



Rapport de Projet de Fin d'Etudes

LICENCE APPLIQUEE EN TECHNOLOGIES DE L'INFORMATIQUE

**PARCOURS : DEVELOPPEMENT DES SYSTEMES D'INFORMATIONS
(DSI)**

**Conception et réalisation d'une plateforme suivi
& pilotage de projets (Api ClickUp)**

Entreprise : AKEOPLUS TUNISIE

Réalisé par :

Iskander ABESSI

Achraf NOUIRA

Encadré par : Sourour MEDDEB

Encadreur(s) entreprise : Mohamed CHAABANE

Encadreur(s) ISET : Sami SEDDIK

Dédicaces

Dédicaces

Remerciement

Table des matières

Remerciement.....	4
Listes des figures	9
Listes des tableaux	10
Introduction Générale.....	1
Chapitre 1 : Présentation du cadre de projet	3
Introduction :	3
1. Présentation de la société :	3
1.1. A Propos :	3
1.2. Les compétences :	3
1.3. Secteurs d'activités :	4
2. Contexte du projet :	4
3. ClickUp :	4
3.1. Qui utilise ClickUp ?	4
3.2. Description de ClickUp :	5
3.3. Informations sur ClickUp :	5
3.4. Prix de ClickUp :	5
4. Description de l'existant :	5
5. Critique de l'existant :	6
6. Solution proposée :	6
7. Méthodologie et formalisme adoptés :	6
7.1. Méthodologie de conception :	6
7.2. Méthodologie de travail :	6
7.2.1. Comparaison des approches classique et agile :	7
7.2.2. Méthode Agile :	8
7.2.3. Etude comparative des méthodes agiles :	8
7.2.4. La méthode adoptée SCRUM :	9

Conclusion :	9
Chapitre 2 : Etude Préalable	9
Introduction :	9
1. Spécification des besoins :	9
1.1. Identification des acteurs :	9
1.2. Besoins fonctionnels :	9
1.3. Besoins non fonctionnels :	9
2. Pilotage du projet avec Scrum :	10
2.1. User Stories :	10
2.2. Equipe Scrum :	11
2.3. Product Backlog :	12
2.4. Planification Des Sprints :	15
3. Diagramme de cas d'utilisation global :	16
4. Conception Architecturale :	17
4.1. Architecture logique :	17
4.2. Architecture de l'application :	17
5. Environnement de travail :	19
5.1. Environnement matériel :	19
5.2. Environnement logiciel :	19
5.3. Les langages :	20
5.4. Les Framework :	21
5.5. Les bibliothèques :	21
Conclusion :	22
Chapitre 3 : Release 1	22
Introduction	22
1. Sprint 1 Module Gestion :	22
1.1. Backlog du sprint :	22

1.2.	Analyse :	23
1.2.1.	Diagramme des cas d'utilisation « Sprint 1 » :	24
1.2.2.	Description Textuelle :	24
5.1.	Conception :	26
5.1.1.	Diagrammes de séquence :	26
5.2.	Réalisation :	27
5.3.	Test :	28
6.	Sprint 2 Dashboard (les statistiques) :	29
6.1.	Backlog du sprint :	29
6.2.	Analyse :	29
6.2.1.	Diagramme du cas d'utilisation « Sprint 2 » :	29
6.2.2.	Description Textuelle :	30
6.3.	Conception :	30
6.4.	Réalisation :	31
6.5.	Test :	32
	Conclusion :	32
Chapitre 4 : Release 2		32
	Introduction	32
1.	Sprint 1 Graphiques de données (widgets) :	32
1.1.	Backlog du sprint :	32
1.2.	Analyse :	33
1.2.1.	Diagramme des cas d'utilisation « Sprint 1 » :	33
1.2.2.	Description Textuelle :	33
1.3.	Conception :	33
1.3.1.	Diagramme de classe :	33
1.3.2.	Diagrammes de séquence :	34
1.4.	Réalisation :	34

1.5. Test :	36
2. Sprint 2 La phase de planification (calendrier) :	36
2.1. Backlog du sprint :	37
2.2. Analyse :	37
2.2.1. Diagramme de cas d'utilisation « Sprint 2 » :	38
2.2.2. Description textuelle :	38
2.3. Conception :	39
2.4. Réalisation :	40
2.5. Test :	41
3. Sprint 3 Module paramétrage :	41
3.1. Backlog du sprint :	42
3.2. Analyse :	42
3.2.1. Diagramme des cas d'utilisation « Sprint 3 » :	43
3.2.2. Description Textuelle :	43
3.3. Conception :	44
3.3.1. Diagramme de classe :	44
3.3.2. Diagrammes de séquence :	44
3.4. Réalisation :	46
3.5. Test :	48
Conclusion :	48
Conclusion Générale	48
Nétographie	49
Annexe :	50

Listes des figures

Figure 1 : Logo Société AkeoPlus.....	4
Figure 2 : Processus Scrum	9
Figure 3 : Diagramme des cas d'utilisation global	16
Figure 4 : architecture 3 tiers.....	17
Figure 5 : Architecture de l'application	18
Figure 6 : Diagramme de cas d'utilisation Sprint 1 Release 1.....	24
Figure 7 : Diagramme de séquence « Ajouter espace de travail »	27
Figure 8 : Interface gestion tâches.....	28
Figure 9 : Interface gestion projets.....	28
Figure 10 : Diagramme de cas d'utilisation sprint 2 release 1	30
Figure 11 : Diagramme de collaboration consulter Dashboard.....	31
Figure 12 : Diagramme de cas d'utilisation sprint 1 release 2.....	33
Figure 13 : Diagramme de classes sprint 1 release 2	34
Figure 14 : Interface ajouter widget	35
Figure 15 : interface indicateurs widget	35
Figure 16 : Interface graphiques de données.....	36
Figure 17 : Diagramme de cas d'utilisation sprint 2 release 2.....	38
Figure 18 : Diagramme de collaboration du cas d'utilisation « Gérer calendrier »	39
Figure 19 : Interface calendrier	40
Figure 20 : Interface de gestion tâche (calendrier).....	40
Figure 21 : Interface ajouter une nouvelle tâche (calendrier)	41
Figure 22 : Diagramme de cas d'utilisation sprint 3 release 2.....	43
Figure 23 : Diagramme de séquence « S'authentifier »	46
Figure 24 : Interface gestion membres	47
Figure 25 : Interface authentification	47

Listes des tableaux

Tableau 1 : Comparaison entre méthodes classiques et méthodes agiles.....	7
Tableau 2 : Tableau comparatif entre méthode Scrum et XP	8
Tableau 3 : Identification d'acteur	9
Tableau 4 : Product Backlog	14
Tableau 5 : Planification des sprints	15
Tableau 6 : Environnement matériel	19
Tableau 7 : Backlog de sprint 1 release 1	23
Tableau 8 : Tableau de test de sprint 1 release 1	29
Tableau 9 : Backlog sprint 2 release 1	29
Tableau 10 : Tableau de test sprint 2 release 1	32
Tableau 11 : Backlog sprint 1 release 2	32
Tableau 12 : Backlog sprint 2 release 2	37
Tableau 13 : Tableau de test de sprint 2 release 2.....	41
Tableau 14 : Backlog sprint 3 release 2	42
Tableau 15 : Tableau de test sprint 3 release 2.....	48

Introduction Générale

La gestion de projet est une discipline qui a pris racine depuis le 19ème siècle. Il ne s'est développé que dans les années 1960. L'informatique a aidé en automatisant la Gestion d'entreprise avec une série de tâches précédemment effectuées par la main.

De nos jours, la plupart des entreprises ont utilisé le système d'information comme un support d'aide à la décision car il permet d'améliorer l'efficacité et la qualité des décisions prises, c'est un outil de travail coopératif autour de données consolidées et correctement partagées.

Les entrepreneurs souhaitent améliorer les niveaux d'efficacité et de productivité, ainsi que la croissance des revenus, qui **dépendent** du bon choix de projets et d'excellentes performances d'utilisation des ressources.

Alors **pour** notre projet de fin d'études, la société AkeoPlus nous a confié le développement d'une application web pour la gestion des projets basé sur API Clickup.

Le présent manuscrit récapitule les étapes de notre mission qui consiste à réaliser une application orientée web qui se **présente** comme suit :

Dans le premier chapitre, intitulé « Présentation du cadre de projet » nous **présentons** l'organisme d'accueil, le contexte général du projet et les méthodologies **adoptées**.

Dans le deuxième chapitre intitulé « Etude préalable » sera consacré pour **l'**analyse des besoins, une présentation des tâches à faire au cours de ce projet et aussi pour la conception générale **suivie** par la présentation de l'architecture générale du projet.

Les deux derniers chapitres « Release 1 » et « Release 2 » constituent le corps de notre rapport. Ces deux chapitres seront consacrés pour le développement de notre système en respectant **les principes fondamentaux de SCRUM.** **{qu'est-ce que tu entends dire principaux ? ou principe}**

Chapitre 1 : Présentation du cadre de projet

Enfin, nous clôturons notre rapport par une conclusion générale qui résume l'ensemble de notre travail.

Chapitre 1 : Présentation du cadre de projet

Introduction :

Ce projet s'inscrit dans le cadre de la préparation de projet de fin d'étude pour l'obtention de la licence appliquée en Technologies de l'Informatique (TI) parcours : Développement Système d'information (DSI) de l'Institut Supérieur des Etudes Technologiques de Radès pour l'année universitaire 2019/2020. Ce projet a été effectué au sein de la société AKEOPLUS Tunisie.

Ce chapitre traite d'abord la présentation de l'organisme d'accueil AKEOPLUS, par la suite le cadre général du projet à savoir, le contexte du projet, la description, les critiques ainsi que la solution proposée et nous présenterons la méthodologie de la gestion du projet et de conception **que** nous avons adaptés.

1. Présentation de la société :

1.1.A Propos :

AKEOPLUS est une société d'ingénierie spécialisée en Mécanique et robotique, Électronique et Mécatronique, Informatique et Data, ainsi que dans la sous-traitance. Elle est créée en 2006 en développant des algorithmes pour améliorer les performances des capteurs sur des robots multi-axes pour obtenir des performances supérieures dans les **procédés** d'inspection. Depuis 2011, AKEOPLUS a commencé une forte croissance en quadruplant son chiffre d'affaires et ses effectifs.

L'expertise d'AKEOPLUS est basée sur l'association des technologies les plus avancées des applications de production, usinage, contrôle de qualité et mobilité/transport. Leurs domaines de compétence sont couplés autour de l'automatisme, la data et l'excellence opérationnelle pour créer des assistants 4.0 adaptatifs et évolutif dédiés.[1]

1.2.Les compétences :

- ✓ Digital : Le digital est central pour AKEOPLUS. Ils numérisent leur savoir-faire technologique combiné au savoir-faire métier de ces clients à travers des outils digitaux, étudiés et sélectionnés pour chaque projet.
- ✓ Automation : L'Automation est une activité historique pour AKEOPLUS. Fort de ces 15 **ans** d'expérience en industrialisation de machine unitaire et de série, ils attachent aujourd'hui à être 100% agnostique afin de toujours proposer la meilleure solution au juste coût et juste besoin à ces clients.

Chapitre 1 : Présentation du cadre de projet

- ✓ safety : La sécurité des biens, des personnes et des données est la pierre angulaire du fonctionnement de toute structure et métier. Crucial mais aussi obligatoire, ils accompagnent ces clients sur les volets SSE (Santé, Sécurité, Environnement), machines et cybersécurité.

1.3.Secteurs d'activités :

- Aérospatial
- Manufacturier
- Agro-Alimentaire
- Mobilité
- Énergie



Figure 1 : Logo Société AkeoPlus

2. Contexte du projet :

La société Akeoplus **met** en place une nouvelle stratégie pour gérer ses projets et son équipe. Cette stratégie poursuit un processus qui commence de l'acceptation de l'offre jusqu'à la livraison au client ainsi que la maintenance. Tout au long de ce processus, les développeurs ont des tâches planifiées avec différents états, des dates fixées, des estimations et des livrables.

Ces tâches planifiées peuvent être livrées dans les dates fixées comme elles peuvent **subir** des retards, des risques (absences imprévues, pannes techniques, une tâche urgente)

La société peut avoir une nouvelle offre ou une demande de maintenance pour un projet alors **il** faut avoir une idée claire sur les tâches en cours et en attente, les charges de l'équipe donc il faut avoir des indices pour la **Prise** de la décision.

3. ClickUp :

3.1.Qui utilise ClickUp ?

Idéal pour toutes les équipes qui souhaitent travailler plus intelligemment et plus efficacement sur une plateforme d'une intuitivité déconcertante.

3.2.Description de ClickUp :

ClickUp est une plateforme simple et intuitive pour gérer tous types de projets. L'objectif principal de ClickUp est d'éliminer les frustrations, les inefficacités et les déconnexions causées par l'écosystème actuel de gestion de projet. Parfaite pour ceux qui trouvent que les autres plateformes de gestion de projet sont trop simples ou trop complexes, la conception et l'expérience utilisateurs exceptionnels de ClickUp font écho à ce que Slack a fait pour la communication.[2]

3.3.Informations sur ClickUp :

Nom de la plateforme : ClickUp

Adresse web : <https://clickup.com/>

Année de Fondation : 2016

3.4.Prix de ClickUp :

ClickUp est disponible en version gratuite et propose un essai gratuit. La version payante de ClickUp est disponible à partir de 5,00 \$US/mois.

Avec une Dashboard payante aussi.

4. Description de l'existant :

L'entreprise **effectue** les travaux de suivi par la **plateforme** ClickUp

- Sur googleDrive Lister les tâches et les estimés
- Sur ClickUp
 1. Ajouter les tâches
 2. Assigner au responsable du tâche
 3. Ajouter l'estimation
 4. Ajouter la date prévue pour commencer la tâche
- Sur Excel
 1. Ajouter les tâches
 2. Assigner au responsable du tâche
 3. Ajouter l'estimation
 4. Ajouter la date prévue pour commencer la tâche
- Avec Mail de suivi fin du jour

- Changer l'état dans le fichier Excel
- Microsoft Excel pour la planification et la documentation des missions des projets.
- Google Calendar pour la planification des projets et des tâches.

5. Critique de l'existant :

- ✗ Perte de temps pour réécrire les tâches
- ✗ Perte l'information (Suppression du fichier).
- ✗ Dépendance de la mise à jour du fichier Excel
- ✗ Pas assez de rapports générés
- ✗ Les graphiques des données sont payants.

6. Solution proposée :

- ✓ Récupérer les tâches du ClickUp (Responsable, estimation, date) dans la plateforme
- ✓ Chaque mise à jour dans ClickUp sera synchronisée dans la plateforme (Responsable, estimation, date)
- ✓ Les tâches avec leurs différentes informations fourni une source pour générer des graphiques qui seront afficher dans la plateforme.
- ✓ Calendrier pour la planification des projets.
- ✓ La gestion des espaces de travail, projets et tâches.

7. Méthodologie et formalisme adoptés :

7.1.Méthodologie de conception :

Dans le cadre de notre projet, nous avons opté pour le langage de modélisation unifié UML comme une approche de conception. UML (Unified Modeling Language) est un langage standard permettant de spécifier, visualiser, construire et documenter les artefacts de systèmes logiciels. Son indépendance par rapport aux langages de programmation, aux domaines de l'application et aux processus, son caractère polyvalent et sa souplesse ont fait de lui un langage universel. [3]

7.2.Méthodologie de travail :

Avant la réalisation d'un projet informatique, il est nécessaire de choisir une méthodologie de travail et un **procédé** de suivi afin d'aboutir à la fin à un logiciel fiable. Cette

méthodologie présente un procédé qui a **pour objectif de formaliser** les étapes préliminaires du développement d'un système afin de rendre ce développement plus fidèle aux besoins du client.

7.2.1. Comparaison des approches classique et agile :

Afin de fixer une méthode particulière, nous avons effectué une étude comparative entre la méthode classique et la méthode agile. Nous présentons dans le tableau 1 une comparaison entre l'approche classique et l'approche agile.

