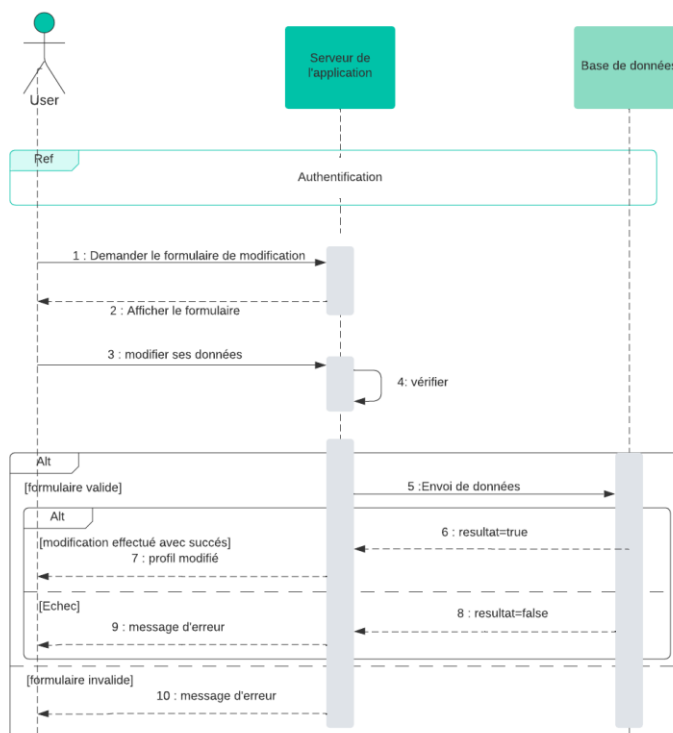


Figure 14:Diagramme de séquence de cas « Modifier son profil »

3. Diagramme de séquence du cas « Consulter liste des utilisateurs »

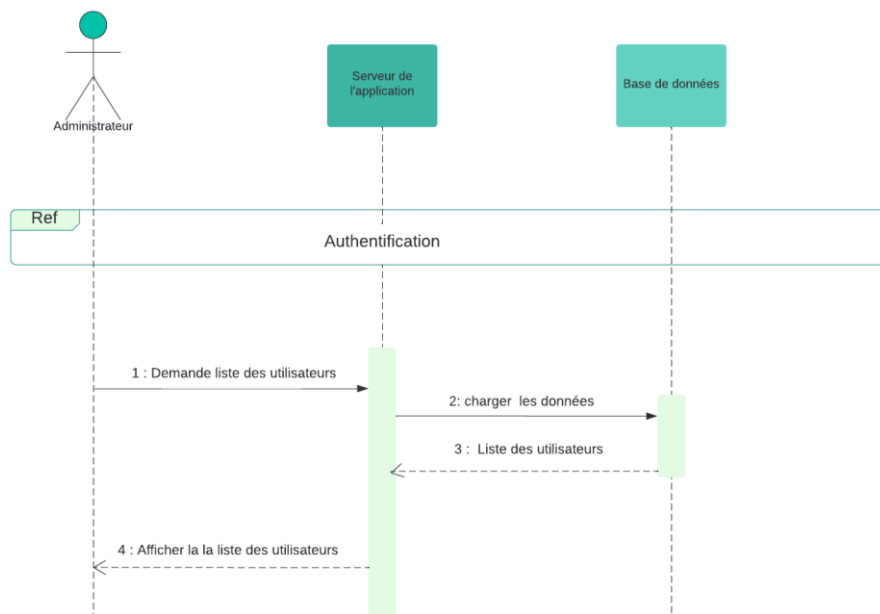


Figure 15:Diagramme de séquence du cas « Consulter liste des utilisateurs »

4. Diagramme de séquence du cas « Supprimer utilisateur »

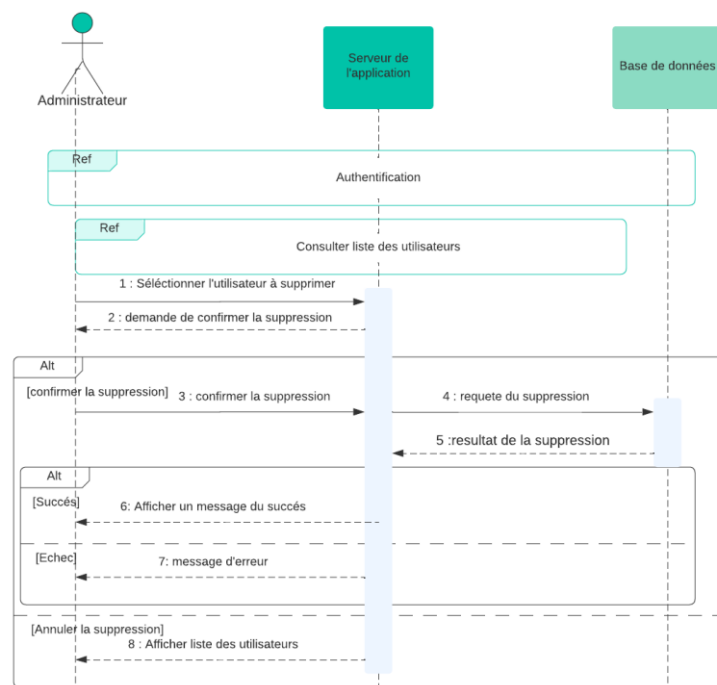


Figure 16:Diagramme de séquence du cas « Supprimer utilisateur »

V. Réalisation

La figure 17 représente l'interface d'inscription pour les utilisateurs, qui doivent saisir leurs données pour pouvoir créer son propre compte.

localhost:4200/choix

Inscription

Inscription

Nom d'utilisateur *
votre nom
Username is required

Email *

Date de naissance *

Téléphone *

Adresse *

Mot de passe *

Choisir un fichier | Aucun fichier choisi

[S'inscrire](#) [Annuler](#) [Se connecter](#)

Figure 17: interface d'inscription d'un utilisateur.

La figure 18 représente l'interface d'authentification pour un administrateur ou un utilisateur qui doivent saisir le mail et le password pour accéder à leurs propres espaces.

localhost:4200/connexion

[Back](#)

Connexion

adresse e-mail

mot de passe

[Se connecter](#)

[S'inscrire maintenant](#)

Figure 18: Interface d'authentification.

La figure 19 représente l'interface d'accueil pour l'espace de l'administrateur qui est un espace dédié seulement à l'administrateur.

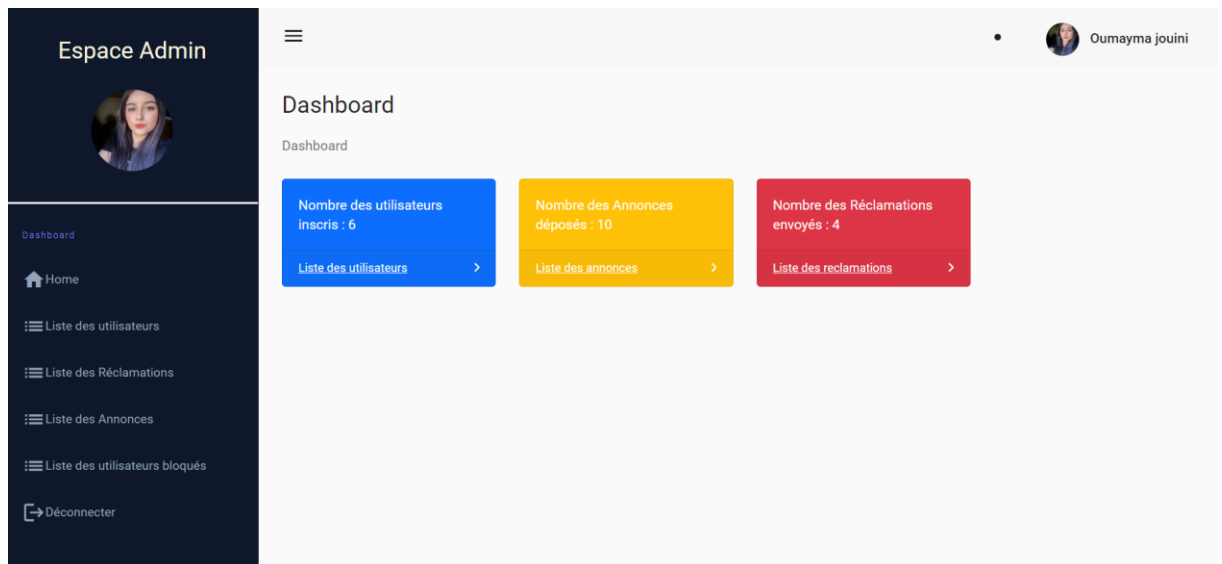


Figure 19: Interface d'accueil « Administrateur ».

Cette fenêtre est gérée par l'administrateur. En fait, à travers cette interface, l'administrateur pourra gérer son profil, consulter la liste des utilisateurs nouveaux-inscrits, les supprimer, les bloquer, même ce dernier pourra consulter la liste des utilisateurs bloqués et les débloquent et se déconnecter.

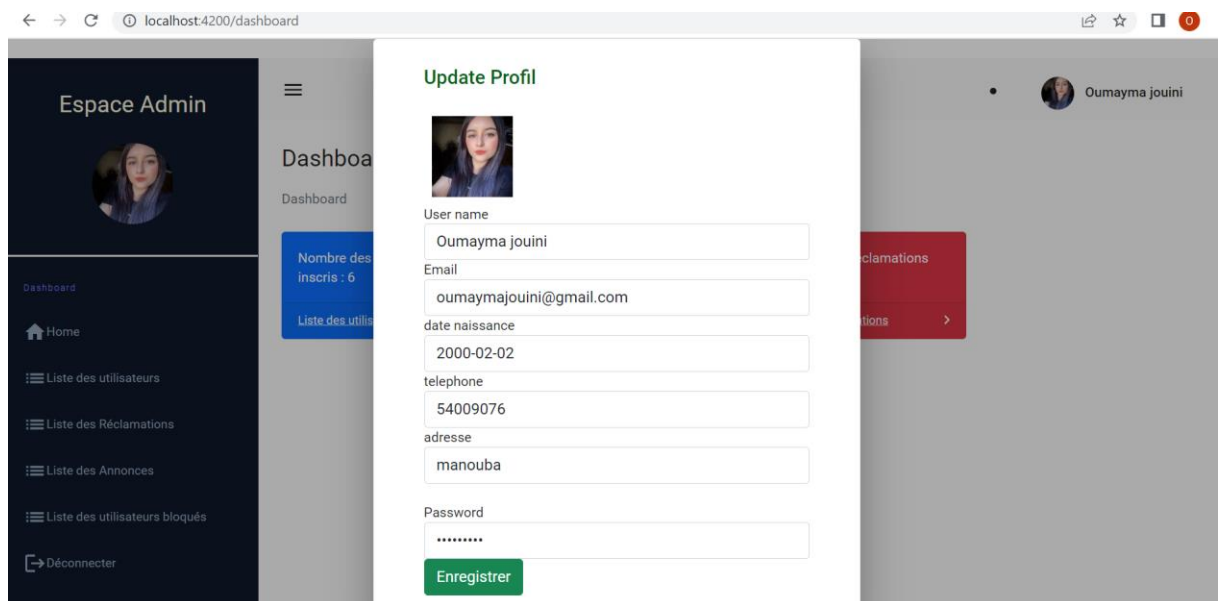


Figure 20: Modifier son profil

Search

[< Back](#)

Liste des utilisateurs





ID	Nom d'utilisateur	Email	Etat	Telephone	Image	Actions
1	Oumayma jouini	oumaymajouini@gmail.com	nonbloqué	54009076		Supprimer Bloquer
2	kamel nefzi	kamel@gmail.com	nonbloqué	56789876		Supprimer Bloquer
3	loulaifa selmi	loulou@gmail.com	nonbloqué	23566876		Supprimer Bloquer
5	khalil taboubi	khalil@gmail.com	nonbloqué	54678987		Supprimer Bloquer

