

RAPPORT DE PROJET DE FIN D'ANNÉE

Présenté en vue de l'obtention de la
LICENCE EN SCIENCE INFORMATIQUE

Spécialité : Génie logiciel et Système d'information

Conception et développement d'une solution de gestion projet pour une entreprise

Par MOHAMED FEFEL

Réalisé au sein de Think Trend



Encadrant Professionel : Dalel Loussaief
Rapporteur : Mme. Wafa Tebourski, Enseignante, ISTIC

Période De Stage : 05/08/2024 - 08/09/2024

Année Universitaire : 2024 - 2025

Dédicaces

Je souhaite consacrer cette réussite à toutes les personnes qui ont joué un rôle important dans ma vie.

Je dédie ce projet à mes parents, Fethi Felfel et Houyem Attia, qui m'ont soutenu et encouragé. Ceci est pour vous, et j'espère vous rendre fiers et heureux.

Mohamed Felfel

Remerciement

Au terme de ce projet de fin d'année, nous tenons à exprimer toute notre gratitude aux responsables et aux enseignants de l'Institut Supérieur des Technologies de l'Information et des Communications de Borj Cedria qui nous avons aidé de près ou de loin dans la réalisation de ce projet.

Nous adressons tous nos remerciements au M.Dalel Loussaief notre encadrante Professionnel pour la gentillesse et la spontanéité avec lesquelles vous avez bien voulu diriger ce travail.

Ce projet de fin d'année est le fruit d'un travail mené au sein de la entreprise Think Trend, dirigée par Mr. Mehdi Chaouch, à qui nous souhaitons exprimer toute notre gratitude pour sa confiance et pour tous les moyens scientifiques qu'il a mis à notre disposition.

Finalement, nous tenons à remercier les membres du jury qui ont accepté d'évaluer mon travail.

Table des matières

Introduction générale	1
1 Contexte du projet	2
1.1 Introduction	2
1.2 Présentation de l'organisme d'accueil	2
1.2.1 Organisme d'accueil :	2
1.2.2 Domaines d'activités	2
1.3 Présentation du projet	3
1.3.1 Cadre de projet	3
1.3.2 Problématique	3
1.3.3 Etude de l'existant	3
1.3.4 Solution proposée	4
1.4 Conclusion	4
2 Spécification des besoins	5
2.1 Introduction	5
2.2 Identification des besoins fonctionnels	5
2.3 Identification des besoins non fonctionnels	5
2.4 Identification des acteurs	6
2.5 Diagramme de cas d'utilisation structurée	6
2.6 Backlog de produit	8
2.7 Environnement de Travail	8
2.7.1 Méthodologie de Conception	8
2.7.2 Environnement Logiciel	9
2.7.3 Technologies utilisées	11
2.8 Architecture utilisée	11
2.9 Conclusion	12
3 Sprint 0 : l'élaboration des projets	13
3.1 Introduction	13
3.2 Identification de Backlog de Sprint 0	13
3.3 Raffinement du Sprint 0	15
3.3.1 Raffinement de cas d'utilisation « Gestion des projets » :	15
3.3.2 Raffinement de cas d'utilisation « Gestion plans d'actions » :	18
3.4 Conception du Sprint 0	21
3.4.1 Conception de cas d'utilisation « Gestion des Projets » :	21
3.4.2 Conception de cas d'utilisation « Gestion des plans d'actions » :	23
3.5 Réalisation du Sprint 0	25
3.5.1 Gestion des projets	25

3.5.2	Gestion des plans d'actions	27
3.6	Conclusion	29
4	Sprint 1 :Le Suivre des projets	30
4.1	Introduction	30
4.2	Identification de Backlog de Sprint 1	30
4.3	Raffinement du Sprint 1	32
4.3.1	Raffinement de cas d'utilisation « Gestion des Listes » :	32
4.3.2	Raffinement de cas d'utilisation « Gestion des Tâches » :	34
4.3.3	Raffinement de cas d'utilisation « Ajouter un Membre » :	36
4.4	Conception du Sprint 1	38
4.4.1	Conception de cas d'utilisation « Ajouter Liste » :	38
4.4.2	Conception de cas d'utilisation « Editer une Tâche » :	39
4.4.3	Conception de cas d'utilisation « Ajouter Membre » :	40
4.5	Diagramme de Classe Globale	41
4.6	Réalisation du Sprint 1	42
4.6.1	Gestion des listes	42
4.6.2	Gestion des Tâches	44
4.6.3	Gestion des Membres	45
4.7	Conclusion	47
	Conclusion Générale	48
	Bibliographie	49

Table des figures

1.1	Logo de société	2
2.2	Diagramme de cas d'utilisation structurée	7
2.3	La processus de Scrum	9
3.4	Raffinement du cas d'utilisation "Gestion des projets"	15
3.5	Raffinement du cas d'utilisation "Gestion plans d'actions"	18
3.6	Diagramme de classes de cas d'utilisation "Ajouter Projet"	21
3.7	Diagramme de séquence de cas d'utilisation "Gestion des Projets"	22
3.8	Diagramme de classe de cas d'utilisation "Supprimer plan d'action"	23
3.9	Diagramme de séquence de cas d'utilisation "Gestion des plans d'actions"	24
3.10	Interface gestion projet	25
3.11	Interface Ajouter projet	25
3.12	Interface Editer projet	26
3.13	Interface supprimer projet	26
3.14	Interface gestion plans d'actions	27
3.15	Interface Ajouter plan d'action	27
3.16	Interface editer plans d'action	28
3.17	Interface supprimer plans d'action	28
4.18	Raffinement du cas d'utilisation "Gestion des Listes"	32
4.19	Raffinement du cas d'utilisation "Gestion des tâches"	34
4.20	Raffinement du cas d'utilisation "Ajouter un Membre"	36
4.21	Diagramme de classe de cas d'utilisation « Ajouter Liste »	38
4.22	Diagramme de Séquence « Ajouter Liste »	38
4.23	Diagramme de classe de cas d'utilisation « Editer une Tâche »	39
4.24	Diagramme de Séquence « Editer une Tâche »	39
4.25	Diagramme de classe de cas d'utilisation « Ajouter Membre »	40
4.26	Diagramme de Séquence « Ajouter Membre »	40
4.27	Diagramme de Classe Globale	41
4.28	Interface Kanban	42
4.29	Interface de création d'une liste	42
4.30	Interface editer liste	43
4.31	Interface de supprimer une liste	43
4.32	Interface Ajouter Tâche	44
4.33	Interface d'editer une tâche	44
4.34	Interface Supprimer Tâche	45
4.35	Interface Ajouter membre	45
4.36	Interface ajout et voir membre	46
4.37	Interface état d'envoi d'email avec succès	46

Liste des tableaux

2.1	Backlog de produit	8
2.2	Logiciels et leurs définitions	10
2.3	Technologies utilisées	11
3.4	Backlog de produit détaillé de sprint 0	14
3.5	Raffinement du cas d'utilisation "Ajouter un projet"	15
3.6	Raffinement du cas d'utilisation "Editer un projet"	16
3.7	Raffinement du cas d'utilisation "Supprimer un projet"	16
3.8	Raffinement du cas d'utilisation "Lister les projets"	17
3.9	Raffinement du cas d'utilisation "Chercher un projet "	17
3.10	Raffinement du cas d'utilisation "Ajouter un plan d'action"	18
3.11	Raffinement du cas d'utilisation "Editer un plan d'action"	19
3.12	Raffinement du cas d'utilisation "Supprimer un plan d'action"	19
3.13	Raffinement du cas d'utilisation "Lister les plans d'action"	20
3.14	Raffinement du cas d'utilisation "Chercher un plan d'action par date"	20
4.15	Backlog de produit détaillé de sprint 1	31
4.16	Raffinement du cas d'utilisation "Ajouter une liste de tâches"	32
4.17	Raffinement du cas d'utilisation "Editer une liste de tâches"	33
4.18	Raffinement du cas d'utilisation "Supprimer une liste de tâches"	33
4.19	Raffinement du cas d'utilisation "Consulter la liste des tâches"	34
4.20	Raffinement du cas d'utilisation "Ajouter une tâche"	35
4.21	Raffinement du cas d'utilisation "Editer une tâche"	35
4.22	Raffinement du cas d'utilisation "Supprimer une tâche"	36
4.23	Raffinement du cas d'utilisation "Ajouter un membre à un projet"	37
4.24	Raffinement du cas d'utilisation "Consulter les membres d'un projet"	37

Introduction générale

Dans un environnement économique de plus en plus compétitif, les entreprises doivent gérer efficacement leurs ressources, leurs projets, leurs employés ainsi que leurs achats pour rester performantes. Cependant, elles sont souvent confrontées à la complexité de la gestion manuelle des processus internes, ce qui entraîne des pertes de temps et des erreurs. De ce fait, les entreprises ont besoin d'outils rapides et performants pour centraliser toutes ces données et permettre une gestion fluide et efficace.

Face à ce besoin, il devient essentiel de disposer d'une plateforme capable de regrouper et d'automatiser l'ensemble des processus internes. Dans le cadre de mon projet, ma mission consiste à concevoir une solution pour gérer efficacement les projets demandés par les clients au sein de l'entreprise. Cette solution vise à faciliter la gestion des projets. Le responsable de projet pourra ainsi saisir toutes les informations relatives aux nouveaux projets, définir les tâches à réaliser et assigner ces tâches aux membres de l'équipe concernés. Grâce à cette plateforme, il sera également possible de suivre, de manière précise et en temps réel, l'avancement des projets et d'ajuster les priorités en conséquence.

En centralisant ces informations sur une seule et même plateforme, la gestion des projets devient plus simple, plus rapide et plus transparente. Chaque jour, les responsables et les membres de l'équipe pourront facilement consulter l'état d'avancement des projets ainsi que les tâches réalisées et à réaliser, permettant ainsi un suivi régulier et efficace. Cette solution permettra à l'entreprise de gérer ses projets de manière proactive et de mieux répondre aux attentes de ses clients, tout en optimisant ses processus internes.

Pour la modélisation de notre futur système, nous avons suivi la méthodologie agile "SCRUM". Notre rapport est organisé de la manière suivante :

Chapitre 1 : Contexte du projet est l'étude de projet permet de comprendre l'environnement.

Chapitre 2 : Spécification des besoins est la définition des besoins et des exigences.

Chapitre 3 : Sprint 0 :l'élaboration des projets à réaliser pour les clients.

Chapitre 4 : Sprint 1 :Le Suivi des projets à partir de la gestion des listes et les tâches.

Nous clôturons notre rapport par une conclusion générale et des perspectives.

Chapitre 1

Contexte du projet

1.1 Introduction

L'étude du projet est une étape essentielle qui permet de comprendre l'environnement de l'entreprise et de définir les objectifs à atteindre. Nous commencerons par présenter l'organisme d'accueil, puis nous aborderons le projet en identifiant la problématique à résoudre. Nous analyserons également l'existant avant de proposer une solution adaptée.

1.2 Présentation de l'organisme d'accueil

1.2.1 Organisme d'accueil :

Think Trend est une entreprise spécialisée dans le développement et la gestion de stratégies de marketing digital innovantes. Nous offrons une gamme complète de services conçus pour aider les entreprises à se démarquer sur le marché numérique, à attirer leur public cible et à atteindre leurs objectifs commerciaux.



FIGURE 1.1 – Logo de société

1.2.2 Domaines d'activités

L'entreprise **Think Trend** propose une gamme variée de services dans plusieurs domaines clés. Voici un aperçu des principales prestations offertes :

- **Stratégie Marketing :**

- Élaboration de stratégies marketing sur mesure, adaptées aux besoins spécifiques de chaque client.
- Développement de plans d'action visant à optimiser la présence numérique des entreprises.
- **Gestion des Réseaux Sociaux :**
 - Création et gestion de contenus engageants pour les différentes plateformes sociales.
 - Analyse des performances des campagnes et ajustements stratégiques pour maximiser leur impact.
- **Création de Contenu :**
 - Rédaction de contenus marketing percutants, parfaitement adaptés à l'audience cible.
 - Conception de visuels attractifs et production de vidéos promotionnelles de qualité.
- **Publicité en Ligne :**
 - Mise en place et gestion de campagnes publicitaires ciblées sur les réseaux sociaux et les moteurs de recherche.
 - Optimisation des dépenses publicitaires afin d'assurer un meilleur retour sur investissement.
- **Développement Web :**
 - Conception et développement de sites web optimisés pour favoriser les conversions.
 - Intégration de fonctionnalités e-commerce et de solutions personnalisées selon les besoins des clients.
- **Analyse et Reporting :**
 - Suivi détaillé des performances des campagnes et des actions marketing.
 - Fourniture de rapports complets accompagnés de recommandations pour améliorer les résultats obtenus.

1.3 Présentation du projet

1.3.1 Cadre de projet

Dans cette partie, nous posons la problématique de notre projet et nous proposons la solution adéquate pour résoudre les difficultés et améliorer l'activité de l'entreprise.

1.3.2 Problématique

La problématique dans mon projet est Comment mettre en place une solution capable d'optimiser la gestion des projets en réduisant les ressources, tout en permettant de suivre efficacement l'évolution des projets et de connaître en temps réel l'avancement des différentes étapes.

1.3.3 Etude de l'existant

Suite à une étude du marché, j'ai identifié la plateforme Trello [18] comme une solution pertinente pour m'aider à résoudre ma problématique.

Trello est une plateforme de gestion de projets en ligne qui repose sur un système de tableaux, de listes et de cartes pour organiser et suivre les tâches. Parmi ses fonctionnalités, on trouve le suivi des tâches avec des descriptions dans des listes différentes et permet aux utilisateurs d'inviter des membres à collaborer sur des projets.

Les avantages de Trello incluent sa facilité d'utilisation, sa flexibilité pour s'adapter à différents types de projets, sa capacité à favoriser une collaboration transparente et son accessibilité sur plusieurs plateformes. Toutefois, Trello présente certaines limites, notamment pour les projets complexes, où il manque des fonctionnalités avancées pour la gestion des dépendances et la difficulté à suivre plusieurs projets simultanément.

1.3.4 Solution proposée

La solution proposée consiste à développer une nouvelle fonctionnalité dans le plateforme de l'entreprise qui faire la gestion de projets centralisée et automatisée. Cette plateforme permettra de regrouper toutes les données liées aux projets sur une seule interface, facilitant ainsi l'organisation et le suivi. Les responsables pourront créer de nouveaux projets, définir les tâches à réaliser et les assigner aux membres de l'équipe. La plateforme offrira une vue d'ensemble en temps réel sur l'avancement des projets, permettant d'ajuster les priorités et d'optimiser les ressources. De plus, elle fournira un historique complet des actions, permettant une traçabilité et un suivi régulier. Cette solution contribuera à accroître la performance, l'efficacité et la réactivité de l'entreprise face aux besoins des clients.

1.4 Conclusion

Ce chapitre a mis en avant l'importance d'une gestion efficace des projets pour optimiser les ressources et améliorer la performance des entreprises. En réponse aux limites des solutions existantes, nous avons proposé le développement d'une plateforme centralisée et automatisée pour faciliter la collaboration et garantir une visibilité en temps réel sur l'avancement des projets. Le prochain chapitre se concentrera sur la spécification des besoins pour la mise en œuvre de cette solution.

chapitre 2

Spécification des besoins

2.1 Introduction

Maintenant, nous allons approfondir l'étude des exigences. Au niveau de ce chapitre, nous allons accéder dans un premier temps à la partie d'analyse des besoins, où nous allons identifier les acteurs et découvrir les besoins fonctionnels et non fonctionnels auxquels doit répondre notre solution. Dans un second temps, nous allons modéliser les fonctionnalités par le diagramme de cas d'utilisation globale, tout en établissant un backlog produit et en précisant la méthodologie choisie pour guider le développement de la solution. .

2.2 Identification des besoins fonctionnels

Les besoins fonctionnels décrivent des fonctionnalités ou des services spécifiques que le système doit fournir pour répondre aux attentes des utilisateurs. Ils se concentrent sur ce que le système doit faire pour accomplir ses objectifs. Les besoins fonctionnels décrivent des fonctionnalités ou des services spécifiques que le système doit fournir pour répondre aux attentes des utilisateurs. Ils se concentrent sur ce que le système doit faire pour accomplir ses objectifs. Notre système permet de :

- **Gestion des projets :**
 - Afficher la liste des projets avec leurs détails (nom, description, date de début, etc.).
 - Ajouter un nouveau projet via un formulaire.
 - Modifier les détails d'un projet existant.
 - Supprimer un projet avec confirmation.
- **Suivi et étude des projets :**
 - Créer, modifier et supprimer des listes de tâches pour chaque projet.
 - Gérer les tâches dans chaque liste (ajouter, éditer, supprimer).
 - Ajouter des membres à un projet via une invitation par email pour collaborer sur les tâches.
- **Gestion des plans d'actions :**
 - Consulter, ajouter, éditer et supprimer les plans d'actions associés aux projets.

