|  |  |  |
| --- | --- | --- |
| **Final_Logo_Mobil-new** | **REPUBLIQUE TUNISIENNE**  **Ministère de l’Enseignement Supérieur, de La Recherche Scientifique**  **Direction Générale des Etudes Technologiques**  **Directions Des ISETs**  **Institut Supérieur des Etudes Technologiques de Charguia** | Résultat de recherche d'images pour "logo iset charguia" |

**Rapport du mini projet**

|  |
| --- |
| **Conception et développement d’une application Web et Mobile de Gestion de Feedback** |

|  |  |
| --- | --- |
| **Réalisé par:**   * **Scrum Master :**   **Hamza Ktata**   * **Membres :**   **Yacine Blidi**  **Moez Abid**  **Nourchène Senan**  **Mehdi Dabboussi**  **Rabii ben tili** | **Filière :**  DSI 301  **Encadré par :**  Mohamed Khalloufi |

**Organisme d’accueil :**

Mintit

**Adresse :**

08 Rue arabie saoudite 1002 Tunis

Sommaire

[Introduction générale 1](#_Toc500228533)

[Chapitre 1 : Cadre du projet 2](#_Toc500228534)

[Introduction 2](#_Toc500228535)

[I. Présentation de l’entreprise d’accueil 2](#_Toc500228536)

[II. La présentation du domaine métier 2](#_Toc500228537)

[III. L’étude de l’existant 3](#_Toc500228538)

[1. Description de l’existant 3](#_Toc500228539)

[2. Critique de l’existant 4](#_Toc500228540)

[3. Solution proposée 4](#_Toc500228541)

[IV. La méthodologie de conception et langage de modélisation 4](#_Toc500228542)

[1. Les méthodes agiles 4](#_Toc500228543)

[2. Les principales méthodes Agiles 4](#_Toc500228544)

[3. SCRUM 5](#_Toc500228545)

[4. UML (Unified Modeling Language) : 6](#_Toc500228546)

[Conclusion 6](#_Toc500228547)

[Chapitre 2 : Sprint 0 Spécification des besoins 7](#_Toc500228548)

[Introduction 7](#_Toc500228549)

[I. Spécification des besoins 7](#_Toc500228550)

[1. Les besoins fonctionnels 7](#_Toc500228551)

[2. Les besoins non fonctionnels 8](#_Toc500228552)

[II. Le Backlog du produit 8](#_Toc500228553)

[III. Le Diagramme de cas d’utilisation général 11](#_Toc500228554)

[1. Présentation des acteurs : 11](#_Toc500228555)

[IV. L’environnement de développement et choix technique 12](#_Toc500228556)

[V. L’architecture du système 17](#_Toc500228557)

[VI. Planification de la release 18](#_Toc500228558)

[Conclusion 18](#_Toc500228559)

[Chapitre 3 : Sprint 1 19](#_Toc500228560)

[Introduction 19](#_Toc500228561)

[I. Spécifications Fonctionnelles 19](#_Toc500228562)

[1. Le Backlog du Sprint 1 19](#_Toc500228563)

[2. Présentation des User Stories 20](#_Toc500228564)

[a. Le prototype relatif au User Story « S’authentifier » 20](#_Toc500228565)

[b. Le prototype relatif au user story « En tant qu’utilisateur je veux m’inscrire » 21](#_Toc500228566)

[c. Le prototype relatif au user story « Télécharger l’application » 22](#_Toc500228567)

[d. Le prototype relatif au user story « En tant que utilisateur je veux consulter la liste des applications » 23](#_Toc500228568)

[e. Le prototype relatif au user story «S’authentifier » 24](#_Toc500228569)

[f. Le prototype relatif au user story « Ajouter une application » 25](#_Toc500228570)

[II. Conception 26](#_Toc500228571)

[3. Diagramme de classes 26](#_Toc500228572)

[4. Diagramme de séquence détaillé 27](#_Toc500228573)

[III. Réalisation 28](#_Toc500228574)

[IV. La Revue du Sprint 30](#_Toc500228575)

[1. Burndown Chart 30](#_Toc500228576)

[2. Calcul de la vélocité 31](#_Toc500228577)

[3. Planification de la release 31](#_Toc500228578)

[Conclusion 31](#_Toc500228579)

[Chapitre 4 : Sprint 2 32](#_Toc500228580)

[Introduction 32](#_Toc500228581)

[I. Spécifications fonctionnelles 32](#_Toc500228582)

[1. Le Backlog du Sprint 2 32](#_Toc500228583)

[a. Le prototype relatif au user story « Description d’une application » 32](#_Toc500228584)

[b. Le prototype relatif au user story « Formulaire Feedback » 34](#_Toc500228585)

[c. Le prototype relatif au user story « En tant que utilisateur je veux convertir mes points» 34](#_Toc500228586)

[d. Le prototype relatif au user story « En tant que administrateur je veux consulter la liste des applications» 36](#_Toc500228587)

[e. Le prototype relatif au user story « En tant qu’administrateur je veux consulter les utilisateurs » 37](#_Toc500228588)

[f. Le prototype relatif au user story «En tant qu’administrateur je veux consulter la liste des Feedbacks » 38](#_Toc500228589)

[g. Le prototype relatif au user story « Modifier application » 39](#_Toc500228590)

[II. Conception 40](#_Toc500228591)

[1. Diagramme de classes 40](#_Toc500228592)

[2. Diagramme de séquence détaillé 41](#_Toc500228593)

[III. Réalisation 42](#_Toc500228594)

[IV. La Revue du Sprint 45](#_Toc500228595)

[1. Burndown Chart 45](#_Toc500228596)

[2. Calcul de la vélocité 46](#_Toc500228597)

[3. Planification de la release 46](#_Toc500228598)

[Conclusion 46](#_Toc500228599)

[Chapitre 5 : Réalisation du Sprint 3 47](#_Toc500228600)

[Introduction 47](#_Toc500228601)

[I. Spécifications fonctionnelles 47](#_Toc500228602)

[1. Le Backlog du Sprint 3 47](#_Toc500228603)

[2. Présentation des User Stories 47](#_Toc500228604)

[a. Le prototype relatif au User Story « En tant que utilisateur je veux modifier mon profil» 47](#_Toc500228605)

[b. Le prototype relatif au user story « En tant que utilisateur je veux rechercher une application précise » 49](#_Toc500228606)

[c. Le prototype relatif au user story «En tant que administrateur je veux supprimer des applications» 51](#_Toc500228607)

[d. Le prototype relatif au user story « En tant que administrateur je veux avoir le suivi des payements» 52](#_Toc500228608)

[Conclusion 56](#_Toc500228609)

[Conclusion générale 57](#_Toc500228610)

[Nétographie 58](#_Toc500228611)

**Liste des figures**

[Figure 1 : Diagramme du Workflow de l’existant. 3](#_Toc500188160)

[Figure 2 : Processus de SCRUM [3]. 5](#_Toc500188161)

[Figure 3 : Diagramme de cas d’utilisation générale 11](#_Toc500188162)

[Figure 4 : Diagramme circulaire de comparaison des systèmes d’exploitation mobile 12](#_Toc500188163)

[Figure 5 : Architecture Firebase 17](#_Toc500188164)

[Figure 6 : Prototype de la page « S’authentifier » 20](#_Toc500188165)

[Figure 7 : Maquette « Inscription » 21](#_Toc500188166)

[Figure 8 : Maquette « Télécharger l’application » 22](#_Toc500188167)

[Figure 9 : Maquette « Liste des applications » 23](#_Toc500188168)

[Figure 10 : Maquette « S’authentifier » 24](#_Toc500188169)

[Figure 11 : Maquette « Ajouter une application » 25](#_Toc500188170)

[Figure 12 : Diagramme de classes 26](#_Toc500188171)

[Figure 13 : Diagramme de séquence détaillé 27](#_Toc500188172)

[Figure 14 : Interface « Inscription » 28](#_Toc500188173)

[Figure 15 : Interface « Authentification utilisateur » 29](#_Toc500188174)

[Figure 16 : Interface « Consultation la liste des applications développées » 29](#_Toc500188175)

[Figure 17 : Interface « Authentification admin » 30](#_Toc500188176)

[Figure 18 : Interface « Ajout application » 30](#_Toc500188177)

[Figure 19 : Burndown Chart du Sprint 1 31](#_Toc500188178)

[Figure 20 : Backlog du Sprint 2 32](#_Toc500188179)

[Figure 21 : Maquette « Description d’une application » 33](#_Toc500188180)

[Figure 22 : Maquette « Formulaire FeedBack » 34](#_Toc500188181)

[Figure 23 : Maquette « Conversion points » 35](#_Toc500188182)

[Figure 24 : Maquette « Consulter application » 36](#_Toc500188183)

[Figure 25 : Maquette « Consulter les utilisateurs » 37](#_Toc500188184)

[Figure 26 : Maquette « Consulter Feedback » 38](#_Toc500188185)

[Figure 27 : Maquette « Modifier application » 39](#_Toc500188186)

[Figure 28 : Diagramme de classes relatif au Sprint 2 40](#_Toc500188187)

[Figure 29 : Diagramme de séquence détaillé relatif au Sprint 2 41](#_Toc500188188)

[Figure 30 : Interface « Consulter description d’une application » 42](#_Toc500188189)

[Figure 31 : Interface « Remplir le feedback » 42](#_Toc500188190)

[Figure 32 : Interface « Conversion des points cumulés » 43](#_Toc500188191)

[Figure 33 : Interface « Consulter les feedback de chaque application » 43](file:///C:\Users\ASUS\Downloads\Rapport_Final_Moez_1.docx#_Toc500188192)

[Figure 34 : Interface « Consulter le cumul de points de chaque user » 44](#_Toc500188193)

[Figure 35 : Interface « Consulter application » 44](#_Toc500188194)

[Figure 36 : Interface « Modifier application » 45](#_Toc500188195)

[Figure 37 : Burdown Chart du Sprint 2 46](#_Toc500188196)

[Figure 38 : Prototype de l’interface «Modifier profil» 47](#_Toc500188197)

[Figure 39 : Prototype de l’interface «Caractéristiques de l’appareil » 48](#_Toc500188198)

[Figure 40 : Prototype de l’interface «Rechercher une application précise» 49](#_Toc500188199)

[Figure 41 : Prototype de l’interface «Supprimer Application» 50](#_Toc500188200)

[Figure 42 : Prototype de l’interface «Avoir le sui des payements» 51](#_Toc500188201)

[Figure 43 : Diagramme de classes relatif au Sprint 3 52](#_Toc500188202)

[Figure 44 : Interface « Consulter les coordonnées de la boîte » 53](#_Toc500188203)

[Figure 45 : Interface « Recherche une application » 53](#_Toc500188204)

[Figure 46 : Interface « Modifier mon profil » 54](#_Toc500188205)

[Figure 47 : Interface « Déconnexion » 54](#_Toc500188206)

[Figure 48 : Interface « Consulter Payements » 55](#_Toc500188207)

**Liste des tableaux**

[Tableau 1 : Tableau comparatif des environnements de développement mobile 13](#_Toc500188208)

[Tableau 2 : Tableau de description des logiciels similaires à Github présents sur le marché 14](#_Toc500188209)

[Tableau 3 : Planification de la release 18](#_Toc500188210)

[Tableau 4 : Backlog du sprint 1 19](#_Toc500188211)

# Introduction générale

Aujourd’hui nous vivons dans un monde où l’information est un facteur important dans nos vies, et nous pouvons maintenant accéder à ces informations très facilement grâce à un outil qui peut se trouver dans nos poches qui n’est d’autre que le smartphone.

Avec l’émergence du Smartphone et sa massification, la téléphonie mobile a profondément touché nos structures perceptives et notre manière d’être au quotidien.

Pour les entreprises et les marques, les applications mobiles sont devenues un moyen essentiel de création de nouveaux services à destination des mobinautes, c’est pour cela que la boîte de développement mobile Mint IT cherche à améliorer son image de marque en testant leurs applications sur une multitude de téléphones afin d’offrir des produits de meilleure qualité.

Le nombre des Smartphones qu'elle possède qui est au nombre de quinze n'est pas suffisant pour assurer la bonne qualité des applications, et avec une communauté aussi puissante qui se trouve sur Internet, la boîte de développement Mint IT a proposé à notre groupe de mini projet « House Targaryen» de développer l’application FeedUp qui permet aux utilisateurs d’évaluer les applications que la boite développe en donnant leurs avis ou en reportant un bug ou un problème quelconque, sous la forme d'un formulaire.

L’élaboration de ce rapport, a été réalisée en cinq chapitres, le premier chapitre est consacré à la présentation de l’organisme d’accueil, la présentation du projet et l’étude de l’existant.

Le deuxième chapitre est réservé à l’analyse des besoins fonctionnels et non fonctionnels du système, l’élaboration du backlog du produit en suivant la méthode agile SCRUM, l’architecture de notre système et le diagramme des cas d’utilisation global.

Et nous enchainerons ensuite avec les trois chapitres « Sprint 1 », « Sprint 2 » et le

« Sprint 3 » où nous détaillerons leurs spécifications fonctionnelles c'est-à-dire ; le Backlog du sprint, la présentation de ses user stories, leurs prototypages, leur conception et leur réalisation.

# Chapitre 1 : Cadre du projet

## Introduction

Dans ce chapitre,nous nous intéressons à une étude préalable du projet. Nous commençons par présenter l’organisation d’accueil et ses différents domaines d’activités. Ensuite, nous détaillons l’étude et la critique de l’existant, ainsi que les solutions proposées.

## Présentation de l’entreprise d’accueil

MINT IT est une société de services d'ingénierie informatique spécialisée dans le développement des applications mobiles.

