

## PROJET DE FIN D'ÉTUDES

### DIPLÔME NATIONAL D'INGÉNIEUR

**SPÉCIALITÉ : INFORMATIQUE**

**Conception et implémentation d'une  
application web dédiée pour la  
gestion du contenu d'un jeu mobile**

Réalisé par : Ahmed Cheikh

Encadrant ESPRIT : Madame Ameni Rommene

Encadrant Entreprise : Monsieur Nадhir Khayati



# Table des matières

Introduction générale.....	1
Chapitre 1 : Contexte général du projet .....	2
Introduction .....	3
1. Présentation de l'organisme d'accueil.....	3
1.1. Présentation .....	3
1.2. Organigramme .....	4
2. Présentation générale du projet .....	5
2.1. Cadre du projet .....	5
2.2. Problématique .....	6
2.3. Etude de l'existant .....	6
2.4. Solution proposée .....	7
3. Méthode de travail.....	7
3.1. Les méthodes agiles.....	8
3.2. Etude comparative des méthodes agiles .....	8
3.3. La méthode adoptée : SCRUM.....	9
3.4. Equipe.....	11
4. Choix technologique .....	12
4.1. Laravel 5.5 .....	12
4.2. Angular 5.2 .....	14
4.3. Swagger 5.5 .....	15
4.5. MySQL 5.6.....	16
5. Outils utilisés.....	16
5.1. Jira .....	16
5.2. Bitbucket.....	17
5.3. Slack .....	17

5.4. Jenkins .....	18
5.5. Amazon S3 .....	19
5.6. Firebase Cloud Messaging (FCM) .....	19
5.7. Redis.....	19
6. Répartition des sprints .....	19
Conclusion.....	21
Chapitre 2 : Analyse et spécification des besoins (Sprint 0).....	22
Introduction .....	23
1. Identification des acteurs.....	23
2. Identification des besoins fonctionnels et non fonctionnels.....	23
2.1. Identification des besoins fonctionnels.....	23
2.2. Identification des besoins non fonctionnels.....	24
3. Backlog de produit .....	26
4. Diagramme de cas d'utilisation.....	33
5. Diagramme de classe d'analyse .....	34
6. Diagramme de déploiement .....	36
Conclusion.....	36
Chapitre 3 : Sprint 1 « Gestion des utilisateurs ».....	37
Introduction .....	38
1. Sprint Backlog.....	38
2. Diagramme de cas d'utilisation raffiné .....	40
3. Description du cas d'utilisation «s'authentifier» .....	41
3.1. Description textuelle du cas d'utilisation « s'authentifier » .....	41
3.2. Diagramme de séquence du cas d'utilisation « s'authentifier » .....	41
4. Description du cas d'utilisation « ajouter un utilisateur ».....	42
4.1. Description textuelle du cas d'utilisation « Ajouter un utilisateur » .....	42
4.2. Diagramme de séquence objet du cas d'utilisation « ajouter un utilisateur » .....	43

5.	Description du cas d'utilisation « modifier un utilisateur ».....	44
5.1.	Description textuelle du cas d'utilisation «modifier un utilisateur».....	44
5.2.	Diagramme de séquence objet du cas d'utilisation « modifier un utilisateur ».....	44
6.	Description du cas d'utilisation « supprimer un utilisateur ».....	45
6.1.	Description textuelle du cas d'utilisation « supprimer un utilisateur » .....	45
6.2.	Diagramme de séquence objet du cas d'utilisation « supprimer un utilisateur » .....	46
7.	Diagramme de classe de conception .....	46
8.	Réalisation .....	48
8.1.	Maquettes.....	48
8.2.	REST API.....	50
8.3.	Sécurisation .....	51
8.3.1.	Passport .....	51
8.3.2.	Autorisations .....	52
8.4.	Interfaces graphiques .....	53
Conclusion.....	55	
Chapitre 4 : Sprint 2 « Gestion du contenu » .....	56	
Introduction .....	57	
1.	Sprint backlog .....	57
2.	Diagramme de cas d'utilisation raffiné .....	67
3.	Description du cas d'utilisation « Ajouter version ».....	70
3.1.	Description textuelle du cas d'utilisation « Ajouter version » .....	70
3.2.	Diagramme de séquence du cas d'utilisation « ajouter une version ».....	71
4.	Description du cas d'utilisation « modifier une version » .....	71
4.1.	Description textuelle du cas d'utilisation « modifier une version ».....	71
4.2.	Diagramme de séquence du cas d'utilisation « modifier une version ».....	72
5.	Description du cas d'utilisation « supprimer une version » .....	73
5.1.	Description textuelle du cas d'utilisation « supprimer une version » .....	73

5.2. Diagramme de séquence du cas d'utilisation « supprimer une version » .....	73
6. Description du cas d'utilisation « changer le statut d'une version ».....	74
6.1. Description textuelle du cas d'utilisation « changer le statut d'une version » .....	74
6.2. Diagramme de séquence du cas d'utilisation « changer le statut une version » .....	75
7. Description du cas d'utilisation « Dupliquer une version ».....	76
7.1. Description textuelle du cas d'utilisation « Dupliquer une version » .....	76
7.2. Utilisation des WebSockets.....	77
8. Diagramme de classe.....	78
9. Réalisation.....	79
9.1. Maquettes .....	79
9.2. Interfaces graphiques.....	82
Conclusion.....	84
Chapitre 5 : Sprint « Gestion des divisions, des notifications et des FAQ ».....	85
Introduction .....	86
1. Sprint backlog .....	86
2. Diagramme de cas d'utilisation raffiné .....	90
3.   Description du cas d'utilisation « ajouter une partition » .....	91
3.1.   Description textuelle du cas d'utilisation « Ajouter une partition ».....	91
3.2.   Diagramme de séquence du cas d'utilisation « ajouter une partition ».....	92
4. Description du cas d'utilisation « modifier une partition » .....	92
4.1. Description textuelle du cas d'utilisation « modifier une partition » .....	92
4.2. Diagramme de séquence du cas d'utilisation « modifier une partition» .....	93
5. Description du cas d'utilisation « supprimer une partition ».....	94
5.1. Description textuelle du cas d'utilisation « supprimer une partition » .....	94
5.2. Diagramme de séquence du cas d'utilisation « supprimer une partition ».....	94
6. Description du cas d'utilisation « envoyer une notification push ».....	95
6.1. Description textuelle du cas d'utilisation « envoyer une notification push » .....	95

6.2. Diagramme de séquence du cas d'utilisation « envoyer une notification push » .....	96
7. Diagramme de classe.....	96
7.1. Diagramme de classe « gestion des partitions ».....	96
7.2. Diagramme de classe « gestion des notifications ».....	97
7.3. Diagramme de classe « gestion des paramètres » .....	98
8. Réalisation.....	99
8.1. Maquettes .....	99
8.2. Interfaces graphiques.....	103
Conclusion.....	106

# Liste des figures

Figure 1 : Logo de 360Medlink.....	3
Figure 2: Organigramme de 360Medlink .....	4
Figure 3 : Aperçu de MyReady Transition.....	6
Figure 4 : Architecture de la solution.....	7
Figure 5 : Cycle de vie Scrum [1] .....	10
Figure 6 : Logo de Laravel .....	12
Figure 7 : Modèle MVC .....	13
Figure 8 : Logo de Angular .....	14
Figure 9 : Interface de la documentation des API via Swagger .....	16
Figure 10 : Plateform JIRA .....	17
Figure 11 : Interface de Slack .....	18
Figure 12 : Interface de Jenkins .....	18
Figure 13 : Principe des jetons JWT .....	24
Figure 14 : Temps de réponse du serveur pour la liste des patients .....	25
Figure 15 : Exemple de documentation générée par compodoc .....	26
Figure 16 : Diagramme de cas d'utilisation global .....	34
Figure 17 : Diagramme de classe global .....	35
Figure 18 : Diagramme de déploiement .....	36
Figure 19 : Diagramme de cas d'utilisation pour le module de gestion des utilisateurs .....	40
Figure 20 : diagramme de séquence du cas d'utilisation « s'authentifier » .....	42
Figure 21 : Diagramme de séquence objet du cas d'utilisation « ajouter un utilisateur » .....	43
Figure 22 : Diagramme de séquence objet du cas d'utilisation « modifier un utilisateur ».....	45
Figure 23 : diagramme de séquence objet du cas d'utilisation « supprimer un utilisateur » ...	46
Figure 24 : Diagramme de classe relatif au module gestion des utilisateurs .....	47
Figure 25 : Maquette de la page d'authentification .....	49
Figure 26 : Maquette de la page de la liste des patients .....	49
Figure 27 : Maquette de la page d'ajout d'un patient.....	50
Figure 28 : API REST pour le module « gestion des utilisateurs » .....	51
Figure 29 : Access token généré par Passport.....	52
Figure 30 : Envoi du token dans le header de la requête.....	52
Figure 31 : Exemple de définition de policy .....	53

Figure 32 : Utilisation des policies.....	53
Figure 33 : Interface d'authentification .....	54
Figure 34 : Interface de la liste des patients .....	54
Figure 35 : Interface d'ajout d'un patient.....	55
Figure 36 : Diagramme de cas d'utilisation raffiné « gestion des versions » .....	67
Figure 37 : Diagramme de cas d'utilisation raffiné « gestion des segments » .....	68
Figure 38 : Diagramme de cas d'utilisation raffiné « gestion des emplacements » .....	68
Figure 39 : Diagramme de cas d'utilisation raffiné « gestion des labels » .....	69
Figure 40 : Diagramme de cas d'utilisation raffiné « gestion des accessoires » .....	70
Figure 41 : Diagramme de séquence du cas d'utilisation « ajouter une version».....	71
Figure 42 : Diagramme de séquence du cas d'utilisation « modifier une version » .....	72
Figure 43 : Diagramme de séquence du cas d'utilisation « supprimer une version ».....	74
Figure 44 : Diagramme de séquence du cas d'utilisation « changer le statut d'une version »..	76
Figure 45 : Flux de diffusion Laravel avec Socket.io et Redis PubSub.....	77
Figure 46 : Diagramme de classe relatif au module gestion du contenu (1 <sup>ère</sup> partie).....	78
Figure 47 : Diagramme de classe relatif au module de gestion du contenu (2 <sup>ème</sup> partie) .....	79
Figure 48 : Maquette de la liste des versions .....	80
Figure 49 : Maquette de la page d'ajout d'une version .....	80
Figure 50 : Maquette d'ajout d'un chapitre .....	81
Figure 51 : Maquette d'ajout d'une session .....	81
Figure 52 : Interface de la gestion des versions .....	82
Figure 53 : Interface d'ajout d'une version .....	82
Figure 54 : Interface d'ajout d'un chapitre .....	83
Figure 55 : Interface d'ajout d'une session .....	83
Figure 56 : Interface d'ajout d'un dictionnaire dans une session .....	84
Figure 57 : Interface de gestion des langues .....	84
Figure 58 : Diagramme de cas d'utilisation raffiné du module « Gestion des partitions » .....	90
Figure 59 : Diagramme de cas d'utilisation raffiné « envoi des notifications » .....	90
Figure 60 : Diagramme de cas d'utilisation raffiné « gestion des FAQ » .....	91
Figure 61 : Diagramme de séquence du cas d'utilisation « ajouter une partition ».....	92
Figure 62 : Diagramme de séquence du cas d'utilisation « modifier une partition » .....	93
Figure 63 : Diagramme de séquence du cas d'utilisation « supprimer une partition » .....	95
Figure 64 : Diagramme de séquence du cas d'utilisation « envoyer une notification push »..	96
Figure 65 : Diagramme de classe relatif au module « gestion des partitions » .....	97

Figure 66 : Diagramme de classe relatif au module « gestion des notifications » .....	98
Figure 67 : Diagramme de classe relatif au module « gestion des paramètres » .....	99
Figure 68 : Maquette d'ajout d'une division .....	100
Figure 69 : Maquette d'ajout d'une region .....	100
Figure 70 : Maquette d'ajout d'un site .....	101
Figure 71 : Maquette d'envoi d'une notification push .....	102
Figure 72 : Maquette d'ajout d'une notification automatique .....	102
Figure 73 : Interface graphique de la liste des partitions .....	103
Figure 74 : Interface graphique d'ajout d'une division .....	103
Figure 75 : Interface graphique d'ajout d'une region .....	104
Figure 76 : Interface graphique d'ajout d'un site .....	104
Figure 77 : Interface graphique de la configuration d'une notification automatique .....	105
Figure 78 : Interface graphique d'envoi d'une notification push .....	105
Figure 79 : Interface graphique de la page de gestion des FAQ .....	106

# Liste des tableaux

Tableau 1 : Tableau comparatif des méthodes agiles.....	9
Tableau 2 : Equipe du projet .....	11
Tableau 3 : Répartition des sprints.....	20
Tableau 4 : Product backlog.....	33
Tableau 5 : Sprint backlog gestion des utilisateurs .....	40
Tableau 6 : Description textuelle du cas d'utilisation « s'authentifier ».....	41
Tableau 7 : Description textuelle du cas d'utilisation « ajouter un utilisateur ».....	43
Tableau 8 : Description textuelle du cas d'utilisation « modifier un utilisateur » .....	44
Tableau 9 : Description textuelle du cas d'utilisation « supprimer un utilisateur » .....	46
Tableau 10 : Sprint backlog « Gestion du contenu » .....	67
Tableau 11 : Description textuelle du cas d'utilisation « ajouter une version » .....	71
Tableau 12 : Description textuelle du cas d'utilisation « modifier une version ».....	72
Tableau 14 : Description textuelle du cas d'utilisation « supprimer une version » .....	73
Tableau 16 : Description textuelle du cas d'utilisation « changer le statut d'une version » .....	75
Tableau 17 : Sprint backlog 3ème sprint.....	89
Tableau 18 : Description textuelle du cas d'utilisation « ajouter une partition ».....	92
Tableau 19 : Description textuelle du cas d'utilisation « modifier une partition ».....	93
Tableau 20 : Description textuelle du cas d'utilisation « supprimer une partition » .....	94
Tableau 21 : Description textuelle du cas d'utilisation « envoyer une notification push » .....	96

# Introduction générale

Environ 10 à 15% des enfants naissent avec des troubles du développement. Beaucoup ont des problèmes de santé chroniques et permanents et nécessitent un soutien à long terme. Précisément au Canada, jusqu'à 850 000 enfants de moins de 14 ans vivent avec une déficience intellectuelle liée au développement cérébral et sont confrontés à des problèmes de mobilité, de langage, d'apprentissage, de socialisation ou de soins personnels qui ont une incidence sur leur qualité de vie.

C'est dans ce cadre que se situe le projet CHILD-BRIGHT. En effet, ce projet se concentre sur la mise en place de thérapies innovantes pour améliorer le développement du cerveau, en utilisant des techniques de stimulation cérébrale pour améliorer le bien-être émotionnel et social des enfants.

Dans ce contexte, la société 360Medlink est amenée à développer un jeu mobile destiné aux enfants atteints par des troubles du développement liés au cerveau, d'un côté, et une application web (CMS) destinée aux chercheurs du réseau CHILD-BRIGHT, afin de leur permettre de gérer le contenu du jeu.

Le but de ce stage est donc, de concevoir et mettre en place un module de gestion de contenu évolutif et flexible supportant tout changement de structure de données dans le temps.

Le présent rapport présentera les différentes étapes de la réalisation de ce projet et s'étalera sur quatre chapitres :

Le premier chapitre présentera le contexte général du projet, ses objectifs, les outils et technologies pour lesquels nous avons opté.

Le deuxième chapitre (Sprint 0) analysera les besoins fonctionnels et non fonctionnels, identifiera les différents acteurs impliqués et élaborera une étude conceptuelle globale du projet.

Les 3èmes, 4èmes et 5èmes chapitres détailleront respectivement les modules « gestion des utilisateurs », « gestion du contenu » et « Gestion des divisions, des notifications et des FAQ », leurs sprints backlogs, leurs diagrammes conceptuels et leurs implantations.

# **Chapitre 1 : Contexte général du projet**

## Introduction

Ce chapitre a pour objectif de situer notre projet dans son contexte général, à savoir, l'organisme d'accueil et le travail demandé. Nous commençons par une présentation de l'organisme d'accueil 360Medlink. Puis nous passons à la description de la mission qu'ils nous ont confiée en définissant le cadre du projet, ensuite la description de l'existant et ses limites. Nous concluons ce chapitre par les besoins de la société et la méthodologie adoptée tout au long de la réalisation du projet.

### 1. Présentation de l'organisme d'accueil

#### 1.1. Présentation

La société montréalaise 360Medlink Inc. est une entreprise mondiale de logiciels de pointe qui se consacre à la création d'innovations personnalisées pour téléphones et tablettes destinées aux plus grandes organisations de la santé et de la biotechnologie. Bénéficiant d'une vaste expérience en tant que professionnels du secteur pharmaceutique pour les sociétés Fortune 500, son équipe rassemble des personnes possédant des compétences exceptionnelles dans les domaines de la technologie et de la santé.

Son produit phare, TAVIERRX, est la première plate-forme d'infirmière virtuelle et application d'accompagnement du patient cliniquement validée, qui combine une éducation médicale interactive à la demande, l'observance du traitement, l'engagement des patients et le coaching humain. Ses autres solutions logicielles sont axées sur le marketing pharmaceutique, les affaires médicales et contribuent à optimiser les performances des essais cliniques mondiaux.



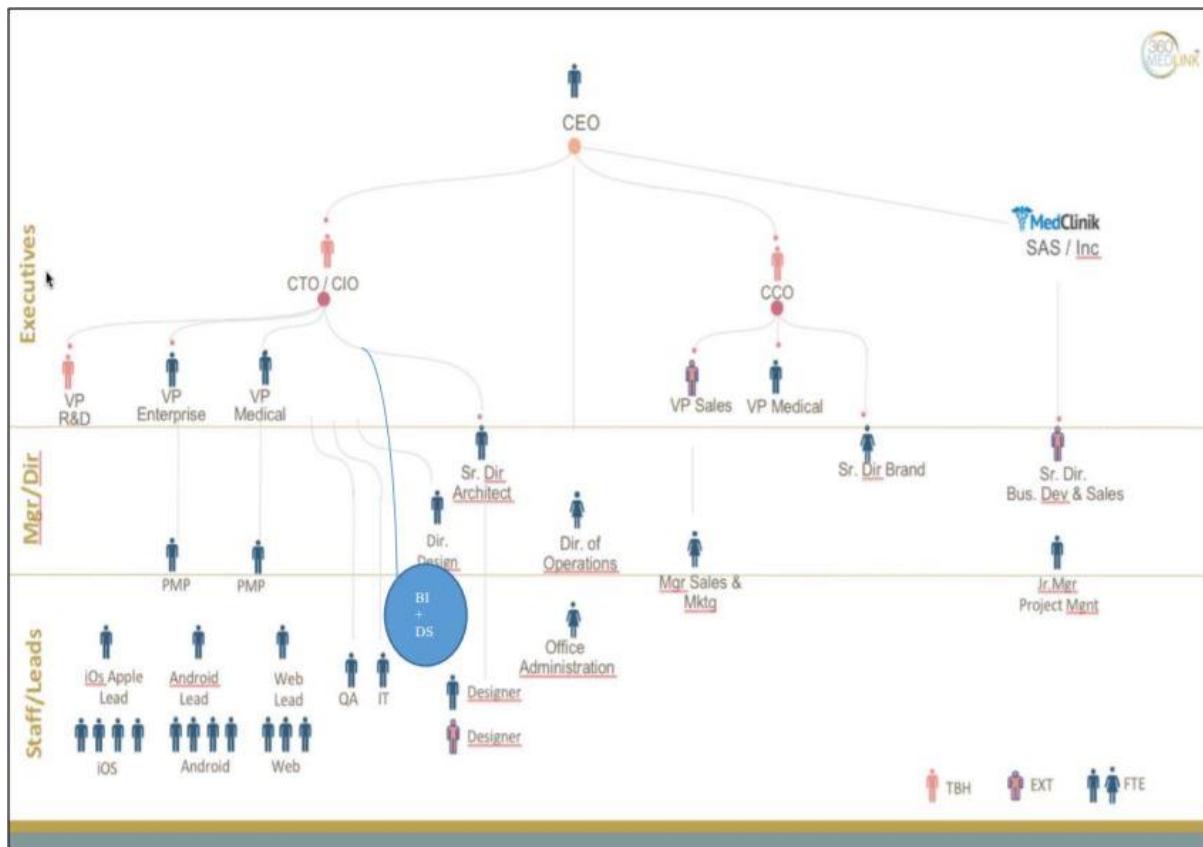
**Figure 1 : Logo de 360Medlink**

Bénéficiant d'une vaste expérience en tant qu'anciens professionnels du secteur pharmaceutique au sein de grandes entreprises, l'équipe 360Medlink possède une expertise unique en matière de technologie et d'environnement de soins de santé.

360Medlink est spécialisée également dans la communication scientifique (médicale et pédagogique) et dans le marketing. Ses applications technologiques sont utilisées pour la communication lors des expositions, en ligne et sur des appareils mobiles. Sa technologie de diffusion innovante lui permet de fournir des programmes qui garantissent la plus large audience possible, la meilleure valeur éducative ou promotionnelle et le meilleur retour sur investissement pour ses clients.

## 1.2. Organigramme

L'organigramme suivant représente la hiérarchie générale de 360Medlink :



**Figure 2: Organigramme de 360Medlink**

Dans le cadre de la préparation de notre projet de fin d'études, nous avons eu l'opportunité d'effectuer un stage en tant que développeur web au sein de l'équipe travaillant sur le projet CHILD-BRIGHT, composée de développeurs mobiles, designers et développeurs web.

## 2. Présentation générale du projet

Dans cette partie, nous définissons le cadre du projet. Ensuite, nous aborderons l'étude de l'existant ainsi que sa critique. Nous articulons après par la problématique et nous concluons par l'exposition des solutions proposées.

### 2.1. Cadre du projet

Le projet CHILD-BRIGHT, mené par l'institut de recherche du centre universitaire de santé McGill au Canada, a pour but de d'améliorer les vies des enfants atteints par des troubles du développement liés au cerveau. CHILD-BRIGHT se concentre sur le développement de thérapies innovantes pouvant inclure de nouvelles techniques de stimulation du cerveau, des combinaisons de médicaments, des applications mobiles et des jeux vidéo.

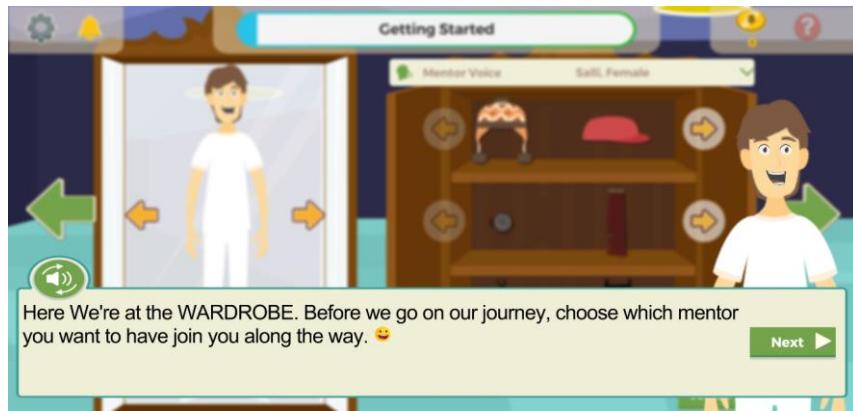
Les professionnels de la santé ont collaboré avec 360Medlink, les patients et les familles pour créer l'application *MyREADY Transition*. L'application *MyREADY Transition* est conçue pour les jeunes de 15 à 17 ans qui bénéficient de soins de santé pédiatriques et qui passeront un jour dans les soins de santé pour adultes. Elle est conçue pour les gens atteints par des troubles de développement liés au cerveau tel que la paralysie cérébrale, spina bifida, épilepsie, autisme ou syndrome d'alcoolisme foetal.

Les utilisateurs ont leur propre mentor pour les aider à se retrouver dans la ville du jeu mobile *MyREADY Transition*. En visitant différents endroits de la ville, en regardant des vidéos et en relevant des défis, l'application *MyREADY Transition* permet d'apprendre et de s'exercer à gérer des informations médicales et d'autres compétences afin d'aider les utilisateurs à se préparer pour les soins de santé pour adultes.

*MyREADY Transition* est représentée par des:

- **Chapitres** : Pour le projet CHILD-BRIGHT, il est prévu d'avoir 5 chapitres (Ceci peut être changé au fil du temps). Chaque chapitre est défini par un groupe de sessions.
- **Sessions** : Il est prévu d'avoir environ 4 sessions pour chaque chapitre. Chaque session est définie par un certain nombre de scènes.
- **Scenes** : Les scènes ce sont les étapes du jeu que l'utilisateur va suivre au fil du temps. Il est prévu d'avoir environ 45 scènes pour chaque session. Chaque scène se déroule dans une chambre/bâtiment.

La figure n° 3 montre un petit aperçu du jeu mobile *MyREADY Transition*.



**Figure 3 : Aperçu de MyReady Transition**

## 2.2. Problématique

Pour que l'équipe CHILD-BRIGHT de McGill puisse mener sa recherche sur les enfants atteints des troubles du développement liés au cerveau, elle a besoin d'avoir un contrôle total sur le jeu mobile et de gérer son contenu conformément à sa recherche scientifique.

D'un côté, les utilisateurs finaux du jeu mobile (les patients) ne vont pas tous avoir une même version. Les différents défis du jeu, les messages du Mentor, les vidéos et les notifications varient selon la langue utilisée par le patient, le type de sa maladie et le site auquel il appartient (Centre de recherche, hôpital ...).

D'un autre côté, l'équipe de chercheurs CHILD-BRIGHT doivent pouvoir mettre à jour le jeu mobile à n'importe quel moment si le besoin l'exige, et ceci sans aucune intervention de la part de l'équipe technique de 360Medlink.

De plus, l'équipe de chercheurs CHILD-BRIGHT a besoin de suivre et surveiller les différentes actions faites par les patients durant leurs avancements dans le jeu et d'avoir des statistiques.

## 2.3. Etude de l'existant

Parmi les causes qui poussent les entreprises à exprimer un besoin particulier est leur volonté de changer ou d'améliorer l'existant. Pour cela, l'étude de l'existant au sein de 360Medlink est cruciale pour analyser son environnement technique en vue d'en saisir la signification et émettre un regard critique permettant l'interprétation des besoins.

L'étude de l'existant nous a permis de dégager un certain nombre d'insuffisances et limites. En effet, le jeu mobile *MyReady Transition* dans sa première version contenait des éléments statiques (Boutons, labels, textes ...etc.). La moindre modification ou mise à jour exigeait une intervention de la part de l'équipe mobile de 360Medlink pour modifier le code et refaire le

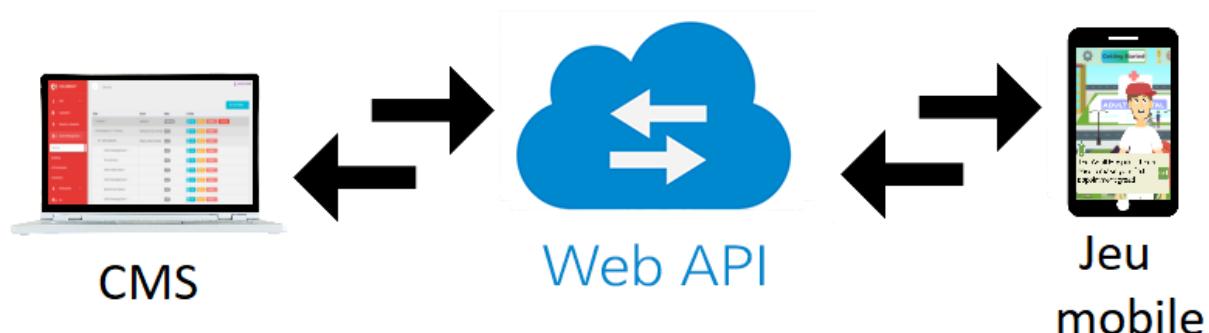
déploiement sur le serveur. Ce qui s'avérait peu pratique et une perte de temps. De plus, une seule version était disponible pour tous les utilisateurs (patients) avec une seule langue (Anglais). Un autre point négatif, c'est l'impossibilité d'avoir des statistiques bien précis sur les actions faites par les utilisateurs du jeu.

#### 2.4. Solution proposée

Après avoir défini la problématique, le challenge de ce projet est de concevoir et élaborer une application pérenne qui comble les insuffisances énumérées précédemment et qui assure d'une part une meilleure dynamisation des données du jeu mobile *MyReady Transition* et d'autre part un module de monitoring des usagers évolutif et flexible supportant tout changement de structure de données dans le temps.

Notre application doit non seulement manipuler le contenu du jeu, mais aussi de gérer ses utilisateurs, les sites auxquels ils peuvent appartenir et de leurs envoyer des notifications instantanément. Les données qui s'afficheront aux utilisateurs finaux (patients) doivent prendre en compte trois paramètres importants : Le site auquel il appartient, sa langue et sa maladie.

Afin d'assurer l'interaction entre le CMS et le jeu mobile, des API web doivent être mis en place.



**Figure 4 : Architecture de la solution**

### 3. Méthode de travail

La finalisation du projet dans les délais de livraison est le souci majeur de chaque équipe de développement d'un logiciel. L'un des problèmes les plus fréquemment affrontés lors de la construction du logiciel est la mal spécification et le changement brusque des besoins. Cela peut influencer non seulement l'équipe de développement en créant un environnement de stress, mais aussi le temps consacré pour la réalisation du projet et donc un retard de livraison et des

dépassements budgétaires. C'est pour cette raison, afin d'éviter ces situations critiques, notre choix s'est orienté vers la méthodologie agile pour la gestion de notre projet.

### **3.1. Les méthodes agiles**

Les méthodes de développement agile, appelé aussi développement adaptatif, sont des méthodes axées sur les personnes avec pour objectif, la satisfaction du client en s'appuyant sur la réalisation d'un logiciel entièrement fonctionnel tout au long de sa fabrication et qu'ont été développées à l'origine pour tenter de résoudre l'équation infernale des projets informatiques de grandes ampleurs. Les 4 valeurs de l'Agilité sont :

- L'équipe : Les individus et leurs interactions avant les processus et les outils.
- L'application : Des fonctionnalités opérationnelles avant la documentation.
- La collaboration : Collaboration avec le client plutôt que contractualisation des relations.
- L'acceptation du changement : Adaptation au changement plutôt que conformité aux plans.