Critères	Approche classique	Approche Agile
Planification	Prédictive	Adaptative
Cycle de vie	Phases séquentielles	Itératif et incrémentale
Changement	Résistance au changement. Processus lourds de gestion des changements acceptés.	Accueil favorable au changement intégré dans le processus
Mesure des succès	Respect des engagements initiaux en termes de couts, de budget et de niveau de qualité.	Satisfaction du client par la livraison de valeur souhaitée
Gestion des risques	Processus district et rigoureux de gestion des risques.	Gestion des risques intégrée dans le processus global
Equipe	Equipe avec ressources spécialisés dirigées par un chef de projet	Equipe responsabilisée, soutenue par le chef de projet
Qualité	Contrôle qualité à la fin de cycle de développement.	Contrôle qualité permanent au niveau du produit et du processus

Tableau 1 : Comparaison enter méthodes classiques et méthodes agiles

7.2.2. Méthode Agile :

D'après la comparaison faite à la sous-section précédente, nous **remarquons** que si le domaine du projet est maîtrisé, la méthode classique suffit. En cas où, nous ne pouvons pas tout prévoir dès le début ou si les besoins sont incomplets comme dans notre cas, il faut utiliser les méthodes itératives et incrémentales telles que les méthodes agiles. Une méthode Agile, c'est une approche incrémentale et itérative, menée dans un esprit collaboratif. **Elle** garantit une meilleure qualité de communication avec l'utilisateur, une meilleure visibilité du client sur l'avancement des travaux, un meilleur contrôle de qualité par le fait que les tests sont effectués en continu, ce qui permet de détecter rapidement les problèmes. [4]

7.2.3. Etude comparative des méthodes agiles :

Vue **le** nombre des méthodes agiles disponibles, une comparaison entre quelques-unes s'impose. Nous avons réalisé ainsi une étude comparative entre Extreme programming XP et Scrum.

Critères	Scrum	Extreme Programming XP
Planification	Au début de chaque sprint	Planning game
Estimation de l'effort	Au début de chaque sprint	Pratique XP
Changement de périmètre	Doit attendre le sprint suivant	Selon besoin
Rôle	Scrum Master Product Owner Développeur	Equipe + client
Boards/Artifacts	Product backlog Scrum board burndown / burnup	Priorisation par le client, pratiques XP
Caractéristiques principales	1. Méthode leader 2. Sprints 3. BurnUP / vélocité	1. Qualité code 2. Craftmanship 3. 3. Outillage

Tableau 2 : Tableau comparatif entre méthode Scrum et XP

7.2.4. La méthode adoptée SCRUM :

La méthodologie Scrum est l'approche agile la plus utilisée des approches agiles existantes et elle est simple à comprendre.

Le principe de la méthode agile SCRUM est de concentrer l'équipe de développement sur un ensemble de fonctionnalités à réaliser de façon itérative, dans des itérations d'une durée de deux à quatre semaines, appelées des Sprints. Chaque Sprint doit aboutir à la livraison d'un produit partiel [5]

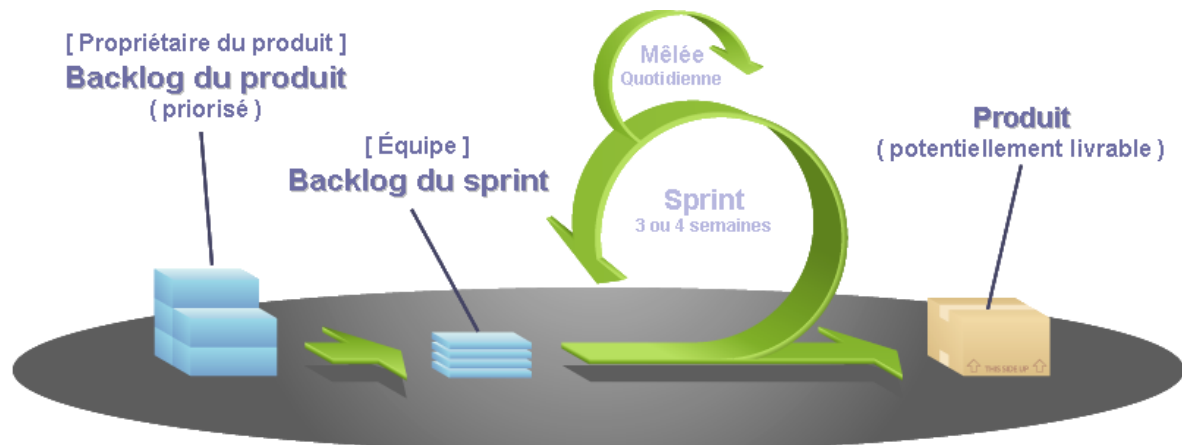


Figure 2 : Processus Scrum

Conclusion :

Tout au long de ce chapitre, nous avons présenté l'organisme d'accueil AkeoPlus Tunisie.

Par ailleurs, nous avons pu dégager le contexte général du projet et présenter le choix de la méthodologie de développement. Le chapitre suivant sera consacré à l'étude préliminaire.

Chapitre 2 : Etude Préalable

Introduction :

Dans ce chapitre, nous **présentons** d'abord l'acteur principal de l'application. Ensuite, nous déterminerons les exigences fonctionnelles et non fonctionnelles du projet. Ensuite, nous **présentons** les user stories, l'équipe Scrum, les tâches du produit et les plans de sprint. Ensuite, nous continuons à modéliser le diagramme de cas d'utilisation global du projet. Enfin, nous **concluons** ce chapitre en expliquant l'environnement matériel et logiciel et en déterminant l'architecture de l'application.

1. Spécification des besoins :

1.1. Identification des acteurs :

Un acteur représente une entité externe qui interagit avec le système par envoi ou réception des messages. En réponse à l'action d'un acteur, le système fournit un service qui répond à ses besoins.

Acteur	Rôles
Administrateur	C'est l'acteur qui s'occupera de toutes les fonctionnalités offertes par l'application.

Tableau 3 : Identification d'acteur

1.2. Besoins fonctionnels :

Les exigences fonctionnelles doivent indiquer les besoins du client. **Dans** notre cas, les **besoins de l'entreprise permettent d'établir les spécifications suivantes :**

- ✓ **S'authentification**
- ✓ Récupération les tâches du ClickUp.
- ✓ **Synchronisation de la plateforme avec ClickUp.**
- ✓ **La génération des graphiques d'informations fournies** par ClickUp.
- ✓ **Affichage des calendriers des tâches selon leurs dates.**

1.3. Besoins non fonctionnels :

Ces besoins sont les contraintes techniques **exigés** et les fonctionnalités nécessaires pour rendre le logiciel plus performant et qui se résument généralement en :

- ✓ Ergonomie : **rendre** les interfaces utilisateur conviviales et faciles à utiliser par les novices.
- ✓ Fiabilité : le fonctionnement de l'application doit être correct et ne contient pas des bugs.
- ✓ Extensibilité : l'application doit faciliter l'ajout de nouvelles fonctionnalités au moindre coût.
- ✓ Performance : il s'agit d'optimiser le temps de chargements des données depuis deux environnements différents ainsi **que l'utilisation** des bonnes pratiques du développement.
- ✓ Portabilité : doit être facile **à utiliser et accessible** par pc, tablette **et** téléphone.
- ✓ Sécurité : L'accès à l'application et les données doit être sécurisé.

2. Pilotage du projet avec Scrum :

2.1. User Stories :

Les **histoires** **{histoires ou historiques ?}** d'utilisateurs sont des brèves descriptions simples d'une fonctionnalité présentée du point de vue de la personne qui souhaite utiliser la nouvelle fonctionnalité, généralement un utilisateur ou un client du système. Ils suivent généralement un modèle simple :

En tant que <type d'utilisateur>, je veux <un objectif> pour que <une raison>.

En tant qu'administrateur, je dois m'authentifier pour accéder à la plateforme

*En tant qu'administrateur, je dois être capable de **créer d'autres** **profils***

En tant qu'administrateur, je dois être capable de consulter les statistiques

*En tant qu'administrateur, je dois être capable de **créer** et gérer des projets dans un espace **de** travail*

*En tant qu'administrateur, je dois être capable de **créer** et gérer des tâches dans un projet*

En tant qu'administrateur, je dois être capable de consulter la planification des projets dans un calendrier

En tant qu'administrateur, je dois être capable de lister les tâches du projet

*En tant qu'administrateur, je dois être capable de **créer** des graphiques et définir leur indicateur*

*En tant qu'administrateur, je dois être capable de suivre **le** progrès des projets*

En tant qu'administrateur, je dois être capable d'inviter d'autres membres

*En tant qu'administrateur, je dois être capable **de** gérer les membres de l'équipe du travail*

*En tant qu'administrateur, je dois être capable de gérer **les** espaces de travail*

En tant qu'administrateur, je dois être capable de consulter la planification des projets dans un calendrier

En tant qu'administrateur, je dois être capable de gérer la planification des projets et les tâches dans un calendrier

*En tant qu'administrateur, je dois être capable **de** positionner les graphiques selon mon choix*

*En tant qu'administrateur, je dois être capable **d'**assigner les tâches aux membres d'équipe de travail*

2.2. Equipe Scrum :

L'équipe Scrum est constituée d'un propriétaire de produit, de l'équipe de développement et d'un Scrum Master. Le modèle d'équipe de Scrum est conçu pour optimiser la flexibilité, la créativité et la productivité. Nous allons tenter de cerner notre équipe Scrum :

➤ **Product Owner (PO) :**

M. Mohamed Chaabane qui est le représentant des clients et des utilisateurs, définit les besoins, les priorités et les fonctionnalités et dirige l'activité de l'équipe de développement.

➤ **Scrum Master (SM) :**

Mme Sourour Meddeb, elle assure globalement la Supervision de l'avancement du projet et des activités de l'équipe. Cela garantira également la bonne organisation de la réunion et la méthode AGILE.

➤ **L'équipe de développement :**

Iskander Abessi et Achraf Nouria **chargés** de la réalisation de projet de la conception au développement.

2.3.Product Backlog :

Le Backlog est un artéfact très important dans Scrum. C'est l'ensemble des caractéristiques fonctionnelles ou techniques qui constituent le produit souhaité. Les éléments du Backlog de produit, appelé aussi les histoires utilisateurs, précisent la fonctionnalité désirée par le client, généralement, écrit sous la forme suivante « En tant que X, je veux Y »

Le Backlog de produit présenté dans le tableau comprend les champs suivants :

- ID : C'est un nombre unique et auto-incrémenté pour chaque histoire utilisateur.
- Thème : Pour mieux ordonner les user stories.
- User Story : comporte la description d'une fonctionnalité désirée par le client suivant la forme « En tant que . . . Je veux . . . ».
- Priorité : par rapport au client (Entreprise) représentée suivant la méthode "MoSCoW" qui est une technique possédant un objectif qui s'articule autour d'un accord entre le maître d'œuvre (MOE) et le maître d'ouvrage (MOA) sur l'importance des tâches que l'on va réaliser par rapport aux délais prévus.

MoSCoW a pour signification (le o ne représente rien c'est juste pour rendre le mot prononçable) :

- M (Must have) : doit être fait (vital).
- S (Should have) : devrait être fait dans la mesure du possible (essentiel).

- C (Could have) : pourrait être fait dans la mesure où cela n'a pas d'impact sur les autres tâches (confort).
- W (Won't have) : ne sera pas fait cette fois mais sera fait plus tard (luxé, c'est la zone d'optimisation budgétaire).
- Complexité : La complexité est évaluée entre 3 niveaux : faible, moyenne et élevée. [6]

ID	Thème	User Story	Priorité	Complexité
1	Authentification	En tant qu'administrateur, je dois m'authentifier pour accéder à la plateforme	Must	Moyenne
2	Utilisateurs plateforme	En tant qu'administrateur, je dois être capable de créer les autres profiles	Must	Moyenne
3	Dashboard	En tant qu'administrateur, je dois être capable de consulter les statistiques	Must	Moyenne
4	Gestion projets	En tant qu'administrateur, je dois être capable de créer et gérer des projets dans un espace de travail	Must	Moyenne
5	Gestion tâches	En tant qu'administrateur, je dois être capable de créer les tâches du projet	Must	Moyenne
6	Graphiques de données	En tant qu'administrateur, je dois être capable de suivre l'avancement des projets	Must	Moyenne
7	Consultation	En tant qu'administrateur, je dois être capable de lister les tâches des projets	Must	Moyenne
8	Graphiques de données	En tant qu'administrateur, je dois être capable de gérer des	Must	Élevée

		graphiques et définir leur indicateur		
9	Module Consultation	En tant qu'administrateur, je dois être capable de suivre progrès des projets	Must	Moyenne
10	Module membres l'équipe	En tant qu'administrateur, je dois être capable d'inviter d'autres membres	Must	Moyenne
11	Module membres l'équipe	En tant qu'administrateur, je dois être capable gérer les membres de l'équipe de travail	Must	Moyenne
12	Module espaces de travail	En tant qu'administrateur, je dois être capable de gérer des espaces de travail	Must	Moyenne
13	Module calendrier	En tant qu'administrateur, je dois être capable de consulter la planification des projets dans un calendrier	Must	Élevée
14	Module calendrier	En tant qu'administrateur, je dois être capable de gérer la planification des projets et les tâches dans un calendrier	Must	Moyenne
15	Graphiques de données	En tant qu'administrateur, je dois être capable positionner les graphiques selon mon choix	Must	Élevée
16	Gestion tâches	En tant qu'administrateur, je dois être capable assigner les tâches aux membres d'équipe de travail	Must	Moyenne
17	Graphiques de données	En tant qu'administrateur, je dois être capable d'avoir un fichier de données de type xls, svg ou pdf par un widget si possible	Should	Moyenne

Tableau 4 : Product Backlog

2.4. Planification Des Sprints :

Les "User stories" précédemment définis dans le Backlog du produit sont triés par ordre de priorités et de valeurs métiers. Le but étant d'implémenter en premier ce qui a le plus de valeur. Le travail sera planifié selon des sprints.

Après une réunion avec l'équipe, on a identifié cinq sprints et deux releases. Dans le tableau, nous présentons la planification des sprints

Release	Sprint	Nom du Sprint	Période
Release1	Sprint 1	Module gestion (projets / tâches / espaces de travail / consultation)	13/02/2020- 19/03/2020
	Sprint 2	Dashboard (les statistiques)	
Release2	Sprint 1	Graphiques de données (widgets)	01/05/2020- 05/07/2020
	Sprint 2	La phase de planification (calendrier)	
	Sprint 3	Module paramétrage (Authentification / Utilisateurs plateforme / Module membres l'équipe)	

Tableau 5 : Planification des sprints

3. Diagramme de cas d'utilisation global :

Nous présentons par la figure une vue globale concernant le comportement fonctionnel en général du système. Ce diagramme permet aussi de représenter les interactions entre les acteurs

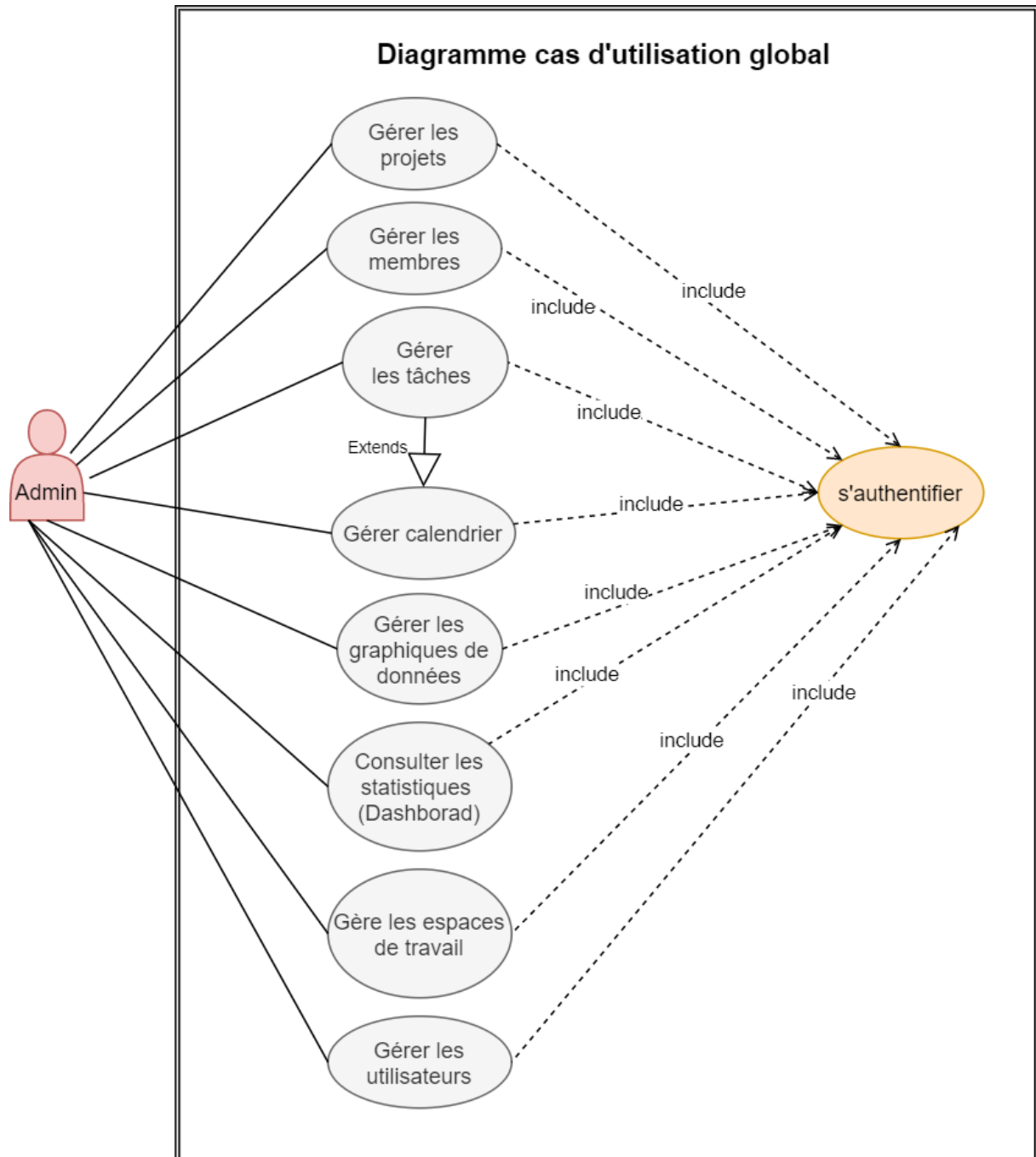


Figure 3 : Diagramme des cas d'utilisation global

4. Conception Architecturale :

4.1. Architecture logique :

L'architecture qu'on a adaptée pour la réalisation de notre solution est de type 3 tiers C'est un modèle logique d'architecture, qui consiste en une séparation en 3 couches logicielles au sein d'un même système.

