

**REPUBLIQUE TUNISIENNE**

**MINISTERE DE L’ENSEIGNEMENT SUPERIEUR ET DE LARECHERCHE SCIENTIFIQUE**

**Direction Générale des Études Technologiques**

**Institut Supérieur des Etudes Technologiques de Djerba**

**Département des Technologies de l’Informatique**

**PROJET DE FIN D’ÉTUDES**

Présenté en vue de l’obtention du

Diplôme de Licence Appliquée en Technologies de

L’Informatique

**Spécialité : D**éveloppement des **S**ystèmes d'**I**nformation

Conception et développement d'applications Web et Mobile pour la logistique et transport « FlexiLogistics »

***Réalisé par :***

**Beltaief Feriel**

**Omrane Karim**

***Encadré par :***

**Mr Mejai Rabii**

***Au sein de l’entreprise TUNVITA***

**Année universitaire : 2021/2022**

**Dédicaces**

Louange à Dieu tout puissant, qui m'a permis de voir ce jour tant attendu

**Je dédie ce travail :**

**À Mon Chère Nour Beltaief**

Je dédie cet événement marquant de ma vie à la mémoire de mon père disparu trop tôt. J’espère que, du monde qui est sien maintenant, il apprécie cet humble geste comme preuve de reconnaissance de la part d’une fille qui a toujours priée pour le salut de son âme.

**À Ma Chère Radhia Jrad**

Aucune dédicace très chère maman, ne pourrait exprimer la profondeur des sentiments que j’éprouve pour vous, vos sacrifices innombrables et votre dévouement firent pour moi un encouragement. Vous avez guetté mes pas, et m’avez couvé de tendresse, ta prière et ta bénédiction m’ont été d’un grand secours pour mener à bien mes études. Puisse Dieu, tout puissant vous combler de santé, de bonheur et vous procurer une longue vie.

**À Mes Chères Ikram, Narimenne et Nawress Beltaief**

Je ne peux trouver les mots justes et sincères pour vous exprimer mon affection et mes pensées, vous êtes pour moi des sœurs et des amies sur qui je peux compter. Je vous dédie ce travail et je vous souhaite une vie pleine de santé et de bonheur.

**À Mon Chère Hamza Sansa**

Je ne saurais exprimer ma profonde reconnaissance pour le soutien continu dont tu as toujours fait preuve. Tu m’as toujours encouragé, incité à faire de mon mieux, ton soutien m’a permis de réaliser le rêve tant attendu. Je te dédie ce travail avec mes vœux de réussite, de prospérité et de bonheur.

Beltaief Feriel

**Dédicaces**

Je dédie ce travail

**À mes parents**

Pour le soutien inconditionnel qu’ils m’ont toujours accordé en toutes circonstances, pour leur amour et leurs précieux conseils. J’espère qu’ils reçoivent à travers ce travail l’expression de mes sincères remerciements et mon profond amour.

Que Dieu les préserve et les procure santé et bonheur.

**À mon frère et ma sœur**

Pour leurs encouragements durant la période de ce projet et pour tous les bons moments qu’on a passé ensemble.

Que Dieu les protège.

**À tous mes amis**

Qui ont partagé avec moi les meilleurs moments de ma vie et qui m’ont aimablement soutenu.

Que Dieu garde notre amitié.

*Omrane Karim*

**Remerciements**

Nous exprimons notre profonde gratitude et respectueuse reconnaissance à notre encadreur Mr Mejai Rabii pour avoir la bonne volonté de bien avoir voulu nous encadrer et pour tout le temps qu’il nous a octroyé et pour tous les conseils qu’il nous a prodigués.

Nous remercions aussi notre encadreur de la société Mr Abdebasset Achour pour la qualité de son suivi durant toute la période de notre projet.

Enfin, nous remercions tous les professeurs de l'ISET de notre formation durant les trois années d'études pour leur sympathie et leur patience, et toutes les personnes qui ont contribué de près ou de loin au bon acheminement de cette formation.

**Table des matières**

[Introduction générale 1](#_Toc106107573)

[Chapitre 1 : Présentation du cadre de projet 6](#_Toc106107574)

[Introduction 6](#_Toc106107575)

[1. Présentation de la société d’accueil 6](#_Toc106107576)

[2. Etude de l’existant 6](#_Toc106107577)

[3. Critique de l’existant 7](#_Toc106107578)

[4. Solution proposée 7](#_Toc106107579)

[5. Contexte générale de projet 8](#_Toc106107580)

[Conclusion 9](#_Toc106107581)

[Chapitre 2 : Spécification des besoins 8](#_Toc106107582)

[Introduction 8](#_Toc106107583)

[1. Méthode de travail 8](#_Toc106107584)

[1.1 La gestion de projet 8](#_Toc106107585)

[1.2 Méthodologies de développement 9](#_Toc106107586)

[1.3 Comparaison entre les méthodes 9](#_Toc106107587)

[1.4 Méthodologie adoptée 11](#_Toc106107588)

[1.4.1 Méthodologie adoptée 11](#_Toc106107588)

[2 Spécification des besoins 13](#_Toc106107589)

[2.1 Les besoins fonctionnels 13](#_Toc106107590)

[2.1.1 Besoins des acteurs 13](#_Toc106107591)

[2.1.2 Le diagramme de cas d’utilisation 16](#_Toc106107591)

[2.2 Les besoins non fonctionnels 23](#_Toc106107592)

[3 Backlog Produit 23](#_Toc106107593)

[4 Planification des releases 26](#_Toc106107594)

[Conclusion 31](#_Toc106107595)

[Chapitre 3 : Conception du système 33](#_Toc106107596)

[Introduction 33](#_Toc106107597)

[1. Conception de l’application 33](#_Toc106107598)

[1.1 Description de la méthode de conception 33](#_Toc106107599)

[1.2 Conception technique détaillée 33](#_Toc106107600)

[1.2.1 Description de la vue statique 33](#_Toc106107600)

[1.2.2 Description de la vue statique 33](#_Toc106107600)

[Conclusion 46](#_Toc106107601)

[Chapitre 4 : Réalisation du système 47](#_Toc106107602)

[Introduction 47](#_Toc106107603)

[1. Environnement de développement 47](#_Toc106107604)

[1.1 Environnement matériel 47](#_Toc106107605)

[1.2 Environnement logiciel 47](#_Toc106107606)

[2. Choix techniques 49](#_Toc106107607)

[2.1 Outils de développement 49](#_Toc106107608)

[4. Réalisation de l’application 56](#_Toc106107609)

[4.1 Site web : 56](#_Toc106107610)

[4.2 Application mobile 69](#_Toc106107611)

[Conclusion 72](#_Toc106107612)

[Conclusion générale 73](#_Toc106107613)

[Bibliographie 74](#_Toc106107614)

**Liste des figures**

[Figure 1 : Société d'accueil 6](#_Toc106107684)

[Figure 2 : Procédure actuel 7](#_Toc106107685)

[Figure 3 : Solution proposée 8](#_Toc106107686)

[Figure 4 : Solution proposée pour optimiser les prix 8](#_Toc106107687)

[Figure 5 : Trois contraintes de projet 8](#_Toc106107688)

[Figure 6 : Méthode agile SCRUM 11](#_Toc106107689)

[Figure 7 : diagramme de cas d'utilisation d'utilisateur 16](#_Toc106107690)

[Figure 8 : Diagramme de cas d'utilisation client 17](#_Toc106107691)

[Figure 9 : Diagramme de cas d’utilisation de Transporteur 18](#_Toc106107692)

[Figure 10 : Planification des releases 26](#_Toc106107693)

[Figure 11 Diagramme de classe 35](#_Toc106107694)

[Figure 12 : diagramme de classe 36](#_Toc106107695)

[Figure 13 : diagramme de séquence authentification 37](#_Toc106107696)

[Figure 14 : diagramme de séquence de suivi livraison 41](#_Toc106107697)

[Figure 15 : diagramme de séquence de gestion des trajets de retour 42](#_Toc106107698)

[Figure 16 : diagramme d’activité de modification profil 43](#_Toc106107699)

[Figure 17 : Diagramme d’activité relatif au cas vérifier la disponibilité de l’offre 44](#_Toc106107700)

[Figure 18 : Diagramme d’états-transitions 45](#_Toc106107701)

[Figure 19 : .Net Core 49](#_Toc106107702)

[Figure 20 : Vue.js 50](#_Toc106107703)

[Figure 21 : Ionic 51](#_Toc106107704)

[Figure 22 : VueX 52](#_Toc106107705)

[Figure 23 : Twilio 52](#_Toc106107706)

[Figure 24 : Mailtrap 52](#_Toc106107707)

[Figure 25 : IntelliTicks 53](#_Toc106107708)

[Figure 26 : Api Restful 53](#_Toc106107709)

[Figure 27 : Machine virtuelle Azure 54](#_Toc106107710)

[Figure 28 : Ngrok 55](#_Toc106107711)

[Figure 29 : Netlify 55](#_Toc106107712)

[Figure 30 : Azure portale 55](#_Toc106107713)

[Figure 31 : Interface de login 56](#_Toc106107714)

[Figure 32 : Interface d’inscription 56](#_Toc106107715)

[Figure 33 : Interface de tableau de bord 57](#_Toc106107716)

[Figure 34 : Interface de profil 58](#_Toc106107717)

[Figure 35 : interface de consultation des personnels 59](#_Toc106107718)

[Figure 36 : Interface de gestion de demandes 59](#_Toc106107719)

[Figure 37 : Interface d’ajout une demande 60](#_Toc106107720)

[Figure 38 : Interface des offres 60](#_Toc106107721)

[Figure 39 : Interface d’offre en détaille 61](#_Toc106107722)

[Figure 40 : Interface de facture 62](#_Toc106107723)

[Figure 41 : Interface de tableau de bord 63](#_Toc106107724)

[Figure 42 : Interface de profil 63](#_Toc106107725)

[Figure 43 : Interface de consultation des villes de destinations 64](#_Toc106107726)

[Figure 44 : Interface d’ajoute une ville 64](#_Toc106107727)

[Figure 45 : Interface de consultation des chauffeurs 65](#_Toc106107728)

[Figure 46 : Interface d’ajoute une chauffeur 65](#_Toc106107729)

[Figure 47 : Interface de gestion des camions 66](#_Toc106107730)

[Figure 48 : Interface d'ajout de camion. 66](#_Toc106107731)

[Figure 49 : Interface de gestion des trajets 67](#_Toc106107732)

[Figure 50 : Interface de gestion des demandes 67](#_Toc106107733)

[Figure 51 : Interface de suggestion d’offre 68](#_Toc106107734)

[Figure 52 : Interface des offres 68](#_Toc106107735)

[Figure 53 : Interface de gestion des factures 69](#_Toc106107736)

[Figure 54 : Interface de login 69](#_Toc106107737)

[Figure 55 : Interface de tableau de bord avec le menu 70](#_Toc106107738)

[Figure 56 : Interface de profil 70](#_Toc106107739)

[Figure 57 : Interface de consultation des offres 71](#_Toc106107740)

[Figure 58 : Interface de consultation des offres 71](#_Toc106107741)

[Figure 59 : Interface de consultation des demandes 72](#_Toc106107742)

**Liste des tableaux**

[Tableau 1 : Tableau comparatif entre les méthodologies de développement [1] 10](#_Toc106107674)

[Tableau 2 : Les besoins des acteurs 14](#_Toc106107675)

[Tableau 3 Description textuelle de cas d'utilisation "s'authentifier" 19](#_Toc106107676)

[Tableau 4 : Description textuelle du cas d’utilisation "Gérer Demande" 20](#_Toc106107677)

[Tableau 5 : Description textuelle de "Accepter une offre" 21](#_Toc106107678)

[Tableau 6 : Description textuelle de "Annuler une acceptation" 22](#_Toc106107679)

[Tableau 7 : Backlog produit 26](#_Toc106107680)

[Tableau 8 : Equipe de Scrum 26](#_Toc106107681)

[Tableau 10 : Environnement matériel 47](#_Toc106107682)

[Tableau 11 : Environnement Logiciel 48](#_Toc106107683)

Introduction générale

L’internet a fait évoluer les modes de consommation au fil des ans. Depuis ces deux dernières décennies, l’achat à distance et la livraison à domicile sont en plein essor. La livraison à domicile présente des atouts et des inconvénients pour l’expéditeur et le destinataire.

En effet, pour les entreprises livreuses, le coût et le temps de transport des marchandises entre le site de production et le client final sont des contraintes qu'il faut maîtriser pour assurer un service satisfaisant aux clients et pour maximiser les profits.

Le présent rapport expose la conception et le développement d’un site web et d'une application mobile qui regroupent plusieurs entreprises de livraison et qui assurent l’automatisation de la gestion des demandes de livraisons.

Le présent rapport détaille nos différentes contributions et réalisation. Il s’articule autour d’une introduction générale de quatre chapitres que nous le décrivons ci-après.

* **Premier chapitre** nommé « Présentation du cadre de projet », s’articule autour de la présentation de l’organisme d’accueil.
* **Deuxième chapitre** nommé « Spécification des besoins », dans ce chapitre nous présentons les principales exigences de notre application (les besoins fonctionnels et non fonctionnels). Pour cela nous aurons recours aux diagrammes de cas d’utilisation UML.
* **Troisième chapitre** qui aborde la phase de conception. Nous présenterons ainsi une conception technique du système.
* **Quatrième chapitre** intitulé « Réalisation » servira à décrire l’environnement de travail, et un aperçu sur les interfaces réalisées.
* **Conclusion** : dans laquelle nous présentons la phase « conclusion et perspectives » qui résume le travail effectué.

|  |  |
| --- | --- |
| Chapitre  1 | Présentation du cadre de projet |

Chapitre 1 : Présentation du cadre de projet

## Introduction

Dans ce chapitre, nous allons positionner le projet dans son contexte organisationnel et méthodologique. Pour ce fait, Nous commençons dans un premier temps par la présentation de l'organisme d’accueil, la problématique et nous définissons dans un deuxième temps le contexte du projet.

## Présentation de la société d’accueil

Créée en 2012, l’agence **Tunvita** dispose d’un large panel de compétences dans le domaine du web, de l’application mobile, des solutions informatiques, dans la conception graphique et réseaux. Elle propose des services tout à fait en phase avec les besoins des TPE, PME, associations, organisations et particuliers.  
Les employés au sein de Tunvita sont des développeurs, des designers, des assistants Marketing, des rédacteurs, des web masters, Community manager, des référenceurs, des intégrateurs, qui sont tous de véritables experts dans leur domaine.



Figure 1 : Société d'accueil

## Etude de l’existant

•Le client navigue sur le internet afin de trouver une société de transport.

•Le client choisi une ou plusieurs sociétés et remplir un formulaire de demande de livraison ou bien il appelle la société par téléphone.

La figure ci-dessous représente la procédure actuelle

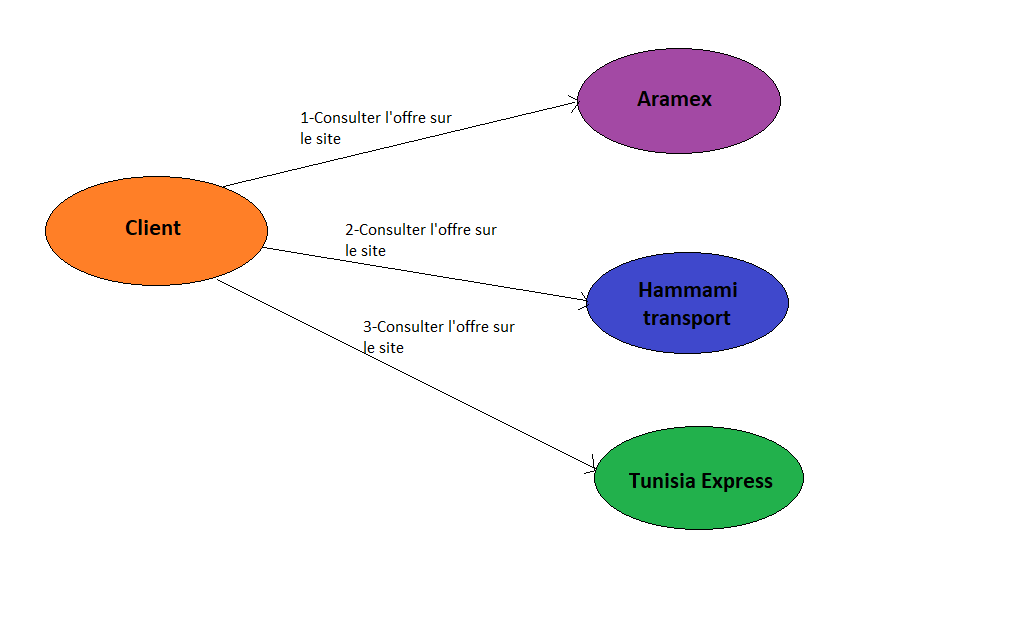


Figure 2 : Procédure actuel

## Critique de l’existant

Il est à noter qu’il n'y a aucun site web tunisien qui regroupe les sociétés de livraison en un seul endroit et permette au client de comparer les prix.

* Le client doit chercher sur plusieurs sites pour trouver l'offre idéale.
* Habituellement, la demande de livraison est soumise par téléphone ou directement sur place.
* Le prix est fixe et n'est pas négociable.

## Solution proposée

Après avoir dégagé les problèmes constatés au niveau de critique de l’existant, la solution qui nous proposons est de développer un site web et une application mobile, performantes qui regroupent les différents transporteurs dans le même endroit. Le client envoi une seule demande et reçoit plusieurs offres avec des prix concurrentiels.

La figure ci-dessous représente la solution proposée



Figure 3 : Solution proposée

La figure ci-dessous représente la solution proposée pour optimiser les prix



Figure 4 : Solution proposée pour optimiser les prix

Nous avons pensé exploiter tout le trajet entre A et B (aller et retour) afin de maximiser les profits de transporteur d’une part et réduire le prix pour le client en d’autre part.

## Contexte générale de projet

Le sujet de notre projet de fin d'études consiste à développer une application web pour les clients qui recherchent une livraison pour leurs marchandises, pour les entreprises qui souhaitent augmenter leurs profits en publiant leurs offres et pour l’intermédiaire le propriétaire de l'application.

L'application mobile a pour but de faciliter le suivi de l'avancement des demandes et des offres en recevant des notifications.

Ce projet s’inscrit dans le cadre du projet de fin d’études l’obtention du diplôme de licence appliquée en informatique : Développement des systèmes d'Informations et a été effectué au sein de l’entreprise « Tunvita ».

## Conclusion

Durant ce premier chapitre nous avons essayé de donner un aperçu général sur le projet, et nous avons présenté, une étude de l'existant dans ce qui suit nous allons présenterons une

spécification des besoins.

|  |  |
| --- | --- |
| **Chapitre 2** | **Spécification des besoins** |

Chapitre 2 : Spécification des besoins

## Introduction

L’analyse de l’existant a permis de dégager les fonctionnalités qui seront mises à la disposition de l’utilisateur. Les besoins ainsi dégagés sont classés en besoins fonctionnels et besoins non fonctionnels.

## Méthode de travail

### 1.1 La gestion de projet

La gestion de projets est très complexe car elle est amenée à évoluer dans différents environnements qui subissent une évolution constante.

On doit donc être multi compétent, c'est-à-dire maîtriser les techniques de gestion de projet, de management d’équipe, avoir un bon relationnel lors des échanges avec le client et enfin comprendre les spécificités du projet.

 La gestion de projets est très complexe car elle est amenée à évoluer dans différents environnements qui subissent une évolution constante.

L’objectif de gestion de projets est de pouvoir mener son projet à termes en respectant les délais et le budget alloué. Pour atteindre cet objectif, on doit prendre en compte les trois contraintes qui constituent le projet.

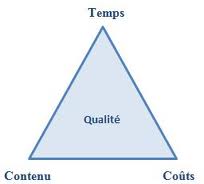


Figure 5 : Trois contraintes de projet

### 1.2 Méthodologies de développement

Une méthode d'analyse et de conception est un procédé qui a pour objectif de permettre de formaliser les étapes préliminaires du développement d'un système afin de rendre ce développement plus fidèle aux besoins du client. Pour ce faire, nous partons d'un énoncé informel (le besoin tel qu'il est exprimé par le client, complété par les futurs utilisateurs d'un logiciel), ainsi que de l'analyse de l'existant éventuel (c'est-à-dire la manière dont les processus à traiter par le système se déroulent actuellement chez le client).

On distingue deux familles de méthodes : Les méthodes agiles et les méthodes du processus unifié.

* **Les méthodes agiles :**

Une méthode agile est une méthode de développement informatique permettant de concevoir des logiciels en impliquant au maximum le demandeur (client). Elles utilisent un principe de développement itératif qui consiste à découper le projet en plusieurs étapes qu’on appelle “itération”.

Ces itérations sont en fait des mini-projets définis avec le client en détaillant les différentes fonctionnalités qui seront développées en fonction de leur priorité. Le chef de projet établi alors une macro planning correspondant aux tâches nécessaires pour le développement de ces fonctionnalités.

* **Le processus unifié (PU)**

Le Processus unifié est une méthode de prise en charge du cycle de vie d’un logiciel et du développement, pour les logiciels orientés objets C’est une méthode générique, itérative et incrémentale.

### 1.3 Comparaison entre les méthodes

Il est très important de bien choisir la méthodologie qui répond parfaitement à nos besoins, s’adapte au mieux aux exigences du client et fournit l’ouvrage dans les plus brefs des délais Pour cette raison, il est nécessaire effectuer une étude comparative entre les méthodologies afin d’en sélectionner la meilleure qui s’adapte au projet Le tableau ci-dessous décrit les méthodologies de développement avec un critère de comparaison, l’aptitude de l’appliquer sur une équipe de développeur de 3 personnes.

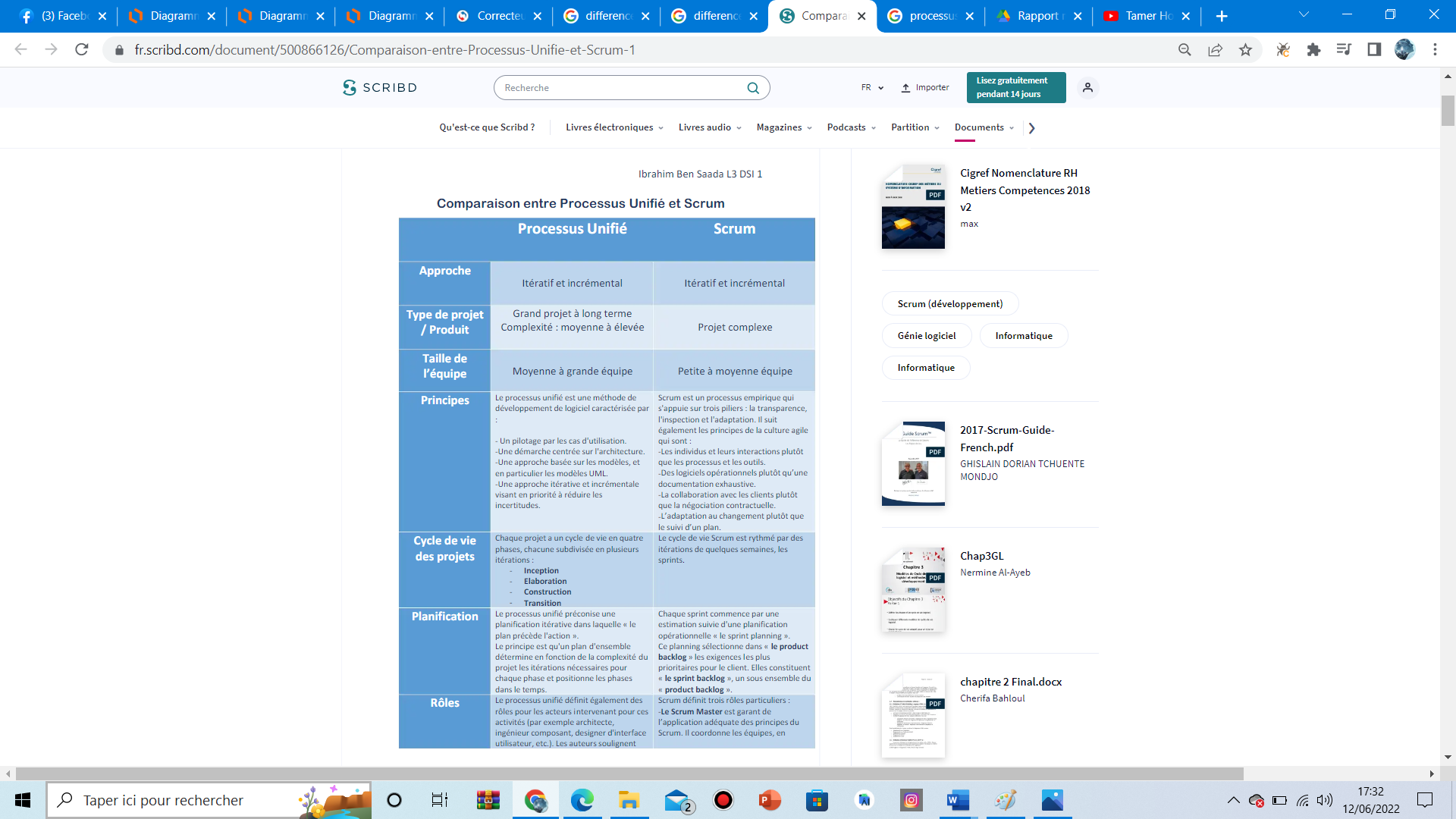


Tableau 1 : Tableau comparatif entre les méthodologies de développement [1]

### 1.4 Méthodologie adoptée

Suite à l’étude comparative nous avons remarqué que les quatre méthodologies étudiées permettent le travail itératif pour un meilleur suivi de l’avancement global du projet, et chacune met l’accent sur une phase bien déterminée du projet. Nous avons opté pour la méthodologie Scrum car elle satisfait les conditions suivantes :

- Scrum convient aux équipes de 3 à 9 personnes.