2.3 Identification des besoins non fonctionnels

Les besoins non fonctionnels sont importants car ils agissent de façon indirecte sur le résultat et sur le rendement de l'utilisateur, ce qui fait qu'ils ne doivent pas être négligés,

pour cela il faut répondre aux exigences suivantes.

- **Performances :**
 - Le système doit être rapide et réactif, même lorsque de nombreux projets, listes et tâches sont gérés simultanément.
- **Fiabilités :**
 - L'application doit fonctionner de façon cohérente sans erreur et doit être satisfaisante.
- **L'utilisabilité :**
 - Les interfaces doivent être conviviales et faciles à utiliser, avec des modales pour les formulaires d'ajout/édition/suppression, afin d'éviter les rechargements de page.
 - Pagination pour naviguer efficacement à travers les projets et les plans d'actions.
- **Gestion des erreurs :**
 - Le système doit afficher des messages d'erreur clairs si un champ obligatoire est manquant ou mal rempli.
 - Validation du formulaire avant soumission pour s'assurer que toutes les informations nécessaires sont présentes.
- **La Maintenabilité :**
 - Le code doit être facile à maintenir pour diverses raisons, telles que les mises à jour techniques ainsi que l'amélioration du contenu, afin d'assurer une meilleure performance de l'application.

2.4 Identification des acteurs

En UML, un acteur représente une entité externe, telle qu'un utilisateur ou un système, interagissant avec le système logiciel en question. Notre conception de projet présente globalement un acteur est le responsable de projet.

Le responsable de projet est chargé de superviser l'ensemble des activités liées à la gestion des projets au sein de l'entreprise. Il maintient à jour la liste des projets, y compris leurs détails, et est responsable de l'ajout, de la modification et de la suppression de projets. En outre, il crée et gère les listes de tâches, et désigne une équipe de membres pour réaliser chaque projet. Le responsable de projet est également impliqué dans la gestion des plans d'action, en consultant, ajoutant, modifiant et supprimant ces derniers selon les besoins. Ce rôle requiert des compétences en gestion, communication et organisation pour garantir l'efficacité et la réussite des projets.

2.5 Diagramme de cas d'utilisation structurée

Diagramme [17] de cas d'utilisation a comme objectif de déterminer ce que chaque utilisateur attend du système. Notre extraction de besoins est basée sur la représentation de l'interaction entre les utilisateurs et les futures fonctions du système.

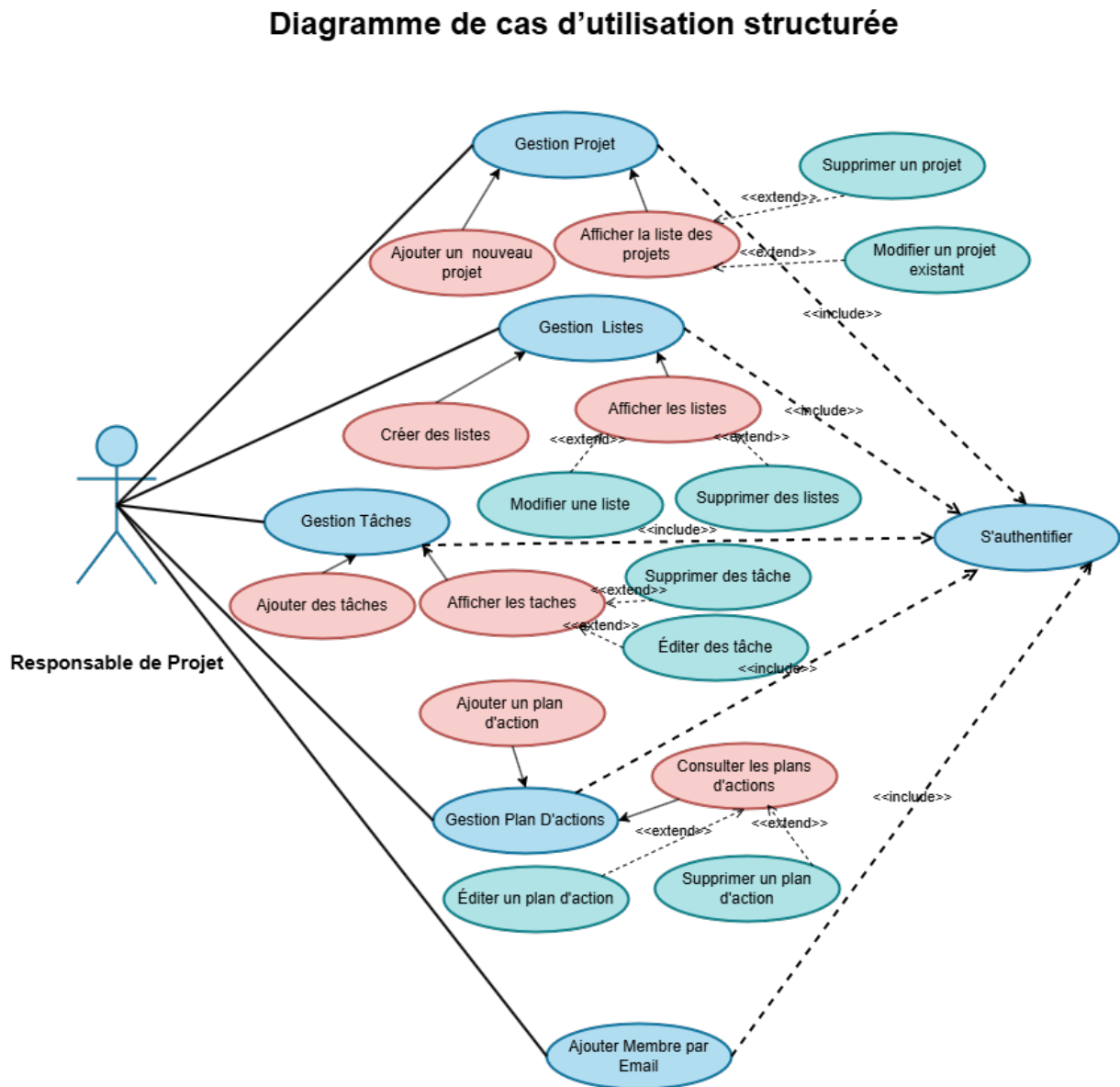


FIGURE 2.2 – Diagramme de cas d'utilisation structurée

2.6 Backlog de produit

Après avoir identifié les exigences fonctionnelles de notre système, nous présentons dans cette section le Backlog de Produit.

Backlog de Produit	Priorité	Estimation	Planification
En tant que responsable de projet, je peux consulter la liste des projets avec leurs détails (nom, description, date de début, etc.)	1	Forte	Sprint 0
En tant que responsable de projet, je peux ajouter un nouveau projet via un formulaire	1	Forte	Sprint 0
En tant que responsable de projet, je peux modifier les détails d'un projet existant	1	Forte	Sprint 0
En tant que responsable de projet, je peux supprimer un projet avec confirmation	1	Forte	Sprint 0
En tant que responsable de projet, je peux créer, modifier et supprimer des listes de tâches d'un projet	2	Forte	Sprint 1
En tant que responsable de projet, je peux gérer les tâches dans chaque liste (ajouter, éditer, supprimer)	2	Moyenne	Sprint 1
En tant que responsable de projet, je peux ajouter des membres à un projet via une invitation par email	3	Moyenne	Sprint 1
En tant que responsable de projet, je peux consulter, ajouter, éditer et supprimer les plans d'actions des projets	4	Forte	Sprint 0

TABLE 2.1 – Backlog de produit

2.7 Environnement de Travail

La méthodologie c'est un système de pratiques techniques et procédures utilisées par ceux qui travaillent dans une discipline.

2.7.1 Méthodologie de Conception

Dans cette section, nous allons présenter la méthodologie Scrum que nous avons choisie pour concevoir et développer notre système, ainsi que le langage de modélisation que nous avons choisi d'utiliser.

Scrum est une méthode de gestion de projet agile qui s'applique principalement dans les projets de développement de logiciels. Il s'agit d'un Framework qui se base sur une

structure d'équipe flexible et collaborative visant à répondre rapidement aux changements et à maximiser la valeur ajoutée au client. L'objectif de Scrum est de développer, maintenir et livrer des produits complexes en favorisant la collaboration, la responsabilité et l'amélioration itérative. Scrum se compose de trois principaux intervenants :

- **Product Owner** : C'est l'expert du produit qui représente les parties prenantes. Il peut également être le client, le propriétaire du projet ou le dirigeant de l'entreprise.
- **Scrum Master** : Il joue le rôle de facilitateur de l'équipe, garantissant que les principes et pratiques Scrum sont bien compris et respectés. Il guide l'équipe et organise le travail de manière à optimiser la performance.
- **Scrum Team** : Il s'agit d'un groupe de professionnels (développeurs, programmeurs, designers, etc.) qui collaborent ensemble pour la réalisation du produit[10].

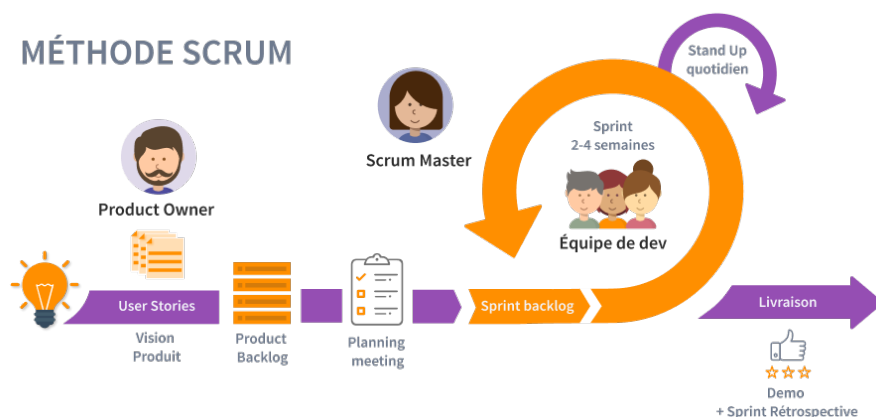





FIGURE 2.3 – La processus de Scrum

2.7.2 Environnement Logiciel

Dans notre projet, nous avons utilisé les logiciels suivants :

	<p>[3]le git est Un système de gestion de version distribué, utilisé pour suivre les modifications dans le code source au fil du temps et collaborer efficacement avec d'autres développeurs.</p>
	<p>[7]Postman est Un outil qui permet de tester et de documenter les API, facilitant ainsi l'interaction avec les services web et les échanges de données.</p>
	<p>GitHub est une plateforme en ligne basée sur Git, un système de contrôle de version distribué. Elle permet aux développeurs de collaborer sur des projets de manière efficace, en hébergeant et en gérant du code source.</p>

	<p>[5] Slack est Un outil de communication d'équipe qui permet de centraliser les discussions, les fichiers et les projets en un seul endroit.</p>
	<p>[9] Google meet est Une plateforme de visioconférence proposée par Google pour organiser des réunions en ligne.</p>
	<p>[13] Un éditeur de code léger et extensible, très populaire chez les développeurs pour écrire et déboguer du code.</p>
	<p>[14]Spring Tool Suite est Un IDE basé sur Eclipse, conçu pour le développement d'applications Spring, en facilitant le développement Java et Spring Boot.</p>
	<p>[15] Xampp Un environnement de serveur web local qui permet aux développeurs de tester des applications PHP, MySQL, Apache et Perl.</p>
	<p>[16] Un système de gestion de base de données relationnelle (SGBDR) open-source, utilisé pour stocker et organiser des données.</p>
	<p>[19]Draw.io est un outil en ligne gratuit permettant de créer, modifier et partager des diagrammes tels que des organigrammes, des diagrammes UML, des diagrammes de flux ou encore des maquettes..</p>

TABLE 2.2 – Logiciels et leurs définitions

2.7.3 Technologies utilisées

Dans notre projet, nous avons utilisé les technologies suivantes :

	<p>[4]Tailwind CSS est un framework CSS utilitaire pour créer des interfaces utilisateur en utilisant des classes prédéfinies.</p>
	<p>HTML [11] (HyperText Markup Language) est le langage de balisage standard pour structurer le contenu des pages web.</p>
	<p>[12]CSS (Cascading Style Sheets) est un langage de style utilisé pour décrire l'apparence visuelle des documents HTML.</p>
	<p>Angular[2] est un framework JavaScript/TypeScript utilisé pour créer des applications web côté client, surtout des applications à une seule page (SPA).</p>
	<p>[1] Spring Boot est un framework Java basé sur Spring, conçu pour créer des applications back-end en simplifiant la configuration et le déploiement.</p>

TABLE 2.3 – Technologies utilisées

2.8 Architecture utilisée

Parmi les exigences du Product Owner, l'architecture de l'application doit être en microservices.

Le terme **micro-service** est utilisé pour décrire une façon de concevoir des logiciels. Cela consiste à créer une application en utilisant plusieurs services indépendants, chacun spécialisé dans une compétence métier particulière. Cela permet une application modulaire et chaque service peut être déployé indépendamment. Les micro-services fonctionnent

comme des petits programmes qui reçoivent des demandes, les traitent et envoient une réponse.

De nombreuses entreprises décident de reconstruire leurs applications monolithiques en utilisant l'approche des micro-services pour leurs avantages, parmi lesquels on cite :

- **Résilience et performance** : Chaque composant étant indépendant, une panne dans un micro-service n'affecte pas les autres, ce qui réduit les dysfonctionnements globaux et facilite la maintenance en localisant rapidement les erreurs.
- **Gestion de données décentralisée** : Grâce à l'approche Polyglot Persistence, chaque micro-service gère sa propre base de données, ce qui améliore la cohérence des données et permet l'utilisation de différentes technologies pour chaque service.
- **Réduction du délai de mise en marché** : Les micro-services étant indépendants, ils peuvent être mis à jour et déployés individuellement sans attendre la finalisation de l'ensemble de l'application.
- **Réutilisation et extensibilité** : Chaque micro-service peut être étendu ou réutilisé de manière indépendante, rendant le système plus flexible et efficace par rapport aux applications monolithiques.
- **Agilité technologique** : Cette architecture permet d'utiliser différentes technologies pour chaque micro-service, offrant plus de flexibilité et la possibilité d'avoir des équipes de développement distinctes pour chaque service.

2.9 Conclusion

Dans ce chapitre, nous avons présenté le diagramme de cas d'utilisation, ce qui nous a permis de réaliser le découpage fonctionnel de notre système. Dans le chapitre suivant, nous détaillerons la première version en exposant la conception et la réalisation.

chapitre 3

Sprint 0 : l'élaboration des projets

3.1 Introduction

Ce chapitre sert à présenter le premier sprint de notre projet. Nous allons modéliser les fonctionnalités par des diagrammes de cas d'utilisation, diagramme de classe et diagrammes de séquences.

3.2 Identification de Backlog de Sprint 0

Dans cette section, nous présentons le BackLog de sprint 0 :

Backlog de Produit	Priorité	Estimation	Planification
En tant que responsable de projet, je peux consulter la liste des projets avec leurs détails (nom, description, date de début, etc.)	1	Forte	Sprint 0
En tant que responsable de projet, je peux ajouter un nouveau projet via un formulaire	1	Forte	Sprint 0
En tant que responsable de projet, je peux modifier les détails d'un projet existant	1	Forte	Sprint 0
En tant que responsable de projet, je peux supprimer un projet avec confirmation	1	Forte	Sprint 0
En tant que responsable de projet, je peux rechercher un projet par son nom	2	Moyenne	Sprint 0
En tant que responsable de projet, je peux consulter les plans d'actions des projets	1	Forte	Sprint 0
En tant que responsable de projet, je peux ajouter un plan d'action pour un projet via un formulaire	1	Forte	Sprint 0
En tant que responsable de projet, je peux éditer un plan d'action existant	1	Forte	Sprint 0
En tant que responsable de projet, je peux supprimer un plan d'action avec confirmation	1	Forte	Sprint 0
En tant que responsable de projet, je peux rechercher un plan d'action par date ou par nom de projet	2	Moyenne	Sprint 0
En tant que responsable de projet, je peux consulter le tableau Kanban des tâches d'un projet à partir d'un lien	1	Forte	Sprint 0

TABLE 3.4 – Backlog de produit détaillé de sprint 0

3.3 Raffinement du Sprint 0

Ce sprint met sur la voie de développer la première partie de notre projet :

- Gestion des projets.
- Gestion des plans d'actions.

3.3.1 Raffinement de cas d'utilisation « Gestion des projets » :

Gestion des projets comportent consulter listes des projets , ajouter un projet , éditer un projet , supprimer un projet et chercher un projet.