Figure 21: Consulter liste utilisateurs

Search

[< Back](#)

Liste des personnes bloquées

ID	Nom d'utilisateur	Email	Role	Telephone	Etat	Actions
4	intidhar jouini	intidhar@gmail.com	Client	27654543	bloqué	Débloquer
5	khalil taboubi	khalil@gmail.com	Client	54678987	bloqué	Débloquer
7	amen jouini	amen@gmail.com	Professionnel	23345678	bloqué	Débloquer

Figure 22: interface représentant la liste des utilisateurs bloqués

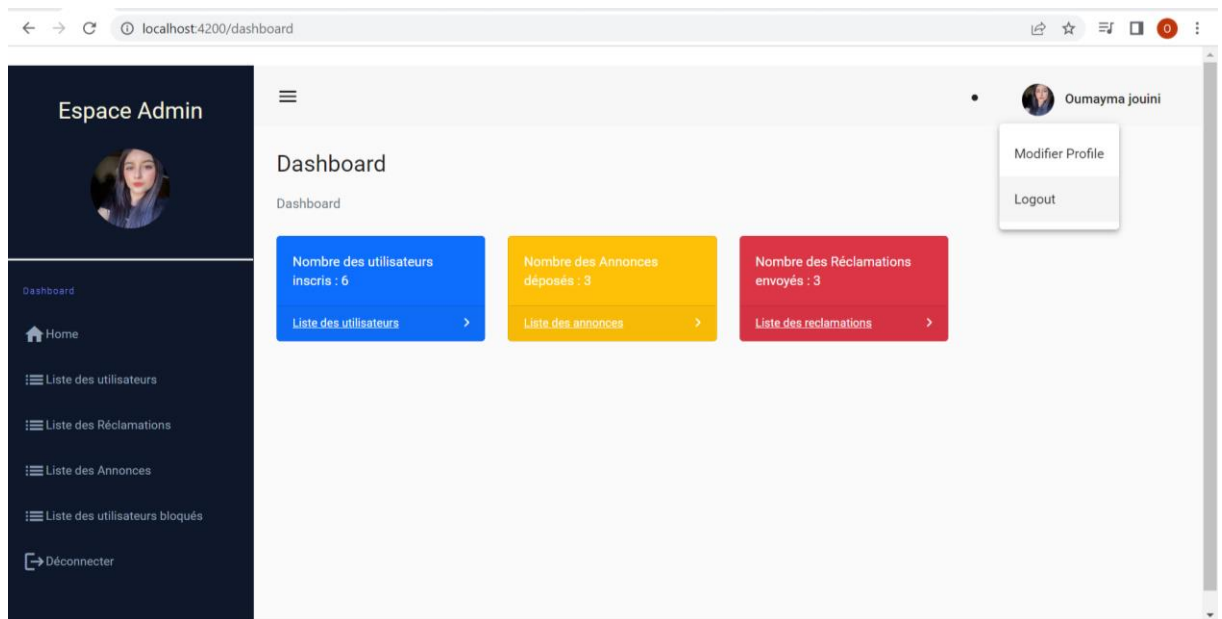


Figure 23: bouton « se déconnecter »

La figure 24 représente l'interface d'accueil pour l'espace du Client qui est un espace dédié seulement au Client.

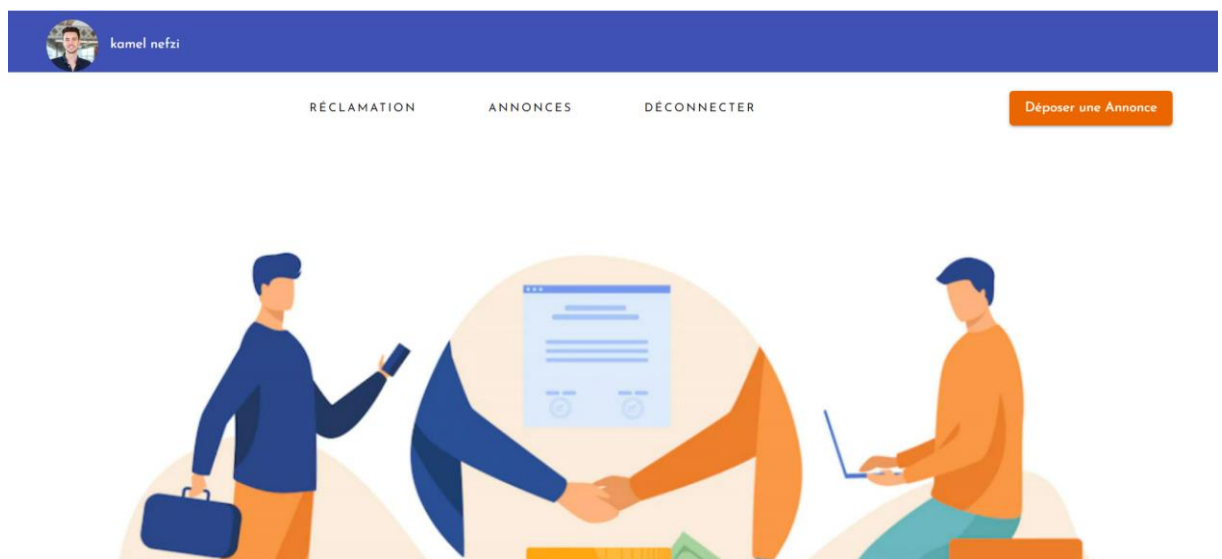



Figure 24: Interface d'accueil d'un client

À travers la figure 25, un client pourra gérer son profil.

localhost:4200/dashclient

kamel nefzi

Update Profil



User name
kamel nefzi

Email
kamel@gmail.com

date naissance
2022-05-18

telephone
56789876

adresse
manouba

Password
.....

Enregistrer

Figure 25: Modifier son profil

Conclusion

Dans ce chapitre nous avons mis l'accent sur la réalisation de sprint 1, nous avons, en premier lieu définir les user story à réaliser dans ce sprint, ensuite nous avons réalisé les diagrammes de cas d'utilisation, de classes et de séquences. Finalement, nous avons illustré les imprimés écrans des interfaces créés durant ce sprint, au niveau de la partie réalisation.

Chapitre 4 : Sprint 2 – Gestion des Annonces

Introduction

Dans ce chapitre nous présentons l'analyse, la conception et la réalisation du deuxième sprint de notre application. Ce sprint a pour objectif de permettre à l'administrateur de gérer les utilisateurs, gérer les annonces déposées, et au client de gérer ses annonces et consulter la liste des autres annonces, enfin on décrit notre application avec des captures d'écran.

I. Backlog Sprint

Les différentes tâches liées à la réalisation du deuxième Sprint sont décrites dans le tableau suivant :

Tableau 8 : Backlog Sprint du Sprint n°2

ID	User Story	Tâches
1	En Tant que Client, je dois pouvoir gérer mes annonces	<ol style="list-style-type: none">1. Création de la table Annonce dans la base de données.2. Effectuation de la relation entre la table « User » et la table « Annonce »3. Réalisation des diagrammes de cas d'utilisation, de séquence et de classe détaillés du cas « Gérer annonce ».4. Création du bouton d'ajout d'une annonce.5. Création du formulaire pour l'ajout d'une annonce.6. Création du « mat-dialog » avec lequel, on va afficher le formulaire.7. Code pour l'ajout (Front-end).8. Code pour l'ajout (Back-end).9. Création de l'interface dans laquelle, on va afficher la liste de ses propres annonces.10. Création du bouton supprimer et modifier d'une annonce.11. Code pour la suppression (Front-end).12. Code pour la suppression (Back-end).13. Code pour la modification (Front-end).14. Code pour la modification (Back-end).15. Code pour voir les détails d'une annonce (Front-end).16. Code pour voir les détails d'une annonce (Back-end).17. Créer l'interface qui affiche le formulaire de la modification d'une annonce.18. Test du cas « Ajout annonce ».19. Test du cas « Supprimer annonce ».20. Test du cas « Modifier annonce ».

		21. Test du cas « Détails annonce ».
2	En Tant qu'Administrateur, je dois pouvoir gérer les annonces déposés	22. Réalisation des diagrammes de cas d'utilisation, et de classe détaillée du cas « Gérer annonces ». 23. Code relatif à la liste des annonces (Back-end). 24. Code relatif à la liste des annonces (Front-End). 25. Création de l'interface dans laquelle, on va afficher les Annonces des Clients. 26. Création du bouton Supprimer. 27. Code pour la suppression (Back-end). 28. Code pour la suppression (Front-End). 29. Test du cas « Liste annonces déposés ». 30. Test du cas « Supprimer annonce ».
3	En Tant que Client, je dois pouvoir consulter la liste des autres annonces	31. Code relatif à la liste des annonces (Back-end). 32. Code relatif à la liste des annonces (Front-End). 33. Création de l'interface dans laquelle, on va afficher les Annonces des autres Clients. 34. Test du cas « Liste des autres annonces ».

II. Diagrammes de cas d'utilisation détaillés du Sprint n°2

Nous présentons ci-dessous le diagramme de cas d'utilisation de sprint 2

1. Cas d'utilisation « Gérer ses annonces »

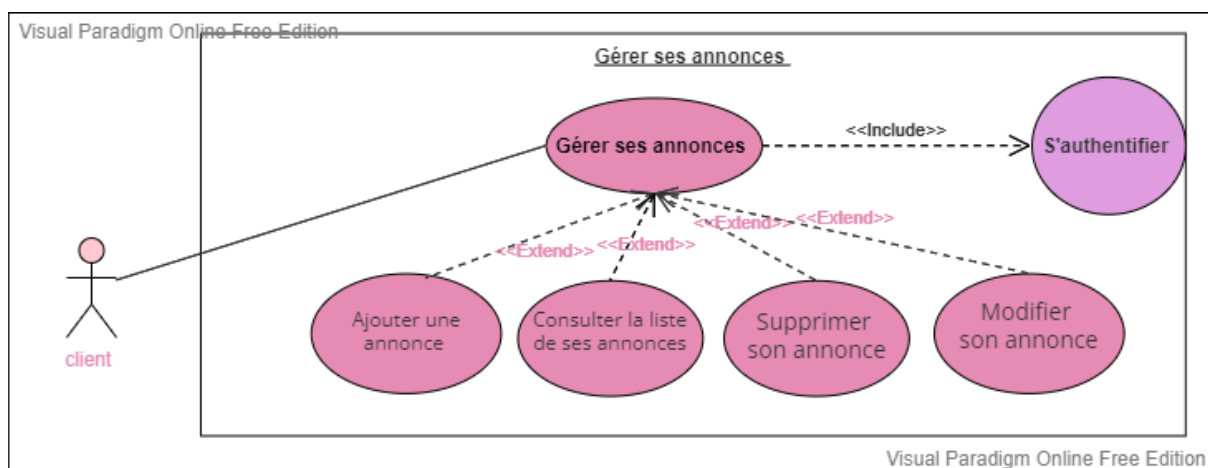


Figure 26:: Diagramme de cas d'utilisation « Gérer ses annonces ».