-Le client est impliqué dans le développement de l’application : notre client est prêt à participer de manière active et régulière à notre projet.

-La progression des tâches s’effectue pendant une durée de développement courte

En se basant sur ces critères, nous pouvons faire des itérations rapides sur le besoin.

Par conséquent, notre choix sera la Méthode Agile (Scrum).

#### Présentation de Scrum

Scrum ou « mêlée, est un terme emprunté au rugby qui désigne la solidarité et la force qui lient les membres de l’équipe au succès de l’itération Le cycle de vie de Scrum est rythmé par des itérations de deux à quatre semaines qu’on appelle Sprint.



Figure 6 : Méthode agile SCRUM

Avant chaque sprint, on effectue une réunion de planification appelée le sprint planning meeting qui consiste à sélectionner les exigences prioritaires pour le client dans le produit backlog qui seront développées, testées et livrées au client : le backlog sprint (sous-ensemble du produit backlog).

Des Scrums sont organisées quotidiennement durant le sprint afin de contrôler l’avancement pour s’assurer que les objectifs sont tenus. A la fin du sprint, une démonstration des derniers développements est présentée au client ce qui donnera lieu à un bilan qualitatif sur le fonctionnement de l’équipe.

Les valeurs mises en avant par cette méthode sont les suivantes :

* **Visibilité** : Avoir une vision réelle sur le résultat
* **Inspection** : Vérifier l’écart par rapport à l’objectif initial
* **Adaptation** : S’adapter en fonction des écarts constatés afin de les ajuster Scrum est favorable à des petits ajustements fréquents.
* **Les membres de l’équipe Scrum :**

On distingue plusieurs acteurs, dont on peut citer :

* **Le directeur de produit** (product owner) est le représentant des clients et des utilisateurs, et fait également parti de l'équipe.
* **Le Scrum Master** veille à l’application de la méthodologie Scrum au sein de l’équipe.
* **Les Développeurs** : créent un incrément Done potentiellement délivrable en production à chaque Sprint et gèrent le Sprint Backlog.
* **Le processus :**

Tous les critères ou exigences du produit sont regroupées dans des journaux ou backlog dont on distingue 2 types :

* Le backlog de produit ou « Product backlog » qui regroupe la liste des fonctionnalités du produit.
* Le backlog de sprint ou « sprint Backlog » en fonction des fonctionnalités du produit, regroupe la liste des tâches qui devra être réalisées à l’itération en cours.

Chaque tâche aurait fait l'objet d’une estimation préalable de charge par l’ensemble des membres de l’équipe afin d’estimer au mieux les taches qui peuvent réaliser un sprint.

* **La Planification :**
* **Release** : Il correspond à la livraison d’une version pour le client comme d’habitude, nous parlons du release pour considérer la période de temps depuis le début du travail sur cette version jusqu’à la livraison et passe à travers une série de sprints successifs.
* **Sprint** : Dès le début d'un projet, la première planification permet de définir le périmètre de chaque itération appelé sprint Chaque sprint dure quelques semaines et regroupe une liste de taches (défini dans le blacklog)
* **La revue du sprint :**

La fin d’un sprint aboutit à la réalisation d’un produit avec des fonctionnalités partielles avec la documentation associée. Dans la plupart des cas, cela conduit à une revue de sprint consistant à faire une démonstration de la réalisation du sprint devant le client afin de valider le travail réalisé et d’avoir un retour pour ajuster le backlog de produit.

## Spécification des besoins

### 2.1 Les besoins fonctionnels

Afin de répondre aux besoins détaillés dans le cahier des charges, nous allons identifier les besoins des acteurs. Puis nous allons produire un diagramme de cas d’utilisation suivi des descriptions textuelles des principaux cas d’utilisation, pour finir avec les diagrammes des séquences système.

### 2.1.1 **Besoins des acteurs**

À ce stade, nous allons décrire les rôles des acteurs communiquent avec le système, plus spécifiant le tableau suivant illustrent les fonctionnalités que doit intégrer l'application à développer.

|  |  |
| --- | --- |
| **Acteurs** | **Description des besoins fonctionnels** |
| **Client**  **Transporteur**      **Transporteur** | L’application permet le client de :   * Demander une inscription * Suivre personnels * Gérer demandes * Ajout d’une demande * Notifier intermédiaire par mail et par message téléphonique * Consulter demande envoyées * Filtrer les demandes envoyées * Gérer offres * Consulter offres reçues * Filtrer les offres reçues * Accepter une offre * Annuler une offre * Notifier intermédiaire par mail et par message téléphonique * Gérer factures client * Consulter facture à payer * Envoyé facture payée * Notifier intermédiaire par mail et par message téléphonique   L’application doit permettre le transporteur de :   * Demander une inscription * Gérer les Camions * Ajout camion * Consulter camion * Modifier camion * Supprimer camion * Filtrer camion * Gérer les chauffeurs * Ajout chauffeur * Consulter chauffeur * Modifier chauffeur * Supprimer chauffeur * Filtrer chauffeur * Gérer les villes * Ajout des villes destinataires * Consulter villes * Supprimer des villes * Gérer les Trajets * Ajouter un trajet * Consulter les trajets * Modifier un trajet * Supprimer un trajet * Filtrer les trajets * Gérer demandes devis * Consulter les demandes reçues * Filtrer les demandes reçues * Envoyer une offre * Notifier intermédiaire par mail et par message téléphonique * Refuser une demande * Gérer offres * Consulter les offres envoyées * Filtrer les offres envoyées * Changer l’état de l’offre * Gérer factures * Ajout facture transporteur * Consulter factures payées |

Tableau 2 : Les besoins des acteurs

#### 2.1.2 Le diagramme de cas d’utilisation

Pour illustrer les fonctionnalités offertes par notre système, nous avons opté pour le diagramme de cas d’utilisation. Ce diagramme donne une vue sur les fonctionnalités de notre système ainsi que les acteurs qui l’utilisent.

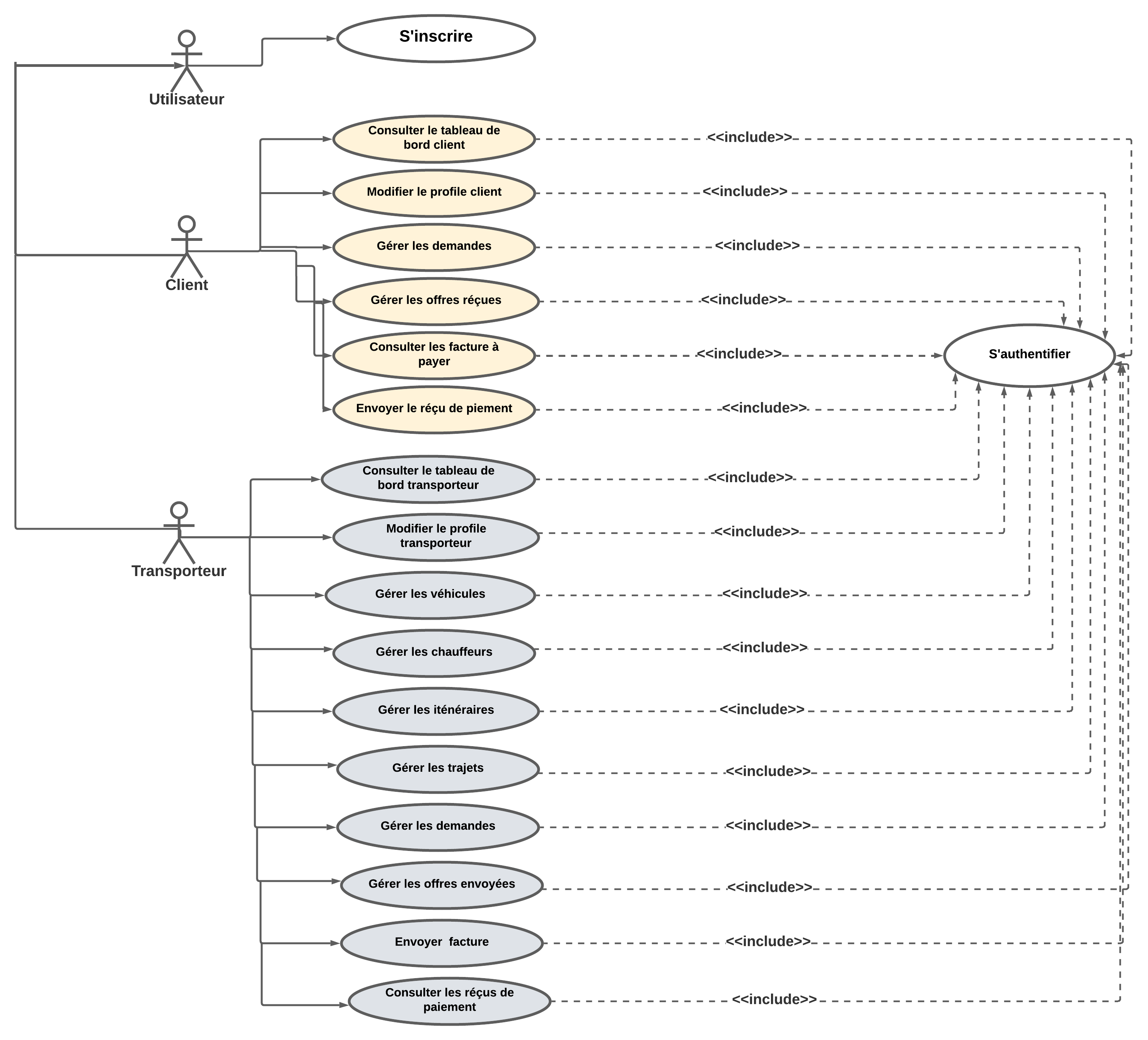


Figure 7 : diagramme de cas d'utilisation d'utilisateur

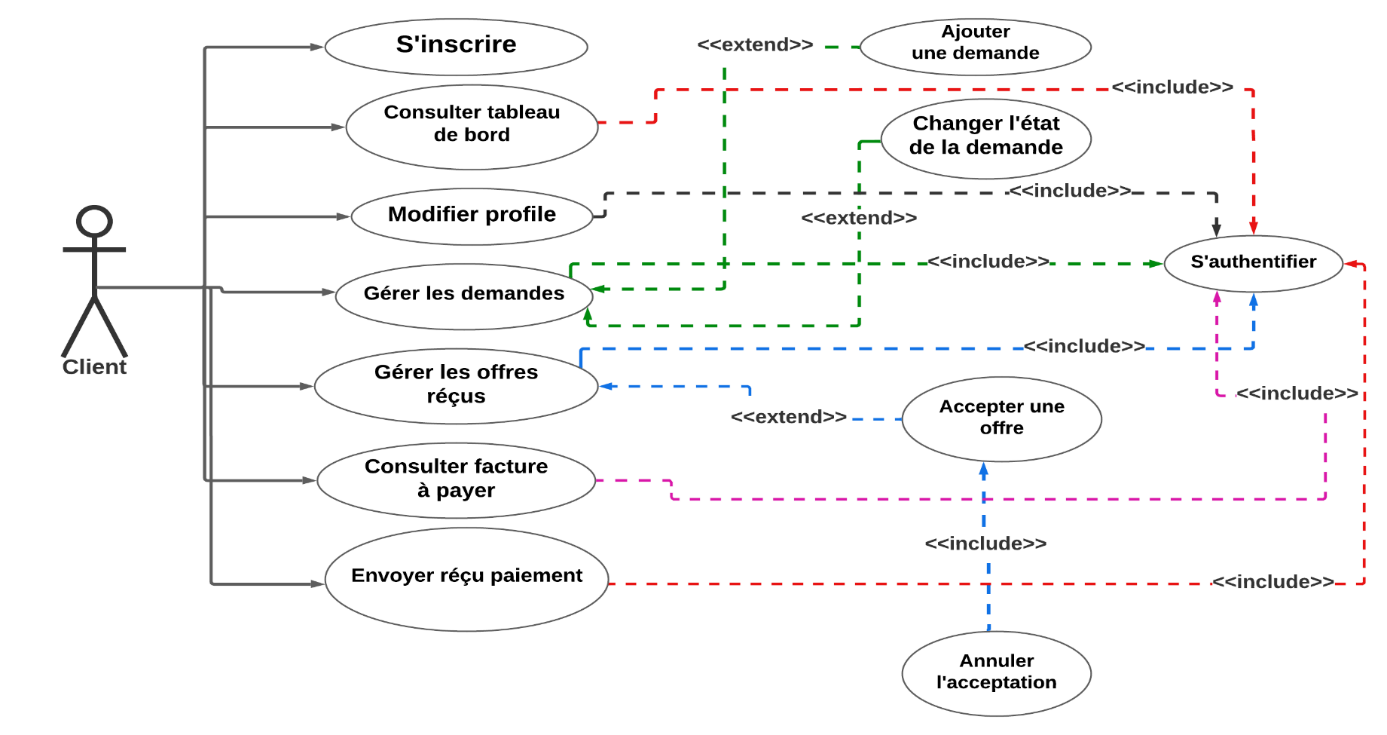
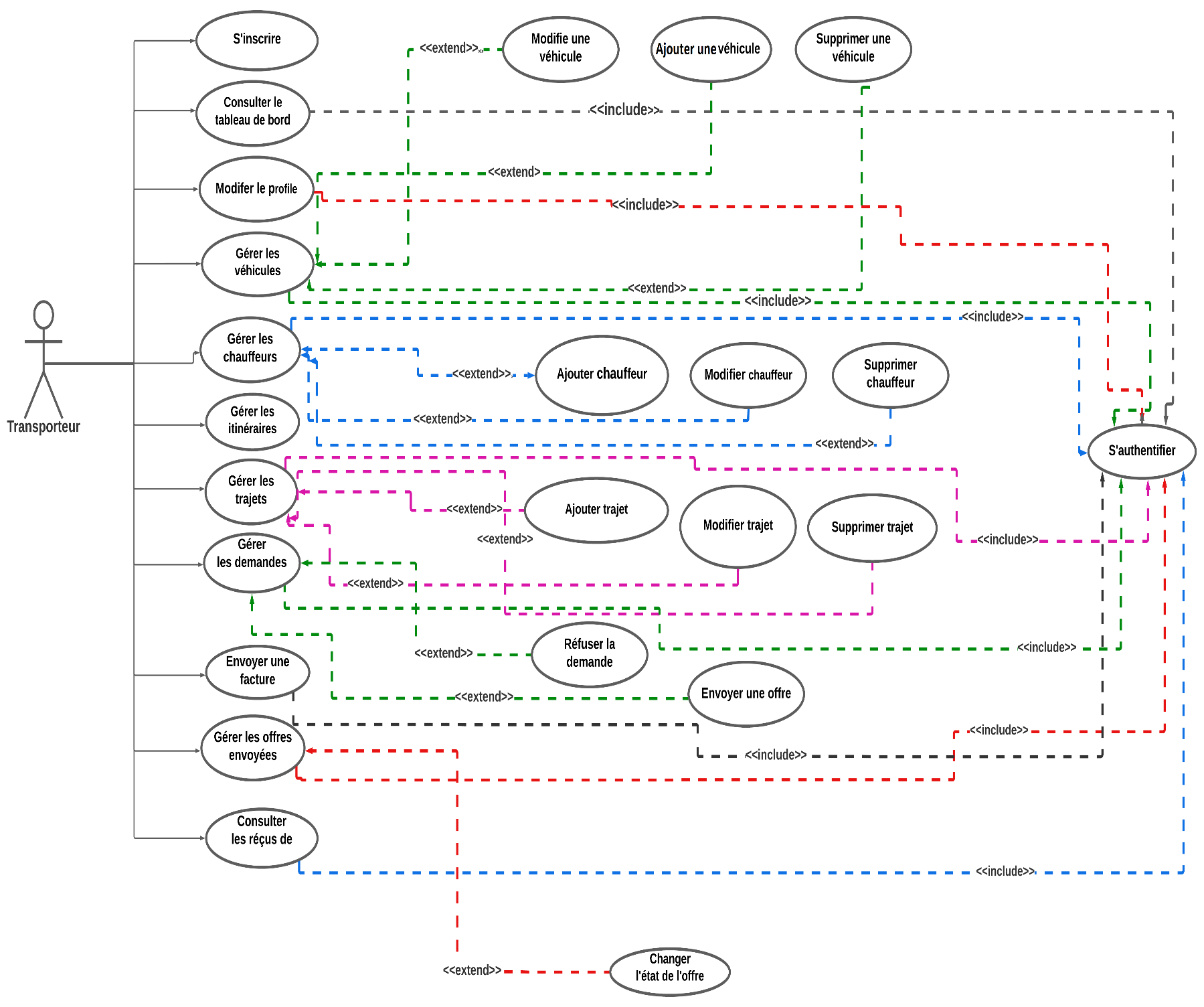


Figure 8 : Diagramme de cas d'utilisation client



**paiement**

Figure 9 : Diagramme de cas d’utilisation de Transporteur

* **Description textuelle du cas d’utilisation « s’authentifier »**

Le tableau ci-dessous décrit de cas d’utilisation « s’authentifier » et la démarche suivie afin d’accéder à l’application :

**Sommaire d’authentification**

* **Titre :** Authentification.
* **Acteur :** Client et transporteur (Utilisateur).
* **Objectif** : Lors du lancement de l’application, le client ou le transporteur doit se connecter pour accéder à son espace.

**Description des enchaînements**

Le cas d’utilisation commence lorsque l’utilisateur saisie l’URL de la page d’authentification.

**Préconditions :** L’utilisateur doit être enregistré dans la base de données.

##### Enchaînement nominal :

1. L’utilisateur accède à l’application.
2. Le système fournit un formulaire d’authentification.
3. L’utilisateur doit saisir son identifiant et son mot de passe.
4. Le système vérifie les coordonnées saisies.
5. Le système redirige l’utilisateur vers la page correspondante.

##### Enchaînement alternatif :

A1. Les coordonnées saisies sont incorrectes : L'enchaînement démarre au point 4 du scénario nominal.

Le système indique que l’identifiant ou le mot de passe est erroné la séquence nominale reprend au point 3.

A2. Le champ Login ou mot de passe est vide :

L'enchaînement démarre au point 4 de la séquence nominale : Le système indique que le champ login est obligatoire

La séquence nominale reprend au point 3.

**Post-conditions :**

En cas de réussite d'authentification, l'utilisateur est dirigé vers la page correspondante.

Tableau 3 Description textuelle de cas d'utilisation "s'authentifier"

* **Description textuelle du cas d’utilisation « Gérer Demande »**

Le tableau ci-dessous représente la description textuelle de cas d’utilisation « Gérer demandes pour la partie transporteur » :

**Description des enchaînements**

**Préconditions :**

Transporteur authentifié, la demande est à l’état non traité.

**Enchaînement nominal :**

1. Le transporteur demande la page « demandes ».
2. Le système affiche la liste de demandes.
3. Le transporteur choisit une demande à traiter.

##### Enchaînement alternatif :

A1. Accepter la demande : le système affiche la page suggestion offre.

A2. Refuser la demande : demande rejetée

**Post-conditions :**

Une nouvelle offre est ajoutée où demande refusée.

**Sommaire Gérer une demande**

**Titre :** Gérer une demande.

**Acteurs :** Transporteur.

**Objectif** : Dans cette étape, le transporteur traite la demande envoyée.

Tableau 4 : Description textuelle du cas d’utilisation "Gérer Demande"

* **Description textuelle de « Accepter une offre »**

Le tableau ci-dessous représente la description textuelle du cas d’utilisation « Accepter une offre » :

**Sommaire d’acception de l’offre**

**Titre :** Accepter une offre.

**Acteurs :** Client.

**Objectif** : Le client choisi la meilleure offre.

**Date de mise à jour :** 28/02/2016

**Description des enchaînements**

**Préconditions :**

Client authentifié, offre à l’état de « en cours de traitement »

**Enchaînement nominal :**

1. L’administrateur demande la page « offres »
2. Le système lui affiche la page demandée
3. Le client accepte l’offre
4. Le système change l’état de l’offre à ‘en attente d’acceptation’

**Post-conditions :**

Attente de confirmation de disponibilité de l’offre

Tableau 5 : Description textuelle de "Accepter une offre"

* **Description textuelle de « Annuler une acceptation »**

Le tableau ci-dessous représente la description textuelle du cas d’utilisation « annuler une acceptation »

**Description des enchaînements**

**Préconditions :**

Client authentifié, offre à l’état de « en attente d’acceptation »

**Enchaînement nominal :**

1. L’administrateur demande la page « offres »
2. Le système lui affiche la page demandée
3. Le client clique sur « annuler »
4. Le système change l’état de l’offre à

**Post-conditions :**

Offre annulée

**Sommaire d’annulation d’une acceptation**

**Titre :** Annuler l’offre.

**Acteurs :** Client.

**Objectif** : Le client annule l’acceptation de l’offre.

Tableau 6 : Description textuelle de "Annuler une acceptation"

### 2.2 Les besoins non fonctionnels

Les besoins non fonctionnels sont importants car ils agissent de façon indirecte sur le résultat et sur le rendement de l’utilisateur, ce qui fait qu’ils ne doivent pas être négligés, pour cela il faut répondre aux exigences suivantes :

* **Fiabilité :**

L’application doit fonctionner de façon cohérente sans erreurs et doit être satisfaisante.

* Les erreurs :

Les ambigüités doivent être signalées par des messages d’erreurs bien organisés pour bien guider l’utilisateur et le familiariser avec notre site web.

* Ergonomie et bonne Interface :

L’application doit être adaptée à l’utilisateur sans qu’il ne fournisse aucun effort (utilisation, claire et facile) de point de vue navigation entre les différentes pages, couleurs et mise en textes utilisés.

* Compatibilité et portabilité :

Un site web quel que soit son domaine, son éditeur et son langage de programmation ne peut être fiable qu’avec une compatibilité avec tous les navigateurs web et tous les moyens que ce soit PC, IPAD ou Mobiles.

