



République Tunisiene
Ministère de l'Enseignement Supérieur
et de la Recherche Scientifique
Université de Tunis El Manar
Faculté des Sciences de Tunis
Département des Sciences de l'informatique



RAPPORT DE PROJET DE FIN D'ÉTUDES

Présenté en vue de l'obtention du :
Diplôme de licence en science de l'informatique
"Computer Science" : Génie logiciel et système d'information
Par :
Ferchichi Zakaria

Développement et conception d'un système de gestion de la qualité

Organisme d'accueil : **Business Management Group**

Soutenu le : 10/06/2024

Encadré par : **Mme Houda Alaya**

Supervisé par : **Mme Derouich Nour**

Président de Jury : **Mme Narjes Doggaz**

Rapporteur : **Mme Rim Dridi**

GLSI3 - S01066

Année Universitaire : 2023-2024

Dédicaces

Je dédie ce travail à :

À ma chère mère , la lanterne qui éclaire mon chemin et m'illumine d'amour, les mots me manquent pour exprimer toute la reconnaissance , la fierté et le profond amour que je te porte pour les sacrifices que tu as consentis à faire pour ma réussite .

À mon cher père , aucun mot ne peut exprimer le sentiment que j'éprouve à ton égard . C'est grâce à ton amour , ta patience , tes sacrifices et tes encouragements tout au long de mon parcours que j'ai pu être là où je suis maintenant.J'espère avoir été digne de ton affection et ta confiance

À mes chers frères, pour leur soutien durant toutes ces années. Je vous dédie ce travail en vous souhaitant un avenir radieux, plein de bonheur et de succès.

À ma chère copine Islem , ton amour, ton soutien et ta compréhension ont été mes piliers tout au long de ce parcours académique. Merci d'avoir été ma source d'inspiration et de m'avoir encouragé à atteindre mes objectifs

À mes amis , pour les moments précieux que nous avons partagé durant toute ses années , pour leur aide , leur patience et leur générosité

Zakaria

Remerciements

C'est un grand plaisir que je réserve cette page pour exprimer ma gratitude à tous ceux qui ont contribué à la réalisation de ce travail.

*Je tiens à exprimer ma profonde gratitude à mon encadrante académique **Mme Houda Alaya** pour son suivi et pour son énorme soutien qu'il n'a cessé de me prodiguer tout au long de la période du projet.*

*Je tiens également à exprimer ma gratitude envers mon encadrante professionnelle, **Mme Derouich Nour**, de la société BMG, pour ses précieux conseils, informations et motivation qu'il m'a fournis et qui m'ont permis de tenir jusqu'au bout pour réussir ce projet. Avec lui, j'ai pu partager sa expertise au quotidien.*

J'aimerai remercier tous mes enseignants d'avoir partagé avec nous leur passion pour l'enseignement. J'ai grandement apprécié leur appui, implication et expérience tout au long de ma formation.

J'adresse aussi mes vifs remerciements à tous les membres de la société BMG auprès desquels j'ai trouvé l'accueil chaleureux, l'aide et l'assistance dont j'ai besoin, un environnement dans lequel j'ai pu m'adapter facilement.

Je tiens finalement à remercier vivement les membres du jury qui ont fait l'honneur de juger ce travail et j'espère qu'il sera à la hauteur de la confiance qu'ils m'ont accordée.

Table des matières

Introduction générale	1
1 Contexte général	4
1 Introduction	5
2 Cadre du projet	5
3 Organisme d'accueil	5
3.1 Présentation générale	5
3.2 Domaine d'activité	6
3.3 Les valeurs de BMG	6
4 Analyse et Critique de l'existant	6
4.1 Étude de l'existant	6
4.2 Critique de l'existant	7
4.3 Solutions proposées	8
5 Choix de la Méthodologie	8
5.1 La méthode agile	9
5.2 La Méthodologie SCRUM	9
6 Conclusion	10
2 Analyse et spécification des besoins	11
1 Introduction	12
2 Spécification des besoins	12
2.1 Identification des acteurs	12
2.2 Besoins fonctionnels	12
2.3 Besoins non fonctionnels	14
3 Pilotage du projet avec SCRUM	14
3.1 Identification de l'équipe de SCRUM	15
3.2 Backlog du projet	15
3.3 Planification des sprints	18
4 Diagramme de cas d'utilisation global	18
5 Environnement de travail	19

5.1	Environnement Matériels	19
5.2	Environnement logiciel et développement	20
6	Architecture de l'application	24
6.1	Architecture physique	24
6.2	Architecture logique	24
7	Conclusion	25
3	Sprint 1 : Authentification , Gestion des ressources humaines et documents	26
1	Introduction	27
2	Backlog Sprint 1	27
3	Besoins fonctionnels	28
4	Mise en place de module Authentification	29
4.1	Diagramme de classes	29
4.2	Diagramme de cas d'utilisation	29
4.3	Diagrammes de séquence	32
5	Mise en place de module Ressources humaines	33
5.1	Diagramme de classes	34
5.2	Diagramme de cas d'utilisation	34
5.3	Diagrammes de séquence	37
6	Mise en place de module Documentation	39
6.1	Diagramme de cas d'utilisation	39
6.2	Diagramme d'activité	42
6.3	Diagrammes de classes	43
6.4	Diagrammes de séquence	44
7	Réalisation	45
8	Test	47
9	Conclusion	49
4	Sprint 2 : Gestion des clients , fournisseurs et indicateurs	50
1	Introduction	51
2	Backlog du sprint	51
3	Besoins fonctionnels	52

4	Mise en place du module client	52
4.1	Diagramme de classes	52
4.2	Diagramme de cas d'utilisation	53
4.3	Diagrammes de séquence	55
5	Mise en place de module Fournisseur	56
5.1	Diagramme de classes	56
5.2	Diagramme de cas d'utilisation	57
5.3	Diagrammes de séquence	58
6	Mise en place de module Indicateur	59
6.1	Diagramme de classes	59
6.2	Diagramme de cas d'utilisation	60
6.3	Diagrammes de séquence	62
7	Réalisation	62
8	Test	64
9	conclusion	65
5	Sprint 3 : Gestion des audits , des actions et la planification des réunions.	66
1	Introduction	67
2	Backlog du sprint	67
3	Besoins fonctionnels	67
4	Mise en place du module Audit	68
4.1	Diagramme de classe	68
4.2	Diagramme de cas d'utilisation	69
4.3	Description textuelle des cas d'utilisation	69
4.4	Diagrammes de séquence	70
5	Mise en place du module réunion	71
5.1	Diagramme de classes	71
5.2	Diagramme de cas d'utilisation	72
5.3	Description textuelle des cas d'utilisation	73
5.4	Diagrammes de séquence	74
6	Mise en place de module actions	75
6.1	Diagramme de classes	75

6.2	Diagramme de cas d'utilisation	75
6.3	Description textuelle des cas d'utilisation	76
6.4	Diagrammes de séquence	78
7	Réalisation	79
8	Test	79
9	conclusion	81
6	Sprint 4 : Gestion des produits non conforme , conformité réglementaire et des risques.	82
1	Introduction	83
2	Backlog du sprint	83
3	Besoins fonctionnels	83
4	Mise en place du module Produit non conforme	84
4.1	Diagramme de classes	84
4.2	Diagramme de cas d'utilisation	84
4.3	Description textuelle des cas d'utilisation	85
4.4	Diagrammes de séquence	86
5	Mise en place du module Conformité réglementaire	87
5.1	Diagramme de classe	87
5.2	Diagramme de cas d'utilisation	88
5.3	Description textuelle des cas d'utilisation	89
5.4	Diagrammes de séquence	89
6	Mise en place de module risque	90
6.1	Diagramme de classes	90
6.2	Diagramme de cas d'utilisation	91
6.3	Description textuelle des cas d'utilisation	92
6.4	Diagrammes de séquence	92
7	Réalisation	93
8	Test	94
9	conclusion	95
Conclusion générale		96

Table des figures

1.1	LOGO BMG	5
1.2	Méthode Agile	9
1.3	Processus de Scrum	10
2.1	Planification sprints	18
2.2	Diagramme de cas d'utilisation global	19
2.3	Logo Visual Studio Code	20
2.4	Logo PostgeSQL	20
2.5	Logo Django	21
2.6	Logo Rest framework	21
2.7	Logo JWT	21
2.8	Logo React	22
2.9	Logo Postman	22
2.10	Logo Draw.io	22
2.11	Latex	23
2.12	GitHub	23
2.13	Gmail	23
2.14	L'architecture 3-tiers	24
2.15	L'architecture MVC	25
3.1	Diagramme de classes d'authentification	29
3.2	Diagramme de cas d'utilisation d'authentification	30
3.3	Diagramme de séquence d'authentification	33
3.4	Diagramme de séquence de "Gérer profile"	33
3.5	Diagramme de classes de module RH	34
3.6	Diagramme de cas d'utilisation de module RH	34
3.7	Diagramme de séquence de Ajouter fiche Employé	37
3.8	Diagramme de séquence de Modifier fiche Employé	38
3.9	Diagramme de séquence de Supprimer fiche Employé	38
3.10	Diagramme de séquence "Ajouter employé"	38

Table des figures

3.11 Diagramme de séquence "Évaluer compétences"	39
3.12 Diagramme de cas d'utilisation de Documentation	40
3.13 Diagramme d'activité de création d'un document	43
3.14 Diagramme de classe de Documentation	44
3.15 Diagramme de séquence de Créer document	45
3.16 Interface de page d'accueil	45
3.17 Interface Profile	46
3.18 Interface Créer compte	46
3.19 Interface Consulter fiche employé	47
3.20 Interface 'valider/refuser demande'	47
3.21 Test méthode créer compte	48
3.22 Test méthode "s'authentifier"	48
3.23 Test méthode "ajouter employé"	49
3.24 Test méthode "valider une demande"	49
4.1 Diagramme de classes de Client	53
4.2 Diagramme de cas d'utilisation de Client	53
4.3 Diagramme de séquence de "ajouter client"	55
4.4 Diagramme de séquence de "Créer réclamation client"	56
4.5 Diagramme de séquence de "suggérer client"	56
4.6 Diagramme de classes du module Fournisseur	57
4.7 Diagramme de cas d'utilisation de Fournisseur	57
4.8 Diagramme de séquence de "Créer réclamation fournisseur"	59
4.9 Diagramme de séquence de "Évaluer fournisseur"	59
4.10 Diagramme de classe d'indicateur	60
4.11 Diagramme de cas d'utilisation d'indicateur	60
4.12 Diagramme de séquence de "suivre indicateur"	62
4.13 Interface ajouter réclamation client	63
4.14 Interface enquête client	63
4.15 Interface évaluer fournisseur	64
4.16 Test méthode "ajouter réclamation client"	64
4.17 Test méthode "Évaluer fournisseur"	65

4.18 Test méthode "Suivre indicateur"	65
5.1 Diagramme de classes d'audit	68
5.2 Diagramme de classe d'audit	69
5.3 Diagramme de séquence "ajouter fiche audit"	70
5.4 Diagramme de séquence "valider audit"	71
5.5 Diagramme de classe de réunion	72
5.6 Diagramme de cas d'utilisation de réunion	73
5.7 Diagramme de séquence de Planifier réunion	74
5.8 Diagramme de séquence de prendre décision	74
5.9 Diagramme de classes d'action	75
5.10 Diagramme de cas d'utilisation d'action	76
5.11 Diagramme de séquence de Modifier action	78
5.12 Diagramme de séquence de Clôturer action	78
5.13 Interface consulter réunion	79
5.14 Interface ajouter action	79
5.15 Test méthode "Créer Audit"	80
5.16 Test méthode "Créer action"	80
5.17 Test méthode "Planifier réunion"	81
6.1 Diagramme de classes Produit non conforme	84
6.2 Diagramme de cas d'utilisation de Produit non conforme	85
6.3 Diagramme de séquence "Créer fiche non conformité"	87
6.4 Diagramme de séquence "Traiter non conformité"	87
6.5 Diagramme de classe de Conformité réglementaire	88
6.6 Diagramme de cas d'utilisation de Conformité réglementaire	88
6.7 Diagramme de séquence de Ajouter fiche conformité	89
6.8 Diagramme de séquence de Modifier fiche conformité	90
6.9 Diagramme de classes de risque	91
6.10 Diagramme de cas d'utilisation de risque	92
6.11 Diagramme de séquence de Traiter risque	92
6.12 Interface traiter risque	93
6.13 Interface Modifier conformité	93

6.14 Interface conformité	94
6.15 Test méthode "Créer non conformité d'un produit"	94
6.16 Test méthode "Créer conformité réglementaire"	95
6.17 Test méthode "Traiter risque"	95

Liste des tableaux

2.1	Equipe Scrum	15
2.2	Product Backlog	17
2.3	Environnement matériel	20
3.1	Backlog Sprint 1	28
3.2	Description textuelle du cas d'utilisation "Grouper les utilisateurs"	30
3.3	Description textuelle du cas d'utilisation "Accorder permissions"	31
3.4	Description textuelle du cas d'utilisation "Créer compte"	31
3.5	Description textuelle du cas d'utilisation "S'authentifier "	32
3.6	Description textuelle du cas d'utilisation "Gérer profile"	32
3.7	Description textuelle du cas d'utilisation "ajouter employé"	35
3.8	Description textuelle du cas d'utilisation "Créer fiches employé"	35
3.9	Description textuelle du cas d'utilisation "Consulter demande"	36
3.10	Description textuelle du cas d'utilisation "Évaluer compétences"	36
3.11	Description textuelle du cas d'utilisation "Évaluer en froid "	36
3.12	Description textuelle du cas d'utilisation "Valider ou refuser demande"	37
3.13	Description textuelle du cas d'utilisation "Créer ou modifier document"	41
3.14	Description textuelle du cas d'utilisation "Accepter ou refuser demande"	41
3.15	Description textuelle du cas d'utilisation "Archiver version antérieur"	41
3.16	Description textuelle du cas d'utilisation "Commenter document"	42
4.1	Backlog Sprint 2	51
4.2	Description textuelle du cas d'utilisation "Ajouter client"	54
4.3	Description textuelle du cas d'utilisation "Ajouter réclamation client"	54
4.4	Description textuelle du cas d'utilisation "Créer enquête client"	54
4.5	Description textuelle du cas d'utilisation "Suivre non conformité"	55
4.6	Description textuelle du cas d'utilisation "Suggérer client"	55
4.7	Description textuelle du cas d'utilisation "Ajouter fournisseur"	58
4.8	Description textuelle du cas d'utilisation "Créer réclamation"	58
4.9	Description textuelle du cas d'utilisation "Évaluer fournisseur"	58

4.10 Description textuelle du cas d'utilisation "Ajouter fournisseur"	61
4.11 Description textuelle du cas d'utilisation "Modifier indicateur"	61
4.12 Description textuelle du cas d'utilisation "Modifier indicateur"	61
4.13 Description textuelle du cas d'utilisation "Suivre indicateur"	62
 5.1 Backlog Sprint 3	67
5.2 Description textuelle du cas d'utilisation "Ajouter fiche audit"	69
5.3 Description textuelle du cas d'utilisation "Ajouter plan audit"	70
5.4 Description textuelle du cas d'utilisation "Valider audit"	70
5.5 Description textuelle du cas d'utilisation "Planifier réunion"	73
5.6 Description textuelle du cas d'utilisation "Prendre décision"	74
5.7 Description textuelle du cas d'utilisation "Visualiser réunion"	74
5.8 Description textuelle du cas d'utilisation "Ajouter action"	76
5.9 Description textuelle du cas d'utilisation "Modifier action"	77
5.10 Description textuelle du cas d'utilisation "ajouter sous action"	77
5.11 Description textuelle du cas d'utilisation "Clôturer action"	78
 6.1 Backlog Sprint 4	83
6.2 Description textuelle du cas d'utilisation "Créer fiche non conformité"	85
6.3 Description textuelle du cas d'utilisation "Remplir fiche de détection"	86
6.4 Description textuelle du cas d'utilisation "Traiter non conformité"	86
6.5 Description textuelle du cas d'utilisation "Clôturer non conformité"	86
6.6 Description textuelle du cas d'utilisation "Ajouter fiche conformité"	89
6.7 Description textuelle du cas d'utilisation "Modifier fiche conformité"	89
6.8 Description textuelle du cas d'utilisation "Traiter risque"	92

Liste des abréviations

- **API** = Application Programming Interface
- **BMG** = Business Management Group
- **ERP** = Enterprise Resource Planning
- **JSON** = JavaScript Objet Notation
- **MVC** = Modèle Vue Contrôleur
- **RH** = Ressources Humaines
- **UML** = Unified Modeling Language

Introduction générale

De nos jours, de nombreuses entreprises, actives dans divers secteurs, envisagent d'investir des sommes importantes dans l'adoption de nouvelles technologies logicielles intégrant des normes de Système de Management de la Qualité (SMQ) telles que l'ISO 9001. Leur objectif est d'améliorer la qualité de leurs services, d'augmenter leur agilité et leur flexibilité, de réduire leurs dépenses, d'augmenter leur capacité de production et de relever les défis du marché. Toutefois, la diversification et le développement des activités au sein de ces entreprises ont provoqué un besoin croissant d'intégrer plusieurs technologies. La gestion efficace de ces diverses activités devient ainsi de plus en plus complexe et difficile .

Pour faire face à ces défis, les entreprises nécessitent l'utilisation des outils optimisés qui puissent répondre à l'ensemble de leurs besoins opérationnels en offrant des fonctionnalités riches et pertinentes. Parmi ces solutions, les systèmes intégrés de gestion, tels que les ERP (Entreprise Ressource Planning), occupent une place centrale. Ces systèmes permettent de gérer et d'analyser efficacement les données, d'améliorer les processus de gestion et d'automatiser les différentes activités et tâches de l'entreprise. Selon Blain, "l'adoption des ERP a considérablement renforcé la réactivité, la productivité et la compétitivité des entreprises".

Notre projet de conception et de réalisation d'un ERP pour la société BMG va bien au-delà de la simple automatisation des processus de gestion. En fait, il s'agit d'un Système de Management de la Qualité (SMQ) conçu selon les normes rigoureuses de l'ISO 9001. En intégrant les principes de qualité et de conformité au cœur même de notre solution ERP. Cet ERP doit permettre d'automatiser les différents processus de gestion, notamment la gestion des ressources humaines, la gestion des audits, la gestion des clients, la gestion des fournisseurs, la gestion des indicateurs, la gestion des risques, la gestion de la documentation, la gestion des non-conformités et la conformité réglementaire

Pour mener à bien ce projet, nous avons utilisé la méthode Scrum, une méthodologie agile qui favorise la collaboration, la flexibilité et l'amélioration continue tout au long du cycle de développement. De plus, nous avons adopté l'architecture MVC (Model-View-Controller) pour la structure de notre application ERP. Cette architecture nous a aidés à séparer les préoccupations

et à assurer une meilleure organisation du code, facilitant ainsi la maintenance et l'évolutivité de notre solution.

En ce qui concerne les technologies utilisées, nous avons opté pour PostgreSQL comme système de gestion de base de données en raison de sa fiabilité, de ses performances et de sa capacité à gérer des volumes importants de données. Pour le développement du backend, notre choix s'est porté sur Django, un framework web Python réputé pour sa solidité et ses fonctionnalités avancées. Pour le frontend, nous avons utilisé React, une bibliothèque JavaScript moderne et puissante, permettant de créer une interface utilisateur réactive et intuitive pour notre ERP.

Le premier chapitre, intitulé "Contexte Général", est consacré à la présentation de l'organisme d'accueil ainsi que de ses services et de ses valeurs. Ensuite, l'étude de l'existant, critique de l'existant et la solution proposée sont abordées. Enfin, la méthodologie choisie pour le développement est détaillée.

Le deuxième chapitre, intitulé "Analyse et Spécification des besoins", porte sur la planification et la spécification des besoins de notre projet, la description de l'architecture de l'application, l'environnement de travail ainsi que les différents choix technologiques effectués.

Ensuite, dans le troisième chapitre intitulé "Authentification, Gestion des ressources humaines et documentation", le premier sprint est abordé, étant la première partie de l'application du cadre méthodologique SCRUM.

Ensuite, dans le quatrième chapitre intitulé "Gestion des clients, fournisseurs et indicateurs", le deuxième sprint est abordé, étant la deuxième partie de l'application du cadre méthodologique SCRUM.

Ensuite, dans le cinquième chapitre intitulé "Gestion des audits, des actions et la planification des réunions", le troisième sprint est abordé, étant la troisième partie de l'application du cadre méthodologique SCRUM.

Enfin, dans le sixième chapitre intitulé "Gestion des produits non conforme,conformité réglementaire et des risques", le quatrième sprint est abordé, étant la dernière partie de l'application du cadre méthodologique SCRUM.

Contexte général

Plan

1	Introduction	5
2	Cadre du projet	5
3	Organisme d'accueil	5
4	Analyse et Critique de l'existant	6
5	Choix de la Méthodologie	8
6	Conclusion	10

1 Introduction

Dans ce premier chapitre, nous allons aborder le contexte général de notre projet. Tout d'abord, nous allons présenter l'entreprise qui nous a permis de réaliser notre projet de fin d'études. Ensuite, nous allons décrire le cadre général de notre projet, qui comprend l'étude de l'existant ainsi qu'une présentation de notre solution, et nous finissons par la Méthodologie adoptée .

2 Cadre du projet

Ce projet s'inscrit dans le cadre de la préparation de projet de fin d'études pour l'obtention d'une licence en "Science informatique" à la Faculté des sciences de Tunis Il a été réalisé au sein de l'entreprise "Business Management Group" sur une durée de 4 mois

3 Organisme d'accueil

3.1 Présentation générale

Business Management Group est un cabinet de Conseil et de Formation dédié à l'amélioration de l'excellence opérationnelle des organisations et des managers. Animé par un esprit visionnaire et innovateur, Business Management Group n'a pas cessé de se développer afin de répondre aux exigences de ses clients soucieux de voir loin pour mieux décider aujourd'hui. Spécialiste dans les domaines de la Formation Professionnelle et du Conseil, BMG met son expertise à la disposition de ses clients afin de renforcer leurs compétences et d'accroître leur performance dans un environnement économique et social de plus en plus exigeant.



Figure 1.1 : LOGO BMG

3.2 Domaine d'activité

- Développement mobile
- Développement web
- Intégration ERP
- Business intelligence
- SEO et optimisation
- Analyse de données
- Commerce électronique et marchés

3.3 Les valeurs de BMG

Les valeurs du BMG sont partagées par tous ses membres :

- **Éthique** : Confidentialité et intégrité dans toutes nos transactions.
- **Collaboration** : Valorisation de travail d'équipe et la participation de toutes les parties prenantes.
- **Agilité** : Des résultats rapides et efficaces appuyés sur des processus solides
- **Expertise** : Nous valorisons l'expertise et le professionnalisme et nous nous entourons de personnes partageant les mêmes idées et exigences.
- **Innovation** : Nous sommes toujours à l'affût des dernières tendances du marché et pensons Out of The Box.

4 Analyse et Critique de l'existant

4.1 Étude de l'existant

Actuellement, les processus de gestion de la qualité à BMG s'appuient sur des méthodes traditionnelles et dispersées. Les documents sont créés et gérés via des logiciels de traitement de texte, ce qui peut entraîner des problèmes de suivi des versions. Les informations sur les employés sont stockées dans des dossiers physiques, et la gestion des formations est informelle. Les réunions sont planifiées par e-mail et le suivi des résultats est manuel.

Les informations fournisseur sont gérées dans des feuilles de calcul partagées, entraînant des difficultés de gestion des réclamations. Une gestion réactive des non-conformités et aucune veille réglementaire formelle n'est en place. Un système ERP pourrait offrir une solution intégrée pour améliorer l'efficacité opérationnelle, la traçabilité des données et la conformité réglementaire.

4.2 Critique de l'existant

Il existe des ERP tels que Odoo , Qali Pro chaque système a ses propres forces et faiblesses, et celles-ci peuvent varier en fonction des besoins spécifiques de chaque entreprise. Voici quelques points faibles potentiels que nous pourrions relever pour ces systèmes :

- **Complexité de mise en oeuvre** : Les ERP comme Odoo ou Kali Pro peuvent nécessiter une configuration complexe et une mise en oeuvre longue et coûteuse, surtout pour les petites et moyennes entreprises qui ont des ressources limitées en termes de temps et de budget.
- **Personnalisation limitée** : Bien que ces ERP offrent une certaine flexibilité et des options de personnalisation, ils peuvent parfois être limités dans leur capacité à répondre aux besoins spécifiques d'une entreprise sans recourir à des développements personnalisés coûteux.
- **Support technique** : Les problèmes techniques ou les besoins de support peuvent survenir pendant l'utilisation de ces ERP, et il est parfois difficile d'obtenir une assistance rapide et efficace de la part des fournisseurs, en particulier pour les entreprises qui utilisent des versions open-source.
- **Intégration avec d'autres systèmes** : L'intégration avec des systèmes externes ou des applications tierces peut être complexe et nécessiter des efforts importants en termes de développement et de configuration, ce qui peut entraîner des retards ou des problèmes d'interopérabilité.
- **Mises à jour et évolutivité** : Les mises à jour fréquentes des ERP peuvent poser des défis en termes de compatibilité avec les personnalisations existantes et nécessiter des efforts importants pour assurer une transition en douceur vers les nouvelles versions. De plus, la capacité des ERP à évoluer avec les besoins changeants de l'entreprise peut être limitée dans certains cas.

4.3 Solutions proposées

Au sein de BMG, notre solution consiste à développer un ERP de gestion de qualité. Ce système sera composé des modules suivants :

- **Document** : La gestion des documents pour un processus de validation de documents.
- **Ressources Humaines** : La gestion des ressources humaines permettra de suivre les compétences et les formations des employés, ce qui contribuera à assurer leur adéquation avec les besoins de l'entreprise en matière de qualité.
- **Audit** : La planification, la réalisation et le suivi des audits sont des éléments clés pour évaluer et améliorer la conformité aux normes de qualité.
- **Fournisseur** : La gestion des fournisseurs et des réclamations associées contribuera à garantir la qualité des matières premières et des services externes.
- **Indicateur** : La surveillance régulière des indicateurs de qualité permettra d'identifier les tendances et d'apporter des améliorations continues aux processus et aux produits.
- **Produit Non Conforme** : La gestion efficace des non-conformités et des réclamations contribuera à résoudre rapidement les problèmes et à prévenir leur récurrence.
- **Évaluation de la conformité réglementaire** : Assurer la conformité aux exigences réglementaires est crucial pour éviter les sanctions et les risques juridiques.
- **Client** : La gestion des clients et de leurs retours permettra d'améliorer la satisfaction client et de fidéliser la clientèle.
- **Risque** : L'identification et le traitement des risques potentiels aideront à anticiper et à atténuer les menaces pour l'entreprise.

5 Choix de la Méthodologie

L'achèvement du projet dans sa date de livraison est le problème principal de chaque équipe de développement d'un logiciel. L'un des problèmes les plus souvent rencontrés lors de la production d'un logiciel est la médiocre spécification, la faible gestion du temps face à l'utilisation d'une technologie apparaissant et le changement soudain des besoins. Afin d'éviter ces problèmes critiques, nous suivons la méthodologie agile pour la gestion du projet.

5.1 La méthode agile

La méthodologie agile est bien différente des méthodes traditionnelles telle que le cycle en V ou Waterfall qui prévoit les planifications totales du projet avant même le développement. Elle se veut plus souple et adaptée, et préconise plutôt la fixation d'objectif à court terme. C'est un mode de gestion qui place le client au cœur de l'action, cela permettra à l'équipe de développement d'avoir des feedbacks réguliers afin d'appliquer directement les changements nécessaires.

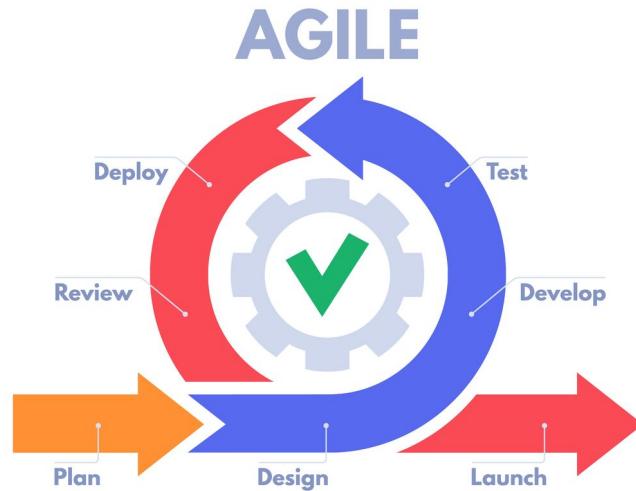


Figure 1.2 : Méthode Agile

5.2 La Méthodologie SCRUM

La méthode scrum ou plus exactement le cadre méthodologique scrum est de loin la méthode Agile la plus utilisée dans le monde. Elle bénéficie aujourd’hui de nombreux retours d’expérience. Les formations, blogs, conférences, communautés, outils et ouvrage à son sujet ne manquent pas [1].

La figure 1.3 présente le processus de scrum.

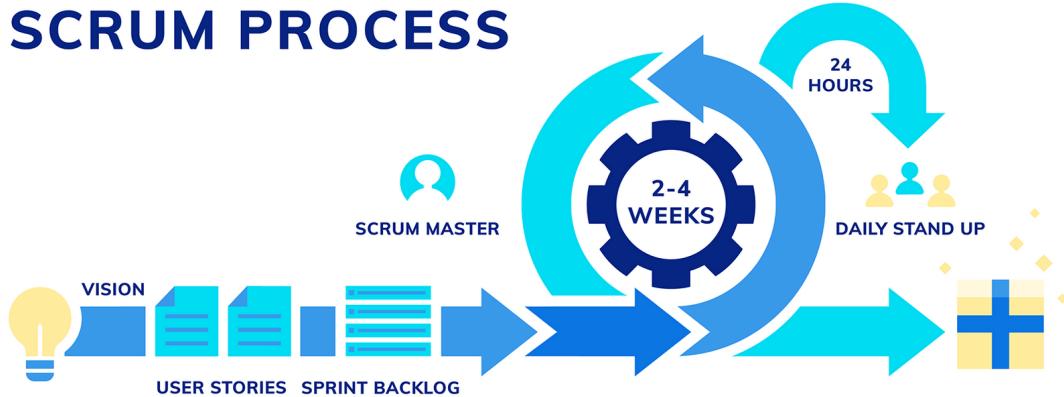


Figure 1.3 : Processus de Scrum

La méthodologie SCRUM est donc un cadre de travail permettant la gestion du développement d'application complexe. Le principe de cette méthode est de développer un logiciel de manière incrémentale en maintenant une liste totalement transparente des demandes dévolution ou de corrections à implémenter "backlog" . Avec des livraisons très fréquentes, toutes les quatre semaines en général. Pour cela, la méthode s'appuie sur des développements itératifs "sprints" à un rythme comptant d'une durée de 2 à 4 semaine. Le sprint englobe les activités d'analyse, de conception, de test et de développement. Au cours de ce dernier l'équipe de développement soulève des question métiers.

6 Conclusion

Dans ce chapitre, nous avons présenté la société BMG. Puis nous avons effectué une étude de l'existant pour sensibiliser l'importance du projet. Ensuite nous avons détaillé la problématique, mentionné les difficultés et résoudre des solutions en utilisant les technologies et les compétences nécessaires au développement.

Le deuxième chapitre nous permettra de commencer la phase d'initialisation du projet.

Analyse et spécification des besoins

Plan

1	Introduction	12
2	Spécification des besoins	12
3	Pilotage du projet avec SCRUM	14
4	Diagramme de cas d'utilisation global	18
5	Environnement de travail	19
6	Architecture de l'application	24
7	Conclusion	25

1 Introduction

Après avoir présenté le cadre de notre projet dans le premier chapitre, nous allons dans ce chapitre identifier les acteurs ainsi que les besoins fonctionnels et non fonctionnels. Nous présenterons ensuite les diagrammes nécessaires pour la spécification et l'analyse des besoins. Enfin, nous détaillerons les technologies utilisées ainsi que l'architecture de l'application.

2 Spécification des besoins

Cette partie va servir à poser les bases du recueil des besoins du système à réaliser, nous nous intéressons à identifier les besoins des utilisateurs dans notre application à travers les spécifications fonctionnelles et non fonctionnelles pour aboutir à une application de bonne qualité qui répond aux besoins des utilisateurs logistiques.

2.1 Identification des acteurs

Un acteur est l'abstraction d'un rôle joué par une entité externe, un processus (personne physique) ou tout un composant qui interagit avec un système. Dans notre outil informatique, il existe 3 acteurs :

- Administrateur : C'est la personne possédant le privilège de plus haut niveau
- Utilisateur : Cela fait plusieurs types d'utilisateurs, chacun ayant des responsabilités spécifiques en fonction du module auquel il est associé.
- Guest : C'est la personne qui peut rejoindre une réunion ou une formation sans être authentifier

2.2 Besoins fonctionnels

Les besoins fonctionnels ou besoins métiers représentent les actions que le système doit exécuter. Le système ne devient opérationnel que s'il satisfait les besoins fonctionnels. BMG a besoin de développer un portail pour la bonne gestion des qualité .

Pour répondre aux besoins des utilisateurs, notre solution devrait permettre de :

1. Module Authentification :

- Gestion des comptes d'utilisateurs et des droits d'accès
- Authentification sécurisée des utilisateurs pour accéder au système.
- Gestion des sessions utilisateur et des connexions sécurisées.

- Suivi des activités des utilisateurs pour des raisons de sécurité et de conformité.

2. Module Ressources humaines :

- Gestion des fiches d'employés et formations
- Suivi des formations et des compétences.

3. Module Documentation :

- Création et modification de documents.
- Suivi des demandes de création et de modification.
- Contrôle des versions et traçabilité des modifications.

4. Module Fournisseur :

- Gestion des informations fournisseurs.
- Suivi des réclamations et des évaluations des fournisseurs.

5. Module Indicateur :

- Gestion des indicateurs

6. Module Client :

- Gestion des informations clients.
- Suivi des réclamations, enquêtes de satisfaction et suggestions.

7. Module Audit :

- Planification, réalisation et suivi des audits.
- Gestion des rapports d'audit et des actions correctives associées.

8. Module Action :

- Gestion des actions correctives et préventives.
- Suivi de leur mise en uvre et de leur efficacité.

9. Module Réunion :

- Planification des réunions.
- Suivi de l'état de réalisation des réunions.

10. Module Produit non conforme :

- Gestion des non-conformités et des réclamations.
- Suivi des actions correctives associées.