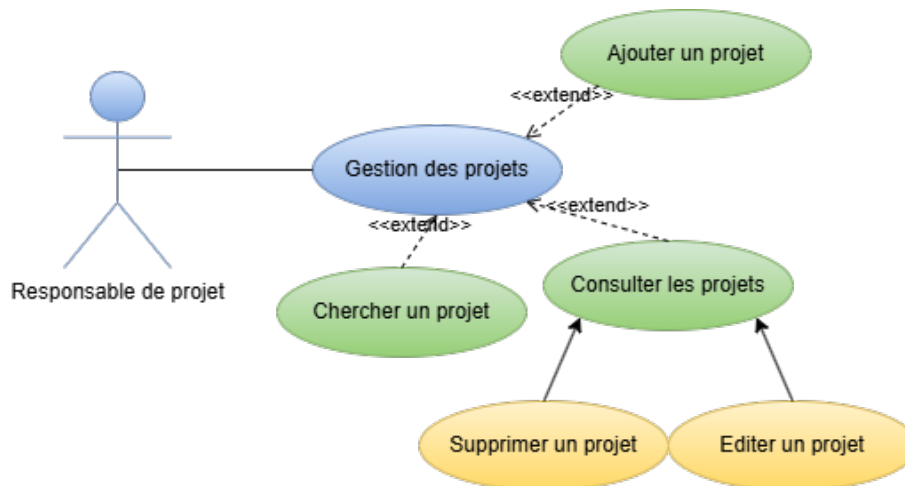


FIGURE 3.4 – Raffinement du cas d'utilisation "Gestion des projets"

Cas d'utilisation	Ajouter un projet
Acteur	Responsable de projet
Pré-condition	-L'utilisateur est authentifié et dispose des droits nécessaires. -Le système est opérationnel.
Post-Condition	-Le projet est ajouté avec succès et visible dans la liste des projets.
Description de scénario principal	-Le système affiche un formulaire d'ajout de projet vide. -L'utilisateur remplit les champs requis et clique sur "Ajouter". -Le projet est ajouté à la base de données.
Scénario alternatif	-Erreur si des champs requis sont vides (le système affiche un message d'erreur).

TABLE 3.5 – Raffinement du cas d'utilisation "Ajouter un projet"

Cas d'utilisation	Editer un projet
Acteur	Responsable de projet
Pré-condition	-L'utilisateur est authentifié et dispose des droits nécessaires. -Le projet existe dans la base de données.
Post-Condition	-Le projet est modifié avec succès et les changements sont sauvegardés.
Description de scénario principal	-Le système affiche un formulaire pré-rempli avec les informations actuelles du projet. -L'utilisateur modifie les champs nécessaires et clique sur "Enregistrer". -Les modifications sont enregistrées dans la base de données.
Scénario alternatif	-Erreur si des champs requis sont laissés vides (le système affiche un message d'erreur).

TABLE 3.6 – Raffinement du cas d'utilisation "Editer un projet"

Cas d'utilisation	Supprimer un projet
Acteur	Responsable de projet
Pré-condition	-L'utilisateur est authentifié et dispose des droits nécessaires. -Le projet existe dans la base de données.
Post-Condition	-Le projet est supprimé avec succès ou l'opération est annulée.
Description de scénario principal	-L'utilisateur clique sur "Supprimer" à côté d'un projet. -Le système affiche un modal de confirmation. -L'utilisateur confirme la suppression, et le projet est supprimé de la base de données.
Scénario alternatif	-L'utilisateur annule l'opération dans le modal de confirmation (le projet n'est pas supprimé).

TABLE 3.7 – Raffinement du cas d'utilisation "Supprimer un projet"

Cas d'utilisation	Consulter la liste des projets
Acteur	Responsable de projet
Pré-condition	-L'utilisateur est authentifié et dispose des droits nécessaires. -Le système est connecté à la base de données.
Post-Condition	-La liste des projets est affichée.
Description de scénario principal	-L'utilisateur clique sur "Liste des projets". -Le système récupère les données et affiche la liste des projets.
Scénario alternatif	-En cas de problème de connexion à la base de données , la liste n'est pas chargée.

TABLE 3.8 – Raffinement du cas d'utilisation "Lister les projets"

Cas d'utilisation	Chercher un projet
Acteur	Responsable de projet
Pré-condition	-L'utilisateur est authentifié et dispose des droits nécessaires. -Le système est opérationnel et la barre de recherche est disponible.
Post-Condition	-Le projet correspondant au nom recherché est affiché dans la liste des projets.
Description de scénario principal	-L'utilisateur entre le nom du projet dans la barre de recherche. -Le système filtre la liste des projets en fonction du nom saisi. -Les résultats correspondants sont affichés à l'utilisateur.
Scénario alternatif	-Aucun projet ne correspond à la recherche ,le système n'affiche aucun projet.

TABLE 3.9 – Raffinement du cas d'utilisation "Chercher un projet "

3.3.2 Raffinement de cas d'utilisation « Gestion plans d'actions » :

Gestion des plans d'actions comportent consulter listes des plans d'actions , ajouter un plan d'action , éditer un plan d'action , supprimer un plan d'action et chercher un plan d'action.

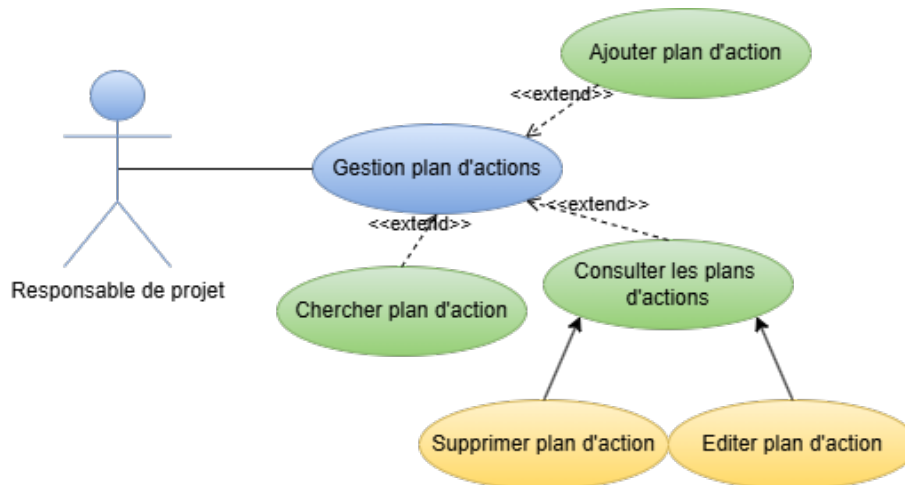


FIGURE 3.5 – Raffinement du cas d'utilisation "Gestion plans d'actions"

Cas d'utilisation	Ajouter un plan d'action
Acteur	Responsable de projet
Pré-condition	-L'utilisateur est authentifié et dispose des droits nécessaires. -Le système est opérationnel.
Post-Condition	-Le plan d'action est ajouté avec succès et visible dans la liste des plans d'action.
Description de scénario principal	-Le système affiche un formulaire d'ajout de plan d'action vide. -L'utilisateur remplit les champs requis et clique sur "Ajouter". -Le plan d'action est ajouté à la base de données.
Scénario alternatif	-Erreur si des champs requis sont vides (le système affiche un message d'erreur).

TABLE 3.10 – Raffinement du cas d'utilisation "Ajouter un plan d'action"

Cas d'utilisation	Editer un plan d'action
Acteur	Responsable de projet
Pré-condition	-L'utilisateur est authentifié et dispose des droits nécessaires. -Le plan d'action existe dans la base de données.
Post-Condition	-Le plan d'action est modifié avec succès et les changements sont sauvegardés.
Description de scénario principal	-Le système affiche un formulaire pré-rempli avec les informations actuelles du plan d'action. -L'utilisateur modifie les champs nécessaires et clique sur "Enregistrer". -Les modifications sont enregistrées dans la base de données.
Scénario alternatif	-Erreur si des champs requis sont laissés vides (le système affiche un message d'erreur).

TABLE 3.11 – Raffinement du cas d'utilisation "Editer un plan d'action"

Cas d'utilisation	Supprimer un plan d'action
Acteur	Responsable de projet
Pré-condition	-L'utilisateur est authentifié et dispose des droits nécessaires. -Le plan d'action existe dans la base de données.
Post-Condition	-Le plan d'action est supprimé avec succès.
Description de scénario principal	-L'utilisateur clique sur "Supprimer" à côté d'un plan d'action. -Le système affiche un modal de confirmation. -L'utilisateur confirme la suppression, et le plan d'action est supprimé de la base de données.
Scénario alternatif	-L'utilisateur annule l'opération dans le modal de confirmation (le plan d'action n'est pas supprimé).

TABLE 3.12 – Raffinement du cas d'utilisation "Supprimer un plan d'action"

Cas d'utilisation	Consulter la liste des plans d'action
Acteur	Responsable de projet
Pré-condition	-L'utilisateur est authentifié et dispose des droits nécessaires. -Le système est connecté à la base de données.
Post-Condition	-La liste des plans d'action est affichée.
Description de scénario principal	-L'utilisateur clique sur "Liste des plans d'action". -Le système récupère les données et affiche la liste des plans d'action.
Scénario alternatif	-En cas de problème de connexion à la base de données, la liste n'est pas chargée.

TABLE 3.13 – Raffinement du cas d'utilisation "Lister les plans d'action"

Cas d'utilisation	Chercher un plan d'action par date
Acteur	Responsable de projet
Pré-condition	-L'utilisateur est authentifié et dispose des droits nécessaires. -Le système est opérationnel et la barre de recherche par date est disponible.
Post-Condition	-Le plan d'action correspondant à la date recherchée est affiché dans la liste.
Description de scénario principal	-L'utilisateur sélectionne une date dans le champ de recherche par date. -Le système filtre la liste des plans d'action en fonction de la date sélectionnée. -Les résultats correspondants sont affichés à l'utilisateur.
Scénario alternatif	-Aucun plan d'action ne correspond à la date sélectionnée, le système n'affiche aucun résultat.

TABLE 3.14 – Raffinement du cas d'utilisation "Chercher un plan d'action par date"

3.4 Conception du Sprint 0

Dans ce niveau, nous allons modéliser notre système avant de le réaliser. On va voir les cas d'utilisations suivants :

- Gestion des projets.
- Gestion des plans d'actions.

3.4.1 Conception de cas d'utilisation « Gestion des Projets » :

Diagramme de classe

Un diagramme de classe est une représentation statique des objets d'un système, montrant leurs classes, attributs, méthodes et les relations qui existent entre eux. Il est utilisé pour modéliser la structure d'un système à un moment donné.

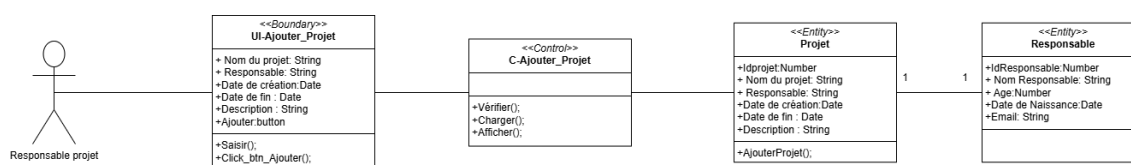


FIGURE 3.6 – Diagramme de classes de cas d'utilisation "Ajouter Projet"

Diagramme de séquence

Un diagramme de séquence est un type de diagramme d'interaction qui illustre la manière dont les objets interagissent entre eux en envoyant des messages dans un ordre chronologique. Il montre la dynamique du système en représentant les échanges entre les différents acteurs et objets.

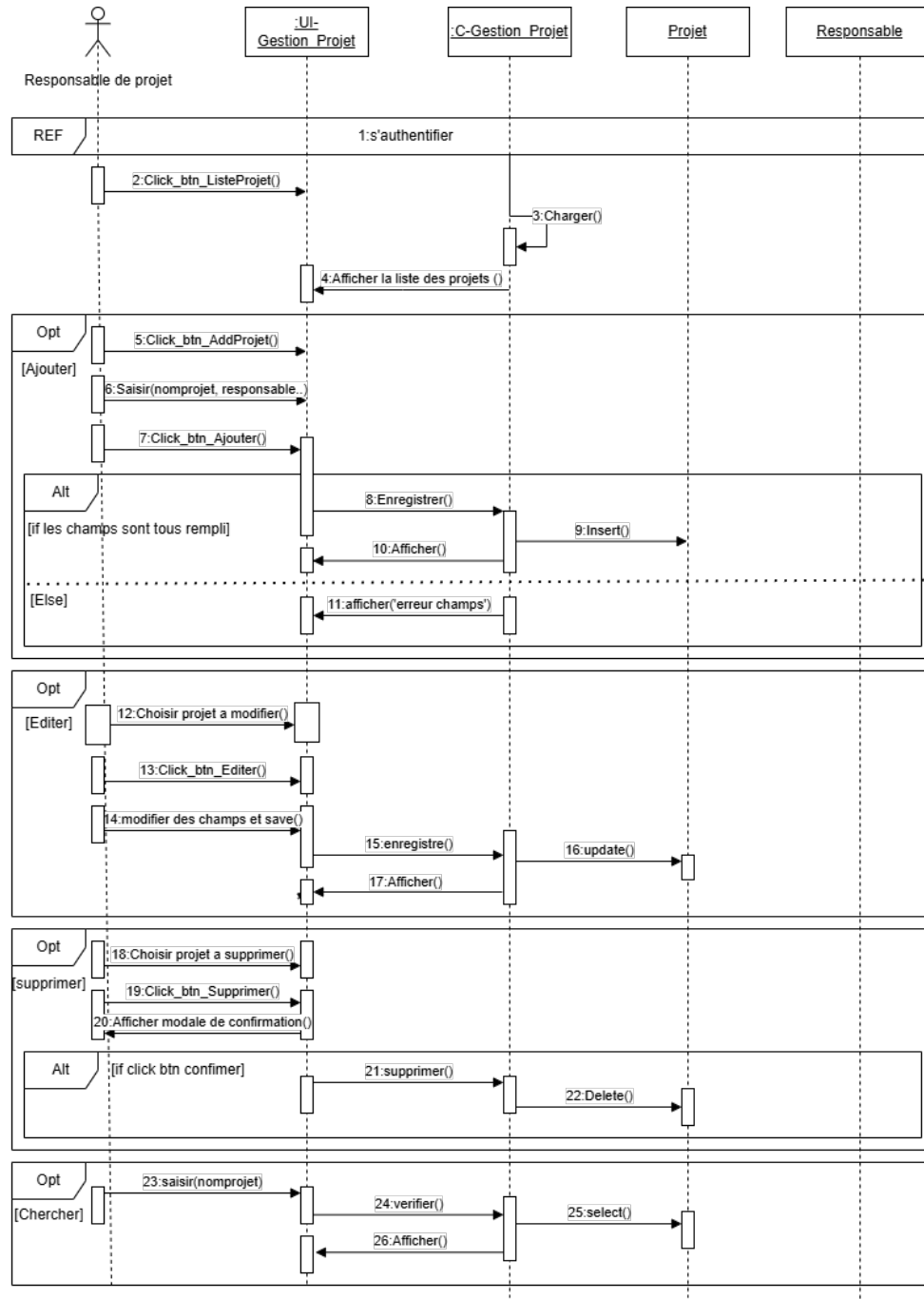


FIGURE 3.7 – Diagramme de séquence de cas d'utilisation "Gestion des Projets"

3.4.2 Conception de cas d'utilisation « Gestion des plans d'actions » :

Diagramme de classe

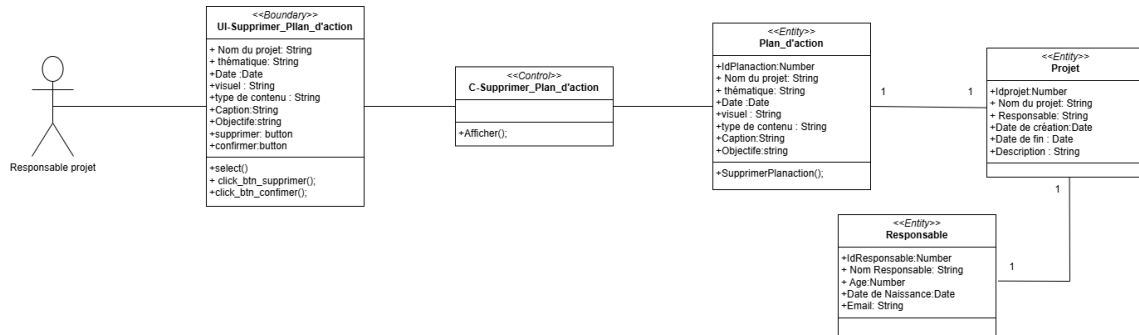


FIGURE 3.8 – Diagramme de classe de cas d'utilisation "Supprimer plan d'action"