2. Cas d'utilisation « Gérer annonces »

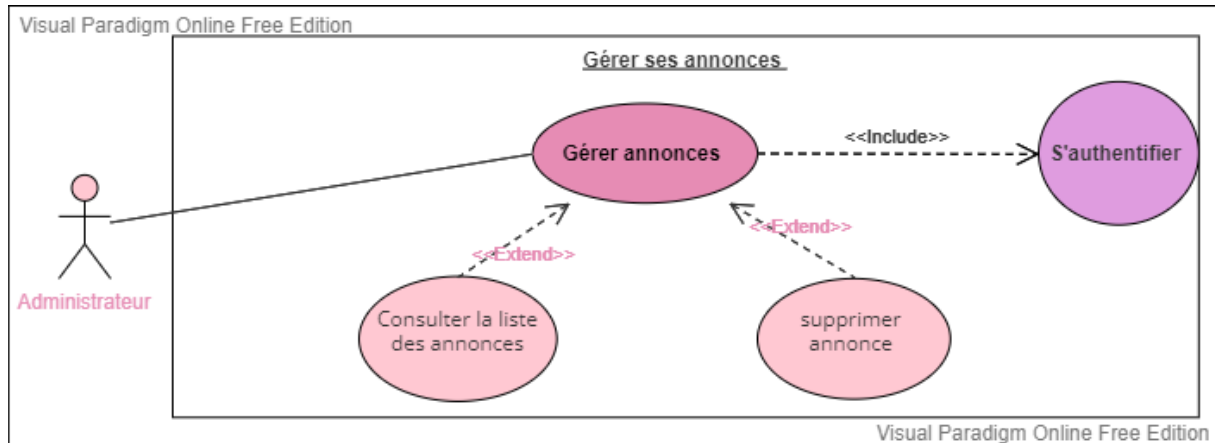


Figure 27: Diagramme de cas d'utilisation « Gérer annonces ».

III. Diagramme de classe du sprint n°2

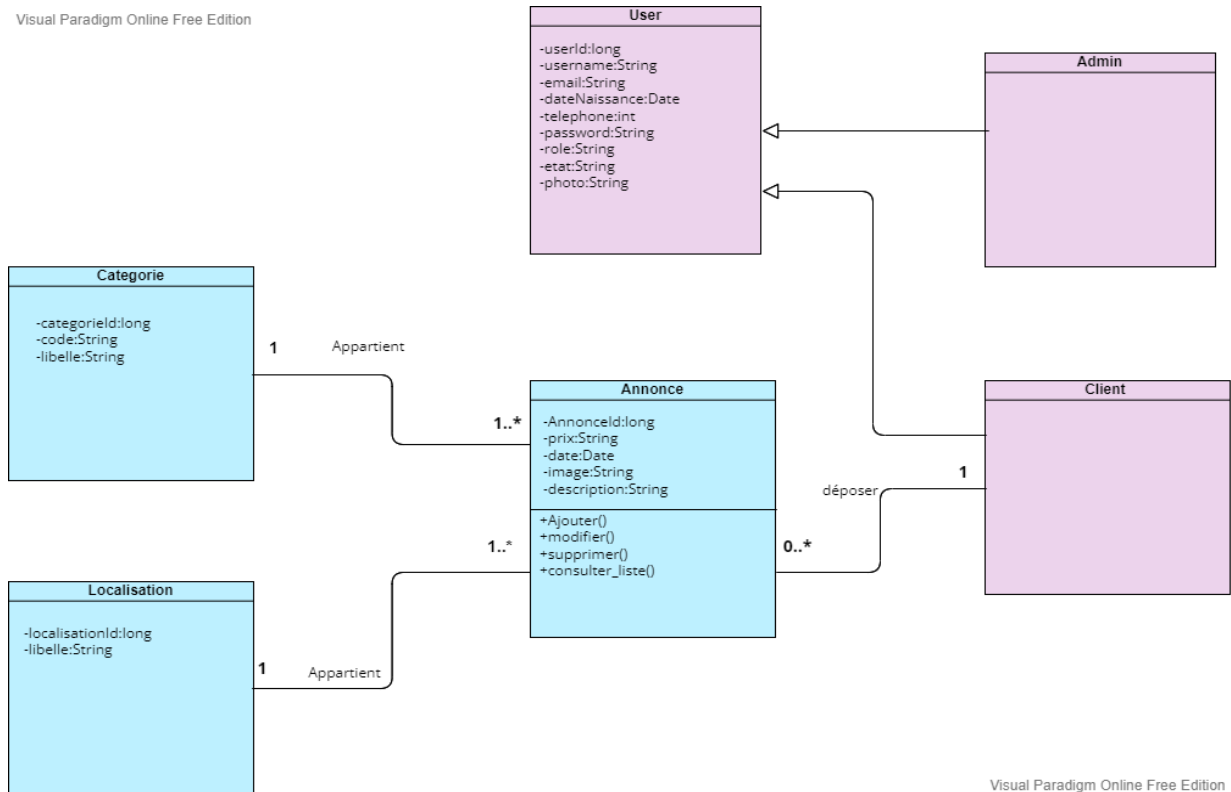


Figure 28:Diagramme de classe du sprint n°2

IV. Diagramme de séquence du sprint n°2

1. Diagramme de séquence du cas « Ajouter une annonce »

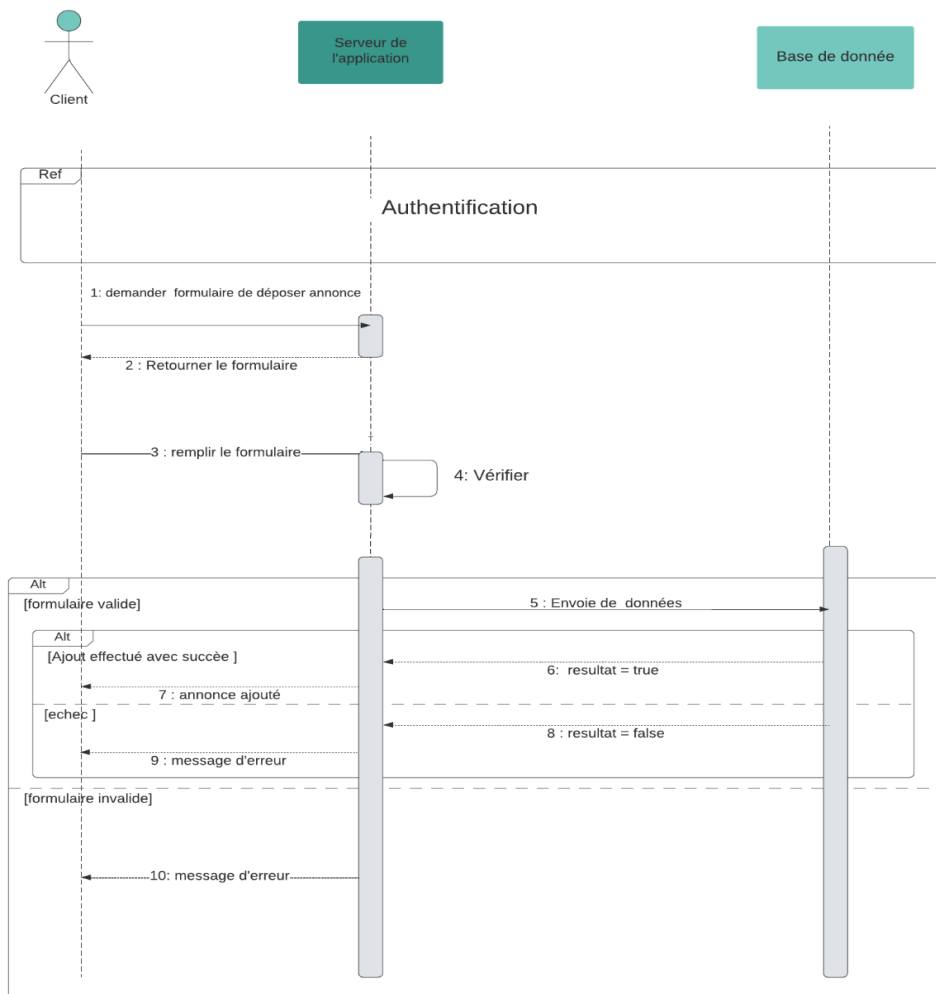


Figure 29:Diagramme de séquence du cas « Ajouter une annonce »

V. Réalisation

A propos la figure 30, un client peut déposer des annonces.

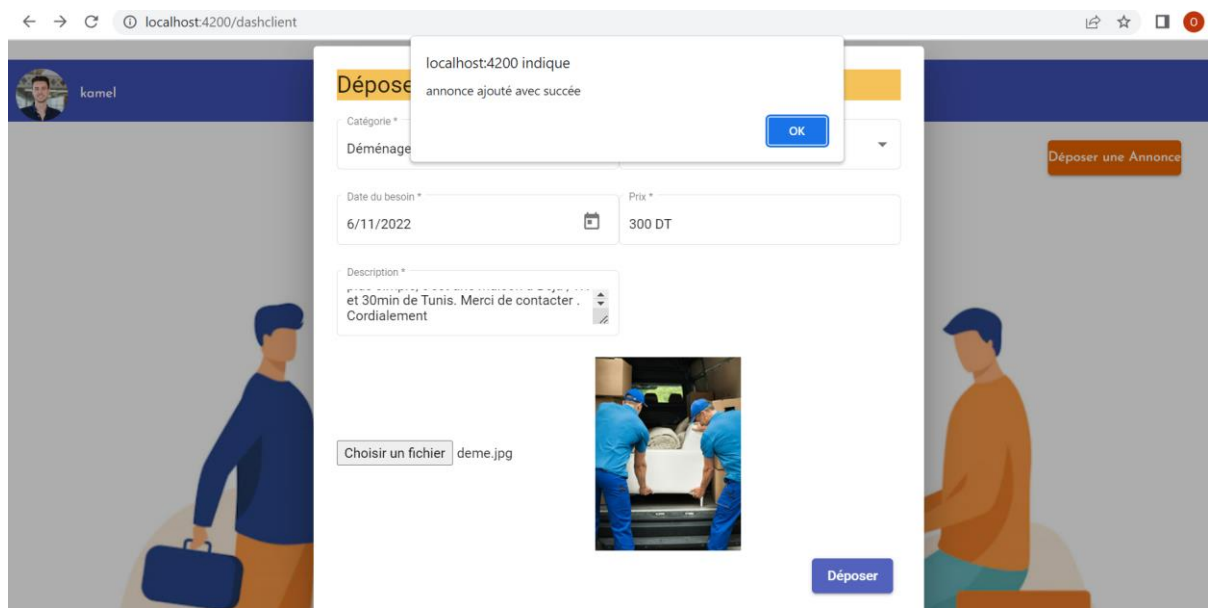


Figure 30: Ajout annonce

A propos les figures 31,32 ,33, un client peut consulter la liste de ses annonces déposées, il peut les consulter, les modifier et les supprimer.