La (Figure 4) illustre une représentation de l'architecture 3 -tiers avec les différents niveaux :

- ❖ La couche présentation des données :
Correspond à l'affichage, la restitution sur le poste de travail du client et le dialogue avec l'utilisateur.
- ❖ La couche traitement métier des données :
Correspond à la mise en œuvre de l'ensemble des règles de gestion et de la logique applicative.
- ❖ La couche accès aux données :

Correspond aux données qui sont destinées à être conservées sur la durée.



Figure 4 : architecture 3 tiers

4.2. Architecture de l'application :

La définition de l'architecture de l'application constitue une étape importante dans le processus de conception. Elle dépend d'un certain nombre de facteurs dont on cite les exigences en matière de performance, les perspectives, d'évolutivité, modularité et extensibilité.

La figure 5 illustre une représentation de l'architecture logicielle de l'application.

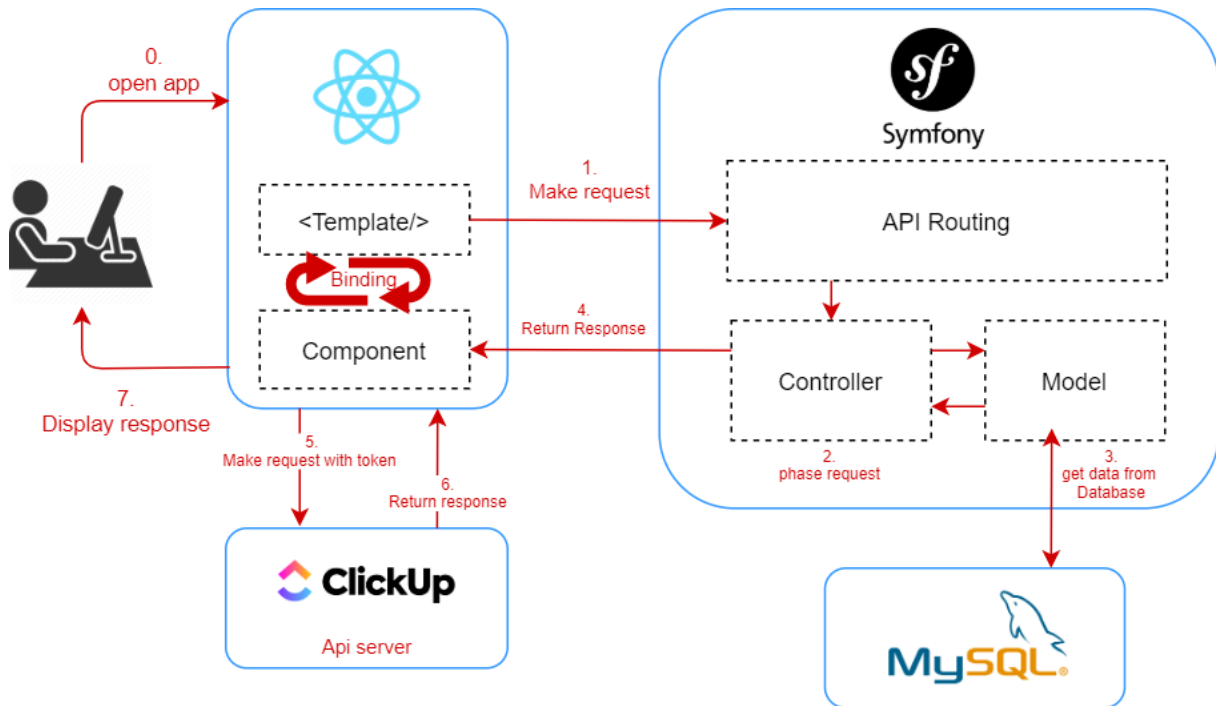


Figure 5 : Architecture de l'application

➤ Couche Présentation :

La couche de présentation assure la logique de navigation. Elle se compose d'une partie frontale en **ReactJs** et la bibliothèque **Ant-desgin**, la bibliothèque **Echarts** et la bibliothèque **Syncfusion React** pour la représentation graphique des informations.

➤ Couche métier :

C'est dans cette couche qu'on doit retrouver tous les traitements métiers de l'application. Ces traitements seront implémentés sous forme de RESTful API développé par le Framework Symfony.

➤ Couche des données :

Les données (position charts, profile admin ...) sont stockées dans une base de données MySql.

➤ Couche serveur Api (ClickUp) :

C'est le serveur de données contient toutes les informations des projets et qui offre un service API.

5. Environnement de travail :

Avant de se lancer dans l'implémentation de notre projet nous allons décrire l'environnement et les outils du travail. On va commencer par définir l'environnement matériel puis on passe à celui logiciel et enfin on présentera les différents langages et Framework utilisés.

5.1. Environnement matériel :

Nous avons développé notre application en utilisant un ordinateur portable dont les caractéristiques techniques sont décrites ci-dessous :

Ordinateur	1	2
Processeur	Core i7	Core i5
RAM	8Go	8Go
Disque Dur	1To	1To
Système d'exploitation	Windows 10 (x64)	Windows 10 (x64)

Tableau 6 : Environnement matériel

5.2. Environnement logiciel :



- **Visual Studio Code** : Visual Studio Code est un éditeur de code source léger et puissant open source disponible pour Windows, MacOS et Linux.



- **PhpStorm** : PhpStorm est un éditeur pour PHP3, HTML, CSS et JavaScript, édité par JetBrains. Il permet d'éditer du code PHP pour les versions allant de la 5.3 à la 7.4.



- **WebStorm** : WebStorm est un IDE pour les langages Web (HTML, CSS et JavaScript), développé par l'entreprise JetBrains et basé sur la plateforme IntelliJ IDEA.



- **Node.js** : Node.js est une plateforme logicielle libre en JavaScript orientée vers les applications réseau événementielles hautement concurrentes qui doivent pouvoir monter en charge. Elle utilise la

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



- **Postman** : Une plateforme API complète permet d'exécuter des requêtes, tester, déboguer, créer des serveurs factices, surveiller, exécuter des tests automatisés et documenter une API.



- **Draw.io** : Draw.io est une application de création des diagrammes en ligne compatible avec Google Drive.



- **MySQL** : MySQL est un système de gestion de bases de données relationnelles (SGBDR).



- **GitHub** est un service web d'hébergement et de gestion de développement de logiciels, utilisant le logiciel de gestion de versions Git.

5.3. Les langages :



- **HTML** (HyperText Markup Language) : HTML est le langage de balisage conçu pour représenter les pages web. C'est un langage permettant d'écrire de l'hypertexte, d'où son nom.



- **JavaScript** : JavaScript est un langage de programmation de scripts principalement employé dans les pages web interactives mais aussi pour les serveurs avec l'utilisation (par exemple) de Node.js



- **PHP** (Hypertext Preprocessor) : PHP est un langage de script serveur, et un outil puissant de création des pages web dynamique et interactives.



- **CSS** (Cascading Style Sheets) : Les feuilles de style en cascade, généralement appelées CSS, forment un langage informatique qui décrit la présentation des documents HTML.



- **Sass** (Syntactically Awesome Stylesheets) : Sass est le langage d'extension CSS de niveau professionnel. Complètement compatible avec toutes les versions de CSS.

5.4. Les Framework :



- **ReactJS** : ReactJS est une bibliothèque JavaScript libre développée par Facebook depuis 2013. Le but principal de cette bibliothèque est de faciliter la création d'application web monopage, via la création de composants dépendant d'un état et générant une page (ou portion) HTML à chaque changement d'état.



- **Symfony4** : Symfony4 est un ensemble de composants PHP ainsi qu'un framework MVC libre écrit en PHP. Il fournit des fonctionnalités modulables et adaptables qui permettent de faciliter et d'accélérer le développement d'un site web.

5.5. Les bibliothèques :



- **Syncfusion** : Syncfusion React (Essential JS 2) est une bibliothèque de composants d'interface utilisateur moderne qui a été conçue à partir de zéro pour être légère, réactive, modulaire et tactile. Il inclut également une prise en charge complète des frameworks Angular, Vue, ASP.NET MVC et ASP.NET Core.



- **ECharts** : ECharts est une bibliothèque de graphiques et de visualisation gratuite et puissante offrant un moyen simple d'ajouter des graphiques intuitifs, interactifs et hautement personnalisables à vos produits commerciaux. Il est écrit en JavaScript pur et basé sur zrender, qui est une toute nouvelle bibliothèque de canevas légère.



- **Ant Design** : Ant Design est un design system web basé sur la bibliothèque React JS qui permet la création d'application web. Utilisé comme une bibliothèque de composants sur des projets React, Ant Design se centre sur le design de l'environnement graphique dans lequel évolue l'utilisateur (UI).

Conclusion :

La spécification des besoins nous a permis d'avoir une vision plus claire du sujet et une compréhension plus profonde des tâches à réaliser. Nous avons essayé de décrire tout au long de ce chapitre brièvement les fonctionnalités de base du système, les choix techniques, les outils de développement et de conception. Nous avons détaillé la première étape de la méthodologie que nous avons choisi à savoir l'identification de l'équipe de travail et la réalisation de Backlog du produit et des sprints.

Dans le chapitre suivant, nous commençons à développer la première release.

Chapitre 3 : Release 1

Introduction

Ce chapitre fait l'objet d'une présentation du premier release du projet qui est la réalisation du module gestion.

Il est composé de deux sprints à savoir le sprint 1 et sprint 2. L'étude de chaque sprint couvre l'analyse, la conception, la réalisation et les tests fonctionnels.

1. Sprint 1 Module Gestion :

Le sprint est le cœur de Scrum. Il s'agit d'un bloc de temps durant lequel un incrément du produit sera réalisé

Ce sprint a pour but de développer la première partie de notre projet qui est :

1.1.Backlog du sprint :

Le backlog du sprint 1 contient une liste des tâches identifiées par l'équipe Scrum qui devront être réalisées avant la fin de sprint.

ID	User Story	ID Tâche	Tâche	Estimation (Jour)
1	Gestion des espaces de travail	1.1	Développement d'une interface de gestion des espaces de travail	5J
		1.2	Utilisation les API ClickUp pour générer les espaces (Ajouter, modifier, supprimer)	
		1.3	Développement d'un pop-up pour l'ajout d'une nouvelle espace de travail	
		1.4	Interface consultation espace de travail	
		1.5	Affichage des espaces de travail	

2	Gestion des projets	2.1	Développement d'une interface de gestion des projets	9J
		2.2	Utilisation les API ClickUp pour générer les projets (Ajouter, modifier, supprimer)	
		2.3	Développement d'un pop-up pour l'ajout d'un nouveau projet	
		2.4	Interface consultation du projet	
		2.5	Affichage des projets	
3	Gestion des tâches	3.1	Développement d'une interface de gestion des tâches	10J
		3.2	Utilisation les API ClickUp pour générer les tâches (Ajouter, modifier, supprimer)	
		3.3	Développement d'un pop-up pour l'ajout d'une nouvelle tâche	
		3.4	Interface consultation taches	
		3.5	Affichage des taches	

Tableau 7 : Backlog de sprint 1 release 1

1.2.Analyse :

Dans cette partie nous présentons la phase d'analyse par un diagramme des cas d'utilisation puis la description textuelle des plus importants d'entre eux.

1.2.1. Diagramme des cas d'utilisation « Sprint 1 » :

Les diagrammes de cas d'utilisation modélisent le comportement d'un système et permettent de capturer les exigences du système. Ils identifient également les interactions entre le système et ses acteurs

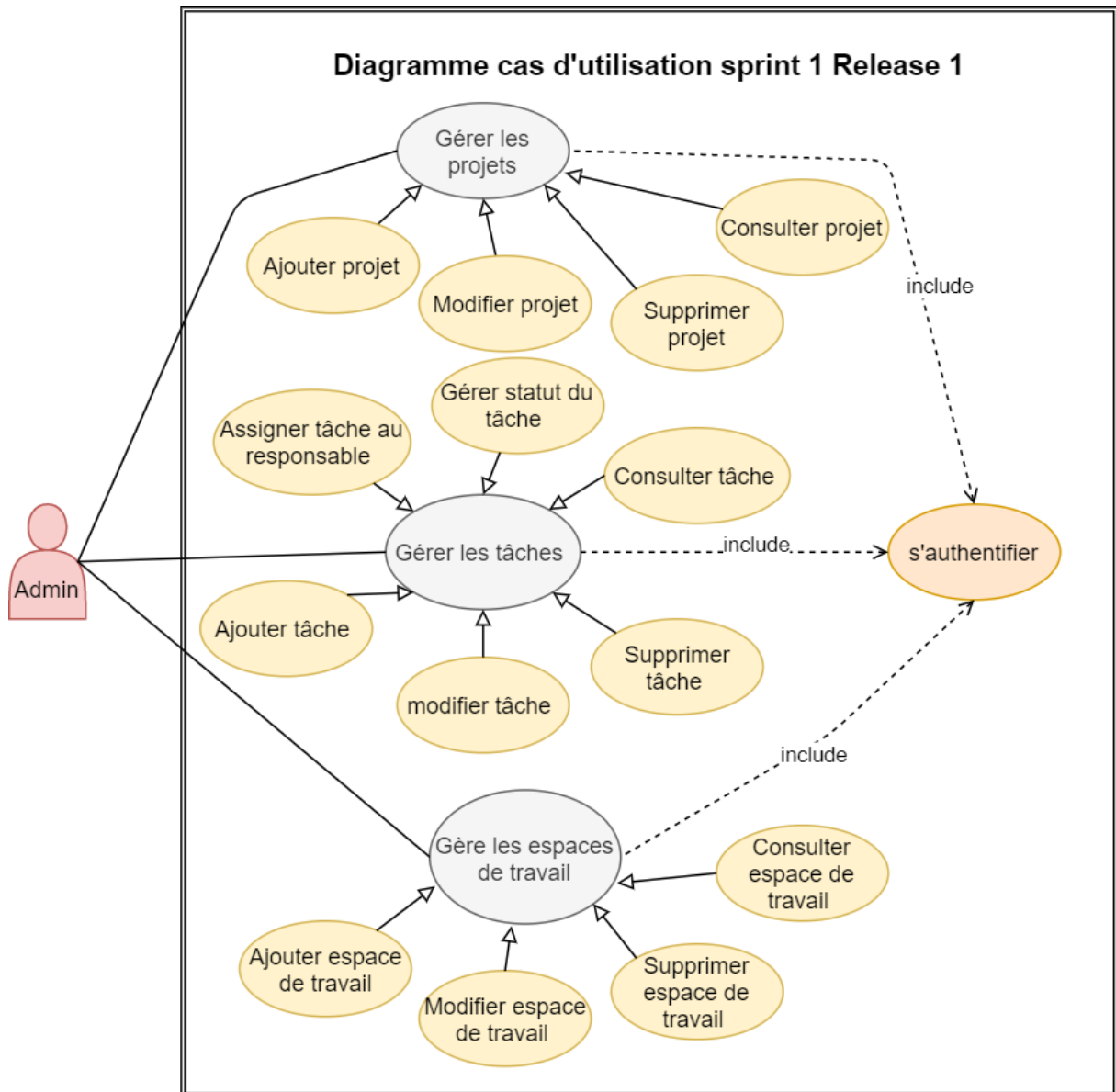


Figure 6 : Diagramme de cas d'utilisation Sprint 1 Release 1

1.2.2. Description Textuelle :

L'objectif de cette activité est de décrire textuellement, pour chaque Sprint, les scénarios des cas d'utilisation. Il faut indiquer comment ce scénario démarre, comment il se termine et les interactions de l'utilisateur avec l'application.

