



Ministère de l'Enseignement Supérieur et Générale Des
Études Technologiques
Institut Supérieur des Études Technologiques de Bizerte



Département Technologies de
l'Informatique (TI)

Rapport de
PROJET DE FIN D'ETUDES

En vue de l'obtention de :
Licence Appliquée en développement des Systèmes d'Information

Migration d'Une Application Mobile d'Assurance

Elaboré par :

Mohamed Ali AYADI

Encadré par :

Mr Hamda MAGHROUM : Zedney Creative

Mr Bilel ZEMZEM : Iset Bizerte

Mr Aymen AMRAOUI : Iset Bizerte

Effectué à:

Zedney Creative Tunisie

Année universitaire : 2020/202

Dédicaces

Du profond de mon cœur, je dédie ce travail à tous ceux qui me sont chers,

A MES CHERS PARENTS

Que ce travail soit l'expression de ma reconnaissance pour vos sacrifices Consentis, Et le soutien moral et matériel que vous n'avez cessé de prodiguer.

Vous avez tout fait pour mon bonheur et ma réussite.

Que dieu vous préserve en bonne santé et vous accorde une longue vie.

A MA SŒUR

Vous étiez toujours présents pour m'aider et m'encourager.

Sachez que vous serez toujours dans mon cœur.

A tous mes amis qui n'ont cessé de m'encourager et de me soutenir.

A TOUS MES AMIS...

Remerciements

J'exprime ma gratitude à toutes les personnes qui m'ont aidé à accomplir ma tâche dans de bonnes conditions et qui ont su m'accorder toute l'attention pour élaborer le présent travail.

Je tiens à exprimer mes vifs remerciements à mes encadrants, Monsieur Hamda MAGHROUM, Monsieur Bilel ZEMZEM et Monsieur Aymen AMRAOUI pour leurs aides, leurs conseils précieux, leurs critiques constructives et leurs suggestions pertinentes qui ont été remarquables tout au long de mon stage.

Je voudrais témoigner par la suite ma reconnaissance à toutes les personnes qui m'ont également fait bénéficier de leurs conseils et leurs expériences.

Mes remerciements les plus distingués sont adressés ensuite aux membres du jury qui m'ont fait l'honneur de bien vouloir accepter d'évaluer ce travail.

Mes remerciements vont aussi à ma famille, à mes amis et à tous ceux qui ont aidé de près ou de loin à la réalisation de ce travail.

Finalement, j'exprime mes sincères reconnaissances à l'égard de tous ceux qui ont contribué à ma formation, particulièrement les enseignants de l'Institut Supérieur des Études Technologiques de Bizerte.

Table des matières

Introduction générale	10
Chapitre1 : Présentation du cadre du projet	12
Introduction	12
I. Présentation du cadre de travail.....	12
I.1. Présentation de l'entreprise d'accueil	12
I.2. Organigramme	13
II. Etude et critique de l'existant	13
III. Méthodologie adoptée	17
IV. Le Framework SCRUM.....	20
V. Backlog product du projet	25
Conclusion.....	27
Chapitre2: Etude préalable et Spécification des besoins	28
Introduction	28
I. Etude préalable	28
Introduction	28
II. Environnement du travail.....	32
III. Installation et configuration	37
IV. Etude des besoins	40
Conclusion.....	43
Chapitre 3 : Release 1	44
Introduction	33
I. Sprint 1	44
1. Sprint Goal.....	44
2. Sprint Back log.....	44
3. Analyse	45
3.1. Diagramme de cas d'utilisation	45
3.2. Description textuelle	45
4. Conception du sprint.....	46
4.1. Diagramme de classe	46
4.2. Diagramme de séquence	47
5. Réalisation et test d'incrimation	49
6. Sprint review	53
II. Sprint 2	53

1. Sprint Goal.....	53
2. Sprint Back log.....	54
3. Analyse	54
3.1. Digramme de cas d'utilisation	55
3.2. Description textuelle	55
4. Conception du sprint 1	57
4.1. Diagramme de classe	57
4.2. Digramme de séquence	57
5. Réalisation et test d'incrimination	59
6. Sprint review	62
 Chapitre 4 : Release 2	63
Introduction	51
I. Sprint 3	63
1. Sprint 3 Goal.....	63
2. Sprint Back log	63
3. Analyse	64
3.1. Digramme de cas d'utilisation	64
3.2. Description textuelle.....	64
4. Conception du sprint 3.....	66
4.1. Digramme de classe	67
4.2. Digramme de séquence	67
5. Réalisation et test d'incrimination.....	68
6. Sprint review	70
II. Sprint 4	70
1. Sprint 3 Goal.....	70
2. Sprint Back log.....	71
3. Analyse	71
3.1. Digramme de cas d'utilisation	71
3.2. Description textuelle	72
4. Conception du sprint 3	73
4.3. Digramme de classe	73
4.4. Digramme de séquence	73
5. Réalisation et test d'incrimination	74
6. Sprint review	75
Conclusion générale	76
Bibliographies	78

Liste des figures

Figure 1 : Organigramme de la société.....	4
Figure 2 : Page Dashboard mobile.....	5
Figure 3 : Page Consulte remboursement	5
Figure 4 : Page filtrer recherche de remboursement.....	6
Figure 5 : Les méthodes classiques VS les méthodes agiles	9
Figure 6 : Scrum Overview	11
Figure 7 : Interface web du Product Back log avec bitrix24.....	15
Figure 8 : Architecture logicielle.....	19
Figure 9 : Architecture MVVM.....	20
Figure 10 : Architecture MVVM vs MVC.....	21
Figure 11 : Détail de travail entre composent de MVVM.....	21
Figure 12 : Logo vscode	22
Figure 13 : Logo Android Studio	23
Figure 14 : logo Postman.....	23
Figure 15 : Logo Bitrix24.....	24
Figure 16 : Logo ionic.....	25
Figure 17 : Logo node.js.....	25
Figure 18 : Logo Python	26
Figure 19 : Logo Angular.....	26
Figure 20 : Installation Plug de l'ionic	27
Figure 21 : Installation Python.....	27
Figure 22: Installation bitrix24	28
Figure 23 : Configuration de API bitrix24	28
Figure 24 : Configuration de Django Framework	29
Figure 25 : Diagramme de contexte statique des acteurs du système	31
Figure 26 : Diagramme de cas d'utilisation général	32
Figure 27 : Digramme de cas d'utilisation pour sprint 1	34
Figure 28 : Digramme de class pour sprint 1.....	35
Figure 29 : Digramme de séquence pour dépôt de demande	36
Figure 30 : Digramme de séquence pour traite une demande	36

Figure 31 : Digramme de séquence générale avec bitrix24 pour oauth 2.0	37
Figure 32 : Interface Login pour Adhérent	38
Figure 33 : Interface dépôt d'une demande.....	38
Figure 34 : Insertion d'un document ou image dans demande	39
Figure 35 : Interface liste de demande.....	39
Figure 36 : Interface Login pour les employés ou L'administrateur.....	40
Figure 37 : Interface Dashboard pour l'administrateur	40
Figure 38 : Interface ajoute remboursement.....	41
Figure 39 : Digramme de cas d'utilisation pour sprint 2	43
Figure 40 : Digramme de class pour sprint 2	45
Figure 41 : Digramme de séquence éditer profile.....	45
Figure 42 : Digramme de séquence consulté remboursement.....	46
Figure 43 : Digramme de séquence éditer données bancaire.....	46
Figure 44 : Interface généré carte tire payent	47
Figure 45 : Interface consulté remboursement.....	47
Figure 46 : Interface édit données bancaire.....	48
Figure 47 : Interface chercher une professionnelle de santé.....	48
Figure 48 : Interface résulta de recherche.....	49
Figure 49 : Interface lien de connexion SSO	49
Figure 50 : Digramme de cas d'utilisation pour sprint 3	52
Figure 51 : Digramme de class pour sprint 3	55
Figure 52 : Digramme de séquence Consultation des remboursements	55
Figure 53 : Digramme de séquence Consultation de la liste des employés.....	56
Figure 54 : Digramme de séquence Consultation du Bénéficiaire du contrat	56
Figure 55 : Interface Demande d'inscription	57
Figure 56 : Interface Consulté du Bénéficiaire du contrat	57
Figure 57 : Interface Consulté les demandes	58
Figure 58 : Digramme de cas d'utilisation pour sprint 4	60
Figure 59 : Digramme de classe pour sprint 4	61
Figure 60 : Digramme de séquence géré temps de travail.....	62
Figure 61 : Interface Dashboard Employée	63
Figure 62 : Interface liste des employés.....	63
Figure 63 : Digramme de classe global	81
Figure 64 : Interface détail d'employée	82

Figure 65 : Interface Dashboard pour l'administrateur	82
Figure 66 : Interface consulter les remboursements pour l'administrateur.....	83
Figure 67 : Interface consulter les remboursements pour l'employée.....	83
Figure 68 : Interface consulter détail de remboursement	84
Figure 69 : Interface activation GPS	84

Liste des tableaux

Tableau 1 : Différence entre méthode Agile et méthode traditionnelle.....	9
Tableau 2 : Product Back log de l'application	16
Tableau 3 : Back log sprint 0	18
Tableau 4 : Matériel utiliser	22
Tableau 5 : Back log sprint 1	33
Tableau 6 : Description du cas d'utilisation « Dépôt d'une demande »	34
Tableau 7 : Description du cas d'utilisation « Consultions du demandes envoyées »	35
Tableau 8 : Description du cas d'utilisation « Traite les demandes »	35
Tableau 9 : Backlog sprint 2	42
Tableau 10 : Description du cas d'utilisation « Consulté de données bancaires ».....	43
Tableau 11 : Description du cas d'utilisation « Modifier données bancaires »	43
Tableau 12 : Description du cas d'utilisation « Chercher localisation un professionnel de santé	44
Tableau 13 : Description du cas d'utilisation « Suivi des remboursements »	44
Tableau 15 : Description du cas d'utilisation « Générer carte de tiers payant »	45
Tableau 16 : Back log sprint 3	52
Tableau 17 : Description du cas d'utilisation « Demande d'inscription »	53
Tableau 18 : Description du cas d'utilisation « Consulte état des demandes »	53
Tableau 19 : Description du cas d'utilisation « Consulte les remboursement »	53
Tableau 20 : Description du cas d'utilisation « Consulte liste des adhérent »	54
Tableau 21 : Description du cas d'utilisation « Consulte liste des employées »	54
Tableau 22 : Back log Sprint 4.....	54
Tableau 23 : Description du cas d'utilisation « Gère temps de travail »	59
Tableau 24 : Description du cas d'utilisation « Consulte statut connexion d'employée »	61
Tableau 25 : Description du cas d'utilisation «Temps d travaille par employée par jours » .	61

Introduction générale

Les secteurs d'activité financière comme notamment la banque, l'assurance a de nos jours largement franchi le cap de la transformation digitale en réorientant ses propres modèles économiques. Le changement a été lent et son adaptation laborieuse ; les avantages offerts par la digitalisation se sont révélés lentement.

Les investissements de l'ensemble du secteur financier dans les nouvelles technologies sont dorénavant considérables. Ils entraînent un profond changement dans la culture même de l'entreprise.

Grâce aux innovations qu'elle apporte, la quatrième révolution technologique offre de nouvelles méthodes de communication et de partage des données. Les produits, les processus, la gestion des données et la relation client ont été profondément bouleversés.

Face à cette mutation interne, les assureurs qui désirent s'imposer sont dans l'obligation de repenser leurs offres traditionnelles. Ils doivent impérativement développer et mettre à jour des solutions qui attirent les clients. Aujourd'hui, ces derniers maîtrisent non seulement les nouvelles technologies mais apprécient leur utilisation au quotidien.

Dans ce cadre (La digitalisation des processus d'assurance / Relation Client), nous avons eu la chance et l'opportunité de concevoir et développer une application mobile que la société Zedney Creative proposera à une large gamme de clients dans le domaine d'assurance.

Pour être à jour et en connaissance de toutes les transactions effectuées, l'assuré peut accéder à toutes ses informations et faciliter la communication avec la mutuelle d'assurance à partir d'une application mobile.

Ce rapport présente en détails tout le travail réalisé au cours de ce stage. Il comporte Six chapitres :

➤ Premier chapitre :

Présentation du cadre du projet dans lequel on donne une idée générale sur la Planification du projet.

➤ Deuxième chapitre :

Etude préalable et Spécification des besoins qui permet de décrit les besoins fonctionnels et non fonctionnels de l'application, le diagramme de cas d'utilisation général.

➤ Les deux derniers chapitres seront consacrés à décrire la conception et la réalisation des sprints.

J'achèverai mon rapport par une conclusion, qui établit le bilan du travail effectué, et ouvre de nouvelles perspectives de du travail réalisé.

Chapitre1 : Présentation du cadre du projet

Introduction

Ce travail s'inscrit dans le cadre de mon projet de fin d'étude dans le but de l'obtention du diplôme de « **Licence Appliquée en Technologie d'Informatique** » auprès de « **l'Institut Supérieur des Etudes Technologiques de Bizerte** ». Ce projet a été effectué au sein de la société « **Zedney Créative Tunisie** ».

Avant de se plonger dans les détails de ce travail, il s'avère indispensable de construire une idée sur l'organisme d'accueil et de savoir plus sur son origine et ses activités, ce qui fait l'objet de ce chapitre.

I. Présentation du cadre de travail

I.1. Présentation de l'entreprise d'accueil

Les travaux de ce projet ont été menés au sein de la société Zedney Créative, éditeur de solutions informatiques principalement pour le marché des assurances.

Zedney Créative est une société de services d'ingénierie informatique (SSII) fondée en 2011. Depuis sa création, elle s'est spécialisée dans le développement des services innovants.

Elle présente pour ces clients des solutions informatiques et les aide à digitaliser leurs communications en tirant le profit des outils collaboratifs. ZEDNEY CREATIVE est un Partenaire de Bitrix24 (Il sera décrit dans le chapitre2) qui est fier d'offrir une gamme de produits avec un support technique.

Ses principales taches c'est la réalisation des analyses des processus d'affaire, les digitaliser et les mettre en œuvre, ainsi que des services de formation.

Zedney Creative dispose de 3 centres de développement :

- En Europe : Paris ; FRANCE.
- Au Moyen-Orient : Riyadh ; L'ARABIE SAOUDITE

- En Afrique : Tunis et Bizerte ; TUNISIE

I.2. Organigramme

Après avoir présenté l'organisme d'accueil et ses principales activités, je vais présenter dans ce qui suit l'organigramme de l'entreprise et les différentes directions.

Figure 1 : Organigramme de la société



Je m'intéresse à la filiale qui présente l'entreprise d'accueil de mon stage (En Tunisie).

La figure 01 montre la hiérarchie au sein de filiale fondée en Tunisie. Il existe 3 directions (informatique, financière et commerciale) dirigées par une direction générale.

Mon stage se déroule au sein de la direction informatique.

II. Etude et critique de l'existant

L'entreprise exploite actuellement une application pour la gestion des relations client pour les compagnies d'assurance. Dans cette partie je vais mettre le point sur les caractéristiques de cette application à savoir les points forts et les points faibles et je termine par proposer une nouvelle solution.

II.1. Description de l'existant

Actuellement Zedney Creative commercialise une application mobile pour une dizaine de clients. Cette application a été développée en 2016 utilisant des technologies basées sur Ionic 3.8 et Angular 5.0.

Nous allons présenter quelques interfaces de cette application afin de mieux comprendre les

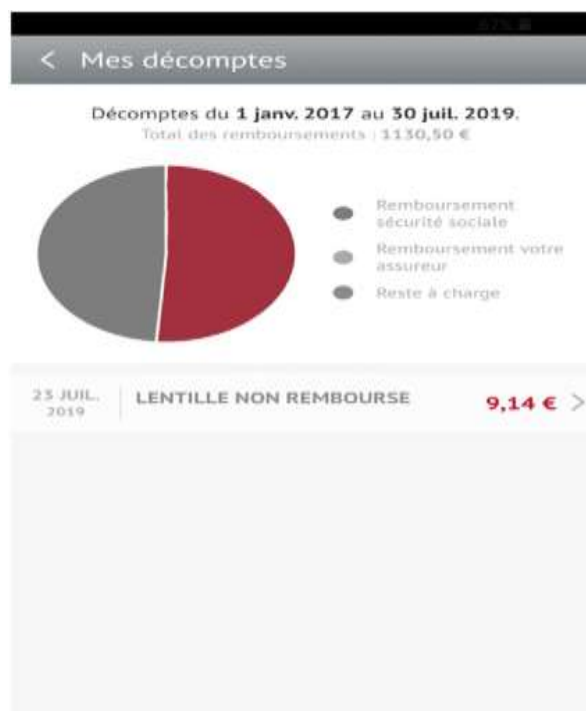
besoins à satisfaire et dégager les limites détectées.

Figure 2 : Page Dashboard mobile



Le figure 02 montre le menu principal de l'application qui peut, auprès des différentes fonctionnalités exposées, naviguer vers les différentes interfaces de l'application.

Figure 3 : Page Consulte remboursement



Le figure 03 montre l'interface où l'adhérent peut consulter et suivre ses remboursements

Figure 4 : page filtrer recherche de remboursement



X

Filtrer vos décomptes

Du 01/01/2019

Au 13/05/2021

RÉINITIALISER

VALIDER

Le figure 0x montre l'interface où l'adhérent peut filtrer ses recherches de remboursement

II.2. Critique de l'existant

Pendant les premières années de l'exploitation de l'application actuelle, les clients étaient satisfaits des services fournis par leurs assureurs à travers cette application qui répond parfaitement aux différents besoins fonctionnels demandés.

Néanmoins, l'évolution des outils de travail et de la technologie ont engendré des problèmes d'une dette technologique. Plusieurs erreurs se posent car les technologies utilisées deviennent obsolètes et plusieurs réclamations client se présentent en ce sens.

Zedney Creative était dans l'obligation de confronter ces obstacles d'où la décision était de

migrer cette solution vers de nouvelles technologies.

L'étude de la solution existante m'a permis de mettre l'accent sur les lacunes suivantes :

- Ionic 3.8.1 : La version utilisée dans l'existant présente de nombreux problèmes au niveau de la vulnérabilité, la manque de stabilité et l'impossibilité de l'utilisation des nouveaux plugins et la migration vers les versions récentes d'Angular
- Le problème avec la version actuelle utilisée d'angular est qu'elle est tellement dépassée, elle ne propose plus les dernières fonctionnalités. En plus, si j'utilise la dernière version de ionic, j'utiliserai automatiquement la dernière version d'Angular qui est bien prise en charge et qui est livrée avec l'application "lazy loading out of the box" (ce qui représente une augmentation importante de la vitesse de chargement de l'application, et la version actuelle de l'application "owlance" ne l'utilise pas).
En plus des nouvelles fonctionnalités d'Angular 8, telles que le nouveau compilateur rapide et la possibilité de récupérer des urls malformées.
- Vu que les versions IONIC et Angular sont très anciennes, il est impossible de mettre à jour les plugins utilisés dans l'application ce qui engendre pas mal de problèmes non gérables (par exemple : Pour pouvoir accéder à la fonctionnalité native de l'appareil comme la caméra, l'explorateur de fichiers, les bibliothèques ou les requêtes http en arrière-plan, la meilleure façon de le faire sur ionic est de passer par @ionic-native, qui est un wrapper pour certains plugins de cordova). Pour l'instant, je ne peux utiliser que la version 4.12 des wrappers @ionicnative, qui à leur tour ne prend pas en charge les dernières versions des plugins Cordova. Par exemple, lorsque je veux faire migrer toutes les requêtes HTTP existantes pour qu'elles utilisent des requêtes d'arrière-plan natives et qu'elle utilise l'épinglage SSL, je n'ai pas pu le faire complètement car la cette version ne prenait pas en charge tous les types de requêtes comme les requêtes "multipart". *

A ne pas oublier de mentionner l'existence de nombreux plugins inutiles tels que firebase (qui a été ajouté pour pouvoir faire tourner la dernière version de firebase sur un ancien Ionic) ou le "cordova-Android-support-gradle-release", qui est également ajouté pour pouvoir faire tourner la dernière version de Cordova Android avec le reste des plugins. Et l'utilisation de ces plugins je limite à utiliser cordova-ios@4.5.5 uniquement parce que le dernier écrase l'application au moment de la

construction ne peut pas fonctionner sur les nouvelle version d'Ionic.

Cette étude de l'existant m'a amenée à décider de redévelopper toute la solution en utilisant des technologies plus récentes (Une proposition qui a été validée par Zedney Creative).

II.3. Solution proposée

Je propose de concevoir et développer une nouvelle solution qui permet de dépasser les lacunes (déjà décrites). Cette solution qui va améliorer les services offerts aux utilisateurs.

Pour ce faire, on propose de développer une application répondant aux services demandés et qui satisfait les contraintes techniques suivantes :

- L'utilisation d'une nouvelle architecture pour le développement (l'architecture MVVM qui sera décrite plus tard).
- Passage à la version IONIC 6.12.4
- Passage à la version Angular 8.0.0.
- L'implémentation du lazy loading et le changement du système de rooting existant.
- La mise à niveau des Plugins nécessaires et la suppression/le remplacement des autres.

III. Méthodologie adoptée

La méthodologie est une démarche organisée rationnellement pour aboutir à un résultat déterminé. Ainsi, la méthodologie correspond à l'ensemble des processus qui permettent d'organiser la conception et le développement du projet dans le but de parvenir au résultat le plus performant et ce dans un climat d'efficacité et de convivialité.

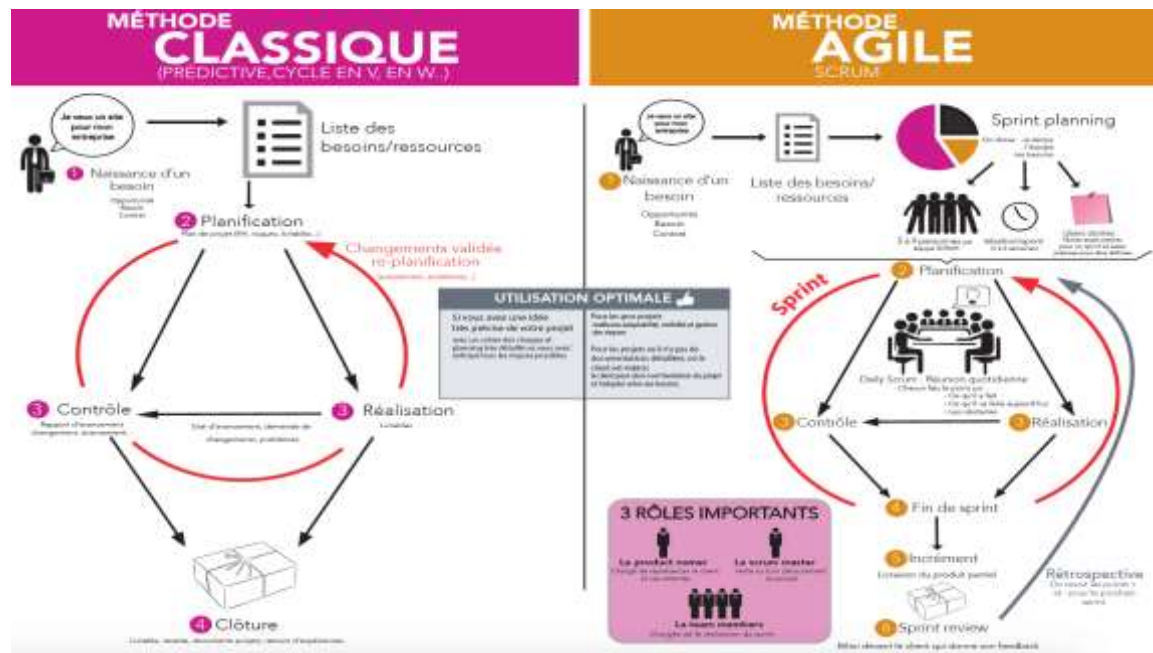
J'étais devant le choix entre l'application de l'une des méthodes de l'approche classique ou bien adopter l'un des méthodes de l'approche agile.