Chez MINT IT, il y’a trois unités différentes :

* **Conseil:** aider les entreprises à transformer leurs idées commerciales en produits et solutions efficaces et rentables,
* **Mobile**: conception et livraison de produits et solutions basés sur des technologies de pointe.
* **UX / UI:** la combinaison d'UI innovantes avec UX attrayante est la compétence fondamentale de nos équipes UX / UI.

## La présentation du domaine métier

On parle aujourd’hui beaucoup du développement mobile ce qui définit le cadre de notre projet.

**Le métier du développeur d’applications mobiles** **:** Le développeur d'application est l'artiste et le créateur des applications pour mobiles, Smartphones et tablettes. Sa mission consiste dans un premier temps à créer techniquement une application mobile et à en assurer le développement informatique.

Il peut intervenir pour concevoir de A à Z une application ou pour améliorer celles qui existent déjà. Après avoir créé les programmes informatiques de l'application et avoir défini les algorithmes permettant le traitement des données, il effectue de nombreux tests de l'application en question pour s'assurer à la fois du respect du cahier des charges et du bon fonctionnement de sa création. Il veille à anticiper les évolutions prévisibles de l'application et des différents problèmes qui pourraient éventuellement survenir.

## L’étude de l’existant

## Description de l’existant

Actuellement, Mintit développe des applications mobiles à des entrepreneurs, des start-ups financées, des entreprises en croissance et à des grandes marques afin de satisfaire les besoins des utilisateurs ainsi que les attentes des clients.

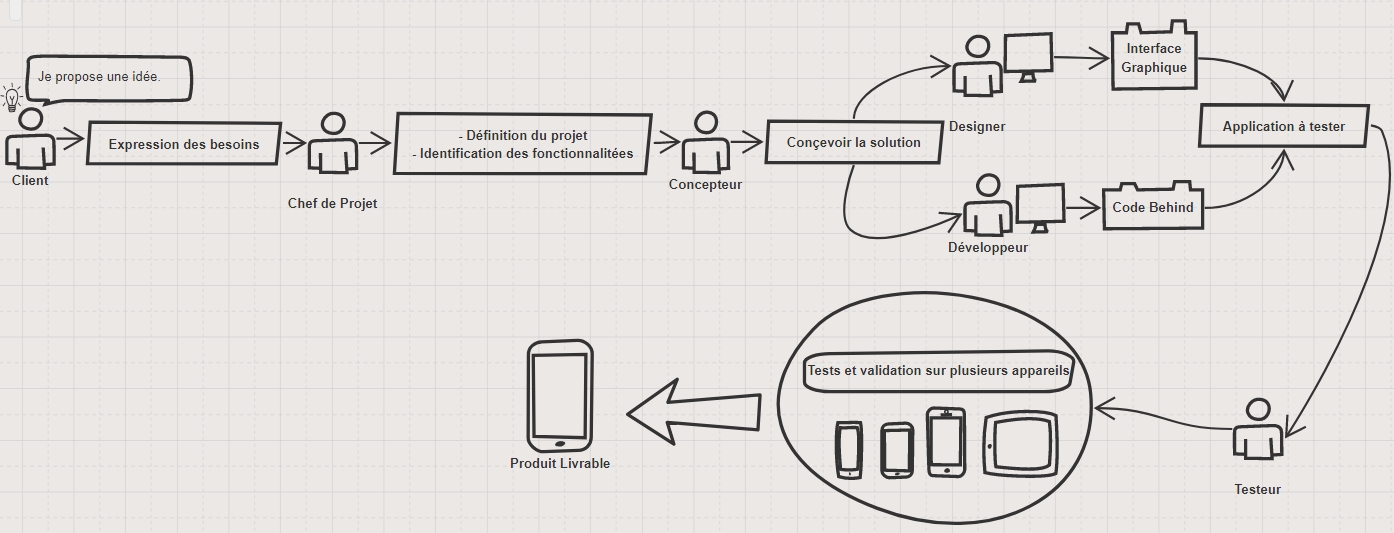


Figure 1 : Diagramme du Workflow de l’existant.

* Le client exprime ses besoins au chef de projet, et un cahier de charges est établi.
* Le chef du projet va définir le projet, identifier les fonctionnalités et les transmettre au concepteur. Il va aussi définir les objectifs du projet.
* Le concepteur va concevoir la solution, et la transmettre aux designers et aux développeurs afin de contribuer à l’ergonomie de l’application.
* Les designers et les développeurs vont travailler en équipe afin de produire chacun les différents composants de l’applicationqui seront intégrésdans une application qui va être transmise vers la phase de test.
* Le testeur va tester l’application sur les différents Smartphones de test disponibles dans la société, s’assurer de la validité de l’application et la livrer, sinon en cas de problème il va transmettre les résultats des tests aux designers et aux développeurs.

Les tests des applications sont gérés par l’équipe de support qui est très réactive au traitement des demandes de correction et d’optimisation et ceux issues des feedbacks des utilisateurs.

## Critique de l’existant

Pendant les phases de test généralement on se trouve devant des grosses difficultés : des problèmes techniques, des bugs, des erreurs de fiabilité, la seule et unique solution pour ces boites de développement est de tester ses applications sur des appareils différents.

Le nombre des appareils que la boite possède qui est au nombre de quinze n'est pas suffisant pour assurer la bonne qualité des applications.

## Solution proposée

Il s’agit de développer deux applications uneapplication mobile qui permet aux internautes de tester les applications que la boîte MintIt développe et de donner leurs Feedbacks, de reporter les bugs qu’ils rencontrent, et de recevoir des récompensessous forme de points qui peuvent être convertis en argentpour chaque Feedback qu’ils donnent.

Et une application Web qui permet à l’administrateur d’ajouter les applications que la boîte a développé ainsi que consulter les feedback afin de remédier aux problèmes rencontrés.

## La méthodologie de conception et langage de modélisation

Chaque personne peut songer à une idée, peut résoudre un problème et trouver des solutions pour faciliter n’importe tache, mais, concrétiser cette idée est le problème majeur. C’est pour cela qu’on doit opter pour les solutions les plus optimales pour avoir recours à une méthodologie efficace qui permet de gérer un cycle de vie d’un projet.

## Les méthodes agiles

Les méthodes agiles se proclament comme une méthodologie efficace qui représente une approche de gestion de projet et de développement itérative et incrémentale et qui vise à réduire le cycle de vie du logiciel et permet de satisfaire les besoins du client. [1] Cette méthode se base sur des cycles courts et permet de découper le projet en petits blocs et les hiérarchiser en fonction des besoins. On fera une étude plus approfondie dans l’annexe de ce rapport à propos les méthodes agiles.

## Les principales méthodes Agiles

* SCRUM
* EXtremeProgramming (XP)
* Rapid Application Development (RAD)
* Crystal clear

Après avoir fait beaucoup de recherches et avoir mené beaucoup d’interviews avec des parties qui travaillent avec les méthodes agiles, on a opté pour l’utilisation de la méthode SCRUM.

## SCRUM

La méthode SCRUM est une méthode agile, créée en 2002, dont le nom est un terme emprunté au rugby qui signifie « la mêlée » qui permet de produire un logiciel dans la durée la plus courte. [2]

Ce processus s'articule en effet autour d'une équipe soudée, qui cherche à atteindre un but. Cette méthodologie se progresse par une série d'itérations appelées sprints, qui durent de deux à quatre semaines. Le produit envisagé est conçu, codé et testé pendant ce sprint. A chaque fin de sprint, on peut voir fonctionner le produit courant et décider soit de le livrer, soit de continuer à l'améliorer pendant un autre sprint.

Voici le processus sur lequel est basé SCRUM :



Figure 2 : Processus de SCRUM [3].

* **Le « Product Backlog» :** C’est l'ensemble des fonctionnalités du produit que l'on veut développer.
* **Le « Sprint Backlog» :** C’est une liste de tâches identifiées par l'équipe à remplir pendant un sprint.
* **« Burndowncharts » :** Met en relief le taux de travail restant dans un sprint.

Ce sont les principaux artéfacts qu’on peut les générer lors de l’utilisation de la méthode SCRUM. Durant un développement d’un projet avec la méthode SCRUMil y a une interaction avec plusieurs intervenants :

* **Le « Product Owner »** qui porte la vision du produit à réaliser.
* **Le « SCRUM Master »** c’est une personne chargée de veiller à la mise en application de la méthode et au respect de ses objectifs.

L’équipe de développement qui réalise le produit. La durée de vie d’un projet en SCRUM est rythmée par un ensemble de réunions clairement définies et strictement limitées dans le temps.

## UML (Unified Modeling Language) :

Une dizaine d'années après le début de son utilisation dans le cadre de projets de développement orienté objet, UML s'est imposé comme standard. Ce langage est né de la fusion de plusieurs méthodes existant auparavant et est devenu désormais la référence en termes de modélisation objet.

La modélisation objet consiste à créer une représentation informatique des éléments du monde réel auxquels on s'intéresse, sans se préoccuper de l'implémentation, ce qui signifie indépendamment d'un langage de programmation. Il s'agit donc de déterminer les objets présents et d'isoler leurs données et les fonctions qui les utilisent. [4] Après le choix de la méthodologie, on a choisi UML comme un langage de modélisation afin de concevoir le projet.

## Conclusion

Dans ce chapitre, nous avons exposé le contexte du sujet de notre projet et présenté l'entreprise qui nous a proposé ce sujet, et nous avons réalisé une étude de l’existant. Le prochain chapitre sera consacré à la spécification des besoins.

# Chapitre 2 : Sprint 0 Spécification des besoins

## Introduction

Dans ce chapitre, la première phase de Scrum qui est le Sprint 0 sera présentée. Il s’agit en fait d’une phase de planification et architecture qui ne se termine pas nécessairement par une livraison. Les travaux réalisés dans cette période conduits à construire une bonne vision du produit, identifier les rôles des utilisateurs, dégager les fonctionnalités principales, préparer l’environnement de développement et déterminer un plan de Release afin de produire le Backlog du produit initial ainsi qu'une première planification des sprints.

## Spécification des besoins

A ce niveau de notre travail, nous nous sommes engagés à présenter les besoins fonctionnels et non-fonctionnels identifiés suite à une présélection préalable des besoins.

## Les besoins fonctionnels

Nous sommes amenées à réaliser un site Web et une application mobile qui répondent aux exigences de la société.

* **L’utilisateur**:
* L’utilisateurpeut s’authentifier s’il possède déjà un compte.
* L’utilisateurpeuts’inscrire en remplissant un formulaire contenant des ses informations personnelles.
* L’utilisateurpeut consulter la liste des applications développées par la boite ainsi que sa description.
* L’utilisateurpeut modifier les informations de son compte.
* Le clientreçoitune notification à chaque application ajoutée par la boite sur son appareil.
* L’utilisateurpeut consulterlescaractéristiques de son appareil.
* L’utilisateurpeut consulter les coordonnés et les informationsde la boite (Mint IT)
* L’utilisateurpeut se déconnecter.
* L’utilisateurpeut demander la conversion des points accumulés en argent à partir d’un certain nombre de points.
* Chaque user peut télécharger l’application choisie.
* L’utilisateurpeut remplirun feedback sur l’application choisie si et seulement si cette dernière est installée sur l’appareil du client.
* **Administrateur :**
* L’administrateurpeut s’authentifier en introduisant un login et un mot de passe.
* L’administrateurpeut consulter la liste des Feedback de chaque application.
* L’administrateurpeut gérer la liste des applications c’est-à-dire y ajouter, modifier et supprimer.
* L’administrateurpeut voir le cumul des points de chaque client et suivre ses paiements.

## Les besoins non fonctionnels

* + **Performance**
    - Décrivent les performances d’exécution du système, généralement en termes de temps de réponse.

Exemple : (L’application mobile) Le temps de chargement d’une page : Le chargement d’une page mobile ne devrait pas prendre plus de 15 secondes en conditions normales.

* + **Disponibilité/Fiabilité**
    - Concernent le niveau de disponibilité qui doit être explicitement défini pour les applications critiques

Exemple : exigence de disponibilité 24/24, 7/7 sauf période de maintenance (à spécifier…)

* + **Matériels**
    - Définissent les configurations matérielles minimales nécessaires au fonctionnement du système.
  + **Facilité d'utilisation**
    - La plate-forme doit être très facile à comprendre et à utiliser grâce à des interfaces utilisateur simples et à une expérience utilisateur exceptionnelle.

## Le Backlog du produit

Avant même de démarrer le premier Sprint, nous avons besoin d’un Backlog du produit, c’est à dire une liste priorisée de caractéristiques orientées client. Le backlog du produit existe tout au long de la vie du produit. C’est la feuille de route du produit.

Tout au long du projet, le Backlog du produit centralise la liste de « tout ce qui peut être fait par l’Equipe, par ordre de priorité ». Il n’existe qu’un seul Backlog pour un produit.

Pour estimer la complexité de nos user story, nous avons utilisé :

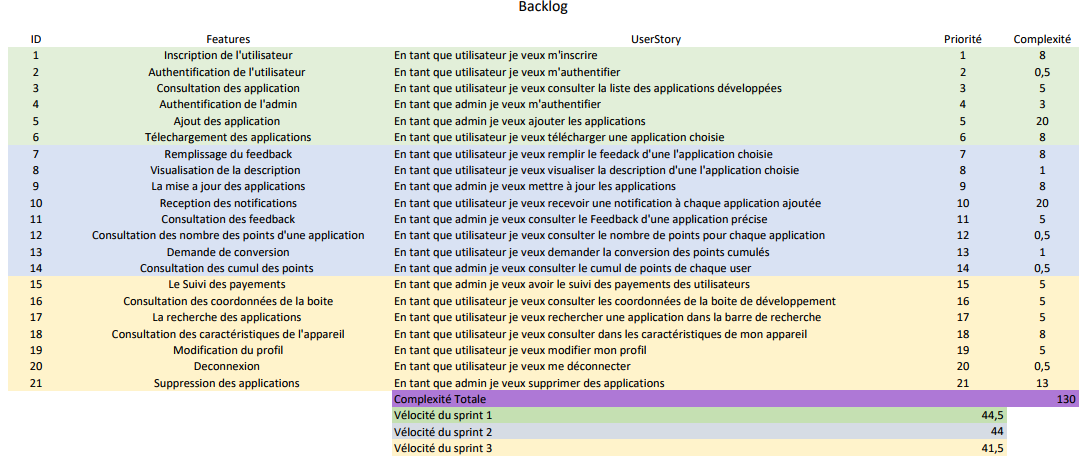
* **2** : Facile
* **3** : Simple
* **5 et 8** : Normale
* **13** : Difficile
* **20 et 40** : Très dure.

