

RÉPUBLIQUE TUNISIENNE
MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA
RECHERCHE SCIENTIFIQUE

INSTITUT SUPÉRIEUR DE GESTION DE SOUSSE



PROJET DE FIN D'ÉTUDE
POUR L'OBTENTION DE DIPLÔME EN INFORMATIQUE APPLIQUÉ À
LA GESTION SPÉCIALITÉ BI

**Conception et Réalisation d'une Plateforme
Web Pour La Gestion Des Projets**

ORGANISME D'ACCUEIL : POULINA GROUP HOLDING



Réalisé par :

Zaabi Manel

Zayri Mourad

Encadré par :

Mr. Hamdi Hassen : Maître Assistant (ISGs)

Mme. Rim Zarrouk : co-encadreur (ISGs)

Mr. Rebhi Ali : Professionnel

Mme. Elouafi Samar : Professionnel

Année Universitaire 2021-2022

Dédicace

Je dédie ce travail :

À ma chère mère,

À mon cher père,

Qui n'ont jamais cessé de formuler des prières à mon égard, de me soutenir et de m'épauler pour que je puisse atteindre mes objectifs,

À mes frères,

À ma soeur,

Ceux qui ont partagé avec moi tous les moments d'émotion lors de la réalisation de ce travail. Ils m'ont chaleureusement supporté et encouragé tout au long de mon parcours,

À ma famille, mes proches,

ceux qui me donnent de l'amour et de la vivacité,

À tous mes amis, qui m'ont toujours encouragé, et à qui je souhaite plus de succès

À tous ceux que j'aime.

Manel & Mourad

Remerciements

Au terme de ce travail, nous tiendrons à adresser nos sincères et vifs remerciements à tous ceux qui nous avons aidé à accomplir notre travail.

Nous adresserons en premier Lieu aux membres du jury que nous remercions d'avoir accepté d'évaluer ce projet.

Nous remercions vivement notre superviseur **Hamdi Hassen** qui s'est montré très collaborateur et qui a su nous consacrer son temps quand j'en avais besoin. Nous lui remercions énormément pour ses précieux conseils et remarques.

Nous remercions particulièrement mes encadrants techniques **Mr. Ali Rebhi et Mme. Elouafi Samar** pour la qualité de l'encadrement qu'ils nous avons assurées, pour ses disponibilités, ses conseils et remarques judicieuses et surtout pour ses confiances et ses encouragements continuels qui nous ont permis d'avancer.

Nous remercions tout le staff de **Poulina Group Holding** pour leur soutien, leur encouragement et l'agréable ambiance du travail que nous avons trouvé. .

Nous ne laisserons pas cette occasion passer sans exprimer nos reconnaissances envers tous mes professeurs de l'École Supérieure de Gestion de Sousse.

Table des matières

Introduction générale	1
1 Cadre Générale du projet	3
1.1 Introduction	3
1.2 Présentation de l'organisme d'accueil	3
1.2.1 Pouлина Group Holding	3
1.2.2 Secteurs d'activité PGH	4
1.2.3 Présentation du service d'accueil	6
1.3 Présentation du projet	7
1.3.1 Contexte et problématique	7
1.3.2 Travail demandé	8
1.4 Choix de la méthodologie	9
1.5 Conclusion	14
2 Sprint 0	15
2.1 Introduction	15
2.2 Identification des besoins	15
2.2.1 Les acteurs	15
2.2.2 Les besoins fonctionnels	16
2.2.3 Les besoins décisionnelles	18
2.2.4 Les besoins non fonctionnels	18
2.3 Pilotage du projet avec SCRUM	19
2.3.1 Les fonctionnalités du backlog	19
2.3.2 Diagramme de cas d'utilisation globale	20
2.4 Planification des sprints	20
2.5 Environnement de travail	22
2.5.1 Environnement matériel	22

2.5.2	Environnement de développement	24
2.5.3	Environnement logiciel	27
2.6	Architecture générale de l'application	30
2.6.1	Architecture logique	30
2.6.2	Architecture physique	31
2.6.3	Architecture du système décisionnelles	32
2.6.4	Méthodologies et Approches	33
2.7	Diagramme de déploiement	34
2.8	Conclusion	35
3	Étude et réalisation du sprint 1	36
3.1	Introduction	36
3.2	Sprint Backlog	36
3.3	Spécification fonctionnelle	38
3.3.1	Diagrammes de cas d'utilisation générale du Sprint 1	38
3.3.2	Diagramme de cas d'utilisation détaillé	38
3.4	Spécification des cas d'utilisations	39
3.4.1	Spécification du cas d'utilisation "modifier information projet"	43
3.5	Conception	46
3.5.1	Diagrammes de séquences détaillés	46
3.5.2	Diagramme de Classe	52
3.6	Réalisation	52
3.6.1	Interface d'authentification	53
3.6.2	Création d'un nouveau projet ou modification d'un projet existant	53
3.6.3	Consultation de toutes les informations d'un projet	54
3.6.4	Consulter liste type objectifs	54
3.6.5	Consultation de toutes les informations d'un projet	55
3.7	Test sprint 1	55
3.7.1	Cas de test d'authentification	55
3.7.2	Cas de test d'ajout d'un projet	57
3.7.3	Cas de test Affecter objectif à un projet	58
3.8	Scrum Board	59
3.9	Conclusion	60

4 Etude et réalisation du sprint 2	61
4.1 Introduction	61
4.2 Sprint Backlog	61
4.3 Spécification fonctionnelle	63
4.3.1 Diagrammes de cas d'utilisation générale du Sprint 2	63
4.3.2 Diagramme de cas d'utilisation détaillé	64
4.4 Spécification des cas d'utilisations	66
4.5 Phase de conception analytique de GIMSI	72
4.6 Conception	72
4.6.1 Diagrammes de séquences détaillés	72
4.6.2 Diagramme de Classes	76
4.7 Conception Data Warhouse	77
4.7.1 Entrepôt de données	77
4.7.2 Le magasin de données (Datamart)	77
4.7.3 Choix des indicateurs	81
4.7.4 Les critères des indicateurs	81
4.7.5 Sélection des indicateurs	82
4.8 Phase mise en œuvre de GIMSI	82
4.8.1 Intégration et déploiement	82
4.8.2 Identification des données et collecte des informations :	82
4.8.3 Développement de la phase ETL	83
4.9 Réalisation	84
4.9.1 Interface "liste des tâches"	84
4.9.2 Interface "ajouter tâche"	84
4.9.3 Interface "Liste des types des tâches"	84
4.9.4 Interface des tableaux de bord	84
4.10 Test Sprint 2	86
4.10.1 Cas de test d'ajout d'un type tâche	86
4.10.2 Cas de test la gestion des tâches	87
4.11 Scrum Board	89
4.12 Conclusion	89
5 Étude et réalisation du sprint 3	93
5.1 Introduction	93

5.2	Sprint Backlog	93
5.3	Spécification fonctionnelle	95
5.3.1	Diagramme de cas d'utilisation générale du Sprint 3	95
5.3.2	Diagrammes de cas d'utilisation détaillés	95
5.4	Spécification des cas d'utilisations	97
5.5	Conception	101
5.5.1	Diagrammes de séquences détaillés	101
5.5.2	Diagramme de Classes	107
5.6	Réalisation	108
5.7	Test Sprint 3	109
5.8	Scrum Board	110
5.9	Conclusion	110

Table des figures

1.1	Le logo de Pouлина Groupe Holding	4
1.2	Structure principale du groupe PGH	4
1.3	Organigramme du service informatique du groupe PGH[1].	6
1.4	La méthodologie Scrum détaillée[2]	13
2.1	Diagramme du cas d'utilisation globale	20
2.2	Logo IBM RSA	27
2.3	Logo Visual Studio Code	28
2.4	Logo Visual Studio 2019	28
2.5	Logo Trello	28
2.6	Logo latex - Overleaf	29
2.7	Logo GitHub	29
2.8	Logo Vertabelo	30
2.9	Architecture logique du projet	31
2.10	Architecture de système décisionnel	32
2.11	Les étapes de la méthodologies GIMSI	34
2.12	Diagramme de déploiement	35
3.1	Diagramme du cas d'utilisation générale du sprint 1	38
3.2	Raffinement du cas d'utilisation "gérer projet"	38
3.3	Raffinement du cas d'utilisation "gérer objectifs"	39
3.4	Diagrammes de séquences cas d'utilisation "s'authentifier"	47
3.5	Diagrammes de séquences cas d'utilisation "Recherche Projet"	48
3.6	Diagrammes de séquences cas d'utilisation "Ajouter Projet"	49
3.7	Diagrammes de séquences cas d'utilisation "modifier Type objectifs"	51
3.8	Diagramme de classe Sprint 1	52
3.9	Interface "Authentification"	53

3.10 Interface "création d'un nouveau projet ou modifier un projet existant"	53
3.11 Interface "Consultation de toutes les informations d'un projet"	54
3.12 Interface "Consultation du liste des types d'objectifs"	54
3.13 Interface "Consultation de toutes les informations d'un projet"	55
3.14 Interface du "responsable"	56
3.15 Interface du "collaborateur"	56
3.16 Interface du "client"	57
3.17 Interface "ajout d'un projet"	58
3.18 Interface "ajout un projet avec succès"	58
3.19 Interface "affectation objectif à un projet"	59
3.20 Scrum Board Sprint 1	59
 4.1 Diagramme du cas d'utilisation générale du sprint 2	63
4.2 Raffinement du cas d'utilisation "gérer tâche"	64
4.3 Raffinement du cas d'utilisation "consulter liste tâche a valider"	64
4.4 Raffinement de cas d'utilisation "consulter liste type tâche"	65
4.5 Raffinement de cas d'utilisation "visualiser tableau de bord"	65
4.6 Cas d'utilisation "consulter liste des tâches"	72
4.7 Cas d'utilisation "supprimer tâche"	73
4.8 Cas d'utilisation "visualiser tableau de bord Suivi des objectifs"	74
4.9 Cas d'utilisation "visualiser tableau de bord Suivi des taches"	75
4.10 Diagramme de classe sprint2	76
4.11 Data mart"Suivi état projet"	78
4.12 Data mart"Suivi objectifs"	79
4.13 Data mart"Suivi taches"	80
4.14 Le processus ETL	83
4.15 Alimentation de la table dimension "dimprojet"	83
4.16 Alimentation de la table de faits "FactVision"	84
4.17 Interface "liste des tâches"	85
4.18 Interface "ajouter tâche"	85
4.19 Interface "Liste des types des tâches"	86
4.20 tableau de bord "État projet"	86
4.21 tableau de bord "Suivi des objectifs"	87
4.22 tableau de bord "Suivi des tâches"	87

4.23 Interface "ajouter type tâche"	88
4.24 Interface "type tâche ajouté avec succès"	89
4.25 Interface "ajouter tâche"	90
4.26 Interface "alert d'ajout avec succé"	90
4.27 Interface "liste tâche apres l'ajout"	91
4.28 Interface "Modifier tâche "	91
4.29 Scrum Board Sprint 2	92
 5.1 Diagrammes du cas d'utilisation générale du Sprint 3	95
5.2 Raffinement du cas d'utilisation "gérer groupe"	96
5.3 Raffinement du cas d'utilisation "gérer réunion"	96
5.4 Raffinement du cas d'utilisation "gérer type réunion"	97
5.5 Raffinement du cas d'utilisation "gérer demande de réunion"	97
5.6 Cas d'utilisation "recherche demande"	101
5.7 Cas d'utilisation "modifier réunion"	104
5.8 Cas d'utilisation "supprimer réunion"	106
5.9 Diagramme de classe sprint3	107
5.10 Interface "ajouter réunion"	108
5.11 Interface "Gestion des Réunions"	108
5.12 Interface "Consultation Liste des groupes"	109
5.13 Interface "afficher détaillles Réunions"	109
5.14 Scrum Board Sprint 3	110

Liste des tableaux

1.1	Les rôles de l'équipe Scrum	10
1.2	Événements Scrum (cérémonies)	12
1.3	Artefacts Scrum	12
2.1	Backlog général	21
2.2	Partitionnement des user stories par sprint	22
3.1	Backlog du sprint 1	37
3.2	Spécification du cas d'utilisation « S'authentifier»	40
3.3	Spécification du cas d'utilisation "ajouter projet"	41
3.4	Spécification du cas d'utilisation "supprimer projet"	42
3.5	Spécification du cas d'utilisation modifier information projet	44
3.6	Spécification du cas d'utilisation consulter liste projet	45
3.7	Spécification du cas d'utilisation consulter liste types objectifs	46
4.1	Backlog du sprint 2	62
4.2	Spécification du cas d'utilisation «ajouter projet»	67
4.3	Spécification du cas d'utilisation "supprimer tâche"	69
4.4	Spécification du cas d'utilisation "modifier information tâche"	71
4.5	Spécification du cas d'utilisation "visualiser tableau de bord"	72
5.1	Backlog du sprint 3	94
5.2	Spécification du cas d'utilisation "Ajouter une demande de réunion"	98
5.3	Spécification du cas d'utilisation "modifier une demande de réunion"	102
5.4	Spécification du cas d'utilisation supprimer une demande de réunion	103

Introduction générale

Dans un monde où tout se monnaie, un monde où toutes les entreprises font la guerre aux minutes gaspillées et aux dépenses superflues, toute innovation qui pourrait nous faire gagner du temps et ainsi de l'argent serait la bienvenue et surtout dans un monde où on fait l'investissement dans de nouveaux projets, une industrie qui ne fait pas exception à la règle. En effet, les acteurs de cet écosystème essaient, par tous les moyens possibles, de grappiller quelques secondes et quelques centimes par-ci et par-là afin d'optimiser leurs dépenses et de diminuer leurs coûts de fonctionnement.

Ce problème se pose encore plus lorsqu'il s'agit d'une activité capitale pour ces institutions : la gestion des projets . En effet, lorsqu'il s'agit de respecter les délais, les objectifs, les normes de qualité ,d'optimiser la productivité de l'entreprise afin de maximiser son chiffre d'affaires et les bénéfices prévus , d'économiser temps et argent et d'identifier les goulets d'étranglement et la baisse des effectifs, un millième de seconde peut faire la différence et ainsi faire gagner ou perdre une entreprise des sommes colossales. Sans oublier le risque de dépassement de délai et de budget qui sera la première raison de la perte de clientèle et revenus.

Toutes ces contraintes obligent, d'une manière ou d'une autre, les acteurs des entreprises à investir massivement dans des systèmes informatiques de plus en plus performants, de plus en plus complexes et donc de plus en plus coûteux afin de réduire, au maximum, la quantité de ressources gâchées. Nous parlons alors du Business intelligence ou appelé aussi l'informatique décisionnelle qui permet à tous les acteurs de l'entreprise d'avoir une vision globale sur leurs activités afin de les aider par la suite à prendre les bonnes décisions, grâce aux outils du business intelligence. Parmi ces outils présents actuellement sur le marché, l'un d'entre eux a spécialement le vent en poupe : Jira. En effet, cette dernière arrive avec la promesse de faire bien gérer un projet , de bien estimer le temps, les coûts ainsi que les jalons à respecter. Sans oublier une dernière promesse, qui n'est pas des moindres, est de bien assurer une flexibilité en adoptant la méthodologie agile.

Or, tout outil qui présente des points forts et des avantages a forcément un talon d'achille. Dans notre cas, c'est le fait que jira rende les transitions indisponibles depuis le portail client , soit un environnement un peu clos ou «trustless» et non extensible et que son utilisation semble un peu complexe. C'est ici que nous intervenons. Notre mission se résume à l'étude d'une alternative pour permettre une meilleure gestion des projets plus adaptée aux besoins de l'entreprise pour appliquer par la suite cette solution à tout type de projet au sein d'un groupe industriel et service tunisien.

Le présent rapport a pour but de poser le contexte du projet, de décrire le sujet ainsi que les méthodes et les outils utilisés et, bien évidemment présenter les résultats obtenus. Il s'articule principalement sur cinq chapitres :

- Le premier chapitre, intitulé "Cadre générale du projet". Il comprend la présentation de l'organisme d'accueil, le contexte du projet et le travail demandé.
- Le deuxième chapitre est consacré à l'analyse et à la spécification des besoins aussi il est consacré à l'exposition de l'environnement de travail en illustrant le cadre de développement et les technologies utilisées dans notre application.
- Le troisième, quatrième et sixième chapitre détaillent le cycle de vie des sprints ayant pour objectif la réalisation des microservices "Gestion Projet", "Gestion Réunion" et "Gestion demande".

Nous clôturerons par une conclusion générale qui résumera notre travail.

Chapitre 1

Cadre Générale du projet

1.1 Introduction

Dans ce chapitre, nous présentons d'abord l'organisation d'accueil, son secteur événementiel et son produit, ensuite nous essaierons d'expliquer la motivation et les objectifs du projet. Finalement, la méthode utilisée et les raisons de ce choix.

1.2 Présentation de l'organisme d'accueil

1.2.1 Pouline Group Holding

Pouline Group Holding (PGH) a été créé en 1967, l'année où tout a commencé avec l'aviculture. Le Groupe a ensuite engagé une stratégie d'intégration de toutes les activités de cette filière : de l'importation des reproducteurs à la distribution des produits transformés. Successivement, une diversification de ces activités a été initiée en investissant dans divers secteurs à forte valeur ajoutée.

En 2010, PGH a lancé une action de restructuration qui a donné naissance à la recentralisation de son groupe autour de 8 métiers pour faciliter la gestion et le suivi des performances.

Les 8 métiers sont l'intégration avicole, les produits de grande consommation, la transformation de l'acier, l'emballage, l'immobilier, le bois et les biens d'équipement, les matériaux de construction et le commerce et les services. D'une dimension internationale, le Groupe possède une vingtaine de filiales à l'étranger, principalement au Maroc, en Algérie, en Libye, en France, au Sénégal et en Chine.

La figure 1.1 présente le logo de Pouline Groupe Holding. [1]



FIGURE 1.1 – Le logo de Poulina Groupe Holding

1.2.2 Secteurs d'activité PGH

La structure principale du Poulina Groupe Holding est illustrée par la figure 1.2 :

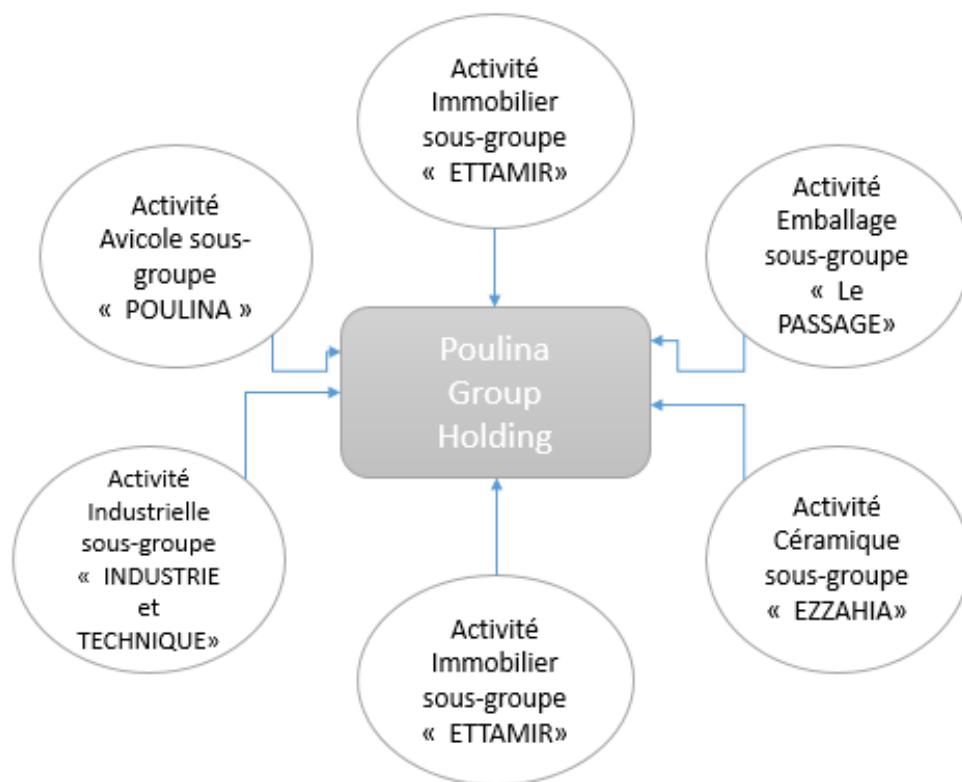


FIGURE 1.2 – Structure principale du groupe PGH

Le Secteur avicole

Le secteur avicole assure l'approvisionnement du pays en viandes à hauteur de 50% du total des viandes (contre 36% en 1994) ainsi que la totalité des besoins en œufs de consommation. Par ailleurs, et bien que les prix des produits avicoles soient très bon marché, ce secteur représente environ 25% de la valeur de l'élevage et 8% des productions agricoles en 2006.

Le Secteur agro-alimentaire et services

La Tunisie, qui compte 12 centrales laitières, est en autosuffisance en lait frais depuis la fin des années 90 (production de 970 millions de litres en 2006) mais, importe toujours du lait en poudre.

La production de yaourts est assurée par 9 entreprises à partir de lait frais, et a fortement augmenté dans les années 90. Cette activité tend à se diversifier avec le développement des desserts lactés (yaourt aux fruits, yaourt à boire, crème dessert).

Le Secteur de la céramique

Le secteur de la céramique connaît depuis le début des années 2000, une véritable mutation grâce à l'introduction d'un nouveau produit : « le grès ». Ce produit connaît un grand succès dans le pays et permet de faire gagner des parts de marché aux revêtements de sols en céramique aux dépend des revêtements concurrents (carreaux de ciment blanc/mosaïque ; le marbre). Grâce au succès de ses produits, la société exporte plus de 20% de sa production vers une trentaine de pays, notamment la France, les pays d'Afrique de l'Ouest, le Maghreb et les pays du Golfe.

Le Secteur industriel

En Tunisie, le secteur du bois et de l'ameublement compte environ 400 entreprises mais reste majoritairement dominé par les petites et micro entreprises indépendantes dont le nombre représente 80% du marché. Sur les 400 sociétés « structurées », seules une dizaine d'entreprises sont de grandes tailles, relativement spécialisées, emploient plus de 100 personnes et écoulent leurs produits via des revendeurs franchisés.

Les débouchés du bois sont principalement l'ameublement (meubles d'intérieur, salons, meubles de cuisine, meubles de bureaux et meubles pour collectivités) et le bâtiment.

Le Secteur de l'emballage

Depuis une dizaine d'années, le secteur de l'emballage connaît une croissance remarquable favorisée par le développement du secteur industriel qui génère des besoins croissants en matière d'emballage.

La société UNIPACK est leader avec 30% du marché dans les segments carton ondulé et carton compact.

Le Secteur immobilier

Poulina Group Holding s'est lancé dans le métier de l'immobilier à travers sa filiale ETTAAMIR, créée en 1997.

Ainsi, plusieurs projets de haut standing ont été réalisés dans les quartiers les plus huppés de la Tunisie, d'autres projets sont en cours de construction et d'autres projets en cours d'études.

À travers sa filiale ETTAAMIR, le groupe PGH est devenu un acteur incontournable du secteur immobilier en Tunisie.

1.2.3 Présentation du service d'accueil

Le service informatique nous a accueilli durant ce stage, le département développement informatique dont nous y effectuons notre projet de fin d'étude est conçu pour le développement des applications web, en fait notre tâche est de réaliser une application web, intitulée "Gestion de projet".

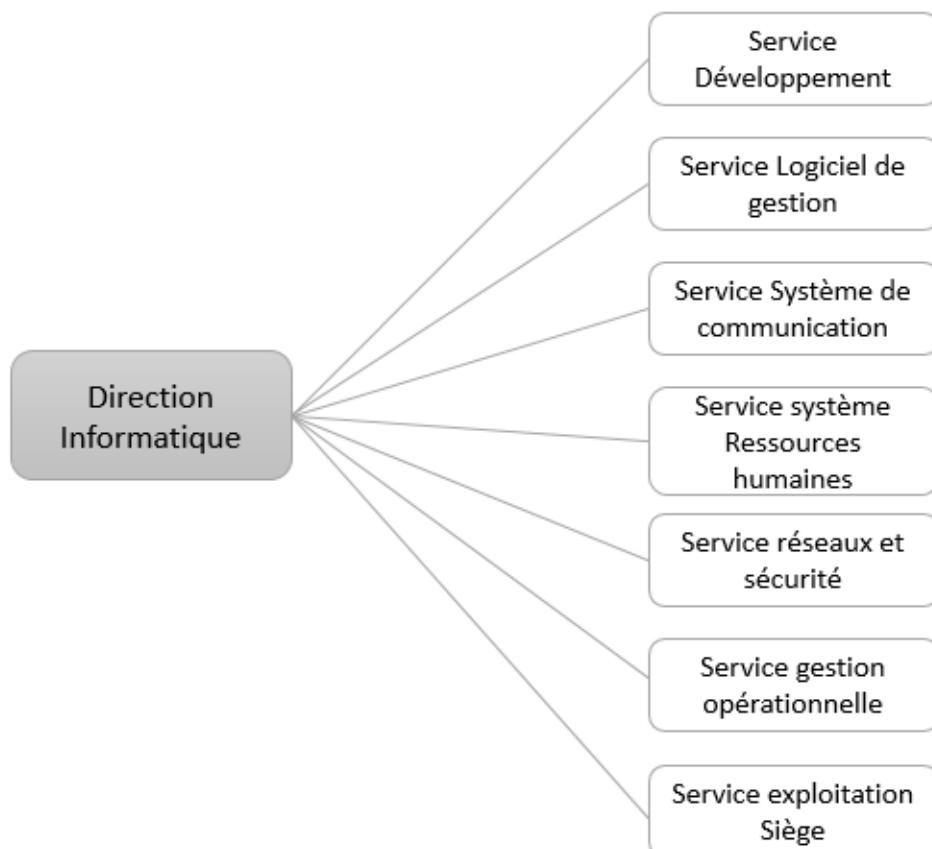


FIGURE 1.3 – Organigramme du service informatique du groupe PGH[1].

1.3 Présentation du projet

1.3.1 Contexte et problématique

Depuis plusieurs années, le nombre des projets en cours de réalisation est en énorme évolution. Le nombre d'entreprises ayant recours à la gestion de projet augmente, tout comme l'apparition d'outils et de méthodes afin de simplifier la vie des chefs de projet et de leurs équipes, afin de conduire le projet vers la réussite.

Dans ce contexte, le processus actuel de gestion de projet, de suivi de l'état d'avancement de projet et de son évaluation souffre d'un nombre considérable de lacunes. Ces défaillances sont à cause de plusieurs problèmes. Parmi les lacunes que connaît notre système actuel pour la méthodologie adoptée pour la gestion des projets, nous pouvons en citer :

- L'absence de confiance vis-à-vis des clients et entre les différents acteurs et intervenants eux-mêmes, des pertes de revenus issues des retards , des goulots d'étranglement se produisent au sein des projets, ce qui a une incidence sur le moral de l'équipe. Les autres membres risquent de devoir attendre pour agir, les projets restent dans le flou et la frustration s'installe.
- Une mauvaise communication est la principale raison pour laquelle un projet n'atteint pas le résultat souhaité ou, tout simplement, échoue.
- Une mauvaise planification : ne pas arriver à définir précisément le périmètre, les responsabilités et le calendrier du nouveau projet à réaliser,l'entreprise aura de sérieux problèmes par la suite. Sans un bon plan d'actions, l'entreprise risque d'un manque de soutien .

Un manque de ressources et de temps et une utilisation des mauvais outils : Les chefs de projet utilisent souvent plusieurs outils pour faire le suivi des tâches, des membres de l'équipe, des livrables et de l'état d'avancement. Cependant,

dans de nombreux cas le logiciel n'est pas assez souple pour leurs accompagner dans la gestion des projets.

Face à cette problématique, un nouvel outil s'avère être très prometteur et incroyablement efficace. C'est celle du software jira. En effet, jira permet de résoudre un certain nombre de défaillances relatives à mettre en œuvre une gestion de projet agile et de prendre en charge des projets et des flux de travail complexes et multidimensionnels . Elle permet aussi à des intervenants qui ne se connaissent pas de se faire confiance.

Mais, par ce qu'il y'a toujours un mais, un seul et dernier problème nous barre la route. Un problème de ne pas être totalement adapté aux besoins de l'entreprise et de rendre ainsi les transitions indisponibles depuis le portail client , aujourd'hui, et bien que ce nouveau système révolutionnera sans aucun doute le monde de réalisation des projets, aucun produit existant n'est suffisamment complet et performant afin de pouvoir assurer toutes les spécificités techniques requises pour construire un système permettant de suivre , de contrôler et de gérer des projets à réaliser sans passer par le circuit conventionnel. C'est ici que ce projet intervient.

1.3.2 Travail demandé

Les logiciels de gestion des projets présents aujourd'hui sur le marché sont certes incomplets mais chacun d'entre eux s'avère être très bon lorsqu'il s'agit d'un traitement en particulier. En effet, certains sont excellents lorsqu'il s'agit d'avoir une visibilité sur les objectifs et les projets à long terme, leur avancement et fournissent également des informations de publication en temps réel, d'autres sont performants au niveau de la collaboration en ligne, le suivi des heures détaillé et les tableaux de bord personnalisés et ainsi de suite..

