

République Algérienne Démocratique et Populaire
Ministère de l'enseignement Supérieur et de la Recherche Scientifique
Université Abderrahmane Mira de Bejaia



جامعة بجاية
Tasdawit n Bgayet
Université de Béjaïa

Faculté des Sciences Exactes
Département d'Informatique

Mémoire de Master Professionnel

Option : Génie Logiciel

Thème

Développement d'une plateforme de doléances d'un
système de gestion de tickets

Réalisé par

—M^{lle} DJIJELI Mylene

—M^{lle} RABIAI Katia

Devant le jury composé de :

Président : M.TOUAZI Djoudi

Encadrant : M.AMROUN Kamal

Examineur : M.MOHAMMEDI Mohamed

Co-encadrant : M.ELSAKAAN Nadim

Maître de conférences B

Professeur

Maître de conférences A

Doctorant

Université A.Mira Béjaïa

Université A.Mira Béjaïa

Université A.Mira Béjaïa

Université A.Mira Béjaïa

Promotion 2021/2022

Remerciements

En guise de reconnaissance, on tient à témoigner nos sincères remerciements toutes les personnes qui ont contribué directement ou indirectement au développement cet humble travail.

Nous tenons tout d'abord à remercier nos deux encadrants M. AMROUN Kamal et M. ELSAKAAN Nadim qui n'ont pas lésiné sur les moyens pour nous aider, orienter et conseiller.

Ce fut une expérience très enrichissante d'apprendre de nouvelles choses, méthode de travail et raisonnement. L'expérience de nos deux encadrants nous a donné Le chemin vers l'esprit de recherche et d'analyse.

Nos sincères gratitudes à M. CHIBOUTI Younes pour la qualité de son encadrement tout au long du déroulement de notre stage pour sa disponibilité, ses précieux conseils durant tout ce temps, sa rigueur et son partage d'expertise.

Nous remercions les membres du jury d'avoir fait l'effort d'examiner notre projet, et d'avoir pris le temps de souligner chaque erreur et de nous avoir conseiller pour l'avenir.

Nous exprimons nos profonds remerciements à nos deux familles, et plus particulièrement à nos parents qui sont le pilier de toute notre vie et qui au goût du sacrifice ont fait de nous ce que nous sommes aujourd'hui.

Sur la même venelle, nous voudrions expliciter notre reconnaissance envers nos amis et toutes les personnes qui ont contribué à cette finalité de près ou de loin.

À tous ces intervenants, nous présentons nos plus sincères respect et gratitude.

Dédicaces

A ma très chère mère

Nulle dédicace ne saurait être assez diserte pour te formuler toute mon affection et mon incommensurable gratitude. Tu m'a élevé et protégé contre vents et marées et m'a soutenu de ma naissance jusqu'à l'âge mûr, en peignant de mille joies l'horizon de ma vie.

Ta prière et ta bénédiction m'ont été d'un grand secours tout au long de ma vie. Quoique je puisse dire et écrire, je ne pourrais exprimer ma profonde reconnaissance. J'espère ne jamais te décevoir, ni trahir ta confiance et tes sacrifices.

Puisse dieu tout puissant, te préserver et t'accorder santé, longue vie et bonheur.

A ma très chère grand-mère

Je voudrais te remercier pour ton amour, ta générosité, ta compréhension et ton soutien fut une lumière dans tout mon parcours. Aucune dédicace ne saurait exprimer l'amour, l'estime et le respect que j'ai toujours pour toi. Puisse Allah te protéger.

A mon cher oncle

Je dédie ce travail marquant de ma vie à toi cher oncle. Tu as été et tu seras toujours un exemple pour moi par tes qualités humaines, ta persévérance et perfectionnisme. En témoignage de brut d'années de sacrifices, de sollicitudes, d'encouragement. En ce jour, j'espère réaliser l'un de tes rêves. Que ce travail soit l'expression de mon estime pour toi et que Dieu te protège.

A ma très chère tante

Ma chère tante adorable Karima qui m'a toujours motivé sans cesse a devenir meilleur et servi d'exemple.

A mon cher frère, cousin et chère cousine

A mon très cher frère SAID, cher cousin SALES et chère cousine IAETICIA. Vous occupez une place particulière dans mon coeur. Je vous dédie ce travail en vous souhaitant un avenir radieux, plein de bonheur et de succès.

A mes chers ami(e)s

À mes cher(e)s ami(e)s YOUCEF, AIDA, FELLA, DYLIA, KHAOULA, SAADI. Pour votre appui, votre assistance sans faille et pour vos encouragements sincères.

A ma très chère binôme

À ma binôme MYLENE, compagne de cette aventure pédagogique. Partager ces moments d'étude et de travail fut mémorable.

KATIA.

Dédicaces

Je dédié ce travail à :

Ma chère mère NOURA qui à su me donner tout l'amour dont j'ai toujours eu besoin, qui a su m'apporter tout le soutien qu'il me fallait et qui a consenti tant de sacrifices.

Ma chère maman, je te remercie pour ton assistance quotidienne, pour tes précieux conseils pour ta présence qui m'a tant apporté et reçoit à travers ce travail aussi modeste soit-il, l'expression de mes sentiments et de mon éternelle gratitude.

Mon cher père MADJID qui peut être fier et trouver ici le résultat de ces longues années de sacrifices et de privations pour m'aider à avancer dans la vie. Puisse Dieu faire en sorte que ce travail porte son fruit de tous ces sacrifices. Merci pour les valeurs nobles, l'éducation et le soutien permanent venu de toi cher papa.

A mes adorables soeurs Melissa, Manel, Nermine qui n'ont pas cessé de m'aider par tous les moyens dans ma vie. A tous les membres de ma famille, petits et grands.

Mon cher chien MILOU ,mon soutien moral ma source de bonheur qui a su faire preuve de patience et de sagesse, en étant toujours à mes cotés et en m'aidant dans la gestion de mon stress.

A tout mes amis (es)qui ont été là à me soutenir tout au long de cette année.

A ma binôme KATIA à qui je souhaite plein de succès et de réussite.

A tout ceux que j'aime, ceux qui sont proches à mon coeur.

Mylene.

Table des matières

Table des matières	i
Liste des tableaux	iii
Table des figures	iv
Liste des acronymes	v
Introduction générale	1
I Présentation de l'organisme d'accueil et objectifs du projet	3
I.1 Introduction	3
I.2 Présentation de l'organisme d'accueil	3
I.2.1 Présentation de l'entreprise Cevital	3
I.2.2 Activité et mission	3
I.2.3 Les facteurs clés du succès de Cevital	4
I.2.4 Objectifs de Cevital	5
I.2.5 Organigramme de l'organisme d'accueil	5
I.3 Présentation du projet	7
I.3.1 Contexte du projet	7
I.3.2 Problématique	7
I.3.3 Objectifs	8
I.4 Conclusion	8
II Spécification et Analyse des besoins	9
II.1 Introduction	9
II.2 Méthodologie de développement	9
II.2.1 Processus Unifié	9
II.2.2 Les caractéristiques du processus unifié	10
II.2.3 Expérience Utilisateur	10
II.2.4 Les piliers d'expérience utilisateur	11
II.2.5 Les étapes d'expérience utilisateur	11
II.3 Langage de modélisation	12
II.3.1 UML(Unified Modeling Language)	12
II.4 Spécification des besoins	13
II.4.1 Les besoins fonctionnelles	13
II.4.2 Les besoins non fonctionnelles	13
II.5 Analyse des besoins	14
II.5.1 Identification des acteurs	14
II.5.2 Identification des cas d'utilisations	14

II.5.3 Diagrammes des cas d'utilisation et leurs descriptions	15
II.6 Diagramme de séquence	30
II.6.1 Diagramme de séquence du cas d'utilisation«Authentification» . . .	30
II.6.2 Diagramme de séquence du cas d'utilisation«Ajouter un compte» .	31
II.6.3 Diagramme de séquence du cas d'utilisation«Modifier un compte» .	32
II.6.4 Diagramme de séquence du cas d'utilisation«Supprimer un compte»	33
II.6.5 Diagramme de séquence du cas d'utilisation«Ajouter un ticket» . .	34
II.6.6 Diagramme de séquence du cas d'utilisation«Modifier un ticket» . .	35
II.6.7 Diagramme de séquence du cas d'utilisation«Supprimer un ticket» .	36
II.6.8 Diagramme de séquence du cas d'utilisation«Traiter un ticket» . . .	37
II.6.9 Diagramme de séquence du cas d'utilisation«Affecter un ticket» . .	38
II.6.10 Diagramme de séquence du cas d'utilisation«Ajouter une catégorie»	39
II.6.11 Diagramme de séquence du cas d'utilisation«Modifier une catégorie»	40
II.6.12 Diagramme de séquence du cas d'utilisation«Supprimer une catégorie»	41
II.7 Conclusion	42
III Conception	43
III.1 Introduction	43
III.1.1 BPMN (Business Process Model and Notation)	43
III.2 Diagramme de classe	45
III.3 Modèle relationnel	46
III.4 Conclusion	46
IV Réalisation	47
IV.1 Introduction	47
IV.2 Langages et environnements de développement	47
IV.2.1 Langage de développement	47
IV.2.2 Environnement de développement	48
IV.2.3 Framework utilisé	49
IV.2.4 Les Systèmes de gestion de base de données relationnelles (SGBDR)	50
IV.3 Captures d'écran de l'application	50
IV.4 Conclusion	58
V Conclusion générale et prespective	59
Bibliographie	60