Notre Backlog est composé de **21** User Stories. On a divisé notre Backlog en 3 Sprints :

**La partie en vert :**représente le sprint 1 qui est composé de 6 user stories.

**La partie en violet :**représente le sprint 2 qui est composé de 8 user stories.

**La partie jaune :**représente le sprint 3 qui est composé de 7 user stories.

C:\Users\Moez\Desktop\Figure 3.bmp

## Le Diagramme de cas d’utilisation général

Dans cette section, nous présentons les besoins de notre système de manière formelle en utilisant le diagramme des cas d’utilisation du langage de modélisation UML. Ce diagramme représente les activités globales de chaque utilisateur dans notre application.

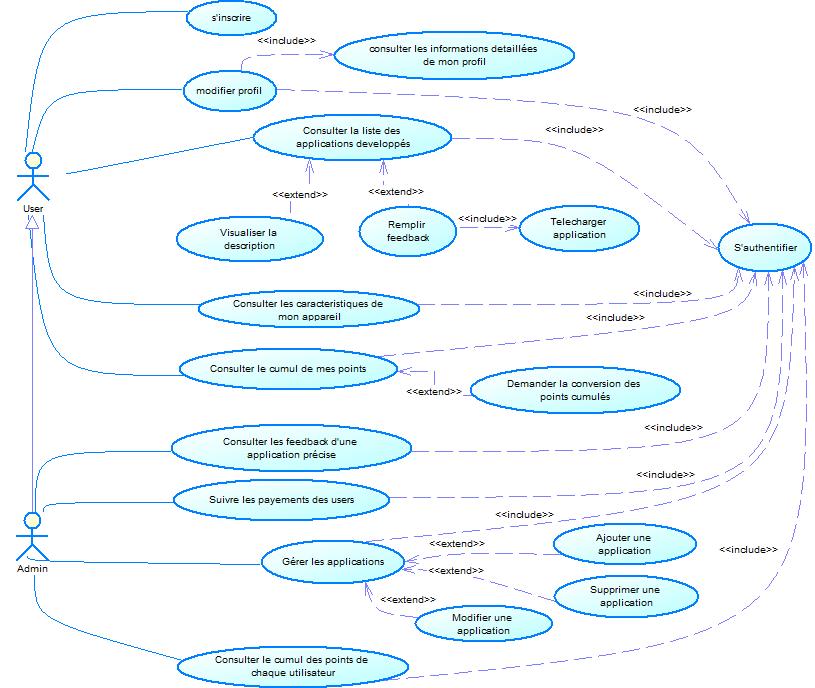
****

Figure 3 : Diagramme de cas d’utilisation générale

## Présentation des acteurs :

Un acteur est une entité externe qui interagit avec le système. Dans notre projet les acteurs sollicitant le système sont :

**L’utilisateur :** qui a pour rôled’évaluer l’application en donnant des feedbacks.

**L’administrateur :**qui a pour rôle de gérer les applicationset consulter les feedbacks des users.

## L’environnement de développement et choix technique

* **Android** **:**Il règne sur 65.53% des Smartphones sur le marché car c'est un système basé sur le noyau LINUX, Open Source qui permet de faciliter la phase de développement d'applications mobiles étendue sur un marché très diversifié.

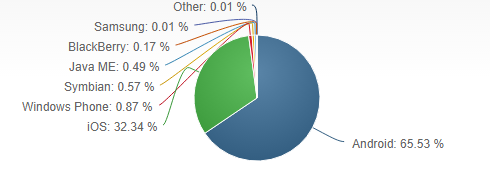


Figure 4 : Diagramme circulaire de comparaison des systèmes d’exploitation mobile

On a choisi Android :

* Tout d’abord, au niveau des outils :

Une application est développée à l'aide du SDK android et de l'IDE eclipse avec le plugin crée par google. À ce niveau-là, le développeur n'a pas besoin de débourser d'argent.

* Une application android est codée à l’aide de deux langages :

Tout d'abord, l'interface se développe en XML et le reste de l'appli en java. Deux langages déjà très populaire et dont les notions sont utiles pas seulement pour Android.

* Aucun OS spécifique ne requit :

Étant donné qu'eclipse est codé en java, cette IDE fonctionne sur tous les OS qui supportent une machine virtuelle java (Qui est présent sous Windows, Mac OSX, ainsi que la plupart des distributions Linux).

* **Pourquoi Android Studio ?**

Android Studio car c'est l'IDE recommandé par le propriétaire d’Android, et c'est un IDE basé sur IntelliJ. Il Permet l'intégration facile de plug-ins et d'extensions.

Les points forts d’Android Studio :

* Il offre un émulateur rapide et facile à utiliser.
* L’intégration Gradle : Android Studio utilise le système de build Gradle qui est très utile.
* L'option "Advanced Code Complétion » qui facilite et accélère la phase d'écriture de code.
* La possibilité faire le design facilement avec le built-in Designer de l'IDE
* L'organisation bien établie du projet
* L'option Drag-and-Drop.

Voici une figure qui fait une comparaison entre les différents environnements de développement mobile :

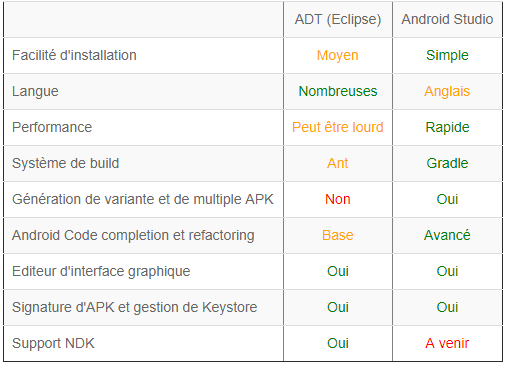


Tableau 1 : Tableau comparatif des environnements de développement mobile

* **GitHub :**

Github est un systéme de contrôle de versions, et c’est un outil extrêmement utile pour avancer sur ses projets et communiquer avec les autres développeurs. Mais surtout, désormais Github est le book des développeurs. Il permet de mettre en avant la qualité de son code et ainsi montrer ses capacités et sa plus-value lorsque l’on recherche un emploi. Considéré comme un véritable réseau social, il permet aussi aux développeurs de contribuer à des projets open source.

Il existe aussi d’autres logiciels de même type que Github, voici quelques-uns ainsi que leur type d’architecture et leurs descriptions :

|  |  |  |  |
| --- | --- | --- | --- |
| **Outil** | **Type** | **Description i** | **Projets qui l’utilisent** |
| [Mercurial](https://www.mercurial-scm.org/) | Distribué | Plus récent, il est complet et puissant. Il est apparu quelques jours après le début du développement de Git et est d’ailleurs comparable à ce dernier sur bien des aspects. Vous trouverez un [tutoriel sur Mercurial](http://www.siteduzero.com/tutoriel-3-196002-gerer-vos-projets-avec-mercurial.html) sur OpenClassrooms. | Mozilla, Python, OpenOffice.org… |
| [Bazaar](http://bazaar.canonical.com/en/) | Distribué | Un autre outil, complet et récent, comme Mercurial. Il est sponsorisé par Canonical, l’entreprise qui édite Ubuntu. Il se focalise sur la facilité d’utilisation et la flexibilité. | Ubuntu, MySQL, Inkscape… |
| [Git](https://git-scm.com/) | Distribué | Très puissant et récent, il a été créé par Linus Torvalds, qui est entre autres l’homme à l’origine de Linux. Il se distingue par sa rapidité et sa gestion des branches qui permettent de développer en parallèle de nouvelles fonctionnalités. | Kernel de Linux, Debian, VLC, Android, Gnome, Qt… |

Tableau 2 : Tableau de description des logiciels similaires à Github présents sur le marché

**Pourquoi Github :**

* Est très rapide ;
* Sait travailler par branches (versions parallèles d’un même projet) de façon très flexible ;
* C’est également valable pour les autres outils ;
* Une interface graphique simplifiée.
* Il est donc à réservé aux développeurs ayant un minimum d’expérience.
* **JAVA :**

**Caractéristiques du langage java :**Java possède un certain nombre de caractéristiques qui ont largement contribué à son énorme succès :

* **Simple** : C’est un langage simple à prendre en main. Basé sur le langage C/C++ mais laisse de coté les sources de problèmes (pointeurs, structures, gestion de la mémoire, héritage multiple, macros etc.). Ö en aucun cas limité.
* **Orienté objet**: Java est un langage purement orienté objet. Tout est classe. Héritage simple. Une librairie plus de classes est fournie.
* **Distribué :** Propose une API réseau standard. Cette dernière permet de manipuler, par exemple, les protocoles HTTP & FTP avec aisance. Des API pour la communication entre des objets distribués (Remote Method Invocation).
* **Interprété** : Un code source doit être traduit dans le langage machine avant d’être exécuté. Compilateur : traduction du code source dans le langage binaire de la machine sur laquelle il sera exécuté.
* **Portable :** Portable d’un système à un autre : Int 32 bits alors qu’en C/C++ 16 ou 32 bits.
* **Robuste**: Pas de pointeurs. Gestion de mémoire indépendante. Mécanisme d’exceptions pour la gestion des erreurs. Compilateur très contraignant. Pas d’héritage multiple ni surcharge des opérateurs.
* **Sécurisé** :
* Langage et son compilateur contraignant.
* Vérifier : vérification du byte code.
* Class Loader : le chargeur de classe.
* Security Manager : protection des fichiers et accès au réseau.
  + **JavaScript**

JavaScript est un langage universel utilisé dans le développement des applications web et mobiles actuelles, le JavaScript peut à la fois être utilisé côté client, c’est-à-dire interprété par le navigateur web de l’internaute, et côté serveur avec l’utilisation de Node.js. Le JavaScript est le langage de prédilection pour interagir avec le HTML permettant ainsi d’apporter du dynamisme à l’intérieur des pages web.

**Pourquoi JavaScript ?**

JavaScript permet la conception d’applications et sites web dynamiques interactifs fonctionnant sur tous les navigateurs web que ce soit Google Chrome, Mozilla Firefox ou encore Safari. De plus, la communauté très active et les nombreux outils et modules disponibles pour utiliser ce langage facilite son utilisation et lui permet d’innover constamment.

* + **Firebase**

Firebase est un ensemble de services d'hébergement pour n'importe quel type d'application. Il propose d'héberger en NoSQL et en temps réel des bases de données, du contenu, de l'authentification.

**Pourquoi Firebase**

Firebase inclut des API intuitives regroupées dans un SDK unique et permet de gagner du temps en limitant le nombre d'intégrations dans notre application. Nousbénéficions ainsi d'un service à la carte et d'une intégration étroite entre les différents produits que nous utilisons.

Firebasebénéficie des avantages de l'infrastructure de Google et s'adapte automatiquement à mesure que notre application évolue. Avec des solutions à chaque étape du développement et de la croissance, nous pouvons développer notre application en toute confiance et bénéficier d'une offre évolutive et complète proposée par Google.

**Principaux avantages de Firebase :**

* **Evolution rapide :**

Firebase inclut des API intuitives regroupées dans un SDK unique pour nous aider à développer rapidement des applications de grande qualité. Il propose également des outils pour attirer des utilisateurs et augmenter nos revenus. Il vous suffit de combiner les fonctionnalités Firebase adaptées à nos besoins.

* **Plus besoin d’avoir recours aux infrastructures :**

Plus besoin de disposer d'infrastructures complexes en interne ni de gérer de multiples tableaux de bord : Firebase prépare le terrain pour que nous pouvons offrir aux clients les fonctionnalités qu'ils recherchent.

## L’architecture du système

L’architecture de notre application est une architecture deux tiers client-serveur. En d’autres termes, là où les applications mobiles, notamment s’appuient généralement sur une base de données et une couche de traitement,Firebase se propose en remplacement. Comme l’explique Andrew Lee qui est le co-fondateur et directeur technique de Firebase, « Les applications intègrent de plus en plus de code natif sur le terminal alors que, traditionnellement, avec les applications Web, le code se trouvait largement sur le serveur ». Avec Firebase, « tout le code est exécuté sur le client » et les données sont côté serveur.

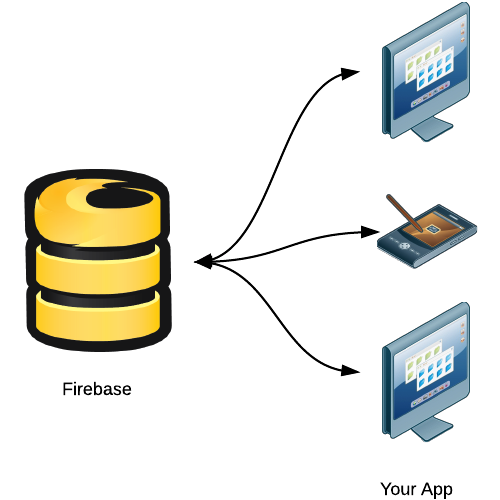


Figure 5 : Architecture Firebase

## Planification de la release

|  |  |  |
| --- | --- | --- |
| Sprint 1  09/10 jusqu'a 28/10 | Sprint 2  30/10 jusqu'a 17/11 | Sprint 3  20/11 jusqu'a 08/12 |
| US Nº 1  US Nº2  US Nº3  US Nº4  US Nº5  US Nº6 | US Nº7  US Nº8  US Nº9  US Nº10  US Nº11  US Nº12  US Nº13  US Nº14 | US Nº15  US Nº16  US Nº17  US Nº18  US Nº19  US Nº20  US Nº21 |
| Estimation 44.5 | Estimation 44 | Estimation 41.5 |

Tableau 3 : Planification de la release

## Conclusion

Dans ce chapitre, nous avons préparé notre plan de travail. Nous avons défini les besoins fonctionnels de notre application, les rôles des utilisateurs. Dans le chapitre qui suit, nous allons présenter le premier sprint.

