

Référence	Dép.	TI
	AN	2022
	N°	.13

Rapport de PROJET DE FIN D'ETUDES

En vue de l'obtention de :

Licence Appliquée en en Développement des systèmes d'information

Elaboré par :

Ayoub Ben Yaala
&
Eya Trabelsi

Encadré par :

Mr Bilel Zemzem

Mme Manoubia Aouadi

Effectué à :

Entreprise : WingedHorse Solutions

- **Adresse :**
- **Tel :**
- **Mail :**

Année universitaire : 2021/2022



Dédicaces

A ma Mère Besma,

« Tu m'as donné la vie, la tendresse et le courage pour réussir.

Tout ce que je peux t'offrir ne pourra exprimer l'amour et la reconnaissance que je te porte.

En témoignage, je t'offre ce modeste travail pour te remercier pour tes sacrifices et

l'affection dont tu m'as toujours entourée. »

A mon Père Nouridine,

*« L'épaule solide, l'œil attentif compréhensif et la personne la plus digne de mon
estime et de mon respect. Aucune dédicace ne saurait exprimer mes sentiments, que*

Dieu te préserve et te procure santé et longue vie. »

A ma sœur Haifa et A mon frère Omar,

A mes amis...

Trabelsi Eya





Dédicaces

Au Dieu tout puissant mon créateur.

À mon père Chokri Ben yaala,

Tu as toujours été à mes côtés pour me soutenir et m'encourager.

Que ce travail traduit ma gratitude et mon affection

À ma mère Monia,

Quoi que je fasse ou que je dise, je ne serai point te remercier comme il se doit. Ton affection me couvre, ta bienveillance me guide et ta présence à mes côtés a toujours été ma source de force pour affronter les différents obstacles.

À mes chers frères Chaker et Wael et ma chère sœur Lojaine pour leur soutien.

À tous mes proches, pour leur bienveillance et amour.

À tous ceux qui m'ont soutenu et cru en moi,

*Je leur dédie ce modeste travail en leur souhaitant un immense bonheur
et une joyeuse vie.*

Ben Yaala Ayoub





Remerciements

Au terme de ce travail je tiens à remercier tous ceux qui ont participé de près ou de loin à la réalisation de ce travail.

À cette occasion je tiens à remercier mon encadrant au sein de WingedHorse Solutions Monsieur **TAREK AOUADI** de m'avoir accordé cette chance de réaliser ce projet.

Je tiens également à remercier mon encadrante à l'Institut Supérieur des Etudes Technologiques à Bizerte, Monsieur **BILEL ZEMZEM** pour la qualité de son encadrement et ses précieuses directives tout au long de la période de stage.

D'un autre part j'exprime mes vifs sentiments et remerciements à tous mes enseignants pour la formation qu'ils ont eu le soin de m'apporter tout au long de mon cursus universitaire.

Finalement, je tiens à exprimer ma gratitude à toute l'équipe au sein de la société WingedHorse Solutions et en particulier Madame **MANOUBIA AOUADI** pour son accueil, son esprit d'équipe et sa collaboration pour me permettre de mener à bien ma mission.

À tous les **membres du jury**, Merci de bien vouloir juger mon travail.



Table des matières

<i>Introduction Générale</i>	<i>9</i>
<i>Chapitre 1 : Présentation du Cadre général du Projet</i>	<i>11</i>
I. Cadre de projet	12
II. Présentation de la société	12
III. Planification du Projet	13
IV. Problématique.....	14
V. Etude de l'Existant et Critique	14
1. Périmètre du projet	14
2. Etude de l'existant.....	15
3. Critique de l'existant.....	17
4. Solution proposée	17
<i>Chapitre 2 : Planification du projet</i>	<i>19</i>
I. Méthodologie de conduite du projet	20
1. Comparaison des méthodes Classiques et Agiles.....	20
2. Choix de la méthode Agile Scrum.....	22
II. Méthode de Modélisation et conception	24
III. Architecture technique du projet.....	24
1. Architecture MVC	24
IV. Outils de gestion de projet.....	25
V. Etablissement du Planning du Projet	26
<i>Chapitre 3 : Planification Backlog produit et environnement technologique.....</i>	<i>27</i>
I. Planification du Backlog Produit	28
1. Besoins fonctionnels.....	28
2. Besoins non fonctionnels.....	28
3. Backlog Produit	29
II. Environnement Technologique.....	34
1. Environnement matériel.....	34
2. Environnement logiciel	34
3. Environnement de Test.....	38
<i>Chapitre 4 : Implémentation de la Release 1.....</i>	<i>41</i>
I. Planification et implémentation du sprint 0	42

1.	But de Sprint	42
2.	Sprint Backlog.....	42
3.	Réalisation et Test	43
II.	Planification et implémentation du sprint 1	44
1.	But de Sprint	44
2.	Sprint Backlog.....	44
3.	Expression des besoins	44
4.	Réalisation.....	46
Chapitre 5 : <u>Implémentation de la Release 2</u>.....		49
I.	Planification et implémentation du sprint 2	50
1.	But de Sprint	50
2.	Sprint Backlog.....	50
3.	Expression des besoins	50
4.	Réalisation.....	52
II.	Planification et implémentation du sprint 3	54
1.	But de Sprint	54
Chapitre 6 : <u>Implémentation de la Release 3</u>.....		56

Table des figures

Figure 1: Logo de l'entreprise	12
Figure 2: Organigramme de l'entreprise	13
Figure 3: Business Processus Management	16
Figure 4: Méthode classique et agiles	20
Figure 5: Modèle en V	21
Figure 6: Modèle SCRUM	23
Figure 7: Scrum par rapport aux méthodes Agiles.....	23
Figure 8: Architecture MVC	24
Figure 9: Le Logiciel Redmine	25
Figure 10: Planning du projet	26
Figure 11: Caractéristiques du PC	34
Figure 12: Diagramme de cas d'utilisation Sprint 1	45
Figure 13: Simulation d'Importation des fichiers.....	46
Figure 14: Simulation pour la création des Fields	46
Figure 15: Simulation de mapping des vols.....	47
Figure 16: Simulation de mapping des vols.....	47
Figure 17: Simulation de dispatching des vols	48
Figure 18 : Diagramme de cas d'utilisation Sprint 2	51
Figure 19 : Diagramme des classe Sprint 2	51
Figure 20 : Base de données MongoDB	52
Figure 21: Interface Login.....	52
Figure 22 : Interface Gestions des utilisateurs	53
Figure 23: Interface Gestion des Vols	53
Figure 24: interface Gestion des avions.....	54

Table des tableaux

Tableau 1: Les phases du stage	26
<i>Tableau 2: Backlog Produit.....</i>	30
Tableau 3: Backlog Sprint 0	42
Tableau 4: Backlog de sprint 1	44
Tableau 5: Backlog Sprint 2	50

Introduction Générale

L'agent aéroportuaire accompagne les passagers tout au long de la circulation dans un aéroport.

Posté au guichet de sa compagnie aérienne, il est de l'accueil des passagers, encadre les embarquements et apporte des renseignements.

Aussi fait la liaison entre la compagnie aérienne, le handling et les passagers.

Ses missions sont donc variées, selon les impératifs du moment (Enregistrement des bagages, Encadrement de l'embarquement, Vérification des protocoles et consignes de sécurité...)

Le « Handling » est une activité industrielle et commerciale exercée sur les aéroports avec un impact direct sur la qualité des prestations offertes aux passagers et aux compagnies. Le handling est un service d'assistance en escale, les prestations vont du personnel chargé de l'enregistrement des passagers au traitement des bagages.

Une compagnie aérienne est une entreprise de transport aérien qui transporte des passagers ou du fret. Les compagnies aériennes louent ou achètent leur avions pour offrir leurs services et peuvent former des partenariats ou des alliances pour des bénéfices mutuels.

Dans ce cadre, notre projet de fin d'étude « **SPOTFLY DEMO** » est axé sur une application web va optimiser le processus de travail de l'entreprise de handling basée sur la démarche BPM.

Le premier chapitre intitulé Cadre du projet est consacré à la présentation de l'environnement du stage au sein de la société « **WingedHorse Solutions** » ainsi que la présentation du projet et le cahier des charges.

Le chapitre suivant est intitulé Etude préalable. Dans ce chapitre, on va présenter la méthodologie de conduite du projet ainsi les environnements du travail aussi on va décrire l'analyse et les critiques de l'existante ainsi que la solution proposée.

Chapitre 1 : Présentation du Cadre général du Projet

Introduction

Le présent chapitre est dédié à la présentation du projet par la fixation des objectifs à atteindre. Dans la première section, nous commençons par présenter en premier lieu le cadre général de notre projet et en second lieu l'entreprise accueillante à savoir WingedHorse Solutions. Dans la deuxième section, nous présentons l'étude de l'existant du projet ainsi que sa critique et nous proposons les solutions. Enfin, nous effectuons l'étude des notions technologiques des solutions et des principes utilisées dans notre projet.

I. Cadre de projet

Les services d'assistance en escale sont essentiels au bon fonctionnement de l'aéroport. Ils désignent un ensemble de services fournis par des entreprises de services d'assistance en escale aux compagnies aériennes. Avec l'évolution du digital, ces entreprises ont besoin de nombreuses solutions pour gagner leurs temps et maximiser leurs potentiels afin de gagner la satisfaction et la confiance des compagnies aériennes et les clients. C'est dans ce contexte que s'inscrit notre projet dont l'objectif est de mettre en place une application « workFly » web et mobile en collaboration avec une équipe informatique.

II. Présentation de la société



Figure 1: Logo de l'entreprise

« WingedHorse Solutions » est filiale d'une société française, située à Tunis. Elle est une startup innovante qui accompagne ses clients dans le développement de leurs solutions informatiques et qui se lance dans le développement de ses propres applications propriétaires autour des nouvelles technologies et la philosophie Scrum. Dans le cadre de la conception d'applications Web et mobiles, l'équipe de WingedHorse Solutions est jeune, dynamique et

formée de trois groupes : développeurs Backend, développeurs Frontend et Mobile, plus un expert qualité du logiciel. Tous les membres fonctionnent dans un environnement Scrum où règne la notion de 4 C (Collaboration, Communication, Coordination et la Confiance). Les groupes mènent une veille active sur les technologies de pointe, l'actualité scientifique, la recherche de solutions innovantes pour le traitement et la gestion des processus métiers. WingedHorse Solutions offre à ses clients d'une part une plateforme technologique de haut niveau constituée d'un environnement de nouvelles technologies qui s'améliorent avec l'amélioration technologique mondiale et d'autre part une possibilité de mettre en place des projets ambitieux et innovateurs.