➤ Description textuelle du cas d'utilisation « Ajouter projet » :

Cas d'utilisation	Ajouter projet
Acteurs	Administrateur
Pré-condition	Administrateur connecté
Scénario nominal	<ol style="list-style-type: none"> 1. L'administrateur s'authentifie 2. Le système affiche la plateforme 3. L'administrateur clique sur gestion projet puis clique sur ajouter projet 4. Le système affiche l'interface d'ajout 5. L'administrateur remplir les champs nécessaires et valide 6. Le système envoie les informations au ClickUp

➤ Description textuelle du cas d'utilisation « Consulter tâche » :

Cas d'utilisation	Consulter tâche
Acteurs	Administrateur
Pré-condition	Administrateur connecté
Scénario nominal	<ol style="list-style-type: none"> 1. L'administrateur s'authentifie 2. Le système affiche la plateforme 3. L'administrateur clique sur gestion projet puis clique sur projet et choisir la tâche à consulter 4. Le système affiche toutes les informations concernant la tâche choisie avec la possibilité de filtrer et grouper les taches selon des critères

➤ Description textuelle du cas d'utilisation « Supprimer espace de travail » :

Cas d'utilisation	Supprimer espace de travail
Acteurs	Administrateur
Pré-condition	Administrateur connecté
Scénario nominal	<ol style="list-style-type: none"> 1. L'administrateur s'authentifie 2. Le système affiche la plateforme 3. L'administrateur clique sur gestion espace de travail puis sur supprimer espace

	<ol style="list-style-type: none">4. Le système affiche un pop-up de confirmation5. L'administrateur confirme la suppression
--	-----------------------------------------------------------------------------------------------------------------------------------------------------

5.1.Conception :

Dans cette section, nous présentons quelques diagrammes de séquence détaillés.

5.1.1. Diagrammes de séquence :

Un diagramme de séquence détaillée permet une représentation détaillée des interactions entre les objets métiers de notre système selon un ordre chronologique. Nous présentons dans ce qui suit les diagrammes des séquences détaillés des cas d'utilisation du premier sprint.

- Diagramme de séquence « Ajouter espace de travail »

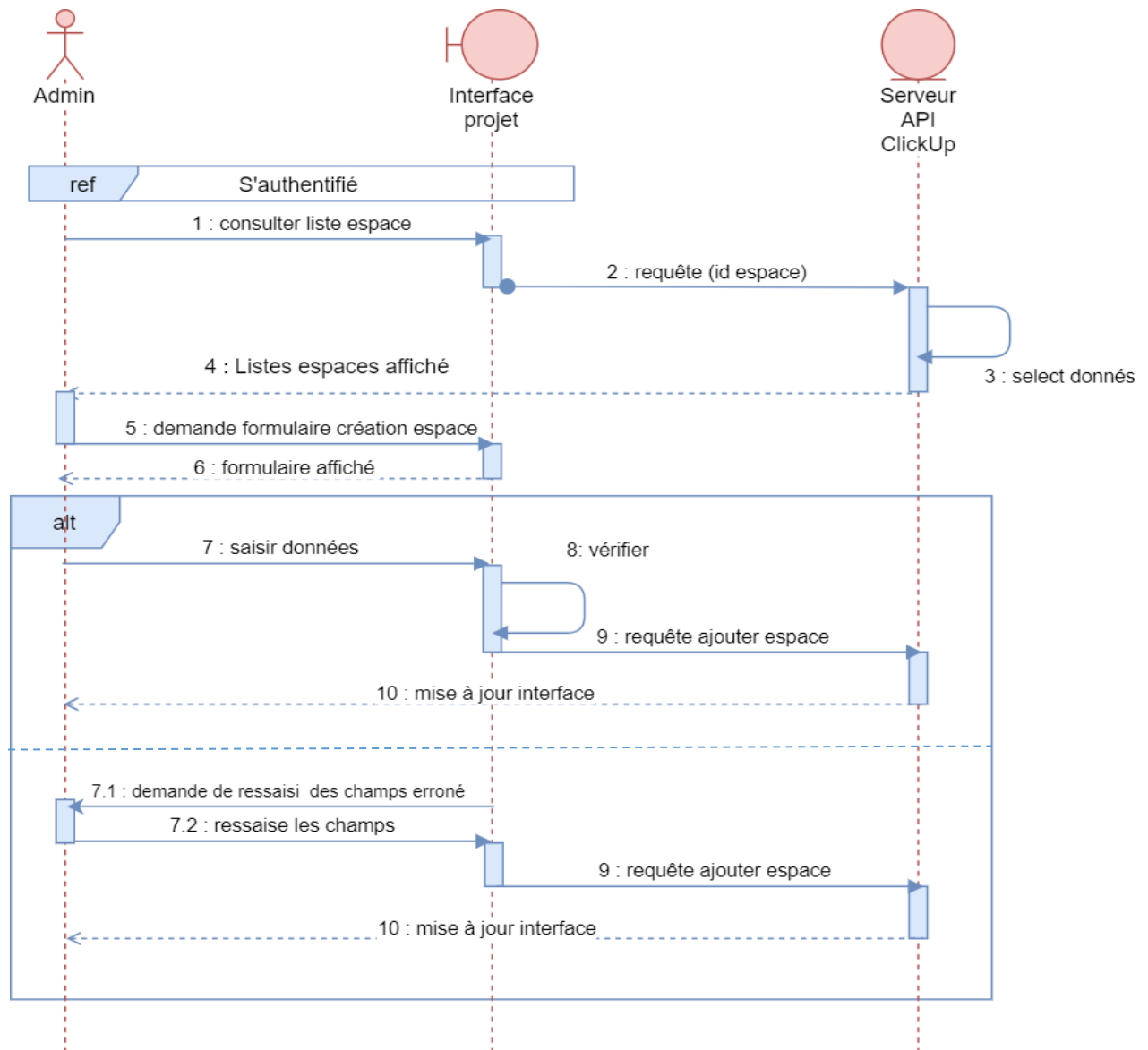
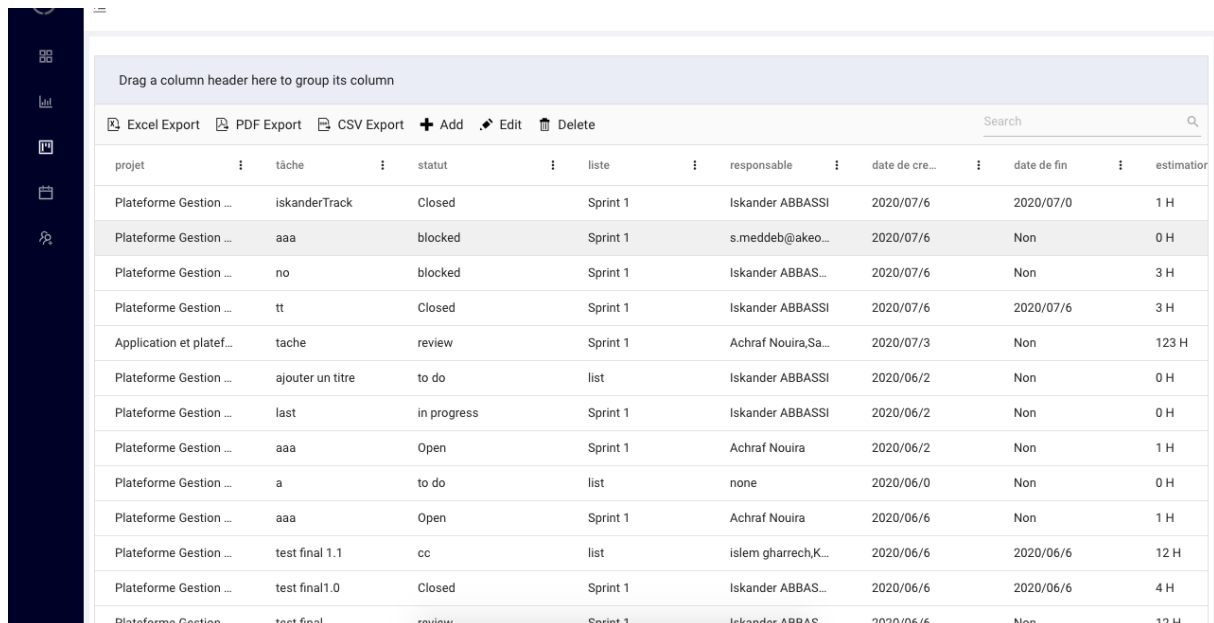


Figure 7 : Diagramme de séquence « Ajouter espace de travail »

5.2.Réalisation :

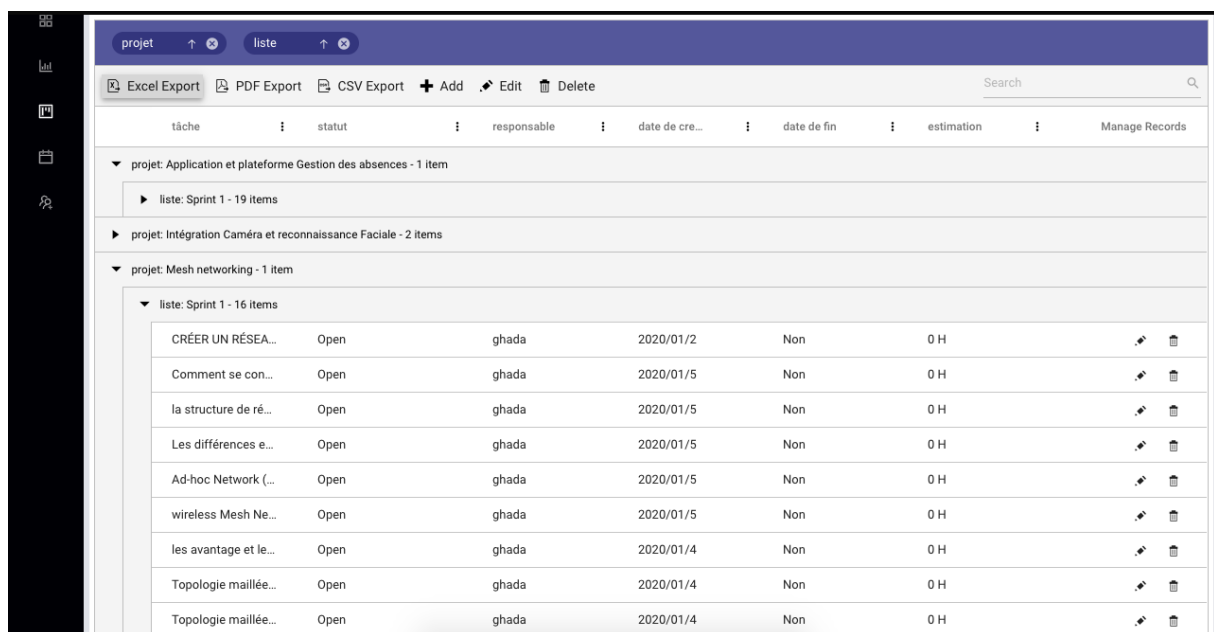
Cette partie est consacrée à l'exposition du travail achevé à travers des captures d'écrans de différentes interfaces développées pendant ce sprint.

Chapitre 3 : Release 1



Drag a column header here to group its column							
Excel Export	PDF Export	CSV Export	+ Add	Edit	Delete	Search	
projet	tâche	statut	liste	responsable	date de cre...	date de fin	estimation
Plateforme Gestion ...	iskanderTrack	Closed	Sprint 1	Iskander ABBASSI	2020/07/6	2020/07/0	1 H
Plateforme Gestion ...	aaa	blocked	Sprint 1	s.meddeb@akeo...	2020/07/6	Non	0 H
Plateforme Gestion ...	no	blocked	Sprint 1	Iskander ABBAS...	2020/07/6	Non	3 H
Plateforme Gestion ...	tt	Closed	Sprint 1	Iskander ABBASSI	2020/07/6	2020/07/6	3 H
Application et platef...	tache	review	Sprint 1	Achraf Nouira,Sa...	2020/07/3	Non	123 H
Plateforme Gestion ...	ajouter un titre	to do	list	Iskander ABBASSI	2020/06/2	Non	0 H
Plateforme Gestion ...	last	in progress	Sprint 1	Iskander ABBASSI	2020/06/2	Non	0 H
Plateforme Gestion ...	aaa	Open	Sprint 1	Achraf Nouira	2020/06/2	Non	1 H
Plateforme Gestion ...	a	to do	list	none	2020/06/0	Non	0 H
Plateforme Gestion ...	aaa	Open	Sprint 1	Achraf Nouira	2020/06/6	Non	1 H
Plateforme Gestion ...	test final 1.1	cc	list	islem gharrech,K...	2020/06/6	2020/06/6	12 H
Plateforme Gestion ...	test final1.0	Closed	Sprint 1	Iskander ABBAS...	2020/06/6	2020/06/6	4 H
Plateforme Gestion ...	test final	review	Sprint 1	Iskander ABBAS...	2020/06/6	Non	12 H

Figure 8 : Interface gestion tâches



tâche	statut	responsable	date de cre...	date de fin	estimation	Manage Records
▼ projet: Application et plateforme Gestion des absences - 1 item						
▶ liste: Sprint 1 - 19 items						
▶ projet: Intégration Caméra et reconnaissance Faciale - 2 items						
▼ projet: Mesh networking - 1 item						
▼ liste: Sprint 1 - 16 items						
CRÉER UN RÉSEA...	Open	ghada	2020/01/2	Non	0 H	Edit Delete
Comment se con...	Open	ghada	2020/01/5	Non	0 H	Edit Delete
la structure de ré...	Open	ghada	2020/01/5	Non	0 H	Edit Delete
Les différences e...	Open	ghada	2020/01/5	Non	0 H	Edit Delete
Ad-hoc Network (...)	Open	ghada	2020/01/5	Non	0 H	Edit Delete
wireless Mesh Ne...	Open	ghada	2020/01/5	Non	0 H	Edit Delete
les avantage et le...	Open	ghada	2020/01/4	Non	0 H	Edit Delete
Topologie maillée...	Open	ghada	2020/01/4	Non	0 H	Edit Delete
Topologie maillée...	Open	ghada	2020/01/4	Non	0 H	Edit Delete

Figure 9 : Interface gestion projets

5.3.Test :

Le test d'un produit logiciel est un processus consistant qui vise à garantir le bon fonctionnement du système à travers une comparaison des comportements attendus et des résultats obtenus.

Avant la fin de chaque Sprint nous avons testé les fonctionnalités du module. Ensuite nous avons validé toutes les fonctionnalités avec le Product Owner. Pour ce fait, nous avons élaboré dans le tableau un ensemble de cas de scénario de tests relatif au sprint 1.

Cas de test	Démarche	Comportement attendu	Résultat
Test de fonctionnalité de la gestion des espaces de travail	Ajouter, modifier, supprimer et consulter	CRUD fonctionnels	Conforme
Test fonctionnalité de la gestion des projets	Ajouter, modifier, supprimer et consulter	CRUD fonctionnels	Conforme
Test fonctionnalité de la gestion des tâches	Ajouter, modifier, supprimer et consulter	CRUD fonctionnels	Conforme

Tableau 8 : Tableau de test de sprint 1 release 1

6. Sprint 2 Dashboard (les statistiques) :

Ce sprint a pour but de développer le module Dashboard qui une représentation visuelle des informations les plus importantes, regroupées sur un écran afin de pouvoir être facilement comprises.