# Chapitre 3 : Sprint 1

## Introduction

Tout au long de ce chapitre, nous allons traiter les user stories de notre premier sprint pour produire un ensemble d’incréments potentiellement livrable.

## Spécifications Fonctionnelles

## bak 1Le Backlog du Sprint 1

Tableau 4 : Backlog du sprint 1

## Présentation des User Stories

## Le prototype relatif au User Story « S’authentifier »

Ce user story permet à l’utilisateur de s’authentifier.



Figure 6 : Prototype de la page « S’authentifier »

Le premier user story représente la première interface.

La priorité métier est égale à (1/2)

* Une complexité représentée par le chiffre 8 (normal)

Les Cas du succès et d’échec du user story « s’authentifier » :

* **Succès**

1. Message affiché « Adresse électronique et mot de passe valide ».
2. Message affiché « Accéder à l’application ».

* **Echec : Affichage des messages**

1. Ces champs sont obligatoires.
2. Veuillez entrer une adresse électronique valide.
3. Email et / ou mot de passe incorrect.

## Le prototype relatif au user story « En tant qu’utilisateur je veux m’inscrire »

Ce user story permet de créer un compte utilisateur.



Figure 7 : Maquette « Inscription »

La deuxième user story représente sur quoi on va commencer à travailler.

La priorité métier est égale à (01)

* Une complexité représentée par le chiffre 1/2 (facile)

Les Cas du succès et d’échec du user story « créer un compte » :

* **Succès**

1. Message affiché « Votre inscription est validée ! »

* **Echec : Affichage des messages**

1. Ces champs sont obligatoires.
2. Veuillez entrer une adresse électronique valide.
3. Le mot de passe doit être tapé deux fois et de longueurs supérieures ou égales à 6 caractères.
4. Le numéro de téléphone doit être valide.
5. Le numéro du RIB doit être valide.
6. Le numéro du CIN doit être valide.

## Le prototype relatif au user story « Télécharger l’application »

Ce user story permet de Télécharger l’application.

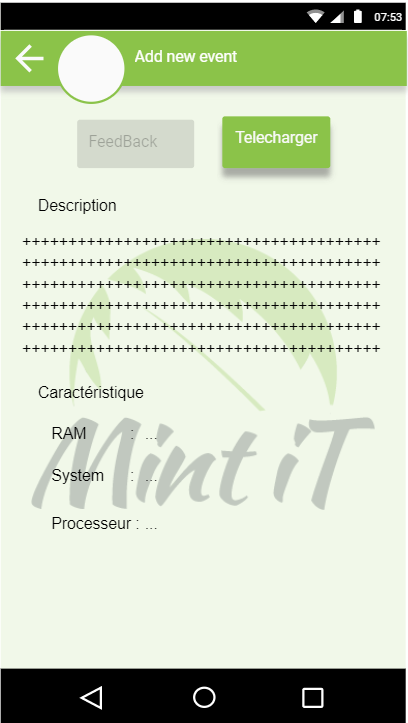


Figure 8 : Maquette « Télécharger l’application »

La troisième user story représente sur quoi on va commencer à travailler. La priorité métier est égale à (05).

* Une complexité représentée par le chiffre 8 (normal)

Les Cas du succès et d’échec du user story « Télécharger une application » :

* **Succès**

1. Message affiché « Télécharger l’application ».
2. Message affiché « Accéder à la page de feedback ».

* **Echec : Affichage des messages**

1. Application non compatible.
2. Vous devez teste l’application pour accéder au feedback.

## Le prototype relatif au user story « En tant que utilisateur je veux consulter la liste des applications »

Ce user story permet de consulter la Liste des applications.

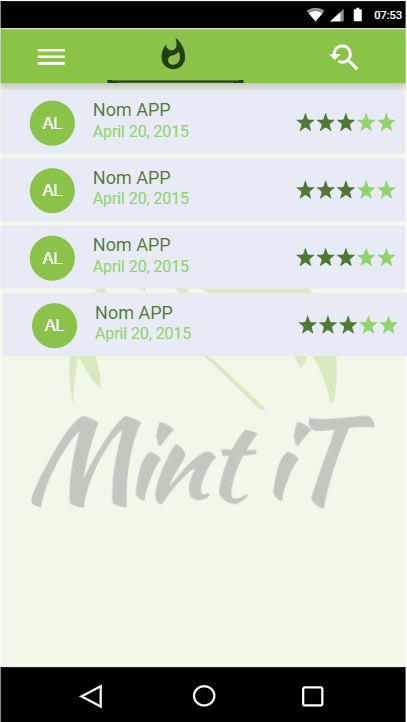


Figure 9 : Maquette « Liste des applications »

La septième user story représente sur quoi on va commencer à travailler. La priorité métier est égale à (07).

* Une complexité représentée par le chiffre 5 (facile)

Les Cas du succès et d’échec du user story « Liste des applications » :

* **Succès**

1. Toutes les applications sont affichées

* **Echec : Affichage des messages**

1. Aucune application disponible.

## Le prototype relatif au user story «S’authentifier »

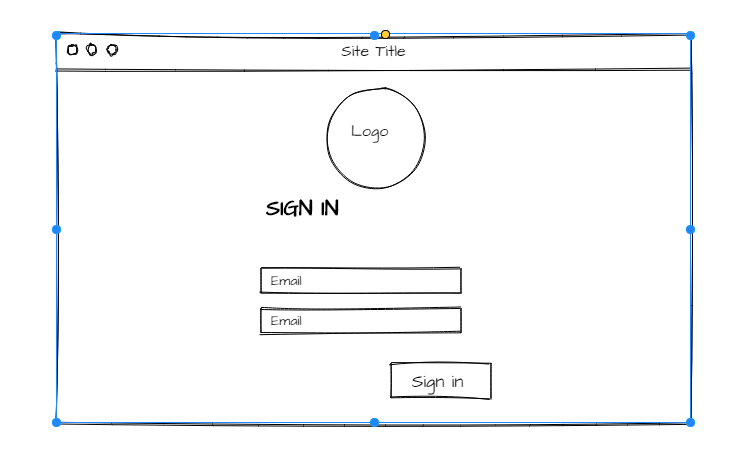


Figure 10 : Maquette « S’authentifier »

La troisième user story représente sur quoi on va commencer à travailler. La priorité métier est égale à (03)

* Une complexité représentée par le chiffre 3 (facile)

Les Cas du succès et d’échec du user story « S’authentifier » :

* **Succès**

1. Message affiché « Adresse électronique et mot de passe valide ».
2. Message affiché « Accéder à l’application ».

* **Echec : Affichage des messages**

1. Ces champs sont obligatoires.
2. Veuillez entrer une adresse électronique valide.
3. Email et / ou mot de passe incorrect.

## Le prototype relatif au user story « Ajouter une application »

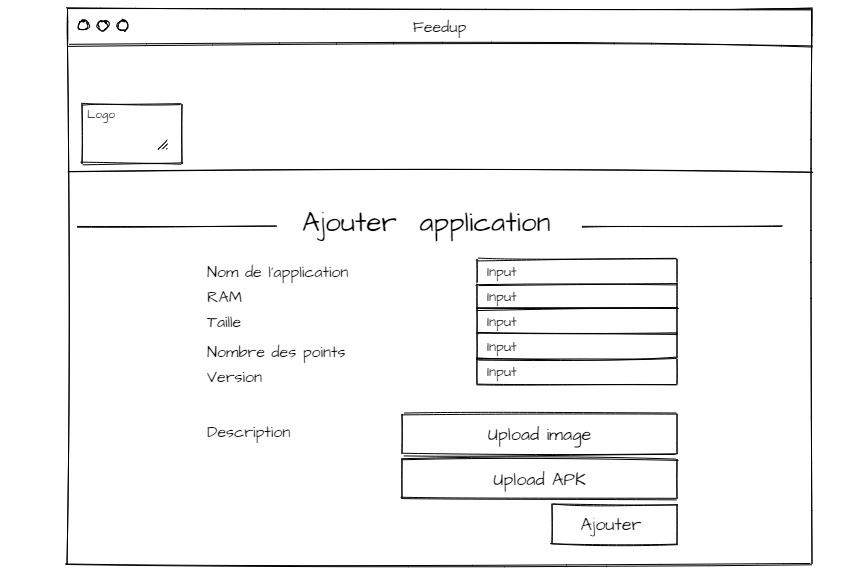


Figure 11 : Maquette « Ajouter une application »

La cinquième user story représente sur quoi on va commencer à travailler. La priorité métier est égale à (05)

* Une complexité représentée par le chiffre 20 (Dur)

Les Cas du succès et d’échec du user story « Ajouter une application » :

* **Succès**

1. Message affiché « Application ajouter avec succès ».

* **Echec : Affichage des messages**

1. Ces champs sont obligatoires.

## Conception

## Diagramme de classes

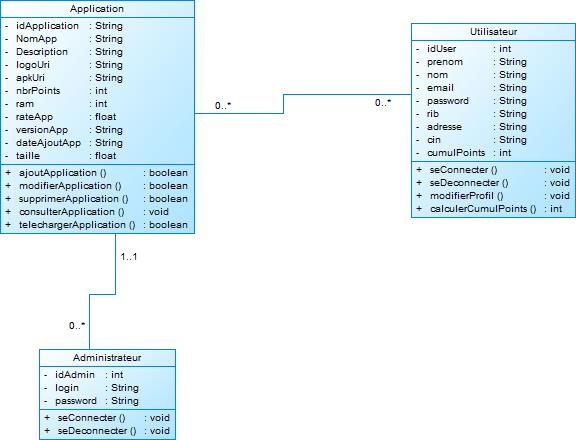


Figure 12 : Diagramme de classes

## Diagramme de séquence détaillé

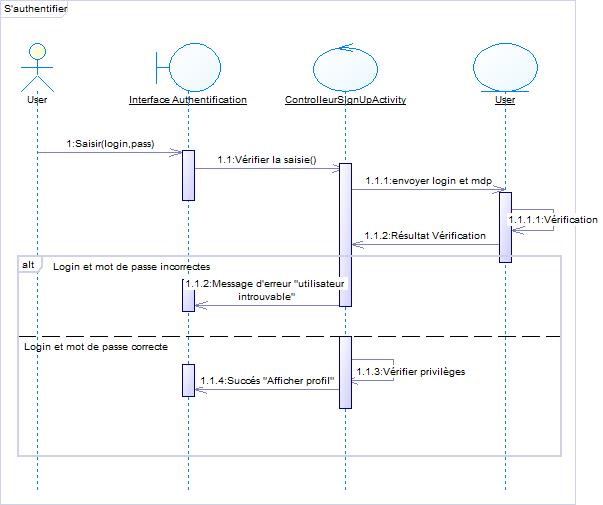


Figure 13 : Diagramme de séquence détaillé

## Réalisation

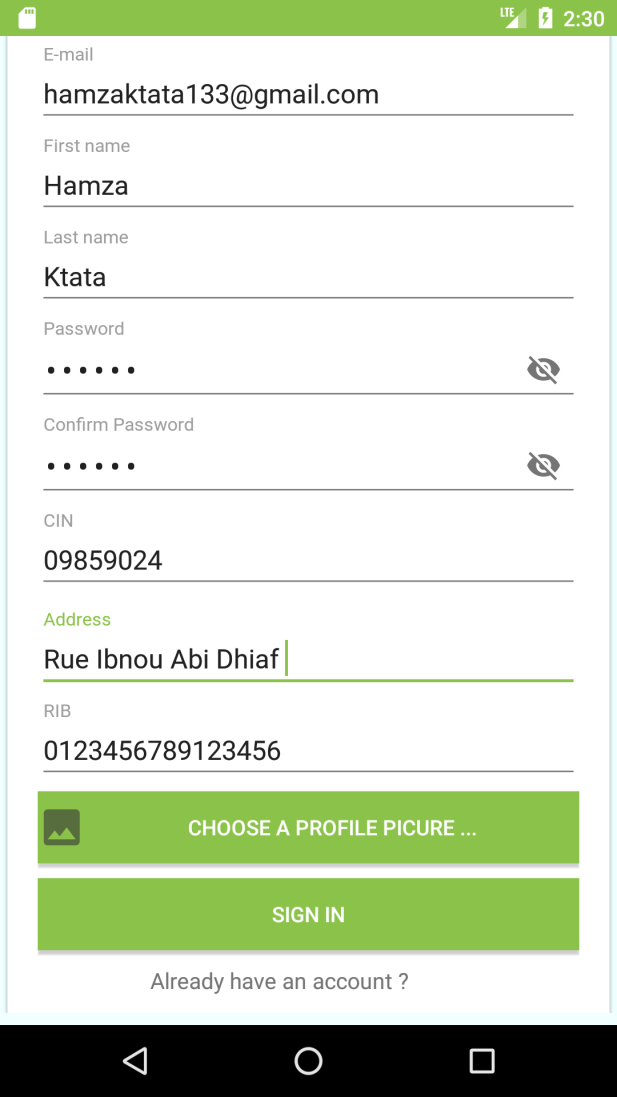
****

Figure 14 : Interface « Inscription »