III.1. Approche classiques vs approches agiles

Pour pouvoir répondre au choix de la méthode de gestion de projet à adopter, j'étais

amené à faire une étude bibliographique sur les deux approches pour prendre la meilleure décision.

Figure 5: Les méthodes classiques VS les méthodes agiles [A]



La figure 5 présente la différence entre la méthode classique et la méthode agile

On privilégiera plutôt la méthode classique lorsqu'on a une idée précise du projet, avec un planning bien détaillé et où on a anticipé tous les risques possibles.

Quant à la méthode Agile, celle-ci permettant une meilleure adaptabilité, visibilité et gestion des risques. On privilégiera également la méthode Agile pour les projets où il n'y a pas de documents détaillés, ou quand vous sentez que votre client est indécis. Le client pourra alors voir l'évolution du projet et l'adapter à ses besoins sans pour autant vous obliger à recommencer tout le travail que vous avez fourni depuis le début.

Tableau 01 : Différence entre méthode Agile et méthode traditionnelle

Caractéristiques	Approche agile	Approche traditionnelle
------------------	----------------	-------------------------

Structure organisationnelle	Itératif	Linéaire
Échelle des projets	Petite et moyenne échelle	Grande échelle
Besoins des utilisateurs	Saisie interactive	Clairement défini avant la mise en œuvre
Implication des clients	Haute	Faible
Modèle de développement	Livraison évolutive	Cycle de la vie
Implication du client	Les clients sont impliqués dès l'exécution des travaux	Les clients s'impliquent tôt dans le projet mais pas une fois que l'exécution a commencé
Gestion des escalades	Lorsque des problèmes surviennent, toute l'équipe travaille ensemble pour les résoudre	Escalade aux gestionnaires en cas de problème
Préférence de modèle	Le modèle agile favorise l'adaptation	Le modèle traditionnel privilégie l'anticipation
Produit ou processus	Moins de concentration sur les processus formels et directifs	Plus sérieux sur les processus que sur le produit
Documents d'essai	Planification complète des tests	Les tests sont planifiés un sprint à la fois
Estimation de l'effort	Scrum master facilite et l'équipe fait l'estimation	Le chef de projet fournit des estimations et obtient l'approbation du PO pour l'ensemble du projet
Examens et approbations	Les revues sont faites après chaque itération	Examens et approbations excessifs par les dirigeants

La figure et le tableau précédents ont montré les avantages de l'approche agile face à l'approche classique.

Tenant compte de tout ce qui a été présenté, j'ai choisi d'appliquer une méthode agile pour la gestion et le suivi de mon projet.

III.2. La méthode choisie

Nous avons choisi d'adopter l'approche agile. Cette approche présente une approche itérative et incrémentale :

« Une méthode agile est une approche itérative et incrémentale, qui est menée dans un esprit collaboratif avec juste ce qu'il faut de formalisme. Elle génère un produit de haute qualité tout en prenant en compte l'évolution des besoins des clients » (Véronique Messenger Rota, Gestion de projet : Vers les méthodes agiles) (A)

Les principes de l'agilité (les méthodes agiles) :

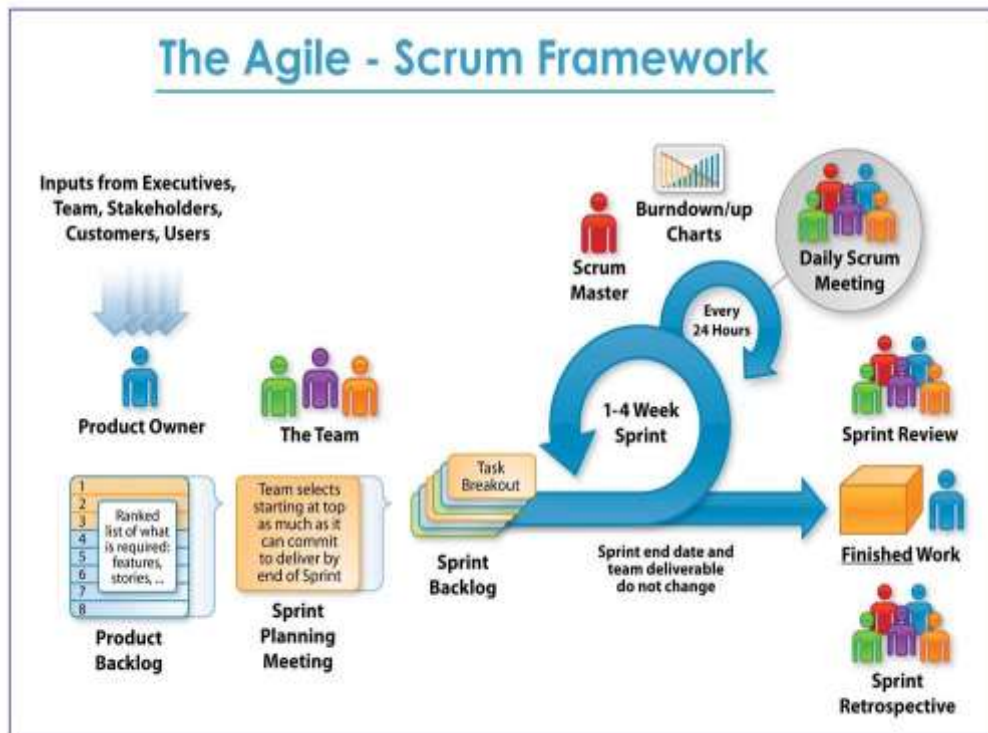
- ❖ La satisfaction client
- ❖ L'excellence technique
- ❖ Les individus
- ❖ L'amélioration continue

Il existe beaucoup de méthodes au niveau de cette approche à savoir scrum, Extreme Programming ... Zedney Creative a choisit d'utiliser la méthode SCRUM qui présente la méthode la plus utilisée dans les entreprises.

IV. Le Framework SCRUM

Scrum est « un cadre de travail permettant de répondre à des problèmes complexes et changeants, tout en livrant de manière productive et créative des produits de la plus grande valeur possible »

Figure 6: Scrum Overview [B]



Il se base sur la théorie du contrôle empirique de processus.

Trois piliers soutiennent l'implémentation du contrôle empirique de processus.

- La transparence
- L'inspection
- L'adaptation

IV.1. L'équipe Scrum

L'équipe SCRUM est une équipe composée de trois rôles, qui sont :

- **Scrum Master** : Il est le responsable de la bonne compréhension de SCRUM et de sa mise en œuvre. Le Scrum Master de notre projet est notre encadrant à l'ISSET de Bizerte qui j'aide à développer un côté d'auto-organisation avantageux et intéressant pour notre savoir être et savoir-faire. Notre Scrum Master j'encourage et je pousse à prendre les décisions qui me concerne par je-même. Il je facilite l'avancement dans le projet en essayant d'éliminer quelques obstacles rencontrés au cours de la réalisation. Il je responsabilise aussi envers le développement et je motive continuellement à l'accomplissement de l'application dans les délais et selon les règles et les normes SCRUM.
- **Product Owner** : Il est le responsable qui établit la vision produit et au même temps

c'est le représentant du client. C'est à lui qui collecte et négocie les exigences et les priorités du client.

Pour notre projet c'est notre encadrant à Zedney Creative qui maintient le Product Backlog. Il exprime clairement les cas d'utilisation, je donne la priorité de chacune et planifie aussi leurs réalisations.

Notre Product Owner s'assure toujours que les cas d'utilisation sont clairs et compréhensibles. Il est à l'écoute de nos interrogations, nos réclamations et nos propositions.

C'est à lui aussi de décider les livraisons des incréments développés tout en tenant en considération l'objectif de maximiser le Business Value de l'application.

- **Développement Team** : L'équipe de développement est responsable de livrer à chaque sprint un incrément « Done » et potentiellement livrable du produit. Je, membres de l'équipe de développement, je développons à travers ce projet une auto-organisation et des compétences variées.

J'ai appris le vrai sens de l'engagement et la discipline. Etre membre d'une équipe de développement SCRUM est une responsabilité à tenir.

IV.2. Les cérémonies

- **Sprint Planning** :

C'est un événement qui démarre chaque sprint et qui réunit la ScrumTeam pendant 4 heures vu que nos sprints s'étalent seulement sur deux semaines.

Au cours de cette cérémonie j'ai fixé les deux objectifs suivants :

- ✓ Définir le sprint goal
- ✓ Elaborer le Sprint Backlog

- **Daily Meeting** :

J'ai consacré 15 minutes chaque matin, pour faire le point sur :

- ✓ Ce que j'ai réalisé la veille pour atteindre l'objectif du sprint.
- ✓ Ce que je réaliserai le jour même pour atteindre l'objectif du sprint.
- ✓ Les obstacles qui me empêchent d'atteindre l'objectif du sprint.

- **Sprint Review :**

Cette cérémonie est tenue à la fin du sprint pour inspecter l'incrément résultant du sprint. C'est aussi une occasion pour adapter le Product Backlog si nécessaire. La Scrum Team participe à cet événement.

J'ai réalisé cet événement au cours de notre projet selon la disponibilité de notre Scrum Master, de notre Product Owner. La présence de ces trois profils de professionnels me permettait d'avoir un avis critique sur l'incrément réalisé.

- **Sprint Rétrospective :**

Cette cérémonie représente une opportunité pour la Scrum Team pour s'inspecter et créer un plan d'amélioration qui sera mis en place au cours du Sprint qui suit. Cette réunion aura lieu après le Sprint Review et avant le prochain Sprint planning.

Notre objectif à travers chaque Sprint Rétrospective est de :

- Revoir la manière dont le sprint s'est déroulé en ce qui concerne nos interactions, nos relations et la fluidité de passage de l'information au sein de toute l'équipe.
- Déterminer les points forts qui nous ont amené à réussir le sprint en cours et de les consolider dans les prochains sprints.

IV.3. Les artefacts

Les artefacts de Scrum sont en nombre de trois que je vais détailler dans les sections ci-dessous. En effet, ce sont des outils nécessaires pour l'accomplissement du travail demandé.

J'ai gardé toujours la transparence, l'inspection et l'adaptation lors de l'élaboration ou l'utilisation de ces artefacts.

- **Le Product Backlog**

C'est l'ensemble des cas d'utilisation exprimés par l'utilisateur final de notre application et rédigés et priorisés par le Product Owner. C'est l'une des manifestations agiles de SCRUM. En effet, c'est un artefact vivant et dynamique vu qu'il s'adapte convenablement aux changements de tout contexte : changement des besoins, de la technologie, du marché, etc. Comme équipe de développement, j'ai toujours recours au Product Backlog et à son rédacteur

le Product Owner.

- **Le Sprint Backlog**

Le Sprint Backlog set un sous ensemble du Product Backlog. Il contient seulement les cas d'utilisation qui permettent de réaliser l'objectif du sprint.

Pour l'élaboration de cet artéfact, j'ai détaillé les taches à effectuer pour chaque cas d'utilisation. J'ai tout droit sur le sprint Backlog car je somme les seuls responsables de son maintien et sa réalisation vu que c'est un document qui j'appartiens. Outre ces tâches, j'ai consacré une partie dans le sprint Backlog pour l'estimation de la charge de travail (le degré de difficulté, la priorité et la durée).

- **L'incrément**

« L'incrément est constitué des éléments du Product Backlog terminés pendant ainsi que la valeur cumulative des incréments livrés dans les sprints précédents. »

Les incréments que j'ai réalisés sont tous utilisables, potentiellement livrables et respectent nécessairement la définition de Done.

Dans le présent rapport je je sommes engagés de montrer dans chaque sprint l'incrément qui en résulte.

IV.4. La définition du Done

Afin d'améliorer la transparence dans l'équipe SCRUM et garantir une inspection efficace et par la suite une meilleure adaptation, je avons défini le Done de la manière suivante :

Un incrément est dit Done si et seulement si tous les efforts de conception, de développement et de test sont totalement terminés.

Je dois respecter cette définition tout au long du projet. C'est un auto-engagement que j'ai pris en considération durant tous les sprints.

Zedney utilise l'outil Bitrix24 qui présente un Framework conformément adapté à la méthode Scrum pour le suivi et la gestion de projet.

Figure 7 : Interface web du Product Back log avec bitrix24

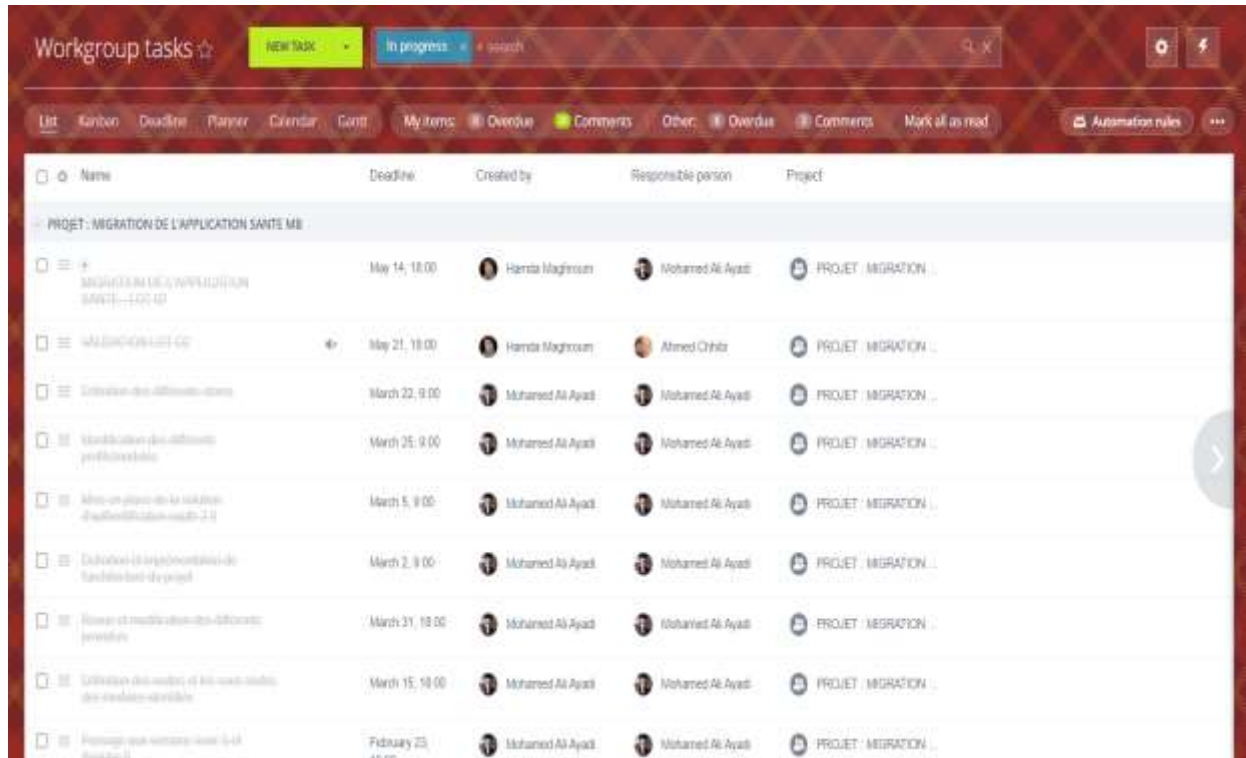


Figure 7 présente un Interface bitrix24 ou les sprints sont effectués et initialisés sous forme de taches

IV.5. Langage de conception

Pour faciliter la mise en place des Sprints et mieux documenter les incréments j'ai recours à l'utilisation du langage de modélisation UML.

UML est un langage de modélisation objet permettant de représenter n'importe quel système d'informations sous formes des modèles autour de trois axes : axe statique, axe dynamique et un axe fonctionnel.

Dans chaque sprint nous allons modéliser le système en présentant des diagrammes UML des différents axes précités. Cette modélisation va donner une vue complète et clarifie au mieux les objectifs à atteindre.

V. Backlog product du projet

L'étude de l'existant m'a permis de dégager les services et les besoins demandés par les clients.

Ces besoins ont été regroupés, triés selon la priorité et répartis sur les différents sprints. Le tableau ci-dessous présente ces besoins qui vont être l'objet de notre travail.

Tableau 2: Product Back log de l'application

N° Sprint	Nom User Story	Description user story	Durée (Jour)	Priorité
0		Architecture Logicielle et Frameworks à maîtriser	14	Élevé
1	Connexion à un compte « adhérent »	En tant qu'adhérent, je veux me connecter à mon compte	3	Elevé
	Dépôt d'une demande	En tant qu'adhérent, je veux déposer une demande de remboursement	4	Elevé
	Consultation des demandes	En tant qu'adhérent, je veux consulter les nouvelles demandes	3	Elevé
	Connexion à un compte « employé »	En tant qu'administrateur, je veux connecter à mon compte	2	Elevé
	Consultation des indicateurs et des statistiques	En tant qu'administrateur, je veux consulter des indicateurs statistiques (Dashboard)	5	Elevé
	Traitent des demandes	En tant qu'employé, je veux traiter la demande	4	Elevé
2	Modification de mes données	En tant qu'adhérent, je veux modifier mon profil	2	Moyenne
	Modification de mes données bancaires	En tant qu'adhérent, je veux modifier mes données bancaires	2	Moyenne
	Recherche et localisation des professionnels de santé	En tant qu'adhérent, je veux rechercher et localiser des professionnels de santé	5	Moyenne
	Génération des carte tiers-payant	En tant qu'adhérent, je veux générer ma carte de tiers-payant	6	Moyenne
	Connexion SSO à l'espace client	En tant qu'adhérent, je veux me connecter à mon espace client sans entrer mes identifiants une autre fois.	3	Moyenne
	Consultation remboursement	En tant qu'adhérent, je veux consulter mes remboursements	3	Moyenne
	Consultation et recherche des statistiques de mes différents remboursements	En tant qu'administrateur, je veux consulter et rechercher des indicateurs statistiques de mes remboursements	6	Faible

3	S'inscrire	En tant qu'un visiteur, je veux m'inscrire	3	Faible
	Consultation des états de mes demandes	En tant qu'administrateur, je veux consulter les états de mes demandes	3	Faible
	Consultation des bénéficiaires du contrat	En tant qu'adhérent, je veux consulter les bénéficiaires de mon contrat	3	Faible
	Consultation de la liste des adhérents	En tant qu'administrateur, je veux consulter les adhérents	3	Faible
	Consultation de la liste des employés	En tant qu'administrateur, je veux consulter et filtrer la liste des employés	3	Faible
4	Gestion du temps de travail	En tant qu'employé, je veux gérer mon temps du travail.	4	Faible
	Consultation des statuts de connexion des différents employés	En tant qu'administrateur, je veux consulter les statuts de connexion des différents employés	4	Faible
	Consultation du temps de travail des différents employés	En tant qu'administrateur, je veux consulter le temps de travail des différents employés par jours	3	Faible

Conclusion

Dans ce qui suit je présenterai le cadre général de l'élaboration de ce projet.

J'enchaînerai par la suite avec l'état de l'art qui sera l'objet du prochain chapitre.

Chapitre2: Etude préalable et Spécification des besoins

Introduction

Afin de mener à bien le processus de développement, il faut tout d'abord déterminer tous les besoins des différents utilisateurs du système à réaliser. Je vais donc me prolonger dans les technologies que je dois utiliser et le maîtriser de façon convenable.

I. Etude préalable

Introduction

Ce projet m'a mis devant un ensemble d'outils qui nécessitent une auto-formation et une architecture logicielle cible à maîtriser.

Dans ce chapitre, je vais présenter en un premier temps les préparatifs faits au niveau des outils de travail pour pouvoir entamer ce projet. En un deuxième temps, je vais collecter les besoins fonctionnels et les organiser dans un « backlog Product ».

I.1. Sprint goal

Le but de ce sprint est de comprendre l'architecture logicielle et la mise en œuvre de cette dernière à travers le Framework de développement, d'autre côté avoir une bonne idée sur Bitrix24 pour mieux manipuler le système et comprendre les interactions avec l'application.

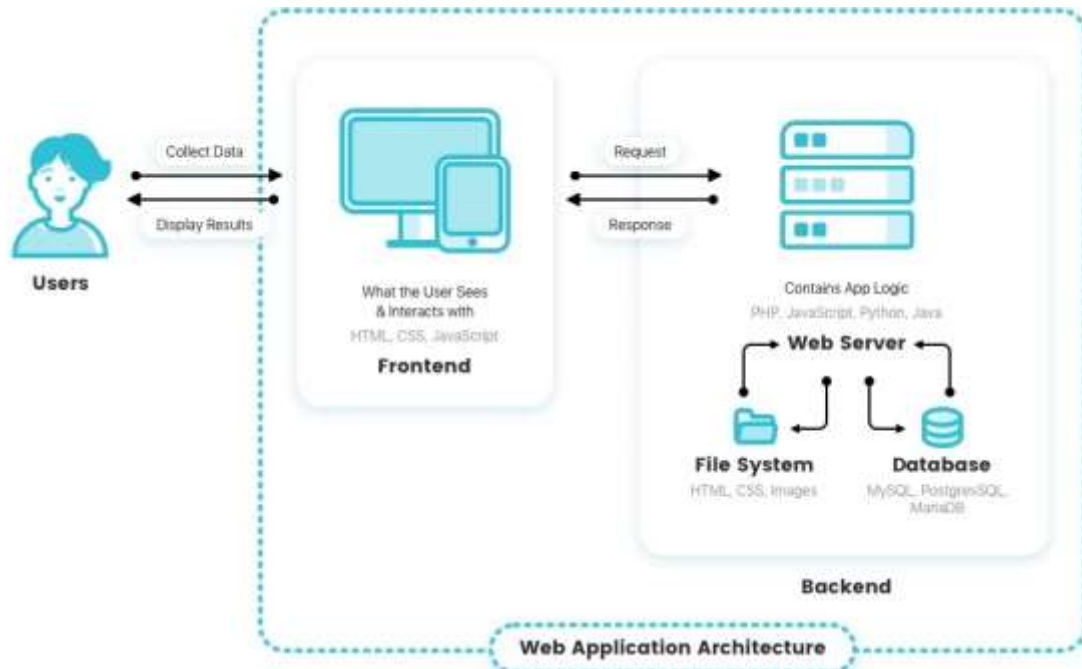
I.2. Sprint Back log

Tableau 03: Back log Sprint 0

0	Architecture Logicielle et Framework	1. Maîtriser la Platform bitrix24 2. Maîtriser les APIs de bitrix24 3. Comprendre les Architectures utilisées	14	Élevé
----------	--------------------------------------	---	----	-------

I.3. Architecture technique de l'application

Figure 8 : Architecture logicielle [C]



« Front end » et « back end » sont des termes utilisés par les programmeurs et les informaticiens pour décrire les couches qui composent le matériel, un programme informatique ou un site Web qui sont délimités en fonction de leur accessibilité à un utilisateur. Dans ce contexte, l'utilisateur fait référence à une entité qui peut être humaine ou numérique.