Nous avons donc pensé à allier un certain nombre de logiciels différents et com-

plémentaires afin de proposer un produit unique qui jouit de la performance et de l'efficacité requise pour un système comme le notre.

Notre travail se fera en deux volets :

- Nous devrons, en premier lieu, proposer une conception qui nous permettrait de mettre en considération les différents services , sous services et filiales de l'entreprise d'accueil , permettraient de répondre aux différents besoins en gestion de projet de l'entreprise et surtout de la généraliser afin d'être valable pour tout type de projet à réaliser .
- En second lieu, nous aurons à développer une application qui permettrait de suivre et de gérer le déroulement complet d'un projet en se basant sur la conception précédemment créée. (Cas pratique)

1.4 Choix de la méthodologie

Nous avons utilisé la méthodologie agile Scrum qui est la méthode utilisée au sein de service développement de Poulina.

Cette méthodologie permet au client de voir et de tester le produit avant qu'il ne soit complètement terminé pour se rendre compte si des changements sont nécessaires. Plus tôt ces changements sont détectés, moins leur impact et leur coût sont réduits.

Rôles de l'équipe Scrum

Le tableau 1.1 présente les rôles de l'équipe Scrum

Rôles de l'équipe Scrum	
ProductOwner	Est la principale partie prenante du projet, généralement un client interne ou externe. Il n'y a qu'un seul Product Owner qui transmet la mission globale et la vision du produit que l'équipe va réaliser. Le propriétaire du produit est ultimement le responsable pour gérer le backlog de produit, vérifier, tester et donner avis vis-à-vis aux produits achevés.
ScrumMaster	Est le leader au service du Product Owner, de l'équipe d'organisation et de développement. Sans autorité hiérarchique sur l'équipe mais plutôt un facilitateur, le ScrumMaster s'assure que l'équipe adhère à la théorie, aux pratiques et aux règles.
Équipe de développement	Est un groupe interfonctionnel auto-organisé armé de toutes les compétences pour livrer des incrémentés livrables à la fin de chaque sprint.

TABLE 1.1 – Les rôles de l'équipe Scrum

Événements Scrum (cérémonies)

Le tableau 1.2 présente les événements scrum (cérémonies).

Artefacts Scrum Le tableau 1.3 présente les Artefacts Scrum.

Règles de Scrum

Les règles de Scrum agile devraient être entièrement à la charge de l'équipe et régies par ce qui fonctionne le mieux pour leurs processus.

Pour résumer ,la figure 1.3 décrit la méthodologie Scrum en montrant ses composants de base et ses concepts détaillés dans les tableaux précédents . Les principales étapes de la méthodologie Scrum se résument comme suit :[3]

- Le Product Owner (ou chef de produit) identifie la vision du produit. La vision du produit est une définition du pourquoi le produit est, comment il va participer à la

Événements Scrum (cérémonies)	
Sprint	Un sprint est une période de temps dans laquelle un travail spécifique est terminé et préparé pour l'examen (review). Les sprints sont généralement de 2 à 4 semaines mais peut être aussi courte qu'une semaine.
Planification du sprint	Les réunions d'équipe de planification sont des événements ponctuels qui déterminent quels articles du backlog de produit seront livrés et comment le travail sera réalisé.
Le stand-up quotidien	Le stand-up quotidien est une courte réunion de communication (non plus de 15 minutes) dans laquelle chaque membre de l'équipe rapidement et en toute transparence couvre les progrès depuis le dernier stand-up, les travaux prévus avant la prochaine réunion et tout obstacle.
La revue de sprint	La revue de sprint est le "show-and-tell" où l'action de démonstration permettant à l'équipe de présenter les travaux réalisés pendant le sprint. Le Product Owner vérifie le travail par rapport aux critères d'acceptation pré-définis et il accepte ou rejette l'oeuvre. Les parties prenantes ou les clients donnent leur avis pour s'assurer que l'incrément livré répondait aux besoins de l'entreprise.

stratégie de l'entreprise et qui va utiliser le produit .

- Le Product Owner du produit crée une feuille de route (Roadmap) du produit. La feuille de route de produits est une vue de haut niveau des exigences du produit, avec un macro planning identifiant les grands jalons avec la date de livraison souhaitée . Cette étape doit permettre d'identifier les besoins de produits, d'hiérarchiser et d'estimer à peu près l'effort nécessaire à la réalisation de ces exigences.

- Le Product Owner crée un plan de release . Un projet agile aura beaucoup de re-

La rétrospective	La rétrospective, ou rétro, est la dernière réunion d'équipe dans le Sprint pour déterminer ce qui s'est bien passé, ce qui ne s'est pas bien passé, et comment l'équipe peut s'améliorer lors du prochain Sprint. Assisté par le Product Owner , les membres de l'équipe et le Scrum Master, la rétrospective est une opportunité pour l'équipe de se concentrer sur sa performance globale et identifier des stratégies d'amélioration continue de ses processus.
------------------	---

TABLE 1.2 – Événements Scrum (cérémonies)

Artefacts Scrum	
Carnet de produit (Product Backlog)	Le carnet de produit est le document le plus important qui décrit chaque exigence d'un projet. Il peut être une liste des tâches composée d'éléments de travail, chacun va produire un livrable à valeur commerciale. Les éléments en retard sont commandés en termes de valeur commerciale par le Product Owner.
Backlog de sprint	Un backlog de sprint est la liste spécifique des éléments extraits du backlog de produits qui doivent être complétés dans un sprint.
Un incrément	Un incrément est la somme de tous les éléments de backlog de produit qui ont été achevés depuis la dernière version du logiciel. Alors que c'est le Product Owner qui décide quand un incrément est libéré, il est de la responsabilité de l'équipe de s'assurer tout ce qui est inclus dans un incrément est prêt à être libéré.

TABLE 1.3 – Artefacts Scrum

lease (versions), avec les caractéristiques les plus prioritaires réalisées en premier et une release comprend trois à cinq sprints.

- Le Product Owner , le Scrum Master, et l'équipe de développement planifient les

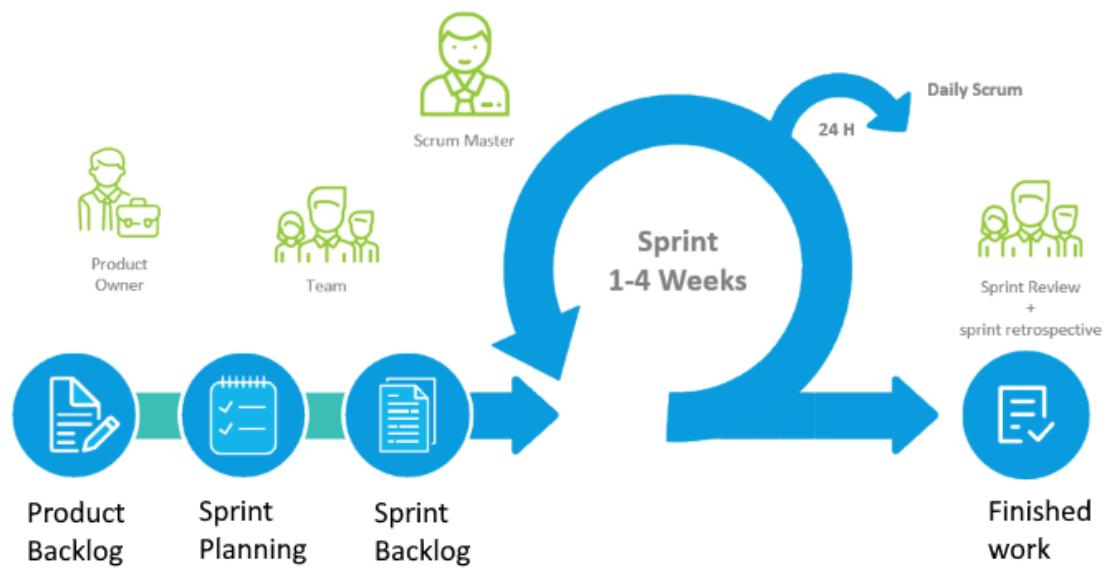


FIGURE 1.4 – La méthodologie Scrum détaillée[2]

sprints ce qui permet de mettre à jour et confirmer le plan de release. Lors de cette séance les intervenants déterminent le contenu du prochain sprint. Le sprint planning a lieu au début de chaque sprint.

- Les récits utilisateurs (User Stories) sont écrits. Les critères d’acceptation sont aussi écrits. Ces éléments sont fournis aux développeurs et permettent de faire les tests de chaque User Story.
- L’équipe de développement commence la réalisation du sprint. Au cours du sprint , des réunions quotidiennes (daily scrum) ont lieu et ne doivent pas dépasser les 15 minutes. Durant ces réunions les membres de l’équipe évoquent les points sur les- quels ils ont avancés, les tâches en cours et les éventuels points de blocage qu’ils ont identifié.
- La revue de sprint (Sprint Review), à la fin de chaque sprint, pour démontrer le produit créé lors du sprint aux parties prenantes de produits. Un échange a lieu sur les problématiques rencontrées.
- L’équipe fait une rétrospective de sprint (Sprint Rétrospective). La rétrospective de

sprint est une réunion où l'équipe explique comment le sprint s'est déroulé et les plans d'amélioration du prochain sprint.

1.5 Conclusion

Comme vu dans ce chapitre, nous avons présenté en premier lieu l'organisme d'accueil, ses chiffres clés ainsi que ses secteurs d'activité et ses produits. Nous avons, ensuite, présenté le projet de façon à donner une idée claire et concise du travail qui sera réalisé par la suite et nous avons clôturé avec le choix de la méthodologie qui sera utilisée tout au long du projet.

Chapitre 2

Sprint 0

2.1 Introduction

Après l'explication des différents concepts théoriques, passons à la prochaine étape, qui nous semble cruciale pour l'élaboration de ce projet. Cette étape englobe les préparatifs nécessaires pour le lancement des sprints dans de bonnes conditions, pour être précis et claire, ces derniers constituent l'étude préliminaire, appelée aussi sprint "0". En effet, cette phase exige l'implication de l'utilisateur afin de ressortir les besoins fonctionnels et non fonctionnels attendus de cette application, nous aborderons par la suite le backlog du produit et planifions les sprints.

2.2 Identification des besoins

La capture des besoins constitue une étape importante qui nous permet d'avoir une idée plus claire et détaillée sur notre système. Deux types de besoins seront détectés : les besoins fonctionnels, non fonctionnels et les différents acteurs du système.

2.2.1 Les acteurs

Un acteur est une entité externe qui interagit avec le système, pouvant être une personne humaine ou un robot.

Les acteurs identifiés dans notre solution sont :

- **Responsable** : Il est chargé de la gestion des projets ,des objectifs , des tâches et encore la gestion des réunions. son objectif est le pilotage au quotidien . Il a besoin des indicateurs pertinents pour suivre l'avancement du projet et s'assurer que tout est sous contrôle.
- **Collaborateur** : Il peut créer des objectifs et des tâches selon le rôle attribué , suivre l'avancement de ses tâches et faire des demandes de réunions.
Chaque collaborateur a besoin d'avoir une vision sur l'avancement de ses propres tâches et sur les projets.
- **Client** : Il peut donner avis suite à chaque réunion et gérer une demande de réunion.

2.2.2 Les besoins fonctionnels

En disposant de plusieurs fonctionnalités, la solution doit nécessairement satisfaire les ensembles des besoins suivants :

- **Authentification** :

Chaque utilisateur doit obligatoirement s'authentifier .

- **Gestion des projets** :

Le Responsable doit avoir la possibilité de créer un projet associé au sous service dont il appartient (il peut y appartenir à plusieurs) , quand il le décide et ce en choisissant le statut de ce projet (active ou inactive) , une durée estimée et un vis-à-vis généralement un client interne appartenant au groupe PGH . Il a la possibilité à tout moment de modifier , de consulter les informations relatives à un projet , de supprimer un projet ou de faire clôturer un projet.

- **Gestion des objectifs** :

Une fois que le projet est crée par le responsable , le responsable doit avoir la possibilité de créer au sein de ce projet (si son statut est actif) un objectif en choisissant le type de cet objectif ,c'est la décomposition du projet en des objectifs.Il a le droit aussi de modifier les informations relatives à un objectif bien déterminé et de supprimer un objectif. Le collaborateur a la possibilité de consulter la liste des objectifs.

- Gestion des groupes :

Le responsable doit avoir la possibilité de créer un groupe à la base des membres du dernier groupe créé associé au dernier objectif d'un projet bien déterminé , à tout moment, un groupe peut y contenir ou non un ou plusieurs membres , Il doit avoir la possibilité de consulter la liste des membre d'un groupe bien déterminé , c'est la gestion des groupes.

- Gestion des tâches :

Notre application doit permettre à un collaborateur d'ajouter une tâche après avoir choisir le type de cette tâche (un type qui varie d'un sous service à un autre) et une durée estimée , de la supprimer , de la modifier , à un responsable d'assigner des tâches à des membres de groupes au sein d'un objectif bien déterminé relatif à un projet . Notre système doit être en mesure de permettre à un collaborateur de consulter la liste de ses propres tâches , leurs états , d'assigner à lui même des tâches à faire ,de demander une assistance sur une tâche en état de blocage et faire passer des tâches à la validation . Le responsable doit aussi avoir la possibilité de valider les tâches en attente de vérification . Il s'agit de la décomposition des objectifs en des tâches.

- Gestion des réunions

Le responsable doit avoir la possibilité de planifier des réunions, à tout moment, de consulter la liste des réunions dans un calendrier ainsi que la liste des membres qui seront présents .Si un membre sélectionné pour atteindre une réunion n'est plus

disponible pour une date et une durée bien déterminées , le système va indiquer la disponibilité du membre sélectionné et le responsable a le choix de sélectionner ce membre ou de l'écartier et permet également à un vis-à-vis de donner avis suite à chaque réunion.

- Gestion des demandes

Notre application doit permettre à un collaborateur d'envoyer, de modifier et de supprimer une demande de réunion , à un vis-à-vis d'envoyer, de modifier , de supprimer une demande de réunion et à un responsable de faire approuver ou refuser une demande . Ils doivent aussi avoir la possibilité de consulter la liste des demandes de réunions .

2.2.3 Les besoins décisionnelles

Visualisation des tableaux de bord :

- suivi des tâches (en cours, non assignées, nombre de tâches terminées, etc.)
- suivi durée des objectifs et des tâches (estimés, réelles.)
- statut des projets (en cours/réalisé/clôturé)

Ce type de tableau de bord facilite la gestion de projet et améliore l'efficacité du pilotage.

2.2.4 Les besoins non fonctionnels

Les besoins non fonctionnels décrivent toutes les contraintes auxquelles est soumis le système pour sa réalisation et son bon fonctionnement. Ainsi, la solution proposée devrait prendre en considération les contraintes suivantes :

- **Sécurité :** Les informations propres à chaque utilisateur doivent rester confidentielles et seulement accessibles par l'utilisateur en question selon le rôle attribué.
- **Performance :** Les exécutions des différents traitements relatifs aux fonctionnalités offertes par l'application doivent être effectuées dans un temps minimal.

- **Ergonomie :** Le système doit disposer d'une interface graphique conviviale, claire et adaptée à ce type d'applications .
- **Fiabilité :** La moyenne du temps de disponibilité de l'application doit être élevée. Elle doit aussi supporter la charge relative à un très grand nombre d'accès simultanés.
- **Évolutivité :** Notre plateforme doit être capable de supporter de nouvelles fonctionnalités si besoin .

2.3 Pilotage du projet avec SCRUM

Nous allons tenter de cerner notre équipe SCRUM :

- **Le Product Owner :** Monsieur Ali Rebhi Il définit les besoins, les priorités et les fonctionnalités et il dirige l'activité de l'équipe de développement.
- **Le Scrum Master :** Monsieur Hamdi Hassen Il assure également l'organisation des réunions et la bonne application de la méthode Scrum .
- **L'équipe de développement :** Manel Zaabi et Mourad Zayri Ils réalisent le travail nécessaire pour aboutir à un incrément du produit livrable à la fin de chaque itération.

2.3.1 Les fonctionnalités du backlog

Après avoir cité les besoins de notre application, nous décrivons dans cette partie le backlog du produit, illustré par le Tableau 2.1, qui représente une liste de tâches exprimées sous forme de besoins.

2.3.2 Diagramme de cas d'utilisation globale

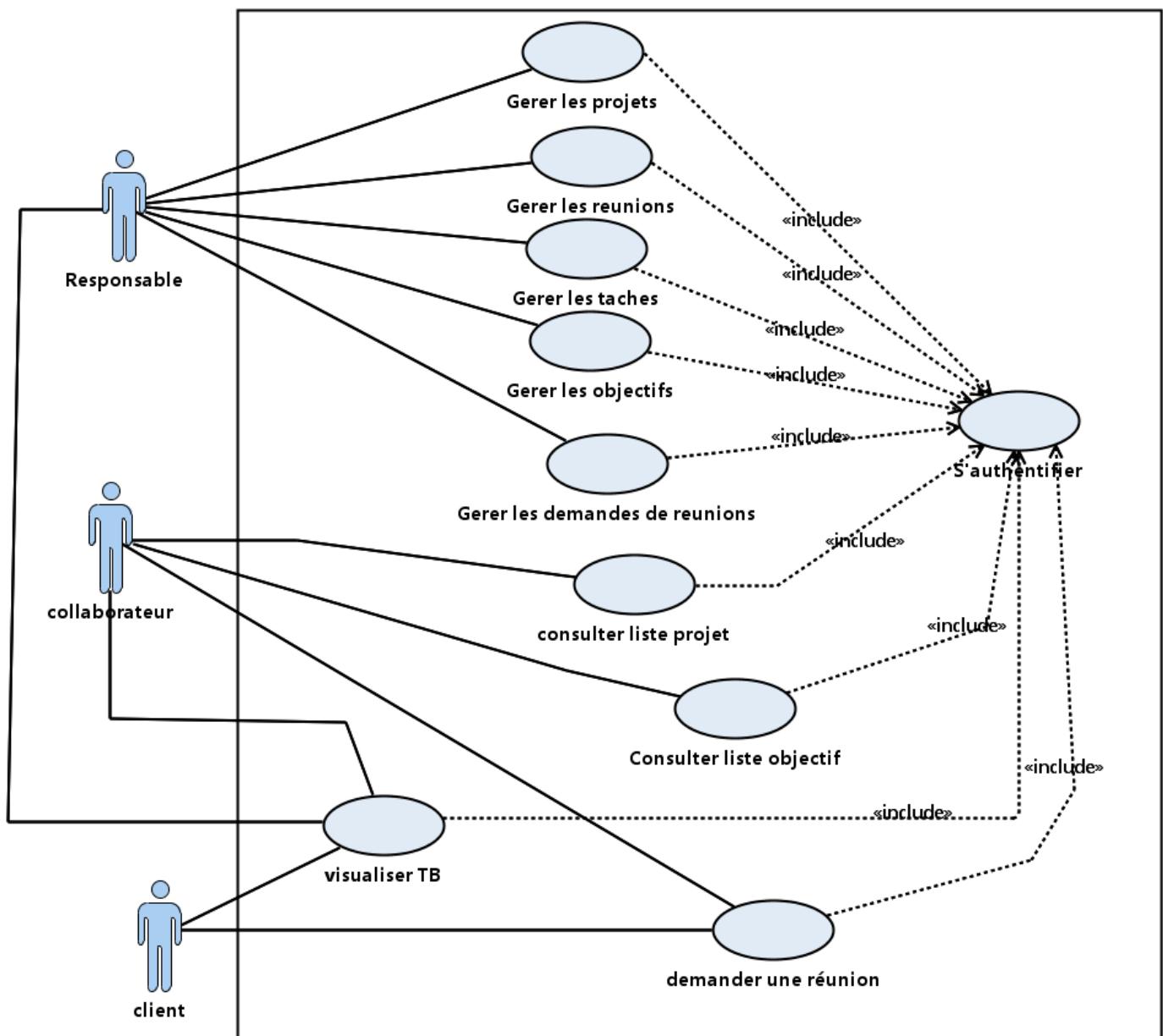


FIGURE 2.1 – Diagramme du cas d'utilisation globale

2.4 Planification des sprints

En se basant sur notre backlog du produit nous divisons notre Release en 3 sprints comme le montre le tableau 2.2 ci-dessous.

lightgray ID	User stories	complexité	Priorité
1	En tant qu'utilisateur, je veux connecter à l'application	Moyenne	2
2	En tant que responsable , je veux gérer un projet	Élevée	1
3	En tant collaborateur, je veux consulter la liste des projets	Faible	2
4	En tant que responsable , je veux gérer les objectifs associé à un projet	Élevée	1
5	En tant que responsable , je veux gérer un groupe de travail	Élevée	1
6	En tant que collaborateur, je veux consulter la liste des groupes	Faible	2
7	En tant que responsable, je veux gérer une tâche	Élevée	1
8	En tant que collaborateur, je veux gérer mes tâches	Faible	2
9	En tant que responsable, je veux gérer les tableaux de bord relatives aux projets , tâches et objectifs	Élevée	1
10	En tant que responsable, je veux gérer les réunions	Élevée	1
11	En tant que collaborateur, je veux gérer les demandes de réunions	Élevée	2
12	En tant que client, je veux pouvoir donner mon avis suite a une réunion	Moyenne	2
13	En tant qu'utilisateur, je veux pouvoir visualiser les tableaux de bord	Élevée	1

Sprint	User Story
Sprint 1 : Authentification, gestion de projet et gestion des objectifs	User Story 1, 2, 3, 4
Sprint2 : gestion des tâches et visualisation du tableau de bord	User Story 7, 8, 9, 13
Sprint 3 : gestion des réunions, gestion des demandes de réunions et gestion des groupes	User Story 5, 6, 10, 11, 12

TABLE 2.2 – Partitionnement des user stories par sprint

2.5 Environnement de travail

2.5.1 Environnement matériel

Notre application a été développée sur deux machines ayant la configuration suivante :

Pc du Manel Zaabi

- **Système d’exploitation :** windows 10 Professionnel 64bits
- **processeur :** Intel(R) Core(TM) i7-7500U CPU @ 2.70GHz 2.90 GHz
- **Mémoire RAM :** 8 Go.
- **Disque dur :** SSD 256 Go et HDD 1000Go .

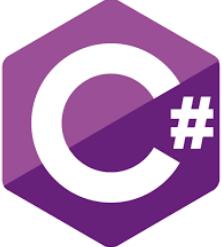
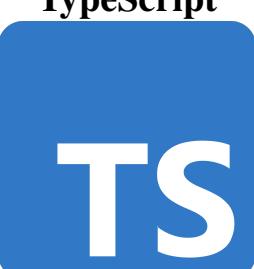
Pc du Mourad Zayri

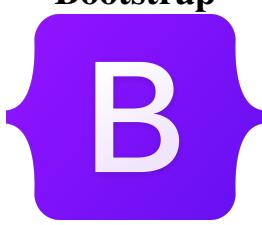
- **Système d’exploitation :** windows 10 Professionnel 64bits

- **processeur :** Intel(R) Core(TM) i7-7500U CPU @ 2.70GHz 2.90 GHz
- **Mémoire RAM :** 8 Go.
- **Disque dur :** SSD 256 Go et HDD 1000Go .

2.5.2 Environnement de développement

Entity Framework 	<p>Entity Framework est le mapping objet-relationnel utilisé par la technologie "Microsoft .NET". Il est écrit en langage C. C'est un framework ORM open source pour les applications .NET pris en charge par Microsoft. Il permet aux développeurs de travailler avec des données à l'aide d'objets de classes spécifiques au domaine sans se concentrer sur les tables et colonnes de base de données sous-jacentes dans lesquelles ces données sont stockées[4].</p>
LINQ 	<p>LINQ (Language-Integrated Query) est un puissant langage de requête introduit avec .Net 3.5 et Visual Studio 2008. C'est un composant du framework .NET de Microsoft qui ajoute des capacités d'interrogation sur des données aux langages .NET en utilisant une syntaxe proche de celle de SQL[5].</p>
.NET Framework 	<p>.NET est une plateforme de développement composée d'outils, de langages de programmation et de bibliothèques pour la création de nombreux types d'applications. C'est un standard proposé par la société Microsoft, pour le développement d'applications d'entreprises multi-niveaux, basées sur des composants[6].</p>

Angular 	Angular est une plate-forme et un cadre pour la création d'applications clientes d'une seule page à l'aide de HTML et de TypeScript. Angular est écrit en TypeScript. Il implémente les fonctionnalités de base et facultatives en tant qu'ensemble de bibliothèques TypeScript pour le développement d'applications.
C sharp 	Est un langage de programmation orientée objet, commercialisé par Microsoft depuis 2002 et destiné à développer sur la plateforme Microsoft .NET[7].
Microsoft SQL Server 	C'est un système de gestion de base de données relationnelles en langage SQL incorporant entre autres un SGBDR (SGBD relationnel) . Il est développé et commercialisé par la société Microsoft. Il fonctionne sous les OS Windows et Linux (depuis mars 2016), mais il est possible de le lancer sur Mac OS via Docker[8].
TypeScript 	TypeScript est un langage de programmation libre et open source développé par Microsoft qui a pour but d'améliorer et de sécuriser la production de code JavaScript. Il s'agit d'un sur-ensemble syntaxique strict de JavaScript. Le code TypeScript est transcompilé en JavaScript, et peut ainsi être interprété par n'importe quel navigateur web ou moteur JavaScript. TypeScript a été cocréé par Anders Hejlsberg, principal inventeur de C .

 <p>NuGet</p>	<p>NuGet est le gestionnaire de paquets de la plate forme de développement Microsoft .NET. C'est un logiciel libre et open source . Il vise à simplifier le processus d'intégration de bibliothèques externes à une application développée dans .NET en automatisant certaines tâches répétitives : télécharger et installer une bibliothèque, modifier les paramètres de configuration, et répéter la même opération pour les autres bibliothèques auxquelles la première fait appel[14].</p>
 <p>Node.js</p>	<p>Node.js est un serveur Web qui permet d'exécuter du code JavaScript en dehors du contexte du navigateur. Il est populaire en raison de sa vitesse d'exécution et de son aspect asynchrone. Par ailleurs, Node.js introduit un certain nombre d'innovations dans le développement des applications Web [10]</p>
 <p>HTML5/CSS3</p>	<p>HTML et CSS sont des langages utilisés pour créer des pages web, les mettre en forme et assurer leurs designs.</p>
 <p>Bootstrap</p>	<p>Bootstrap est une collection d'outils utile à la création du design (graphisme, animation et interactions avec la page dans le navigateur , etc...) de sites et d'applications web. C'est un ensemble qui contient des codes HTML et CSS, des formulaires, boutons, outils de navigation et autres éléments interactifs, ainsi que des extensions JavaScript en option[11].</p>

 <p>JQuery write less, do more.</p>	<p>JQuery est un framework Javascript sous licence libre, qui permet de faciliter des fonctionnalités communes de Javascript. L'utilisation de cette bibliothèque permet de gagner du temps de développement. JQuery possède l'avantage d'être utilisable sur plusieurs navigateurs web (Internet Explorer, Firefox, Chrome, Safari ou Opera). La bibliothèque JQuery possède les fonctionnalités suivantes : Manipulation du DOM (HTML ou CSS), la gestion des évènements (clic, survol, soumettre un formulaire, etc), AJAX et les effet d'animation[12].</p>
---	---

2.5.3 Environnement logiciel

Rational Software Architect

Nous avons utilisé Rational Software Architect pour bien présenter nos diagramme UML : Rational Software Architect est un environnement de modélisation et de développement utilisant le langage UML pour la conception d'architecture d'applications et de services Web(Rational Software Architect, 2019).

La figure 2.2 représente le logo du IBM RSA



FIGURE 2.2 – Logo IBM RSA

Visual Studio Code

Visual Studio Code est un IDE développé par Microsoft pour Windows, Linux et OS X. Il combine simplicité et puissance pour proposer aux développeurs un éditeur de code performant pour une utilisation quotidienne dont il supporte une centaine de langages [7].

La figure 2.3 représente le logo du Visual Studio Code



FIGURE 2.3 – Logo Visual Studio Code

Visual Studio 2019

Visual Studio 2019 est un IDE développé par Microsoft pour Windows, Linux et OS X. Visual Studio est un ensemble complet d'outils de développement permettant de générer des applications web ASP.NET, des services web XML, des applications bureautiques et des applications mobiles. Il facilite la création de solutions faisant appel à plusieurs langages. Par ailleurs, ces langages permettent de mieux tirer parti des fonctionnalités du framework .NET[8].

La figure 2.4 représente le logo du Visual Studio 2019



FIGURE 2.4 – Logo Visual Studio 2019

Trello

Trello est un outil informatique qui offre des tableaux, des listes et des cartes qui permettent d'organiser les projets et de définir leur ordre de priorité de façon amusante, souple et enrichissante (Trello, 2019) [9].