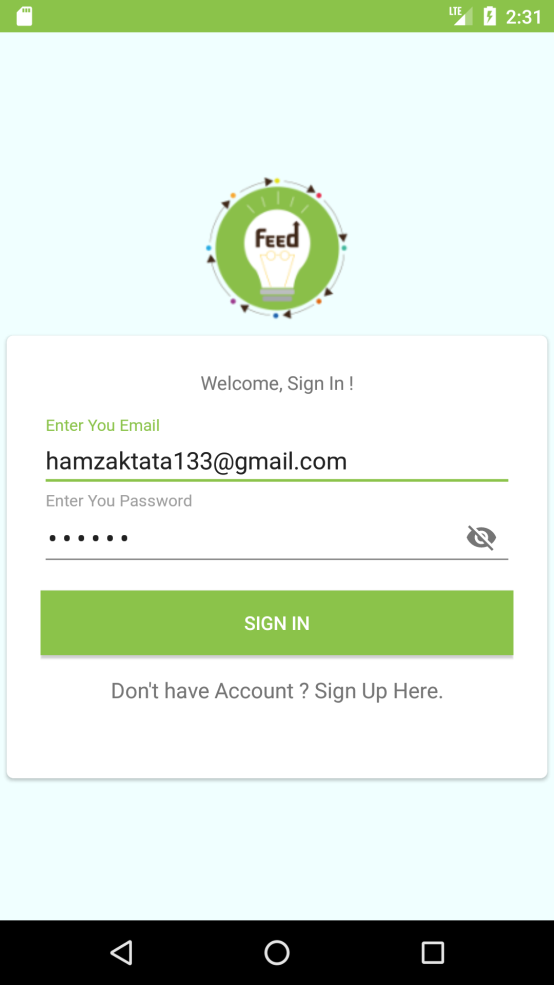
****

Figure 15 : Interface « Authentification utilisateur »

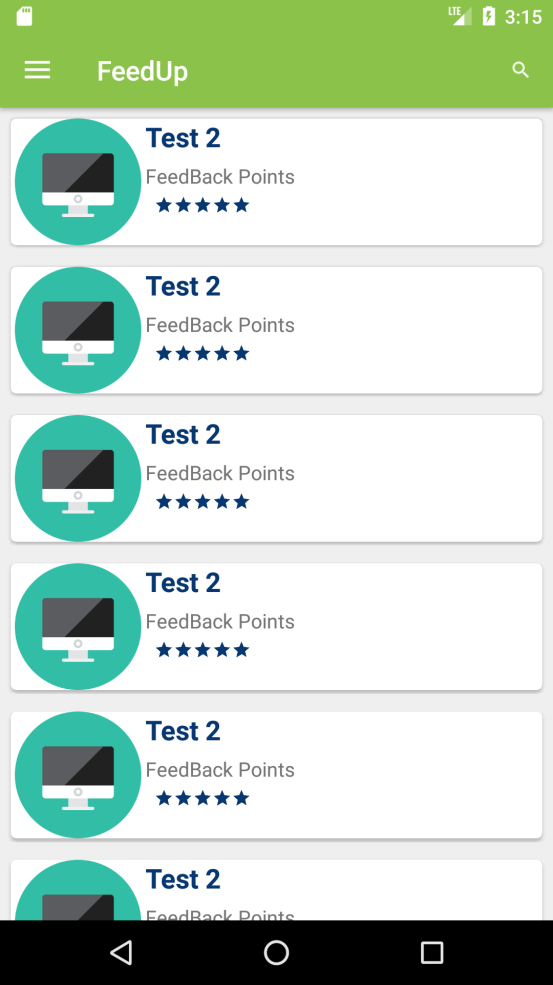


Figure 16 : Interface « Consultation la liste des applications développées »

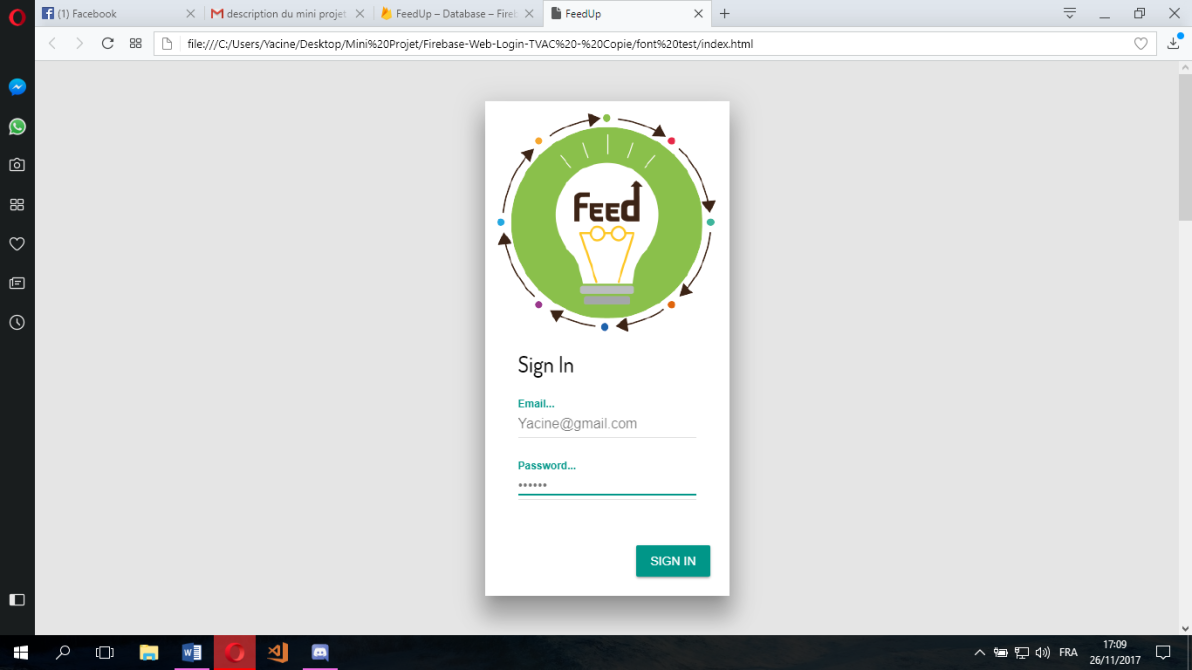


Figure 17: Interface « Authentification admin »

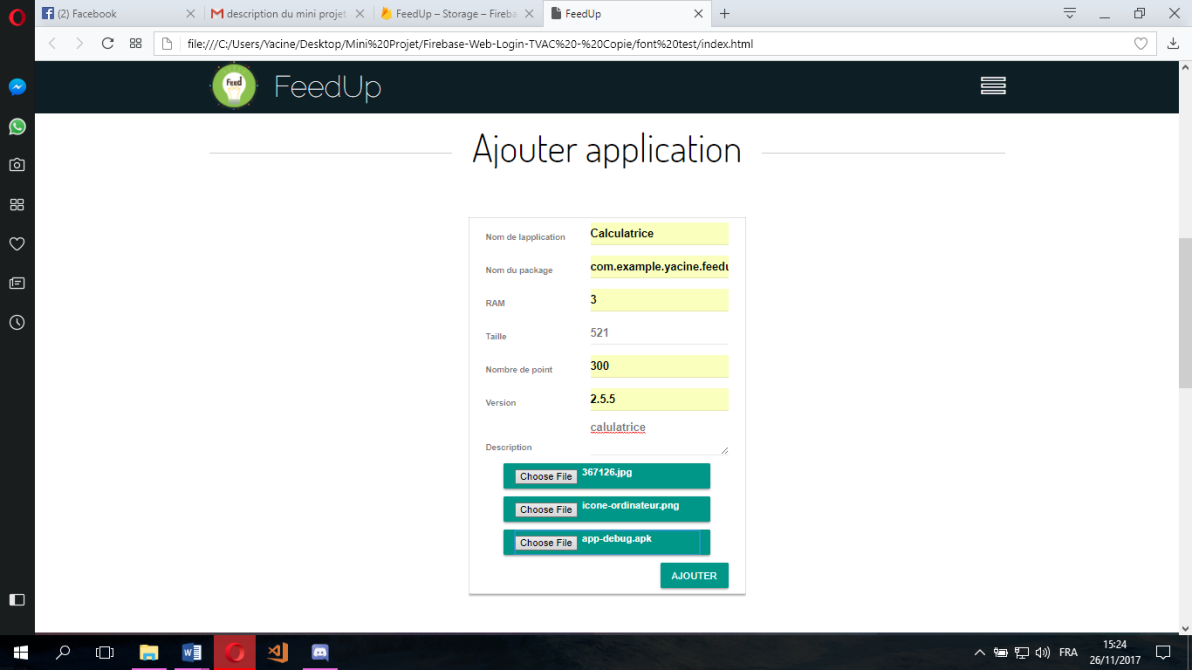


Figure 18 : Interface « Ajout application »

## La Revue du Sprint

## Burndown Chart

Ce graphe reflète l’avancement des travaux au sein de l’équipe. Il est formé de deux axes ; L’axe vertical indique la charge totale du sprint alors que l’axe horizontal représente les jours du sprint.En effet, dans notre cas le graphique(figure 16), fait apparaitre un retard un retard à cause du manque deconnaissance de Firebase.Ce retard a pu être rattrapé par la suite. Mais, on a eu un autre retard causé par nos études. En fin, on a pu terminer toutes les tâches a la fin du sprint.

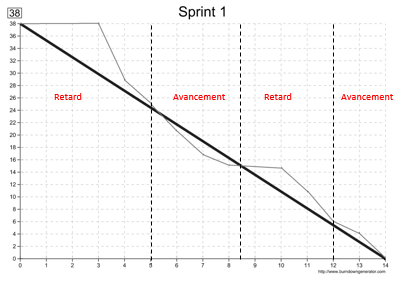


Figure 19 : Burndown Chart du Sprint 1

## Calcul de la vélocité

La vélocité estimée du début du sprint n’a pas changé car on a réussi à réaliser toutes les user stories du sprint, donc la vélocité réelle à la fin de ce sprint reste la même est égale à 45,5 points.

## Planification de la release

Au niveau de ce sprint, il n’y aura pas une mise à jour de notre planification puisque toutes les User stories du sprint ont été réalisées et acceptées.

## Conclusion

Au cours de ce chapitre nous avons présenté les différents user stories du premier sprint ainsi que la conception, les maquettes et les interfaces de ces user stories. Au chapitre suivant, nous allons présenter les user stories du deuxième sprint.

# Chapitre 4 : Sprint 2

## Introduction

Tout au long de ce chapitre, nous allons traiter les user stories de notre deuxième sprint pour produire un ensemble d’incréments potentiellement livrable.

## Spécifications fonctionnelles

## C:\Users\Yacine\AppData\Local\Microsoft\Windows\INetCache\Content.Word\bak 2.pngLe Backlog du Sprint 2

Figure 20 : Backlog du Sprint 2

**Présentation des user stories**

## Le prototype relatif au user story « Description d’une application »

Ceuser story permet à l’utilisateur de consulter la description d’une application :

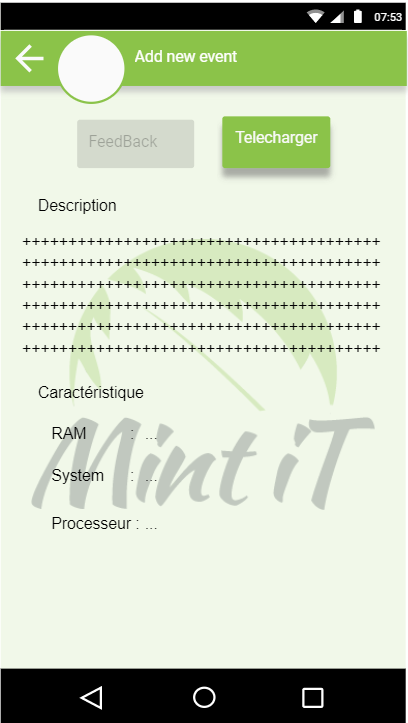
****

Figure 21 : Maquette « Description d’une application »

Le huitième user story représente la première interface.

La priorité métier est égale à (08)

* Une complexité représentée par le chiffre 1 (facile)

Les Cas du succès et d’échec du user story « description d’une applications » :

* **Succès**

1. Message affiché « Télécharger l’application ».
2. Message affiché « Accéder à la page de feedback ».

* **Echec : Affichage des messages**

1. Application non compatible.
2. Vous devez teste l’application pour accéder au feedback.

## Le prototype relatif au user story « Formulaire Feedback »

Ce user story permet à l’utilisateur de remplir un formulaire de FeedBack :

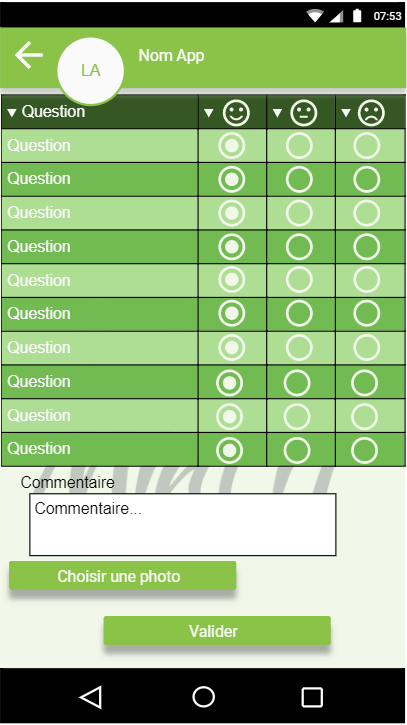


Figure 22 : Maquette « Formulaire FeedBack »

Le septième user story représente la première interface.

La priorité métier est égale à (07)

* Une complexité représentée par le chiffre 8 (Dur)

Les Cas du succès et d’échec du user story « Formulaire FeedBack » :

* **Succès**

1. Message affiché « Formulaire Valider avec succès »

* **Echec : Affichage des messages**

1. Vous devez répondre à tous les questions.

## Le prototype relatif au user story « En tant que utilisateur je veux convertir mes points»

Ce user story permet à l’utilisateur de convertir les points cumuler :

****

Figure 23 : Maquette « Conversion points »

Le treizième user story représente la première interface.