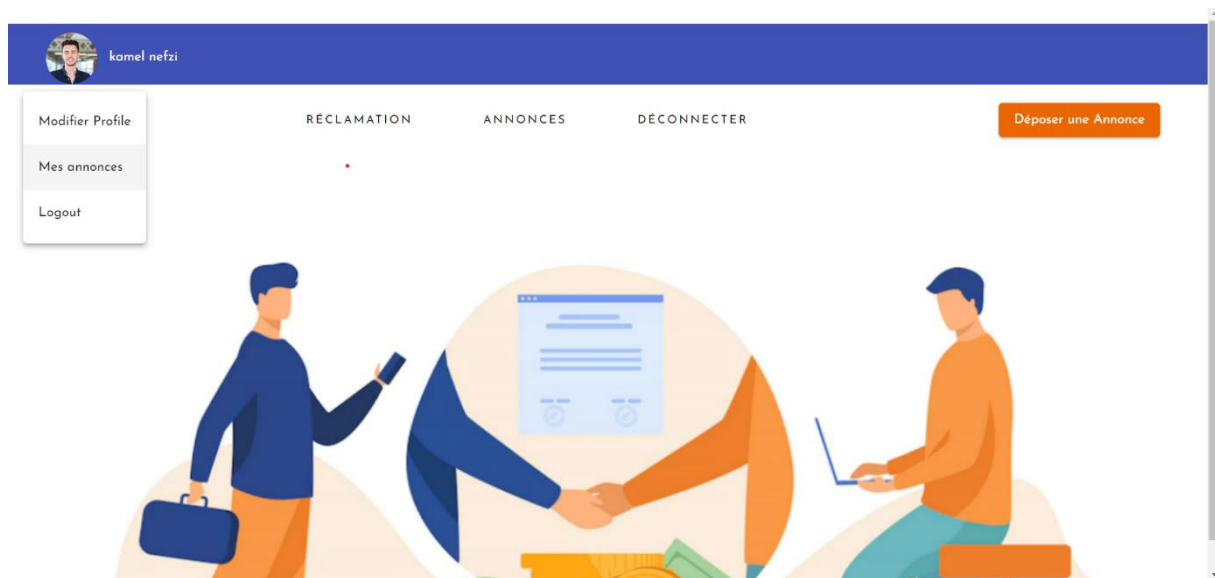


Figure 31:interface pour consulter ses annonces

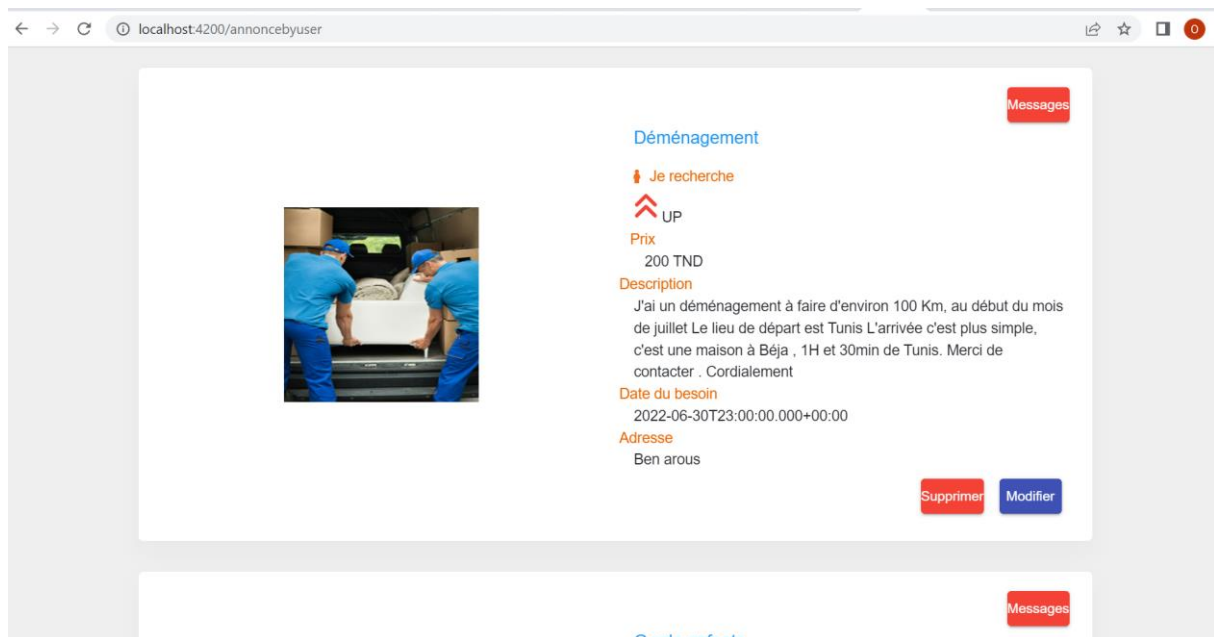


Figure 32: Liste de ses annonces

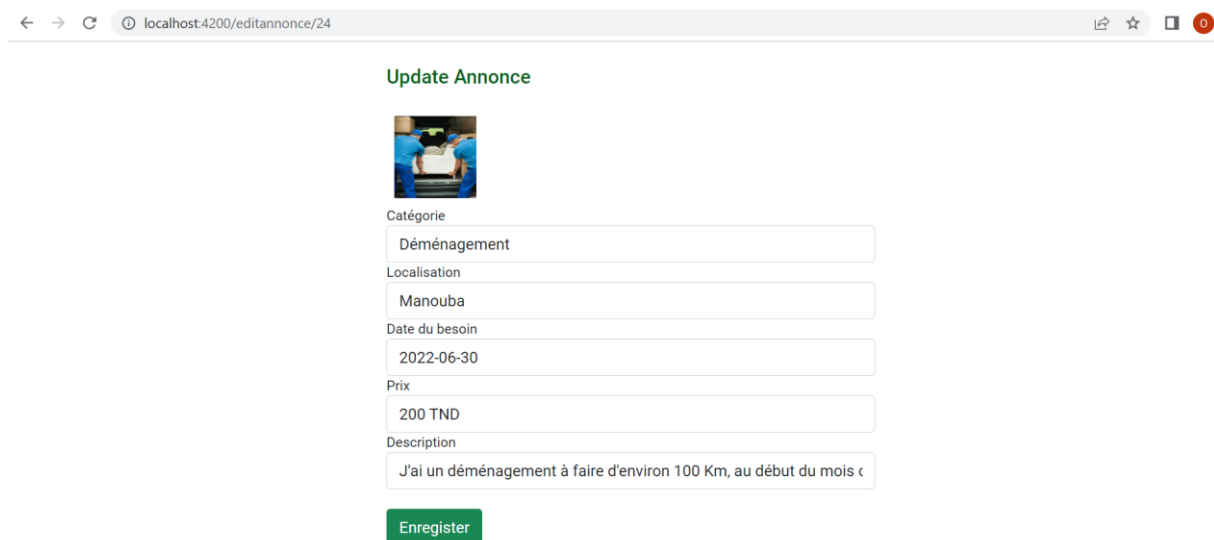


Figure 33: interface de modifier son annonce

A propos la figure 34,35 ,36, l'administrateur peut consulter la liste des annonces déposés par les clients, il peut les consulter, voir les détails et les supprimer.

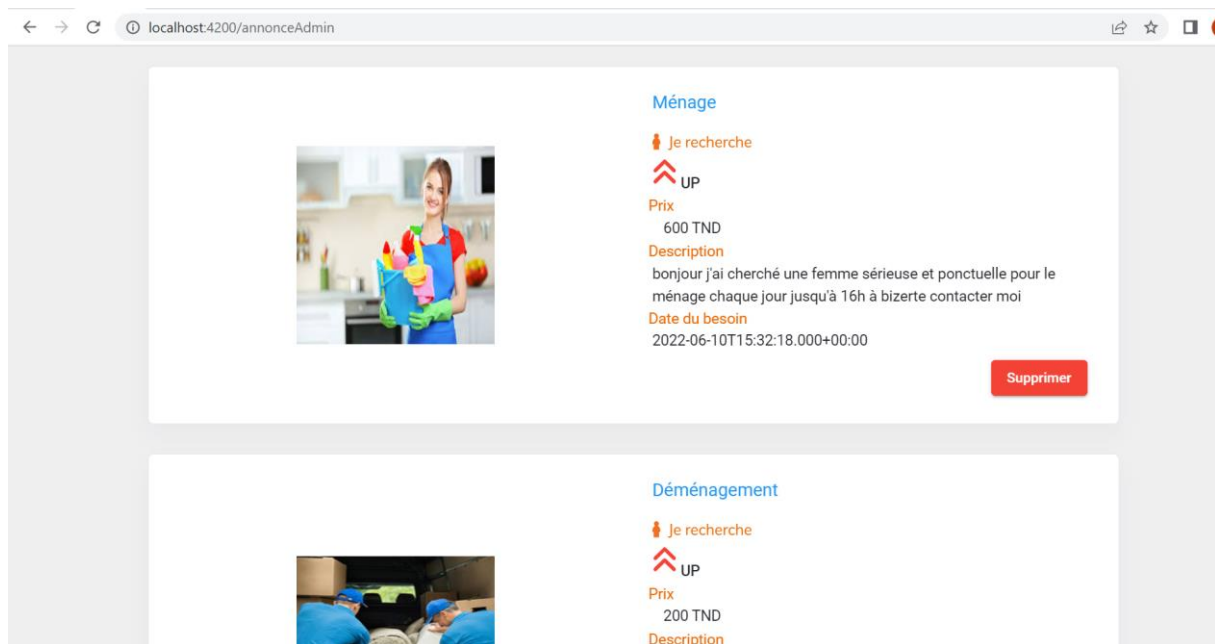


Figure 34: interface représentant la liste des annonces de clients

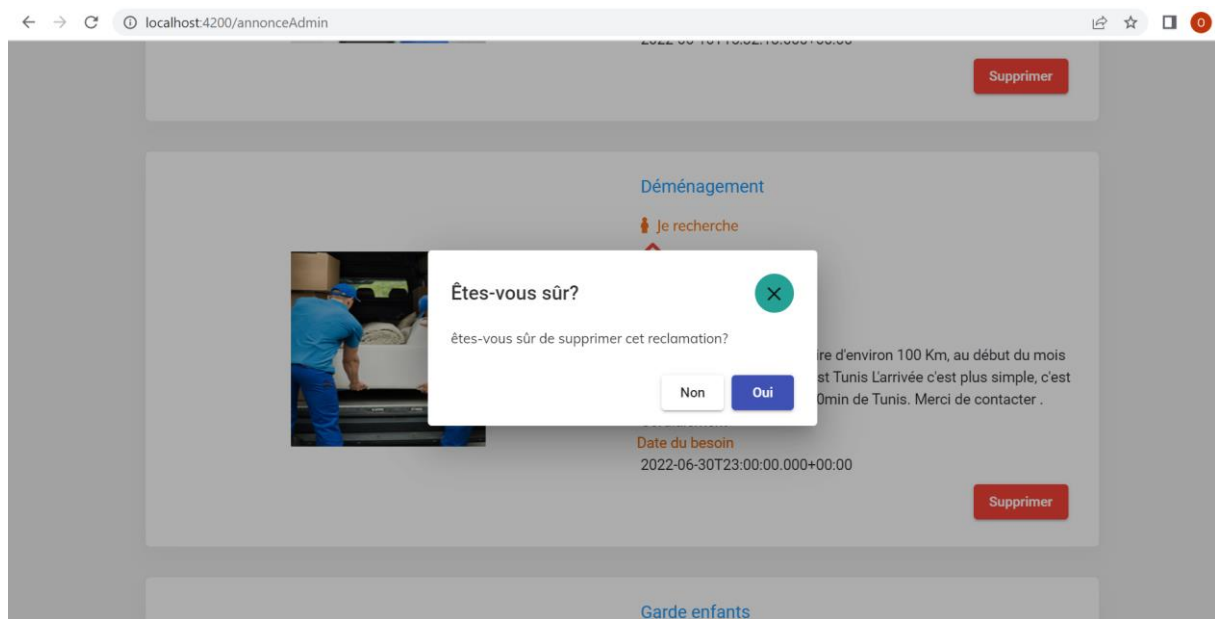


Figure 35: interface pour la suppression des annonces

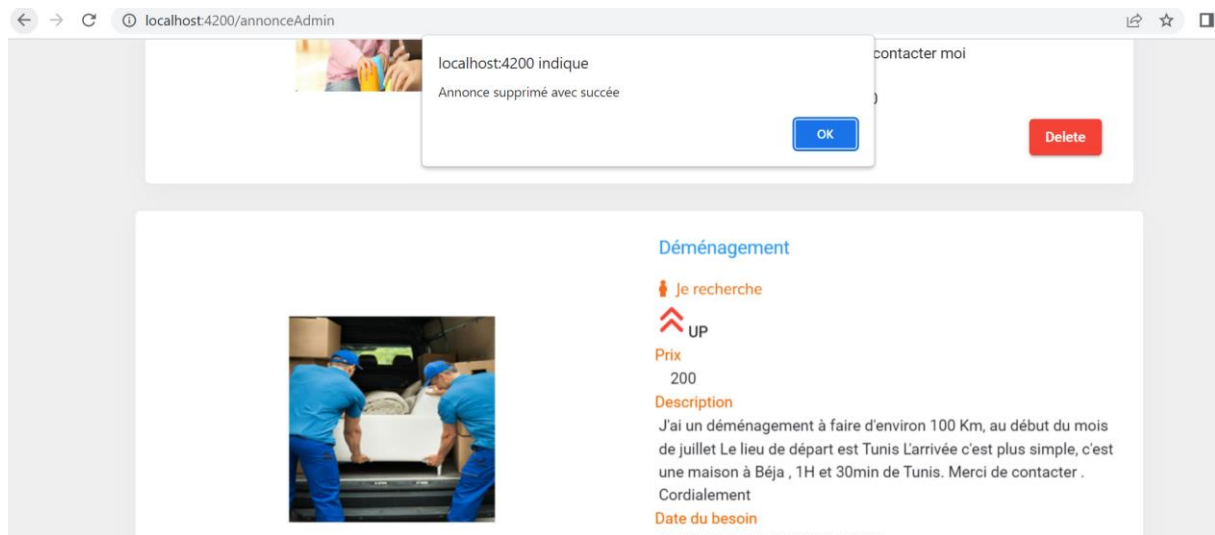


Figure 36:Supprimer Annonce

La figure 37 représente la liste des annonces déposées par les autres clients.