La figure 2.5 représente le logo du Trello



FIGURE 2.5 – Logo Trello

latex - Overleaf

une plateforme en ligne gratuite permettant d’éditer du texte en LATEX sans aucun téléchargement d’application.[15]

La figure 2.6 représente le logo du latex - Overleaf



FIGURE 2.6 – Logo latex - Overleaf

GitHub

GitHub est un site web et un service de cloud qui aide les développeurs à stocker et à gérer leur code, ainsi qu'à suivre et contrôler les modifications qui lui sont apportées.[16]

La figure 2.7 représente le logo GitHub



FIGURE 2.7 – Logo GitHub

Vertabelo

Vertabelo est un outil de conception de base de données visuelle disponible via un navigateur Web. Ce ne prend en charge que la modélisation physique des données. Vertabelo propose la génération SQL, la rétro-ingénierie, le support du travail d’équipe, le lien public vers le modèle, le versioning du modèle et l’import/export au format XML[17].

La figure 2.8 représente le logo Vertabelo



FIGURE 2.8 – Logo Vertabelo

2.6 Architecture générale de l'application

2.6.1 Architecture logique

Dès les premières phases de conception de notre application, nous nous sommes mis comme objectif de, non seulement, proposer une application qui permettrait à ses utilisateurs de gérer les projets mais surtout de concevoir des applications puissantes et simples à maintenir .

On a utilisé l'architecture logicielle CQRS , un acronyme pour Command Query Responsibility Segregation , il s'agit d'un pattern d'architecture qui vise à séparer les couches de lecture et d'écriture ,on va séparer (Segregation) la responsabilité (Responsability) des lectures (Query) et écriture (Command) , il permet d'accroître l'évolutivité de l'application ainsi que d'en faciliter la maintenance. Il met en avant le fait que les responsabilités de lecture et d'écriture sont très différentes et qu'il faut donc les séparer logiquement et physiquement.

La Figure 2.8 décrit en détails l'architecture logique de notre l'application .

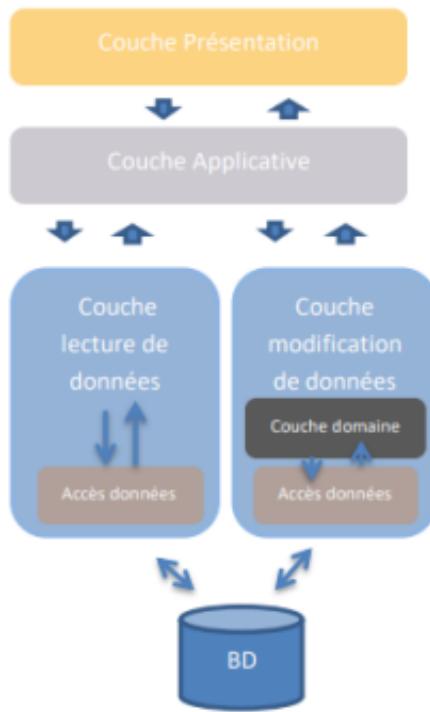


FIGURE 2.9 – Architecture logique du projet

2.6.2 Architecture physique

Suite à ce que nous avons mentionné sur l'architecture logique, maintenant il est temps de répartir celle-ci sur l'architecture physique, pour cela nous avons opté pour une architecture d'applications Web N-tiers basé sur les microservices.

Une architecture microservices est principalement orientée vers le back end, même si l'approche est également utilisée pour le front end.

Chaque service s'exécute dans son propre processus et communique avec d'autres processus par l'intermédiaire des protocoles à savoir HTTP/HTTPS, Web Sockets.. Il implémente une fonctionnalité métier ou de domaine spécifique de bout en bout dans une certaine limite de contexte, et doit être développé de manière autonome pour pouvoir être déployé indépendamment.

Enfin, chaque microservice doit avoir son modèle de données de domaine et sa logique de domaine associés (souveraineté et gestion des données décentralisée). Il peut être basé sur différentes technologies de stockage de données (SQL, NoSQL) et

sur différents langages de programmation.

2.6.3 Architecture du système décisionnelles

la figure 2.9 présente l'architecture de système décisionnel

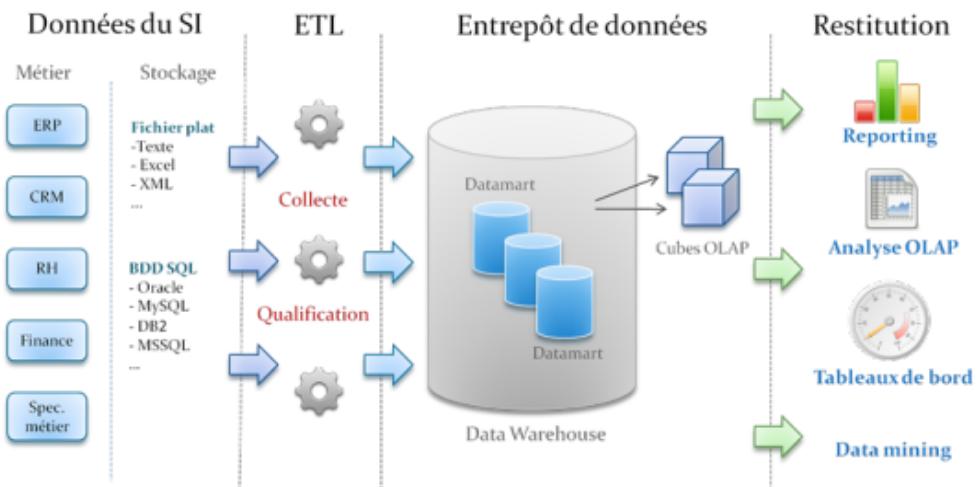


FIGURE 2.10 – Architecture de système décisionnel

Un système d'information décisionnel est composé de trois principaux composants :

- Le premier composant est la zone ETL (Extract, Transform, Load en anglais) où l'on effectue les traitements sur les données : cette zone doit être réservée aux développeurs. En aucun cas un utilisateur final ne devrait pouvoir y avoir accès car les données n'y sont pas encore forcément dans un état cohérent.

- Le second composant est la zone de stockage des données : historiquement nous avions ici un entrepôt de données (data warehouse en anglais) reposant sur une technologie OLAP. Désormais nous pouvons imaginer d'autres types de stockage de données comme le stockage in-memory ou les systèmes de données distribués (HDFS). Cependant, quel que soit le format ou le support de stockage choisi, la modélisation dimensionnelle est une excellente approche pour structurer les données de manière logique et de permettre d'y accéder.

- Le dernier composant est la zone de restitution des données qui couvre tous les outils qui génèrent des rapports ou des tableaux de bord. Cette zone peut permettre aussi de fournir des données à un niveau atomique pour des tâches de machine learning.[14]

2.6.4 Méthodologies et Approches

Dans le cadre de notre projet de fin d'études, nous choisissons la méthode GIMSI pour la répartition et la réalisation du tableau de bord.

Méthode Gimsi

Alain fernandez a mis en place la démarche GIMSI pour la conception des tableaux de bord de pilotage, qui se base sur la coopération et le partage de connaissances.

Se compose de 4 phases principales découpées en 10 étapes, chacune traitent une préoccupation particulière du projet tableaux de bord.comme indiqué dans le tableau

Phase	Etape	Objectifs
Identification	Etude de l'environnement de l'entreprise	Il s'agit d'identifier l'environnement économique et de stratégie de l'entreprise afin de définir le champ d'étude
	Identification de l'entreprise	Analyse des structures de l'entreprise pour identifier les processus, activités et acteurs concernés

Conception	Définition des objectifs	Sélection des objectifs tactiques de chaque équipe en fonction de la stratégie générale
	Construction du tableau de bord	Définition du tableau de bord de chaque équipe
	Choix des indicateurs	Choix des indicateurs en fonction des objectifs choisis, du contexte et des acteurs concernés
	Collecte des informations	Identification des informations nécessaires à la construction des indicateurs
	Le système de tableau de bord	Construction du système de tableau de bord, contrôle de la cohérence globale
Mise en oeuvre	Le choix des progiciels	Elaboration de la grille de sélection pour le choix des progiciels adéquats
	Intégration et déploiement	Implantation des progiciels, déploiement à l'entreprise
Amélioration permanente	Audit	Suivi permanent du système

FIGURE 2.11 – Les étapes de la méthodologies GIMSI

2.7 Diagramme de déploiement

Le diagramme de déploiement est utilisé pour visualiser la topologie des composants physiques d'un système dans lequel les composants logiciels sont déployés. (Edraw, 2019)

La figure présente le diagramme de déploiement de notre application.

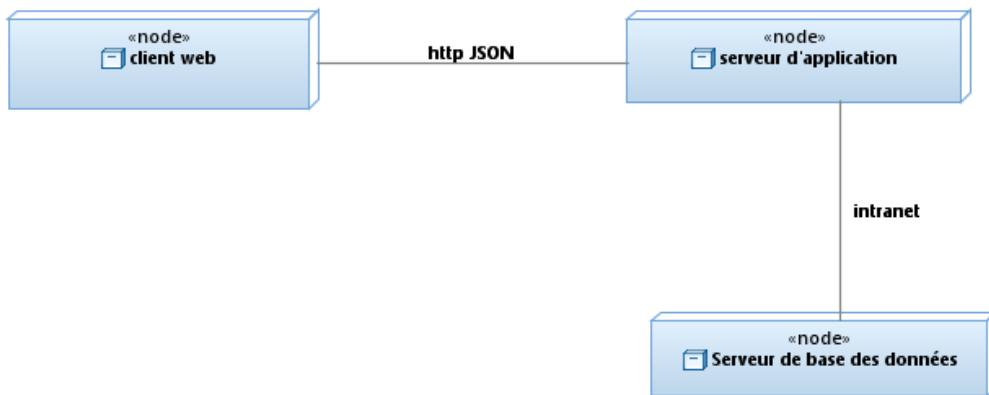


FIGURE 2.12 – Diagramme de déploiement

2.8 Conclusion

Dans ce chapitre nous avons préparé le plan du travail et énuméré les besoins fonctionnels et non fonctionnels de notre application. Nous avons aussi présenté le backlog et le diagramme des cas d'utilisation général. Par la suite, nous avons planifié et organisé le temps consacré à la réalisation du projet en identifiant les sprints. Finalement, nous avons défini et justifié nos choix technique.

Chapitre 3

Étude et réalisation du sprint 1

3.1 Introduction

Tout d'abord, nous allons commencer par les différents aspects du Sprint 1 qui inclut l'authentification, la gestion des projets et la gestion des objectifs.

3.2 Sprint Backlog

ID	User Story	priorité	complexité
1	En tant qu'utilisateur, je veux me connecter à l'application avec les bons droits.	2	moyenne
2	En tant que responsable , je veux créer un projet.	1	élevée
3	En tant que responsable , je veux modifier un projet.	1	élevée
4	En tant que responsable , je veux supprimer un projet.	1	élevée
5	En tant qu'utilisateur , je veux consulter les informations relatives à un projet.	2	faible
6	En tant que collaborateur,je veux consulter la liste des projets.	2	faible
7	En tant que responsable, je veux créer un objectif associé à un projet.	1	élevée
8	En tant que responsable , je veux modifier un objectif.	1	élevée
9	En tant que responsable , je veux supprimer un objectif.	1	élevée
10	En tant que collaborateur,je veux consulter la liste des objectifs d'un projet bien déterminé.	2	faible
11	En tant que responsable , je veux gérer les types d'objectifs.	1	élevée

TABLE 3.1 – Backlog du sprint 1

3.3 Spécification fonctionnelle

3.3.1 Diagrammes de cas d'utilisation générale du Sprint 1

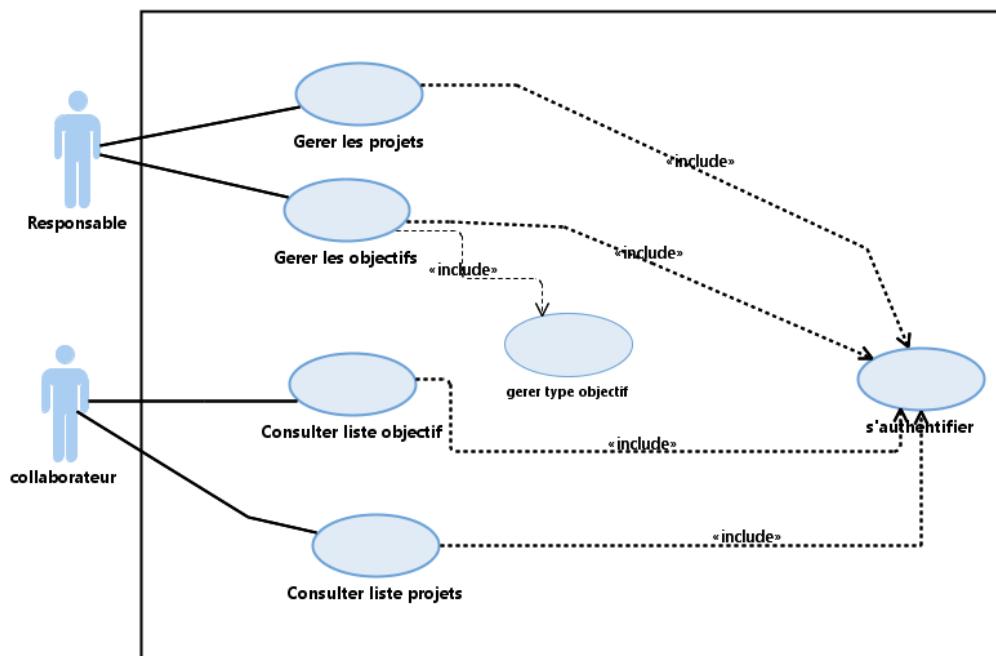


FIGURE 3.1 – Diagramme du cas d'utilisation générale du sprint 1

3.3.2 Diagramme de cas d'utilisation détaillé

Pour présenter un diagramme de cas d'utilisation détaillé, nous le divisons en sous-diagramme. Chacun présente un raffinement d'un cas d'utilisation.

Raffinement du cas d'utilisation gérer projet

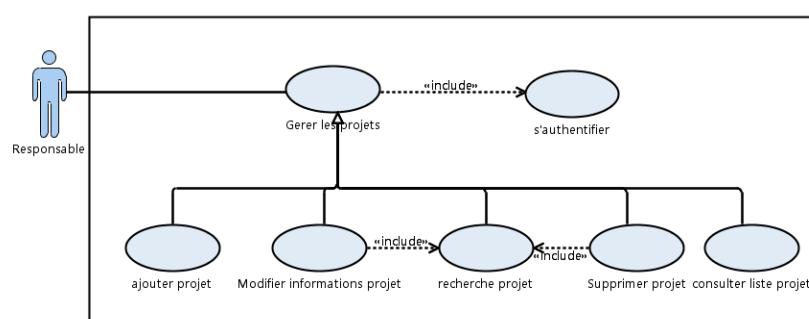


FIGURE 3.2 – Raffinement du cas d'utilisation "gérer projet"

Raffinement de cas d'utilisation gérer les objectifs

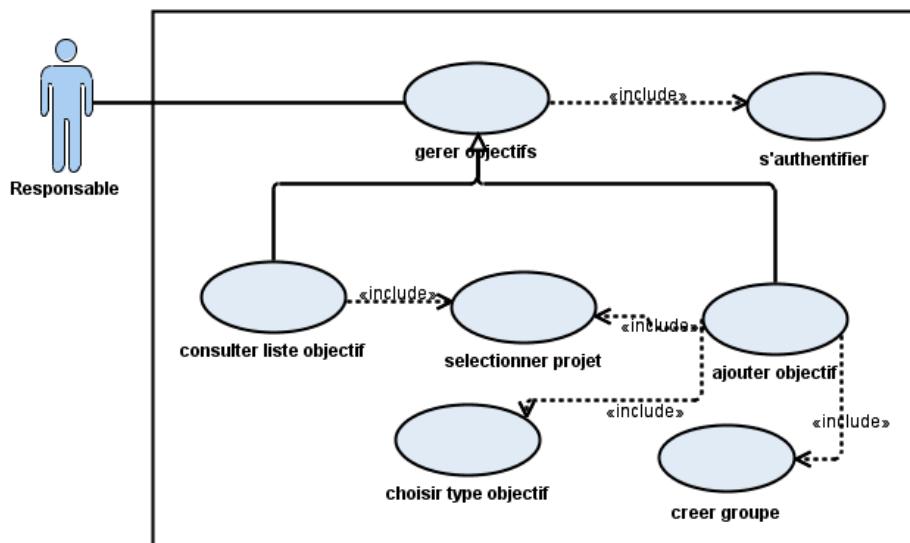


FIGURE 3.3 – Raffinement du cas d'utilisation "gérer objectifs"

3.4 Spécification des cas d'utilisations

Spécification du cas d'utilisation "s'authentifier"

Titre :S'authentifier	
Résumé : L'acteur s'authentifie en utilisant son login et mot de passe	
Acteur : Responsable ,Collaborateur , Client	
Pré-condition :	L'application accessible
Post-condition :	Utilisateur authentifié
Scénario de base :	<ol style="list-style-type: none"> 1. Le système donne la main à l'utilisateur pour introduire son Code Utilisateur et son mot de passe 2. L'utilisateur introduit son code utilisateur et mot de passe. 3. L'utilisateur valide la connexion. 4. Le système vérifie la combinaison code utilisateur et mot de passe. 5. Le système autorise l'accès approprié à l'utilisateur
Scénario d'exception :	<p>3.a. Si L'utilisateur ne remplit pas les deux données :</p> <ol style="list-style-type: none"> 1.Le système n'autorise pas l'utilisateur de valider la connexion. 2.Retour à l'étape 1 du scénario de base. <p>3.b. si la combinaison code utilisateur et mot de passe est incorrecte :</p> <ol style="list-style-type: none"> 1.Le système informe l'utilisateur que le code utilisateur ou le mot de passe sont invalides et la connexion a échoué. 2. Retour à l'étape 1 de scénario de base.

TABLE 3.2 – Spécification du cas d'utilisation « S'authentifier»

Spécification du cas d'utilisation "ajouter projet"

Titre : Ajouter projet	
Résumé : le responsable ajoute les informations du projet	
Acteur : Responsable	
Pré-condition :	L'application est en service, le responsable est authentifié via le système et il est associé à un sous service bien déterminé à travers l'authentification
Post-condition :	Projet ajouté
Scénario de base :	<ol style="list-style-type: none"> 1. Le responsable demande l'ajout du projet 2. L'application lui affiche la formulaire d'ajout d'un nouveau projet. 3. Le système donne la main au responsable pour introduire les données du nouveau projet 4. Le responsable introduit les données du projet 5. Le responsable valide l'ajout du projet. 6. Le système vérifie la validité des données introduites. 7. Le système enregistre le nouveau projet. 8. Le système affiche la liste des projets actualisée avec un message d'enregistrement avec succès.
Scénario d'exception :	<p>6.a. Si une donnée manquante ou invalide :</p> <ol style="list-style-type: none"> 1. Système affiche une erreur indiquant que cette donnée est obligatoire ou invalide. 2. Système retourne à l'étape 4 du scénario de base <p>6.b. Si une donnée valide :</p> <ul style="list-style-type: none"> - Passer à l'enregistrement du projet.

TABLE 3.3 – Spécification du cas d'utilisation "ajouter projet"

Spécification du cas d'utilisation "supprimer projet"

Titre :supprimer projet	
Résumé : Le responsable a la possibilité de supprimer un projet	
Acteur : Responsable	
Pré-condition :	Le responsable est authentifié et au moins un projet existant
Post-condition :	Projet supprimé
Scénario de base :	<ol style="list-style-type: none"> 1.«include» rechercher projet 2. Le responsable choisit l'action de suppression. 3. Le système demande de confirmer la suppression du projet 4. Le responsable valide la suppression du projet 5. Le système supprime le projet 6. Le système affiche la liste des projets actualisée avec un message de suppression avec succès
Scénario d'exception :	<ol style="list-style-type: none"> 4.a. Si le responsable annule la suppression : <ol style="list-style-type: none"> 1. La suppression ne s'effectue pas 2. Le système affiche la liste des projets

TABLE 3.4 – Spécification du cas d'utilisation "supprimer projet"

3.4.1 Spécification du cas d'utilisation "modifier information projet"

Titre :Modifier Information projet	
Résumé : Le responsable a la possibilité de modifier les données d'un projet	
Acteur : Responsable	
Pré-condition :	Le responsable est authentifié et au moins un projet existant
Post-condition :	Les données du projet sont mises à jour
Scénario de base :	<p>1. «include» rechercher projet</p> <p>2. Le responsable choisit l'action de modification des données</p> <p>3. Le système affiche le formulaire de modification des données du projet</p> <p>4. Le système donne la main à le responsable pour introduire les nouvelles données de projet.</p> <p>5. Le responsable introduit les nouvelles données du projet</p> <p>6. Le responsable enregistre les données entrées.</p> <p>7. Le système vérifie la validité des données introduites.</p> <p>8. Le système enregistre la modification du projet</p> <p>9. Le système informe le responsable que les modifications ont été effectuées avec succès.</p> <p>10. Le système affiche les informations du projet mises à jour</p>
Scénario d'exception :	<p>6.a. Si Le responsable choisit l'action d'annulation :</p> <p>1. Le système annule la modification</p>

spécification du cas d'utilisation consulter projets

Titre :consulter listes projets	
Résumé : Le collaborateur a la possibilité de consulter la liste des projets	
Acteur : Responsable ou collaborateur	
Pré-condition :	L'acteur est authentifié
Post-condition :	La liste des projets est affichée
Scénario de base :	<ol style="list-style-type: none">1.Le collaborateur demande la consultation de la liste des projets2. Le système affiche la liste des projets

TABLE 3.6 – Spécification du cas d'utilisation consulter liste projet

Spécification du cas d'utilisation consulter listes types objectifs

Titre :consulter listes types objectifs	
Résumé :Le Responsable ou Le collaborateur a la possibilité de consulter la liste des types des objectifs	
Acteur : Responsable et collaborateur	
Pré-condition :	L'utilisateur ou Le collaborateur est authentifié
Post-condition :	La liste des objectifs est affichée
Scénario de base :	<ol style="list-style-type: none"> 1.L'acteur demande la consultation de la liste des types objectifs 2. Le système affiche la liste des types objectifs

TABLE 3.7 – Spécification du cas d'utilisation consulter liste types objectifs

3.5 Conception

3.5.1 Diagrammes de séquences détaillés

Cas d'utilisation s'authentifier

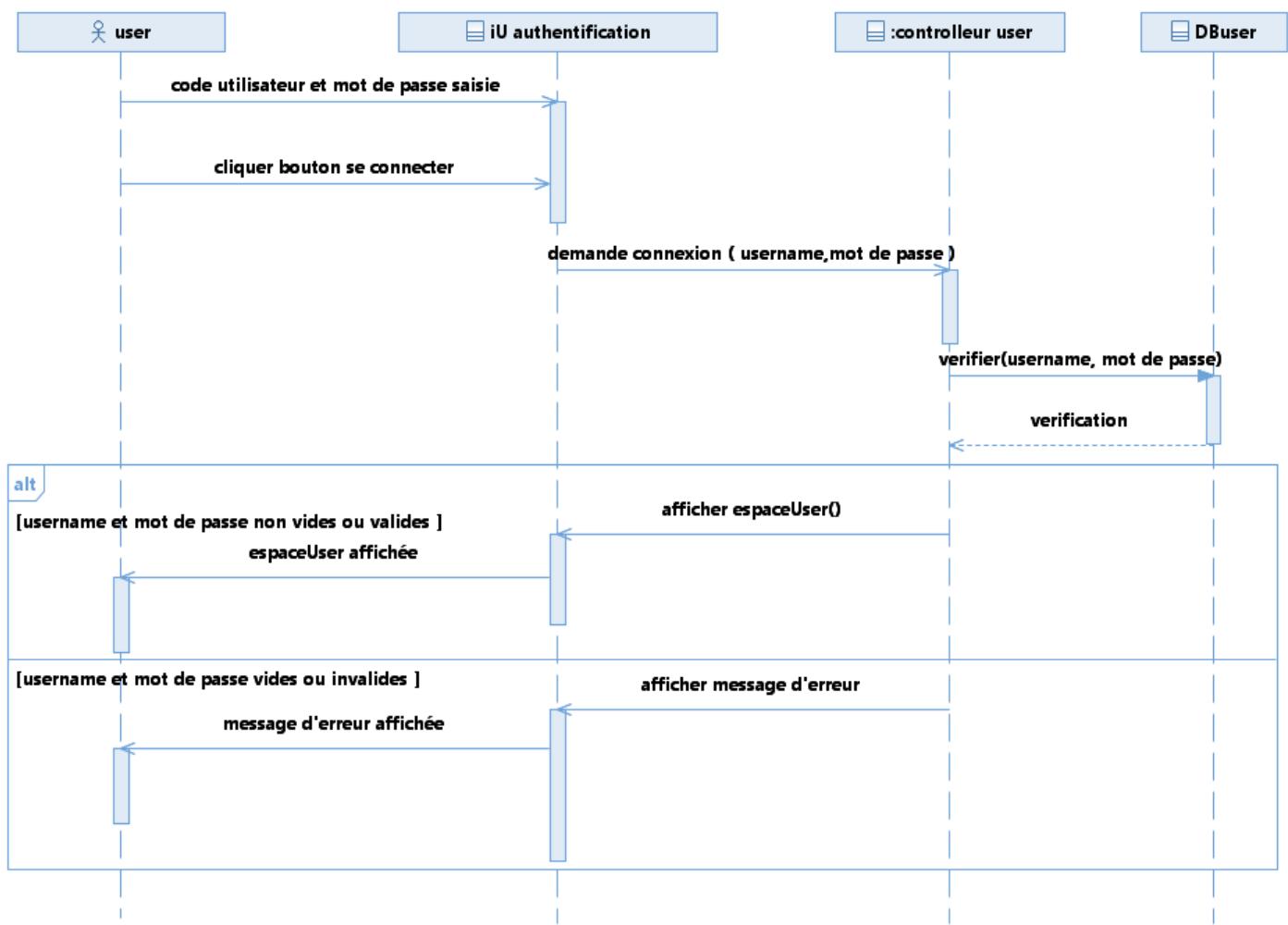


FIGURE 3.4 – Diagrammes de séquences cas d'utilisation "s'authentifier"

Diagramme de séquences du cas d'utilisation «Recherche Projet»

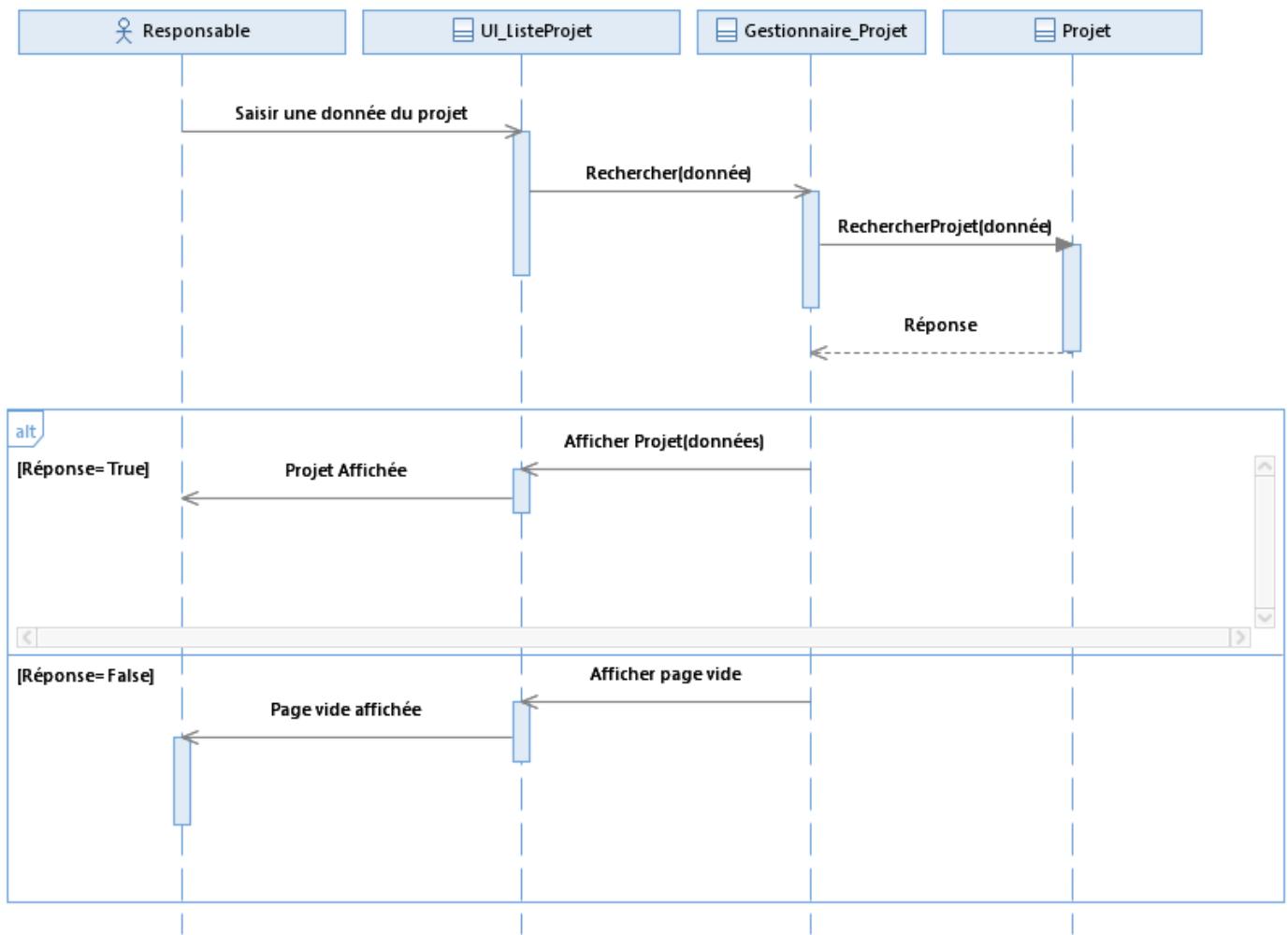


FIGURE 3.5 – Diagrammes de séquences cas d'utilisation "Recherche Projet"