### **3.2. Etude comparative des méthodes agiles**

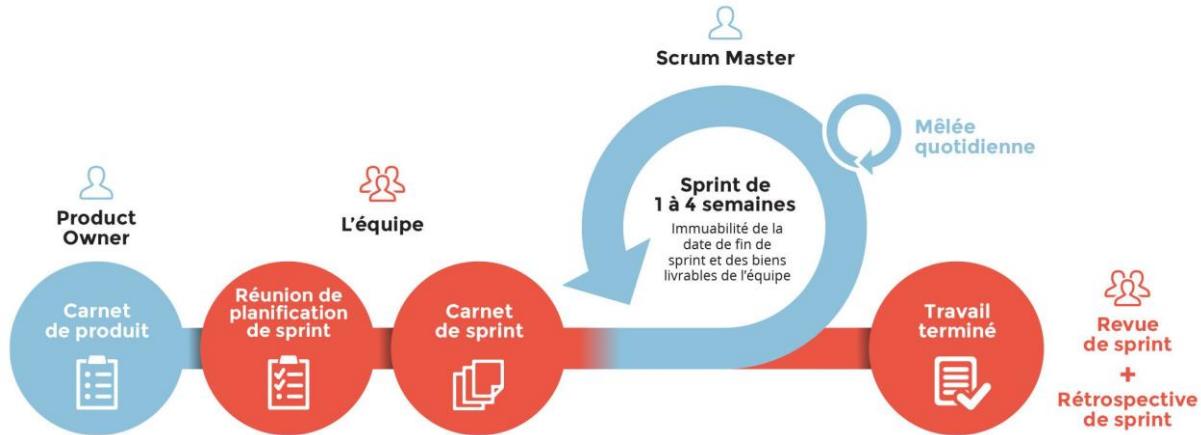
Méthode	Description	Avantages	Inconvénients
FDD	FDD regroupe les meilleures pratiques qui sont toutes dictées par ce qui est important pour le client. Cela signifie que les développeurs se concentrent sur les fonctionnalités que le client valorise.	<ul style="list-style-type: none"> <li>- Prend en charge plusieurs équipes travaillant en parallèle</li> <li>- Tous les aspects d'un projet sont suivis par une fonctionnalité</li> <li>- S'adapte bien aux grandes équipes/projets</li> </ul>	<ul style="list-style-type: none"> <li>- Favorise la propriété individuelle du code à la propriété partagée</li> <li>- Les itérations ne sont pas bien définies par le processus comme les autres méthodologies agiles</li> </ul>
XP	Ensemble de bonnes pratiques de développement	<ul style="list-style-type: none"> <li>-Itératif et simple à mettre en œuvre.</li> </ul>	<ul style="list-style-type: none"> <li>- Ne couvre pas les phases en amont et</li> </ul>

	(travail en équipe, transfert de compétences...)	<ul style="list-style-type: none"> <li>- Fait une large place aux aspects techniques.</li> <li>- Innovant : programmation en duo</li> </ul>	<ul style="list-style-type: none"> <li>en aval au développement</li> <li>- Assez flou dans sa mise en œuvre</li> </ul>
Scrum	<ul style="list-style-type: none"> <li>- Basé sur des itérations appelées des sprints.</li> <li>- Utilisé pour le développement agile</li> </ul>	<ul style="list-style-type: none"> <li>- Simplicité des processus. -Règles définies clairement.</li> <li>- Augmentation de productivité.</li> <li>- Chaque équipe a son lot de responsabilité</li> <li>- Amélioration de la communication</li> </ul>	<ul style="list-style-type: none"> <li>- Nécessite une équipe engagée sinon le projet va échouer.</li> <li>- Le processus rapide et répétable met une forte pression sur l'équipe de développement.</li> </ul>

**Tableau 1 : Tableau comparatif des méthodes agiles**

### 3.3. La méthode adoptée : SCRUM

Le développement de logiciels avec la méthode SCRUM repose sur les développements itératifs à un rythme constant de 2-4 semaines. L'avantage le développement de logiciels Agile avec la méthode SCRUM consiste à réduire la documentation au minimum afin de gagner en productivité pour nous permettre de sauvegarder l'historique des décisions prises sur le projet et d'effectuer facilement des interventions sur le logiciel quand il va dans la phase maintenance.



**Figure 5 : Cycle de vie Scrum [1]**

## Les sprints

Le cycle de vie Scrum est rythmé par des itérations de quelques semaines, les sprints.

## Le product backlog

Le référentiel des exigences initiales est dressé et hiérarchisé avec le client. Il constitue ce que l'on nomme le product backlog. Il ne doit pas nécessairement contenir toutes les fonctionnalités attendues dès le début du projet, il va évoluer durant le projet en parallèle des besoins du client.

## User Story

Les fonctionnalités décrites portent le nom de User Stories et sont décrites en employant la terminologie utilisée par le client.

## Le sprint planning meeting

On organise, avant chaque sprint, une réunion de planification, le sprint planning meeting. Ce planning sélectionne dans le product backlog les exigences les plus prioritaires pour le client. Elles seront développées, testées et livrées au client à la fin du sprint. Elles constituent le sprint backlog, un sous ensemble du product backlog.

## La mêlée

Au cours du sprint, il est organisé, chaque jour, une réunion d'avancement (environ 15 min) avec tous les membres de l'équipe afin de s'assurer que les objectifs du sprint seront tenus, c'est le Scrum ou mêlée. Chaque jour, après la réunion Scrum, le Scrum Master maintient un graphique appelé sprint burndown chart. Ce graphique donne une très bonne vision de ce qui a

été fait et du rythme de travail de l'équipe. Il permet également d'anticiper si toutes les stories du Sprint Backlog seront terminées à la fin de l'itération ou non.

### 3.4. Equipe

Rôle	Missions	Acteur
Propriétaire de produit	Présentation des caractéristiques et des fonctionnalités du produit à développer et approbation du produit à livrer	Via-Dufresne Ley Alicia
Scrum master	Supervision de l'avancement du projet et des activités de l'équipe et éliminer les obstacles.	Saviana Stéphane
Equipe de développement	<ul style="list-style-type: none"> <li>• Conception de la structure de données</li> <li>• Mise en place des entités de la base de données</li> <li>• Implémentation des Services Web</li> <li>• Implémentation des interfaces utilisateurs</li> <li>• Consommation et intégration des Service Web</li> </ul>	Cheikh Ahmed
	Architecte logiciel	Khayati Nadhir
	Développeur full-stack	Abdelmoumen Samia
	Ingénieur en assurance de qualité	Jayaraman Raghuram
	Designer	Jenaina Araujo
	Développeur mobile (Unity)	Gharbi Ayoub – Ben Dhifallah Alaeddine
	Consultant Business Intelligence	Bouzid Taher
	Ingénieur DevOps	Rajasekar Vignesh

Tableau 2 : Equipe du projet

## 4. Choix technologique

La sélection d'une pile technologique pertinente est cruciale pour la réussite d'un projet, mais faire le bon choix peut s'avérer un défi. Il y a plusieurs critères à prendre en considération avant de faire le bon choix, à savoir : La taille du projet, la performance, l'évolutivité, la maintenance, la documentation...

### 4.1. Laravel 5.5

Laravel est un framework bien documenté <https://laravel.fr/>, et un framework bien documenté qui permet de faire gagner du temps face à ces concurrents. Il possède une communauté importante, ce qui nous donne la possibilité de trouver rapidement une réponse en cas de doute, c'est aussi le gage d'une certaine pérennité pour le framework, plus les développeurs l'utiliseront, plus longtemps il continuera d'exister.

Laravel est aussi très complet sans trop l'être, et grâce à sa documentation bien fournie, la durée d'apprentissage reste contenue. Son plus gros point fort est sa simplicité, le code que l'on écrit avec Laravel est toujours simple, lisible, concis et agréable à lire. On y retrouve les standards de programmation ainsi que la possibilité d'ajouter facilement des extensions. [2]

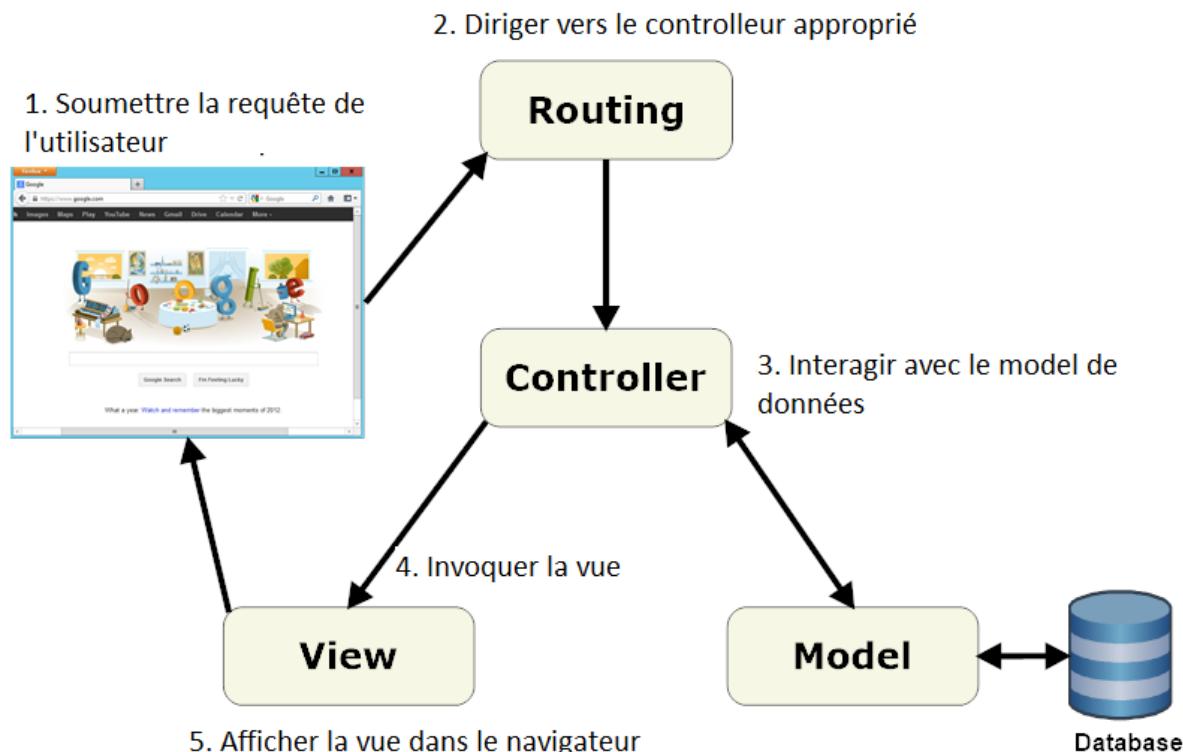


**Figure 6 : Logo de Laravel**

Nous avons choisi Laravel 5.5 comme Framework pour de multiples raisons :

- **Techniques d'autorisation:** Laravel simplifie très largement la mise en œuvre des techniques d'authentification, et fournit un moyen simple d'organiser la logique d'autorisation et de contrôler l'accès aux ressources.
- **Artisan :** Laravel fournit un outil intégré à la ligne de commande appelé Artisan. Cet outil nous permet d'effectuer la majorité des tâches de programmation répétitives et fastidieuses que la plupart des développeurs évitent d'effectuer manuellement.

- **Prise en charge de l'architecture MVC :** Ceci garantit la clarté entre la logique et la présentation. MVC contribue à améliorer les performances, permet une meilleure documentation et intègre de nombreuses fonctionnalités. Voici comment fonctionne le MVC pour Laravel. :



**Figure 7 : Modèle MVC**

- **Sécurité :** Laravel fournit plusieurs mécanismes pour sécuriser une application web, tel que : Le cryptage des mots de passe, la protection des routes et l'utilisation des instructions SQL préparées qui rendent les attaques par injection inimaginables.
- **Migration de la base de données :** Un problème pour les développeurs est de maintenir la base de données synchronisée entre les machines de développement. Avec les migrations de bases de données Laravel, c'est extrêmement facile.

## 4.2. Angular 5.2

C'est un framework front-end côté client. Il est utilisé pour créer des applications web très puissantes. Angular est souvent considéré comme une sorte de langage HTML dynamique.

De plus, il fait partie de la technologie ‘MEAN’ (MongoDB, Express, Angular, Node) stack qui est considérée comme une combinaison très puissante pour créer des applications complètes.

Enfin, il a été créé et est toujours maintenu par Google, dont la réputation n'est plus à faire.



**Figure 8 : Logo de Angular**

Nous avons utilisé Angular 5.2 comme framework front-end pour les raisons suivantes :

- **Angular nous présente non seulement les outils, mais également les modèles de conception permettant de construire notre projet de manière maintenable.** Lorsqu'une application Angular est correctement conçue, nous ne nous retrouvons pas avec un désordre de classes et de méthodes difficiles à modifier et encore plus difficiles à tester. Le code est structuré de manière pratique et nous n'aurons pas besoin de passer beaucoup de temps pour comprendre ce qui se passe.
- **C'est JavaScript, mais c'est mieux.** Angular est construit avec TypeScript, qui à son tour s'appuie sur JS ES6. Nous n'avons pas besoin d'apprendre un nouveau langage, mais nous recevons toujours des fonctionnalités telles que le typage statique, les interfaces, les classes, les namespaces, les décorateurs, etc.
- **Pas besoin de réinventer le vélo.** Avec Angular, nous disposons déjà de nombreux outils pour commencer à concevoir l'application tout de suite. Nous avons des directives pour donner un comportement dynamique aux éléments HTML. Nous pouvons activer

les formulaires à l'aide de FormControl et introduire diverses règles de validation. Nous pouvons facilement envoyer des requêtes HTTP asynchrones de différents types.

- **Les composants sont découplés.** Angular s'est efforcé de supprimer le couplage étroit entre les différents composants de l'application. L'injection se fait dans le style NodeJS et nous pouvons facilement remplacer divers composants.
- **Le test est au cœur.** Angular est conçu pour des tests approfondis et prend en charge les tests unitaires et de bout en bout.
- **La popularité d'Angular.** Angular est activement entretenu et possède une grande communauté.

### 4.3.      Swagger 5.5

La plupart des applications Web prennent en charge les API RESTful, mais contrairement aux API SOAP, les API REST s'appuient sur des méthodes HTTP et ne disposent pas d'un équivalent WSDL (Web Services Description Language) pour définir les structures de requête et de réponse entre consommateurs et fournisseurs. Sans un service contractuel adéquat, de nombreux fournisseurs d'API REST utilisent des documents Microsoft Word ou des pages wiki pour documenter l'utilisation de l'API. Ces formats peuvent rendre la collaboration et le contrôle de version de document difficiles, en particulier pour les applications comportant de nombreuses API ou ressources, ou lorsque les API sont en cours de développement itératif. Ces types de documents sont également plus difficiles à intégrer dans une application de test automatisée.

Le framework Swagger open source aide à résoudre ces problèmes pour les consommateurs et les développeurs d'API. La structure fournit la spécification OpenAPI permettant de créer une documentation facile à utiliser sur l'API RESTful au format JSON ou YAML. Les documents Swagger peuvent être traités par différents langages de programmation et peuvent être intégrés au contrôle de source pour la gestion des versions au cours du cycle de développement du logiciel.

swagger https://api.childbright.360medlink.net/docs/api-docs.json Explore

## ChildBright API Documentation v1.0.0

[ Base URL: / ]  
https://api.childbright.360medlink.net/docs/api-docs.json

---

### auth

- POST** /api/auth/activation/byType/{type} Activate a user base on the type
- GET** /api/auth/app/init retrieve the init elements
- POST** /api/auth/login Returns an access token and refresh token
- POST** /api/auth/refresh Refresh access token for a user
- POST** /api/auth/logout Destroys access token and refresh token

**Figure 9 : Interface de la documentation des API via Swagger**

## 4.5. MySQL 5.6

MySQL est système de gestion de base de données open-source, puissant et gratuit, existant depuis des années. Il est très stable et a une grande communauté qui aide à le maintenir, le déboguer et le mettre à jour.

## 5. Outils utilisés

Etant donné que l'équipe travaillant sur le projet CHILD-BRIGHT est divisée entre ceux qui travaillent en Tunisie, et ceux qui travaillent au Canada, l'utilisation des outils de communication et de collaboration s'impose, pour aboutir à des meilleurs résultats et gagner du temps. Voici donc la liste des outils que nous avons utilisée durant ce projet :

### 5.1. Jira

Jira est un outil de gestion de projet qui permet de planifier les sprints, créer des user stories, répartir les tâches sur l'ensemble de l'équipe et suivre les bugs.

The screenshot shows the JIRA interface for the 'CHILD BRIGHT' project. On the left, there's a sidebar with various project management options like Board, Backlog, Active sprints, Reports, Releases, Issues and filters, Pages, Components, Timesheet, and Add-ons. The main area displays an 'All sprints' board with a grid of tasks. The columns represent different stages of development: OPEN, IN PROGRESS, CODE REVIEW, READY TO TEST, REOPENED, DONE, READY TO DEPLOY, and CLOSED. In the OPEN column, there are several issues, one of which is expanded to show its details: '[InterfaceLabel - Create] InterfaceLabel CHILD...-1421'. Another issue in the same column is '[Notification Automatic] Implement the CHILD...-1447'. Below the OPEN column, there's a section for 'Other Issues 87 issues' with more task cards.

**Figure 10 : Plateform JIRA**

## 5.2. Bitbucket

Bitbucket est une solution de gestion de référentiel (repository) Git conçue pour les équipes professionnelles. Il offre un emplacement central pour gérer les dépôts git, collaborer sur un code source et guider ses utilisateurs tout au long du processus de développement. Il fournit des fonctionnalités impressionnantes qui incluent [4] :

- Contrôle d'accès pour restreindre l'accès à votre code source
- Contrôle du workflow pour appliquer un workflow de projet ou d'équipe
- Extraction des requêtes avec des commentaires en ligne pour la collaboration lors de la révision du code
- Intégration Jira pour une traçabilité complète du développement

## 5.3. Slack

Afin d'assurer une bonne communication entre les différents membres de l'équipe, nous avons opté pour Slack. Slack nous permet de :

- Collaborer en ligne comme on le ferait en personne.
- Réunir les bonnes personnes et les informations en un seul endroit.
- Communiquer efficacement, rester en contact et faire les choses plus rapidement. [5]

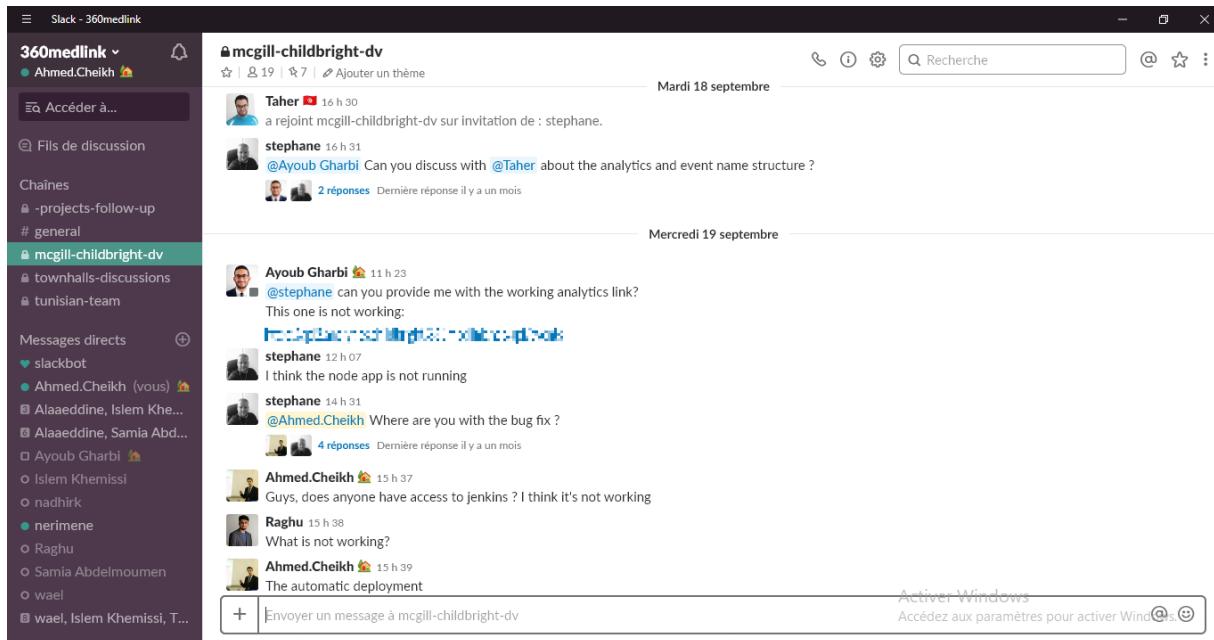


Figure 11 : Interface de Slack

## 5.4. Jenkins

Jenkins offre un moyen simple de configurer un environnement d'intégration continue ou de livraison continue pour presque toutes les combinaisons de langages et de référentiels de code source utilisant des pipelines, ainsi que pour automatiser d'autres tâches de développement courantes. Jenkins n'élimine pas le besoin de créer des scripts pour des étapes individuelles, mais il nous offre un moyen plus rapide et plus robuste d'intégrer toute notre chaîne d'outils de construction, de test et de déploiement que nous ne pouvons le faire nous-même.

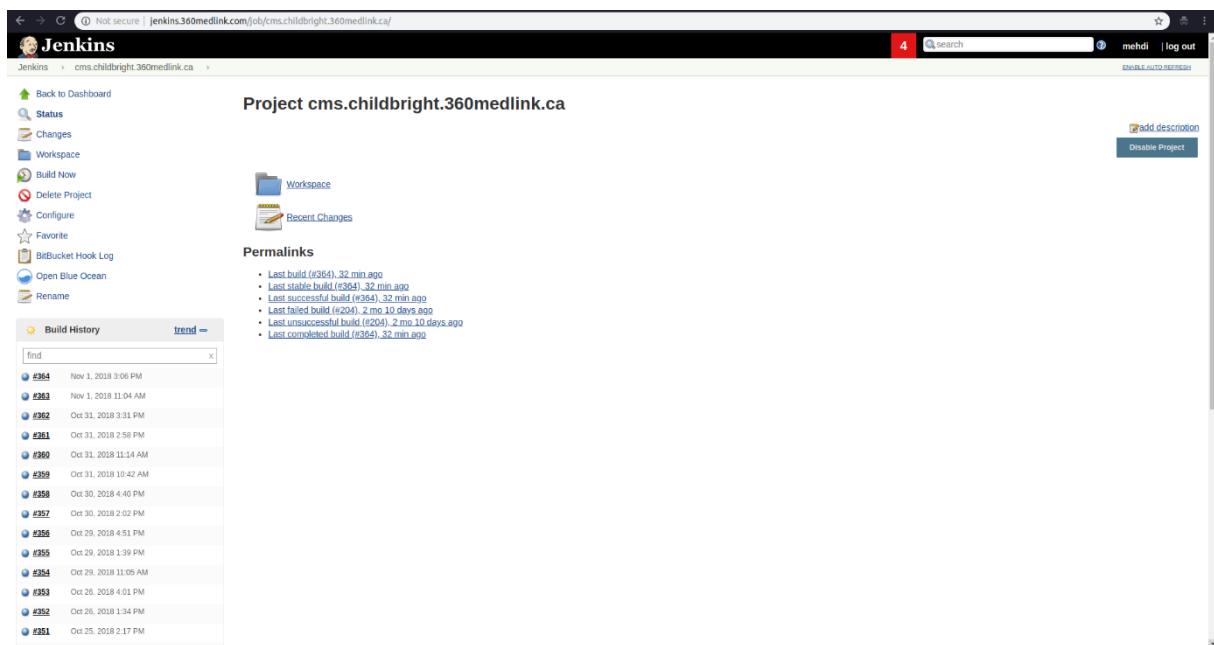


Figure 12 : Interface de Jenkins

## 5.5. Amazon S3

Amazon S3 (Simple Storage Service) est un service Web de cloud computing proposé par Amazon Web Services (AWS) d'Amazon. Amazon S3 fournit un stockage d'objet via des interfaces de services Web (REST, SOAP et BitTorrent).

Le but d'utiliser Amazon S3 dans notre projet est de stocker tous les fichiers nécessaires pour le fonctionnement du jeu mobile MyReady Transition (Fichiers audio, vidéos, images...)

## 5.6. Firebase Cloud Messaging (FCM)

Firebase Cloud Messaging (FCM) est une solution de messagerie cross-platform qui permet de livrer des messages de manière fiable, gratuitement.

À l'aide de FCM, nous pouvons notifier une application cliente qu'un nouveau courrier électronique ou d'autres données sont disponibles pour la synchronisation. Nous pouvons également envoyer des messages de notification pour encourager le réengagement et la rétention des utilisateurs. Pour les cas d'utilisation tels que la messagerie instantanée, un message peut transférer une charge utile maximale de 4 Ko vers une application client. [6]

## 5.7. Redis

Redis est un magasin de structure de données clé/valeur en mémoire open source rapide. Redis propose un ensemble de structures de données en mémoire polyvalentes qui vous permet de créer facilement un large éventail d'applications personnalisées. Les principaux cas d'utilisation de Redis comprennent la mise en cache, la gestion des sessions, la fonctionnalité pub/sub et les classements. Il s'agit du magasin clé/valeur le plus populaire à l'heure actuelle. Il est distribué sous licence BSD, écrit en code C optimisé et prend en charge plusieurs langages de développement. Redis est l'acronyme de REmote DIctionary Server.

# 6. Répartition des sprints

Les sprints sont répartis comme l'indique le tableau 3 :

Sprint	Description	Durée
Sprint 0 : Analyse et spécification des besoins	- Partager une vision claire du projet entre	3 semaines

	<p>les membres de l'équipe.</p> <ul style="list-style-type: none"> <li>- Produire une première version du backlog du produit.</li> <li>- Réfléchir et se mettre d'accord sur une architecture globale du projet.</li> <li>- Préparer l'environnement du développement.</li> </ul>	
Sprint 1 : Gestion des utilisateurs	<ul style="list-style-type: none"> <li>- Développer le module pour gérer les utilisateurs d'une part, et assurer la sécurisation de l'application par une authentification d'autre part.</li> </ul>	6 semaines
Sprint 2 : Gestion du contenu	<ul style="list-style-type: none"> <li>- Développer le module pour gérer le contenu du jeu mobile.</li> </ul>	8 semaines
Sprint 3 : Gestion des sites, des notifications et des FAQs	<ul style="list-style-type: none"> <li>- Développer le module pour gérer les sites, les FAQs et l'envoi des notifications.</li> </ul>	6 semaines

**Tableau 3 : Répartition des sprints**

## Conclusion

Dans ce premier chapitre, nous avons situé notre projet dans son contexte, en présentant l'organisme d'accueil et le contexte du projet. Nous nous sommes aussi intéressés à l'étude de l'existant et nous l'avons critiqué pour pouvoir déterminer nos objectifs d'une façon claire et objective. D'un autre côté nous avons opté pour la méthodologie Scrum qui s'avère la plus appropriée pour notre projet grâce aux avantages présentés par cette dernière.

Le chapitre suivant sera consacré à l'analyse et spécification des besoins.

# **Chapitre 2 : Analyse et spécification des besoins (Sprint 0)**

## Introduction

L'analyse fonctionnelle est une démarche qui consiste à la caractérisation et la hiérarchisation des différentes fonctions d'un projet. Étant considérée comme une étape essentielle dans le cycle de développement d'une application, cette phase permet un passage d'une vue extérieure du comportement d'un système à une vue intérieure permettant d'expliquer son fonctionnement. Les besoins présentés avec des diagrammes de cas d'utilisation sont regroupés dans des diagrammes de séquence système pour mieux détailler le projet et amoindrir la complexité.

### 1. Identification des acteurs

Un acteur est une entité externe au système. Il représente une personne ou un autre système informatique qui attend un ou plusieurs services offerts par une interface d'accès.

Dans notre contexte, l'acteur principal est le chercheur/administrateur du réseau CHILD-BRIGHT qui va utiliser le CMS afin de gérer le contenu du jeu mobile *MyReady Transition* à sa guise.

### 2. Identification des besoins fonctionnels et non fonctionnels

Les besoins recueillis ont été répartis en deux catégories: les besoins fonctionnels et les besoins non fonctionnels.

#### 2.1. Identification des besoins fonctionnels

Les besoins fonctionnels ou besoin métiers sont les tâches exécutées par le système afin de satisfaire une requête donnée. Notre application couvre les besoins fonctionnels suivants :

- Authentification : L'utilisateur doit pouvoir s'authentifier sur l'application web via un email et un mot de passe.
- Gestion des utilisateurs : Trois types d'utilisateurs à gérer : Administrateurs, chercheurs et patients.
- Gestion des segments : Versions, chapitres et sessions.
- Gestion des langues disponibles pour une version du jeu mobile / Ajouts des traductions
- Gestion des sites : Les sites ce sont les établissements auxquels les patients peuvent appartenir (Centre de recherche, hôpital ...)
- Envoi des notifications : On peut distinguer deux type de notifications, les notifications push qui seront instantanés, et les notifications automatiques qui seront préprogrammés à l'avance.

- Gestion des trophées/plaques : Lorsque l'utilisateur réussit à accomplir un chapitre du jeu, il reçoit un trophée. De même, lorsqu'il réussit à accomplir une session, il reçoit une plaque.
- Gestion des localisations : Les localisations ce sont les bâtiments du jeu, les chambres et leurs équipements.
- Gestion des labels des interfaces.
- Gestion des FAQ (frequently asked questions).
- Gestion des dictionnaires.

## 2.2. Identification des besoins non fonctionnels

- Ergonomie :

Etant donné que notre module va être utilisé par des non connaisseurs d'informatique, les interfaces doivent être réalisées de telle façon que son utilisateur s'y trouve facile à manipuler.

- Sécurité :

L'application doit respecter la confidentialité des données. L'échange des données entre utilisateur et application Web constitue le principe fondamental du Web 2.0.