Le back-end fait référence aux parties d'une application informatique ou du code d'un programme qui lui permettent de fonctionner et qui ne sont pas accessibles par un utilisateur. La plupart des données et la syntaxe d'exploitation sont stockées et accessibles dans le back-end d'un système informatique. Typiquement, le code est composé d'un ou plusieurs langages de programmation. Le back-end est également appelé couche d'accès aux données du logiciel ou du matériel et comprend toute fonctionnalité qui doit être accessible et accessible par des moyens numériques.

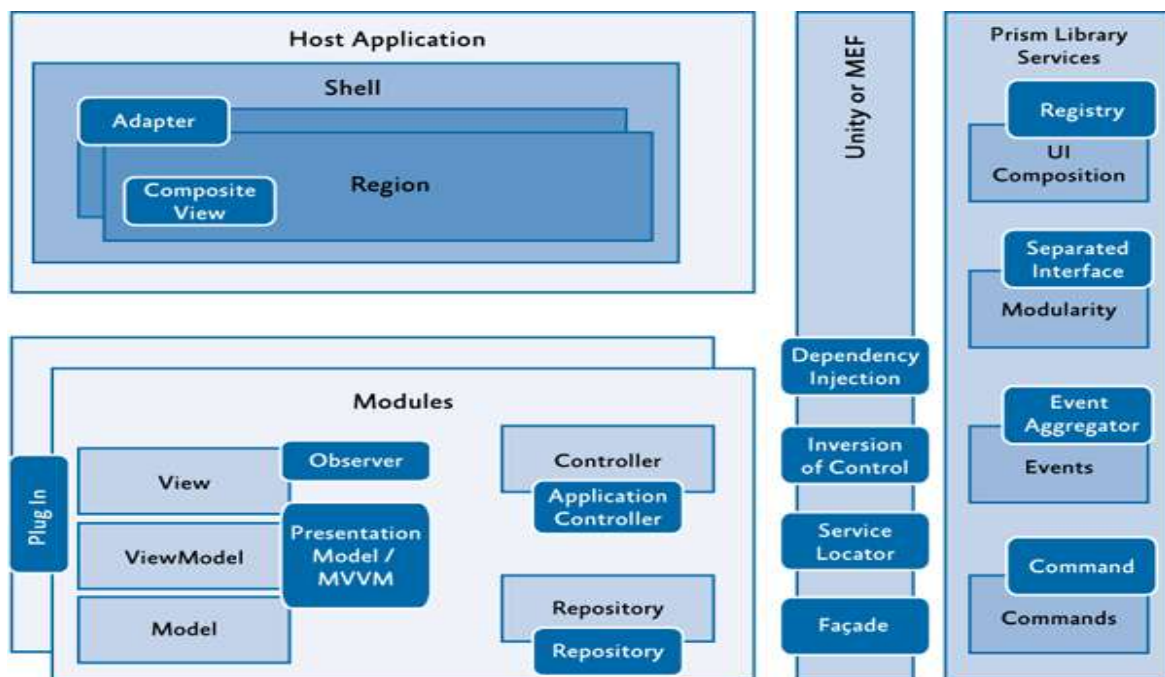
La couche au-dessus du back-end est le front-end et comprend tous les logiciels ou matériels faisant partie d'une interface utilisateur. Les utilisateurs humains ou numériques interagissent

directement avec divers aspects du front-end d'un programme, y compris les données saisies par l'utilisateur, les boutons, les programmes, les sites Web et d'autres fonctionnalités. La plupart de ces fonctionnalités sont conçues par des professionnels de l'expérience utilisateur (UX) pour être accessibles, agréables et faciles à utiliser.

Une application ou un programme back-end prend en charge les services utilisateurs frontaux et s'interface avec toutes les ressources requises. L'application back-end peut interagir directement avec le front-end ou elle peut être appelée à partir d'un programme intermédiaire qui assure la médiation des activités front-end et back-end.

I.4. Architecture MVVM

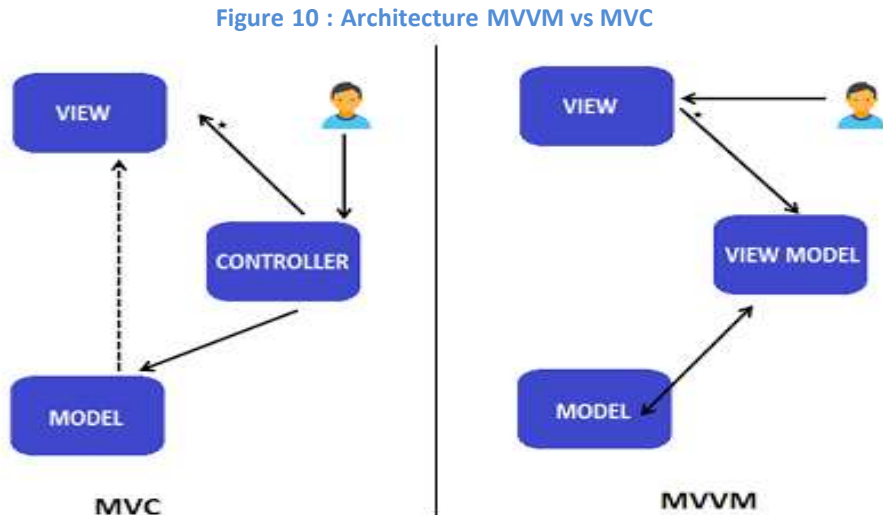
Figure 9 : Architecture MVVM [D]



Le modèle-vue-vue modèle (Model View ViewModel) est une architecture et une méthode de conception utilisée dans le génie logiciel.

MVVM est originaire de Microsoft et adapté pour le développement des applications basées sur les technologies Windows Présentation Fondation et Silverlight4 via l'outil MVVM Light5 par exemple. Cette méthode permet, tel le modèle MVC (modèle-vue-contrôleur), de séparer la vue de la logique et de l'accès aux données en accentuant les principes de liaison et

- Différence de mvvm que mvc :



Le framework MVC est un modèle architectural qui sépare une application en trois composants logiques principaux: modèle, vue et contrôleur. D'autre part MVVM facilite une séparation du développement de l'interface utilisateur graphique à l'aide d'un langage de balisage ou d'un code GUI.

Dans MVC, le contrôleur est le point d'entrée de l'application, tandis que dans MVVM, la vue est le point d'entrée de l'application.

Le composant MVC Model peut être testé séparément de l'utilisateur, tandis que MVVM est facile pour les tests unitaires séparés et le code est piloté par les événements.

L'architecture MVC a des relations « un à plusieurs » entre le contrôleur et la vue tandis que dans l'architecture MVVC, des relations « un à plusieurs » entre la vue et le modèle de vue.

Figure 11 : Détail de travail entre composant de MVVM [E]

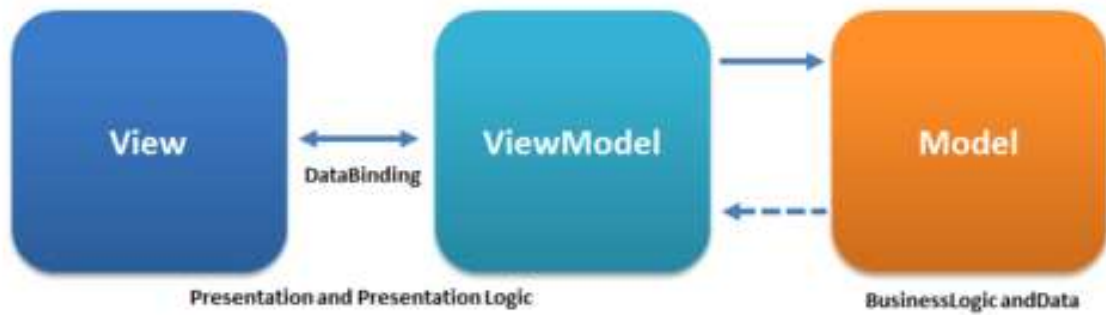


Figure 11 présente les fonctionnalités et la liaison entre les différents composants de l'architecture MVVM

II. Environnement du travail

II.1. Environnement matériel

Pour le développement de notre application, j'ai utilisé :

Tableau 04: Matériel utiliser

Modèle	Processeur	RAM	Disque Dur	Système d'exploitation
DELL G5 15	Intel I7	8GO	1To	Windows 10

II.2. Environnement logiciel

Concernant l'environnement logiciel de notre projet, j'utilisé parmi les outils et logiciels ce qui suit :

Figure 12 : Logo vscode [F]



Visual Studio Code est un éditeur de code extensible développé par Microsoft pour Windows, Linux et macOS2.

Les fonctionnalités incluent la prise en charge du débogage, la mise en évidence de la syntaxe,

la complétion intelligente du code, les snippets, la refactorisation du code et Git intégré. Les utilisateurs peuvent modifier le thème, les raccourcis clavier, les préférences et l'installateur des extensions qui ajoutent des fonctionnalités supplémentaires.

Le code source de Visual Studio Code provient du projet logiciel libre et open source VS Code de Microsoft publié sous la licence MIT permissive, mais les binaires compilés sont des logiciels gratuits pour toute utilisation.

Figure 13 : Logo Android Studio [G]



Android Studio est l'environnement de développement intégré (IDE) officiel pour le système d'exploitation Android de Google, basé sur le logiciel IntelliJ IDEA de JetBrains et conçu spécifiquement pour le développement Android. Il est disponible en téléchargement sur les systèmes d'exploitation Windows, macOS et Linux ou sous forme de service par abonnement en 2020. Il remplace les outils de développement Android Eclipse (E-ADT) en tant que principal IDE pour le développement d'applications Android natives.

Android Studio a été annoncé le 16 mai 2013 lors de la conférence Google I/O. Il était en phase de prévisualisation d'accès anticipé à partir de la version 0.1 en mai 2013, puis est entré en phase bêta à partir de la version 0.8 qui a été publiée en juin 2014. La première version stable a été publiée en décembre 2014, à partir de la version 1.0.

Le 7 mai 2019, Kotlin a remplacé Java comme langage préféré de Google pour le développement d'applications Android. Java est toujours pris en charge, tout comme C++.

Figure 14 : logo Postman [H]



Postman est la plateforme de collaboration pour le développement d'API il simplifie chaque étape de la création d'une API et rationalise la collaboration afin que vous puissiez créer de meilleures API, plus rapidement.

Figure 15 : Logo Bitrix24 [I]



Bitrix24 est un logiciel de collaboration avec des outils complets de gestion, de collaboration et de communication.

Bitrix24 vous offre une plate-forme unifiée pour vos fichiers, projets, messages, tâches et contacts. Vous pouvez immédiatement utiliser la plateforme en créant simplement un compte et en invitant ensuite vos collègues. Grâce au flux d'activités, tous les membres de votre équipe peuvent suivre les tâches et travailler efficacement ensemble pour faire avancer le projet.

Ses principales solutions incluent le réseau social, les tâches et les projets, le CRM, la gestion de documents, le partage de fichiers, le calendrier et la planification, la gestion des équipes de vente, la messagerie électronique, le téléphone et la gestion des ressources humaines. Il dispose également d'une solution mobile qui fonctionne sur votre smartphone ou votre tablette, qu'il s'agisse d'Android, d'iPhone ou d'iPad.

Les utilisateurs peuvent commencer avec un plan gratuit avec tous les outils et fonctionnalités essentiels : jusqu'à 12 utilisateurs, stockage en ligne et outils de collaboration. Les plans premium hébergés dans le cloud sont hautement évolutifs pour être accessibles sur n'importe quel appareil. Une solution sur site est également disponible pour un contrôle complet des données, un accès au code source et une gamme d'outils supplémentaires. Le marché des applications Bitrix est disponible pour créer vos propres applications ou utiliser vos systèmes existants

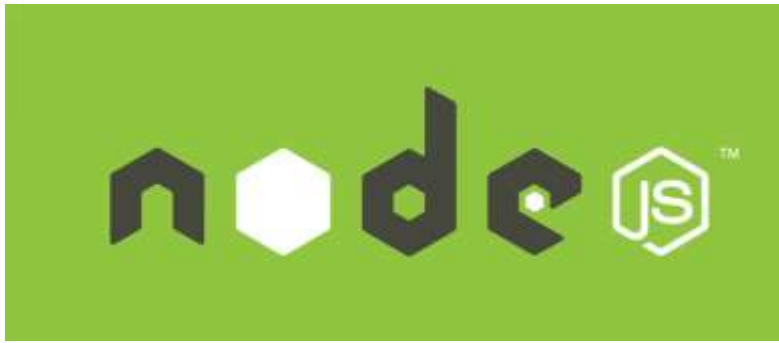
II.3. Langages de développement :

Figure 16 : Logo ionic [J]



Ionic est un SDK open source complet pour le développement d'applications mobiles hybrides créé par Max Lynch, Ben Sperry et Adam Bradley de Drifty Co. en 2013. La version originale a été publiée en 2013 et construite sur AngularJS et Apache Cordova. Cependant, la dernière version a été reconstruite en tant qu'ensemble de composants Web, permettant à l'utilisateur de choisir n'importe quel framework d'interface utilisateur, tel que Angular, React ou Vue.js. Il permet également l'utilisation de composants ioniques sans aucun cadre d'interface utilisateur. Ionic fournit des outils et des services pour développer des applications Web hybrides mobiles, de bureau et progressives basées sur des technologies et des pratiques de développement Web modernes, à l'aide de technologies Web telles que CSS, HTML5 et Sass. En particulier, les applications mobiles peuvent être créées avec ces technologies Web, puis distribuées via des magasins d'applications natifs pour être installées sur des appareils à l'aide de Cordova ou de Capacitor.

Figure 17 : Logo node.js [K]



Node.js est une plateforme logicielle libre en JavaScript, orientée vers les applications réseau événementielles hautement concurrentes qui doivent pouvoir monter en charge.

Elle utilise la machine virtuelle V8, la librairie libuv pour sa boucle d'évènements, et implémente sous licence MIT les spécifications Common JS.

Parmi les modules natifs de Node.js, on retrouve http qui permet le développement de serveur HTTP. Il est donc possible de se passer de serveurs web tels que Nginx ou Apache lors du déploiement de sites et d'applications web développés avec Node.js.

Concrètement, Node.js est un environnement bas niveau permettant l'exécution de JavaScript côté serveur.

Figure 18 : Logo Python [L]



Python est un langage de programmation interprété, multi-paradigme et multiplateformes. Il favorise la programmation impérative structurée, fonctionnelle et orientée objet. Il est doté d'un typage dynamique fort, d'une gestion automatique de la mémoire par ramasse-miettes et

D'un système de gestion d'exceptions ; il est ainsi similaire à Perl, Ruby, Scheme, Smalltalk et Tcl.

Le langage Python est placé sous une licence libre proche de la licence BSD et fonctionne sur

la plupart des plates-formes informatiques, des smartphones aux ordinateurs centraux, de Windows à Unix avec notamment GNU/Linux en passant par macOS, ou encore Android, iOS, et peut aussi être traduit en Java ou .NET. Il est conçu pour optimiser la productivité des programmeurs en offrant des outils de haut niveau et une syntaxe simple à utiliser.

Il est également apprécié par certains pédagogues qui y trouvent un langage où la syntaxe, clairement séparée des mécanismes de bas niveau, permet une initiation aisée aux concepts de base de la programmation.

Figure 19 : Logo angular [M]

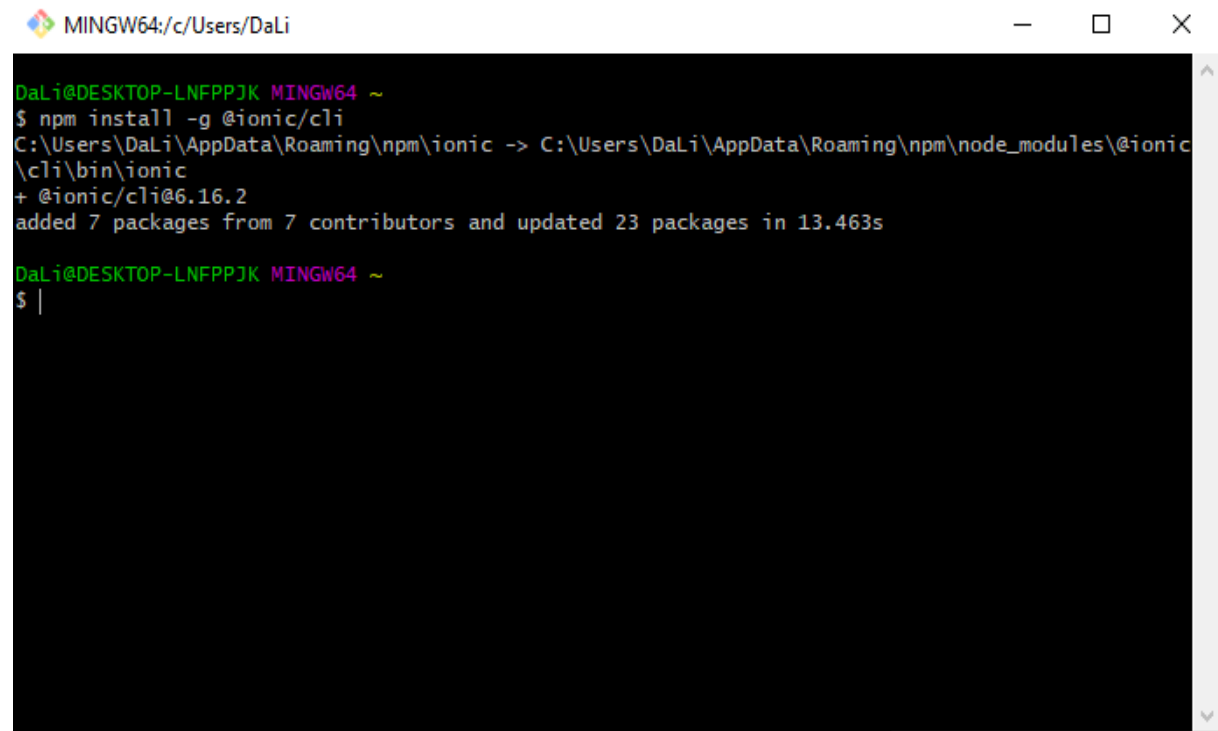


Angular est un cadre côté client, source, basé sur TypeScript, et co-dirigé par l'équipe du projet « Angular » à Google et par une communauté de particuliers et de sociétés. Angular est une réécriture complète de AngularJS, cadre construit par la même équipe. Il permet la création d'applications Web et plus particulièrement de ce qu'on appelle des « Single Page Applications » : des applications web accessibles via une page web unique qui permet de fluidifier l'expérience utilisateur et d'éviter les chargements de pages à chaque nouvelle action. Le Framework est basé sur une architecture du type MVC et permet donc de séparer les données, le visuel et les actions pour une meilleure gestion des responsabilités. Un type d'architecture qui a largement fait ses preuves et qui permet une forte maintenabilité et une amélioration du travail collaboratif.

III. Installation et configuration

❓ Installation Ionic 5 angular 9

Figure 20: Installation Plug de l'ionic

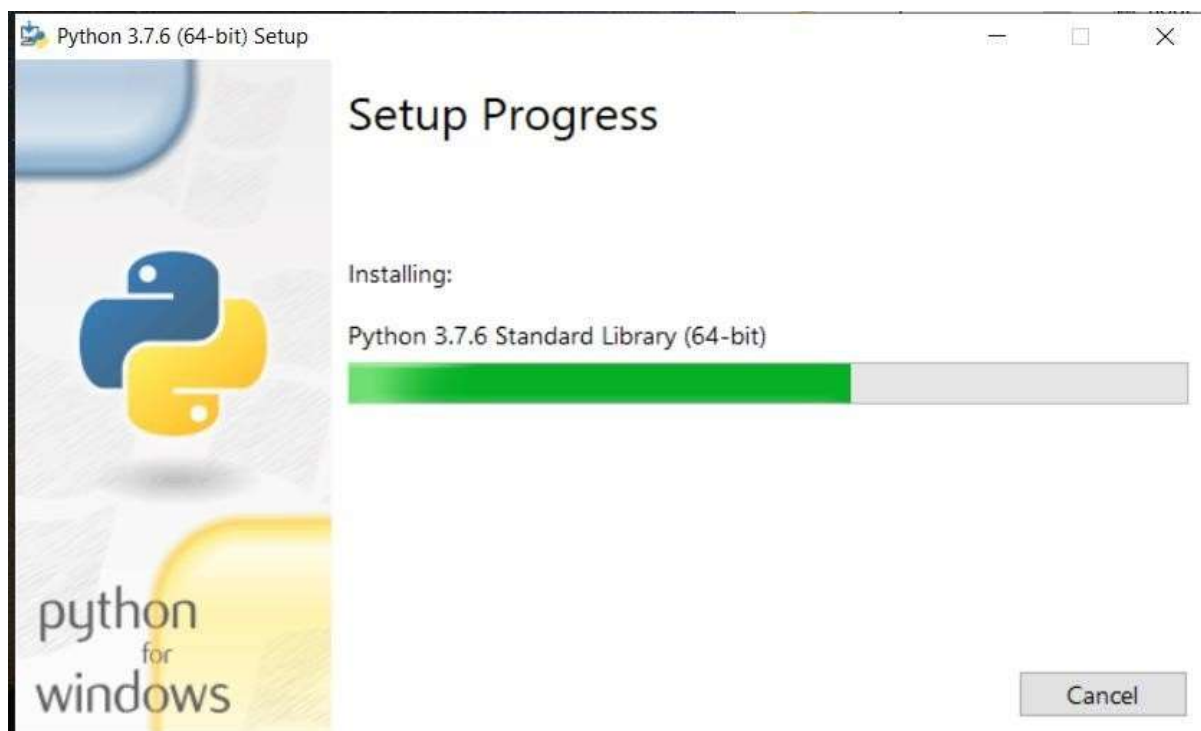


A terminal window titled 'MINGW64:/c/Users/DaLi' showing the execution of the command 'npm install -g @ionic/cli'. The output indicates that the command was successful, adding 7 packages from 7 contributors and updating 23 packages in 13.463 seconds. The terminal prompt is now '\$ |'.

```
DaLi@DESKTOP-LNFPPJK MINGW64 ~  
$ npm install -g @ionic/cli  
C:\Users\DaLi\AppData\Roaming\npm\ionic -> C:\Users\DaLi\AppData\Roaming\npm\node_modules\@ionic  
\cli\bin\ionic  
+ @ionic/cli@6.16.2  
added 7 packages from 7 contributors and updated 23 packages in 13.463s  
  
DaLi@DESKTOP-LNFPPJK MINGW64 ~  
$ |
```