Liste des tableaux

II.1	Description textuelle du cas d'utilisation «Authentification».	21
II.2	Description textuelle du cas d'utilisation «Créer un ticket».	22
II.3	Description textuelle du cas d'utilisation «Modifier un ticket».	22
II.4	Description textuelle du cas d'utilisation «Supprimer un ticket».	23
II.5	Description textuelle du cas d'utilisation «Modifier un profil».	24
II.6	Description textuelle du cas d'utilisation «Consulter la liste des tickets».	25
II.7	Description textuelle du cas d'utilisation «Traiter les tickets».	25
II.8	Description textuelle du cas d'utilisation «Affecter un ticket».	26
II.9	Description textuelle du cas d'utilisation «Gérer les comptes».	27
II.10	Description textuelle du cas d'utilisation « S'affecter les tickets »	28
II.11	Description textuelle du cas d'utilisation «Gérer les catégories».	29

Table des figures

I.1	Organigramme de l'entreprise Cévital [2].	6
I.2	Direction Système d'Information [2].	7
II.1	Les phases du UP [3].	10
II.2	Les piliers d'UX [7].	11
II.3	Les étapes d'UX [8].	12
II.4	Les différentes diagrammes d'UML [9].	13
II.5	Diagramme de cas d'utilisation global.	16
II.6	Diagramme de cas d'utilisation «Demandeur».	17
II.7	Diagramme de cas d'utilisation «Technicien».	18
II.8	Diagramme de cas d'utilisation «Responsable IT».	19
II.9	Diagramme de cas d'utilisation «Responsable métier».	20
II.10	Diagramme de séquence du cas d'utilisation«Authentification».	30
II.11	Diagramme de séquence du cas d'utilisation«Ajouter un compte».	31
II.12	Diagramme de séquence du cas d'utilisation«Modifier un compte».	32
II.13	Diagramme de séquence du cas d'utilisation«Supprimer un compte».	33
II.14	Diagramme de séquence du cas d'utilisation«Ajouter un ticket».	34
II.15	Diagramme de séquence du cas d'utilisation«Modifier un ticket».	35
II.16	Diagramme de séquence du cas d'utilisation«Supprimer un ticket».	36
II.17	Diagramme de séquence du cas d'utilisation«Traiter un ticket».	37
II.18	Diagramme de séquence du cas d'utilisation«Affecter un ticket».	38
II.19	Diagramme de séquence du cas d'utilisation«Ajouter une catégorie».	39
II.20	Diagramme de séquence du cas d'utilisation«Modifier une catégorie».	40
II.21	Diagramme de séquence du cas d'utilisation«Supprimer une catégorie».	41
III.1	Diagramme BPMN.	44
III.2	Diagramme de classe.	45
IV.1	Achitecture MVC [22].	48
IV.2	Interface d'Authentification.	50
IV.3	Interface Technicien.	51
IV.4	Interface Responsable IT.	52
IV.5	Interface liste des utilisateurs.	53
IV.6	Interface formulaire d'ajout.	54
IV.7	Interface de modification.	55
IV.8	Interface Responsable Métier.	56
IV.9	Interface la liste des tickets.	57
IV.10	Interface Modifier compte.	58

Liste des acronymes

BDD	Base De Données
BPMI	Business Process Management Initiative
BPMN	Business Process Model and Notation
CSS	Cascading Style Sheets
HTML	Hyper Text Markup Language
IT	Information Technology
JS	JavaScript
JPEG	Joint Photographic Experts Group
MVC	Model-View-Controller
OMG	Object Management Group
PHP	Hypertexte Preprocessor
PNG	Portable Network Graphics
SQL	structured Query language
UML	Unifed Modeling Language
UP	Unified Process
UX	User Experience
VSC	Visuel Studio Code
XHTML	EXtensible HyperText Markup Language

Introduction générale

Notre époque actuelle se justifie par l'essor de la technologie, qui s'est emparée de divers domaines, ou ce que l'on appelle les outils numériques, qui ont touché les champs professionnels.

Les outils numériques sont apparus pour faciliter le travail et fournir des informations fluides. Ils ont évolué à partir des systèmes d'information qui peuvent être définis comme un ensemble organisé de ressources [1].

Le système d'information joue un rôle important dans les domaines professionnels et sont même indispensables à leur bon fonctionnement.

L'importance des applications métiers dans nos vies augmente avec le temps en raison des services qu'elles fournissent. Elles permettent de faciliter la gestion des activités, de simplifier les tâches et d'automatiser les processus des utilisateurs au sein de l'entreprise.

Cependant, les entreprises, organisations et les associations d'aujourd'hui ont de la difficulté à trouver des solutions appropriées à leurs problèmes. les choses ne se passent pas toujours comme prévu. En fin de compte, anticiper les besoins des employés n'est pas toujours facile. Par conséquent, il est très important d'établir un mécanisme efficace de gestion des doléances ou ce qu'on appelle gestion des tickets.

A l'issue du cursus Master, l'objectif principal de ce projet est de créer une plateforme de réclamations basée sur le système de gestion des tickets, qui permettra aux employés la soumission sécurisée des demandes, réclamations et demandes d'augmentation.

Notre mémoire sera divisé en quatre chapitres :

Dans le premier chapitre intitulé «**Présentation de l'organisme d'accueil et objectifs du projet**», nous définirons l'organisme d'accueil puis la présentation du projet et nous extrairons la problématique.

Dans le deuxième chapitre intitulé «**Analyse et spécification des besoins**», nous allons déterminer les acteurs de notre système et l'analyse des besoins.

Dans le troisième chapitre qui concerne «**La conception**», nous présenterons la solution que nous souhaitons développer, ainsi que la manière dans laquelle notre système devra être déployé.

Dans le dernier chapitre dédié à «**La réalisation**» de notre plateforme, nous présenterons dans un premier lieu l'environnement de développement et les différents outils et langages de programmation utilisés, ensuite nous exposerons des captures d'écran qui représentent les interfaces de notre application.

Enfin, nous clôturerons ce mémoire par une conclusion générale tout en donnant quelques perspectives que nous souhaiterons accomplir dans des travaux futurs.

Chapitre I

Présentation de l'organisme d'accueil et objectifs du projet

I.1 Introduction

Dans ce chapitre, nous allons mettre l'accent sur la présentation de l'organisme d'accueil, puis nous décrirons le contexte et nous poserons la problématique et les objectifs de notre projet.

I.2 Présentation de l'organisme d'accueil

I.2.1 Présentation de l'entreprise Cevital

Cevital est une entreprise algérienne qui a vu le jour dès l'entrée de notre pays en économie de marché en faisant passer notre pays du statut d'importateur à celui d'exportateur. Elle a été créée par des fonds privés en 1998. Elle est implantée à l'extrême Est du port de Béjaïa.

Elle est l'un des fleurons de l'industrie agro-alimentaire en Algérie qui est constituée de plusieurs unités de production équipées de la dernière technologie et poursuit son développement par divers projets en cours de réalisation. Son expansion et son développement durant les 5 dernières années, font d'elle un important pourvoyeur d'emplois et de richesses.

Ce méga complexe est composé concentré en première partie dans le secteur de l'agro-alimentaire : Raffinage d'huile et de sucre, produits dérivés, négoce de céréales, distribution de produits destinés à l'alimentation humaine et animale.

I.2.2 Activité et mission

L'ensemble des activités de Cevital est concentré sur la production des huiles végétales et de margarine et se présente comme suit :

- Raffinage d'huile.
- Conditionnement d'huile.

- Production de margarine.
- Fabrication d'emballage en P.E.T.
- Raffinerie de sucre en cours de réalisation.
- Stockage céréales en cours de réalisation.
- Minoterie en cours d'étude.
- Savonnerie en cours d'étude.

Les huiles de Cevital disponibles sur le marché sont :

- Fleurial : 100 tournesol commercialisée depuis août 1999.
- Soya : 100 soya commercialisée depuis septembre 1999.
- Canola : 100 colza commercialisée depuis fin septembre 1999.
- Oleol : 100 tournesol commercialisée depuis début avril 2000.
- Olivia : huile d'olive raffinée.

I.2.3 Les facteurs clés du succès de Cevital

Les facteurs clés du succès de l'entreprise Cevital sont :

- Capacité à manager des projets : dans la production et la distribution de grandes envergures.
- Maîtrise de la technologie : les unités industrielles utilisent les dernières innovations en matière d'automatisation des procès.
- Jeunesse des salariés : encadrement à fort potentiel pour assurer une gestion pérenne de l'entreprise.
- Choix du site : l'avantage de la localisation est un facteur clé du succès, car il représente un avantage compétitif de taille sur le plan logistique, c'est-à-dire la proximité des installations portuaires réduisant les coûts des matières importées et des produits finis exportés.
- Force de négociation : la taille de l'entreprise, les parts de marchés investis, une comparaison taux entreprises qui évolue dans les mêmes secteurs d'activité.

- Présence d'un réseau de distribution : couvrant l'ensemble du territoire national.

I.2.4 Objectifs de Cevital

Cevital de Béjaïa a pour objectifs :

- L'extension de ses produits sur tout le territoire national.
- L'importation des graines oléagineuses pour l'extraction directe des huiles brutes.
- L'encouragement des agriculteurs par des aides financières pour la production de ces graines localement.
- L'optimisation de ses offres d'emploi sur le marché du travail.
- La modernisation des installations en termes de marché étranger et leur exportation.
- La réponse aux besoins du client.
- L'accroissement du chiffre d'affaire.
- L'amélioration de ses relations de partenariat étrangères.