Pour assurer la sécurité, nous avons protégé l'application par l'utilisation des jetons JWT (Json Web Token). Le principe des jetons est décrit dans la figure 13:

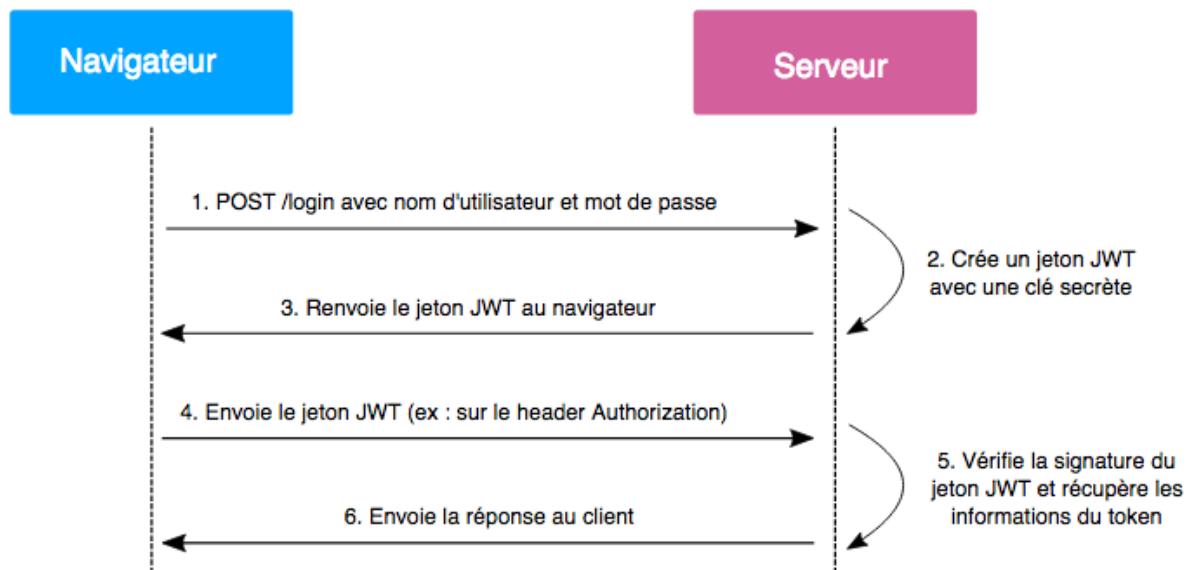


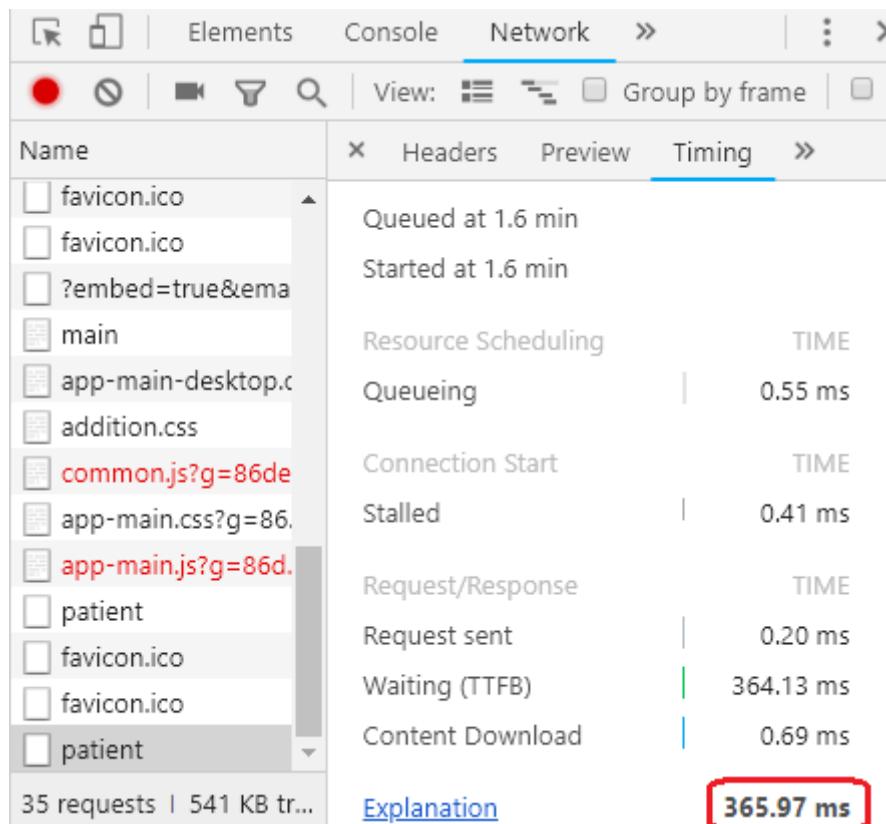
Figure 13 : Principe des jetons JWT

- **La performance :**

La rapidité de réponse est l'un des piliers fondamentaux de l'UX (expérience utilisateur). Il est donc essentiel de donner à la personne naviguant sur notre application web l'accès à l'information qu'il désire de la façon la plus simple, rapide et efficace possible.

C'est pourquoi nous avons choisi que notre application soit mono-page, c'est-à-dire que l'utilisateur peut naviguer entre les pages sans qu'il y ait un rafraîchissement. En effet, sur une application mono-page, après le chargement initial de la page, seules les données sont demandées par le serveur (ou envoyées au serveur), ce qui prend beaucoup moins de temps et de bande passante que l'envoi constant de HTML.

La figure 14 montre le temps de réponse du serveur pour la liste des patients (0.36 secondes), ce qui s'avère assez rapide.



**Figure 14 : Temps de réponse du serveur pour la liste des patients**

Un autre moyen pour améliorer la performance de notre application est l'utilisation de la notion du **Lazy Loading**. En effet, étant donné que notre application contient un nombre important de modules, les charger tous en même temps peut entraîner des problématiques lors de son démarrage. Avec le lazy loading, un module n'est chargé que lorsque sa route est sollicitée.

Ceci nous permet d'améliorer le temps de chargement de l'application, améliorant ainsi l'expérience utilisateur.

- **La maintenabilité :**

Un code maintenable signifie simplement « code facile à modifier ou à étendre ». La facilité de maintenance repose sur un code soigneusement construit et facile à lire afin de localiser le composant spécifique relatif à une demande de modification donnée.

Afin d'assurer la maintenabilité de notre code source, nous avons eu recours à **compodoc**, un outil intéressant qui nous permet de générer une documentation complète pour notre application Angular. La figure 15 illustre un exemple de documentation du module `SitesModule`, générée par compodoc.

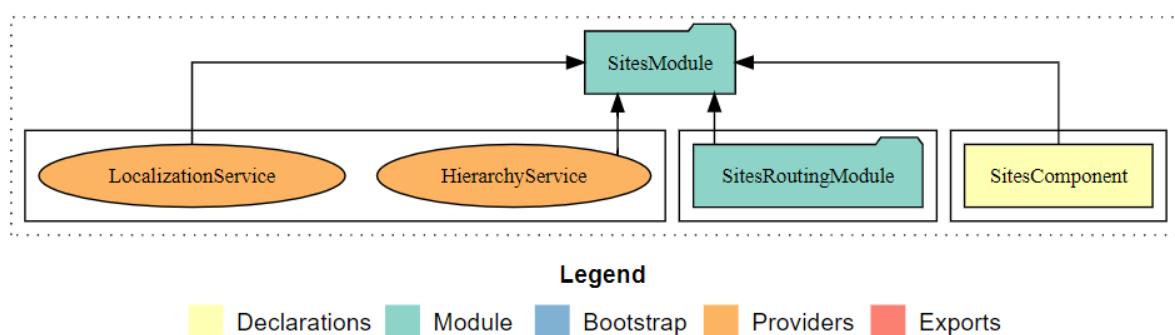


Figure 15 : Exemple de documentation générée par compodoc

### 3. Backlog de produit

Le backlog de produit dans Scrum est une liste de fonctionnalités hiérarchisées, contenant de brèves descriptions de toutes les fonctionnalités souhaitées dans le produit. Lors de l'application de Scrum, il n'est pas nécessaire de démarrer un projet avec un long effort initial pour documenter toutes les exigences. En règle générale, une équipe Scrum et son propriétaire de produit commencent par écrire tout ce à quoi ils peuvent penser pour hiérarchiser leurs backlogs de manière agile. Ce product backlog agile est presque toujours plus que suffisant pour un premier sprint. Le product backlog Scrum est alors autorisé à croître et à évoluer à mesure que nous en apprenons davantage sur le produit et ses clients.

Un backlog Scrum typique comprend les différents types d'éléments suivants:

- Fonctionnalités

- Bogues
- Travail technique
- Acquisition de connaissances

De loin, le principal moyen utilisé par une équipe Scrum pour exprimer les fonctionnalités du backlog de produit consiste en des user stories, des descriptions courtes et simples de la fonctionnalité souhaitée, présentées du point de vue de l'utilisateur.

Le Backlog du produit comprend les champs suivants :

- IDF : un identifiant de fonctionnalité.
- Fonctionnalité : un ensemble de user stories et qui construit un service, observable de l'extérieur.
- Histoire Utilisateur « User story » : une phrase décrivant la fonctionnalité désirée par le client.
- ID-US : un nombre unique et auto-incrémenté pour chaque histoire utilisateur.
- Business Value (B.V) : la valeur métier qui dirige la priorisation du développement des histoires utilisateurs suivant les attentes et les besoins du client, allant de 0 à 100 (étant 100 le plus important)
- Story point (S.P) : Les Story Points permettent de mesurer une tâche en termes d'effort d'une équipe concrète, au lieu de simplement faire des estimations en jour-hommes, qui fluctuent selon les personnes. On utilise souvent, et on utilisera ici, la suite de Fibonacci (1, 2, 3, 5, 8, 13, 21, ...) pour décliner la quantité d'effort à fournir.

Le tableau 4 présente le Backlog de notre produit :

IDF	Fonctionnalité	ID-US	User Story	B.V	S.P
<b>F1</b>	S'authentifier	1.1	En tant que administrateur, je souhaite m'authentifier au CMS via un e-mail et un mot de passe.	100	8
		1.2	En tant que chercheur, je souhaite m'authentifier au		

			CMS via un e-mail et un mot de passe.		
		1.3	En tant que patient, je souhaite m'authentifier à l'application mobile via un e-mail et un mot de passe.		
F2	Gérer les utilisateurs	2.1	En tant que administrateur, je souhaite ajouter, lister, modifier, afficher les détails et supprimer un utilisateur (Administrateur, chercheur ou patient).	100	13
		2.2	En tant que chercheur, je souhaite ajouter, lister, modifier, afficher les détails et supprimer un patient.		
F3	Gérer les segments	3.1	En tant que administrateur, je souhaite lister, ajouter, modifier, supprimer, dupliquer, afficher les détails et publier les versions.	100	21
		3.2	En tant que chercheur, je souhaite lister, afficher les détails des versions.		
		3.3	En tant que administrateur, je souhaite lister, ajouter, modifier, supprimer, afficher les détails des chapitres pour une version donnée.		

		3.4	En tant que chercheur, je souhaite lister, afficher les détails des chapitres pour une version donnée.		
		3.5	En tant que administrateur, je souhaite lister, ajouter, modifier, supprimer, afficher les détails des sessions pour un chapitre donné.		
		3.6	En tant que chercheur, je souhaite lister, afficher les détails des sessions pour un chapitre donné.		
		3.7	En tant que administrateur, je souhaite lister, ajouter, modifier, supprimer, afficher les détails des scenes pour une session donnée.		
		3.8	En tant que chercheur, je souhaite lister, afficher les détails des scenes pour une session donnée.		
F4	Gérer les langues	4.1	En tant que administrateur, je souhaite lister, activer/désactiver les langues disponibles.	20	2
		4.2	En tant que chercheur, je souhaite lister, activer/désactiver les langues disponibles.		

F5	Gérer les sites	5.1	En tant que administrateur, je souhaite lister, ajouter, modifier, supprimer, afficher les détails des divisions.	60	5
		5.2	En tant que chercheur, je souhaite lister, afficher les détails des divisions.		
		5.3	En tant que administrateur, je souhaite lister, ajouter, modifier, supprimer, afficher les détails des régions pour une division donnée.		
		5.4	En tant que chercheur, je souhaite lister, afficher les détails des régions pour une division donnée.		
		5.5	En tant que administrateur, je souhaite lister, ajouter, modifier, supprimer, afficher les détails des sites pour une région donnée.		
		5.6	En tant que chercheur, je souhaite lister, afficher les détails des sites pour une région donnée.		
F6	Gérer les localisations	6.1	En tant que administrateur, je souhaite lister, ajouter, modifier, supprimer, afficher les détails des bâtiments.	40	5

		6.2	En tant que chercheur, je souhaite lister, afficher les détails des bâtiments.		
		6.3	En tant que administrateur, je souhaite lister, ajouter, modifier, supprimer, afficher les détails des chambres pour un bâtiment donné.		
		6.4	En tant que chercheur, je souhaite lister, afficher les détails des chambres pour un bâtiment donné.		
		6.5	En tant que administrateur, je souhaite lister, ajouter, modifier, supprimer, afficher les détails des équipements pour une chambre donnée.		
		6.6	En tant que chercheur, je souhaite lister, afficher les détails des équipements pour une chambre donnée.		
F7	Envoyer des notifications	7.1	En tant que administrateur, je souhaite envoyer des notifications push aux appareils des patients appartenant à un site bien donné.	60	8
		7.2	En tant que administrateur, je souhaite programmer des notifications automatiques.		

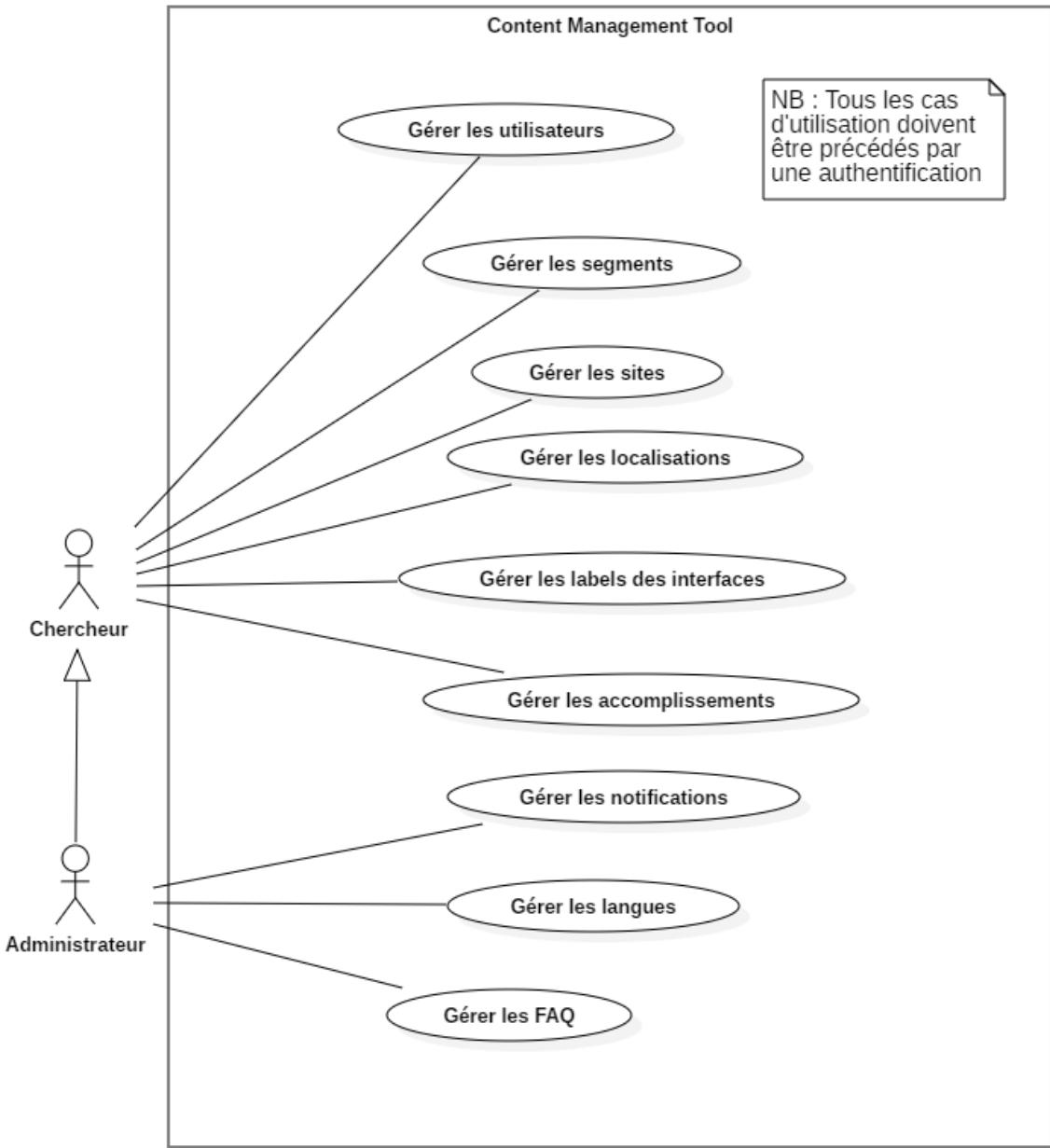
F8	Gérer les labels des interfaces	8.1	En tant que administrateur, je souhaite lister, ajouter, modifier, supprimer, activer/désactiver, afficher les détails des labels des interfaces de l'application mobile.	60	5
		8.2	En tant que chercheur, je souhaite lister, afficher les détails des labels des interfaces de l'application mobile.		
F9	Gérer les FAQs	9.1	En tant que administrateur, je souhaite ajouter, lister, modifier les questions/réponses.	30	5
F10	Gérer les accomplissements	9.1	En tant que administrateur, je souhaite ajouter et modifier et afficher les trophées pour un chapitre donné.	30	3
		9.2	En tant que chercheur, je souhaite afficher les trophées pour un chapitre donné.		
		9.3	En tant que administrateur, je souhaite ajouter et modifier et afficher les plaques pour une session donnée.		
		9.4	En tant que chercheur, je souhaite afficher les plaques pour une session donnée.		

F11	Gérer les dictionnaires	10.1	En tant que administrateur, je souhaite ajouter, modifier, supprimer, afficher les détails des dictionnaires pour une session donnée.	60	8
		10.2	En tant que chercheur, je souhaite afficher les détails des dictionnaires pour une session donnée.		

**Tableau 4 : Product backlog**

#### **4. Diagramme de cas d'utilisation**

Dans cette partie, nous présentons le diagramme de cas d'utilisation qui modélise l'interaction entre le système informatique à développer et les acteurs interagissant avec le système. Également, il permet de définir les besoins des utilisateurs et les fonctionnalités du système de façon synthétique. .



**Figure 16 : Diagramme de cas d'utilisation global**

## 5. Diagramme de classe d'analyse

Le diagramme de classes est l'élément le plus important de la modélisation UML. Il est le plus proche de l'application que nous souhaitons développer, puisqu'il met en évidence les classes, qui vont se traduire par la suite lors de l'écriture du code en des classes et objets logiciel réels.

La figure 17 montre une partie du diagramme de classe global (ne prenant en compte que la partie du projet sur laquelle j'ai travaillé).

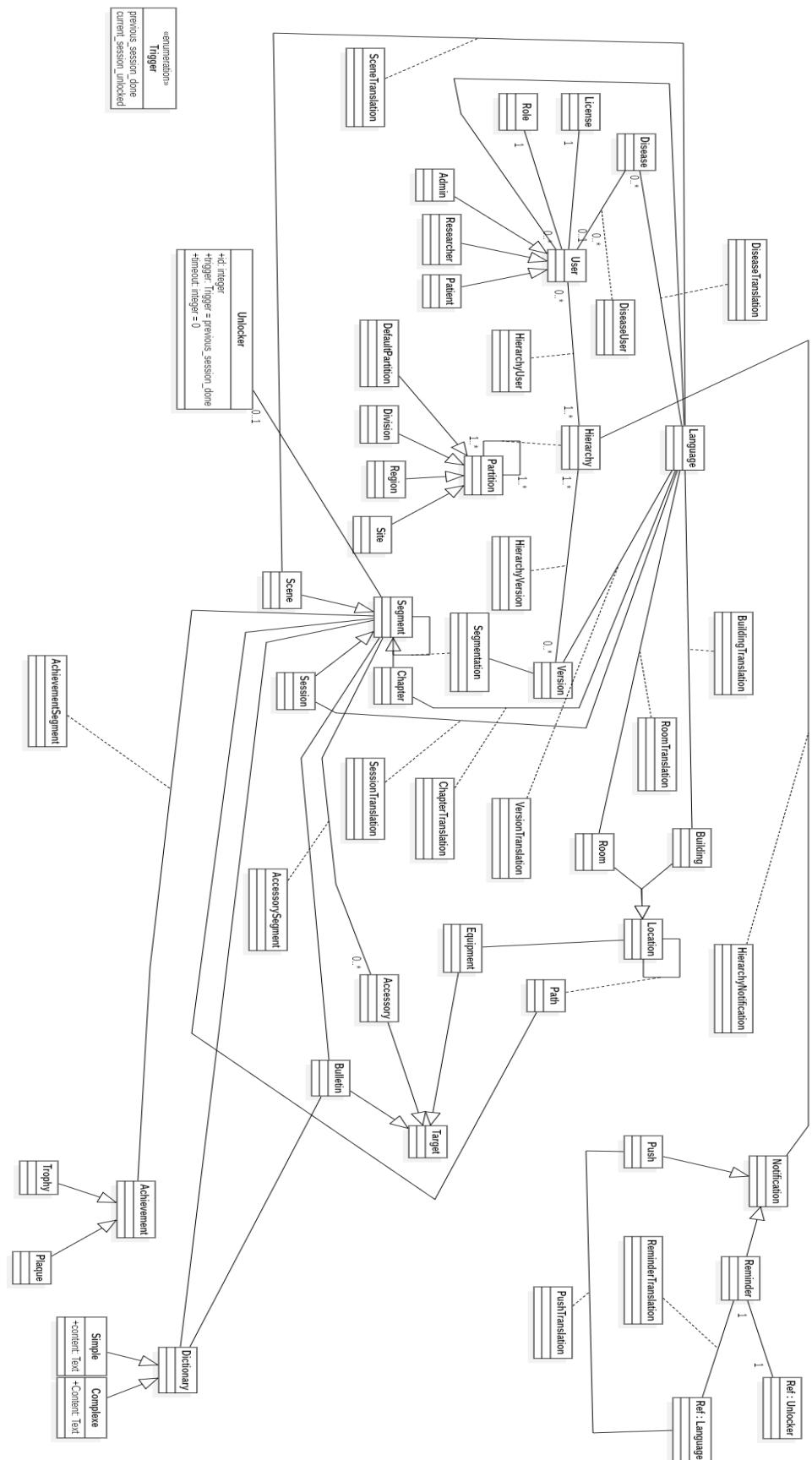
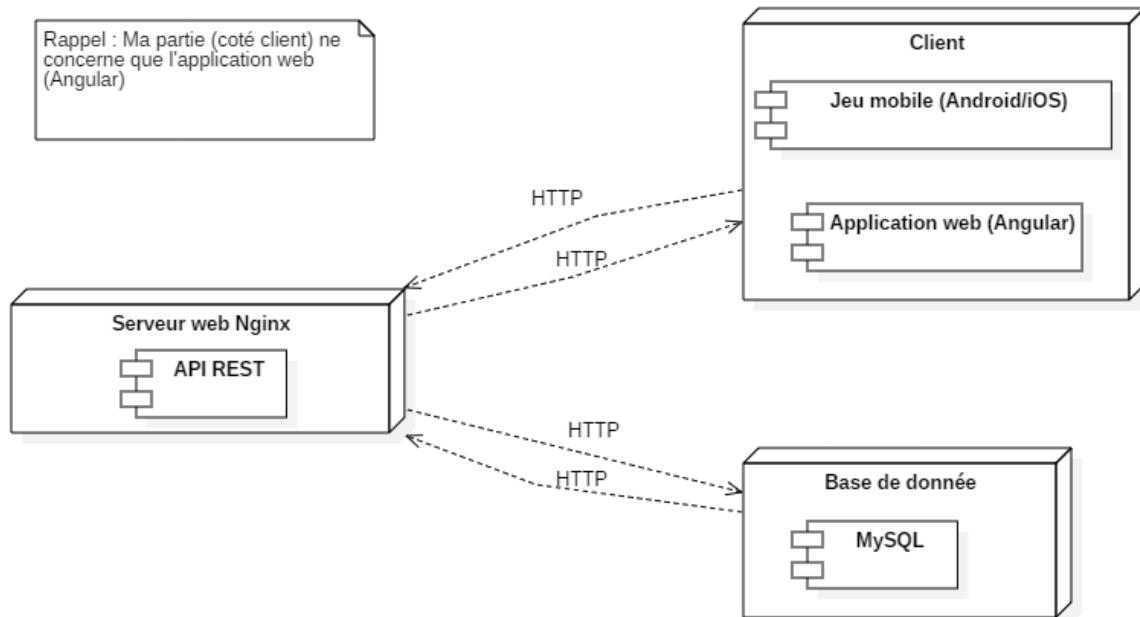


Figure 17 : Diagramme de classe global

## 6. Diagramme de déploiement

Le diagramme de déploiement suivant permet de visualiser la topologie des composants physiques du système, où les composants logiciels sont déployés.



**Figure 18 : Diagramme de déploiement**

Nous avons utilisé le modèle d'architecture 3 tiers, qui vise à modéliser notre système sous forme d'empilement de 3 couches, à savoir :

- Couche présentation : Regroupe nos deux applications client (web et mobile) qui servent à interagir avec les APIs REST.
- Couche traitement : Contient les différents APIs, elle contrôle les fonctionnalités d'une application en effectuant un traitement détaillé.
- Couche accès aux données : Comprend le système de stockage de base de données.

## Conclusion

Dans ce chapitre, nous avons présenté l'objectif de ce sprint (sprint 0). Pour ce faire, nous avons identifié les différents acteurs de l'application ainsi que les besoins fonctionnels et non fonctionnels. Nous avons également élaboré le product backlog et les différents diagrammes conceptuels afin d'avoir une meilleure vision sur le projet et les prochains sprints.

Dans le prochain chapitre, nous allons attaquer le premier sprint : La gestion des utilisateurs.

# **Chapitre 3 : Sprint 1 « Gestion des utilisateurs »**

## Introduction

Dans ce chapitre, nous allons nous intéresser au module de la gestion des utilisateurs. Pour ce faire, nous allons commencer par l'élaboration du sprint backlog, les différents diagrammes conceptuels, puis nous détaillerons la réalisation de différentes fonctionnalités de ce sprint.

### 1. Sprint Backlog

Dans cette partie, nous allons élaborer les différents user stories de ce sprint, ainsi que les tâches qui devront être accomplies à la fin du sprint et leurs estimations.

ID	User story	Tâche	Estimation (H)
S1	En tant que administrateur, je veux m'authentifier au CMS via un e-mail et un mot de passe.	<ul style="list-style-type: none"> <li>Mettre en place les entités de la base de données</li> <li>Implémenter Passport (middleware)</li> </ul>	8
S2	En tant que chercheur, je veux m'authentifier au CMS via un e-mail et un mot de passe.	<ul style="list-style-type: none"> <li>Implémenter les méthodes d'authentification et exposer les web services</li> <li>Implémenter les interfaces d'authentification</li> <li>Consommer et intégrer les web services d'authentification</li> </ul>	12
S3	En tant que administrateur, je veux consulter la liste des utilisateurs.	<ul style="list-style-type: none"> <li>Implémenter la méthode pour afficher la liste des utilisateurs.</li> <li>Exposer la méthode en tant que web service.</li> </ul>	16
	En tant que chercheur, je veux consulter la liste des utilisateurs.	<ul style="list-style-type: none"> <li>Implémenter les interfaces de l'affichage de la liste des utilisateurs (par type).</li> </ul>	6
			4

		<ul style="list-style-type: none"> <li>• Consommer et intégrer l'API de la liste des utilisateurs.</li> </ul>	8
S4	<p>En tant que administrateur, je veux consulter les détails d'un utilisateur.</p> <p>En tant que chercheur, je veux consulter les détails d'un utilisateur.</p>	<ul style="list-style-type: none"> <li>• Implémenter la méthode pour afficher les détails d'un utilisateur.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Implémenter les interfaces de l'affichage des détails d'un utilisateur.</li> <li>• Consommer et intégrer l'API des détails des utilisateurs.</li> </ul>	16 6 4 8
S5	<p>En tant que administrateur, je veux chercher un utilisateur spécifique.</p> <p>En tant que chercheur, je veux chercher un utilisateur spécifique.</p>	<ul style="list-style-type: none"> <li>• Ajouter une fonctionnalité cotée client qui permet d'effectuer une recherche sur un utilisateur spécifique.</li> </ul>	4
S6	En tant que administrateur, je veux ajouter un utilisateur.	<ul style="list-style-type: none"> <li>• Implémenter la méthode pour ajouter un utilisateur.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Implémenter l'interface du formulaire d'ajout d'un utilisateur.</li> <li>• Consommer et intégrer l'API de l'ajout d'un utilisateur.</li> </ul>	16 6 4 8
S7	En tant que administrateur, je veux modifier un utilisateur.	<ul style="list-style-type: none"> <li>• Implémenter la méthode pour modifier un utilisateur.</li> <li>• Exposer la méthode en tant que web service.</li> </ul>	16 6

		<ul style="list-style-type: none"> <li>• Implémenter l'interface du formulaire de modification d'un utilisateur.</li> <li>• Consommer et intégrer l'API de modification d'un utilisateur.</li> </ul>	4 8
S8	En tant que administrateur, je veux supprimer un utilisateur.	<ul style="list-style-type: none"> <li>• Implémenter la méthode pour supprimer un utilisateur.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Consommer et intégrer l'API de suppression d'un utilisateur.</li> </ul>	16 6 8

Tableau 5 : Sprint backlog gestion des utilisateurs

## 2. Diagramme de cas d'utilisation raffiné

La figure 19 illustre le diagramme de cas d'utilisation raffiné pour le module de gestion des utilisateurs.

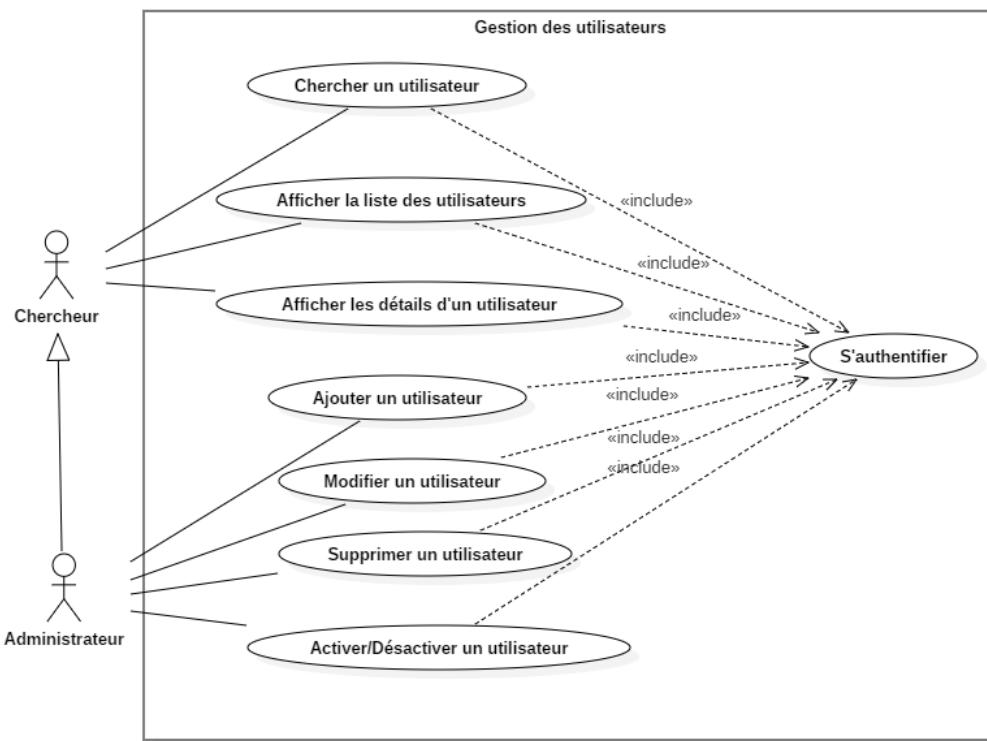


Figure 19 : Diagramme de cas d'utilisation pour le module de gestion des utilisateurs

### 3. Description du cas d'utilisation « s'authentifier »

Dans ce qui suit, nous allons détailler le cas d'utilisation « s'authentifier » en utilisant une description textuelle et un diagramme de séquence.