11. Module Évaluation de la conformité réglementaire :

- Veille réglementaire pour identifier les exigences applicables.
- Gestion de la conformité aux normes et réglementations.

12. Module Risque :

- Identification et évaluation des risques potentiels.

2.3 Besoins non fonctionnels

Les besoins non fonctionnels sont indispensables et permettent l'amélioration de la qualité logicielle de notre système. Ils agissent comme des contraintes sur les solutions, mais leur prise en considération fait éviter plusieurs incohérences dans le système. Celui-ci doit répondre aux exigences suivantes :

- **Performance** : Il est impératif que l'application soit performante et ne rencontre aucun problème de lenteur lors de son utilisation.
- **Fiabilité** : Il est essentiel que l'application respecte la fiabilité des données en garantissant que les données utilisées par l'application soient exactes, cohérentes, complètes et disponibles pour les utilisateurs qui en ont besoin.
- **Exactitude** : Il est primordial que l'application garantisse l'exactitude des données en s'assurant que les données utilisées par l'application sont précises et que les résultats produits par l'application correspondent aux attentes et aux spécifications de l'utilisateur.
- **Ergonomie** : l'interface de notre application doit être ergonomique et conviviale. Aussi, elle doit être cohérente, homogènes et facile à manipuler par tous les acteurs de système pour la simplicité d'utilisation tout en garantissant le confort visuel.

3 Pilotage du projet avec SCRUM

Lors de l'utilisation de la méthodologie agile Scrum, les réunions sont très importantes pour l'avancement du projet. Le choix de la méthodologie agile de Scrum offre la possibilité d'un développement rapide, ce qui permettra la réutilisation des fonctions séparément de la plateforme. Cette démarche a évoqué la participation de plusieurs acteurs dans le cadre de notre projet. Avant d'entrer dans le vif du sujet, il faut discuter de la planification et de la structure du projet, qui est l'épine dorsale des méthodes agiles.

3.1 Identification de l'équipe de SCRUM

L'équipe scrum joue un rôle très important dans le choix de la meilleure manière d'achever leur travail. Dans cette section, nous présentons les différents acteurs impliqués dans l'avancement du projet à différentes étapes et leurs rôles associés. Cette équipe comprend : Product Owner, Scrum Master et équipe de développement.

Dans notre projet, la distribution des rôles aux différents participants est établie comme suit dans le tableau 2.1 :

Le Product Owner	Le Scrum Master	L'équipe de développement
Aatizez Mourou	Nour Derouich	Zakaria Ferchichi

Tableau 2.1 : Equipe Scrum

3.2 Backlog du projet

Cette partie sert à jeter les bases pour recueillir les besoins du système à mettre en place. Nous sommes intéressés à identifier les besoins des utilisateurs d'applications à travers des spécifications fonctionnelles et non fonctionnelles afin de réaliser des applications de haute qualité qui répondent aux besoins des utilisateurs logistiques. Nous expliquons en détail la signification des différents termes utilisés dans le Product Backlog :

- **Id** : représente l'identifiant des modules.
- **Module** : Une première répartition du backlog pour faciliter le travail de priorisation des Product Owners.
- **User Story** (récit utilisateur en français) : représente le besoin fonctionnel du client exprimé selon ce format "En tant que. . .Je veux. . .Afin que . . ."
- **Sprint** : représente le sprint auquel appartient le module

Le tableau si dessous reprsénte le Product Backlog :

ID	Module	User Story	Sprint
1	Authentification	<ul style="list-style-type: none"> - En tant qu'administrateur, je veux m'authentifier afin d'accéder à l'application web. - En tant qu'utilisateur, je veux m'authentifier afin d'accéder à l'application web. 	Sprint 1
2	Documentation	<ul style="list-style-type: none"> - En tant que utilisateur , je peux créer ou modifier un document. -En tant que utilisateur je peux valider un document - En tant que utilisateur je peux approuver un document - En tant que utilisateur je peux télécharger , visualiser ou commenter un document 	Sprint 1
3	Ressources humaines	<ul style="list-style-type: none"> - En tant que utilisateur , je peux créer une fiche employé et/ou une fiche formation -En tant que utilisateur je peux ajouter un employé ou un responsable formation . - En tant que utilisateur je peux évaluer les compétences - En tant que utilisateur je peux créer ou consulter une demande de formation - En tant que utilisateur je peux faire évaluation froid ou chaud - En tant que guest je peux accéder à une formation 	Sprint 1
4	Fournisseur	<ul style="list-style-type: none"> - En tant que utilisateur je peux ajouter un fournisseur - En tant que utilisateur je peux créer une réclamation pour un fournisseur - En tant que utilisateur je peux évaluer un fournisseur 	Sprint 2
5	Indicateur	<ul style="list-style-type: none"> - En tant que utilisateur je peux ajouter un indicateur - En tant que utilisateur je peux suivre un indicateur - En tant que utilisateur je peux modifier ou supprimer un indicateur 	Sprint 2

6	Client	- En tant que utilisateur je peux ajouter un client	Sprint 2
		- En tant que utilisateur je peux ajouter une réclamation client	
		- En tant que utilisateur je peux créer une enquête satisfaction client	
		- En tant que utilisateur je peux ajouter une suggestion client	
7	Réunion	- En tant que utilisateur je peux planifier une réunion	Sprint 3
		-En tant que utilisateur je peux suivre l'état de réalisation de réunion	
		- En tant que guest ou utilisateur je peux participer à une réunion	
8	Actions	- En tant que utilisateur je peux ajouter , modifier action	Sprint 3
9	Audit	- En tant que utilisateur je peux planifier un audit .	Sprint 3
		- En tant que utilisateur je peux ajouter un plan audit	
		- En tant que utilisateur je peux consulter l'audit	
		- En tant que utilisateur je peux modifier ou supprimer un audit	
10	Produit non conforme	- En tant que utilisateur je peux ajouter non conformité	Sprint 4
		- En tant que utilisateur je peux traiter non conformité	
		- En tant que utilisateur je peux suivre non conformité	
11	Conformité réglementaire	- En tant que utilisateur je peux examiner les informations relatives à la conformité réglementaire	Sprint 4
12	Risque	- En tant que utilisateur je peux identifier et de traiter les risques potentiels	Sprint 4

Tableau 2.2 : Product Backlog

3.3 Planification des sprints

Après avoir réalisé notre Backlog du produit, nous avons organisé une réunion de planification avec l'équipe Scrum et nous avons divisé le travail en des sprints comme le montre la figure 2.1 ci-dessous :

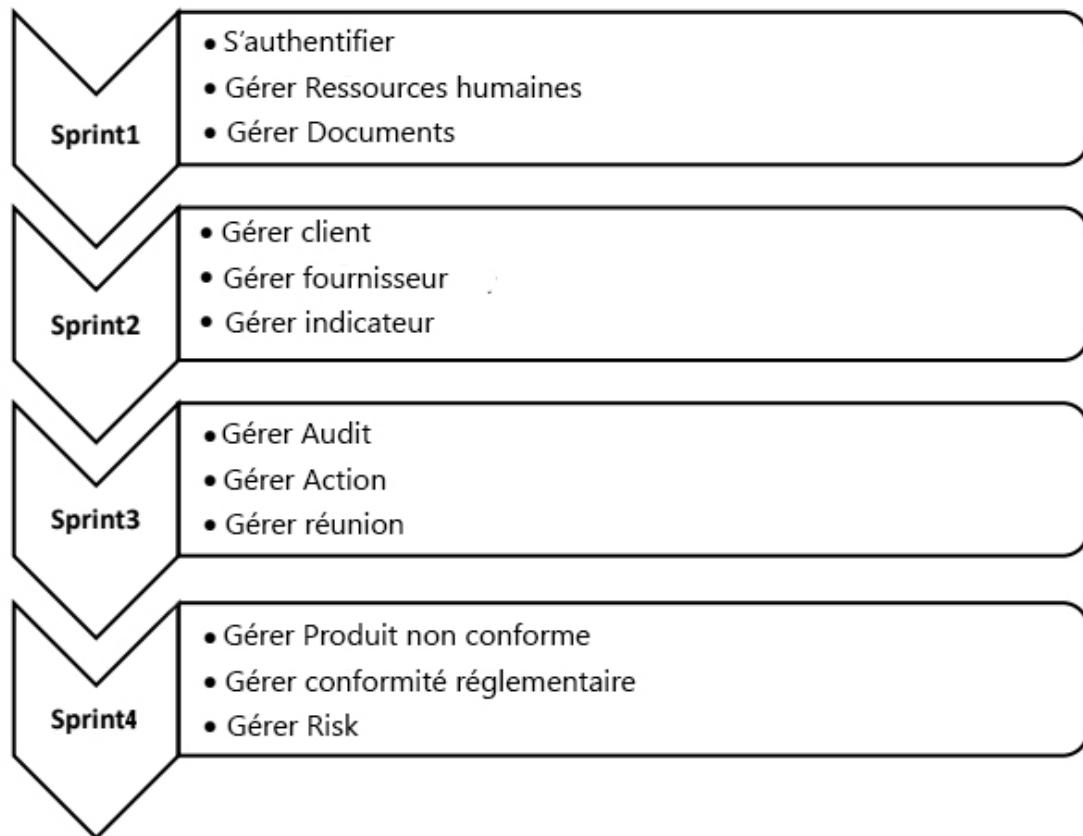


Figure 2.1 : Planification sprints

4 Diagramme de cas d'utilisation global

Les diagrammes de cas d'utilisation illustrent et définissent le contexte et les exigences d'un système entier, ou des parties essentielles d'un système. Ces diagrammes identifient également les interactions entre le système et ses acteurs. Nous présentons dans la figure ci-dessous notre diagramme de cas d'utilisation globale décrivant les grandes fonctionnalités du déroulement de notre projet, mais ce diagramme général ne montre pas de façon détaillée le dialogue entre les acteurs et les différents profils de chaque utilisateur. Dans les chapitres qui suivent, nous détaillerons, avec raffinement itératif, les différents cas d'utilisation et nous mettrons les différents acteurs par profile.

Nous allons dans la figure 2.2 présenter le diagramme des cas d'utilisation générale

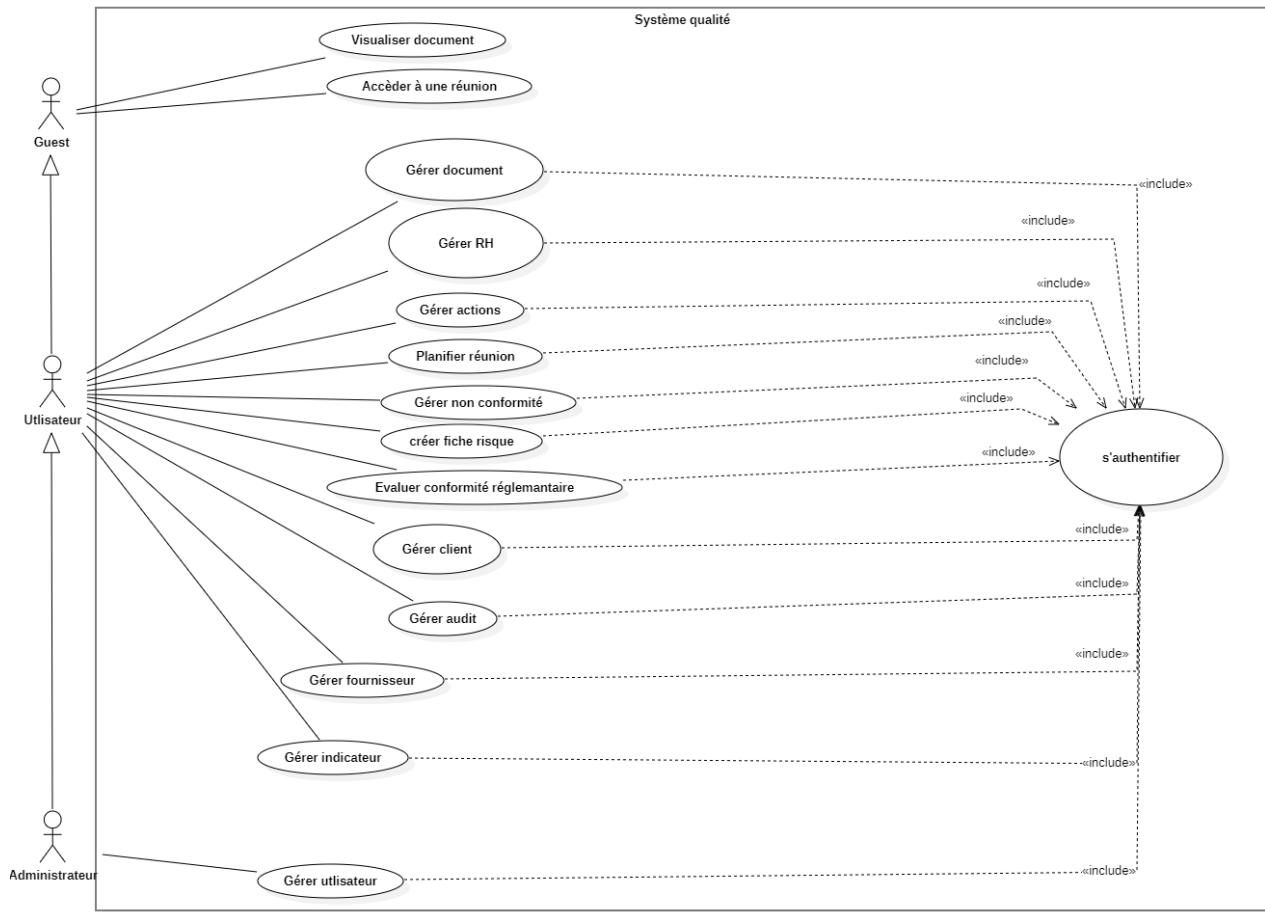


Figure 2.2 : Diagramme de cas d'utilisation global

5 Environnement de travail

Nous allons présenter dans cette section l'environnement de la réalisation de l'application. Ainsi nous avons eu recours à des outils de développements, et les caractéristiques de la machine sur laquelle nous avons travaillé.

5.1 Environnement Matériels

Notre application a été développée sur un ordinateur DELL G15 5520 dont la configuration est présentée comme suit dans le tableau 2.3 :

Processeur	Intel Core i7-12700H
Mémoire Ram	16 GO
Disque Dur	512 GO
Système d'exploitation	Windows 11

Tableau 2.3 : Environnement matériel

5.2 Environnement logiciel et développement

Dans cette section, nous aborderons les outils logiciels que nous avons utilisés pour développer et concevoir notre application .

L'éditeur de code

Visual Studio Code est un éditeur de code extensible développé par Microsoft pour Windows, Linux et macOS[2] .



Figure 2.3 : Logo Visual Studio Code

Base de données

PostgreSQL est un système de gestion de base de données relationnelle orienté objet puissant et open source qui est capable de prendre en charge en toute sécurité les charges de travail de données les plus complexes [3].

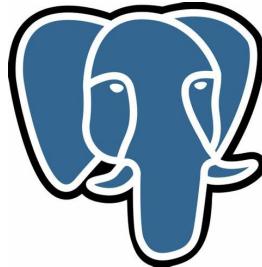


Figure 2.4 : Logo PostgreSQL

Backend

Django est un framework web Python de haut niveau qui encourage un développement rapide et une conception propre et pragmatique. Construit par des développeurs expérimentés, il s'occupe d'une grande partie des tracas du développement Web, afin que vous puissiez vous concentrer sur l'écriture de votre application sans avoir besoin de réinventer la roue. C'est gratuit et open source[4].



Figure 2.5 : Logo Django

django REST framework

Django Rest Framework (DRF) est une boîte à outils puissante et flexible pour la création d'API Web. Elle offre une manière simple et cohérente de construire des API, et est construite sur le framework Django. DRF fournit de nombreuses fonctionnalités intégrées, y compris l'authentification, les permissions, la sérialisation, et bien plus encore. Elle offre également une grande flexibilité, vous permettant de la personnaliser et de l'étendre pour répondre à vos besoins [5].



Figure 2.6 : Logo Rest framework

JSON Web Token

JWT est un mécanisme d'authentification basé sur des jetons qui permet à un serveur d'authentifier un client (par exemple, un utilisateur) sans stocker d'informations de session sur le serveur. C'est un mécanisme d'authentification sans état, ce qui signifie que le serveur n'a pas besoin de stocker d'informations sur le client pour vérifier son identité [6].



Figure 2.7 : Logo JWT

Frontend

React (également connu sous le nom de React.js ou ReactJS) est une bibliothèque JavaScript front-end pour la création d'interfaces utilisateur ou de composants d'interface utilisateur. Il est maintenu par Facebook et une communauté de développeurs individuels et d'entreprises. React vous permet de concevoir des vues simples pour chaque état de votre application, et il mettra à jour et rendra efficacement les bons composants lorsque vos données changent. Il utilise un DOM virtuel pour augmenter les performances de l'application et permet aux développeurs de créer des composants d'interface utilisateur réutilisables[7].

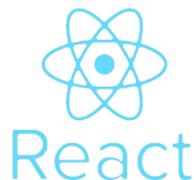


Figure 2.8 : Logo React

Postman

Postman est une plate-forme d'API pour la création et l'utilisation d'API. Postman simplifie chaque étape du cycle de vie des API et rationalise la collaboration afin que vous puissiez créer de meilleures API, plus rapidement[8].



Figure 2.9 : Logo Postman

Conception

StarUML est une application de modélisation UML (Unified Modeling Language) et de modélisation logicielle. Conçue pour être une plateforme extensible, StarUML permet aux développeurs de créer des diagrammes UML pour représenter les structures et les comportements des systèmes logiciels. Elle offre diverses fonctionnalités pour faciliter la modélisation[9].



Figure 2.10 : Logo Draw.io

Rapport

LaTeX (prononcé soit "LAY-tek" soit "LAH-tek" selon à qui vous parlez) est un outil qui permet à quiconque de créer des documents magnifiquement mis en page. C'est le code que vous écrivez lorsque vous utilisez l'Éditeur de Code d'Overleaf et il est à la base de ce que vous voyez dans notre Éditeur Visuel[10].



Figure 2.11 : Latex

Outils de collaboration

GitHub est un service d'hébergement Open-Source, permettant aux programmeurs et aux développeurs de partager le code informatique de leurs projets afin de travailler dessus de façon collaborative. Nous pouvons le considérer comme un Cloud dédié au code informatique[11].

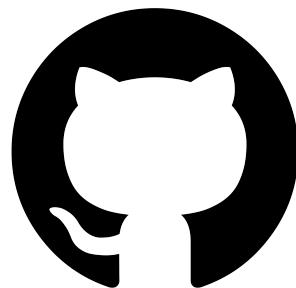


Figure 2.12 : GitHub

Gmail est l'abréviation de Google Mail. Il s'agit du service de messagerie que propose Google. Très pratique et simple à utiliser, elle est gratuite pour les particuliers et payante pour les professionnels[12].



Figure 2.13 : Gmail

6 Architecture de l'application

6.1 Architecture physique

Notre application suit une architecture 3-tiers, dont les couches sont les suivantes :

- La couche de présentation : liée à l'interface utilisateur. Elle permet d'afficher les données et de laisser l'utilisateur final d'interagir avec ces derniers.
- La couche métier : est en charge d'appliquer et respecter les algorithmes métiers. Elle correspond à la partie fonctionnelle de l'application, celle qui implémente la logique applicative et la sécurité.
- La couche de données : correspond au serveur qui comprend les bases de données de configuration, les bases de données de contenu et les bases de données liées au applications de serveur.

La figure 2.14 présente l'architecture physique de l'application

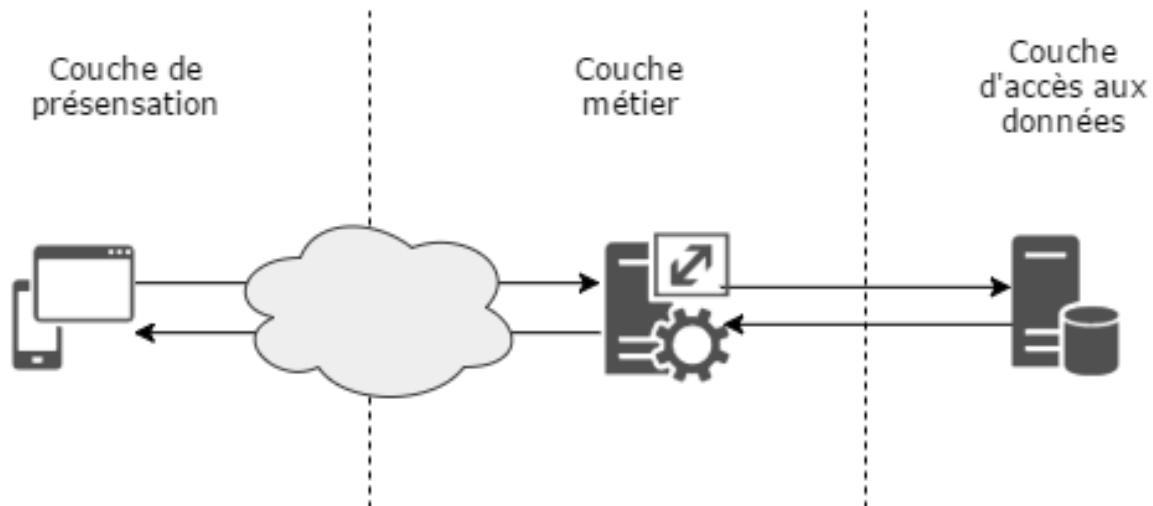


Figure 2.14 : L'architecture 3-tiers

6.2 Architecture logique

Nous avons décidé d'utiliser le modèle MVC dans la conception de notre application. Le modèle découpe littéralement l'application en des entités distinctes, et de ce fait impacte très fortement l'organisation du code.

Ces entités sont :

- Le Modèle (le M de MVC) : cette entité contient tout ce qui concerne le traitement, le stockage et la mise à jour des données d'application.

- La Vue (le V de MVC) : correspond à l'IHM (interface homme machine), elle permet la présentation des données du modèle (affichage).
- Le Contrôleur (le C de MVC) : cette entité contient tout ce qui concerne le contrôle des actions de l'utilisateur et des données, elle assure l'échange entre la vue et le modèle. .

La figure 2.15 présente l'architecture logique MVC :

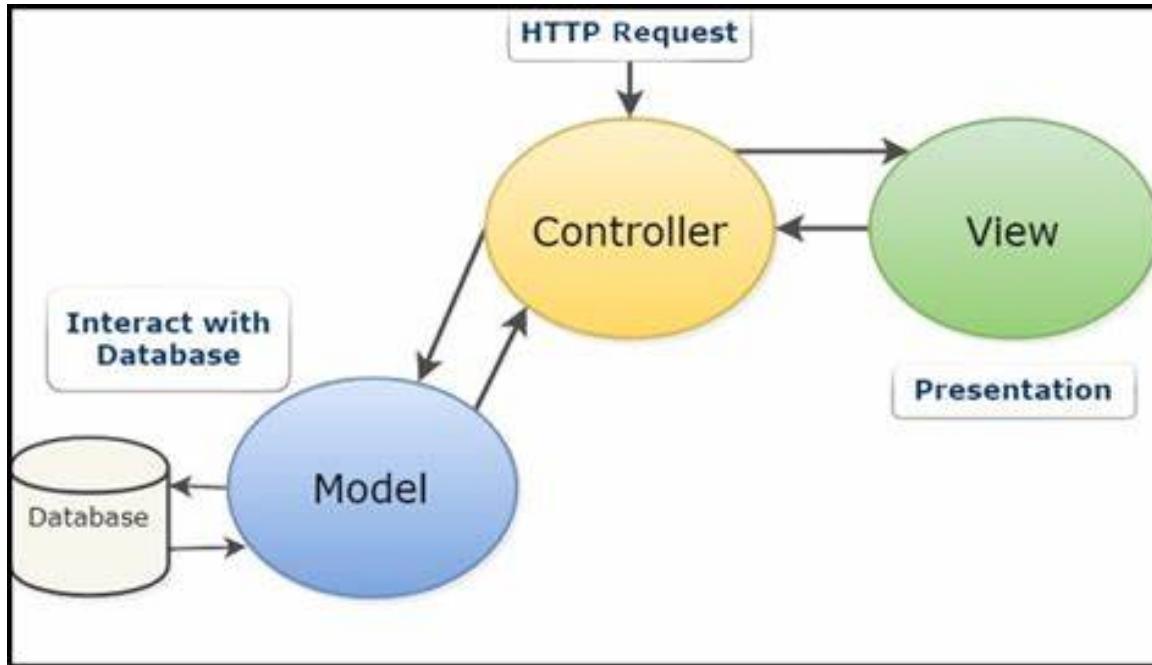


Figure 2.15 : L'architecture MVC

7 Conclusion

Dans ce chapitre, nous avons présenté les besoins de notre projet, aussi bien fonctionnels et non fonctionnels, le backlog du projet ainsi que la planification des sprints. De plus, nous avons élaboré le diagramme de cas d'utilisation général. À ce niveau, nous pouvons passer les sections suivants qui comporteront la planification et l'architecture du projet. Enfin, nous avons présenté l'architecture de notre application.

Dans le chapitre suivant, nous commencerons par le premier sprint qui couvrira la première étape du développement d'application

Sprint 1 : Authentification , Gestion des ressources humaines et documents

Plan

1	Introduction	27
2	Backlog Sprint 1	27
3	Besoins fonctionnels	28
4	Mise en place de module Authentification	29
5	Mise en place de module Ressources humaines	33
6	Mise en place de module Documentation	39
7	Réalisation	45
8	Test	47
9	Conclusion	49

1 Introduction

Dans ce chapitre, nous allons traiter le premier sprint de notre projet, qui est consacré à l'authentification , la gestion des RH et la gestion des documents.

Nous commençons par présenter le backlog du sprint à faire, puis nous montrons la conception de cette partie. Nous terminons avec l'implémentation et les tests pour s'assurer que l'application répond à nos besoins.

2 Backlog Sprint 1

Le sprint backlog est un ensemble de user stories qui sont des descriptions simples et courtes des fonctionnalités attendues du produit.

Chaque user story est accompagné d'une estimation de l'effort nécessaire pour la réaliser.

Module	User Story	Taches	Estimation
Authentification	- En tant qu'administrateur, je veux m'authentifier afin d'accéder à l'application web.	-Conception et création de l'interface	1
	- En tant qu'administrateur, je peux grouper les utilisateurs accorder les permissions et gérer les utilisateurs.	-Réalisation -Implémentation de code	
	- En tant que utilisateur, je veux m'authentifier afin d'accéder à l'application web.	-Test de l'interface	
Ressources humaines	- En tant que responsable RH , je peux créer une fiche employé et/ou fiche formation	-Implémentation des fonctionnalités	1
	-En tant que Responsable RH je peux ajouter un employé ou un responsable formation .	-Mise en place de l'interface	
	- En tant que responsable RH je peux évaluer les compétences		

	<ul style="list-style-type: none"> - En tant que Employé je peux créer ou consulter une demande de formation - En tant que participant ou responsable formation je peux faire évaluation froid ou chaud - En tant que Guest ou employé je peux accéder à une formation 	<ul style="list-style-type: none"> -Test des fonctionnalités - Réalisation 	
Documentation	<ul style="list-style-type: none"> - En tant que Redacteur ou superviseur , je peux créer ou modifier document 	<ul style="list-style-type: none"> -Implémentation des fonctionnalités 	
	<ul style="list-style-type: none"> -En tant que vérificateur je peux valider document 	<ul style="list-style-type: none"> -Test des fonctionnalités 	1
	<ul style="list-style-type: none"> - En tant que approuvateur je peux approuver document 	<ul style="list-style-type: none"> -Mise en place de l'interface 	
	<ul style="list-style-type: none"> - En tant que utilisateur je peux télécharger , visualiser ou commenter un document 	<ul style="list-style-type: none"> -Réalisation 	
	<ul style="list-style-type: none"> - En tant que superviseur je peux archiver documents 		

Tableau 3.1 : Backlog Sprint 1

3 Besoins fonctionnels

Notre application doit assurer plusieurs fonctionnalités, notamment :

- Authentification : Cette fonctionnalité offre aux utilisateurs les autorisations pour accéder au différents interfaces
- Créer un document : Cette fonctionnalité offre aux utilisateurs la possibilité de créer un document avec un processus bien déterminé
- Visualiser un document : Cette fonctionnalité offre aux utilisateurs la possibilité d'accéder à un document et l'imprimer ou commenter.
- Gérer les fiches employés : Cette fonctionnalité offre aux responsable RH la possibilité de

créer,modifier,consulter ou supprimer les fiches

- Gérer les formations : Cette fonctionnalité offre aux responsable RH la possibilité de créer,modifier,consulter ou supprimer les formations
- Gérer les employés,participants et responsable formations : Cette fonctionnalité offre aux responsable RH la possibilité de créer,modifier,consulter ou supprimer les formations
- Évaluer compétences : Cette fonctionnalité offre aux responsable RH la possibilité de évaluer les compétences d'un employé

4 Mise en place de module Authentification

Dans cette section,nous allons représenté les diagrammes de classes, de cas d'utilisation et de séquence pour le module authentification

4.1 Diagramme de classes

La figure 3.1 représente le diagramme de classes d'authentification

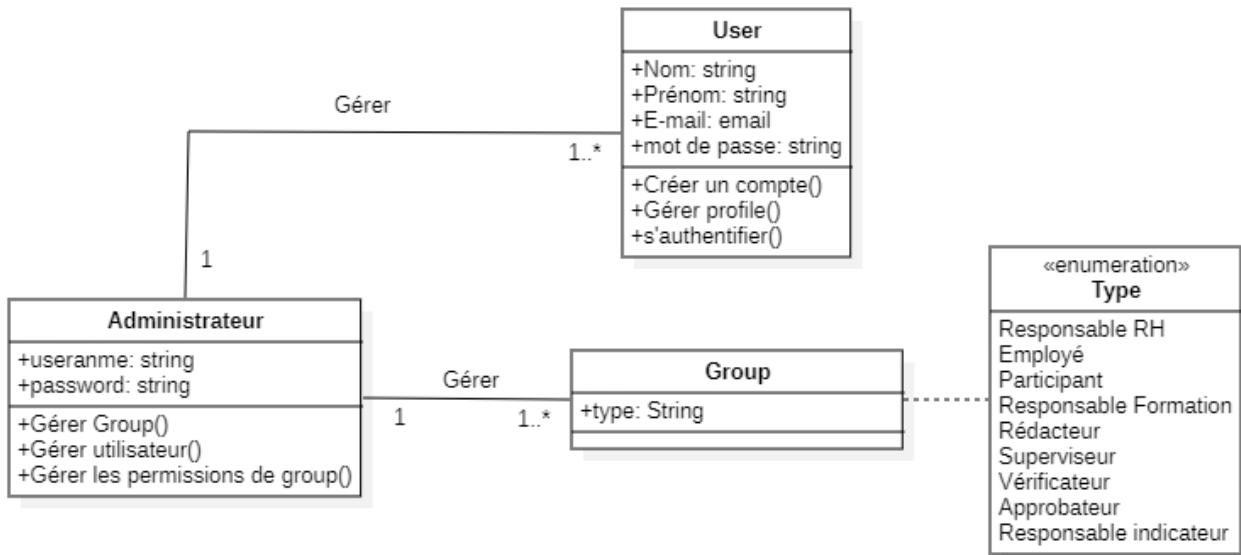


Figure 3.1 : Diagramme de classes d'authentification

4.2 Diagramme de cas d'utilisation

La figure 3.2 représente le diagramme de cas d'utilisation d'authentification

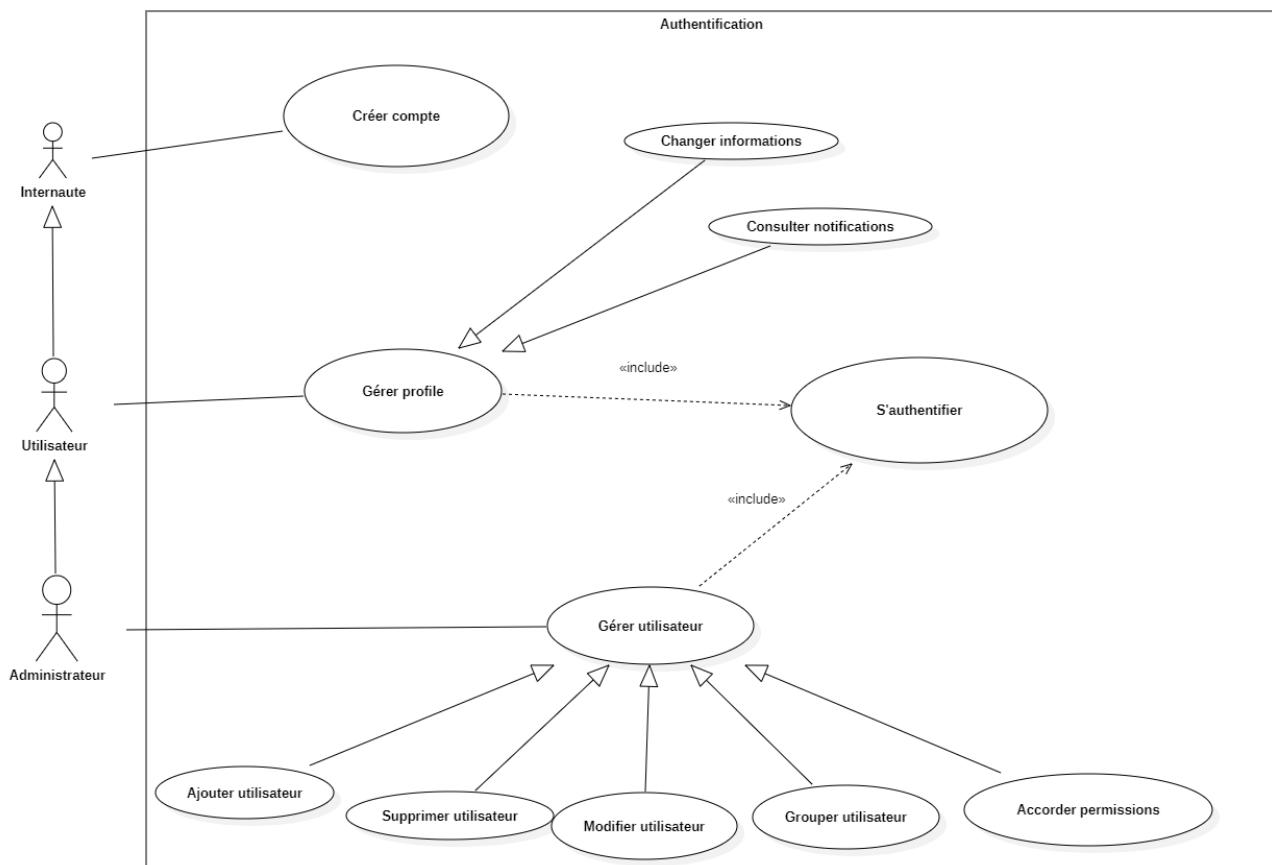


Figure 3.2 : Diagramme de cas d'utilisation d'authentification

4.2.1 Description textuelle des cas d'utilisation

Dans le but de mieux comprendre notre système et ses interactions avec les utilisateurs, dans cette partie nous allons détailler les scénarios des principaux cas d'utilisation.