## Backlog Produit

Le backlog est élaboré avant le lancement des sprints sous la base des besoins de l’utilisateur citée précédemment, dans la phase de préparation (ou sprint 0). Il est utilisé d’une part pour planifier les releases, d’autre part pour distinguer le contenu de chaque sprint. [2]

Nous présentons à travers le tableau ci-dessous le Backlog du produit relatif à notre solution. Il énumère les champs suivants :

* ID : C’est un nombre unique et auto-incrémenté pour chaque User Story.
* Titre : C’est le résumé de l’User Story.
* User Story : C’est une description courte de la tâche à réalise

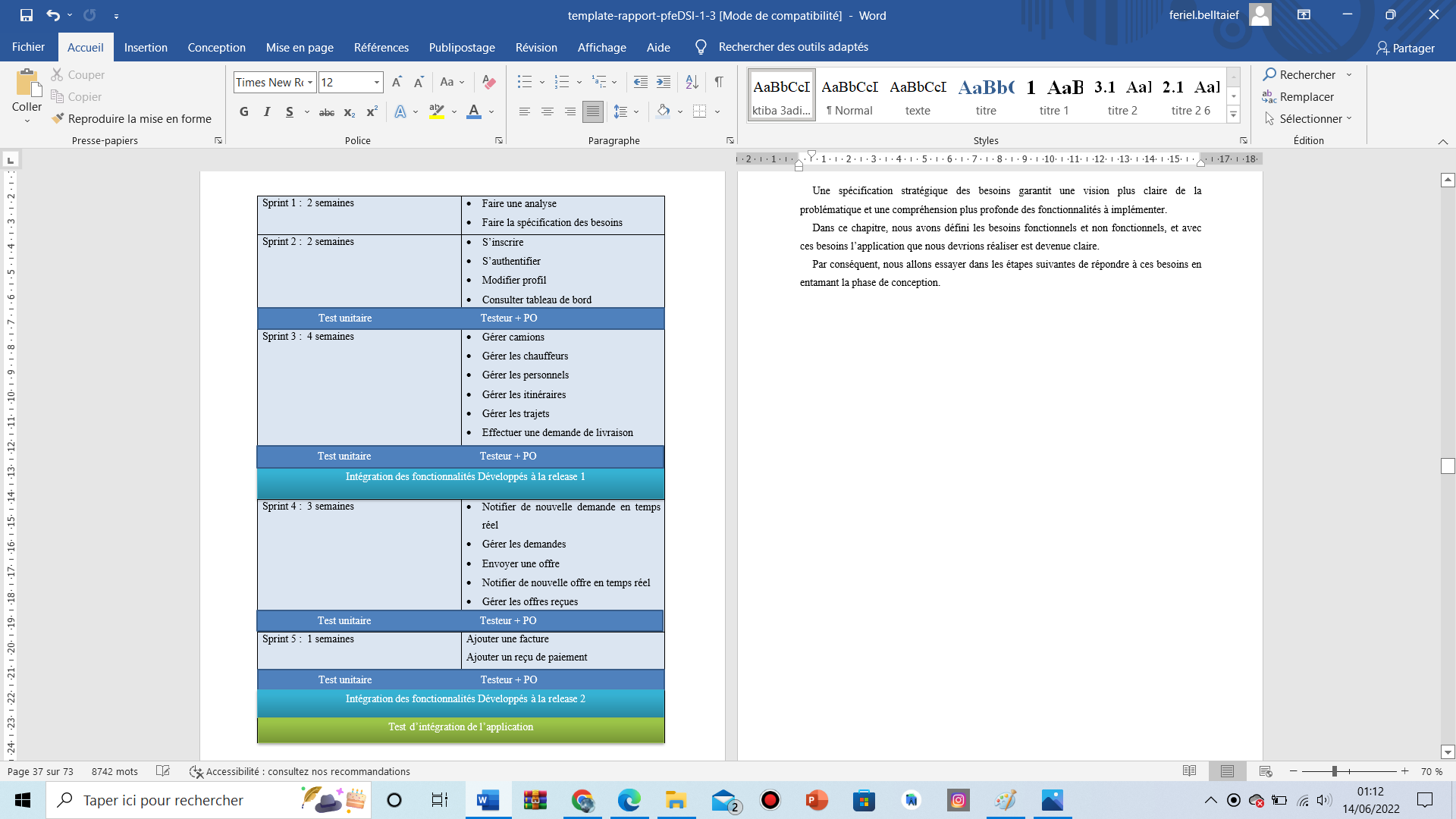
|  |  |  |
| --- | --- | --- |
| **ID** | **User Story** | **Description** |
| **1** | S’inscrire | En tant qu’utilisateur (client - transporteur), je peux m’inscrire. |
| **2** | S’authentifier | En tant qu’utilisateur (client - transporteur - intermédiaire), je peux m’authentifier. |
| **3** | Activer un compte | En tant qu’intermédiaire, je peux activer les comptes des clients et des transporteurs. |
| **4** | Gérer les clients | En tant qu’intermédiaire je peux gérer les clients. |
| **5** | Gérer les transporteurs | En tant qu’intermédiaire, je peux gérer les transporteurs. |
| **6** | Gérer les personnels | En tant qu’intermédiaire, je peux gérer les personnels. |
| **7** | Consulter les permissions | En tant qu’intermédiaire, je peux consulter les permissions. |
| **8** | Affecter une permission à un personnel | En tant qu'intermédiaire, je peux affecter une ou plusieurs permissions au personnel. |
| **9** | Modifier l’état de statuts | En tant qu'intermédiaire, je peux activer ou désactiver les comptes (clients, transporteurs). |
| **10** | Gérer les sociétés | En tant qu'intermédiaire, je peux gérer les sociétés. (Modifier-supprimer-ajouter) |
| **11** | Gérer les villes | En tant qu'intermédiaire, je peux gérer les villes.  (Modifier-supprimer-ajouter) |
| **12** | Gérer les véhicules | En tant que transporteur, je peux gérer les véhicules.  (Modifier-supprimer-ajouter) |
| **13** | Gérer les chauffeurs | En tant que transporteur, je peux gérer les chauffeurs.  (Modifier-supprimer-ajouter) |
| **14** | Affecter des destinations à un transporteur | En tant que transporteur, je peux ajouter des villes destinataires. |
| **15** | Ajouter des trajets | En tant que transporteur, je peux ajouter des trajets. |
| **16** | Consulter les trajets | En tant qu'intermédiaire, je peux suivre trajets. |
| **17** | Gérer les rôles | En tant qu’intermédiaire, je peux gérer les rôles. |
| **18** | Consulter le Dashboard | En tant qu’utilisateur (client - transporteur - intermédiaire) , je peux consulter le Dashboard qui donne une vue globale sur le statistique de l’application |
| **19** | Modifier profil | En tant qu’utilisateur (client - transporteur - intermédiaire), je peux modifier mon profil |
| **20** | Passer une demande de livraison | En tant que client de l’application, je peux passer une commande de livraison. |
| **21** | Modifier l’état de demande de livraison | En tant qu’utilisateur (client - intermédiaire) de l’application, je peux modifier l’état de demande de livraison |
| **22** | Consulter les demandes de livraison | En tant qu’utilisateur (client - intermédiaire) de l’application, je peux lister les demandes de livraison. |
| **23** | Consulter les factures transporteurs | En tant qu’utilisateur (transporteur - intermédiaire), je peux consulter les factures transporteurs. |
| **24** | Consulter les factures clients | En tant qu’utilisateur (client - intermédiaire), je peux consulter les factures clients. |
| **25** | Envoyer factures | En tant que transporteur, je peux envoyer des factures à l'intermédiaire.  En tant qu'intermédiaire, je peux envoyer des factures au transporteur. |
| **26** | Envoyer payement | En tant que client, je peux envoyer des paiements à l'intermédiaire.  En tant qu'intermédiaire, je peux envoyer des paiements au transporteur. |

Tableau 7 : Backlog produit

|  |  |
| --- | --- |
| Product Owner | Mr Abdelbasset Achour |
| Scrum Master | Notre encadrant pédagogique  Mr Mejai Rabii |
| Equipe de développements | Omrane Karim  Beltaief Feriel  Bouzayeni Malek |

Tableau 8 : Equipe de Scrum

## Planification des releases



Test unitaire

jhdfkjjkfddjkfdfkdffdgdfdekl Testeur + PO

Figure 10 : Planification des releases

## Conclusion

Une spécification stratégique des besoins garantit une vision plus claire de la problématique et une compréhension plus profonde des fonctionnalités à implémenter. Dans ce chapitre, nous avons défini les besoins fonctionnels et non fonctionnels, et avec ces besoins l’application que nous devrions réaliser est devenue claire. Par conséquent, nous allons essayer dans les étapes suivantes de répondre à ces besoins en entamant la phase de conception.

|  |  |
| --- | --- |
| **Chapitre 3** | **Conception du système** |

Chapitre 3 : Conception du système

## Introduction

Après avoir fixé les besoins de notre projet, nous allons détailler le système et en associer la forme et l’architecture correspondante. De ce fait, nous allons procéder dans cette partie à la conception technique de l'application.

## Conception de l’application

La conception de systèmes est le processus de définition de l'architecture, des composants, des modules, des interfaces et des données pour un système pour satisfaire les exigences spécifiées. D’ailleurs la conception de systèmes pourrait être considérée comme l'application de la théorie des systèmes au développement de produits.

### 1.1 Description de la méthode de conception

UML (langage de modélisation unifié) c’est une méthode normalisée qui cadre l’analyse et la conception d'un système, En effet c’est une méthode qui a des architectures logicielles fondées sur les objets du système elle représente un support de communication performant en utilisant les techniques orientée objet.

### 1.2 Conception technique détaillée

Nous allons détailler dans cette partie la conception de notre projet en commençant par la vue statique du système et utiliser par la suite une représentation dynamique pour détailler quelques scenarios

#### 1.2.1 Description de la vue statique

Dans cette partie nous allons présenter les règles des gestions qui nous permettent de faire une conception de la base de données et le diagramme de classe.

* **Règle de gestion :**

Le diagramme de classe étudiée est basé sur les règles de gestion suivantes :

R1 : Un utilisateur peut être un transporteur, client, chauffeur ou intermédiaire.

R2 : Chaque utilisateur appartient à une société

R3 : Chaque transporteur possède une ou plusieurs camions

R4 : Chaque camion est conduit par un chauffeur

R5 : Chaque camion possède un type

R6 : Chaque camion peut avoir plusieurs trajets

R7 : Le trajet est associé à deux villes

R8 : Chaque transporteur livre sur plusieurs itinéraires

R9 : Chaque itinéraire est associé à une ville

R10 : Chaque client peut envoyer une ou plusieurs demandes de livraison

R11 : Chaque demande de livraison possède une ou plusieurs fichiers et possède un seul état

R12 : Chaque demande livraison peut avoir une ou plusieurs factures

R13 : Chaque demande de livraison est transmise à une demande de devis

R14 : Le transporteur peut recevoir une ou plusieurs demandes devis

R15 : Chaque demande devis possède un état

R16 : Le transporteur peut envoyer une ou plusieurs offres, chacune correspondante à une demande déterminée

R17 : Chaque offre peut avoir une ou plusieurs factures

R18 : Chaque offre peut avoir une ou plusieurs fichiers

* **Diagramme de classe :**

La vue structurelle du modèle UML est la vue la plus utilisée pour spécifier une application. L’objectif de cette vue consiste à modéliser la structure des différentes classes d’une application orientée objet ainsi que leurs relations.

La figure ce dessous représente la vue statique de notre application illustrée par le diagramme de classe suivant :

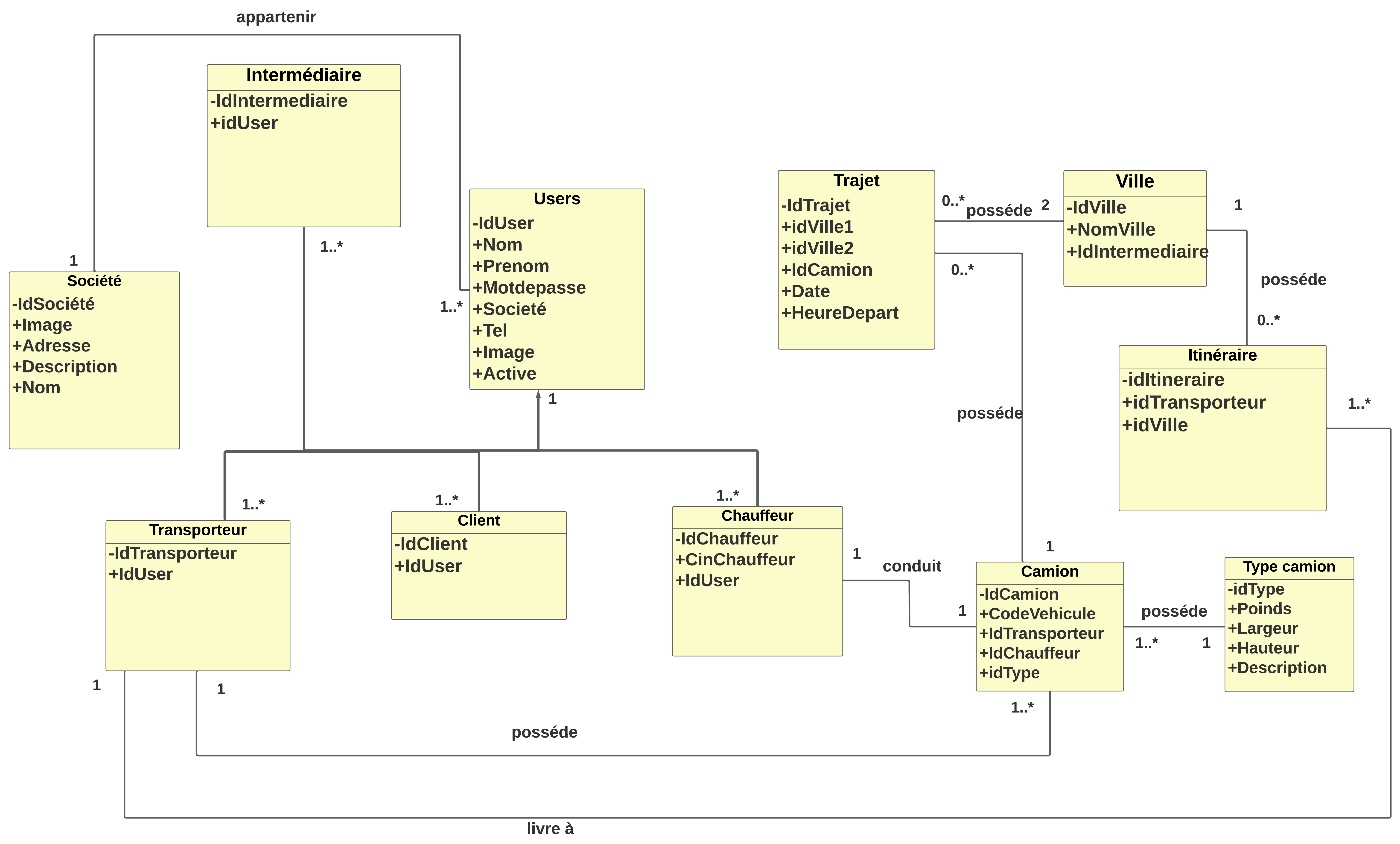


Figure 11 Diagramme de classe

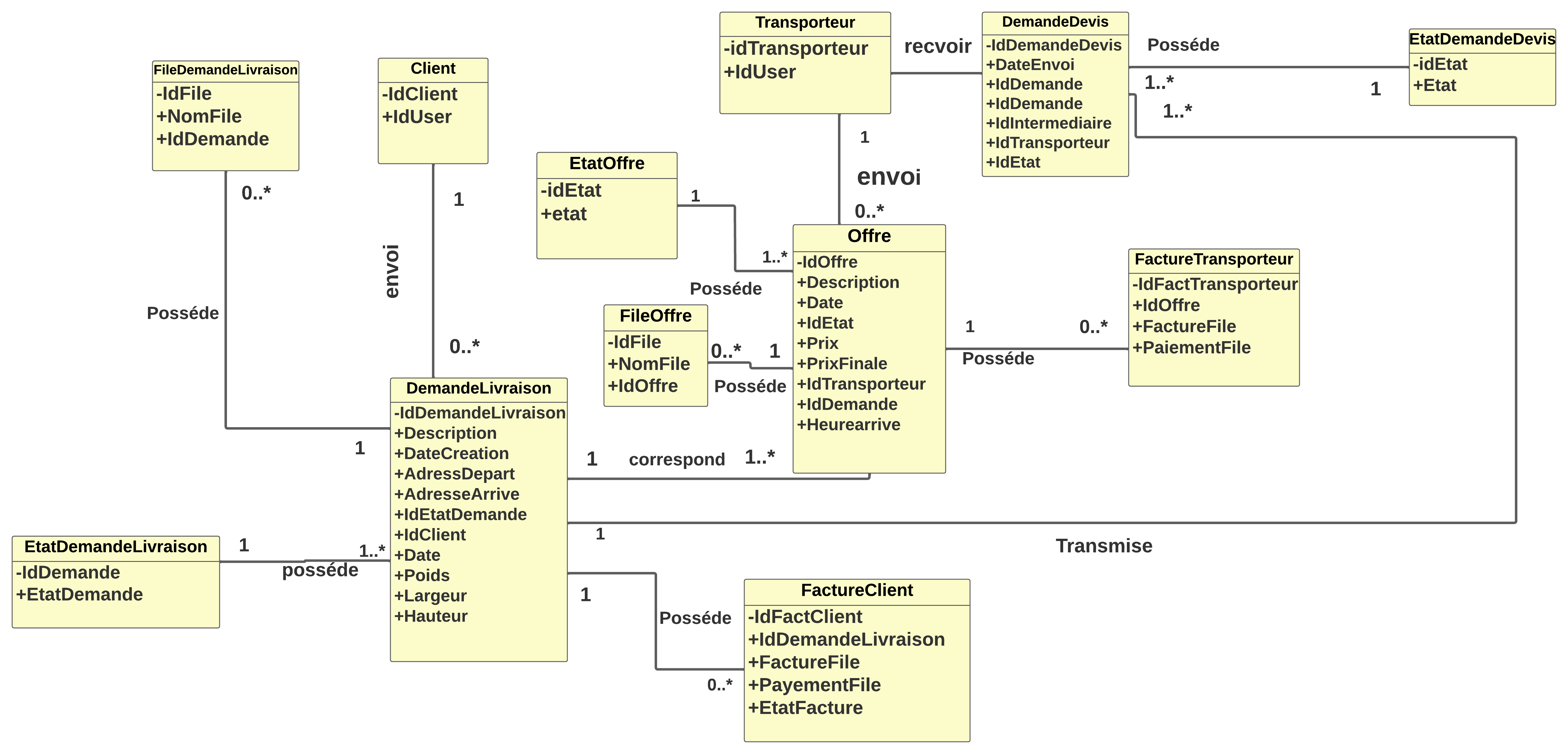


Figure 12 : diagramme de classe

#### 1.2.2 Description de la vue dynamique

La vue dynamique de notre application est représentée par les différents diagrammes de séquences, quelques diagrammes d’activités et un diagramme d’états- transitions.

* **Diagrammes séquence objet**

Nous allons présenter dans cette partie quelques diagrammes de séquence objet.

* **Diagramme de séquence objet relatif au cas « S’authentifier »**

L’utilisateur se connecte au système, le système lui affiche un formulaire contenant deux champs : l’identifiant et le mot de passe, le système vérifie l'existence du l’utilisateur :

* Si l’utilisateur existe, il y a un succès d’accès sinon le système lui affiche un message d’erreur. Ce diagramme est valable pour l’authentification de client, transporteur et intermédiaire.
* Si l’utilisateur est un client il sera dirigé vers l’espace client
* Si l’utilisateur est un transporteur il sera dirigé vers l’espace transporteur
* Si l’utilisateur est un intermédiaire il sera dirigé vers l’espace intermédiaire

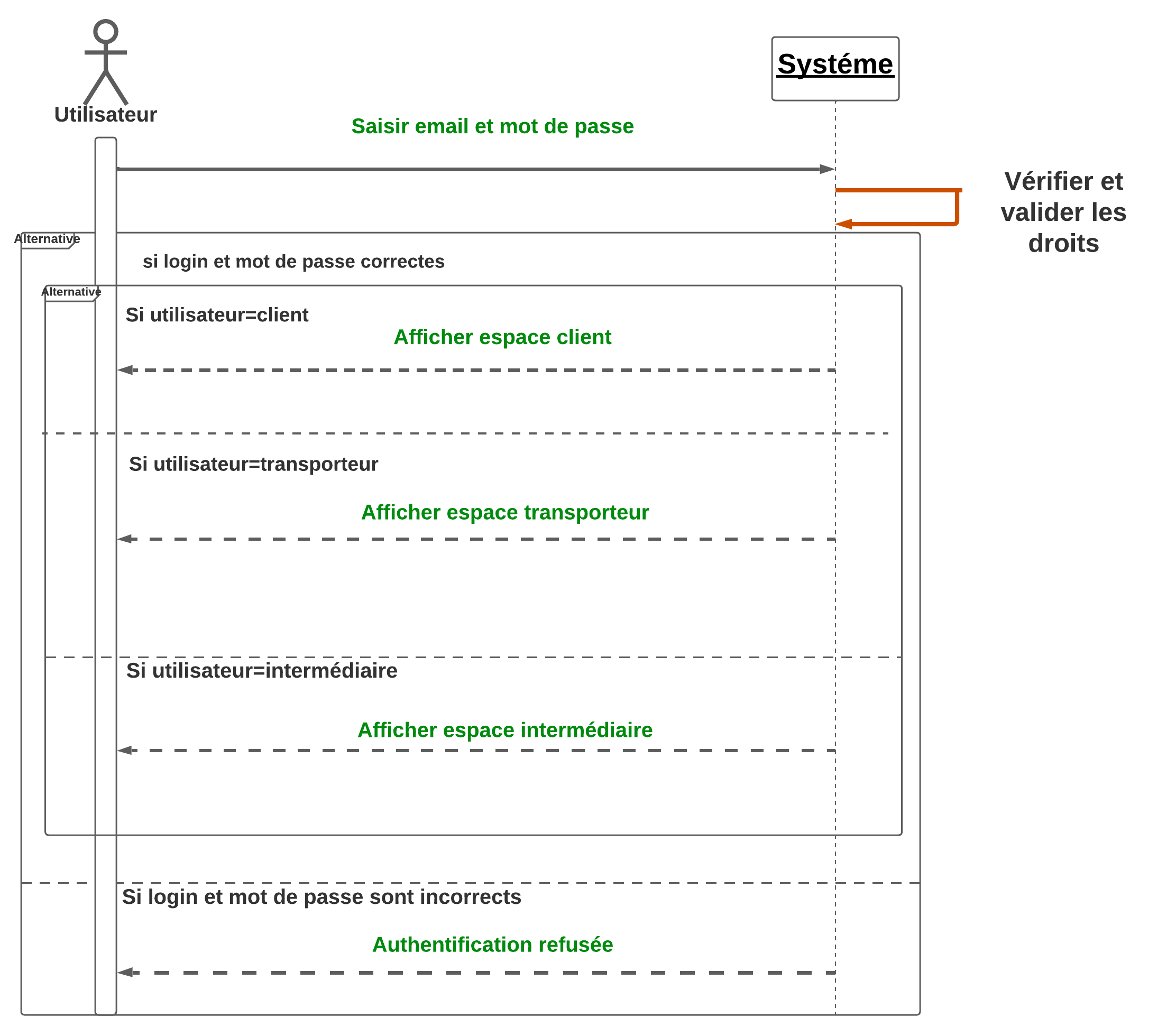
La figure ci-dessous représente le diagramme de séquence objet relatif au cas « S’authentifier » :

Figure 13 : diagramme de séquence authentification

* + Diagramme de séquence système du cas d’utilisation « Gérer les demandes »

Le transporteur se connecte au système pour consulter les demandes envoyées, le système lui affiche la page demandes. Deux cas sont possible :

* Si le transporteur accepte la demande un formulaire suggestion offre s’affiche et il doit le remplir correctement si non un message d’erreur s’affiche sur l’écran.
* L’offre sera envoyée à l’intermédiaire, il le traite et la transmettre au client.
* Le client choisi une seule offre à accepter, une fois une offre est acceptée les autres seront automatiquement annulées.
* Lorsque la livraison est en cours, le transporteur envoi le facture à payer, cette facture est transmise au client, le client envoi un reçu de paiement.