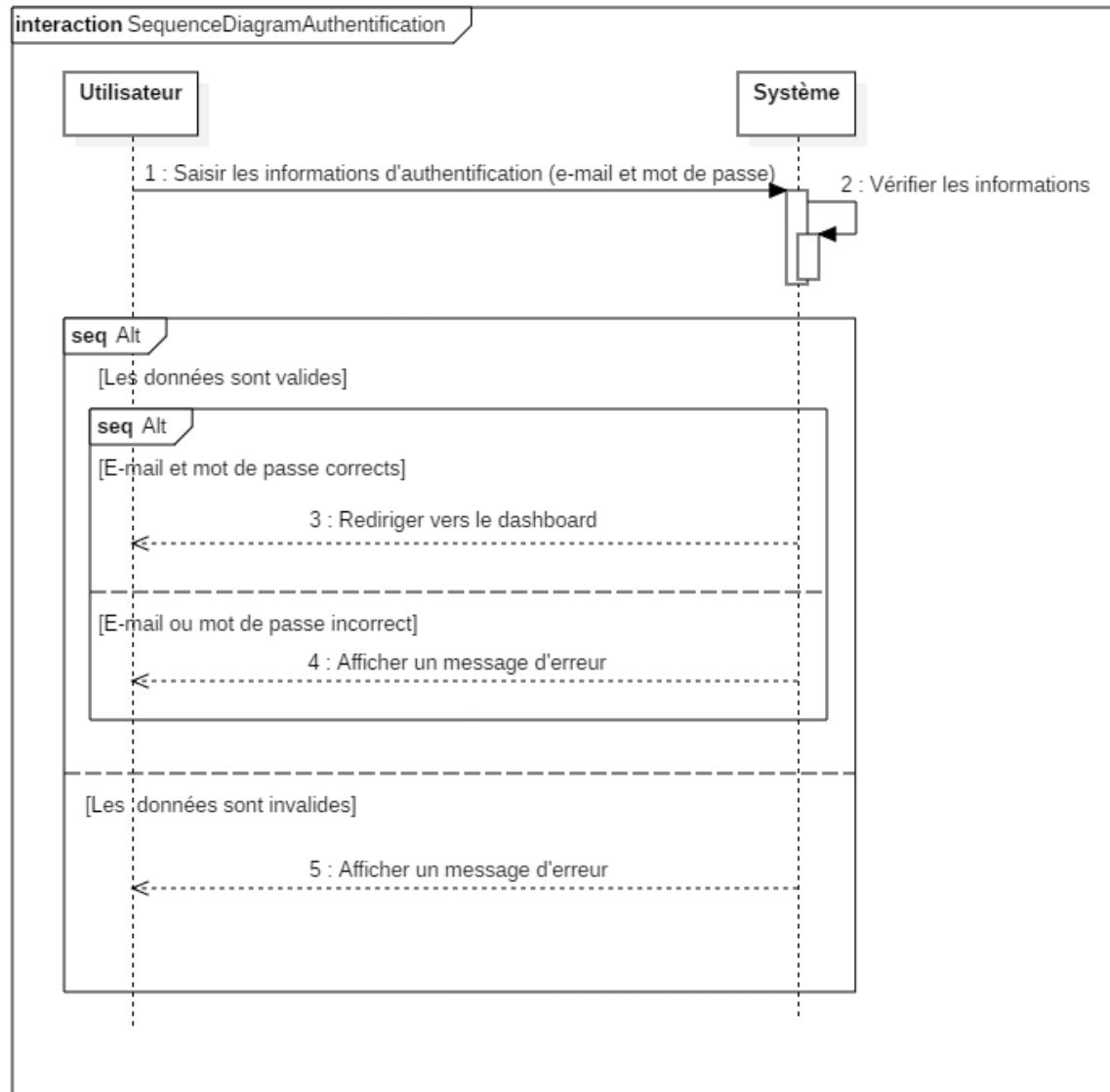
#### 3.1. Description textuelle du cas d'utilisation « s'authentifier »

<b>Description</b>	L'utilisateur a la possibilité de se connecter à l'application.
<b>Précondition</b>	L'utilisateur n'est pas connecté.
<b>Scénario nominal</b>	<ol style="list-style-type: none"> <li>1) Le système affiche la page de connexion</li> <li>2) L'utilisateur saisit ses données de connexion</li> <li>3) L'utilisateur clique sur le bouton « Login »</li> <li>4) Le système vérifie les informations de connexion</li> <li>5) Le système synchronise les données utilisateur avec le serveur</li> </ol>
<b>Exceptions</b>	<p>4.1 – Les données ne sont pas valides</p> <p>4.1.1 – Le système affiche un message d'erreur et retourne au 1<sup>er</sup> point</p> <p>4.2 – Les informations de connexion sont erronées</p> <p>4.2.1 – Le système affiche un message d'erreur et retourne au premier point</p>
<b>Post-condition</b>	L'utilisateur est connecté

**Tableau 6 : Description textuelle du cas d'utilisation « s'authentifier »**

#### 3.2. Diagramme de séquence du cas d'utilisation « s'authentifier »

La figure 20 représente le diagramme de séquence du cas d'utilisation « s'authentifier » :



**Figure 20 : diagramme de séquence du cas d'utilisation « s'authentifier »**

## 4. Description du cas d'utilisation « ajouter un utilisateur »

Dans ce qui suit, nous allons détailler le cas d'utilisation « ajouter un utilisateur » à travers la description textuelle et le diagramme de séquence.

### 4.1. Description textuelle du cas d'utilisation « Ajouter un utilisateur »

<b>Description</b>	L'utilisateur a la possibilité d'ajouter un utilisateur
<b>Précondition</b>	L'utilisateur est connecté et a le rôle « administrateur ».

<b>Scénario nominal</b>	1) L'utilisateur demande le formulaire d'ajout 2) Le système affiche le formulaire d'ajout 3) L'utilisateur saisit les données 4) Le système vérifie la validité des données saisies 5) Le système enregistre les données dans la base de données
<b>Exceptions</b>	4.1 – Les données ne sont pas valides  4.1.1 – Le système affiche un message d'erreur et retourne au 2 <sup>ème</sup> point
<b>Post-condition</b>	Utilisateur enregistré dans la base de données

Tableau 7 : Description textuelle du cas d'utilisation « ajouter un utilisateur »

#### 4.2. Diagramme de séquence objet du cas d'utilisation « ajouter un utilisateur »

La figure 21 représente le diagramme de séquence objet du cas d'utilisation « ajouter un utilisateur » :

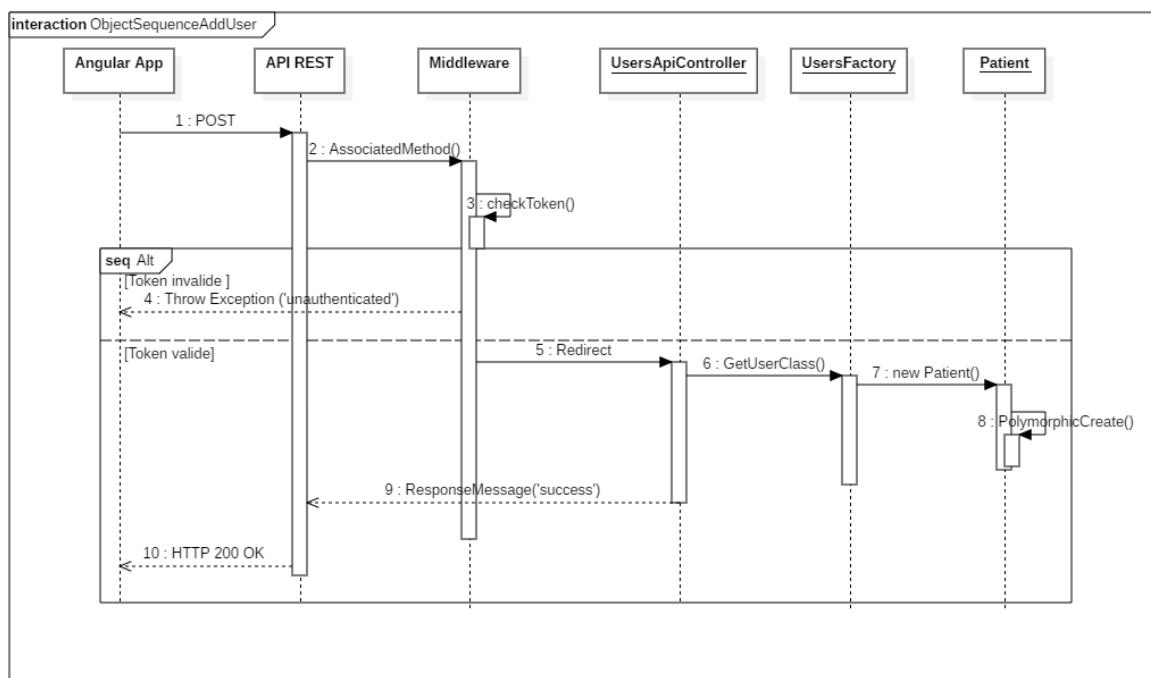


Figure 21 : Diagramme de séquence objet du cas d'utilisation « ajouter un utilisateur »

## 5. Description du cas d'utilisation « modifier un utilisateur »

Dans ce qui suit, nous allons détailler le cas d'utilisation « modifier un utilisateur » à travers la description textuelle et le diagramme de séquence.

### 5.1. Description textuelle du cas d'utilisation « modifier un utilisateur »

<b>Description</b>	L'utilisateur a la possibilité de modifier un utilisateur
<b>Précondition</b>	L'utilisateur est connecté et a le rôle « administrateur ».
<b>Scénario nominal</b>	<ol style="list-style-type: none"> <li>1) L'utilisateur demande le formulaire de modification, ceci peut être effectué de deux manières :           <ol style="list-style-type: none"> <li>a) Dans la page de la liste des utilisateurs</li> <li>b) Dans la page des détails d'un utilisateur</li> </ol> </li> <li>2) Le système affiche le formulaire de modification</li> <li>3) L'utilisateur saisit les modifications</li> <li>4) Le système vérifie la validité des données saisies</li> <li>5) Le système enregistre les modifications dans la base données</li> </ol>
<b>Exceptions</b>	4.1 – Les données ne sont pas valides 4.1.1 – Le système affiche un message d'erreur et retourne au 2 <sup>ème</sup> point
<b>Post-condition</b>	Modifications enregistrés dans la base de données

Tableau 8 : Description textuelle du cas d'utilisation « modifier un utilisateur »

### 5.2. Diagramme de séquence objet du cas d'utilisation « modifier un utilisateur »

La figure 22 représente le diagramme de séquence objet du cas d'utilisation « modifier un utilisateur » :

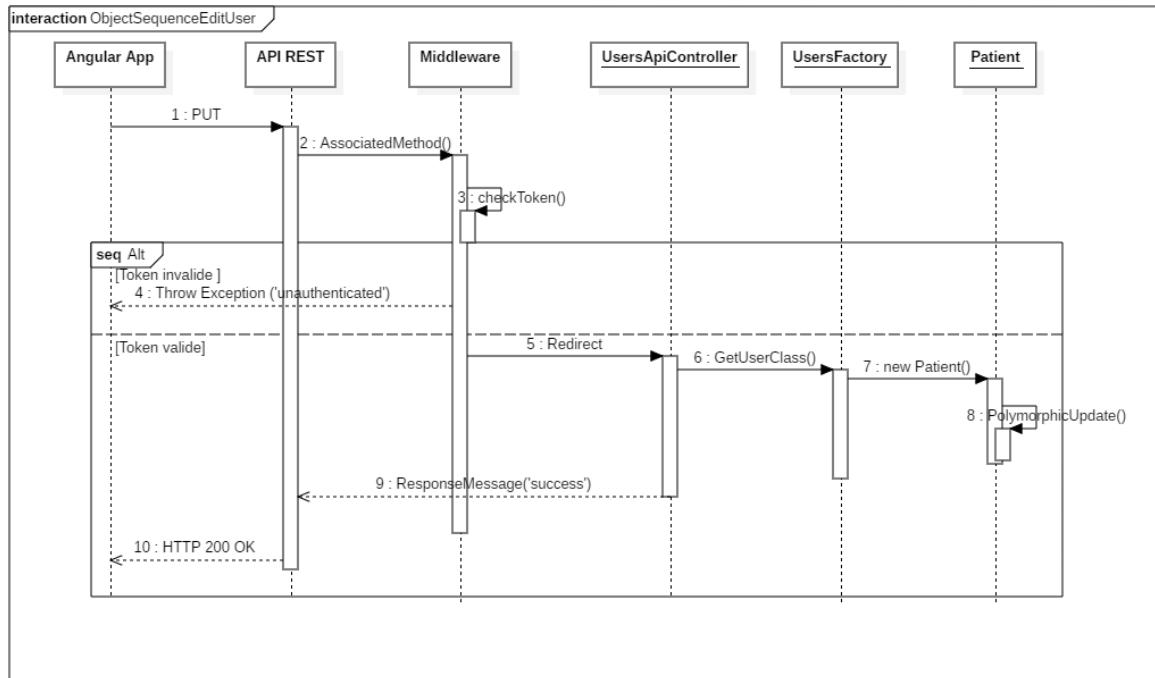


Figure 22 : Diagramme de séquence objet du cas d'utilisation « modifier un utilisateur »

## 6. Description du cas d'utilisation « supprimer un utilisateur »

Dans ce qui suit, nous allons détailler le cas d'utilisation « modifier un utilisateur » à travers la description textuelle et le diagramme de séquence.

### 6.1. Description textuelle du cas d'utilisation « supprimer un utilisateur »

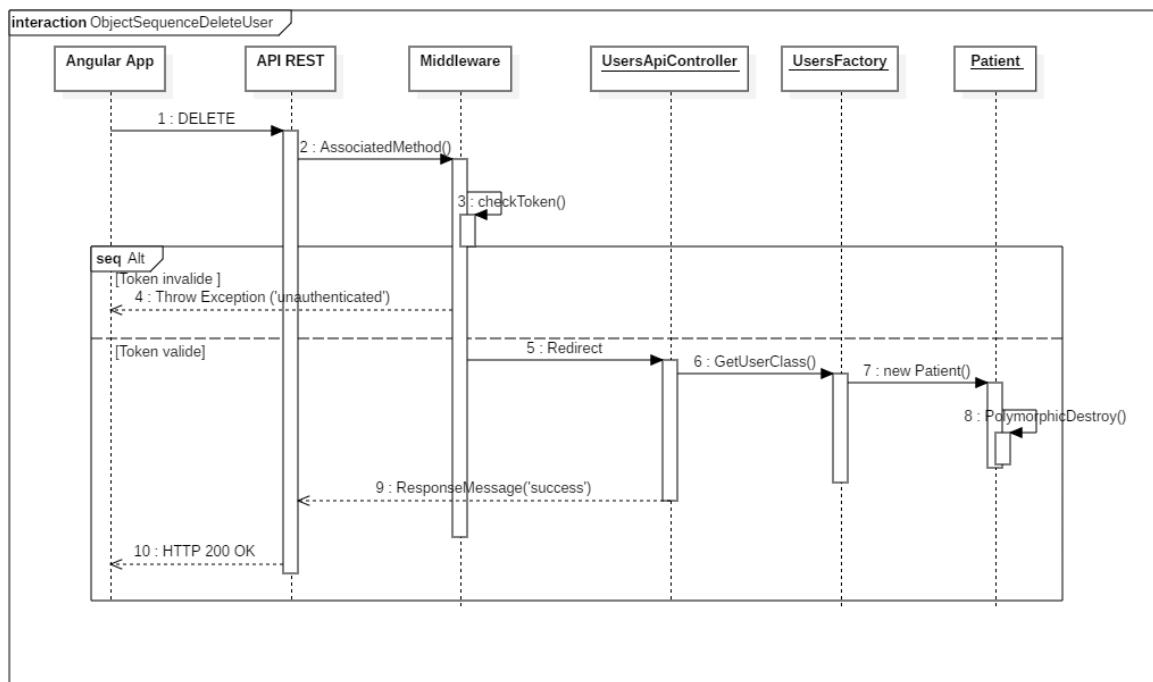
<b>Description</b>	L'utilisateur a la possibilité de supprimer un utilisateur
<b>Précondition</b>	L'utilisateur est connecté et a le rôle « administrateur ».
<b>Scénario nominal</b>	<ol style="list-style-type: none"> <li>Le système affiche la liste des utilisateurs</li> <li>L'utilisateur clique sur le bouton « delete » pour un utilisateur spécifique</li> <li>Le système supprime les données de l'utilisateur de la base de données</li> <li>Le système rafraîchit la liste des utilisateurs</li> </ol>
<b>Exceptions</b>	aucune

<b>Post-condition</b>	Utilisateur supprimé au niveau de la base de données
-----------------------	--

**Tableau 9 : Description textuelle du cas d'utilisation « supprimer un utilisateur »**

## 6.2. Diagramme de séquence objet du cas d'utilisation « supprimer un utilisateur »

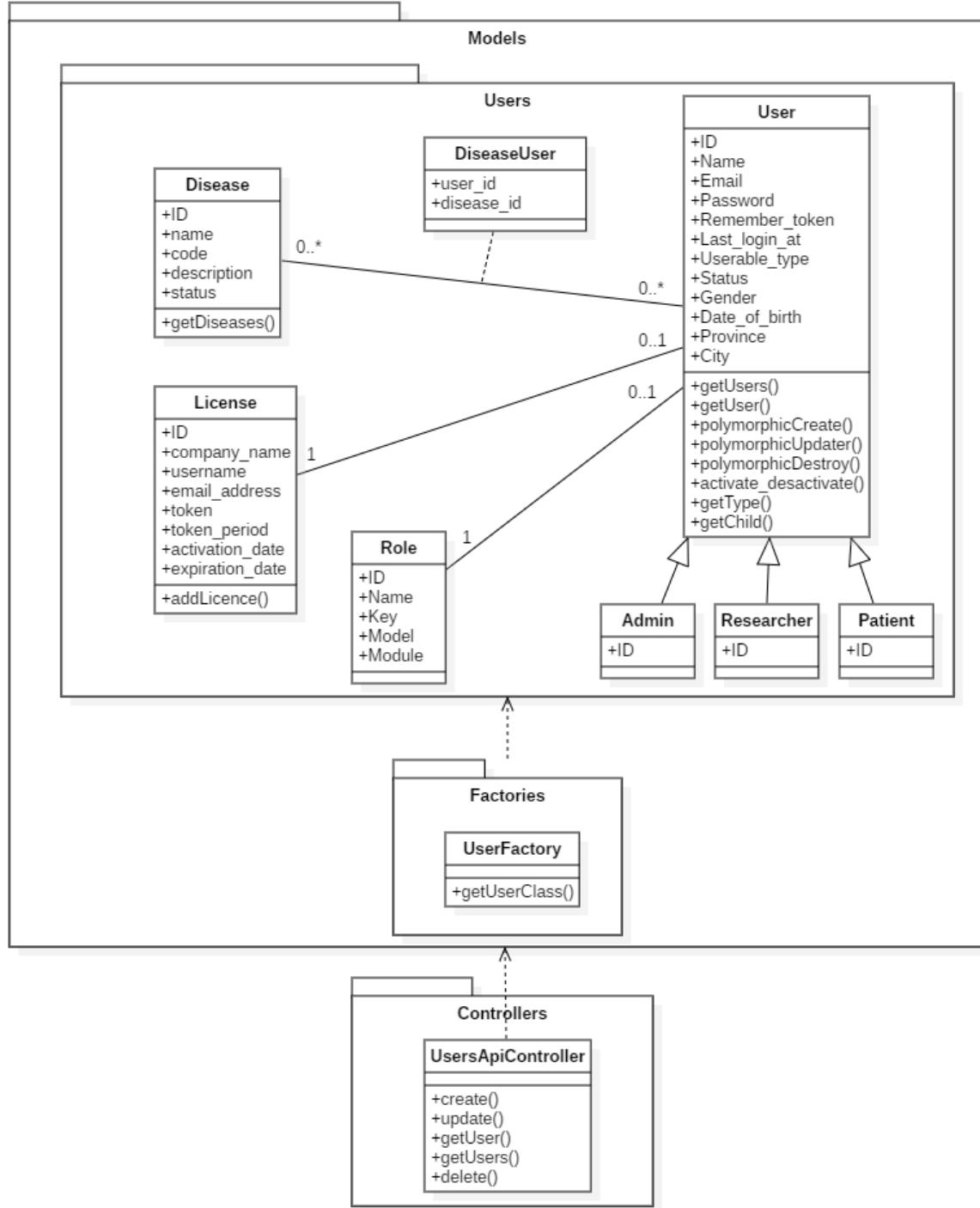
La figure 23 représente le diagramme de séquence objet du cas d'utilisation « supprimer un utilisateur » :



**Figure 23 : diagramme de séquence objet du cas d'utilisation « supprimer un utilisateur »**

## 7. Diagramme de classe de conception

La figure 24 représente le diagramme de classe relatif au module « gestion des utilisateurs » :



**Figure 24 : Diagramme de classe relatif au module gestion des utilisateurs**

Approche pour laquelle nous avons opté dans notre conception :

- L'utilisation du patron de conception Factory :

Le patron de conception Factory est utilisé lorsque nous avons une super-classe (User) avec plusieurs sous-classes (Admin, researcher, patient ...) et en fonction des entrées, nous devons instancier l'une des sous-classes à travers la classe Factory.

Ceci nous permet de tirer plusieurs avantages, à savoir :

- Fournir une abstraction entre les classes d'implémentation et contrôleurs via l'héritage.
- Factory rend notre code plus robuste, moins couplé et facile à étendre. Par exemple, nous pouvons facilement modifier l'implémentation de la classe Admin.
- Le contrôleur n'a pas à changer radicalement.
- Facile à implémenter.

- **L'utilisation de la notion du polymorphisme :**

Polymorphisme signifie la capacité de prendre plus d'une forme. Une opération peut présenter différents comportements dans différents cas. Le comportement dépend des types de données utilisés dans l'opération. Voici les avantages d'utiliser le principe du polymorphisme :

- Simplicité : Si nous devons écrire du code qui traite une famille de types, le code peut ignorer les détails spécifiques au type et simplement interagir avec le type de base de la famille. Cela rend notre code plus facile à écrire et plus facile à comprendre pour les autres.
- Extensibilité : D'autres sous-classes pourraient être ajoutées ultérieurement à la famille de types et les objets de ces nouvelles sous-classes fonctionneraient également avec le code existant.

## 8. Réalisation

Après l'étude conceptuelle de notre sprint « Gestion des utilisateurs » à travers les descriptions textuelles et les diagrammes de séquence, nous passerons à la partie de l'implémentation.

### 8.1. Maquettes

Les figures ci-dessous montrent les maquettes que nous avons établies afin de les valider avec le product owner.

A Web Page

https://

Email

Password

For research purposes only. Confidential.

[Forgot password?](#)

**Figure 25 : Maquette de la page d'authentification**

A Web Page

https://

User Name

Users - Patients						
<a href="#">+ Add</a>						
Show <input type="button" value="10"/> entries <input style="width: 100px;" type="text"/> Search: <input type="text"/>						
Id	Study Number	Name	Condition	Site	Status	Actions
1	123	Pertti Suibhne	SPINA BIFIDA	Site 1	Active	<a href="#">edit   inactivate</a>
2	222	Moss Ankit	CEREBRAL PALSY	Site 2	Active	<a href="#">edit   inactivate</a>
3	333	Axel Frans	SPINA BIFIDA	Site 2	Active	<a href="#">edit   inactivate</a>
4	444	Elias Esaias	AUTISM SPECTRUM DISORDER	Site 3	Active	<a href="#">edit   inactivate</a>
5	321	Eduard Ruben	FETAL ALCOHOL SYNDROME	Site 4	Active	<a href="#">edit   inactivate</a>
6	555	Rudolf Karen	SPINA BIFIDA	Site 1	Inactive	<a href="#">edit</a>

Showign <x> of <y> entries Previous 1 2 3 Next

**Figure 26 : Maquette de la page de la liste des patients**

The screenshot shows a web application interface for adding a patient. The top navigation bar includes a logo for 'My Ready Transition', a user icon, and a 'User Name' field. On the left, a sidebar menu lists various sections: Users (Administrators, Patients, Researchers), Languages, Hierarchy of partitions, Content Management (Versions, Buildings, InterfaceLabels), Notifications (Automatic notifications, Push notifications), and Analytics. The main content area is titled 'Users - Patients' and contains two sections: 'User Credentials' and 'User Profile'. In 'User Credentials', fields include 'Study Number \*' (text input), 'Patient Number \*' (text input), 'Division \*' (dropdown: List of divisions), 'Region \*' (dropdown: List of regions), and 'Site \*' (dropdown: List of sites). In 'User Profile', fields include 'Full Name (Not required)' (text input), 'Email (Not required)' (text input), 'Phone number (Not required)' (text input), 'Gender (Not required)' (dropdown: Male or Female), 'Date of birth (Not required)' (text input with calendar icon), 'Condition (Not required)' (dropdown: CEREBRAL PALSY, AUTISM SPECTRUM DISORDER, FETAL ALC), 'Address (Not required)' (text input), 'Province (Not required)' (dropdown: List of Canada Provinces), 'City (Not required)' (dropdown: List of City according to the province selected), and 'Status \*' (dropdown: Active / Inactive). At the bottom right are 'Cancel' and 'Save' buttons.

**Figure 27 : Maquette de la page d'ajout d'un patient**

## 8.2. REST API

Une API RESTful est une interface de programme d'application (API) qui utilise des requêtes HTTP pour les données GET, PUT, POST et DELETE. Elle est basée sur la technologie de pointe (REST), un style architectural et une approche de la communication souvent utilisés dans le développement de services Web.

Voici donc les API REST que nous avons créées pour ce sprint, et que nous allons utiliser plus tard dans la partie front-end avec Angular.

users	
GET	/api/users/byType/{type} List all users by type
POST	/api/users/byType/{type} Create a new user base on the type
GET	/api/users/{id}/byType/{type} Get a user by its ID and Type
PUT	/api/users/{id}/byType/{type} Updates a given user
DELETE	/api/users/{id}/byType/{type} Get a user by its ID
PUT	/api/users/finalizeRegistration Updates a given user
PUT	/api/users/{id}/activate/byType/{type} Activates a given user
PUT	/api/users/{id}/inactivate/byType/{type} Inactivates a given user

**Figure 28 : API REST pour le module « gestion des utilisateurs »**

Afin de faciliter l'utilisation, la maintenance et le contrôle des versions des API, nous avons opté pour le framework Swagger, qui nous permet de documenter nos API d'une façon très simple.

### 8.3. Sécurisation

#### 8.3.1. Passport

Laravel facilite déjà l'authentification via les formulaires de connexion traditionnels, mais qu'en est-il des API? Les API utilisent généralement des jetons pour authentifier les utilisateurs et ne conservent pas l'état de session entre les demandes. Laravel simplifie l'authentification des API avec **Laravel Passport**, qui fournit une implémentation complète du serveur OAuth2 pour une application Laravel en quelques minutes.

Maintenant, voici comment tout va fonctionner :

- 1) Lorsque l'utilisateur veut se connecter, il envoie une requête avec les paramètres requis (email et mot de passe), le serveur vérifie la validité de celles-ci. Une fois validés, le serveur crée un jeton signé et le renvoie en réponse (Figure 25).
- 2) Le client enregistre le jeton localement (généralement dans localStorage) et le renvoie à chaque demande ultérieure nécessitant une authentification.
- 3) Toutes les demandes nécessitant une authentification passent par un middleware qui vérifie le jeton fourni et n'autorise la demande que si le jeton est vérifié. En effet, lorsque nous appelons des routes sécurisées par Passport, les utilisateurs d'API de l'application doivent spécifier leur jeton d'accès en tant que **Bearer Token** dans le header Authorization de leur requête (Figure 26).