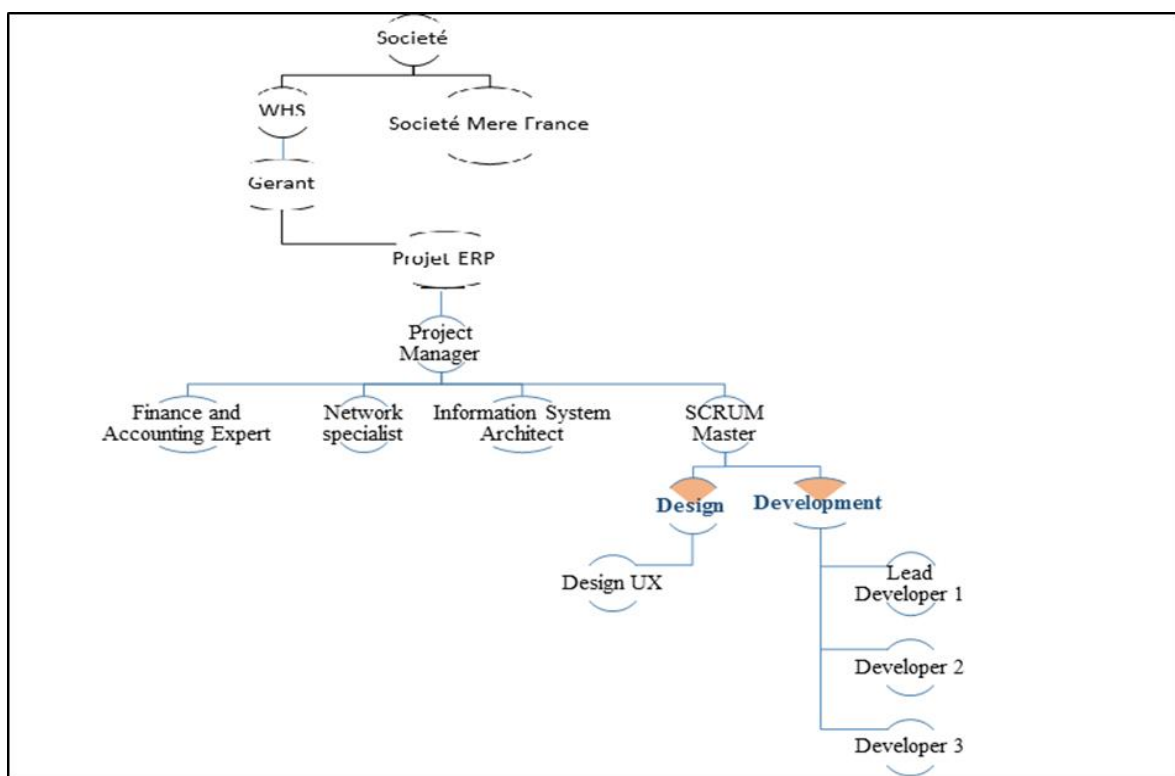


Figure 2: Organigramme de l'entreprise

III. Planification du Projet

Aujourd'hui, mettre en place un système ne constitue pas un projet complet, il faut que ce projet ait une qualité logicielle élevée. Pour atteindre ce niveau de qualité, le processus de l'élaboration du projet doit se baser sur le choix d'une méthodologie ou cycle de vie de conduite de projet pour mieux gérer l'équipe, d'un modèle de découpage du projet pour mieux définir les fonctionnalités à regrouper en sous-systèmes selon leur priorité et d'une méthode de gestion de projet que le responsable doit suivre quotidiennement en comparant le planning prévisionnel et le rendement réel de l'équipe informatique. Pour déterminer cet environnement de travail

avant de démarrer le projet, une réunion avec le comité de pilotage a été tenue. Sont présentes toutes les personnes intéressées par ce projet.

Lors de cette réunion, tous les membres informaticiens et gestionnaires ont discuté le projet pour en premier lieu choisir une méthode de conduite de projet et en second lieu comprendre l'objectif du projet et connaître les besoins fonctionnels et non fonctionnels qu'elle doit implémenter en suivant un planning bien précis.

IV. Problématique

Un aéroport gère une multitude de services imposant la coordination de nombreux acteurs sur le terrain. Ceux qui garantissent aux compagnies aériennes les services nécessaires au bon déroulement des opérations aéroportuaires, en particulier dans les domaines clés que sont le décollage, l'atterrissage et le balisage. Ceux qui gèrent la gestion du chargement et déchargement, transfert et tri-bagages, transports passagers et équipages ainsi que les opérations diverses sur avion comme le guidage arrivée/départ, assistance au placement, fueling, assistance au démarrage.

Les plateformes existantes qui gèrent le trafic aérien manquent de communication, de collaboration entre les différents intervenants d'un aéroport : Compagnie aérienne, Handler, Contrôleur aérien, en plus toutes ces solutions sont basées sur une démarche basique et classique.

Notre objectif est d'offrir à nos clients un logiciel simple à déployer, partageant la même source d'informations mais au même temps il doit sécuriser tout en se basant sur le temps réel et la notion des 3 C (Collaboration, Communication et Coordination).

V. Etude de l'Existant et Critique

Avant de se lancer dans un projet, une étude et une analyse de l'existant s'impose. Cette étude a pour fin la critique de l'existant et la proposition d'une solution innovante pour le client.

1. Périmètre du projet

Nous avons intégré une équipe SCUM qui doit mettre en place un système configurable de gestion pour une compagnie aérienne se basant sur la notion « ERM ». Le logiciel ERM est implémenté par une équipe de la société en déployant la notion de « Business Process Management ». Tout au long de notre stage, nous serons responsables de développement de certains modules et fonctionnalités que le Product Owner et le Scrum Master nous les ont assignés au fur et à mesure de la constitution des releases et sprints.

Pour ces raisons, le périmètre de notre projet est en réalité le domaine relatif au trafic aérien en déployant « ERM ».

2. Etude de l'existant

Vu que tout notre travail se base sur le concept de l'ERM (1), nous allons en premier lieu définir cette notion avant d'entamer l'étude de l'existant.

L'ERM est une technique de management destinée à aider les entreprises à gérer tous les risques auxquels elles doivent faire face. Il s'agit d'adopter des stratégies organisationnelles pour limiter les impacts de ces risques afin de créer de la valeur pour l'actionnaire et d'avantage concurrentiel pour l'entreprise. Quand nous parlons des activités généralement nous parlons aussi du « Business Process Management » (2).

La démarche BPM (19) (Business Process Management) ou gestion des processus métier en français est une démarche qui vise à analyser et fluidifier les processus mis en place par l'entreprise pour réaliser ses activités dans une optique d'amélioration de sa performance. La notion de processus. Les processus structurent l'ensemble des activités quotidiennes nécessaires à l'exercice de chaque métier. Ils consistent en une série de tâches et d'actions, parfois complexes, que les collaborateurs doivent accomplir afin d'obtenir un résultat déterminé.

Pour un processus, donné il est possible d'avoir plusieurs procédures : l'enchaînement des activités est semblable, mais les ressources liées à son exécution peuvent varier d'un acteur à l'autre. Sa représentation est faite grâce au workflow. En effet un workflow correspond à la représentation et la gestion informatique de toutes les tâches à accomplir associées aux différents acteurs impliqués dans la réalisation d'un processus (3).

Le BPM (Business Process Management) (18) permet de représenter les activités d'une organisation c'est-à-dire ses processus et il permet aussi d'analyser et de fluidifier les processus mis en place dans une optique d'amélioration de la performance. Il met en évidence les interactions humaines et les échanges de données avec les systèmes d'informations de l'organisme. Il assure que les processus sont mis en œuvre pour répondre aux besoins clients/acteurs avec la meilleure performance au niveau de tous les domaines de l'entreprise.

Le BPM crée de la valeur pour l'entreprise en améliorant la performance, la productivité, l'efficacité du personnel et la qualité du service au client que toute entreprise cherche sa fidélité et sa confiance

L'objectif principal du BPM est d'apporter une visibilité totale sur l'activité et les échanges au sein de l'entreprise. Il consiste à diminuer les erreurs humaines et les problèmes de communication au sein de ces processus, ainsi de permettre aux différents intervenants de se concentrer sur leurs tâches à fortes valeurs ajoutées.

Pour toute entreprise, son but en déployant le BPM est d'analyser et adapter régulièrement ses processus, workflows et procédures en fonction des contraintes identifiées pour développer sa performance ou sa qualité de service. En effet, l'entreprise peut bénéficier des avantages du BPM qui sont multiples et dépendent de la nature des processus modélisés

[4]. Nous citons les avantages suivants :

- **Visibilité** : un processus bien décrit favorise la compréhension de ses enjeux par les utilisateurs afin d'y contribuer plus efficacement.
- **Rentabilité** : le BPM permet de réduire les délais de réalisation, d'accélérer les prises de décision, d'optimiser les coûts et donc de maximiser l'efficacité et les résultats de l'entreprise.
- **Performance** : L'économie actuelle vise à faire toujours plus avec moins de ressources. Les principes du BPM favorisent l'amélioration de la productivité.
- **Traçabilité** : le suivi d'avancement (qui ? quoi ? quand ? comment ? pourquoi ?) est facilité par la mise en place d'indicateurs de suivi et permet de remonter immédiatement toutes problématiques identifiées.
- **Agilité** : l'identification et la correction des dysfonctionnements rencontrés sont des enjeux majeurs du BPM. La capacité à adapter en temps réel les processus modélisés selon les contraintes et les exceptions identifiées génère une agilité nouvelle.

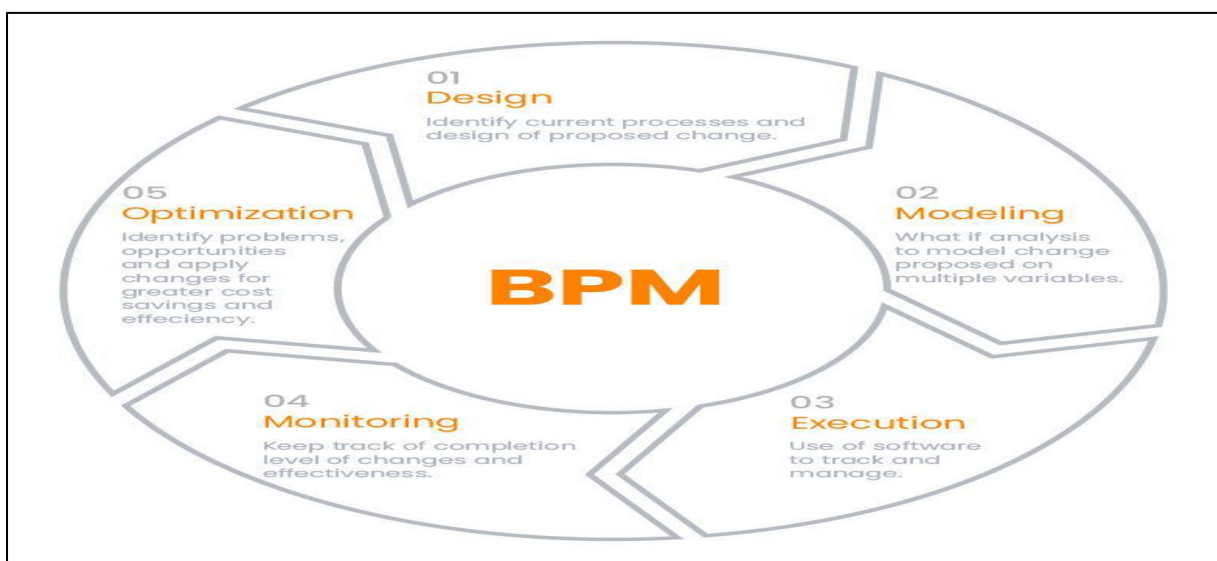


Figure 3: Business Process Management

3. Critique de l'existant

Au cours de la période de l'étude de l'existant et la présentation du système ERM, nous avons bien compris respectivement le processus à mettre en place et le choix de la démarche BPM pour l'implémentation de notre projet.