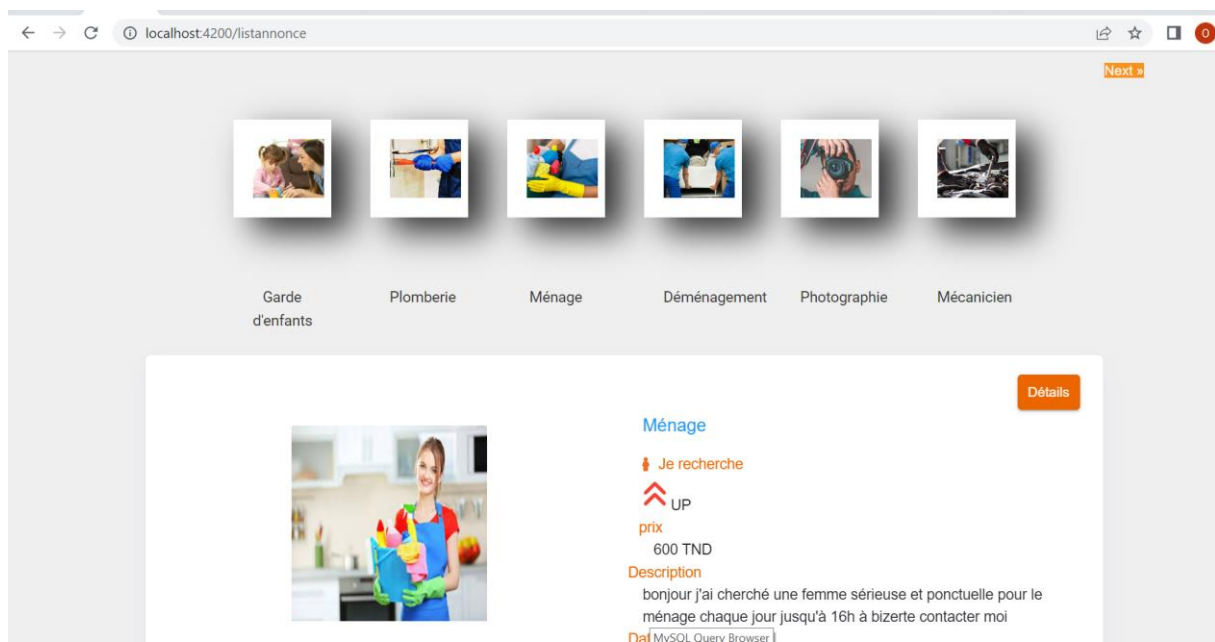


Figure 37:Interface représentant la liste des autres annonces

La figure 38 représente les détails d'une annonce.

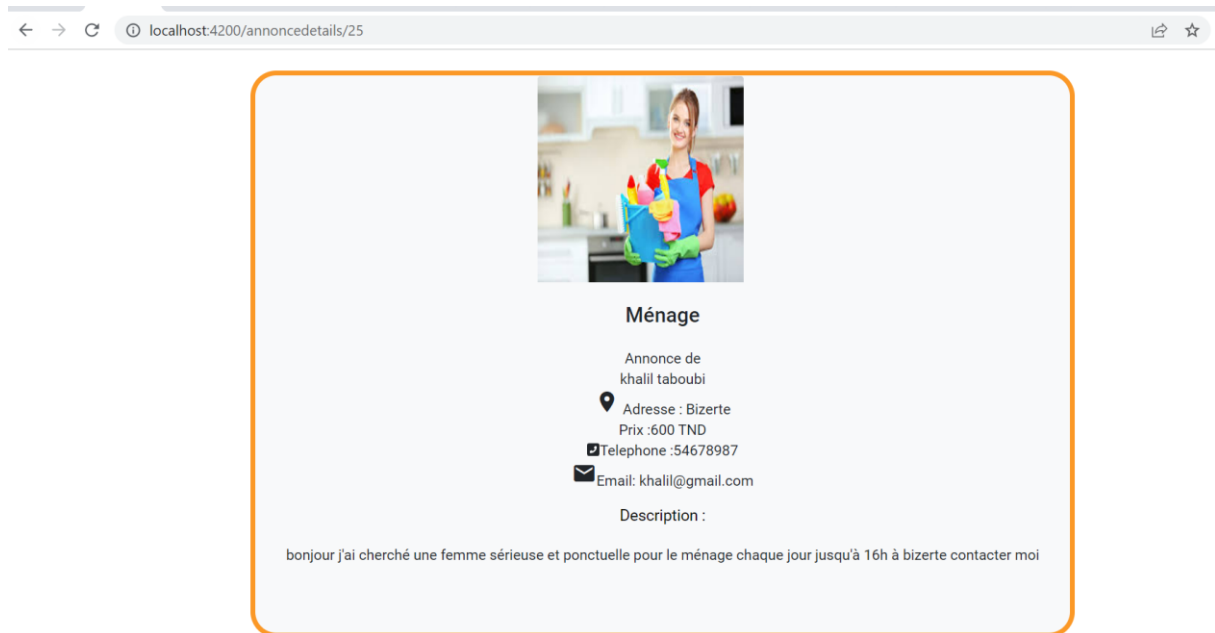


Figure 38:Détails annonce

Conclusion

Dans le chapitre qui suit, nous allons entamer le Sprint n°3 qui est dédié à la Gestion des Réclamations des utilisateurs.

Chapitre 5 : Sprint 3 – Gestion des Réclamations

Introduction

I. Backlog Sprint

Les différentes tâches liées à la réalisation du 3 Sprint sont décrites dans le tableau suivant :

Tableau 9: Backlog Sprint du Sprint n°3

ID	User Story	Tâches
1	En tant que Client/Professionnel je dois pouvoir faire des réclamations.	<ol style="list-style-type: none">1. Création de la table réclamation dans la base de données.2. Effectuation de la relation entre la table « User » et la table « Réclamation »3. Réalisation des diagrammes de séquence et de classe détaillés du cas « Faire réclamation ».4. Création du formulaire pour faire une réclamation.5. Création du « mat-dialog » avec lequel, on va afficher le formulaire.6. Code pour l'ajout d'une réclamation (Front-end).7. Code pour l'ajout d'une réclamation (Back-end).8. Test du cas « Ajout réclamation ».
2	En tant que Admin je dois pouvoir gérer les réclamations des utilisateurs.	<ol style="list-style-type: none">9. Réalisation des diagrammes de cas d'utilisation, de séquence et de classe détaillée du cas « Gérer réclamations ».10. Code relatif à la liste des réclamations (Back-end).11. Code relatif à la liste des réclamations (Front-End).12. Création de l'interface dans laquelle, on va afficher les réclamations des utilisateurs.13. Test du cas « Liste réclamations ».
3	En tant que Admin je dois pouvoir répondre aux réclamations.	<ol style="list-style-type: none">14. Réalisation des diagrammes de cas d'utilisation, de séquence et de classe détaillée du cas « Gérer réclamations ».15. Code de configuration Spring-Mail (Back-end).16. Création du bouton Répondre.17. Code pour répondre par mail (Back-end).18. Code pour répondre par mail (Front-End).19. Création du formulaire pour répondre aux réclamations.20. Création du « mat-dialog » avec lequel, on va afficher le formulaire.21. Test du cas « Répondre réclamation par mail ».

II. Diagrammes de cas d'utilisation détaillés du Sprint n°3

Nous présentons ci-dessous le diagramme de cas d'utilisation de sprint 3

1. Cas d'utilisation « Gérer réclamation »

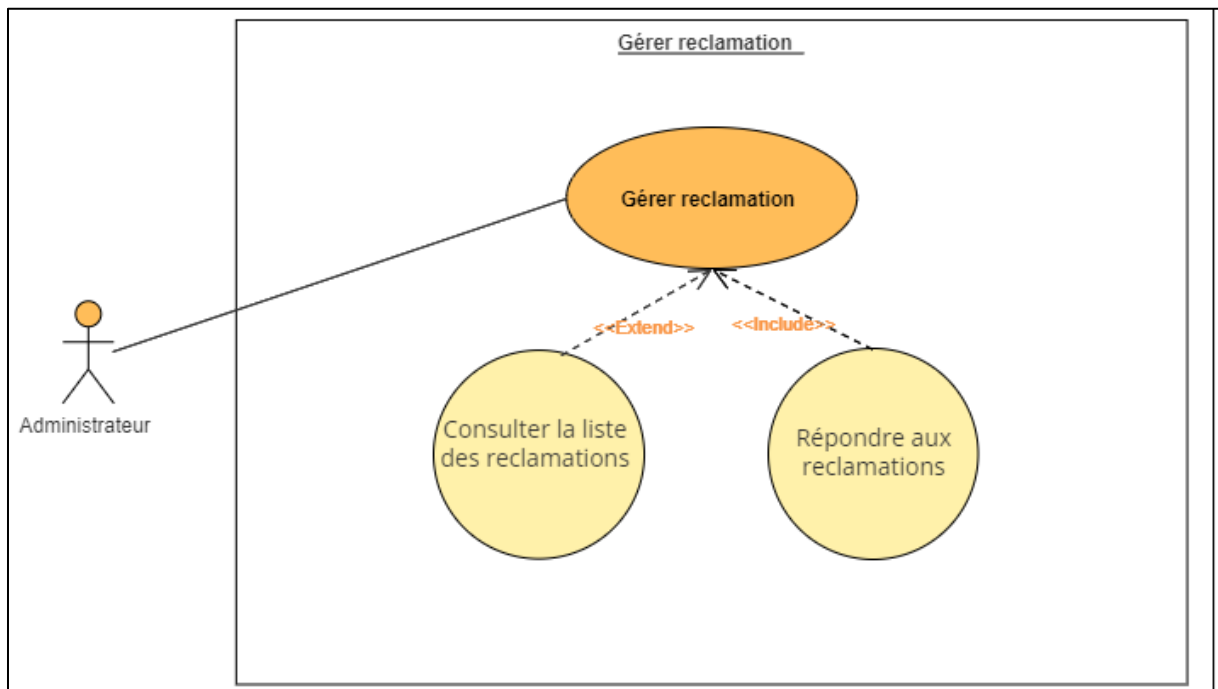


Figure 39: Cas d'utilisation « Gérer réclamation »