Diagramme de séquence

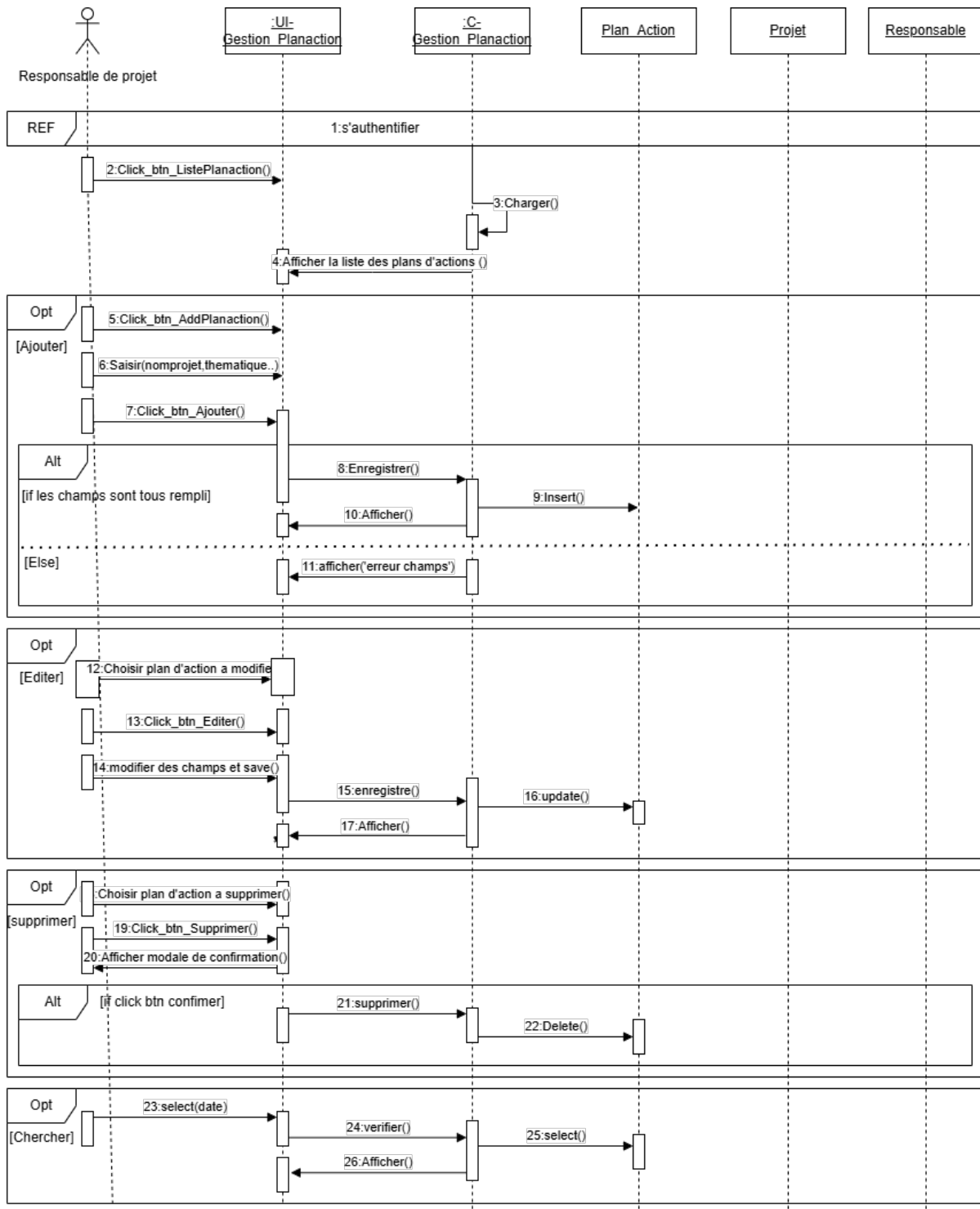


FIGURE 3.9 – Diagramme de séquence de cas d'utilisation "Gestion des plans d'actions"

3.5 Réalisation du Sprint 0

3.5.1 Gestion des projets

-Cette interface permet à l'utilisateur de gérer les projets.

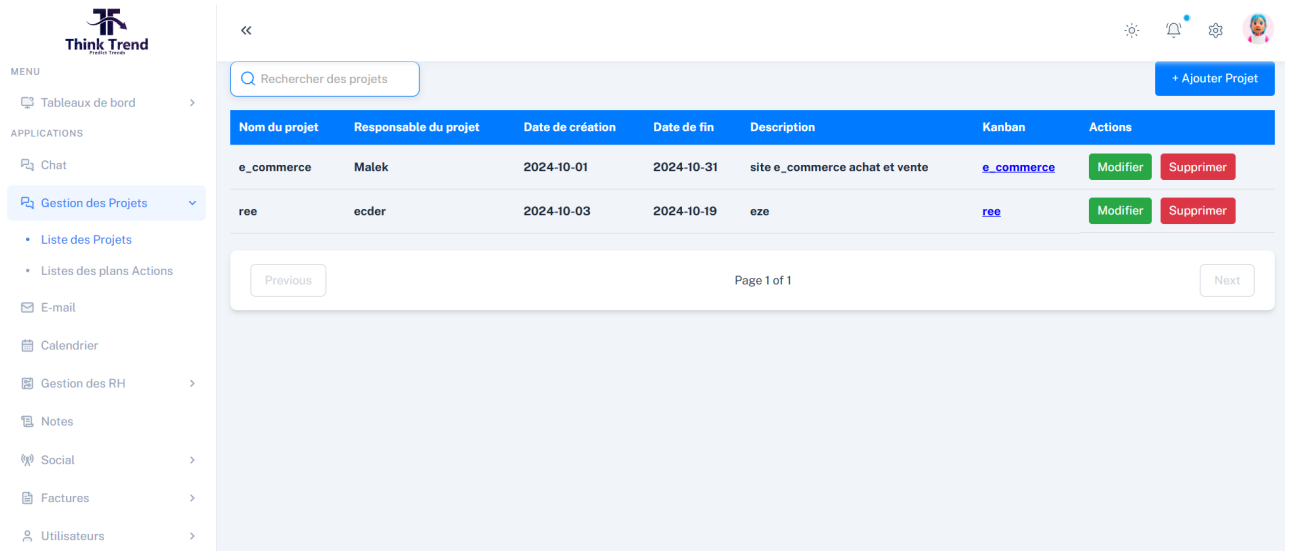


FIGURE 3.10 – Interface gestion projet

-Cette interface permet à l'utilisateur d'ajouter projet.

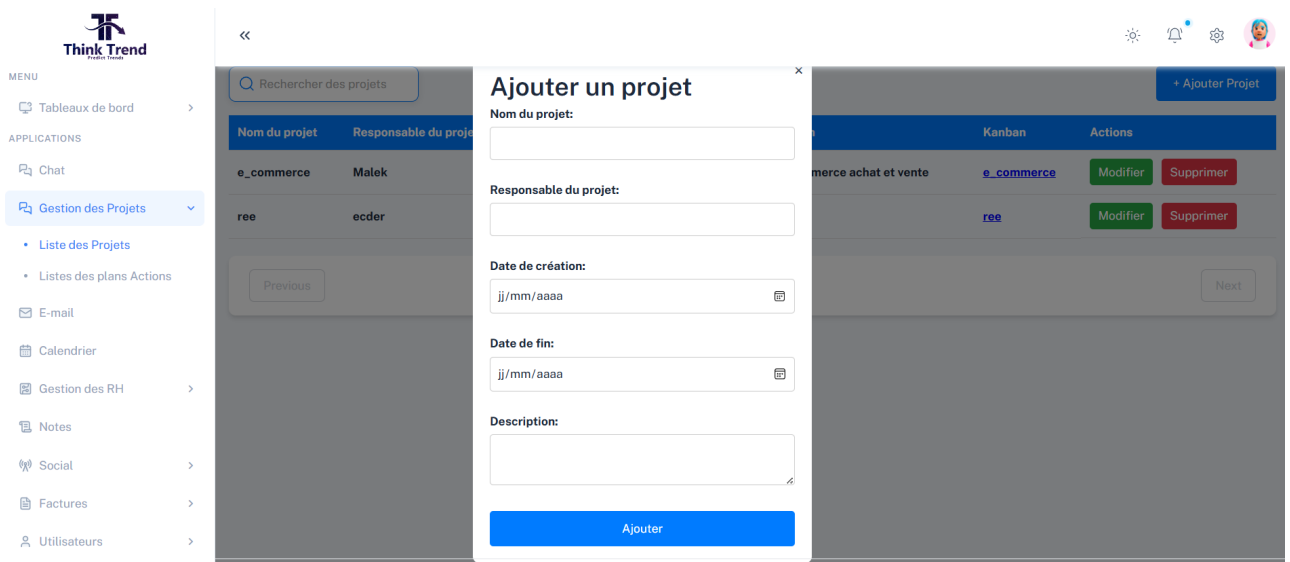


FIGURE 3.11 – Interface Ajouter projet

-Cette interface permet à l'utilisateur d'editer projet.

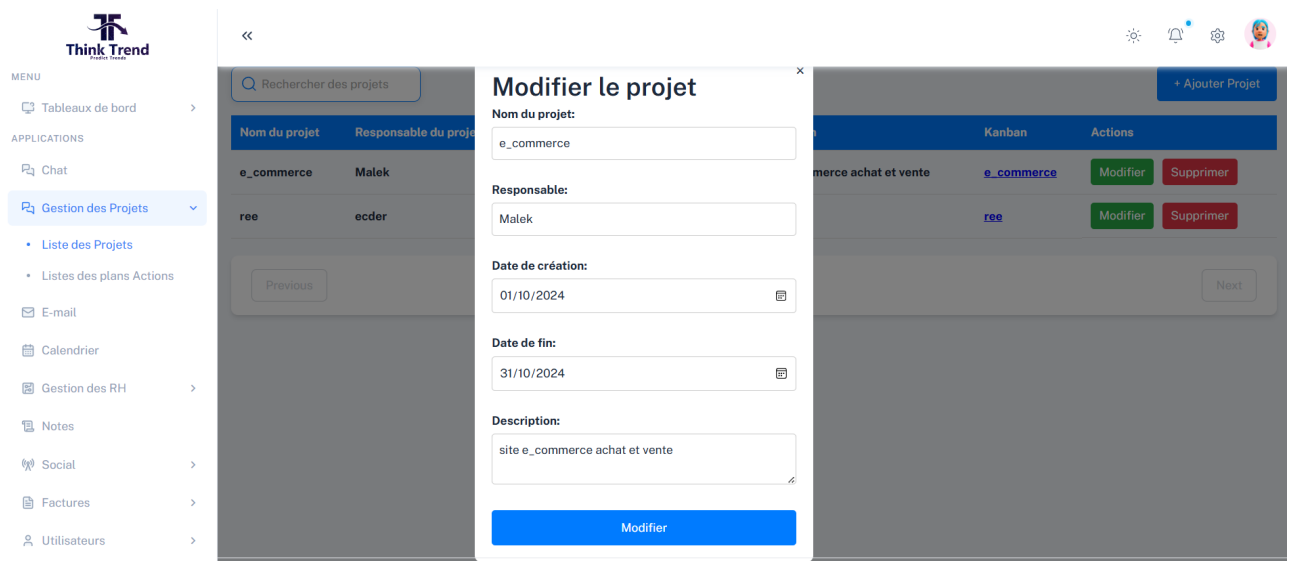


FIGURE 3.12 – Interface Editer projet

-Cette interface permet à l'utilisateur de supprimer projet.

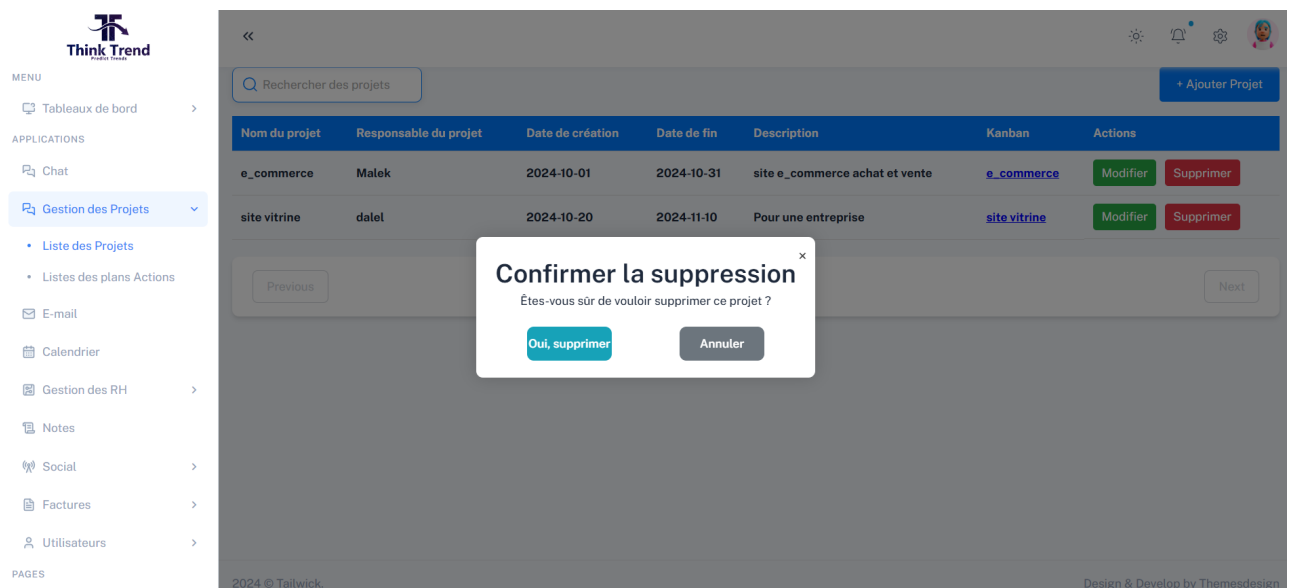


FIGURE 3.13 – Interface supprimer projet

3.5.2 Gestion des plans d'actions

-Cette interface permet à l'utilisateur de gérer les plans d'actions.

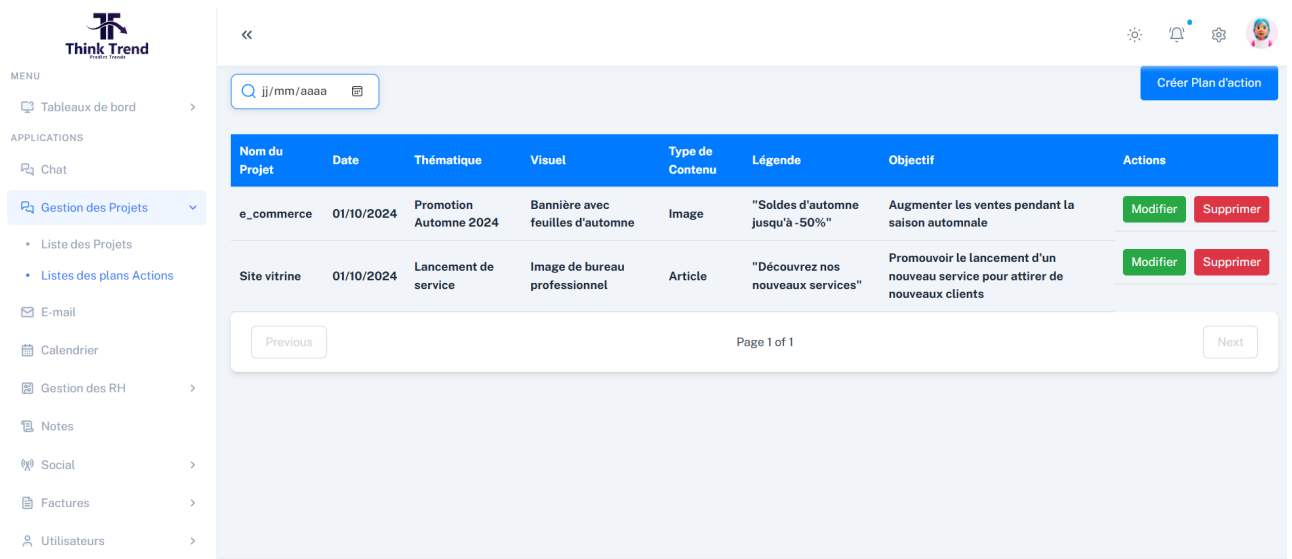


FIGURE 3.14 – Interface gestion plans d'actions

-Cette interface permet à l'utilisateur d'ajouter plan d'action.