Suite à des recherches sur internet, nous avons remarqué que c'est vrai, il y'a quelques plateformes comme A-CDM qui gère le trafic aérien et la communication entre les différents intervenants d'un aéroport, mais toutes ces solutions sont basées sur une démarche basique et classique.

4. Solution proposée

Notre application web et mobile va optimiser le processus de travail de l'entreprise de handling basée sur la démarche de BPM. C'est une application qui va permettre le suivi et le traitement des vols entre les différents acteurs de l'aéroport. Pour une mise à jour de la solution, la partie code doit être riche, bien structurée et admet les propriétés de la qualité logicielle [6] suivantes :

- Facile à lire, par soi-même mais aussi par les autres.
- Avoir une organisation logique et évidente.
- Être explicite, montrer clairement les intentions du développeur.
- Être soigné et robuste au temps qui passe.
- Avoir la méthodologie et de la stratégie de développement adaptées par la société.

Cette solution aura deux catégories d'acteurs :

- **L'Administrateur** : c'est un utilisateur authentifié, il possède tous les privilèges de l'administration de l'application. Il a aussi l'accès à certains services de l'application tels que l'affichage des vols, ...
- **Le client** : c'est un utilisateur authentifié, il possède certains privilèges, à savoir l'accès aux différents services de l'application tels que l'affichage des vols, l'affectation des agents et le traitement des services.

Ce client peut être :

- Handler,
- Compagnie aérienne.

Conclusion

Tout au long de ce chapitre, nous avons présenté l'organisme d'accueil et une brève description du projet à réaliser, en déterminant la problématique et en proposant la solution envisagée. Par la suite nous avons présenté la réunion de planification de notre projet.

Le chapitre suivant sera consacré d'une part à la constitution de la liste des besoins fonctionnels et non fonctionnels, et la planification du Backlog de produit par notre Product-Owner et d'autre part à l'implémentation de l'environnement technologique pour la mise en place de notre projet par l'équipe Scrum en collaboration du Scrum Master.

Chapitre 2 : Planification du projet

Introduction

Pour réussir un projet, il faut déterminer ses facteurs de réussite comme le budget, le délai, les collaborateurs, la méthode de conduite de projet et les technologies à déployer au niveau des outils de développement, de l'architecture et de l'environnement matériel et logiciel. Ces facteurs sont identifiés lors de la réunion de la planification du projet ainsi que la présentation de la liste des besoins des utilisateurs.

I. Méthodologie de conduite du projet

Pour garantir la réussite de notre projet, nous sommes contraints de choisir une méthode de conduite de projet. Nous avons le choix entre les méthodes Classiques (Modèle en cascade, Modèle en V, Modèle Spirale,) et les méthodes Agile (ADS, Crystal, Scrum, ...).

1. Comparaison des méthodes Classiques et Agiles

La définition et la comparaison des méthodes Classiques et Agiles peuvent nous servir de référence afin faire le choix d'une méthode pour réaliser la conduite de notre projet.

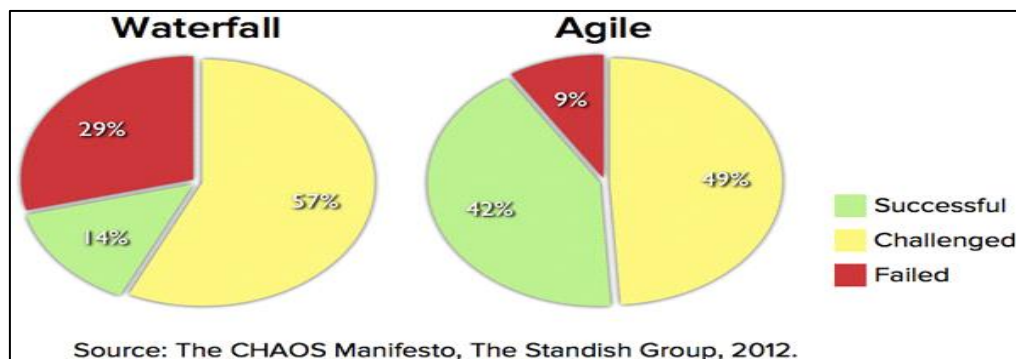


Figure 4: Méthode classique et agiles

- **Méthodes Classiques**

Les méthodes de gestion de projet « classiques » ou traditionnelles se basent sur des approches **structurées**, fondées sur des **processus rigoureux** afin de livrer des projets selon des **exigences strictes et établies d'avance sous la forme d'un cahier des charges**.

Pour cette méthode, nous allons prendre comme exemple le modèle du cycle en V [...], qui est un modèle conceptuel de gestion de projet créé à la suite du problème de réactivité du modèle en cascade. Ce cycle de vie 'V' présente des

validations intermédiaires favorisant un bon suivi du projet par des points de mesure concrets de l'avancement du travail, la décomposition fonctionnelle de l'activité et la limitation des risques d'erreurs d'analyse ou de conception très coûteuses. Pour chaque phase de la branche gauche du modèle V, qui constitue en réalité une démarche en cascade, nous allons préparer la phase de test de la branche droite qui lui correspond.

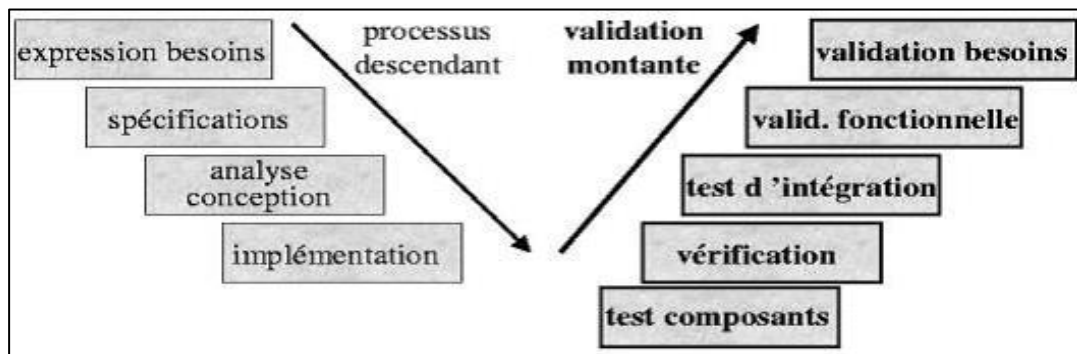


Figure 5: Modèle en V

Ce modèle en V présente des avantages et des inconvénients dont nous citons quelques exemples :

Avantages :

- ❖ Une forte notion de vérification et validation.
- ❖ Permet de discipliner le processus de développement.

Inconvénients :

- ❖ Rigidité des phases qui sont linéaires et manque de souplesse : pour passer à l'étape suivante, il est important de terminer la précédente, ce qui entraîne des dérives du planning et donc de coût.
- ❖ Difficulté d'intégrer et d'anticiper le changement.
- ❖ Projet monolithique : un produit global fourni en bloc.
- ❖ Manque de communication : la communication est peu facilitée du fait du cloisonnement des rôles de chaque acteur : ils ont chacun un rôle à jouer mais ils ne le jouent jamais tous ensemble et communiquent principalement au travers de documents.

- **Méthode Agiles**

Le terme "agile" définit une approche de gestion de projet qui prend le contre-pied des approches traditionnelles prédictives et séquentielles de type cycle en V ou en cascade. La notion même de "gestion de projet" est remise en question au profit de "gestion de produit". De façon à raisonner davantage "produit" que "projet", en effet l'objectif d'un projet consiste bien à donner naissance à un produit.

Les méthodes Agiles sont nées du constat que les modèles classiques ont atteint leurs limites en particulier sur des projets complexes.

Les méthodes Agiles ont été créées et éprouvées par des acteurs de terrains ce qui les rend particulièrement pragmatiques. Les méthodes agiles reposent toutes sur des valeurs fondatrices [7] :

- ❖ Priorité des personnes et des interactions sur les procédures et les outils,
- ❖ Priorité d'applications opérationnelles sur une documentation exhaustive,
- ❖ Priorité de la collaboration avec le client sur la négociation de contrat,
- ❖ Priorité de l'acceptation du changement sur la planification.

Comme exemple de méthode agile, nous prenons la méthode Scrum qui est une méthode qui se base sur ses quatre (04) valeurs fondatrices. Durant les réunions, l'équipe informatique ainsi que le **scrum-master** et **product-Owner** doivent travailler en 4 C (Collaboration, Communication, Coordination et Confiance). Ces quatre termes permettront à l'équipe du projet de travailler en transparence, ce qui permet :

- ❖ D'avoir des indicateurs relatifs à l'état d'avancement visibles et bien compris par toute personne impliquée au projet,
- ❖ D'avoir à l'esprit une inspection continue pour détecter les variations par rapport à l'attente prévue,
- ❖ D'avoir une adaptation qui permet à tout processus d'être ajusté rapidement afin de minimiser les futures déviations et d'éliminer la probabilité que le produit résultant sera inacceptable

2. Choix de la méthode Agile Scrum

Pour ses avantages, nous avons choisi la méthode Agile Scrum qui offre une souplesse de travail à toute l'équipe impliquée au projet par rapport aux méthodes classiques :

- ❖ Répond mieux au besoin de faire face au problématique de la gestion de projet c'est à dire que la capacité d'agir et la connaissance du projet se déroule de façon parallèle contrairement à la méthode classique ou notre capacité d'agir tend à diminuer à travers le temps de déroulement du projet.
- ❖ S'adapte au changement et les exploites pour donner un avantage compétitif au client et assurer une meilleure qualité du produit,
- ❖ Offre une meilleure qualité (délai, cout et objectifs) en un temps minimum,
- ❖ Une meilleure communication grâce au mêler quotidienne et à la mise en valeurs du travail en équipe,
- ❖ Réalise un bilan à la fin de chaque Sprint pour capitaliser le sprint suivant.