III. Diagramme de classe du sprint n°3

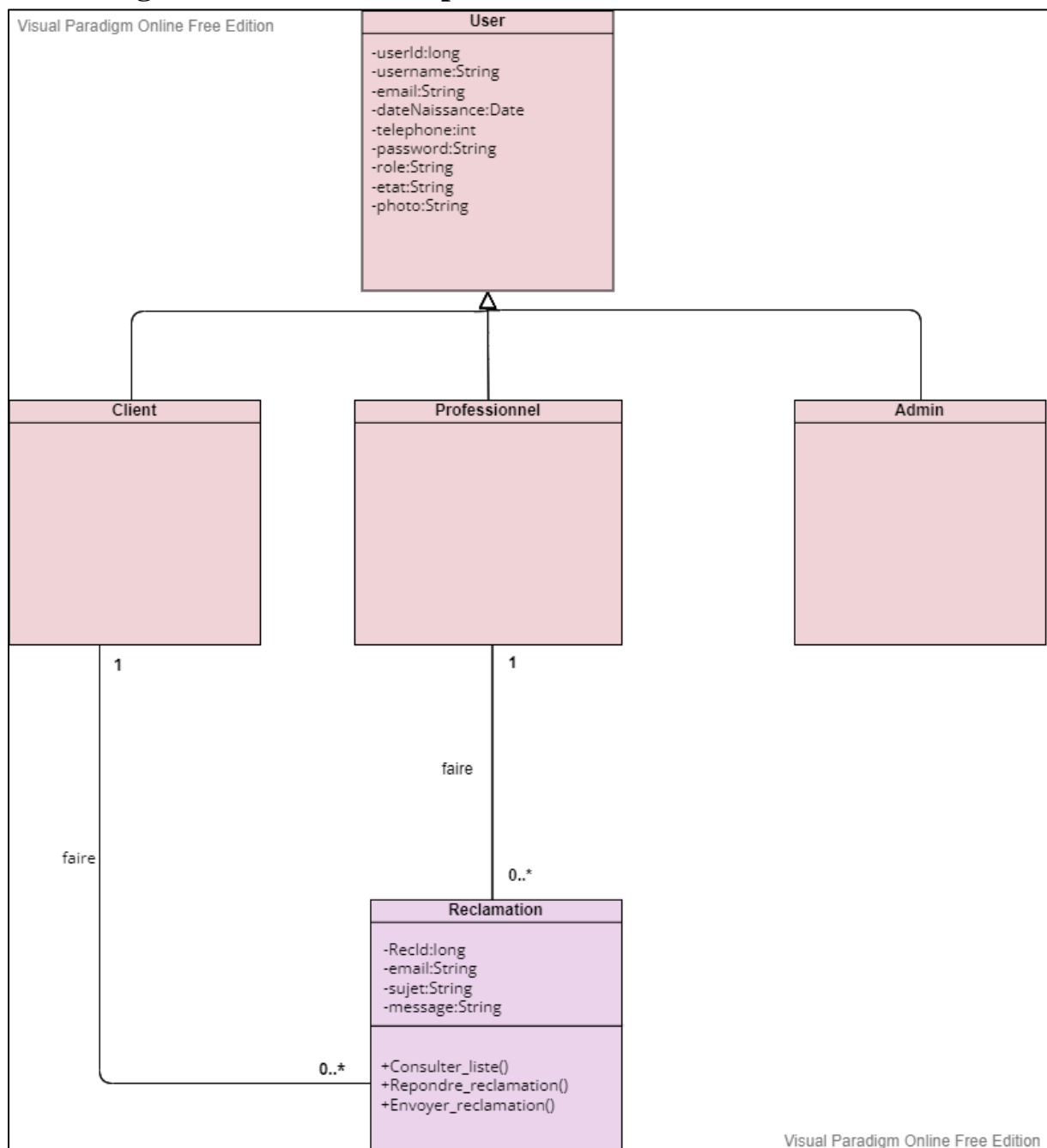


Figure 40:Diagramme de classe du sprint n°3

IV. Diagrammes de séquence du Sprint n°3

1. Diagramme de séquence du cas « Envoyer réclamation »

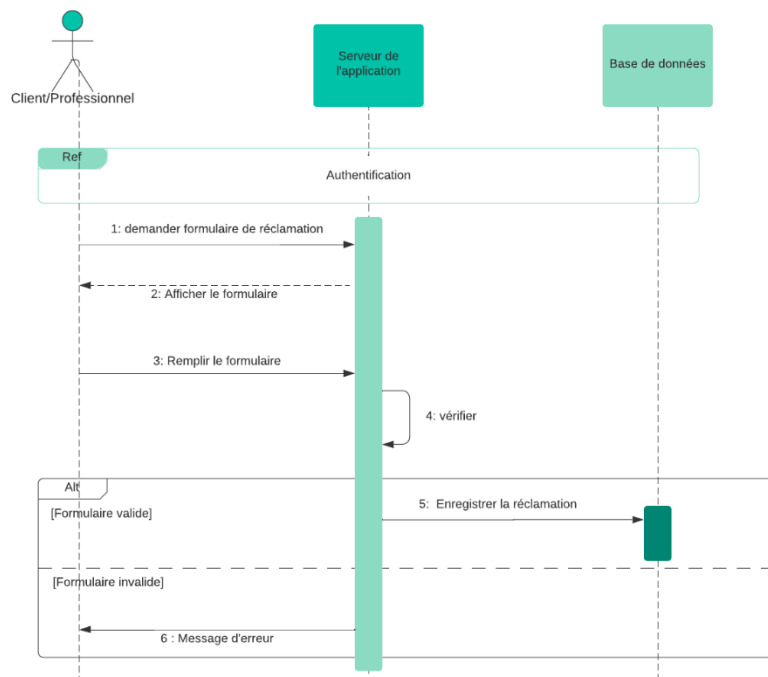


Figure 41: Diagramme de séquence du cas « Envoyer réclamation »

2. Diagramme de séquence du cas « Consulter liste des réclamations »

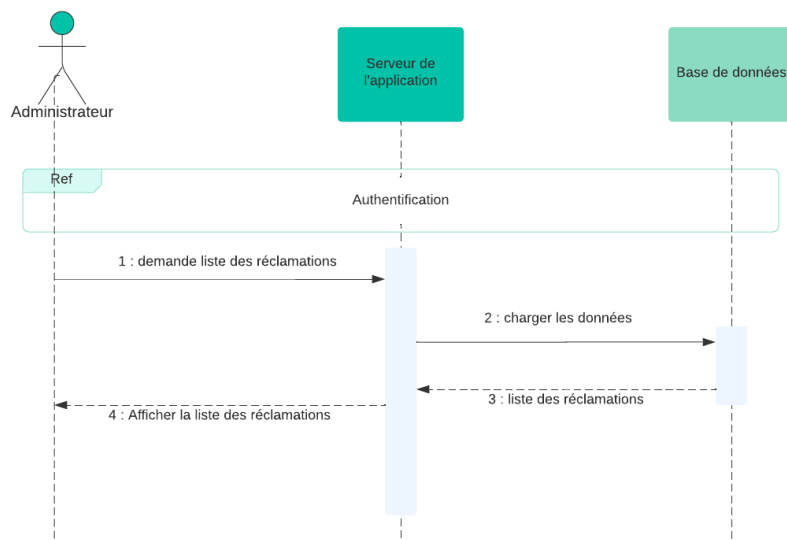


Figure 42: Diagramme de séquence du cas « Consulter liste des réclamations »

3. Diagramme de séquence du cas « Répondre au réclamation »

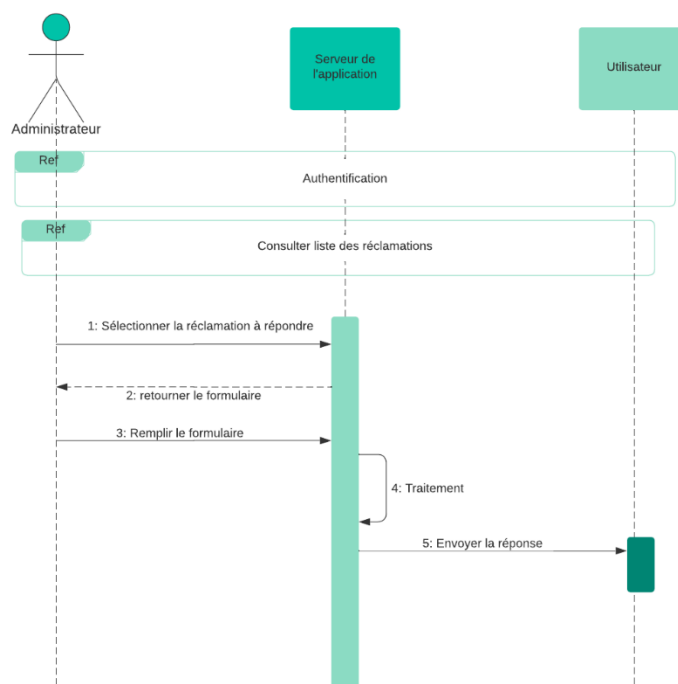


Figure 43:Diagramme de séquence du cas « Répondre au réclamation »

V. Réalisation

Les figures 44,45 représente le formulaire de réclamation.

The screenshot shows a web interface with a dark blue header. On the left, a user profile for 'kamel' is visible. The main content area has a light gray background with a large illustration of two people. A white modal form titled 'Réclamation' is centered on the screen. The form contains three input fields: 'Email *' (with a red border and a red error message 'Email is required' below it), 'Sujet *', and 'Message *' (with a text area and a small icon). At the bottom of the form are two buttons: 'Fermer' (red) and 'Envoyer' (blue). In the top right corner of the page, there is a button labeled 'Déposer une Annonce'.

Figure 44:Formulaire de réclamation

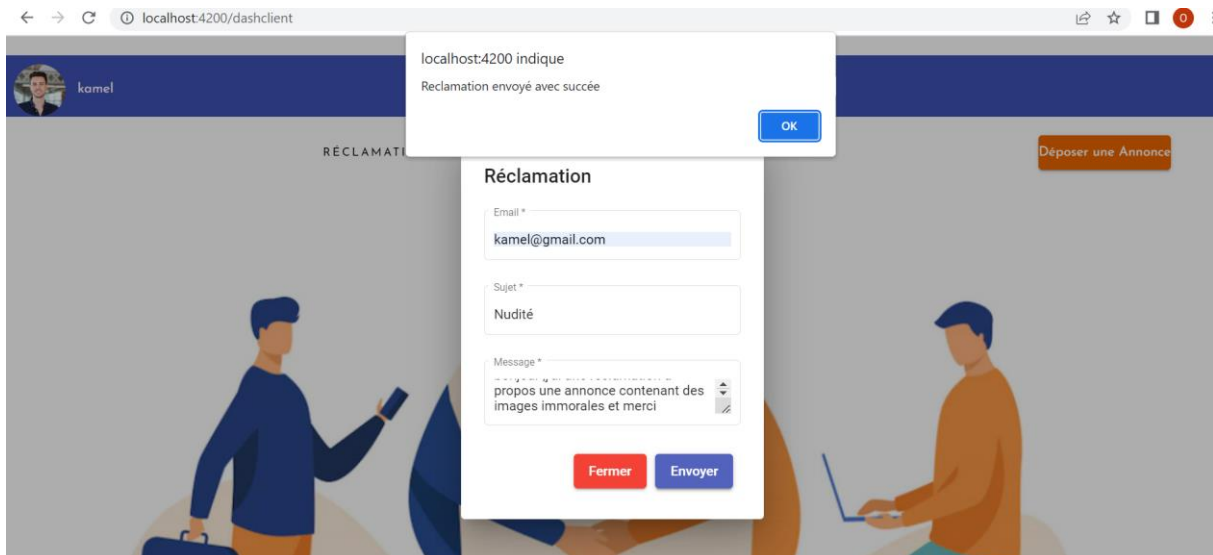


Figure 45:Envoyer réclamation

La figure 46, représente la liste des réclamations envoyés par les utilisateurs.

localhost:4200/listreclamation

< Back

Liste des réclamations					
ID	Email	Sujet	Message	Etat	Actions
20	khalil@gmail.com	Fraude	bonsoir j'ai une plainte concernant un professionnel (mécanicien) pas sérieux .	traité	Répondre
17	kamel@gmail.com	Nudité	bonjour ,j'ai une réclamation à propos une annonce contenant des images immorales et merci	non traité	Répondre
16	selmiloulaifa@gmail.com	Confiance	bonjour ,j'ai une réclamation a propos une femme de ménage qui m'a contacter puis elle venue chez moi et m'a volé merci d'avance	traité	Répondre

Figure 46:Liste des réclamations

A propos la figure 47, l'administrateur appui sur le bouton répondre pour répondre aux réclamations reçues par mail.