Response body

```
{
  "status": "success",
  "code": 0,
  "message": "user logged in",
  "payload": {
    "auth": {
      "token_type": "Bearer",
      "expires_in": 31535998,
      "access_token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzIiNiIsp0aS16ImNkOGFiNmY3Ng1N2ExYWZiYzgZYjBkZDM4ZTUy0DgzZmVLNwIx0Tk5YTkwNDkwOTM2MjNhMTYxZDBmYmVmZmUzN2EsNmIzZDQx0GZ1NGZk1n8.eJhdWQ10i1xi1iwanRpIjoiY2QAYWI2Zj3ODU3YTFhZmJjODN1MGRkfzhlNTI40DNmZWU1YjE50TlhTA0OTA5MzYh2ExNjFkMGZ1ZWZmZTM3YTk2YjNkNDE4ZmI8ZmQ1LCjpxXQj0jE1NDE20TM0MzAsIm5iZi16MTU0MTy5MzQzMwCwizXhwIjoxNTczMjI5NDI5LCJzdWI0i1iwi2NvcGVzIjpbIioiXXo.PgUK91ZxcUajzcHpi7gkSFK_UWv._Jx1DVB-yNm@815z8HeLuAkhJ1R5y_axX97296Zjb09QkaqVJlt08_ZA_NdYnkImPECDdp0y7hSol1-GwsIJ5enq-d89sFTHDyFvJolcwhJ_19qxqrEnY38ujBh1j0@TbnLqjT16awhHizHA-gSvn0rQdv0rRpUnYRfs0yxlHwexdx1pETXmn1IgcePTMwCT0GAVM2k4mt-Vwx9Mxr1Lfbsbn9IgL7shWnpkj98SdyWcdabVnvoA1Zymk6Wq211eiyYNCL4gg6tSo03Kw6x-UbrcksTr4VlcHHVxLozRD_7RyKRTs2xq1AD8kfrW8fTVyXfsx12_FUmpH3mMnt10y9eqfseoJRd33ft1IXI_91_-zFvZ9aEwrqTmQx_0yayVAhmCs0tc-VUGtfousUej71KxYEpqK0x8X2-tK-6yQSL34FDzHYd3w89tRJMZh9xyeQZHSQj6dpkF1rkAvZRI1S7yCPkywUB4anHoIIIXeY9vUcUenrLxVNeB-x41CcSUSGUr3ep6rmzthJ6sBMOAWJuGaqtftHlnogPku0zw0_oTaXXyC6S9R2s-EsUa0432sTo-FBZB9PLq455KwWJG2KMcpeyUAER3VrzjS90pdvs-BAPOUkWVjqacpr4BWozW_vJbh1ik",
      "refresh_token": "def8209a3b24d0476aef76bd3b9bf1b4b0874b07aa677321f9d00b3ed8e027c5857cd6346f3a6f7fb9723cf2fed5a472c84540e8cf327d7a82b837b937e0060a597392912d0d6e6a9fd78a24ae6b6346e291e8925142096c1d30377188e45dfedc8da914f5232304836fd0502a0a3b4b7daaf4f0b65c3969cc418d2e55c9ca8ca7743241d7682da2b9c6c581144ef3a919abd970d7be95232534af10137296df9316559916ae073f1bad81c61357d1827fa26137ba01b70de0994f11ede9562c2feab86f261bb469aaft786c533c4eebfc75e6716f7teeb5f34978ddad3b0b361c9ec13aa9d78c20689a9b9d423f5e0fb96549d11422f0624434e0212066a9fd99eceeb136c601c06a99208df2787e0254d7eefa24a82df6b7da3d79a701dee0f0d3323070d410f20a2436ae6d5dc21938c755558ebca67bd653134955294915cd756b1c44fe3691787fb70354f6a9fdbce30d12c72eaf57a14f619fdh5d1234e"
    }
  }
}
```

[Download](#)

Figure 29 : Access token généré par Passport

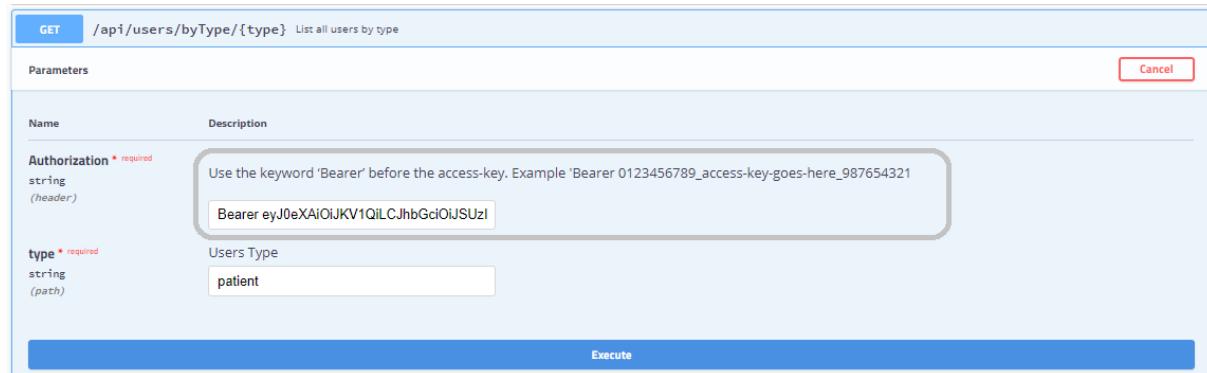


Figure 30 : Envoi du token dans le header de la requête

### 8.3.2. Autorisations

Étant donné que la définition de toute la logique d'autorisation peut devenir fastidieuse dans les applications volumineuses, Laravel nous permet de diviser notre logique d'autorisation en classes "Policies". Policies sont des classes PHP simples qui regroupent la logique d'autorisation en fonction de la ressource qu'elles autorisent. Ceci nous permet d'ajouter une autre couche de sécurité qui teste sur le rôle de l'utilisateur connecté, la ressource à accéder et l'opération à effectuer. Voici un exemple de la classe AdminPolicy qui sert à implémenter la logique des autorisations sur la ressource « Admin » :

```

    /**
     * Determine whether the user can list admins.
     *
     * @param \App\User $user
     * @return mixed
     */
    public function list_admins_only(User $user)
    {
        return ($user->hasRole(Role::$ADMIN));
    }

```

**Figure 31 : Exemple de définition de policy**

Ceci se traduit par : Seuls les utilisateurs ayant le rôle Admin peuvent consulter la liste des Administrateurs. La policy peut être ainsi utilisé dans nos contrôleurs.

```

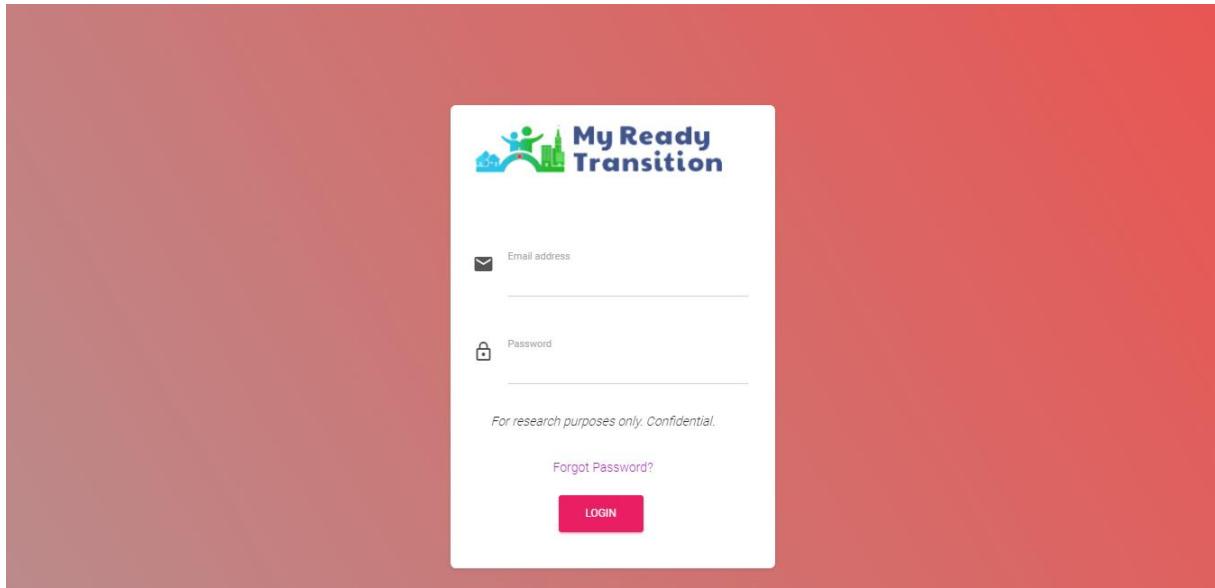
public function getUsers(Request $request, $type)
{
    /* this factory is to get the real user's class type */
    /* $userClass is the className related to the $type : Admin, Patient or Researcher */
    $userClass = (new UserFactory())->getUser($type);
    /* We check the policy to see if the connected user [$request->user()] has the right to perform this action on the model type */
    if ($request->user()->can('list_all', $userClass)) {
        $users = User::getUsers($userClass::all());
        //event(new GetApiCalledEvent($request, $users));
        return ResponseMessage::Report("users list", "success", $users);
    } elseif ($request->user()->can('list_admins_only', $userClass)) {
        $users = User::getUsers($userClass::whereHas('user.role', function ($q) {
            $q->where('role_key', Role::$ADMIN);
        })->get());
        return ResponseMessage::Report("users list", "success", $users);
    }
}

```

**Figure 32 : Utilisation des policies**

## 8.4. Interfaces graphiques

La première étape pour utiliser le CMS et accéder aux différentes fonctionnalités est l'authentification. L'utilisateur doit posséder déjà un compte pour se connecter. Dans le cas échéant, il doit demander à un administrateur de lui en fournir un. Voici donc l'interface d'authentification :



**Figure 33 : Interface d'authentification**

La figure 34 représente l'interface de la gestion des patients. Chaque ligne représente un patient spécifique avec ses informations relatifs (Numéro d'étude, nom ... etc.) Depuis cette interface, l'administrateur peut ajouter, éditer, activer/désactiver ou afficher les détails d'un patient.

Study number	Name	Site	Status	Actions
1044	Mark	McGILL	Active	VIEW EDIT INACTIVATE
6575	Jennifer	McGILL	Active	VIEW EDIT INACTIVATE
3485	Ayoub Gharbi	McGILL	Active	VIEW EDIT INACTIVATE
3377	Alicia Vla-Dufresne Ley	McGILL	Active	VIEW EDIT INACTIVATE
6369	Alex	McGILL	Active	VIEW EDIT INACTIVATE

**Figure 34 : Interface de la liste des patients**

En cliquant sur le bouton « Add », le formulaire d'ajout d'un patient s'affiche contenant tous les champs nécessaires. Certains champs étant obligatoires, l'administrateur ne peut ajouter un patient qu'en remplissant celles-ci. Dans le cas contraire, des messages d'erreur s'affichent sous les champs laissés vides.

The screenshot shows the ChildBright application interface. On the left is a red sidebar with navigation links: Users, Administrators, Patients (selected), Researchers, Languages, Hierarchy of partitions, Content Management, Notifications, FAQ, and Analytics. The main area is titled 'User Credentials' and contains fields for Study Number (001) and Patient Number (001). Below this, dropdown menus show division (RCT001CB), region (Quebec), and site (McGILL). The 'User Profile' section includes fields for Full Name (Ahmed), Email (ahmedc@360medlink.com), Phone number (123456789), Gender (Male), Date of birth, and Address.

**Figure 35 : Interface d'ajout d'un patient**

## Conclusion

Dans ce chapitre, nous nous sommes intéressés à la réalisation du module de la gestion des utilisateurs. Pour ce faire, nous avons commencé par l'élaboration du sprint backlog, puis une étude conceptuelle et finalement la réalisation.

Dans le chapitre suivant nous allons passer au 2<sup>ème</sup> sprint qui est la gestion du contenu.

# **Chapitre 4 : Sprint 2 « Gestion du contenu »**

## Introduction

Le but de ce sprint est de mettre en œuvre les différentes fonctionnalités qui vont servir à gérer le contenu du jeu mobile *MyReady Transition*. Pour ce faire, nous allons élaborer le sprint backlog et les diagrammes relatifs à la conception, puis nous clôturerons cette partie par des captures d'écran expliquant la réalisation.

### 1. Sprint backlog

Dans cette partie, nous allons élaborer les différents user stories de ce sprint, ainsi que les tâches qui devront être accomplies à la fin du sprint et leurs estimations.

ID	User Story	Tâche	Estimation (H)
S1	En tant que administrateur, je veux ajouter une version.	<ul style="list-style-type: none"> <li>• Mettre en place les entités de la base de données</li> <li>• Implémenter la méthode pour ajouter une version.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Implémenter l'interface de l'ajout d'une version.</li> <li>• Consommer et intégrer l'API de l'ajout d'une version.</li> </ul>	6 4 1 2 4
S2	En tant que administrateur, je veux supprimer une version.	<ul style="list-style-type: none"> <li>• Implémenter la méthode pour supprimer une version.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Consommer et intégrer l'API de la suppression d'une version.</li> </ul>	2 1 2

S3	En tant que administrateur, je veux modifier une version.	<ul style="list-style-type: none"> <li>• Implémenter la méthode pour modifier une version.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Implémenter l'interface de modification d'une version.</li> <li>• Consommer et intégrer l'API de la modification d'une version.</li> </ul>	4 1 2 4
S4	En tant que administrateur, je veux changer le statut d'une version.	<ul style="list-style-type: none"> <li>• Implémenter la méthode pour changer le statut d'une version.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Consommer et intégrer l'API de la modification d'une version.</li> </ul>	16 2 4
S5	En tant que administrateur/chercheur, je veux consulter la liste des versions.	<ul style="list-style-type: none"> <li>• Implémenter la méthode pour afficher la liste des versions.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Implémenter l'interface d'affichage de la liste des versions.</li> <li>• Consommer et intégrer l'API de la liste des versions.</li> </ul>	4 1 2 4
S6	En tant que administrateur/chercheur, je veux consulter les détails d'une version.	<ul style="list-style-type: none"> <li>• Implémenter la méthode pour afficher les détails d'une version.</li> <li>• Exposer la méthode en tant que web service.</li> </ul>	4 1

		<ul style="list-style-type: none"> <li>• Implémenter l'interface d'affichage des détails d'une version.</li> <li>• Consommer et intégrer l'API de l'affichage des détails d'une version.</li> </ul>	2 4
S7	En tant que administrateur, je veux ajouter un segment (chapitre, session).	<ul style="list-style-type: none"> <li>• Mettre en place les entités de la base de données</li> <li>• Implémenter la méthode pour ajouter un segment.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Consommer et intégrer l'API de l'ajout d'un segment.</li> </ul>	6 8 2 4
S8	En tant que administrateur, je veux supprimer un segment (chapitre, session).	<ul style="list-style-type: none"> <li>• Implémenter la méthode pour supprimer un segment.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Consommer et intégrer l'API de la suppression d'un segment.</li> </ul>	4 1 4
S9	En tant que administrateur, je veux modifier un segment (chapitre, session).	<ul style="list-style-type: none"> <li>• Implémenter la méthode pour modifier un segment.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Implémenter l'interface de modification d'un segment.</li> <li>• Consommer et intégrer l'API de modification d'un segment.</li> </ul>	4 1 2 4

S10	En tant que administrateur/chercheur, je veux consulter la liste des segments (chapitres, sessions)	<ul style="list-style-type: none"> <li>• Implémenter la méthode pour afficher la liste des segments.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Mettre à jour la tree table pour afficher les chapitres et les sessions.</li> <li>• Consommer et intégrer l'API de la liste des segments.</li> </ul>	8 1 4 4
S11	En tant que administrateur/chercheur, je veux consulter les détails d'un segment (chapitre, session)	<ul style="list-style-type: none"> <li>• Implémenter la méthode pour afficher les détails d'un segment.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Implémenter l'interface d'affichage des détails d'un segment.</li> <li>• Consommer et intégrer l'API de l'affichage des détails d'un segment.</li> </ul>	4 1 2 4
S12	En tant que administrateur, je veux ajouter un emplacement (bâtiment, chambre).	<ul style="list-style-type: none"> <li>• Mettre en place les entités de la base de données</li> <li>• Implémenter la méthode pour ajouter un emplacement.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Consommer et intégrer l'API de l'ajout d'un emplacement.</li> </ul>	6 8 2 4
S13	En tant que administrateur, je veux supprimer un	<ul style="list-style-type: none"> <li>• Implémenter la méthode pour supprimer un emplacement.</li> </ul>	4

	emplacement (bâtiment, chambre).	<ul style="list-style-type: none"> <li>• Exposer la méthode en tant que web service.</li> <li>• Consommer et intégrer l'API de la suppression d'un emplacement.</li> </ul>	1 4
S14	En tant que administrateur, je veux modifier un emplacement (bâtiment, chambre).	<ul style="list-style-type: none"> <li>• Implémenter la méthode pour modifier un emplacement.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Implémenter l'interface de modification d'un emplacement.</li> <li>• Consommer et intégrer l'API de modification d'un emplacement.</li> </ul>	4 1 2 4
S15	En tant que administrateur/chercheur, je veux consulter la liste des emplacements (bâtiments, chambres)	<ul style="list-style-type: none"> <li>• Implémenter la méthode pour afficher la liste des emplacements.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Implémenter la tree table pour afficher les bâtiments et les chambres</li> <li>• Consommer et intégrer l'API de la liste des emplacements.</li> </ul>	8 1 4 4
S16	En tant que administrateur/chercheur, je veux consulter les détails d'un emplacement (bâtiment, chambre).	<ul style="list-style-type: none"> <li>• Implémenter la méthode pour afficher les détails d'un emplacement.</li> <li>• Exposer la méthode en tant que web service.</li> </ul>	4 1

		<ul style="list-style-type: none"> <li>• Implémenter l'interface d'affichage des détails d'un emplacement.</li> <li>• Consommer et intégrer l'API de l'affichage des détails d'un emplacement.</li> </ul>	2 4
S17	En tant que administrateur, je veux ajouter un équipement pour une chambre.	<ul style="list-style-type: none"> <li>• Mettre en place les entités de la base de données</li> <li>• Implémenter la méthode pour ajouter un équipement.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Consommer et intégrer l'API de l'ajout d'un équipement.</li> </ul>	6 8 2 4
S18	En tant que administrateur, je veux supprimer un équipement.	<ul style="list-style-type: none"> <li>• Implémenter la méthode pour supprimer un équipement.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Consommer et intégrer l'API de la suppression d'un équipement.</li> </ul>	4 1 4
S19	En tant que administrateur, je veux modifier un équipement.	<ul style="list-style-type: none"> <li>• Implémenter la méthode pour modifier un équipement.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Implémenter l'interface de modification d'un équipement.</li> <li>• Consommer et intégrer l'API de modification d'un équipement.</li> </ul>	4 1 2 4

S20	En tant que administrateur/chercheur, je veux consulter la liste des équipements.	<ul style="list-style-type: none"> <li>• Implémenter la méthode pour afficher la liste des équipements.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Mettre à jour la tree table pour afficher les équipements.</li> <li>• Consommer et intégrer l'API de la liste des équipements.</li> </ul>	8 1 4 4
S21	En tant que administrateur/chercheur, je veux consulter les détails d'un équipement.	<ul style="list-style-type: none"> <li>• Implémenter la méthode pour afficher les détails d'un équipement.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Implémenter l'interface d'affichage des détails d'un équipement.</li> <li>• Consommer et intégrer l'API de l'affichage des détails d'un équipement.</li> </ul>	4 1 2 4
S22	En tant que administrateur, je veux ajouter un label.	<ul style="list-style-type: none"> <li>• Mettre en place les entités de la base de données</li> <li>• Implémenter la méthode pour ajouter un label.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Consommer et intégrer l'API de l'ajout d'un label.</li> </ul>	6 8 2 4
S23	En tant que administrateur, je veux	<ul style="list-style-type: none"> <li>• Implémenter la méthode pour activer/désactiver un label.</li> </ul>	4

	activer/désactiver un label.	<ul style="list-style-type: none"> <li>• Exposer la méthode en tant que web service.</li> <li>• Consommer et intégrer l'API de l'activation/désactivation d'un label.</li> </ul>	1 4
S24	En tant que administrateur, je veux modifier un label.	<ul style="list-style-type: none"> <li>• Implémenter la méthode pour modifier un label.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Implémenter l'interface de modification d'un label.</li> <li>• Consommer et intégrer l'API de modification d'un label.</li> </ul>	4 1 2 4
S25	En tant que administrateur/chercheur, je veux consulter la liste des labels.	<ul style="list-style-type: none"> <li>• Implémenter la méthode pour afficher la liste des labels.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Implémenter l'interface de l'affichage de la liste des labels.</li> <li>• Consommer et intégrer l'API de la liste des labels.</li> </ul>	4 1 2 4
S26	En tant que administrateur/chercheur, je veux consulter les détails d'un label.	<ul style="list-style-type: none"> <li>• Implémenter la méthode pour afficher les détails d'un label.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Implémenter l'interface d'affichage des détails d'un label.</li> <li>• Consommer et intégrer l'API de l'affichage des détails d'un label.</li> </ul>	4 1 2 4

S27	En tant que administrateur, je veux ajouter un accessoire.	<ul style="list-style-type: none"> <li>• Mettre en place les entités de la base de données</li> <li>• Implémenter la méthode pour ajouter un accessoire.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Consommer et intégrer l'API de l'ajout d'un accessoire.</li> </ul>	6 8 2 4
S28	En tant que administrateur, je veux activer/désactiver un label.	<ul style="list-style-type: none"> <li>• Implémenter la méthode pour activer/désactiver un accessoire.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Consommer et intégrer l'API de l'activation/désactivation d'un accessoire.</li> </ul>	4 1 4
S29	En tant que administrateur, je veux modifier un accessoire.	<ul style="list-style-type: none"> <li>• Implémenter la méthode pour modifier un accessoire.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Implémenter l'interface de modification d'un accessoire.</li> <li>• Consommer et intégrer l'API de modification d'un accessoire.</li> </ul>	4 1 2 4
S30	En tant que administrateur/chercheur, je veux consulter la liste des accessoires.	<ul style="list-style-type: none"> <li>• Implémenter la méthode pour afficher la liste des accessoires.</li> <li>• Exposer la méthode en tant que web service.</li> </ul>	4 1

		<ul style="list-style-type: none"> <li>• Implémenter l'interface de l'affichage de la liste des accessoires.</li> <li>• Consommer et intégrer l'API de la liste des accessoires.</li> </ul>	2 4
S31	En tant que administrateur/chercheur, je veux consulter les détails d'un accessoire.	<ul style="list-style-type: none"> <li>• Implémenter la méthode pour afficher les détails d'un accessoire.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Implémenter l'interface d'affichage des détails d'un accessoire.</li> <li>• Consommer et intégrer l'API de l'affichage des détails d'un accessoire.</li> </ul>	4 1 2 4
S32	En tant que administrateur, je veux consulter la liste des langues.	<ul style="list-style-type: none"> <li>• Implémenter la méthode pour afficher la liste des langues.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Implémenter l'interface de l'affichage de la liste des langues.</li> <li>• Consommer et intégrer l'API de la liste des langues.</li> </ul>	4 1 2 4
S33	En tant que administrateur, je veux activer/désactiver une langue.	<ul style="list-style-type: none"> <li>• Implémenter la méthode pour activer/désactiver une langue.</li> <li>• Exposer la méthode en tant que web service.</li> </ul>	2 1

		<ul style="list-style-type: none"> <li>• Consommer et intégrer l'API de l'activation/désactivation des langues.</li> </ul>	2
--	--	--	---

Tableau 10 : Sprint backlog « Gestion du contenu »

## 2. Diagramme de cas d'utilisation raffiné

La figure 36 illustre le diagramme de cas d'utilisation raffiné pour le module de la gestion des versions.

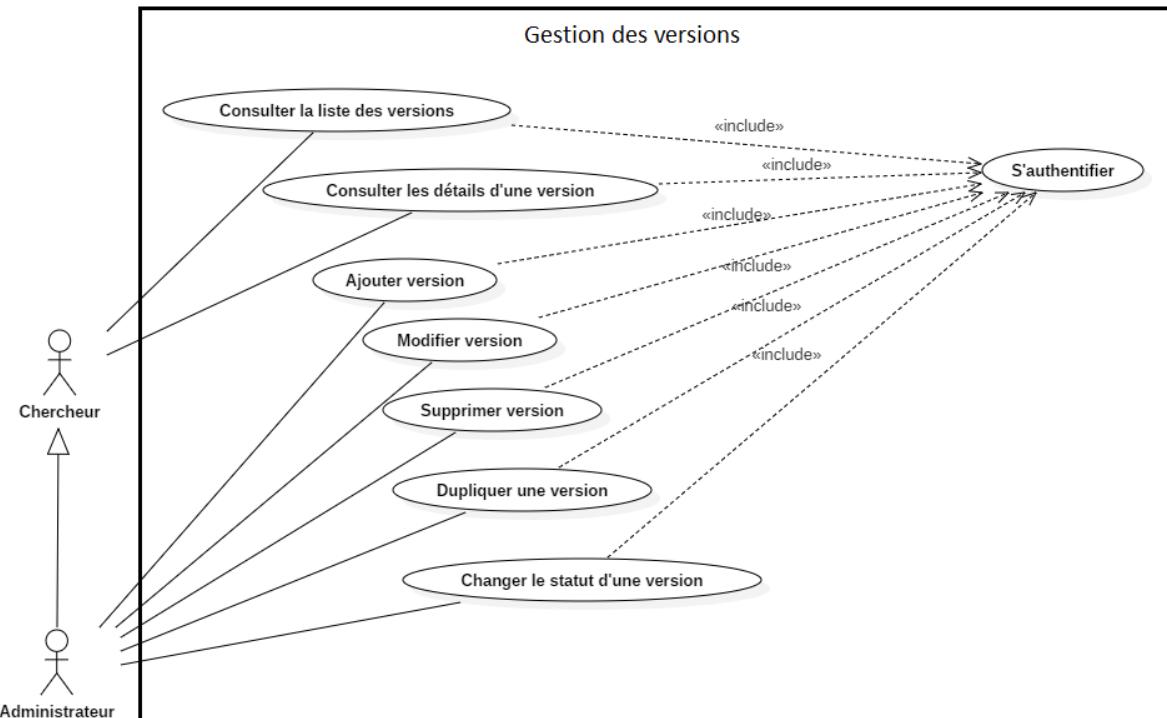
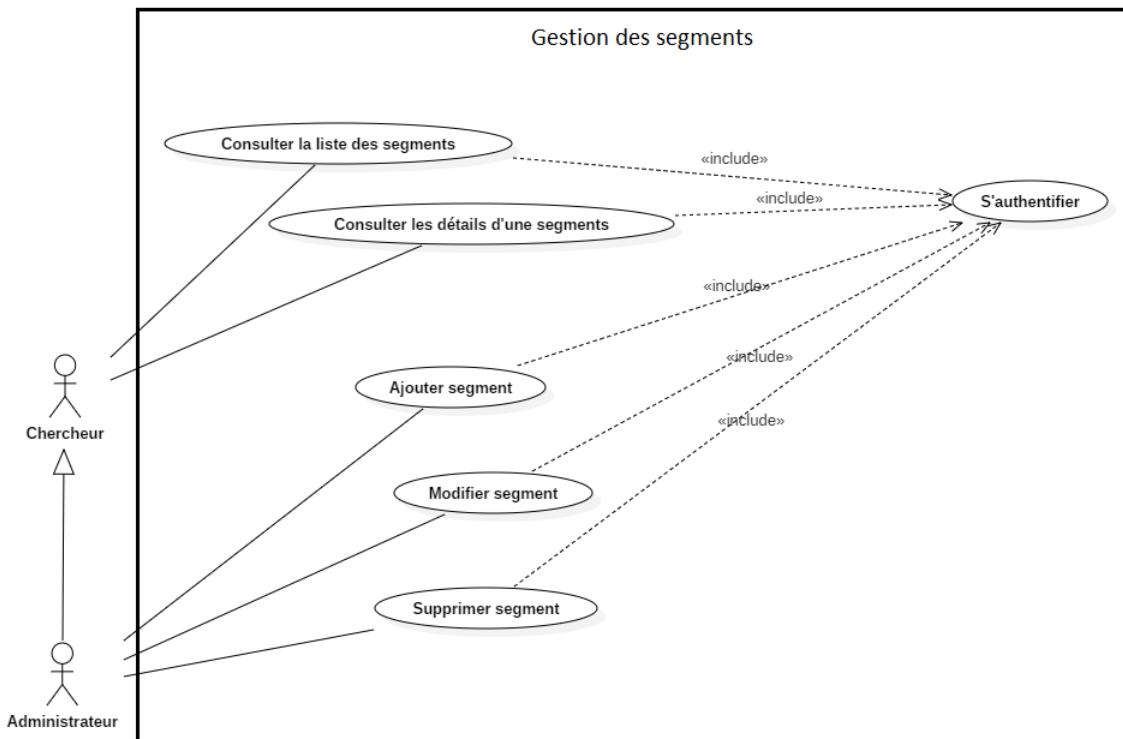


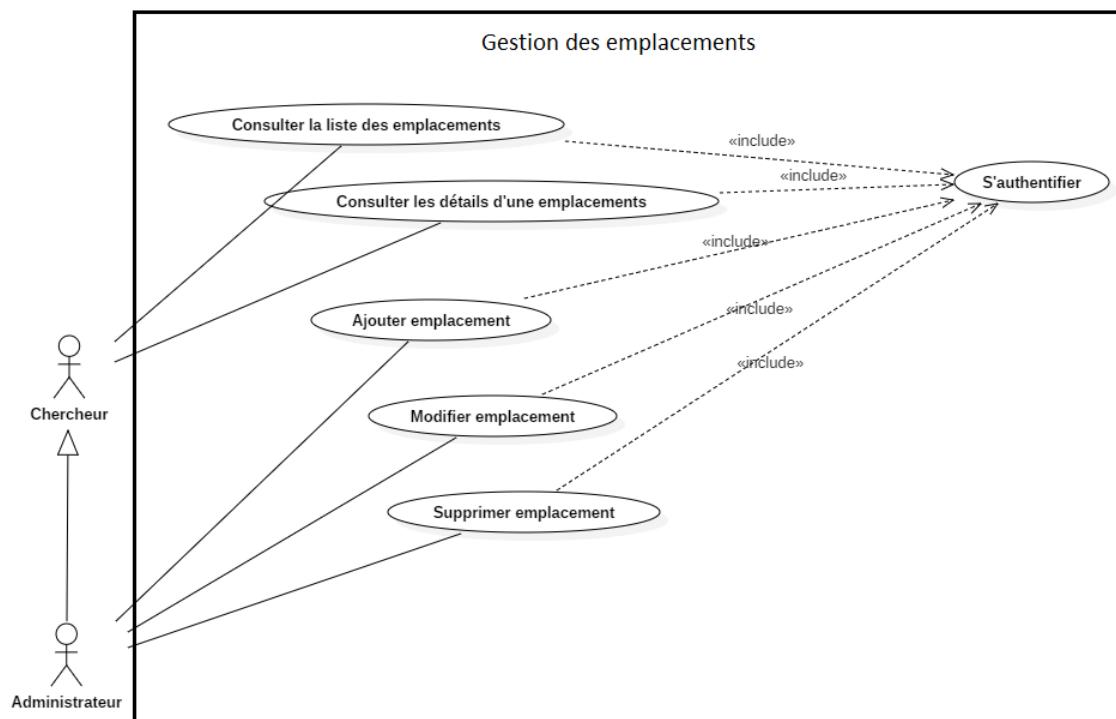
Figure 36 : Diagramme de cas d'utilisation raffiné « gestion des versions »

La figure 37 illustre le diagramme de cas d'utilisation raffiné pour le module de la gestion des segments.



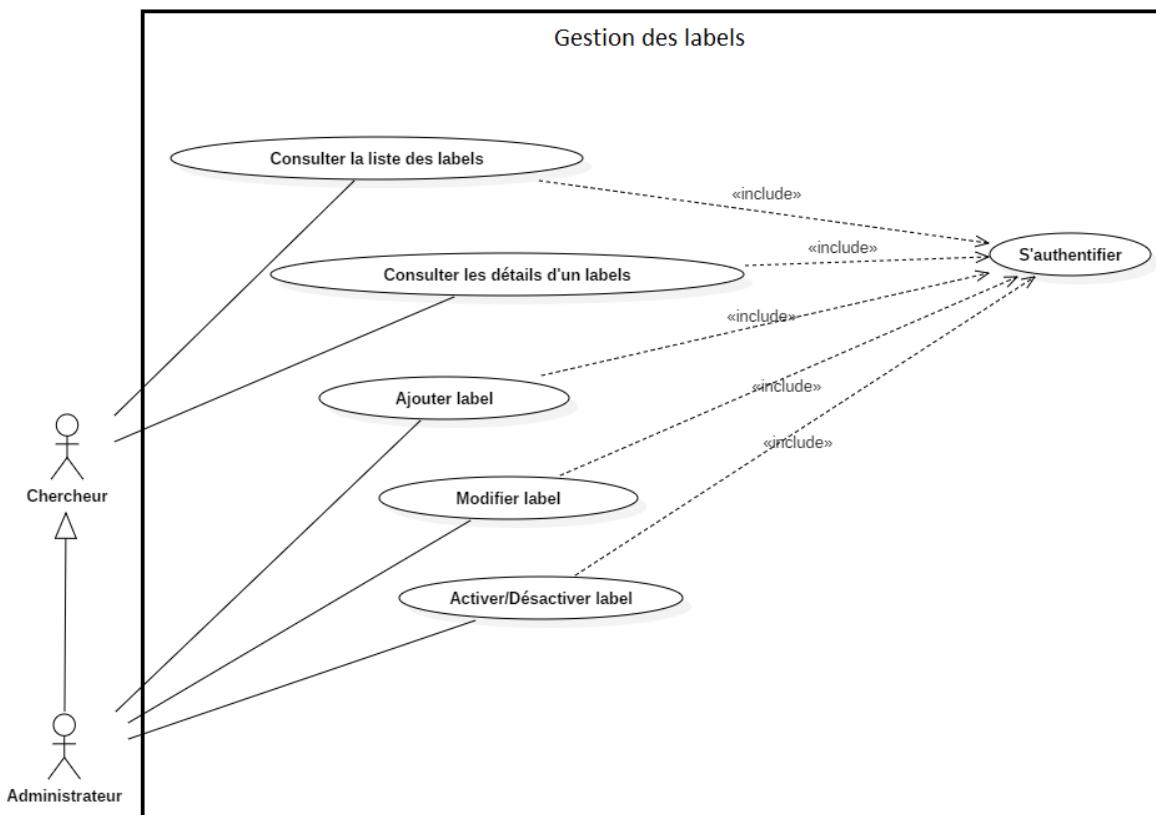
**Figure 37 : Diagramme de cas d'utilisation raffiné « gestion des segments »**

La figure 38 illustre le diagramme de cas d'utilisation raffiné pour le module de la gestion des emplacements.