Diagramme de séquences du cas d'utilisation «ajouter Projet»

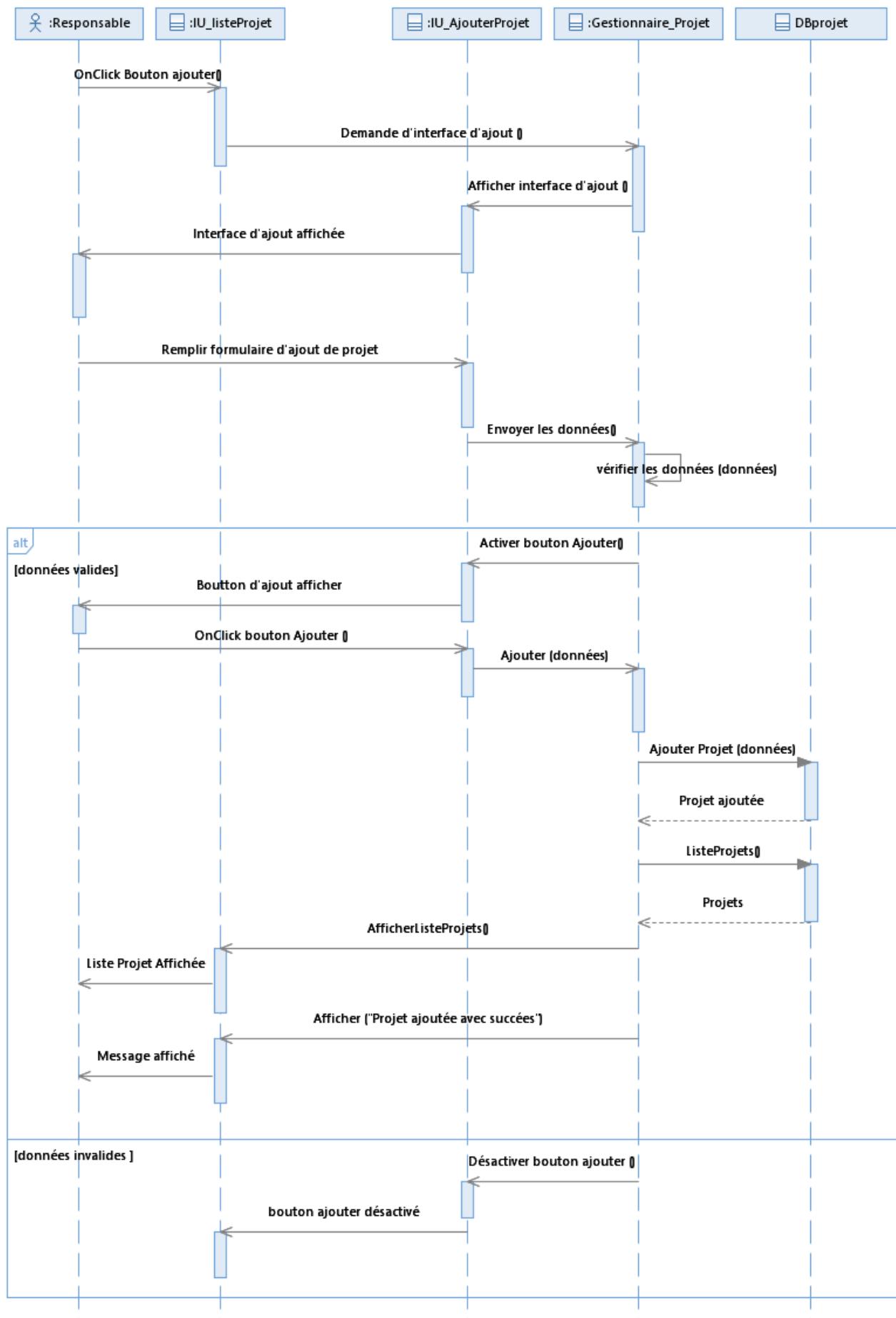


FIGURE 3.6 – Diagrammes de séquences cas d'utilisation "Ajouter Projet"

Diagramme de séquences du cas d'utilisation «Modifier type objectif»

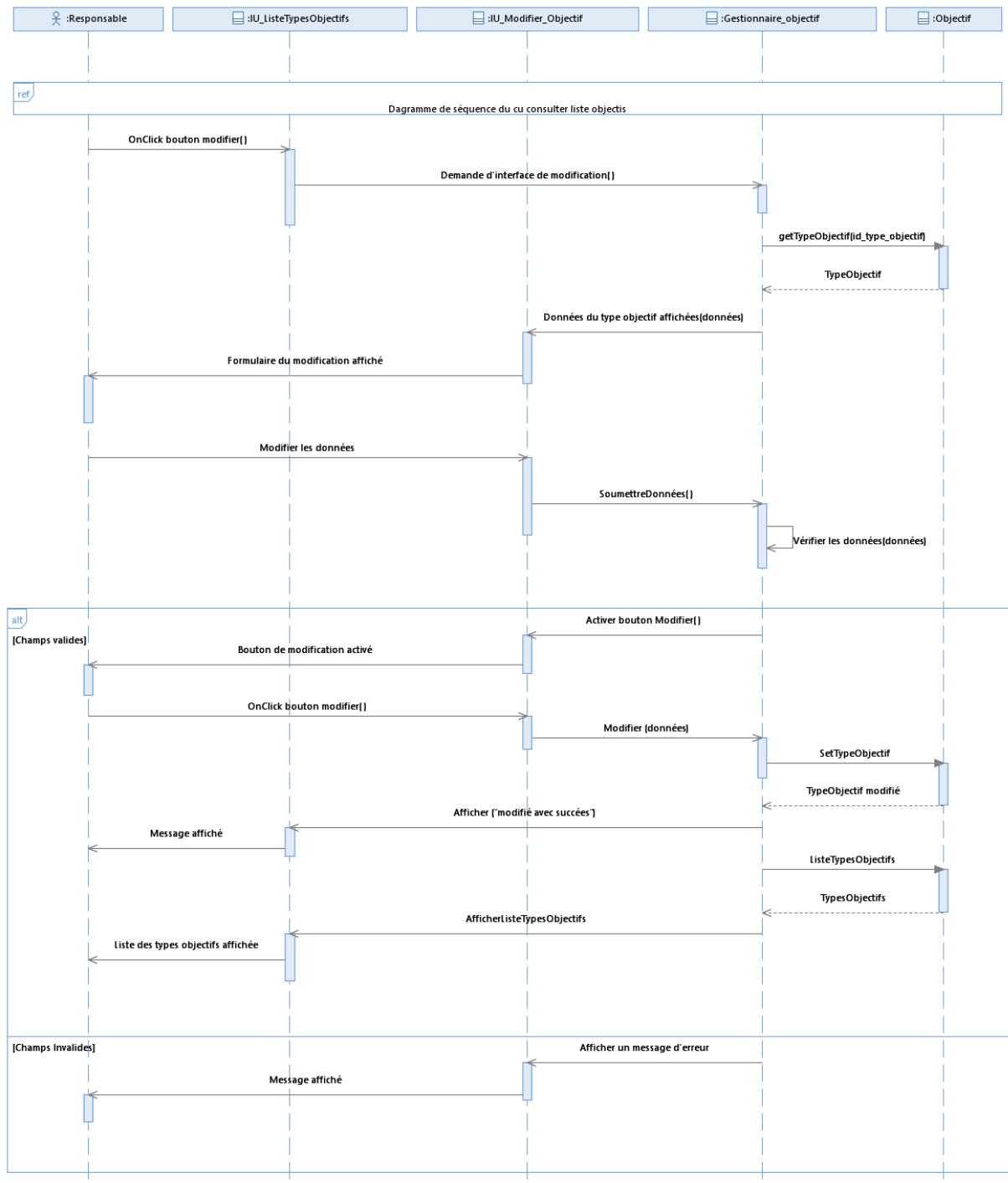


FIGURE 3.7 – Diagrammes de séquences cas d'utilisation "modifier Type objectifs"

3.5.2 Diagramme de Classe

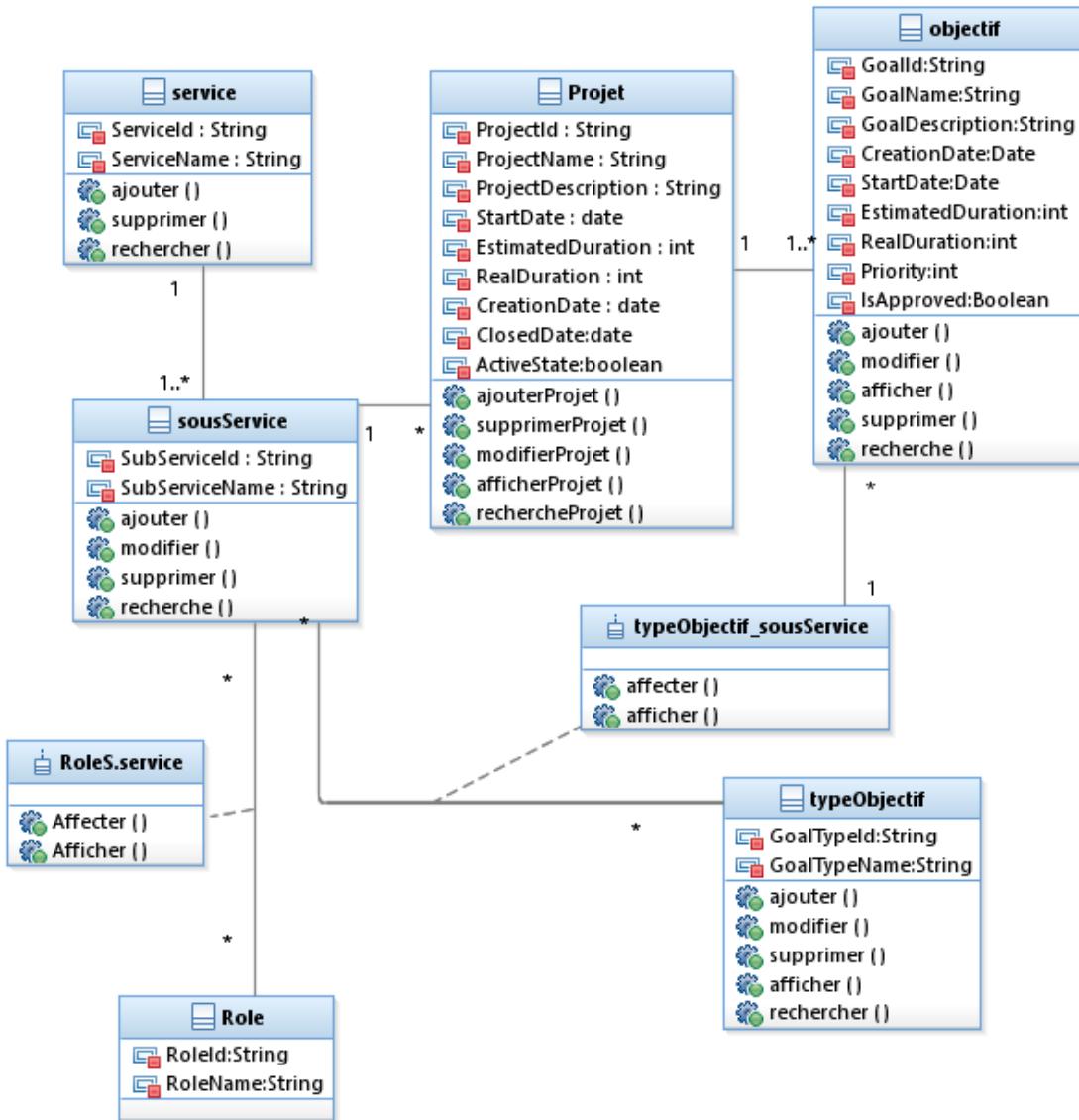


FIGURE 3.8 – Diagramme de classe Sprint 1

3.6 Réalisation

Dans cette partie nous présentons des captures d'écran des principales interfaces de notre application.

3.6.1 Interface d'authentification



FIGURE 3.9 – Interface "Authentification"

3.6.2 Crédation d'un nouveau projet ou modification d'un projet existant

FIGURE 3.10 – Interface "crédation d'un nouveau projet ou modifier un projet existant"

3.6.3 Consultation de toutes les informations d'un projet

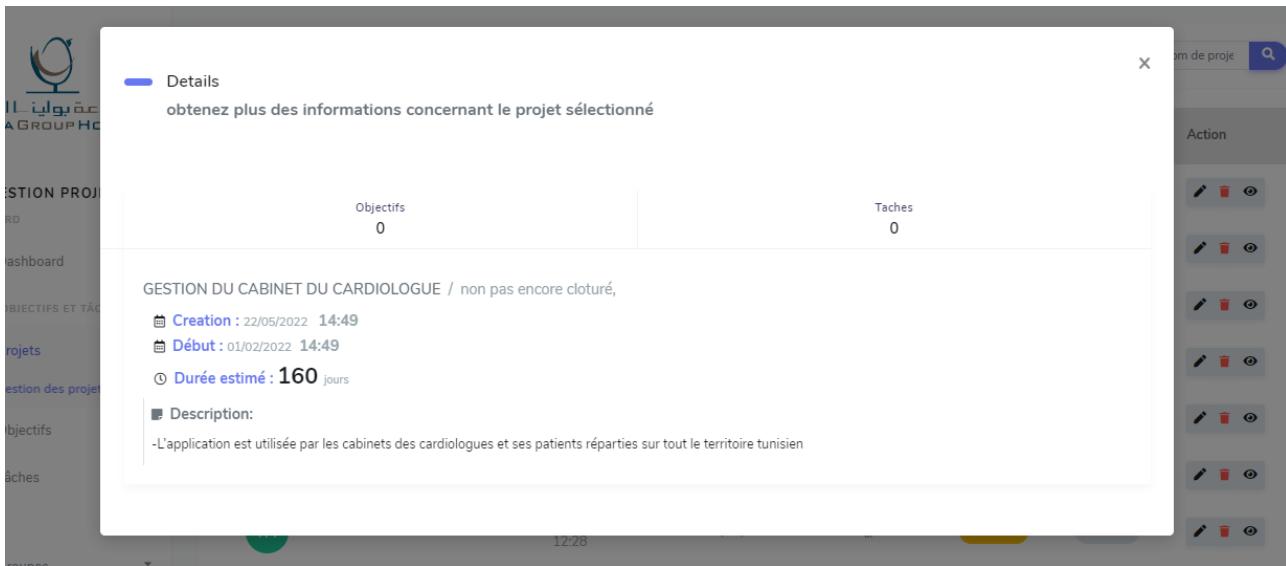


FIGURE 3.11 – Interface "Consultation de toutes les informations d'un projet"

3.6.4 Consulter liste type objectifs

The screenshot shows a list of objective types in a project management application. The table has two columns: "Nom" (Name) and "Action".

Nom	Action
conception	[Edit] [Delete]
réalisation	[Edit] [Delete]
Migration	[Edit] [Delete]
Front	[Edit] [Delete]
Back	[Edit] [Delete]

At the bottom, there are navigation links: « Previous » (disabled), a page number (1), and « Next ».

FIGURE 3.12 – Interface "Consultation du liste des types d'objectifs"

3.6.5 Consultation de toutes les informations d'un projet



FIGURE 3.13 – Interface "Consultation de toutes les informations d'un projet"

3.7 Test sprint 1

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 développées. Pour ce fait, nous allons présenter quelques scénarios de test ainsi que les captures d'écran de tests fonctionnels correspondants relatifs au sprint 1.

3.7.1 Cas de test d'authentification

Nous avons ajouté une responsable qui s'appelle « Zaabi Manel », un collaborateur qui s'appelle « Bennour haifa », une cliente qui s'appellent « Ben ayed Mariem ». Le Responsable a le droit de gérer les projets comme illustre dans la figure suivante.

Interface d'un responsable

The screenshot shows the 'Responsable' dashboard. At the top right, it says 'MANEL ZAABI -- Responsable'. Below the header, there's a sidebar with 'GESTION PROJET' and 'DASHBOARD' sections, and a 'PROJETS OBJECTIFS ET TÂCHES' section containing 'Projets', 'Objectifs', 'Tâches', 'Groupes', and 'Réunions'. The main area is titled 'Liste des Projets' with the sub-instruction 'ceci est la liste de tous les projets.' It features a blue 'Ajouter un projet' button and a search bar. A table lists five projects:

Nom	Date Création	Date Début	Durée estimé()	Etat	Statut	Action
GD Gestion De Projet	18/01/2022 15:16	28/01/2022	21(j)	En cours	Active	edit trash eye
CD Calcul De Stock	10/01/2022 15:18	Pas encore	22(j)	En cours	Active	edit trash eye
FF Foncier	12/01/2022 09:07	Pas encore	3(j)	En cours	Active	edit trash eye
PP PFE	02/06/2022 11:41	01/02/2022	200(j)	En cours	Active	edit trash eye
AA ABCD	02/06/2022 11:42	01/02/2020	10(j)	En cours	Inactive	edit trash eye

FIGURE 3.14 – Interface du "responsable"

Interface d'un collaborateur

The screenshot shows the 'Collaborateur' dashboard. At the top right, it says 'HAIFA BENNOUR -- Collaborateur'. Below the header, there's a sidebar with 'GESTION PROJET' and 'DASHBOARD' sections, and a 'PROJETS OBJECTIFS ET TÂCHES' section containing 'Projets', 'Objectifs', 'Tâches' (with a note 'Gestion des tâches collaborateur'), and 'Réunions'. The main area is titled 'Liste des Projets' with the sub-instruction 'ceci est la liste de tous les projets.' It features a search bar. A table lists the same five projects as the responsible interface:

Nom	Date Création	Date Début	Durée estimé()	Etat	Statut	Action
GD Gestion De Projet	18/01/2022 15:16	28/01/2022	21(j)	En cours	Active	eye
CD Calcul De Stock	10/01/2022 15:18	Pas encore	22(j)	En cours	Active	eye
FF Foncier	12/01/2022 09:07	Pas encore	3(j)	En cours	Active	eye
PP PFE	02/06/2022 11:41	01/02/2022	200(j)	En cours	Active	eye
AA ABCD	02/06/2022 11:42	01/02/2020	10(j)	En cours	Inactive	eye

FIGURE 3.15 – Interface du "collaborateur"

Interface d'un client

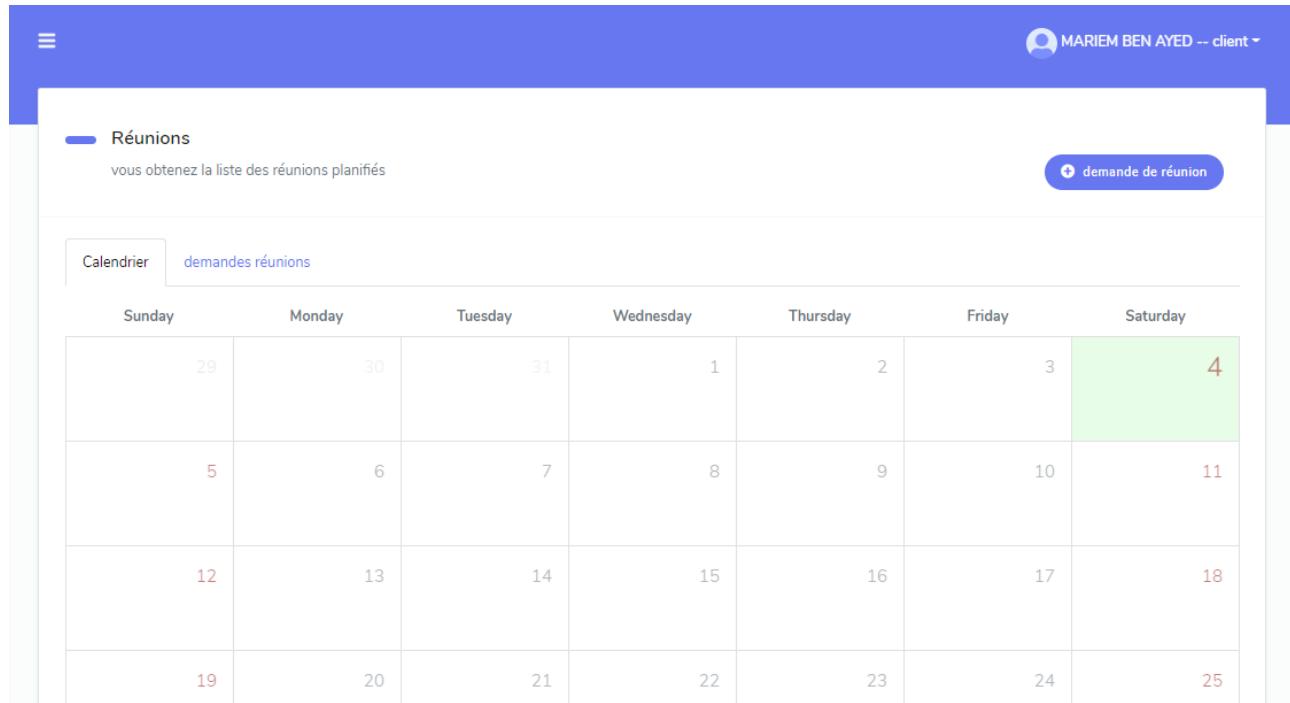


FIGURE 3.16 – Interface du "client"

3.7.2 Cas de test d'ajout d'un projet

Interface d'ajout d'un projet

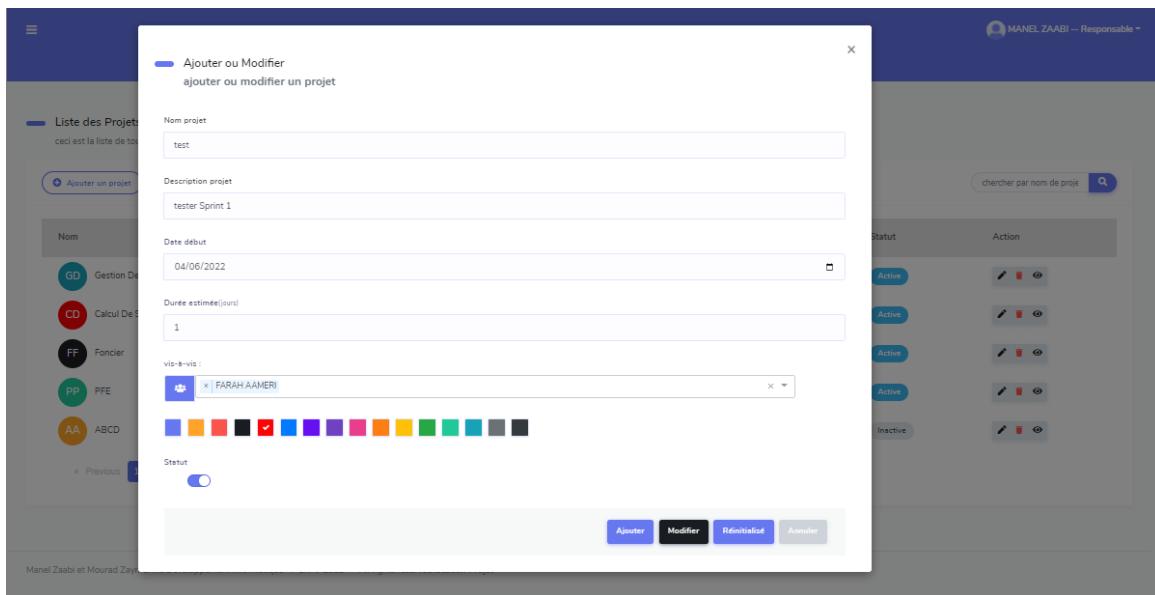


FIGURE 3.17 – Interface "ajout d'un projet"

Interface d'ajout d'un projet avec succès

Nom	Date Création	Date Début	Durée estimé()	Etat	Statut	Action
Gestion De Projet	04/06/2022 14:17	04/06/2022	21()	En cours	Active	

FIGURE 3.18 – Interface "ajout un projet avec succès"

3.7.3 Cas de test Affecter objectif à un projet

Interface d'affectation objectif à un projet

The screenshot shows a web-based project management application. At the top, there's a blue header bar with a user profile icon and the name "MANEL ZAABI -- Responsable". Below the header, a sidebar on the left lists "Objectifs" and "Sélectionner projet" with a dropdown menu set to "gestion de projet". A central table displays four rows of objective data:

Nom	Approuvé	Date Création	Date Début	Durée estimé(heure(s))	Action
Réalisation	Pas encore	04/06/2022 14:20	Pas encore	1	
Conception	Pas encore	04/06/2022 14:21	Pas encore	1	
Conception	Approuvé	04/06/2022 14:33	04/01/2022 14:18	8	
Elaboration cahier de charge	Approuvé	04/06/2022 14:33	04/01/2022 14:18	13	

A small blue button labeled "Ajouter objectif" is located in the top right corner of the table area.

FIGURE 3.19 – Interface "affectation objectif à un projet"

3.8 Scrum Board

Nous présentons ci-dessous le scrum board de sprint 1 au cours de la 2ème semaine de sa période de réalisation.

The screenshot shows a Trello Scrum Board titled "sprint1". The board has three columns: "To Do", "Doing", and "Done".

- To Do:** Contains one card: "gestion des objectifs".
- Doing:** Contains two cards: "Gestion de projet" (due 29 mai) and "Authentification" (due 25 mai).
- Done:** Contains one card: "Authentification".

The left sidebar shows the project structure: "PFE Gratuit" with "Tableaux", "Membres", and "Paramètres". Under "Vues de l'espace de travail", "sprint1" is selected. The bottom left corner features a purple button for "Essayer Premium gratuitement".

FIGURE 3.20 – Scrum Board Sprint 1

3.9 Conclusion

Au cours de ce chapitre, nous avons présenté le premier sprint. Pour ce faire, nous sommes passées par l'analyse, la conception et la réalisation. Dans le chapitre suivant nous entamons le deuxième sprint.

Chapitre 4

Etude et réalisation du sprint 2

4.1 Introduction

Dans cette partie, nous allons présenter les différents aspects du Sprint 2 qui inclut la gestion des tâches et la gestion des tableaux de bord.