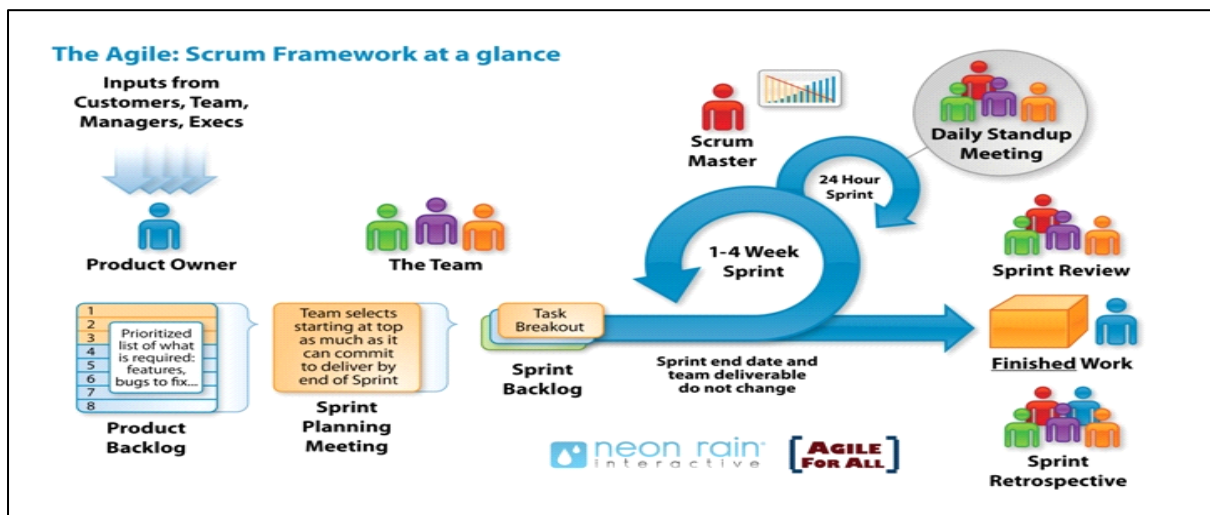


Figure 6: Modèle SCRUM

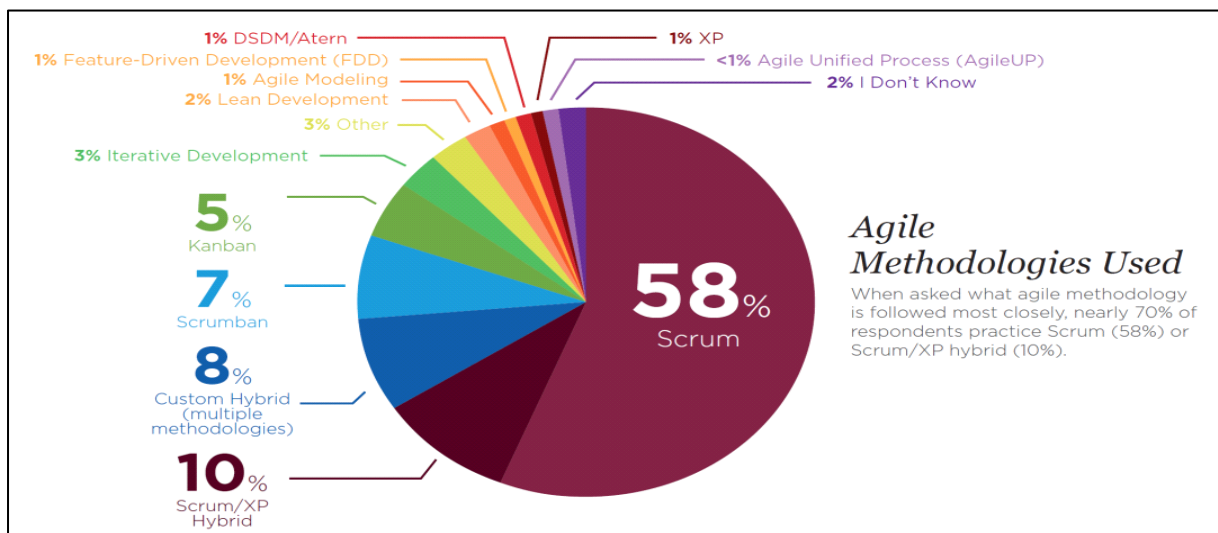


Figure 7: Scrum par rapport aux méthodes Agiles

II. Méthode de Modélisation et conception

Modéliser, c'est décrire de manière visuelle et graphique les besoins et les solutions fonctionnelles et techniques du projet. Dans le cadre de notre projet, nous allons réaliser la modélisation avec le langage UML. C'est l'acronyme anglais pour « Unified Modeling Language ». Nous le traduisons par « Langage de modélisation unifié ».

La notation UML est un langage visuel constitué d'un ensemble de schémas appelés des diagrammes, qui donnent chacun une vision différente pour représenter le logiciel à développer : son fonctionnement, sa mise en route, les actions susceptibles d'être effectuées.

Pour notre projet, nous avons besoin d'un langage de modélisation unifiée c'est pourquoi nous avons choisi UML. Notre choix s'est basé sur les points forts de ce langage notamment sa standardisation et les divers diagrammes qu'il propose. Aussi UML présente le meilleur outil pour schématiser des systèmes complexes sous un format graphique et textuel simplifié et normalisé. [12]

III. Architecture technique du projet

En informatique, l'architecture désigne la structure générale inhérente à un système informatique, l'organisation des différents éléments du système : logiciels et/ou matériels et/ou humains et/ou informations et des relations entre les éléments. Cette structure fait suite à un ensemble de décisions stratégiques prises par le comité de pilotage du projet avec l'équipe informatique. [13]

1. Architecture MVC

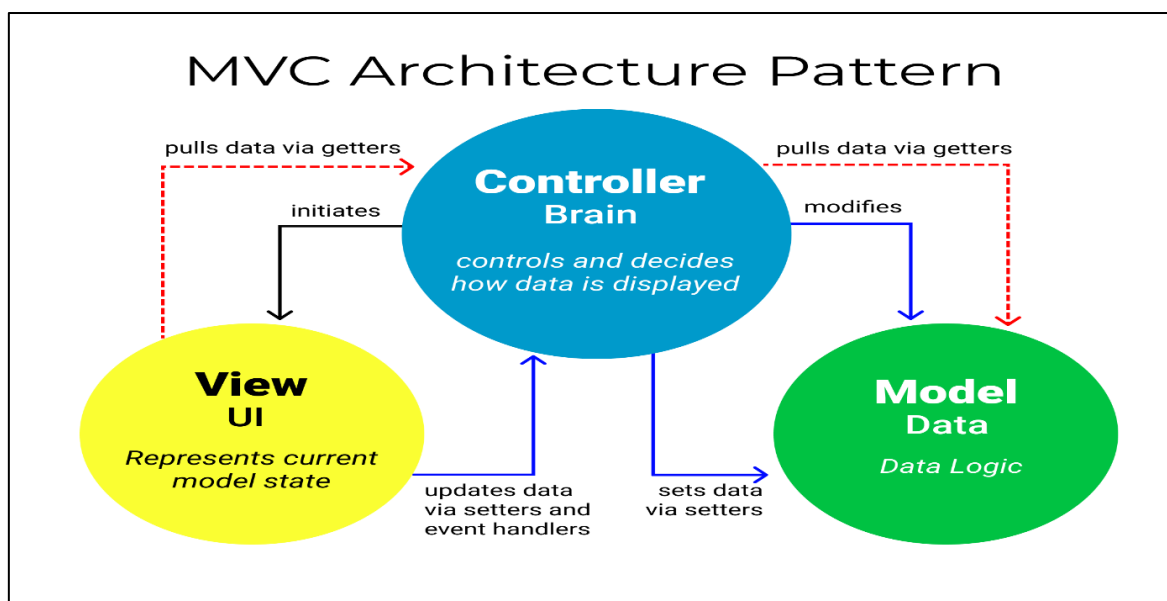


Figure 8: Architecture MVC

La figure ci-dessus représente l'architecture de notre application qui fait apparaître l'infrastructure « MVC » (Modèle-Vue-Contrôleur) composée des éléments suivants :

- **Le modèle** représente le comportement de l'application : traitements des données, interactions avec la base de données, services, etc. le modèle comporte toute les classes et les méthodes qui décrivent les données manipulées par l'application.
- **La vue** correspond à l'interface avec laquelle l'utilisateur interagit. Les résultats renvoyés sont dénués de toute présentation mais sont soutenus par les vues. La vue n'effectue aucun traitement, elle se contente d'afficher les résultats des traitements effectués par le modèle, et de permettre à l'utilisateur d'interagir avec elles.
- **Le contrôleur** prend en charge la gestion des événements de synchronisation pour mettre à jour la vue ou le modèle. Il n'effectue aucun traitement, ne modifie aucune donnée, il analyse la requête du client et se contente d'appeler le modèle adéquat et de renvoyer la vue correspondante à la demande tout en des requête http.

IV. Outils de gestion de projet

Nous avons utilisé « Redmine » pour schématiser les rôles de Scrum et les Sprints. IL s'agit d'un outil en ligne qui couvre tous les scénarios d'utilisation courants dans Scrum. Il présente plusieurs caractéristiques telles que la gestion des équipes et des rôles, la gestion du Backlog produit, la planification des Sprints, etc. [14].



Figure 9: Le Logiciel Redmine

V. Etablissement du Planning du Projet

Le manque de méthode de conduite de projet, de budget, de ressources humaines compétentes ou d'un délai suffisant, tout projet peut être voué à l'échec. Après avoir évalué le degré de réussite possible et la faisabilité de notre projet et déterminer tous les critères de réussite du projet, nous devons établir un planning qui sera adapté tout le long de notre stage.

Tableau 1: Les phases du stage

	Début	Jours	Date de Fin
Intégration à l'entreprise	14 février 2022	10	23 février 2022
Installation de la plateforme technologique	24 février 2022	10	5 mars 2022
Apprentissage des technologies déployées par l'entreprise	6 mars 2022	35	9 avril 2022
Implémentation des fonctionnalités + Validation & Test	10 avril 2022	55	3 juin 2022
Elaboration du Rapport du PFE	14 février 2022	110	3 juin 2022

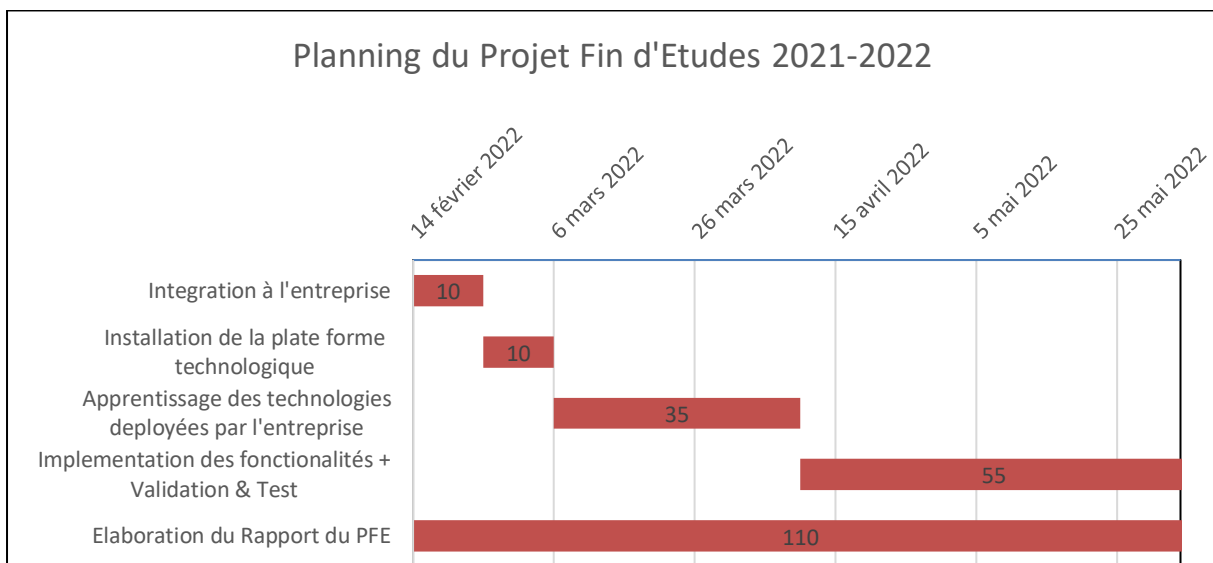


Figure 10: Planning du projet

Conclusion

Ce chapitre a été consacré au choix de la méthodologie de conduite de projet, à la méthode de modélisation et à l'établissement du planning de la mise en place de notre projet. Le chapitre suivant, nous allons établir notre backlog produit en collaboration avec le PO et le Scrum Master.