I.2.5 Organigramme de l'organisme d'accueil

La figure suivante présente l'organigramme de l'entreprise Cévital :

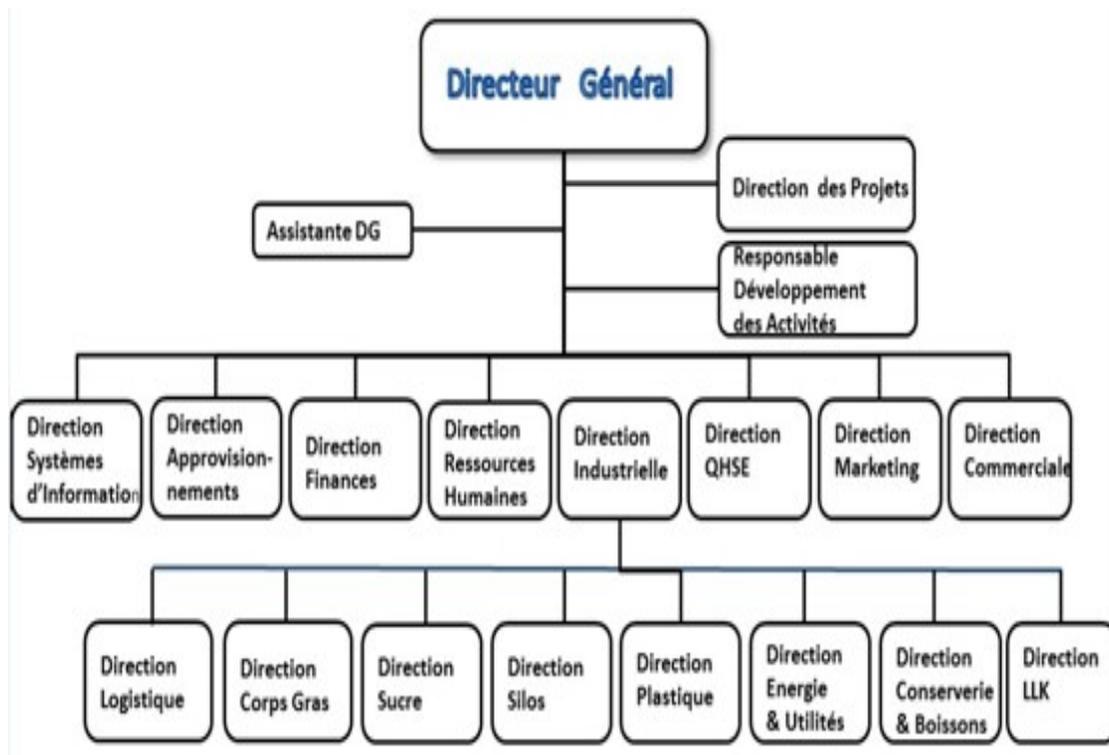


FIGURE I.1 – Organigramme de l'entreprise Cévital [2].

- **Direction des Ressources Humaines** : Cette direction a pour rôle de définir et proposer à la direction générale les principes de gestion ressources humaines en support avec les objectifs du business. Elle assure le recrutement et la gestion des carrières. Elle se charge de la formation du personnel et participe avec la direction générale à l'élaboration de la politique de communication afin de développer l'adhésion du personnel aux objectifs fixés par l'organisation.

- **Direction des Finances** : Le rôle de cette direction est de préparer et mettre à jour les budgets, tenir la comptabilité et préparer les états comptables et financiers selon les normes et pratiquer le contrôle de gestion.

- **Direction Commerciale** : Elle a en charge de commercialiser toutes les gammes des produits et le développement du Fichier clients de l'entreprise, au moyen d'actions de détection ou de promotion de projets à base de hautes technologies. En relation directe avec la clientèle, elle possède des qualités relationnelles pour susciter l'intérêt des prospects.

- **Direction Système d'Informations** : Elle assure la mise en place des moyens des technologies de l'information nécessaires pour supporter et améliorer l'activité, la stratégie et la performance de l'entreprise. Elle doit ainsi veiller à la cohérence des moyens informatiques et de communication mis à la disposition des utilisateurs, à leur mise à niveau, à leur maîtrise technique et à leur disponibilité et opérationnalité permanente et en toute sécurité.

Elle définit également dans le cadre des plans pluriannuels les évolutions nécessaires en fonction des objectifs de l'entreprise et des nouvelles technologies.

La figure suivante présente la direction système information :

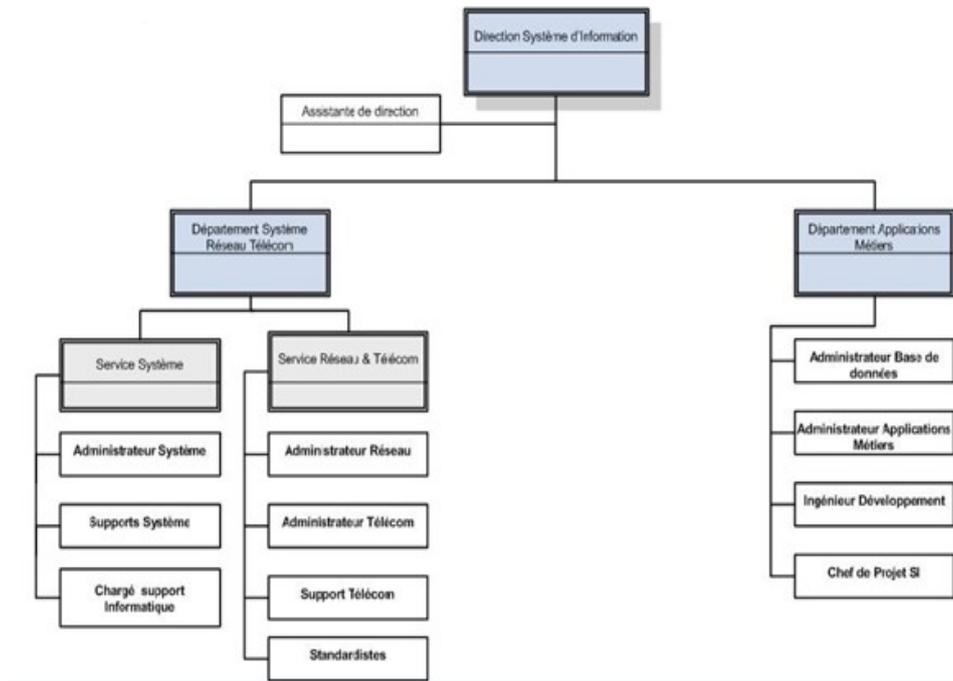


FIGURE I.2 – Direction Système d'Information [2].

I.3 Présentation du projet

I.3.1 Contexte du projet

Tous les métiers de l'entreprise sont différents mais partagent une problématique in-
muable, satisfaire leurs employés et leurs exigences, qu'ils soient internes ou externes.

Afin de gérer les demandes, les plaintes, les incidents, il est nécessaire de construire un
système de gestion des tickets, qui se charge de répondre aux demandes d'aide des em-
ployés.

I.3.2 Problématique

Durant notre stage effectué au sein de l'entreprise Cevital. Nous avons détecté un cer-
tain nombre de problèmes qui ralentissent le bon fonctionnement du service informatique.
Ces problèmes sont :

- Manque de suivi et traçabilité de matériels.
- Manque de contact entre individus.
- Possibilité de perte de données.

- Perte de temps.
- La difficulté pour attribuer les problèmes pour chaque informaticien.
- La difficulté pour faire remonter les informations exprimées par l'utilisateur aux autres techniciens.

I.3.3 Objectifs

Notre objectif est de réaliser une plateforme de doléances à base d'un ticket qui permettra de soumettre les réclamations et plaintes des utilisateurs.

Grâce à cela, il n'y aura pas de double travail, pas de perte de communication donc pas de perte de temps.

A long terme, cela nous permettrait d'améliorer notre système d'information et donc le confort de nos utilisateurs sans devoir interagir avec le personnel.

I.4 Conclusion

Tout au long de ce chapitre, nous avons donné un aperçu du projet en décrivant l'organisme d'accueil et le contexte du projet. Puis, nous avons imposé problématique et donné un aperçu sur l'objectif principal de notre projet.

Le chapitre suivant sera consacré à la spécification et analyse des besoins.

Chapitre II

Spécification et Analyse des besoins

II.1 Introduction

Dans ce chapitre, nous allons définir les démarches de développement. Ensuite, nous spécifions les besoins en définissant les différents besoins fonctionnels et non fonctionnels. Puis, nous identifions les acteurs de notre système et la présentation des diagrammes de cas d'utilisation ainsi que leurs descriptions textuelles. Nous terminerons par illustrer les différents diagrammes de séquences qui sont des interactions entre les acteurs et le système selon un ordre chronologique dans la formulation UML.

II.2 Méthodologie de développement

II.2.1 Processus Unifié

Processus Unifié est un processus de développement logiciel itératif permettant de décrire les besoins, les exigences des utilisateurs et qui consiste à diviser le projet en parties et pour chaque partie on effectue les étapes suivantes :

-Analyse des besoins : Déterminer les spécifications des besoins et les exigences des utilisateurs.

-Conception : La phase de conception se base sur le langage UML, c'est la phase créative d'un projet. Le but de la conception est de permettre de créer un système ou un processus répondant à un besoin spécifique.

-Réalisation : C'est la phase de développement et construction de produit.

-Test et validation : C'est la dernière tâche permet de tester le module développé et le validé [3] [4].

II.2.2 Les caractéristiques du processus unifié

Le processus unifié est une méthode de développement de logiciel caractérisée par :

-Itératif et incrémental : Les itérations se succèdent dans un ordre logique permettant de donner lieu à un incrément et donc d'établir un développement plus optimisé et efficace.