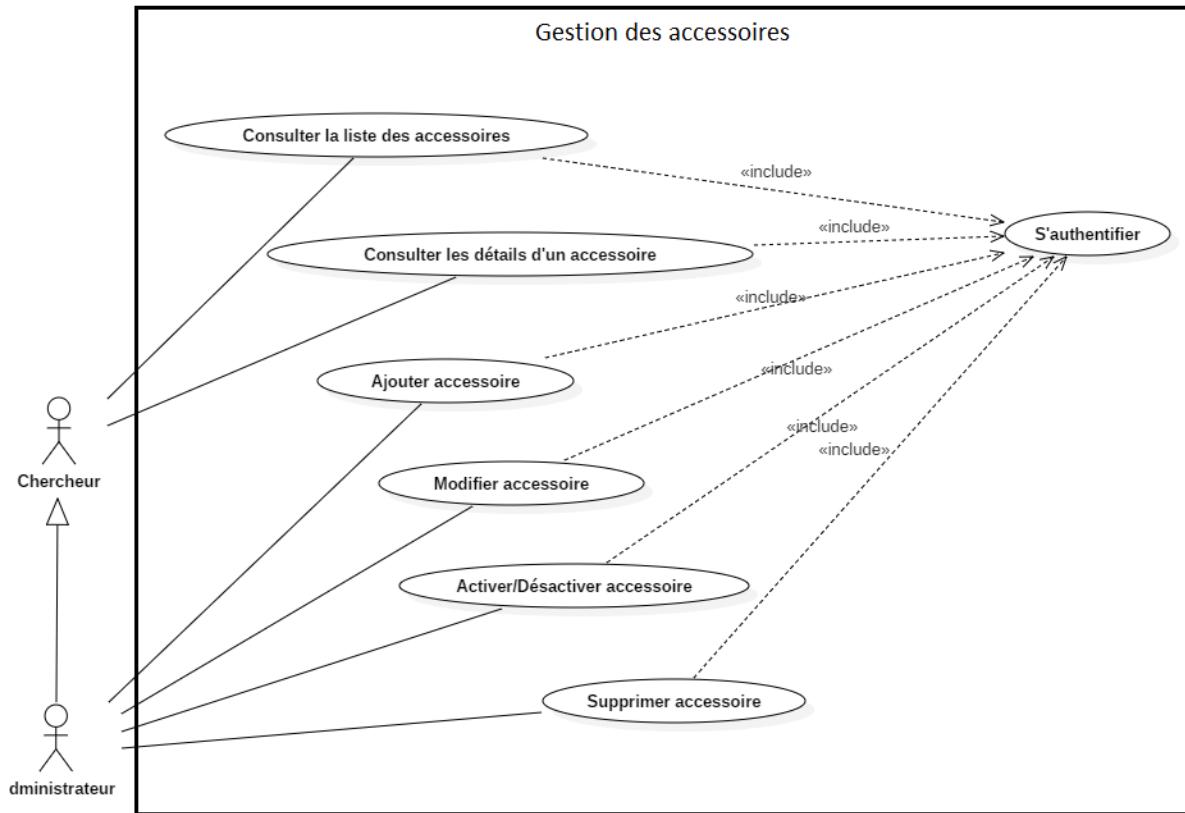
**Figure 38 : Diagramme de cas d'utilisation raffiné « gestion des emplacements »**

La figure 39 illustre le diagramme de cas d'utilisation raffiné pour le module de la gestion des labels.



**Figure 39 : Diagramme de cas d'utilisation raffiné « gestion des labels »**

La figure 40 illustre le diagramme de cas d'utilisation raffiné pour le module de la gestion des accessoires.



**Figure 40 : Diagramme de cas d'utilisation raffiné « gestion des accessoires »**

### 3. Description du cas d'utilisation « Ajouter version »

Dans ce qui suit, nous allons détailler le cas d'utilisation « Ajouter version » en utilisant une description textuelle et un diagramme de séquence.

#### 3.1. Description textuelle du cas d'utilisation « Ajouter version »

<b>Description</b>	L'utilisateur a la possibilité d'ajouter une version
<b>Précondition</b>	L'utilisateur est connecté et a le rôle « administrateur ».
<b>Scénario nominal</b>	<ol style="list-style-type: none"> <li>1) L'utilisateur demande le formulaire d'ajout</li> <li>2) Le système affiche le formulaire d'ajout</li> <li>3) L'utilisateur saisit les données</li> <li>4) Le système vérifie la validité des données saisies</li> <li>5) Le système enregistre les données dans la base de données</li> </ol>
<b>Exceptions</b>	4.1 – Les données ne sont pas valides

	4.1.1 – Le système affiche un message d'erreur et retourne au 2 <sup>ème</sup> point
<b>Post-condition</b>	Version enregistrée dans la base de données

Tableau 11 : Description textuelle du cas d'utilisation « ajouter une version »

### 3.2. Diagramme de séquence du cas d'utilisation « ajouter une version »

La figure 41 représente le diagramme de séquence du cas d'utilisation « ajouter une version » :

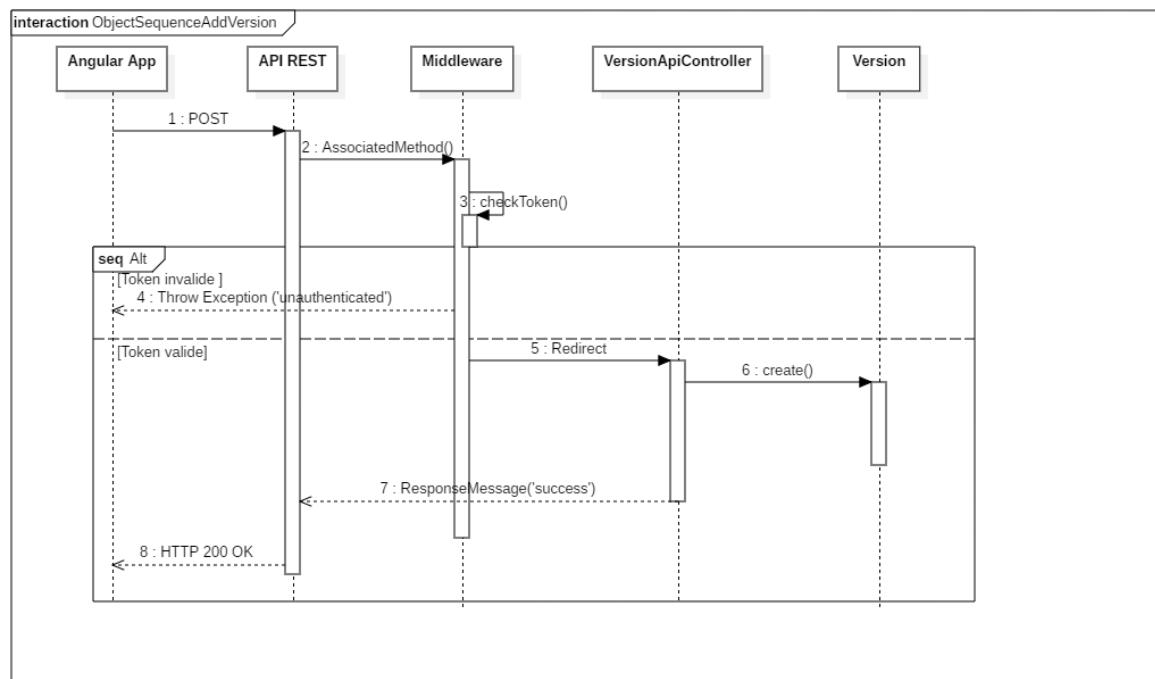


Figure 41 : Diagramme de séquence du cas d'utilisation « ajouter une version»

## 4. Description du cas d'utilisation « modifier une version »

Dans ce qui suit, nous allons détailler le cas d'utilisation « modifier une version » à travers la description textuelle et le diagramme de séquence.

### 4.1. Description textuelle du cas d'utilisation « modifier une version »

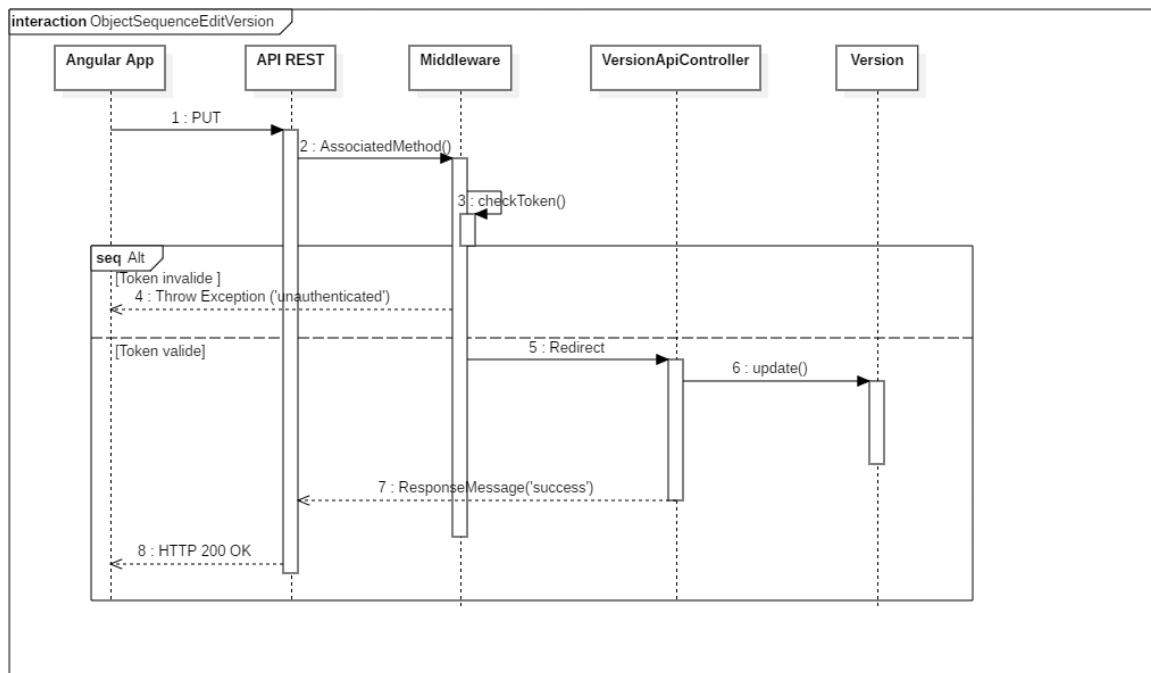
<b>Description</b>	L'utilisateur a la possibilité de modifier une version
<b>Précondition</b>	L'utilisateur est connecté et a le rôle « administrateur ».

<b>Scénario nominal</b>	<ol style="list-style-type: none"> <li>1) L'utilisateur demande le formulaire de modification, ceci peut être effectué de deux manières :             <ol style="list-style-type: none"> <li>a) Dans la page de la liste des versions</li> <li>b) Dans la page des détails d'une version</li> </ol> </li> <li>2) Le système affiche le formulaire de modification</li> <li>3) L'utilisateur saisit les modifications</li> <li>4) Le système vérifie la validité des données saisies</li> <li>5) Le système enregistre les modifications dans la base données</li> </ol>
<b>Exceptions</b>	4.1 – Les données ne sont pas valides 4.1.1 – Le système affiche un message d'erreur et retourne au 2 <sup>ème</sup> point
<b>Post-condition</b>	Modifications enregistrés dans la base de données

**Tableau 12 : Description textuelle du cas d'utilisation « modifier une version »**

#### 4.2. Diagramme de séquence du cas d'utilisation « modifier une version »

La figure 42 représente le diagramme de séquence du cas d'utilisation « modifier une version » :



**Figure 42 : Diagramme de séquence du cas d'utilisation « modifier une version »**

## 5. Description du cas d'utilisation « supprimer une version »

Dans ce qui suit, nous allons détailler le cas d'utilisation « modifier une version » à travers la description textuelle et le diagramme de séquence.

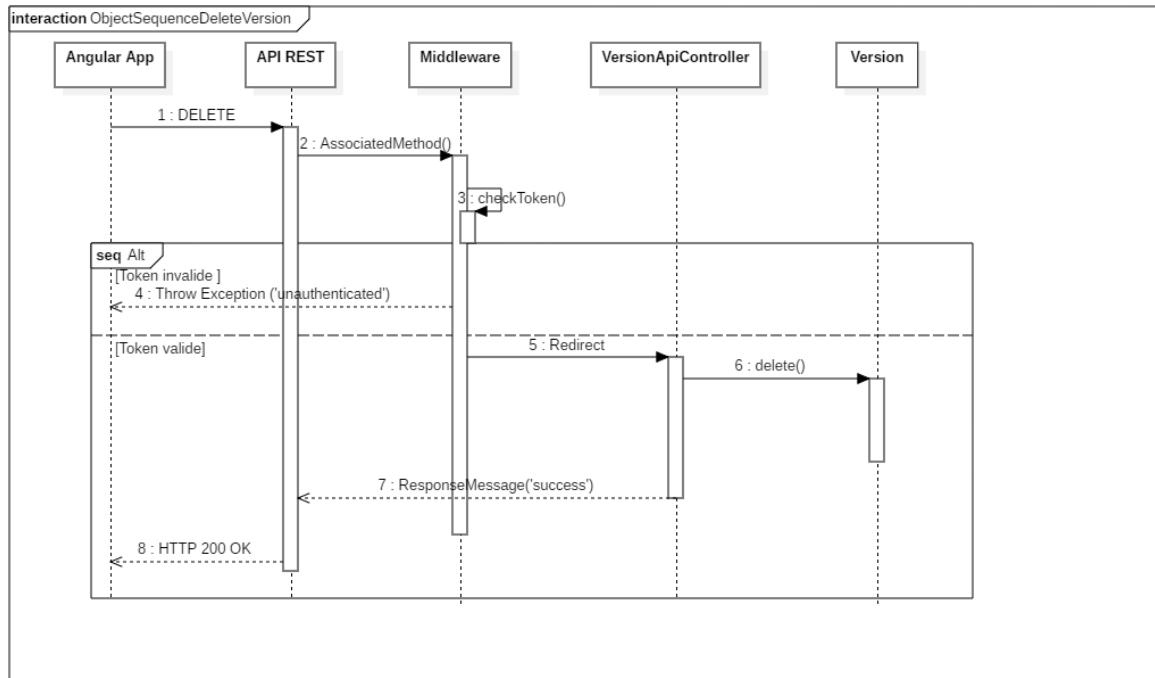
### 5.1. Description textuelle du cas d'utilisation « supprimer une version »

<b>Description</b>	L'utilisateur a la possibilité de supprimer une version
<b>Précondition</b>	L'utilisateur est connecté et a le rôle « administrateur ».
<b>Scénario nominal</b>	<ol style="list-style-type: none"> <li>1) Le système affiche la liste des versions</li> <li>2) L'utilisateur clique sur le bouton « delete » pour une version spécifique</li> <li>3) Le système supprime les données de la version de la base de données</li> <li>4) Le système rafraîchit la liste des versions</li> </ol>
<b>Exceptions</b>	aucune
<b>Post-condition</b>	Version supprimé au niveau de la base de données

Tableau 13 : Description textuelle du cas d'utilisation « supprimer une version »

### 5.2. Diagramme de séquence du cas d'utilisation « supprimer une version »

La figure 43 représente le diagramme de séquence du cas d'utilisation « supprimer une version » :



**Figure 43 : Diagramme de séquence du cas d'utilisation « supprimer une version »**

## 6. Description du cas d'utilisation « changer le statut d'une version »

Dans ce qui suit, nous allons détailler le cas d'utilisation « changer le statut d'une version » à travers la description textuelle et le diagramme de séquence.

### 6.1. Description textuelle du cas d'utilisation « changer le statut d'une version »

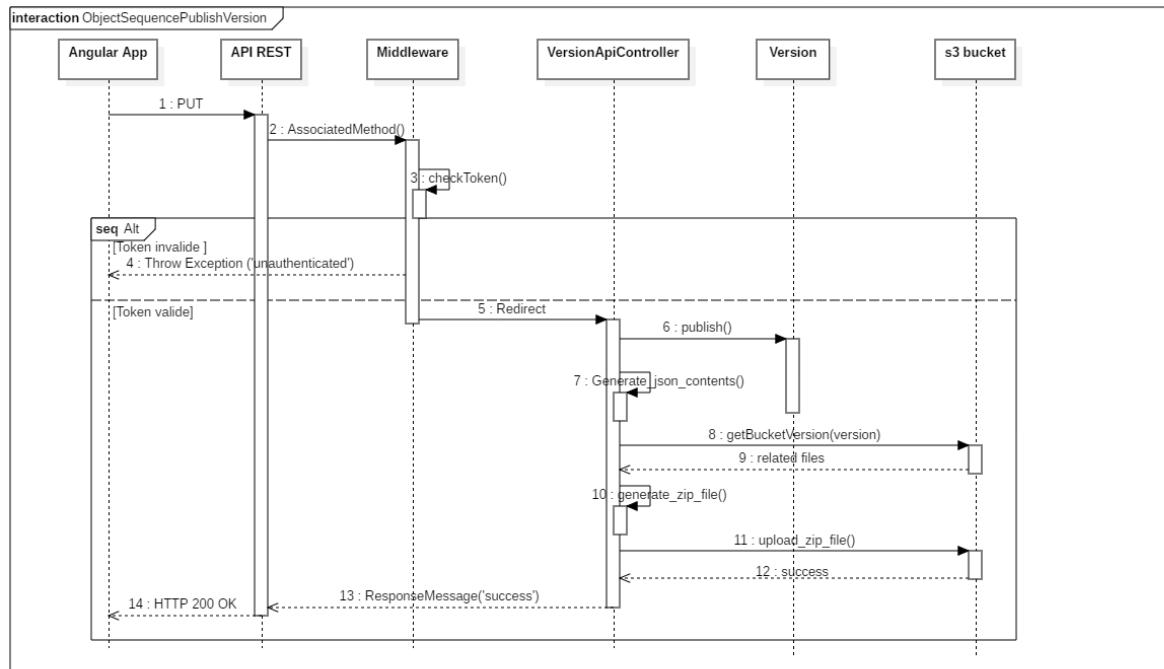
<b>Description</b>	L'utilisateur a la possibilité de changer le statut d'une version
<b>Précondition</b>	L'utilisateur est connecté et a le rôle « administrateur ».
<b>Scénario nominal</b>	<ol style="list-style-type: none"> <li>1) Le système affiche la liste des versions (Le statut par défaut est « offline »)</li> <li>2) L'utilisateur clique sur le bouton « review » pour une version ayant le statut « offline »</li> <li>3) Le système met à jour le statut de la version (offline -&gt; review)</li> <li>4) L'utilisateur clique sur le bouton « publish » pour une version ayant le statut « review »</li> </ol>

	<p>5) Le système :</p> <ul style="list-style-type: none"> <li>- Met à jour le statut de toute autre version publiée appartenant au même site (published-&gt;archived)</li> <li>- Met à jour le statut de la version (review -&gt; published)</li> <li>- Désactive la possibilité de modifier cette version</li> </ul>
<b>Exceptions</b>	aucune
<b>Post-condition</b>	<ul style="list-style-type: none"> <li>- Statuts des versions changés au niveau de la base de données</li> <li>- 4<sup>ème</sup> et 5<sup>ème</sup> point : <ul style="list-style-type: none"> <li>• Fichier .json généré au niveau du serveur, contenant tout le contenu nécessaire pour le jeu mobile (Chapitres, sessions, scenes ...)</li> <li>• Fichier .zip généré au niveau du serveur s3 bucket d'Amazon</li> </ul> </li> <li>-</li> </ul>

**Tableau 14 : Description textuelle du cas d'utilisation « changer le statut d'une version »**

## **6.2. Diagramme de séquence du cas d'utilisation « changer le statut une version »**

La figure 44 représente le diagramme de séquence du cas d'utilisation « changer le statut d'une version » :



**Figure 44 : Diagramme de séquence du cas d'utilisation « changer le statut d'une version »**

## 7. Description du cas d'utilisation « Dupliquer une version »

Dans ce qui suit, nous allons détailler le cas d'utilisation « dupliquer une version » à travers la description textuelle et un schéma explicatif.

### 7.1. Description textuelle du cas d'utilisation « Dupliquer une version »

<b>Description</b>	L'utilisateur a la possibilité de dupliquer une version
<b>Précondition</b>	L'utilisateur est connecté et a le rôle « administrateur ».
<b>Scénario nominal</b>	<ol style="list-style-type: none"> <li>1) Le système affiche la liste des versions</li> <li>2) L'utilisateur clique sur le bouton « Duplicate »</li> <li>3) Le système : <ul style="list-style-type: none"> <li>- Change le bouton de « duplicate » à « in progress »</li> <li>- Commence le processus de duplication</li> </ul> </li> <li>4) L'utilisateur clique sur le bouton « in progress »</li> <li>5) Le système affiche un pop-up contenant l'avancement du processus de duplication en temps réel</li> <li>6) L'utilisateur peut éventuellement cliquer sur « Cancel »</li> <li>7) Le système annule la duplication</li> </ol>

<b>Exceptions</b>	6.1. L'utilisateur n'est pas celui qui a lancé le processus de duplication.
<b>Post-condition</b>	Version dupliquée, avec tous ses chapitres, sessions et scenes.

Tableau 15 : Description textuelle du cas d'utilisation « Dupliquer une version »

## 7.2. Utilisation des WebSockets

Le processus de duplication prend en charge la duplication d'une version donnée et tout son contenu associé (chapitres, sessions, scenes ...) ce qui prend un temps de traitement énorme (environ 10 minutes). En effet, une version contient à peu près : 5 chapitres \* 4 sessions \* 58 scenes = 1160 segments.

Avec notre architecture actuelle, la communication entre le client et le serveur est asynchrone, c'est-à-dire que le client envoie une requête au serveur et doit attendre sa réponse. En d'autres termes, si l'utilisateur choisit de dupliquer une version, il doit attendre environ 10 minutes (le temps que le serveur réponde) avant de pouvoir effectuer une autre opération. D'où la nécessité d'avoir une communication plus réactive et immédiate. Ceci est tout simplement résolu par les **WebSockets**.

WebSocket est une technologie évoluée qui permet d'ouvrir un canal de communication interactif entre un navigateur (côté client) et un serveur. Avec cette API vous pouvez envoyer des messages à un serveur et recevoir ses réponses de manière événementielle sans avoir à aller consulter le serveur pour obtenir une réponse [7].

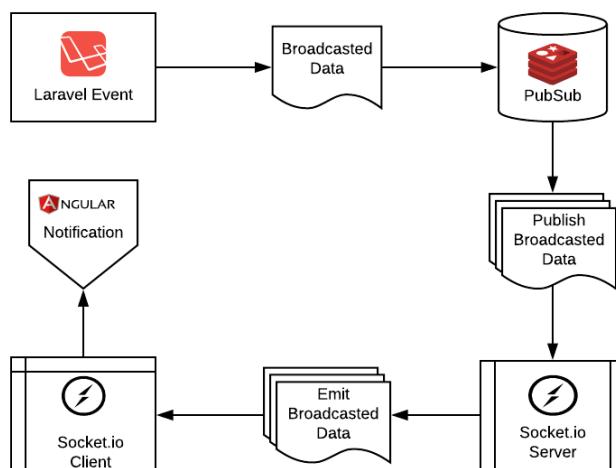


Figure 45 : Flux de diffusion Laravel avec Socket.io et Redis PubSub

Le flux est comme ceci : Un événement est déclenché lorsque nous devons diffuser quelque chose, puis publier des données sur Redis PubSub. Le serveur Socket.io qui écoute et s'abonne au canal Redis PubSub, reçoit les données, puis les émet via un WebSocket. Un script client Socket.io qui a écouté le serveur Socket.io reçoit la diffusion et fait quelque chose avec les données. Dans notre cas, nous informons l'utilisateur sur l'avancement du processus du duplication.

## 8. Diagramme de classe

Nous avons divisé notre diagramme sur deux parties pour une meilleure lisibilité. La figure 46 représente le diagramme de classe relatif au module de gestion des versions et la figure 47 représente le diagramme de classe relatif au module de gestion des emplacements (Buildings, room) et la gestion des cibles (Accessories, equipments, bulletins).

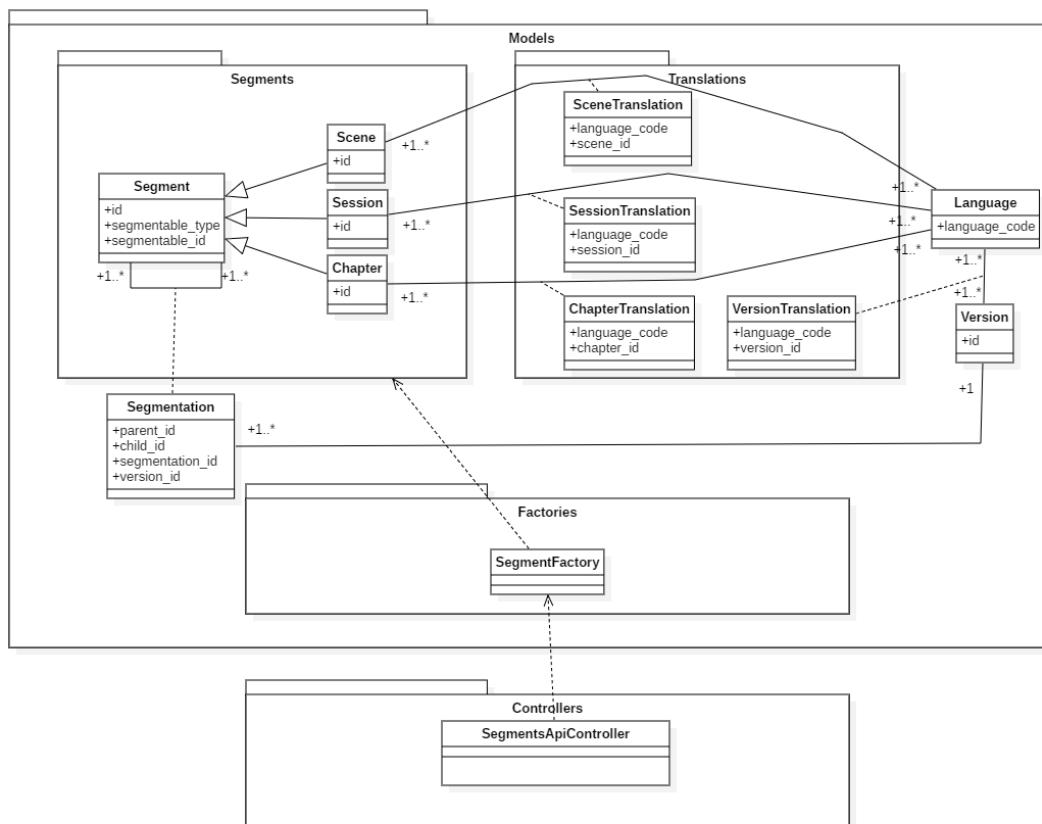
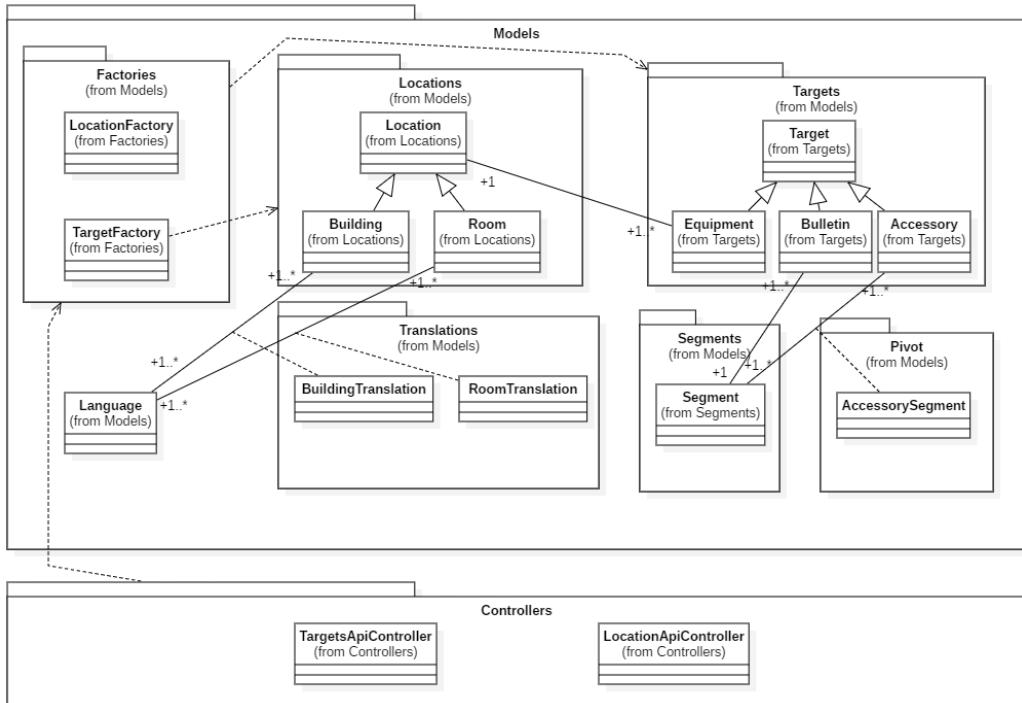


Figure 46 : Diagramme de classe relatif au module gestion du contenu (1<sup>ère</sup> partie)



**Figure 47 : Diagramme de classe relatif au module de gestion du contenu (2ème partie)**

## 9. Réalisation

Après l'étude conceptuelle de notre sprint « Gestion du contenu » à travers les descriptions textuelles, les diagrammes de séquence et le diagramme de classe, nous passerons à la partie de l'implémentation.

### 9.1. Maquettes

Les figures ci-dessous montrent quelques maquettes que nous avons établies afin de les valider avec le product owner.