La priorité métier est égale à (13)

* Une complexité représentée par le chiffre 1 (facile)

Les Cas du succès et d’échec du user story « Conversion points » :

* **Succès**

1. Message affiché « Sélectionner le mentant que vous voulez convertir »

* **Echec : Affichage des messages**

1. Message affiché « Vous ne pouvait pas sélectionner se choix »

## Le prototype relatif au user story « En tant que administrateur je veux consulter la liste des applications»

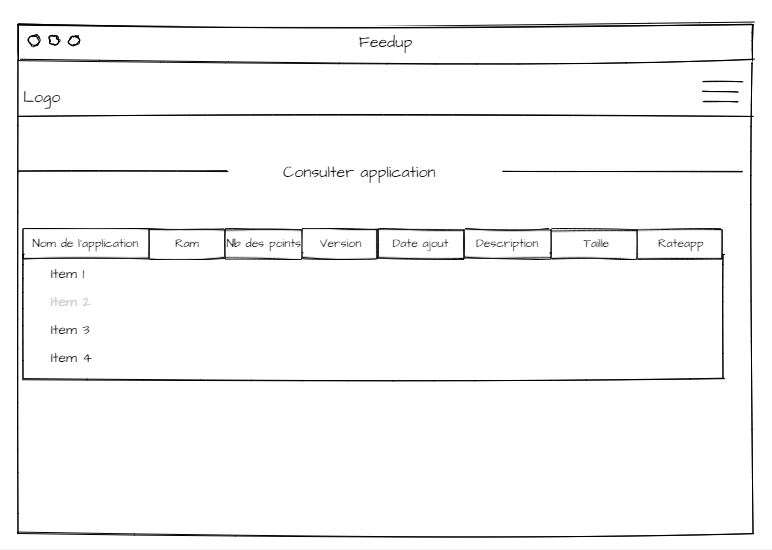
****

Figure 24 : Maquette « Consulter application »

La neuvième user story représente sur quoi on va commencer à travailler. La priorité métier est égale à (09)

* Une complexité représentée par le chiffre 1 (facile)

## Le prototype relatif au user story « En tant qu’administrateur je veux consulter les utilisateurs »

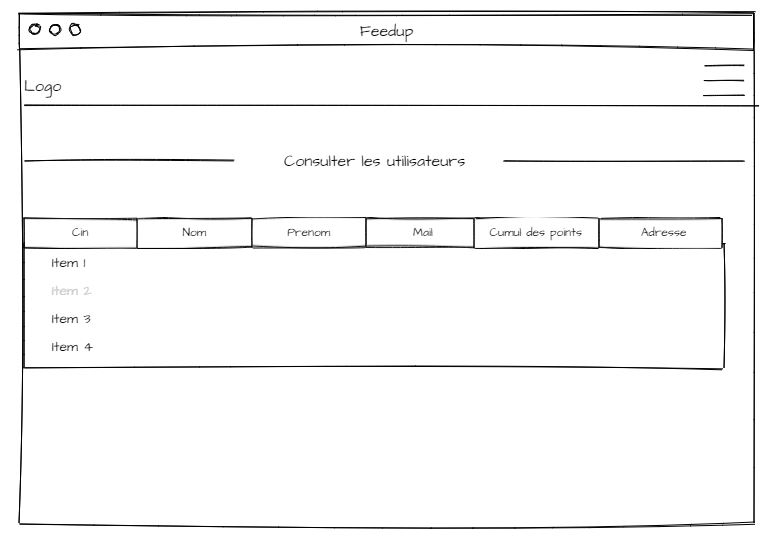
****

Figure 25 : Maquette « Consulter les utilisateurs »

La quatorzième user story représente sur quoi on va commencer à travailler. La priorité métier est égale à (14)

* Une complexité représentée par le chiffre 1/2 (facile)

## Le prototype relatif au user story «En tant qu’administrateur je veux consulter la liste des Feedbacks »

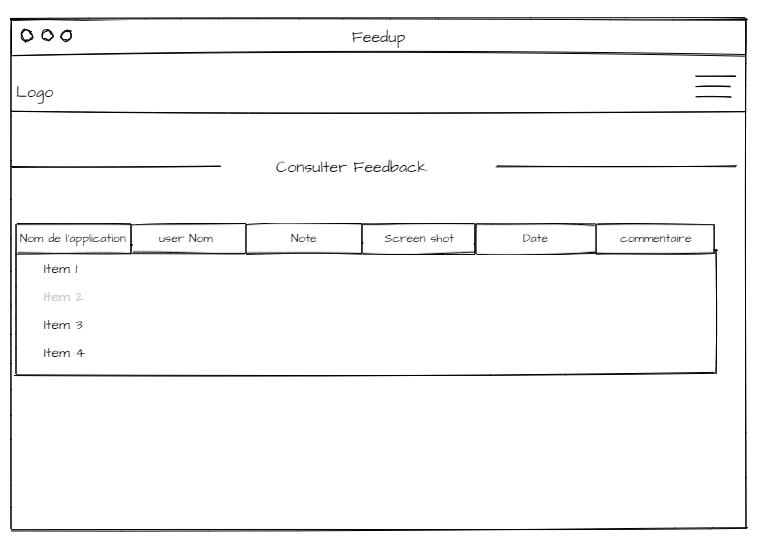
****

Figure 26 : Maquette « Consulter Feedback »

La onzième user story représente sur quoi on va commencer à travailler. La priorité métier est égale à (11)

* Une complexité représentée par le chiffre 5 (normal)

## Le prototype relatif au user story « Modifier application »

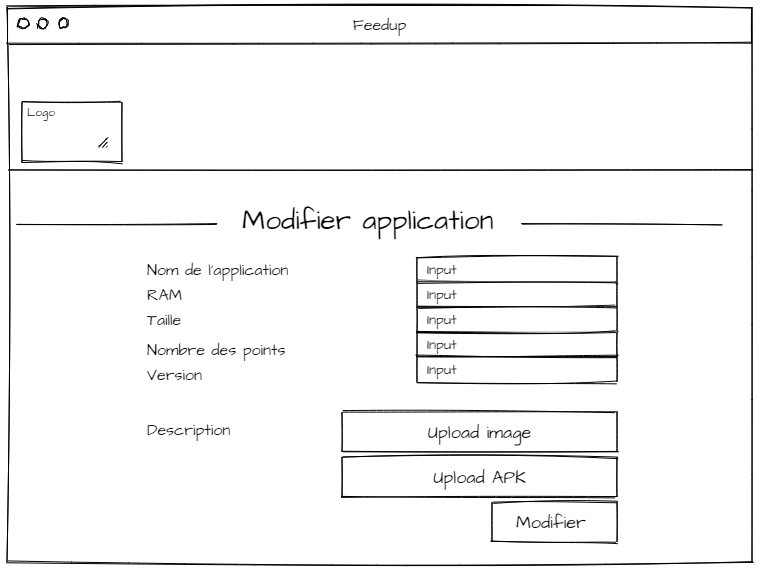
****

Figure 27 : Maquette « Modifier application »

La neuvième user story représente sur quoi on va commencer à travailler. La priorité métier est égale à (09)

* Une complexité représentée par le chiffre 13 (Dur)

Les Cas du succès et d’échec du user story « modifier Application » :

* **Succès :**

1. Message affiché « Application modifiée avec succès »

* **Echec : Affichage des messages**

## Conception

## Diagramme de classes

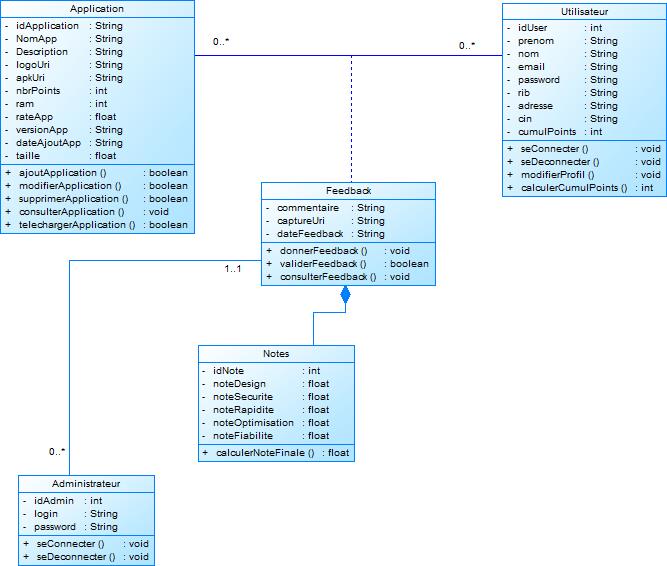
****

Figure 28 : Diagramme de classes relatif au Sprint 2

## Diagramme de séquence détaillé

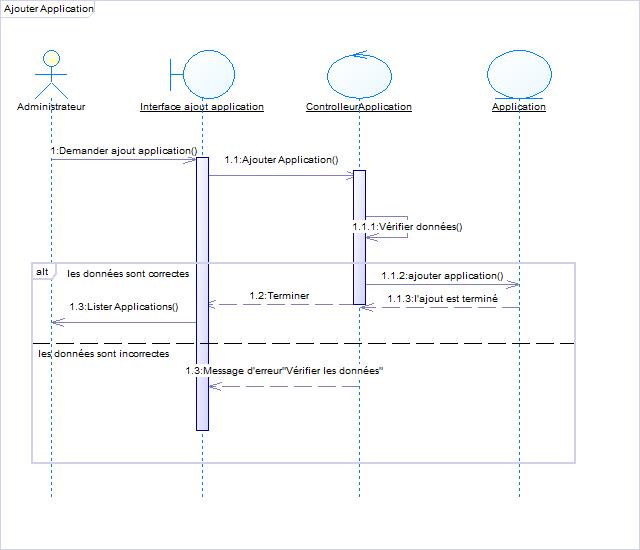
****

Figure 29 : Diagramme de séquence détaillé relatif au Sprint2

## Réalisation

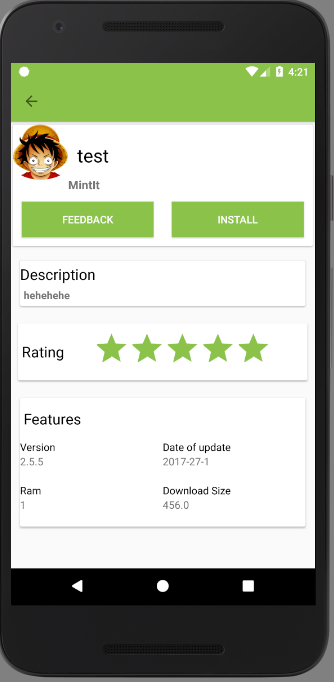
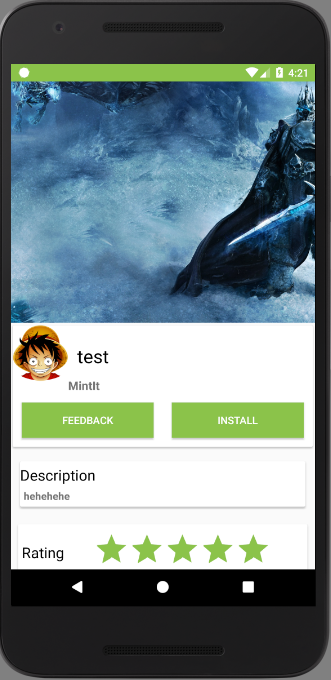
****

Figure 30 : Interface « Consulter description d’une application »

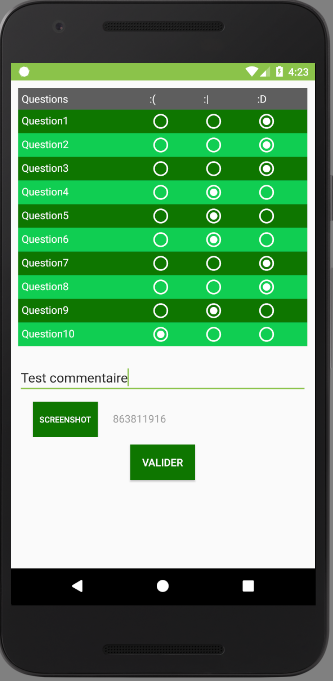
****

Figure 31 : Interface « Remplir le feedback »

****

Figure 32 : Interface « Conversion des points cumulés »

****

Figure 33 : Interface « Consulter les feedback de chaque application »

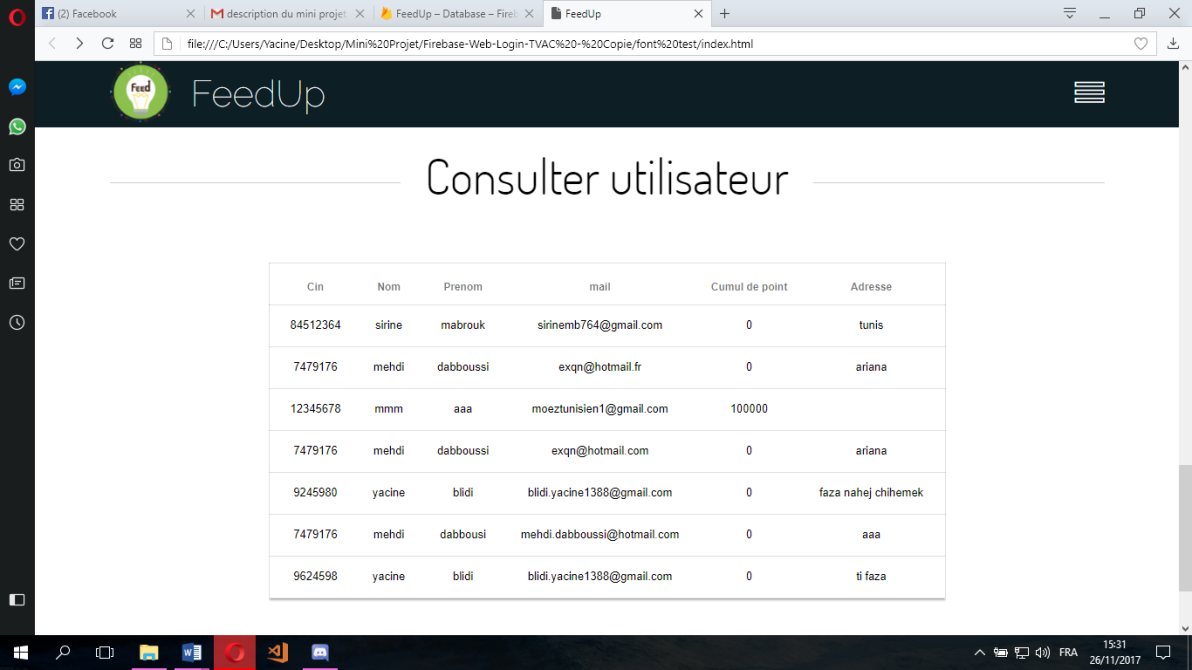
****

Figure 34 : Interface « Consulter le cumul de points de chaque user »