6.1.Backlog du sprint :

Ce tableau décrit les tâches identifiées pour ce sprint.

ID	User Story	ID Tâche	Tâche	Estimation (Jour)
1	Dashboard	1.1	Développement d'un Dashboard	7J
		1.2	Utilisation API ClickUp pour la récupération des données	
		1.3	Filtrage les données récupérer	
		1.4	Insertion des panneaux de contrôle pour afficher statistiques	

Tableau 9 : Backlog sprint 2 release 1

6.2.Analyse :

Dans cette partie, nous présentons les diagrammes de cas d'utilisation et la description textuelle des plus importants d'entre eux.

6.2.1. Diagramme du cas d'utilisation « Sprint 2 » :

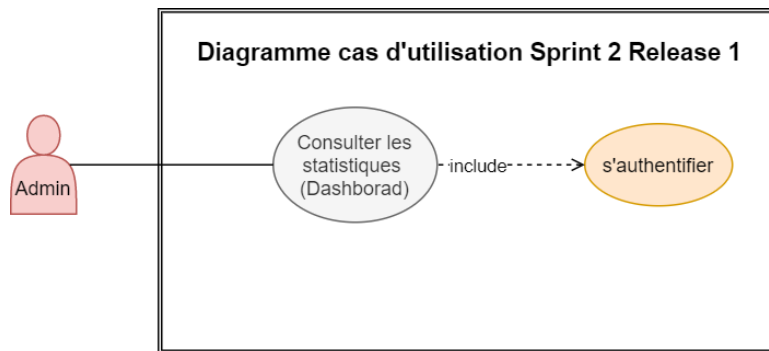


Figure 10 : Diagramme de cas d'utilisation sprint 2 release 1

6.2.2. Description Textuelle :

- Description textuelle du cas d'utilisation « Consulter Dashboard (les statistiques) » :

Cas d'utilisation	Consulter Dashboard
Acteurs	Administrateur
Pré-condition	Administrateur connecté
Description	Le Dashboard aide l'administrateur à suivre les projets, l'état des tâches, les membres d'équipe et encore le nombre des projets et tâches en cours

6.3.Conception :

- Diagramme de collaboration de cas d'utilisation « Consulter Dashboard » :

Un diagramme de collaboration est un diagramme d'interactions est une représentation simplifiée d'un digramme de séquence se concentrant sur les échanges de messages entre les objets.

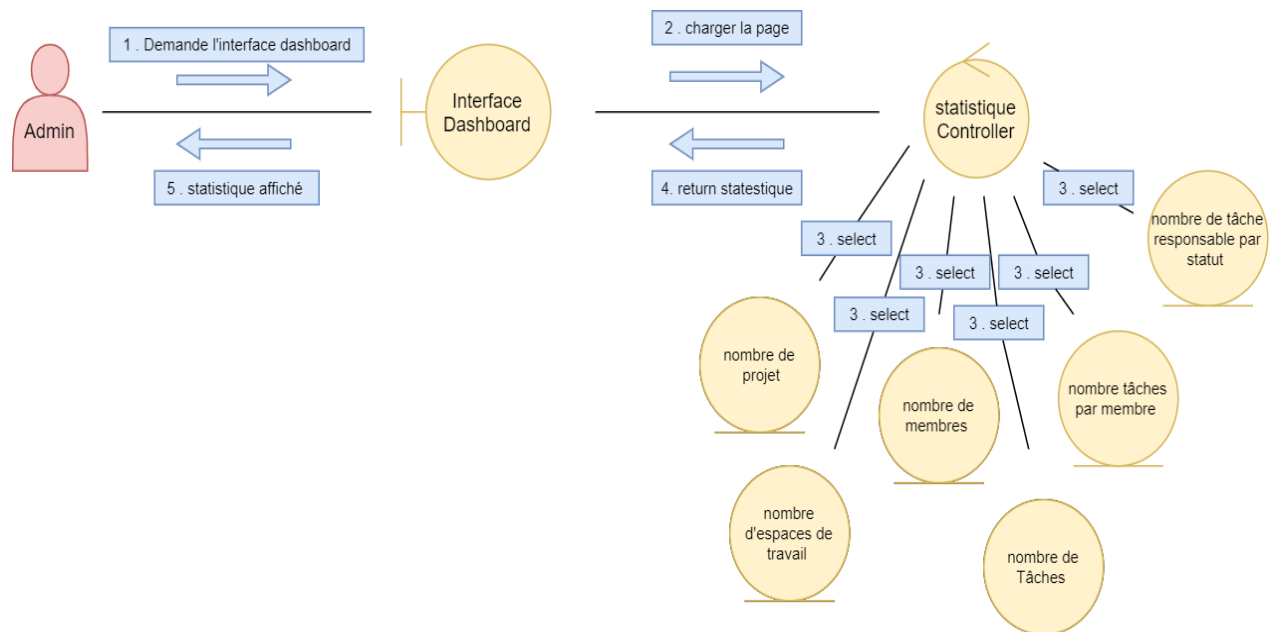


Figure 11 : Diagramme de collaboration consulter Dashboard

6.4.Réalisation :

Afin de montrer les résultats de ce sprint, nous allons présenter quelques captures d'écran.

➤ Interface Dashboard :

Cette interface est une représentation visuelle des informations les plus important regroupées sur un écran afin de pouvoir être facilement comprises.

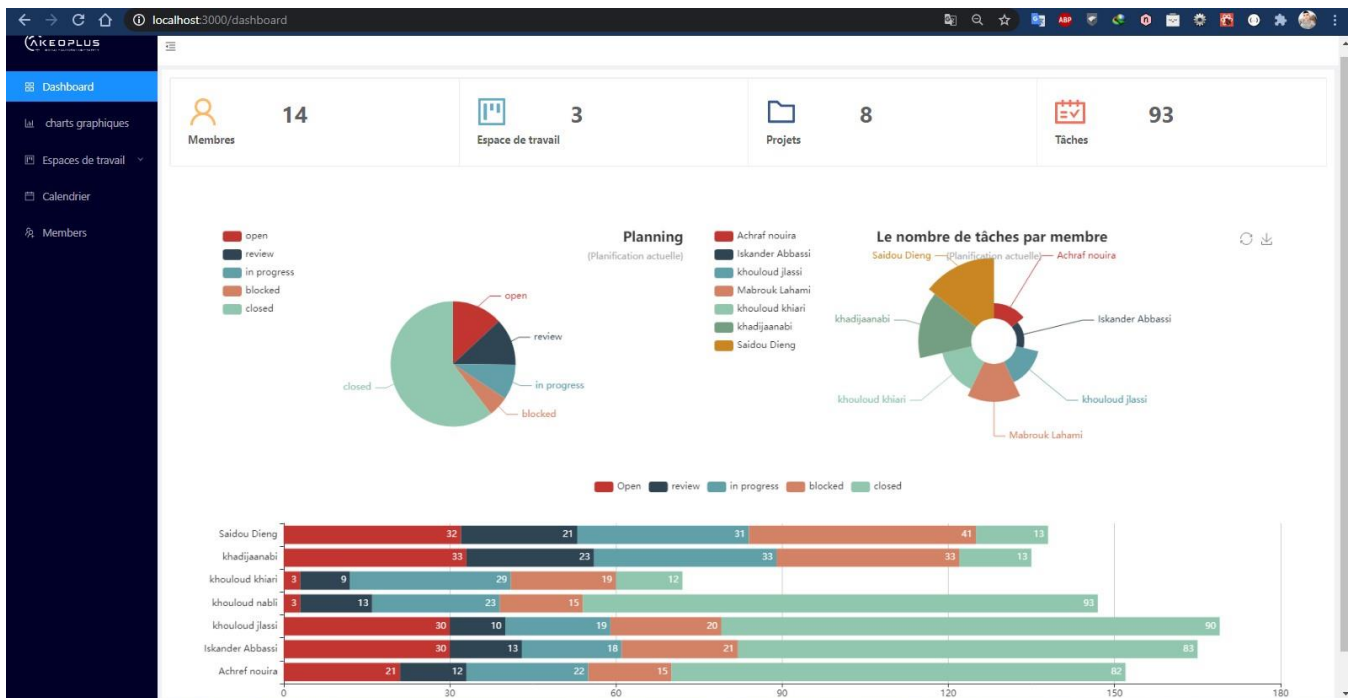


Figure 12 : Interface Dashboard

6.5. Test :

Le tableau présente un ensemble de cas de tests relatif au sprint 2.

Cas de test	Démarche	Comportement attendu	Résultat
Test de statistiques correctes	Consulter Dashboard	Dashboard	Conforme aux besoins

Tableau 10 : Tableau de test sprint 2 release 1

Conclusion :

Au cours de ce chapitre, nous avons présenté le premier release. Pour ce faire, nous avons passé par l'analyse, la conception, la réalisation et les tests pour chaque sprint. Ce release permet la gestion (projets / tâches /espaces de travail) et facilite la vision avec un Dashboard conviviale. Dans le chapitre suivant nous entamons le deuxième release.

Chapitre 4 : Release 2

Introduction

Ce chapitre a pour objectif d'exposer le travail achevé durant la deuxième itération de notre Projet. Dans cette partie, nous nous intéressons à des tâches principales étant les chartes graphiques qui permettent de la visualisation graphique des données et le calendrier des planifications.

Ce Release est composé de trois sprints une pour les chartes graphiques et la deuxième pour la phase de planification et la dernière pour l'authentification, utilisateurs plateforme et la gestion des membres d'équipe. L'étude de sprint couvre l'analyse, la conception, la réalisation et les tests fonctionnels.

1. Sprint 1 Graphiques de données (widgets) :

Ce sprint permet à l'administrateur de générer des graphiques (widgets) selon le besoin pour représenter les données de manière facile à comprendre et aussi permettent en un seul coup d'œil d'en saisir la tendance générale.

1.1.Backlog du sprint :

Le Backlog du sprint contient une liste des tâches identifiées par l'équipe Scrum qui devront être réalisées avant la fin de sprint.

ID	User Story	ID Tâche	Tâche	Estimation (Jour)
1	Gérer les graphiques de données	1.1	Interface pour ajouter les panneaux dynamiques	
		1.2	Utilisation API ClickUp pour la récupération des données	
		1.3	Intégrer widget tâche par membre d'équipe	
		1.4	Intégrer widget charge de travail par statut	
		1.5	Intégrer widget KPI	

Tableau 11 : Backlog sprint 1 release 2

1.2.Analyse :

Dans cette partie nous présentons la phase d'analyse par un diagramme des cas d'utilisation puis la description textuelle des plus importants d'entre eux.

1.2.1. Diagramme des cas d'utilisation « Sprint 1 » :

Les diagrammes de cas d'utilisation modélisent le comportement d'un système et permettent de capturer les exigences du système. Ils identifient également les interactions entre le système et l'acteur

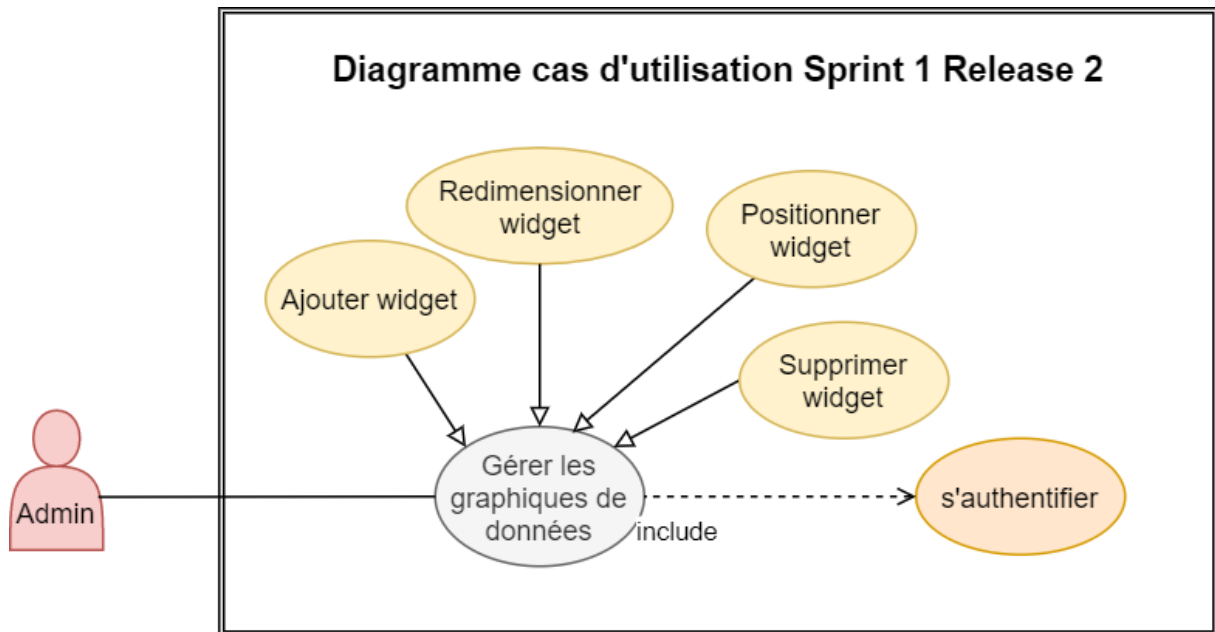


Figure 13 : Diagramme de cas d'utilisation sprint 1 release 2

1.2.2. Description Textuelle :

L'objectif de cette activité est de décrire textuellement, pour chaque Sprint, les scénarios des cas d'utilisation. Il faut indiquer comment ce scénario démarre, comment il se termine et les interactions de l'utilisateur avec l'application.

1.3.Conception :

1.3.1. Diagramme de classe :

La figure représente le diagramme de classe utilisé pour le développement du Sprint 1

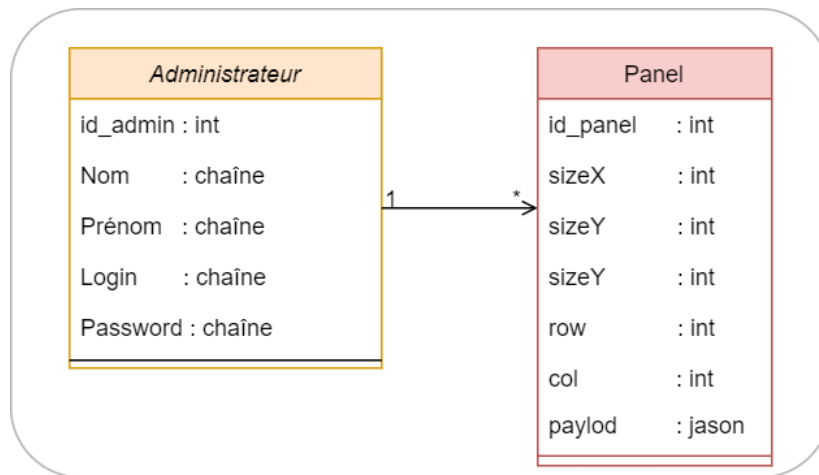


Figure 14 : Diagramme de classes sprint 1 release 2

1.3.2. Diagrammes de séquence :

Un diagramme de séquence détaillée permet une représentation détaillée des interactions entre les objets métiers de notre système selon un ordre chronologique. Nous présentons dans ce qui suit les diagrammes des séquences détaillés des cas d'utilisation du premier sprint.

1.4.Réalisation :

Cette partie est consacrée à l'exposition du travail achevé à travers des captures d'écrans de différentes interfaces développées pendant ce sprint.