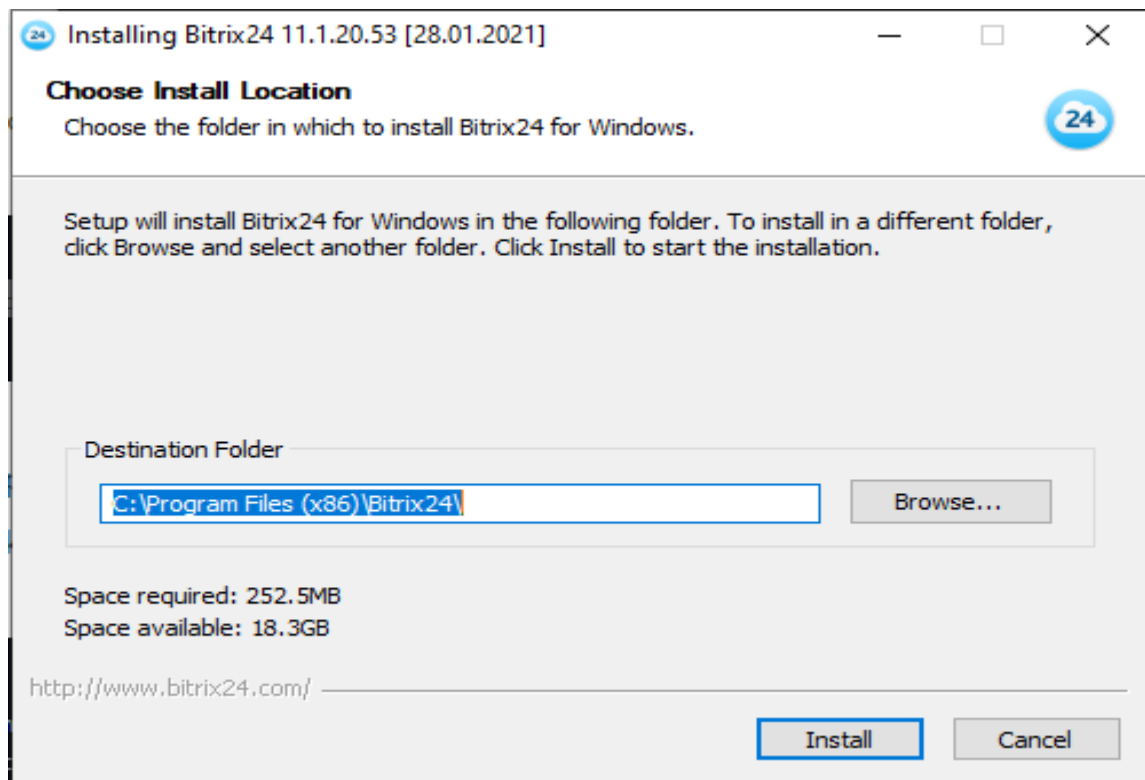
2 Installation Python -3.7.6

Figure 21: Installation Python



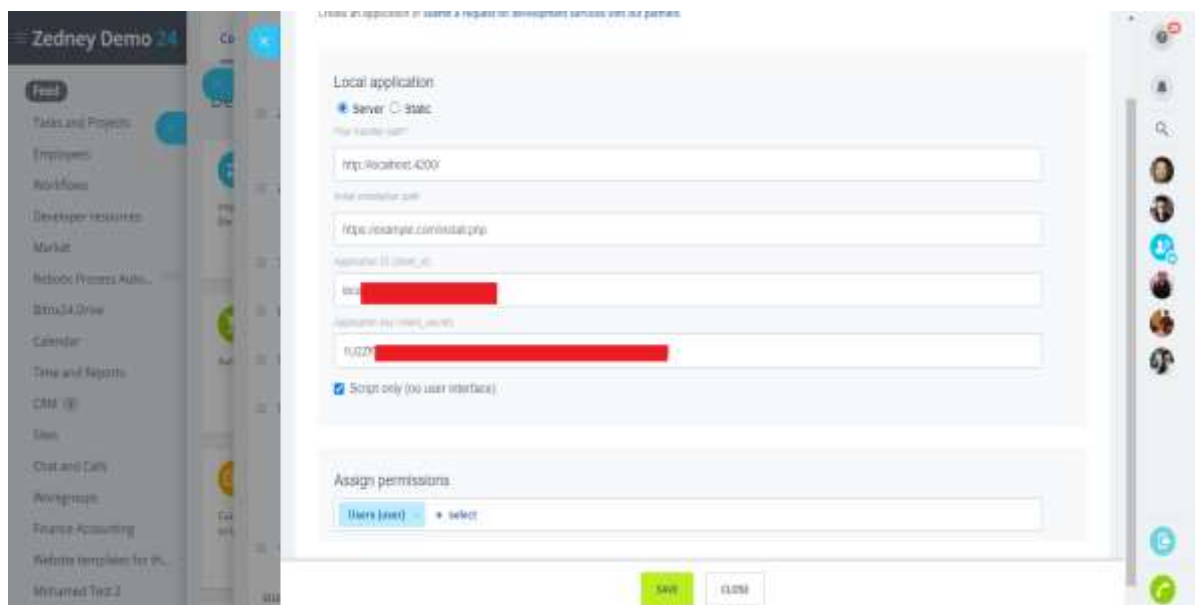
2 Installation Bitrix24

Figure 22: Installation bitrix24



2 Créer une application de développement via bitrix24

Figure 23: Configuration de API bitrix24



- ❑ Créer un Workspace Django API et Importer le package Django API et ses modules
- ❑ Ajouter python 3.7 et ses librairies

Figure 24: Configuration de Django Framework

```

1  """
2  Django settings for MyWebSite project.
3
4  Generated by 'django-admin startproject' using Django 1.10.1.
5
6  For more information on this file, see
7  https://docs.djangoproject.com/en/1.10/topics/settings/
8
9  For the full list of settings and their values, see
10 https://docs.djangoproject.com/en/1.10/ref/settings/
11 """
12
13 import os
14
15 # Build paths inside the project like this: os.path.join(BASE_DIR, ...)
16 BASE_DIR = os.path.dirname(os.path.dirname(os.path.abspath(__file__)))
17
18
19 # If you are not using the default Django admin interface, you may need to
20 # see https://docs.djangoproject.com/en/1.10/howto/deployment/checklist/
21
22 # SECURITY WARNING: keep the secret key used in production secret!
23 SECRET_KEY = '7p7y5*+}004g_2ro3-60^h-w15_-3z{nhq{)p0d^p0m=004!'
24
25 # SECURITY WARNING: don't run with debug turned on in production!
26 DEBUG = True
27
28 ALLOWED_HOSTS = []
29

```

IV. Etude des besoins

IV.1. Besoins fonctionnels

L'application à mettre en œuvre doit principalement satisfaire les besoins suivants :

❑ Besoins fonctionnels:

L'objectif de ce travail est de modéliser et développer une application mobile tout en modifiant l'architecture logicielle et utiliser des nouvelles technologies. Cette application présente 03 acteurs principaux à savoir : l'administrateur, l'employé et l'adhérent. Je vais décrire dans ce qui suit les services à satisfaire pour chaque acteur.

Le système doit permettre à l'administrateur de :

- Consulter les statistiques

- Consulter la liste des remboursements
- Consulter la liste des demandes de remboursement
- Gérer la liste des adhérents
- Gérer la liste des employés

Le système doit permettre à l'employé de :

- Valider les demandes des adhérents
- Initialiser les jours de travail
- Ajouter les remboursements et les actes d'adhérent

Le système doit permettre à l'adhérent de :

- Consulter les données de profil
- Modifier données de profil
- Consulter les données bancaires
- Modifier les données bancaires
- Chercher et, localiser les prestataires de santé
- Envoyer une demande (plusieurs types de demande)
- Suivre les demandes envoyées
- Consulter les bénéficiaires du contrat
- Consulter les détails des remboursements
- Filtrer la recherche des remboursements
- Contacter l'assistance
- Suivre des remboursements
- Générer les cartes de tiers payant
- Gérer les contrats

IV.2. Besoins non fonctionnels

Parmi les considérations et les contraintes additionnelles dont je dois tenir compte lors de la réalisation du projet je peux citer :

- 🔍 **Rapidité des traitements** : Vu que le système est utilisé par plusieurs acteurs, il est impérativement nécessaire que la durée d'exécution soit minimale.
- 🔍 **La performance** : Un logiciel doit être performant à travers ses fonctionnalités, il doit répondre à toutes les exigences des usagers d'une manière optimale...
- 🔍 **La convivialité** : Le futur logiciel doit être facile à utiliser. En effet, les interfaces

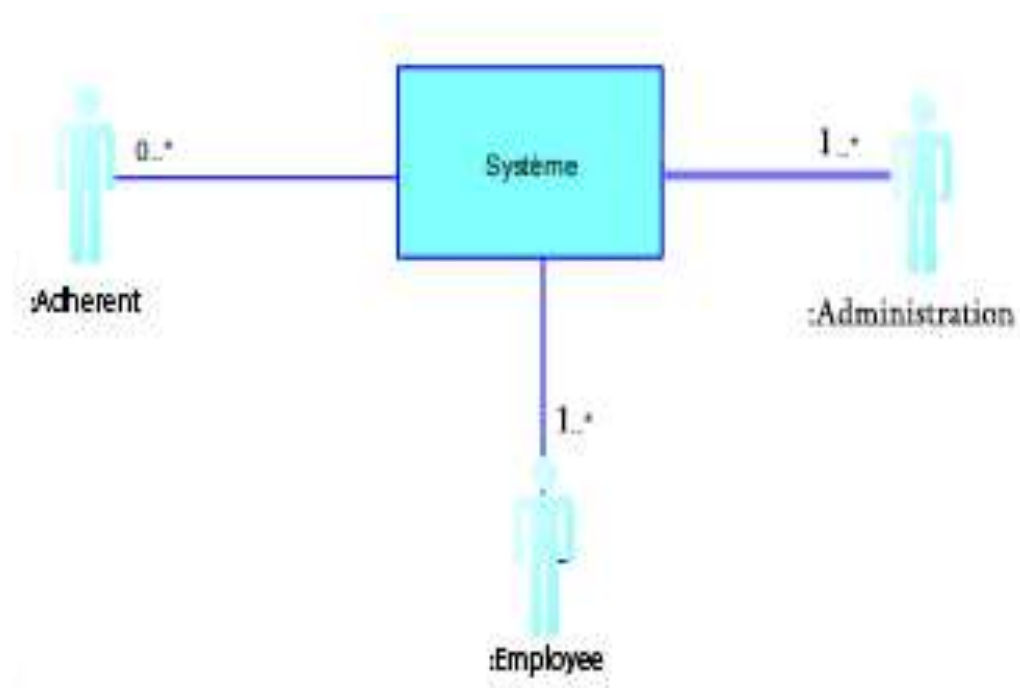
utilisateurs doivent être conviviales, simples, ergonomiques et adaptées à l'utilisateur, etc.

- ☐ **La sécurité** : Le système doit être sécurisé (identification des utilisateurs, définition des droits d'accès pour chaque type d'utilisateur, etc.
- ☐ **La disponibilité** : Le système doit être opérationnel à chaque fois que le service en a besoin, etc.
- ☐ **La fiabilité**: Le système doit résister à différentes pannes...

IV.3. Présentation des acteurs

Les acteurs du système sont représentés dans la figure suivante :

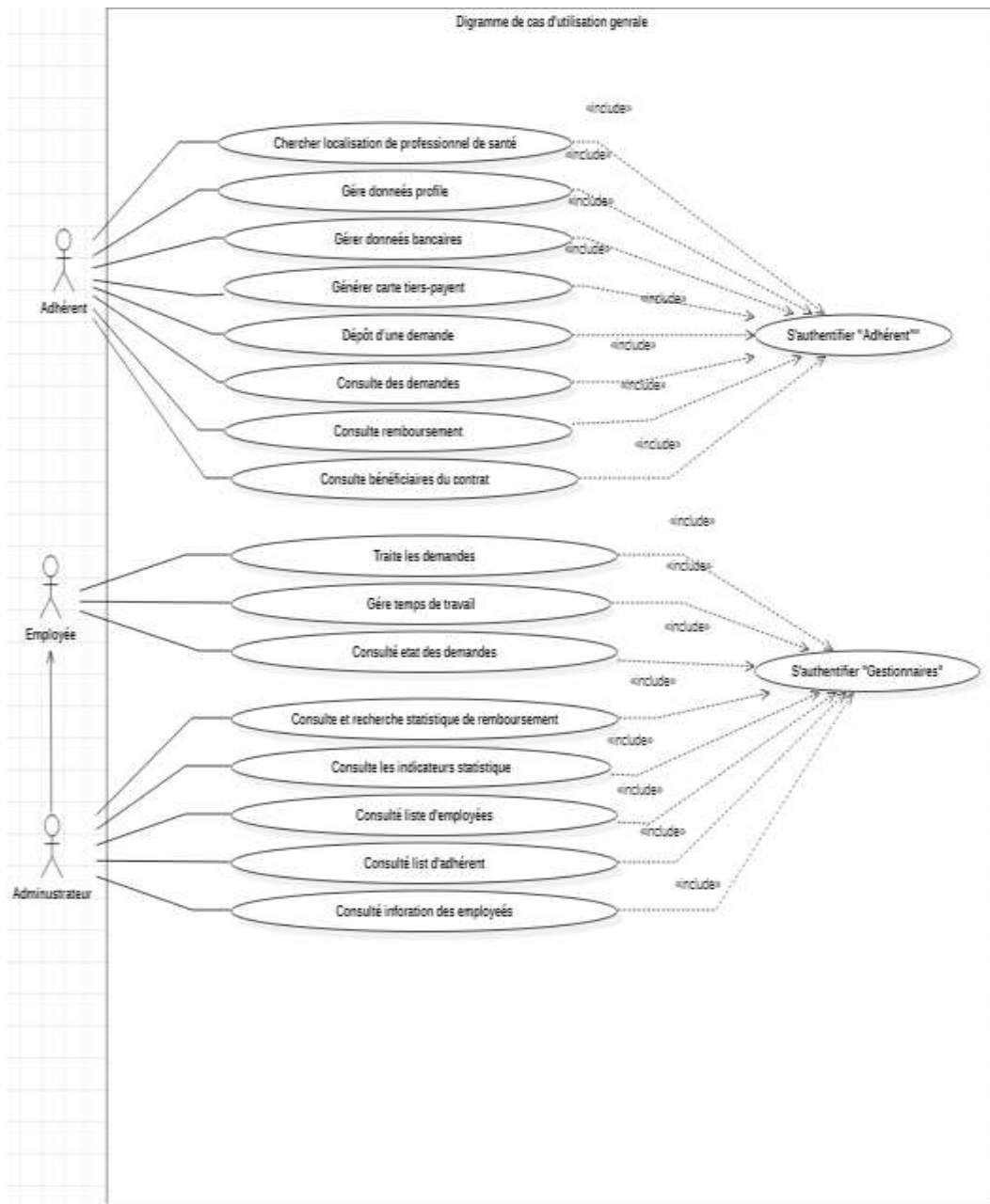
Figure 25: Diagramme de contexte statique des acteurs du système



IV.4. Diagramme de cas d'utilisation

Dans ce qui suit, une présentation générale des cas d'utilisation sera faite, ensuite elle sera étalée sur plusieurs sous cas d'utilisation qui seront brièvement analysés un par un dans chaque sprint. La figure suivante présente les divers cas d'utilisation assurés par le système.

Figure 26 : Diagramme de cas d'utilisation général



Conclusion

Ce chapitre a été consacré pour la spécification des besoins fonctionnels et non Fonctionnels du système résultant, le diagramme d'acteurs et le diagramme de cas d'utilisation générale.

Dans le prochain chapitre, je vais détailler les deux premiers sprints.

Chapitre 3 : Release 1

Introduction

Dans ce chapitre nous allons détailler les sprints de la première release avec l'expression des besoins, la conception et les interfaces réalisées.

I. Sprint 1

1. Sprint Goal

Le but de ce Sprint est d'avoir un livrable présentant des fonctionnalités les plus sollicitées par les 03 acteurs de l'application.

Ce livrable comporte 6 modules :

- Le premier module permet à l'adhérent de se connecter à son compte
- Le deuxième module permet à l'adhérent de déposer une demande
- La troisième partie permet à l'administrateur de se connecter à son espace
- La quatrième partie permet à l'administrateur de consulter les nouveaux remboursements
- La cinquième partie permet à l'administrateur de consulter les statistiques
- La sixième partie permet à l'employé de traiter les demandes

2. Sprint Back log

Après avoir défini le but de ce premier sprint, une réunion a eu lieu pour définir le périmètre de ce sprint (les éléments du back log de produit qui vont être réalisés), l'identification, l'affectation et l'estimation des tâches à réaliser. Le résultat de cette réunion donne naissance au sprint back log (tableau)

Tableau 05: Back log Sprint 1

Connexion à un compte « adhérent »	En tant qu'adhérent, je veux me connecter à mon compte	3	Elevé
Dépôt d'une demande	En tant qu'adhérent, je veux déposer une demande de remboursement	4	Elevé

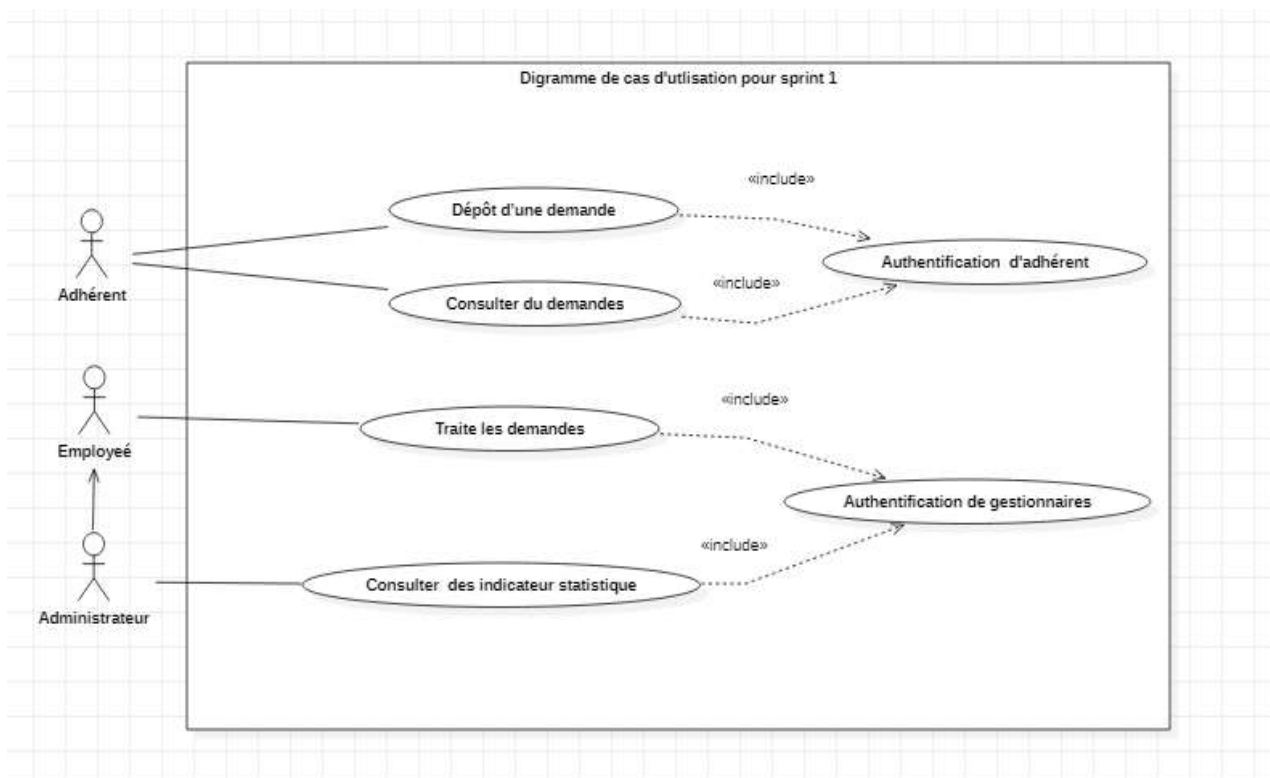
1	Consultation des demandes	En tant qu'adhérent, je veux consulter les nouvelles demandes	3	Elevé
	Connexion à un compte « employé »	En tant qu'administrateur, je veux connecter à mon compte	2	Elevé
	Consultation des indicateurs et des statistiques	En tant qu'administrateur, je veux consulter des indicateurs statistiques (Dashboard)	5	Elevé
	Traient des demandes	En tant qu'employé, je veux traiter la demande	4	Elevé

3. Analyse

L'étape de spécification des besoins est très importante puisque la réussite de toute Étude dépend de la qualité de son départ. Il faut donc bien déterminer les fonctions Attendues par le système. La représentation des besoins sera exprimée sous la forme des Diagrammes des cas d'utilisations qui permettent de représenter les utilisateurs ainsi que les interactions entre les acteurs et le système.