My Ready Transition

User Name

Users  
---- Administrators  
---- Patients  
---- Researchers  
Languages  
Hierarchy of partitions  
Content Management >  
---- Versions  
---- Buildings  
---- InterfaceLabels  
Notifications >  
---- Automatic notifications  
---- Push notifications  
Analytics

### Versions

ID	Title	Action
Version 01	TRC-2017-ENGLISH	<a href="#">VIEW</a> <a href="#">EDIT</a> <a href="#">REMOVE</a> <a href="#">DUPLICATE</a> <a href="#">PUBLISHED</a>
Chapter 01	Introduction to Transition	<a href="#">VIEW</a> <a href="#">EDIT</a> <a href="#">REMOVE</a> <a href="#">DUPLICATE</a>
Session A	Introduction to the App	<a href="#">VIEW</a> <a href="#">EDIT</a> <a href="#">REMOVE</a> <a href="#">DUPLICATE</a>
Session B	The Basics: What is Transition?	<a href="#">VIEW</a> <a href="#">EDIT</a> <a href="#">REMOVE</a> <a href="#">DUPLICATE</a>
Scene 01	THELOUNGE-ENTRANCE-MENTOR MESSAGE	<a href="#">VIEW</a> <a href="#">EDIT</a> <a href="#">REMOVE</a> <a href="#">DUPLICATE</a>
Scene 02	THELOUNGE-ENTRANCE-POP-UP QUIZZ	<a href="#">VIEW</a> <a href="#">EDIT</a> <a href="#">REMOVE</a> <a href="#">DUPLICATE</a>
Scene 03	THELOUNGE-ENTRANCE-MENTOR MESSAGE	<a href="#">VIEW</a> <a href="#">EDIT</a> <a href="#">REMOVE</a> <a href="#">DUPLICATE</a>
Scene 04	THELOUNGE-ENTRANCE-RECEIVE COINS	<a href="#">VIEW</a> <a href="#">EDIT</a> <a href="#">REMOVE</a> <a href="#">DUPLICATE</a>
Scene 05	THELOUNGE-FREE	<a href="#">VIEW</a> <a href="#">EDIT</a> <a href="#">REMOVE</a> <a href="#">DUPLICATE</a>
Scene 06	CITY-FREE	<a href="#">VIEW</a> <a href="#">EDIT</a> <a href="#">REMOVE</a> <a href="#">DUPLICATE</a>
Scene 07	PEDIATRIC-HOSPITAL-KIDS LOUNGE-MENTOR MESSAGE	<a href="#">VIEW</a> <a href="#">EDIT</a> <a href="#">REMOVE</a> <a href="#">DUPLICATE</a>
Scene 08	PEDIATRIC-HOSPITAL-FREE	<a href="#">VIEW</a> <a href="#">EDIT</a> <a href="#">REMOVE</a> <a href="#">DUPLICATE</a>
Scene 09	PEDIATRIC-HOSPITAL-KIDS LOUNGE-TV	<a href="#">VIEW</a> <a href="#">EDIT</a> <a href="#">REMOVE</a> <a href="#">DUPLICATE</a>
Scene 10	PEDIATRIC-HOSPITAL-KIDS LOUNGE-RECEIVE COINS	<a href="#">VIEW</a> <a href="#">EDIT</a> <a href="#">REMOVE</a> <a href="#">DUPLICATE</a>
<a href="#">ADD NEW</a>		
Session C	Paediatrics to Adult Health Care Experiences	<a href="#">VIEW</a> <a href="#">EDIT</a> <a href="#">REMOVE</a> <a href="#">DUPLICATE</a>
Session D	Building Your Own PathBridge	<a href="#">VIEW</a> <a href="#">EDIT</a> <a href="#">REMOVE</a> <a href="#">DUPLICATE</a>
Chapter 02	Knowledge is Power	<a href="#">VIEW</a> <a href="#">EDIT</a> <a href="#">REMOVE</a> <a href="#">DUPLICATE</a>
Chapter 03	Communication is Key	<a href="#">VIEW</a> <a href="#">EDIT</a> <a href="#">REMOVE</a> <a href="#">DUPLICATE</a>
Chapter 04	Time to Take Charge	<a href="#">VIEW</a> <a href="#">EDIT</a> <a href="#">REMOVE</a> <a href="#">DUPLICATE</a>
Chapter 05	Good 2 Go	<a href="#">VIEW</a> <a href="#">EDIT</a> <a href="#">REMOVE</a> <a href="#">DUPLICATE</a>
Version 02	TRC-2017-FRENCH	<a href="#">VIEW</a> <a href="#">EDIT</a> <a href="#">REMOVE</a> <a href="#">DUPLICATE</a> <a href="#">PUBLISHED</a>
Version 02	TRC-2016-ENGLISH	<a href="#">VIEW</a> <a href="#">EDIT</a> <a href="#">REMOVE</a> <a href="#">DUPLICATE</a> <a href="#">PUBLISH</a>
Version 02	TRC-2017-FRENCH	<a href="#">VIEW</a> <a href="#">EDIT</a> <a href="#">REMOVE</a> <a href="#">DUPLICATE</a> <a href="#">PUBLISH</a>

Figure 48 : Maquette de la liste des versions

**Versions**

Code \*

Description

Division \*  
    
 This is selected by default

New empty version  From previous version

Languages \*

Please select a division from the list of divisions above

Region \*  
    
 This will appear if "New empty version" is selected and this is a multiple select box

This will appear if "From previous version" is selected And there will be an warning message displayed if

Figure 49 : Maquette de la page d'ajout d'une version

## Chapters

Name \*

Code \*

Description

Accomplishment Trophy \*

English French

Figure 50 : Maquette d'ajout d'un chapitre

## Sessions

Name \*

Code \*

Description

Accomplishment Plaque \*

English French

Accessory \*

Assign this Accessory to \*

1) Head accessories

2) T-shirt and arms accessories

3) Pants and shoes accessories

Figure 51 : Maquette d'ajout d'une session

## 9.2. Interfaces graphiques

La figure 52 représente l'interface de la gestion des versions du jeu, chaque version contient une liste de chapitres, chaque chapitre contient une liste de sessions et chaque session contient une liste de scènes.

Name	Code	Status	Action
Version 1	version1	<span>Published</span>	<span>VIEW</span> <span>EDIT</span> <span>REMOVE</span> <span>DUPLICATE</span>
Introduction to Transition	introduction_to_transiti	<span>active</span>	<span>VIEW</span> <span>EDIT</span> <span>REMOVE</span>
Getting Started	ready_to_be_myready	<span>active</span>	<span>VIEW</span> <span>EDIT</span> <span>REMOVE</span>
MentorMessage (Scene 2)		<span>active</span>	<span>VIEW</span> <span>EDIT</span> <span>REMOVE</span>
Free (Scene 3)		<span>active</span>	<span>VIEW</span> <span>EDIT</span> <span>REMOVE</span>
MentorVideo (Scene 4)		<span>active</span>	<span>VIEW</span> <span>EDIT</span> <span>REMOVE</span>
MentorMessage (Scene 5)		<span>active</span>	<span>VIEW</span> <span>EDIT</span> <span>REMOVE</span>
SelectMentor (Scene 6)		<span>active</span>	<span>VIEW</span> <span>EDIT</span> <span>REMOVE</span>
MentorMessage (Scene 7)		<span>active</span>	<span>VIEW</span> <span>EDIT</span> <span>REMOVE</span>

**Figure 52 : Interface de la gestion des versions**

Après avoir cliqué sur le bouton « add version », l'utilisateur est redirigé vers la page d'ajout d'une version comme le montre la figure 53:

**Figure 53 : Interface d'ajout d'une version**

Après avoir ajouté une version, l'utilisateur peut y ajouter des chapitres. La figure 54 montre l'interface de l'ajout d'un chapitre :

**Figure 54 : Interface d'ajout d'un chapitre**

L'utilisateur peut également ajouter des sessions pour un chapitre. La figure 55 montre l'interface d'ajout d'une session :

**Figure 55 : Interface d'ajout d'une session**

Lors de l'ajout d'une session, nous pouvons optionnellement ajouter des dictionnaires. Un dictionnaire peut être soit commun pour toutes les maladies, c'est-à-dire tous les patients auront le même contenu, soit spécifique pour chaque maladie, et là, chaque patient aura un contenu spécifique selon sa maladie (et sa langue, bien entendu.).

The screenshot shows the ChildBright application interface. On the left, a red sidebar contains navigation links: Users, Languages, Hierarchy of partitions, Content Management (with a dropdown), Versions, Buildings, InterfacesLabels, Accessories, Notifications (with a dropdown), and FAQ. The main area is titled 'Dictionary n°1' and shows 'MyDictionary' selected in the dropdown. It includes options for 'Simple' (selected) and 'Complex'. A 'Name' field contains 'Bulletin 1' and a 'Code' field contains 'bulletin\_1'. There is a checkbox for 'Show all answers'. Below this is a 'Content' section with tabs for 'ENGLISH' (selected) and 'FRENCH'. Under 'ENGLISH', terms like 'CEREBRAL PALSY', 'AUTISM SPECTRUM DISORDER', 'FETAL ALCOHOL SYNDROME', 'EPILEPSY', and 'SPINA BIFIDA' are listed. A toolbar below the tabs includes icons for text styling (B, I, U, etc.) and alignment. A text input field says 'Insert text here ...'. At the bottom right, there is a note about activating Windows.

**Figure 56 : Interface d'ajout d'un dictionnaire dans une session**

The screenshot shows the ChildBright application interface. The sidebar is identical to Figure 56. The main area is titled 'Languages' and shows a list of languages: Arabic (ar), English (en), French (fr), and Spanish (es). Each language entry has a status toggle switch. Below the table, it says 'Showing 1 to 4 of 4 entries'. At the bottom right, there is a note about activating Windows.

**Figure 57 : Interface de gestion des langues**

## Conclusion

Dans ce chapitre, nous nous sommes intéressés à la réalisation du module de la gestion du contenu. Pour ce faire, nous avons commencé par l'élaboration du sprint backlog, puis une étude conceptuelle et finalement la réalisation. Dans le chapitre suivant nous allons passer au 3ème sprint qui est la gestion des divisions, des notifications et des FAQs.

# **Chapitre 5 : Sprint « Gestion des divisions, des notifications et des FAQ »**

## Introduction

Le but de ce sprint est de mettre en œuvre les différentes fonctionnalités qui vont servir à gérer les divisions, régions et sites auxquels un patient peut appartenir, d'une part. D'autre part, nous allons implémenter deux autres modules, l'un pour gérer les FAQ de l'application et l'autre pour envoyer des notifications aux patients. Pour ce faire, nous allons élaborer le sprint backlog et les diagrammes relatifs à la conception, puis nous clôturerons cette partie par des captures d'écran expliquant la réalisation.

### 1. Sprint backlog

Dans cette partie, nous allons élaborer les différents user stories de ce sprint, ainsi que les tâches qui devront être accomplies à la fin du sprint et leurs estimations.

ID	User story	Tâche	Estimation(H)
S1	En tant que administrateur, je veux ajouter une partition (Division, région ou site).	<ul style="list-style-type: none"> <li>Mettre en place les entités de la base de données</li> <li>Implémenter la méthode pour ajouter une partition.</li> <li>Exposer la méthode en tant que web service.</li> <li>Implémenter l'interface de l'ajout d'une partition.</li> <li>Consommer et intégrer l'API de l'ajout d'une partition.</li> </ul>	6 4 1 2 4
S2	En tant que administrateur, je veux supprimer une partition.	<ul style="list-style-type: none"> <li>Implémenter la méthode pour supprimer une partition.</li> <li>Exposer la méthode en tant que web service.</li> <li>Consommer et intégrer l'API de la suppression d'une partition.</li> </ul>	2 1 2
S3	En tant que administrateur, je veux modifier une partition.	<ul style="list-style-type: none"> <li>Implémenter la méthode pour modifier une partition.</li> <li>Exposer la méthode en tant que web service.</li> </ul>	4 1

		<ul style="list-style-type: none"> <li>• Implémenter l'interface de modification d'une partition.</li> <li>• Consommer et intégrer l'API de la modification d'une partition.</li> </ul>	2 4
S4	En tant que administrateur/chercheur, je veux consulter la liste des partitions.	<ul style="list-style-type: none"> <li>• Implémenter la méthode pour afficher la liste des partitions.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Implémenter l'interface d'affichage de la liste des partitions.</li> <li>• Consommer et intégrer l'API de la liste des partitions.</li> </ul>	4 1 2 4
S5	En tant que administrateur/chercheur, je veux consulter les détails d'une partition.	<ul style="list-style-type: none"> <li>• Implémenter la méthode pour afficher les détails d'une partition.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Implémenter l'interface d'affichage des détails d'une partition.</li> <li>• Consommer et intégrer l'API de l'affichage des détails d'une partition.</li> </ul>	4 1 2 4
S6	En tant que administrateur, je veux ajouter une FAQ (Question/Réponse).	<ul style="list-style-type: none"> <li>• Mettre en place les entités de la base de données</li> <li>• Implémenter la méthode pour ajouter une FAQ.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Implémenter l'interface de l'ajout d'une FAQ.</li> </ul>	6 4 1 2 4

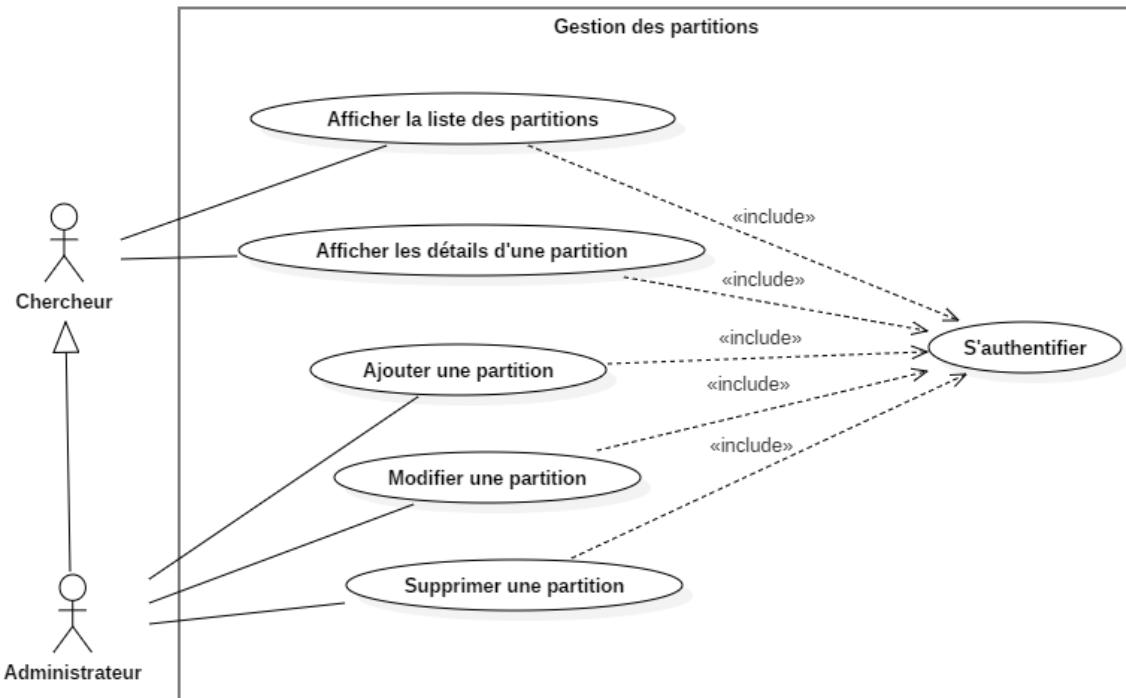
		<ul style="list-style-type: none"> <li>• Consommer et intégrer l'API de l'ajout d'une FAQ.</li> </ul>	
S7	En tant que administrateur, je veux supprimer une FAQ.	<ul style="list-style-type: none"> <li>• Implémenter la méthode pour supprimer une FAQ.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Consommer et intégrer l'API de la suppression d'une FAQ.</li> </ul>	2 1 2
S8	En tant que administrateur, je veux modifier une FAQ.	<ul style="list-style-type: none"> <li>• Implémenter la méthode pour modifier une FAQ.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Implémenter l'interface de modification d'une FAQ.</li> <li>• Consommer et intégrer l'API de la modification d'une FAQ.</li> </ul>	4 1 2 4
S9	En tant que administrateur, je veux consulter la liste des FAQ.	<ul style="list-style-type: none"> <li>• Implémenter la méthode pour afficher la liste des FAQ.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Implémenter l'interface d'affichage de la liste des FAQ.</li> <li>• Consommer et intégrer l'API de la liste des FAQ.</li> </ul>	4 1 2 4
S10	En tant que administrateur, je souhaite envoyer des notifications push aux appareils des patients.	<ul style="list-style-type: none"> <li>• Configurer Firebase.</li> <li>• Mettre en place les entités de la base de données.</li> <li>• Implémenter la méthode pour envoyer des notifications push.</li> <li>• Exposer la méthode en tant que web service.</li> </ul>	2 4 4 1

		<ul style="list-style-type: none"> <li>• Implémenter l'interface d'envoi des notifications push.</li> <li>• Consommer et intégrer l'API de l'envoi des notifications push.</li> </ul>	2 4
S11	En tant que administrateur, je souhaite préconfigurer des notifications (reminders) à envoyer automatiquement aux appareils des patients.	<ul style="list-style-type: none"> <li>• Mettre en place les entités de la base de données.</li> <li>• Implémenter la méthode pour configurer des notifications automatiques.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Implémenter l'interface de configuration des notifications automatiques.</li> <li>• Consommer et intégrer l'API de l'envoi des notifications automatiques.</li> </ul>	4 4 1 3 4
S12	En tant que administrateur, je souhaite modifier les termes d'utilisation et la politique de confidentialité du jeu.	<ul style="list-style-type: none"> <li>• Mettre en place les entités de la base de données.</li> <li>• Implémenter la méthode pour la modification des termes d'utilisation et la politique de confidentialité du jeu.</li> <li>• Exposer la méthode en tant que web service.</li> <li>• Implémenter l'interface de la modification des termes d'utilisation et la politique de confidentialité du jeu.</li> <li>• Consommer et intégrer l'API.</li> </ul>	4 4 1 2 4

**Tableau 16 : Sprint backlog 3ème sprint**

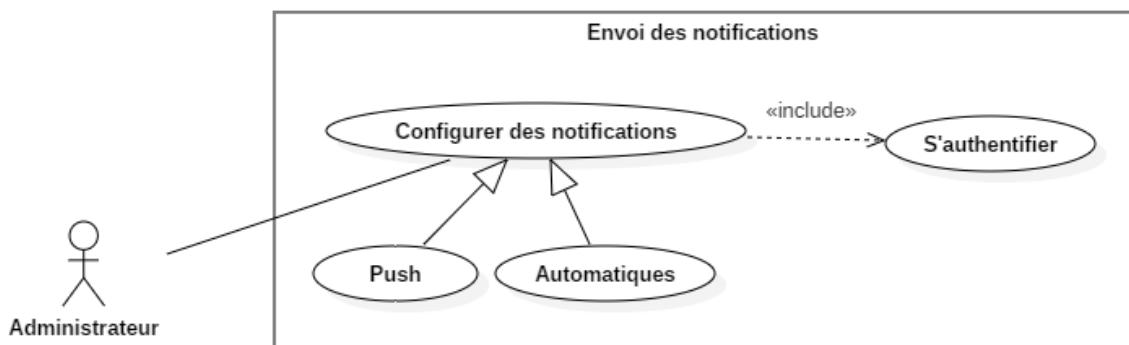
## 2. Diagramme de cas d'utilisation raffiné

La figure 58 illustre le diagramme de cas d'utilisation raffiné pour le module de la gestion des partitions.



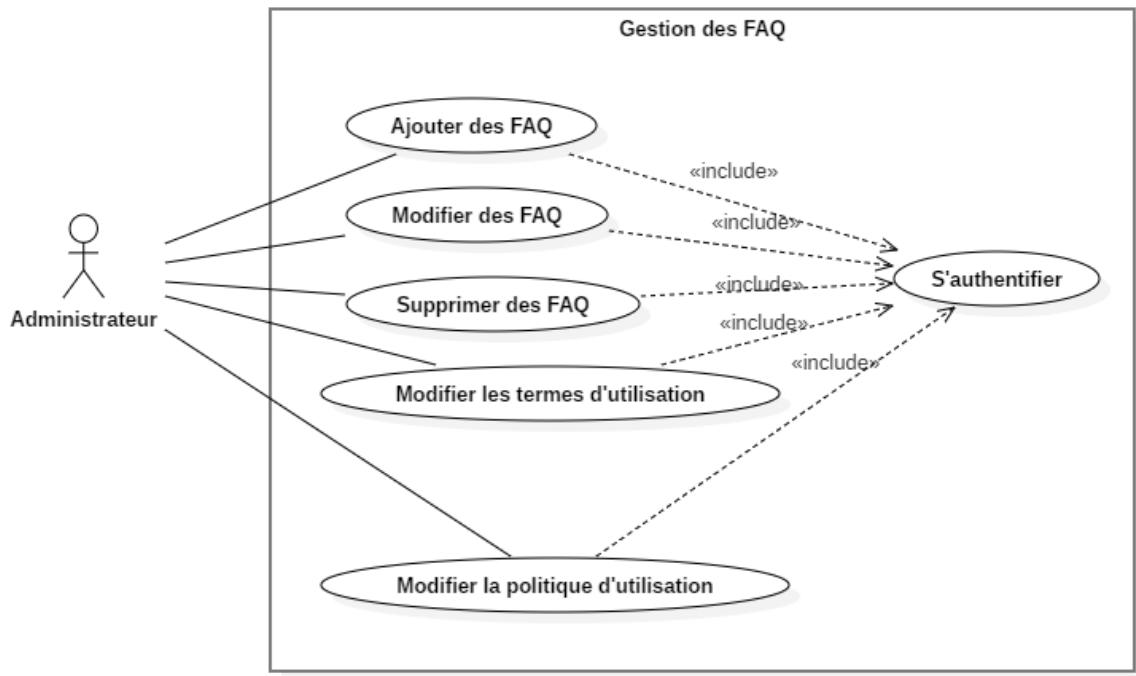
**Figure 58 : Diagramme de cas d'utilisation raffiné du module « Gestion des partitions »**

La figure 59 illustre le diagramme de cas d'utilisation raffiné pour le module de l'envoi des notifications.



**Figure 59 : Diagramme de cas d'utilisation raffiné « envoi des notifications »**

La figure ci-60 illustre le diagramme de cas d'utilisation raffiné pour le module de la gestion des FAQ.



**Figure 60 : Diagramme de cas d'utilisation raffiné « gestion des FAQ »**

### 3. Description du cas d'utilisation « ajouter une partition »

Dans ce qui suit, nous allons détailler le cas d'utilisation « ajouter une partition » à travers la description textuelle et le diagramme de séquence.

#### 3.1. Description textuelle du cas d'utilisation « Ajouter une partition »

<b>Description</b>	L'utilisateur a la possibilité d'ajouter une partition
<b>Précondition</b>	L'utilisateur est connecté et a le rôle « administrateur ».
<b>Scénario nominal</b>	<ol style="list-style-type: none"> <li>1) L'utilisateur demande le formulaire d'ajout</li> <li>2) Le système affiche le formulaire d'ajout</li> <li>3) L'utilisateur saisit les données</li> <li>4) Le système vérifie la validité des données saisies</li> <li>5) Le système enregistre les données dans la base de données</li> </ol>
<b>Exceptions</b>	<p>4.1 – Les données ne sont pas valides</p> <p>4.1.1 – Le système affiche un message d'erreur et retourne au 2<sup>ème</sup> point</p>

<b>Post-condition</b>	Partition enregistré dans la base de données
-----------------------	--

Tableau 17 : Description textuelle du cas d'utilisation « ajouter une partition »

### 3.2. Diagramme de séquence du cas d'utilisation « ajouter une partition »

La figure 61 représente le diagramme de séquence du cas d'utilisation « ajouter une partition » :

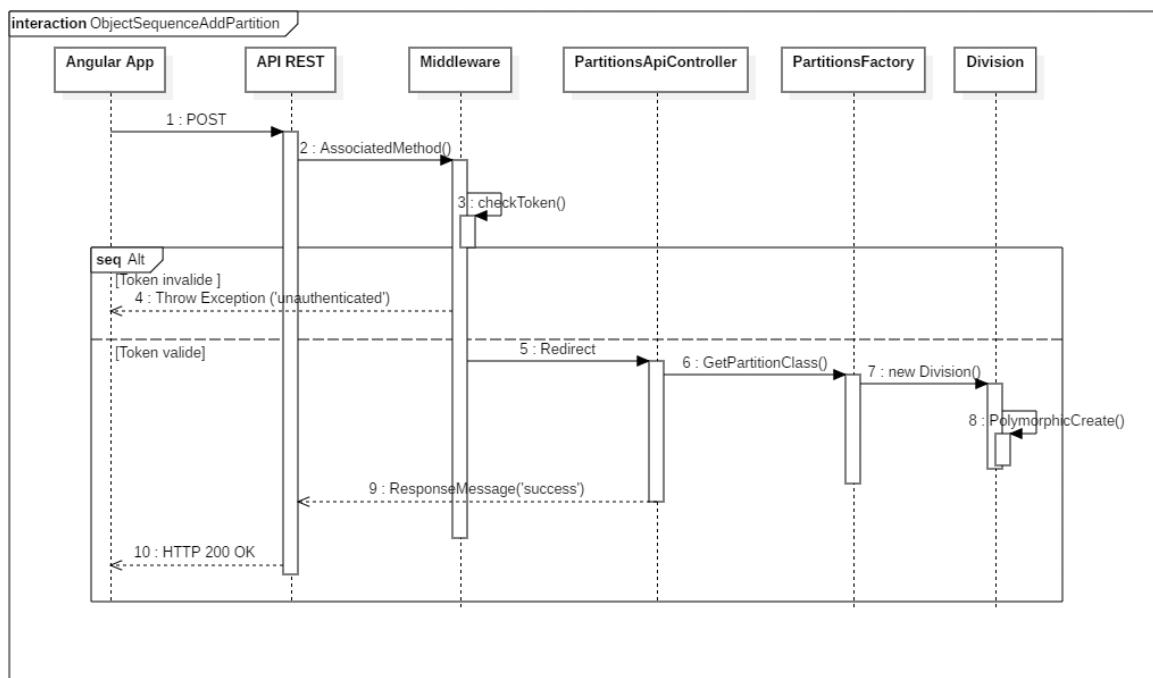


Figure 61 : Diagramme de séquence du cas d'utilisation « ajouter une partition »

## 4. Description du cas d'utilisation « modifier une partition »

Dans ce qui suit, nous allons détailler le cas d'utilisation « modifier une partition » à travers la description textuelle et le diagramme de séquence.

### 4.1. Description textuelle du cas d'utilisation « modifier une partition »

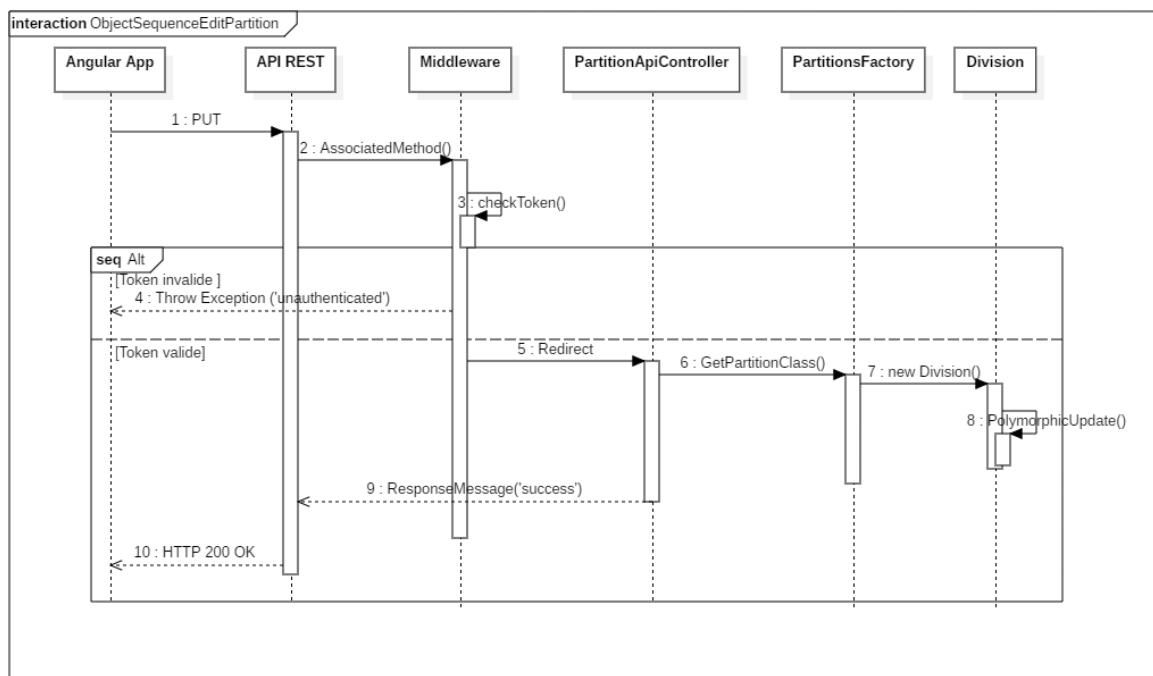
<b>Description</b>	L'utilisateur a la possibilité de modifier une partition
<b>Précondition</b>	L'utilisateur est connecté et a le rôle « administrateur ».
<b>Scénario nominal</b>	1) L'utilisateur demande le formulaire de modification, ceci peut être effectué de deux manières :

	a) Dans la page de la liste des partitions b) Dans la page des détails d'une partition 2) Le système affiche le formulaire de modification 3) L'utilisateur saisit les modifications 4) Le système vérifie la validité des données saisies 5) Le système enregistre les modifications dans la base données
<b>Exceptions</b>	4.1 – Les données ne sont pas valides  4.1.1 – Le système affiche un message d'erreur et retourne au 2 <sup>ème</sup> point
<b>Post-condition</b>	Modifications enregistrés dans la base de données

**Tableau 18 : Description textuelle du cas d'utilisation « modifier une partition »**

#### **4.2. Diagramme de séquence du cas d'utilisation « modifier une partition»**

La figure 62 représente le diagramme de séquence du cas d'utilisation « modifier une partition » :



**Figure 62 : Diagramme de séquence du cas d'utilisation « modifier une partition »**

## 5. Description du cas d'utilisation « supprimer une partition »

Dans ce qui suit, nous allons détailler le cas d'utilisation « supprimer une partition » à travers la description textuelle et le diagramme de séquence.

### 5.1. Description textuelle du cas d'utilisation « supprimer une partition »

<b>Description</b>	L'utilisateur a la possibilité de supprimer une partition
<b>Précondition</b>	L'utilisateur est connecté et a le rôle « administrateur ».
<b>Scénario nominal</b>	<ol style="list-style-type: none"> <li>1) Le système affiche la liste des partitions</li> <li>2) L'utilisateur clique sur le bouton « delete » pour une partition spécifique</li> <li>3) Le système supprime les données de la partition de la base de données</li> <li>4) Le système rafraîchit la liste des partitions</li> </ol>
<b>Exceptions</b>	aucune
<b>Post-condition</b>	Partition supprimée au niveau de la base de données

Tableau 19 : Description textuelle du cas d'utilisation « supprimer une partition »

### 5.2. Diagramme de séquence du cas d'utilisation « supprimer une partition »

La figure 63 représente le diagramme de séquence du cas d'utilisation « supprimer une partition » :

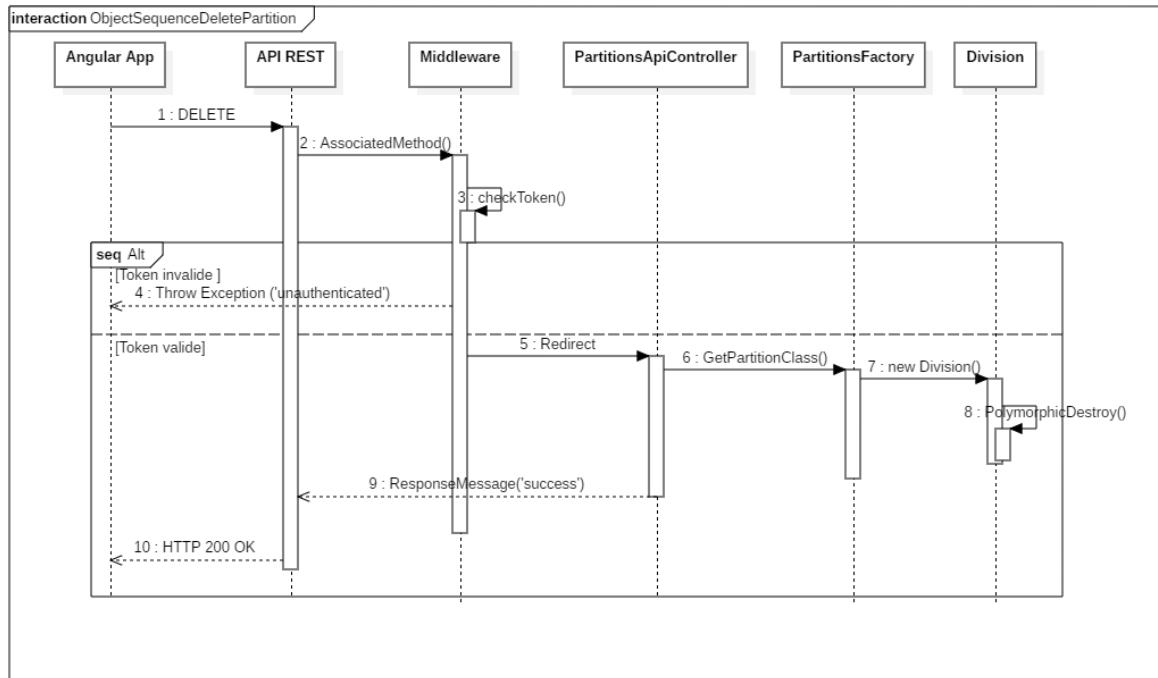


Figure 63 : Diagramme de séquence du cas d'utilisation « supprimer une partition »

## 6. Description du cas d'utilisation « envoyer une notification push »

»

Dans ce qui suit, nous allons détailler le cas d'utilisation « envoyer une notification push » à travers la description textuelle et le diagramme de séquence.