Chapitre 3 : Planification Backlog produit et environnement technologique

Introduction

Ce chapitre présente le premier pas de la réalisation de notre projet. En premier lieu, nous identifions les acteurs de notre application, nous listons les besoins fonctionnels et non fonctionnels de notre système et le backlog produit et en second lieu, nous donnons un bref aperçu sur l'environnement technologique au niveau du matériel, des technologies et des langages de programmation utilisés pour la mise en place de notre projet.

I. Planification du Backlog Produit

Dans cette section notre Product Owner (PO) va présenter les spécifications de notre projet. Cette phase n'est que la traduction des besoins fonctionnels et non fonctionnels des utilisateurs. De ces besoins, le PO va élaborer le backlog Produit qui est la liste des fonctionnalités à implémenter.

1. Besoins fonctionnels

La spécification des besoins fonctionnels consiste à recenser et caractériser les différentes fonctions que la future application doit offrir pour satisfaire les besoins de ses futurs utilisateurs. Pour spécifier ces exigences fonctionnelles, notre PO va se baser sur le cahier des charges élaboré par nos collaborateurs en France.

La liste des fonctionnalités de notre application sont les suivantes :

- ✓ S'authentifier.
- ✓ Gérer les différents acteurs : Compagnie aérienne, Handler, Aéroport, Salariés.
- ✓ Gérer les droits d'accès.
- ✓ Générer les données des vols.
- ✓ Générer le processus des vols.
- ✓ Tester le processus de l'application.
- ✓ Adapter le processus au mapping des données des vols.
- ✓ Etablir les tableaux de bord (Statistiques par acteur)

2. Besoins non fonctionnels

Exigences qui identifient des contraintes internes et externes du système. Ce sont les caractéristiques de la qualité du logiciel en tant que facteurs et critères qui ne touchent pas aux objectifs métiers espérés mais contribuent à une meilleure qualité de la solution obtenue.

Pour notre application, nous nous intéressons aux contraintes ci-dessous citées dont certaines font partie implicitement de notre application au niveau des nouvelles technologies à exploiter :

- **La sécurité** : Grâce à une interface d'authentification, chaque utilisateur saisit son login, son mot de passe afin de pouvoir accéder à son espace personnel renfermant les interfaces des fonctionnalités qui lui sont attribuées.
- **Stabilité** : Notre application doit être efficace et capable de s'adapter à une forte augmentation de la demande sans que son niveau de performance n'en soit impacté.
- **Haute Disponibilité** : Maintenir le bon fonctionnement de notre application.
- **La maintenabilité** : Les programmes doivent être modulaires et bien structurés. Le code doit être lisible et compréhensible, il doit avoir l'opportunité d'être amélioré facilement et rapidement par tout ingénieur développeur.
- **Performance et efficacité** : Contenir un minimum d'erreurs, satisfaire les spécifications, remplir ses missions et permettre d'effectuer les opérations rapidement et de manière efficace pour gagner du temps.
- ❖ **Utilisabilité** : Ce besoin touche l'utilisateur par la compréhensibilité, la facilité d'apprentissage et l'attractivité. L'utilisateur doit quantifier la qualité du logiciel en regroupant les critères suivants :
 - La facilité à trouver l'information cherchée.
 - La compréhensibilité de l'information.
 - La capacité à présenter une information exacte avec une belle ergonomie.
- ❖ **Testabilité** : Faciliter les procédures de test permettant de s'assurer de l'adéquation des fonctionnalités.

3. Backlog Produit

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

- **Rel** : Identifiant de la **release** – Numéro séquentiel-

- **SP** : Identifiant du **sprint** – Numero séquentiel-
- **ID story** : C'est un nombre unique et auto-incrémenté pour chaque Nom de User Story.

- **Nom** : Le titre d'user story
- **User Story** : C'est une description courte de l'user story à réaliser et qui se définit de la manière suivante : En tant que <rôle>, je veux <faire quelque chose>.
- **Priorité « Pr »** : C'est l'importance attribuée par le Product Owner à une user story, la valeur (1) est plus prioritaire que (2), etc....
- **Comp** : C'est la complexité de chaque user story
- **Sem** : C'est la période du chaque user story.

Tableau 2: Backlog Produit

Rel.	SP.	ID Story	Nom	User Story	Pr.	Comp.	Sem.
1	0	0.1	Configuration et gestion de la plateforme « SPOTFLY DEMO ».	En tant qu'administrateur, je veux Paramétrer l'application.	1	M	W1
				En tant qu'administrateur, je veux gérer des clients.	1	M	
				En tant qu'administrateur, je veux gérer des formulaires.	1	M	
				En tant qu'administrateur, je veux gérer des états et Statu.	1	M	
				En tant qu'administrateur, je veux gérer des processus.	1	M	
				En tant qu'administrateur, je veux gérer des services.	1	M	
	0.2		Test de la plateforme « SPOTFLY DEMO » avec Squash Tm.	En tant que testeur, je veux tester le processus SPOTFLY DEMO avec squash TM.	2	M	W2
	1	1.1	Importation du fichier des vols.	En tant qu'administrateur, je veux importer le fichier des vols.	1	D	W3-4
		1.2	Création des Fields	En tant qu'administrateur, je veux Créer des Fields.	2	M	
		1.3	Mapping des vols	En tant qu'administrateur, je veux effectuer le mapping des vols.	3	D	
		1.4	Dispatching des vols	En tant qu'administrateur, je veux effectuer le Dispatching des vols.	4	M	
2	2	2.1	Préparation de Base de données « MongoDB »	En tant qu'administrateur, je veux préparer un Base de données	1	L	W5
			Authentification Administrateur	En tant qu'administrateur, je dois m'authentifier pour accéder à mon espace personnel.	1	M	
		2.3	Gestion des utilisateurs	En tant qu'administrateur, je veux gérer des utilisateurs (Crud).	1	M	
		2.4	Gestion des vols	En tant qu'administrateur, je veux gérer des vols (Crud).	1	L	
		2.5	Gestion des avions	En tant qu'administrateur, je veux gérer des avions (Crud).	1	L	

		3.2	Affectation des tâches aux différents utilisateurs	En tant qu'administrateur, je veux affecter les tâches aux différents utilisateurs.	2	D	W6-7
		3.3	Consultation les historiques des tâches réalisés.	En tant qu'administrateur, je veux consulter les historiques des réalisés.	2	D	
		3.4	Envoie des notifications.	En tant qu'administrateur, je peux envoyer des notifications.	2	D	
		3.5	Réception des notifications.	En tant qu'utilisateur, je veux recevoir des notifications.	2	D	
3	4	4.1	Authentification utilisateur via une application web.	En tant qu'utilisateur, je dois m'authentifier pour exploiter mon espace personnel.	1	M	W8-9
		4.2	Consultation l'historique des tâches réalisées.	En tant qu'utilisateur, je veux consulter les tâches réalisées (historiques) via une application web.	1	M	
		4.3	Consultation des nouvelles tâches affectées.	En tant qu'utilisateur, je veux consulter et exécuter les nouvelles tâches affectées via une application web.	1	D	
		4.4	Envoie des notifications.	En tant qu'utilisateur, je peux envoyer des notifications via une application web.	1	M	
		4.5	Réception des notifications.	En tant qu'administrateur, je veux recevoir des notifications via une application web.	1	M	
	5	5.1	Authentification utilisateur via une application mobile.	En tant qu'utilisateur, je dois m'authentifier pour exploiter mon espace personnel.	1	M	W10-11
		5.2	Consultation de l'historique	En tant qu'utilisateur, je veux consulter les tâches réalisées (historiques) via une application mobile.	1	M	
		5.3	Consultation de l'historique	En tant qu'utilisateur, je veux consulter et exécuter les nouvelles tâches affectées via une application mobile.	1	M	
		5.4	Envoie des notifications.	En tant qu'utilisateur, je peux envoyer des notifications via une application mobile.	1	M	
		5.5	Réception des notifications.	En tant qu'administrateur, je veux recevoir des notifications via une application mobile.	1	M	

II. Environnement Technologique

Dans cette section, nous allons présenter les logiciels de développement et de test de processus que nous allons déployer tout au long de notre projet.

1. Environnement matériel

L'environnement matériel utilisé lors de la réalisation de l'application, est :

Élément	Valeur
Nom du système d'exploitation	Microsoft Windows 10 Professionnel
Version	10.0.19044 Build 19044
Ordinateur	DESKTOP-R2LP77J
Fabricant	LENOVO
Type	PC à base de x64
Processeur	Intel(R) Core(TM) i5-7200U CPU @ 2.50GHz, 2701 MHz, 2 cœur(s), 4 processe...
Version du contrôleur embarqué	1.19
Mémoire physique (RAM) instal...	8,00 Go

Figure 11: Caractéristiques du PC

2. Environnement logiciel



Angular est un Framework coté client, Pratiquement base sur TypeScript.

Il permet de créer des applications web (SPA). <<Single Page Application>> ce sont des interfaces accessibles via une page unique pour éviter les chargements des pages à chaque nouvelle action.

Angular aussi est basé sur une architecture de type MVC (Model, View, Controller) qui permet de séparer les données.



PrimeNG est une collection de composants d'interface utilisateur riches pour Angular. Tous les widgets sont open source et utilisables gratuitement sous licence MIT.

PrimeNG est développé par PrimeTek Informatics , un fournisseur avec des années d'expertise dans le développement de solutions d'interface utilisateur open source. Pour les nouvelles et les mises à jour du projet.



TypeScript est un langage de programmation libre et open source développé par Microsoft ayant pour but d'améliorer et de sécuriser la production de code JavaScript. Il est un sur-ensemble syntaxique strict de JavaScript (c'est-à-dire que tout code JavaScript correct peut être utilisé avec TypeScript). Le code TypeScript est transcompilé en JavaScript, et peut ainsi être interprété par n'importe quel navigateur web ou moteur JavaScript.^[Y]



MongoDB est une base de données NoSQL orientée documents. Il permet de manipuler des objets structurés au format BSON (JSON binaire), sans schéma prédéterminé. Des clés peuvent être ajoutées à tout moment « à la volée », sans reconfiguration de la base.

Les données prennent la forme de documents enregistrés eux-mêmes dans des collections, une collection contenant un nombre quelconque de documents. Les collections sont l'équivalent aux tables, et les documents aux enregistrements des bases de données relationnelles. Contrairement aux bases de données relationnelles, les champs d'un enregistrement sont libres et peuvent être différents d'un enregistrement à un autre au sein d'une même collection. Le seul champ commun et obligatoire est le champ de la clé principale ("id").

Le point fort de MongoDB est que l'ensemble du système tourne autour de la gestion de documents y compris le langage d'interrogation, par ailleurs, il présente un inconvénient qu'il ne permet ni les requêtes très complexes standardisées, ni les JOIN, mais permet de programmer des requêtes spécifiques en JavaScript [hh].