La figure ci-dessous représente le diagramme de séquence objet relatif au cas « Gérer demandes » :

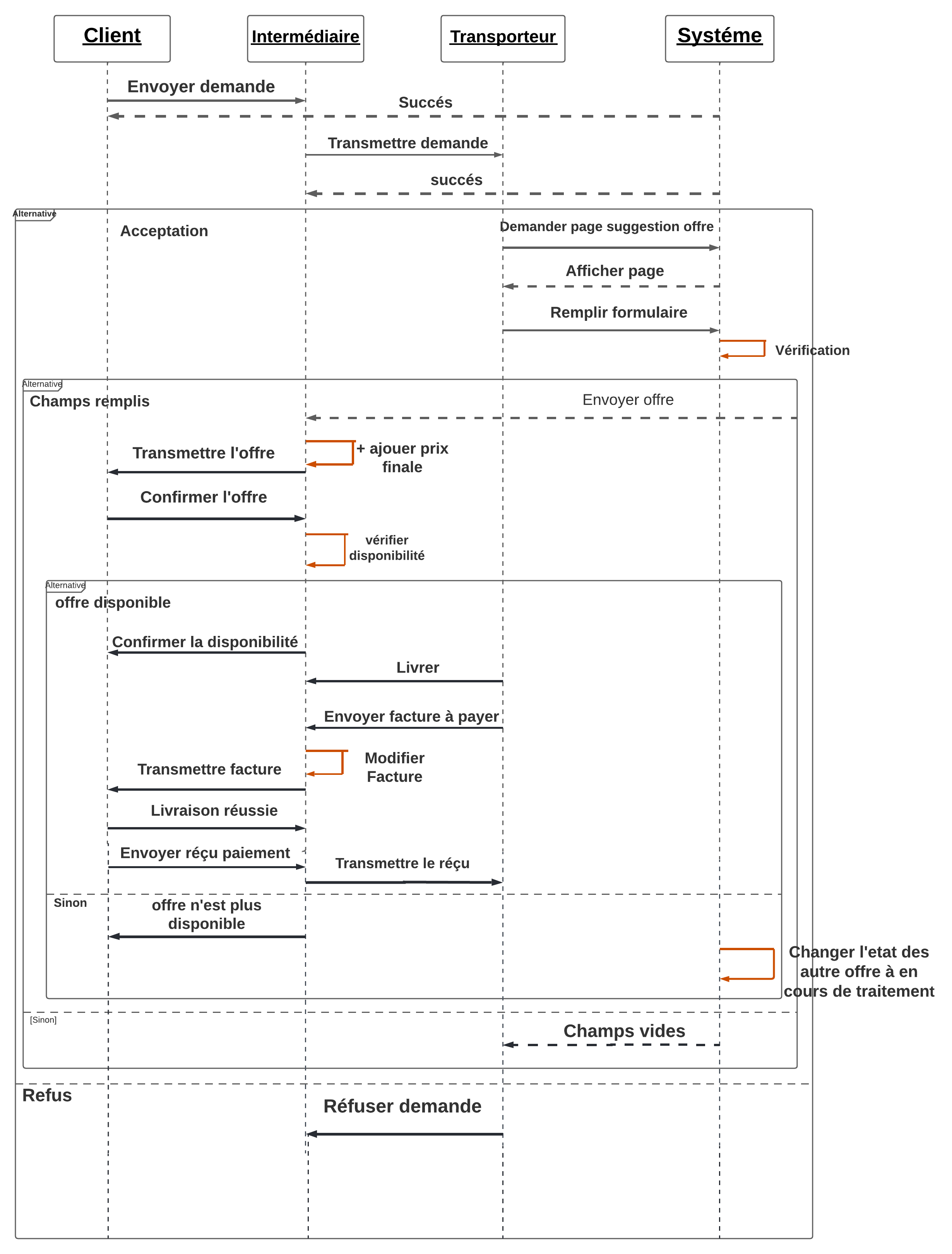


Figure 14 : diagramme de séquence de suivi livraison

* + - * Diagramme de séquence système du cas d’utilisation « Gérer les trajets »

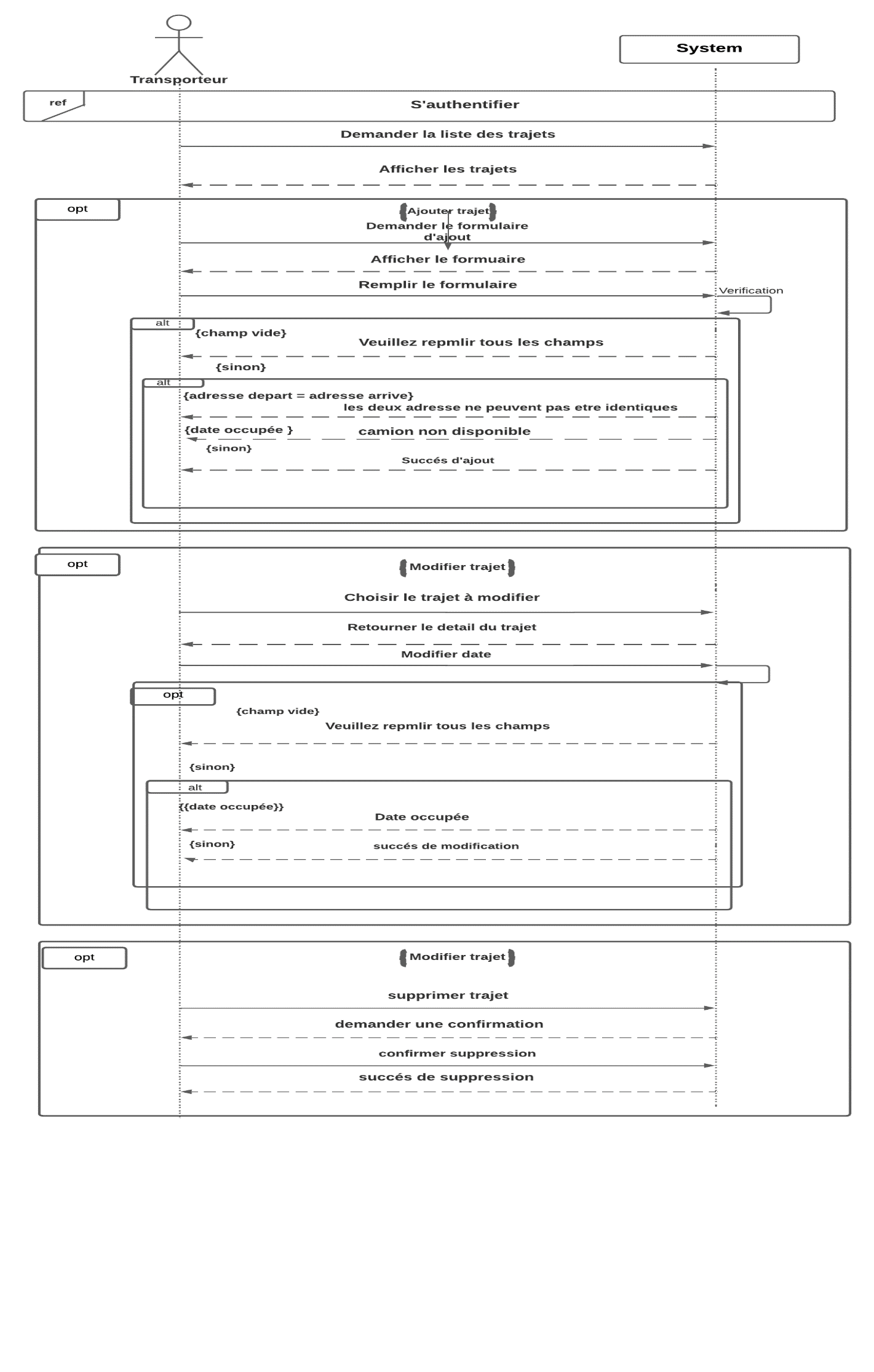


Figure 15 : diagramme de séquence de gestion des trajets de retour

* Diagramme d’activité

Le diagramme d’activité représente les règles d’enchaînement des activités et des actions dans le système. Il permet d’une part de consolider la spécification d’un cas d’utilisation et d’autre part de concevoir une méthode. [B2]

Dans ce qui suit, nous allons présenter des diagrammes d’activités de quelques cas d’utilisation.

* **Diagramme d’activité relatif au cas « Modifier profile »**

Dans la figure ci-dessous nous présentons le diagramme d’activité relatif au cas « modifier profil » :

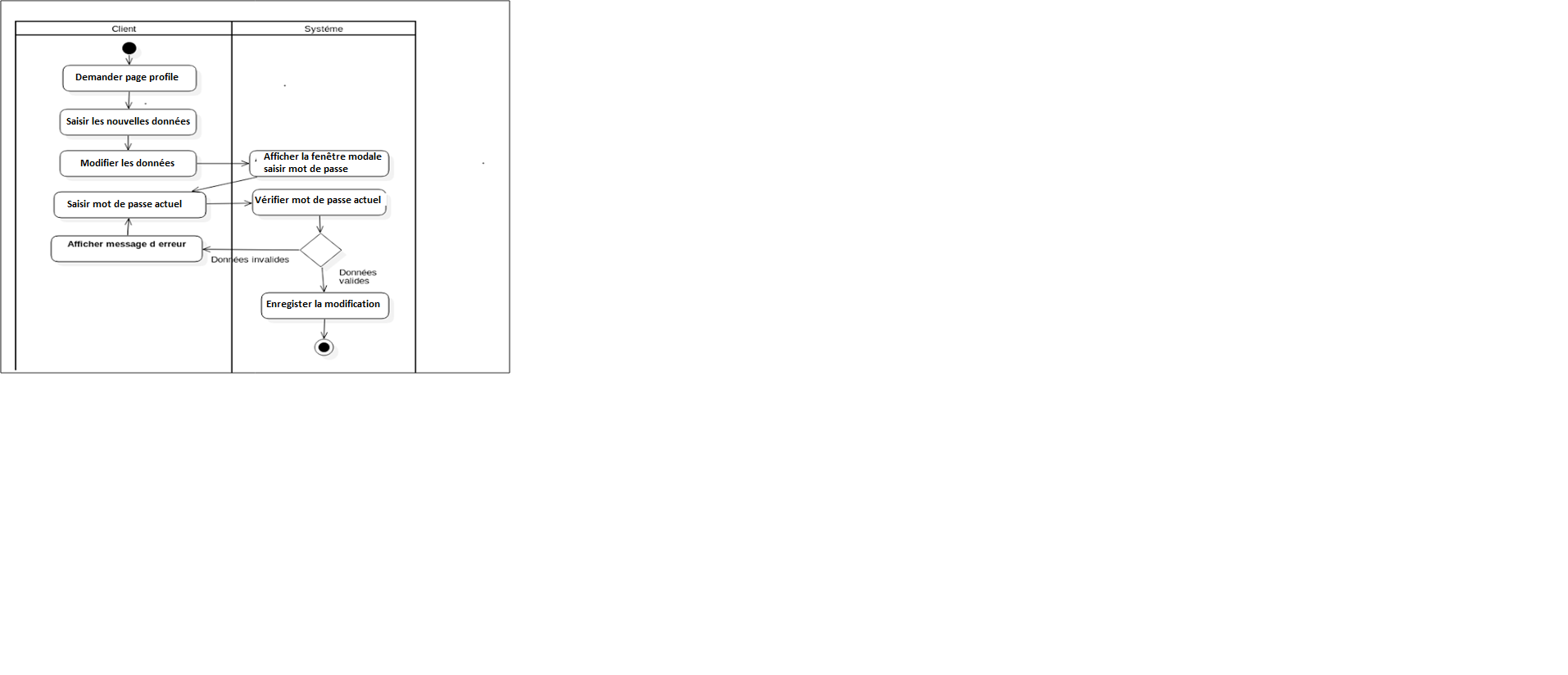


Figure 16 : diagramme d’activité de modification profil

* Diagramme d’activité relatif au cas « Vérifier la disponibilité de l’offre »

Dans la figure ci-dessous nous présentons le diagramme d’activité relatif au cas « modifier profile » :



Figure 17 : Diagramme d’activité relatif au cas vérifier la disponibilité de l’offr

* Diagramme d’états-transitions

Un diagramme d’états-transitions est une description des changements d’états d’un objet en réponse aux interactions avec d’autres objets ou avec des acteurs.

La figure ci-dessous décrit l’objet offre dans leur interaction avec les acteurs transporteur, client et intermédiaire.

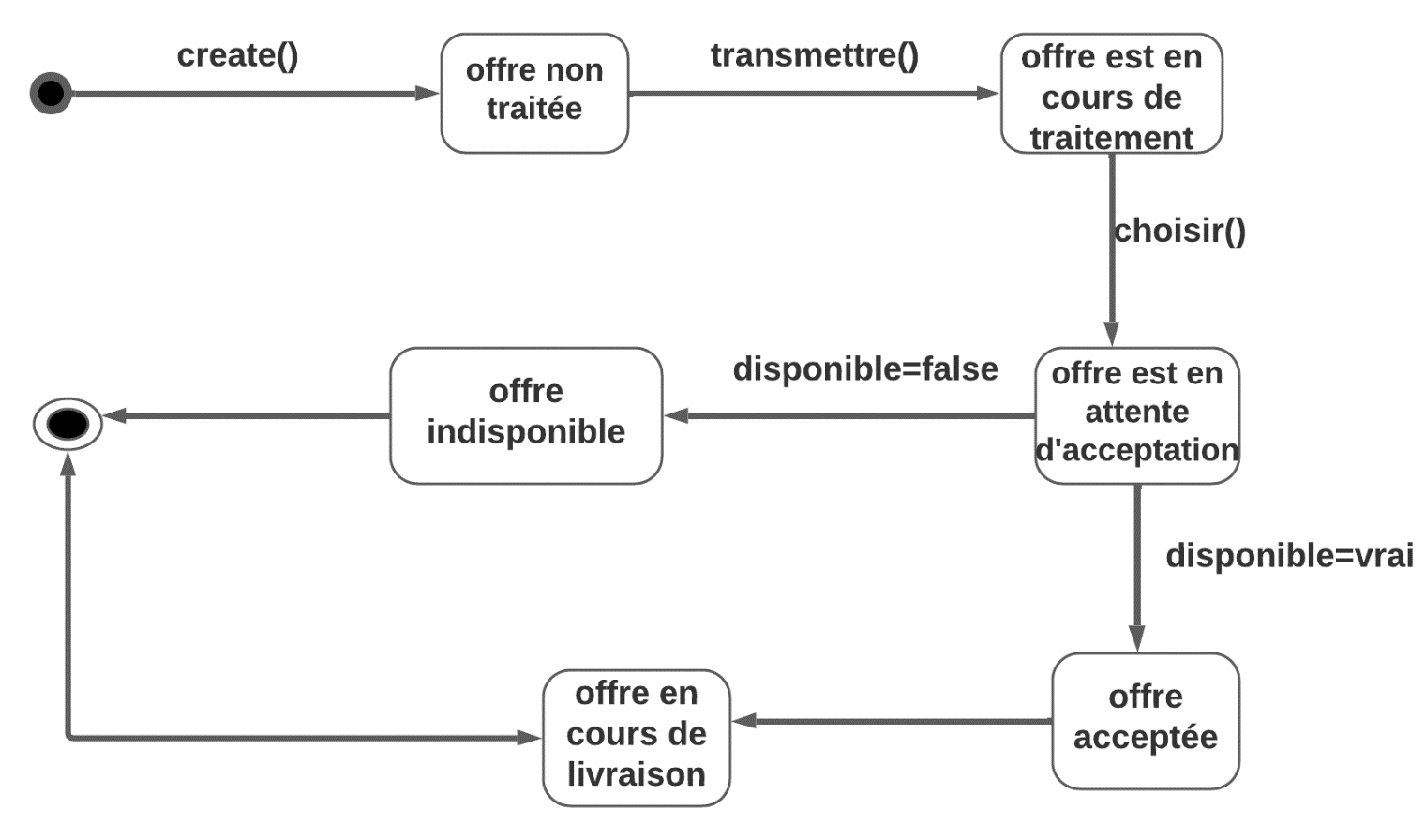


Figure 18 : Diagramme d’états-transitions

## Conclusion

Dans ce chapitre, nous avons présenté une grande partie de l'application, ainsi la présentation de diagramme de classe, diagramme de séquence, diagramme d’états-transition et le diagramme d’activité. Dans le chapitre suivant, nous allons terminer notre projet avec la présentation de réalisation.

|  |  |
| --- | --- |
| **Chapitre 4** | **Réalisation du système** |

Chapitre 4 : Réalisation du système

## Introduction

Dans ce chapitre nous allons décrire l’environnement de travail logiciel et matériel permettant la réalisation de notre application. Par la suite, nous présenterons le travail réalisé et les résultats obtenus.

## Environnement de développement

### 1.1 Environnement matériel

Notre application a été réalisée par deux ordinateurs ayant les caractéristiques suivantes :

|  |  |  |
| --- | --- | --- |
| **Marque** | **Dell** | **HP** |
| **Processeur** | Intel® Core ™ i5-6300U CPU @ 2.40GHz 2.50GHz | Intel® Core ™ i5-8265U CPU @ 1.60GHz 1.80GHz |
| **RAM** | 16 Go | 16 Go |
| **Disque Dur** | 256 Go SSD | 480 Go SSD |
| **Carte Graphique** | Intel ® HD Graphics 520 | Intel ® UHD Graphics 620 |
| **Système d’exploitation** | Windows 10 x64 | Windows 11 x64 |

Tableau 9 : Environnement matériel

### 1.2 Environnement logiciel

La mise en œuvre de nos applications Web et Mobile nécessite plusieurs logiciels. Nous présenterons en détail les différents outils utilisés dans le processus de développement de notre projet sous forme d’un tableau.

|  |  |  |
| --- | --- | --- |
| **Nom de Logiciel** | **Logo** | **Utilisation** |
| **Visual Studio Code** |  | **Visual Studio Code** est un éditeur de code extensible développé par Microsoft.[4] |
| **Visual Studio 2022** |  | Est l’IDE que nous avons utilisée pour le développement du module Web Api. Il s’agit d’un éditeur de code contenant des fonctionnalités de déverminage et de déploiement. Visual Studio comporte aussi différentes extensions permettant de déployer facilement l’API sur Azure. [5] |
| **Lucidchart** | Lucidchart - 2022 Reviews, Pricing, Features  How to Change SQL Server Authentication Mode using SSMS | **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. [6] |
| **SQL Server Management Studio (SSMS)** |  | **SQL Server Management Studio** est une application logicielle lancée pour la première fois avec Microsoft qui est utilisée pour configurer, gérer et administrer tous les composants de Microsoft SQL Server. [7] |
| **GitHub** |  | C’est le service web d’hébergement et de gestion de développement de logiciels. [8] |
| **Postman** | Postman Logo PNG Vector (SVG) Free Download | Permet de lancer une série de requêtes HTTP les unes après les autres en les enregistrant dans une collection puis en exécutant cette collection. [9] |
| **Android Studio** | Icon Android Studio Logo, HD Png Download , Transparent Png Image - PNGitem | Android Studio est un environnement de développement pour développer des applications mobiles Android. Il est basé sur IntelliJ IDEA et utilise le moteur de production Gradle. [10] |

Tableau 10 : Environnement Logiciel

## Choix techniques

### 2.1 Outils de développement

* L’application back end
* .Net Core :

Pour développer notre système de suivi, nous avons utilisé la technologie .Net Core de Microsoft, version 3.1. Ce choix est imposé par l’entreprise qui a déjà mis en place des applications en utilisant la technologie Microsoft et qui possède donc les licences requises.

.NET : est un [Framework](https://fr.wikipedia.org/wiki/Framework) [Libre et Open Source](https://fr.wikipedia.org/wiki/Free/Libre_Open_Source_Software) pour les systèmes d'exploitation [Windows](https://fr.wikipedia.org/wiki/Windows), [macOS](https://fr.wikipedia.org/wiki/MacOS) et [Linux](https://fr.wikipedia.org/wiki/Linux)[3](https://fr.wikipedia.org/wiki/.NET_Core#cite_note-3).

Bien que .NET Core partage un sous-ensemble d'API [.NET Framework](https://fr.wikipedia.org/wiki/.NET_Framework), il est livré avec sa propre [API](https://fr.wikipedia.org/wiki/Interface_de_programmation) qui ne fait pas partie de [.NET Framework](https://fr.wikipedia.org/wiki/.NET_Framework).

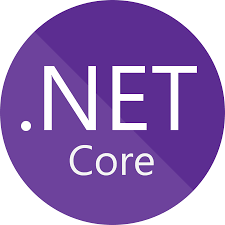


Figure 19 : .Net Core

Pourquoi .net Core ?

Les raisons principales données par Microsoft :

* Modularité : n’utilisez que ce dont vous avez besoin.
* Moderne : il s’agit de la dernière version de .NET
* À jour : le déploiement via Nugget permet des mises à jour plus fréquentes et lorsque vous (le développeur) le décidez.
* Cross-plateformes : en utilisant .Net Core vous vous assurez que votre code sera portable aisément sur d’autres plateformes.
* Performance : les gains annoncés et constatés sont au rendez-vous.
* Application Front-end
* Front web
* Vue.js :

[Vue.js](https://mobiskill.fr/emploi/developpeur-vue-js/) est décrit comme « un Framework progressif pour la création d’interfaces utilisateur ». Ce qui différencie ce Framework des autres Framework de développement web [JavaScript](https://mobiskill.fr/emploi/developpeur-javascript/) est que son architecture est conçue pour être adaptable de manière incrémentielle. [11]



Figure 20 : Vue.js

Pourquoi Vue.js ?

L’écosystème Vue.js se compose d’une bibliothèque de base, de Framework et d’autres outils qui permettent un développement front-end facile et rapide.

Bien que Vue, lui-même, soit léger et minimaliste, il donne tous les outils aux développeurs pour créer des applications web très fonctionnelles.

Il existe d’autre raison comme :

* La flexibilité

L’écriture d’une application à l’aide de Vue.JS est rapide dû au fait qu’il est possible de l’exécuter via son navigateur. Cela facilite également le processus de test.

* L’approche simple

Il est très facile d’intégrer Vue.JS dans un projet Web existant. Ceux qui ont le savoir-faire de JavaScript et de HTML sont encore plus à l’aise avec ce Framework. Le passage à ce Framework est donc très facile et avantageux pour un environnement de développement rapide.

* Petite taille
* Documentation détaillée
* **Front Mobile**
* **Ionic :**

Ionic est un Framework et un mélange d’outils et de technos pour développer des applications mobiles hybrides rapidement et facilement. Il s’appuie sur AngularJS pour la partie application web du Framework et sur Cordova pour la partie construction des applications natives. [12]



Figure 21 : Ionic

**Pourquoi Ionic ?**

* Rapidité, facilité et qualité sont donc les points forts de ce Framework open source permettant de développer une application pour plusieurs environnements simultanément.
* Une large communauté : Avec Ionic, il y a quasiment peu de chance que cela vous arrive. En plus d'une documentation déjà très riche et simple à prendre en main, le Framework fédère une très large communauté de développeurs enthousiastes près à vous aider en cas de pépin.
* Coder une fois, Déployer partout : l’un des plus gros avantages de Ionic : la possibilité de développer votre application une fois, et la déployer sur plusieurs terminaux mobiles. Alors que la plupart des outils de développement d'applications mobiles hybrides sont difficiles d'accès, parfois peu ou mal documentés, Ionic propose une architecture beaucoup plus simple utiliser et une documentation claire et concise.
* Des composants logiciels élégants : Le Framework propose de nombreux composants et Template qui vous faciliteront énormément la vie. Pas besoin d'être un expert UX pour commencer à créer des applications mobiles élégantes et ergonomiques : quelques clics et quelques lignes de codes suffisent.
* VueX:

**VueX** est une bibliothèque de gestion d’état pour Vue.js. Elle s’intègre aux applications Vue en créant une interface propre et isolée pour lier les données situées aux données de l’API Web.