- Description textuelle du cas d'utilisation "Grouper les utilisateurs"

Cas d'utilisation	Grouper les utilisateurs
Acteur	Administrateur
Pré-condition	L'administrateur doit être authentifié
Scénario nominal	-L'administrateur crée un groupe -L'administrateur classe l'utilisateur dans ce groupe
Post-condition	L'utilisateur est associé à un groupe

Tableau 3.2 : Description textuelle du cas d'utilisation "Grouper les utilisateurs"

- Description textuelle du cas d'utilisation "Accorder permissions"

Cas d'utilisation	Accorder permissions
--------------------------	----------------------

Acteur	Administrateur
Pré-condition	L'administrateur doit être authentifié
Scénario nominal	<ul style="list-style-type: none"> -L'administrateur sélectionne l'utilisateur -L'administrateur attribue des privilèges à l'utilisateur
Post-condition	L'utilisateur prend les accès

Tableau 3.3 : Description textuelle du cas d'utilisation "Accorder permissions"

- Description textuelle du cas d'utilisation "Créer compte"

Cas d'utilisation	Créer compte
Acteur	Utilisateur
Scénario nominal	<ul style="list-style-type: none"> -l'utilisateur saisit le nom,prénom,nom d'utilisateur,e-mail,mot de passe -Utilisateur confirme la saisie de ces données
Exception	<ul style="list-style-type: none"> - Dans le cas où les informations saisies ne sont pas valides ou l' e-mail déjà existe le système envoie un message d'erreur.
Post-condition	Compte créé et l'utilisateur peut authentifier

Tableau 3.4 : Description textuelle du cas d'utilisation "Créer compte"

- Description textuelle du cas d'utilisation "S'authentifier "

Cas d'utilisation	S'authentifier
Acteur	Utilisateur
Pré-condition	Utilisateur approuvé.
Scénario nominal	<ul style="list-style-type: none"> -Utilisateur saisit son e-mail et son mot de passe. -Utilisateur confirme la saisie de ces données d'identification. -Le système vérifie les données d'identification. -Le système affiche l'interface d'accueil.

Scénario alternatif	-l'utilisateur clique sur la button mot de passe oublié . -L'utilisateur reçoit un mail de ré initialisation -L'utilisateur réinitialise son mot de passe .
Post-condition	Utilisateur authentifié
Exception	Si les identifiants(e-mail,nom mot de passe) sont non valides, le système envoie un message d'erreur.

Tableau 3.5 : Description textuelle du cas d'utilisation "S'authentifier "

- Description textuelle du cas d'utilisation "Gérer profile"

Cas d'utilisation	Gérer profile
Acteur	Utilisateur
Pré-condition	Utilisateur authentifié.
Scénario nominal	-Utilisateur accède à son profile. -Utilisateur peut consulter les notifications ou état d'un demande . -L'utilisateur peut changer ses informations tel que mot de passe,nom d'utilisateur
Post-condition	Informations changés

Tableau 3.6 : Description textuelle du cas d'utilisation "Gérer profile"

4.3 Diagrammes de séquence

La figure 3.3 représente le diagramme de séquence d'authentification

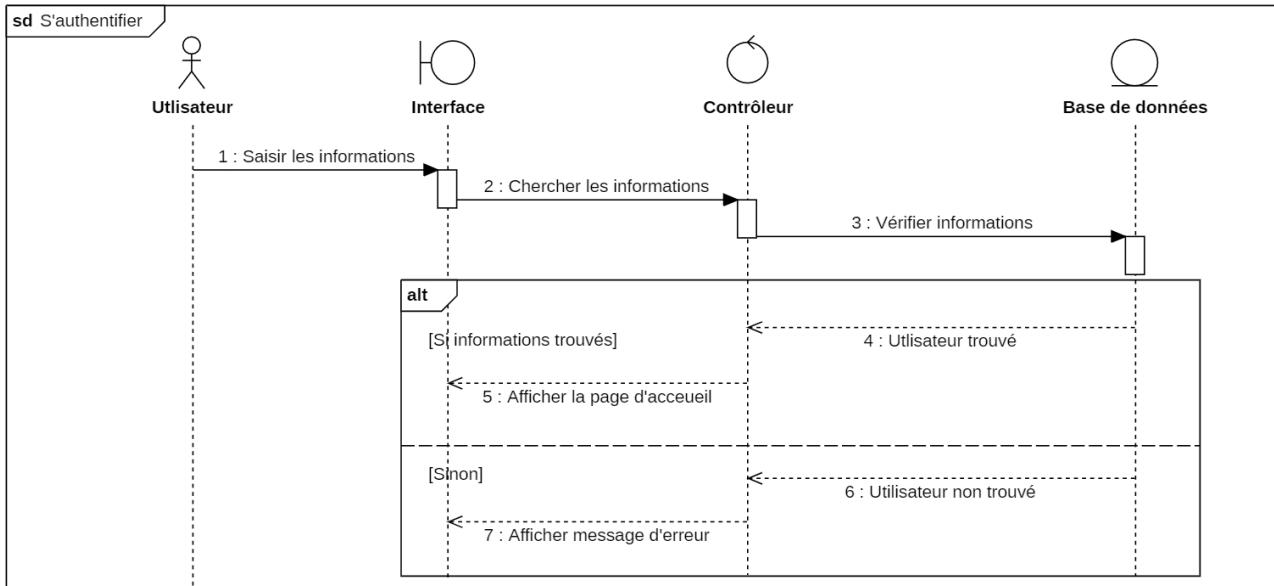


Figure 3.3 : Diagramme de séquence d'authentification

La figure 3.4 représente le diagramme de séquence de gérer profile

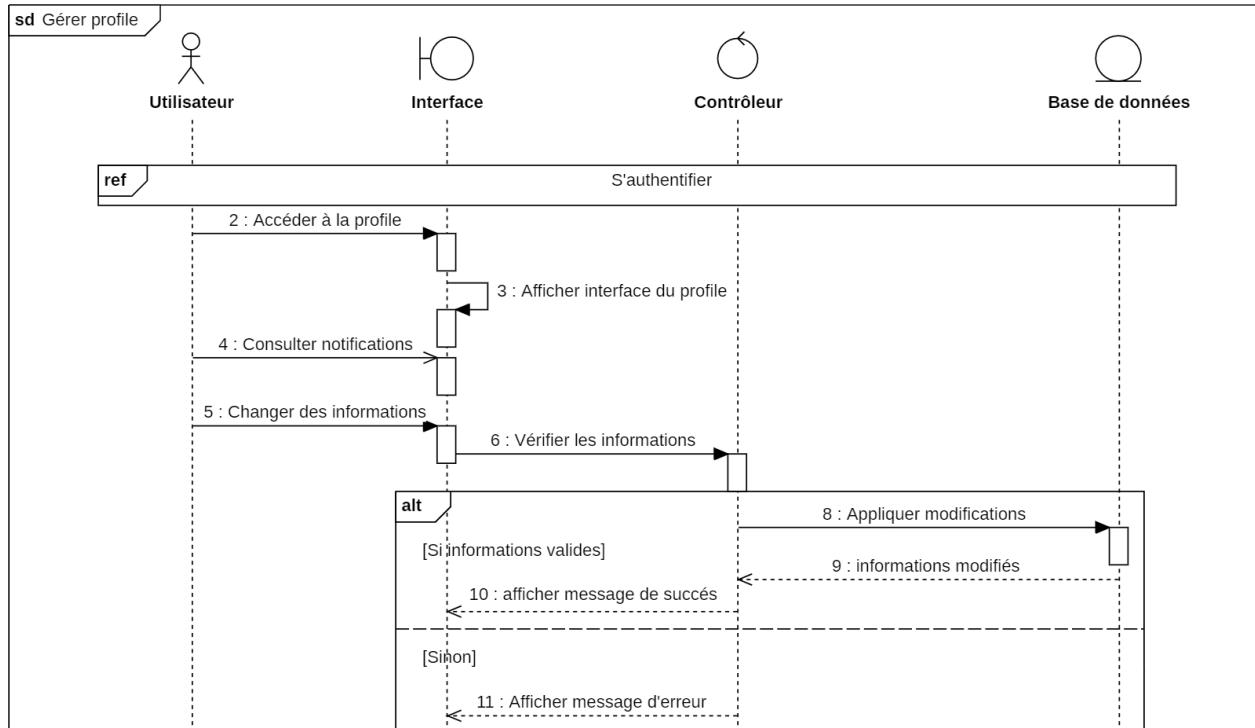


Figure 3.4 : Diagramme de séquence de "Gérer profile"

5 Mise en place de module Ressources humaines

Dans cette section,nous allons représenté les diagrammes de classe,cas d'utilisation et de séquence pour le module RH

5.1 Diagramme de classes

La figure 3.5 représente le diagramme de module RH

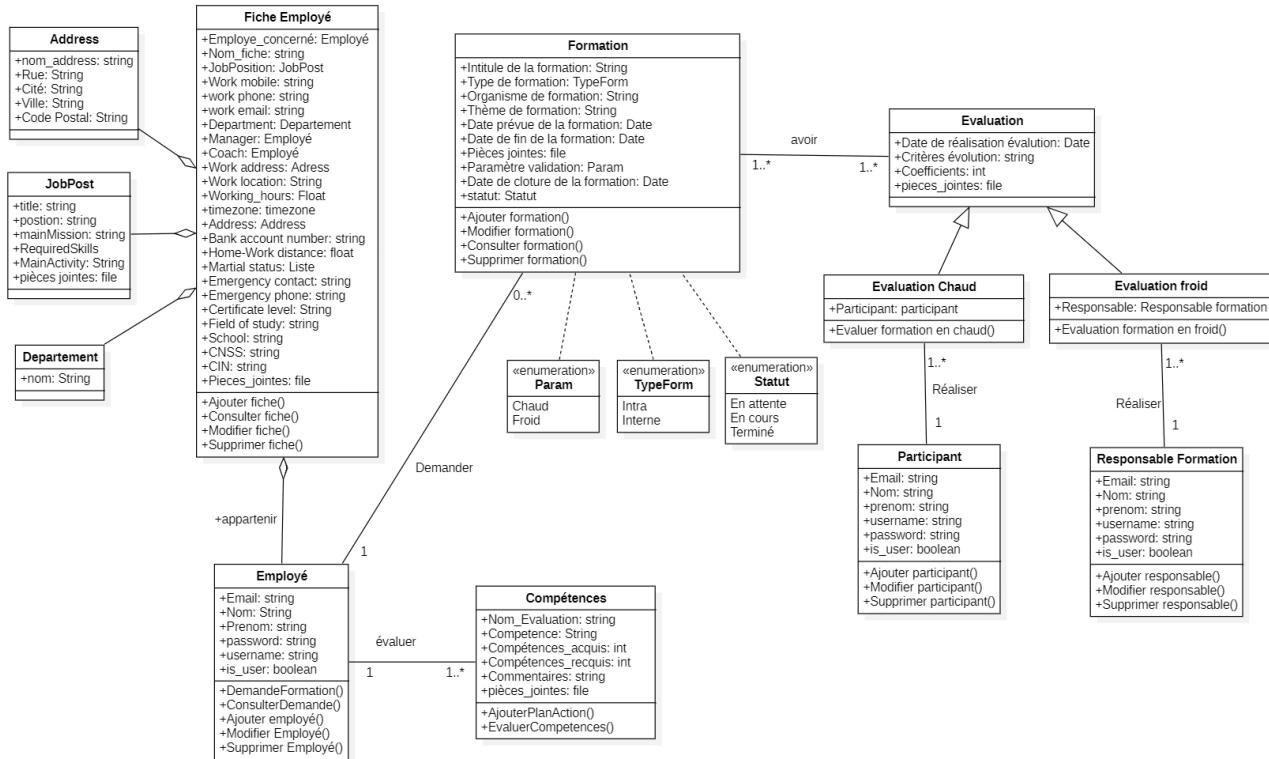


Figure 3.5 : Diagramme de classes de module RH

5.2 Diagramme de cas d'utilisation

La figure ci dessous représente le diagramme de cas d'utilisation de module RH

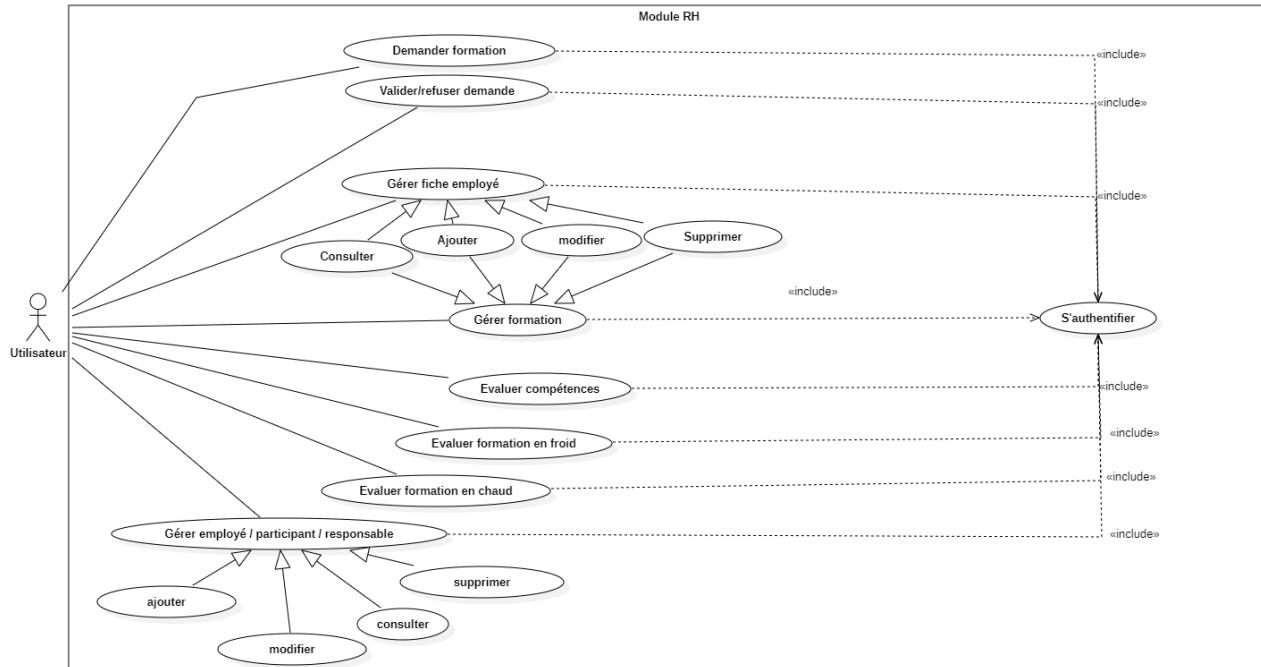


Figure 3.6 : Diagramme de cas d'utilisation de module RH

5.2.1 Description textuelle des cas d'utilisation

Dans le but de mieux comprendre notre système et ses interactions avec les utilisateurs, dans cette partie nous allons détailler les scénarios des principaux cas d'utilisation.

- Description textuelle du cas d'utilisation "ajouter employé"

Cas d'utilisation	ajouter employé
Acteur	Utilisateur
Pré-condition	L'utilisateur doit appartenir au groupe responsable RH et authentifié
Scénario nominal	-L'utilisateur saisit les données d'un employé tel que (nom,prénom,email,isuser...)
Scénario alternatif	<ul style="list-style-type: none"> - Si l'utilisateur coche le champ "isuser" , le système génère un mot de passe est l'envoyé à son gmail - L'employé devient un utilisateur et peut autntifié
Post-condition	L'ajout d'un nouveau employé

Tableau 3.7 : Description textuelle du cas d'utilisation "ajouter employé"

- Description textuelle du cas d'utilisation "Créer fiches employé"

Cas d'utilisation	Créer fiche employé
Acteur	Utilisateur
Pré-condition	L'utilisateur doit appartenir au groupe responsable RH et authentifié
Scénario nominal	-L'utilisateur remplit une fiche qui contient les données d'un employé tel que (poste,numéro,adresse,département...)
Post-condition	L'ajout d'une nouvelle fiche

Tableau 3.8 : Description textuelle du cas d'utilisation "Créer fiches employé"

- Description textuelle du cas d'utilisation "Consulter demande"

Cas d'utilisation	Consulter demande
--------------------------	-------------------

Acteur	Employé
Pré-condition	L'utilisateur doit être authentifié, la demande est envoyée et appartient au groupe Employé
Scénario nominal	-L'employé examine le statut de la demande (acceptée, refusée, en attente)
Post-condition	L'employé vérifie le statut

Tableau 3.9 : Description textuelle du cas d'utilisation "Consulter demande"

- Description textuelle du cas d'utilisation "Évaluer compétences "

Cas d'utilisation	Évaluer compétences
Acteur	Utilisateur
Scénario nominal	-le responsable RH sélectionne un employé et faire une évaluations selon le niveau acquis par rapport niveau requis
Post-condition	Employé évaluer
Exception	Si niveau acquis inférieur à niveau requis , un plan d'action va être créer

Tableau 3.10 : Description textuelle du cas d'utilisation "Évaluer compétences"

- Description textuelle du cas d'utilisation "Évaluer en froid "

Cas d'utilisation	Évaluer en froid
Acteur	Utilisateur
Pré-condition	L'utilisateur doit être authentifié, appartient au groupe Responsable formation et la formation est terminée.
Scénario nominal	-Le responsable formation sélectionne une formation -Le responsable formation remplit une fiche qui contient des informations tel que (commentaires, date, coefficients) .
Post-condition	Formation évaluée en froid

Tableau 3.11 : Description textuelle du cas d'utilisation "Évaluer en froid "

- Description textuelle du cas d'utilisation "Valider ou refuser demande"

Cas d'utilisation	Valider ou refuser demande
Acteur	Utilisateur
Pré-condition	L'utilisateur doit appartenir au groupe responsable RH et authentifié.
Scénario nominal	<ul style="list-style-type: none"> -Le responsable RH sélectionne une demande -Le responsable RH modifie le statut pour accepter ou refuser -Le responsable RH envoie la réponse à l'employé(demandeur)
Post-condition	Demande changée

Tableau 3.12 : Description textuelle du cas d'utilisation "Valider ou refuser demande"

5.3 Diagrammes de séquence

La figure 3.7 représente le diagramme de séquence de "Ajouter fiche Employé"

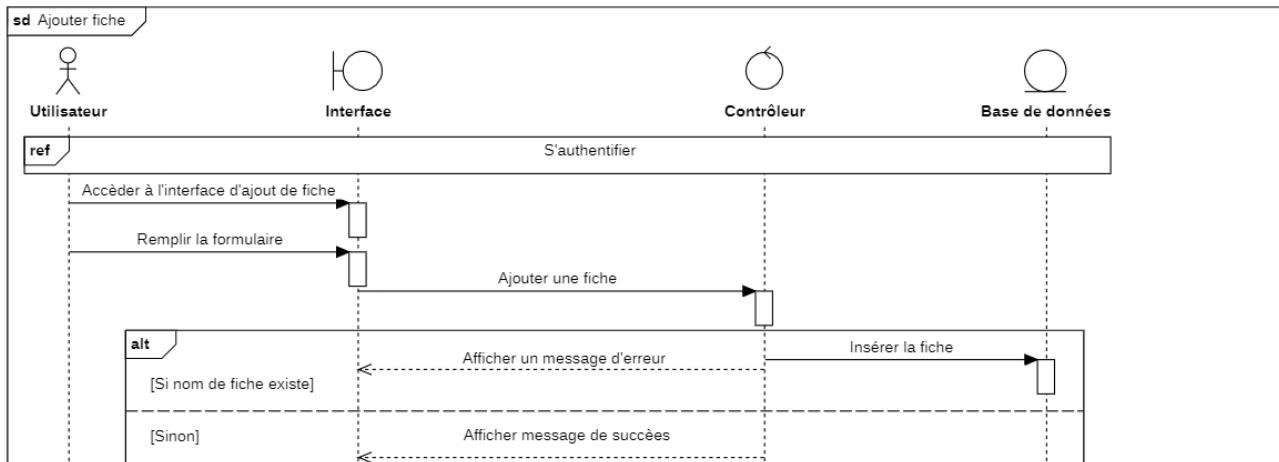


Figure 3.7 : Diagramme de séquence de Ajouter fiche Employé

La figure 3.8 représente le diagramme de séquence de "Modifier fiche Employé"

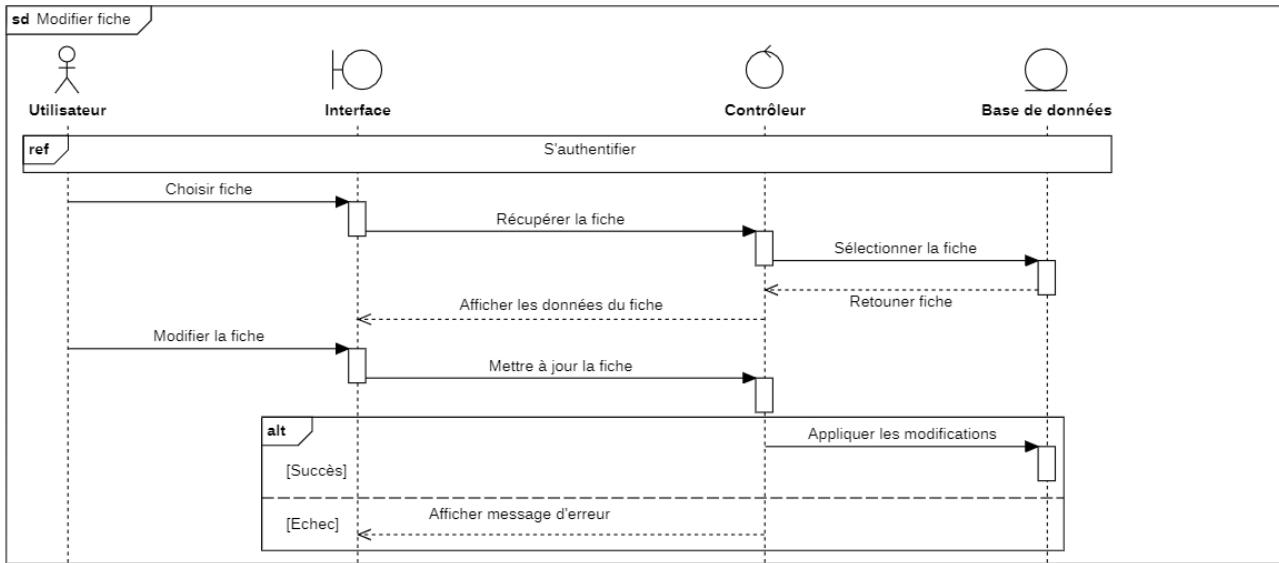


Figure 3.8 : Diagramme de séquence de Modifier fiche Employé

La figure 3.9 représente le diagramme de séquence de "Supprimer fiche Employé"

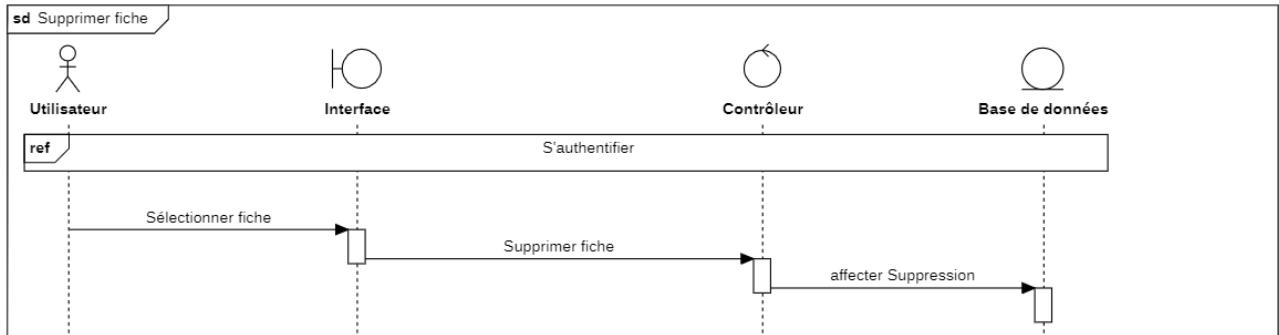


Figure 3.9 : Diagramme de séquence de Supprimer fiche Employé

La figure 3.10 représente le diagramme de séquence de Ajouter employé"

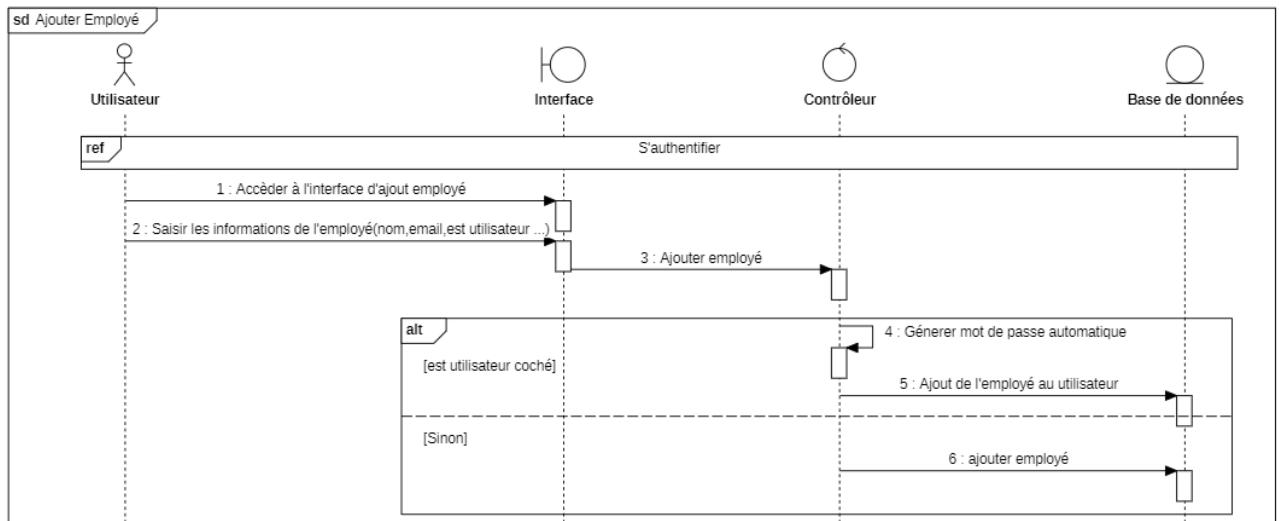


Figure 3.10 : Diagramme de séquence "Ajouter employé"

La figure 3.11 représente le diagramme de séquence de Évaluer compétences

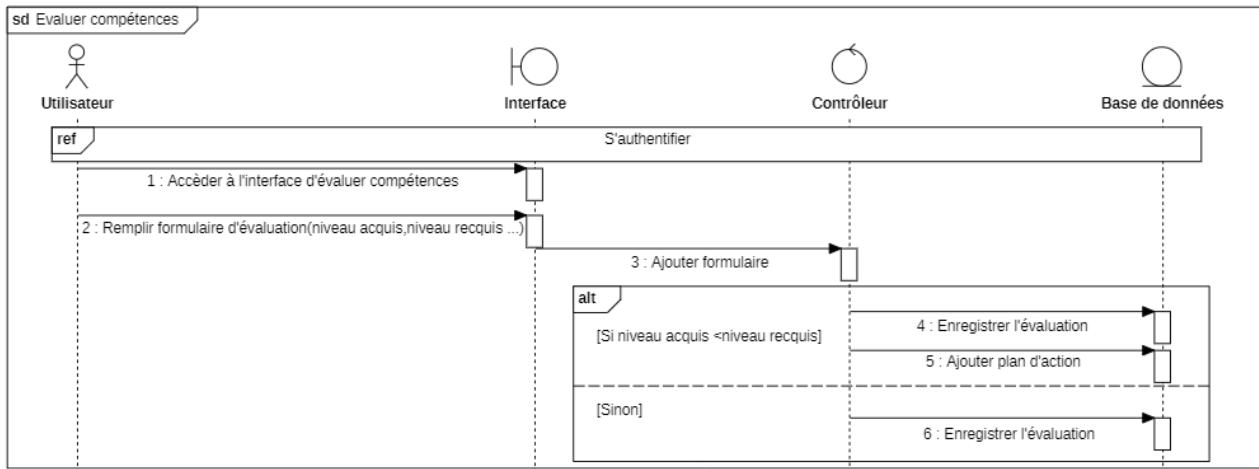


Figure 3.11 : Diagramme de séquence "Évaluer compétences"

6 Mise en place de module Documentation

Dans cette section , nous allons représenté les diagrammes de classes , cas d'utilisation et de séquence pour le module Documentation .

6.1 Diagramme de cas d'utilisation

La figure 3.12 représente le diagramme de cas d'utilisation de Documentation

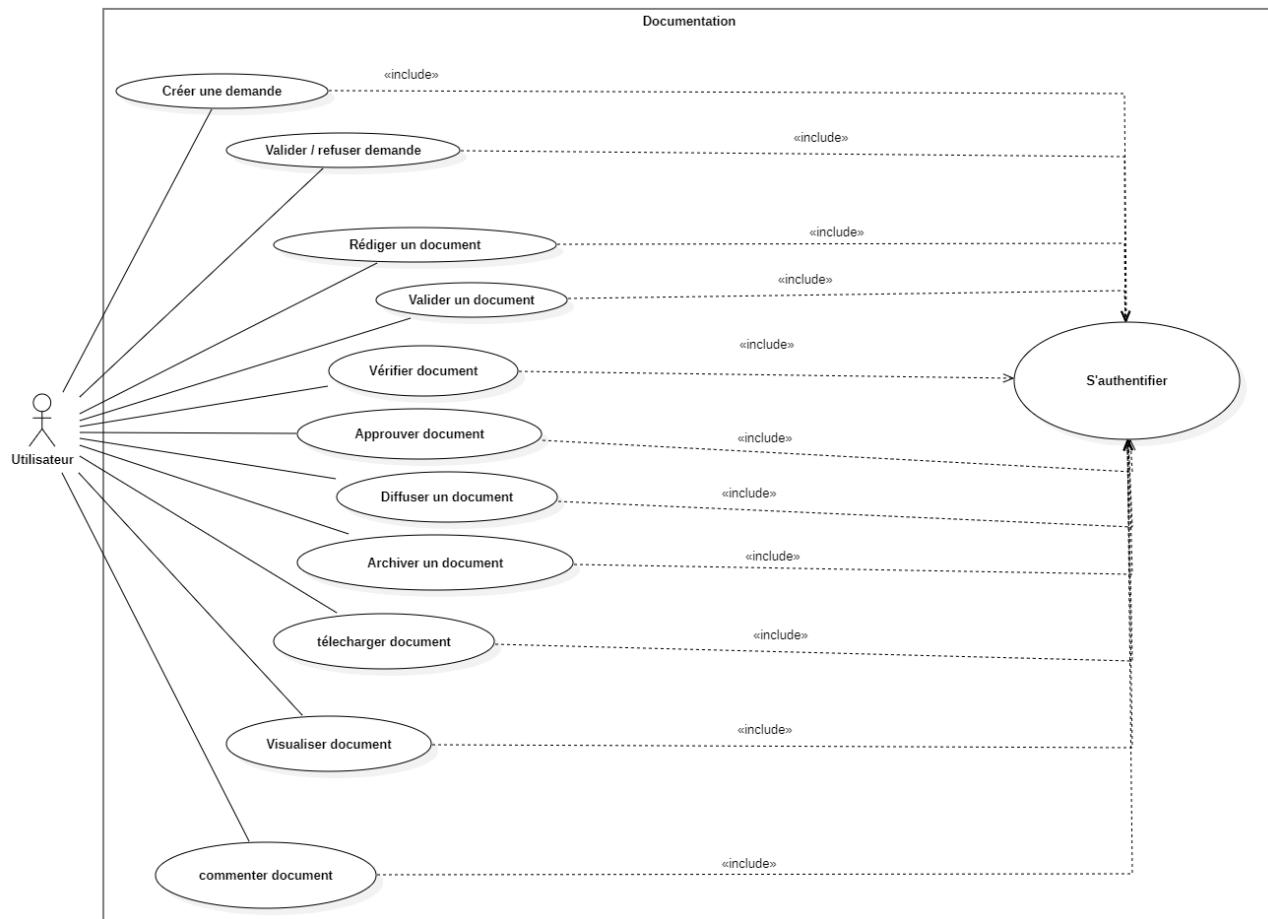


Figure 3.12 : Diagramme de cas d'utilisation de Documentation

6.1.1 Description textuelle des cas d'utilisation

Dans le but de mieux comprendre notre système et ses interactions avec les utilisateurs, dans cette partie nous allons détailler les scénarios des principaux cas d'utilisation.

- Description textuelle du cas d'utilisation "Créer ou modifier document"

Cas d'utilisation	Créer ou modifier document
Acteur	Utilisateur
Pré-condition	L'utilisateur doit appartenir au groupe rédacteur ou superviseur , authentifié et la demande acceptée
Scénario nominal	-L'utilisateur crée un document en remplissant les champs tel que (code,libelle...) -Le rédacteur/superviseur valide les informations

Post-condition	L'ajout d'un nouveau document en attendant qu'il soit valider depuis un utilisateur de type superviseur
-----------------------	---

Tableau 3.13 : Description textuelle du cas d'utilisation "Créer ou modifier document"

- Description textuelle du cas d'utilisation "Accepter ou refuser demande"

Cas d'utilisation	Accepter ou refuser demande
Acteur	Utilisateur
Pré-condition	L'utilisateur doit être authentifié, la demande est envoyée et appartient au groupe Superviseur
Scénario nominal	-Le superviseur examine la demande -Le superviseur accepte la demande en changeant le statut en valider et notifie le rédacteur ou refuse la demande en changeant le statut en refuser et notifie le rédacteur en donnant un motif de refus et commentaire
Post-condition	L'utilisateur examine la demande

Tableau 3.14 : Description textuelle du cas d'utilisation "Accepter ou refuser demande"

- Description textuelle du cas d'utilisation "Archiver version antérieur"

Cas d'utilisation	Archiver version antérieur
Acteur	Utilisateur
Pré-condition	L'utilisateur doit être authentifié et appartient au groupe Superviseur
Scénario nominal	-Quand le Superviseur modifie un document, l'ancienne version est conservée dans un interface historique
Post-condition	Document archivé

Tableau 3.15 : Description textuelle du cas d'utilisation "Archiver version antérieur"

- Description textuelle du cas d'utilisation "Commenter document"

Cas d'utilisation	Commenter document
Acteur	Utilisateur
Pré-condition	L'utilisateur doit être authentifié
Scénario nominal	-L'utilisateur sélectionne un document -L'utilisateur commente le document .
Post-condition	Document commenté

Tableau 3.16 : Description textuelle du cas d'utilisation "Commenter document"

6.2 Diagramme d'activité

Nous avons mentionné dans les besoins fonctionnels que la création d'un document nécessite un processus bien déterminé. Par conséquent, nous avons besoin d'un diagramme d'activité pour mieux comprendre ce processus.