Envoie Email

Email

selmiloulaifa@gmail.com

Objet

Confiance

Message

Enregister

Figure 47: Envoyer Mail

La figure 48 représente un exemple d'un email envoyé par l'administrateur.

réclamation



jouinioumayma721@gmail.com

À selmiloulaifa ▼

bonjour on a pris votre réclamation en considération et merci

...

[Message tronqué] [Afficher l'intégralité du message](#)

Figure 48: Mail envoyé par l'administrateur

Conclusion

Dans le chapitre qui suit, nous allons entamer le Sprint n°4 qui est dédié à la gestion des messages.

Chapitre 6 : Sprint 4 – Gestion des Messages

Introduction

Ce chapitre est consacré pour décrire l'analyse, la conception et la réalisation du quatrième sprint. Ce sprint a pour objectif de la gestion des messages.

I. Backlog Sprint

Les différentes tâches liées à la réalisation du 4 Sprint sont décrites dans le tableau suivant :

Tableau 10 : Backlog Sprint du Sprint n°4

ID	User Story	Tâches
1	En tant que Professionnel, je dois consulter la liste des annonces déposés	<ol style="list-style-type: none">1. Réalisation des diagrammes de cas d'utilisation, de séquence et de classe détaillée du cas « Gérer annonce ».2. Code relatif à la liste des annonces déposés (Back-end).3. Code relatif à la liste des annonces déposés (Front-End).4. Création de l'interface dans laquelle, on va afficher les annonces déposées par les utilisateurs.5. Code pour voir les détails d'une annonce (Back-end).6. Code pour voir les détails d'une annonce (Front-End).7. Test du cas « Liste annonces déposés ».8. Test du cas « détails annonce ».
2	En tant que Professionnel, je dois pouvoir contacter les clients	<ol style="list-style-type: none">9. Configuration de la base de données en temps réel Firebase10. Installer et configurer le SDK Firebase11. Création du bouton Contacter.12. Code relatif pour le chat (Front-End).13. Test du cas « Contacter client ».
3	En Tant que client, je dois gérer les messages	<ol style="list-style-type: none">14. Création du bouton Messages.15. Code relatif pour voir ces messages (Front-End).16. Test du cas « Gérer messages ».

II. Réalisation

1. Configuration de la base de données en temps réel Firebase

Nous avons Connecté sur la [Console Google Firebase](#) et nous cliquons sur « Ajouter un projet ».

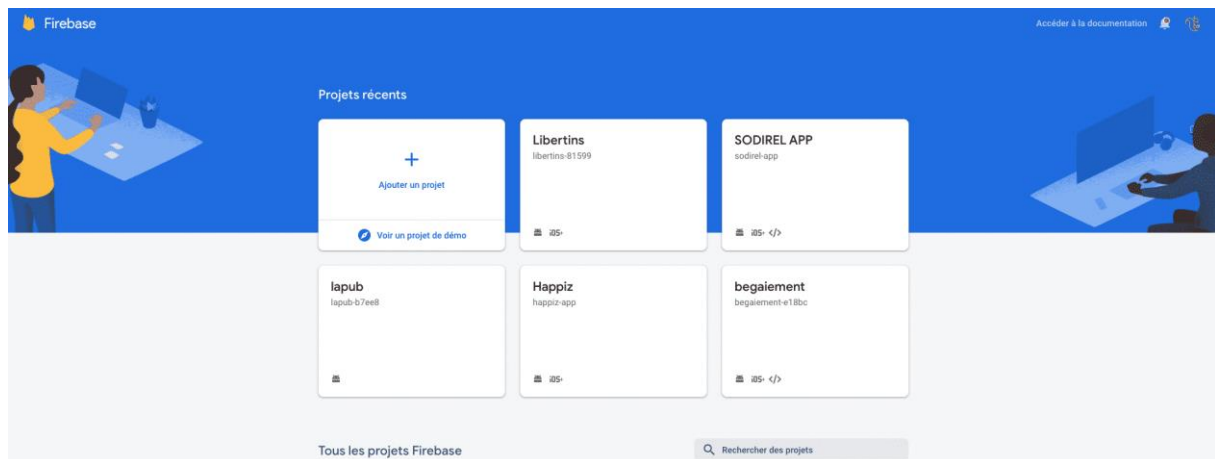


Figure 49: Interface Console Google Firebase

Nous avons créé le nom du projet puis nous avons cliqué sur « Continuer »

 This screenshot shows the first step of creating a project in the Firebase console. The title is "Créer un projet (Étape 1 sur 3)" (Create a project (Step 1 of 3)). The main heading is "Commençons par donner un nom à votre projet" (Let's start by giving a name to your project). Below this, there's a text input field with the placeholder "Saisissez le nom de votre projet" (Enter the name of your project). A sample value "my-awesome-project-id" is shown in a rounded rectangle below the input field. At the bottom, there's a "Continuer" (Continue) button.

Figure 50 : Ajout d'un projet

 This screenshot shows the third step of creating a project in the Firebase console. The title is "Créer un projet (Étape 3 sur 3)" (Create a project (Step 3 of 3)). The main heading is "Configurer Google Analytics" (Configure Google Analytics). Below this, there's a section titled "Sélectionner ou créer un compte Google Analytics" (Select or create a Google Analytics account). It shows a dropdown menu with "Default Account for Firebase" selected. Below the dropdown, there's a "Créer un compte" (Create account) button. At the bottom, there's a "Précédent" (Previous) button and a "Créer un projet" (Create project) button.

Figure 51: interface ajout projet

Après avoir cliqué sur le bouton Continuer, il sera redirigé vers cette page.

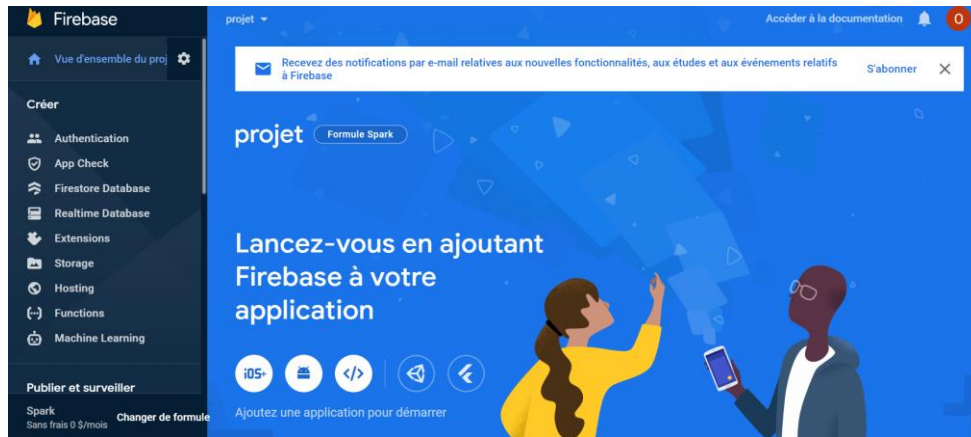


Figure 52: Interface de Firebase

Nous avons cliqué sur créer une base de données

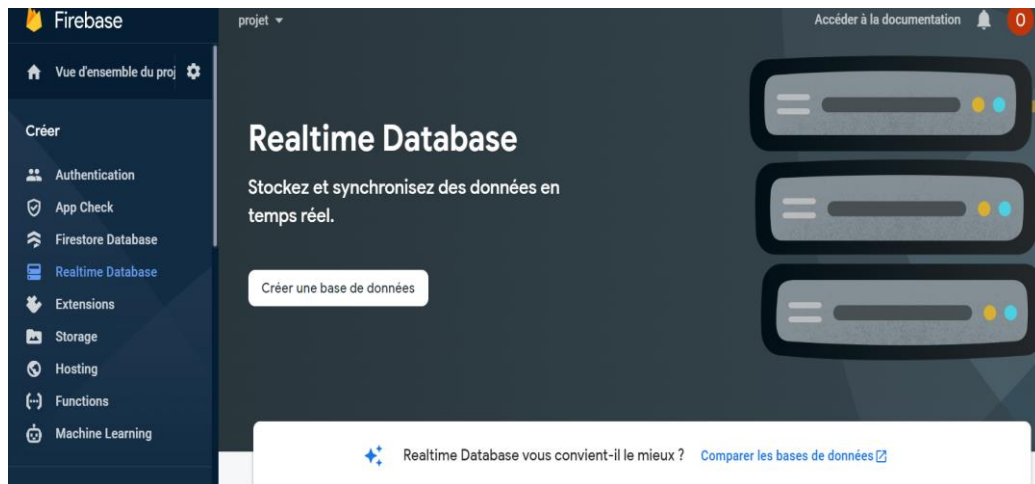


Figure 53: Interface de la création d'une base de données



Figure 54: Interface configuration de la base de données



Figure 55: Aperçu de la base

2. Installation et configuration du SDK Firebase

Nous utiliserons le SDK pour accéder à la base de données en temps réel Firebase. Pour cela, tapez cette commande pour installer le module :

```
PROBLÈMES  SORTIE  CONSOLE DE DÉBOGAGE  TERMINAL

Microsoft Windows [version 10.0.19043.1586]
(c) Microsoft Corporation. Tous droits réservés.

C:\myprojects\Angular>npm install --save firebase
```

Figure 56: Interface de l'installation du module Firebase sur Angular

Ensuite, nous avons enregistré le module Firebase SDK dans l'application Angular en ouvrant et en éditant ce fichier `src/app/app.component.ts` puis nous avons ajouté ces importations de Firebase :

```
import * as firebase from 'firebase';
```

Figure 57: Importation de Firebase

Nous avons déclaré une variable constante pour conserver le paramètre Firebase qui contient la variable de configuration permettant d'accéder à Firebase à l'aide d'apiKey et de databaseURL.

```
apiKey: '',
databaseURL: 'https://oumima-chat-default-rtdb.firebaseio.com'
```

Figure 58: Lien de la base de données

Ensuite, nous avons initialisé les paramètres de configuration de Firebase dans le constructeur du composant angulaire.

```
export class AppComponent {  
  title = 'Angular';  
  constructor() {  
    firebase.initializeApp(config);  
  }  
}
```

Figure 59:Initialisation du paramètre de configuration de Firebase

Maintenant, la base de données en temps réel Firebase est prête à être utilisée.