4.2 Sprint Backlog

ID	User Story	priorité	complexité
1	En tant qu'utilisateur, je veux visualiser les tableaux de bord	1	moyenne
2	En tant que responsable , je veux créer une tâche	1	élevée
3	En tant que responsable , je veux modifier une tâche	1	élevée
4	En tant que responsable , je veux supprimer une tâche	1	élevée
5	En tant qu'utilisateur , je veux consulter les informations relatives à une tâche	2	faible
6	En tant que collaborateur, je veux consulter la liste des tâches	2	faible
7	En tant que responsable , je veux ajouter un type de tâche	1	élevée
8	En tant que responsable , je veux supprimer un type de tâche	1	élevée
9	En tant que responsable, je veux modifier un type de tâche	2	faible

TABLE 4.1 – Backlog du sprint 2

4.3 Spécification fonctionnelle

4.3.1 Diagrammes de cas d'utilisation générale du Sprint 2

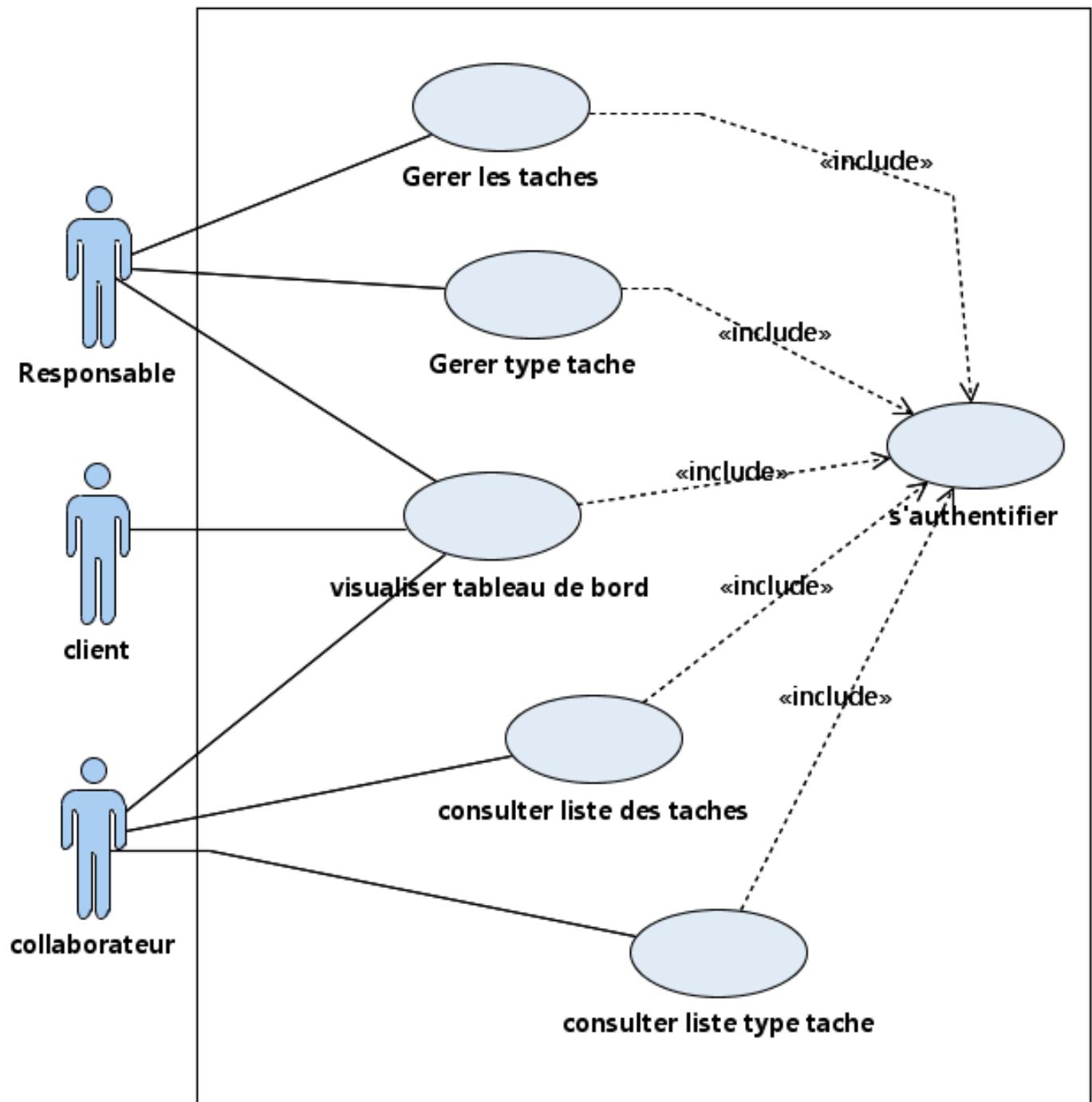


FIGURE 4.1 – Diagramme du cas d'utilisation générale du sprint 2

4.3.2 Diagramme de cas d'utilisation détaillé

Pour présenter un diagramme de cas d'utilisation détaillé, nous le divisons en sous-diagramme. Chacun présente un raffinement d'un cas d'utilisation.

Raffinement du cas d'utilisation "gérer tâche"

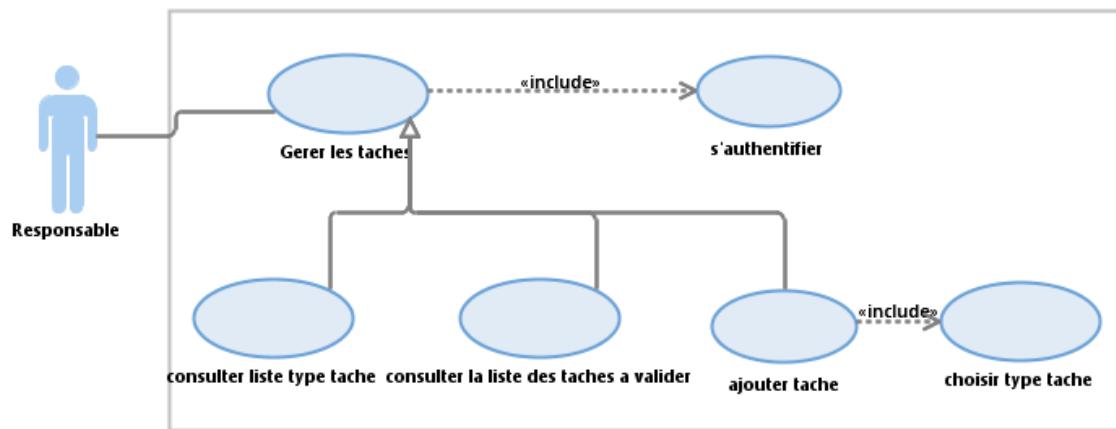


FIGURE 4.2 – Raffinement du cas d'utilisation "gérer tâche"

Raffinement du cas d'utilisation "consulter liste tâche à valider"

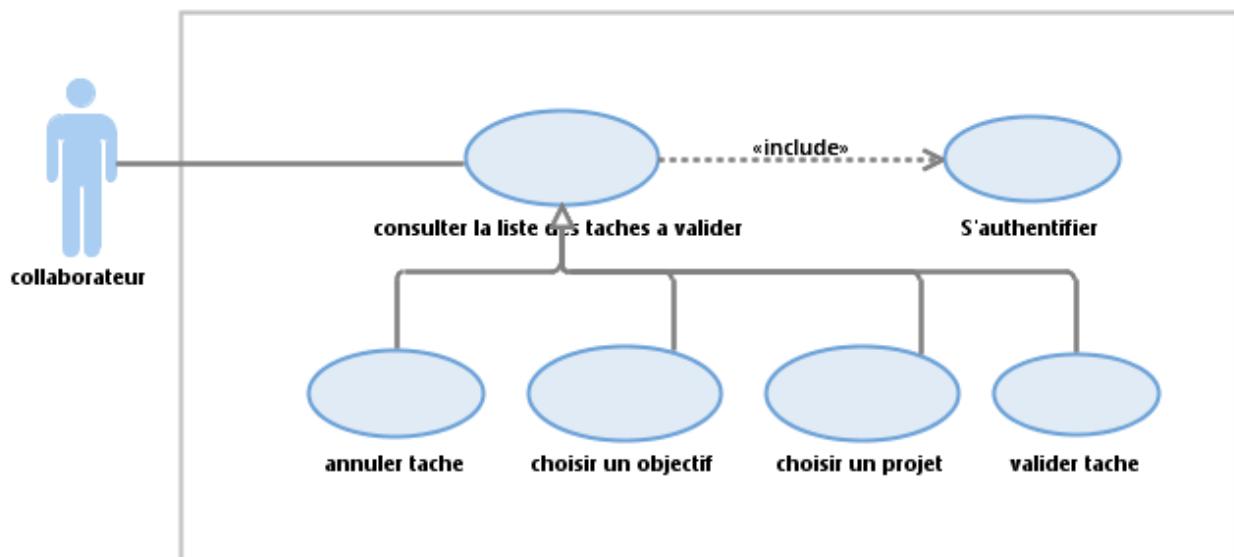


FIGURE 4.3 – Raffinement du cas d'utilisation "consulter liste tâche à valider"

Raffinement du cas d'utilisation "consulter liste type tâche"

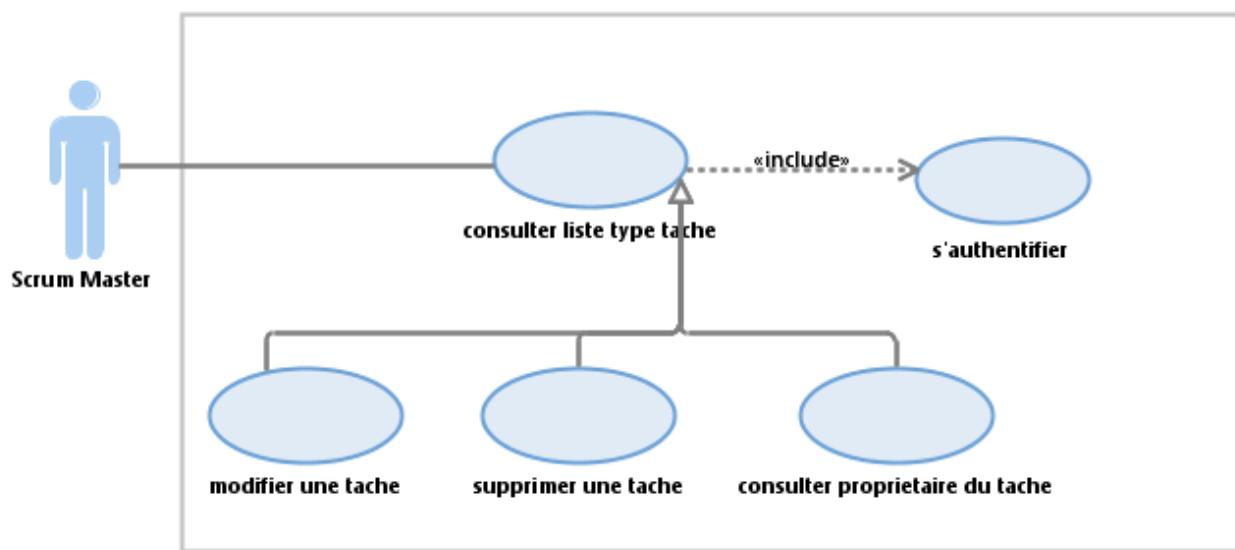


FIGURE 4.4 – Raffinement de cas d'utilisation "consulter liste type tâche"

Raffinement du cas d'utilisation "visualiser tableau de bord"

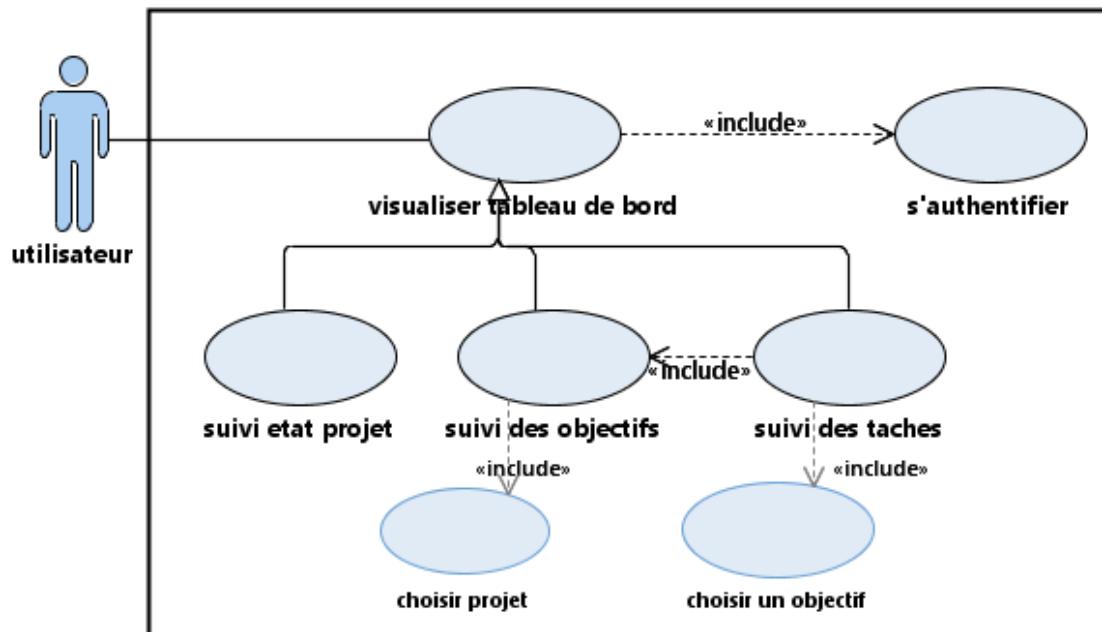


FIGURE 4.5 – Raffinement de cas d'utilisation "visualiser tableau de bord"

4.4 Spécification des cas d'utilisations

spécification du cas d'utilisation "ajouter tâche"

Titre : Ajouter tâche	
Résumé : le responsable ajoute une tâche	
Acteur : Responsable	
Pré-condition :	L'application est en service, le responsable est authentifié via le système et il est associé à un sous service bien déterminé à travers l'authentification
Post-condition :	Projet ajouté
Scénario de base :	<p>1. Le responsable demande l'ajout du tâche</p> <p>2.L'application lui affiche la formulaire d'ajout d'un nouveau tâche.</p> <p>3. Le système donne la main à le responsable pour introduire les données au formulaire</p> <p>4. Le responsable introduit les données du tâche</p> <p>5. le responsable valide l'ajout du tâche.</p> <p>6. Le système vérifie la validité des données introduites.</p> <p>8.Le système enregistre la nouvelle tâche.</p> <p>9. Le système affiche la liste la liste des tâche avec un message d'enregistrement avec succès.</p>
Scénario d'exception :	7.a. Si une donnée manquante ou invalide : 67

Spécification du cas d'utilisation "supprimer tâche"

Titre : supprimer tâche	
Résumé : Le responsable a la possibilité de supprimer une tâche	
Acteur : Responsable	
Pré-condition :	Le responsable est authentifié et au moins une tâche existante
Post-condition :	tâche supprimé
Scénario de base :	<p>1. «include» rechercher tâche</p> <p>2. Le responsable choisit l'action de suppression.</p> <p>3. Le système demande de confirmer la suppression du tâche</p> <p>4. le responsable valide la suppression du tâche</p> <p>5. Le système supprime la tâche</p> <p>6. Le système affiche la liste des tâches actualisée avec un message de suppression avec succès</p>
Scénario d'exception :	<p>4.a. Si le responsable annule la suppression :</p> <p>1. La suppression ne s'effectue pas</p> <p>2. Le système affiche la liste des tâches</p>

TABLE 4.3 – Spécification du cas d'utilisation "supprimer tâche"⁶⁹

Spécification du cas d'utilisation "modifier information tâche"

Titre :Modifier information tâche	
Résumé : Le responsable a la possibilité de modifier les données d'une tâche	
Acteur : Responsable	
Pré-condition :	Le responsable est authentifié et au moins une tâche existante
Post-condition :	Les données du tâche sont mises à jour
Scénario de base :	<p>1. «include» rechercher tâche</p> <p>2. Le responsable choisit l'action de modification des données</p> <p>3. Le système affiche le formulaire de modification des données du tâche</p> <p>4. Le système donne la main à le responsable pour introduire les nouvelles données du tâche.</p> <p>5. Le responsable introduit les nouvelles données du tâche</p> <p>6. Le responsable enregistre les données entrées.</p> <p>7. Le système vérifie la validité des données introduites.</p> <p>8. Le système enregistre la modification du tâche</p> <p>9. Le système informe le responsable que les modifications ont été effectuées avec succès.</p> <p>10. Le système affiche les informations du tâche mises à jour</p>
Scénario d'exception :	<p>6.a. Si Le responsable choisit l'action d'annulation :</p> <p>1. Le système annule la modification</p>

Spécification du cas d'utilisation "visualiser tableau de bord"

Titre : Visualiser liste des tableaux de bord	
Résumé : L'acteur a la possibilité de visualiser la liste des tableaux de bord	
Acteur : Responsable ou collaborateur	
Pré-condition :	L'acteur est authentifié
Post-condition :	La liste des tableaux de bord est affichée
Scénario de base :	<ol style="list-style-type: none"> 1. L'acteur demande la visualisation de la liste des tableaux de bord 2. Le système affiche la liste des tableaux de bord

TABLE 4.5 – Spécification du cas d'utilisation "visualiser tableau de bord"

4.5 Phase de conception analytique de GIMSI

4.6 Conception

4.6.1 Diagrammes de séquences détaillés

Cas d'utilisation "consulter liste des tâches"

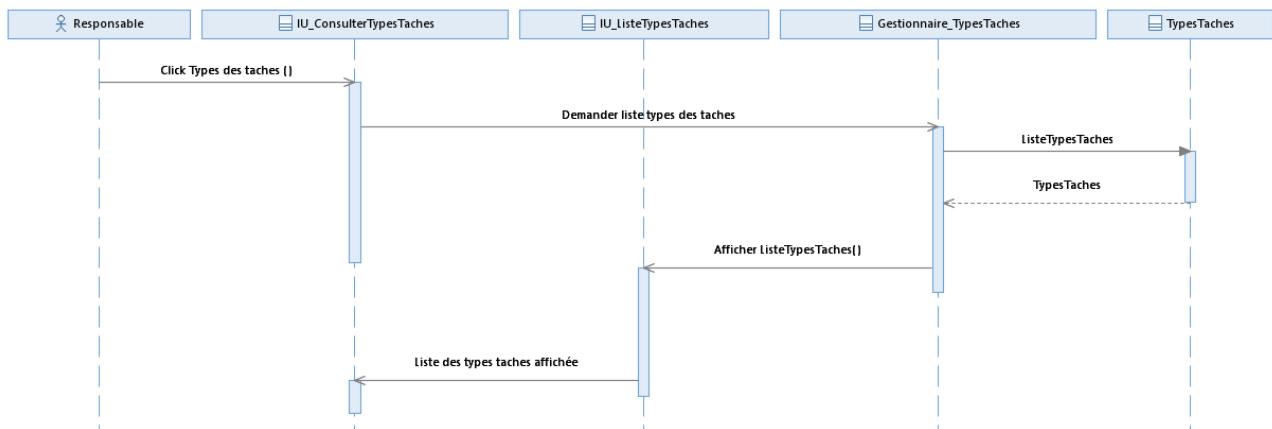


FIGURE 4.6 – Cas d'utilisation "consulter liste des tâches"

Cas d'utilisation "supprimer tâche"

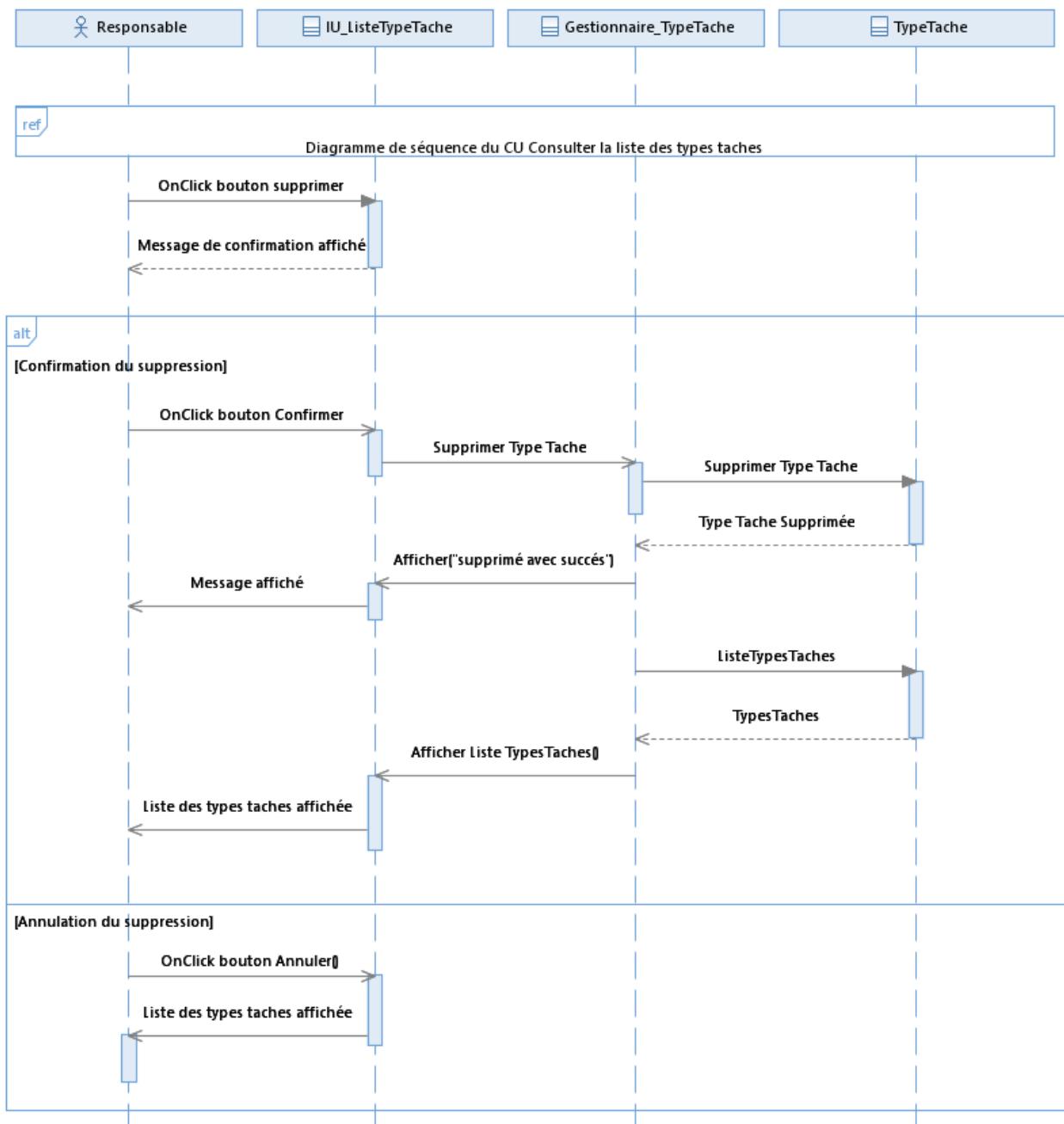


FIGURE 4.7 – Cas d'utilisation "supprimer tâche"

Cas d'utilisation "visualiser tableau de bord Suivi des objectifs "

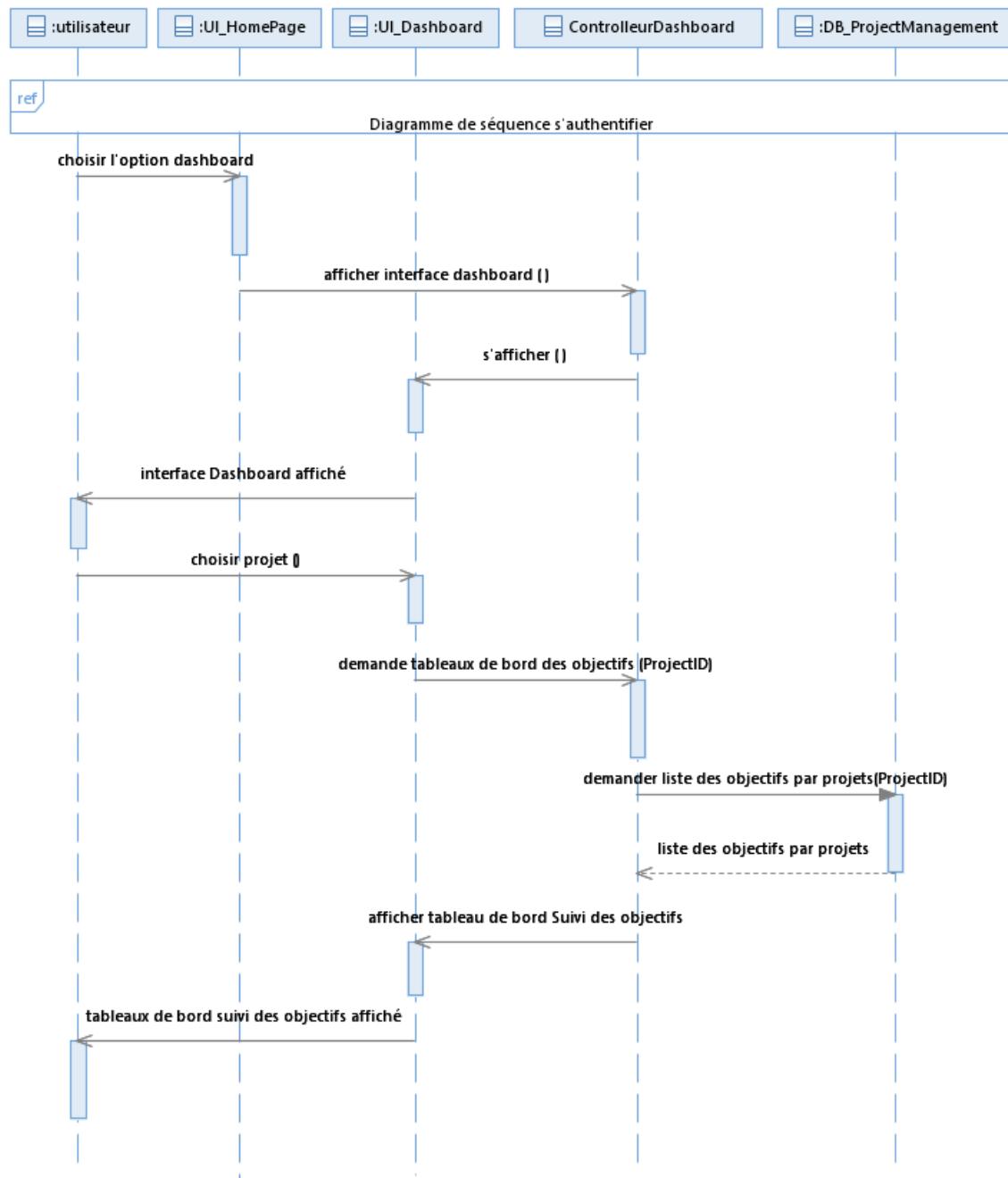


FIGURE 4.8 – Cas d'utilisation "visualiser tableau de bord Suivi des objectifs"

Cas d'utilisation "visualiser tableau de bord Suivi des taches "

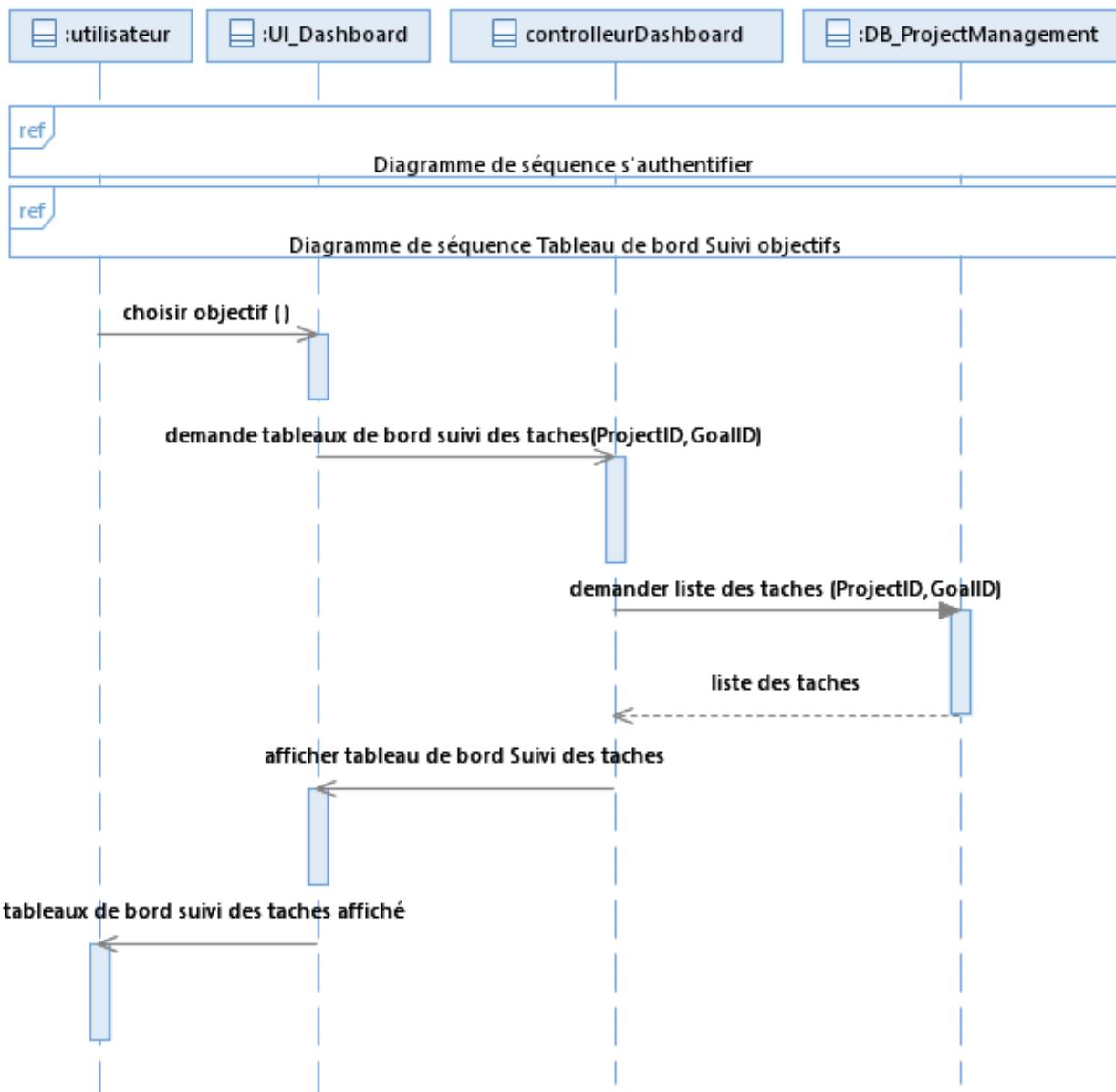


FIGURE 4.9 – Cas d'utilisation "visualiser tableau de bord Suivi des taches"

4.6.2 Diagramme de Classes



FIGURE 4.10 – Diagramme de classe sprint2

4.7 Conception Data Warehouse

4.7.1 Entrepôt de données

Un entrepôt de données ou un datawarehouse est une base de données non classique qui gère des données historiées qui permettent de faire des analyses en ligne ou de prendre des décisions. Les données d'un entrepôt sont intégrées périodiquement à l'entrepôt, ils ne sont pas directement modifiables, elles sont dérivées des données des bases opérationnelles (DataMart) qui régissent les activités de l'entreprise.