3.1. Diagramme de cas d'utilisation

Figure 27: Diagramme de cas d'utilisation pour sprint 1



3.2. Description textuelle

Tableau 06 : Description du cas d'utilisation « Dépôt d'une demande »

Acteur(s) :	Adhérent
Objectif :	Dépôt d'une demande
Précondition(s) :	Avoir un mot de passe et un identifiant Sécuriser
Scénario nominal :	<ol style="list-style-type: none"> 1. Naviguer vers la tab bar 2. Cliquer sur « Faire une demande à mon conseiller » 3. Choisir l'objectif 4. Entrer un message 5. Ajouter un document 6. Cliquer sur « Valider »
Scénario alternatif :	Si le nom d'utilisateur ou le mot de passe ne sont pas valides, le système affiche un message d'erreur

Tableau 07 : Description du cas d'utilisation « Consultations des demandes envoyées »

Acteur(s) :	Adhérent
Objectif :	Suivre les demandes envoyées
Pré-condition(s) :	Client Authentifié
Scénario nominal :	<ol style="list-style-type: none"> 1. Naviguer vers tab bar 2. Cliquer sur « Faire une demande à mon conseiller » 3. Cliquer sur SUIVRE
Scénario alternatif :	Si le temps d'accès de jeton est fini, il faut générer un autre jeton

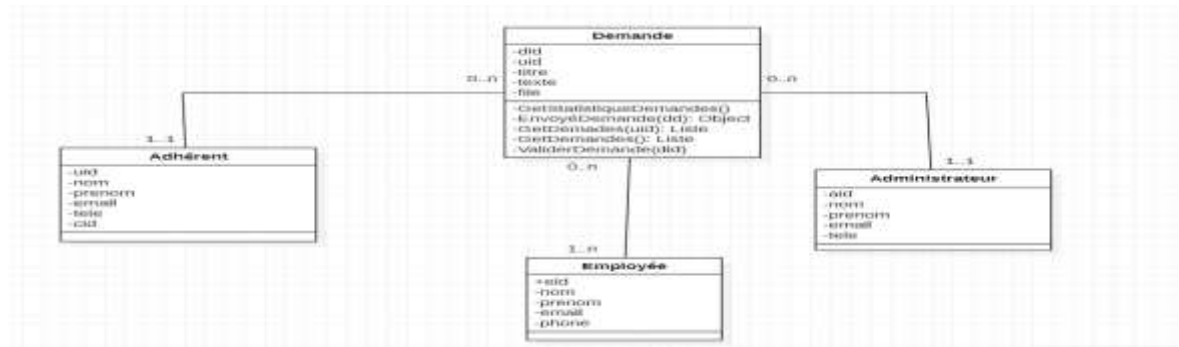
Tableau 08 : Description du cas d'utilisation « Traiter les demandes »

Acteur(s) :	Adhérent
Objectif :	Traiter les demandes
Pré-condition(s) :	Client Authentifié
Scénario nominal :	<ol style="list-style-type: none"> 1. Ouvrir le navbar 2. Cliquer sur une demande 3. Consulter une demande et valider
Scénario alternatif :	Si le temps d'accès de jeton fini il faut générer un autre jeton

4. Conception du sprint

4.1. Diagramme de classe

Figure 28: Diagramme de class pour sprint 1



4.2. Diagramme de séquence

Figure 29: Diagramme de séquence pour dépôt de demande

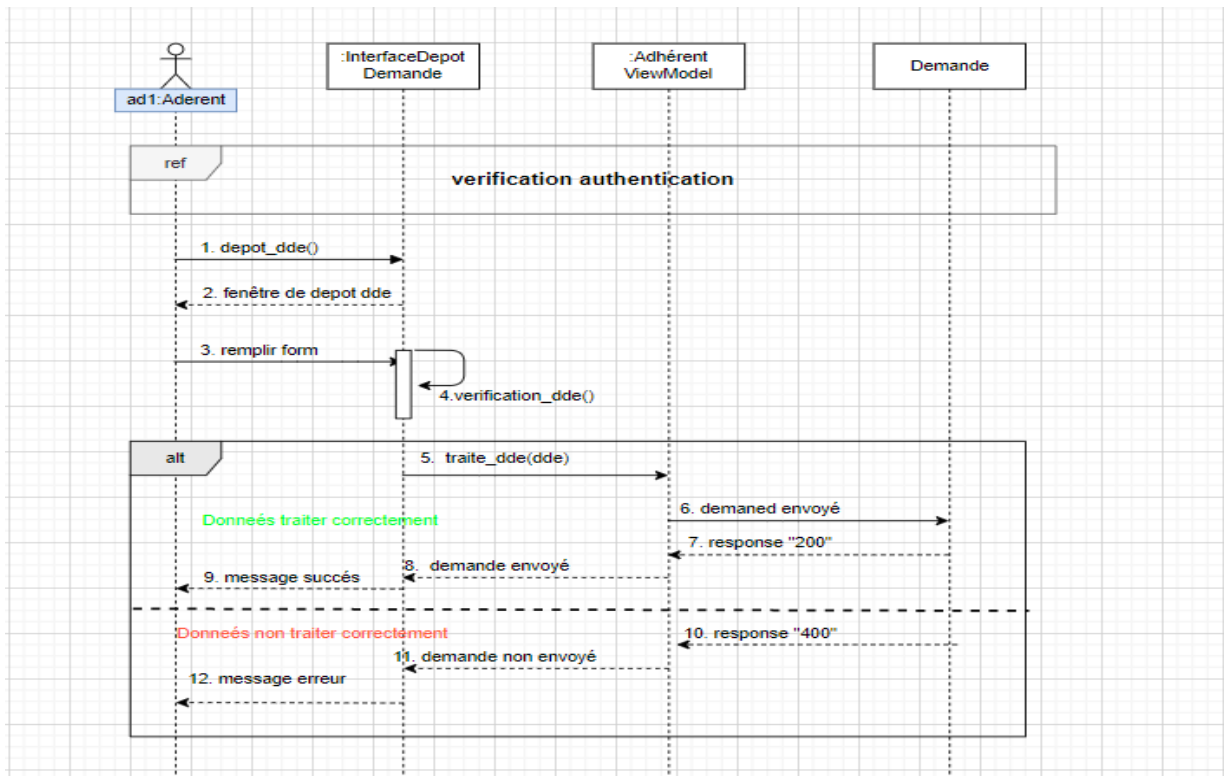


Figure 30: Diagramme de séquence pour traite une demande

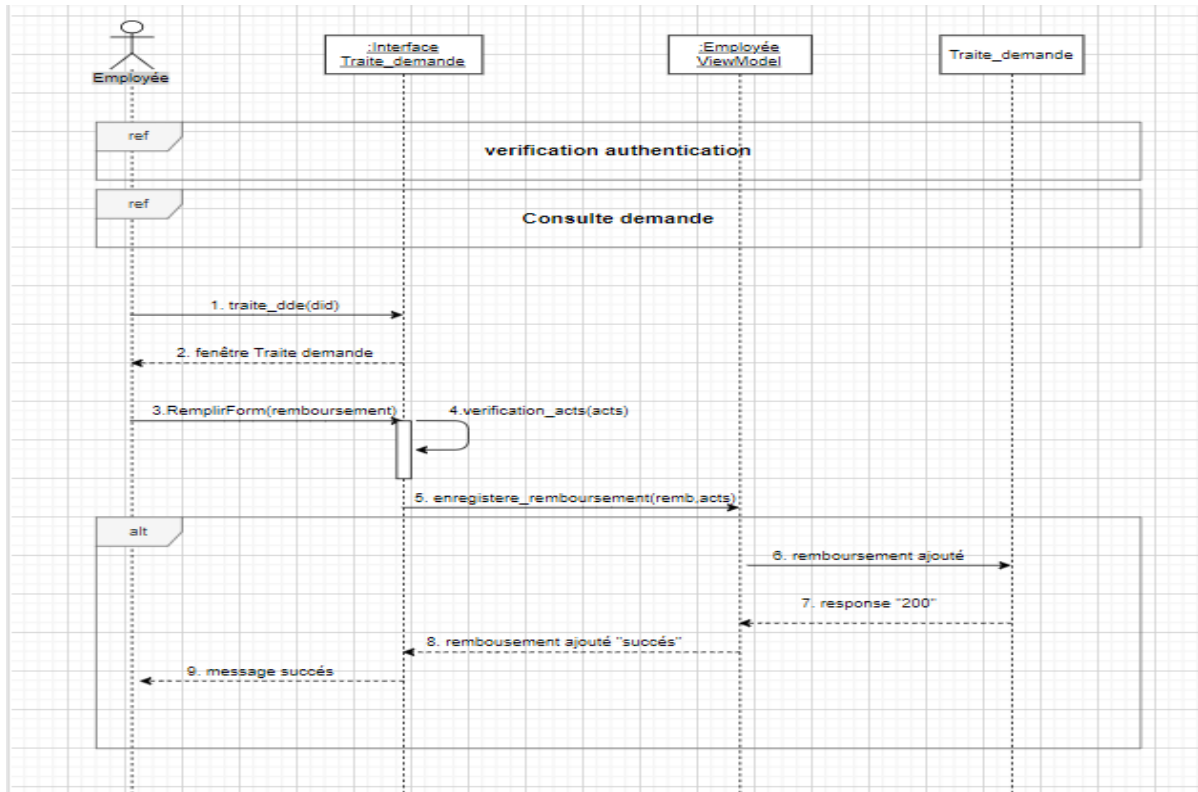
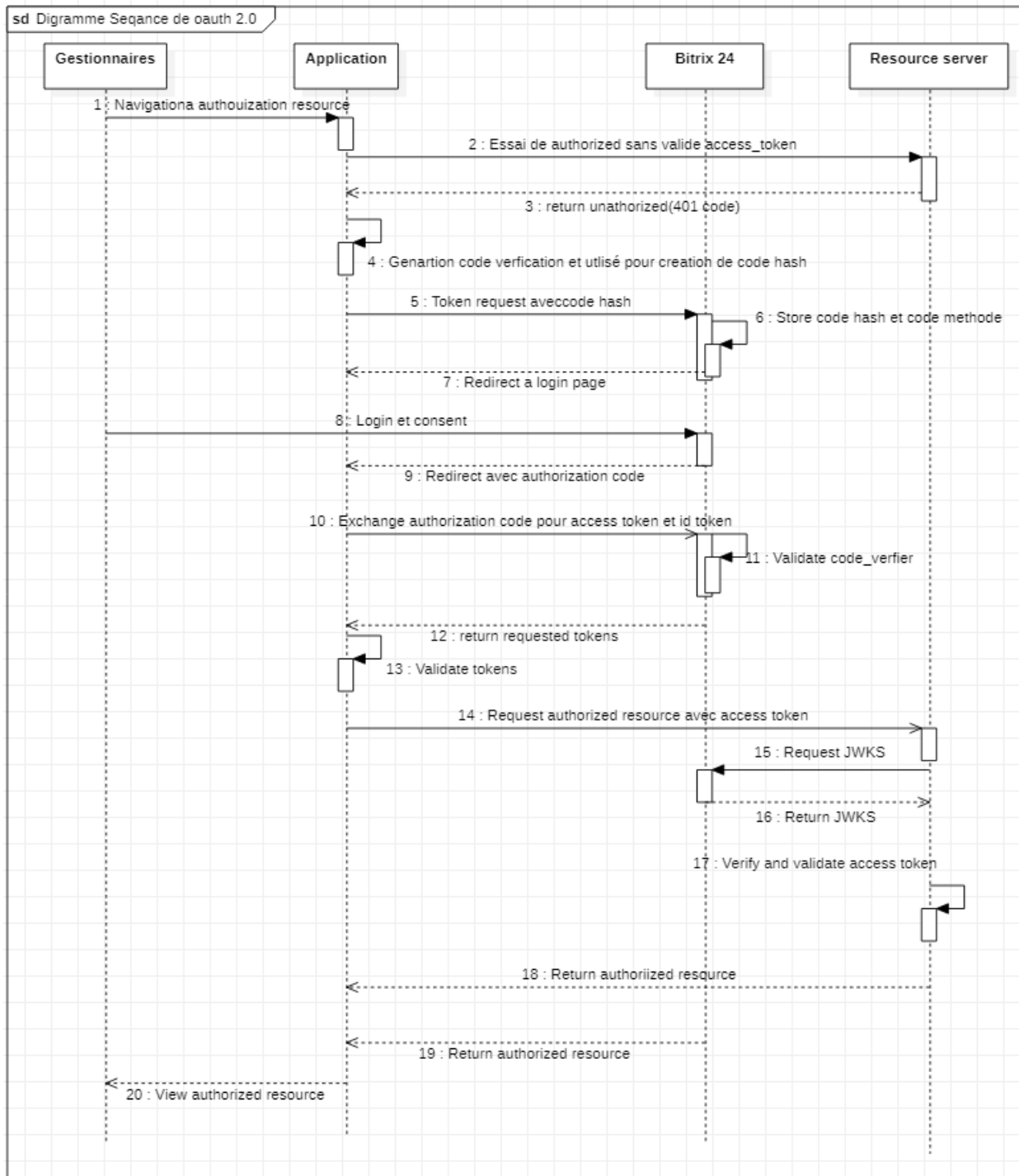


Figure 31: Digramme de séquence générale avec bitrix24 pour oauth 2.0



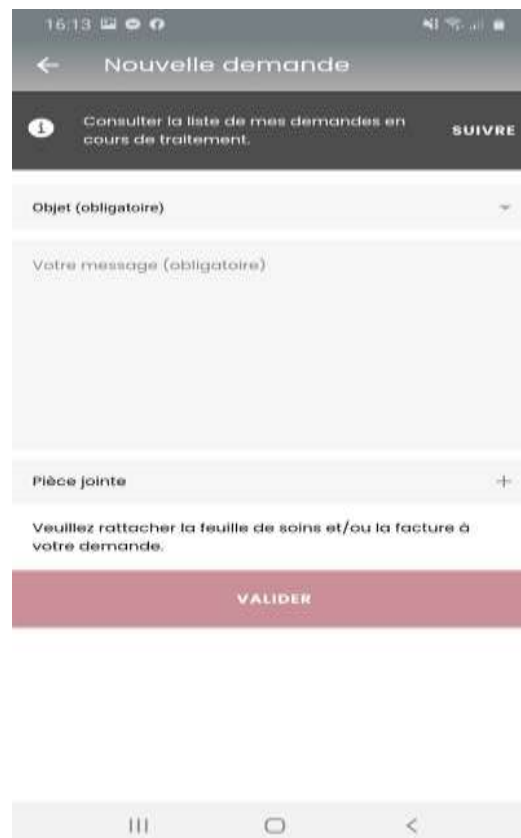
5. Réalisation et test d'incrimination

Figure 32: Interface Login pour Adhérent



La figure 32 présente l'interface où l'adhérent s'authentifie

Figure 33: Interface dépôt d'une demande



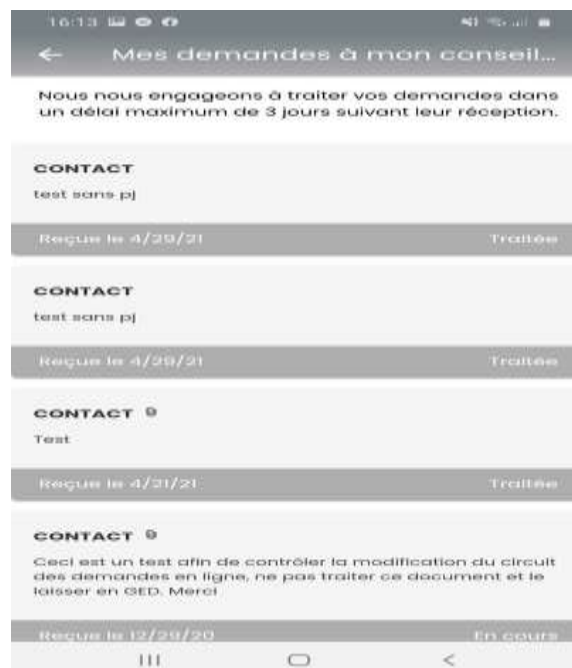
La figure 33 présente l'interface où l'adhérent peut déposer une demande de remboursement.

Figure 34: Insertion d'un document ou image dans demande



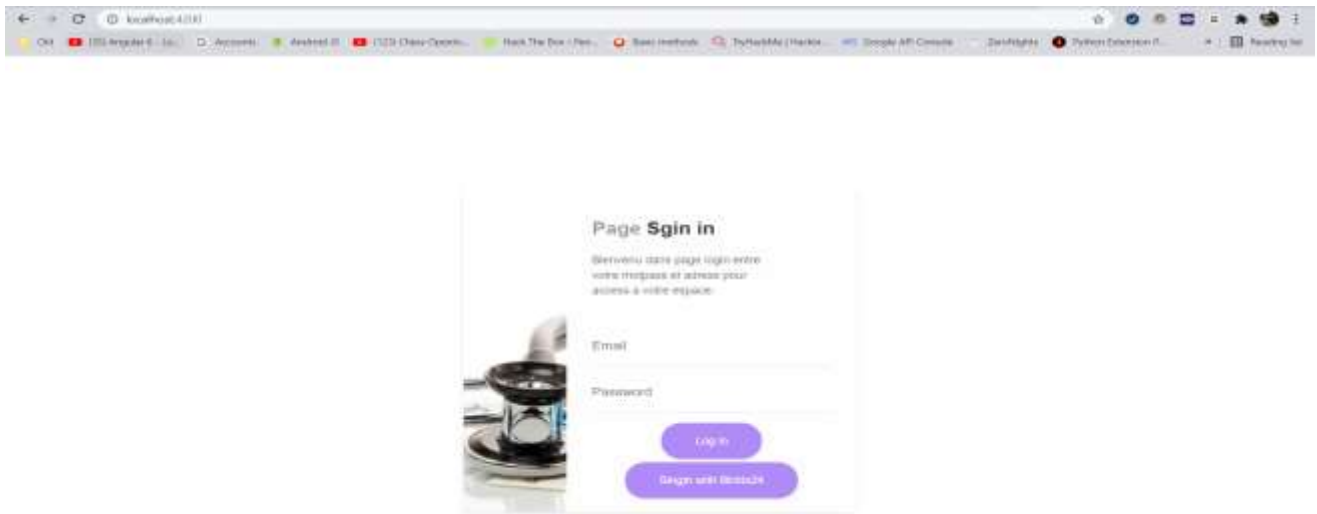
La figure 34 affiche les différents type d'ajout des documents

Figure 35: Interface liste de demande



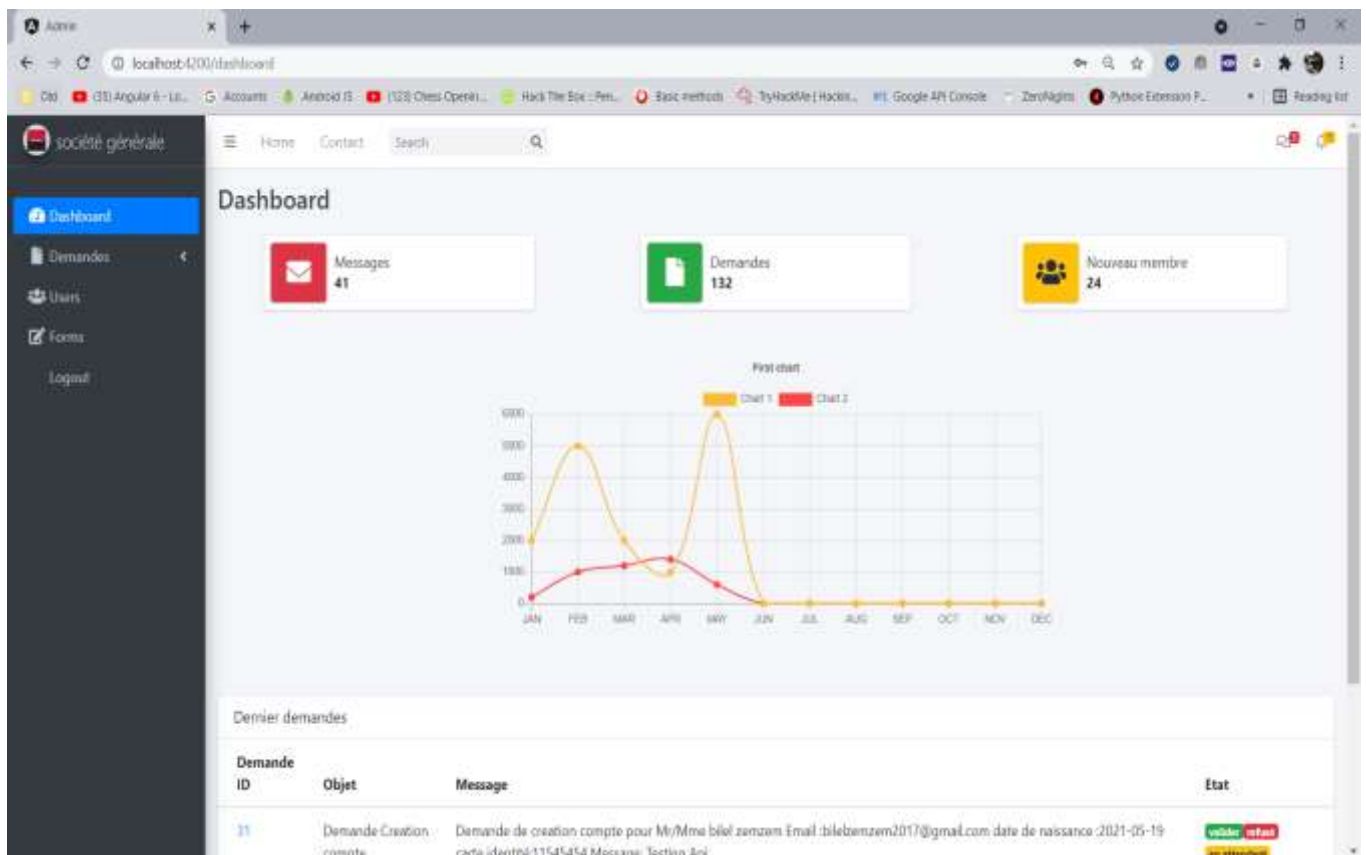
La figure 35 affiche l'archive des demandes

Figure 36: Interface Login pour les employés ou L'administrateur



La figure 36 présente l'interface d'administration - login

Figure 37: Interface Dashboard pour l'administrateur



La figure 37 affiche le Dashboard de l'administrateur

Figure 38: Interface ajoute remboursement

La figure 38 depuis cette interface, l'employé peut ajouter un remboursement a un adhérent quel quant

6. Sprint review

Durant le Sprint Review, j'ai présenté l'incrément réalisé au Product Owner en présence du scrum master et l'équipe de développement. Ce livrable a été validé par le product owner. Aucun problème à signaler n'a été détecté durant ce sprint et les stories du client n'ont pas été modifiés.

II. Sprint 2

1. Sprint Goal

Le but de ce Sprint est de réaliser 6 modules :

- Le premier module permet à l'adhérent de consulter et modifier son profil
- Le deuxième module permet à l'adhérent de consulter et modifier ses données bancaires

- La troisième partie permet à l'adhérent rechercher et localiser des professionnels de santé
- La quatrième partie permet à l'adhérent de générer une carte de tiers-payant
- La cinquième partie permet l'adhérent de se connecter à son espace client
- La sixième partie permet adhérent de consulter ses remboursements

2. Sprint Back log

Une fois le but de notre Sprint est défini, il est temps de décider quelles sont ses fonctionnalités et leurs degrés d'importance. Le tableau ci-dessous résume le Back log de notre sprint :

Tableau 09 : Back log Sprint 2

2	Modification de mes données	En tant qu'adhérent, je veux modifier mon profil	2	Moyenne
	Modification de mes données bancaires	En tant qu'adhérent, je veux modifier mes données bancaires	2	Moyenne
	Recherche et localisation des professionnels de santé	En tant qu'adhérent, je veux rechercher et localiser des professionnels de santé	5	Moyenne
	Génération des carte tiers-payant	En tant qu'adhérent, je veux générer ma carte de tiers-payant	6	Moyenne
	Connexion SSO à l'espace client	En tant qu'adhérent, je veux me connecter à mon espace client sans entrer mes identifiants une autre fois.	3	Moyenne
	Consultation remboursement	En tant qu'adhérent, je veux consulter mes remboursements	3	Moyenne

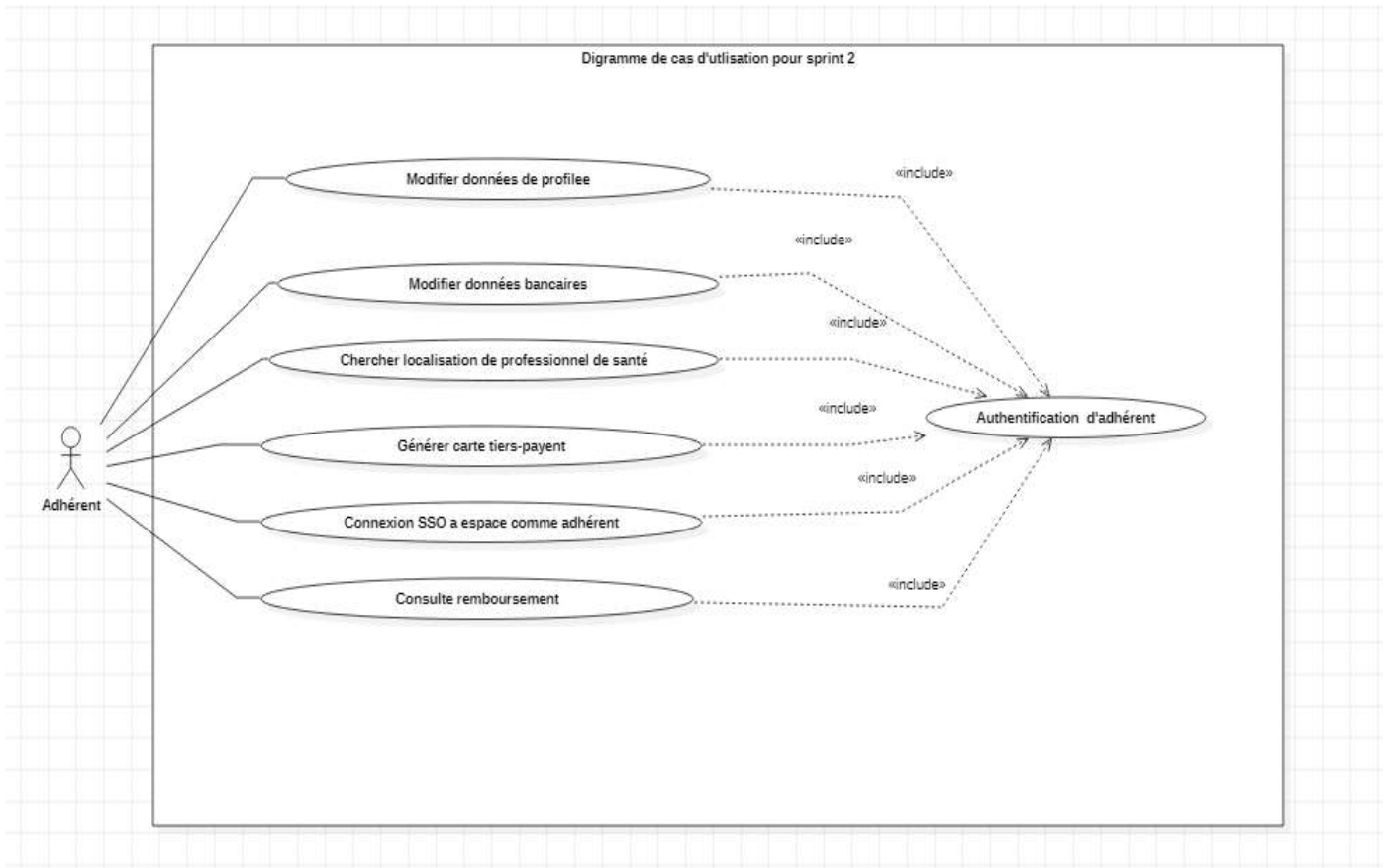
3. Analyse

L'étape de spécification des besoins est très importante puisque la réussite de toute Étude dépend de la qualité de son départ. Il faut donc bien déterminer les fonctions Attendues par le système. La représentation des besoins sera exprimée sous la forme des Diagrammes des

cas d'utilisations qui permettent de représenter les utilisateurs ainsi quelles interactions entre les acteurs et le système.