La figure 3.13 représente le diagramme de d'activité de création d'un document

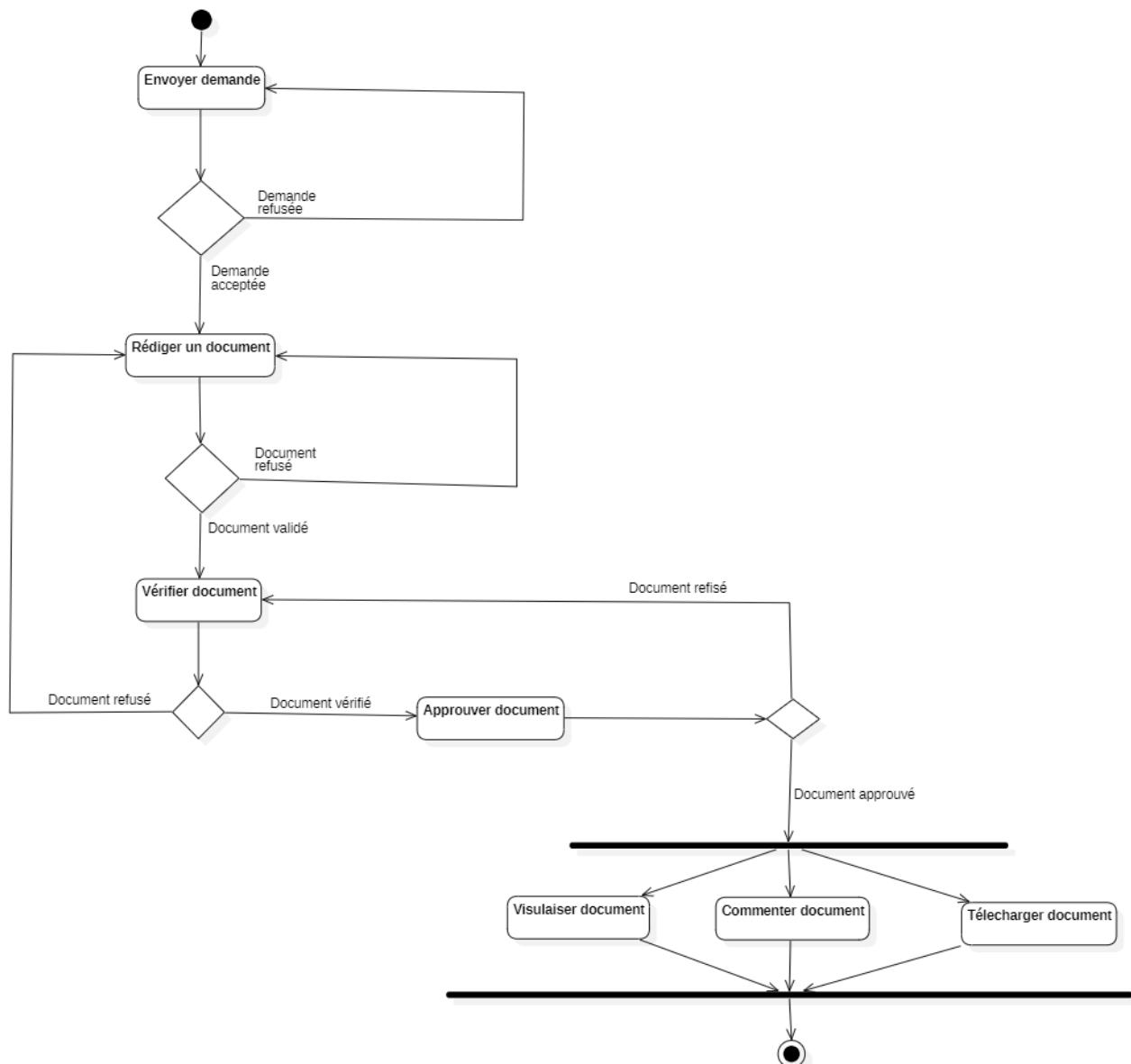


Figure 3.13 : Diagramme d'activité de création d'un document

6.3 Diagrammes de classes

La figure ci dessous représente le diagramme de classe de module Documentation

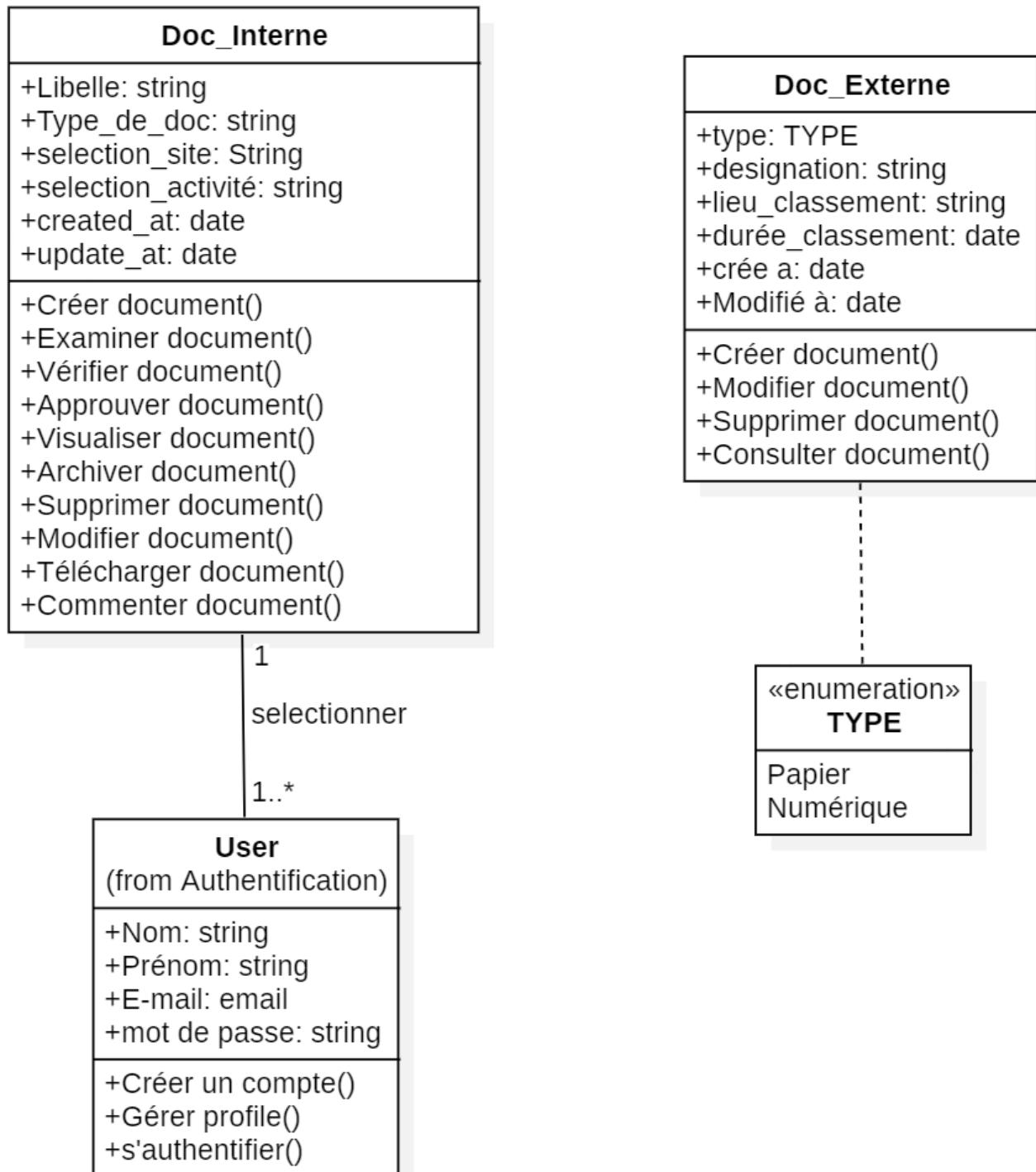


Figure 3.14 : Diagramme de classe de Documentation

6.4 Diagrammes de séquence

La figure ci dessous représente le diagramme de séquence de Crée document

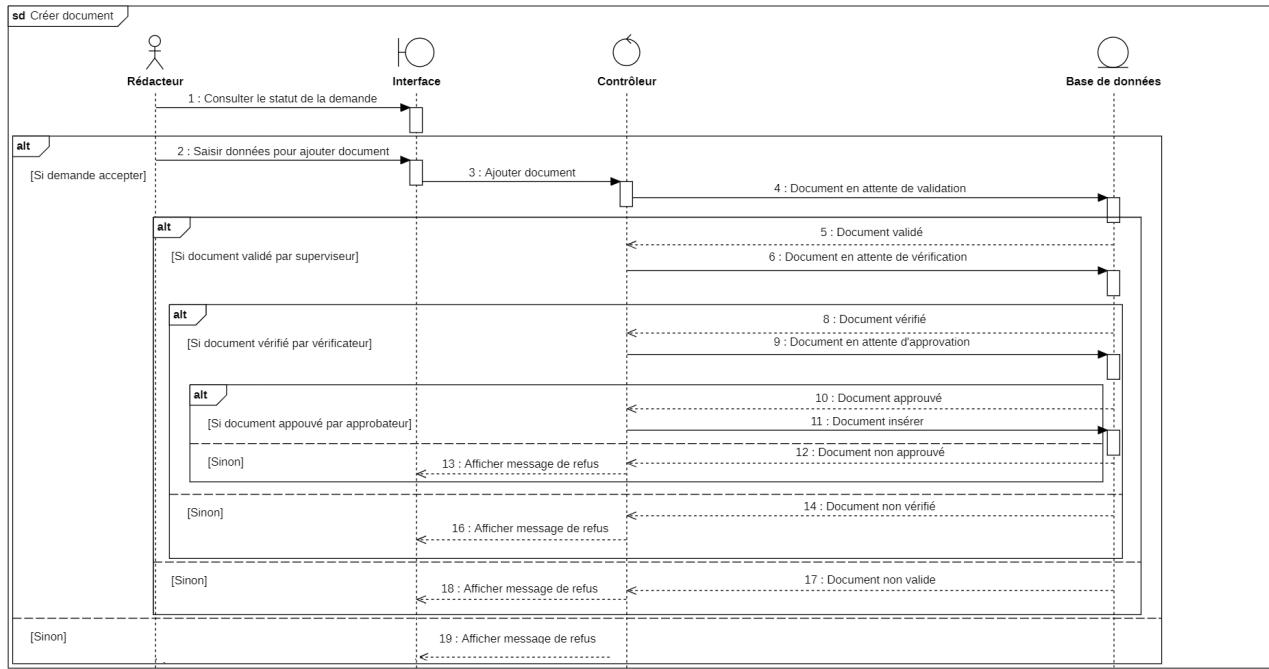


Figure 3.15 : Diagramme de séquence de Crée document

7 Réalisation

Cette section présente les images des interfaces utilisateurs offrant un visuel des fonctionnalités de ce sprint. Ces illustrations mettent en évidence l'ergonomie et la convivialité de l'application web.

La figure 3.16 présente l'interface page d'accueil

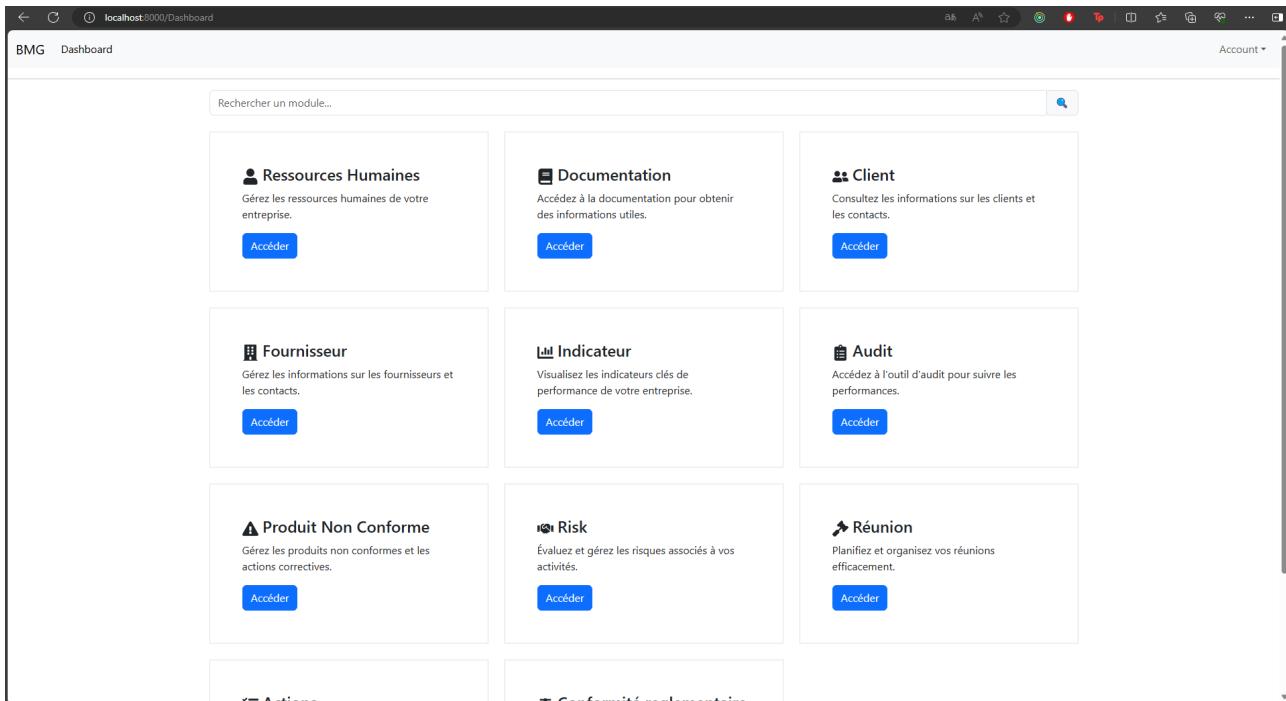


Figure 3.16 : Interface de page d'accueil

La figure 3.17 présente l'interface profile d'utilisateur

Profil Utilisateur

E-mail:	user@user.com
Prénom:	user prenom
Nom:	user nom

Notifications

Expéditeur:	user prenom user nom
Message:	demande numéro 4 validé
Reçu le:	2024-06-01 17:24:21

X

Demandes

Demande N°:	4
Type:	Procédure
Objet du document:	TEST
Statut:	Validé
Créé le:	2024-06-01 17:23:54

Modifier Profil

Figure 3.17 : Interface Profile

La figure 3.18 présente l'interface créer compte

The screenshot shows a registration form titled "Inscrivez-vous". It contains the following fields:

- Nom:
- Prénom:
- Email:
- Mot de passe:
- Confirmez Mot de passe:

S'inscrire

Vous avez déjà un compte ? [Se connecter](#)

Figure 3.18 : Interface Créer compte

La figure 3.19 présente présente l'interface "consulter fiche employé"

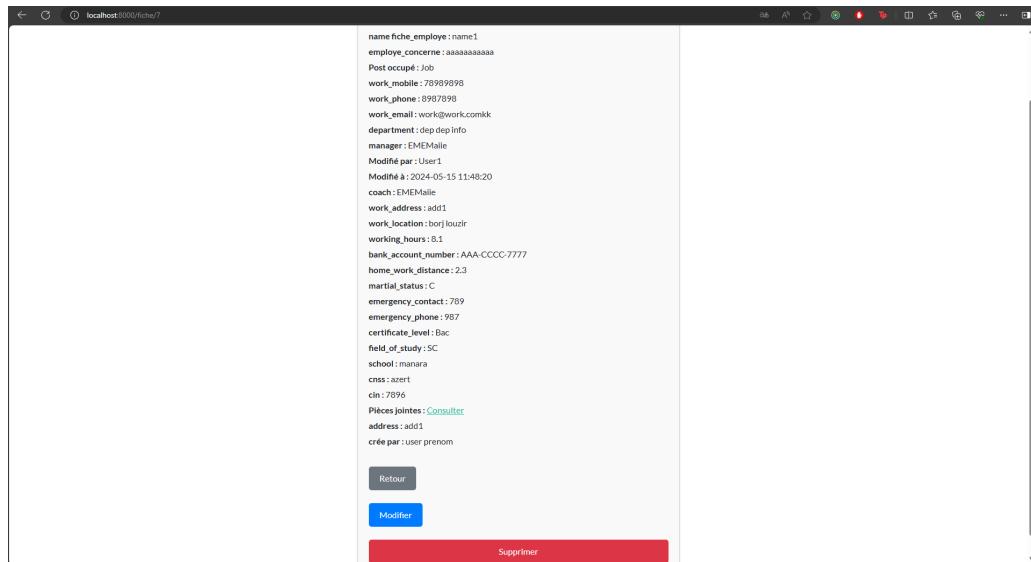


Figure 3.19 : Interface Consulter fiche employé

La figure 3.20 présente l'interface "valider/refuser demande"

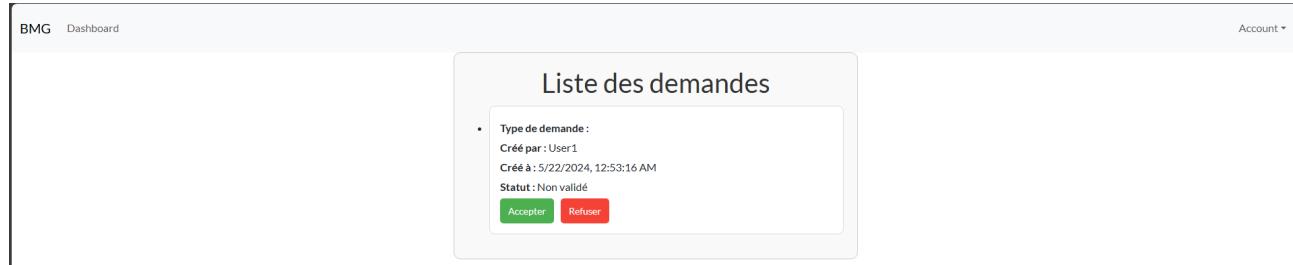


Figure 3.20 : Interface 'valider/refuser demande'

8 Test

La partie test des APIs a été réalisé par l'outil Postman.

Cette interface permet de tester la méthode créer compte

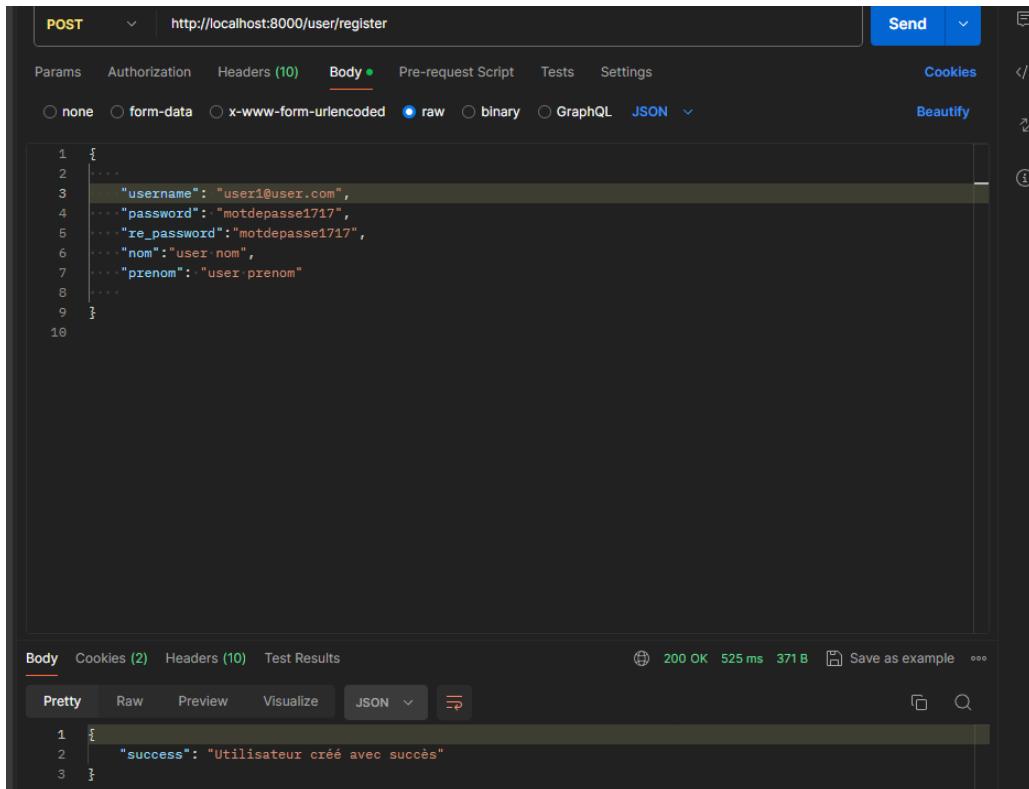


Figure 3.21 : Test méthode créer compte

Cette interface permet de tester la méthode "s'authentifier"

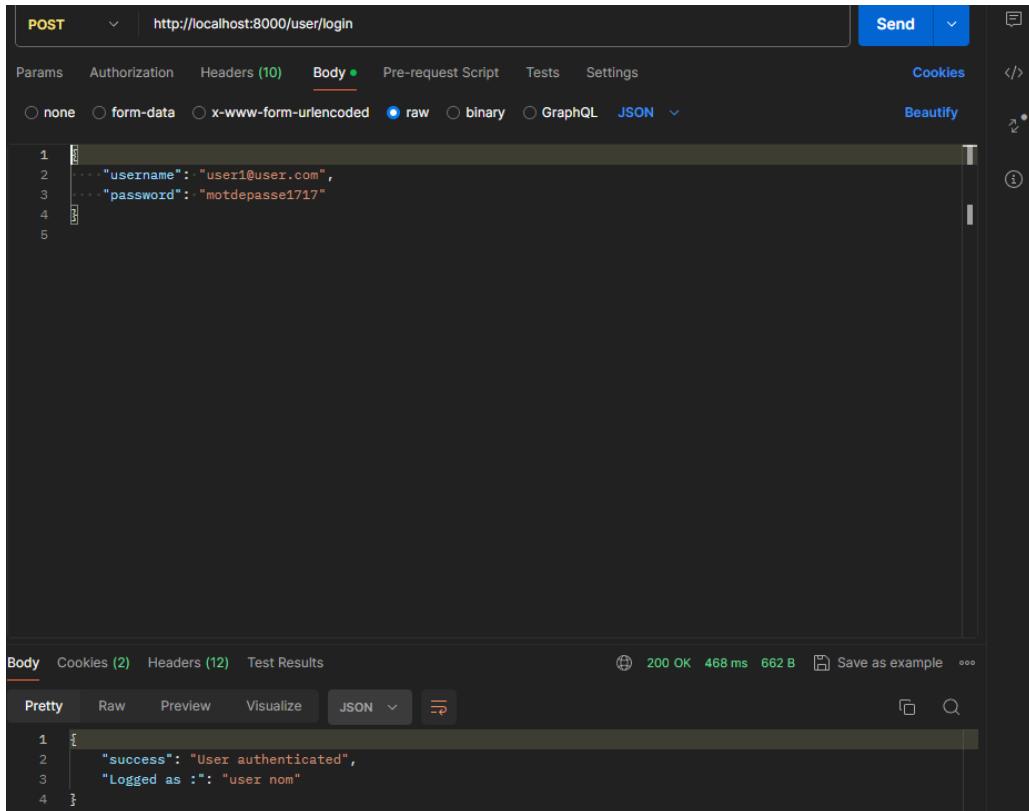


Figure 3.22 : Test méthode "s'authentifier"

Cette interface permet de tester la méthode "ajouter employé"

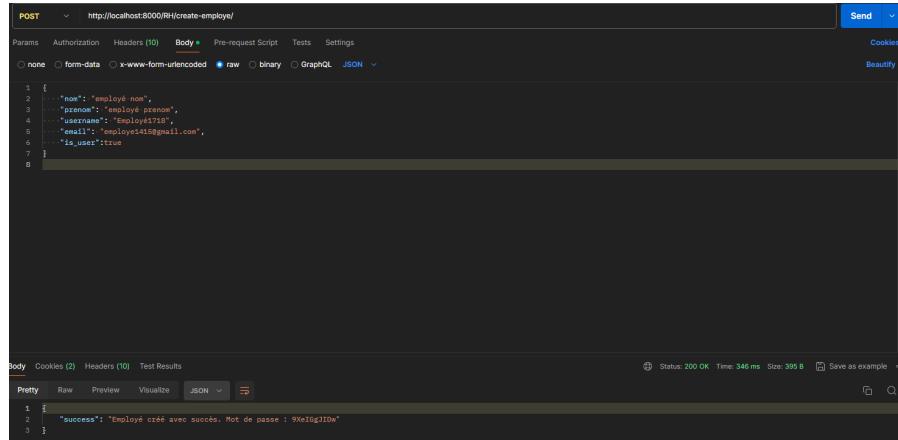


Figure 3.23 : Test méthode "ajouter employé"

Cette interface permet de tester la méthode "valider une demande"

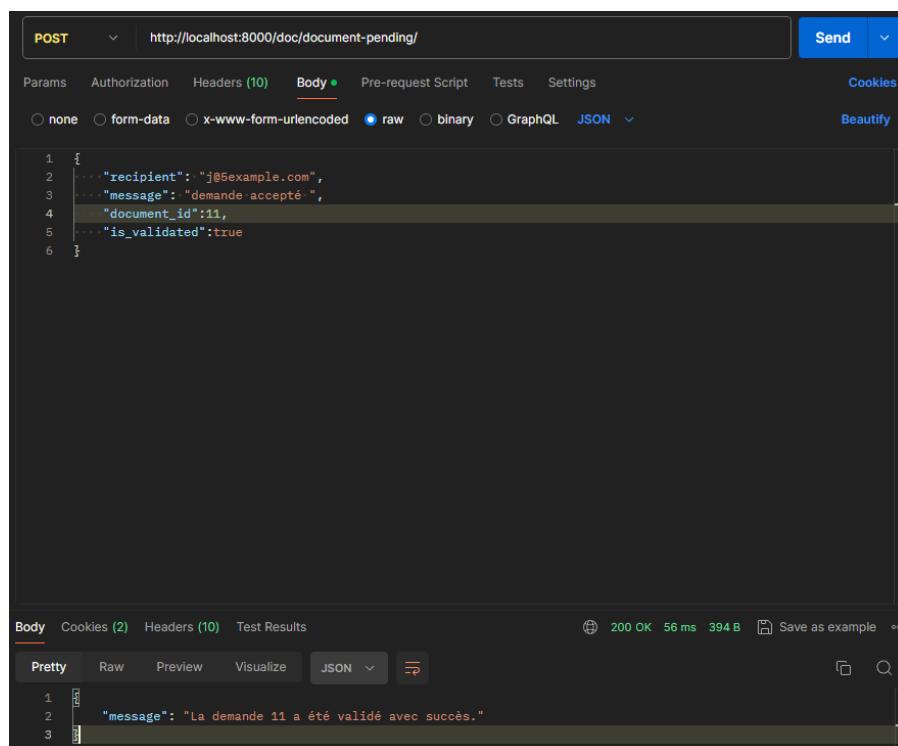


Figure 3.24 : Test méthode "valider une demande"

9 Conclusion

Dans ce chapitre nous avons mis l'accent sur la réalisation du premier sprint où nous avons présenté les cas d'utilisations, détaillé la conception, les tests des API et en intégrant les interfaces de plateforme.

Dans le chapitre qui suit, nous allons présenter le deuxième sprint couvrant les fonctionnalités suivantes : Gestion des clients , Fournisseurs et indicateurs.

Sprint 2 : Gestion des clients , fournisseurs et indicateurs

Plan

1	Introduction	51
2	Backlog du sprint	51
3	Besoins fonctionnels	52
4	Mise en place du module client	52
5	Mise en place de module Fournisseur	56
6	Mise en place de module Indicateur	59
7	Réalisation	62
8	Test	64
9	conclusion	65

1 Introduction

Notre deuxième sprint traitera la gestion des clients, des fournisseurs et des indicateurs, pendant ce sprint nous allons suivre les mêmes étapes du premier sprint.

2 Backlog du sprint

Le tableau ci dessous représente le Backlog du sprint 2

Module	User Story	Taches	Estimation
Client	- En tant que utilisateur je peux ajouter un client	-Conception et création de l'interface -Implémentation de code	1
	- En tant que utilisateur je peux ajouter une réclamation client.	-Test de l'interface	
	- En tant que utilisateur je peux créer une enquête satisfaction client	- Réalisation	
	-En tant que utilisateur je peux ajouter une suggestion client	-Test des fonctionnalités	
Fournisseur	- En tant que utilisateur je peux ajouter un fournisseur	-Implémentation des fonctionnalités	1
	- En tant que utilisateur je peux évaluer un fournisseur	-Réalisation -Test	
	-En tant que utilisateur je peux créer une réclamation pour un fournisseur .	-Mise en place de l'interface	
Indicateur	- En tant que utilisateur je peux ajouter un indicateur	-Implémentation des fonctionnalités	1
	-En tant que utilisateur je peux suivre un indicateur	-Test des fonctionnalités -Réalisation	
	- En tant que utilisateur je peux modifier ou supprimer un indicateur	-Mise en place de l'interface	

Tableau 4.1 : Backlog Sprint 2

3 Besoins fonctionnels

Notre application doit assurer plusieurs fonctionnalités , notamment ;

- Gérer client : Cette fonctionnalité offre aux utilisateurs la possibilité d'introduire un client, modifier un client ou le supprimer
- Créer réclamation client : Cette fonctionnalité offre aux utilisateurs la possibilité de créer une réclamation d'un client
- Créer enquête de satisfaction d'un client : Cette fonctionnalité offre aux utilisateurs la possibilité de créer une enquête de satisfaction pour un client.
- Créer suggestion client : Cette fonctionnalité offre aux utilisateurs la possibilité d'ajouter une suggestion pour un client
- Gérer fournisseur : Cette fonctionnalité offre aux utilisateurs la possibilité d'introduire un fournisseur, modifier un fournisseur ou le supprimer
- Évaluer fournisseur : Cette fonctionnalité offre aux utilisateurs la possibilité d'évaluer un fournisseur
- Créer réclamation fournisseur : Cette fonctionnalité offre aux utilisateurs la possibilité de créer une réclamation d'un fournisseur
- Gérer indicateur : Cette fonctionnalité offre aux utilisateurs la possibilité d'ajouter un indicateur, modifier un indicateur ou le supprimer
- suivre indicateur : Cette fonctionnalité offre aux utilisateurs la possibilité de suivre un indicateur

4 Mise en place du module client

Dans cette section,nous allons représenté les diagrammes de classe, de cas d'utilisation et de séquence pour le module client