Modélisation de documents

Documents JSON

Comment seront présentées les données sous formes des documents JSON. Le modèle est très simple :

- Tout est clé/valeur : “clé” : “valeur”,
- Un document est encapsulé dans des accolades {...}, pouvant contenir des listes de clés/valeurs,
- Une valeur peut être un type scalaire (entier, nombre, texte, booléen, null), des listes de valeurs [...], ou des documents imbriqués.

Comment faire pour passer du relationnel vers JSON?

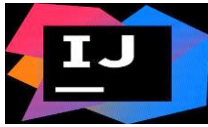
En effet, les données utilisées de manière classique sont en relationnel Mais dans ce cas, nous devons faire face à des bases de données NoSQL avec une proscription des jointures. De fait, si nous avons un SGBDR avec deux tables, nous n'allons pas faire de requêtes de jointure mais Il faut dans ce cas réaliser une fusion des deux tables pour n'en produire qu'une seule en sortie, ce que nous appelons la dénormalisation.



Robo 3T (anciennement Robomongo) est un outil de gestion MongoDB multiplateforme centré sur le shell. Contrairement à la plupart des autres outils d'interface utilisateur d'administration MongoDB, Robo 3T intègre le shell mongo réel dans une interface à onglets avec accès à une ligne de commande shell ainsi qu'à une interaction avec l'interface graphique.



Spring est un framework open source pour construire et définir l'infrastructure d'une application Java, dont il facilite le développement et les tests.



IntelliJ IDEA également appelé « IntelliJ », « IDEA » ou « IDJ » est un environnement de développement intégré (en anglais Integrated Development Environment - IDE) destiné au développement de logiciels informatiques reposant sur la technologie Java.

IntelliJ IDEA est un IDE développé par le studio JetBrains. Il est disponible en deux éditions différentes, une édition communautaire gratuite et une édition commerciale payante.

Avec cet IDE, tous les outils nécessaires pour développer efficacement en Java sont disponibles. Pour faciliter la vie des développeurs, IntelliJ IDEA offre de la complétion en plus de la prise en charge de Java et d'une multitude de frameworks Java.



Git C'est un outil qui permet de gérer différents projets en les envoyant sur un serveur. Les développeurs connectés envoient leur code et récupèrent.



Node JS est un environnement d'exécution open source et multiplateforme qui permet aux développeurs de créer diverses applications et outils côté serveur en JavaScript. Cet environnement est conçu pour être utilisé en dehors du navigateur (Il s'exécute directement sur votre ordinateur ou dans le système d'exploitation du serveur).



Est le framework actuellement le plus populaire dans Node. Il ne fournit que quelques fonctionnalités, il peut en revanche être complété par de nombreuses librairies disponibles sur **npm**. Vous êtes libre de choisir les librairies et l'architecture backend qui vous conviennent le mieux.



Est une bibliothèque JavaScript qui permet de simplifier la création d'interface. Elle permet une synchronisation plus simple et plus efficace entre les données de votre application et le rendu HTML.



React Native est un Framework conçu par Facebook nous permet aussi d'accélérer le développement de vos applications mobiles. Nos développeurs mobiles misent aussi sur sa grande modularité. Elle permet de vous garantir la maintenance du code de votre application et sa capacité à évoluer en fonction des besoins de vos utilisateurs mobinautes.



Expo est à la fois un framework et une plateforme qui simplifie la création et le déploiement d'applications mobiles à l'aide de React Native. Expo intègre de nombreux outils utiles et bibliothèques natives pour React Native. Il gère également les mises à jour de ces bibliothèques. C'est donc un moyen simple et rapide de démarrer un projet.

3. Environnement de Test

Dans cette section, nous allons présenter des logiciels de test des processus et nous allons choisir en collaboration de l'équipe **SCRUM** le logiciel à déployer.



Squash est une suite d'outils permettant de concevoir, automatiser, exécuter et industrialiser des tests. Basée sur un socle open source, la solution est modulaire et facile à intégrer. Il fonctionne dans tous les environnements de projet : Agilité à l'échelle de type V-Cycle, Agile et SAFe. [x].



Sélénium est un cadre de test automatisé gratuit (open source) pour valider les applications Web et mobiles sur différents navigateurs et plates-formes. Il peut utiliser Java, C #, Python et d'autres langages de programmation. Créer des scripts de test Sélénium. Les tests effectués à l'aide des outils de test Sélénium sont souvent appelés tests Sélénium. [x1].

❖ **Choix du logiciel de test du processus**

Pour gérer l'ensemble des étapes d'une recette : de la gestion des exigences à l'exécution des campagnes de test d'un projet, nous avons choisis SQUASH TM.

SQUASH TM est un gestionnaire de cas de test. Il permet de collecter un ensemble de cas de test, de les rédiger, de les organiser, de les exécuter, et de présenter leurs résultats. Il fonctionne comme une application Web et supporte les Navigateurs : Firefox, Chrome, Microsoft Edge. Squash présente 4 principales étapes et quatre espaces que nous allons citer.

✓ **Les étapes de test avec Squash TM :**

1. Création du projet de test
2. Définition de l'exigence
3. Définition du cas de test
 - 3.1 Rédaction du cas de test
 - 3.2 Paramétrage du cas de test
 - 3.3 Préparation du jeu de données
4. Exécution de cas de test

✓ **Squash TM présente 4 espaces :**

○ **Espace des exigences :**

Une exigence est un objet dans l'espace des exigences. Toute exigence ou règle de gestion qualifiant le système peut être présentée dans Squash TM comme une exigence qualifiante ("Criticité", "Statut", "Catégorie" etc.) pour créer un référentiel d'exigences souhaitées.

○ **Espace cas de test :**

Un cas de test est un objet dans l'espace de cas de test. Il est défini par la condition préalable, les jeux de données à compiler, les activités à réaliser et les résultats attendus. L'exécution d'un cas de test doit permettre de vérifier étape par étape la conformité du comportement attendu du système testé, et ainsi l'identification d'éventuelles anomalies.

- **Espace Campagne :**

Les phases d'exécution ou "activités" sont régies par certains principes :

- Les activités commencent avec la première livraison de développement et se terminent avec la production (MEP).
- Une activité (ou phase d'exécution) est découpée en itérations (ou cycles), Chaque itération est définie par l'intervalle de temps entre deux livrables de développement (évolution et/ou révision).

- **Espace Pilotage :**

L'espace Pilotage permet de créer des éléments de reporting tels que des graphiques ou des tableaux de bord personnalisés.

Conclusion

Dans ce chapitre, notre PO a identifié les besoins fonctionnels et non fonctionnels de notre système. Ensuite, le diagramme de cas d'utilisation global et le diagramme de classe ainsi que le backlog produit ont été établis. Alors que l'équipe Scrum a présenté l'environnement matériel et logiciel à utiliser pour développer notre projet. Dans le chapitre suivant, Nous entamons le développement de la première Release.

Chapitre 4 :
Implémentation de la Release 1

Introduction

En se basant sur le principe “Scrum”, nous commençons par définir l’objectif de cette release. Suite à une conversation entre les membres de l’équipe, nous avons décidé l’objectif suivant : développer, tester et valider les différents sprints de notre première release.

I. Planification et implémentation du sprint 0

1. But de Sprint

Le but de ce sprint consiste à configurer et paramétrer l’application « SPOTFLY DEMO » avec le logiciel ERM et tester le processus de ces fonctionnalités avec Squash TM.

2. Sprint Backlog

Ce tableau présente le backlog de ce sprint ainsi que l’estimation de la période et la complexité de chaque User Story.

Tableau 3: Backlog Sprint 0

Rel.	SP.	ID Story	Nom	User Story	Pr.	Comp.	Sem.
1	0	0.1	Configuration et gestion de la plateforme « SPOTFLY DEMO ».	En tant qu’administrateur, je veux Paramétrer l’application.	1	M	W1
				En tant qu’administrateur, je veux gérer des clients.	1	M	
				En tant qu’administrateur, je veux gérer des formulaires.	1	M	
				En tant qu’administrateur, je veux gérer des états et Status.	1	M	
				En tant qu’administrateur, je veux gérer des processus.	1	M	
				En tant qu’administrateur, je veux gérer des services.	1	M	
		0.2	Test de la plateforme « SPOTFLY DEMO » avec Squash Tm.	En tant que testeur, je veux tester le processus SPOTFLY DEMO avec squash TM.	2	M	W2

3. Réalisation et Test

II. Planification et implémentation du sprint 1

1. But de Sprint

Le but de ce sprint consiste à développer le module gestion des flux des vols avec une simulation externe pour l'intégrer au projet.

Qui permet l'utilisateur de l'application d'importer un fichier Excel contenant des vols pour faire le mapping de chaque source colonne avec le Field crée pour le remplir avec le contenu de chaque colonne de fichier Excel pour obtenir un document qui représente un vol.

2. Sprint Backlog

Ce tableau présente le backlog de ce sprint ainsi que l'estimation de la période et la complexité de chaque User Story.

Tableau 4: Backlog de sprint 1

Rel.	SP.	ID Story	Nom	User Story	Pr.	Comp.	Sem.
	1	1.1	Importation du fichier des vols.	En tant qu'administrateur, je veux importer le fichier des vols.	1	D	W3-4
		1.2	Création des Fields	En tant qu'administrateur, je veux Créer des Fields.	2	M	
		1.3	Mapping des vols	En tant qu'administrateur, je veux effectuer le mapping des vols.	3	D	
		1.4	Dispatching des vols	En tant qu'administrateur, je veux effectuer le Dispatching des vols.	4	M	

3. Expression des besoins

3.1 Diagramme de cas d'utilisation

Acteur principal : l'administrateur.

Pré condition : Administrateur connecté

Post condition : Mise a jour de la base de données.

Description : Tout d'abord, l'administrateur importe un fichier de vol et récupère la source de la colonne du fichier Excel dans la base de données.

En tant que deuxième lien, il crée le champ et l'enregistre dans la base de données.

Il fait ensuite un mappage "d'assignation" entre chaque colonne source et un champ pour obtenir des documents contenant des vols.

Enfin, il expédie le vol (chaque fichier représente un formulaire).