-Centré sur l'architecture : Tout système complexe doit être décomposé en partie modulaire afin d'en faciliter la maintenance et l'évolution. Cette architecture (fonctionnelle, logique, matérielle, etc) doit être modéliser en UML, et pas seulement documentée en texte.

-Guidé par les cas d'utilisation d'UML : Le modèle des cas d'utilisations guide le processus unifié et décrit les fonctionnalités du système.

-Piloté par les risques : Les risques majeurs du projet doivent être identifié au plus tôt mais surtout levés le plus rapidement. Les mesures à prendre dans ce cadre déterminent l'ordre des interactions [3] [4].

La figure suivante présente les phases du UP :

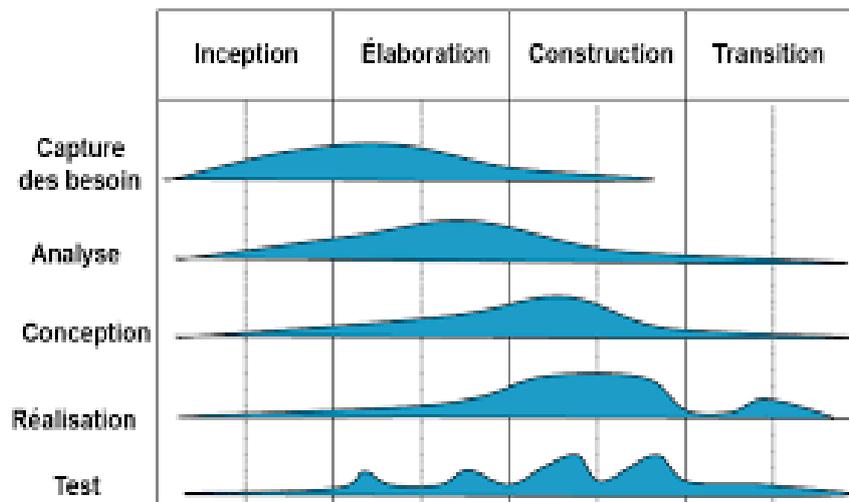


FIGURE II.1 – Les phases du UP [3].

II.2.3 Expérience Utilisateur

L'expérience utilisateur est un processus de conception centré sur l'utilisateur. Cette approche fait référence au sentiment que ressentent les utilisateurs lors de l'utilisation d'un produit, d'une application, d'un système ou d'un service. Il s'agit d'un terme large qui peut couvrir n'importe quoi, de la facilité avec laquelle l'utilisateur peut naviguer dans le produit, de sa facilité d'utilisation et de la pertinence du contenu affiché [5].

II.2.4 Les piliers d'expérience utilisateur

L'expérience utilisateur se base sur six piliers :

- **Utile** : Le contenu est original et répond à un besoin.
- **Utilisable** : Le site est facile à utiliser.
- **Désirable** : Les éléments servent l'émotion et la satisfaction de l'utilisateur.
- **Trouvable** : Le contenu est navigable et localisable (sur le site et hors du site).
- **Accessible** : Le contenu est accessible aux personnes handicapées.
- **Crédible** : Les utilisateurs peuvent faire confiance [6] [7].

La figure suivante présente les piliers d'UX :

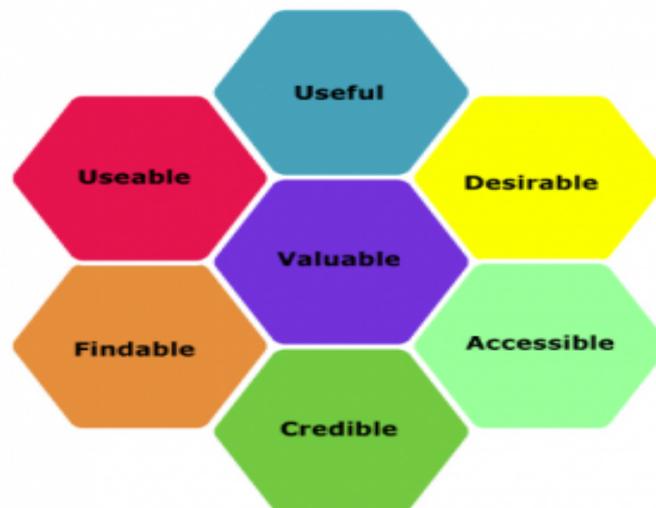


FIGURE II.2 – Les piliers d'UX [7].

II.2.5 Les étapes d'expérience utilisateur

Les différentes étapes de la méthode UX sont schématisées en 5 étapes principales. Ils assurent la réalisation de la vision de la meilleure expérience utilisateur et contribuent à l'amélioration des produits et services associés [8].

La figure suivante présente les étapes d'UX :

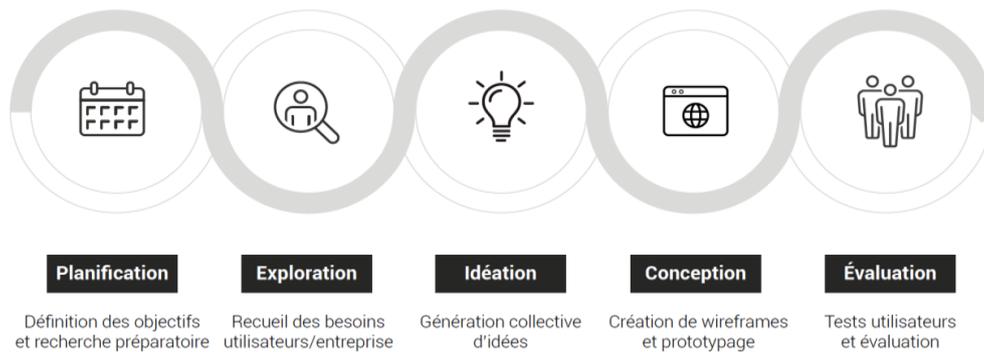


FIGURE II.3 – Les étapes d'UX [8].

II.3 Langage de modélisation

II.3.1 UML(Unified Modeling Language)

UML est un langage de modélisation graphique et textuel conçu pour comprendre et décrire les exigences en développant différents diagrammes pour modéliser toutes les données et tous les traitements.

UML est construit autour de 13 diagrammes, répartis en deux catégories : Les diagrammes structurels et les diagrammes comportementaux [9].

L'ensemble des 13 types de diagrammes UML peuvent ainsi être résumé sur la figure ci- dessous :

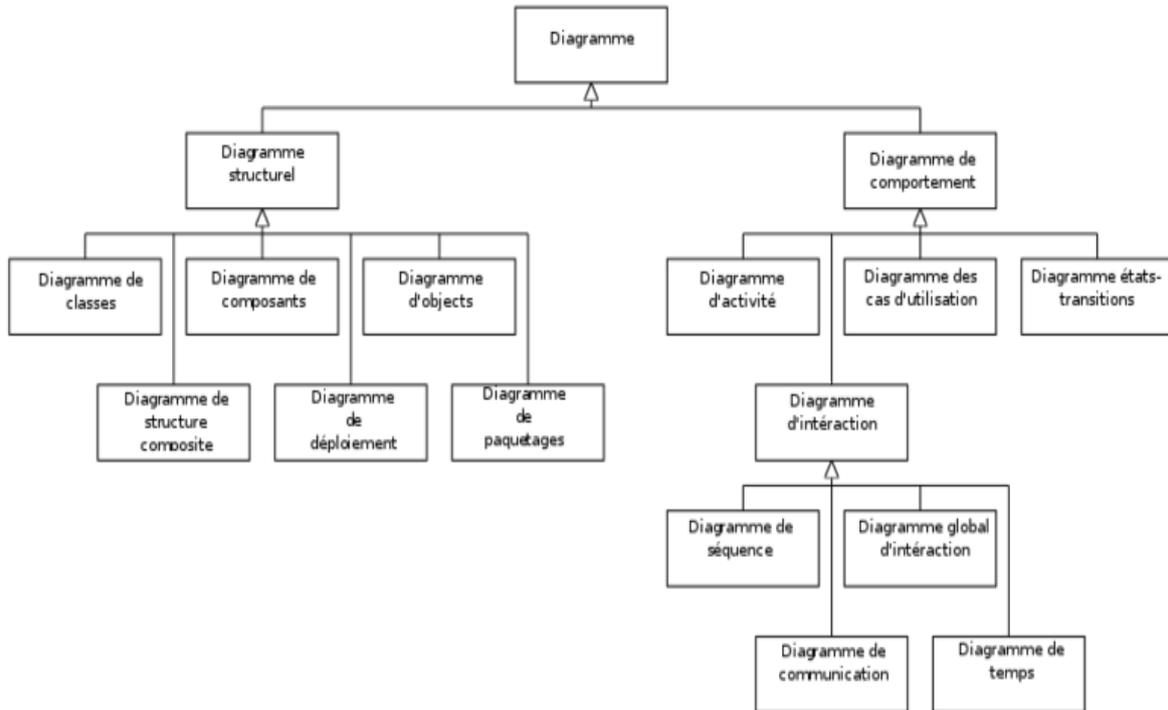


FIGURE II.4 – Les différents diagrammes d’UML [9].

II.4 Spécification des besoins

II.4.1 Les besoins fonctionnelles

Les besoins fonctionnels expriment une action que le système doit effectuer en réponse à une demande [10].