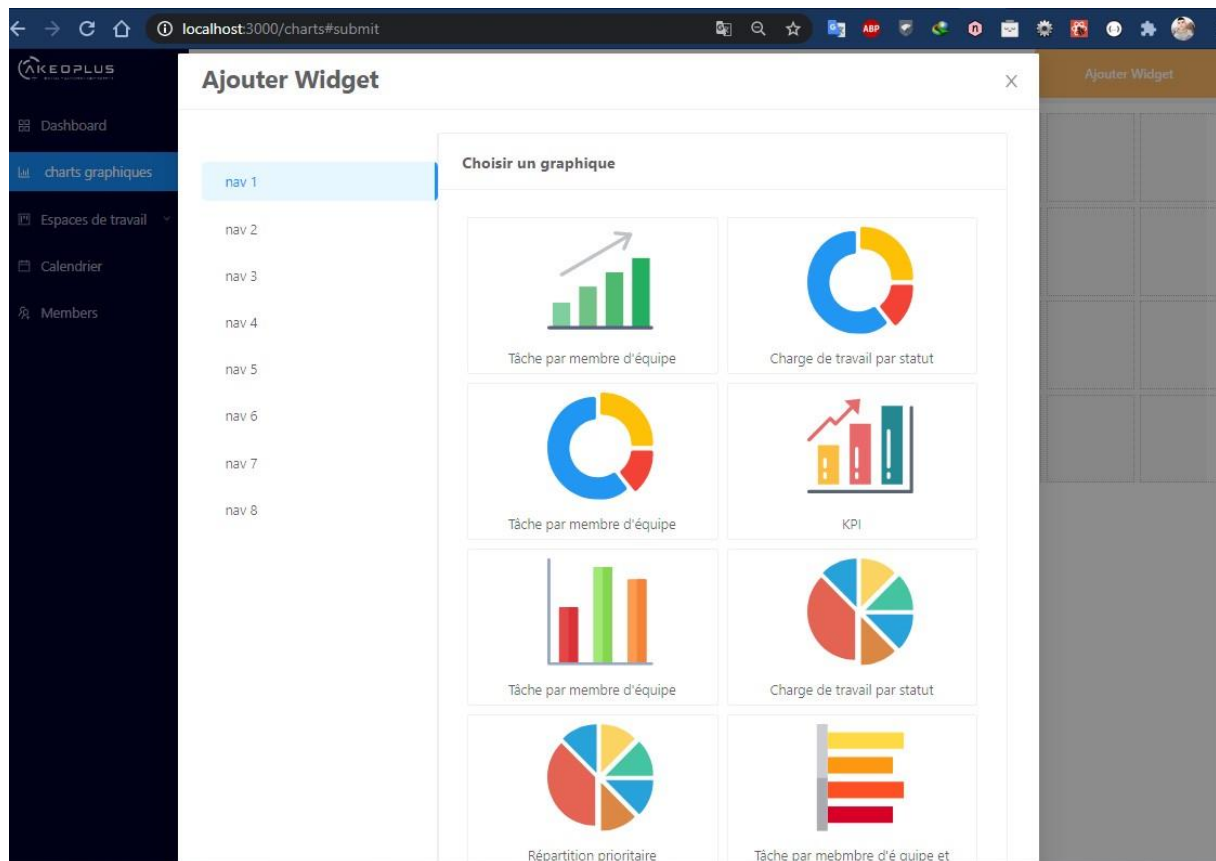


Figure 15 : Interface ajouter widget

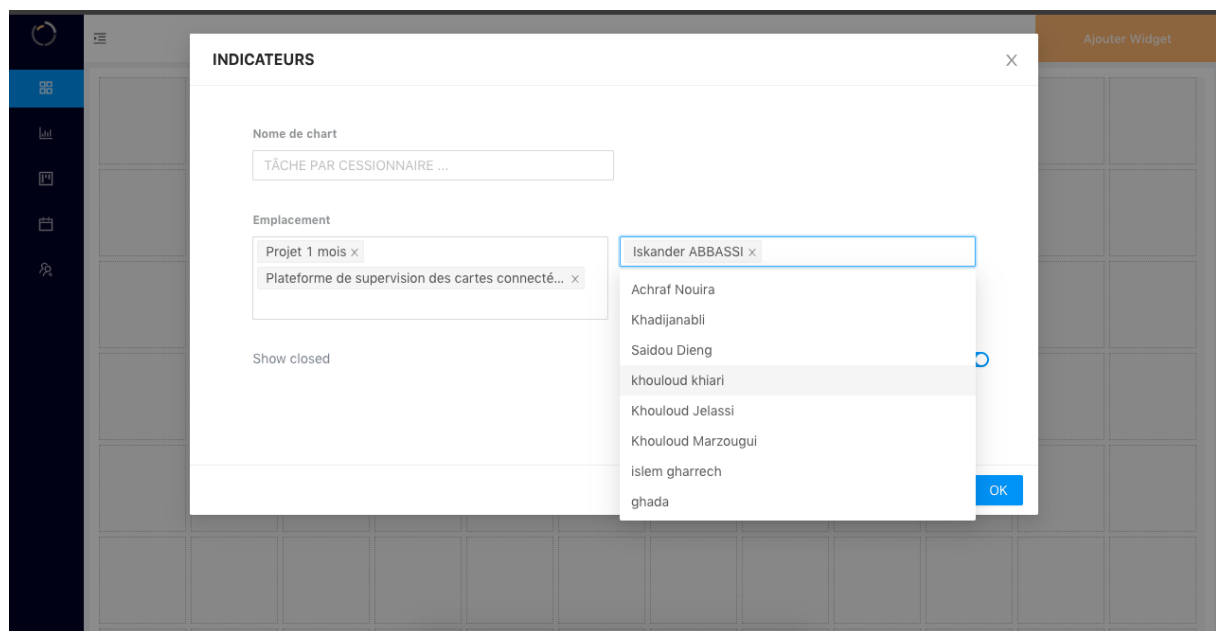


Figure 16 : interface indicateurs widget

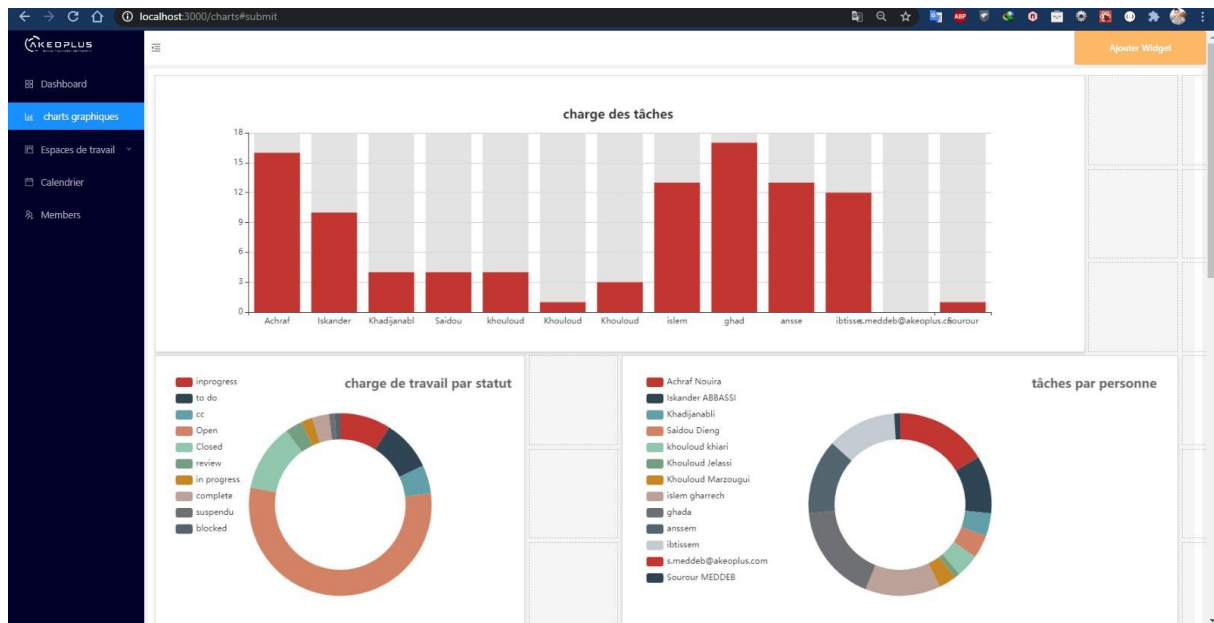


Figure 17 : Interface graphiques de données

1.5.Test :

Cas de test	Démarche	Comportement attendu	Résultat
Test d'ajout du widget	Ajouter widget	Widget ajouté	Conforme
Test enregistrement Positions des widgets	Enregistrer position widget pour la prochaine login	Même widgets trouvés pour un nouveau login	Conforme
Test de suppression des widget	Supprimer widget	Widget supprimé	Conforme
Test de redémentions des widgets	Redimensionner widget	Widget redimensionné	Conforme

2. Sprint 2 La phase de planification (calendrier) :

Ce sprint permet à l'administrateur de consulter le planning des projets et les taches dans une calendrier avec la possibilité de gérer les tâches directement.

Les tâches sont organisées selon la date début et la date fin et chaque tâche est caractérisée par une couleur qui détermine la statue.

L'affichage du planning peut être en jour, semaine ou mois selon le choix.

2.1.Backlog du sprint :

ID	User Story	ID Tâche	Tâche	Estimation (Jour)
1	Gestion des planifications (calendrier)	1.1	Interface de planning projets	9J
		1.2	Consommer les données par les API ClickUp	
		1.3	Filtrage des données	
		1.4	Affichage des données organisées par date début, date fin et durée tâche	
2	Gestion tâche (calendrier)	2.1	Interface d'ajout d'une tâche	5J
		2.2	Interface modification tâche	
		2.3	Développement d'un bouton pour supprimer une tache avec une pop-up de confirmation	
		2.4	La transmission des données au serveur ClickUp par les API ClickUp	

Tableau 12 : Backlog sprint 2 release 2

2.2.Analyse :

Dans cette partie, nous présentons les diagrammes de cas d'utilisation et la description textuelle des plus importants d'entre eux.

2.2.1. Diagramme de cas d'utilisation « Sprint 2 » :

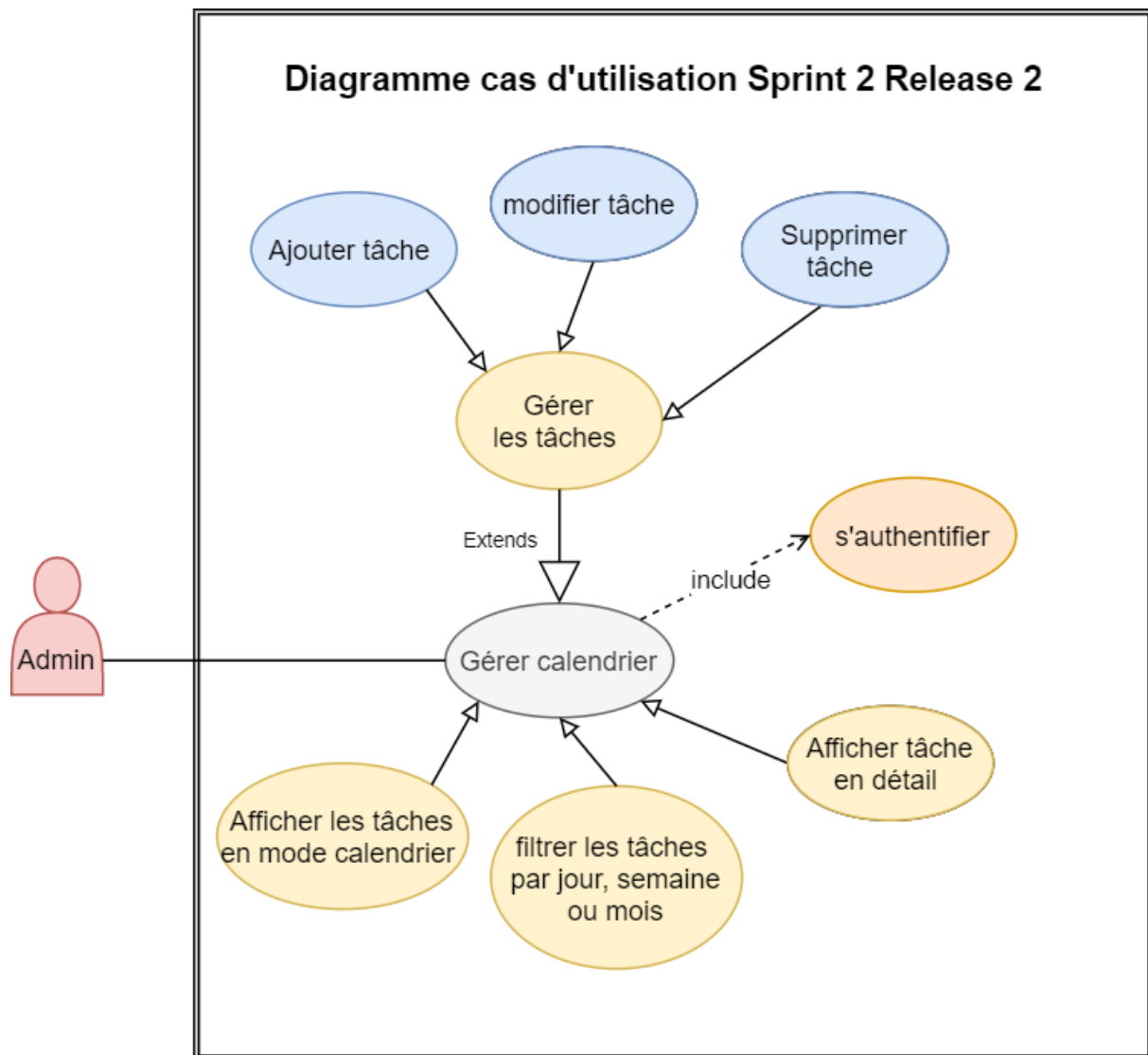


Figure 18 : Diagramme de cas d'utilisation sprint 2 release 2

2.2.2. Description textuelle :

➤ Description textuelle du cas d'utilisation « Gérer les tâches » :

Cas d'utilisation	Gérer les tâches
Acteurs	Administrateur
Pré-condition	Administrateur connecté
Description	<p>La gestion des tâches dans calendrier consiste à :</p> <ul style="list-style-type: none"> ✓ Créer une nouvelle tâche ✓ Modifier la date, responsable, une tâche ✓ Supprimer une tâche ✓ Consulter les tâches

	✓ Possibilité de glisser déposer une tâche pour changer sa date planifier
--	---------------------------------------------------------------------------

➤ Description textuelle du cas d'utilisation « Gérer calendrier » :

Cas d'utilisation	Gérer calendrier
Acteurs	Administrateur
Pré-condition	Administrateur connecté
Description	<p>La gestion du calendrier consiste à :</p> <ul style="list-style-type: none"> ✓ Afficher les tâches organisées selon ses dates ✓ Afficher les tâches par jour, par semaine ou par mois ✓ Chaque tâche est colorée par statut

2.3.Conception :

➤ Diagramme de collaboration du cas d'utilisation « Gérer calendrier » :

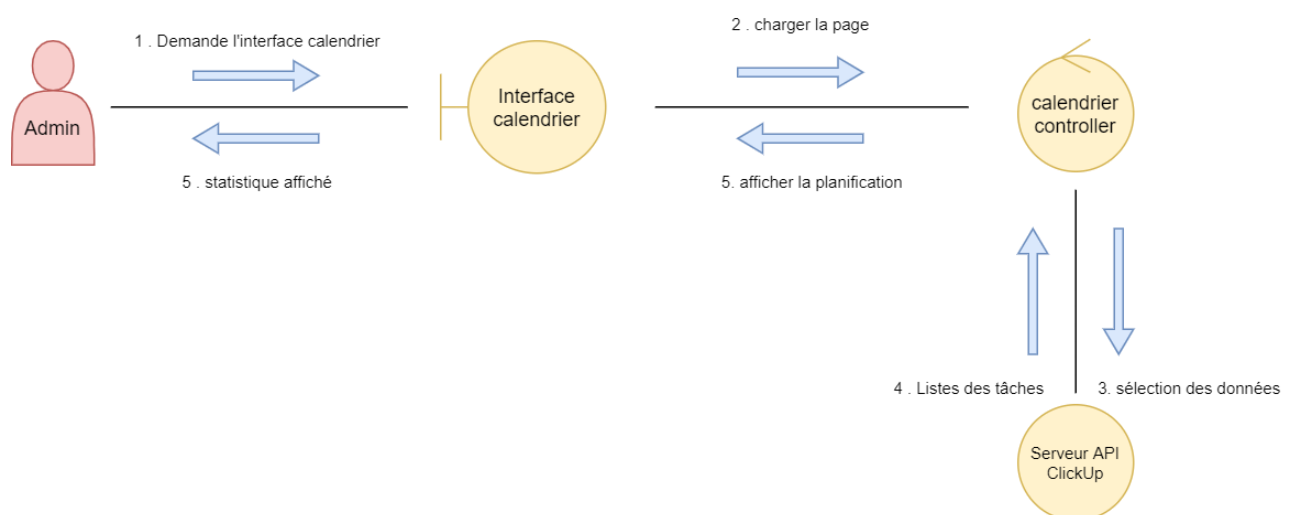


Figure 19 : Diagramme de collaboration du cas d'utilisation « Gérer calendrier »