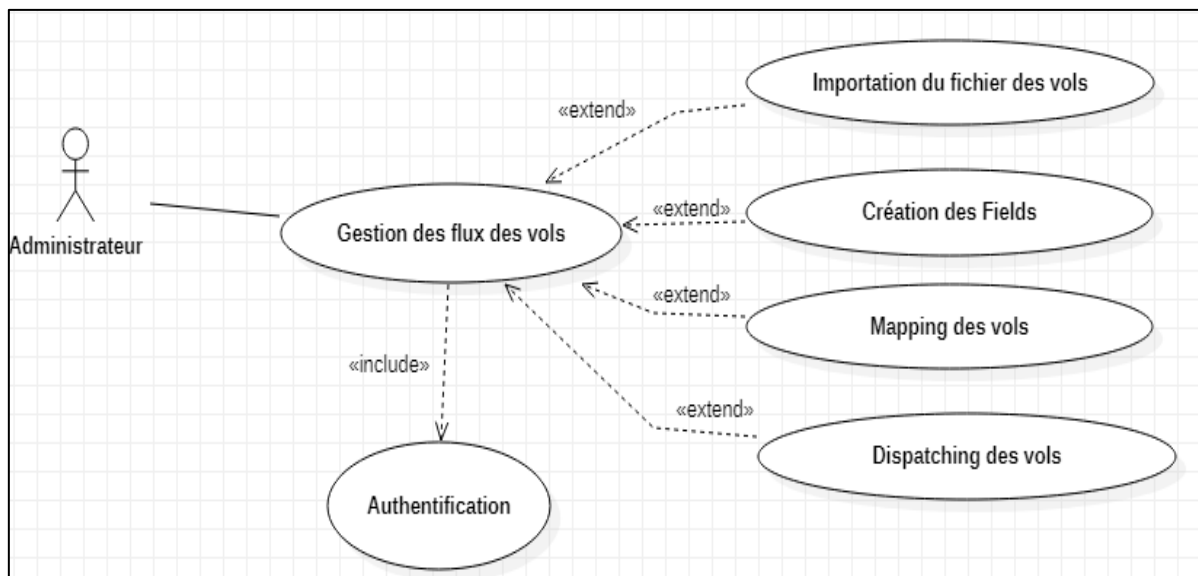
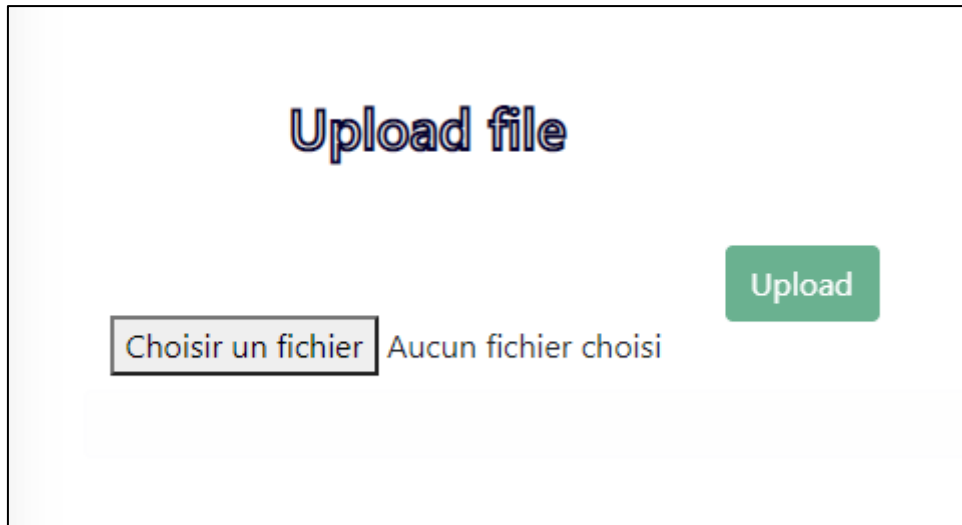


Figure 12: Diagramme de cas d'utilisation Sprint 1

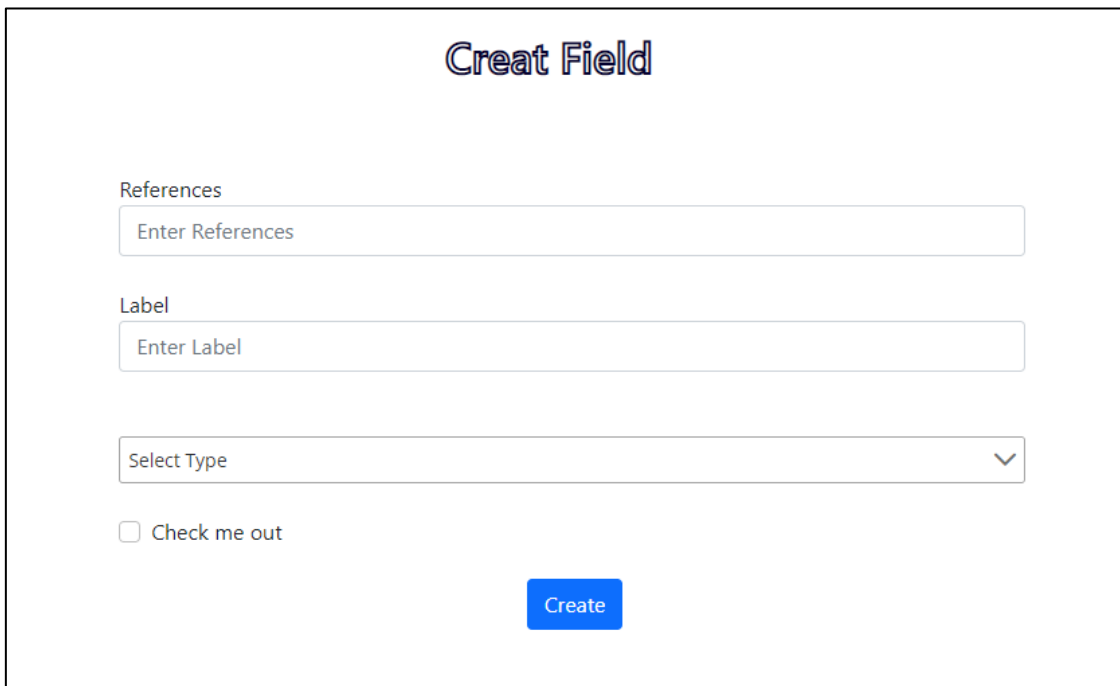
3.2 Diagramme des classes

4. Réalisation



The image shows a web interface for uploading a file. At the top, the text "Upload file" is displayed in a large, stylized font. Below this, there is a green button labeled "Upload". To the left of the button, there is a text input field with the placeholder text "Choisir un fichier" and a label "Aucun fichier choisi" next to it. The input field is highlighted with a red border.

Figure 13: Simulation d'Importation des fichiers



The image shows a web interface for creating a new field. At the top, the text "Creat Field" is displayed in a large, stylized font. Below this, there are three input fields: "References" with the placeholder text "Enter References", "Label" with the placeholder text "Enter Label", and "Select Type" with a dropdown arrow. Below these fields, there is a checkbox labeled "Check me out". At the bottom, there is a blue button labeled "Create".

Figure 14: Simulation pour la création des Fields

Mapping

Fly_PFE.xlsx ✕ ▾

Column	Field	PK	Save	
Select Column ▾	Select Field ▾	■	■	Delete

OK

Figure 15: Simulation de mapping des vols

Fly_PFE.xlsx ✕ ▾

Column	Field	PK	Save	
Country ✕ ▾	pays ✕ ▾	■	■	Delete
Year ✕ ▾	Année ✕ ▾	■	■	Delete
Rank ✕ ▾	Rang ✕ ▾	■	■	Delete
Total ✕ ▾	Totale ✕ ▾	■	■	Delete

OK

Figure 16: Simulation de mapping des vols

pays	Année	Rang	Totale
Yemen	2021	1st	111.7
Somalia	2021	2nd	110.9
Syria	2021	3rd	110.69999999999999
South Sudan	2021	4th	109.4
Congo Democratic Republic	2021	5th	108.39999999999999
Central African Republic	2021	6th	107
Chad	2021	7th	105.8
Sudan	2021	8th	105.19999999999999
Afghanistan	2021	9th	102.1
Zimbabwe	2021	10th	99.10000000000001
Ethiopia	2021	11th	99.00000000000001
Nigeria	2021	12th	98

Figure 17: Simulation de dispatching des vols

Conclusion

Dans ce chapitre, nous sommes arrivés à terminer la simulation du module gestion des flux des vols. De ce fait, je suis amené à étudier les étapes de la mise en œuvre du prochaine release que je vais détailler dans le chapitre suivant.

Chapitre 5 :
Implémentation de la Release 2

Introduction

En se basant sur le principe “Scrum”, nous commençons par définir l’objectif de cette release. Suite à une conversation entre les membres de l’équipe, nous avons décidé l’objectif suivant : développer, tester et valider les différents sprints de notre première release.

I. Planification et implémentation du sprint 2

1. But de Sprint

Le but de ce sprint consiste à développer les modules gestion des utilisateur, vols, avion.

2. Sprint Backlog

La planification des différentes fonctionnalités qui constituent le présent incrément est présentée dans le tableau suivant :

Tableau 5: Backlog Sprint 2

Rel.	SP.	ID Story	Nom	User Story	Pr.	Comp.	Sem.
2	2	2.1	Préparation de Base de données « MongoDB »	En tant qu’administrateur, je veux préparer un Base de données	1	L	W5
		2.2	Authentification Administrateur	En tant qu’administrateur, je dois m’authentifier pour accéder à mon espace personnel.	1	M	
		2.3	Gestion des utilisateurs	En tant qu’administrateur, je veux gérer des utilisateurs (Crud).	1	M	
		2.4	Gestion des vols	En tant qu’administrateur, je veux gérer des vols (Crud).	1	L	
		2.5	Gestion des avions	En tant qu’administrateur, je veux gérer des avions (Crud).	1	L	

3. Expression des besoins

3.1 Diagramme de cas d’utilisation

Acteur principal : l’administrateur (Responsable), (compagnie aérienne).

Pré condition : Administrateur connecté

Post condition : Mise à jour de la base de données

Description : Après une authentification l’administrateur a le droit de faire la gestion de différents modules (utilisateur « technicien », « handler », Vols, Avions)