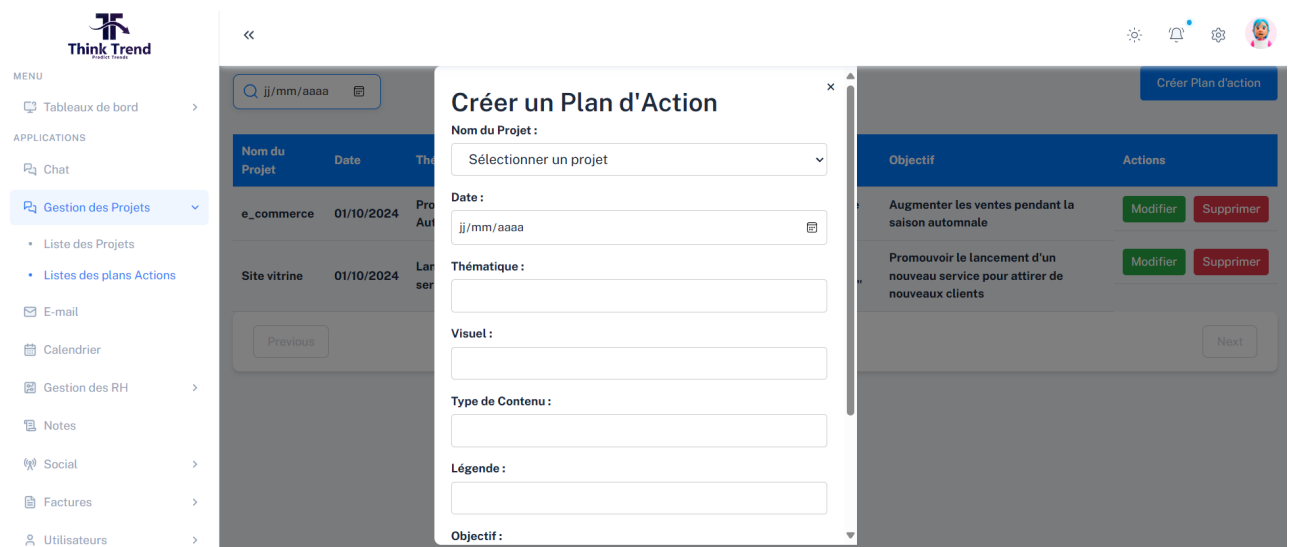


FIGURE 3.15 – Interface Ajouter plan d'action

-Cette interface permet à l'utilisateur d'editer plan d'action.

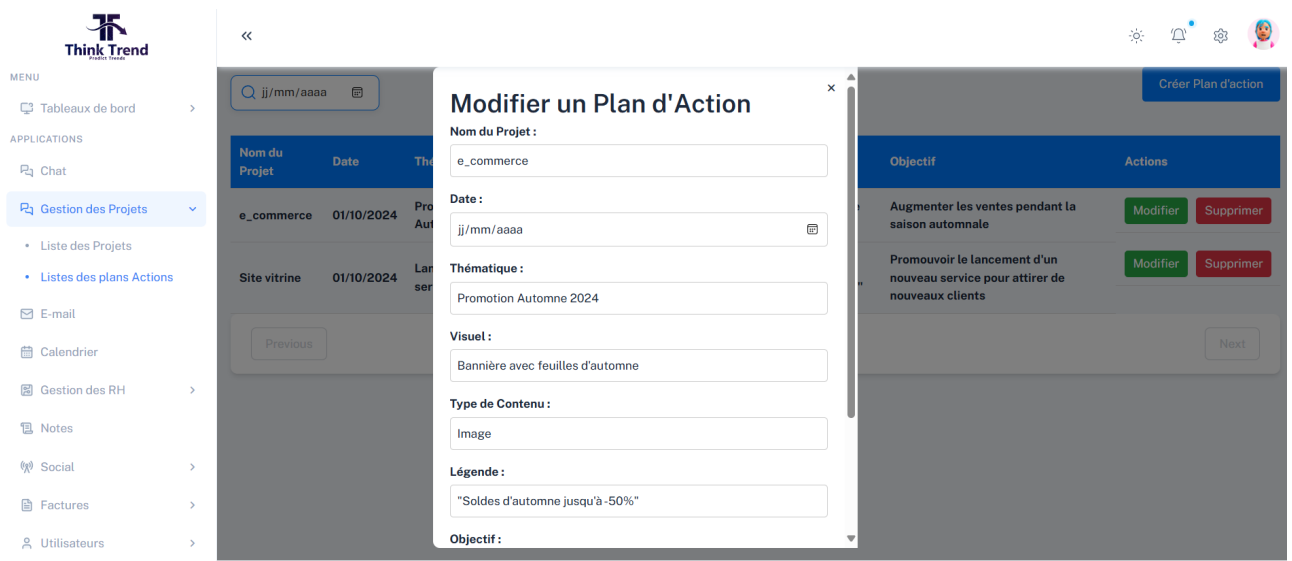


FIGURE 3.16 – Interface editer plans d'action

-Cette interface permet à l'utilisateur de supprimer plan d'action.

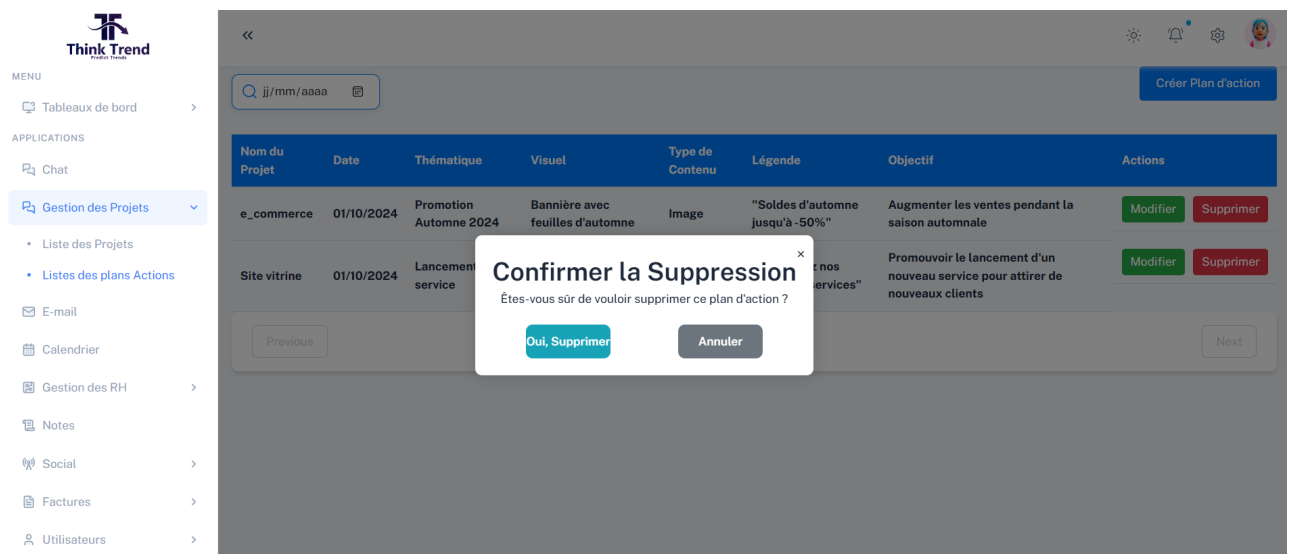


FIGURE 3.17 – Interface supprimer plans d'action

3.6 Conclusion

À travers ce sprint, nous avons effectué une conception détaillée et nous sommes ensuite intéressés par la réalisation du module Gestion projet et Gestion plan d'action.

Nous passons au sprint 1 pour assurer le Suivi des projets.

chapitre 4

Sprint 1 :Le Suivie des projets

4.1 Introduction

Ce chapitre présente le deuxième sprint du projet. L'étude de ce sprint inclut le raffinement des cas d'utilisation ainsi que la conception et la réalisation de trois cas d'utilisation portant sur le suivi des projets.

4.2 Identification de Backlog de Sprint 1

Dans cette section, nous présentons le BackLog de sprint 1 :

Backlog de Produit	Priorité	Estimation	Planification
En tant que responsable de projet, je peux consulter la liste des listes de tâches	1	Forte	Sprint 1
En tant que responsable de projet, je peux ajouter une nouvelle liste de tâches	1	Forte	Sprint 1
En tant que responsable de projet, je peux modifier une liste de tâches existante	2	Forte	Sprint 1
En tant que responsable de projet, je peux supprimer une liste de tâches avec confirmation	2	Forte	Sprint 1
En tant que responsable de projet, je peux consulter la liste des tâches dans une liste de tâches	1	Forte	Sprint 1
En tant que responsable de projet, je peux ajouter une nouvelle tâche dans une liste de tâches	1	Forte	Sprint 1
En tant que responsable de projet, je peux modifier une tâche existante dans une liste de tâches	2	Forte	Sprint 1
En tant que responsable de projet, je peux supprimer une tâche avec confirmation dans une liste de tâches	2	Forte	Sprint 1
En tant que responsable de projet, je peux inviter des membres à collaborer sur un projet via leur email	1	Forte	Sprint 1

TABLE 4.15 – Backlog de produit détaillé de sprint 1

4.3 Raffinement du Sprint 1

Ce sprint met sur la voie de développer la deuxième partie de notre projet :

- Gestion des Listes.
- Gestion des Tâches.
- Ajouter Membres au projet.

4.3.1 Raffinement de cas d'utilisation « Gestion des Listes » :

Gestion des Listes comportent consulter les listes des tâches , ajouter une liste , éditer une liste et supprimer une liste.

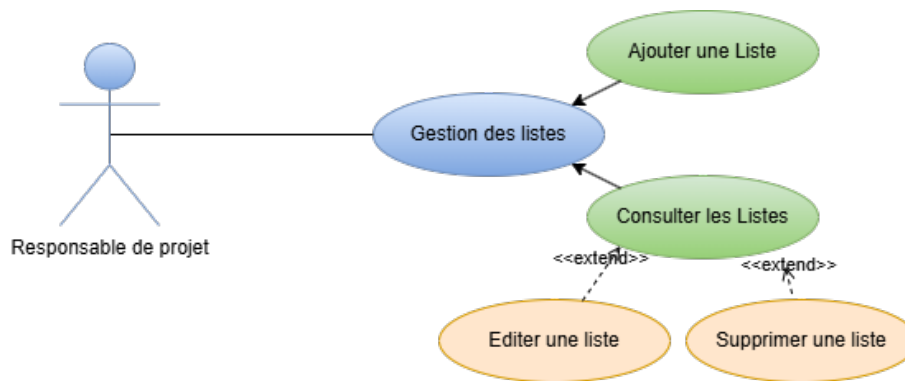


FIGURE 4.18 – Raffinement du cas d'utilisation "Gestion des Listes"

Cas d'utilisation	Ajouter une liste de tâches
Acteur	Responsable de projet
Pré-condition	-L'utilisateur est authentifié et dispose des droits nécessaires. -Le système est opérationnel.
Post-Condition	-La nouvelle liste de tâches est ajoutée à un projet avec succès et visible dans le tableau Kanban.
Description de scénario principal	-Le système affiche un formulaire d'ajout de liste de tâches. -L'utilisateur entre le nom de la liste et clique sur "Ajouter". -La liste de tâches est ajoutée à la base de données et visible dans le tableau Kanban.
Scénario alternatif	-Erreur si le nom de la liste est vide (le système affiche un message d'erreur).

TABLE 4.16 – Raffinement du cas d'utilisation "Ajouter une liste de tâches"

Cas d'utilisation	Editer une liste de tâches
Acteur	Responsable de projet
Pré-condition	-L'utilisateur est authentifié et dispose des droits nécessaires. -La liste de tâches existe dans la base de données.
Post-Condition	-Le nom de la liste de tâches est modifié avec succès et visible dans le tableau Kanban.
Description de scénario principal	-Le système affiche un formulaire avec le nom actuel de la liste de tâches. -L'utilisateur modifie le nom et clique sur "Enregistrer". -Les modifications sont enregistrées dans la base de données.
Scénario alternatif	-Erreur si le nouveau nom est vide (le système affiche un message d'erreur).

TABLE 4.17 – Raffinement du cas d'utilisation "Editer une liste de tâches"

Cas d'utilisation	Supprimer une liste de tâches
Acteur	Responsable de projet
Pré-condition	-L'utilisateur est authentifié et dispose des droits nécessaires. -La liste de tâches existe dans la base de données.
Post-Condition	-La liste de tâches est supprimée avec succès.
Description de scénario principal	-L'utilisateur clique sur "Supprimer" à côté d'une liste de tâches. -Le système affiche un modal de confirmation. -L'utilisateur confirme la suppression, et la liste de tâches est supprimée de la base de données.
Scénario alternatif	-L'utilisateur annule l'opération dans le modal de confirmation (la liste de tâches n'est pas supprimée).

TABLE 4.18 – Raffinement du cas d'utilisation "Supprimer une liste de tâches"

Cas d'utilisation	Consulter la liste des tâches
Acteur	Responsable de projet
Pré-condition	-L'utilisateur est authentifié et dispose des droits nécessaires. -Le système est connecté à la base de données.
Post-Condition	-Les listes de tâches sont affichées dans le tableau Kanban.
Description de scénario principal	-L'utilisateur clique sur un lien du tableau Kanbandd'un projet pour consulter les listes de tâches. -Le système récupère les données et affiche les listes de tâches dans le tableau Kanban.
Scénario alternatif	-En cas de problème de connexion à la base de données, les listes de tâches ne sont pas chargées.

TABLE 4.19 – Raffinement du cas d'utilisation "Consulter la liste des tâches"

4.3.2 Raffinement de cas d'utilisation « Gestion des Tâches » :

Gestion des Tâches comportent ajouter une tâche dans une liste , éditer une tâche et supprimer une tâche.

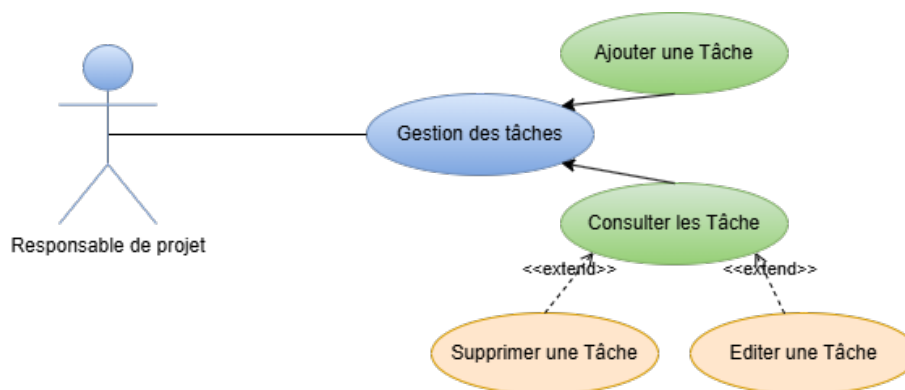


FIGURE 4.19 – Raffinement du cas d'utilisation "Gestion des tâches"

Cas d'utilisation	Ajouter une tâche
Acteur	Responsable de projet
Pré-condition	<ul style="list-style-type: none"> - L'utilisateur est authentifié et dispose des droits nécessaires. - La liste de tâches existe dans le projet.
Post-Condition	<ul style="list-style-type: none"> - La nouvelle tâche est ajoutée à la liste de tâches avec succès et est visible dans le tableau Kanban.
Description de scénario principal	<ul style="list-style-type: none"> - Le système affiche un formulaire d'ajout de tâche. - L'utilisateur entre les détails de la tâche et clique sur "Ajouter". - La tâche est ajoutée à la base de données et visible dans la liste de tâches du tableau Kanban.
Scénario alternatif	<ul style="list-style-type: none"> - Erreur si le nom de la tâche est vide (le système affiche un message d'erreur).

TABLE 4.20 – Raffinement du cas d'utilisation "Ajouter une tâche"

Cas d'utilisation	Editer une tâche
Acteur	Responsable de projet
Pré-condition	<ul style="list-style-type: none"> - L'utilisateur est authentifié et dispose des droits nécessaires. - La tâche existe dans la liste de tâches.
Post-Condition	<ul style="list-style-type: none"> - La tâche est modifiée avec succès et est visible dans le tableau Kanban.
Description de scénario principal	<ul style="list-style-type: none"> - Le système affiche un formulaire avec les informations actuelles de la tâche. - L'utilisateur modifie la description et clique sur "Enregistrer". - Les modifications sont enregistrées dans la base de données.
Scénario alternatif	<ul style="list-style-type: none"> - Erreur si des champs requis sont vides (le système affiche un message d'erreur).

TABLE 4.21 – Raffinement du cas d'utilisation "Editer une tâche"

Cas d'utilisation	Supprimer une tâche
Acteur	Responsable de projet
Pré-condition	<ul style="list-style-type: none"> - L'utilisateur est authentifié et dispose des droits nécessaires. - La tâche existe dans la liste de tâches.
Post-Condition	<ul style="list-style-type: none"> - La tâche est supprimée avec succès de la liste de tâches.
Description de scénario principal	<ul style="list-style-type: none"> - L'utilisateur clique sur "Supprimer" à côté d'une tâche. - Le système affiche un modal de confirmation. - L'utilisateur confirme la suppression, et la tâche est supprimée de la base de données.
Scénario alternatif	<ul style="list-style-type: none"> - L'utilisateur annule l'opération dans le modal de confirmation (la tâche n'est pas supprimée).

TABLE 4.22 – Raffinement du cas d'utilisation "Supprimer une tâche"

4.3.3 Raffinement de cas d'utilisation « Ajouter un Membre » :

Gestion des Tâches comportent ajouter un membre à un projet et Consulter les membres d'un projet.