Les bibliothèques comme VueX permettent de gérer les états partagés entre les composants d’une manière structurée et facile à maintenir. Elles créent un arbre d’état global auquel chaque composant peut accéder et qu’il peut mettre à jour de manière progressive. [13]



Figure 22 : VueX

* **TWILIO**

Twilio est une entreprise américaine spécialisée dans les [communications unifiées](https://fr.wikipedia.org/wiki/Communications_unifi%C3%A9es) via une plateforme cloud. Twilio permet aux développeurs de logiciels de créer et de recevoir des appels téléphoniques, d'envoyer et de recevoir des messages texte et d'effectuer d'autres fonctions de communication à l'aide de ses API de [service Web](https://fr.wikipedia.org/wiki/Service_web). [14]

****

Figure 23 : Twilio

* **MailTrap**

Mailtrap est un outil de test et n'est pas conçu pour envoyer des e-mails à de vraies adresses. Cependant, vous pouvez transférer certains messages à vos collègues, clients ou à vos propres boîtes de réception à des fins de test. [15]

****

Figure 24 : Mailtrap

* **Intelliticks**

IntelliTicks est une plate-forme de chat en direct alimentée par l'IA pour les ventes (avec une solution de secours humaine). Il aide les entreprises à obtenir 2 fois plus de prospects en étant 70 % plus en ligne que le chat en direct traditionnel. [16]



Figure 25 : IntelliTicks

* **API RESTful**

Une API compatible REST, est une interface de programmation d'application qui fait appel à des requêtes HTTP pour obtenir (GET), placer (PUT), publier (POST) et supprimer (DELETE) des données. [17]

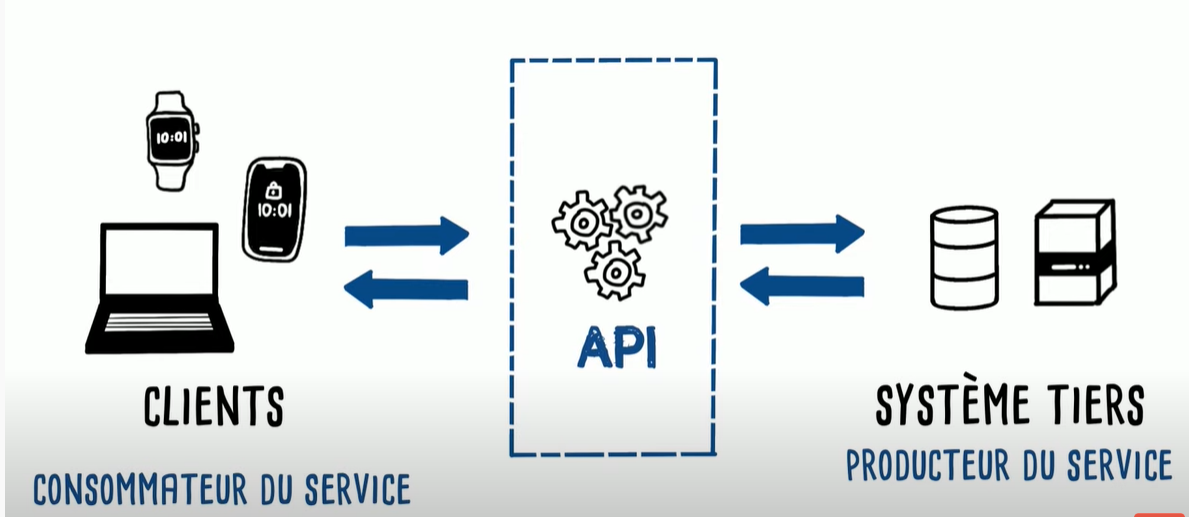


Figure 26 : Api Restful

1. Déploiement

* **Machine virtuelle Azure :**

Les machines virtuelles Azure sont des instances de service d'images qui fournissent des ressources informatiques à la demande et évolutives, avec une tarification basée sur l'utilisation.

Plus largement, une machine virtuelle se comporte comme un serveur : il s'agit d'un ordinateur à l'intérieur d'un ordinateur qui offre à l'utilisateur la même expérience que celle qu'il aurait sur le système d'exploitation hôte. En général, les machines virtuelles sont placées dans un « bac à sable » qui les isole du reste du système, de sorte que les logiciels installés sur la machine virtuelle ne peuvent ni s’échapper, ni modifier le serveur sous-jacent.

Chaque machine virtuelle fournit son propre matériel virtuel, à savoir les processeurs, la mémoire, les disques durs, les interfaces réseau et les autres périphériques nécessaires. [18]

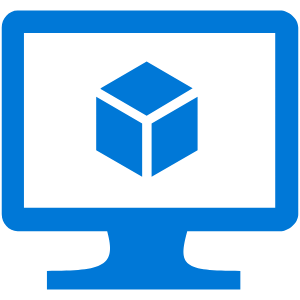


Figure 27 : Machine virtuelle Azure

* Ngrok

Ngrok est un utilitaire puissant qui permet aux utilisateurs de partager localement ou d’accéder à une ou plusieurs applications Web hébergées localement via une URL Web accessible au public. Cette URL publique est ensuite exposée au domaine Ngrok Io, puis tunnel le trafic qui arrive à ce point final à travers un port local host spécifié. [19]



Figure 28 : Ngrok

* **Netlify :**

Netlify est une entreprise américano-danoise fondée en 2014 qui propose des services d'hébergement et de cloud computing sans serveur pour les sites web statiques. [20]

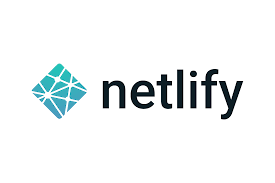


Figure 29 : Netlify

* **Azure portale :**

Une console web unifiée qui offre une alternative aux outils en ligne de commande. [21]

****

Figure 30 : Azure portale

## Réalisation de l’application

Afin de tester notre application, nous présentons dans cette partie des figures qui illustre les captures écrans sur les différentes parties de notre application.

### 4.1 Site web :

- La figure suivante illustre l’interface correspond à la page login :

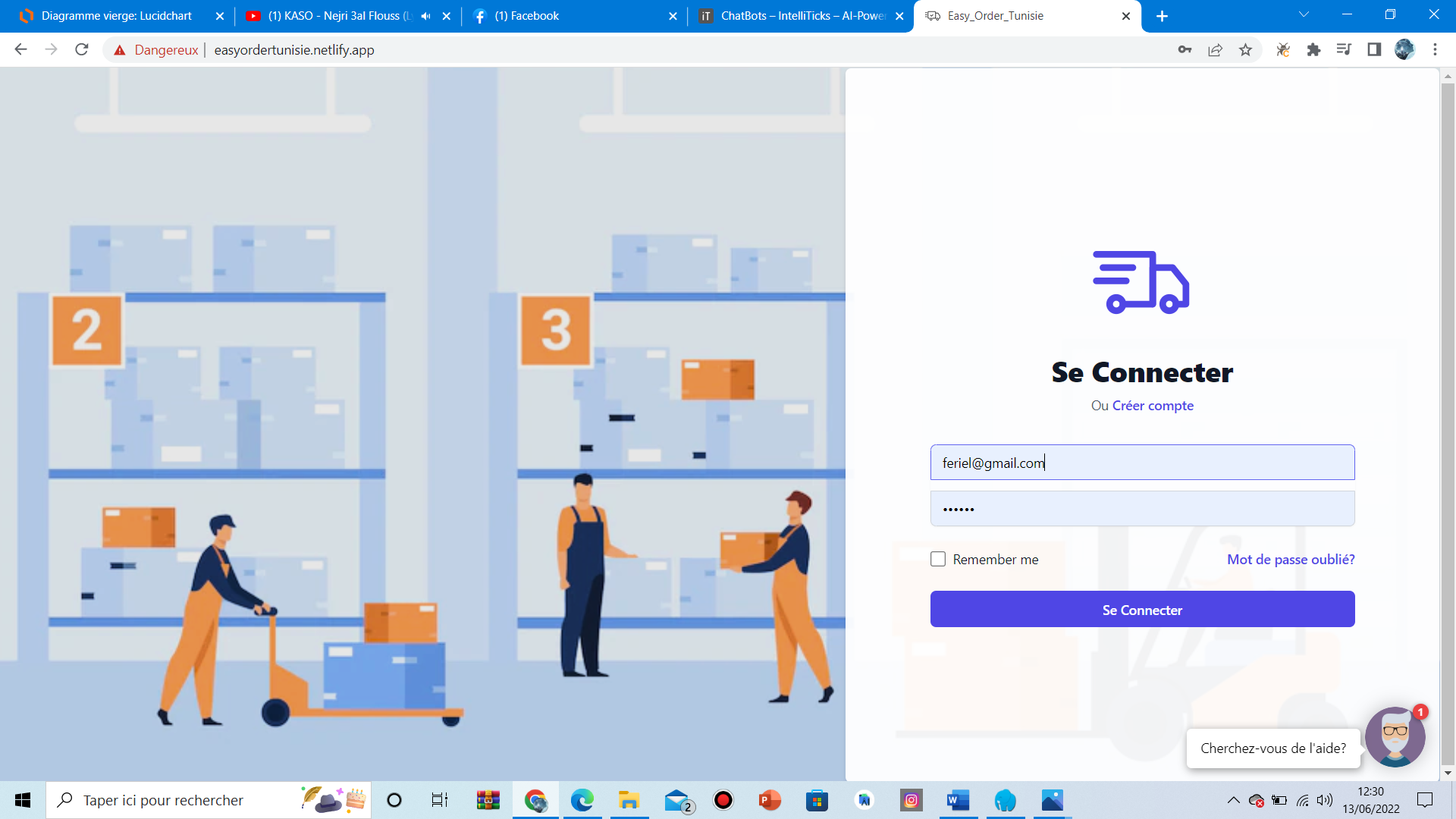


Figure 31 : Interface de login

* La figure suivante illustre l’interface correspond à la page d’inscription :

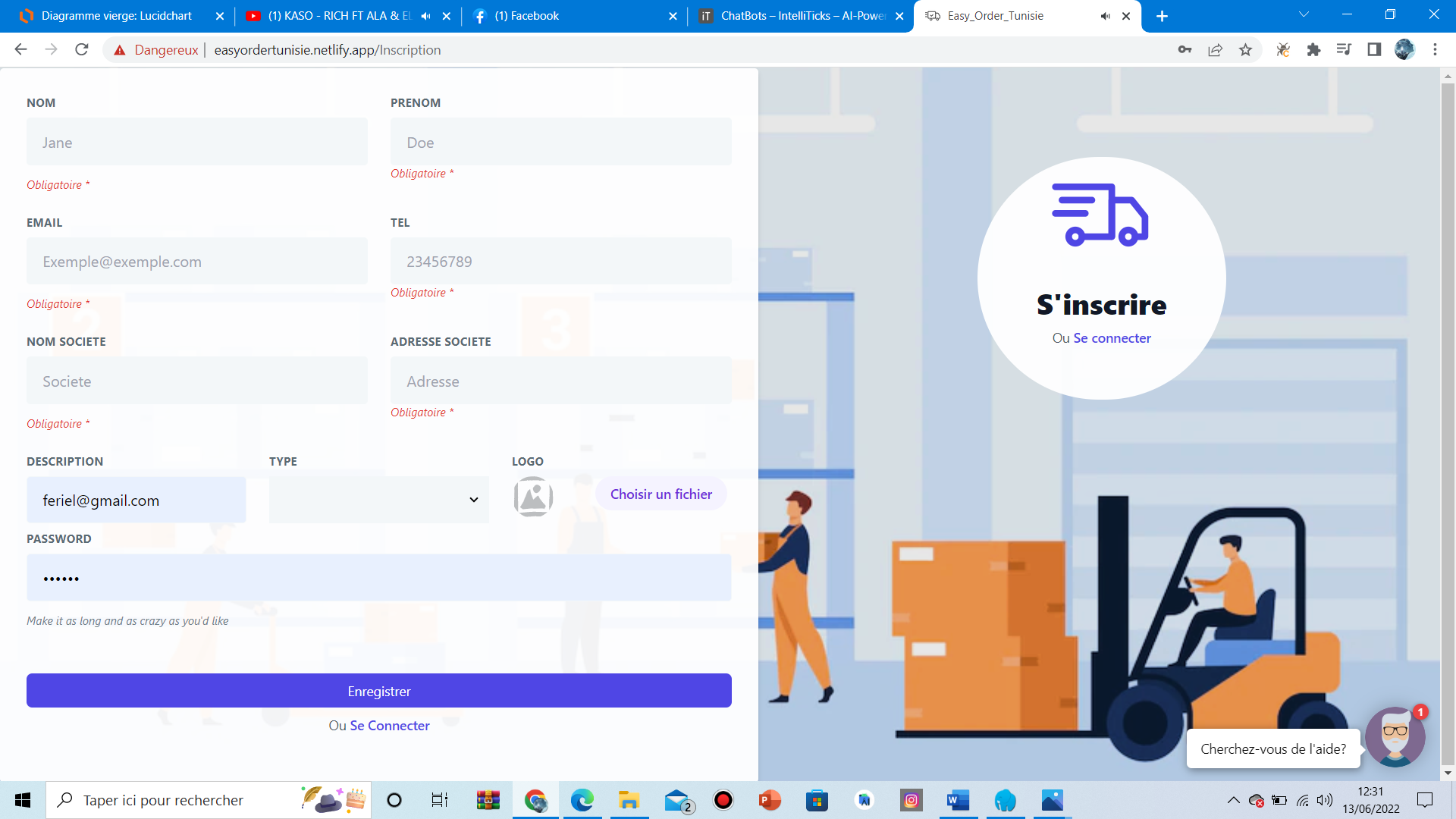
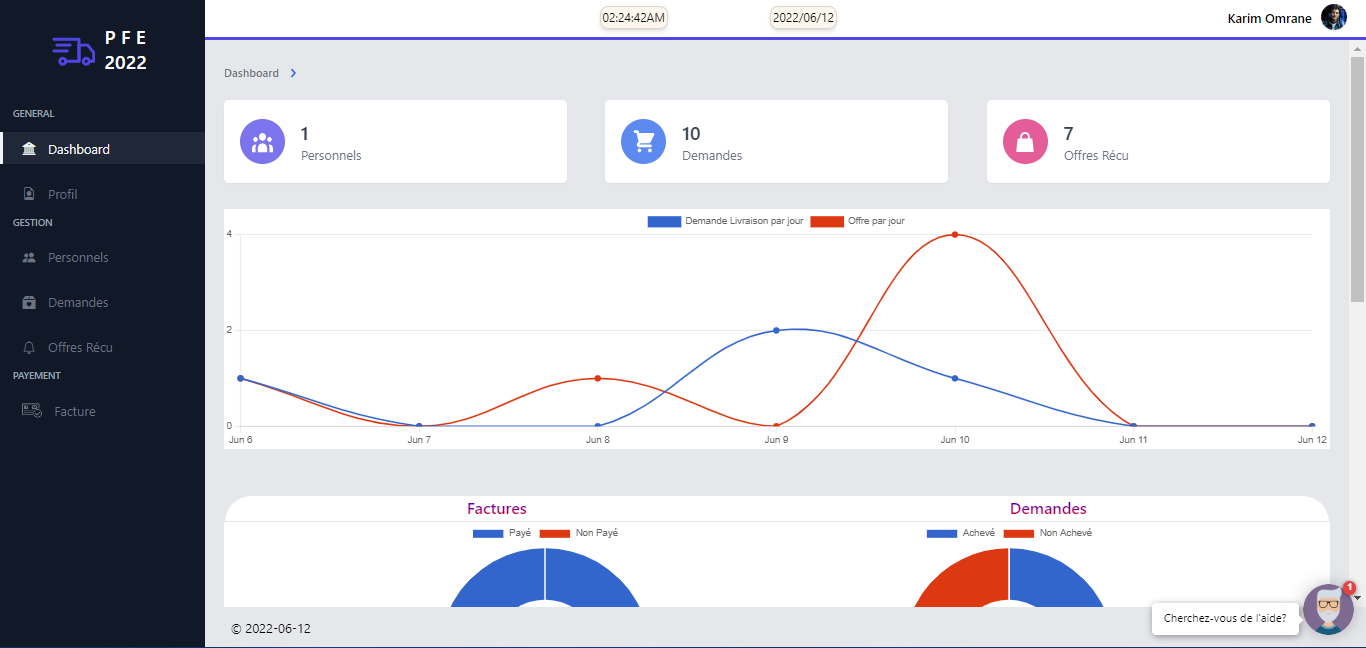


Figure 32 : Interface d’inscription

* **Partie Client :**
* La figure suivante illustre l’interface correspond au tableau de bord :



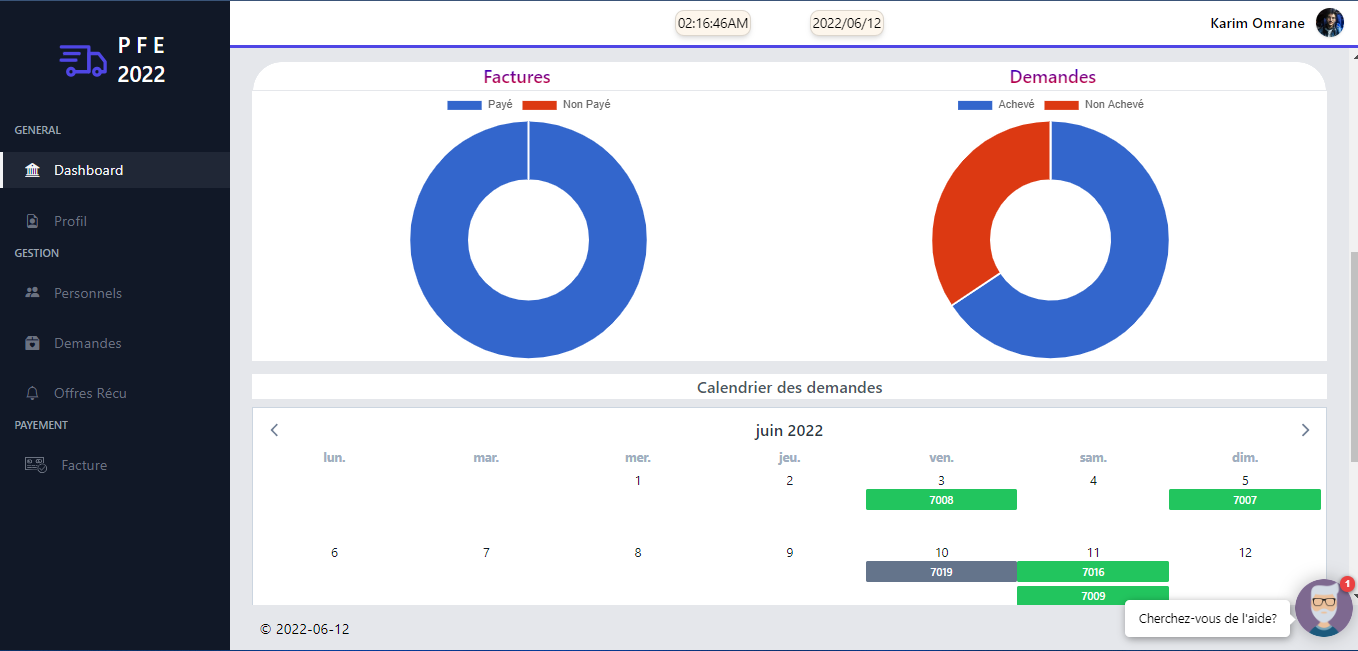
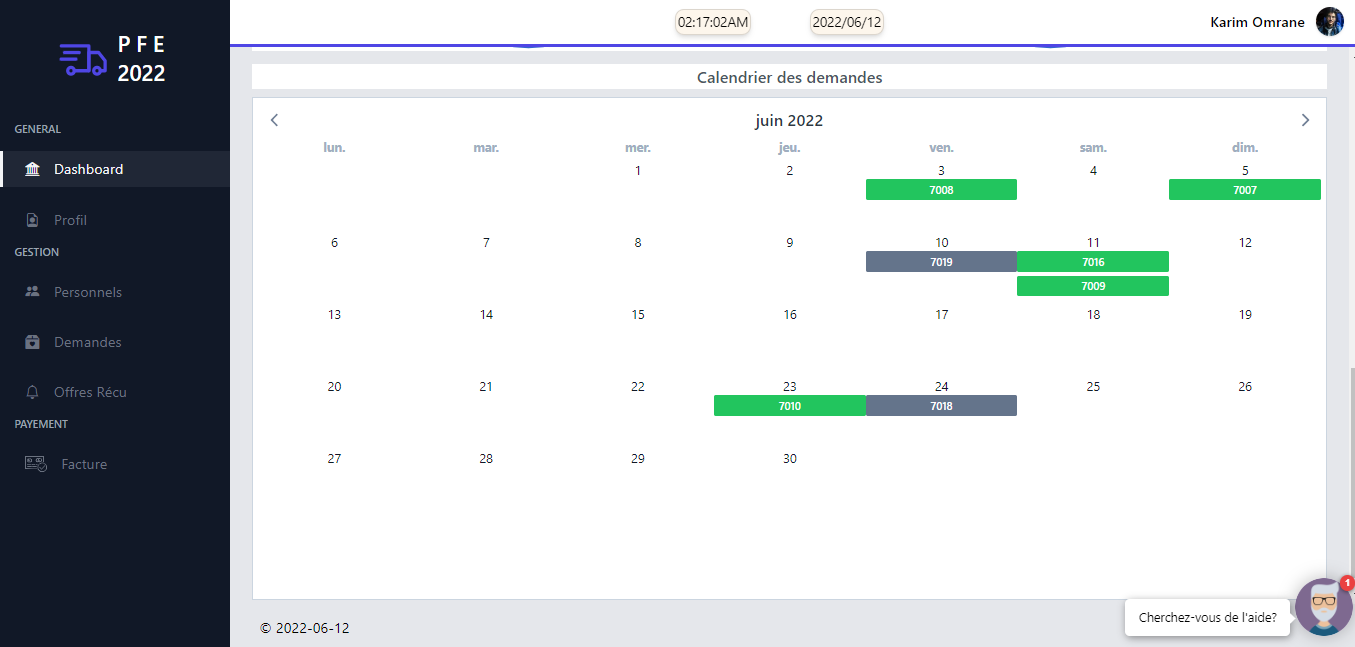


Figure 33 : Interface de tableau de bord

* Les figures suivantes illustrent l’interface correspond au profil du client. Le client peut modifier son nom et prénom, son email, son numéro et son mot de passe ainsi que son image :