3.1. Digramme de cas d'utilisation

Figure 39: Digramme de cas d'utilisation pour sprint 2



3.2. Description textuelle

Tableau 10 : Description du cas d'utilisation « Consulté de données bancaires »

Acteur(s) :	Adhérent
Objectif :	Consulter les données personnelles
Précondition(s) :	Adhérent Authentifié
Scénario nominal :	<ol style="list-style-type: none"> 1. Naviguer vers le menu de gauche 2. Cliquer sur Mes données bancaires
Scénario alternatif :	Si le temps d'accès de jeton fini if faux générer un autre jeton

Tableau 11 : Description du cas d'utilisation « Modifier données bancaires »

Acteur(s) :	Adhérent
Objectif :	Modifier mes données bancaires
Précondition(s) :	Adhérent Authentifié
Scénario nominal :	<ol style="list-style-type: none"> 1. Naviguer vers le menu de gauche 2. Cliquer sur Mes données bancaires 3. Cliquer sur l'icône de stylo 4. Editer les données du compte de virement 5. Valider les données correctes 6. Cliquer sur le bouton valider
Scénario alternatif :	Si une case doit être remplie, un message s'affiche

Tableau 12 : Description du cas d'utilisation « Chercher localisation un professionnel de santé »

Acteur(s) :	Adhérent
Objectif :	rechercher et localiser des professionnels de santé
Précondition(s) :	Adhérent Authentifié
Scénario nominal :	<ol style="list-style-type: none"> 1. Naviguer vers le tab bar 2. Cliquer sur Trouvez un professionnel de santé 3. Choisir une spécialité 4. Activer le GPS ou entrer une adresse 5. Entre le nom du médecin (Optionnel) 6. Cliquer sur recherche 7. Consulter carte
Scénario alternatif :	Si le temps d'accès de jeton est terminé, il faut générer un autre jeton

Tableau 13 : Description du cas d'utilisation « Suivi des remboursements »

Acteur(s) :	Adhérent
Objectif :	Consulter les Décomptes
Pré-condition(s) :	Adhérent Authentifié

Scénario nominal :	<ol style="list-style-type: none"> 1. Naviguer vers la page principale 2. Cliquer sur « Consulter tous mes décompte » 3. Visualisation du graphique de remboursement
Scénario alternatif :	S'il n'y a aucun décompte, un message s'affiche

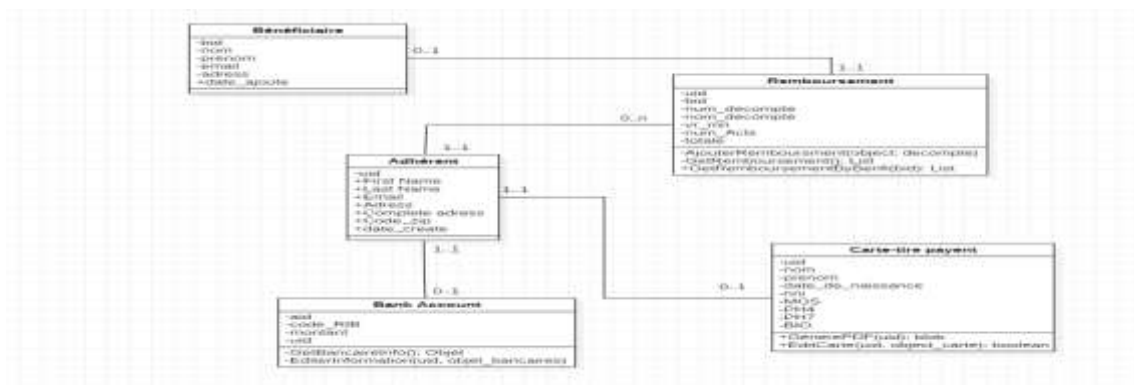
Tableau 14 : Description du cas d'utilisation « Générer carte de tiers payant en fichier PDF »

Acteur(s) :	Adhérent
Objectif :	Générer carte de tiers payant en fichier PDF
Pré-condition(s) :	Client Authentifié
Scénario nominal :	<ol style="list-style-type: none"> 1. Naviguer vers la page principale 2. Cliquer sur « Ma carte de tiers-payant » 3. Visualisation du fichier PDF
Scénario alternatif :	S'il n'y a aucun décompte, un message s'affiche

4. Conception du sprint 1

4.1. Diagramme de classe

Figure 40: Diagramme de class pour sprint 2



4.2. Diagramme de séquence

Figure 41: Diagramme de séquence éditer profile

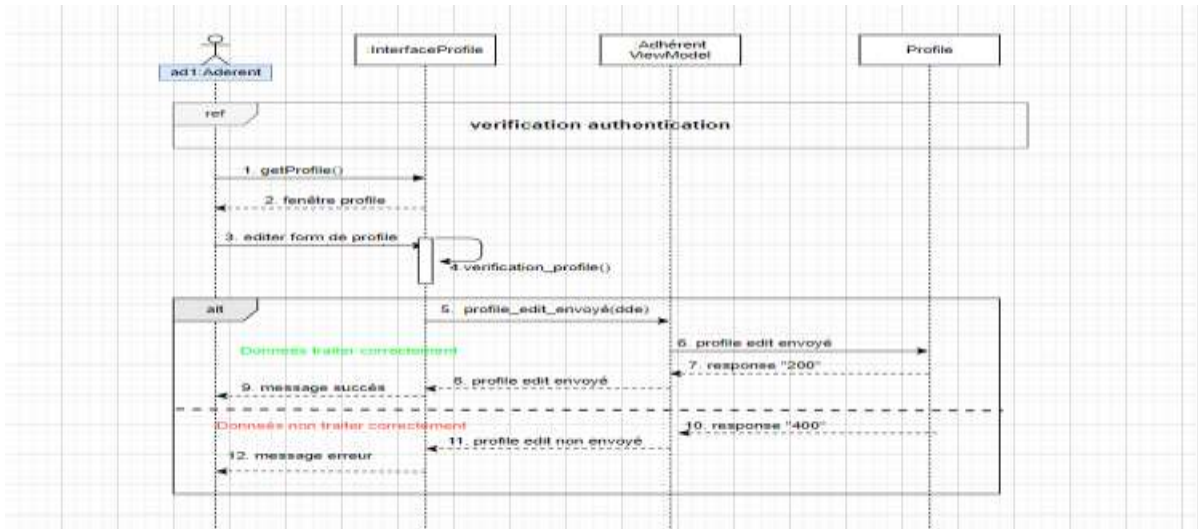


Figure 42: Diagramme de séquence consulté remboursement

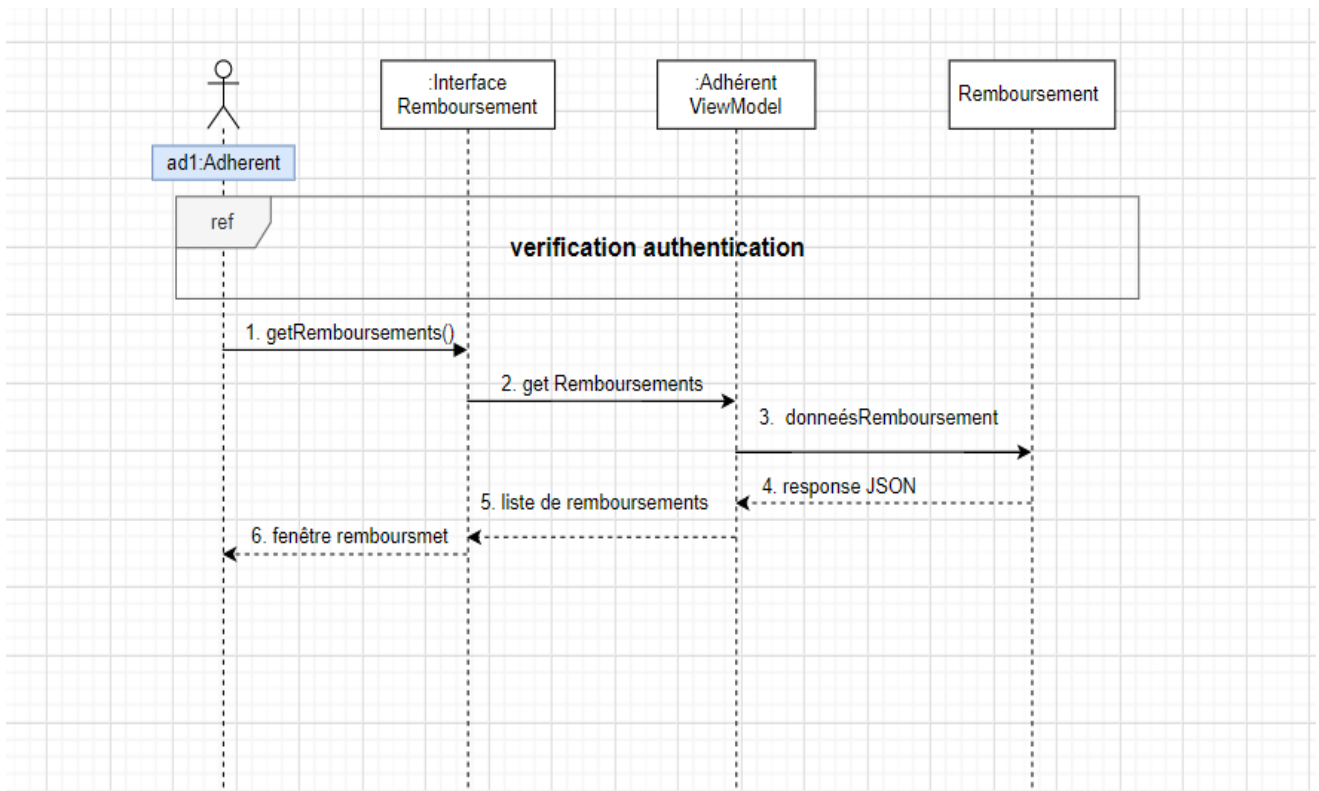
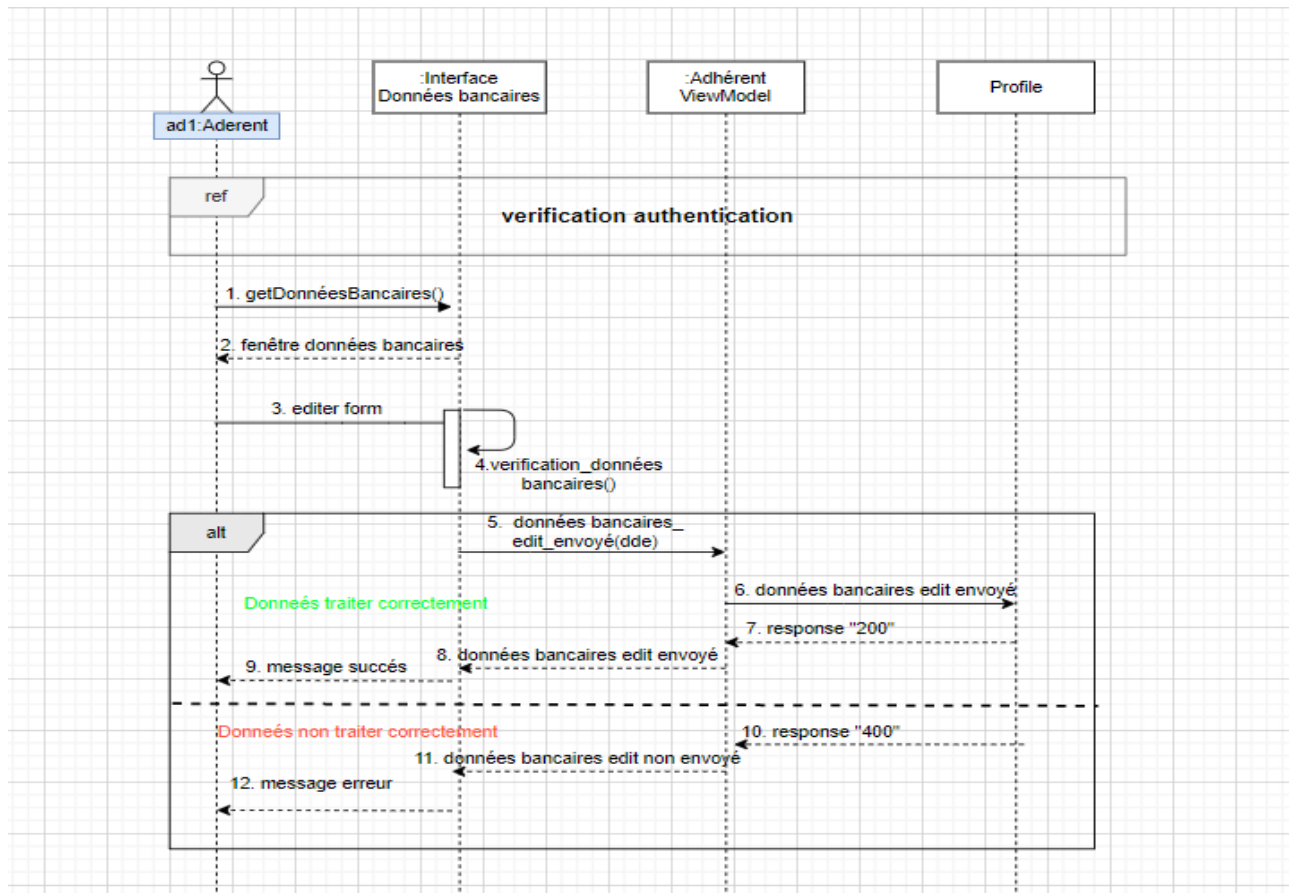


Figure 43: Diagramme de séquence éditer données bancaire



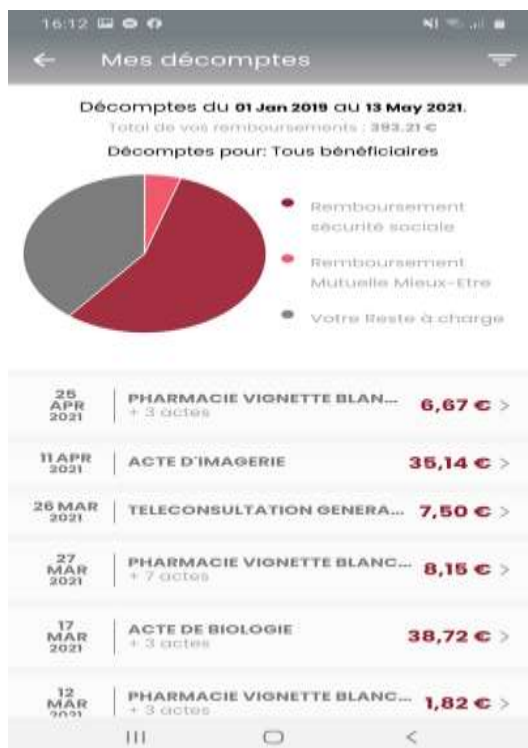
5. Réalisation et test d'incrimination

Figure 44: Interface générée carte tire payent



La figure 44 présente le chargement lors de la génération de la carte tiers-payant en PDF. Pour générer cette carte.

Figure 45: Interface consulté remboursement



La figure 45 présente l'interface de consultation des statistiques de remboursement

Figure 46: Interface édit données bancaire

16:17

← Modifier mes données bancaires

MR BORDE

[Redacted Address]

Sélectionner le type d'opération (o... Sélectionner une ville

Date d'effet souhaitée 13/05/2021

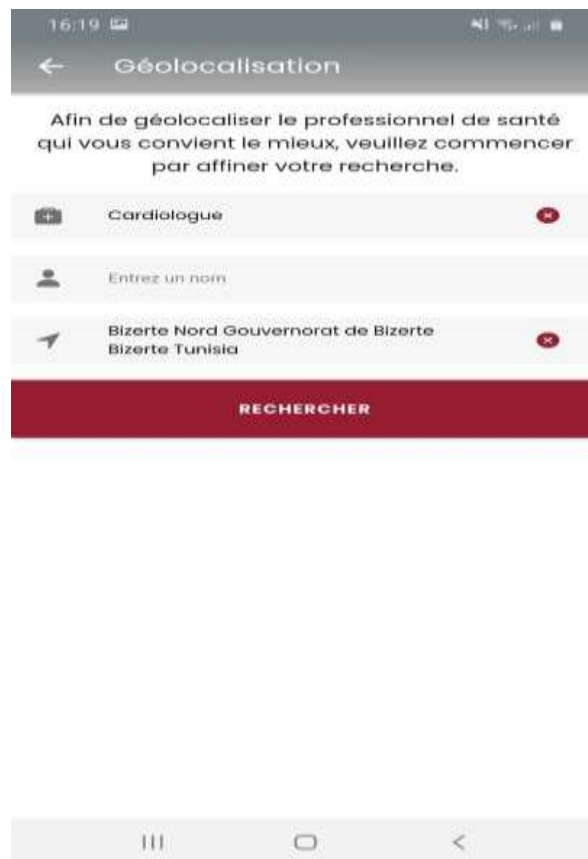
Ajouter un relevé d'identité bancaire +

Ajouter un mandat de prélèvement SEPA +

VALIDER

La figure 46 présente l'interface de modification des données bancaires

Figure 47: Interface chercher une professionnelle de santé



La figure 47 présente l'interface qu'il faut remplir pour activer et lancer la recherche du professionnel de santé

Figure 48: Interface résultat de recherche



La figure 48 affiche les résultats de recherche des professionnels de santé

Figure 49: Interface lien de connexion SSO



La figure 49 affiche les liens SSO « Single sign-on » implémentés dans notre application

6. Sprint review

Durant le Sprint Review, j'ai présenté l'incrément réalisé au Product Owner en présence du scrum master et l'équipe de développement. Ce livrable a été validé par le product owner. Aucun problème à signaler n'a été détecté durant ce sprint et les stories du client n'ont pas été modifiés.

Chapitre 4 : Release 2

Introduction

Dans ce chapitre je vais détailler les sprints de la deuxième release avec l'expression des besoins, la conception et les interfaces réalisées.

I. Sprint 3

1. Sprint 3 Goal

Le but de ce Sprint est de réaliser 6 modules :

- Le premier module permet à l'administrateur de consulter et rechercher les indicateurs statistiques de remboursement
- Le deuxième module permet à l'adhérent de faire une demande d'inscription
- La troisième partie permet à l'administrateur de consulter les demandes
- La quatrième partie permet à l'adhérent de générer une carte de tiers-payant
- La cinquième partie permet à l'adhérent de consulter les bénéficiaires de son contrat
- La sixième partie permet à l'administrateur de consulter les listes des adhérents
- La septième partie permet à l'administrateur de consulter les listes des employés

2. Sprint Back log

Une fois le but de notre Sprint est défini, il est temps de décider quelles sont ses fonctionnalités et leurs degrés d'importance. Le tableau ci-dessous résume le Backlog de notre sprint :

Tableau 15 : Back log Sprint 3

Consultation et recherche des statistiques de mes différents remboursements	En tant qu'administrateur, je veux consulter et rechercher des indicateurs statistiques de mes remboursements	6	Faible
S'inscrire	En tant qu'un visiteur, je veux m'inscrire	3	Faible

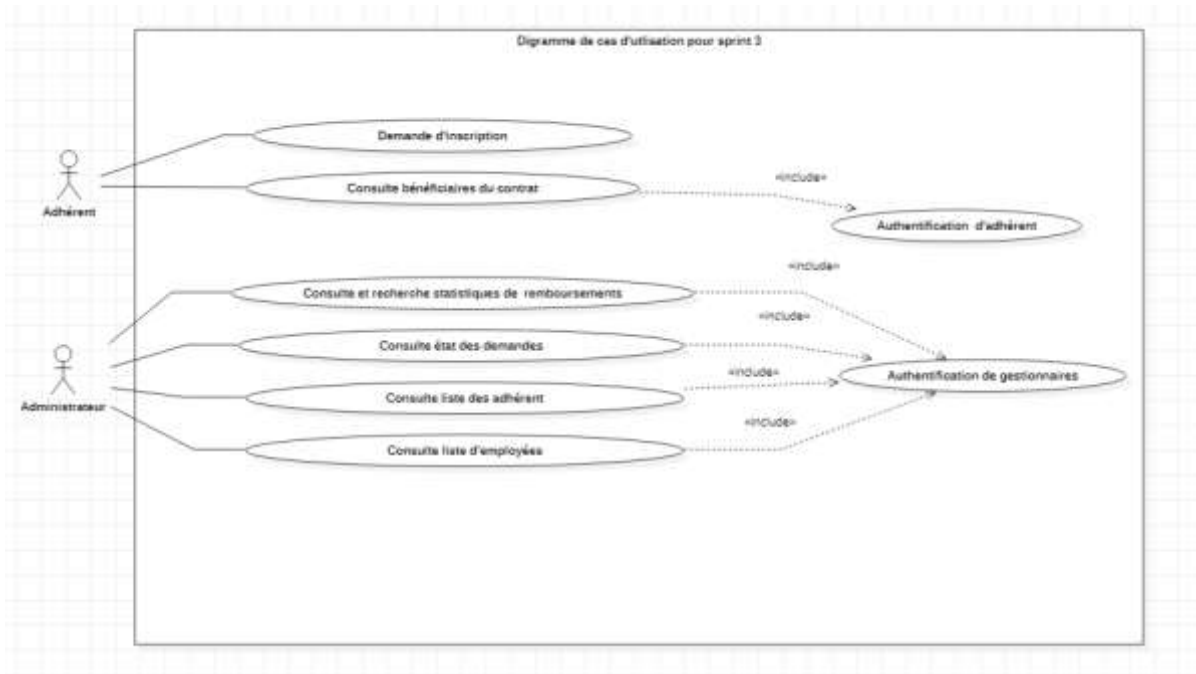
3	Consultation des états de mes demandes	En tant qu'administrateur, je veux consulter les états de mes demandes	3	Faible
	Consultation des bénéficiaires du contrat	En tant qu'adhérent, je veux consulter les bénéficiaires de mon contrat	3	Faible
	Consultation de la liste des adhérents	En tant qu'administrateur, je veux consulter les adhérents	3	Faible
	Consultation de la liste des employés	En tant qu'administrateur, je veux consulter et filtrer la liste des employés	3	Faible

3. Analyse

3.1. Diagramme de cas d'utilisation

Ci-dessous la liste des diagrammes de cas d'utilisation qui permettent de représenter les utilisateurs ainsi que les interactions entre les acteurs et le système.