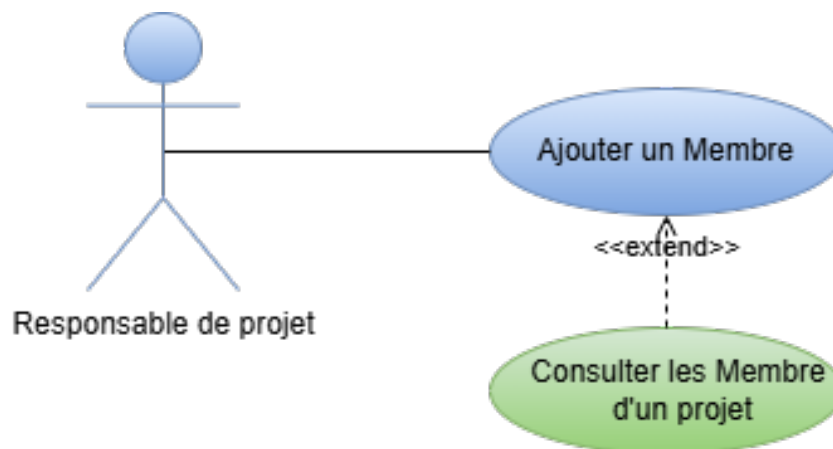


FIGURE 4.20 – Raffinement du cas d'utilisation "Ajouter un Membre"

Cas d'utilisation	Ajouter un membre à un projet à partir d'email
Acteur	Responsable de projet
Pré-condition	<ul style="list-style-type: none"> - L'utilisateur est authentifié et dispose des droits nécessaires. - Le projet existe dans le système.
Post-Condition	<ul style="list-style-type: none"> - Le nouvel membre est ajouté au projet avec succès et reçoit une notification par email.
Description de scénario principal	<ul style="list-style-type: none"> - Le système affiche un formulaire d'ajout de membre. - L'utilisateur entre l'adresse email du membre et clique sur "Ajouter". - Le système valide l'email et ajoute le membre à la base de données. - Un email de notification est envoyé au nouvel utilisateur.
Scénario alternatif	<ul style="list-style-type: none"> - Erreur si l'email est vide ou invalide (le système affiche un message d'erreur). - Erreur si l'utilisateur existe déjà dans le projet (le système affiche un message d'erreur).

TABLE 4.23 – Raffinement du cas d'utilisation "Ajouter un membre à un projet"

Cas d'utilisation	Consulter les membres d'un projet en cliquant sur "Ajouter un membre"
Acteur	Responsable de projet
Pré-condition	<ul style="list-style-type: none"> - L'utilisateur est authentifié et dispose des droits nécessaires. - Le projet existe dans le système.
Post-Condition	<ul style="list-style-type: none"> - La liste des membres du projet est affichée à l'utilisateur.
Description de scénario principal	<ul style="list-style-type: none"> - L'utilisateur clique sur le bouton "Ajouter un membre". - Le système affiche un modal contenant la liste des membres du projet. - L'utilisateur peut voir les informations des membres et éventuellement ajouter de nouveaux membres.
Scénario alternatif	<ul style="list-style-type: none"> - Erreur si le projet n'a pas de membres (le système affiche liste vide).

TABLE 4.24 – Raffinement du cas d'utilisation "Consulter les membres d'un projet"

4.4 Conception du Sprint 1

Dans ce niveau, nous allons modéliser notre système avant de le réaliser. On va voir les cas d'utilisations suivants :

- Ajouter Liste à un projet.
- éditer une Tâche d'un liste.
- Ajouter Membres au projet.

4.4.1 Conception de cas d'utilisation « Ajouter Liste » :

Diagramme de classe

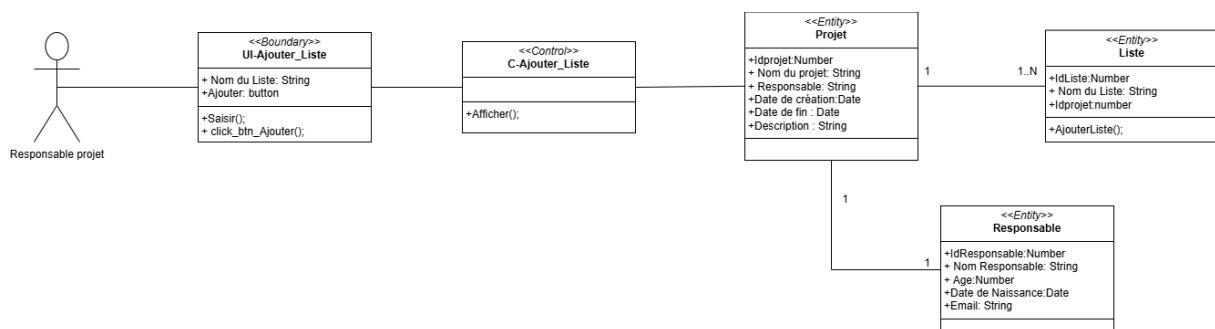


FIGURE 4.21 – Diagramme de classe de cas d'utilisation « Ajouter Liste »

Diagramme de séquence

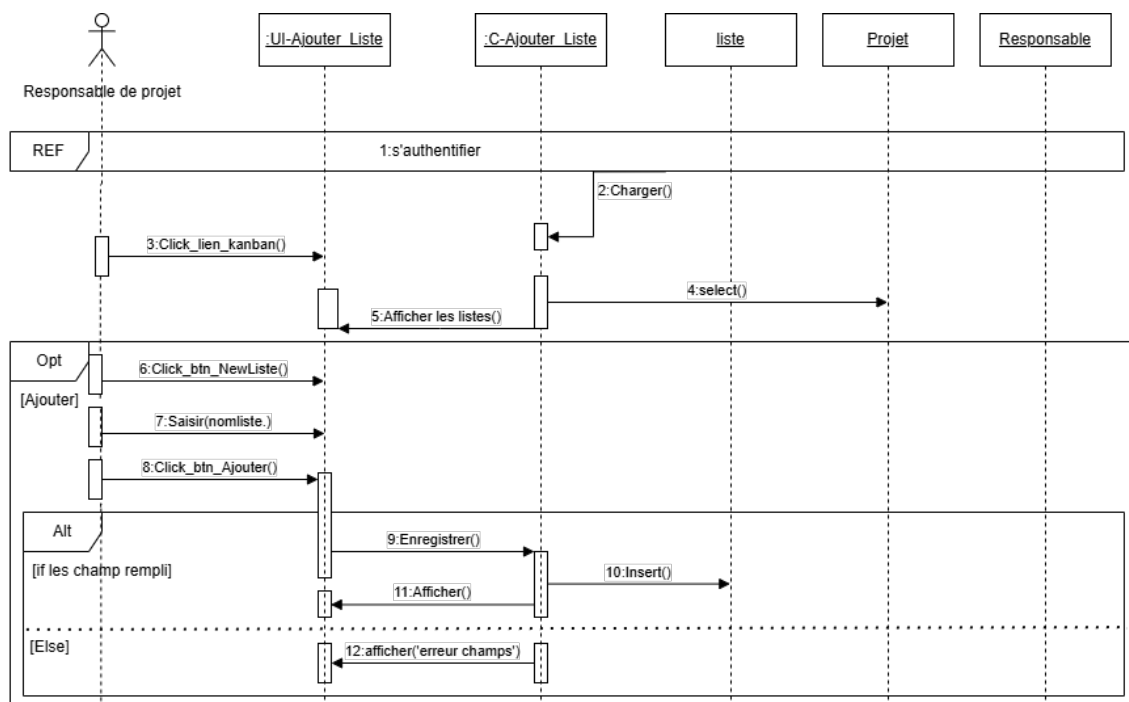


FIGURE 4.22 – Diagramme de Séquence « Ajouter Liste »

4.4.2 Conception de cas d'utilisation « Editer une Tâche » :

Diagramme de classe

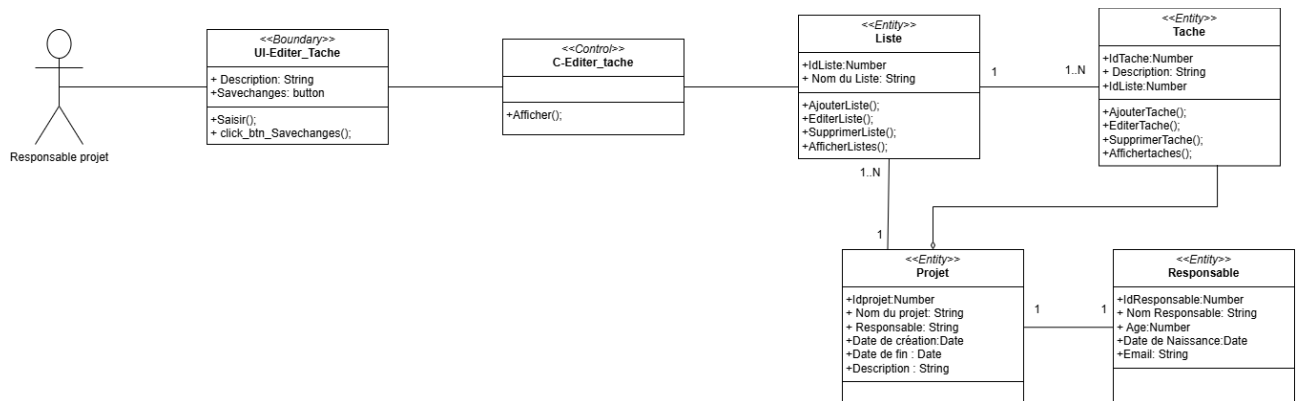


FIGURE 4.23 – Diagramme de classe de cas d'utilisation « Editer une Tâche »

Diagramme de séquence

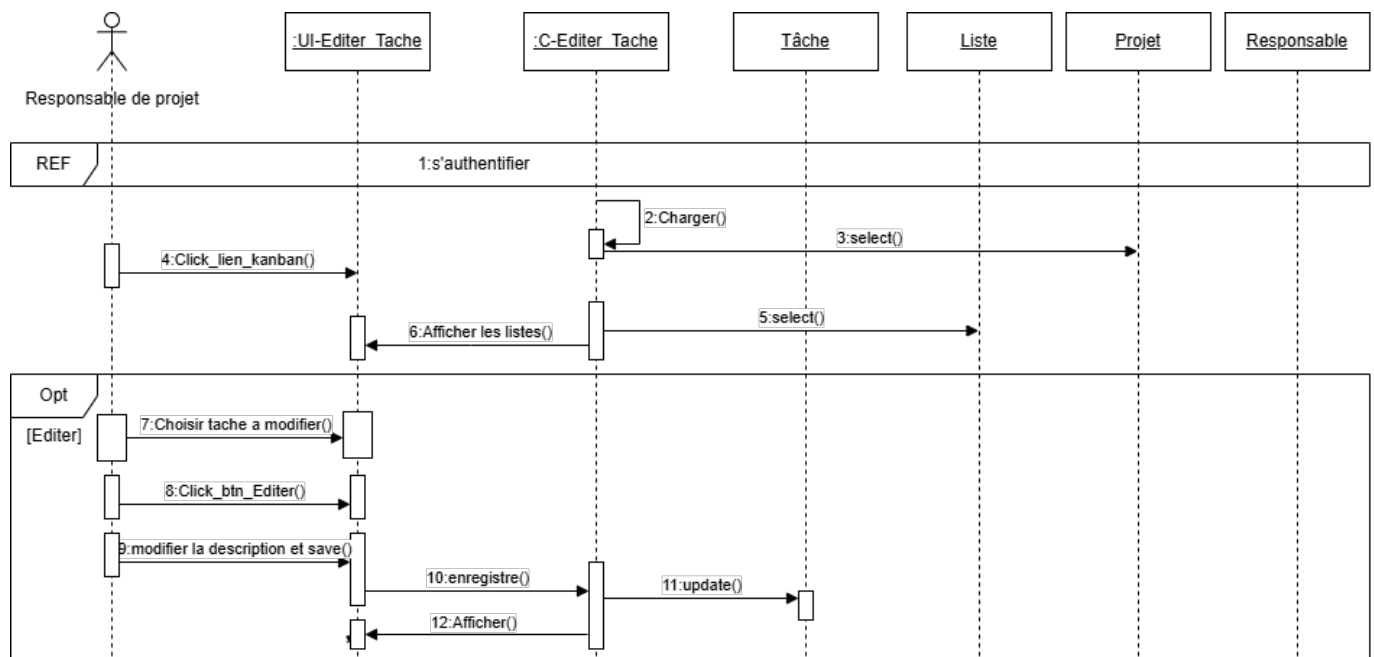


FIGURE 4.24 – Diagramme de Séquence « Editer une Tâche »

4.4.3 Conception de cas d'utilisation « Ajouter Membre » :

Diagramme de classe

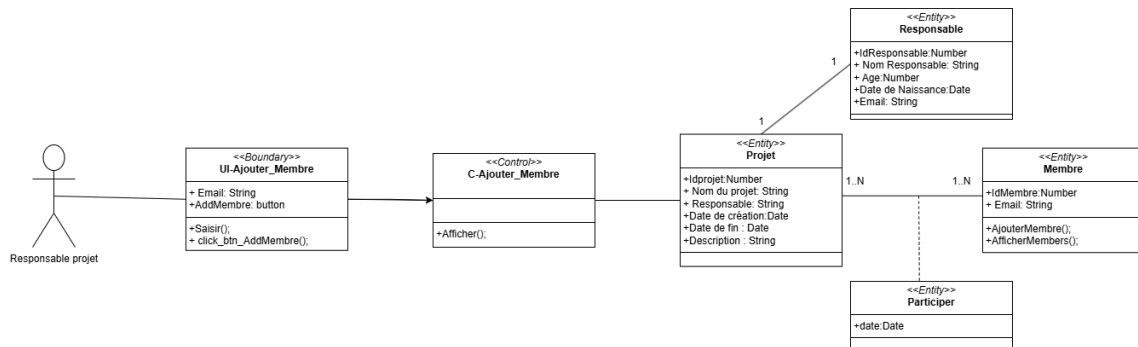


FIGURE 4.25 – Diagramme de classe de cas d'utilisation « Ajouter Membre »

Diagramme de séquence

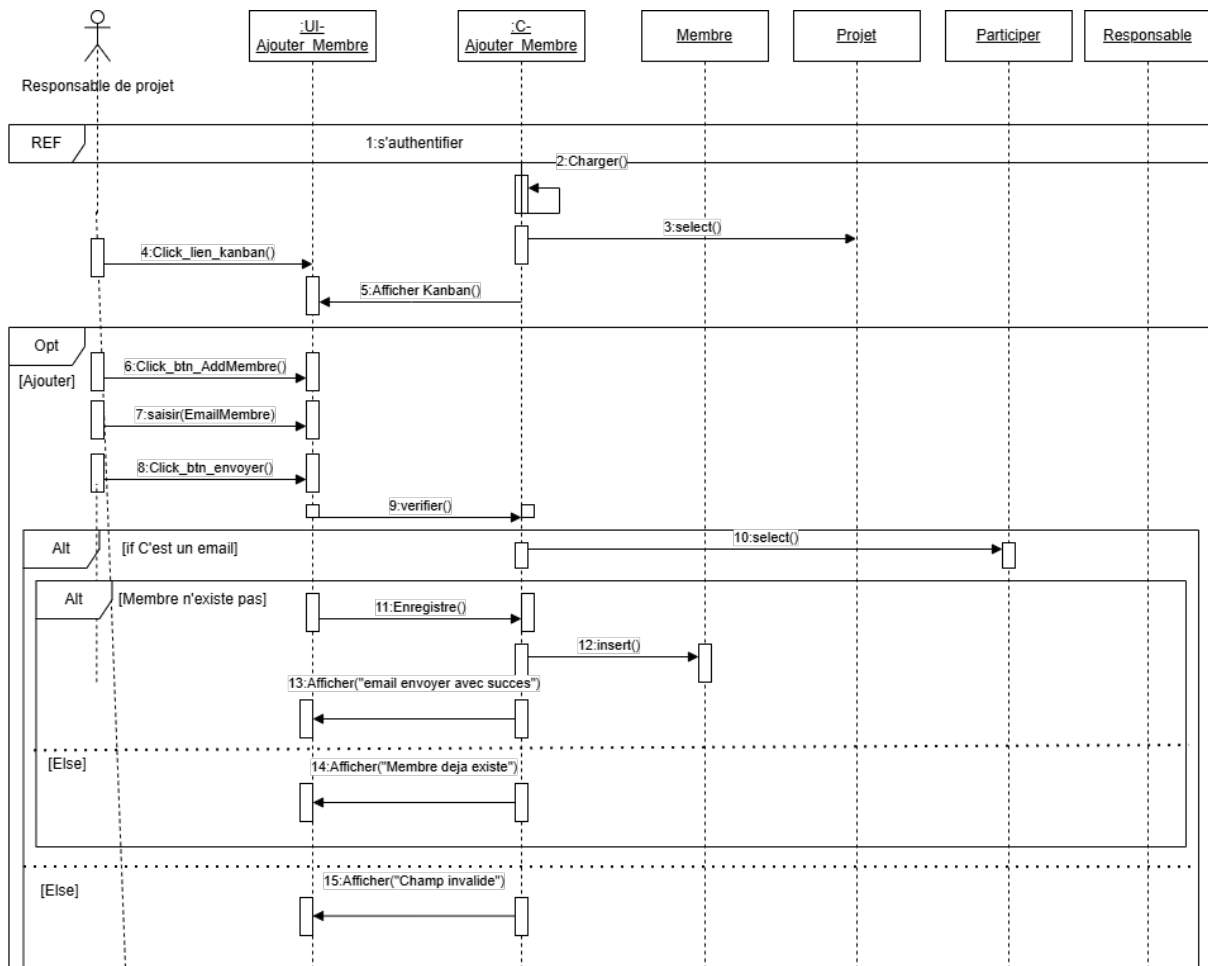


FIGURE 4.26 – Diagramme de Séquence « Ajouter Membre »