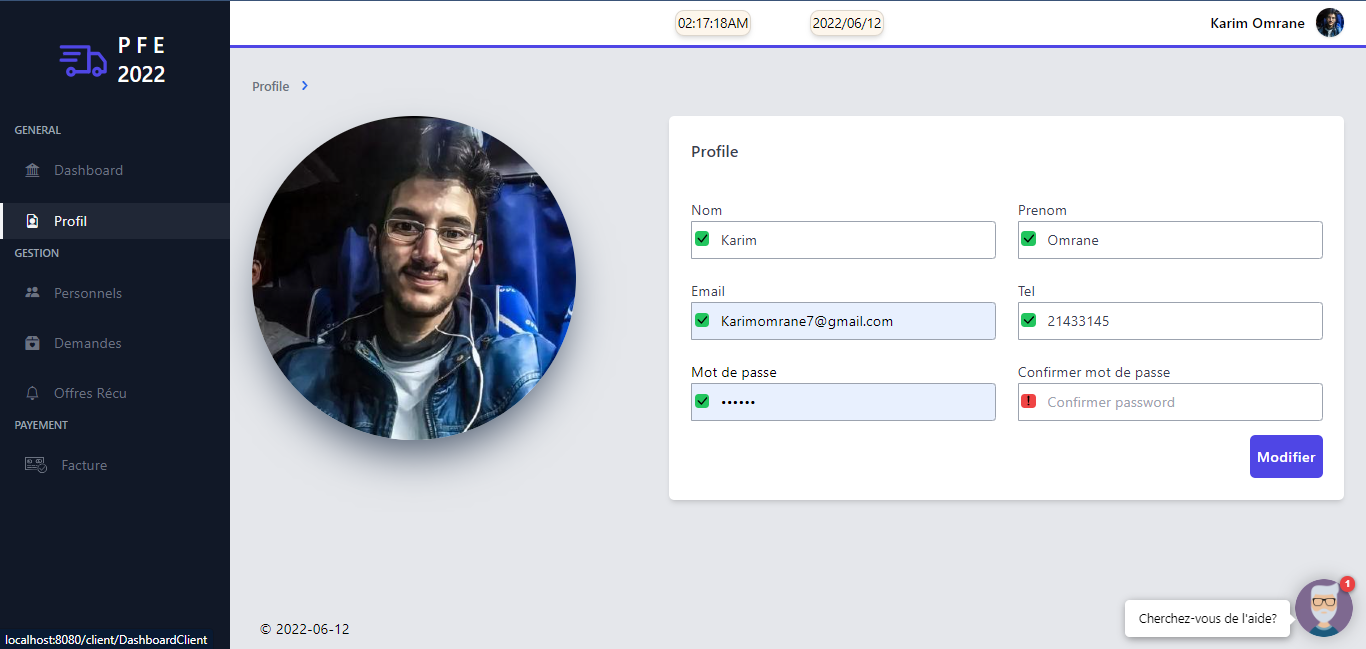


Figure 34 : Interface de profil

* La figure suivante illustre l’interface correspond à la page de personnels où le

client peut consulter la liste de personnels :

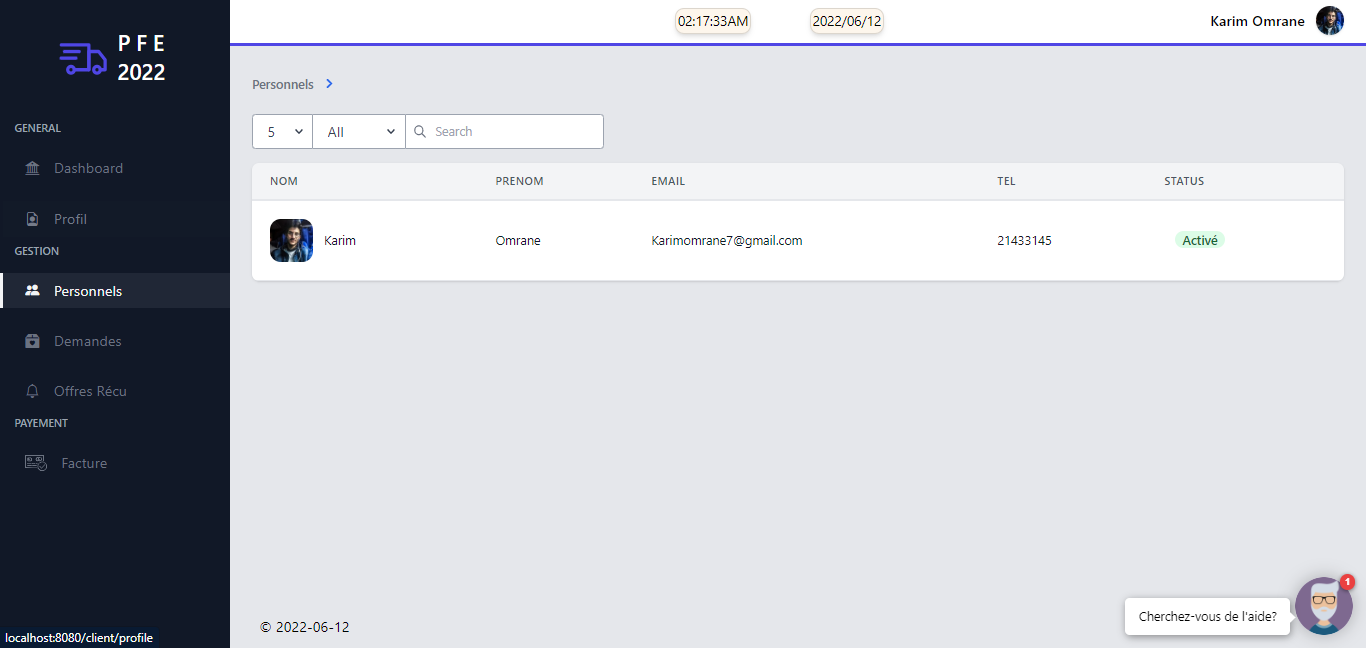


Figure 35 : interface de consultation des personnels

* La figure suivante illustre l’interface correspond à la page de demandes. Le client peut consulter, ajouter et changer l’état des demandes :

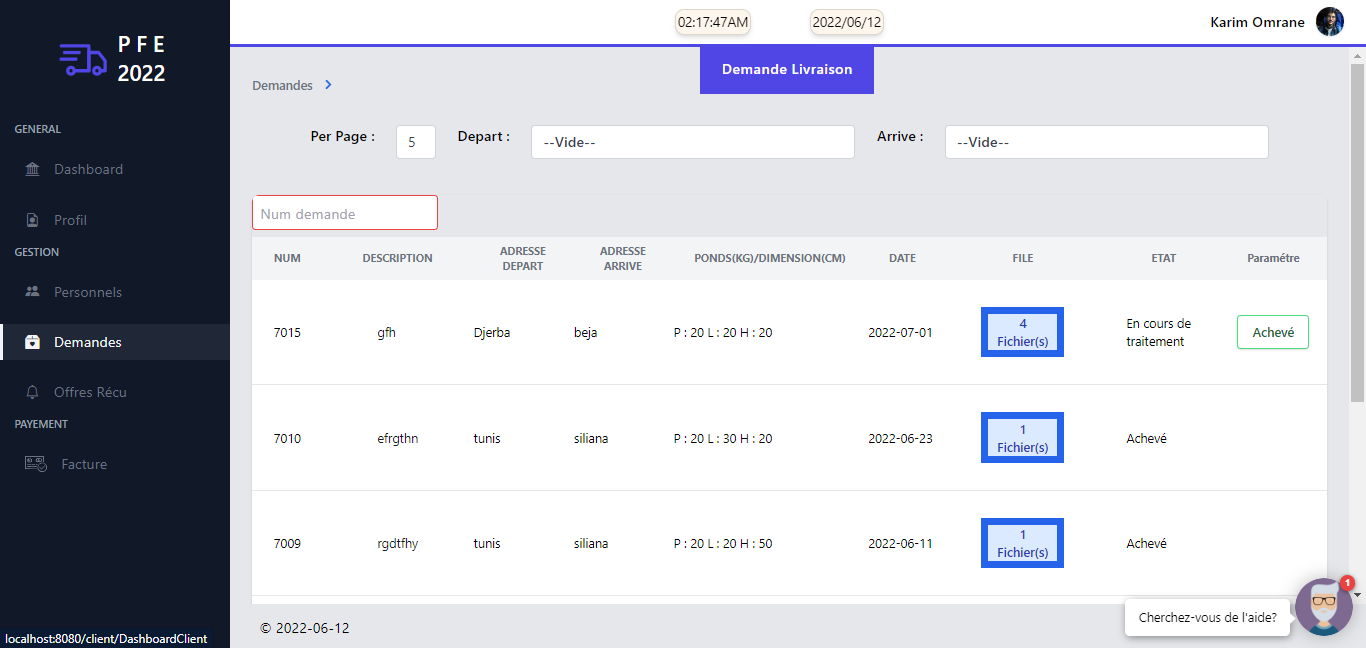


Figure 36 : Interface de gestion de demandes

* La figure suivante illustre l’interface correspond au formulaire d’ajout d’une demande. Le client doit saisir la description, le poids, l’adresse du départ et l’adresse de l’arrive, largeur, hauteur, date et une ou plusieurs images.

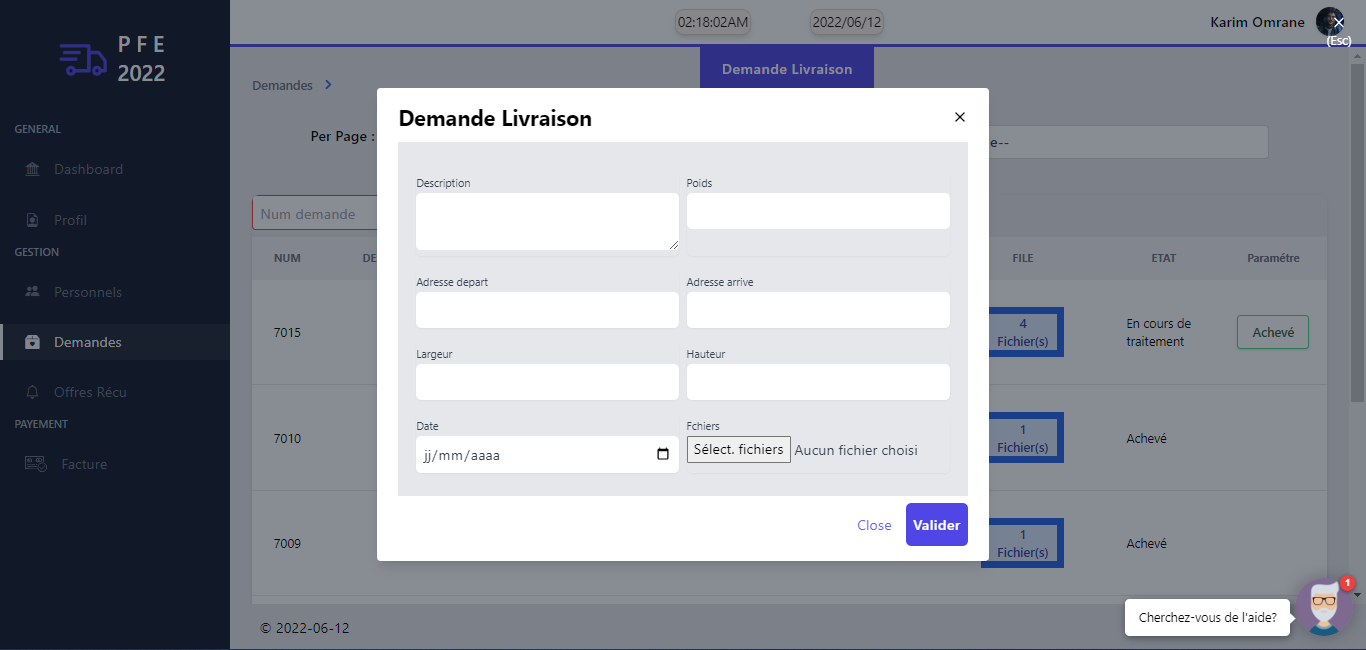


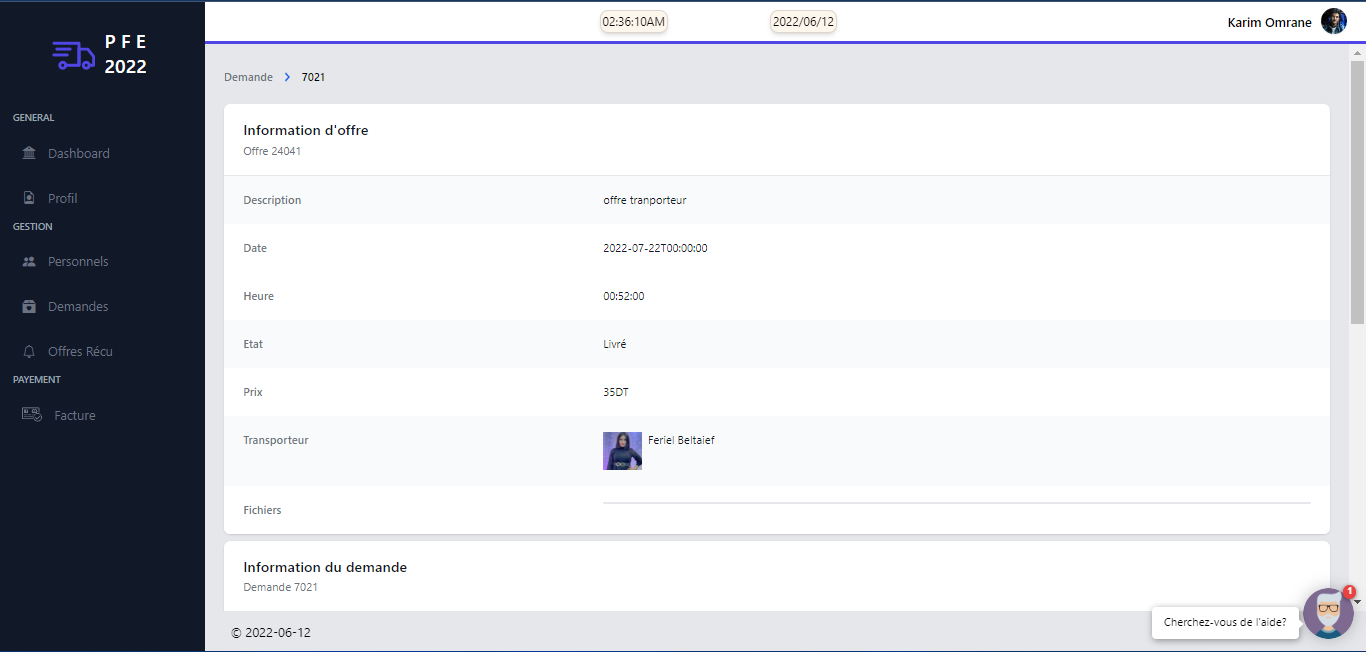
Figure 37 : Interface d’ajout une demande

* La figure suivante illustre l’interface correspond à la page d’offres. Le client peut consulter, accepter et annuler les offres :

###### 

Figure 38 : Interface des offres

* Les figures suivantes illustrent l’interface correspond à la page d’offre. Le client peut consulter l'offre et sa demande correspondante en détaille :



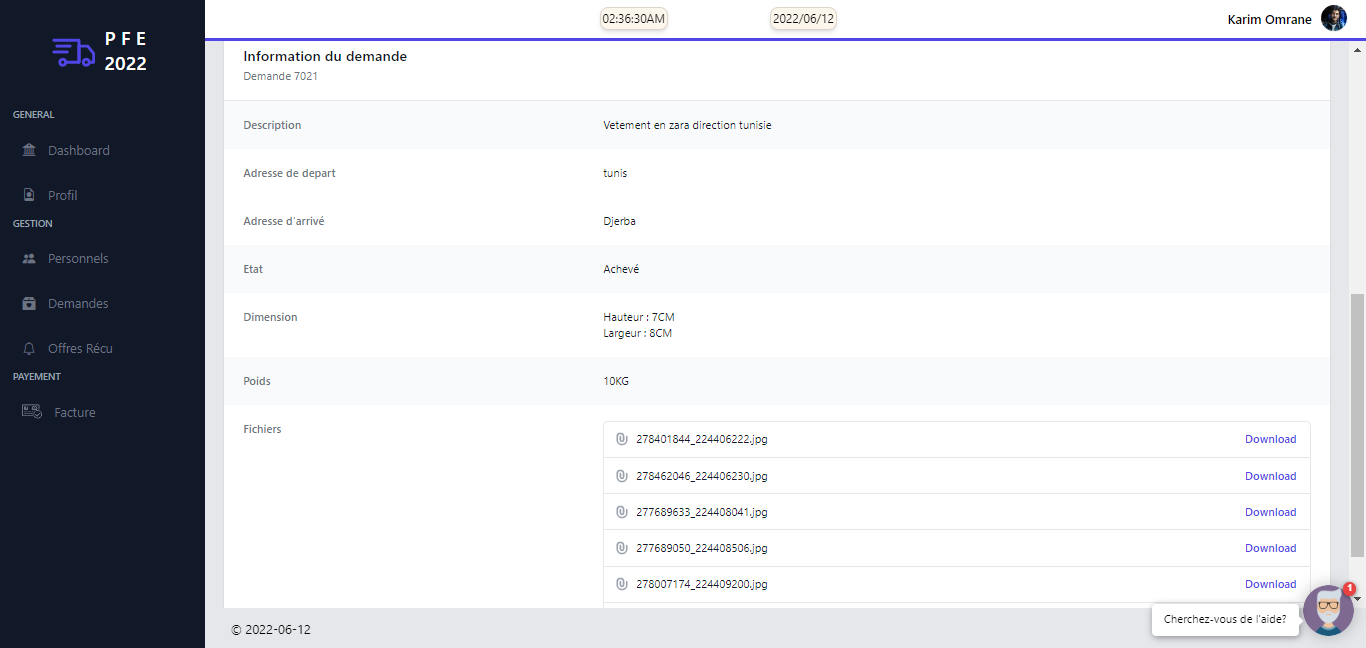
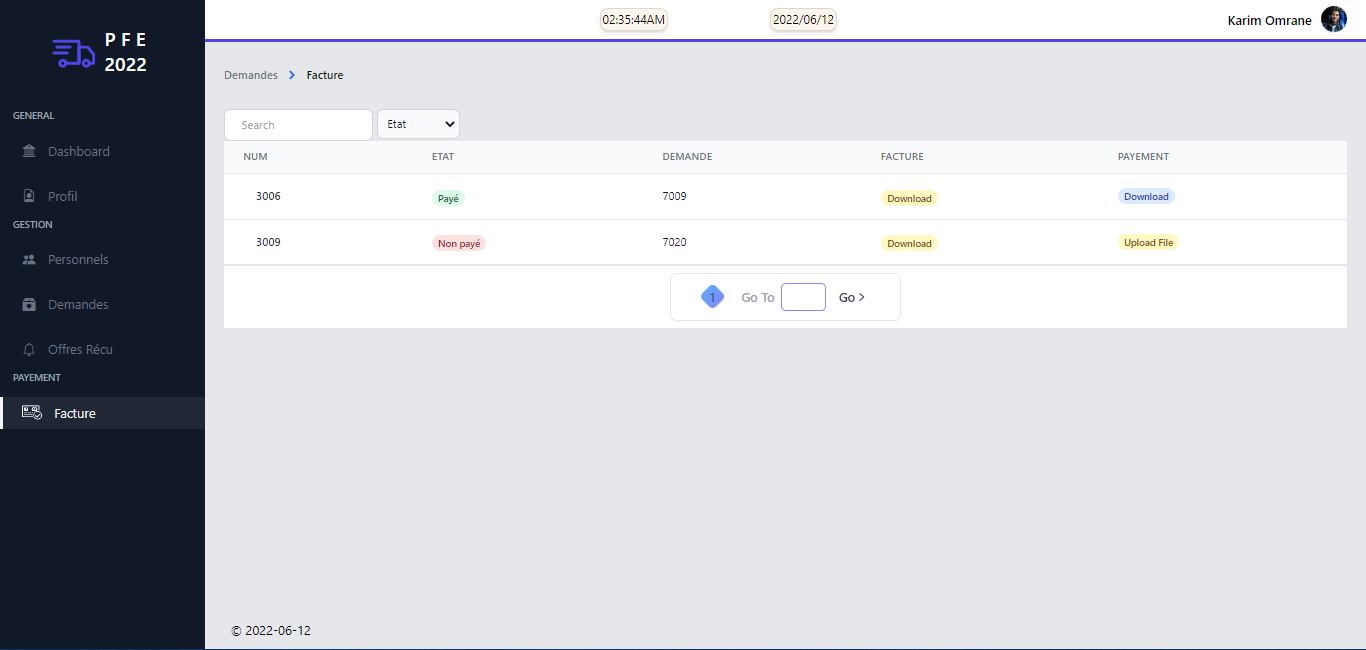


Figure 39 : Interface d’offre en détaille

* Les figures suivantes illustrent l’interface correspond à la page de factures. Le client peut consulter les factures à payer et valider le paiement :



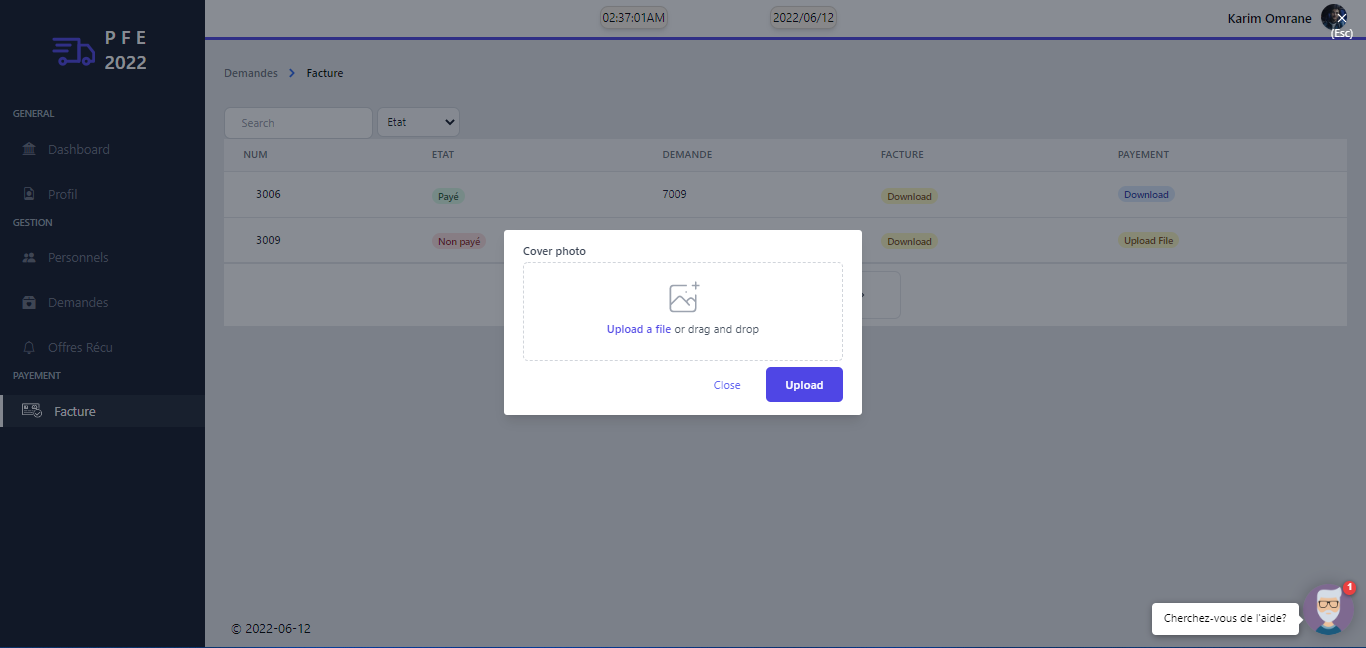


Figure 40 : Interface de facture

* **Partie Transporteur :**
* La figure suivante illustre l’interface correspond au tableau de bord :