Par ailleurs, un Data Warehouse n'est pas une simple copie des données. En effet, les données y sont organisées et structurées selon 4 caractéristiques essentielles :

- Données orientées sujet
- Données intégrées
- Données historiées
- Données non-volatiles

4.7.2 Le magasin de données (Datamart)

Un « data mart » est une base de données dont le contenu est en rapport avec une activité de l'entreprise et qui est créée pour répondre aux besoins spécifiques d'un groupe d'utilisateurs. Il s'agit souvent (mais pas toujours) d'un segment partitionné dans le data warehouse de l'entreprise.

Les data marts accélèrent les processus en permettant de définir un accès aux données stockées dans un data warehouse (ou autre gisement de données opérationnelles) en quelques jours au lieu de plusieurs mois ou plus. Il s'agit d'une solution peu coûteuse qui permet de transformer les données en connaissance exploitable.

Nous avons décidé de concevoir des datamarts destinées pour le suivi d'état de projet, le suivi des objectifs en fonction des projets et le suivi des tâches .

le figure 4.11 présente la data mart "Suivi état projet "

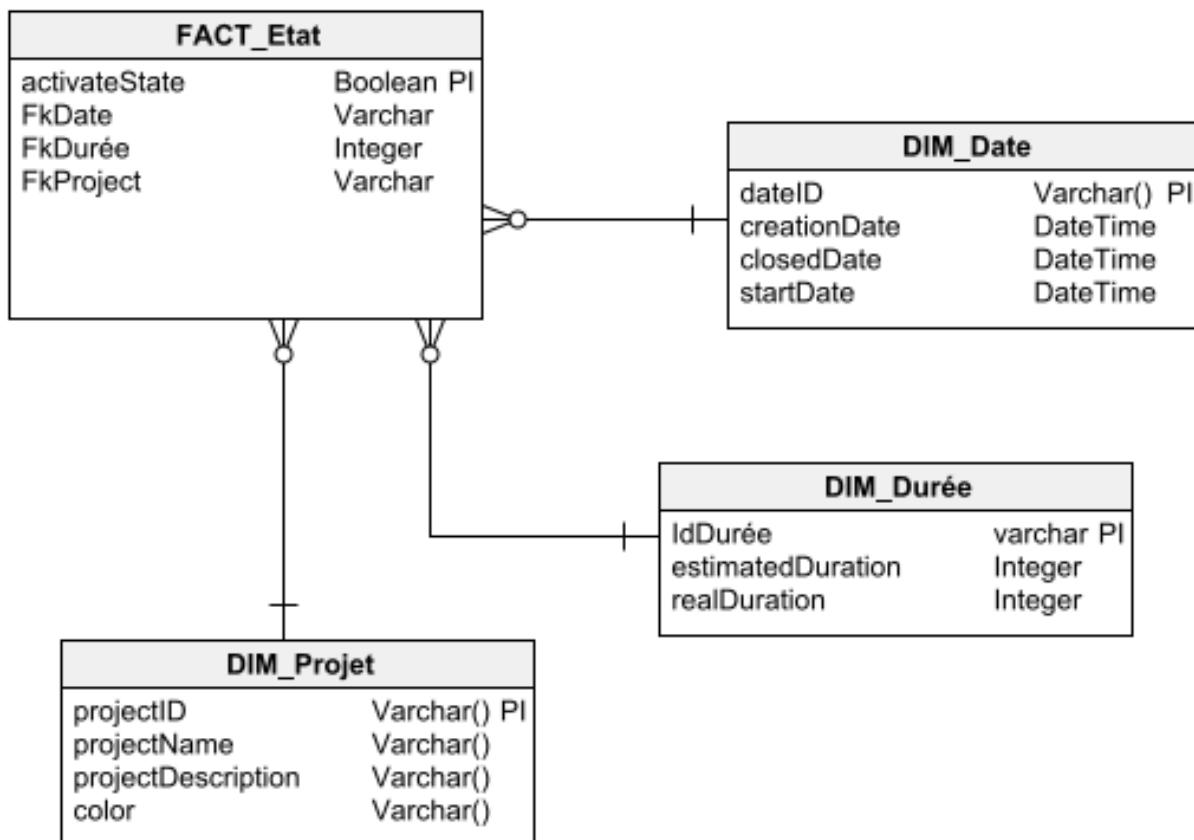


FIGURE 4.11 – Data mart "Suivi état projet"

Schéma physique

Structure de datamart « Suivi état projet»

- **FACT-Etat**(FkProject, FKDate, FkDurée, ActivateState)
- **DIM-Projet**(projectID, ProjectName, ProjectDescription, Color)
- **DIM-Date**(dateID, creationDate, closedDate, startDate)
- **DIM-Durée**(IdDurée, estimatedDuration, realDuration)

le figure 4.12 présente la data mart "Suivi objectifs "

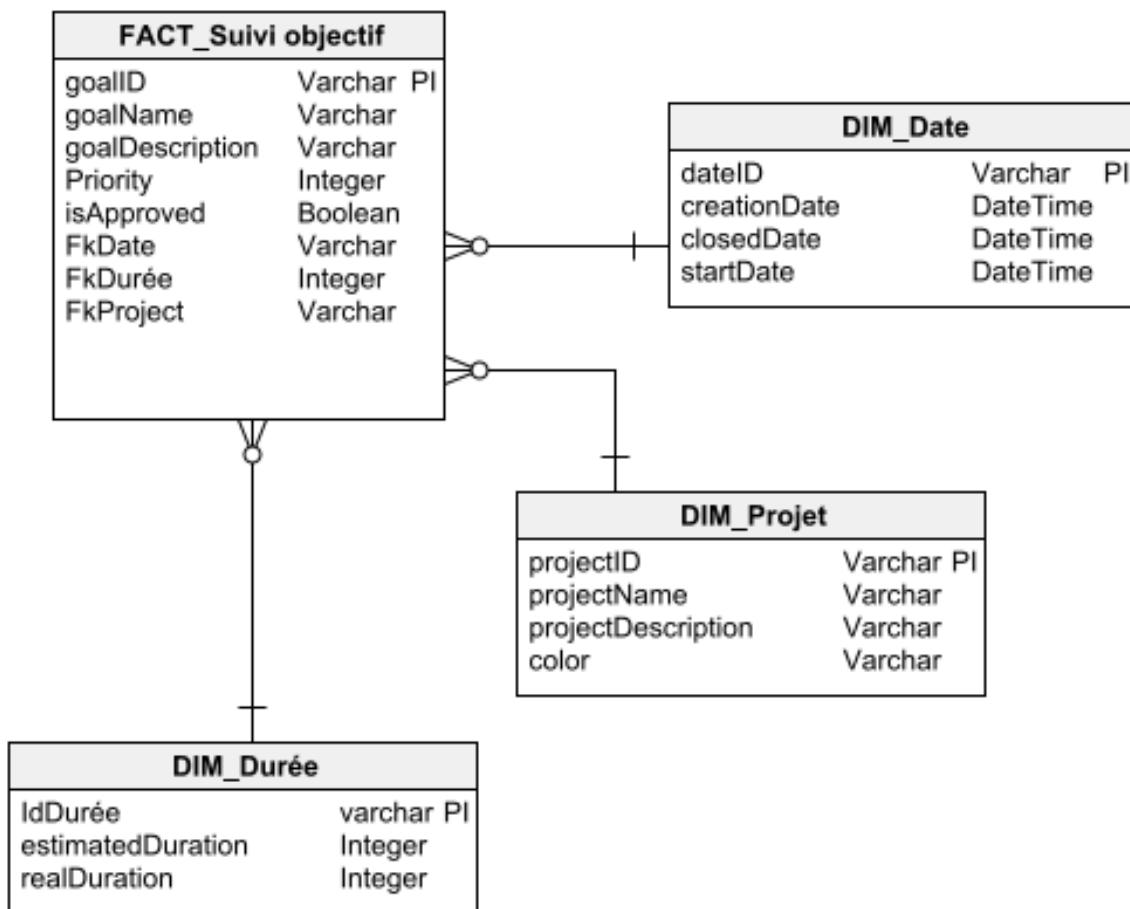


FIGURE 4.12 – Data mart "Suivi objectifs"

Schéma physique

Structure de data mart "Suivi objectifs"

- **FACT-Suivi objectif**(FkProject, FKDate, FkDurée, goalID, goalName, goalDescription, Priority, isApproved)
- **DIM-Projet**(projetID, ProjectName, ProjectDescription, Color)
- **DIM-Date**(dateID, creationDate, closedDate, startDate)
- **DIM-Durée**(idDurée, estimatedDuration, realDuration)

le figure 4.13 présente la data mart "Suivi taches "

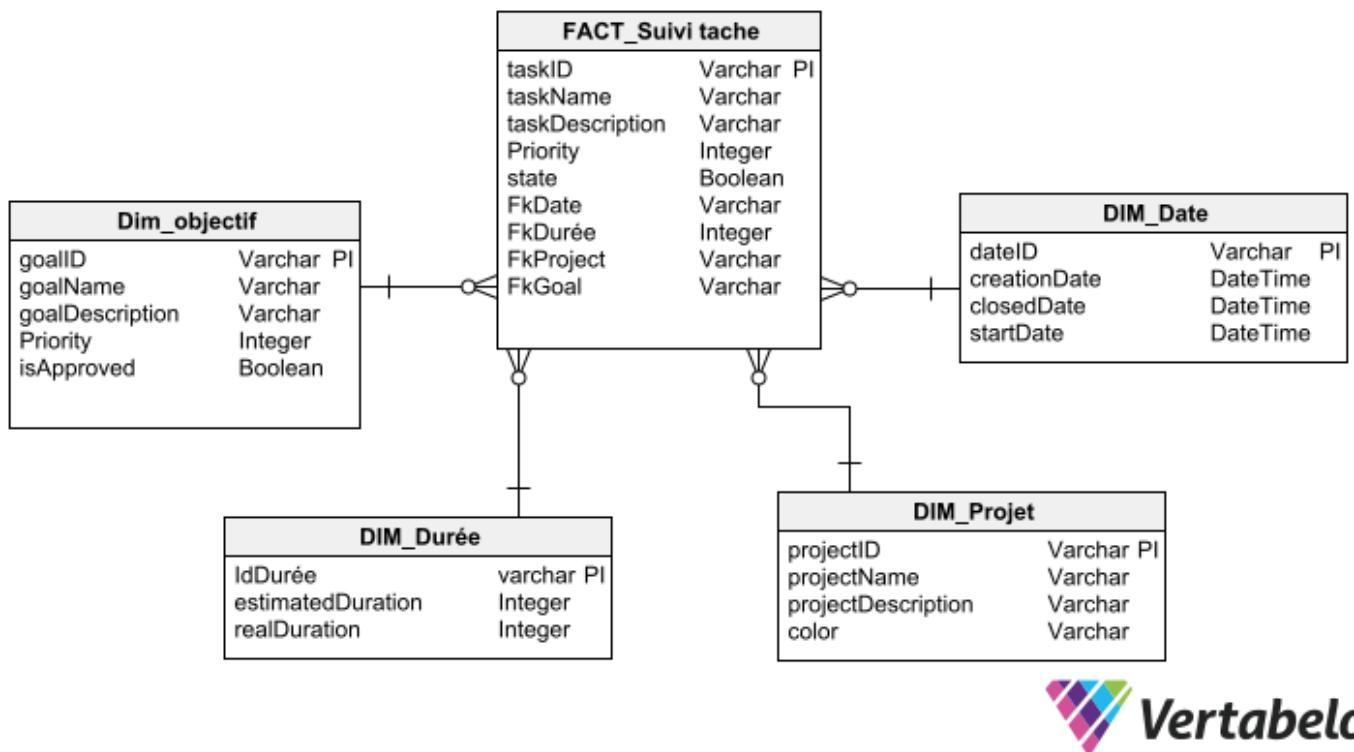


FIGURE 4.13 – Data mart "Suivi taches"

Schéma physique

Structure de Data mart "Suivi taches"

- **FACT-Suivi tache**(FkProject, FKDate, FkDurée, Fkgoal, taskID, taskName, taskDescription, Priority, state)
- **DIM-Projet**(projetID, ProjectName, ProjectDescription, Color)
- **DIM-Date**(dateID, creationDate, closedDate, startDate)
- **DIM-Durée**(IdDurée, estimatedDuration, realDuration)
- **DIM-objectif**(goalID, goalName, goalDescription, Priority, isApproved)

4.7.3 Choix des indicateurs

Parmi les étapes de la phase analytique de GIMSI , nous avons concentrés sur les choix des indicateurs.

Le choix des indicateurs représente la principale difficulté lors de l’élaboration d’un système de tableau de bord. Les indicateurs doivent en effet offrir une vision suffisante du système en fonction des objectifs choisis, et aussi ils doivent permettre aux utilisateurs de prendre les décisions nécessaire. Et comme nous l’avons fait pour le choix des objectifs, l’étape de choix des indicateurs se compose de deux parties :

- Les critères des indicateurs
- La sélection des indicateurs

4.7.4 Les critères des indicateurs

L’objectif de cet outil est de choisir les indicateurs les plus adéquats en fonction des objectifs poursuivis, de l’activité exercée, des comportements et des besoins propres du décideur. Le choix de l’indicateur sera effectué par les décideurs qui devant utiliser le tableau de bord.

Un bon indicateur doit répondre aux caractéristiques suivantes :

- Un indicateur doit être mesurable en temps réel.
- L’indicateur doit mesurer un ou plusieurs objectifs.
- L’indicateur doit pouvoir être présentée sur le poste de travail.
- L’indicateur doit être constructible.
- L’indicateur doit mener à l’action.

En effet, un indicateur clé de performance ou « Key Performance Indicator » est une mesure ou un ensemble de mesures braquées sur un aspect critique de la performance globale de l’organisation.

Le choix de la sélection de bons indicateurs représente la principale difficulté lors de l'élaboration d'un système de tableau de bord, puisqu'ils vont non seulement quantifier l'activité de l'équipe, mais doivent en effet offrir une vision suffisante du système en fonction des objectifs choisis.

4.7.5 Sélection des indicateurs

Ces indicateurs clés de performance sont dégagés lors de la phase de capture des besoins exprimés par les décideurs, qui se résument en :

- nombre des tâches (done , assignée ,..) par objectif, date, projet et durée.
- nombre des objectifs par projet, durée et date.
- état des projet par date et durée (clôturé , en cour,...)

4.8 Phase mise en œuvre de GIMSI

4.8.1 Intégration et déploiement

Dans cette partie, nous allons présenter en détail toutes les étapes de la mise en œuvre de système décisionnel c'est-à-dire nous nous intéressons à la partie technique de notre travail. Cette étape est principalement basée sur la phase d'extraction, transformation et chargement des données (ETL), la phase de génération des tableaux de bord et la phase data mining.

4.8.2 Identification des données et collecte des informations :

Comme présenté dans la Phase de conception analytique de GIMSI, nous avons localisé les données nécessaires à notre solution et décidé la quantité de ce qui est pertinent pour nos besoins.

4.8.3 Développement de la phase ETL

Cette alimentation (le plus souvent appelée processus ETL « Extract-Transform-Load ») se déroule en 3 phases qui sont :

- Extraction des données
- Transformation des données
- Le chargement des données traitées dans l'entrepôt de données

Cette phase nous permettra de démontrer les différentes étapes du processus ETL qui préparent les données pour la mise en œuvre du tableau de bord.

La figure 4.12 décrit le processus d'un ETL :

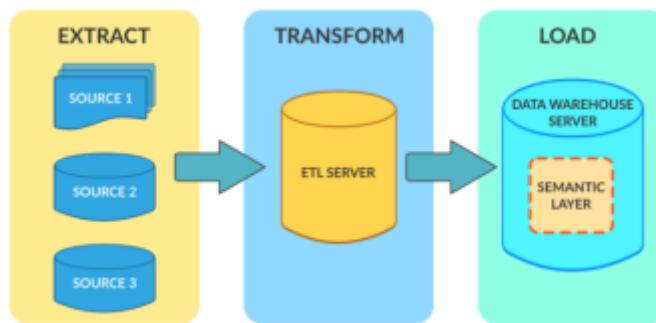


FIGURE 4.14 – Le processus ETL

Afin de mieux comprendre cette étape, nous prenons comme exemple la dimension "DimProjet" modélisée dans la figure 4.13 suivante :

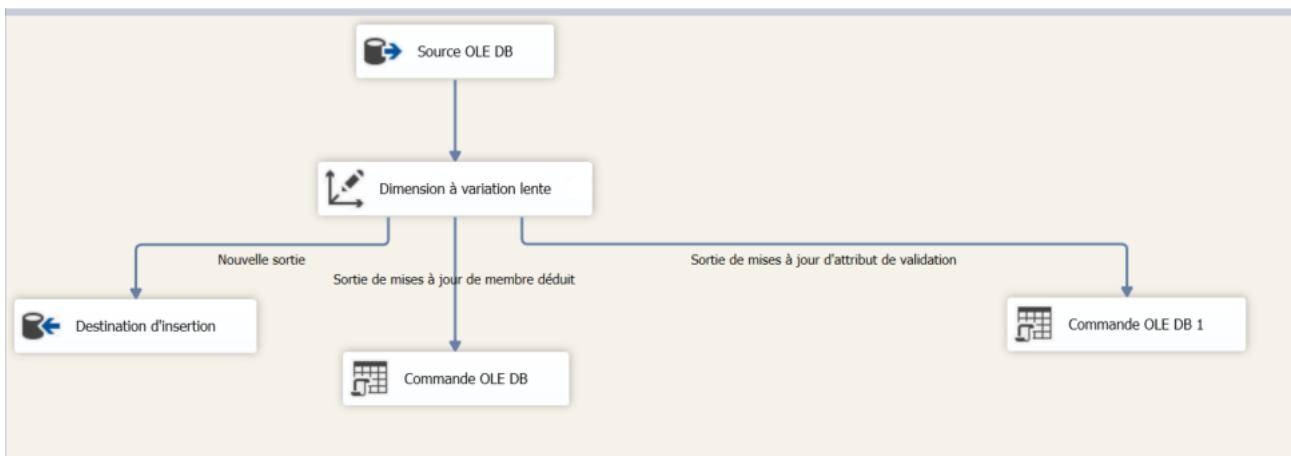


FIGURE 4.15 – Alimentation de la table dimension "dimprojet"

Et encore tel que présenté dans la figure ci-dessus, nous avons défini le table fait

"FactVision" :

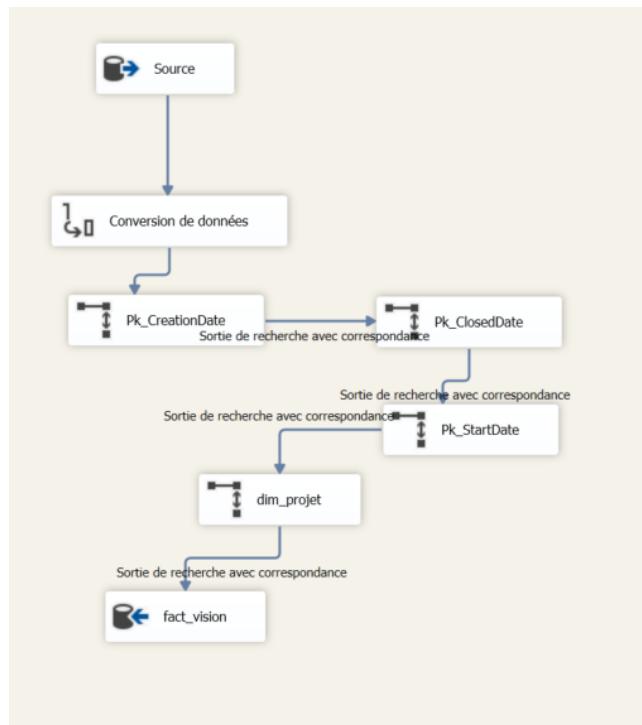


FIGURE 4.16 – Alimentation de la table de faits "FactVision"

4.9 Réalisation

Dans cette partie nous allons présenter les captures d'écran des principales interfaces de notre application.

4.9.1 Interface "liste des tâches"

4.9.2 Interface "ajouter tâche"

4.9.3 Interface "Liste des types des tâches"

4.9.4 Interface des tableaux de bord

Pour assurer une utilisation accrue de notre application, nous avons conçu des interfaces qui fournissent à l'utilisateur une visibilité sur la totalité des tableaux de bord.

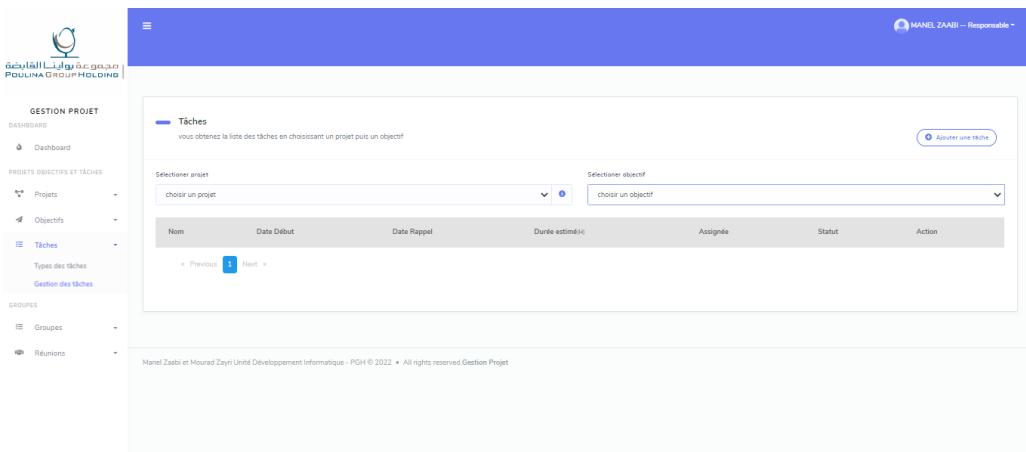


FIGURE 4.17 – Interface "liste des tâches"

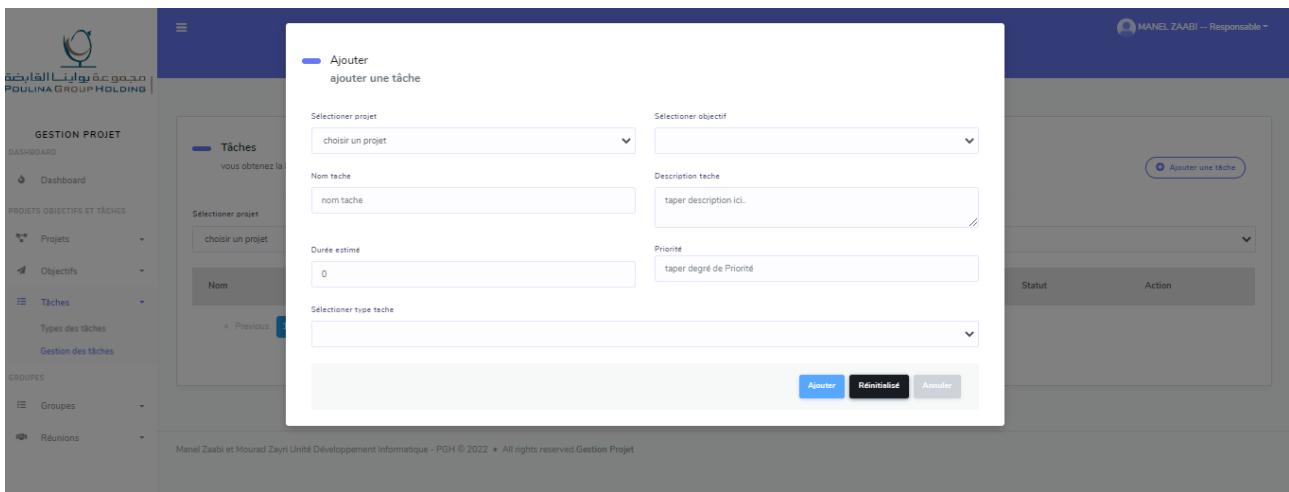


FIGURE 4.18 – Interface "ajouter tâche"

Tableau de bord État projet

Tableau de bord suivi des objectifs

MANEL ZAABI -- Responsable --

Liste des types des tâches
ceci est la liste de tous les projets.