Les besoins fonctionnels de notre plateforme sont les suivants :

- Chaque utilisateur doit avoir un identifiant et un mot de passe unique pour accéder à cette plateforme.
- Chaque utilisateur peut mettre à jour ses informations personnelles .
- Le demandeur peut créer des tickets.
- Le technicien peut consulter et traiter les tickets.
- Le responsable métier peut gérer les catégories des tickets.
- Le responsable IT peut gérer les utilisateurs et affecter les tickets

II.4.2 Les besoins non fonctionnelles

Une fois les besoins fonctionnels sont bien définis, les besoins non fonctionnels doivent être pris en compte tout au long du processus de développement de la plateforme à savoir :

- **Extensibilité** : Avoir une possibilité d'ajouter ou de modifier de nouvelles fonctionnalités.
- **Efficacité** : Le temps de réponse doit être en temps réel.
- **Disponibilité** : L'accès à la plateforme doit se faire à tout moment.
- **La sécurité** : L'utilisateur doit s'authentifier avant toute tâche pour avoir l'autorisation d'accès à l'interface qui lui convient, et cela par la saisie obligatoire d'un identifiant et mot de passe unique.

II.5 Analyse des besoins

II.5.1 Identification des acteurs

Un acteur est un rôle joué par une personne ou une entité externe qui interagit avec le système [11].

Notre plateforme fait intervenir les 4 acteurs principaux :

- **Demandeur** : Représente toute personne qui peut créer, consulter et supprimer les tickets.
- **Technicien** : C'est une personne qui peut consulter la liste des tickets et les traiter.
- **Responsable IT** : Est le responsable de gérer les utilisateurs.
- **Responsable métier** : Est le responsable de la gestion des catégories de tickets.

II.5.2 Identification des cas d'utilisations

Un cas d'utilisation est une description d'un ensemble de séquences d'actions qu'un système exécute pour produire un résultat observable pour un acteur.

Il représente une exigence fonctionnelle de l'ensemble du système [11].

Nous allons spécifier pour chaque acteur ses cas d'utilisations :

1. Le demandeur

Il a le rôle de :

- Inscrire sur la plateforme.
- Créer, modifier et supprimer ses tickets.
- Consulter la liste de ses tickets.
- Gérer son profil.

2. Le technicien

Il a le rôle de :

- Inscrire sur la plateforme.
- Consulter les liste des tickets.
- Traiter les tickets.

- Clôturer les tickets.

3. Le responsable métier

Il a le rôle de :

- Inscrire dans la plateforme.
- Gérer la gestion des catégories de tickets.

4. Le responsable IT

Il a le rôle de :

- Inscrire dans la plateforme.
- Gérer les utilisateurs.
- Affecter les tickets.

II.5.3 Diagrammes des cas d'utilisation et leurs descriptions

1. Définition

Un diagramme de cas d'utilisation est un diagramme UML qui décrit le comportement fonctionnel d'un système logiciel.

Il représente une unité discrète d'interaction entre un utilisateur et le système [11].

2. Diagramme de cas d'utilisation global

Après avoir identifié les acteurs, le diagramme de cas d'utilisation général donne une vision globale du comportement fonctionnel de notre plateforme.

La figure suivante présente le diagramme de cas d'utilisation global :

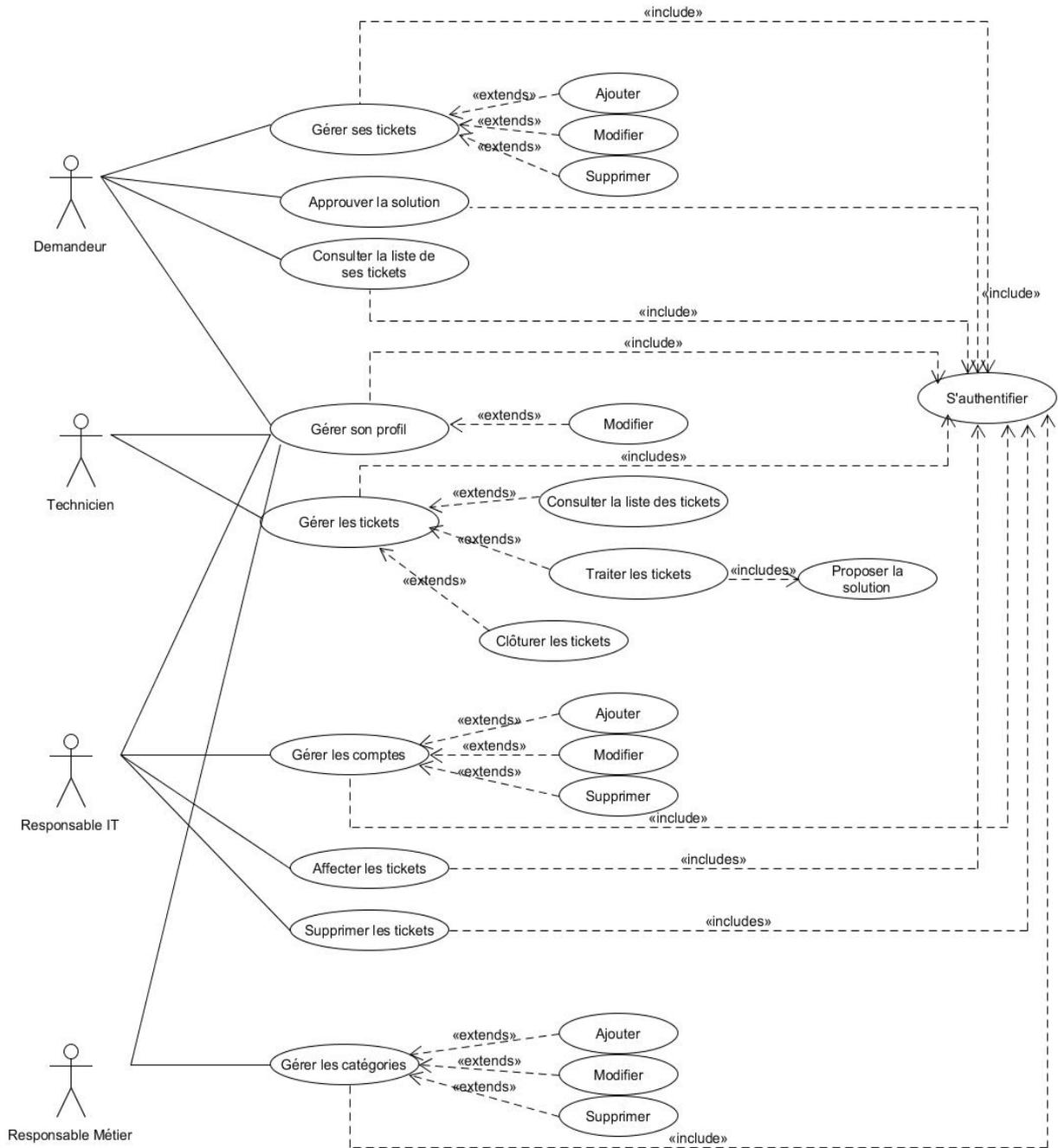


FIGURE II.5 – Diagramme de cas d'utilisation global.

Les diagrammes des cas d'utilisation pour chaque acteur principal :

3. Diagramme de cas d'utilisation «Demandeur»

Diagramme de cas d'utilisation suivant expose que le demandeur peut modifier son profil et gérer ses tickets, avant tout ça l'utilisateur doit être authentifié. Il a le droit de consulter la liste des tickets.

La figure suivante présente le diagramme de cas d'utilisation "Demandeur" :

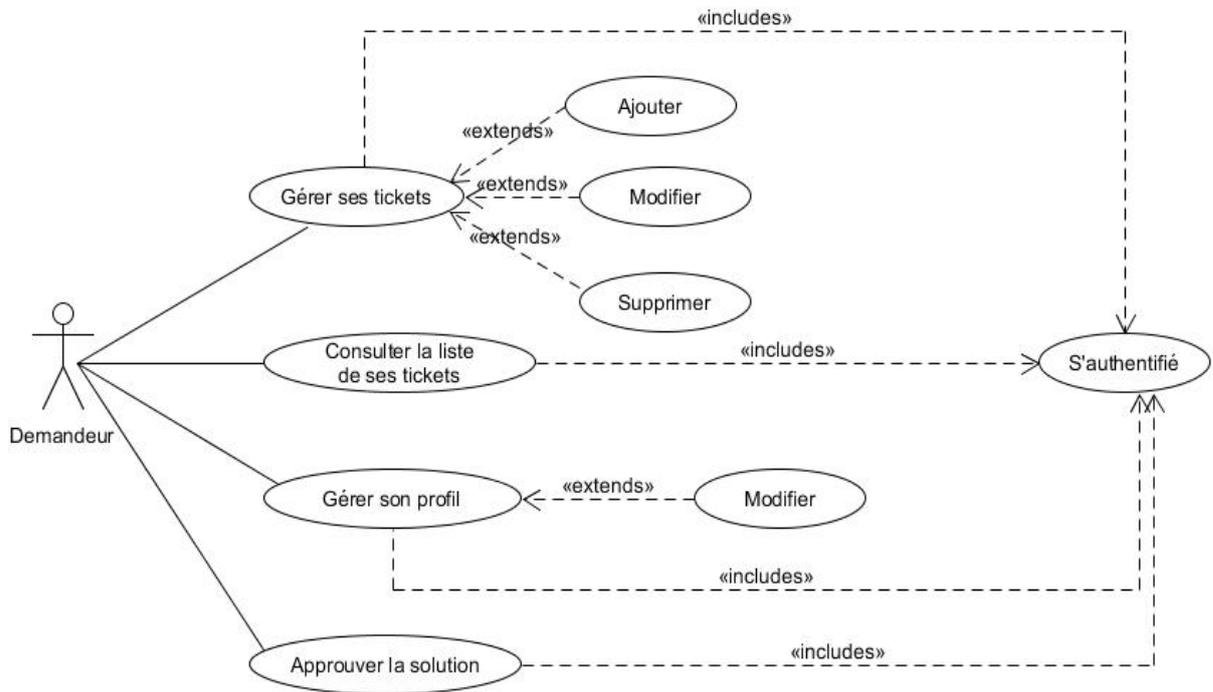


FIGURE II.6 – Diagramme de cas d'utilisation «Demandeur».