Figure 50: Diagramme de cas d'utilisation pour sprint 3



3.2. Description textuelle

Tableau 16 : Description du cas d'utilisation « Demande d'inscription »

Acteur(s) :	Adhérent
Objectif :	Envoyer une demande d'inscription
Précondition(s) :	Client Authentifié
Scénario nominal :	<ol style="list-style-type: none"> 1. Ouvrir l'application 2. Naviguer vers la création du compte 3. Remplir le formulaire 4. Click sur « SIGN UP »
Scénario alternatif :	Si la forme manque de données, un message d'erreur s'affiche

Tableau 17 : Description du cas d'utilisation « Consulter les bénéficiaires de mon contrat »

Acteur(s) :	Adhérent
Objectif :	Consulter les bénéficiaires de mon contrat
Précondition(s) :	Client Authentifié
Scénario nominal :	<ol style="list-style-type: none"> 1. Naviguer vers le Navbar dans l'application 2. Cliquer sur Bénéficiaires du contrat
Scénario alternatif :	Si la forme manque de données, un message d'erreur s'affiche

Tableau 18 : Description du cas d'utilisation « Consulter les états des demandes »

Acteur(s) :	Administrateur
Objectif :	Consulte des indicateurs de remboursement
Précondition(s) :	Administrateur Authentifié
Scénario nominal :	<ol style="list-style-type: none"> 1. Ouvrir l'application 2. Cliquer sur le navbar 3. Cliquer sur demandes 4. Consulter les demandes
Scénario alternatif :	Si aucune demande, un message s'affiche

Tableau 19 : Description du cas d'utilisation « Consulter des indicateurs de remboursement »

Acteur(s) :	Administrateur
Objectif :	Consulter des indicateurs de remboursement
Précondition(s) :	Administrateur Authentifié
Scénario nominal :	<ol style="list-style-type: none"> 1. Ouvrir l'application 2. Naviguer vers le Dashboard 3. Consulter les statistiques de remboursement
Scénario alternatif :	Si pas de remboursements, un message s'affiche

Tableau 20 : Description du cas d'utilisation « Consulter la liste des adhérent »

Acteur(s) :	Administrateur
Objectif :	Consulte liste des employés
Précondition(s) :	Administrateur Authentifié
Scénario nominal :	<ol style="list-style-type: none"> 1. Ouvrir l'application 2. Naviguer vers Clients 3. Consulter la liste de client
Scénario alternatif :	Si pas de d'adhérents, un message s'affiche

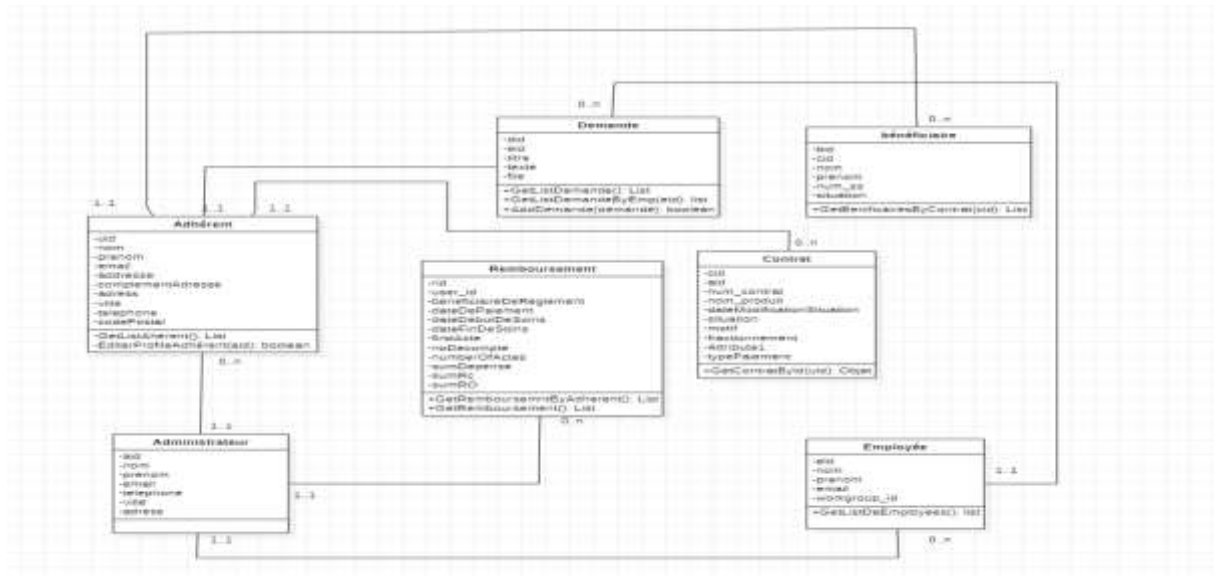
Tableau 21 : Description du cas d'utilisation « Consulter la liste des employées »

Acteur(s) :	Administrateur
Objectif :	Consulter liste des employés
Précondition(s) :	Administrateur Authentifié
Scénario nominal :	<ol style="list-style-type: none"> 1. Ouvrier l'application 2. Naviguer vers Employées 3. Consulter liste des employés
Scénario alternatif :	Si aucun décompte sur compte, un message s'affiche

4. Conception du sprint 3

4.1. Diagramme de classe

Figure 51: Diagramme de class pour sprint 3



4.2. Diagramme de séquence

Figure 52: Diagramme de séquence Consultation des remboursements

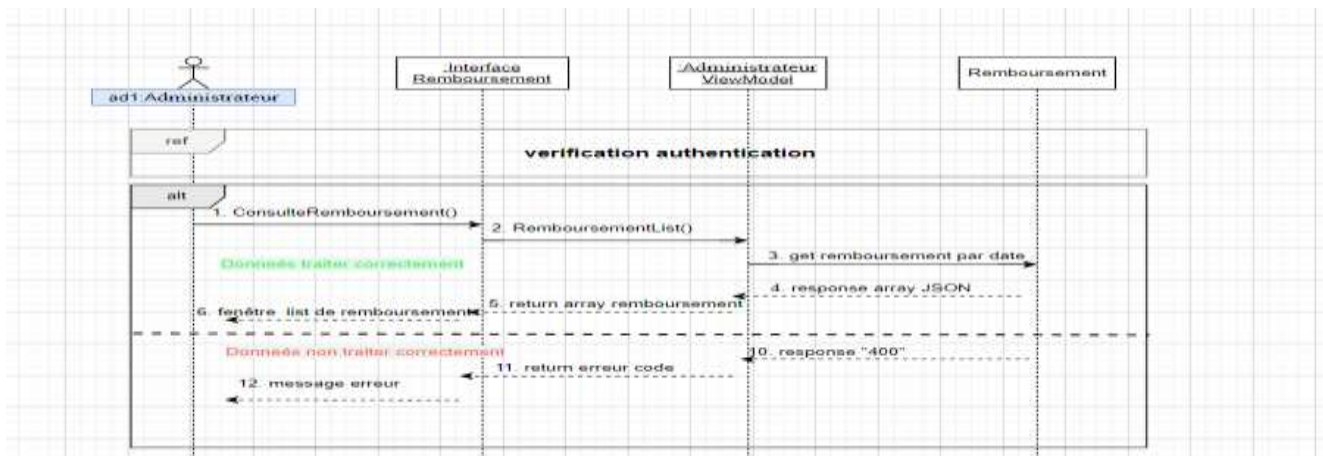


Figure 53: Diagramme de séquence Consultation de la liste des employés

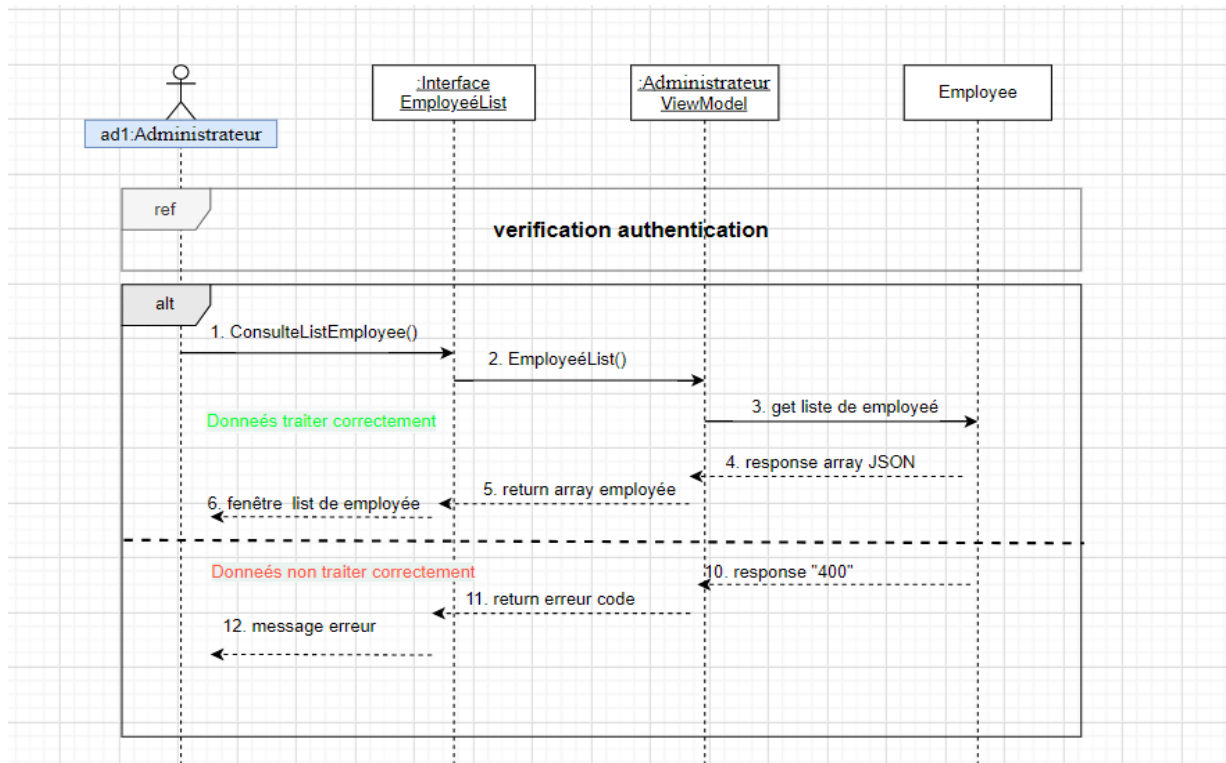
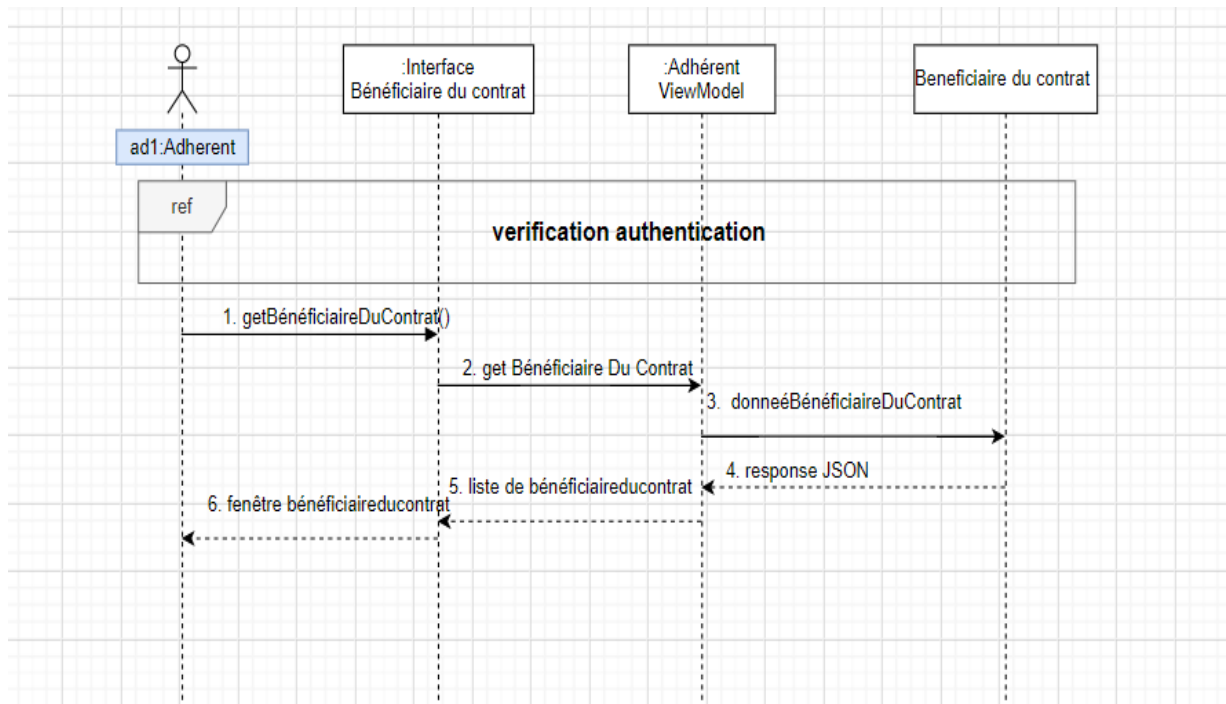


Figure 54: Diagramme de séquence Consultation du Bénéficiaire du contrat



5. Réalisation

Figure 55: Interface Demande d'inscription

16:46

← Demande Creation du Compte

Create Account

Last name

First name

Secande name

Email

+216 ** *** **

Carte identité

Description

La figure 55 présente l'interface de demande d'inscription

Figure 56: Interface Consulté du Bénéficiaire du contrat

16:18

← Mes bénéficiaires

NOM	Rôle
BORDE VINCENT	Assuré
Numéro de sécurité sociale	
Situation	
En cours	
GAMBARD SANDRINE	Conjoint
Numéro de sécurité sociale	
Situation	
En cours	
DOURI ALI	Enfant
Numéro de sécurité sociale	
Situation	
En cours	
DOURI SILVIA	Enfant

La figure 56 présente l'interface de consultation des bénéficiaire du contrat

Figure 57: Interface Consulté les demandes



La figure 57 présente l'interface de consultation des demandes de l'adhérent

6. Sprint review

Durant le Sprint Review, j'ai présenté l'incrément réalisé au Product Owner qui l'a validée. Le cahier des charges de départ a été validé par le client.

Après une discussion avec le Product Owner, ce dernier a présenté de nouvelles fonctionnalités qui les a jugées indispensables. Ces fonctionnalités ont été acceptées par l'équipe de travail et par conséquent, elles ont été ajoutées dans le back log Product. Ces nouvelles fonctionnalités sont les suivantes :

- Gérer le temps de travail
- Consulter les statuts des connexions pour les différents employés
- Consulter le temps de travail des employés

II. Sprint 4

1. Sprint 4 Goal

Le but de ce Sprint est de réaliser 6 modules :

- Le premier module permet à l'employé de gérer son temps du travail.

- Le deuxième module permet à l'administrateur de consulter le statut de connexion de l'employé.
- La troisième partie permet à l'administrateur de consulter le temps de travail des employés par jour.

2. Sprint Back log

Une fois le but de notre Sprint est défini, il est temps de décider quelles sont ses fonctionnalités et leurs degrés d'importance. Le tableau ci-dessous résume le Back log de notre sprint :

Tableau 22 : Back log Sprint 4

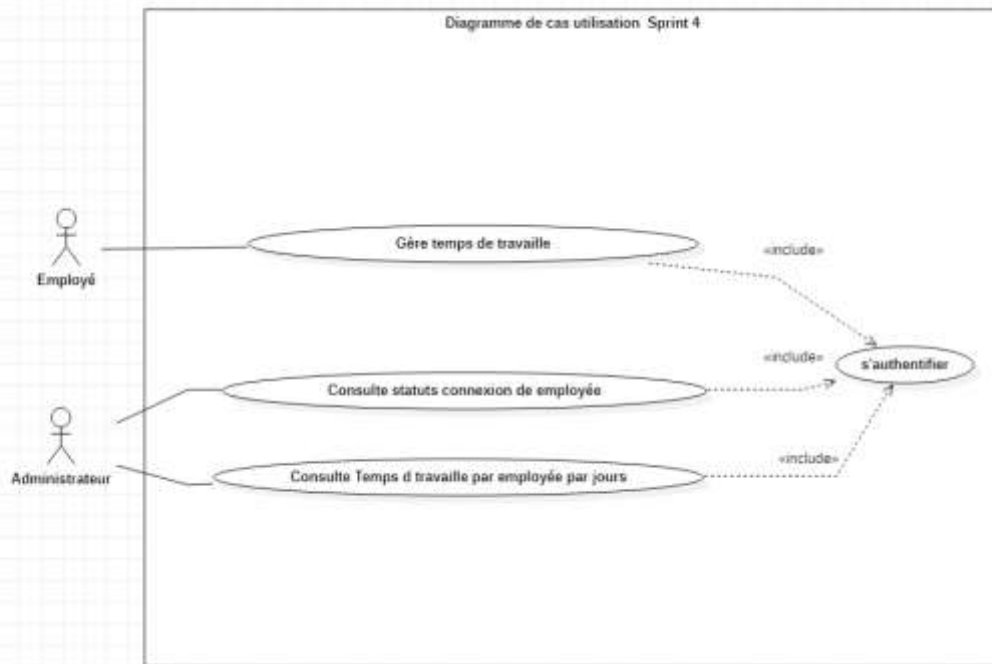
4	Gestion du temps de travail	En tant qu'employé, je veux gérer mon temps du travail.	4	Faible
	Consultation des statuts de connexion des différents employés	En tant qu'administrateur, je veux consulter les statuts de connexion des différents employés	4	Faible
	Consultation du temps de travail des différents employés	En tant qu'administrateur, je veux consulter le temps de travail des différents employés par jours	3	Faible

3. Analyse

3.1. Diagramme de cas d'utilisation

Diagrammes des cas d'utilisations qui permettent de représenter les utilisateurs ainsi que les interactions entre les acteurs et le système.

Figure 58: Diagramme de cas d'utilisation pour sprint 4



3.2. Description textuelle

Tableau 23 : Description du cas d'utilisation « Gérer le temps de travail »

Acteur(s) :	Employé
Objectif :	Consulter les statuts de connexion des employés
Précondition(s) :	Employé Authentifié
Scénario nominal :	<ol style="list-style-type: none"> 1. Ouvrir l'application 2. Naviguer vers le Dashboard 3. Cliquer sur l'icône Play ou pause 4. Contrôler votre temps
Scénario alternatif :	S'il y a des données manquantes, un message d'erreur s'affiche

Tableau 24 : Description du cas d'utilisation « Consulter les statuts de connexion des employés »

Acteur(s) :	Administrateur
Objectif :	Consulter les statuts de connexion des employés
Précondition(s) :	Administrateur Authentifié

Scénario nominal :	<ol style="list-style-type: none"> 1. Ouvrir l'application 2. Naviguer vers Employés 3. Consulter les statuts de connexion des employés
Scénario alternatif :	S'il y a des données manquantes, un message d'erreur s'affiche

Tableau 25 : Description du cas d'utilisation « Consulter les Temps de travail des employés »

Acteur(s) :	Administrateur
Objectif :	Consulter les Temps de travail des employé
Précondition(s) :	Administrateur Authentifié
Scénario nominal :	<ol style="list-style-type: none"> 1. Ouvrir l'application 2. Naviguer vers Employé 3. Choisir un employé 4. Consulter le temps de travail
Scénario alternatif :	S'il y a des données manquantes, un message d'erreur s'affiche