Ajouter un type de tâche

chercher par type de tâche

DASHBOARD

GESTION PROJET

PROJETS OBJECTIFS ET TÂCHES

- Projects
- Objectifs
- Tâches

GROUPES

- Groupes
- Réunions

Nom

Action

Nom	Action
étude besoins	
réalisation	
Conception	
Test	
Développement	

< Previous Next >

Manel Zaabi et Mourad Zayri Unité Développement Informatique - PGH © 2022 • All rights reserved. Gestion Projet

FIGURE 4.19 – Interface "Liste des types des tâches"

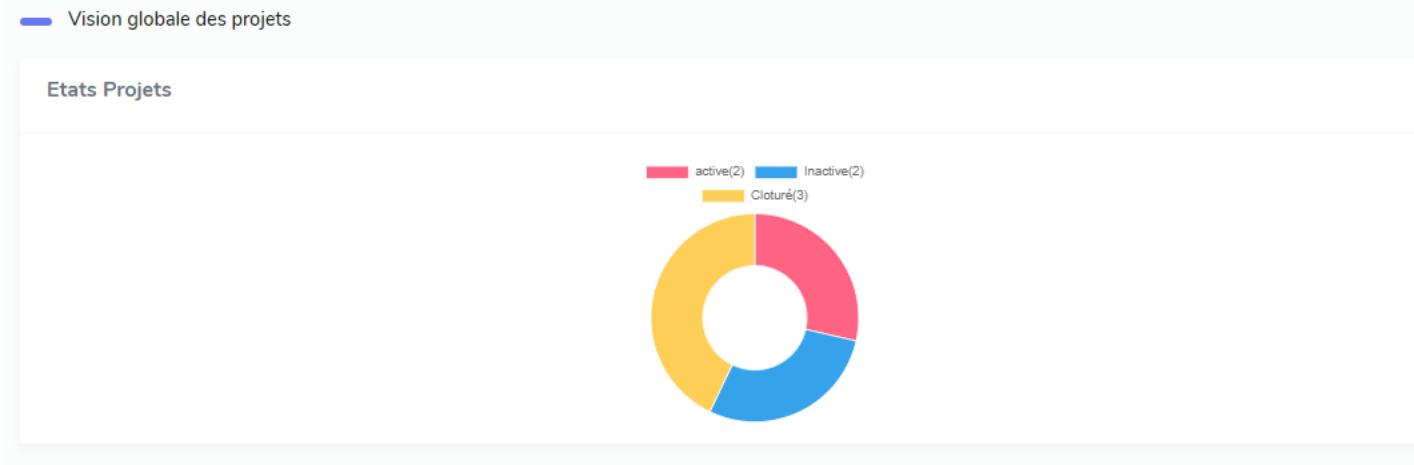


FIGURE 4.20 – tableau de bord "État projet"

Tableau de bord suivi des tâches

4.10 Test Sprint 2

4.10.1 Cas de test d'ajout d'un type tâche

Interface d'ajout d'un type de tâche

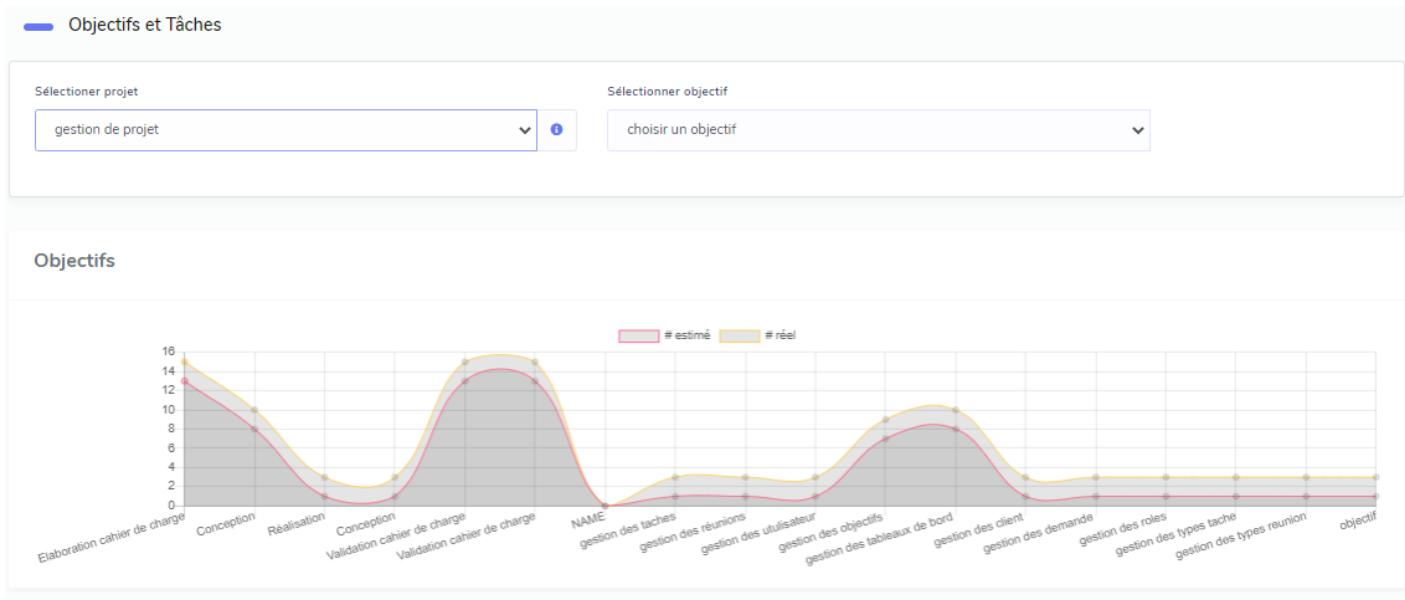


FIGURE 4.21 – tableau de bord "Suivi des objectifs"

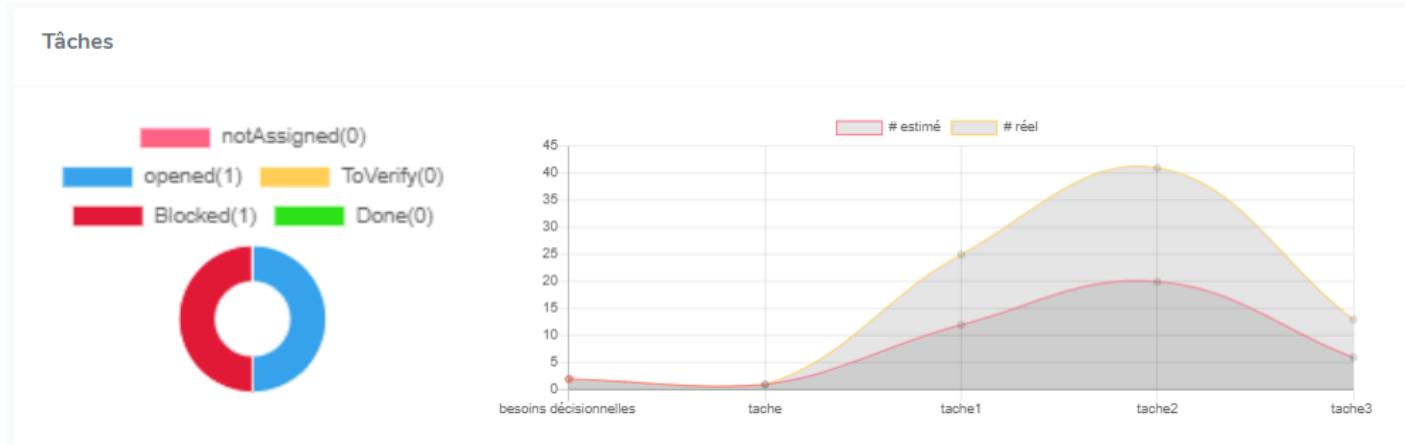


FIGURE 4.22 – tableau de bord "Suivi des tâches"

Interface type tâche ajouté avec succès

4.10.2 Cas de test la gestion des tâches

Tester L'ajout d'une tâche

Interface "ajouter tâche"

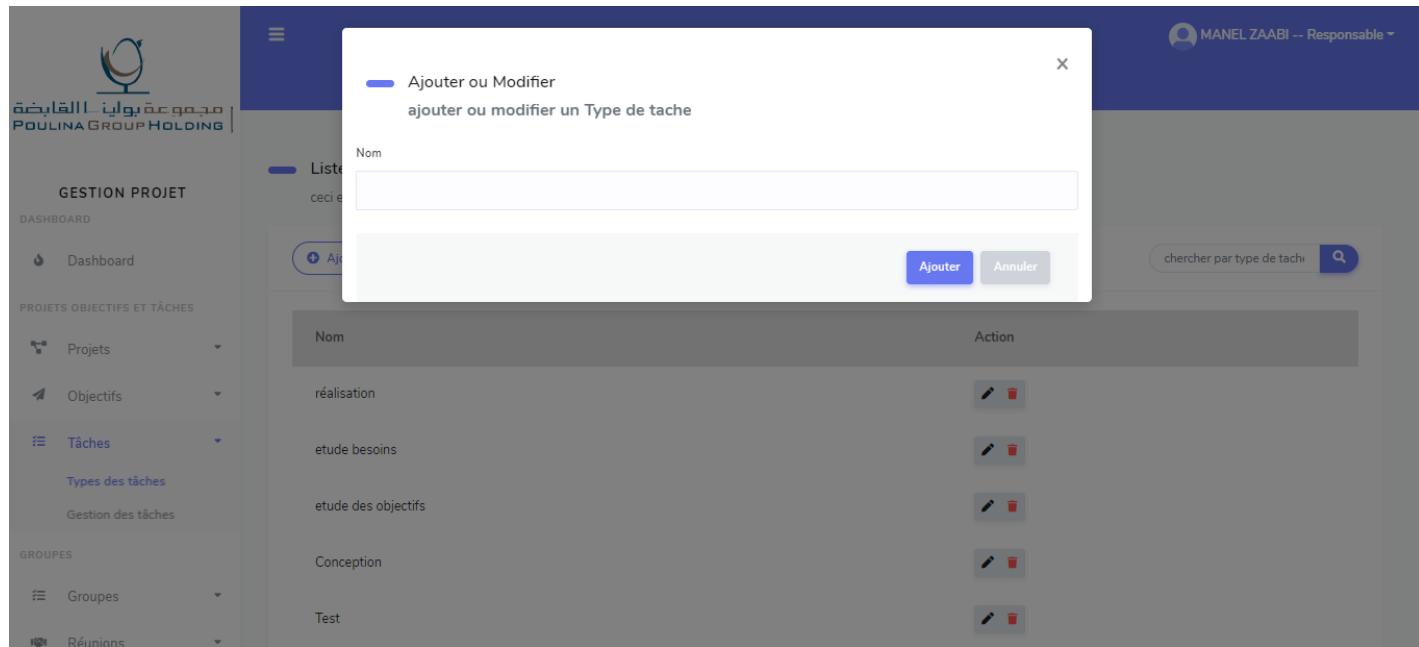


FIGURE 4.23 – Interface "ajouter type tâche"

Interface "alert d'ajout avec succès"

Interface "liste tâche apres l'ajout"

The screenshot shows a user interface for managing task types. At the top right, there is a message "ajouté avec succès" (added successfully) with a checkmark icon. Below this, a navigation bar includes a "Liste des types de tâches" link and a "Ajouter un type de tâche" button. A search bar at the top right says "chercher par type de tâche". The main content area is a table with columns "Nom" (Name) and "Action". The table lists six task types: "réalisation", "étude besoins", "étude des objectifs", "Conception", "Test", and "Développement". Each row has edit and delete icons in the "Action" column. At the bottom left, there are navigation buttons: "« Previous", a blue "1" button, and "Next »".

FIGURE 4.24 – Interface "type tâche ajouté avec succès"

Interface "Modifier tâche "

4.11 Scrum Board

Nous présentons ci-dessous le scrum board de sprint 2 au cours de la 2ème semaine de sa période de réalisation.

4.12 Conclusion

Tout au long de ce chapitre, nous avons présenté les différentes User Stories du sprint 2 ainsi que leur conception, leur réalisation et leurs tests. Ainsi que les différentes phases de réalisation allant de la phase ETL à la génération des tableaux de bord. Dans le chapitre prochain nous allons montrer plus de détails sur le sprint 3.

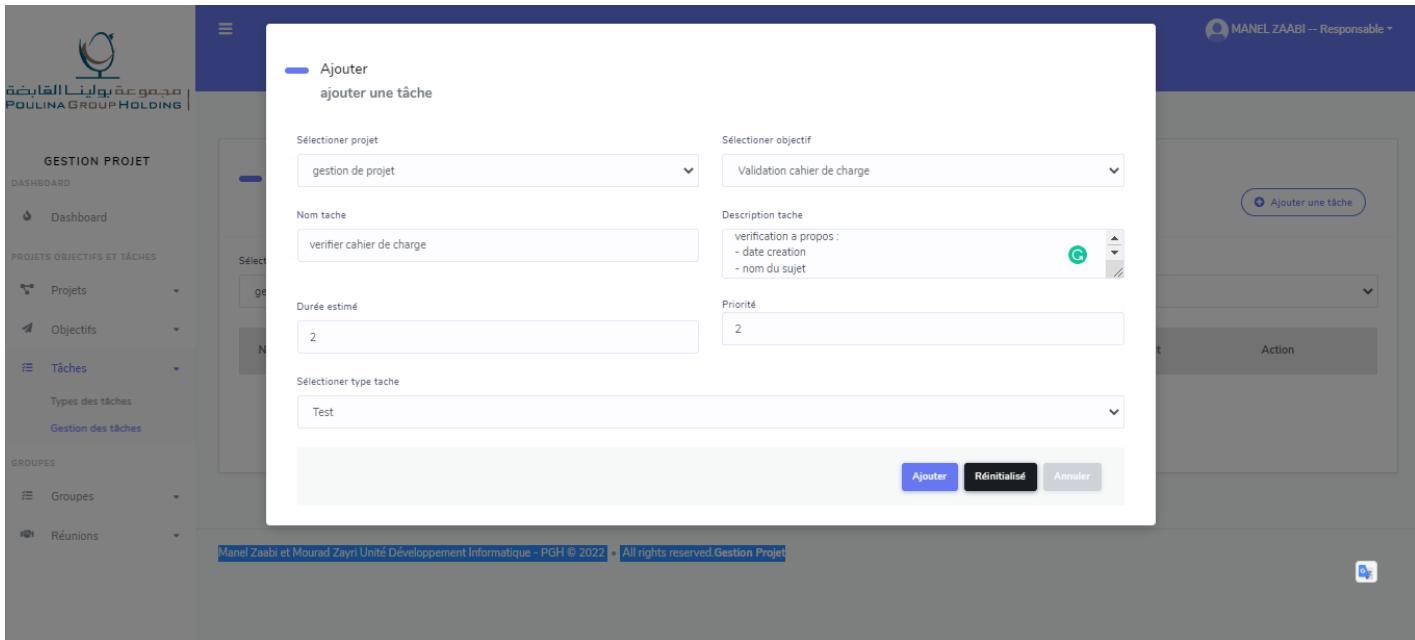


FIGURE 4.25 – Interface "ajouter tâche"

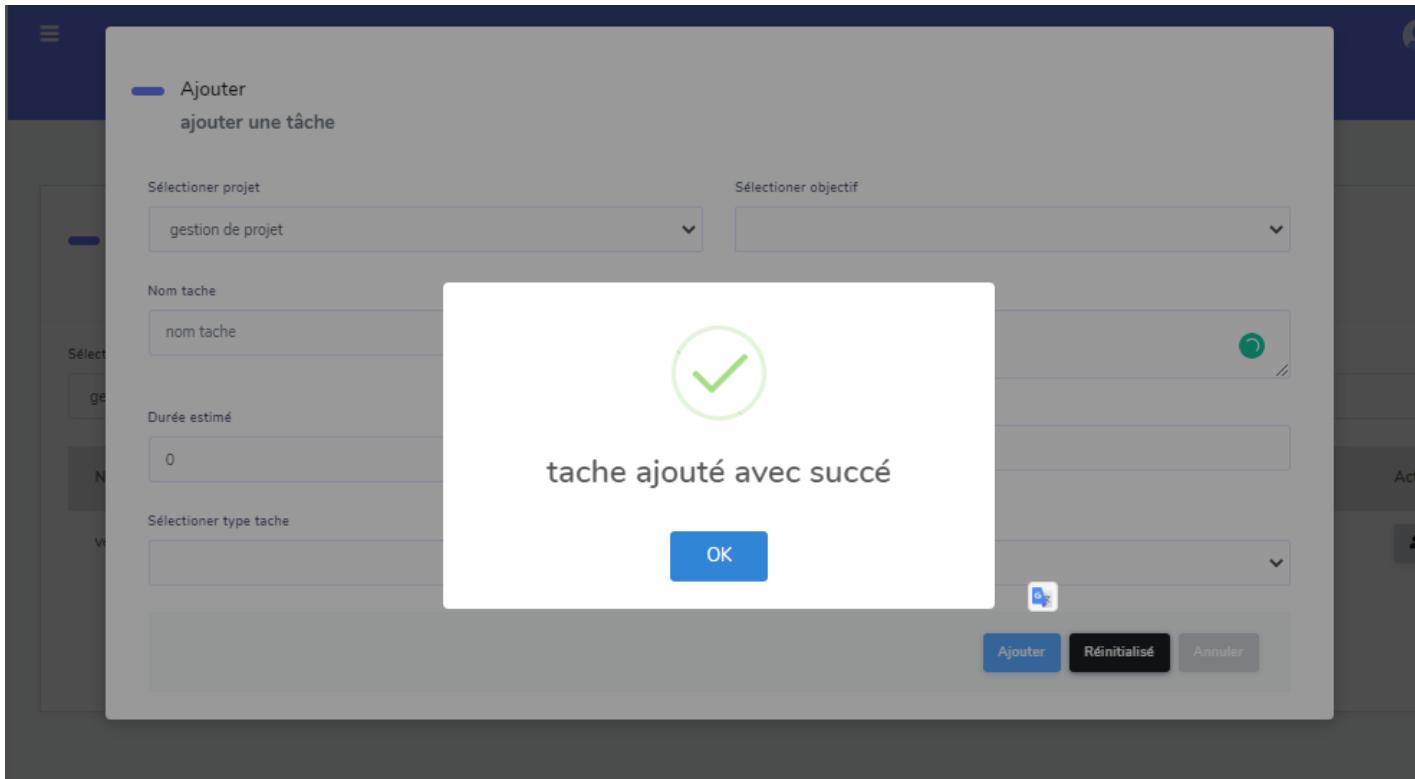


FIGURE 4.26 – Interface "alert d'ajout avec succès"

Tâches

vous obtenez la liste des tâches en choisissant un projet puis un objectif

Ajouter une tâche

Sélectionner projet: gestion de projet

Sélectionner objectif: Validation cahier de charge

Nom	Date Début	Date Rappel	Durée estimé[h]	Assignée	Statut	Action
vérifier cahier de charge	Pas encore	Pas encore	2	Non assignée	Open	

« Previous 1 Next »

FIGURE 4.27 – Interface "liste tâche apres l'ajout"

POULINA GROUP HOLDING

GESTION PROJET

Dashboard

PROJETS OBJECTIFS ET TÂCHES

- Projets
- Objectifs
- Tâches** (highlighted)
- Types des tâches
- Gestion des tâches

GROUPES

- Groupes
- Réunions

Ajouter une tâche

Modifier
modifier une tâche

Tâches

vous obtenez la liste des tâches en choisissant

Sélectionner projet: gestion de projet

Nom	Date Début	Date Rappel	Durée estimé[h]	Assignée	Statut	Action
type1	Pas encore	Pas encore	1	Non assignée	Open	

nom tâche

tâche modifiée avec succès

OK

Previous 1 Next

FIGURE 4.28 – Interface "Modifier tâche "

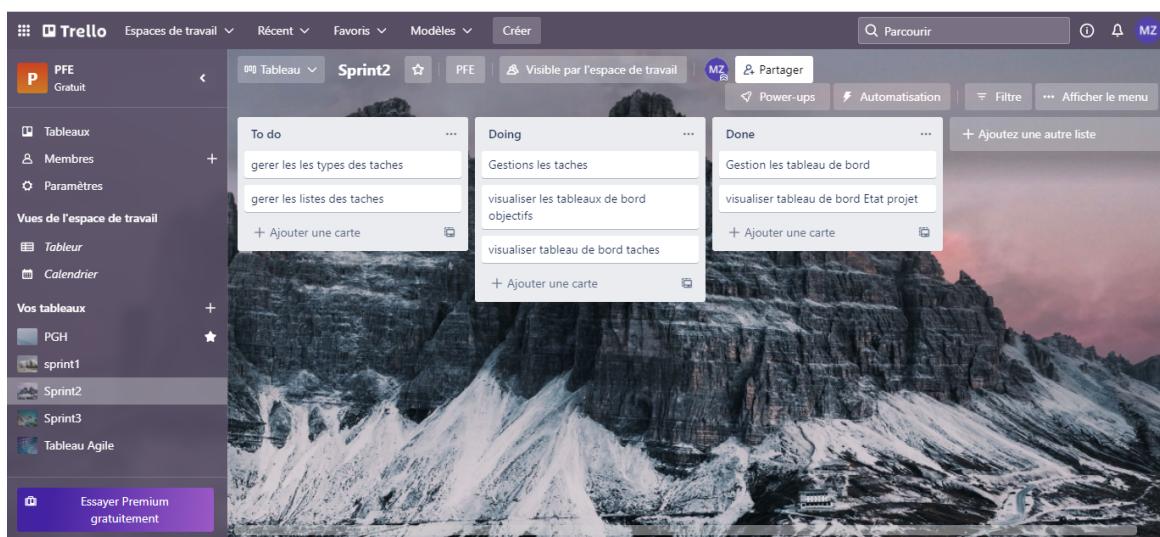


FIGURE 4.29 – Scrum Board Sprint 2

Chapitre 5

Étude et réalisation du sprint 3

5.1 Introduction

Comme les deux premiers sprint nous présenterons le diagramme de cas d'utilisation général puis les détails relatifs aux fonctionnalités offertes. Nous présenterons par la suite la description textuelle et le diagramme de séquences de certains cas d'utilisations. Enfin nous présenterons le diagramme de classes du sprint3.

5.2 Sprint Backlog

ID	User Story	priorité	complexité
1	En tant qu'utilisateur, je veux demander une réunion	1	élevée
2	En tant que responsable , je veux consulter la liste des groupes	2	moyenne
3	En tant que responsable , je veux gérer les réunions	1	élevée
4	En tant que responsable , je veux gérer les demandes de réunions	1	élevée
5	En tant que client , je veux donner mon avis suite a une réunion	3	faible
6	En tant que responsable, je veux consulter la liste des types de réunions	2	moyenne
7	En tant que responsable, je veux ajouter un type de réunion	1	élevée

TABLE 5.1 – Backlog du sprint 3

5.3 Spécification fonctionnelle

5.3.1 Diagramme de cas d'utilisation générale du Sprint 3

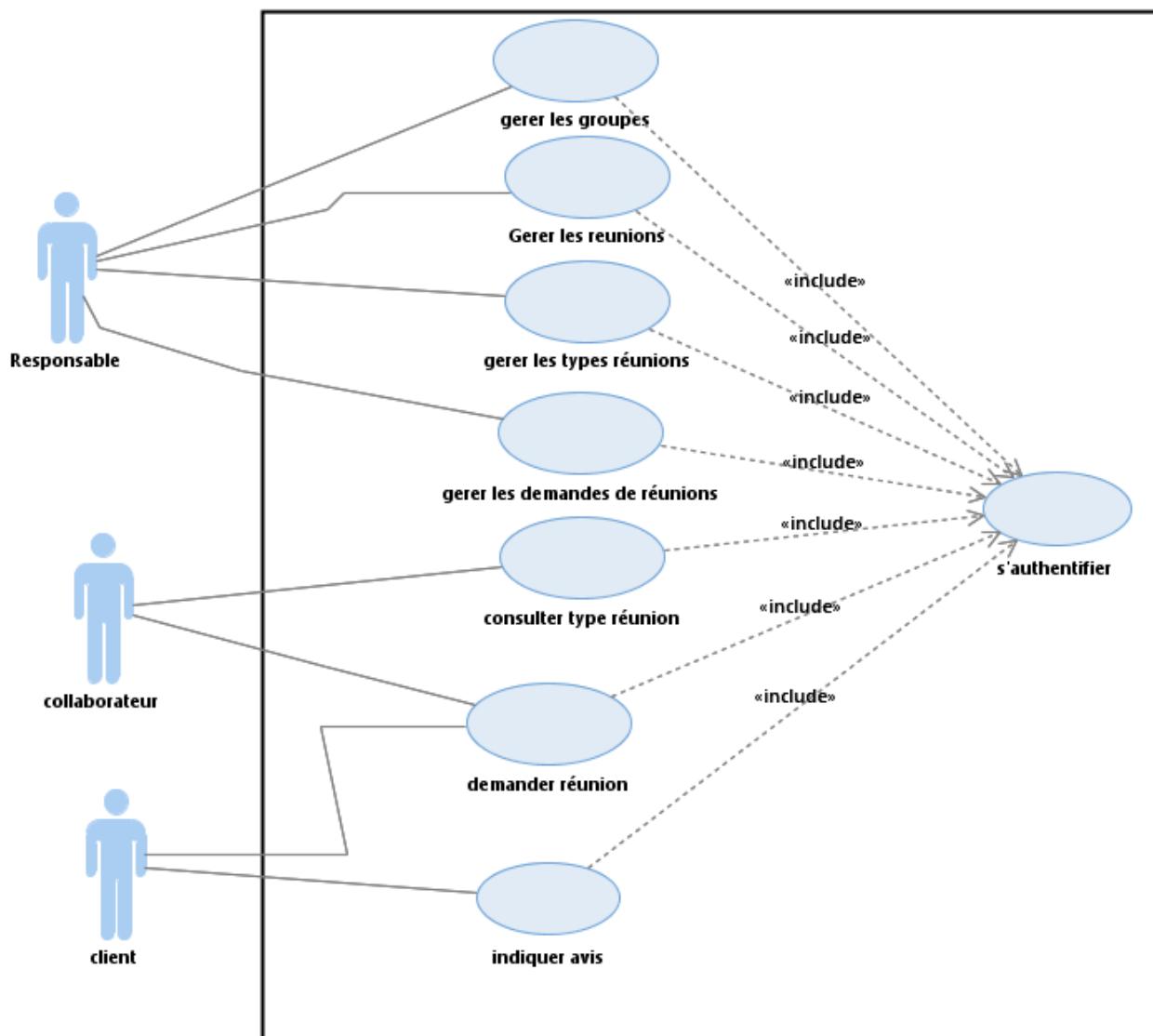


FIGURE 5.1 – Diagrammes du cas d'utilisation générale du Sprint 3

5.3.2 Diagrammes de cas d'utilisation détaillés

Pour présenter un diagramme de cas d'utilisation détaillé, nous le divisons en sous-diagramme. Chacun présente un raffinement d'un cas d'utilisation

Raffinement du cas d'utilisation "gérer groupe"

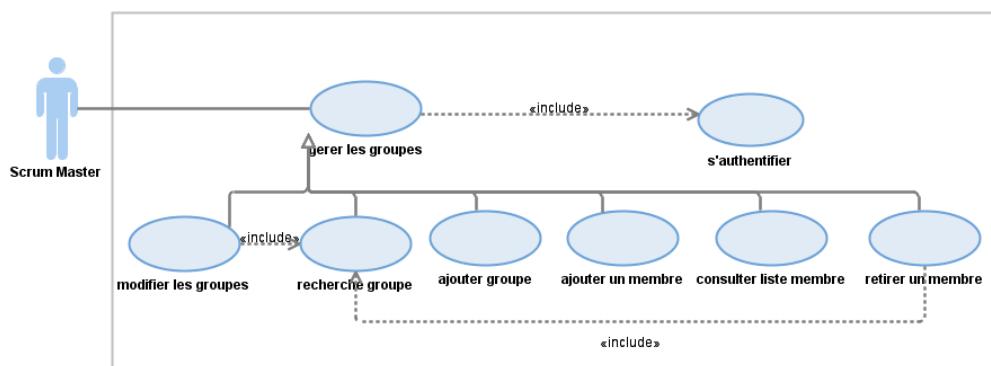


FIGURE 5.2 – Raffinement du cas d'utilisation "gérer groupe"

Raffinement du cas d'utilisation "gérer réunion"

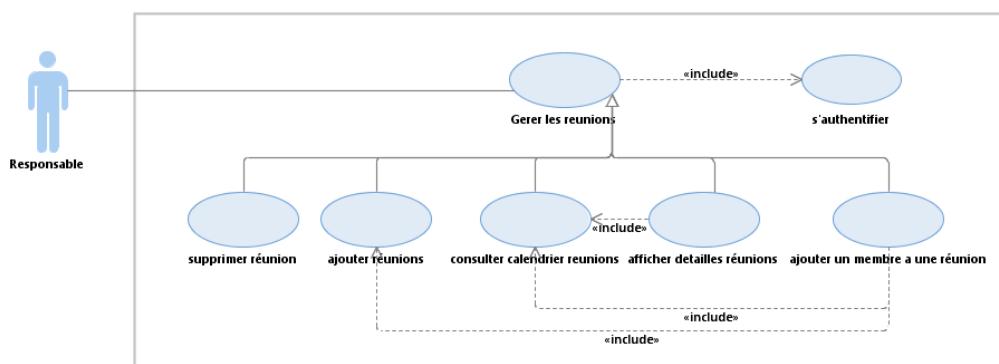


FIGURE 5.3 – Raffinement du cas d'utilisation "gérer réunion"

Raffinement du cas d'utilisation "gérer type réunion"

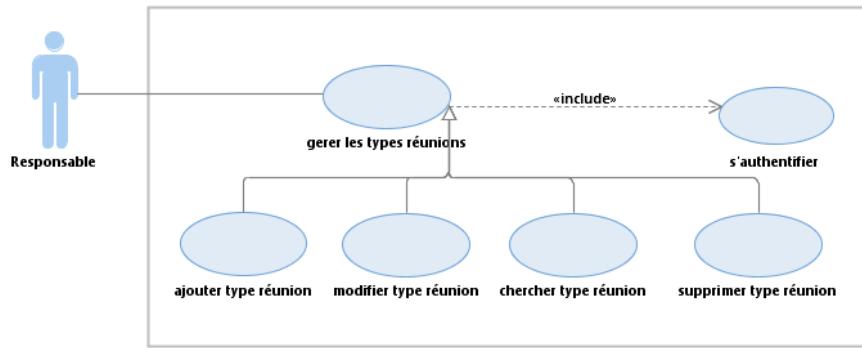


FIGURE 5.4 – Raffinement du cas d'utilisation "gérer type réunion"

Raffinement du cas d'utilisation "gérer demande de réunion"

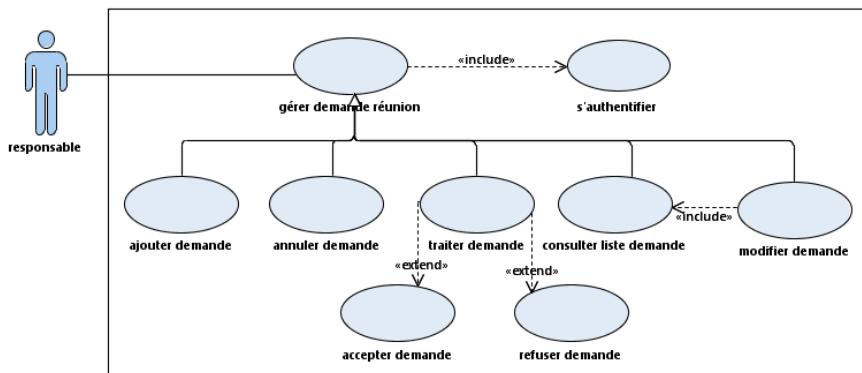


FIGURE 5.5 – Raffinement du cas d'utilisation "gérer demande de réunion"

5.4 Spécification des cas d'utilisations

Spécification du cas d'utilisation "ajouter une demande de réunion"

<p style="text-align: center;">Titre :Ajouter une demande</p>	
<p>Résumé : Tout acteur doit avoir la possibilité de lancer une demande de Réunion.</p>	
<p style="text-align: center;">Acteur : Responsable , Collaborateur, client</p>	
Pré-condition :	Les acteurs doivent s'authentifier puis consulter la calendrier de leurs demandes de réunions personnelles
Post-condition :	La demande de réunion est ajoutée avec succès
Scénario de base :	<ol style="list-style-type: none"> 1. L'acteur choisit de lancer une demande de réunion. 2. Le système affiche le formulaire d'ajout. 3. L'acteur remplit le formulaire avec les renseignements nécessaires. 4. L'acteur choisit d'enregistrer la demande. 5. Le système vérifie la validation du formulaire. 6. Le système affiche un message de succès d'ajout.
Scénario d'exception :	<ol style="list-style-type: none"> 1. a- Le système trouve le formulaire invalide : <ol style="list-style-type: none"> 1. Le système affiche un message d'erreur. 2. Le cas d'utilisation retourne à l'étape 2.