4. Diagramme de cas d'utilisation «Technicien»

Diagramme de cas d'utilisation suivant expose que le technicien peut consulter les tickets et les traiter, avant tout ça il doit être authentifié. Il peut aussi modifier son profil.

La figure suivante présente le diagramme de cas d'utilisation "Technicien" :

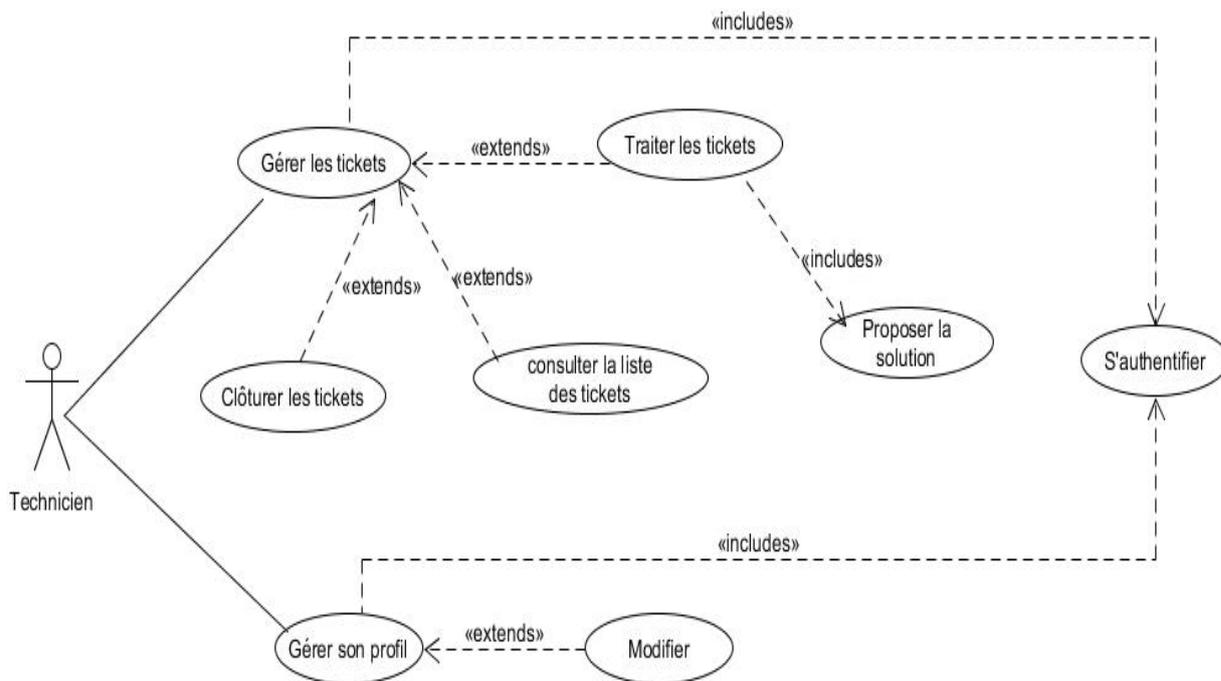


FIGURE II.7 – Diagramme de cas d'utilisation «Technicien».

5. Diagramme de cas d'utilisation «Responsable IT»

Diagramme de cas d'utilisation suivant montre que le responsable peut gérer les compte : Ajouter, modifier et supprimer des comptes avant tout ça il doit être authentifié.

La figure suivante présente le diagramme de cas d'utilisation "Responsable IT" :

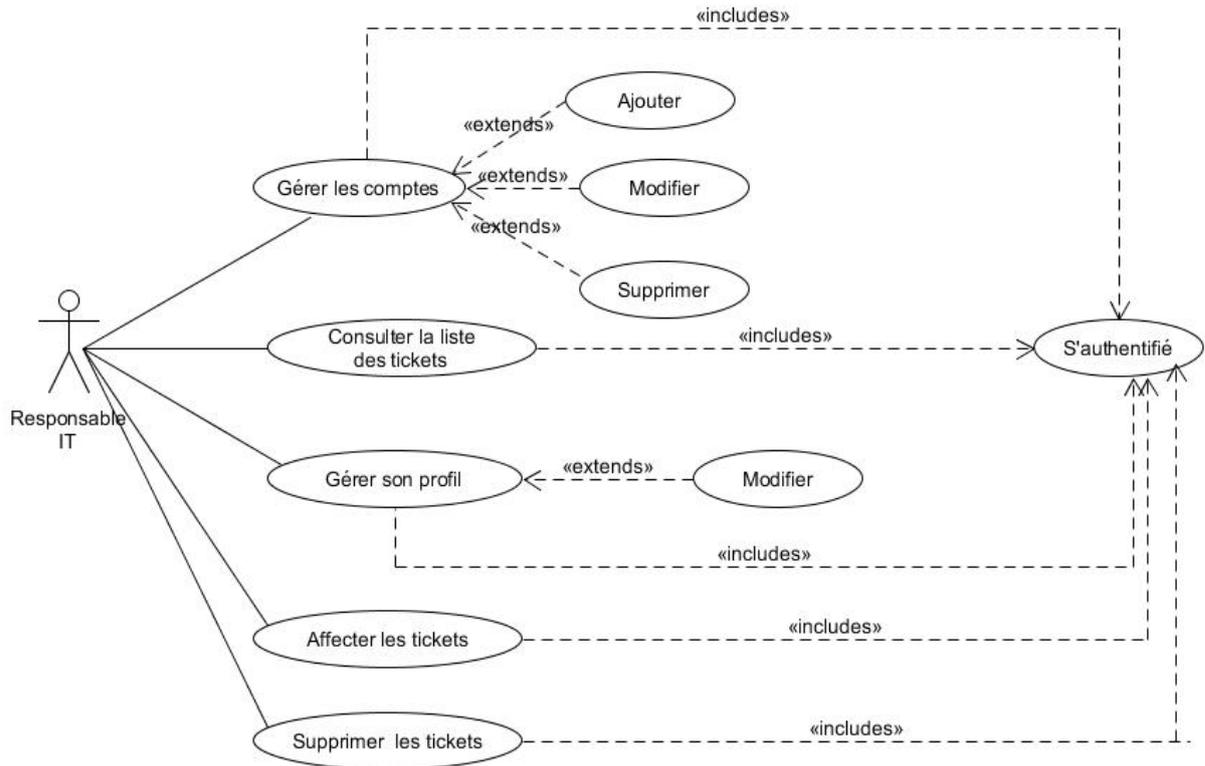


FIGURE II.8 – Diagramme de cas d'utilisation «Responsable IT».

6. Diagramme de cas d'utilisation «Responsable métier»

Diagramme de cas d'utilisation suivant montre que le responsable peut gérer les catégories de tickets, avant tout ça il doit être authentifié.

La figure suivante présente le diagramme de cas d'utilisation "Responsable métier" :

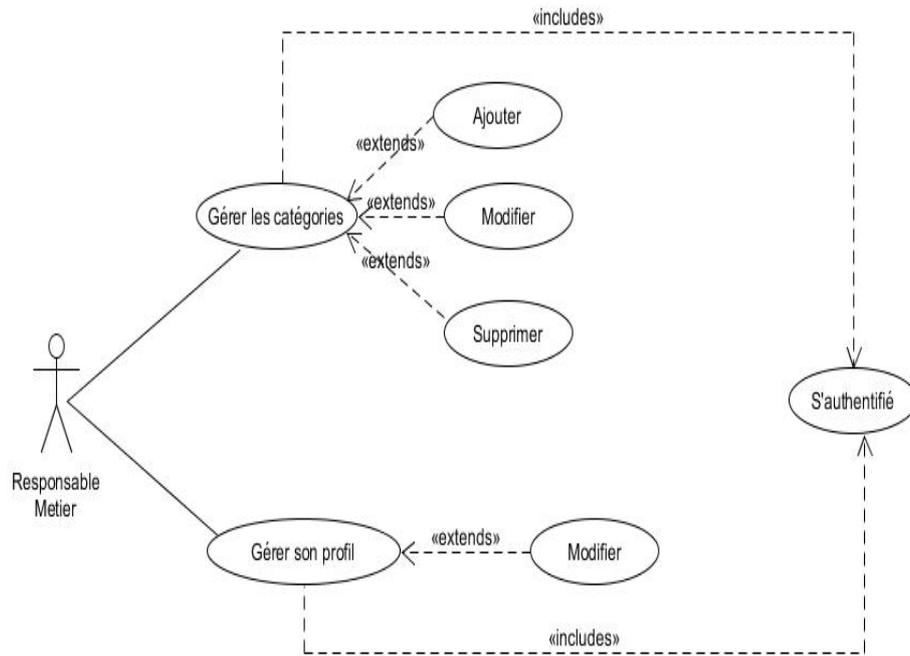


FIGURE II.9 – Diagramme de cas d'utilisation «Responsable métier».

7. Description textuelle de chaque cas d'utilisation

Elle est organisée de la façon suivante :

- **Un sommaire d'identification** : Va résumer les propriétés du cas d'utilisation.
- **Une description des enchaînements** : Pré conditions au déclenchement du cas d'utilisation doivent être spécifiées, un scénario nominal décrivant celui-ci additionné à des scénarios alternatifs et d'exceptions [16].