4. Conception du sprint 3

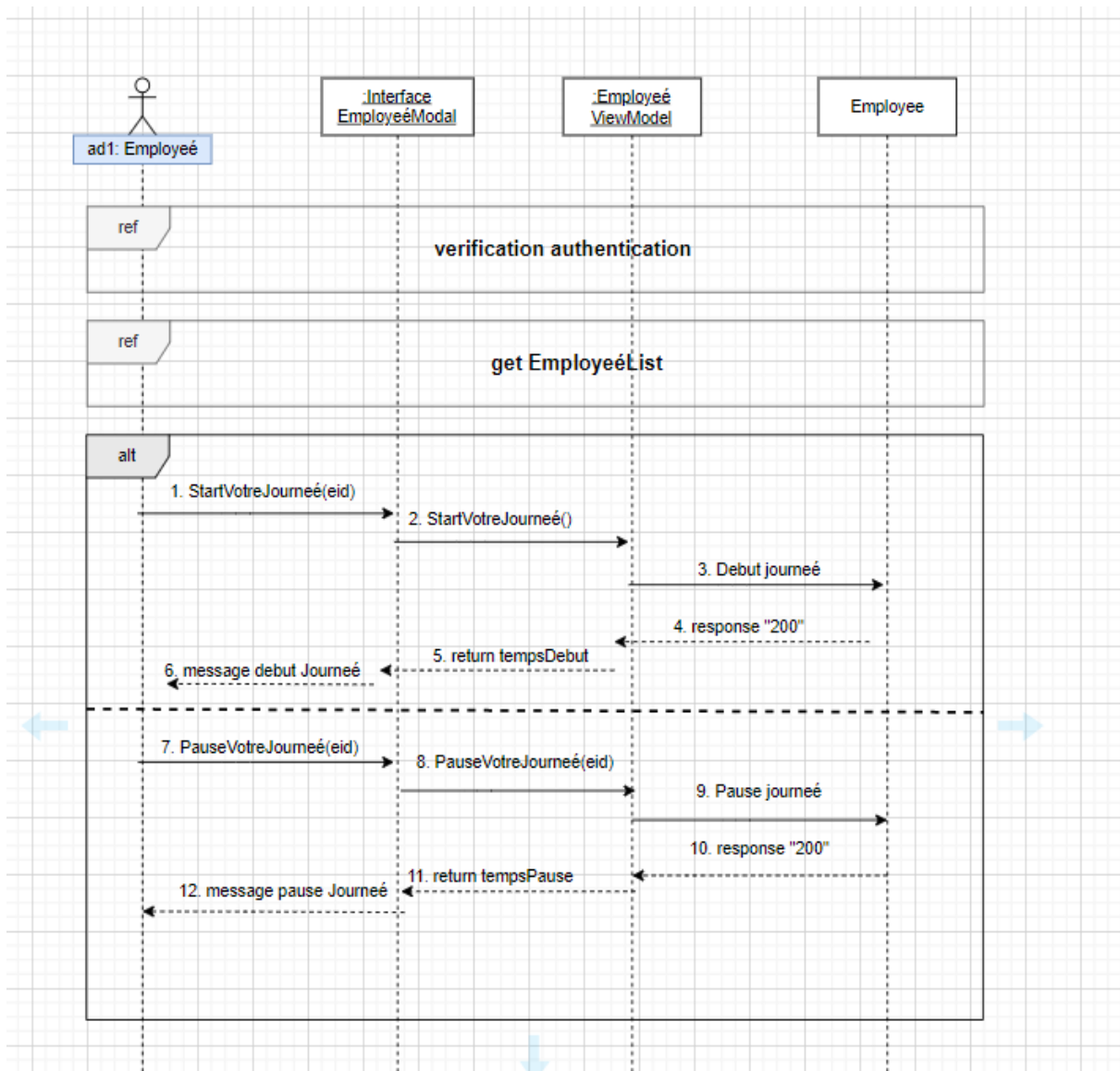
4.3. Digramme de classe

Figure 59: Digramme de classe pour sprint 4



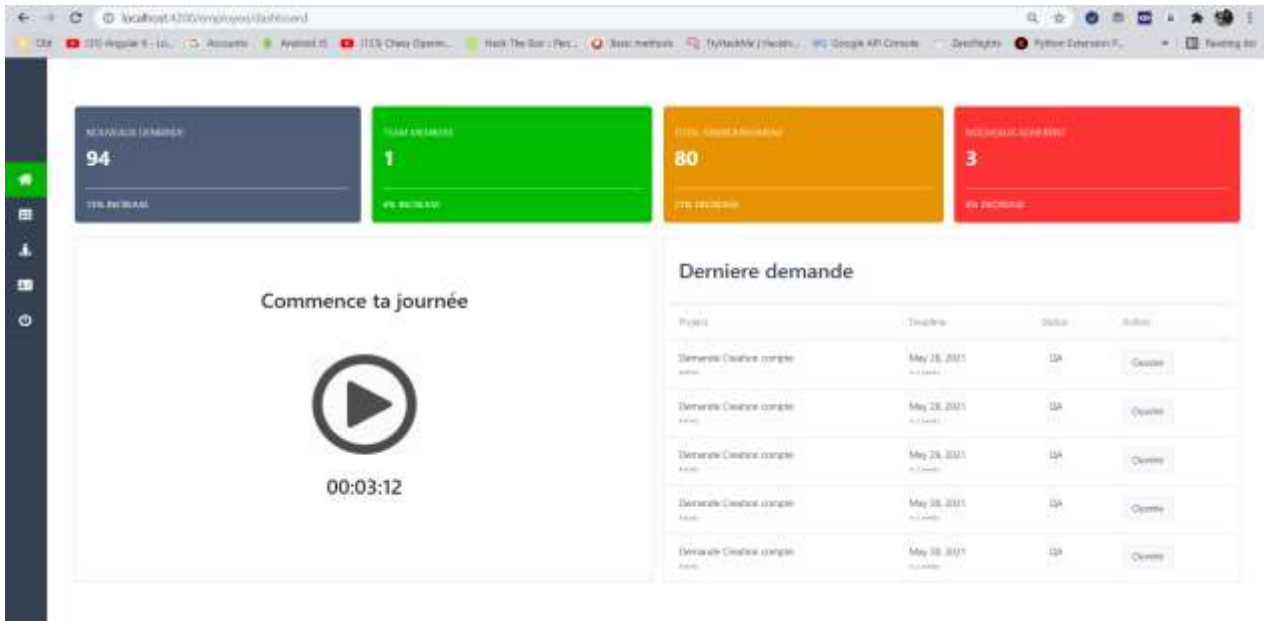
4.4. Digramme de séquence

Figure 60: Digramme de séquence géré temps de travail



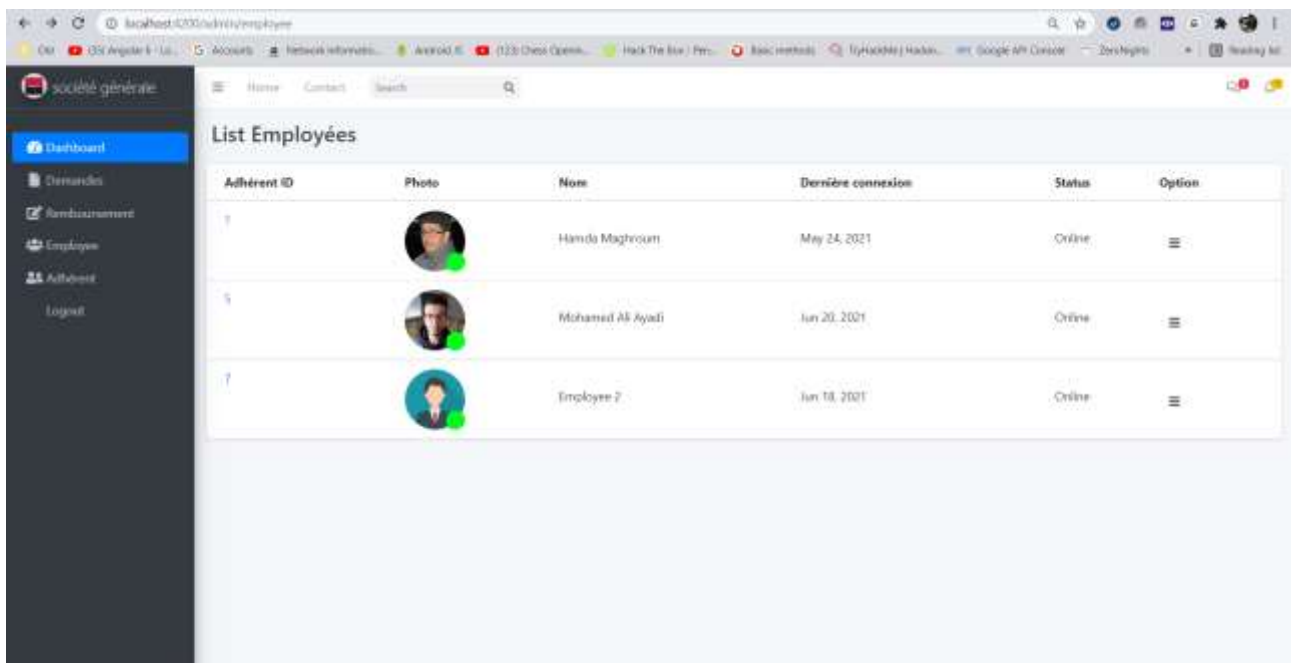
5. Réalisation et test d'incrimination

Figure 61: Interface Dashboard Employée



La figure 61 présente l'interface Dashboard dédiée à l'employé

Figure 62: Interface liste des employés



La figure 62 présente la liste des employés

6. Sprint review

Durant le Sprint Review, j'ai présenté l'incrément réalisé au Product Owner en présence du scrum master et l'équipe de développement. Ce livrable a été validé par le product owner. Aucun problème à signaler n'a été détecté durant ce sprint et les stories du client n'ont pas été modifiés.

Conclusion générale

Ce projet de fin d'études a été réalisé au sein de l'entreprise Zedney Creative. Il a consisté au développement et la mise en place d'un Système d'assurance pour la gestion et le suivi des remboursements.

Tout d'abord, j'ai commencé par comprendre le contexte général de notre application et identifier les différentes exigences de notre futur système. J'ai préparé par la suite notre planning de travail en respectant les priorités de mon besoins suite à une discussion au sein de l'équipe. Ensuite, j'ai conçu et développé une solution qui permet de gérer la totalité des services demandés par le client selon la méthodologie Scrum.

Ce stage de fin d'études au sein de Zedney Creative, s'est déroulé dans des très bonnes conditions. Il m'a permis de mettre en pratique mes connaissances théoriques. C'était une expérience très riche qui m'a permis de renforcer mes compétences d'une part dans le domaine de développement mobile et d'autre part en soft skills (travail en équipe, communication, collaboration ...).

Comme tout projet, certaines circonstances étaient défavorables pour le bon déroulement de travail. Par exemple je n'avais pas un accès complet à la base des données ce qui a augmenté la complexité du travail demandé.

Ce projet a satisfait le cahier des charges de départ. Néanmoins, ce système peut être amélioré au niveau des axes suivants :

- Une Mise à niveau côté design.
- L'intégration d'un outil permettant de digitaliser les consultations des patients.

Conclusion générale

- L'ajout d'un chat bot intégré au niveau du système

Pour conclure, je souhaite que ce modeste travail apporte la satisfaction aux responsables de la société Zedney Creative et aux membres de Jury.

Bibliographies

- [A] <https://www.marine-guyot.ovh/methode-agile-ou-classique/>
- [B] <https://medium.com/jorgeacetozi/agile-and-scrum-overview-fb68fc04aff8>
- [C] <https://reinvently.com/blog/fundamentals-web-application-architecture/>
- [D] <https://resources.scrumalliance.org/>
- [E] <https://fr.slideshare.net/TechnetFrance/alm210>
- [F] <https://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93viewmodel>
- [G] https://fr.wikipedia.org/wiki/Visual_Studio_Code
- [H] https://en.wikipedia.org/wiki/Android_Studio
- [I] <https://www.postman.com/company/about-postman/>
- [J] <https://www.bitrix24.com/>
- [K] [https://fr.wikipedia.org/wiki/Ionic_\(framework\)](https://fr.wikipedia.org/wiki/Ionic_(framework))
- [L] <https://fr.wikipedia.org/wiki/Node.js>
- [M] [https://en.wikipedia.org/wiki/Python_\(programming_language\)](https://en.wikipedia.org/wiki/Python_(programming_language))
- [N] [https://en.wikipedia.org/wiki/Angular_\(web_framework\)](https://en.wikipedia.org/wiki/Angular_(web_framework))

Remarque :

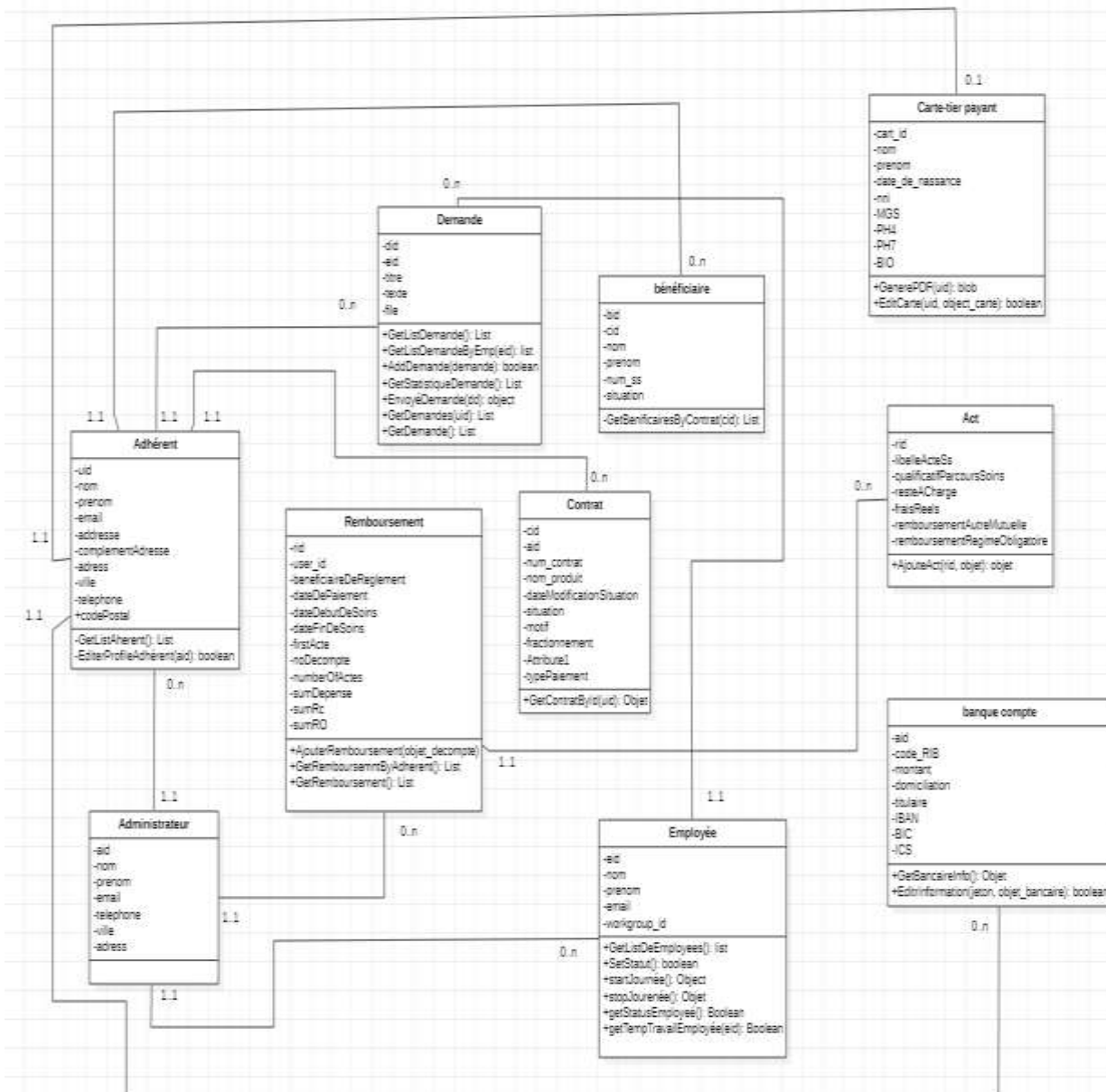
Durant mon auto formation qui s'est déroulé au sein de la société Zedney creative j'ai bénéficié des plusieurs ressources durant sprint 0 je site quelque ressource ici :

- [A] <https://ionicframework.com/docs/angular/your-first-app>
- [B] <https://www.tutorialspoint.com/ionic/index.htm>
- [C] https://www.youtube.com/playlist?list=PLrnPJCHvNZuCfAe7QK2BoMPkv2TGM_b0E
- [D] <https://www.youtube.com/playlist?list=PLrnPJCHvNZuDihTpkRs6SpZhqgBqPU118>
- [E] <https://helpdesk.bitrix24.com/documentation.php>

[F] <https://cordova.apache.org/docs/en/10.x/guide/hybrid/plugins/>

Digramme de classe global

Figure 63 : Digramme de classe global



Annexe2: Autres interfaces graphiques

Figure 64 : Interface détail d'employée

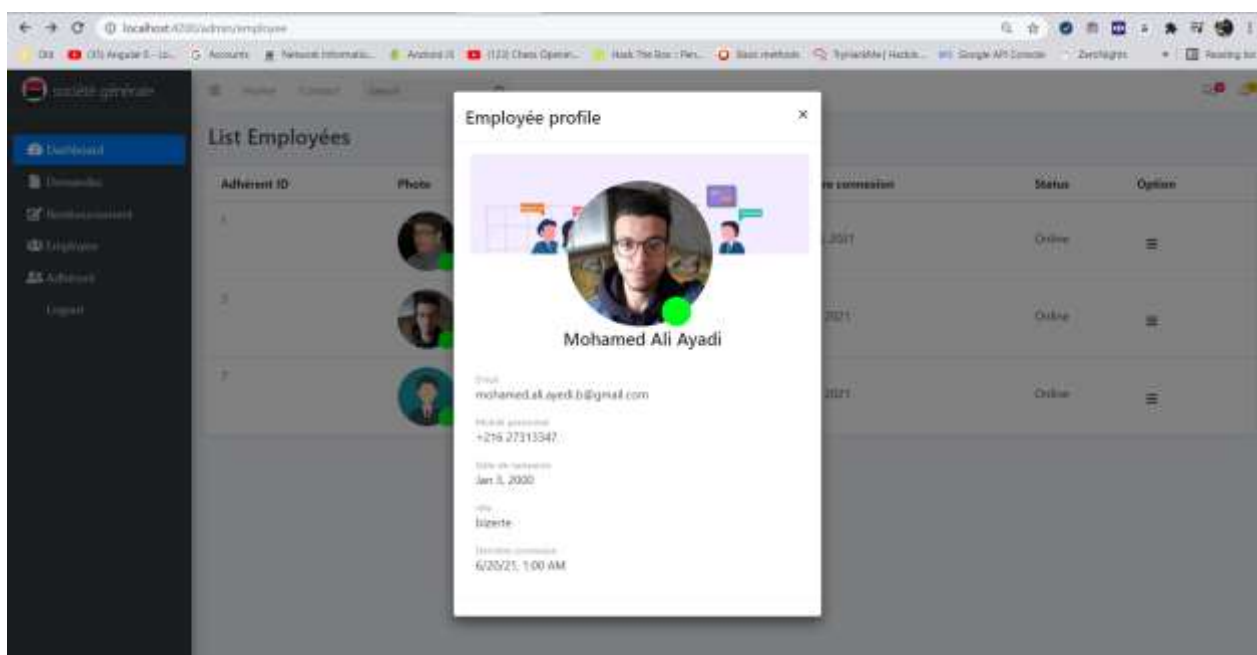


Figure 65 : Interface Dashboard pour l'administrateur

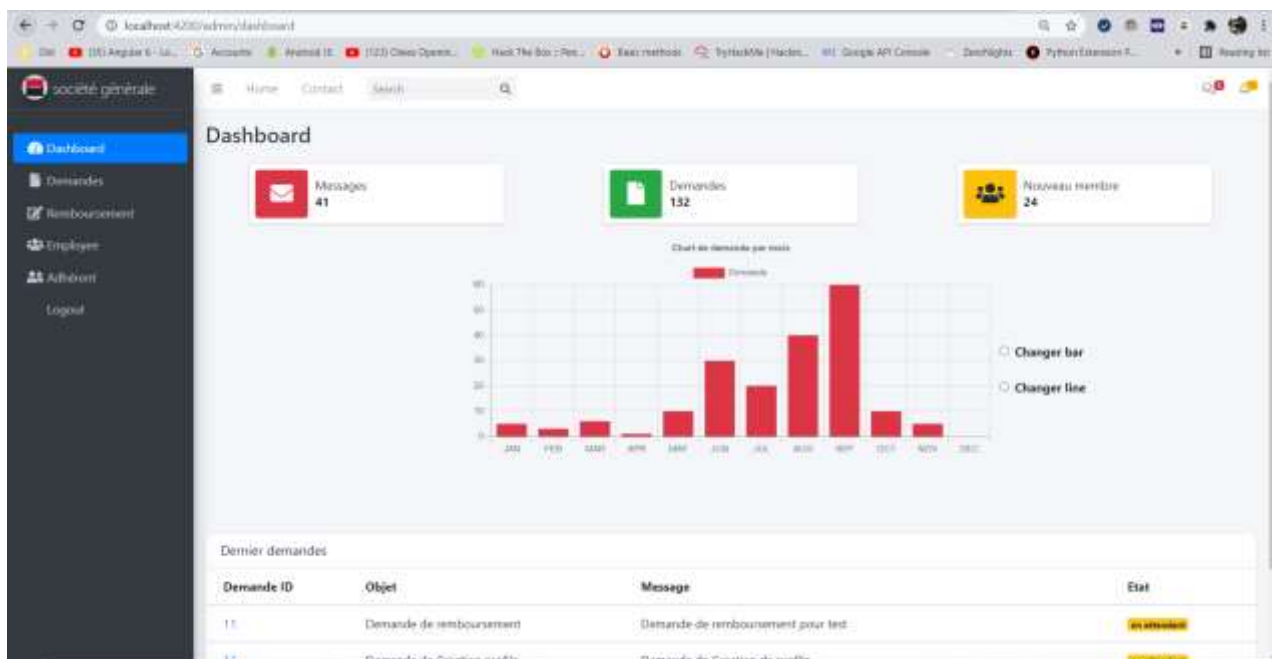


Figure 66 : Interface consulter les remboursements pour l'administrateur

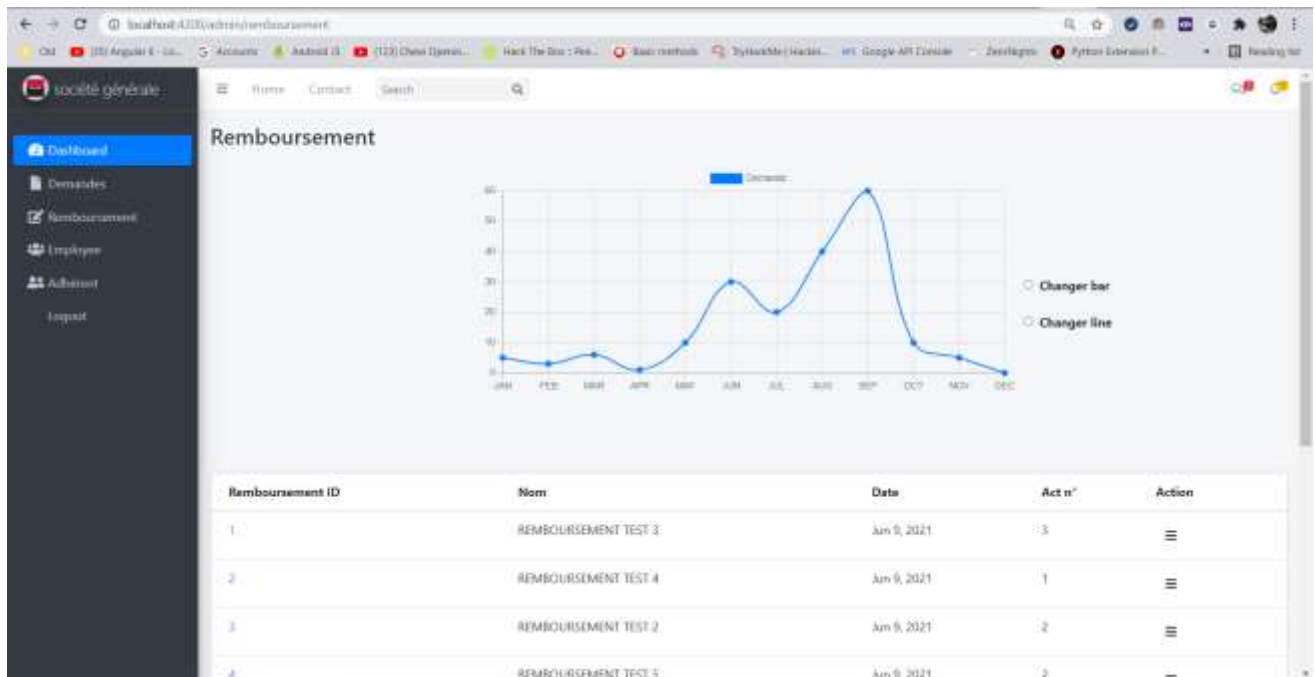


Figure 67 : Interface consulter les remboursements pour l'employée

Welcome

Remboursement

Remboursement	Date de fin	Actes	Status
REMBOURSEMENT TEST 3	Jun 9, 2021 (12:00 AM)	3	Open
REMBOURSEMENT TEST 4	Jun 9, 2021 (12:00 AM)	1	Open
REMBOURSEMENT TEST 2	Jun 9, 2021 (12:00 AM)	2	Open
REMBOURSEMENT TEST 5	Jun 9, 2021 (12:00 AM)	2	Open
REMBOURSEMENT TEST 6	Jun 9, 2021 (12:00 AM)	1	Open
REMBOURSEMENT TEST 7	Jun 9, 2021 (12:00 AM)	1	Open
REMBOURSEMENT TEST 8	Jun 9, 2021 (12:00 AM)	2	Open
REMBOURSEMENT TEST 1	Jun 9, 2021 (12:00 AM)	3	Open

Figure 68 : Interface consulter détail de remboursement



Figure 69 : Interface activation GPS