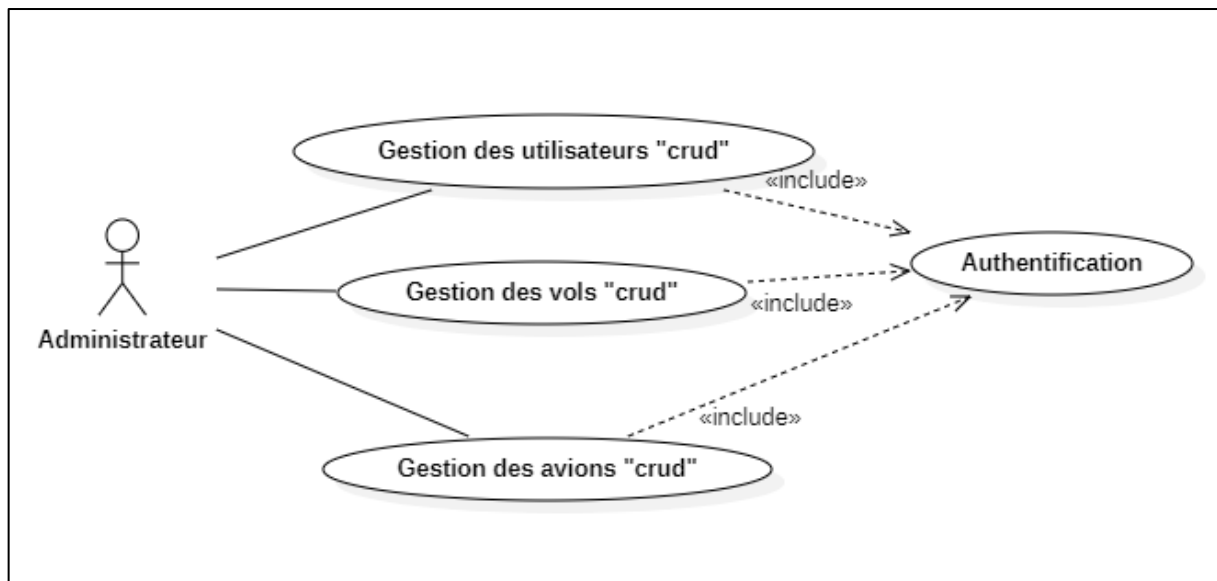


Figure 18 : Diagramme de cas d'utilisation Sprint 2

3.2 Diagramme des classes

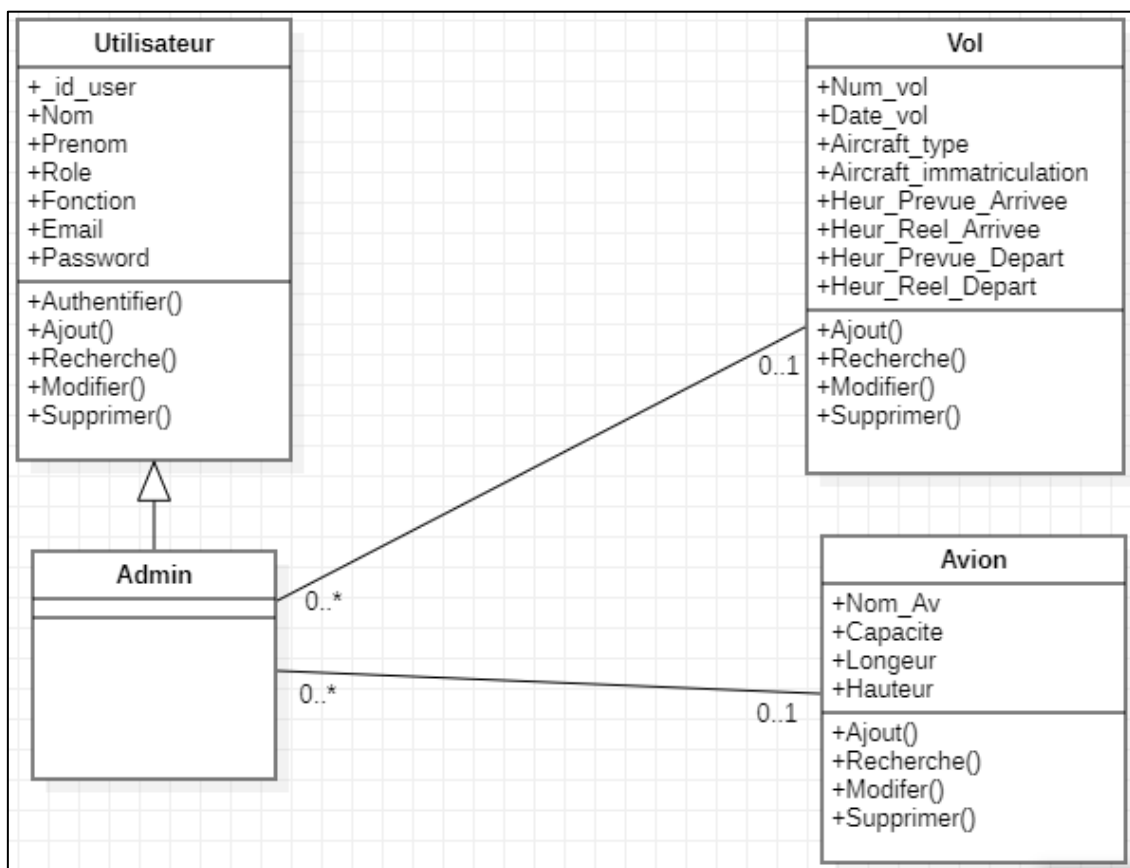


Figure 19 : Diagramme des classe Sprint 2

4. Réalisation

4.1 Préparation de la base de données

Cette interface présente la préparation d'une base de données sur MongoDB.com

Comme montre la figure suivante :

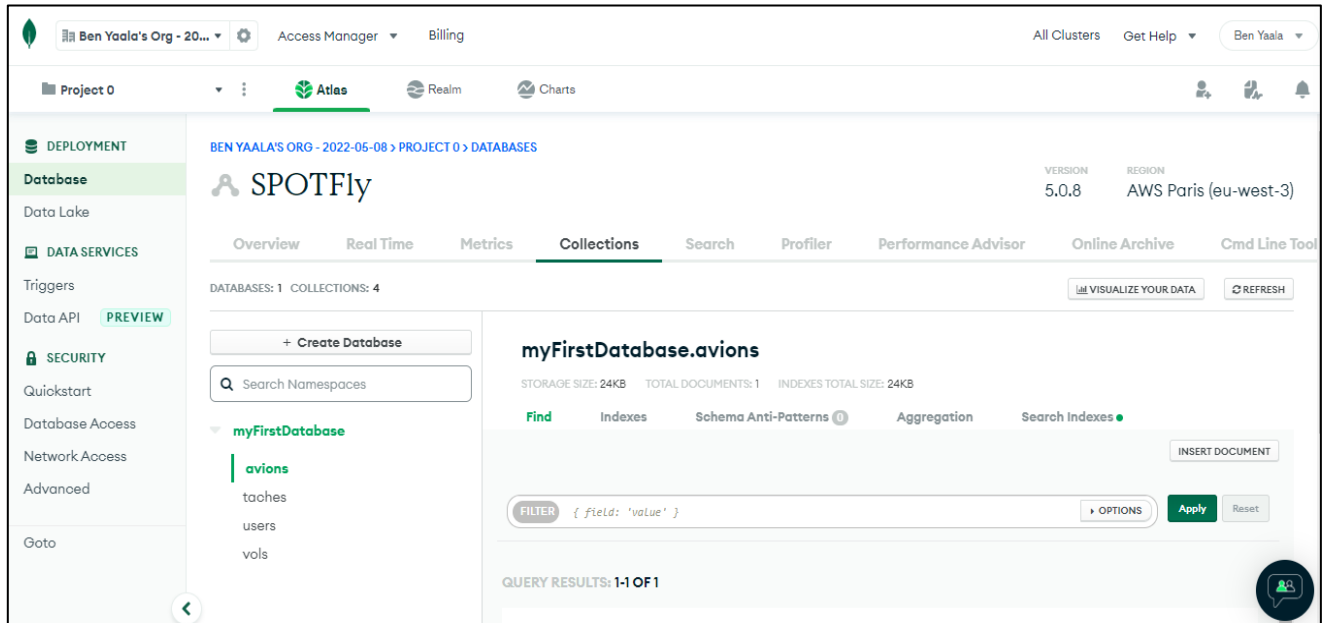


Figure 20 : Base de données MongoDB

4.2 Interface d'Authentification

Après la création de la base de données No SQL MongoDB nous commençons à préparer l'interface de page login (web) qui permet l'administrateur de s'authentifier à leur propre compte.

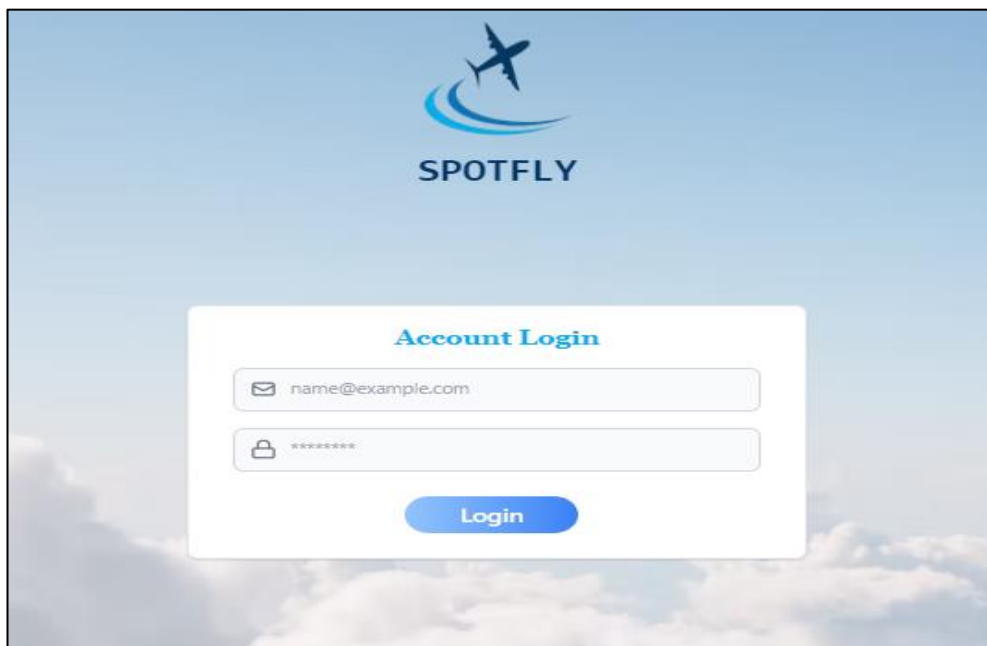


Figure 21: Interface Login

4.3 Gestion des utilisateurs « technicien »

Après l'authentification, l'administrateur « Responsable » prend l'accès à faire les gestions des utilisateurs « technicien », l'ajout de technicien, la modification des cordonnées ainsi qu'il peut le supprimer.

<input type="text" value="Search.."/>					<input type="button" value="Add User"/>	
NOM	PRENOM	EMAIL	ROLE	FONCTION		
azerty	azert	chefescale@gmail.com	technicien	chef d'escale		
Ben yaala	Ayoub	ayoubbenyaala@gmail.com	responsable	Chef		
Trabelsi	Eya	eyatrabelsi@gmail.com	responsable	Chef		

Figure 22 : Interface Gestions des utilisateurs

4.4 Gestion des vols

L'administrateur peut ajouter des vols selon le besoin, modifier les heure de départ, les heure d'arrivée, ...(ect) ainsi que supprimer des vols annuler.

<input type="text" value="Search.."/>								<input type="button" value="Add Vol"/>	
NUM VOL	DATE VOL	AIRCRAFT TYPE	AIRCRAFT IMMATRICULATION	HEURE PRÉVUE ARRIVÉE	HEURE RÉEL ARRIVÉE	HEURE PRÉVUE DÉPART	HEURE RÉEL DÉPART		
TU354	2022-11-04T23:00:00.000Z	TUNISAIRE	123456TU	20:00	21:00	15:00	16:00		
TU354	2022-11-04T23:00:00.000Z	TUNISAIRE	123456TU	20:00	21:00	15:00	16:00		
TU354	2022-11-04T23:00:00.000Z	TUNISAIRE	123456TU	20:00	21:00	15:00	16:00		
TU450	2022-05-29T00:00:00.000Z	TUNISAIR	TU123456	15:00	15:30	18:00	20:00		

Figure 23: Interface Gestion des Vols

4.5 Gestion des avions

Nous avons terminé ce sprint par le module de gestion des avions qui permettrait à l'administrateur « responsable » d'ajouter des nouveaux avions et les modifier ces caractéristiques (capacité, nombre des vols par jours), ainsi de supprimer des avions en panne ou détruites.

<input type="text" value="Search.."/>				<input type="button" value="Add avion"/>	
NOM AVION	CAPACITE	LONGUEUR	HAUTEUR		
Tunisair	500	15	8		

Figure 24: interface Gestion des avions

II. Planification et implémentation du sprint 3

1. But de Sprint

Conclusion

À travers ce chapitre, nous sommes arrivés à préparer la majorité des modules de gestion « CRUD ». Dans le chapitre suivant nous amenons les étapes de la mise en œuvre du prochain sprint.

Chapitre 6 :

Implémentation de la Release 3

Introduction