### 6.1. Description textuelle du cas d'utilisation « envoyer une notification push »

<b>Description</b>	L'utilisateur a la possibilité d'envoyer une notification push
<b>Précondition</b>	L'utilisateur est connecté et a le rôle « administrateur ».
<b>Scénario nominal</b>	<ol style="list-style-type: none"> <li>1) L'utilisateur demande la page d'envoi de notifications push</li> <li>2) Le système affiche le formulaire</li> <li>3) L'utilisateur saisit les données de la notification (Titre, contenu)</li> <li>4) Le système vérifie la validité des données saisies</li> <li>5) Le système envoi une requête au serveur firebase</li> <li>6) Firebase diffuse la notification aux appareils mobiles</li> </ol>
<b>Exceptions</b>	4.1 – Les données ne sont pas valides

	4.1.1 – Le système affiche un message d’erreur et retourne au 2ème point
<b>Post-condition</b>	<ul style="list-style-type: none"> <li>Notification envoyé aux appareils mobiles</li> <li>Notification enregistrée dans la base de données</li> </ul>

Tableau 20 : Description textuelle du cas d’utilisation « envoyer une notification push »

## 6.2. Diagramme de séquence du cas d’utilisation « envoyer une notification push »

La figure 64 représente le diagramme de séquence du cas d’utilisation « envoyer une notification push » :

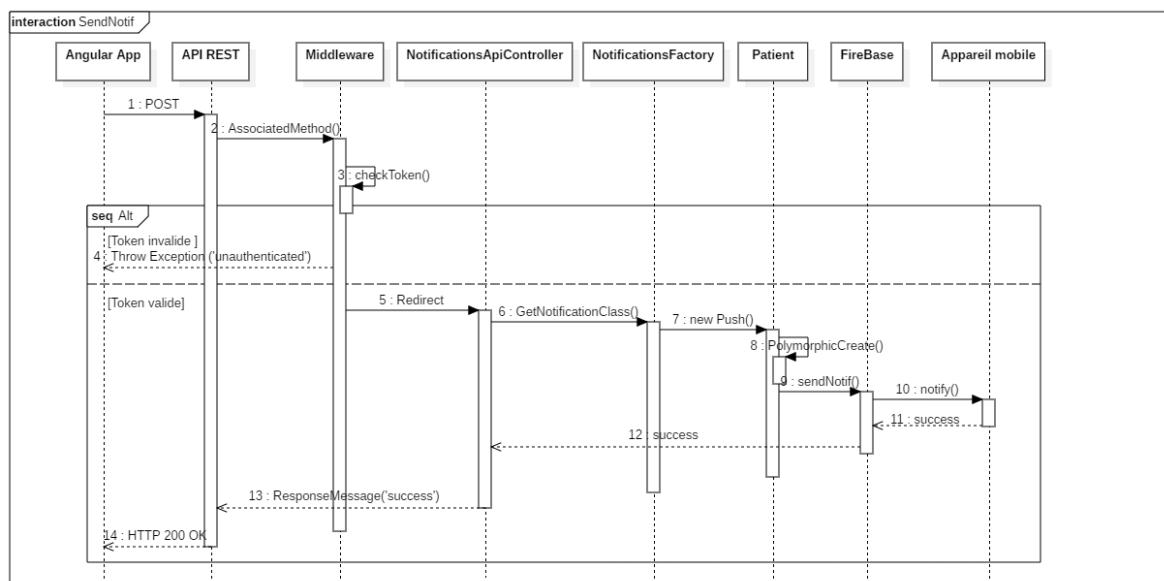
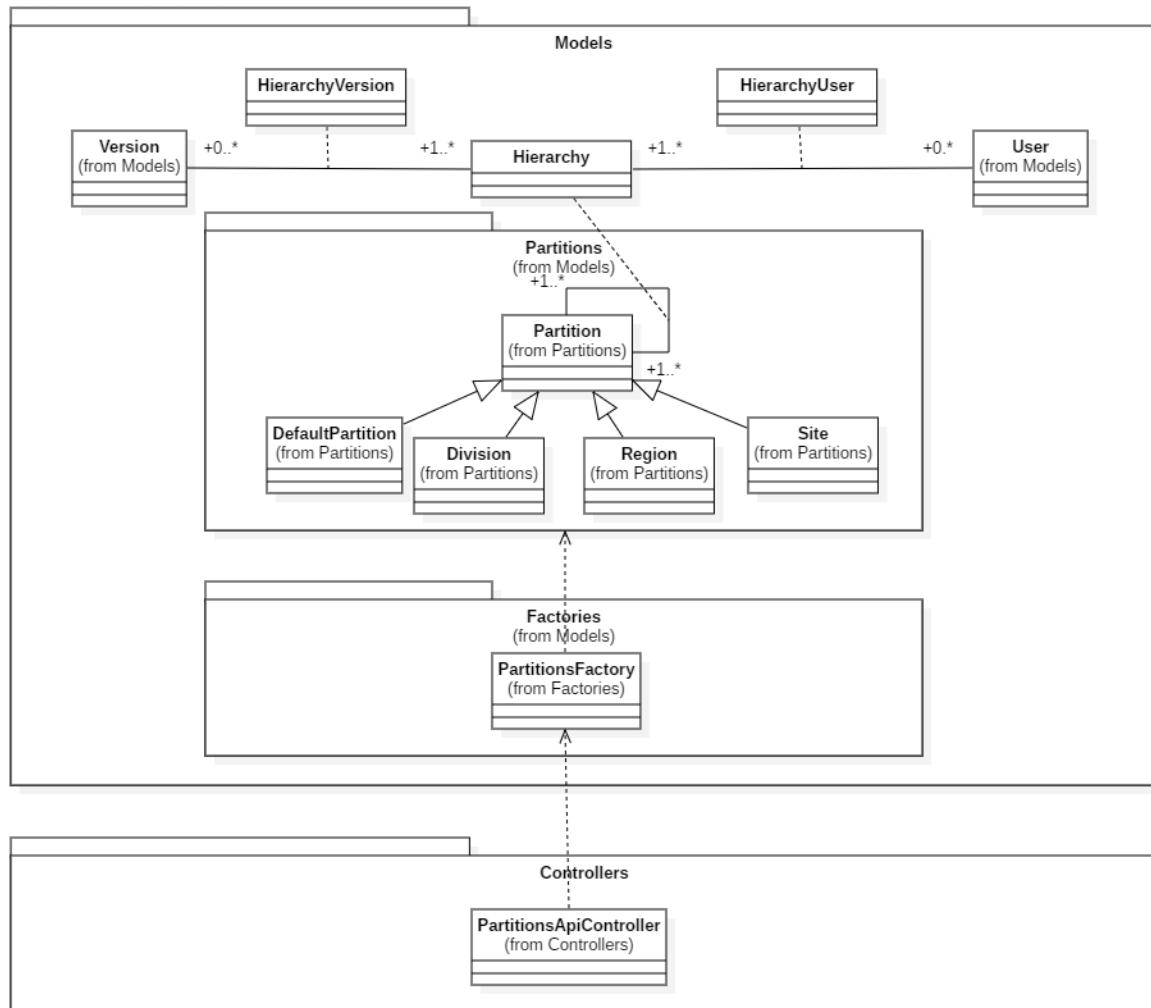


Figure 64 : Diagramme de séquence du cas d’utilisation « envoyer une notification push »

## 7. Diagramme de classe

### 7.1. Diagramme de classe « gestion des partitions »

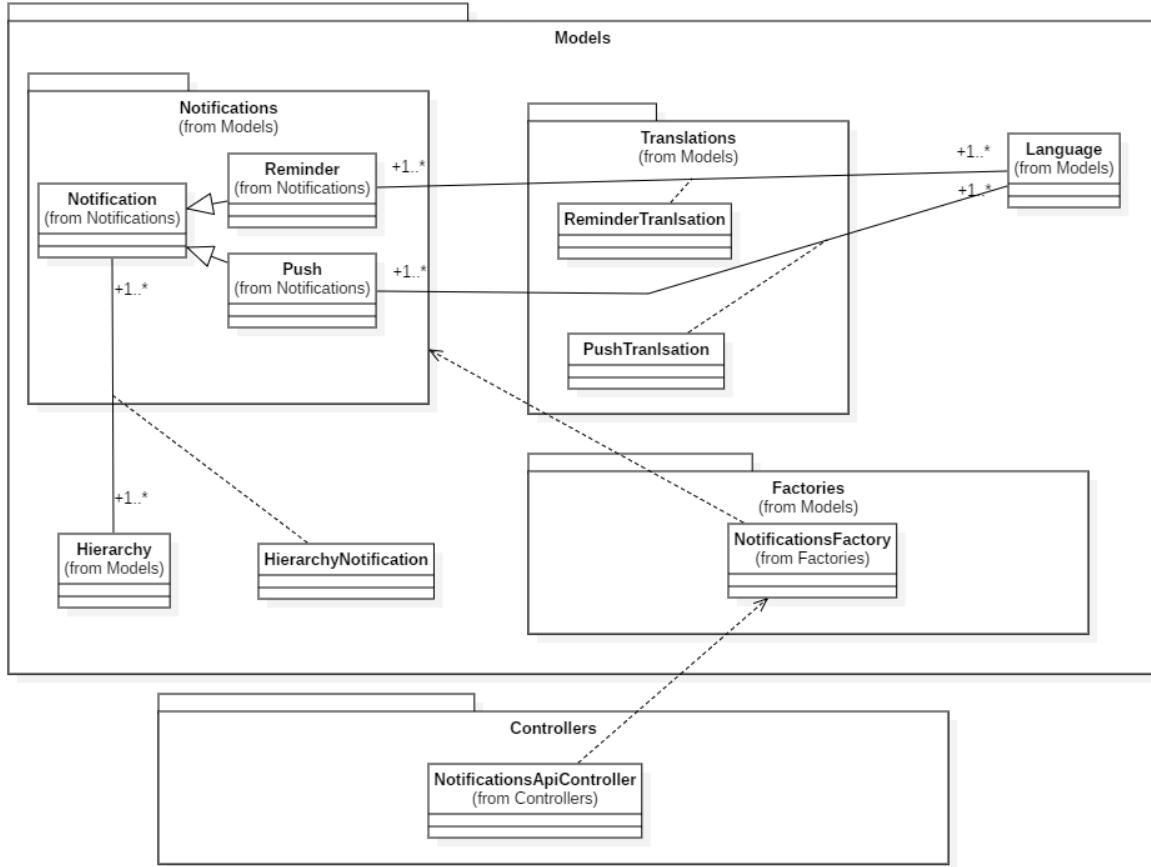
La figure 65 représente le diagramme de classe relatif au module « gestion des partitions » :



**Figure 65 : Diagramme de classe relatif au module « gestion des partitions »**

## 7.2. Diagramme de classe « gestion des notifications »

La figure 66 représente le diagramme de classe relatif au module « gestion des notifications » :



**Figure 66 : Diagramme de classe relatif au module « gestion des notifications »**

### 7.3. Diagramme de classe « gestion des paramètres »

La figure 67 représente le diagramme de classe relatif au module « gestion des paramètres » :

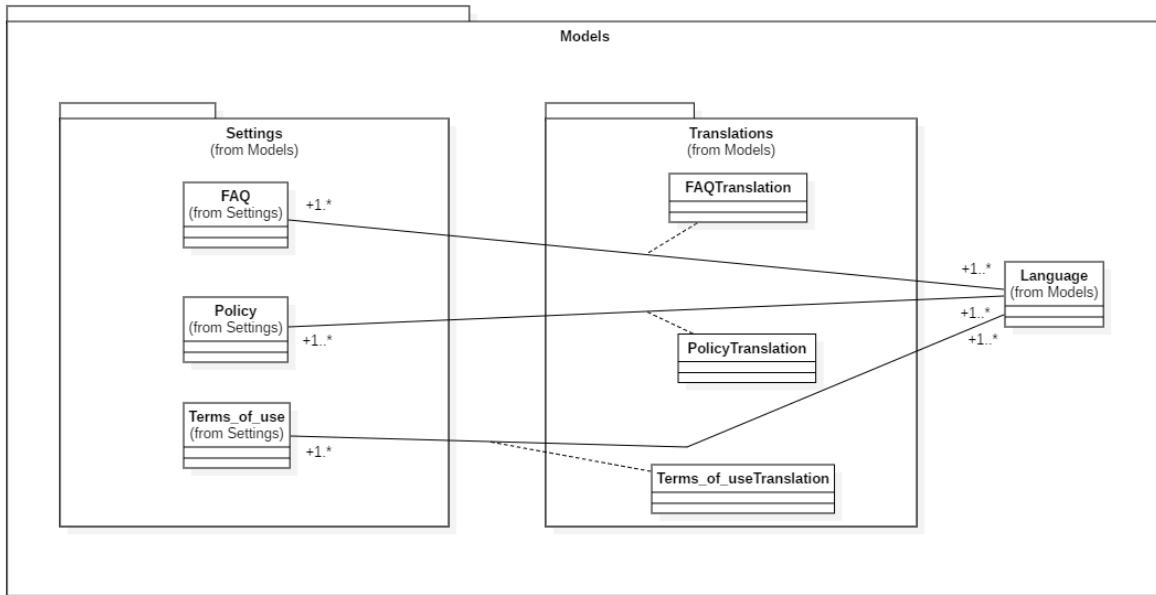


Figure 67 : Diagramme de classe relatif au module « gestion des paramètres »

## 8. Réalisation

Après l'étude conceptuelle de ce dernier sprint, à travers les descriptions textuelles et les diagrammes de séquence, nous passerons à la partie de l'implémentation

### 8.1. Maquettes

Les figures ci-dessous montrent les maquettes que nous avons établies afin de les valider avec le product owner.

A Web Page

https://

**Mu Readu**

**Regions**

Region information

Name \*

Code \*

Division \*  List of divisions

Status \*  Active / Inactive

**Figure 68 : Maquette d'ajout d'une division**

A Web Page

https://

**Mu Readu**

**Divisions**

Division information

Name \*

Code \*

Status \*  Active / Inactive

**Figure 69 : Maquette d'ajout d'une region**

A Web Page

https://

Mu Readu

Sites

Site information

Site Name \* Site Code \*

Division \* Region (not required)

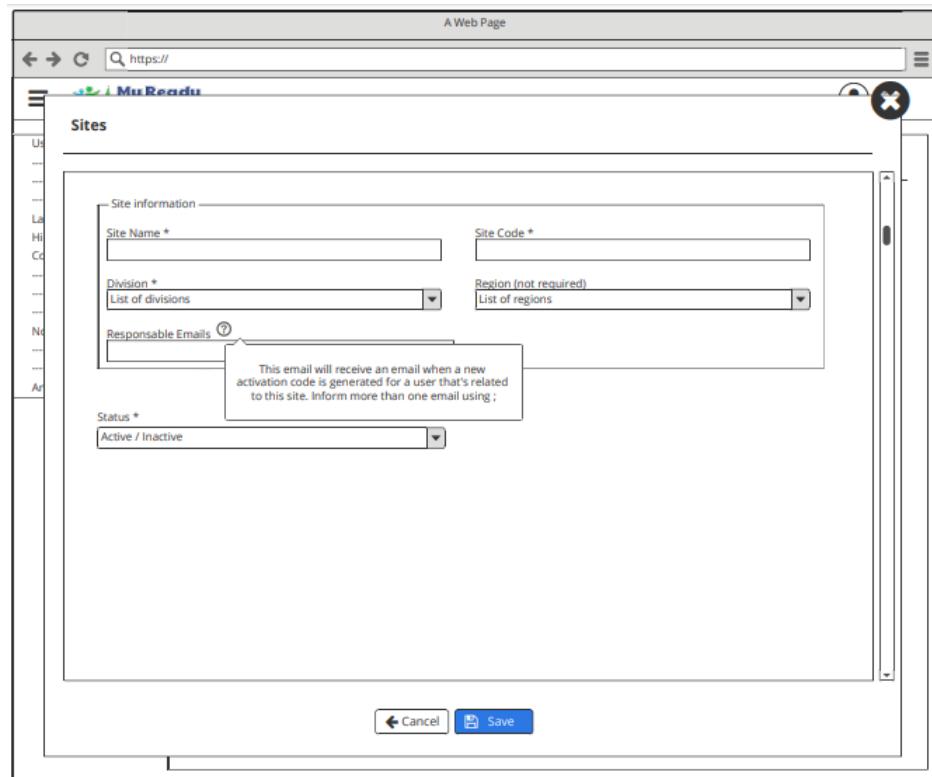
List of divisions List of regions

Responsable Emails ⓘ

This email will receive an email when a new activation code is generated for a user that's related to this site. Inform more than one email using ;

Status + Active / Inactive

Cancel Save



**Figure 70 : Maquette d'ajout d'un site**

A Web Page

<https://>

User Name

**Notifications - Push**

Send a new message

Division \* Region (optional) Site (optional)

List of divisions List of regions List of sites

English\* French

Message

Previous Messages

Show 10 entries Search:

ID	Division	Region	Site	Message
1	My Ready Transition	Quebec	McGill	Hi there's a new version available...
1	My Ready Transition	Quebec	McGill	Hi there's a new version available...
1	My Ready Transition	Quebec	McGill	Hi there's a new version available...
1	My Ready Transition	Quebec	McGill	Hi there's a new version available...
1	My Ready Transition	Quebec	McGill	Hi there's a new version available...

Showign <> of <> entries Previous 1 2 3 Next

**Figure 71 : Maquette d'envoi d'une notification push**

A Web Page

<https://>

User Name

**Notifications - Automatic**

First reminder

English\* French

First reminder message

This reminder should be sent in the next day after the user complete one session, at 9 am. Sent only once.

Second reminder

English\* French

Second reminder messa

Send the second reminder in:  days. at:

Third reminder

English\* French

Third reminder message

Send the third reminder in:  days. at:

**Figure 72 : Maquette d'ajout d'une notification automatique**

## 8.2. Interfaces graphiques

The screenshot shows the 'CHILDRIGHT' application interface. On the left, a red sidebar contains navigation links: Users, Languages (highlighted in red), Hierarchy of partitions, Content Management, Notifications, FAQ, and Analytics. The main content area is titled 'Hierarchy of partitions' and displays a table of divisions:

Name	Code	Action
RCT001CB	division1	<button>VIEW</button> <button>EDIT</button> <button>REMOVE</button>
Division 2	division2	<button>VIEW</button> <button>EDIT</button> <button>REMOVE</button>

A blue button '+ ADD DIVISION' is located in the top right corner of the main content area.

Figure 73 : Interface graphique de la liste des partitions

The screenshot shows the 'CHILDRIGHT' application interface. On the left, a red sidebar contains navigation links: Users, Languages, Hierarchy of partitions, Content Management, Notifications, FAQ, and Analytics. The main content area is titled 'Division informations' and contains the following form fields:

- Name \*: RCT001CB
- Code \*: (Field is empty, highlighted in red with error message: 'Code is required.')
- Status \*: Active

At the bottom of the form are two buttons: '< CANCEL' and a purple 'SAVE' button. A small note at the bottom right says 'Activer Windows' and 'Accédez aux paramètres pour activer Windows.'

Figure 74 : Interface graphique d'ajout d'une division

The screenshot shows the ChildBright application interface. On the left is a red sidebar with the following navigation items:

- Users
- Languages
- Hierarchy of partitions
- Content Management
- Notifications
- FAQ
- Analytics

The main content area has a header with the user name "Ahmed Cheikh". Below the header is a form titled "Regions informations". The form contains the following fields:

Name *	Code *
RCT001CB	
Status *	

At the bottom right of the form are two buttons: "< CANCEL" and "SAVE".

**Figure 75 : Interface graphique d'ajout d'une région**

The screenshot shows the ChildBright application interface. On the left is a red sidebar with the following navigation items:

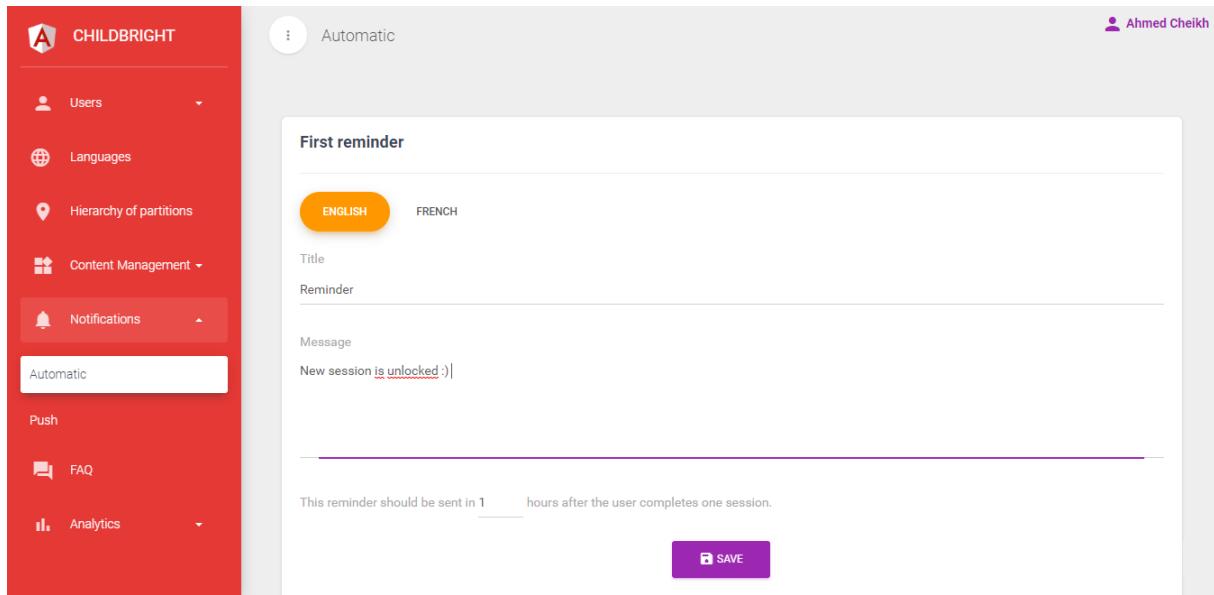
- Users
- Languages
- Hierarchy of partitions
- Content Management
- Notifications
- FAQ
- Analytics

The main content area has a header with the user name "Ahmed Cheikh". Below the header is a form titled "Site informations". The form contains the following fields:

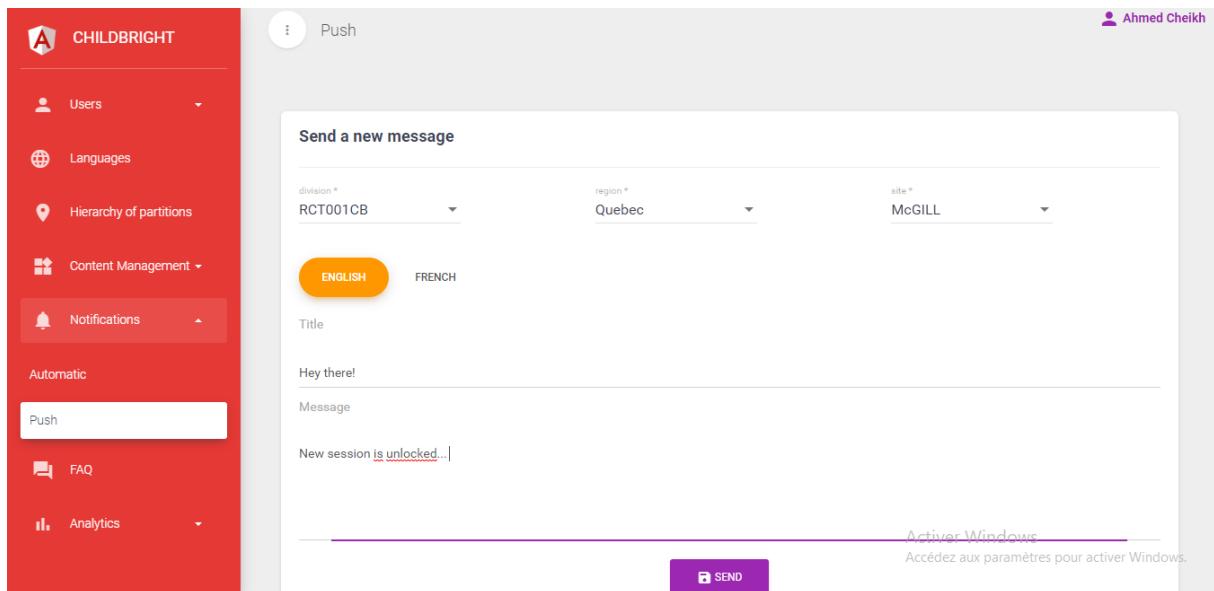
Name *	Code *
RCT001CB	Region
Quebec	
Responsible Emails ⓘ	
Status *	

At the bottom right of the form are two buttons: "< CANCEL" and "SAVE".

**Figure 76 : Interface graphique d'ajout d'un site**



**Figure 77 : Interface graphique de la configuration d'une notification automatique**



**Figure 78 : Interface graphique d'envoi d'une notification push**

The screenshot shows the ChildBright application interface. On the left, there is a red sidebar with the following navigation options:

- Users
- Languages
- Hierarchy of partitions
- Content Management
- Notifications
- Automatic
- Push
- FAQ** (highlighted)
- Analytics

The main content area is titled "FAQ" and contains a list of frequently asked questions:

- I have forgotten my MyREADY Transition login. How can I retrieve it?
- How do I get technical support?
- I have a question regarding the content of the MyREADY Transition App. How do I get a response?
- I have a question regarding the Clinical Research Study. How do I address my request for information?
- I have started a new section. How do I go back to review a previous one?

Below the questions is a toolbar with various icons for text styling (B, I, U, G, " " , H1, H2, etc.) and alignment (Normal, A, etc.).

A note at the bottom of the page reads: "The application is meant to be completed sequentially and one time. If you want to learn more about a topic or go back to review what you learned you can go to the Bulletin Board where you will find the resources available."

At the bottom right, there are buttons for "Activer Windows" (Activate Windows), "EDIT FAQ", "DELETE", and "SAVE".

**Figure 79 : Interface graphique de la page de gestion des FAQ**

## Conclusion

Dans ce chapitre, nous nous sommes intéressés à la réalisation du module de la gestion des partitions, notification et FAQ. Pour ce faire, nous avons commencé par l'élaboration du sprint backlog, puis une étude conceptuelle et finalement la réalisation.

# Conclusion générale

Les soins de santé est un environnement en constante évolution. Les nouvelles technologies et les nouveaux moyens viennent renforcer les capacités des médecins et des chercheurs, permettant aux patients de bénéficier de nouveaux traitements, moniteurs et médicaments.

Ce projet de fin d'études s'installe dans le cadre de cette évolution, permettant ainsi d'améliorer la vie des enfants atteints par des troubles liés aux cerveaux et faciliter leur transition de la vie d'adolescent à la vie d'adulte.

Il convient de rappeler que ce projet se déroule autour de l'application web CMS, qui assure non seulement la manipulation du contenu du jeu *MyReady Transition*, mais aussi de gérer les différents sites de recherche, les utilisateurs (qu'ils soient patients, administrateurs ou chercheurs) et d'envoyer des notifications (instantanées et pré-configurables).

Pour ce faire, nous avons procédé, en premier lieu, à la spécification des besoins fonctionnels et non fonctionnels de notre projet ainsi que la mise en évidence des différentes fonctionnalités à travers les histoires utilisateurs. Ce qui nous a conduits à la phase suivante qui consiste à concevoir et implémenter les différents sprints. Malgré les contraintes de temps et la complexité technique du projet, nous avons réussi au final à mettre en place une solution évolutive, flexible, supportant tout changement de structure de données dans le temps.

Le cadre du ce projet nous a permis de développer nos connaissances requises pendant notre cursus universitaire et surtout d'appliquer les bonnes pratiques d'un ingénieur en informatique. Il nous a permis également de mieux s'approcher du milieu professionnel et de savoir comment gérer et optimiser le temps dans le but d'en profiter au maximum.

Finalement, notre travail ne s'arrête pas là. En effet, il reste ouvert à toute amélioration ou ajout de nouvelles fonctionnalités. Notre solution étant flexible, l'apparition de nouveaux besoins dans le futur ne sera pas d'une grande complexité (comme par exemple l'ajout d'un nouveau segment, peu importe son emplacement dans la hiérarchie des segments).

# Webographie

- [1] : <https://bubbleplan.net/blog/agile-scrum-gestion-projet/> (Consulté le 08-01-2018)
- [2] : <https://www.anthedesign.fr/developpement-web/php-avec-laravel/> (Consulté le 20-01-2018)
- [3] : <https://getcodingknowledge.com/introduction-a-angular-5/> (Consulté le 20-01-2018)
- [4] : <https://confluence.atlassian.com/confeval/development-tools-evaluator-resources/bitbucket/bitbucket-what-is-bitbucket> (Consulté le 02-02-2018)
- [5] : <https://get.slack.help/hc/en-us/articles/115004071768-What-is-Slack> (Consulté le 09-02-2018)
- [6] : <https://firebase.google.com/docs/cloud-messaging/> (Consulté le 02-09-2018)
- [7] : <https://developer.mozilla.org/fr/docs/WebSockets> (Consulté le 15-06-2018)



## ECOLE SUPÉRIEURE PRIVÉE D'INGÉNIERIE ET DE TECHNOLOGIES

**www.esprit.tn - E-mail : contact@esprit.tn**

Siège Social : 18 rue de l'Usine - Charguia II - 2035 - Tél. : +216 71 941 541 - Fax. : +216 71 941 889

Annexe : Z.I. Chotrana II - B.P. 160 - 2083 - Pôle Technologique - El Ghazala - Tél. : +216 70 685 685 - Fax. : +216 70 685 454