TABLE 5.2 – Spécification du cas d'utilisation "Ajouter une demande de réunion"

Spécification du cas d'utilisation "modifier demande de réunion"

Spécification du cas d'utilisation "supprimer demande de réunion"

5.5 Conception

5.5.1 Diagrammes de séquences détaillés

Cas d'utilisation "recherche demande"

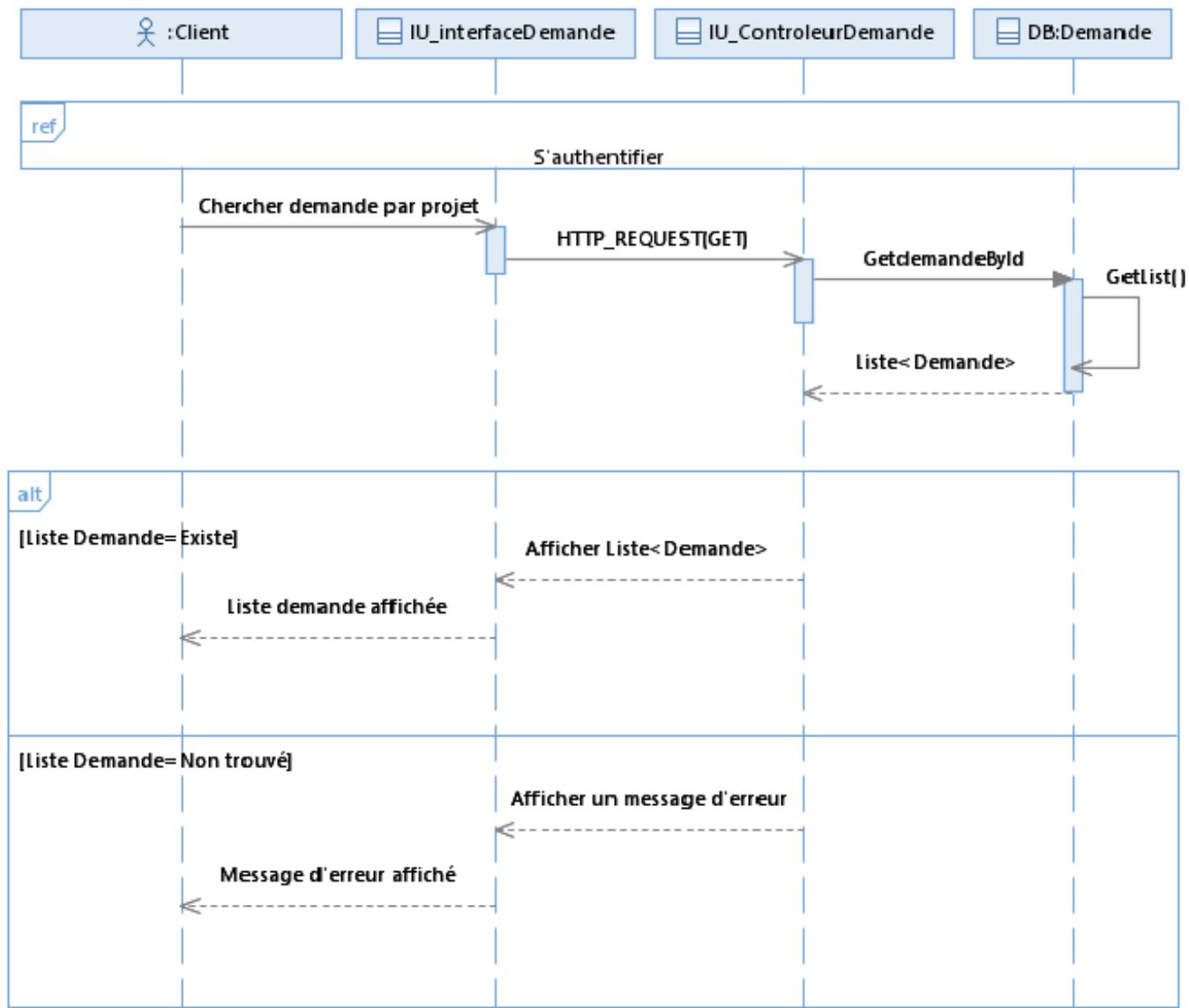


FIGURE 5.6 – Cas d'utilisation "recherche demande"

Titre : modifier demande de réunion	
Résumé : L'acteur doit pouvoir modifier l'une de ses demandes avant qu'elle soit étudiée.	
Acteur : Responsable , Collaborateur, client	
Pré-condition :	L'utilisateur doit s'authentifier puis consulter la liste de ses demandes de réunion personnelles. La demande de congé doit être non étudiée. (sauf pour le responsable)
Post-condition :	la demande à modifier doit être enregistrée avec ses nouvelles informations.
Scénario de base :	<ol style="list-style-type: none"> 1. L'utilisateur choisit la demande de réunion à modifier. 2. L'utilisateur choisit de modifier la demande. 3. Le système affiche les informations de la demande de réunion. 4. L'utilisateur change ses données. 5. L'utilisateur confirme la modification. 6. Le système enregistre la demande avec ses nouvelles informations. 7. Le système affiche un message de succès de modification.
Scénario d'exception :	<p>4. a- l'acteur ne respecte pas les contraintes des champs à modifier. Le système informe l'acteur de la contrainte violée.</p>

<p style="text-align: center;">Titre :supprimer demande de réunion</p>	
<p>Résumé : L'acteur doit pouvoir supprimer l'une de ses demandes avant qu'elle soit étudiée.</p>	
<p style="text-align: center;">Acteur : Responsable , Collaborateur, client</p>	
Pré-condition :	<p>L'acteur doit s'authentifier puis consulter la liste des demandes</p> <p>La demande doit être non encore étudiée.</p>
Post-condition :	la demande est supprimée avec succès.
Scénario de base :	<ol style="list-style-type: none"> 1. L'acteur choisit la demande à supprimer. 2. l'acteur clique sur le bouton supprimer. 3. Le système affiche un message de confirmation de suppression. 4. L'acteur confirme la suppression. 5. Le système supprime la demande. 6. Le système affiche un message de succès de suppression.
Scénario d'exception :	<p>4.a- l'acteur ne confirme pas la suppression -le système afficher la liste des demandes de l'acteur sans supprimer la demande.</p>

TABLE 5.4 – Spécification du cas d'utilisation supprimer une demande de réunion

Cas d'utilisation "modifier réunion"

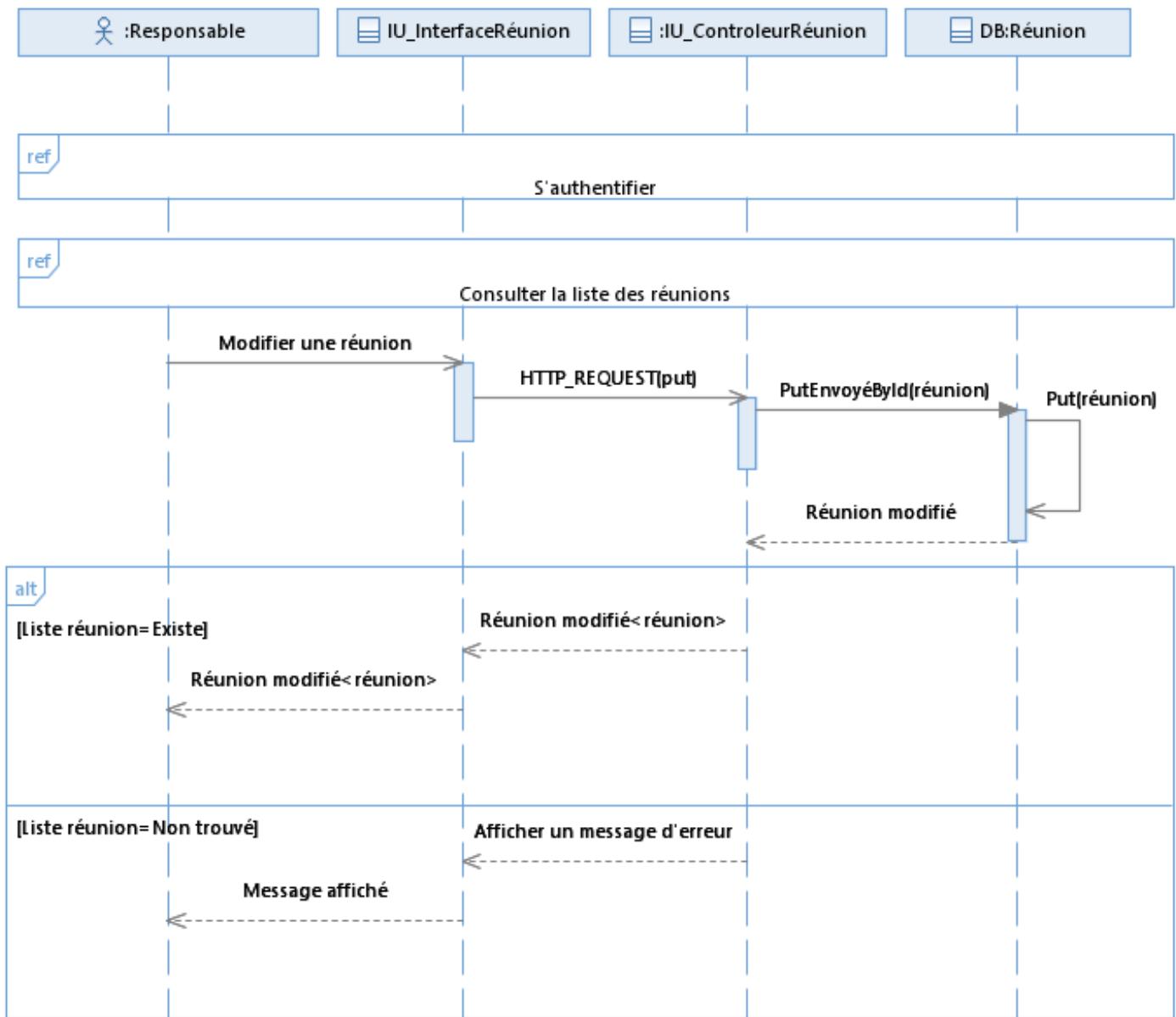


FIGURE 5.7 – Cas d'utilisation "modifier réunion"

Cas d'utilisation "supprimer réunion"

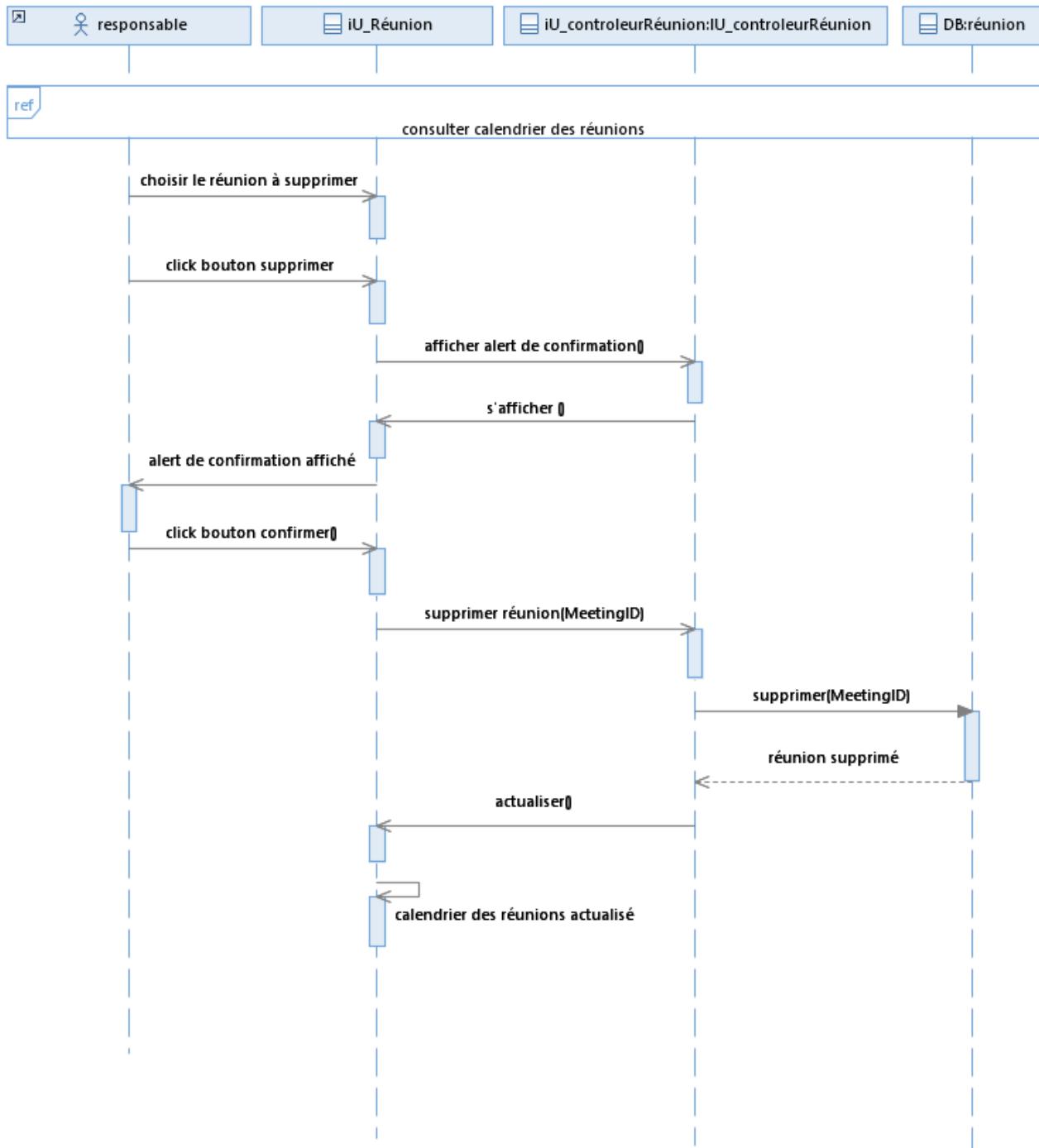


FIGURE 5.8 – Cas d'utilisation "supprimer réunion"

5.5.2 Diagramme de Classes

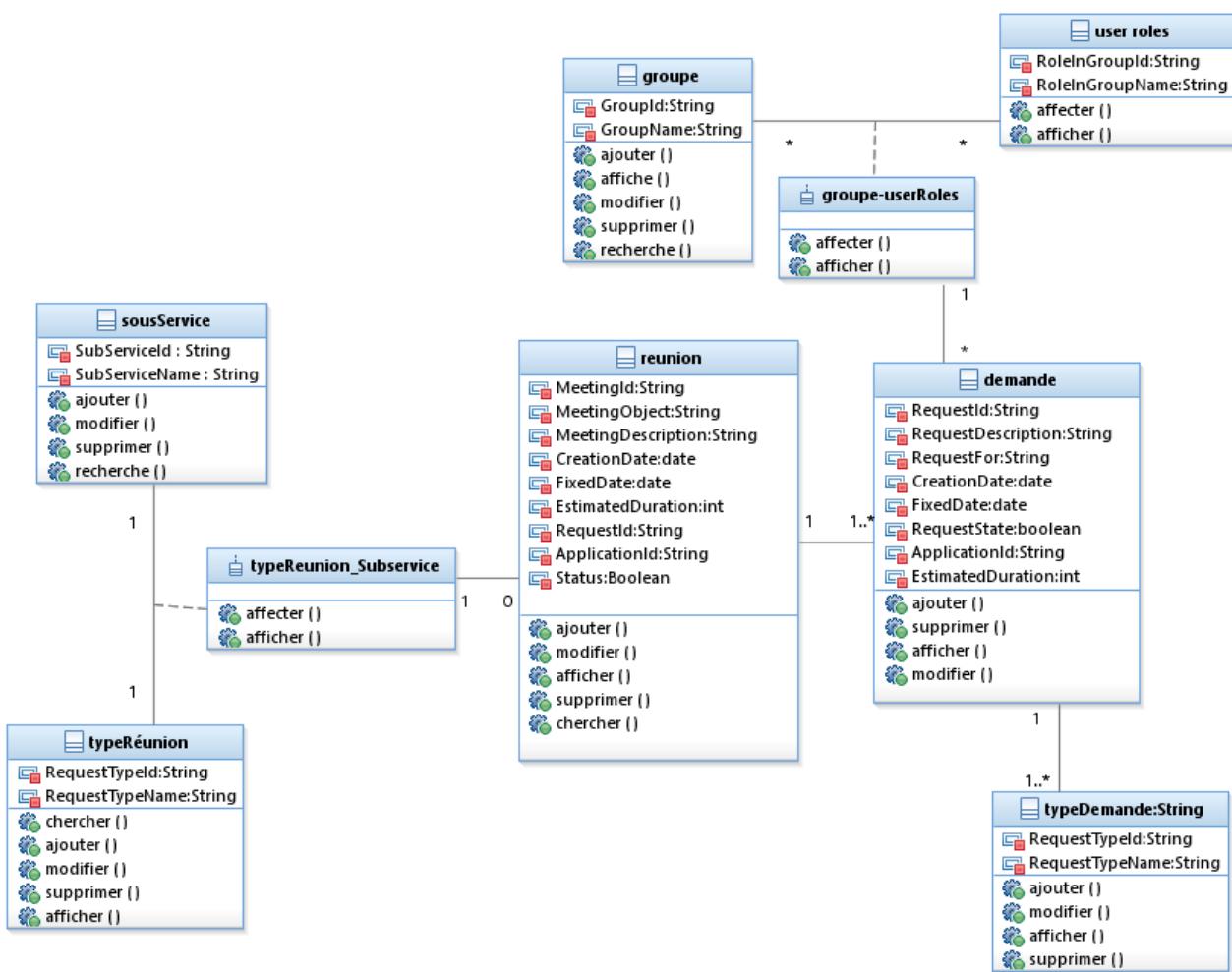


FIGURE 5.9 – Diagramme de classe sprint3

5.6 Réalisation

Interface "ajouter réunion"

Ajouter
ajouter une réunion

Sélectionner projet: gestion de projet

Sélectionner objectif: Réalisation

Sélectionner type réunion: Réunion

Membre Réunion: Utilisateur: Manel Zaabi et Mourad Zaytoun

Objet de Réunion: objet de réunion

Description de Réunion: taper description ici.

Date et heure de réunion: jj/mm/aaaa --::-- 0

Ajouter, Réinitialisé, Annuler

FIGURE 5.10 – Interface "ajouter réunion"

Interface "Gestion des Réunions"

Réunions
vous obtenez la planification des réunions dans un calendrier

Ajouter une réunion

Sélectionner projet: gestion de projet

Sélectionner objectif: Réalisation

Calendrier Gérer demandes

Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	1	2

FIGURE 5.11 – Interface "Gestion des Réunions"

5.7 Test Sprint 3

Interface "Consultation Liste des groupes"

The screenshot shows a web-based application interface for project management. At the top, there is a header with the logo of 'POULINA GROUP HOLDING' and a user profile for 'MANEL ZAABI -- Responsable'. The left sidebar contains navigation links for 'GESTION PROJET' (Dashboard, Projets, Objectifs, Tâches, Groupes, Réunions), 'PROJETS OBJECTIFS ET TÂCHES' (Projets, Objectifs, Tâches), and 'GROUPES' (Groupes, Gestion des groupes). The main content area is titled 'Groupes' and displays a table titled 'Liste des groupes' with columns: 'projet', 'Objectif', 'Groupe', and 'Membres'. The table lists several groups: 'Elaboration cahier de charge' (concepteurs), 'Conception' (Business analysts), 'spécification des besoins' (Analysateurs), 'gestion de projet' (Business Analysts), 'Conception' (Cloud Computing), 'Spécification' (testeur), and 'Elaboration' (développeur). A search bar is located at the top right of the table.

FIGURE 5.12 – Interface "Consultation Liste des groupes"

Interface "afficher détaillées Réunions"

The screenshot shows a calendar interface for managing meetings. At the top, there are dropdown menus for 'Sélectionner projet' (PFE) and 'Sélectionner objectif' (Elaboration). Below the calendar, there are tabs for 'Calendrier' and 'Gérer demandes', with 'Calendrier' selected. The calendar grid shows dates from Sunday 29 to Saturday 25. Various meeting icons are placed on specific days and times, such as a yellow circle for '4:52 PM réunion conception' on Wednesday, August 8th. A tooltip for this meeting provides details: 'Réunion de conception à 16h52 le 08/08/2024'. The bottom of the calendar shows a legend for different meeting types: 'Réunion de conception', 'Réunion de test', 'Réunion de développement', and 'Réunion de gestion'.

FIGURE 5.13 – Interface "afficher détaillées Réunions"

5.8 Scrum Board

Nous présentons ci-dessous le scrum board de sprint 3 au cours de la 2ème semaine de sa période de réalisation.

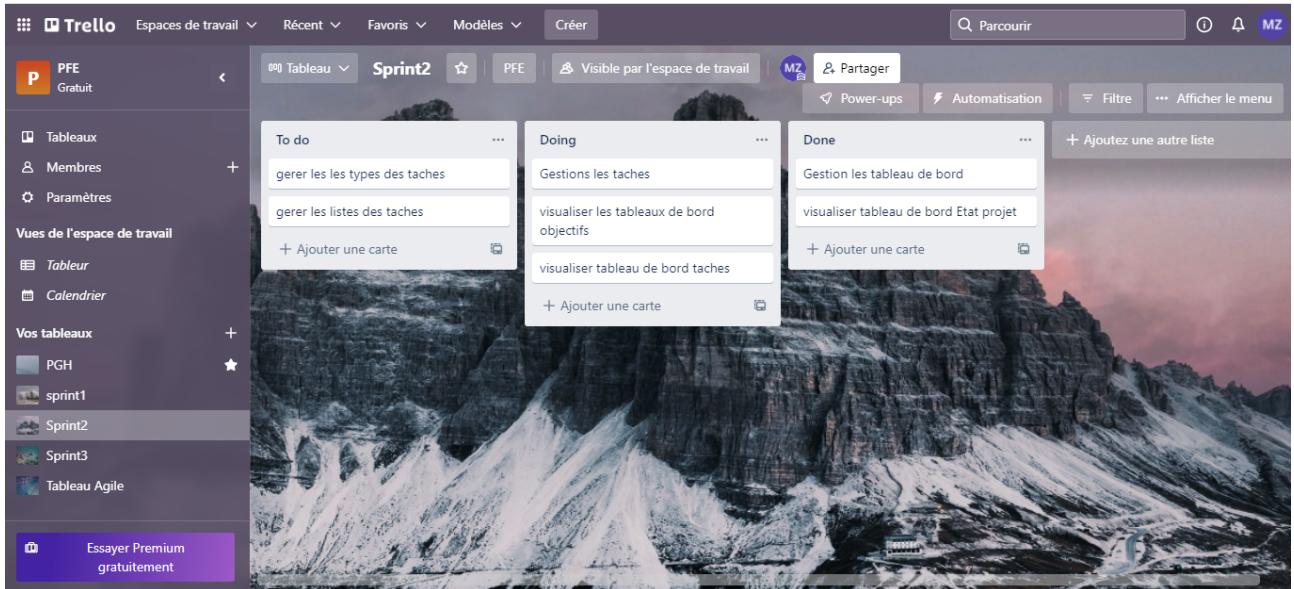


FIGURE 5.14 – Scrum Board Sprint 3

5.9 Conclusion

A ce stade, nous avons réussi à développer le dernier sprint de notre application pour arriver à un produit complet et fonctionnel. Notre application Web est donc prête à être exploité en offrant aux partenaires la possibilité de gérer convenablement toutes les fonctionnalités de la plateforme.

Conclusion générale

Aujourd’hui, le déroulement et le suivi des projets par les entreprises via les processus traditionnelles et classiques de gestion des projets s’avère être un processus complexe, et très peu fiable à cause du suivi de l’équipe projet du cahier des charges à la lettre et le travail sur la totalité du projet jusqu’à sa livraison. Il n’y a pas d’interaction avec le client qui recevra son projet une fois que celui-ci est terminé, ne laissant pas de place pour les changements et les imprévus, il est donc préférable de tout réussir du premier coup.

Notre projet de fin d’études au sein de l’entreprise Poulina Group Holding était donc de développer une application qui permettrait de gérer les projets qui seront réalisés au sein de notre entreprise d’accueil. Ainsi que l’ajout du touche business intelligence a notre application va aider les décideurs du prendre les bonnes décisions grâce aux tableaux de bord qui seront claires et constructives.

La réalisation de ce travail a nécessité au début de faire une collecte d’informations qui nous a permis à son tour de préparer une étude théorique au cours de laquelle nous avons étudié et critiqué l’existant. Cette étude nous a amené à proposer quelques solutions et en choisir une.

La solution retenue était étudiée d’une manière détaillée et nous avons pu grâce à cette étude déterminer les besoins fonctionnels et non fonctionnels de l’application. Arrivant à cette étape nous étions capables de passer à concevoir notre système. Toutes les étapes sus indiquées nous ont finalement amené à la dernière étape à savoir : la réalisation.

Malgré quelques difficultés rencontrées, qui a nécessité un travail de réflexion pour délimiter le projet et le manque d'expérience en programmation angular avec le Framework ASP.net et l'architecture microservices, qui ont nécessité une période intense d'auto apprentissage, nous sommes arrivées enfin à implémenter un système qui sera exploité par les différents services de PGH pour faciliter énormément leurs tâches et gérer leurs projets.

Ce projet a été très bénéfique aussi bien au niveau informatique qu'au niveau professionnel. D'une part, il nous a permis de mettre en pratique nos connaissances acquises pendant trois années d'étude à l'institut supérieur de gestion de Sousse et de raffiner nos capacités.

D'autre part, il nous a permis d'apprendre un nouveau langage de programmation et un nouveau Framework.

Enfin, il nous a permis de vivre de près l'expérience du travail au sein d'une société, d'améliorer nos capacités de communication, et de nous adapter à la vie professionnelle.

Bien que l'application développée répond aux besoins spécifiés, certaines lacunes restent à déplorer, dont la participation du client dans la gestion de notre application restent limitées.

Comme perspectives, notre présent travail peut être amélioré, en une application mobile pouvant être exploitée sur une tablette ou un Smartphone. Cette mobilité facilitera davantage l'utilisation du système et permettra une meilleure flexibilité

Pour conclure, nous souhaitons que ce modeste travail apporte satisfaction aux responsables de Poulin et aux membres de Jury.

Bibliographie

Bibliographie

- [1] <http://www.poulinagroupholding.com/fr/>
- [2] <https://nicolas-vidal.com/articles-focus-sur-la-methode-de-gestion-de-projet-agile-scrum/>
- [3] <http://gestion-projets.com/scrum-8-etapes/>
- [4] <https://docs.microsoft.com/en-us/ef/>
- [5] <https://docs.microsoft.com/en-us/dotnet/csharp/programming-guide/concepts/linq/>
- [6] <https://docs.microsoft.com/en-us/dotnet/>
- [7] <https://docs.microsoft.com/en-us/dotnet/csharp/>
- [8] <https://docs.microsoft.com/en-us/sql/sql-server/?view=sql-server-ver15>
- [9] <https://docs.microsoft.com/en-us/nuget/>
- [10] <https://nodejs.org/>
- [11] <https://getbootstrap.com/docs/4.1/getting-started/introduction/>
- [12] <https://jquery.com/>
- [13] <https://www.rabbitmq.com/documentation.html>
- [14] http://formations.imt-atlantique.fr/bi/bi_architectures.html
- [15] <https://paris-sorbonne.libguides.com/>
- [16] <https://kinsta.com/fr/base-de-connaissances/base-de-connaissances-github/>

Résumé

Le projet intitulé "Conception et Réalisation d'une Plateforme Web Pour La Gestion Des Projets" consiste à la création d'une plateforme de gestion des projets. Notre application nous a permis de gérer les projets, les groupes, les objectifs, les taches, les réunions et la demande des réunions, de même elle a répondu à un ensemble de besoins décisionnelles telle que la creation des dashboards, la visualisation des tableaux de bord des états projet, suivi des objectifs et des taches. Scrum est utilisé comme méthodologie de travail. Angular utulisé comme langage de programmation, visual studio et visual studio code sont utilisés comme logiciels pour développer notre application et visualiser nos résultats. Notre application peut être améliorée en ajoutant d'autre fonctionnalités à savoir la notification par mail. Nous pouvons aussi l'améliorer en une application mobile facilitera l'utilisation du système.

Abstract

The project entitled "Design and Realization of a Web Platform for Project Management" consists on the creation of a project management platform. Our application allowed us to manage projects, groups, objectives, tasks, meetings and requests for meetings, as well as meeting a set of decision-making needs such as the creation of dashboards, the visualization of dashboards of project states, monitoring of objectives and tasks. Scrum is used as a work methodology, Angular used as programming language, visual studio and visual studio code are used as software to develop our application and visualize our results. Our application can be improved by adding other features, namely email notification. We can also upgrade it to a mobile application that will make the system easier to use.