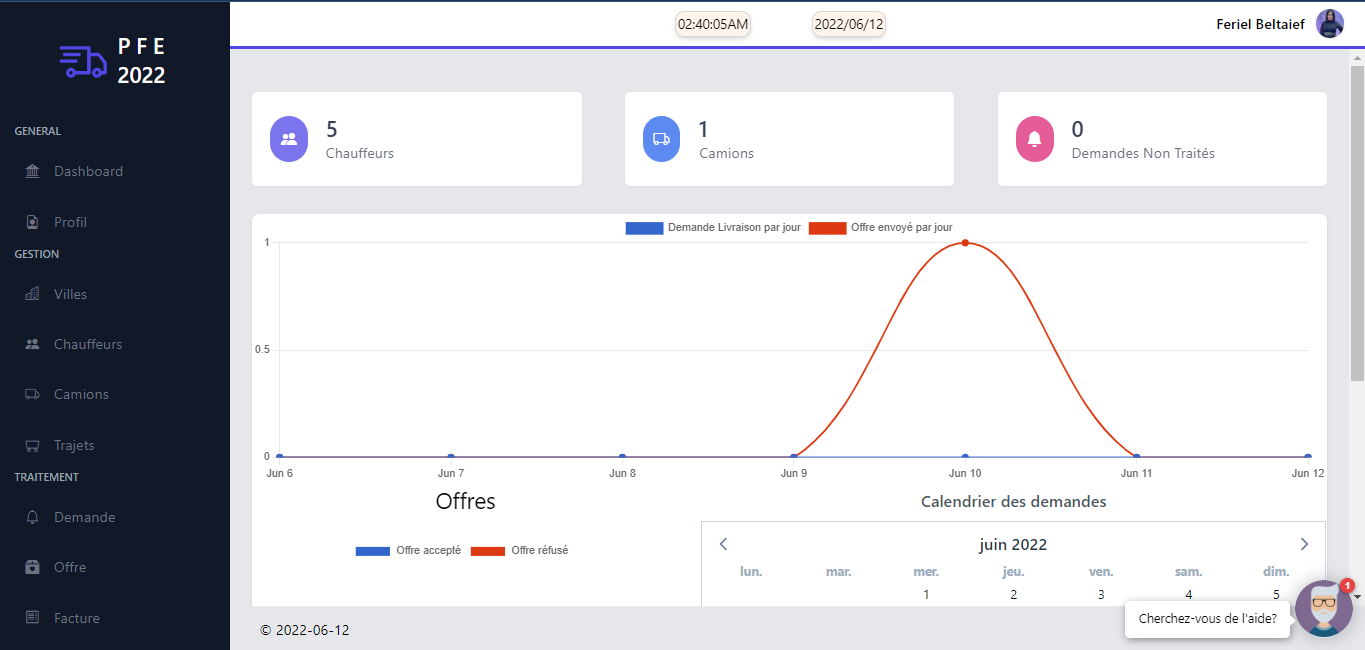


Figure 41 : Interface de tableau de bord

* La figure suivante illustre l’interface correspond au profil du client où le client peut modifier son nom et prénom, son email, son numéro et son mot de passe ainsi que son image :

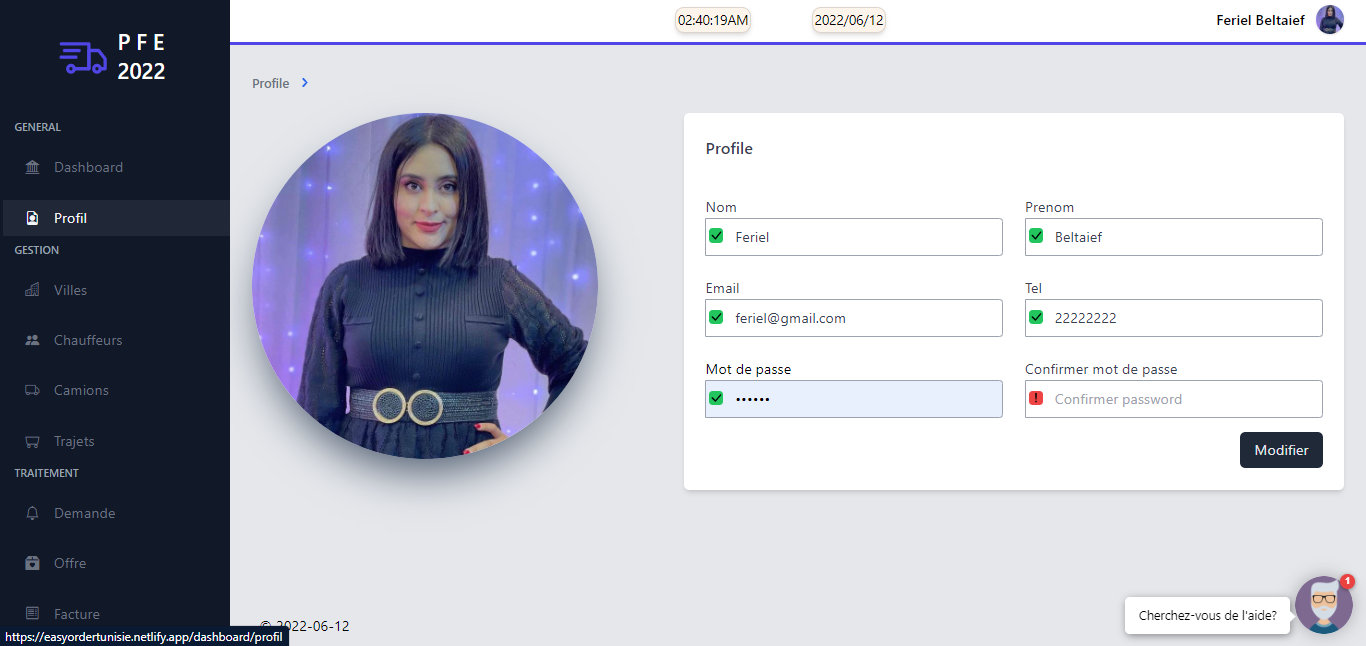


Figure 42 : Interface de profil

* Les figures suivantes illustrent l’interface correspond à la page où le transporteur peut sélectionner les villes de destinations:

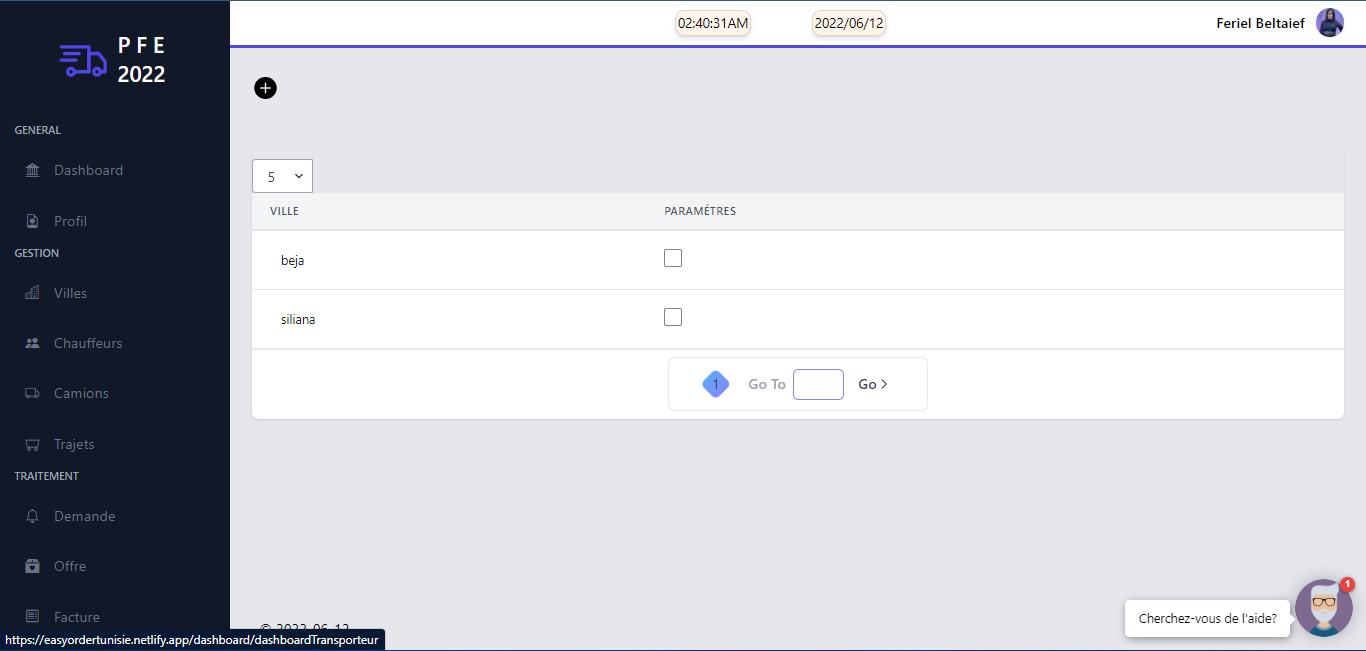


Figure 43 : Interface de consultation des villes de destinations

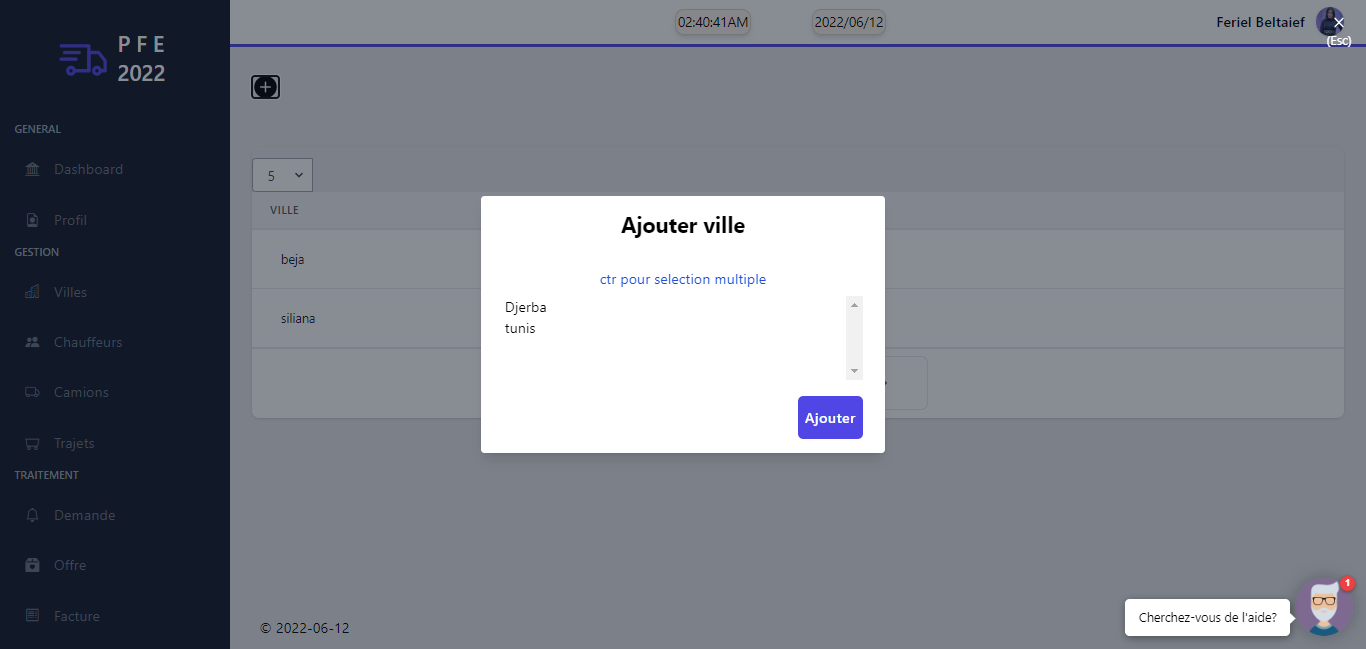


Figure 44 : Interface d’ajoute une ville

* Les figures suivantes illustrent l’interface correspond à la page de chauffeurs où le transporteur peut ajouter, modifier et supprimer un chauffeur

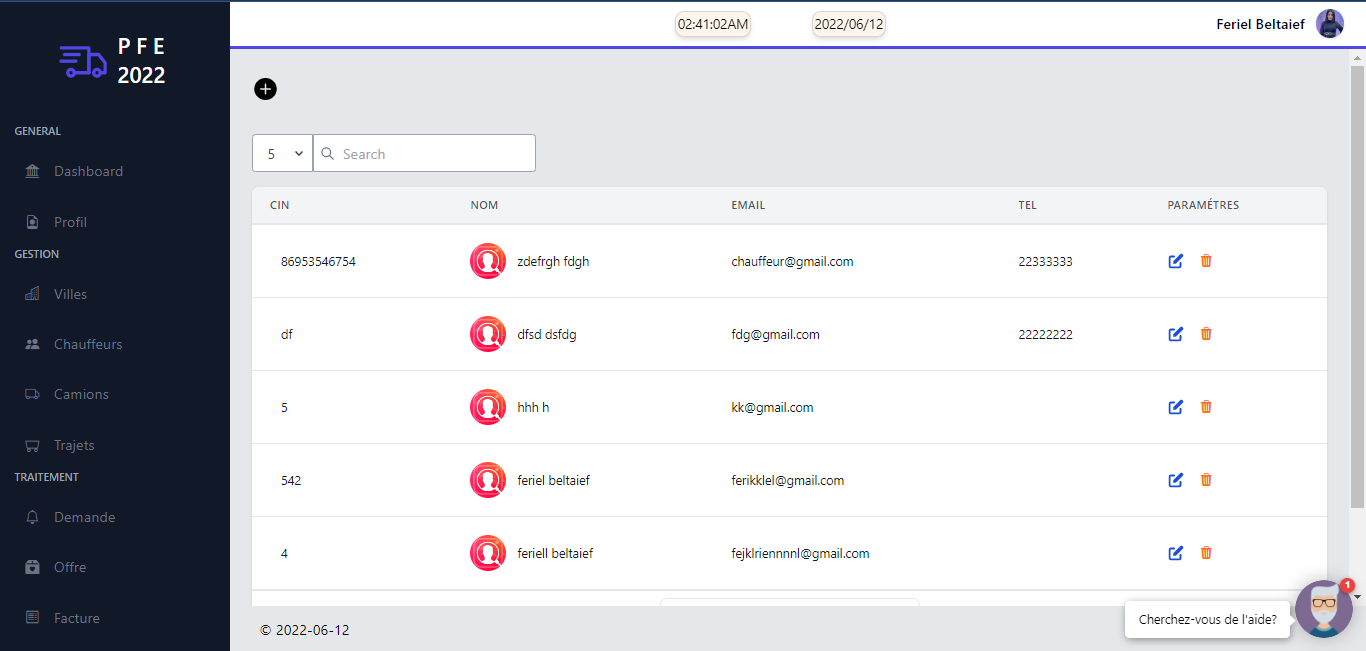


Figure 45 : Interface de consultation des chauffeurs

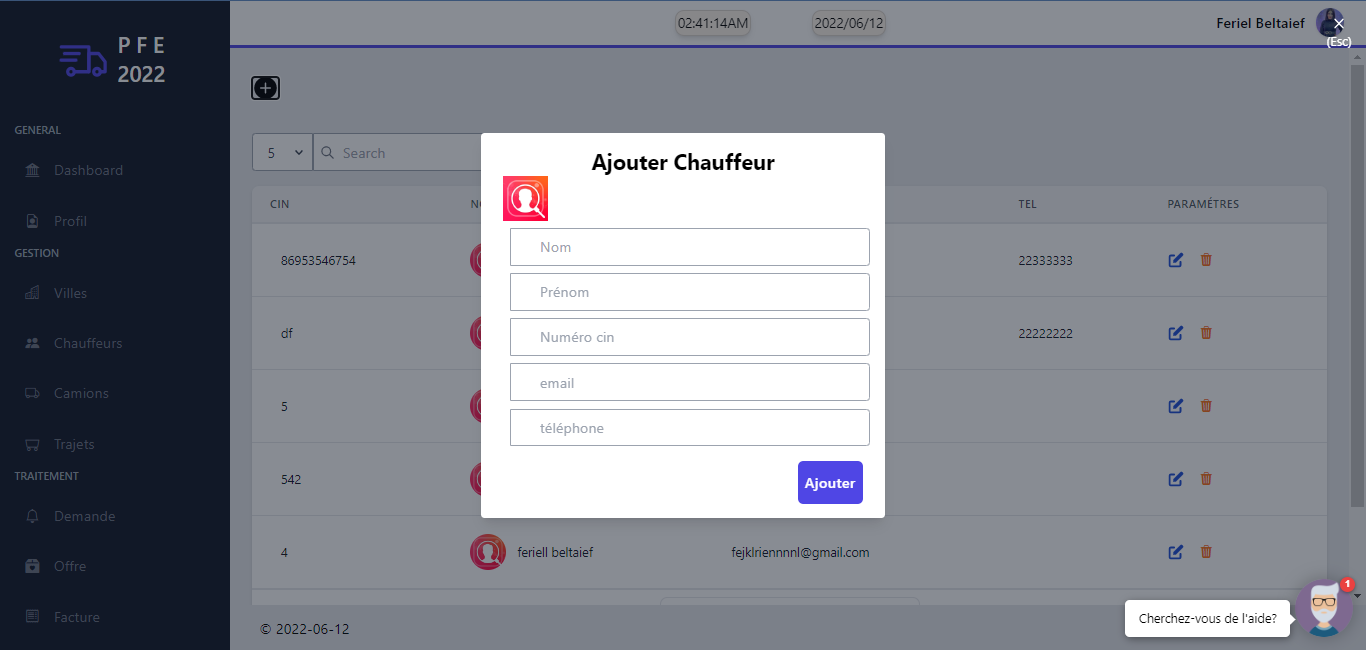


Figure 46 : Interface d’ajoute une chauffeur

* Les figures suivantes illustrent l’interface correspond à la page de camions où le transporteur peut ajouter, modifier et supprimer un camion et lui affecter un chauffeur et consulter ses trajets :

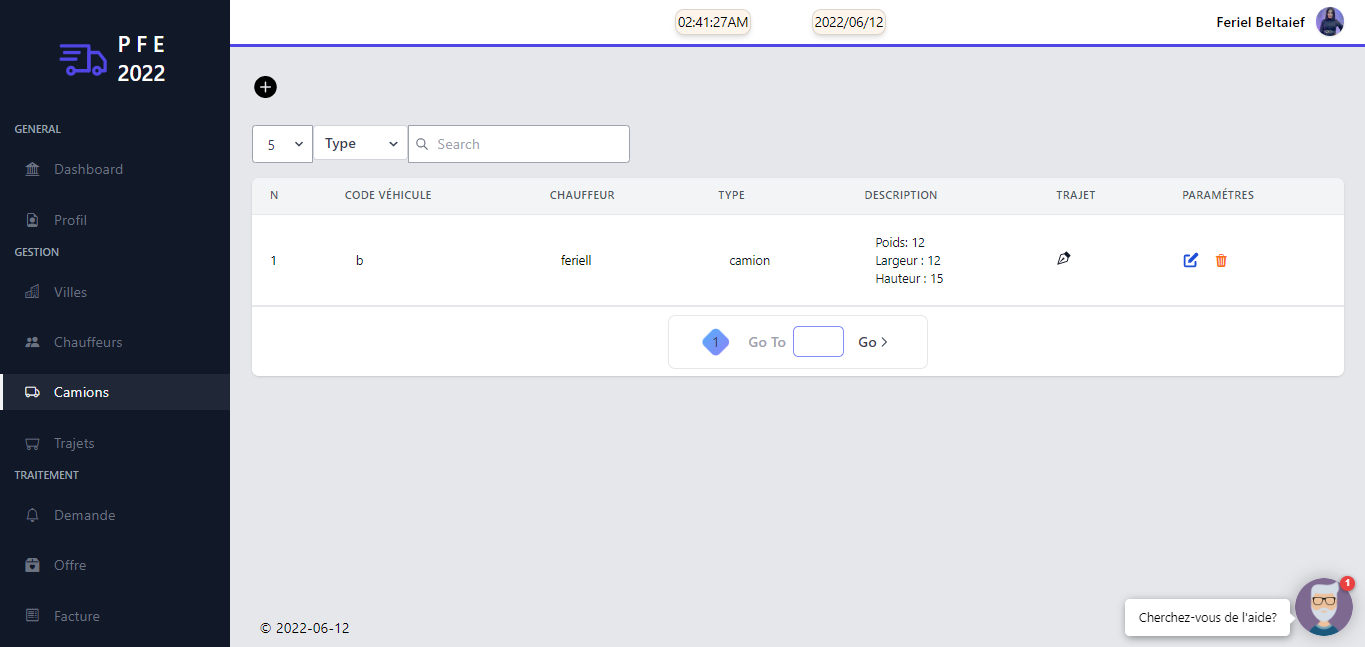


Figure 47 : Interface de gestion des camions

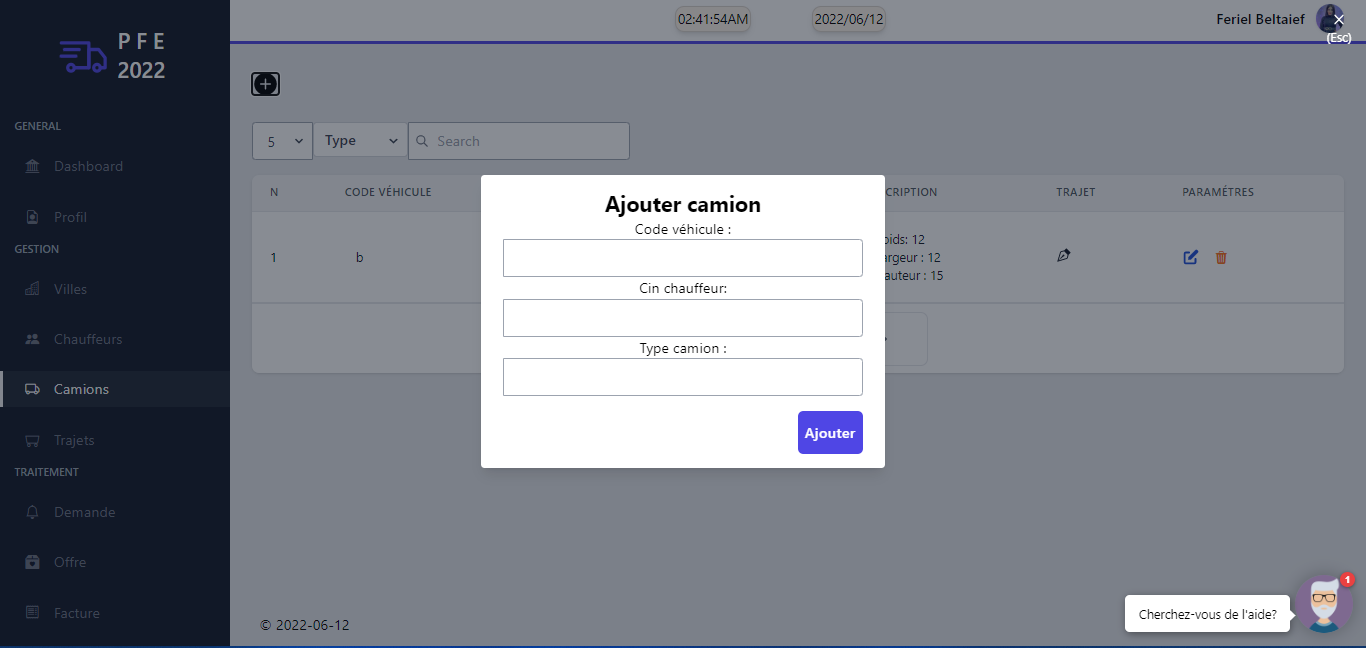


Figure 48 : Interface d'ajout de camion.