Les figures 60,61,62 représentent la liste des annonces pour les professionnels pour les consulter, voir plus détails et contactent les clients

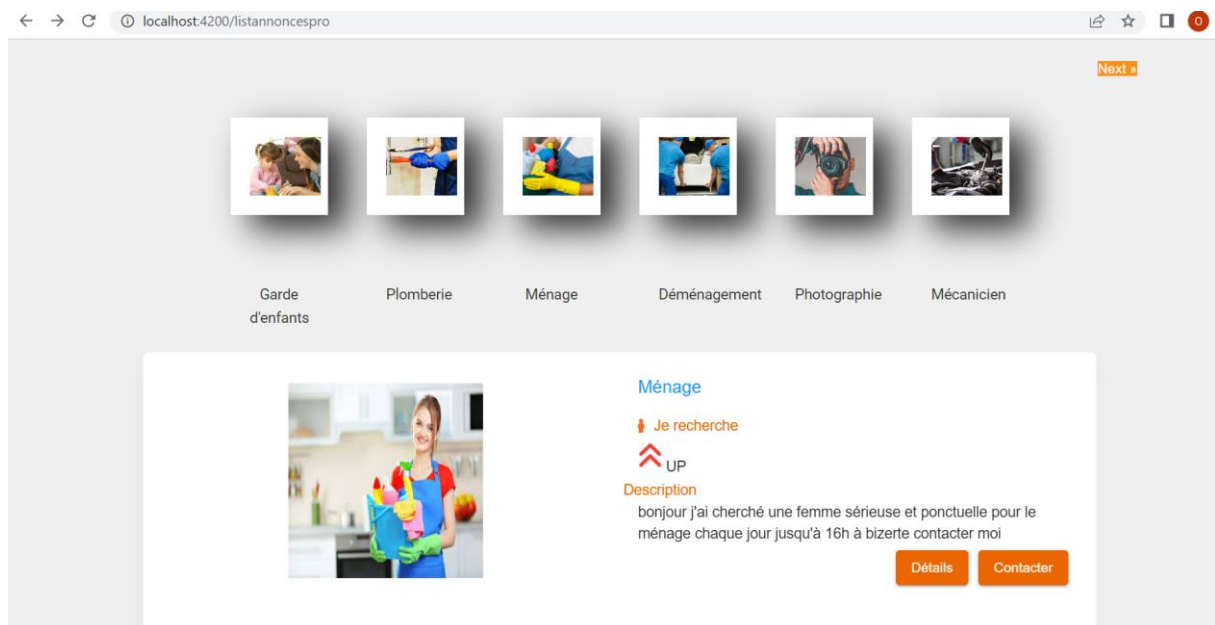


Figure 60:Liste des annonces des clients

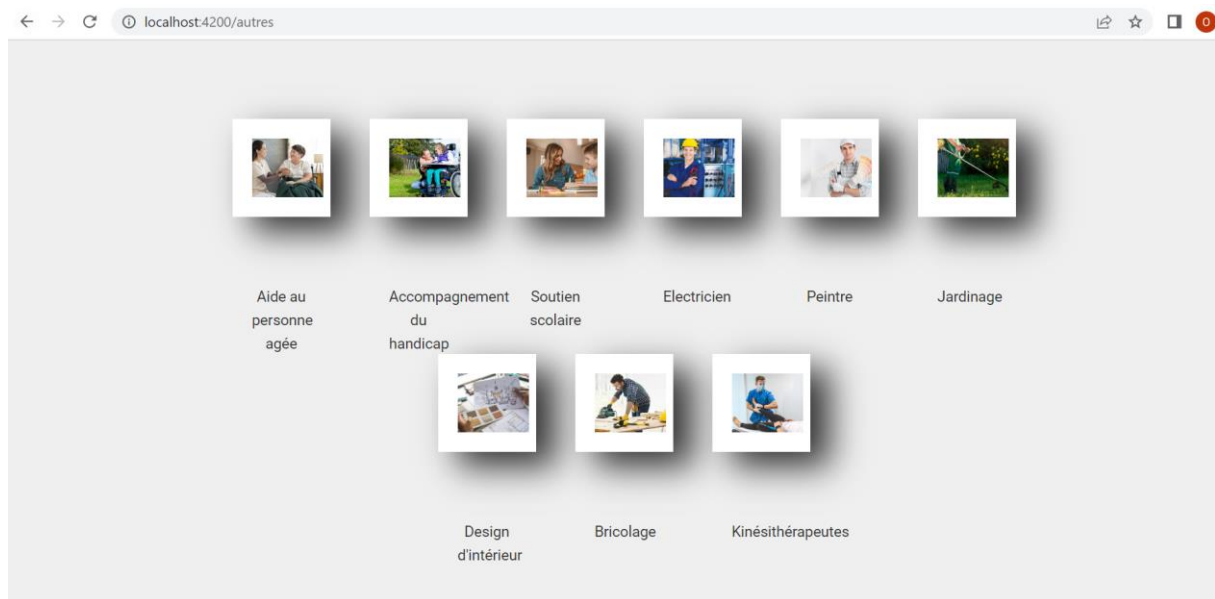


Figure 61: Liste des annonces par catégorie

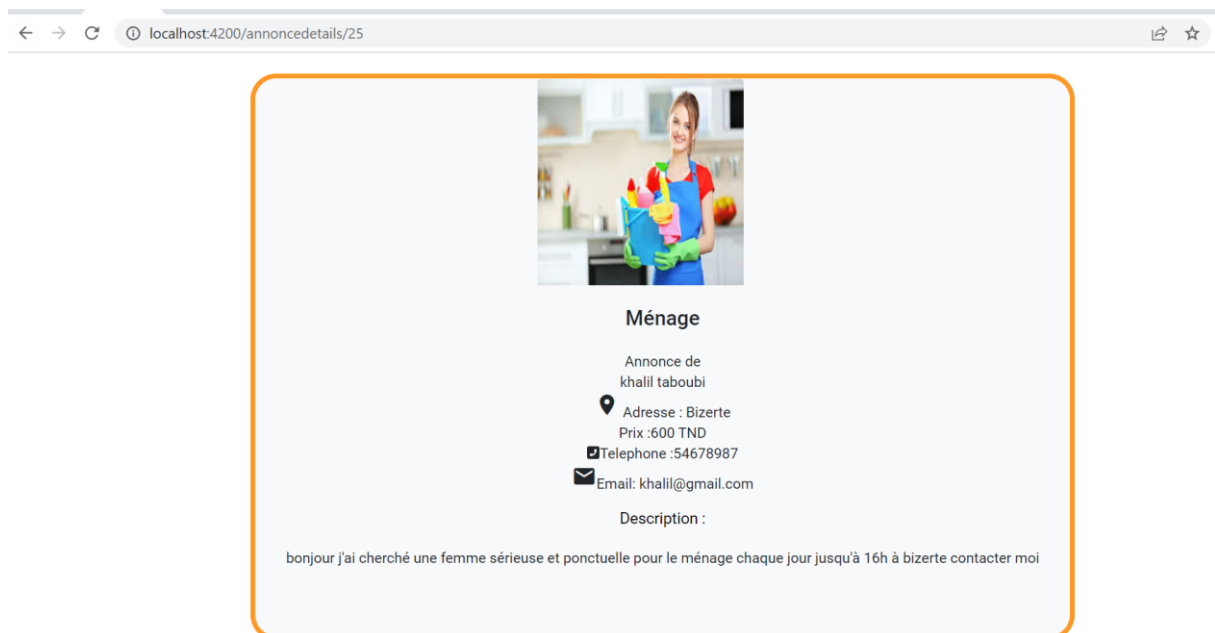


Figure 62: Détails annonce

La figure 63 représente une interface du chat pour les professionnels contactent directement les clients.

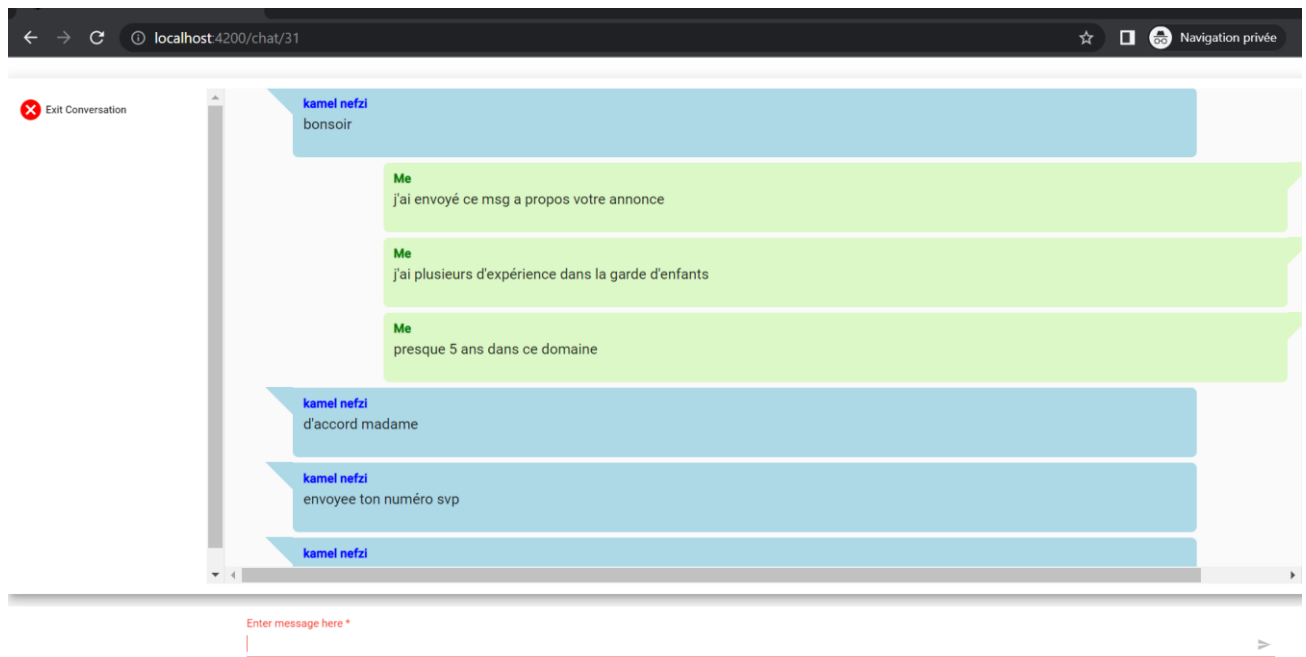


Figure 63: chat pour professionnel

La figure 64 représente une interface du chat pour les clients reçoivent des messages et les répondent.



Figure 64: chat pour clients

Conclusion

Dans ce chapitre nous avons mis l'accent sur la réalisation de sprint 4, nous avons, en premier lieu définir les user story à réaliser dans ce sprint, ensuite nous avons réalisé les diagrammes de cas d'utilisation, de classes et de séquences. Finalement, nous avons illustré les imprimés écrans des interfaces créés durant ce sprint, au niveau de la partie réalisation

Conclusion générale

Ce projet a été réalisé dans le cadre du mémoire de fin d'étude pendant quatre mois de stage au sein de la société « beecoders » de Tunis.

Le présent travail avait comme objectif la conception et la réalisation d'une application web qui fournit des services à domicile et qui permet aux professionnels à trouver des offres d'emploi plus facilement.

Nous avons essayé tout au long de notre travail de construire notre module incrément par incrément en utilisant la méthodologie Scrum.

En ce qui concerne la démarche, j'ai en premier lieu par comprendre le contexte général de notre projet et pourquoi ce projet doit-il être réalisé, ensuite nous avons fait une étude préalable afin de découvrir les techniques et les outils à utiliser, et choisir la méthodologie à suivre. Nous avons passé par la suite à analyser les besoins en déduisant le Backlog Product, la priorisation de ce dernier nous a conduit à la première planification. Nous avons, par ailleurs, étudié les différents Sprints de notre Backlog, pour obtenir une conception qui met l'accent sur l'aspect statique et dynamique du système décrivant minutieusement l'aspect métier de la solution. Ensuite, pour chaque sprint, nous avons abordé l'étape du développement au cours de laquelle nous avons expliqué les principales tâches effectuées, et nous le finissons par une revue et rétrospective.

Finalement, Ce projet nous a donné l'opportunité de manipuler des techniques innovantes et évolutives et nous a permis aussi de tester et d'appliquer nos connaissances et les améliorer.

Bibliographie