2.4.Réalisation :

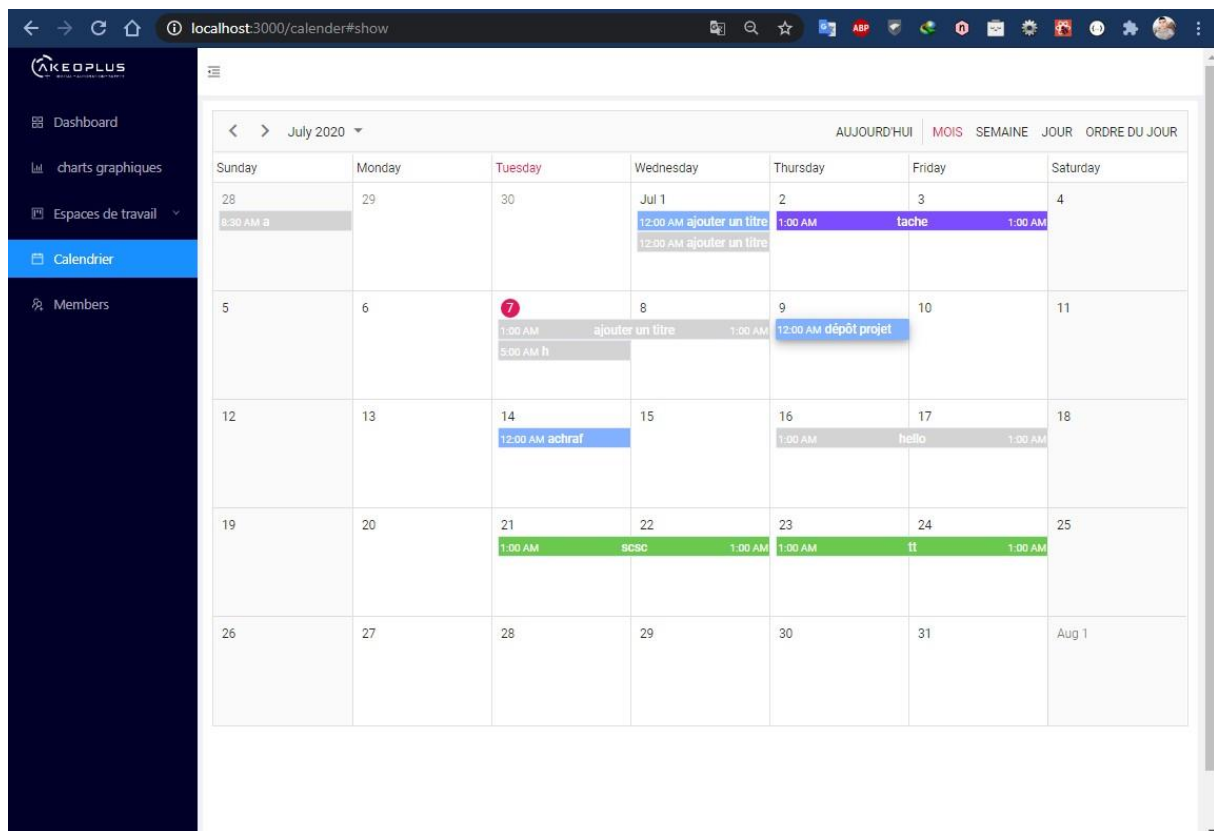


Figure 20 : Interface calendrier

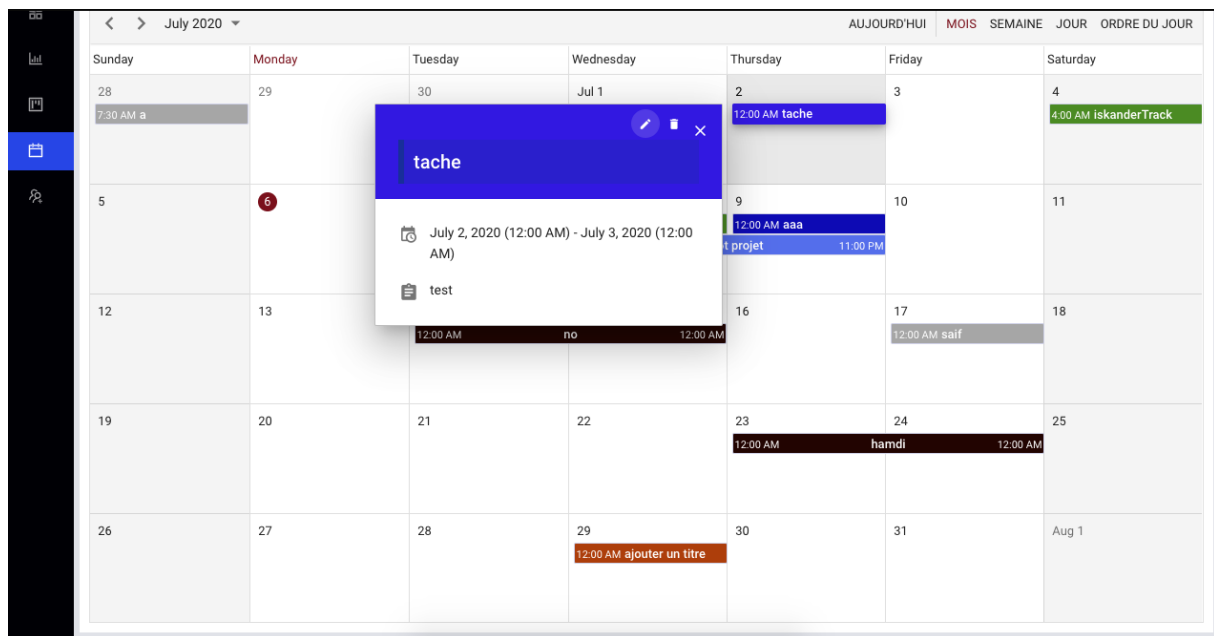


Figure 21 : Interface de gestion tâche (calendrier)

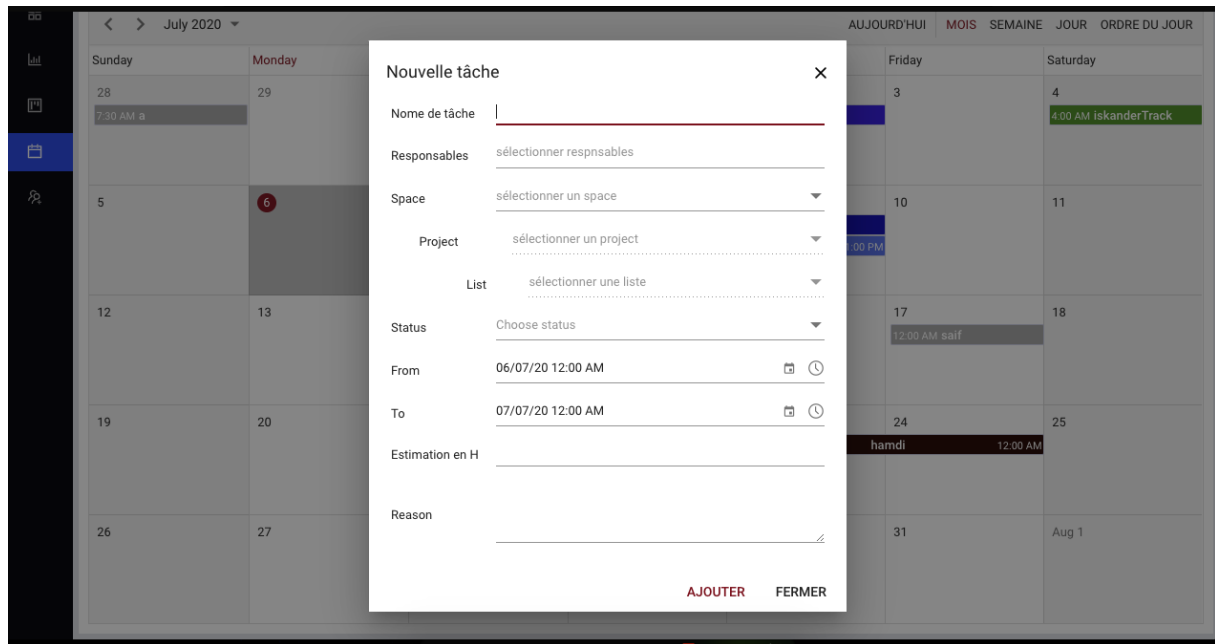


Figure 22 : Interface ajouter une nouvelle tâche (calendrier)

2.5. Test :

Cas de test	Démarche	Comportement attendu	Résultat
Test de la fonctionnalité de la gestion des planifications	Afficher les tâches dans un calendrier et filtrer par jour, semaine et mois	Planification calendrier fonctionnelle	Conforme
Test de la fonctionnalité de gestion des tâches dans le calendrier	Ajouter, modifier, changer le statut et supprimer des tâches	Gestion complète fonctionnelle	Conforme

Tableau 13 : Tableau de test de sprint 2 release 2

3. Sprint 3 Module paramétrage :

Ce sprint a pour but de développer la module paramétrage de notre projet qui est :

- L'authentification
- La gestion des utilisateurs de la plateforme
- La gestion des membres d'équipe de travail

3.1.Backlog du sprint :

ID	User Story	ID Tâche	Tâche	Estimation (Jour)
1	Authentification	1.1	Interface Authentification	
		1.2	Développement d'un API d'authentification	
2	Utilisateurs plateforme	2.1	Interface de gestion d'utilisateur	
		2.2	Ajouter, afficher, modifier utilisateur	
		2.3	Chercher utilisateur, changer Mot de passe, consulter les informations de l'utilisateur	
3	Module membres l'équipe	3.1	Interface de gestion des membres d'équipe de travail	
		3.2	Inviter, lister, supprimer un membre d'équipe	
		3.3	Chercher membre, consulter les informations du membre	

Tableau 14 : Backlog sprint 3 release 2

3.2.Analyse :

3.2.1. Diagramme des cas d'utilisation « Sprint 3 » :

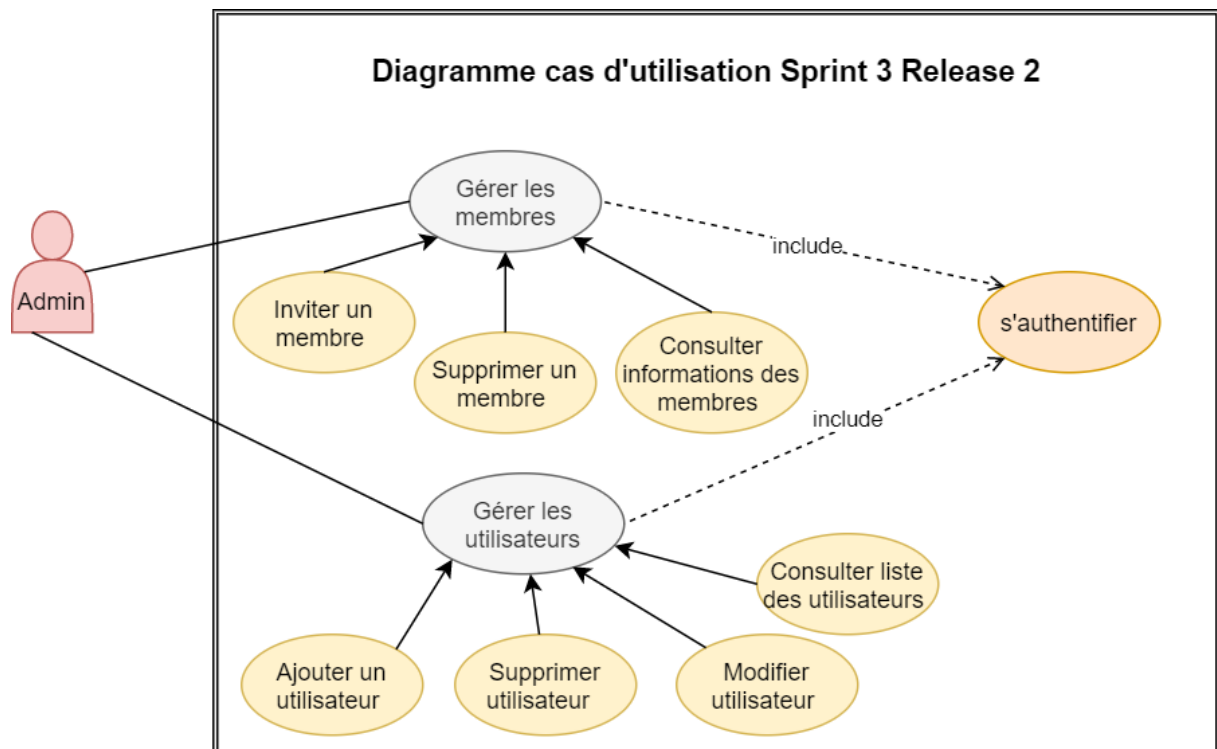


Figure 23 : Diagramme de cas d'utilisation sprint 3 release 2

3.2.2. Description Textuelle :

➤ Description textuelle du cas d'utilisation « Authentification » :

Cas d'utilisation	Authentification
Acteurs	Administrateur
Pré-condition	Profil déjà créée
Scénario nominal	<ol style="list-style-type: none"> 1. L'utilisateur demande la connexion 2. Le système affiche le formulaire d'authentification contenant le login et le mot de passe 3. L'acteur remplit les champs nécessaires et valide 4. Le système vérifie les données saisies 5. Le système ouvre l'espace personnel de l'acteur en question
Scénario alternatif	4- l'acteur saisie des données incorrectes

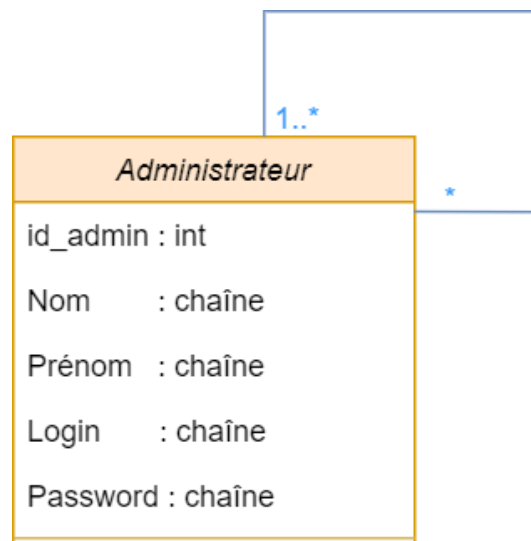
	<p>4-1- le système affiche des messages d'erreurs</p> <p>4-2- reprise de l'étape 3 du scénario nominal</p>
--	------------------------------------------------------------------------------------------------------------

➤ Description textuelle du cas d'utilisation « Gérer les utilisateurs » :

Cas d'utilisation	Gérer les utilisateurs
Acteurs	Administrateur
Pré-condition	Administrateur connecté
Description	La gestion des utilisateurs permet de gérer les A

3.3.Conception :