(a) Cas d'utilisation «Authentification»

Titre	Authentification.
Acteur	Tous les acteurs.
Description	Permet aux utilisateurs de s'authentifier pour accéder à notre système.
Scenario nominal	<ul style="list-style-type: none"> — L'utilisateur saisit son email et son mot de passe. — Le système vérifie les informations introduites. — Le système affiche l'interface approprié pour chaque utilisateur.
Scenario alternatif	<p>Un champ de saisie est vide alors, le système indique qu'un champ est incomplet et demande de ressaisir.</p> <p>Email ou mot de passe incorrect, alors Le système affiche un message d'erreur et le retour vers la page d'authentification sera effectué.</p>
Pré condition	L'utilisateur doit être connecter et avoir un compte.

TABLE II.1 – Description textuelle du cas d'utilisation «Authentification».

(b) Cas d'utilisation «Créer un ticket»

Titre	Créer un ticket.
Acteur	Demandeur.
Description	Le demandeur a la possibilité de créer des tickets.
Scenario nominal	<ul style="list-style-type: none"> — Le demandeur s'authentifie. — Il demande de créer un ticket. — Le système affiche un formulaire. — Le demandeur saisie les informations et valider. — Le système enregistre puis affiche un message de confirmation.
Scenario alternatif	Un champ de saisie est vide ,alors le système indique qu'un champ est incomplet et demande de ressaisir.
Pré condition	Le demandeur doit s'authentifier.

TABLE II.2 – Description textuelle du cas d'utilisation «Créer un ticket».

(c) Cas d'utilisation «Modifier un ticket»

Titre	Modifier son ticket.
Acteur	Demandeur.
Description	Le demandeur a la possibilité de modifier ses tickets.
Scenario nominal	<ul style="list-style-type: none"> — Le demandeur s'authentifie. — Le demandeur choisit le ticket à modifier. — Le système affiche le formulaire. — Le demandeur modifie les champs voulus. — Le système mise à jour les informations dans la base . — Le système actualise la liste des tickets et l'afficher.
Scenario alternatif	Ticket inexistant alors le système affiche un message d'erreur
Pré condition	Le demandeur doit s'authentifier.

TABLE II.3 – Description textuelle du cas d'utilisation «Modifier un ticket».

(d) Cas d'utilisation «Supprimer un ticket»

Titre	Supprimer un ticket.
Acteur	Demandeur.
Description	Le demandeur a la possibilité de supprimer des tickets .
Scenario nominal	<ul style="list-style-type: none"> — Le demandeur s'authentifie. — Le demandeur choisit le ticket à supprimer. — Le système affiche le formulaire. — Le demandeur supprime les champs voulus. — Le système mise à jour les informations dans la base. — Le système actualise la liste des tickets et l'afficher.
Scenario alternatif	Ticket inexistant alors le système affiche un message d'erreur.
Pré condition	Le demandeur doit s'authentifier.

TABLE II.4 – Description textuelle du cas d'utilisation «Supprimer un ticket».

(e) Cas d'utilisation «Modifier un profil»

Titre	Modifier un profil.
Acteur	Les utilisateurs.
Description	Pouvoir modifier un champ.
Scenario nominal	<ul style="list-style-type: none"> — Les utilisateurs s'authentifient. — Ils accèdent à leurs profils. — Le système affiche le formulaire. — Ils modifient les champs voulus. — Le système mise à jour les informations dans la base. — Le système actualise les champs et l'affiche.
Scenario alternatif	<p>Champs obligatoires non valides et/ou vides</p> <p>Le système affiche un message d'erreur.</p>
Pré condition	<ul style="list-style-type: none"> — Les utilisateurs doivent avoir un compte. — ils doivent s'authentifier.

TABLE II.5 – Description textuelle du cas d'utilisation «Modifier un profil».

(f) Cas d'utilisation «Consulter la liste des tickets»

Titre	Consulter la liste des tickets.
Acteur	Demandeur,Technicien.
Description	Le demandeur et le technicien ont la possibilité de consulter la liste des des tickets.
Scenario nominal	<ul style="list-style-type: none"> — Le demandeur et technicien s'authentifient. — Le demandeur et technicien demandent de consulter la liste des tickets. — Le système affiche la liste des tickets.
Pré condition	<ul style="list-style-type: none"> — Demandeur,Technicien doivent avoir un compte. — Demandeur,Technicien doivent s'authentifier.

TABLE II.6 – Description textuelle du cas d'utilisation «Consulter la liste des tickets».

(g) Cas d'utilisation «Traiter les tickets»

Titre	Traiter les tickets.
Acteur	Technicien.
Description	Le technicien a la possibilité de traiter la liste des tickets.
Scenario nominal	<ul style="list-style-type: none"> — Le technicien s'authentifier. — Le technicien demande de consulter la liste des tickets. — Le système affiche la liste des ticket. — Le technicien sélectionne un ticket à traiter et valider.
Scenario alternatif	Champs non valides et/ou vides Le système affiche un message d'erreur.
Pré condition	Technicien doit s'authentifier.

TABLE II.7 – Description textuelle du cas d'utilisation «Traiter les tickets».

(h) Cas d'utilisation «Affecter un ticket»

Titre	Affecter un ticket.
Acteur	Responsable métier.
Description	Le responsable métier a la possibilité d'affecter les tickets.
Scenario nominal	<ul style="list-style-type: none"> — Le responsable métier s'authentifier. — Le responsable métier demande de consulter la liste des tickets. — Le système affiche la liste des ticket. — Le responsable métier sélectionne un ticket. — Le système affiche les données du ticket sélectionné. — Le responsable métier choisit un technicien. — Le responsable métier valide . — Le système met à jour les informations dans la base. — Le système actualise la liste des tickets et l'affiche.
Scenario alternatif	<p>Champs non valides et/ou vides Le système affiche un message d'erreur.</p>
Pré condition	Le responsable métier doit s'authentifier.

TABLE II.8 – Description textuelle du cas d'utilisation «Affecter un ticket».

(i) Cas d'utilisation «Gérer les comptes»

Titre	Gérer les comptes.
Acteur	Responsable IT.
Description	Pouvoir d'ajouter, supprimer et modifier un compte.
Scenario nominal	<p style="text-align: center;">Ajouter un compte</p> <ul style="list-style-type: none"> — Le responsable IT demande d'afficher le formulaire d'ajout. — Le système affiche le formulaire à remplir. — Le responsable IT remplit et valide le formulaire. — Le système ajoute les informations dans la base. — Le système actualise la liste des comptes et l'affiche. <p style="text-align: center;">Modifier un compte</p> <ul style="list-style-type: none"> — Responsable IT demande la liste des comptes. — Le système affiche la liste des comptes. — Le responsable IT choisit un compte à modifier. — Le système affiche le formulaire de modification. — Il modifie les champs voulus. — Le responsable IT valide ou annule la modification. — Le système mise à jour les informations dans la base. — Le système actualise la liste des comptes et l'affiche. <p style="text-align: center;">Supprimer un compte</p> <ul style="list-style-type: none"> — Le responsable IT demande la liste des comptes. — Le système affiche la liste des comptes. — Le responsable IT choisit le compte à supprimer. — Le système demande une confirmation. — Le responsable IT confirme ou annule la suppression. — Le système supprime le compte de la base. — Le système actualise la liste des comptes.
Scenario alternatif	<p style="text-align: center;">Compte existe déjà ou champ vide Le système affiche un message d'erreur.</p> <p style="text-align: center;">Modification avec des champs vides Le système affiche un message d'erreur.</p> <p style="text-align: center;">Comptes inexistant Un message d'erreur sera affiché.</p>
Pré condition	Le responsable IT doit être authentifié.

TABLE II.9 – Description textuelle du cas d'utilisation «Gérer les comptes».

(j) Cas d'utilisation «S'affecter les tickets»

Titre	S'affecter les tickets.
Acteur	Technicien.
Description	Le technicien a la possibilité de s'affecter les tickets.
Scenario nominal	<ul style="list-style-type: none"> — Le technicien s'authentifier. — Le technicien demande de consulter la liste des tickets. — Le système affiche la liste des ticket. — Le technicien choisit un ticket. — technicien valide . — Le système mise à jour les informations dans la base. — Le système actualise la liste des tickets et l'affiche.
Pré condition	Le technicien doit s'authentifier .

TABLE II.10 – Description textuelle du cas d'utilisation « S'affecter les tickets »

(k) Cas d'utilisation «Gérer les catégories»

Titre	Gérer les catégories.
Acteur	Responsable métier.
Description	Pouvoir d'ajouter, supprimer et modifier une catégorie.
Scenario nominal	<p style="text-align: center;">Ajouter une catégorie</p> <ul style="list-style-type: none"> — Le responsable métier demande d'afficher le formulaire d'ajout. — Le système affiche le formulaire à remplir. — Le responsable métier remplit et valide le formulaire. — Le système ajoute les informations dans la base. — Le système actualise la liste des catégories et l'affiche. <p style="text-align: center;">Modifier une catégorie</p> <ul style="list-style-type: none"> — Responsable métier demande la liste des catégories. — Le système affiche la liste des catégories. — Le responsable métier choisit une catégorie à modifier. — Le système affiche le formulaire de modification. — Il modifie les champs voulus. — Le responsable métier valide ou annule la modification. — Le système mise à jour les informations dans la base. — Le système actualise la liste des catégories et l'affiche. <p style="text-align: center;">Supprimer une catégorie</p> <ul style="list-style-type: none"> — Le responsable métier demande la liste des catégories. — Le système affiche la liste des catégories. — Le responsable métier choisit la catégorie à supprimer. — Le système demande une confirmation. — Le système actualise la liste des catégories et l'affiche.
Scenario alternatif	<p style="text-align: center;">Catégorie existe déjà ou champ vide Le système affiche un message d'erreur.</p> <p style="text-align: center;">Modification avec des champs vides Le système affiche un message d'erreur .</p> <p style="text-align: center;">Catégorie inexistant Un message d'erreur sera affiché.</p>
Pré condition	— Le responsable métier doit être authentifié.