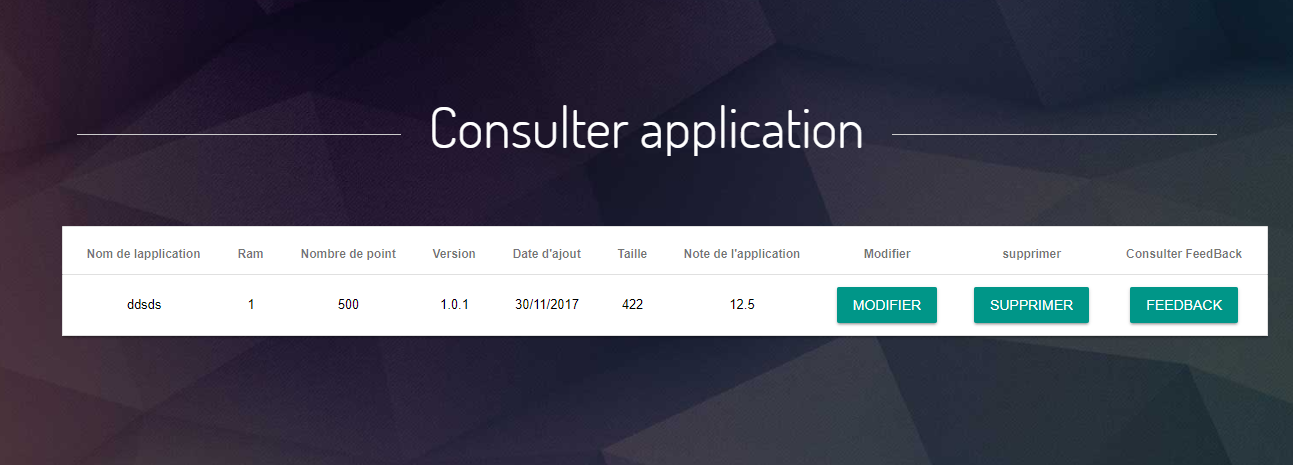


Figure 35 : Interface « Consulter application »



Figure 36 : Interface « Modifier application »

## La Revue du Sprint

## Burndown Chart

Motivée et bien organisée, l'équipe a commencé le Sprint 2 avec un rythme de travail élevé, ceci était durant les quatre premiers jours du sprint. Ensuite, du 4éme au 9éme jour, l'équipe a connu un retard dans la réalisation des tâches à cause de problèmes personnels survenant dans un des membres de l'équipe. Certains problèmes techniques sont apparus. Du 10éme jusqu’au 21éme jour (Fin du Sprint), l'équipe a repris le travail, et a rattrapé le retard qui a été fait, pour enfin pouvoir terminer le sprint avec succès dans les délais impartis.

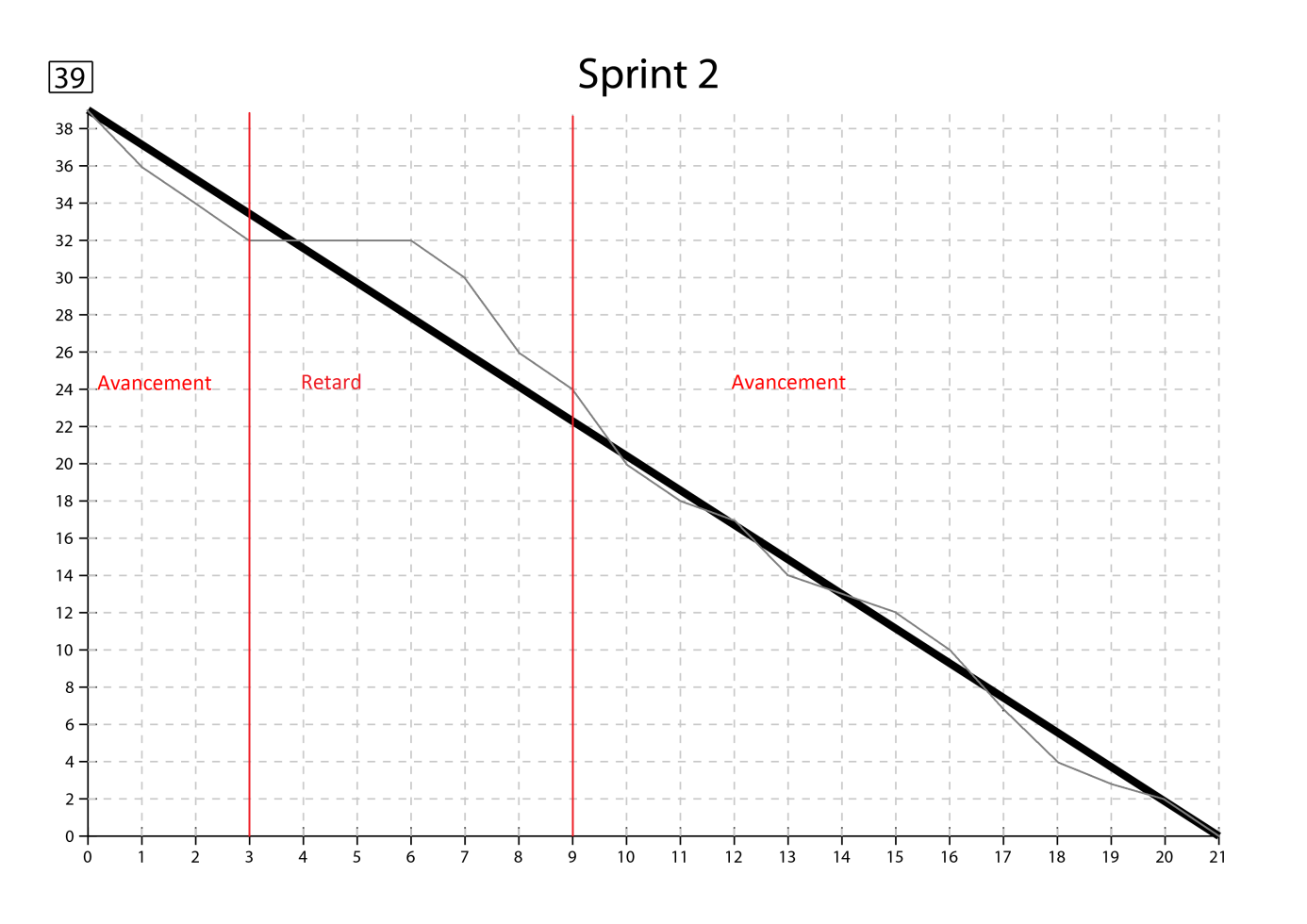


Figure 37 : Burdown Chart du Sprint 2

## Calcul de la vélocité

La vélocité estimée du début du sprint n’a pas changé car on a réussi à réaliser toutes les user stories du sprint, donc la vélocité réelle à la fin de ce sprint reste la même est égale à 44 points.

## Planification de la release

Au niveau de ce sprint, il n’y aura pas une mise à jour de notre planification puisque toutes les User stories du sprint ont été réalisées et acceptées.

## Conclusion

Durant ce chapitre, nous avons présenté les différentes user stories de sprint 2 ainsi que leur conception, leurs maquettes et leurs interfaces. Nous allons présenter les user stories du dernier sprint (3) au chapitre suivant.

# Chapitre 5 : Réalisation du Sprint 3

## Introduction

Tout au long de ce chapitre, nous allons traiter les user stories de notre deuxième sprint pour produire un ensemble d’incréments potentiellement livrable.

## Spécifications fonctionnelles

## Le Backlog du Sprint 3

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **User Story** | **Taches** | **Estimation en heure** |
| **1** | En tant que admin je veux avoir le suivi des payements des utilisateurs | Interface | 1h |
| Code Behind | 2h |
| **2** | En tant que utilisateur je veux Consulter les coordonnées de la boite de développement | Interface | 3h |
| Code behind | 1h |
| **3** | En tant que utilisateur je veux rechercher une application dans la barre de recherche | Code behind | 3h |
| **4** | En tant que utilisateur je veux consulter dans les caractéristiques de mon appareil | Interface | 3h |
| Code behind | 1h |
| **5** | En tant que utilisateur je veux modifier mon profil | Interface | 1h |
| Code behind | 3h |
| **6** | En tant que utilisateur je veux me déconnecter | Code behind | 3h |
| **7** | En tant que admin je veux supprimer des applications | Code behind | 3h |

Tableau 5:Backlog du sprint 3

## Présentation des User Stories

## Le prototype relatif au User Story « En tant que utilisateur je veux modifier mon profil»

Ce user story permet à l’utilisateur de modifier son profil.

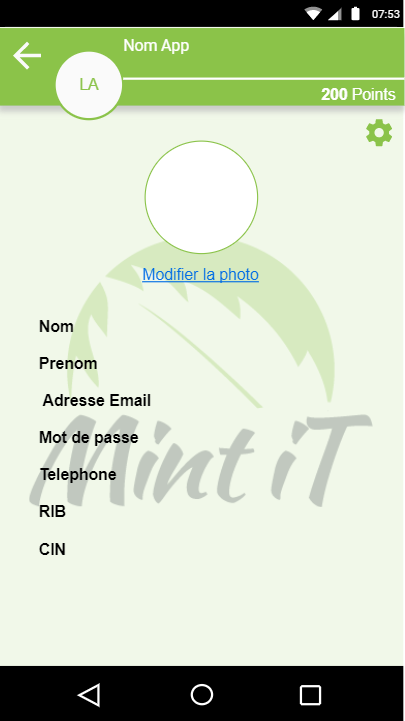


Figure 38 : Prototype de l’interface «Modifier profil»

La priorité métier de ce user story est égale à (19)

* Une complexité représentée par le chiffre 5 (normal)

Les Cas du succès et d’échec du user story «Modifier profil» :

* **Succès**

1. Message affiché « Formulaire affiché avec succès».

b) Message affiché «La modification a été effectuée avec succès».

* **Echec : Affichage des messages**

1. Ces champs sont obligatoires.
2. Veuillez entrer une adresse électronique valide.
3. Le mot de passe doit être tapé deux fois et de longueurs supérieures ou égales à 6 caractères.
4. Le numéro de téléphone doit être valide.
5. Le numéro du RIB doit être valide.
6. Le numéro du CIN doit être valide.
7. **Le prototype relatif au user story « En tant qu’utilisateur je veux consulter les caractéristiques de mon appareil»**

Ce user story permet à un utilisateur de visualiser les caractéristiques de son appareil.

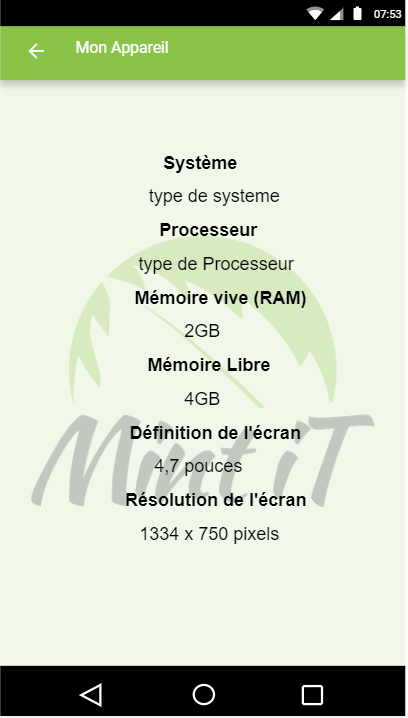


Figure 39 : Prototype de l’interface «Caractéristiques de l’appareil »

La priorité métier est égale à (18)

* Une complexité représentée par le chiffre 8(normal).

Les Cas du succès et d’échec du user story «Consulter les caractéristiques de l’appareil» :

## Le prototype relatif au user story « En tant que utilisateur je veux rechercher une application précise »

Ce user story permet de rechercher une application.