4.1 Diagramme de classes

La figure 4.1 représente le diagramme de classe du module client

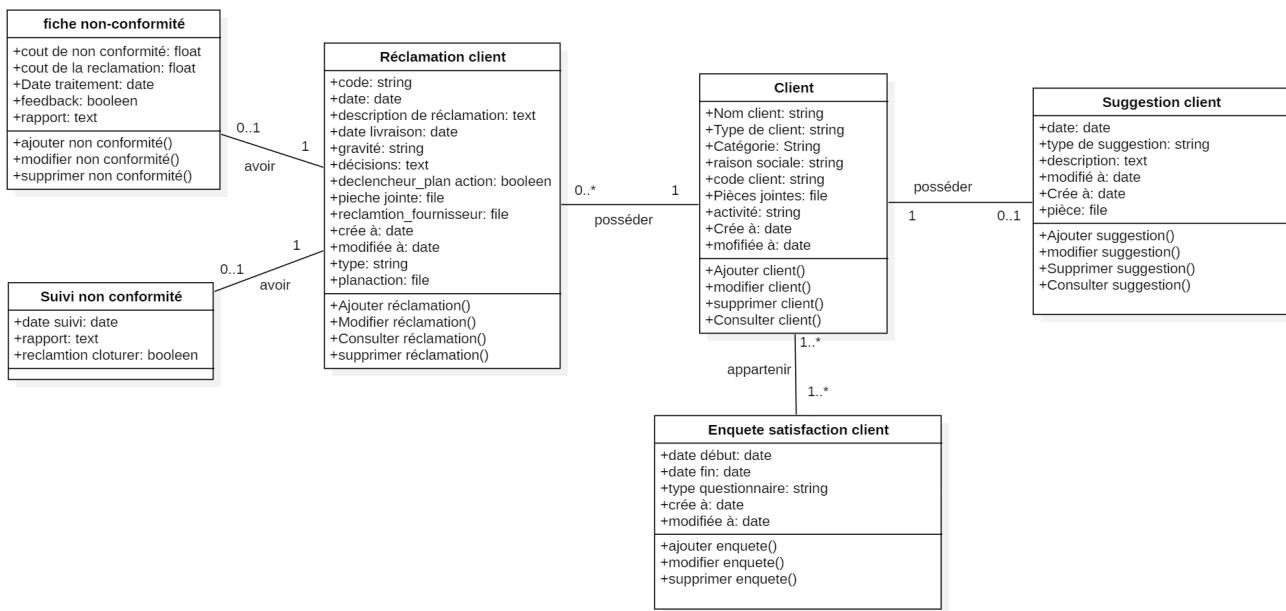


Figure 4.1 : Diagramme de classes de Client

4.2 Diagramme de cas d'utilisation

La figure 4.2 représente le diagramme de cas d'utilisation du module client

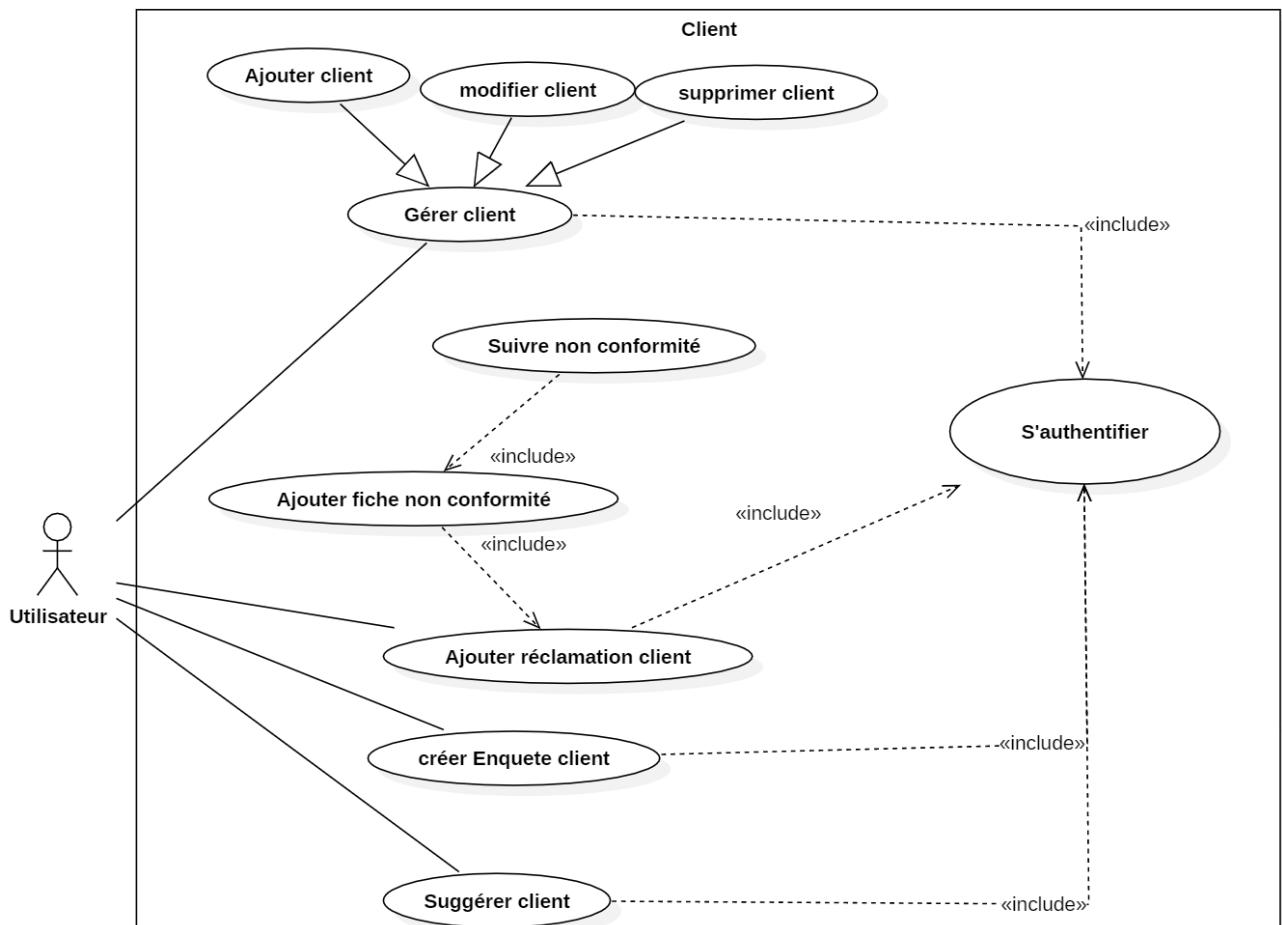


Figure 4.2 : Diagramme de cas d'utilisation de Client

4.2.1 Description textuelle des cas d'utilisation

- Description textuelle du cas d'utilisation "Ajouter client"

Cas d'utilisation	Ajouter client
Acteur	Utilisateur
Pré-condition	L'utilisateur doit être authentifié
Scénario nominal	-L'utilisateur remplit une fiche d'ajout -L'utilisateur valide les informations
Post-condition	L'ajout d'un nouveau client

Tableau 4.2 : Description textuelle du cas d'utilisation "Ajouter client"

- Description textuelle du cas d'utilisation "Ajouter réclamation client"

Cas d'utilisation	Ajouter réclamation client
Acteur	Utilisateur
Pré-condition	L'utilisateur doit être authentifié
Scénario nominal	-L'utilisateur remplit les champs du formulaire -L'utilisateur valide les données
Post-condition	Réclamation ajouté
Extension	Si le champ déclencheur plan action est sélectionné lors de l'ajout , une fiche de non conformité va être ajouter

Tableau 4.3 : Description textuelle du cas d'utilisation "Ajouter réclamation client"

- Description textuelle du cas d'utilisation "Créer enquête client"

Cas d'utilisation	Créer enquête client
Acteur	Utilisateur
Scénario nominal	-L'utilisateur accède à l'interface de l'ajout -l'utilisateur remplit les champs du formulaire
Post-condition	Enquête ajouté -L'enquête est envoyée par mail aux clients concernés

Tableau 4.4 : Description textuelle du cas d'utilisation "Créer enquête client"

- Description textuelle du cas d'utilisation "Suivre non conformité"

Cas d'utilisation	Suivre non conformité
Acteur	Responsable suivi
Pré-condition	-L'utilisateur doit être authentifié , appartient au groupe responsable suivi et fiche de non conformité créée
Scénario nominal	-Le responsable suivi rempli une fiche de suivi
Post-condition	Non conformité suivi

Tableau 4.5 : Description textuelle du cas d'utilisation "Suivre non conformité"

- Description textuelle du cas d'utilisation "Suggérer client"

Cas d'utilisation	Suggérer client
Acteurs	Utilisateur
Pré-condition	L'utilisateur doit être authentifié.
Scénario nominal	-L'utilisateur remplit une formulaire de suggestion en sélectionnant le client concerné,type de suggestion
Post-condition	Suggestion sauvegardée .

Tableau 4.6 : Description textuelle du cas d'utilisation "Suggérer client"

4.3 Diagrammes de séquence

La figure 4.3 représente le diagramme de séquence de "ajouter client"

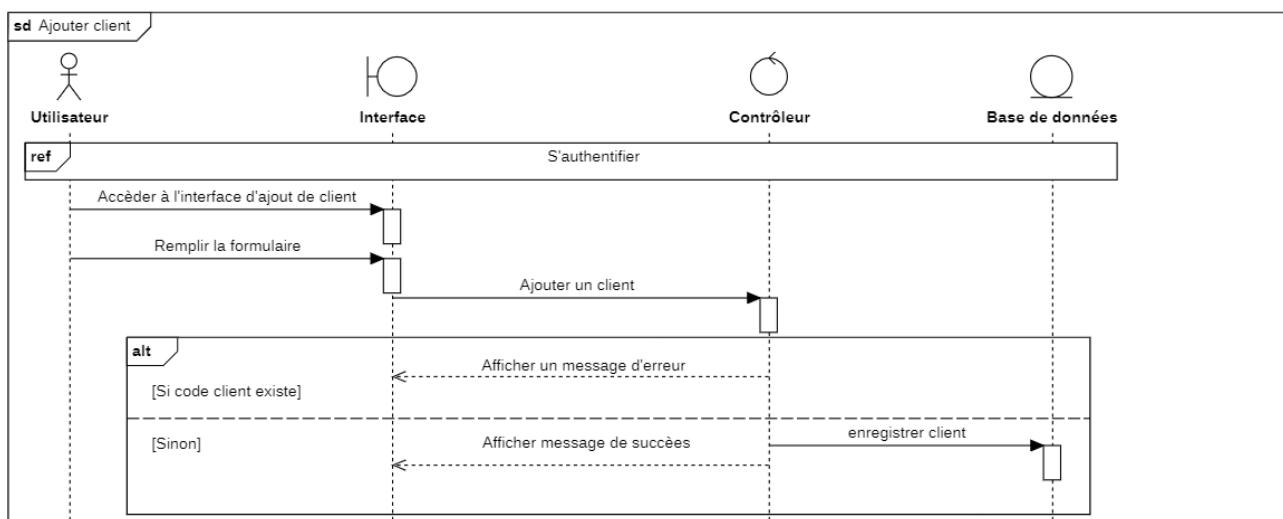


Figure 4.3 : Diagramme de séquence de "ajouter client"

La figure 4.4 représente le diagramme de séquence de "Créer réclamation client"

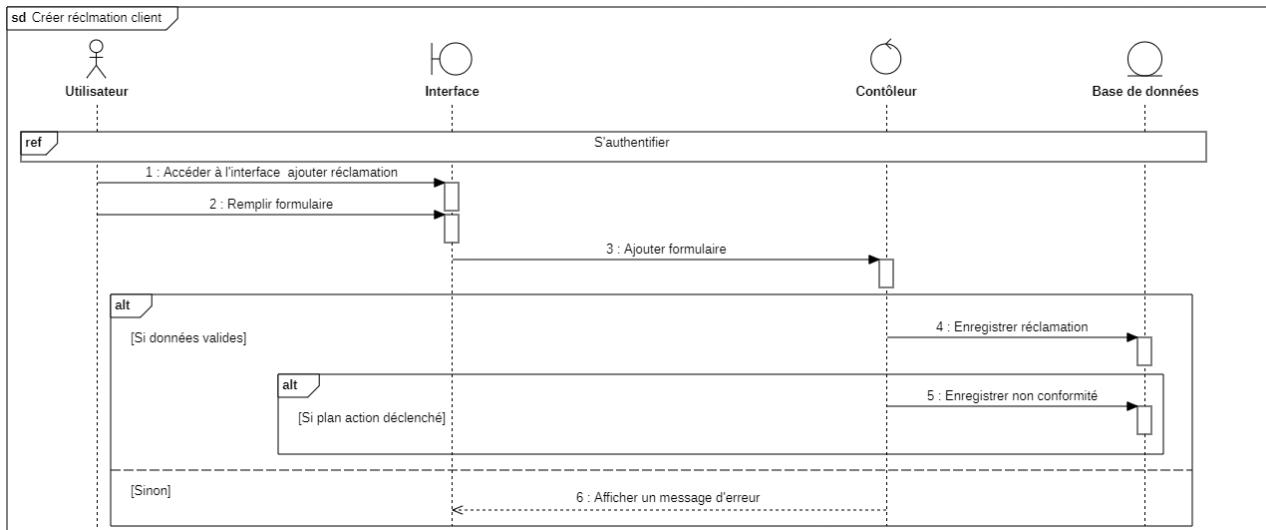


Figure 4.4 : Diagramme de séquence de "Créer réclamation client"

La figure 4.5 représente le diagramme de séquence de "suggérer client"

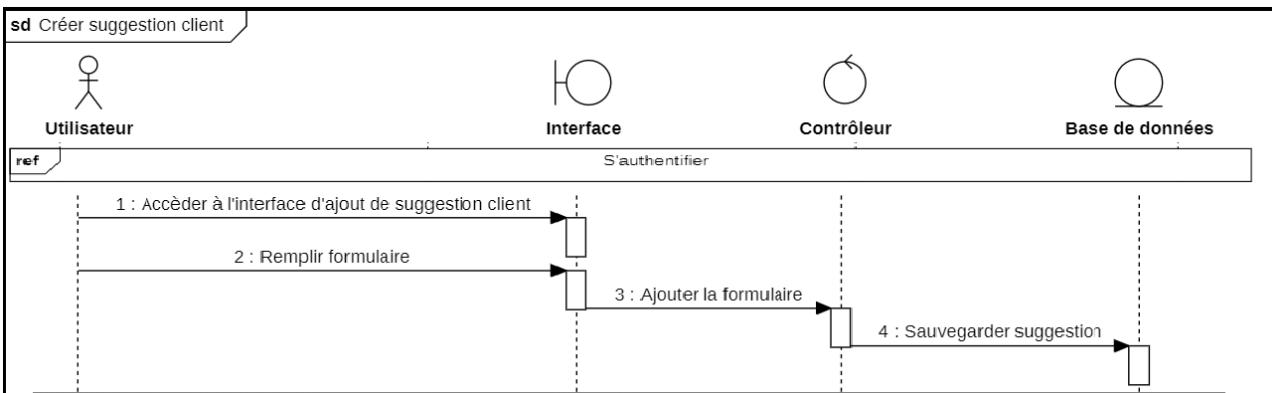


Figure 4.5 : Diagramme de séquence de "suggérer client"

5 Mise en place de module Fournisseur

Dans cette section, nous allons représenter les diagrammes de classe, cas d'utilisation et de séquence pour le module fournisseur

5.1 Diagramme de classes

La figure 4.6 représente le diagramme de classes du module fournisseur

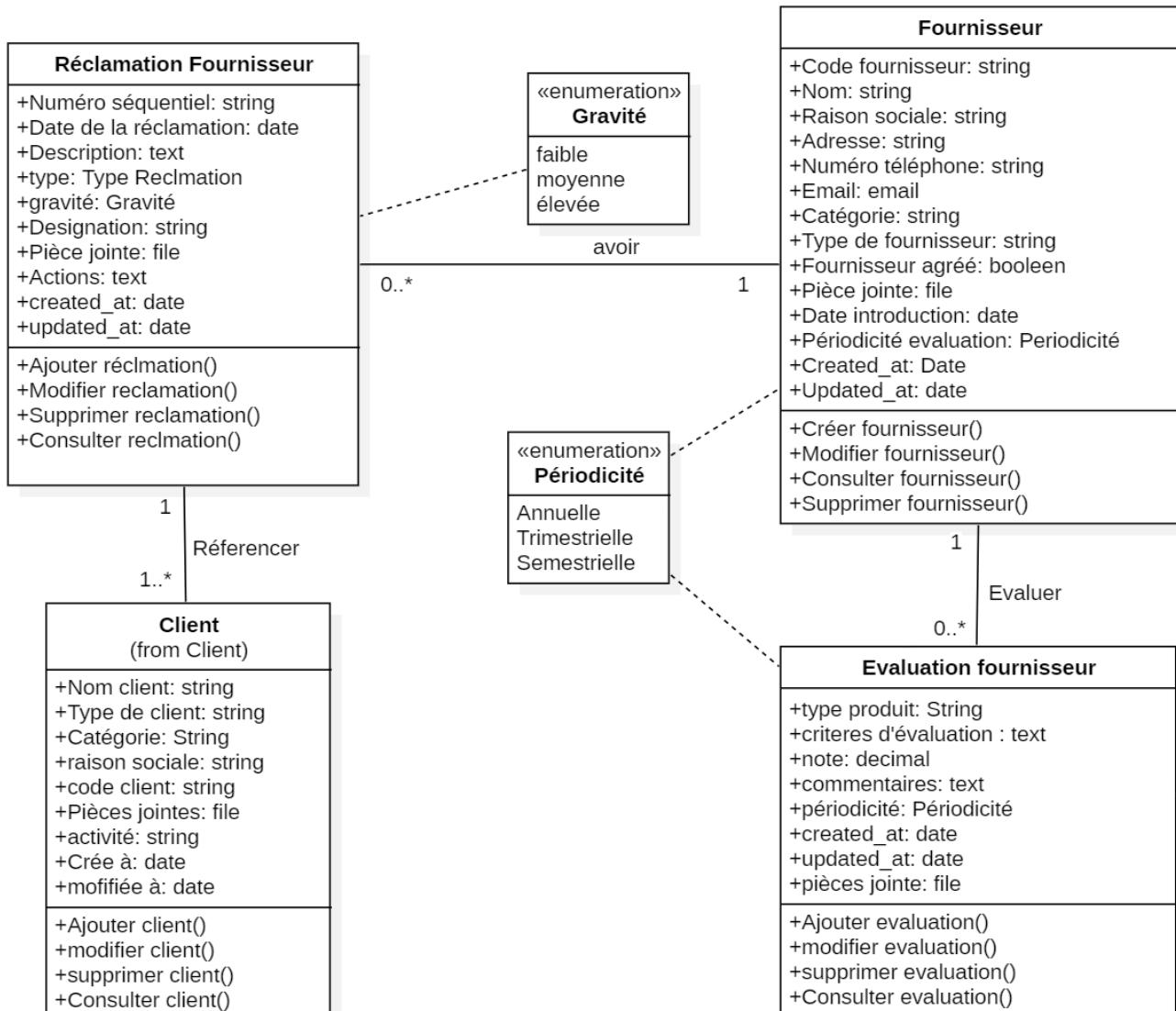


Figure 4.6 : Diagramme de classes du module Fournisseur

5.2 Diagramme de cas d'utilisation

La figure 4.7 représente le diagramme de cas d'utilisation du module fournisseur

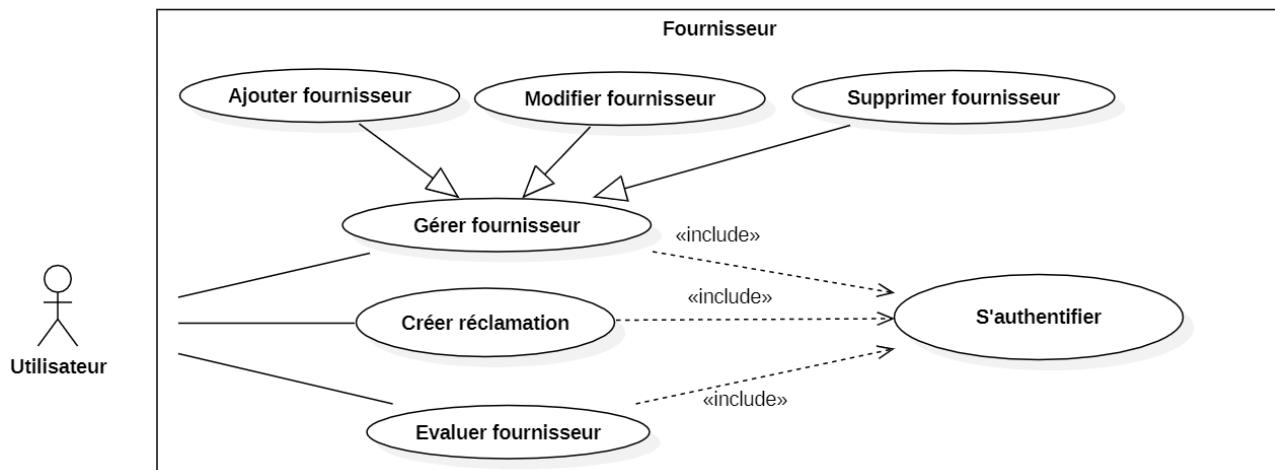


Figure 4.7 : Diagramme de cas d'utilisation de Fournisseur

5.2.1 Raffinement des cas d'utilisation

- Description textuelle du cas d'utilisation "Ajouter fournisseur"

Cas d'utilisation	Ajouter fournisseur
Acteur	Utilisateur
Pré-condition	L'utilisateur doit être authentifié
Scénario nominal	-L'utilisateur remplit une fiche d'ajout -L'utilisateur valide les informations
Post-condition	L'ajout d'une nouveau fournisseur

Tableau 4.7 : Description textuelle du cas d'utilisation "Ajouter fournisseur"

- Description textuelle du cas d'utilisation "Créer réclamation"

Cas d'utilisation	Créer réclamation
Acteurs	Utilisateur
Pré-condition	L'utilisateur doit être authentifié
Scénario nominal	-L'utilisateur remplit les champs du formulaire -L'utilisateur valide les données
Post-condition	Réclamation ajouté

Tableau 4.8 : Description textuelle du cas d'utilisation "Créer réclamation"

- Description textuelle du cas d'utilisation "Évaluer fournisseur"

Cas d'utilisation	Évaluer fournisseur
Acteurs	Utilisateur
Scénario nominal	-L'utilisateur sélectionne un fournisseur -l'utilisateur remplit les champs du formulaire tel que (critères, commentaires,notes...)
Post-condition	fournisseur évalué

Tableau 4.9 : Description textuelle du cas d'utilisation "Évaluer fournisseur"

5.3 Diagrammes de séquence

La figure représente le diagramme de séquence de "Créer réclamation fournisseur"

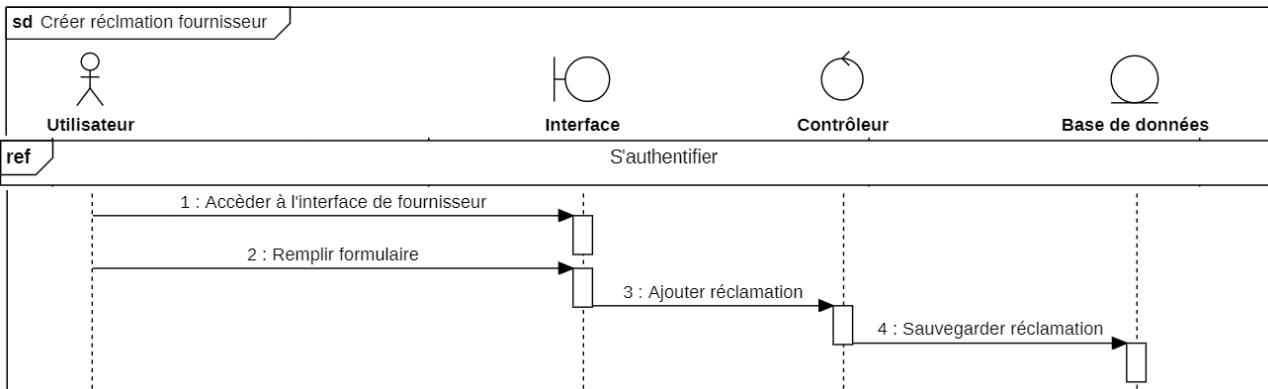


Figure 4.8 : Diagramme de séquence de "Créer réclamation fournisseur"

La figure représente le diagramme de séquence de "Évaluer fournisseur"

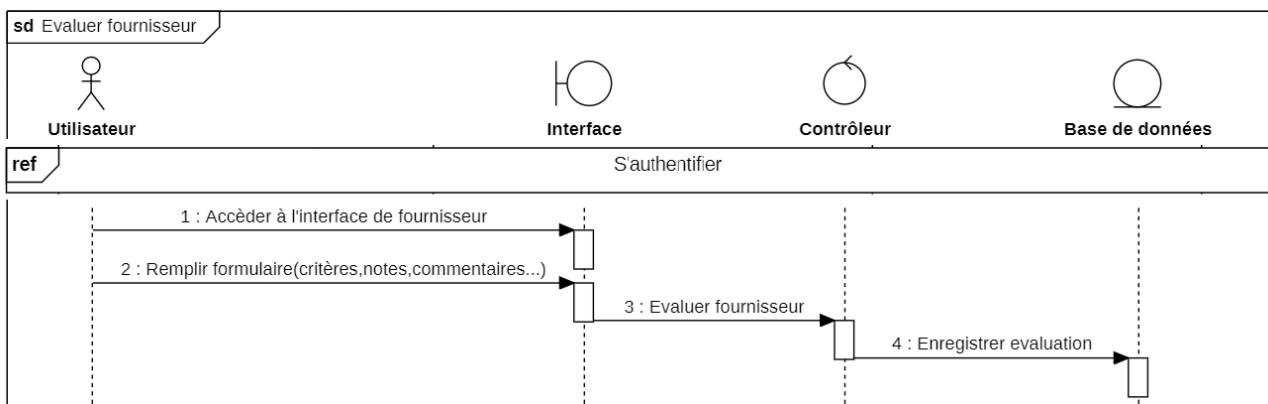


Figure 4.9 : Diagramme de séquence de "Évaluer fournisseur"

6 Mise en place de module Indicateur

Dans cette section,nous allons représenté les diagrammes de classes , cas d'utilisation et de séquence pour le module indicateur

6.1 Diagramme de classes

La figure 4.10 représente le diagramme de classes du module indicateur

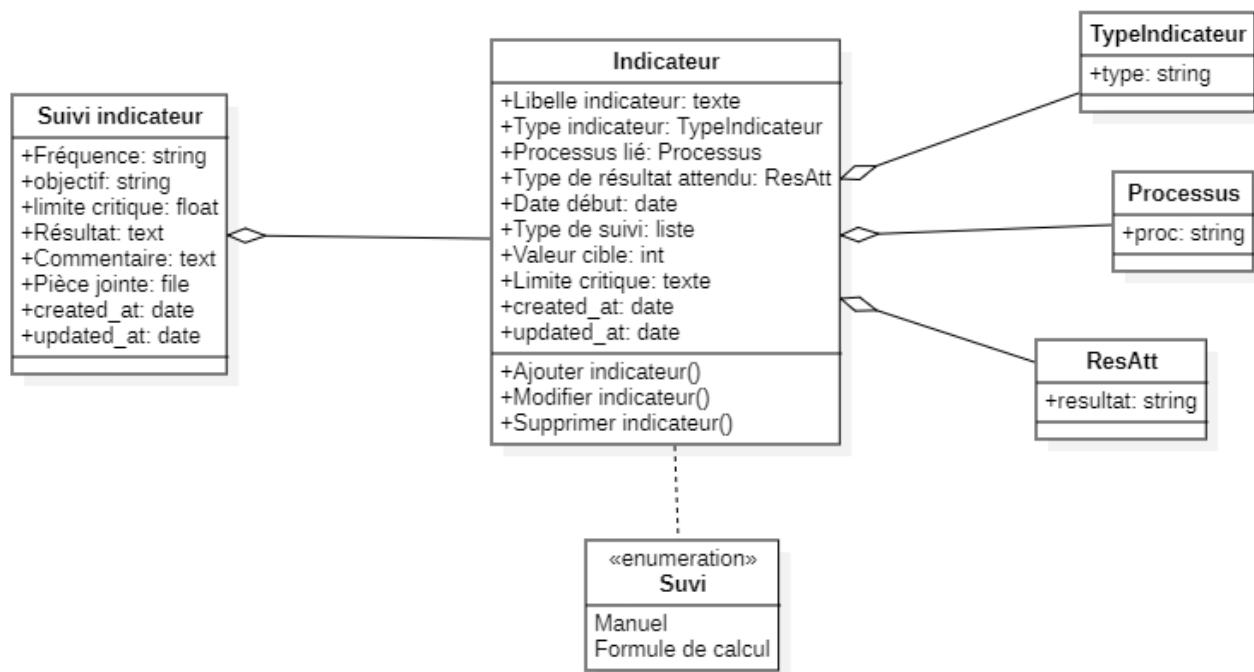


Figure 4.10 : Diagramme de classe d'indicateur

6.2 Diagramme de cas d'utilisation

La figure 4.11 représente le diagramme de cas d'utilisation du module indicateur

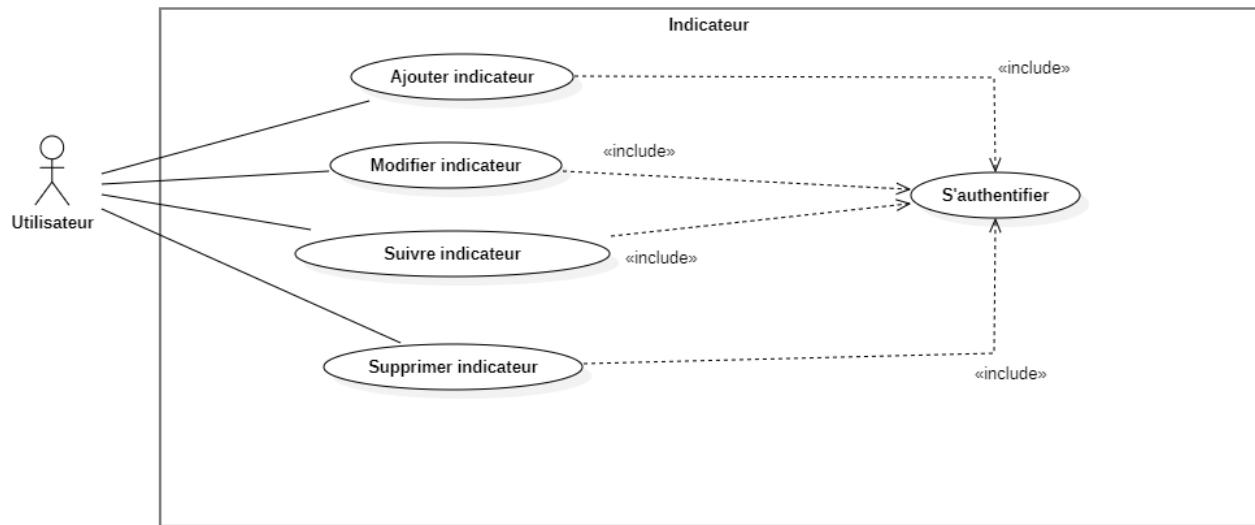


Figure 4.11 : Diagramme de cas d'utilisation d'indicateur

6.2.1 Raffinement des cas d'utilisation

- Description textuelle du cas d'utilisation "Ajouter indicateur"

Cas d'utilisation	Ajouter fournisseur
Acteur	Utilisateur

Pré-condition	L'utilisateur doit être authentifié
Scénario nominal	-L'utilisateur remplit une fiche d'ajout -L'utilisateur valide les informations
Post-condition	L'ajout d'une nouveau indicateur

Tableau 4.10 : Description textuelle du cas d'utilisation "Ajouter fournisseur"

- Description textuelle du cas d'utilisation "Modifier indicateur"

Cas d'utilisation	Modifier indicateur
Acteurs	Utilisateur
Pré-condition	L'utilisateur doit être authentifié et l'indicateur existe
Scénario nominal	-L'utilisateur sélectionne l'indicateur -L'utilisateur modifie les champs de l'indicateur
Post-condition	Indicateur modifié

Tableau 4.11 : Description textuelle du cas d'utilisation "Modifier indicateur"

- Description textuelle du cas d'utilisation "Supprimer indicateur"

Cas d'utilisation	Supprimer indicateur
Acteur	Utilisateur
Pré-condition	L'utilisateur doit être authentifié, appartient au groupe responsable indicateur et l'indicateur existe
Scénario nominal	-L'utilisateur sélectionne l'indicateur -L'utilisateur supprime l'indicateur
Post-condition	Indicateur supprimé

Tableau 4.12 : Description textuelle du cas d'utilisation "Modifier indicateur"

- Description textuelle du cas d'utilisation "Suivre indicateur"

Cas d'utilisation	Suivre Indicateur
Acteur	Utilisateur

Scénario nominal	<ul style="list-style-type: none"> -L'utilisateur sélectionne l'indicateur -Si le suivi est manuel,l'utilisateur écrit la formule à suivre -L'utilisateur ajoute les informations de suivi -Si le suivi par formule ,l'utilisateur sélectionne des formules existants
Post-condition	Indicateur suivi

Tableau 4.13 : Description textuelle du cas d'utilisation "Suivre indicateur"

6.3 Diagrammes de séquence

La figure 4.12 représente le diagramme de séquence de "suivre indicateur"

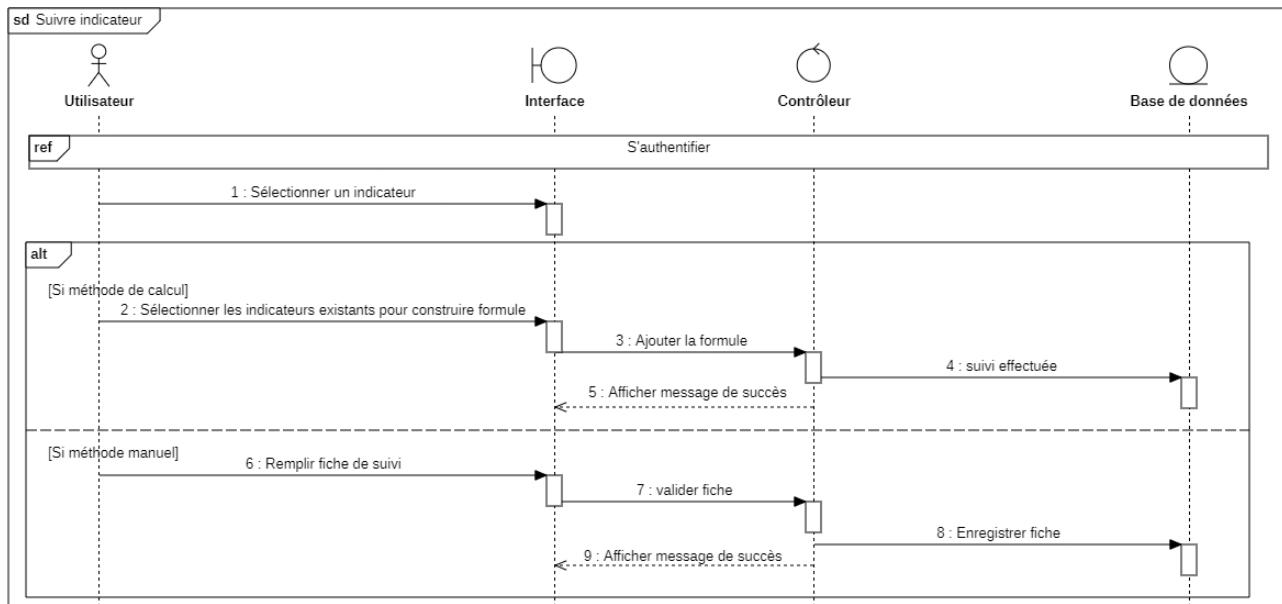


Figure 4.12 : Diagramme de séquence de "suivre indicateur"

7 Réalisation

La figure 4.13 présente l'interface ajouter réclamation client

Ajouter Réclamation

Code:

Description:

Décisions:

Type Réclamation: Sélectionner...

Gravité : Sélectionner...

Date de Livraison : mm/dd/yyyy

Responsable traitement: Sélectionner...