* La figure suivante illustre l’interface correspond à la page de trajets où le transporteur peut ajouter, supprimer et modifier un trajet et lui affecter un camion :

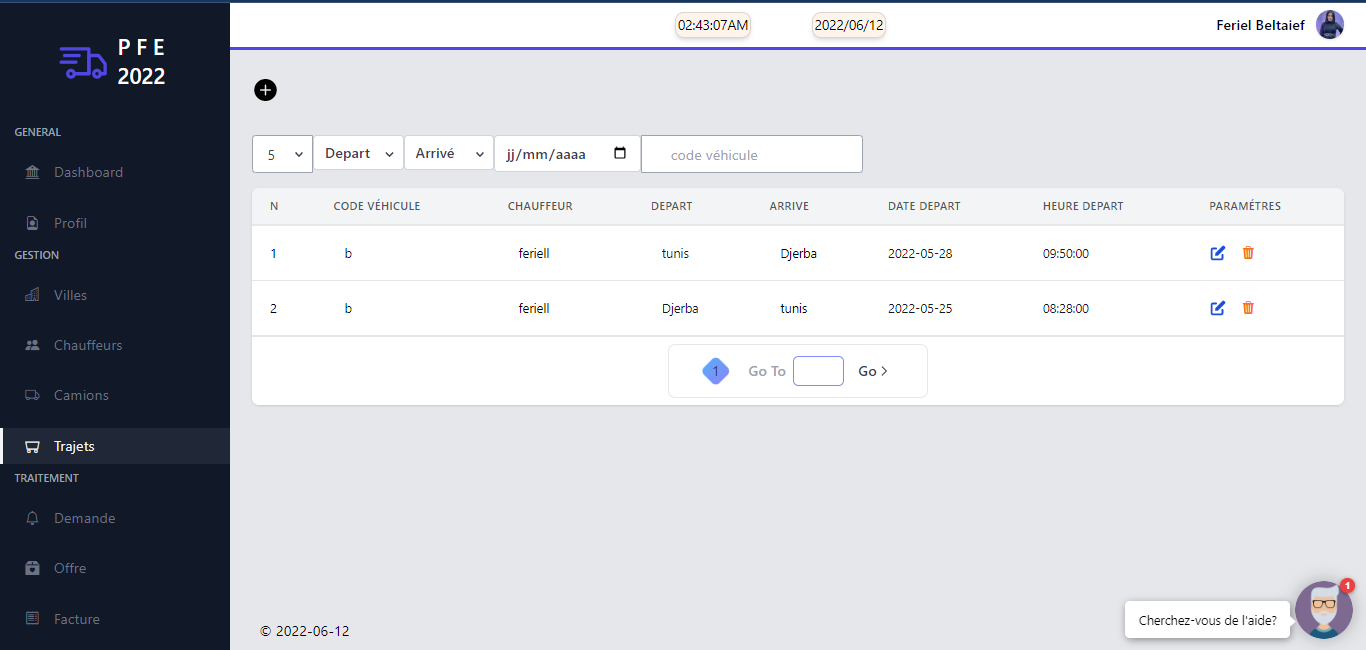


Figure 49 : Interface de gestion des trajets

* Les figures suivantes illustrent l’interface correspond à la page de demandes où le transporteur peut refuser ou accepter la demande en ajoutant une offre :

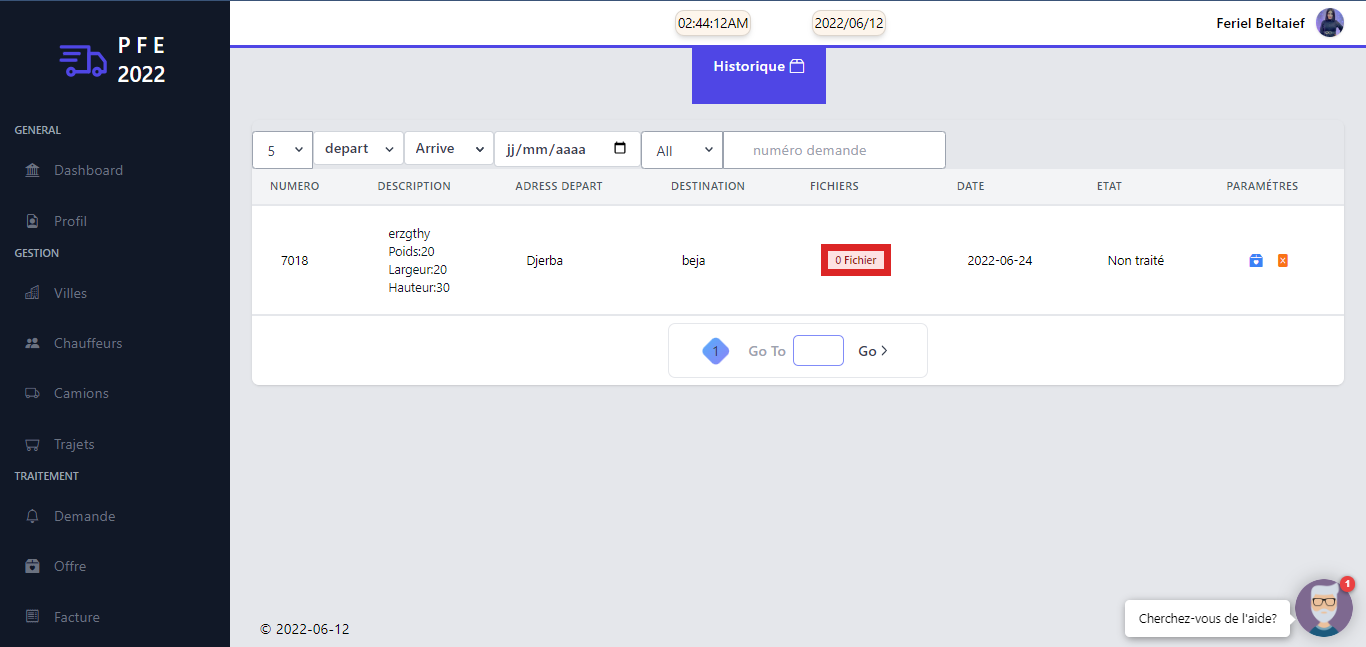


Figure 50 : Interface de gestion des demandes

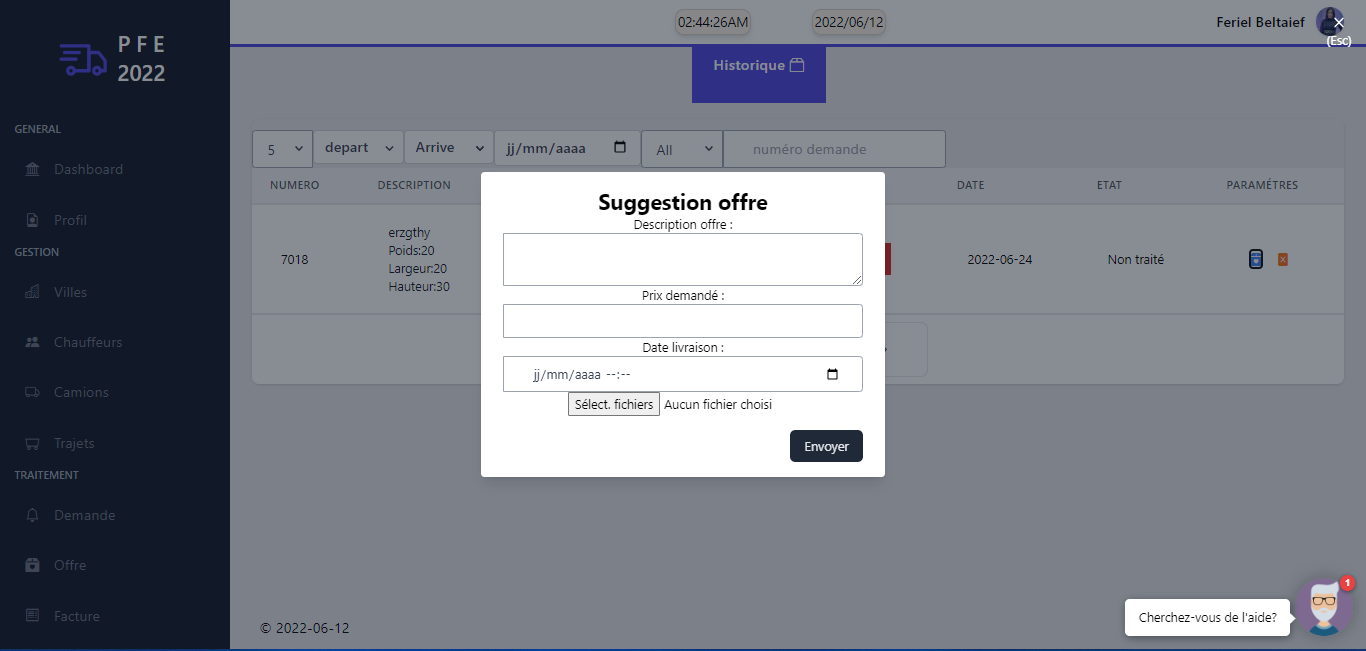


Figure 51 : Interface de suggestion d’offre

* La figure suivante illustre l’interface correspond à la page des offres où le transporteur peut consulter ses offres et modifier son état :

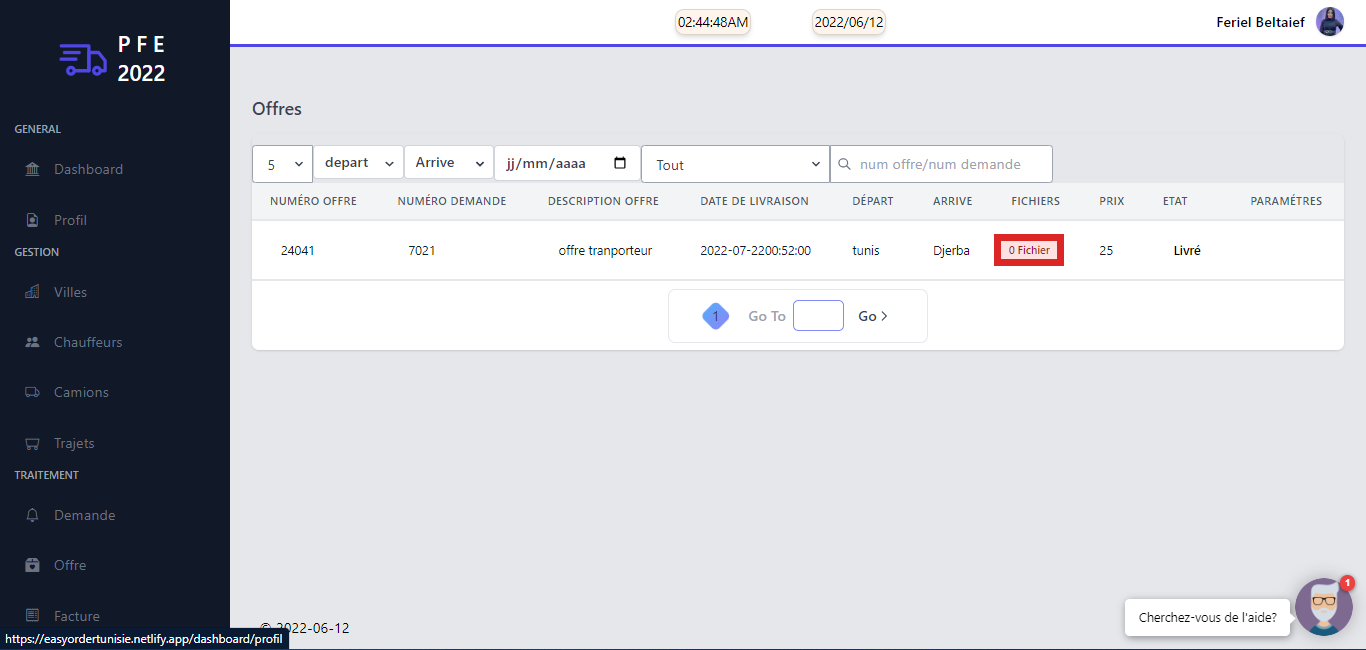


Figure 52 : Interface des offres

* La figure suivante illustre l’interface correspond à la page de factures où le transporteur peut consulter les reçus de paiement et ajouter les factures à payer :

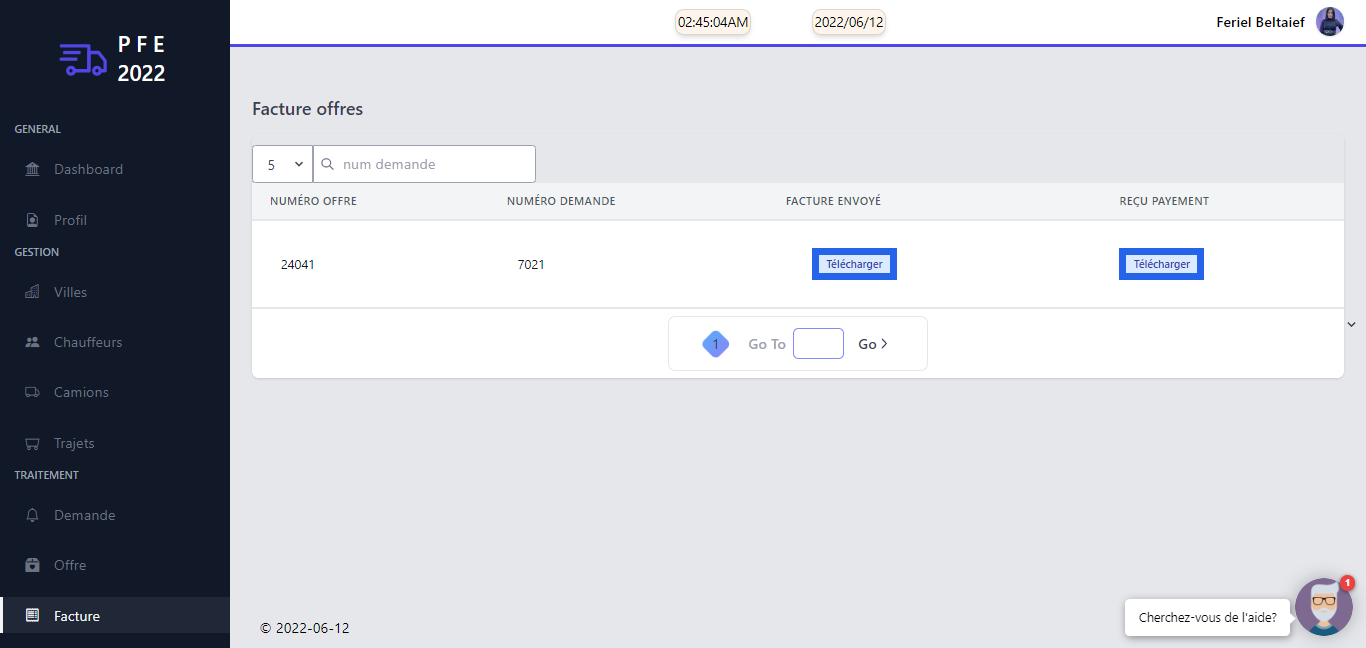


Figure 53 : Interface de gestion des factures

### 4.2 Application mobile

* Page d’authentification :

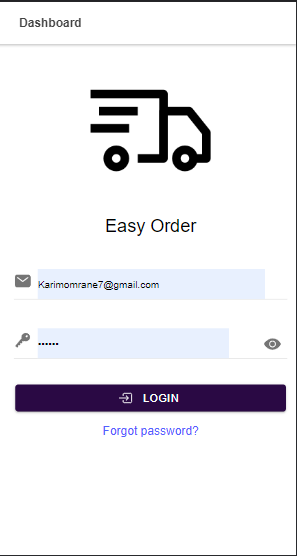


Figure 54 : Interface de login

* Les figures suivantes illustrent l’interface correspond au tableau de bord:

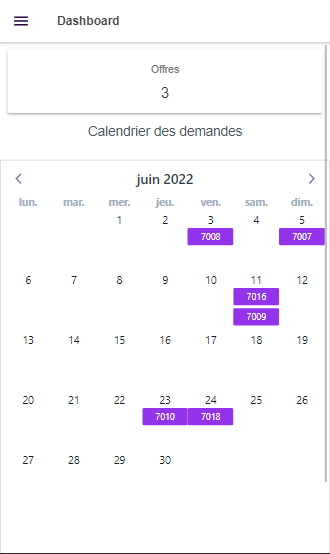
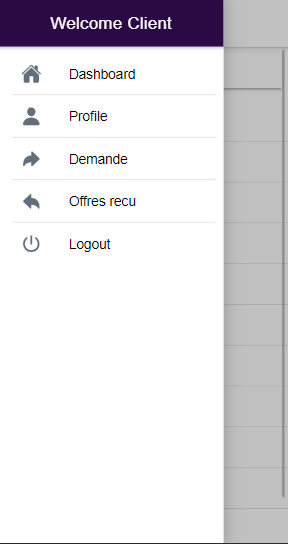


Figure 55 : Interface de tableau de bord avec le menu

* Les figures suivantes illustrent l’interface correspond à la page de profil :

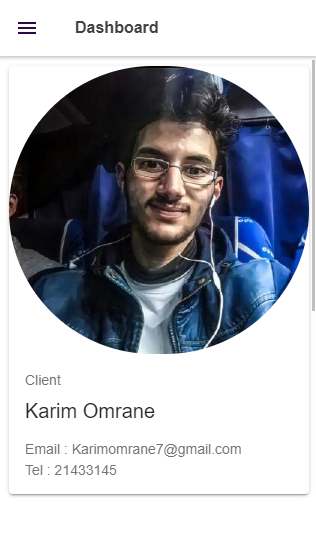


Figure 56 : Interface de profil

* Les figures suivantes illustrent l’interface correspond à la page des offres où le client peut consulter les offres :

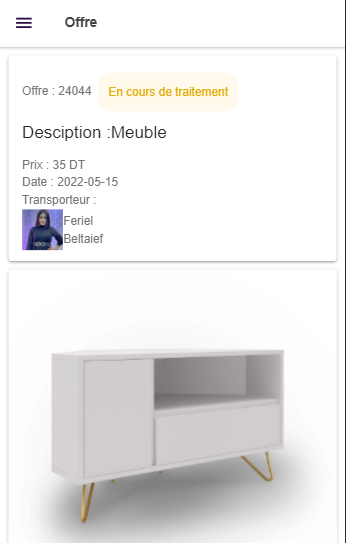


Figure 57 : Interface de consultation des offres

* Les figures suivantes illustrent l’interface correspond à la page ou le transporteur peut consulter les offres

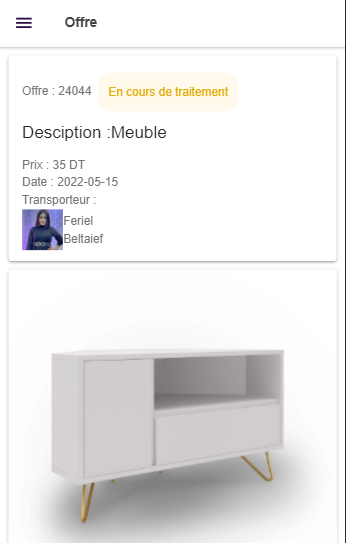
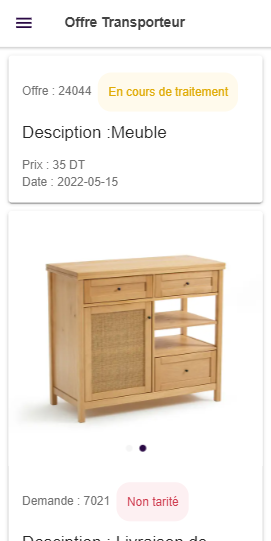


Figure 58 : Interface de consultation des offres

* Les figures suivantes illustrent l’interface correspond à la page ou le transporteur peut consulter les demandes

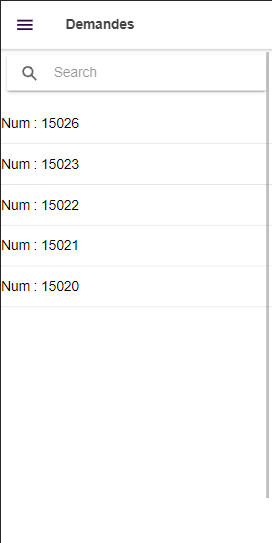
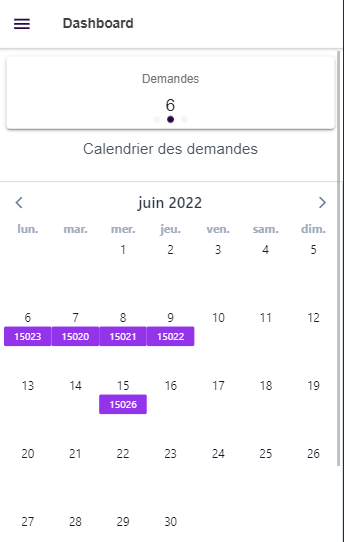
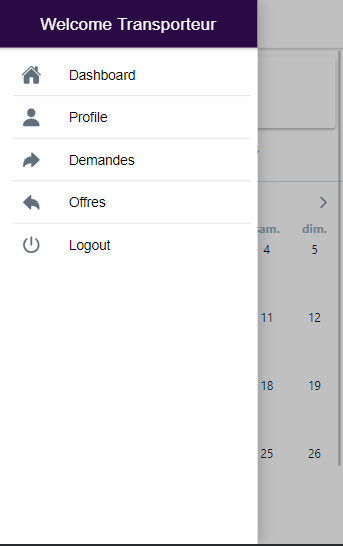




Figure 59 : Interface de consultation des demandes

## Conclusion

Dans ce chapitre, nous avons présenté les aspects de réalisation de notre système à travers une description de l’environnement de travail matériel et logiciel. Nous avons également présenté les principales interfaces de l’application.

Conclusion générale

L’objectif principal du projet est la conception et la réalisation d'un site web et application mobile pour la logistique et transport au sein de la société « Tunvita », afin d’achever notre premier cycle d'étude universitaire, en technologies de l’informatique à l’ISET de Djerba.

Afin de mener à bien le projet, la solution que nous proposons réside dans l'adoption d'une démarche qui consiste à suivre une méthode de travail. Nous avons commencé par la présentation générale du projet, spécifié les besoins des utilisateurs, procédé la conception et finalement nous avons recours à la réalisation.

Nous estimons que ce projet nous a été bénéfique sur tous les plans à savoir le plan technique et le plan humain.

Pour ce qui concerne le plan technique, le projet a été une véritable occasion pour découvrir et maitriser la technologie .net et la technologie Ionic pour le développement mobile.

Pour ce qui concerne le plan humain, nous avons eu la chance d'améliorer nos capacités de se communiquer et du travailler au sein d'une équipe.

Nous avons essayé dans notre travail de respecter les exigences présentées par le cahier de charges et nous estimons par ailleurs que notre application peut être enrichie par d'autres fonctionnalités qui sont les suivantes :

* Améliorer l'aspect sécuritaire en sécurisant d’avantage l’accès à la base de données.
* Ajout d'autres fonctionnalités :
* Une fonctionnalité de suivi GPS des colis en cours de livraison.
* Une migration future de la base donnée vers un système NOSQL pour éviter la saturation.

Bibliographie

[1] [Méthode Agile vs Méthode Classique, la Comparaison | Marine Guyot (marine guyot.ovh)](https://www.marine-guyot.ovh/methode-agile-ou-classique/) , [Consulter le 01/06/2022]

**[**2] <https://hubvisory.com/blog/qu-est-qu-un-backlog-comment-le-construire-et-le-gerer/>, , [Consulter le 03/06/2022]

[3] <https://sites.google.com/site/mortadhafadhlaoui3im1/conception>, [Consulter le 06/06/2022]

[4] <http://tvaira.free.fr/projets/activites/vscode.pdf> , [Consulter le 05/06/2022]

[5] <https://www.redhat.com/fr/topics/middleware/what-is-ide> , [Consulter le 05/06/2022]

[6] <https://fr.wikipedia.org/wiki/Lucidchart> , [Consulter le 05/06/2022]

[7] <https://en.wikipedia.org/wiki/SQL_Server_Management_Studio> ,[Consulter le 05/06/2022]

[8] <https://fr.wikipedia.org/wiki/GitHub> , [Consulter le 05/06/2022]

[9] <https://practicalprogramming.fr/postman/> ,[Consulter le 05/06/2022]

[10] <https://fr.wikipedia.org/wiki/Android_Studio> , [Consulter le 05/06/2022]

[11] <https://mobiskill.fr/blog/conseils-emploi-tech/vue-js-quels-sont-les-avantages-et-les-inconvenients/>, [Consulter le 06/06/2022]

[12] <https://www.duchess-france.org/presentation-de-ionic-framework/> , [Consulter le 06/06/2022]

[13] <https://qdmana.com/2021/12/20211201184921927w.html> ,[Consulter le 06/06/2022]

[14] <https://fr.wikipedia.org/wiki/Twilio>, [Consulter le 06/06/2022]

[15] <https://mailtrap.io/> , [Consulter le 06/06/2022]

[16] <https://docs.intelliticks.com/> ,[Consulter le 07/06/2022]

[17] <https://www.lemagit.fr/definition/API-RESTful> , [Consulter le 07/06/2022]

[18] <https://docs.microsoft.com/fr-fr/azure/virtual-machines/windows/overview> , [Consulter le 07/06/2022]

[19] <https://www.cachem.fr/ngrok-tunnel-applications-locales/> ,[Consulter le 07/06/2022]

[20] <https://fr.wikipedia.org/wiki/Netlify> ,[Consulter le 08/06/2022]

[21] <https://docs.microsoft.com/fr-fr/azure/azure-portal/> , [Consulter le 08/06/2022]