Figure 40 : Prototype de l’interface «Rechercher une application précise»

* La priorité métier est égale à (15).
* Une complexité représentée par le chiffre 5 (normal)

Les Cas du succès et d’échec du user story « Rechercher une application précise » :

* **Succès**

1. L’application est affichée avec succès.

* **Echec : Affichage des messages**

1. Le nom de l’application saisie est incorrecte.
2. Aucune application n’est disponible.

## Le prototype relatif au user story «En tant que administrateur je veux supprimer des applications»

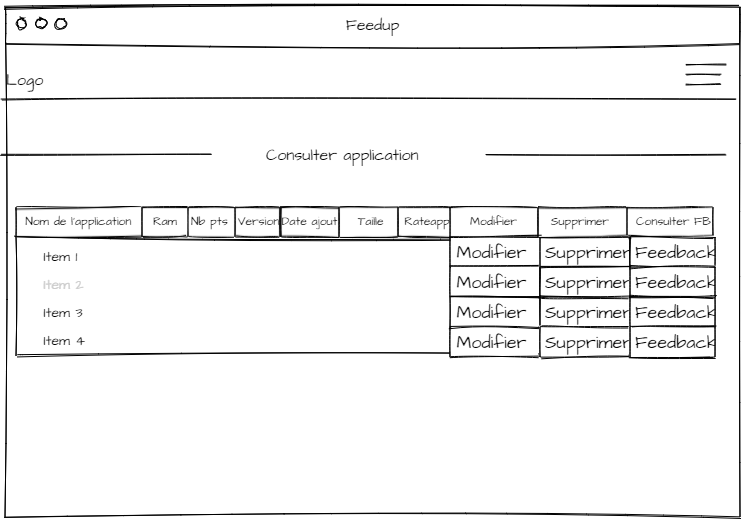


Figure 41 : Prototype de l’interface «Supprimer Application»

La priorité métier est égale à (21)

* Une complexité représentée par le chiffre 13 (difficile)

Les Cas du succès et d’échec du user story «Supprimer une application précise» :

* **Succès**

1. Message affiché « Application supprimé avec succès».

* **Echec : Affichage des messages**

1. Il faut sélectionner l’application à supprimer.

## Le prototype relatif au user story « En tant que administrateur je veux avoir le suivi des payements»

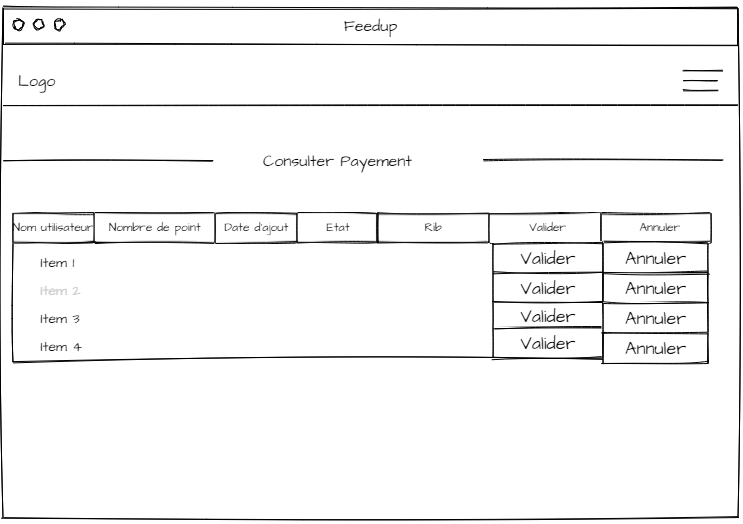


Figure 42 : Prototype de l’interface «Avoir le sui des payements»

La priorité métier est égale à (15)

* Une complexité représentée par le chiffre 5 (normal)

Les Cas du succès et d’échec du user story « Ajouter une application » :

* **Succès**

1. La liste des payements est affichée avec succès.

* **Echec : Affichage des messages**

1. La liste n’est pas disponible.
2. **Conception**
   * + 1. **Diagramme de classes**

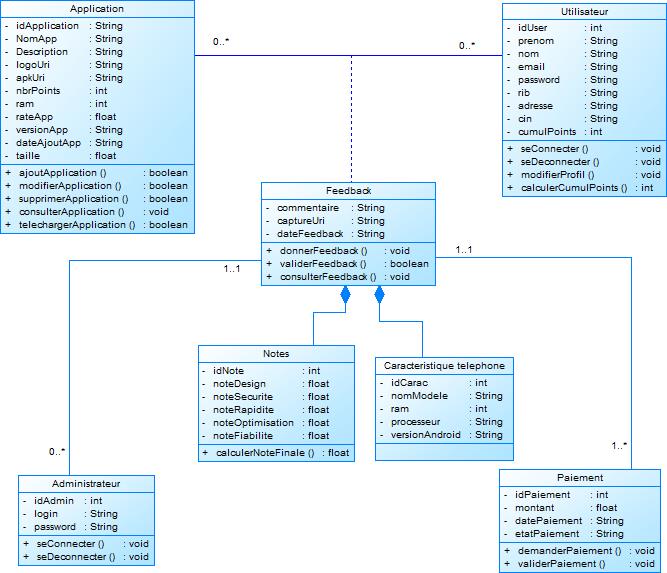


Figure 43 : Diagramme de classes relatif au Sprint 3

1. **Réalisation**

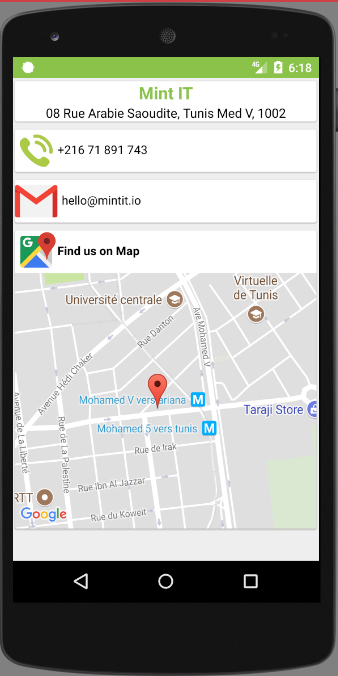


Figure 44 : Interface « Consulter les coordonnées de la boîte »

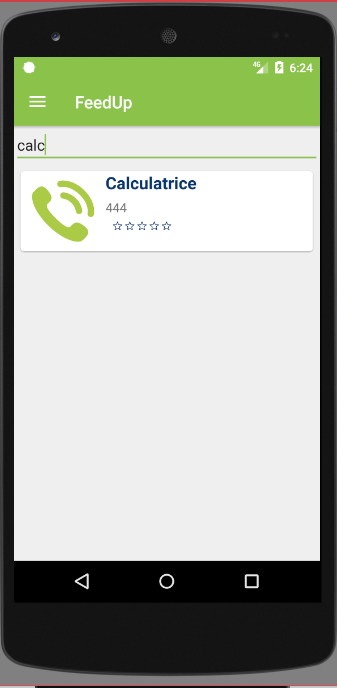
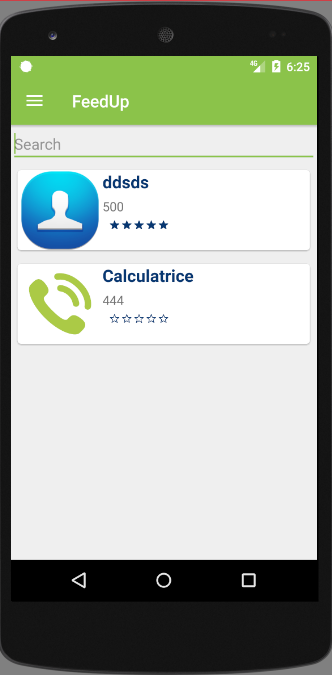


Figure 45 : Interface « Recherche une application »

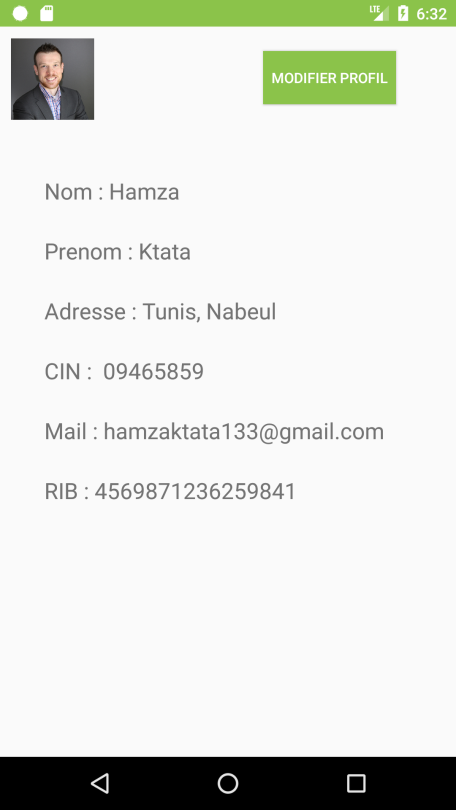


Figure 46 : Interface « Modifier mon profil »

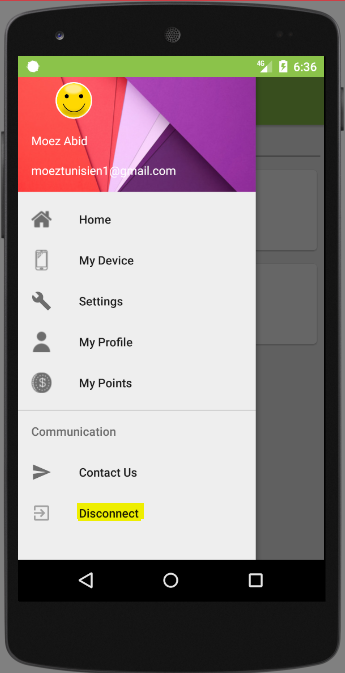


Figure 47 : Interface « Déconnexion »

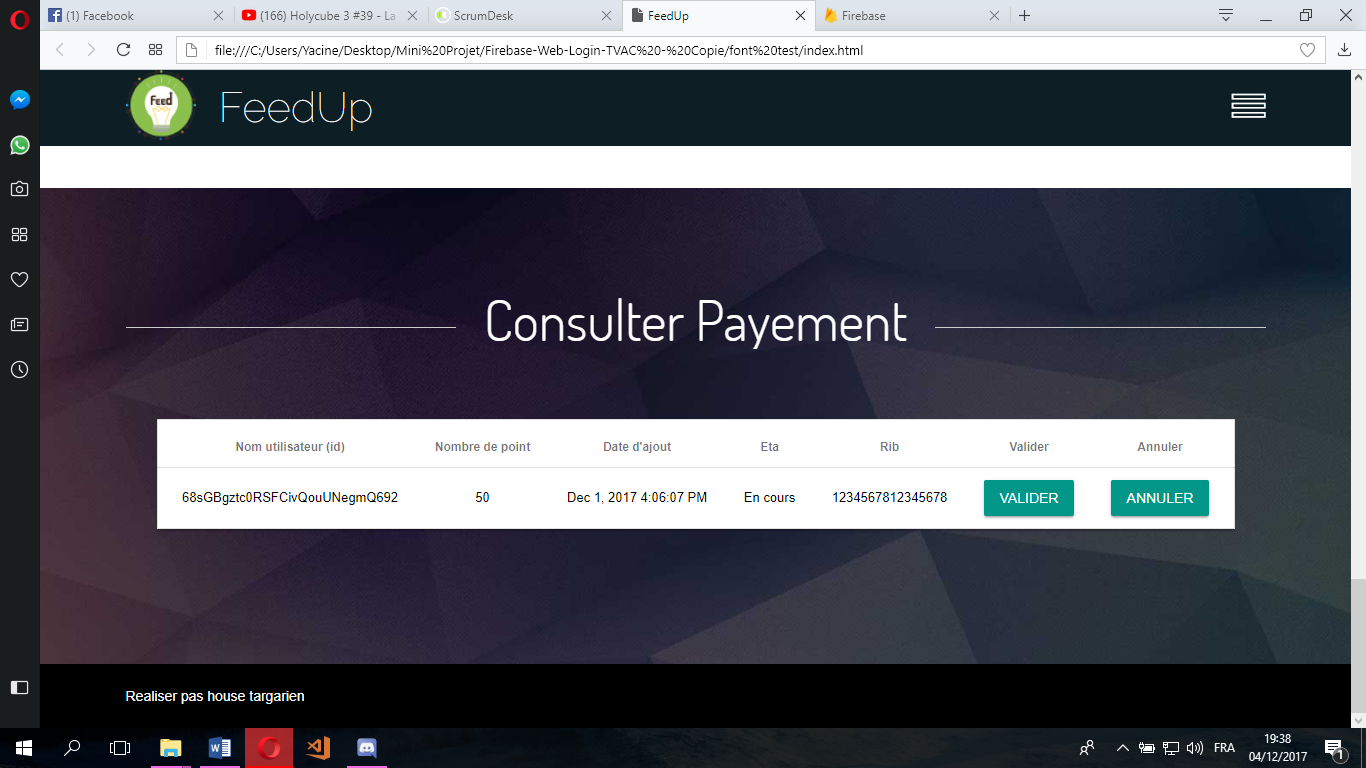


Figure 48 : Interface « Consulter Payements »

## Conclusion

Tout au long de ce chapitre nous avons présenté les différents user story,la conception,les maquettes et les interfaces du dernier sprint .

# Conclusion générale

Nous avons déployé notre application, afin d’effectuer des tests et repérer les éventuels bugs.

La réalisation de ce travail a été bénéfique sur plusieurs plans ; En effet, il nous a appris comment surmonter les difficultés techniques que nous avons rencontré et surtout comment faire face aux ambigüités et aux contraintes techniques pour tester de nouvelles alternatives aboutissant à l’accomplissement du projet.

Le travail était très intéressant puisqu’il nous a permis de comprendre les différents types de maintenance et leurs procédures. Il nous a permis de mettre en oeuvre et de consolider les connaissances théoriques acquises tout au long de notre cursus universitaire. Nous avons alors utilisé la programmation orientée objet avec plusieurs concepts du génie logiciel. Nous avons aussi suivi la méthode agile SCRUM qui nous a paru intéressante et adéquate avec notre projet.

# Nétographie

[1] C. Aubry, Le guide pratique de la méthode agile la plus populaire.

[2] «Affinity Software,» [En ligne]. Available: http://www.affinity-software.fr/v1/formations/les-methodesagiles.

[Accès le 15 Février 2015].

[3] «mountaingoatsoftware,» [En ligne]. Available:

http://www.mountaingoatsoftware.com/agile/scrum/images. [Accès le 26 février 2015].

[4] P. Roques, UML 2 en action.