Fichier Réclamation fournisseur: Choose File No file chosen

Fichier plan action : Choose File No file chosen

Pièces jointes : Choose File No file chosen

Déclencher plan action :

Figure 4.13 : Interface ajouter réclamation client

La figure 4.14 présente l'interface enquête client

Liste des Enquêtes

Name Enquete: Name Date debut: 2024-06-01 Date fin: 2024-06-10 Type questionnaire: Satisfaction les clients: Nom Client Pièces jointes : null Crée à: 2024-05-31 12:06:59 Crée par: user prenom Modifié à: Pas de modification Modifié par: Pas de modification Modifier Supprimer	Name Enquete: Name2 client3 Date debut: 2024-06-01 Date fin: 2024-06-10 Type questionnaire: Satisfaction les clients: Nom Client Pièces jointes : null Crée à: 2024-05-31 12:14:04 Crée par: user prenom Modifié à: Pas de modification Modifié par: Pas de modification Modifier Supprimer	Name Enquete: Name name Date debut: 2024-06-01 Date fin: 2024-06-10 Type questionnaire: Satisfaction les clients: Nom Client Pièces jointes : null Crée à: 2024-05-31 13:10:38 Crée par: user prenom Modifié à: Pas de modification Modifié par: Pas de modification Modifier Supprimer
--	---	---

[Ajouter Enquête](#)

[Retour](#)

Figure 4.14 : Interface enquête client

La figure 4.15 présente l'interface évaluer fournisseur

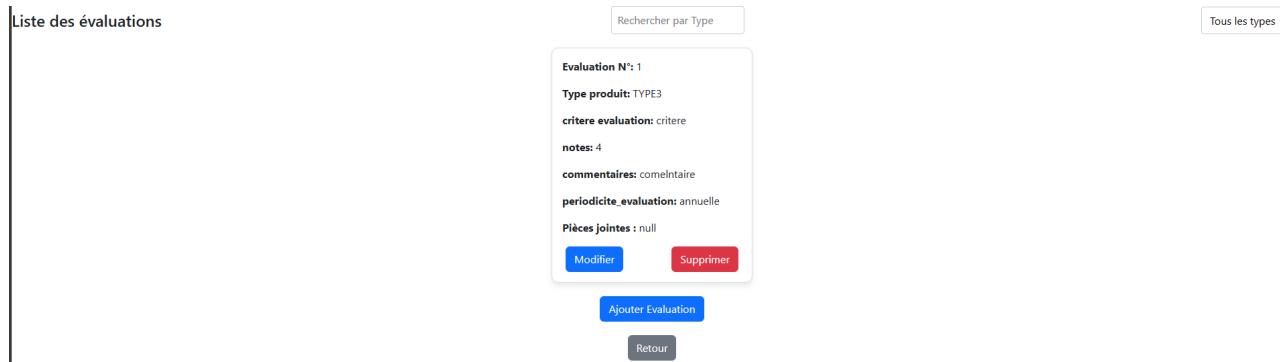


Figure 4.15 : Interface évaluer fournisseur

8 Test

Cette interface permet de tester la méthode de "ajouter réclamation client"

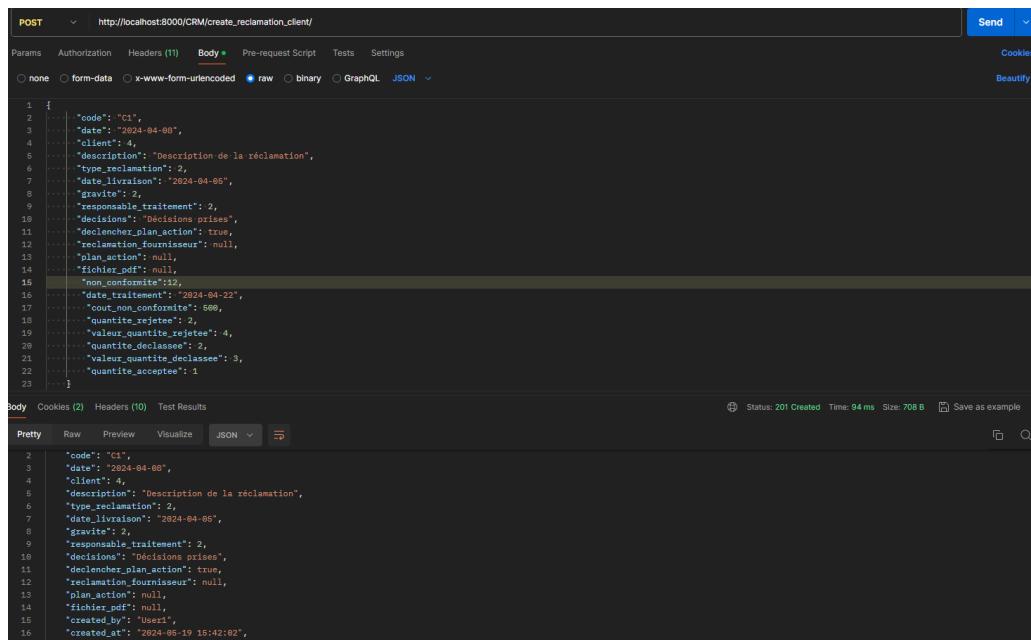


Figure 4.16 : Test méthode "ajouter réclamation client"

Cette interface permet de tester la méthode de "Évaluer fournisseur"

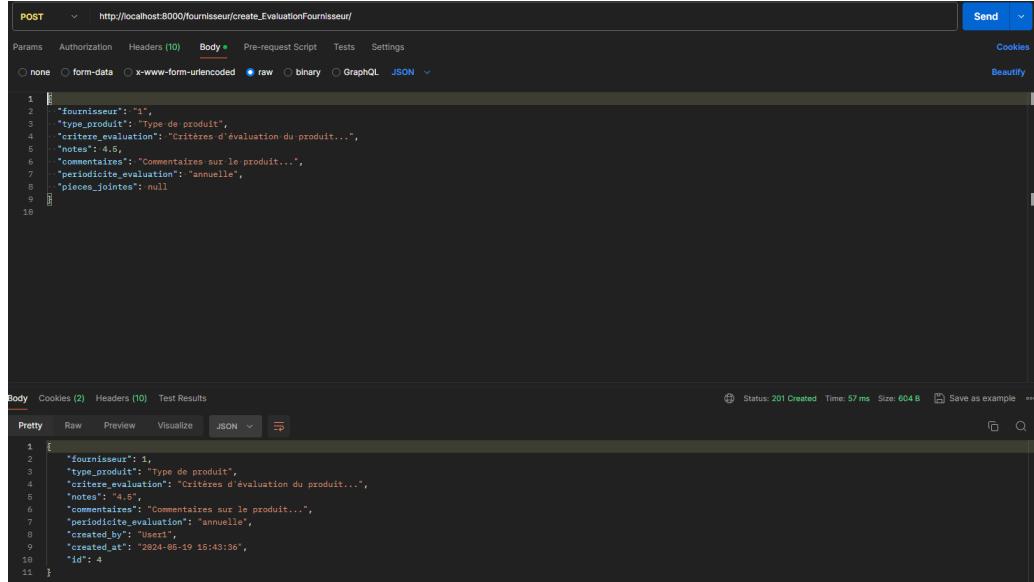


Figure 4.17 : Test méthode "Évaluer fournisseur"

Cette interface permet de tester la méthode de "Suivre indicateur"

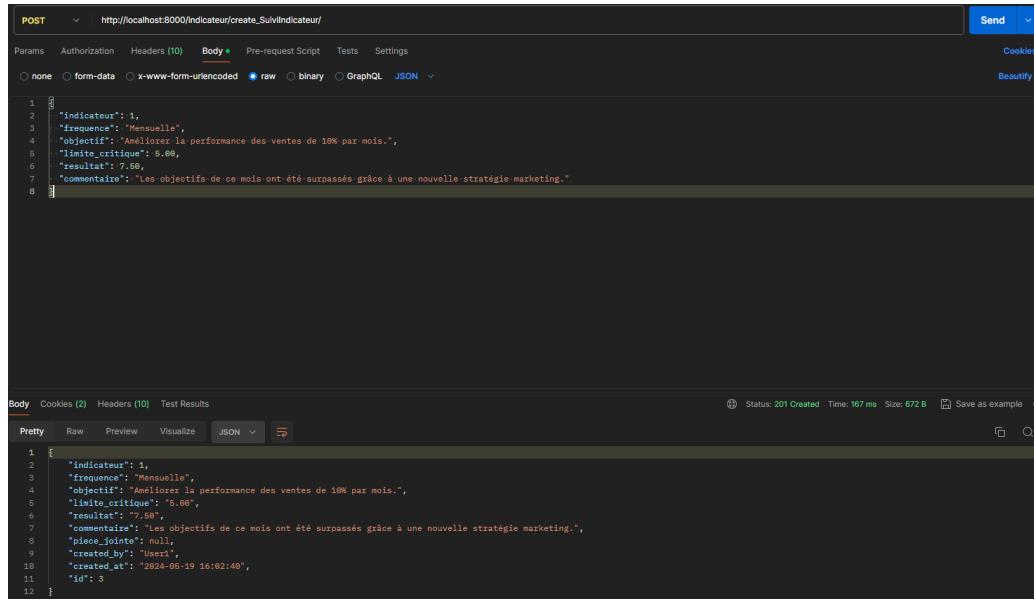


Figure 4.18 : Test méthode "Suivre indicateur"

9 conclusion

En conclusion de ce sprint, nous avons réalisé des avancées significatives dans le développement de notre projet, en mettant en place des fonctionnalités clés telles que la gestion des clients, la gestion des fournisseurs et la gestion des indicateurs. Dans le prochain sprint nous nous concentrerons sur la gestion des audits , les actions et la planification des réunions.

Sprint 3 : Gestion des audits , des actions et la planification des réunions.

Plan

1	Introduction	67
2	Backlog du sprint	67
3	Besoins fonctionnels	67
4	Mise en place du module Audit	68
5	Mise en place du module réunion	71
6	Mise en place de module actions	75
7	Réalisation	79
8	Test	79
9	conclusion	81

1 Introduction

Notre troisième sprint traitera la gestion des audits, des actions et la planification des réunions, pendant ce sprint nous allons suivre les mêmes étapes du premier et deuxième sprint.

2 Backlog du sprint

Le tableau ci dessous représente le Backlog du sprint 2

Module	User Story	Taches	Estimation
Audit	- En tant que utilisateur je peux planifier un audit	-Conception et création de l'interface	1
	- En tant que utilisateur je peux ajouter un plan audit.	-Implémentation de code	
	- En tant que utilisateur je peux suivre laudit	-Test de l'interface - Réalisation	
	-En tant que utilisateur je peux consulter , modifier ou supprimer un audit	-Test des fonctionnalités	
Réunion	- En tant que utilisateur je peux planifier une réunion	-Implémentation des fonctionnalités	1
	- En tant que utilisateur je peux suivre létat de réalisation de réunion	-Réalisation -Test	
	-En tant que guest ou utilisateur je peux participer à une réunion .	-Mise en place de l'interface	
Actions	- En tant que utilisateur je peux ajouter , modifier action	-Mise en place de l'interface - Réalisation	1

Tableau 5.1 : Backlog Sprint 3

3 Besoins fonctionnels

- Gérer audits : Cette fonctionnalité permet aux utilisateurs la possibilité de planifier, réaliser et suivre des audits au sein d'une organisation.

- Gérer les actions : Cette fonctionnalité offre aux utilisateurs la possibilité d'ajouter une action, modifier une action ou la supprimer
- Planifier réunion : Cette fonctionnalité offre aux utilisateurs la possibilité de planifier une réunion et suivre l'état de réalisation

4 Mise en place du module Audit

Dans cette section, nous allons représenter les diagrammes de classes, de cas d'utilisation et de séquence pour le module Audit

4.1 Diagramme de classe

La figure ci dessous représente le diagramme de classes d'audit

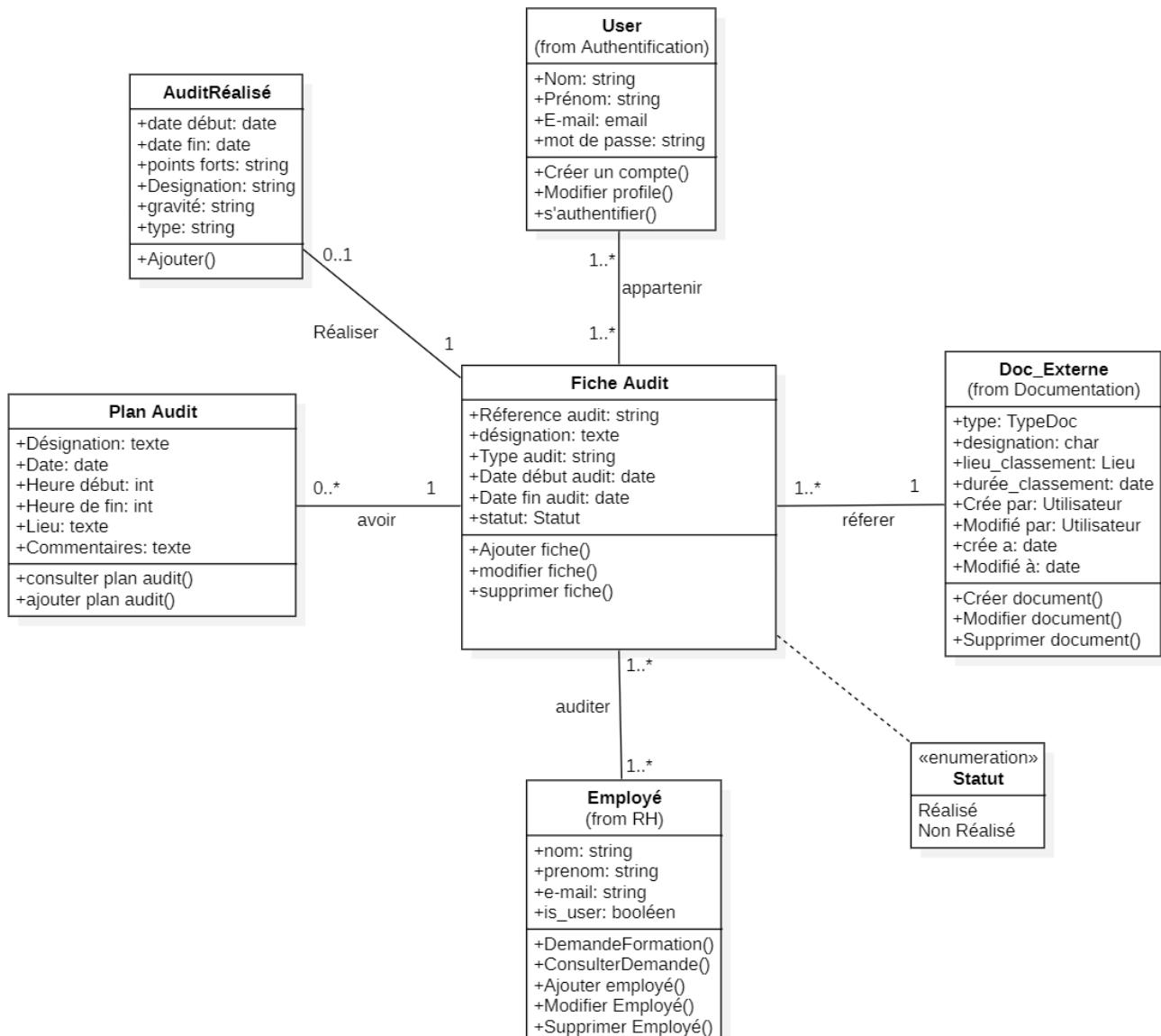


Figure 5.1 : Diagramme de classes d'audit

4.2 Diagramme de cas d'utilisation

La figure ci dessous représente le diagramme de cas d'utilisation d'audit

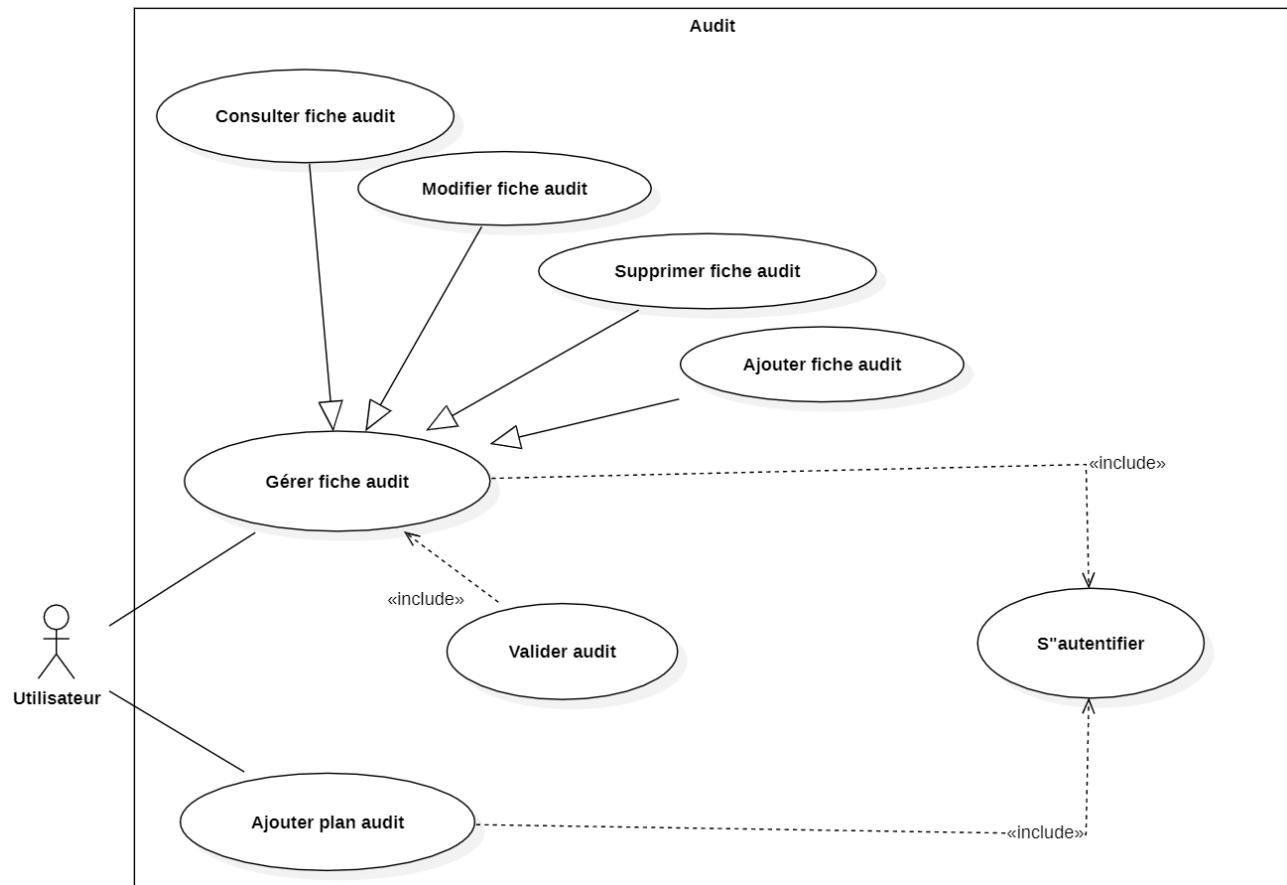


Figure 5.2 : Diagramme de classe d'audit

4.3 Description textuelle des cas d'utilisation

- Description textuelle du cas d'utilisation "Ajouter fiche audit"

Cas d'utilisation	Ajouter fiche audit
Acteur	Utilisateur
Pré-condition	L'utilisateur doit être authentifié
Scénario nominal	-L'utilisateur remplit une fiche d'ajout -L'utilisateur valide les informations
Post-condition	L'ajout d'une fiche audit

Tableau 5.2 : Description textuelle du cas d'utilisation "Ajouter fiche audit"

- Description textuelle du cas d'utilisation "Ajouter plan audit"

Cas d'utilisationn	Ajouter plan audit
Acteur	Utilisateur
Pré-condition	L'utilisateur doit être authentifié
Scénario nominal	<ul style="list-style-type: none"> -L'utilisateur sélectionne la fiche d'audit -L'utilisateur ajoute un plan sous forme d'un tableau -L'utilisateur valide les données
Post-condition	plan audit ajouté

Tableau 5.3 : Description textuelle du cas d'utilisation "Ajouter plan audit"

- Description textuelle du cas d'utilisation "Valider audit"

Cas d'utilisation	Valider audit
Acteur	Utilisateur
Scénario nominal	<ul style="list-style-type: none"> -L'utilisateur sélectionne fiche d'audit -l'utilisateur change le statut à "Réalisé" - L'utilisateur remplit fiche de réalisation
Post-condition	audit validé

Tableau 5.4 : Description textuelle du cas d'utilisation "Valider audit"

4.4 Diagrammes de séquence

La figure ci dessous représente le diagramme de séquence "Ajouter fiche audit"

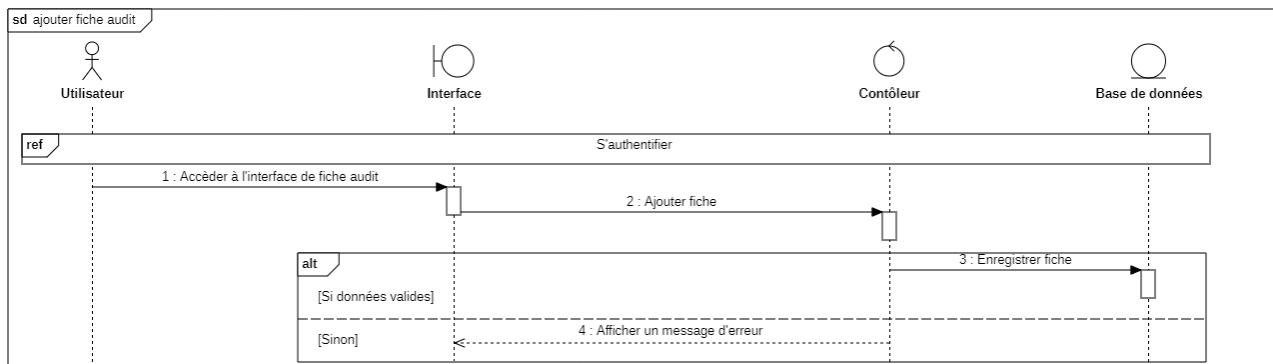


Figure 5.3 : Diagramme de séquence "ajouter fiche audit"

La figure ci dessous représente le diagramme de séquence "Valider audit"

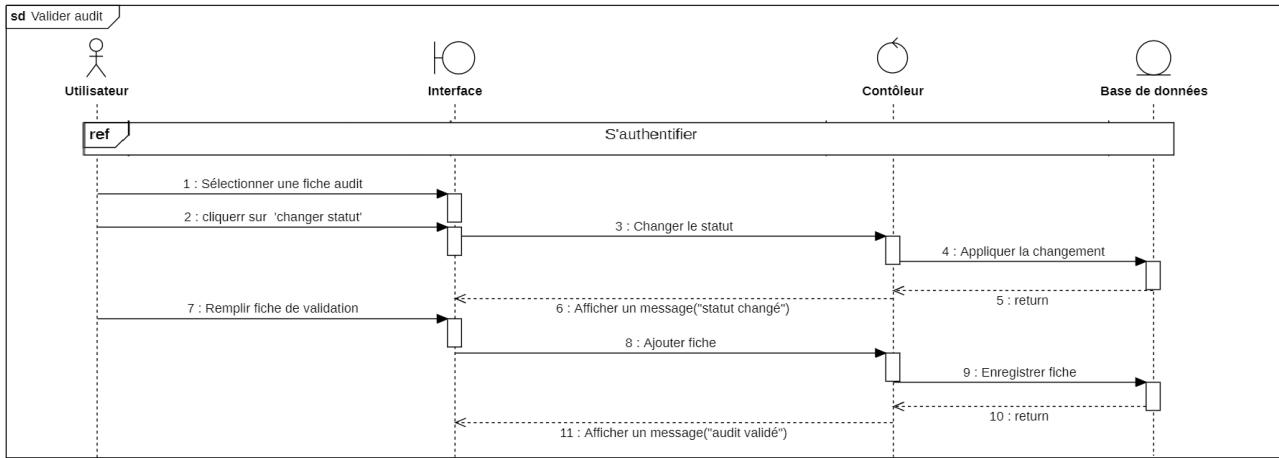


Figure 5.4 : Diagramme de séquence "valider audit"

5 Mise en place du module réunion

Dans cette section, nous allons représenter les diagrammes de classe, de cas d'utilisation et de séquence pour le module réunion.

5.1 Diagramme de classes

La figure ci dessous représente le diagramme de classe de réunion

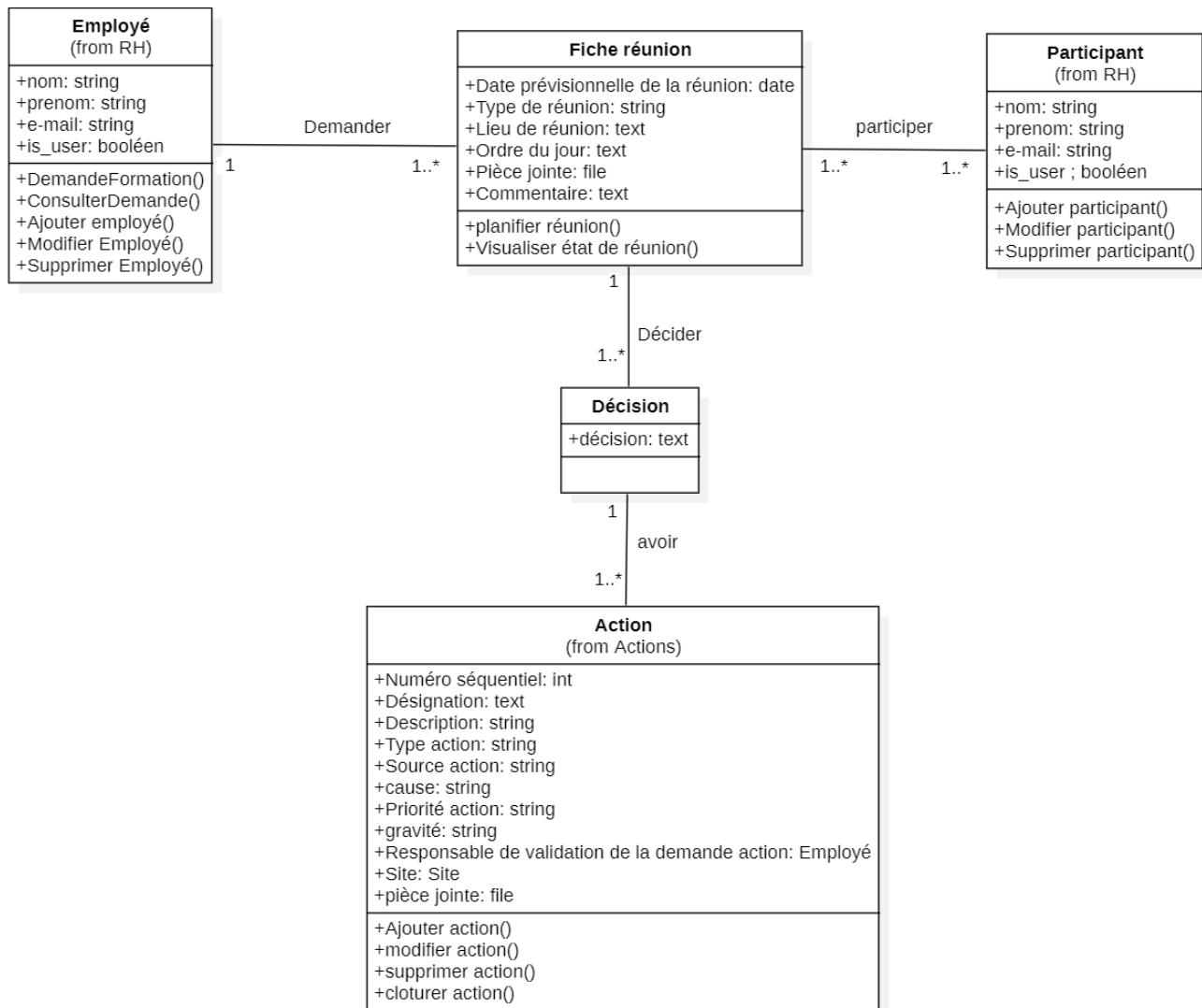


Figure 5.5 : Diagramme de classe de réunion

5.2 Diagramme de cas d'utilisation

La figure ci dessous représente le diagramme de cas d'utilisation de réunion

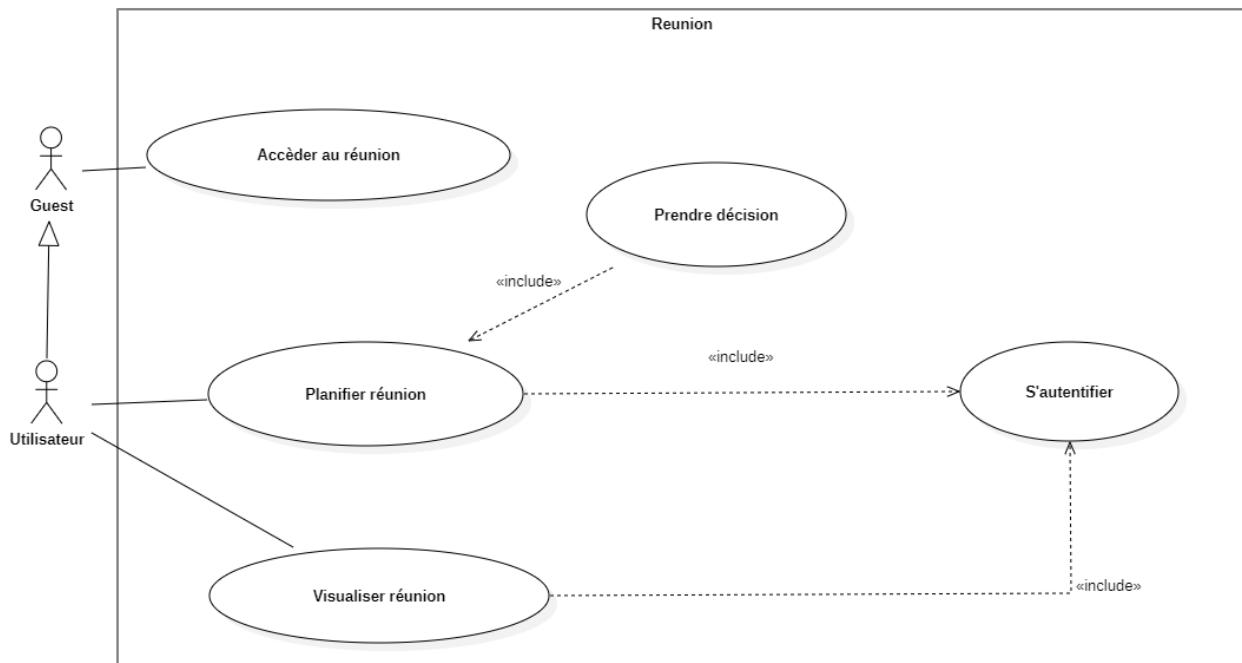


Figure 5.6 : Diagramme de cas d'utilisation de réunion

5.3 Description textuelle des cas d'utilisation

- Description textuelle du cas d'utilisation "Planifier réunion"

Cas d'utilisation	Planifier réunion
Acteur	Utilisateur
Pré-condition	L'utilisateur doit être authentifié
Scénario nominal	<ul style="list-style-type: none"> -L'utilisateur remplit une fiche de réunion tel que (la date,type,liste des participants...) -L'utilisateur valide les informations - L'utilisateur notifie les participants
Post-condition	Réunion planifiée

Tableau 5.5 : Description textuelle du cas d'utilisation "Planifier réunion"

- Description textuelle du cas d'utilisation "Prendre décision"

Cas d'utilisation	Prendre décision
Acteurs	Utilisateur
Pré-condition	L'utilisateur doit être authentifié

Scénario nominal	<ul style="list-style-type: none"> -L'utilisateur sélectionne la réunion -L'utilisateur ajoute des champs tel que (actions ,décision prise) -L'utilisateur valide les données
Post-condition	Décision prise

Tableau 5.6 : Description textuelle du cas d'utilisation "Prendre décision"

- Description textuelle du cas d'utilisation "Visualiser réunion"

Cas d'utilisation	Visualiser réunion
Acteurs	Utilisateur
Scénario nominal	<ul style="list-style-type: none"> -L'utilisateur sélectionne la réunion -l'utilisateur consulte l'état de la réunion (réalisée ou non réalisée)

Tableau 5.7 : Description textuelle du cas d'utilisation "Visualiser réunion"

5.4 Diagrammes de séquence

La figure ci dessous représente le diagramme de séquence "Planifier réunion"

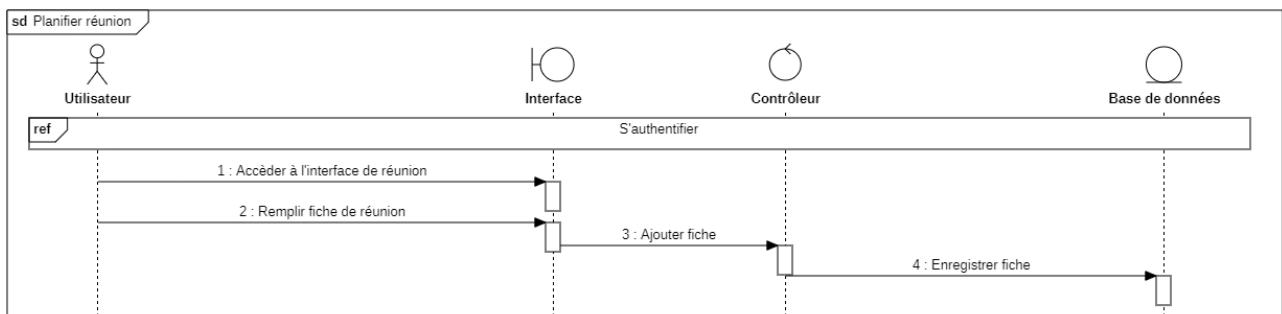


Figure 5.7 : Diagramme de séquence de Planifier réunion

La figure ci dessous représente le diagramme de séquence "Prendre décision"

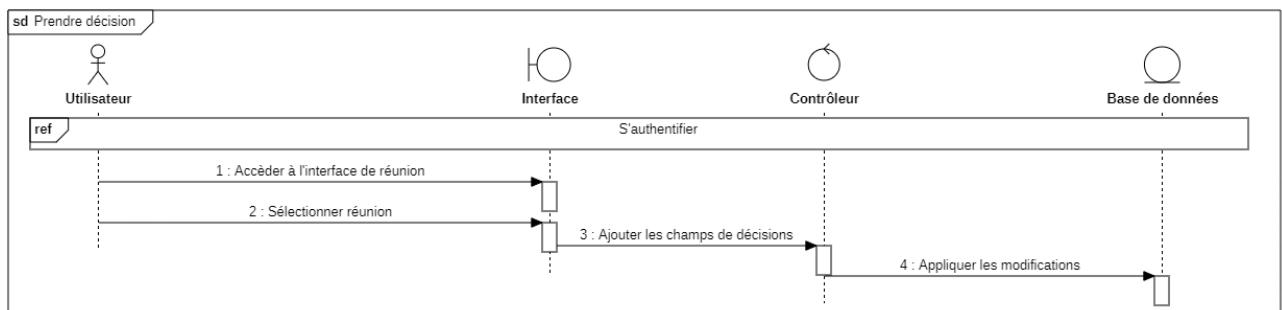


Figure 5.8 : Diagramme de séquence de prendre décision

6 Mise en place de module actions

Dans cette section,nous allons représenté les diagrammes de classe, de cas d'utilisation et de séquence pour le module Actions