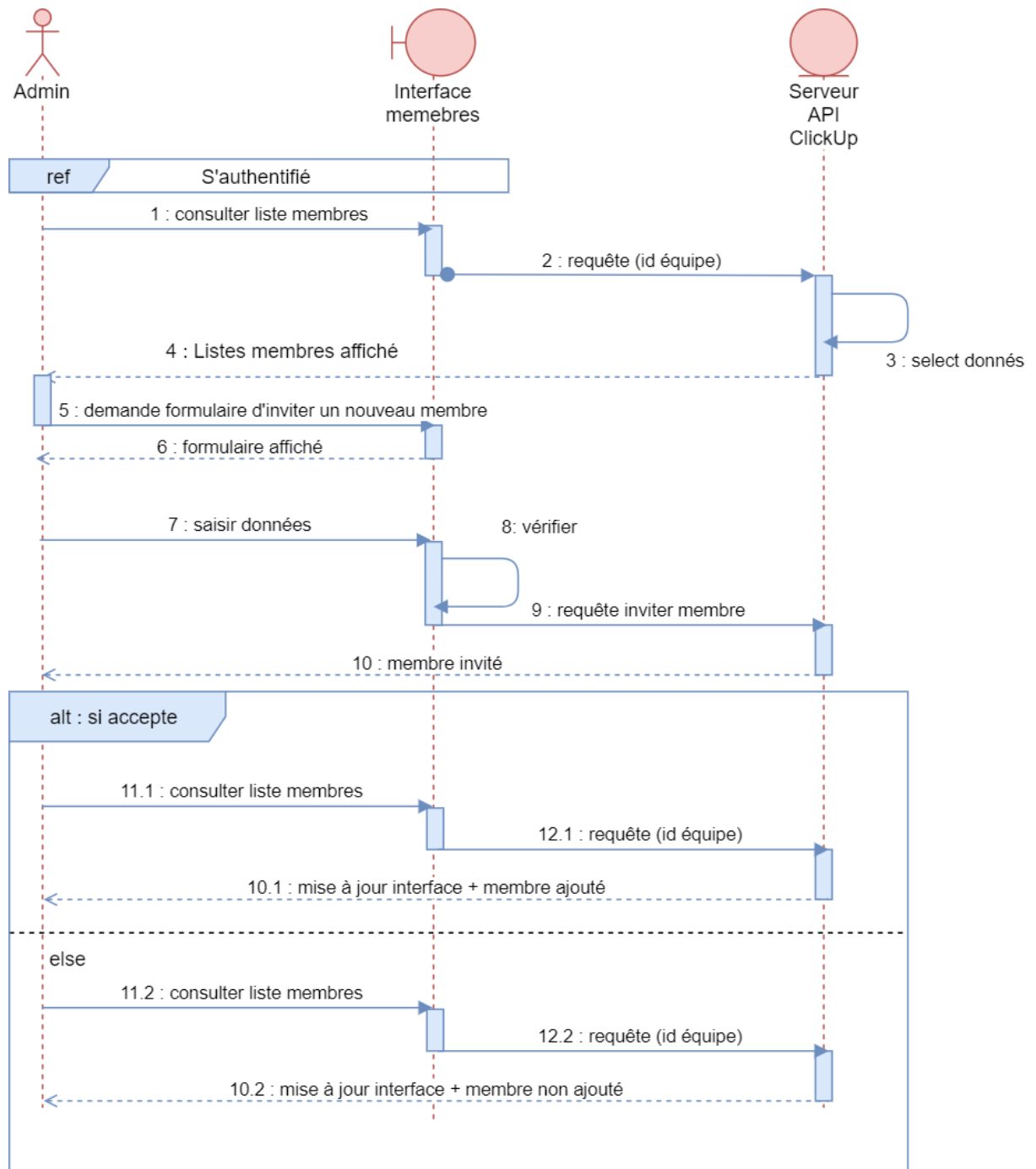
3.3.1. Diagramme de classe :



3.3.2. Diagrammes de séquence :

➤ Diagramme de séquence d'authentification :

L'administrateur doit s'authentifier et fournir son nom de connexion et son mot de passe afin d'accéder à la plateforme. Si les données saisies sont incorrectes, le contrôleur Demander un message d'erreur.



➤ **Diagramme de séquence d'authentification :**

Chaque administrateur doit s'authentifier, en fournissant son login et son mot de passe afin d'accéder à son espace personnel. Si les données saisies sont erronées, le contrôleur demande d'afficher un message d'erreur

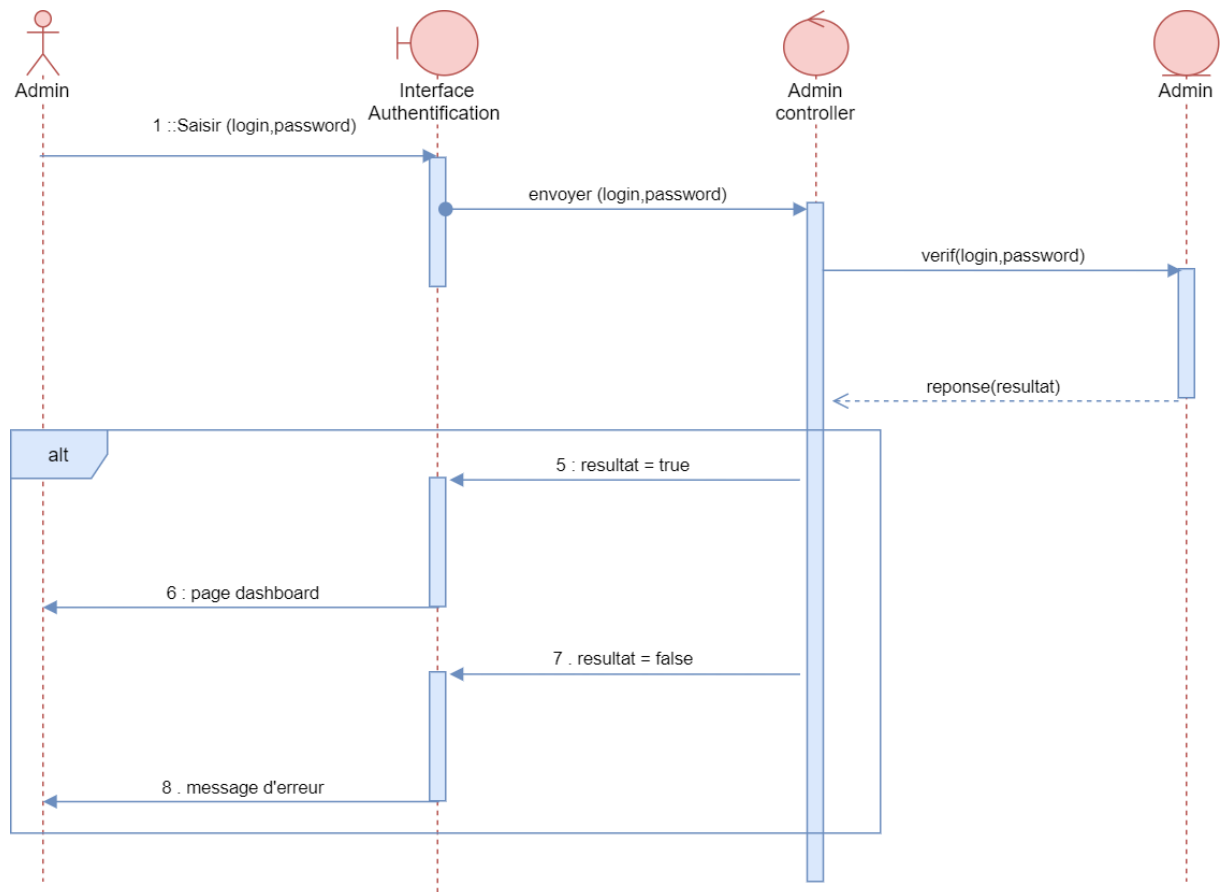


Figure 24 : Diagramme de s  quence « S'authenti   »

3.4.R  alisation :

➤ Interface gestion membres :

Cette interface repr  sente l'interface « gestion membres » o   l'administrateur peut inviter, modifier et supprimer un membre d'une   quipe de travail.

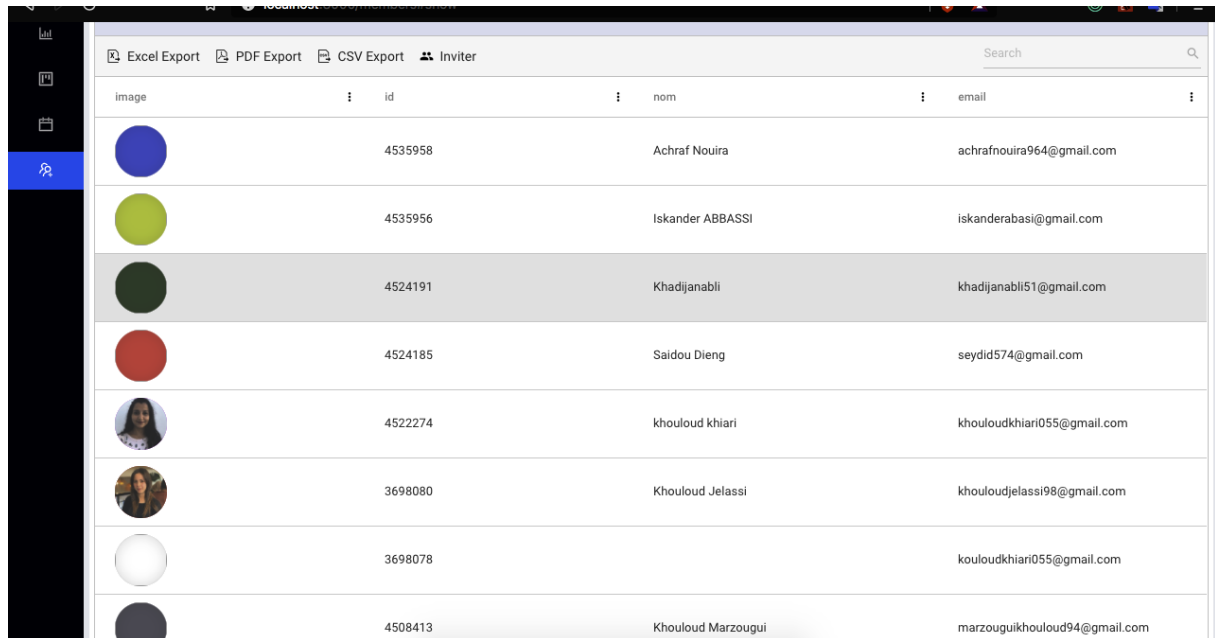










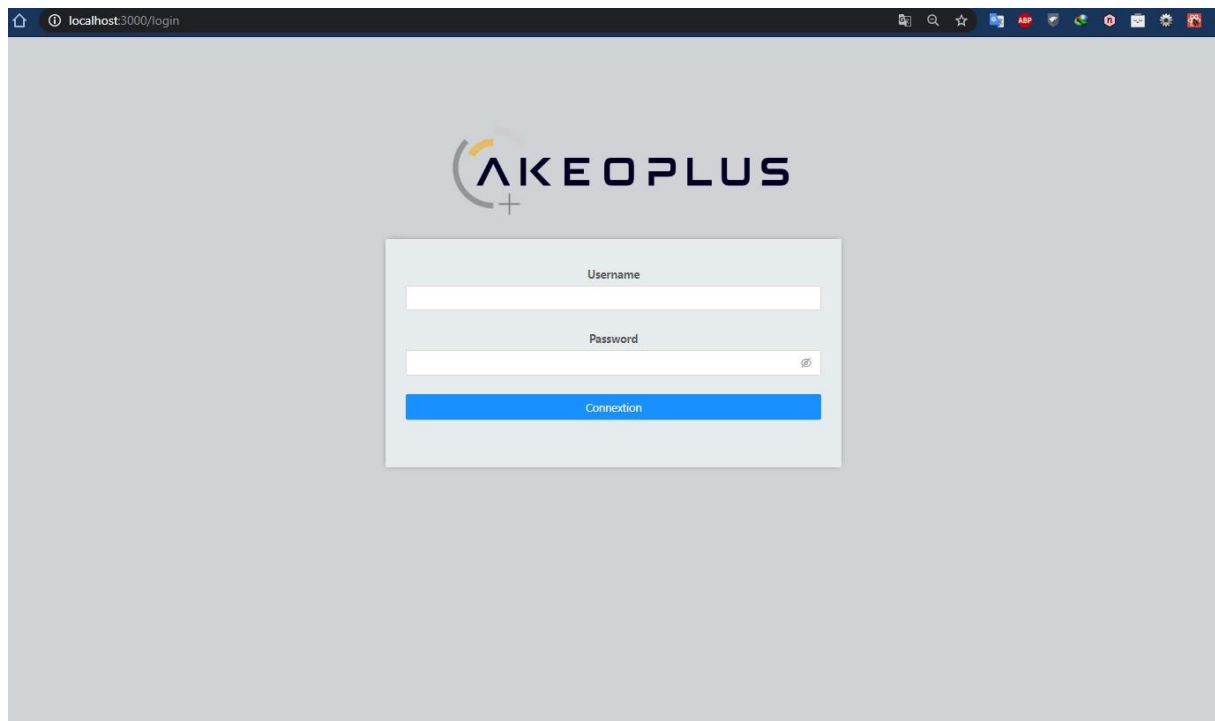
image	id	nom	email
	4535958	Achraf Nouira	achrafnouira964@gmail.com
	4535956	Iskander ABBASSI	iskanderabasi@gmail.com
	4524191	Khadjanabli	khadjanabli51@gmail.com
	4524185	Saidou Dieng	seydid574@gmail.com
	4522274	khouloud khiari	khouloudkhiari055@gmail.com
	3698080	Khouloud Jelassi	khouloudjelassi98@gmail.com
	3698078		kouloudkhiari055@gmail.com
	4508413	Khouloud Marzougui	marzouguikhouloud94@gmail.com

Figure 25 : Interface gestion membres

➤ Interface Authentification :

L'interface Authentification permet à l'utilisateur d'accéder au système après une validation de son login et son mot de passe.



localhost:3000/login

AKEOPLUS

Username

Password

Connexion

Figure 26 : Interface authentification

3.5.Test :

Cas de test	Démarche	Comportement attendu	Résultat
Test d'authentification	S'authentifier	Authentification réussie	Conforme
Test de fonctionnement de la gestion des utilisateurs	Ajouter, Modifier, supprimer utilisateurs	Gestion utilisateurs réussie	Conforme
Test de fonctionnement de la gestion des membres	Inviter, modifier, supprimer des membres	Gestion membres réussie	Conforme mais il faut que le type du compte ClickUp qui appartienne à l'administrateur actuel soit de type entreprise

Tableau 15 : Tableau de test sprint 3 release 2

Conclusion :

Durant ce chapitre nous avons réalisé le module paramétrage tout ce qui concerne la gestion utilisateurs, membres d'équipe et l'authentification. Aussi la réalisation de la planification par un calendrier convivial et gérer des graphiques de données. D'abord, nous avons fait l'analyse détaillée puis la conception et enfin la réalisation de ce dernier incrément.

Conclusion Générale

Le présent rapport est réalisé dans le cadre de notre projet de fin d'études chez AkeoPlus Tunisie en vue de l'obtention du diplôme de la licence appliquée en Technologie de l'Informatique (Développement des Systèmes Informatiques).

Ce travail a pour objectif de développer une application Web pour la gestion des projets basé sur API ClickUp.

Nous avons commencé dans un premier lieu par comprendre le contexte général de notre application. Puis, nous avons préparé par la suite notre planning de travail en respectant les priorités de nos besoins.

Pour pouvoir compléter notre mission, nous avons détaillé les différentes étapes d'analyse, de conception et de réalisation de ce système et nous avons essayé tout au long de notre travail de construire notre application étape par étape en utilisant la méthodologie SCRUM.

Ce projet était bénéfique pour nous dans plusieurs sens. Il nous ouvrir sur le monde professionnel qui implique de la responsabilité et de la discipline, il a été une occasion pour améliorer nos connaissances dans les environnements de travail et de maîtriser les langages de programmation pour qui seront certes utiles au niveau professionnel.

La réalisation de cette application web nous a été très bénéfique sur le plan technique. En effet, nous avons eu une opportunité de mieux maîtriser les Framework ReactJs et Symfony4 et découvrir les nouveaux outils de développement d'application Web.

En conclusion, ce travail a atteint ses objectifs, mais ce n'est qu'un début. Il existe des extensions qui peuvent enrichir notre application, comme l'ajout d'autres services mettent en œuvre des méthodes de gestion des risques avec l'intelligence artificielle et améliorent l'expérience utilisateur en ajoutant des systèmes de notification en temps réel.

Nétographie

- [1] <https://www.akeoplus.com/> [Accès mars 2020]
- [2] <https://www.capterra.fr/software/158833/clickup#reviews> [Accès Mars 2020]
- [3] <https://www.lucidchart.com/pages/fr/quest-ce-que-le-langage-de-modelisation-unifie> [Accès Mars 2020]
- [4] <https://www.journaldunet.com/solutions/dsi/1419985-comparatif-des-methodes-agiles/> [Accès Mars 2020]
- [5] http://www.scrum-institute.org/images_scrum/Simple_Burndown_Chart.jpg [Accès Mars 2020]
- [6] https://en.wikipedia.org/wiki/MoSCoW_method [Accès Mars 2020]

Annexe

Annexe :