4.5 Diagramme de Classe Globale

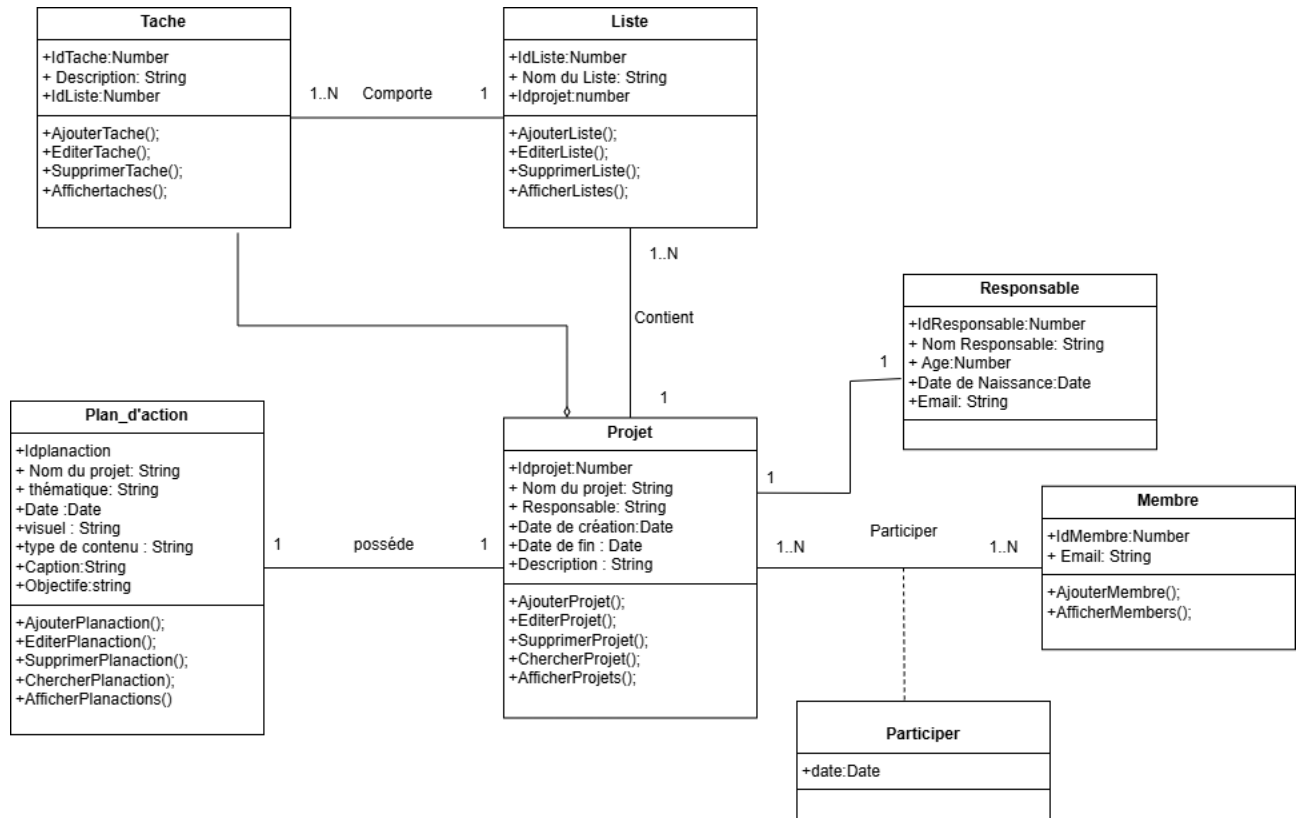


FIGURE 4.27 – Diagramme de Classe Globale

4.6 Réalisation du Sprint 1

4.6.1 Gestion des listes

-Cette interface permet à l'utilisateur consulter l'interface kanban d'un projet.

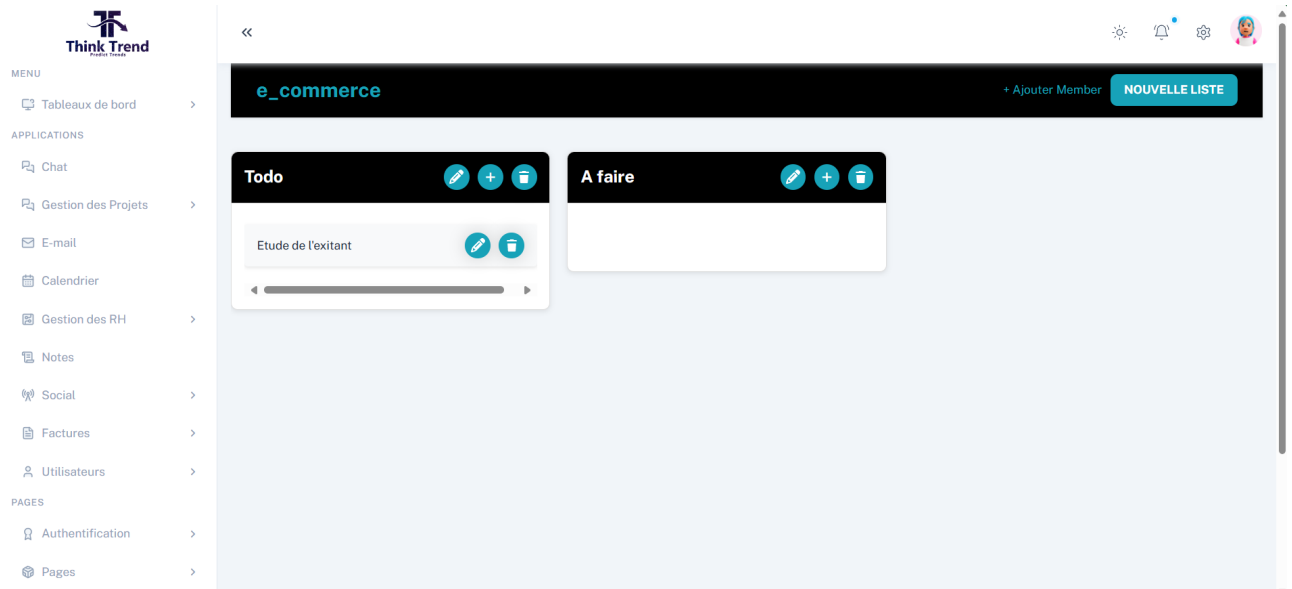


FIGURE 4.28 – Interface Kanban

-Cette interface permet à l'utilisateur de créer une liste dans un projet en cliquant sur New List.

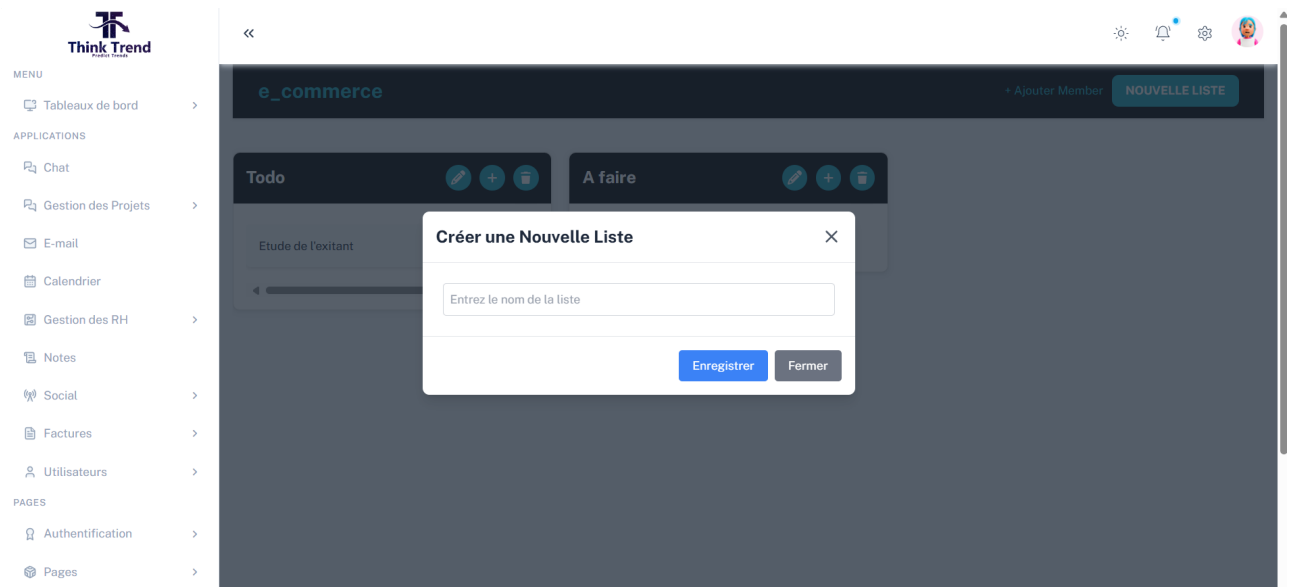


FIGURE 4.29 – Interface de création d'une liste

-Cette interface permet à l'utilisateur de modifier le nom d'une liste existante.

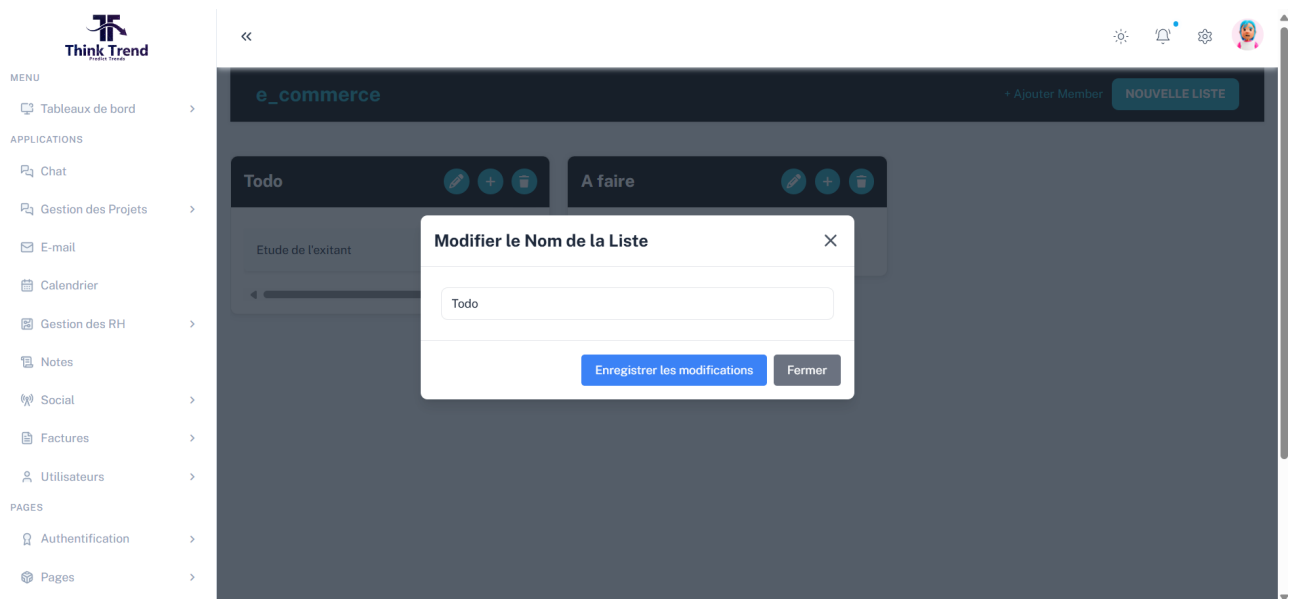


FIGURE 4.30 – Interface editer liste

-Cette interface permet à l'utilisateur de supprimer une liste.

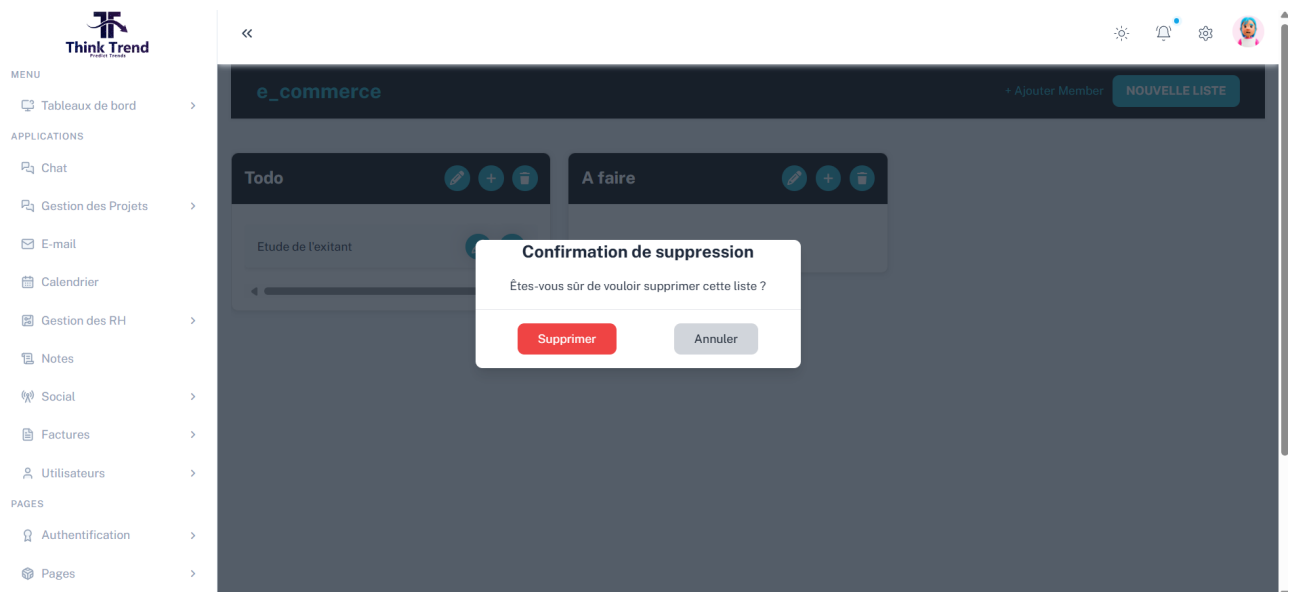


FIGURE 4.31 – Interface de supprimer une liste

4.6.2 Gestion des Tâches

-Cette interface permet à l'utilisateur de créer une tâche dans une liste en cliquant sur button +.

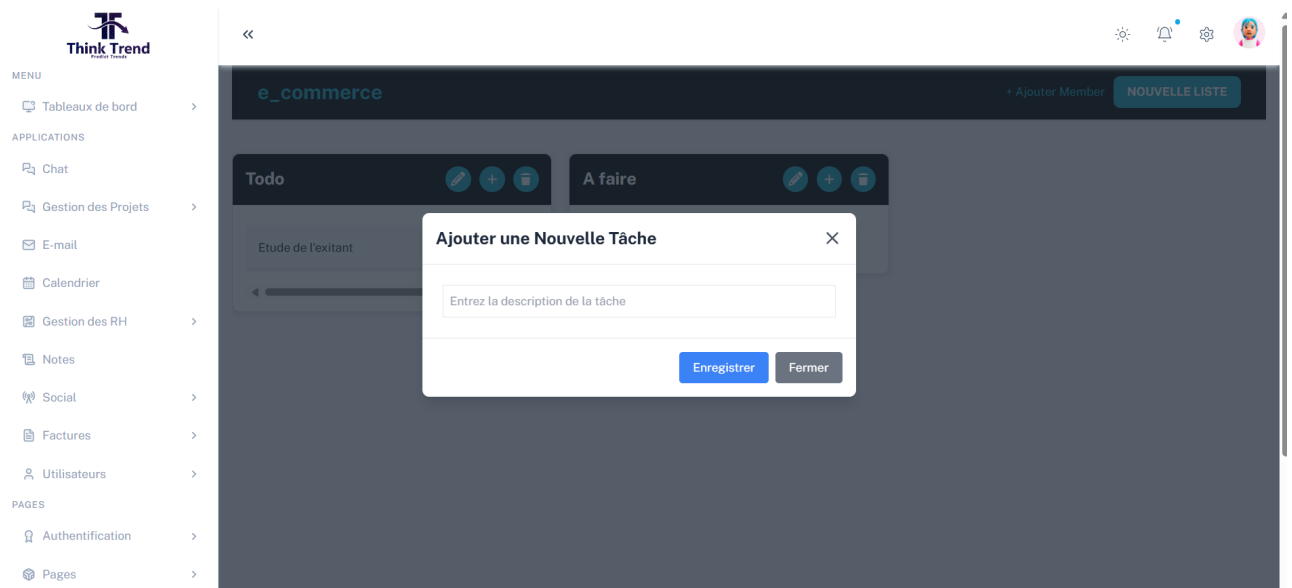


FIGURE 4.32 – Interface Ajouter Tâche

-Cette interface permet à l'utilisateur de modifier la description de la tâche.