6.1 Diagramme de classes

La figure ci dessous représente le diagramme de classes d'action

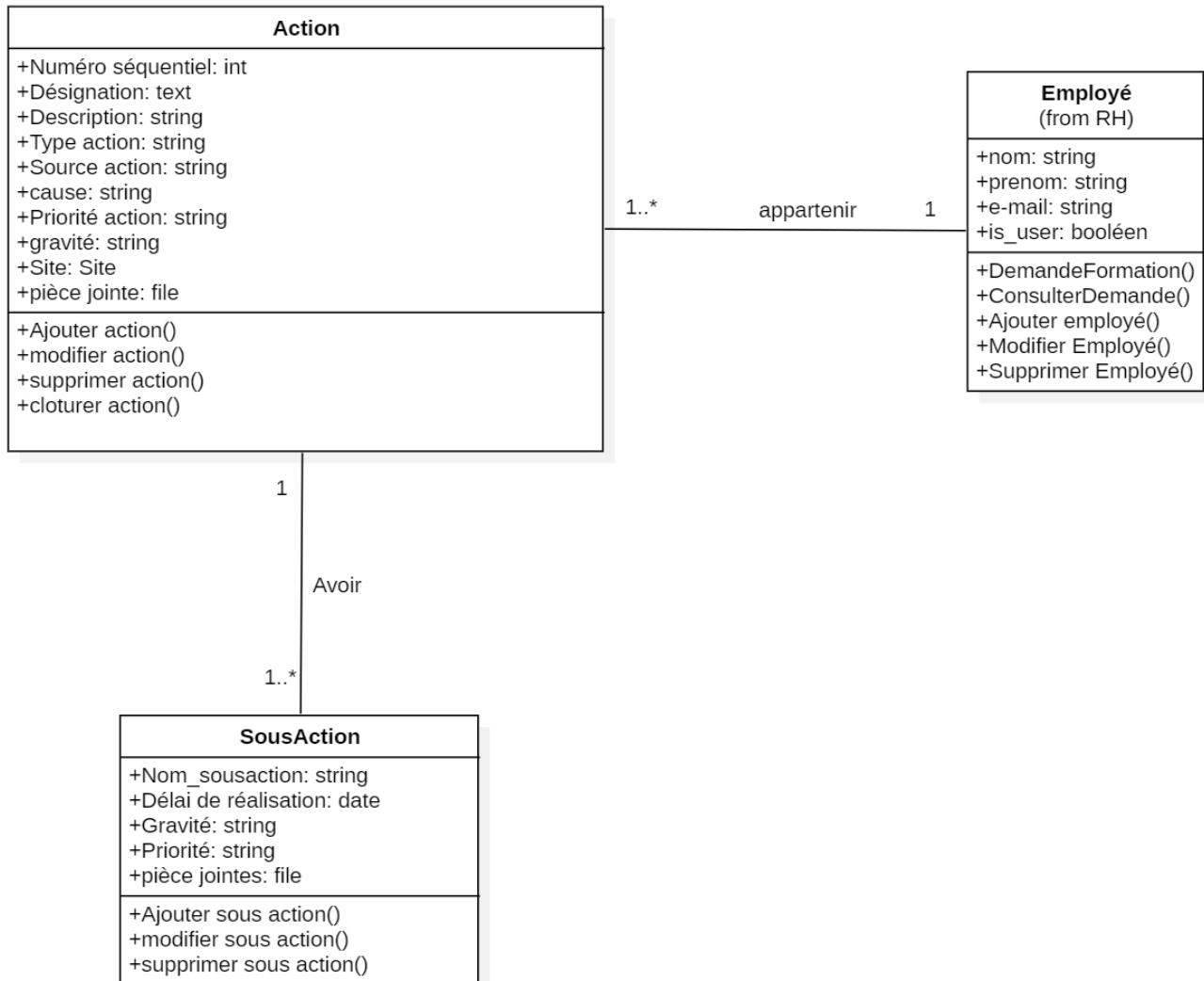


Figure 5.9 : Diagramme de classes d'action

6.2 Diagramme de cas d'utilisation

La figure ci dessous représente le diagramme de cas d'utilisation d'action

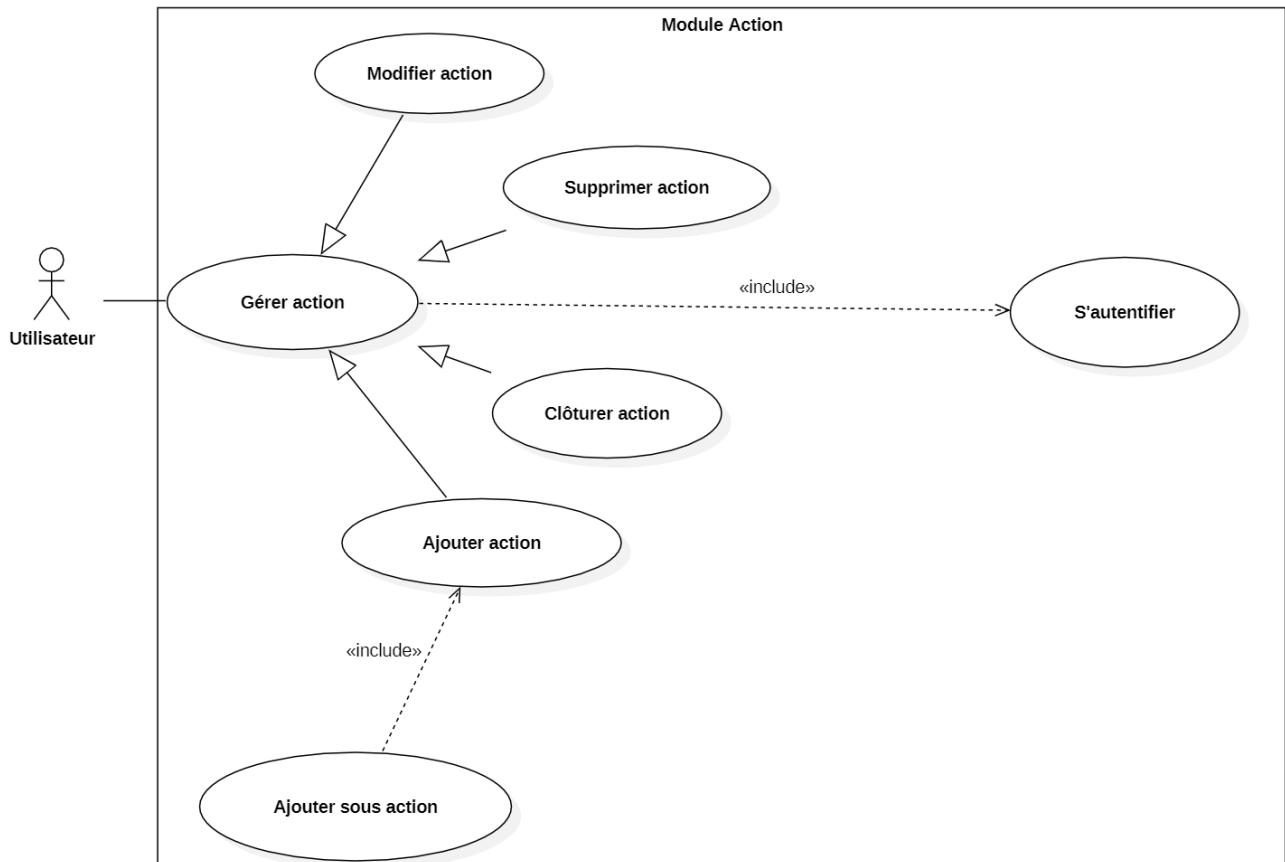


Figure 5.10 : Diagramme de cas d'utilisation d'action

6.3 Description textuelle des cas d'utilisation

- Description textuelle du cas d'utilisation "Ajouter action"

Cas d'utilisation	Ajouter action
Acteur	Utilisateur
Pré-condition	L'utilisateur doit être authentifié
Scénario nominal	-L'utilisateur remplit une fiche d'action -L'utilisateur valide les informations
Post-condition	Action créée

Tableau 5.8 : Description textuelle du cas d'utilisation "Ajouter action"

- Description textuelle du cas d'utilisation "Modifier action"

Cas d'utilisation	Modifier action
Acteurs	Utilisateur

Pré-condition	L'utilisateur doit être authentifié et autorisé par le responsable d'action
Scénario nominal	<ul style="list-style-type: none"> -L'utilisateur envoie une demande de modification au responsable d'action - Si la demande acceptée , l'utilisateur modifie l'action -L'utilisateur valide les données
Post-condition	Action modifiée et modifications sauvegardées dans la traçabilité

Tableau 5.9 : Description textuelle du cas d'utilisation "Modifier action"

- Description textuelle du cas d'utilisation "ajouter sous action"

Cas d'utilisation	ajouter sous action
Acteurs	Utilisateur
Pré-condition	L'utilisateur doit être authentifié et il existe une action
Scénario nominal	<ul style="list-style-type: none"> -L'utilisateur sélectionne l'action -l'utilisateur saisit les informations de sous actions -L'utilisateur valide les informations
Post-condition	sous action ajouté

Tableau 5.10 : Description textuelle du cas d'utilisation "ajouter sous action"

- Description textuelle du cas d'utilisation "Clôturer action"

Cas d'utilisation	Clôturer action
Acteurs	Utilisateur
Pré-condition	L'utilisateur doit être authentifié

Scénario nominal	<ul style="list-style-type: none"> -L'utilisateur sélectionne la fiche d'action -l'utilisateur saisit les informations de clôture tel que(Le responsable de clôture , délai clôture , efficacité d'action) -L'utilisateur envoie la fiche au responsable validation
Post-condition	Action clôturé

Tableau 5.11 : Description textuelle du cas d'utilisation "Clôturer action"

6.4 Diagrammes de séquence

La figure ci dessous représente le diagramme de séquence "Modifier action"

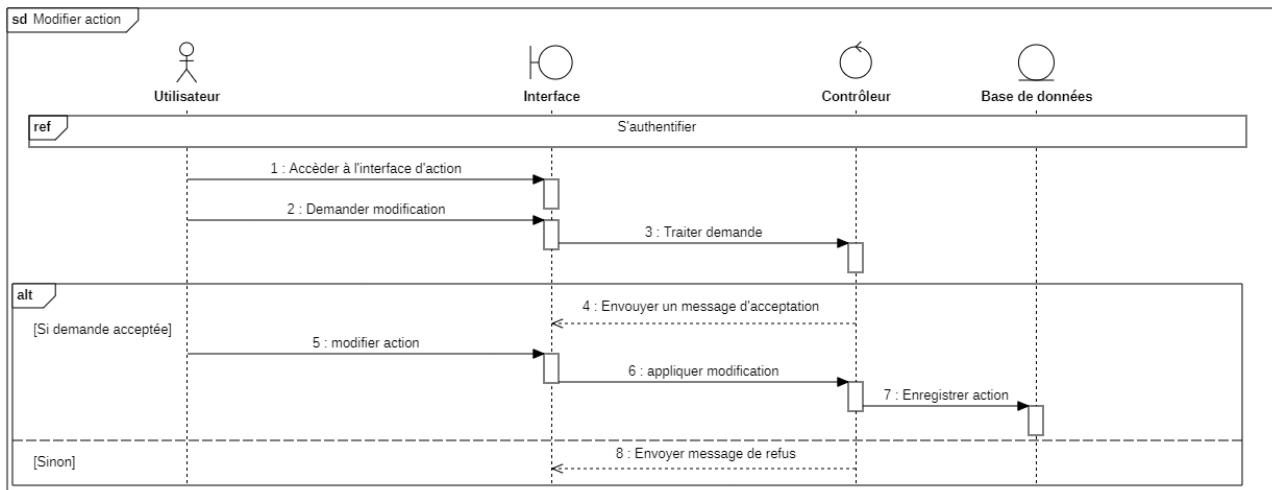


Figure 5.11 : Diagramme de séquence de Modifier action

La figure ci dessous représente le diagramme de séquence "Clôturer action"

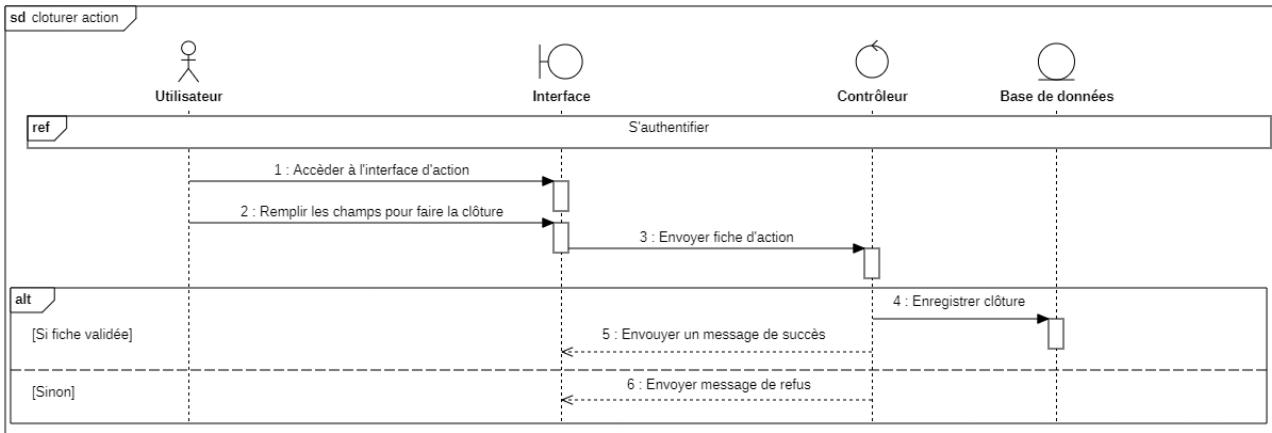


Figure 5.12 : Diagramme de séquence de Clôturer action

7 Réalisation

La figure 5.13 présente l'interface consulter réunion

L'interface de consultation de réunions affiche trois réunions séparées dans des boîtes :

- Réunion 1** : Crée par: user prenom, type_reunion: One-on-One, lieu: Salle de conférence A, ordre_du_jour: 1. Introduction 2. Revue des actions précédentes 3. Nouveaux sujets 4. Conclusion. Bouton "Consulter".
- Réunion 2** : Crée par: user prenom, type_reunion: One-on-One, lieu: Salle de conférence A, ordre_du_jour: 1. Introduction 2. Revue des actions précédentes 3. Nouveaux sujets 4. Conclusion. Bouton "Consulter".
- Réunion 3** : Crée par: user prenom, type_reunion: One-on-One, lieu: Salle de conférence A, ordre_du_jour: 1. Introduction 2. Revue des actions précédentes 3. Nouveaux sujets 4. Conclusion. Bouton "Consulter".

En bas à gauche, un bouton "Ajouter Réunion". En bas à droite, un bouton "Retour".

Figure 5.13 : Interface consulter réunion

La figure 5.14 présente l'interface ajouter action

L'interface pour ajouter une action contient les champs suivants :

- Type Action : Select Type
- Source Action : Select Source
- Cause Action : Select Cause
- Gravité Action : Select Gravité
- Priorité Action : Select Priorité
- Site : Select Site
- Nom Action : champ texte
- Designation : champ texte
- Description : champ texte
- Responsable Validation (ID de l'employé) : champ texte

Figure 5.14 : Interface ajouter action

8 Test

Cette interface permet de tester la méthode de "Créer Audit"

The screenshot shows a Postman interface for a POST request to `http://localhost:8000/audit/create_audit/`. The request body is a JSON object representing an audit:

```

1 {
2     "id": 3,
3     "reference_audit": "AUD2024-001",
4     "designation": "Audit de conformité des processus internes.",
5     "champ_audit": [40],
6     "type_audit": "Interne",
7     "auditeurs": [40],
8     "date_debut_audit": "2024-04-22",
9     "date_fin_audit": "2024-04-25",
10    "document_reference": 6,
11    "audités": [6, 7],
12    "responsable_validation": 8
13 }

```

The response status is `201 Created` with a time of `104 ms` and a size of `676 B`.

Figure 5.15 : Test méthode "Créer Audit"

Cette interface permet de tester la méthode de "Créer Action"

The screenshot shows a Postman interface for a POST request to `http://localhost:8000/action/create_action/`. The request body is a JSON object representing an action:

```

1 {
2     "nom_action": "Réparation d'urgence",
3     "designation": "Réparation des machines",
4     "description": "La réparation immédiate des machines suite à une panne majeure.",
5     "type_action": 1,
6     "source_action": 1,
7     "cause_action": 1,
8     "gravite_action": 1,
9     "priorite_action": 1,
10    "responsable_validation": 18,
11    "site": 1,
12    "plan": null
13 }

```

The response status is `201 Created` with a time of `82 ms` and a size of `716 B`.

Figure 5.16 : Test méthode "Créer action"

Cette interface permet de tester la méthode de "Planifier réunion"

The screenshot shows a POST request in Postman. The URL is `http://localhost:8000/reunion/create_Meet/`. The Body tab is selected, showing the following JSON payload:

```

1  {
2   ... "date_previsionnelle": "2024-06-15",
3   ... "type_reunion": 1,
4   ... "lieu": "Salle de conférence A",
5   ... "ordre_du_jour": "1. Introduction\n2. Revue des actions précédentes\n3. Nouveaux sujets\n4. Conclusion",
6   ... "participants": [62],
7   ... "piece_jointe": null,
8   ... "commentaire": "Préparer les documents pour la présentation."
9 }
10
11
12
13
14
    
```

The response status is 201 Created, time 100 ms, size 687 B. The JSON response body is identical to the sent payload.

Figure 5.17 : Test méthode "Planifier réunion"

9 conclusion

En conclusion de ce sprint, nous avons réalisé des avancées significatives dans le développement de notre projet, en mettant en place des fonctionnalités clés telles que la gestion des audits, la gestion des actions et la planification des réunions. Dans notre prochain chapitre nous nous concentrerons sur nos finales fonctionnalités notamment la gestion des produit non conforme , la gestion des risques et la gestion de conformité réglementaire.

Sprint 4 : Gestion des produits non conforme , conformité réglementaire et des risques.

Plan

1	Introduction	83
2	Backlog du sprint	83
3	Besoins fonctionnels	83
4	Mise en place du module Produit non conforme	84
5	Mise en place du module Conformité réglementaire	87
6	Mise en place de module risque	90
7	Réalisation	93
8	Test	94
9	conclusion	95

1 Introduction

Ce chapitre marque la conclusion de notre projet, qui se concentre . la gestion des produits non conforme, conformité réglementaire et des risques .

2 Backlog du sprint

Le tableau ci dessous représente le Backlog du sprint 2

Module	User Story	Taches	Estimation
Produit non conforme	- En tant que utilisateur je peux ajouter non conformité	-Implémentation des fonctionnalités	1
	- En tant que utilisateur je peux traiter non conformité	-Réalisation -Test	
	-En tant que utilisateur je peux suivre non conformité .	-Mise en place de l'interface	
Conformité réglementaire	- En tant que utilisateur je peux examiner les informations relatives à la conformité réglementaire	-Mise en place de l'interface - Réalisation	1
Risque	- En tant que utilisateur je peux identifier et de traiter les risques potentiels	-Mise en place de l'interface - Réalisation	1

Tableau 6.1 : Backlog Sprint 4

3 Besoins fonctionnels

- produit non conforme : Cette fonctionnalité permet aux utilisateurs la possibilité de gérer les non-conformités et les réclamations de manière efficace en permettant d'enregistrer les informations pertinentes, d'identifier les causes et les sources de non-conformité, d'associer les réclamations aux produits non conformes, de traiter les non-conformités de manière rapide et efficace, de suivre les actions correctives et préventives, et de prévenir la récurrence des non-conformités..
- Gérer conformité réglementaire : Cette fonctionnalité permet aux utilisateurs d'assurer

la veille sur les exigences réglementaires applicables à l'activité de l'entreprise

- Gérer risque : Cette fonctionnalité permet aux utilisateurs d'identifier et de traiter les risques potentiels pour l'entreprise

4 Mise en place du module Produit non conforme

Dans cette section,nous allons représenté les diagrammes de classe, de cas d'utilisation et de séquence pour le module Produit non conforme

4.1 Diagramme de classes

La figure 6.1 représente le diagramme de classes de Produit non conforme

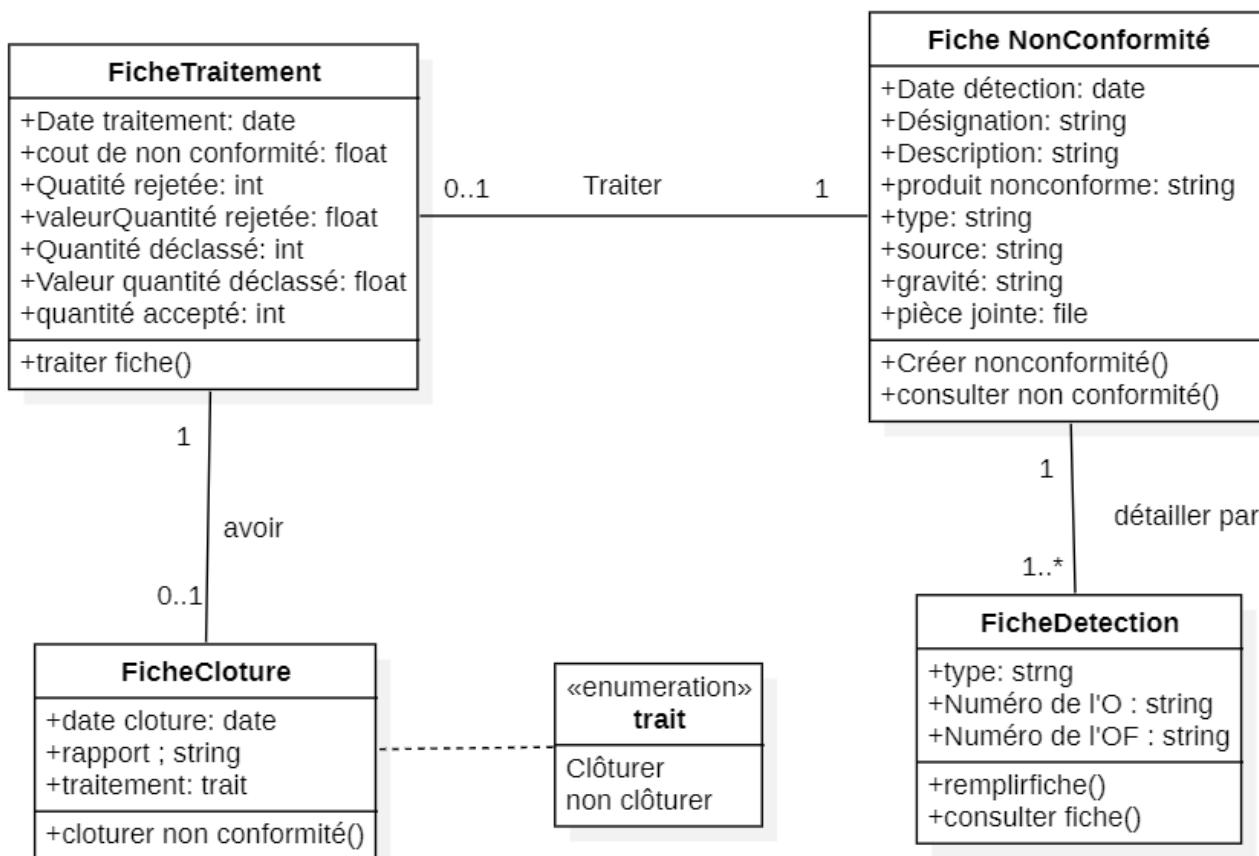


Figure 6.1 : Diagramme de classes Produit non conforme

4.2 Diagramme de cas d'utilisation

La figure 6.2 représente le diagramme de cas d'utilisation de Produit non conforme

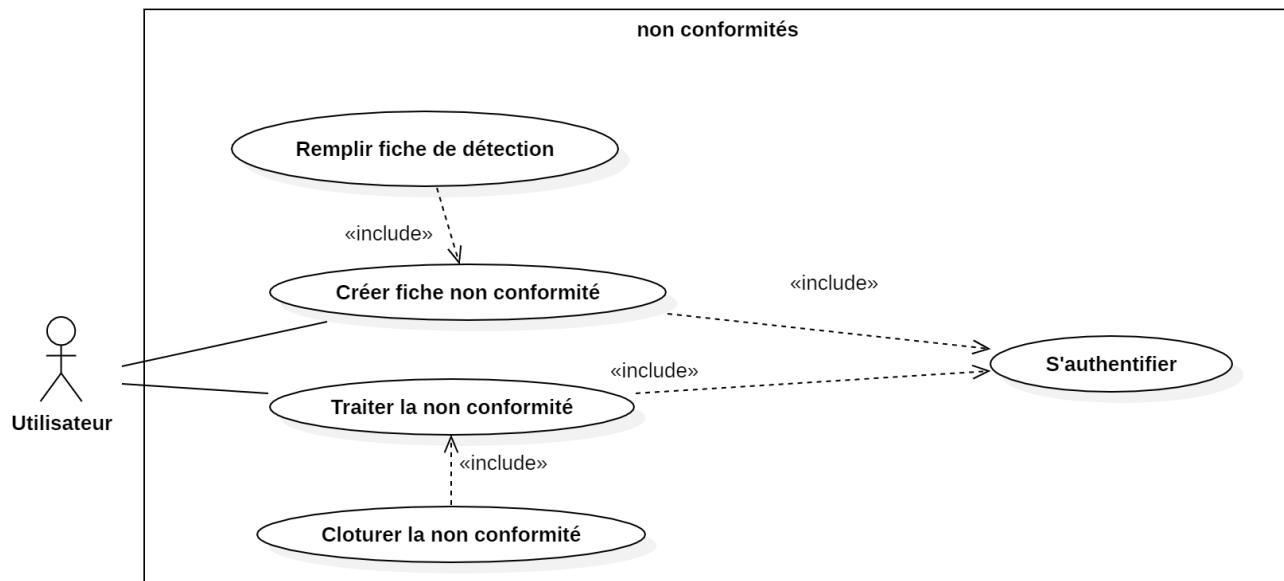


Figure 6.2 : Diagramme de cas d'utilisation de Produit non conforme

4.3 Description textuelle des cas d'utilisation

- Description textuelle du cas d'utilisation "Créer fiche non conformité"

Cas d'utilisation	Créer fiche non conformité
Acteur	Utilisateur
Pré-condition	L'utilisateur doit être authentifié
Scénario nominal	-L'utilisateur remplit une fiche d'ajout -L'utilisateur valide les informations
Post-condition	L'ajout d'une non conformité

Tableau 6.2 : Description textuelle du cas d'utilisation "Créer fiche non conformité"

- Description textuelle du cas d'utilisation "Remplir fiche de détection"

Cas d'utilisation	Remplir fiche de détection
Acteurs	Utilisateur
Pré-condition	L'utilisateur doit être authentifié et une non conformité existe
Scénario nominal	-L'utilisateur remplit une fiche de détection -L'utilisateur sélectionne la non conformité concernée -L'utilisateur valide les données

Post-condition	Fiche de détection remplit
-----------------------	----------------------------

Tableau 6.3 : Description textuelle du cas d'utilisation "Remplir fiche de détection"

- Description textuelle du cas d'utilisation "Traiter non conformité"

Cas d'utilisation	Traiter non conformité
Acteurs	Utilisateur
Pré-condition	L'utilisateur doit être authentifié et une non conformité déclenchée
Scénario nominal	-L'utilisateur sélectionne la non conformité -l'utilisateur remplit une fiche de traitement - L'utilisateur valide la fiche
Post-condition	non conformité traité

Tableau 6.4 : Description textuelle du cas d'utilisation "Traiter non conformité"

- Description textuelle du cas d'utilisation "Clôturer non conformité"

Cas d'utilisation	Clôturer non conformité
Acteurs	Utilisateur
Pré-condition	L'utilisateur doit être authentifié et une non conformité traitée
Scénario nominal	-L'utilisateur sélectionne la non conformité -l'utilisateur remplit une fiche de clôture - L'utilisateur valide la fiche
Post-condition	non conformité clôturé

Tableau 6.5 : Description textuelle du cas d'utilisation "Clôturer non conformité"

4.4 Diagrammes de séquence

La figure ci dessous représente le diagramme de séquence "Créer fiche non conformité"

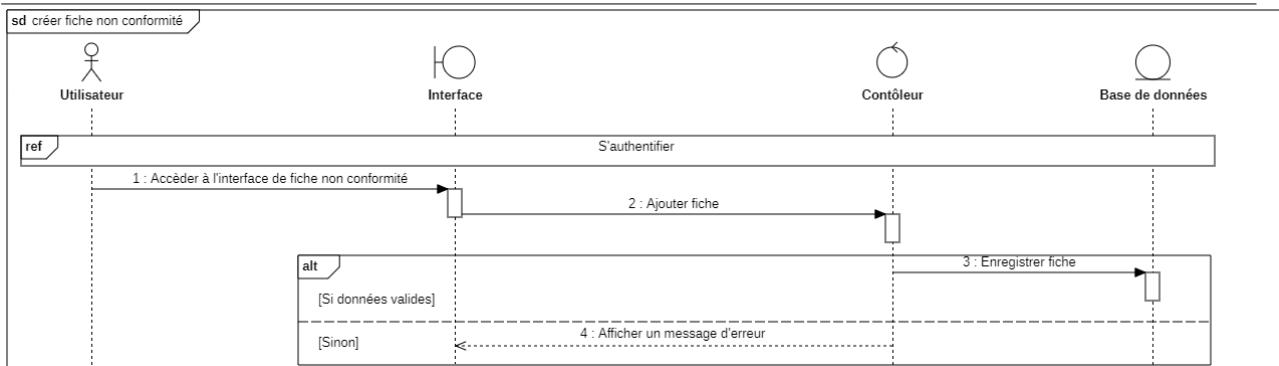


Figure 6.3 : Diagramme de séquence "Créer fiche non conformité"

La figure ci dessous représente le diagramme de séquence "Traiter non conformité"

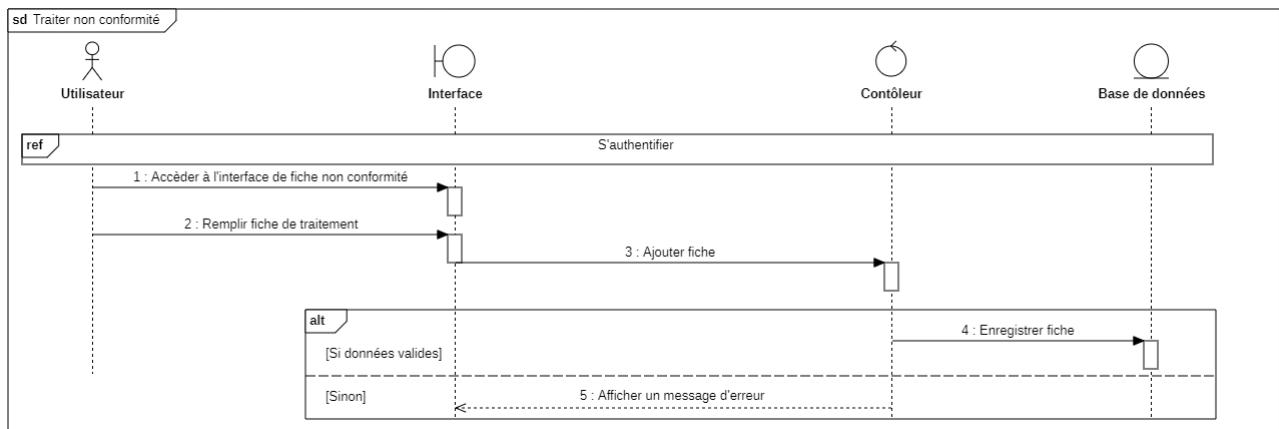


Figure 6.4 : Diagramme de séquence "Traiter non conformité"

5 Mise en place du module Conformité réglementaire

Dans cette section, nous allons représenter les diagrammes de classe, de cas d'utilisation et de séquence pour le module Conformité réglementaire.

5.1 Diagramme de classe

La figure ci dessous représente le diagramme de classe de Conformité réglementaire

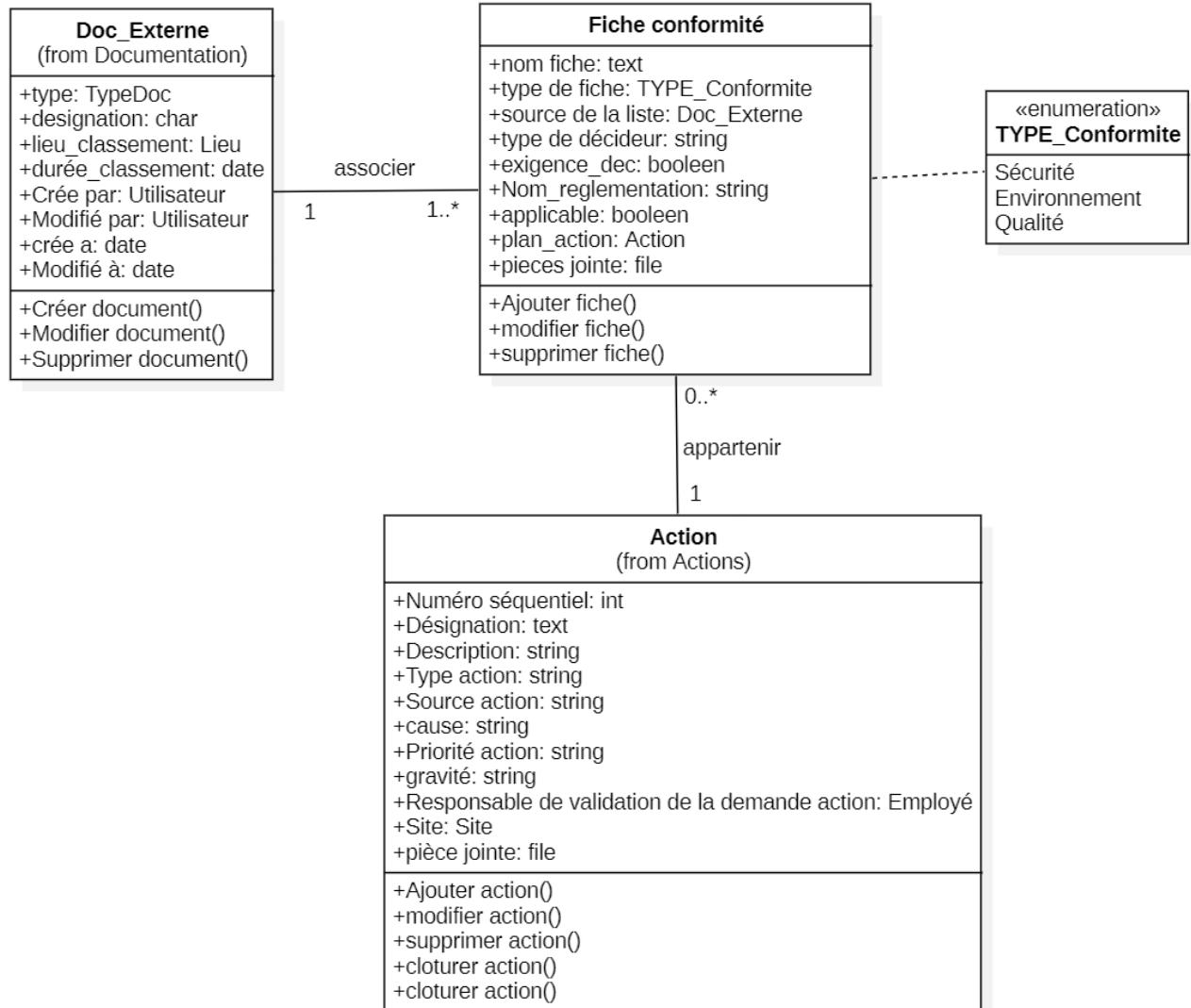


Figure 6.5 : Diagramme de classe de Conformité réglementaire

5.2 Diagramme de cas d'utilisation

La figure ci dessous représente le diagramme de cas d'utilisation de Conformité réglementaire

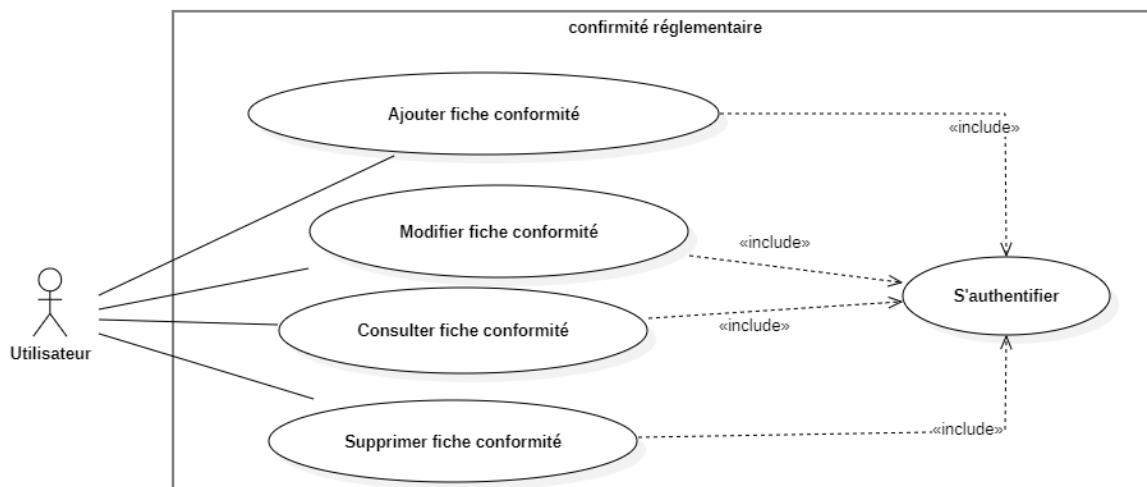


Figure 6.6 : Diagramme de cas d'utilisation de Conformité réglementaire

5.3 Description textuelle des cas d'utilisation

- Description textuelle du cas d'utilisation "Ajouter fiche conformité"

Cas d'utilisation	Ajouter fiche conformité
Acteur	Utilisateur
Pré-condition	L'utilisateur doit être authentifié
Scénario nominal	-L'utilisateur remplit une fiche de conformité -L'utilisateur enregistre la fiche
Post-condition	Conformité enregistré

Tableau 6.6 : Description textuelle du cas d'utilisation "Ajouter fiche conformité"

- Description textuelle du cas d'utilisation "Modifier fiche conformité"

Cas d'utilisation	Modifier fiche conformité
Acteur	Utilisateur
Pré-condition	L'utilisateur doit être authentifié et fiche existe
Scénario nominal	-L'utilisateur sélectionne la fiche à modifier -L'utilisateur valide les informations
Post-condition	fiche modifiée

Tableau 6.7 : Description textuelle du cas d'utilisation "Modifier fiche conformité"

5.4 Diagrammes de séquence

La figure ci dessous représente le diagramme de séquence "Ajouter fiche conformité"

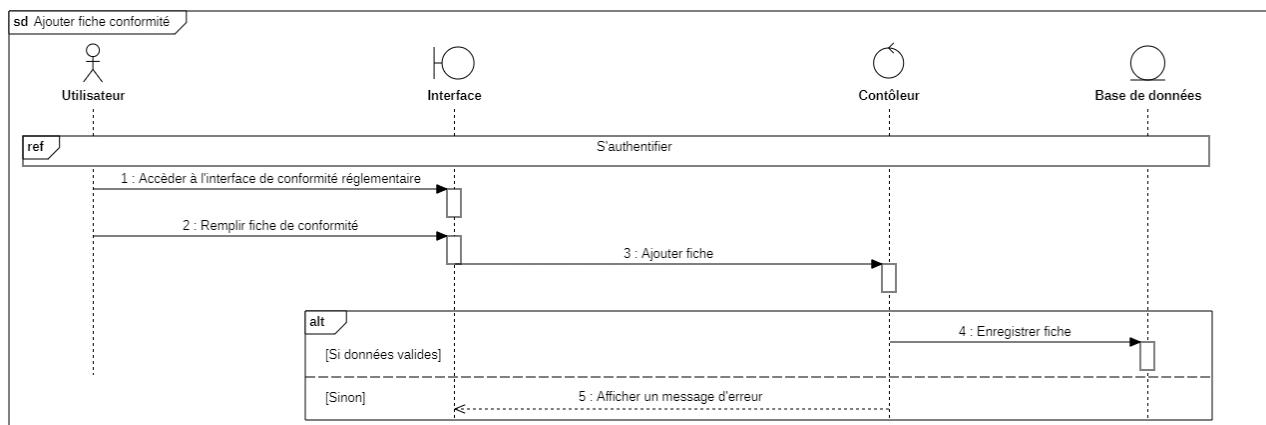


Figure 6.7 : Diagramme de séquence de Ajouter fiche conformité

La figure ci dessous représente le diagramme de séquence "Modifier fiche conformité"

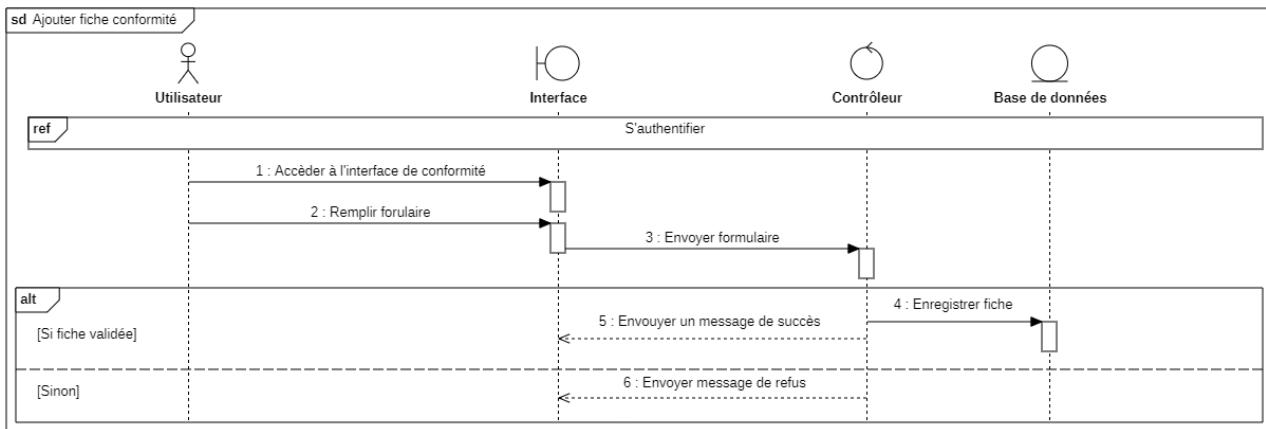


Figure 6.8 : Diagramme de séquence de Modifier fiche conformité

6 Mise en place de module risque

Dans cette section,nous allons représenté les diagrammes de classe, de cas d'utilisation et de séquence pour le module Risque

6.1 Diagramme de classes

La figure ci dessous représente le diagramme de classes risque

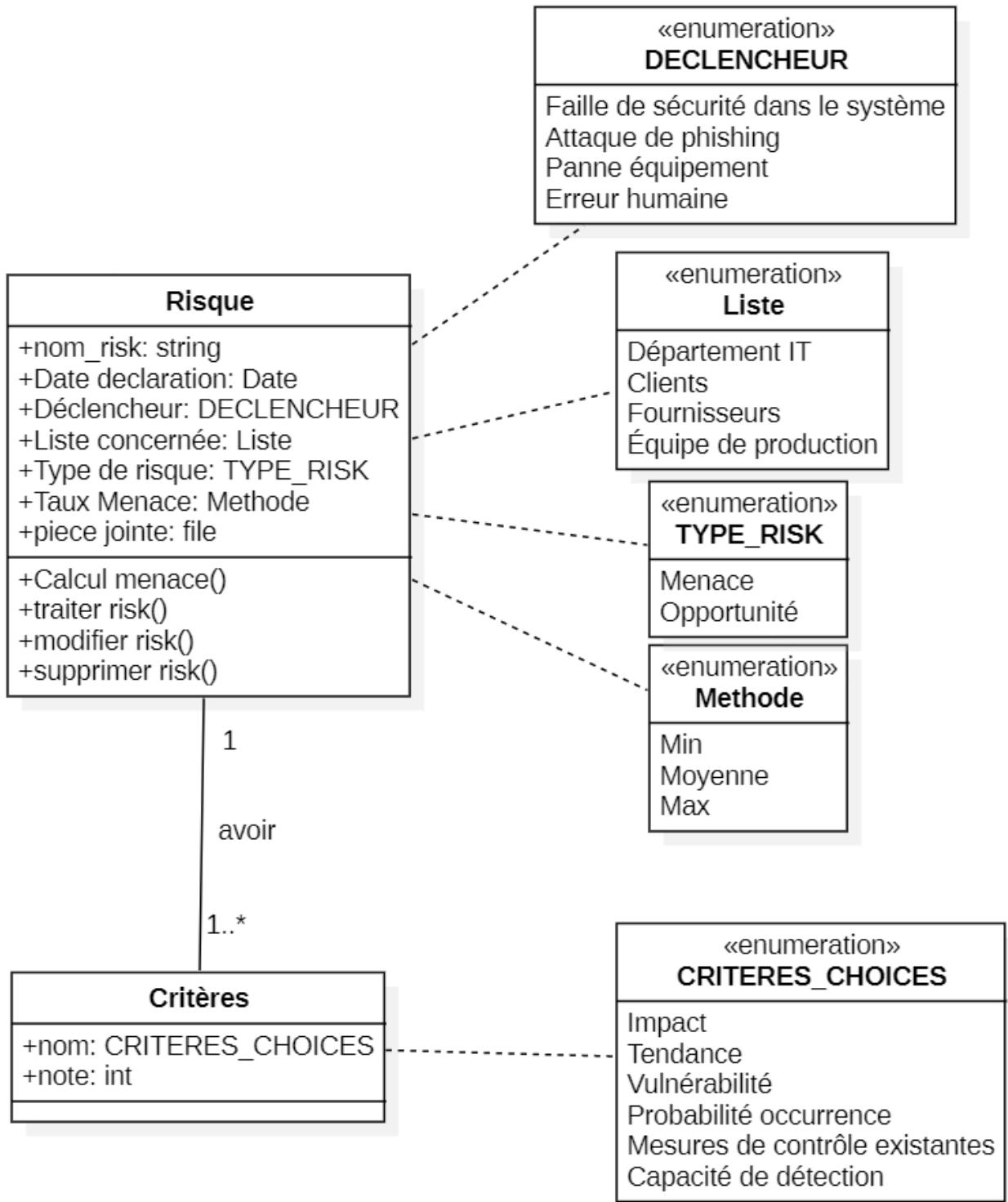


Figure 6.9 : Diagramme de classes de risque

6.2 Diagramme de cas d'utilisation

La figure ci dessous représente le diagramme de cas d'utilisation de risque

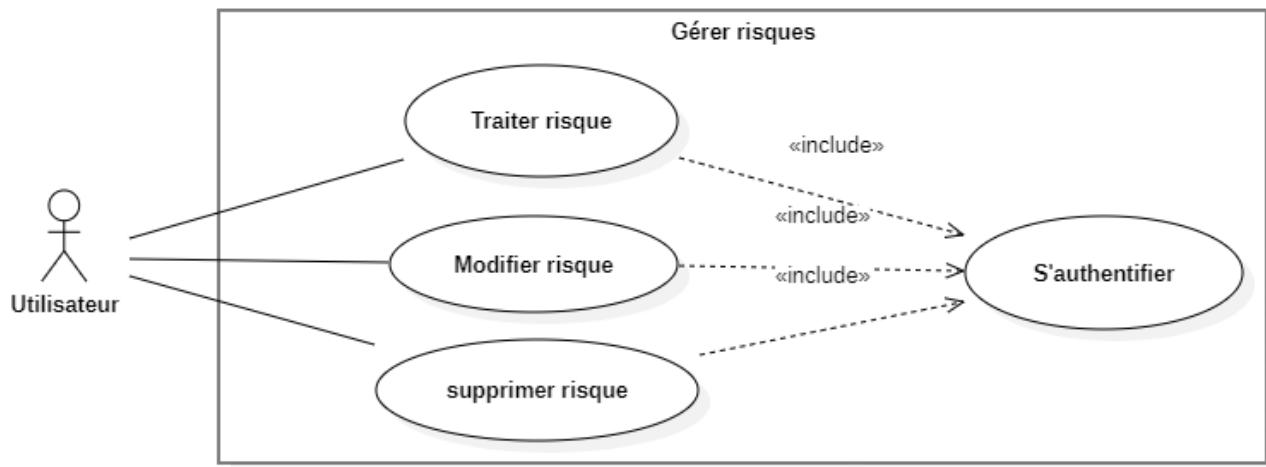


Figure 6.10 : Diagramme de cas d'utilisation de risque

6.3 Description textuelle des cas d'utilisation

- Description textuelle du cas d'utilisation "Traiter risque"

Cas d'utilisation	Traiter risque
Acteur	Utilisateur
Pré-condition	L'utilisateur doit être authentifié
Scénario nominal	-L'utilisateur remplit une fiche de risque -L'utilisateur valide les données
Post-condition	Fiche de risque ajoutée

Tableau 6.8 : Description textuelle du cas d'utilisation "Traiter risque"

6.4 Diagrammes de séquence

La figure ci dessous représente le diagramme de séquence "Traiter risque"

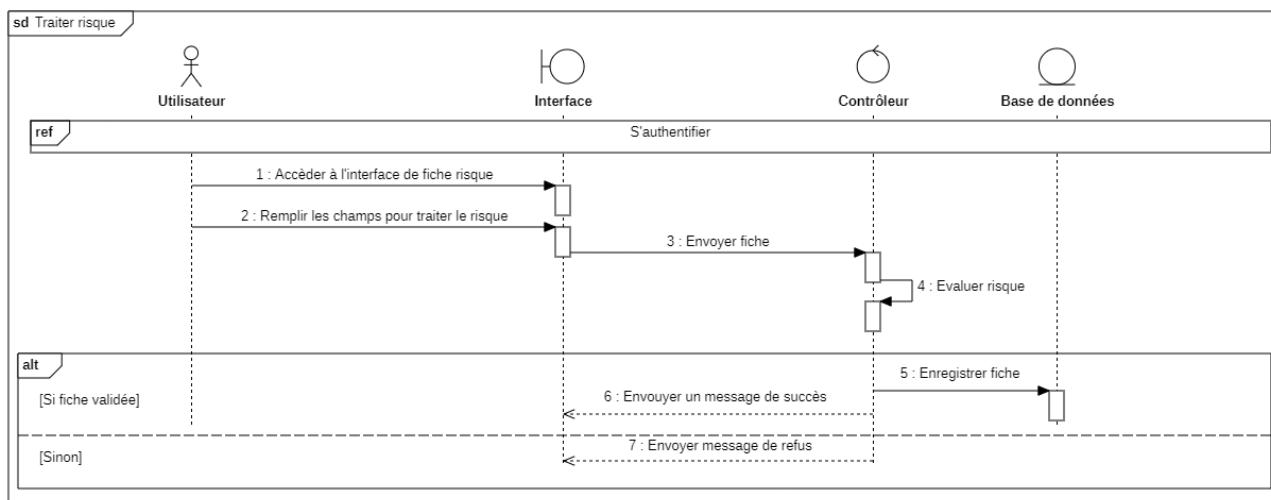
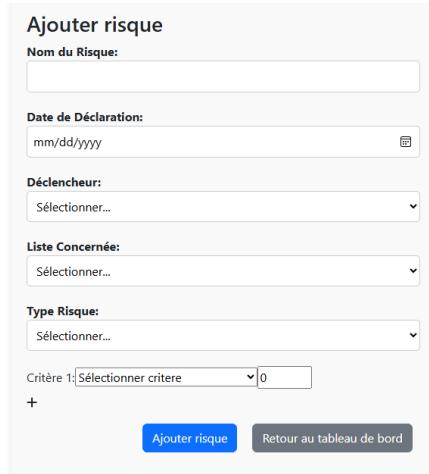


Figure 6.11 : Diagramme de séquence de Traiter risque

7 Réalisation

La figure 6.12 présente l'interface traiter risque



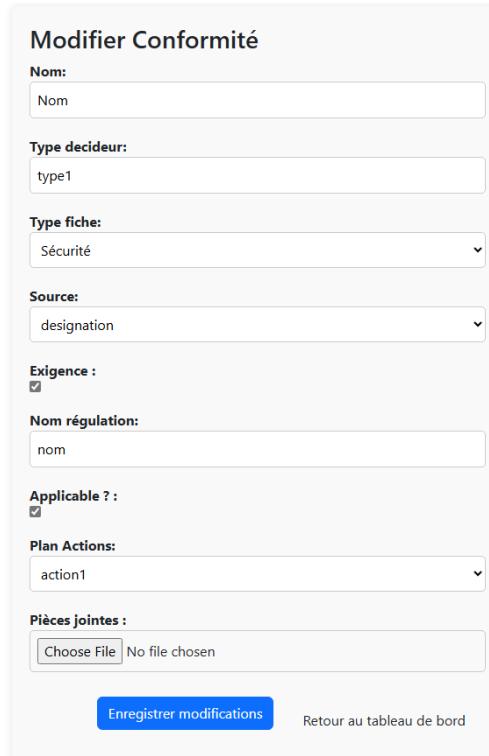
L'interface "Ajouter risque" est une page de formulaire. Le titre est "Ajouter risque". Il contient les champs suivants :

- Nom du Risque:** champ de texte.
- Date de Déclaration:** champ de date format mm/dd/yyyy.
- Déclencheur:** champ déroulant avec option "Sélectionner...".
- Liste Concernnée:** champ déroulant avec option "Sélectionner...".
- Type Risque:** champ déroulant avec option "Sélectionner...".
- Critère 1:** champ de texte avec placeholder "Sélectionner critere" et un bouton "+" pour ajouter plus d'un critère.

En bas de la page, il y a deux boutons : "Ajouter risque" (bleu) et "Retour au tableau de bord" (gris).

Figure 6.12 : Interface traiter risque

La figure 6.13 présente l'interface Modifier conformité



L'interface "Modifier Conformité" est une page de formulaire. Le titre est "Modifier Conformité". Il contient les champs suivants :

- Nom:** champ de texte avec valeur "Nom".
- Type décideur:** champ de texte avec valeur "type1".
- Type fiche:** champ déroulant avec valeur "Sécurité".
- Source:** champ déroulant avec valeur "designation".
- Exigence :** champ à cocher avec valeur checked.
- Nom régulation:** champ de texte avec valeur "nom".
- Applicable ? :** champ à cocher avec valeur checked.
- Plan Actions:** champ déroulant avec valeur "action1".
- Pièces jointes :** champ de type fichier avec bouton "Choose File" et message "No file chosen".

En bas de la page, il y a deux boutons : "Enregistrer modifications" (bleu) et "Retour au tableau de bord" (gris).

Figure 6.13 : Interface Modifier conformité

La figure 6.14 présente l'interface Conformité

Liste des Conformité réglementaire

nom_conforme: Exigence Sécurité 2028	nom_conforme: Nom
type_fiche: Sécurité	type_fiche: Sécurité
source: designation	source: designation
type_decideur: manager	type_decideur: type1
Crée par: user prenom	Crée par: user prenom
Crée à: 2024-06-02 10:13:19	Crée à: 2024-06-02 11:49:04
Modifié par: pas de modification	Modifié le : pas de modification
Pièces jointes : null	Pièces jointes : null
exigence_dec: exist	exigence_dec: exist
nom_reglementation: nom1	nom_reglementation: nom
applicable: oui	applicable: oui
plan_action: action1	plan_action: action1

nom_conforme: Exigence Sécurité 2028	nom_conforme: Nom
type_fiche: Sécurité	type_fiche: Sécurité
source: designation	source: designation
type_decideur: manager	type_decideur: type1
Crée par: user prenom	Crée par: user prenom
Crée à: 2024-06-02 10:13:19	Crée à: 2024-06-02 11:49:04
Modifié par: pas de modification	Modifié le : pas de modification
Pièces jointes : null	Pièces jointes : null
exigence_dec: exist	exigence_dec: exist
nom_reglementation: nom1	nom_reglementation: nom
applicable: oui	applicable: oui
plan_action: action1	plan_action: action1

[Modifier](#)
[Supprimer](#)

[Ajouter Conformité](#)

[Retour](#)

Figure 6.14 : Interface conformité

8 Test

Cette interface permet de tester la méthode de "Créer non conformité d'un produit"

```

POST http://localhost:8000/produit/create_Produit/
Body (raw)
{
  "date_detection": "2024-04-22",
  "designation_produit_non_conforme": "le premier produit",
  "description_non_conforme": "Description de la non-conformité...",
  "produits_non_conformes": [1],
  "type_non_conforme": 1,
  "source_non_conforme": 1,
  "niveau_gravite": 1,
  "piecesJointes": null,
  "personnes_a_notifier": [1, 2, 3]
}

```

```

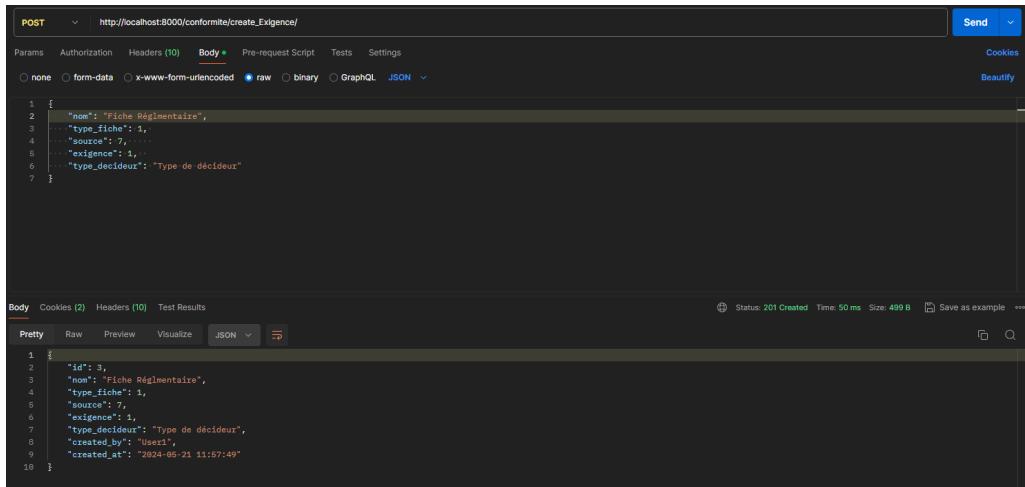
1 {
2   "id": 13,
3   "date_detection": "2024-04-22",
4   "designation_produit_non_conforme": "le premier produit",
5   "description_non_conforme": "Description de la non-conformité...",
6   "type_non_conforme": 1,
7   "source_non_conforme": 1,
8   "niveau_gravite": 1,
9   "piecesJointes": null,
10  "personnes_a_notifier": [
11    1,
12    2,
13    3
14  ],
15  "produits_non_conformes": [
16    1
17  ],
18  "created_by": "User1",
19  "created_at": "2024-05-21 11:54:33"
20 }

```

Figure 6.15 : Test méthode "Créer non conformité d'un produit"

Chapitre 6. Sprint 4 : Gestion des produits non conforme , conformité réglementaire et des risques.

Cette interface permet de tester la méthode de "Créer conformité réglementaire"



The screenshot shows a Postman interface with a POST request to `http://localhost:8000/conformite/create_Exigence/`. The request body is a JSON object:

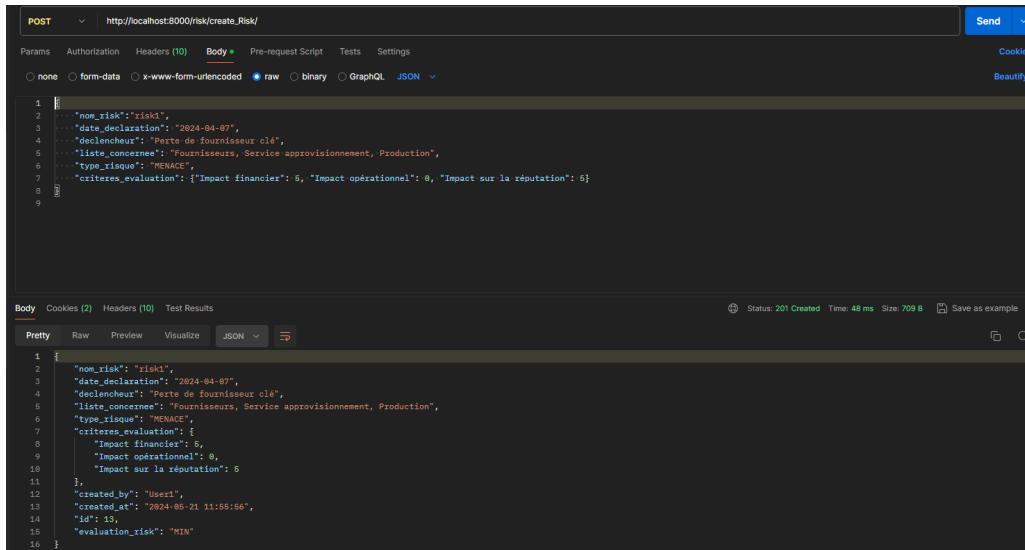
```
1 {
2     "nom": "Fiche Réglementaire",
3     "type_fiche": 1,
4     "source": 7,
5     "exigence": 1,
6     "type_decideur": "Type de décideur"
7 }
```

The response status is 201 Created, and the response body is:

```
1 {
2     "id": 3,
3     "nom": "Fiche Réglementaire",
4     "type_fiche": 1,
5     "source": 7,
6     "exigence": 1,
7     "type_decideur": "Type de décideur",
8     "created_by": "User1",
9     "created_at": "2024-06-21 11:57:49"
10 }
```

Figure 6.16 : Test méthode "Créer conformité réglementaire"

Cette interface permet de tester la méthode de "Traiter risque"



The screenshot shows a Postman interface with a POST request to `http://localhost:8000/risk/create_Risk/`. The request body is a JSON object:

```
1 {
2     "nom_risque": "risk1",
3     "date_declaration": "2024-04-07",
4     "declencheur": "Perte de fournisseur clé",
5     "liste_concerne": "Fournisseurs, Service approvisionnement, Production",
6     "type_risque": "MENACE",
7     "critères_evaluation": {"Impact financier": 8, "Impact opérationnel": 0, "Impact sur la réputation": 6}
8 }
```

The response status is 201 Created, and the response body is:

```
1 {
2     "nom_risque": "risk1",
3     "date_declaration": "2024-04-07",
4     "declencheur": "Perte de fournisseur clé",
5     "liste_concerne": "Fournisseurs, Service approvisionnement, Production",
6     "type_risque": "MENACE",
7     "critères_evaluation": [
8         {"Impact financier": 8, "Impact opérationnel": 0, "Impact sur la réputation": 6}
9     ],
10    "created_by": "User1",
11    "created_at": "2024-06-21 11:58:56",
12    "id": 13,
13    "évaluation_risque": "MIN"
14 }
```

Figure 6.17 : Test méthode "Traiter risque"

9 conclusion

En conclusion de ce sprint, nous avons réalisé des avancées significatives dans le développement de notre projet, en mettant en place des fonctionnalités clés telles que la gestion des audits, la gestion des actions et la planification des réunions. Dans notre prochain chapitre nous nous concentrerons sur nos finales fonctionnalités notamment la gestion des produit non conforme , la gestion des risques et la gestion de conformité réglementaire.

Conclusion générale

En conclusion, la conception et la mise en oeuvre d'un ERP de gestion de la qualité pour la société BMG représentent une démarche stratégique visant à répondre aux défis croissants des entreprises modernes. En intégrant les normes rigoureuses du Système de Management de la Qualité (SMQ) tel que l'ISO 9001, notre projet vise à améliorer de manière significative la qualité des services, l'agilité, la flexibilité et la compétitivité de l'entreprise.

L'ERP que nous proposons ne se limite pas à l'automatisation des processus de gestion. Il vise à intégrer les principes de qualité et de conformité au coeur des opérations de l'entreprise. En automatisant les différents processus de gestion tels que la gestion des ressources humaines, des audits, des clients, des fournisseurs, des indicateurs, des risques, de la documentation et des non-conformités, ainsi qu'en assurant la conformité réglementaire, cet ERP permettra à BMG de gérer et d'analyser efficacement ses données, d'améliorer ses processus de gestion et d'automatiser ses activités.

Pour garantir le succès de ce projet, nous avons adopté la méthodologie Scrum, une approche Agile permettant une gestion de projet flexible et itérative. Cette méthodologie favorise la collaboration entre les équipes, l'adaptation rapide aux changements et la livraison fréquente de fonctionnalités opérationnelles. Grâce à des sprints réguliers, des revues de sprint et des rétrospectives, nous assurons une amélioration continue du produit et une réactivité accrue aux besoins des utilisateurs. L'implication active des parties prenantes à chaque étape du projet permet de s'assurer que l'ERP répond précisément aux exigences de BMG tout en restant aligné sur les objectifs stratégiques de l'entreprise.

Ainsi, l'adoption de cet ERP renforcera la réactivité, la productivité et la compétitivité de BMG. Elle permettra également de créer un environnement où la qualité est intrinsèquement liée à chaque processus opérationnel, garantissant une amélioration continue et une conformité constante aux normes de l'ISO 9001. En fin de compte, ce projet contribuera de manière significative à la croissance et au succès durable de l'entreprise.

Pour les perspectives futures, nous envisageons d'intégrer un chatbot intelligent au sein de l'ERP. Ce chatbot, doté d'une intelligence artificielle avancée, pourra assister les utilisateurs en temps réel, répondre à leurs questions, les guider dans l'utilisation du système et résoudre les problèmes courants. Cette fonctionnalité permettra d'améliorer l'expérience utilisateur, de réduire le temps de formation et de diminuer la charge de travail du support technique. De plus, le chatbot pourra collecter des données sur les interactions et les utiliser pour fournir des analyses et des suggestions visant à améliorer les processus de gestion.

Webographie

- [1] SCRUM.ORG. *Qu'est-ce que Scrum ?* URL : <https://www.scrum.org/resources/what-scrum-module>.
- [2] VIRTUAL STUDIO CODE. *Documentation de Visual Studio Code.* URL : <https://code.visualstudio.com/docs>.
- [3] ORACLE CORPORATION. *Définition de PostgreSQL.* URL : <https://www.oracle.com/fr/database/definition-postgresql/>.
- [4] DJANGO SOFTWARE FOUNDATION. *Django - The Web framework for perfectionists with deadlines.* URL : <https://www.djangoproject.com/>.
- [5] ENCODE OSS. *Django Rest Framework.* URL : <https://www.django-rest-framework.org/>.
- [6] AUTH0. *JWT.IO - JSON Web Tokens Introduction.* URL : <https://jwt.io/>.
- [7] W3SCHOOLS. *React - Introduction.* URL : https://www.w3schools.com/whatis/whatis_react.asp.
- [8] POSTMAN. *Qu'est-ce que Postman ?* URL : <https://www.postman.com/product/what-is-postman/>.
- [9] MKLAB. *StarUML - Outil de modélisation.* URL : <https://staruml.io/>.
- [10] OVERLEAF. *Pourquoi utiliser LaTeX ?* URL : <https://fr.overleaf.com/about/why-latex>.
- [11] DATASCIENTEST. *Tout savoir sur GitHub.* URL : <https://datascientest.com/github-tout-savoir>.
- [12] GOOGLE. *À propos de Gmail.* URL : <https://www.google.com/intl/fr/gmail/about/>.

Résumé

Le présent travail s’inscrit dans le cadre d’un projet de fin d’études à la Faculté des Sciences de Tunis, en vue de l’obtention du diplôme national de licence en science de l’informatique, spécialité génie logiciel et systèmes d’information. Réalisé au sein de l’organisme d’accueil BMG, ce projet a pour objectif de concevoir et développer un système de gestion de la qualité. L’application envisagée sera construite sur les technologies Django, React et PostgreSQL. L’objectif est de gérer efficacement divers aspects de l’entreprise tout en garantissant la qualité de ses opérations.

Mots clés : ERP , RH , MVC

Abstract

This project is part of a final-year project at the Faculty of Sciences of Tunis, aimed at obtaining a national license degree in Computer Science, specializing in software engineering and information systems. Conducted within the hosting organization BMG, the project aims to design and develop a quality management system. The envisioned application will be built using Django, React, and PostgreSQL technologies. The ultimate goal is to effectively manage various aspects of the enterprise while ensuring the quality of its operations.

Keywords : ERP , RH , MVC

ملخص

ويأتي هذا العمل ضمن مشروع نهاية الدراسة في الكلية للعلوم بتونس، قصد الحصول على الشهادة الوطنية للإجازة في علوم الكمبيوتر، متخصصة في هندسة البرمجيات ونظم المعلومات. يهدف هذا المشروع إلى تصميم وتطوير نظام إدارة الجودة. الهدف هو إدارة الجوانب المختلفة للأعمال بشكل فعال مع ضمان جودة عملياتها.

الكلمات المفتاحية: تخطيط موارد المؤسسات، الموارد البشرية