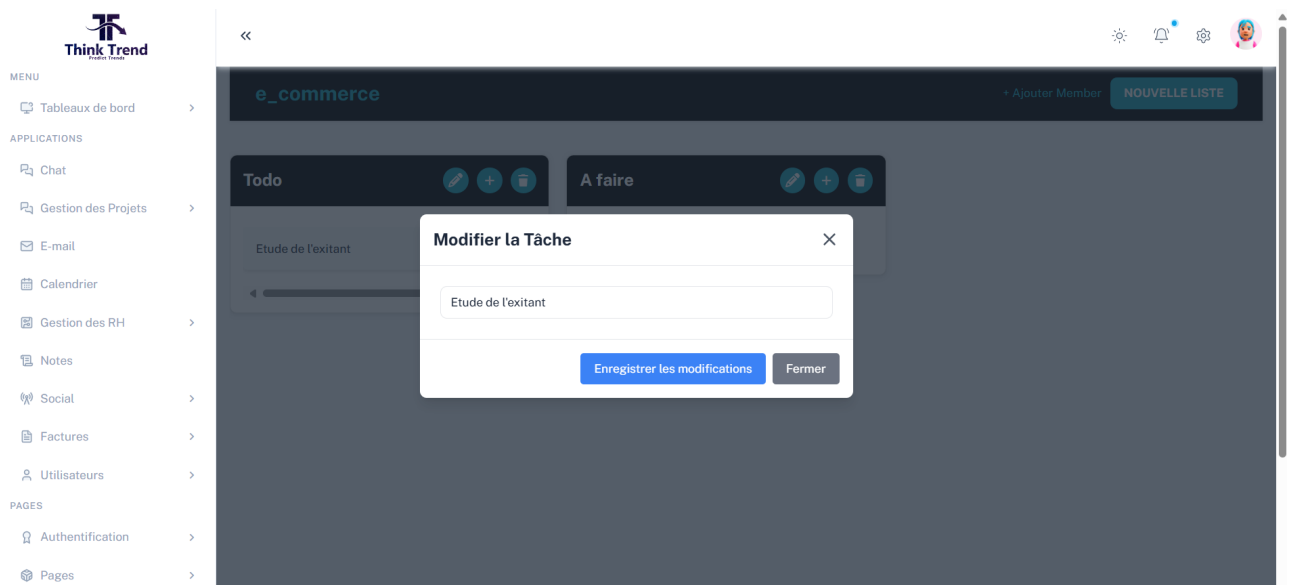


FIGURE 4.33 – Interface d'editer une tâche

-Cette interface permet à l'utilisateur de supprimer une tâche d'une liste.

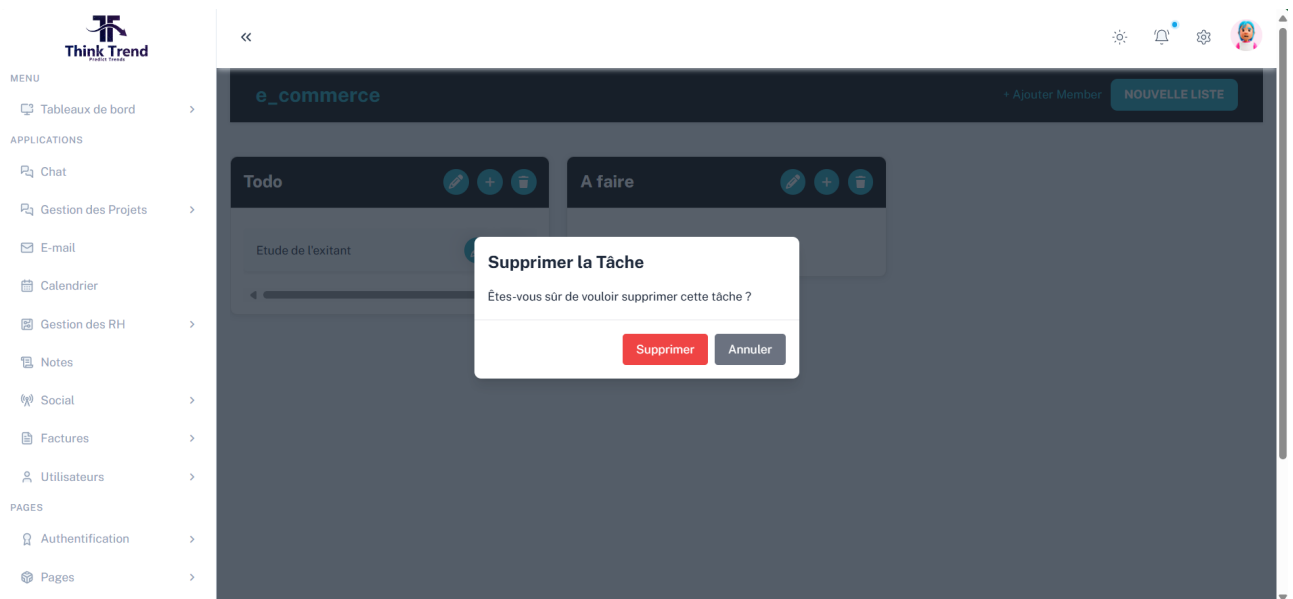


FIGURE 4.34 – Interface Supprimer Tâche

4.6.3 Gestion des Membres

-Cette interface permet à l'utilisateur d'ajouter des membres à un projet à partir de son email.

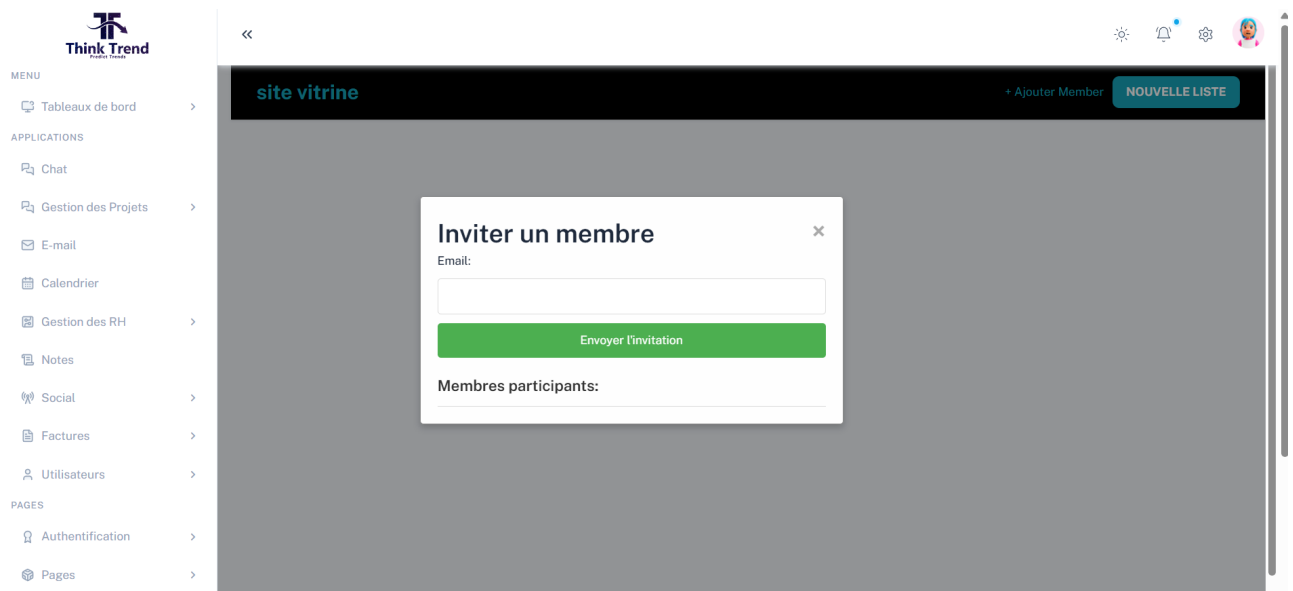


FIGURE 4.35 – Interface Ajouter membre

-Cette interface permet à ajouter un membre et consulter les membres existe déjà dans le projet.

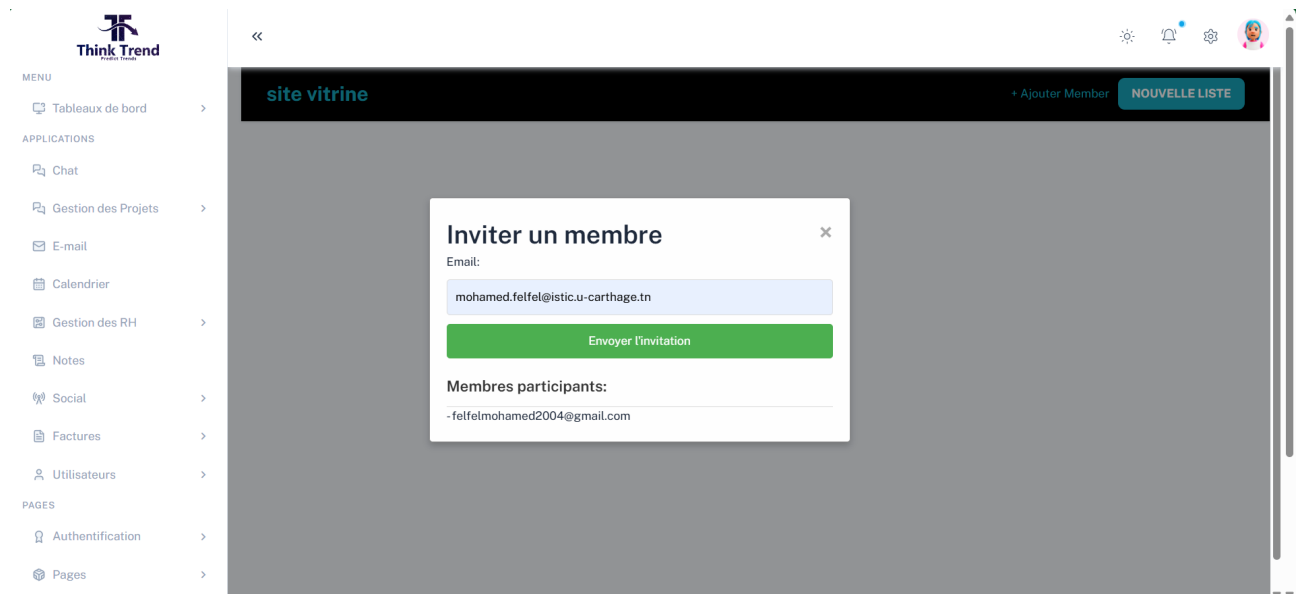


FIGURE 4.36 – Interface ajout et voir membre

-Cette interface permet à l'utilisateur de consulter l'état de l'envoi.

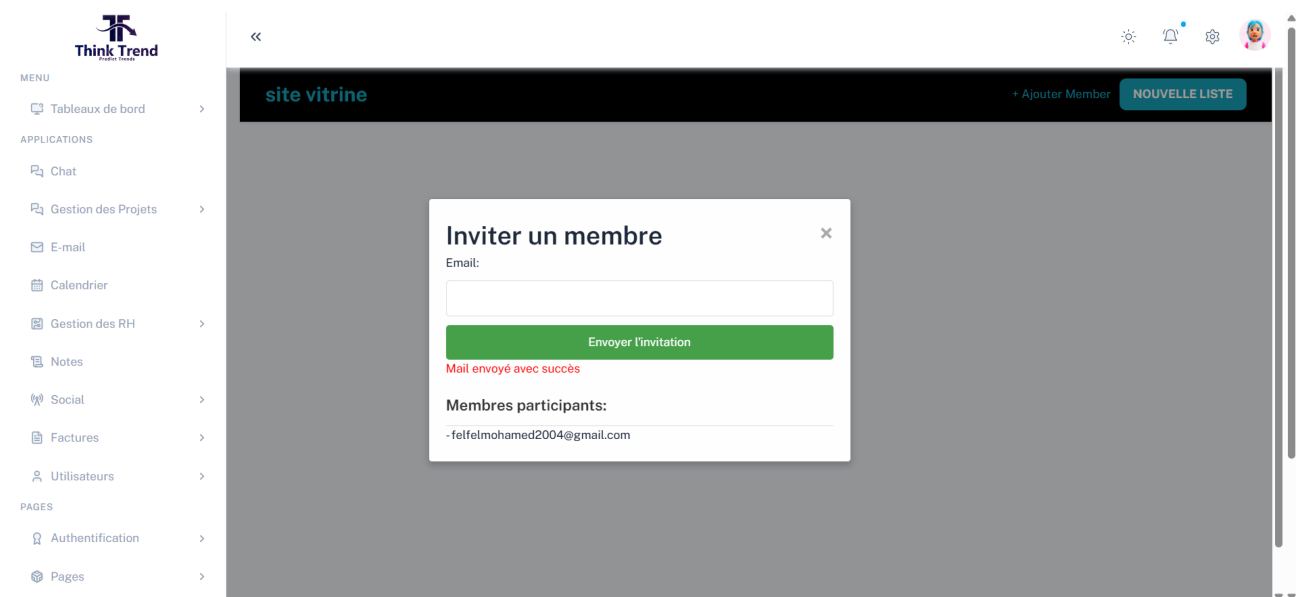


FIGURE 4.37 – Interface état d'envoi d'email avec succès

4.7 Conclusion

Au cours de ce dernier sprint, nous avons présenté le raffinement des cas d'utilisation. Nous avons également effectué une conception détaillée et nous nous sommes ensuite intéressés à la réalisation du module Suivi des projets.

Conclusion Générale

À la fin de ce rapport, nous pouvons affirmer que ce projet a constitué une occasion précieuse de confronter nos connaissances théoriques à un environnement pratique.

En effet, ce projet nous a permis d'assumer diverses responsabilités, ce qui a contribué à renforcer nos compétences tant pratiques que théoriques.

Notre stage a été bénéfique, car il nous a permis de perfectionner nos connaissances en programmation et en conception.

Tout au long de ce rapport, nous avons expliqué les différentes étapes par lesquelles nous avons élaboré une application web de gestion et de suivi des projets. La plateforme, que nous avons développée selon une architecture micro-services et en suivant la méthodologie SCRUM.

Les résultats obtenus montrent une amélioration significative de la gestion des projets, avec une meilleure visibilité sur l'avancement des tâches, une assignation claire des responsabilités et une réactivité accrue face aux attentes des clients. En outre, cette solution ouvre la voie à de futures évolutions, telles que l'intégration d'outils d'analyse de données pour un suivi encore plus précis des performances des projets.

Les perspectives d'amélioration et d'enrichissement de cette solution sont nombreuses, et nous sommes impatients de continuer à travailler sur ce projet afin d'en maximiser les bénéfices pour l'ensemble des parties prenantes.

Bibliographie

- [1] <https://spring.io/projects/spring-boot>.
- [10] Scrum. <https://www.scrum.org>. Cadre de gestion de projet agile.
- [11] Html. <https://developer.mozilla.org/en-US/docs/Web/HTML>. Langage de balisage pour la création de pages web.
- [12] Css. <https://developer.mozilla.org/en-US/docs/Web/CSS>. Langage de style pour la présentation des pages web.
- [13] Visual studio code. <https://code.visualstudio.com>. Éditeur de code open-source.
- [14] Spring tool suite (sts). <https://spring.io/tools>. Environnement de développement pour Spring.
- [15] Xampp. <https://www.apachefriends.org/index.html>. Environnement de développement PHP local.
- [16] Mysql. <https://www.mysql.com>. Système de gestion de base de données relationnelle.
- [17] Uml. <https://www.uml.org>. Langage de modélisation unifié.
- [18] Trello. <https://trello.com>. Outil de gestion de projets basé sur des tableaux et cartes.
- [19] draw.io. <https://app.diagrams.net>. Outil en ligne pour créer des diagrammes et des schémas techniques.
- [2] Angular. <https://angular.io>. Framework JavaScript pour le développement d'applications web.
- [3] Git. <https://git-scm.com>. Système de gestion de versions distribué.
- [4] Tailwind css. <https://tailwindcss.com>. Framework CSS utilitaire pour un design rapide.
- [5] Slack. <https://slack.com>.
- [7] Postman. <https://www.postman.com>. Plateforme de collaboration pour les API.
- [9] Google meet. <https://meet.google.com>.