TABLE II.11 – Description textuelle du cas d'utilisation «Gérer les catégories».

II.6 Diagramme de séquence

Un diagramme de séquence est un diagramme uml qui représente les interactions entre les objets en mettant l'accent sur le classement chronologique des messages échangés. Les scénarios sont des instances des cas d'utilisation et ils sont traduits en diagrammes de séquences [13].

II.6.1 Diagramme de séquence du cas d'utilisation «Authentification»

Pour accéder au système, l'utilisateur introduit son adresse email et son mot de passe : Si les informations sont correctes, la page demandée s'affiche, sinon le système renvoie un message d'erreur.

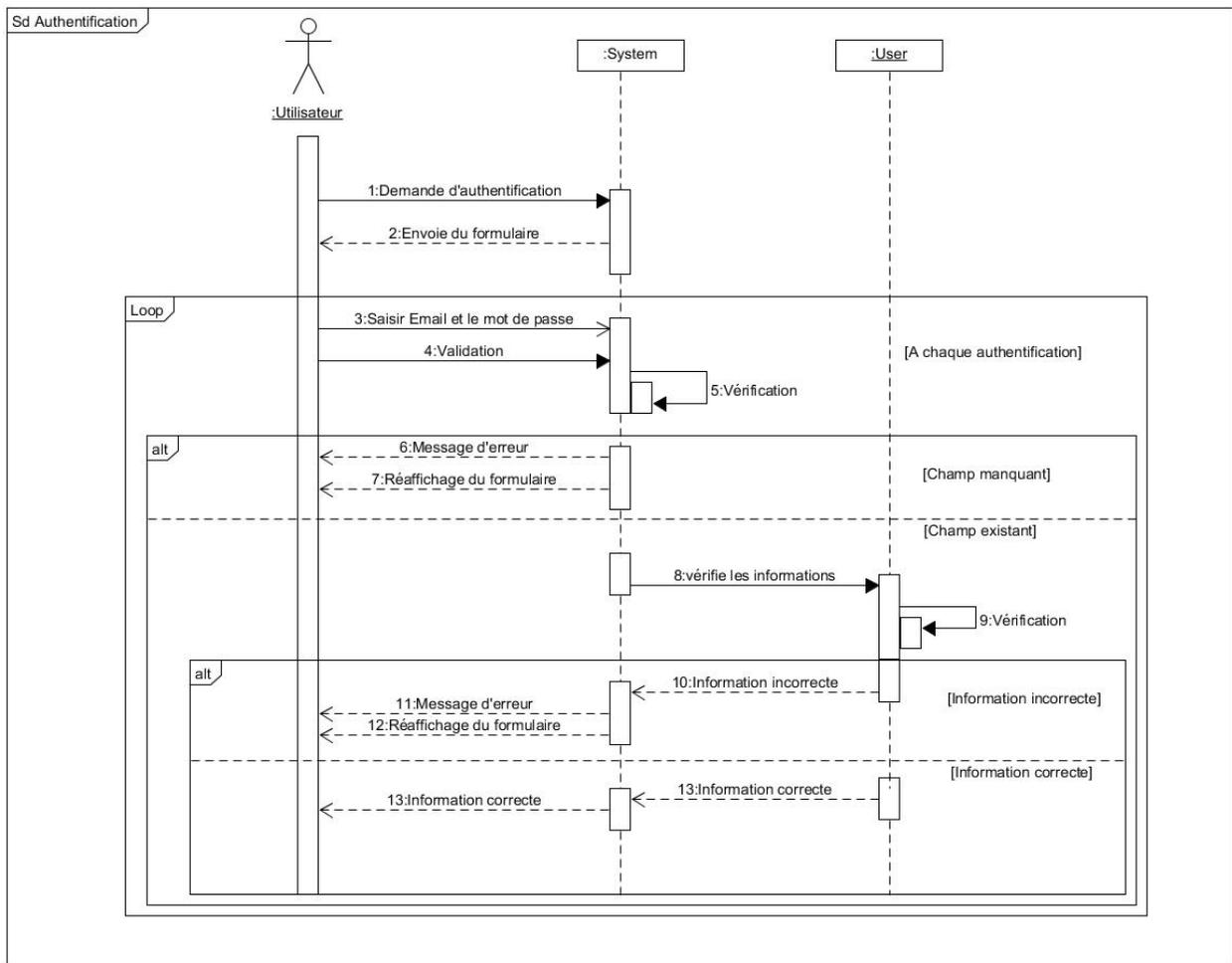


FIGURE II.10 – Diagramme de séquence du cas d'utilisation «Authentification».

II.6.2 Diagramme de séquence du cas d'utilisation «Ajouter un compte»

Une fois authentifié, le responsable IT a la possibilité d'ajouter un ou plusieurs utilisateurs.

Pour cela, il demande d'ajouter un utilisateur après le système lui renvoie un formulaire a remplir. Une fois terminé, il valide : si aucune erreur n'est commise, l'ajout sera fait sinon le système lui renvoie un message d'erreur.

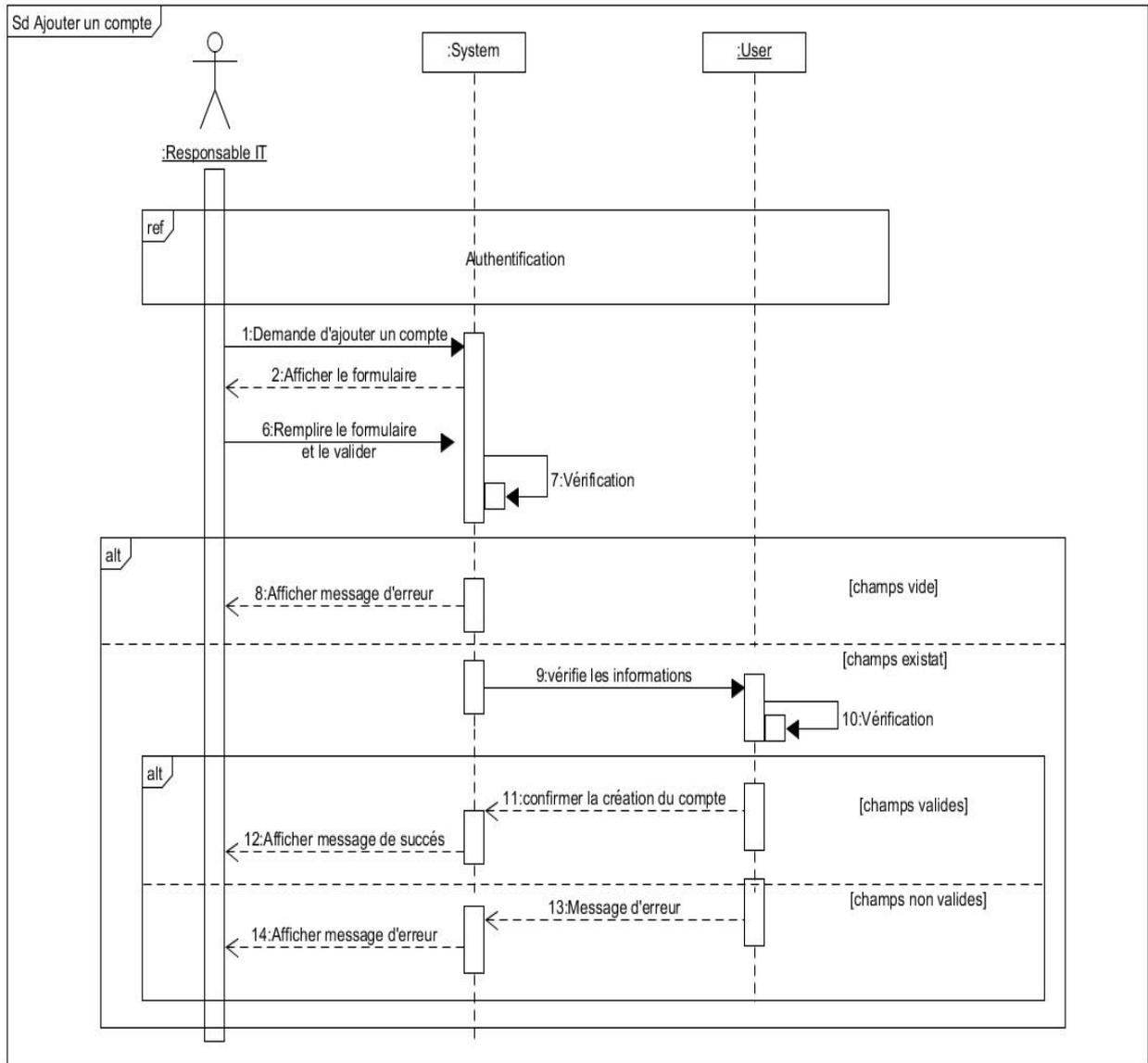


FIGURE II.11 – Diagramme de séquence du cas d'utilisation «Ajouter un compte».

II.6.3 Diagramme de séquence du cas d'utilisation «Modifier un compte»

Une fois authentifié, le responsable IT a la possibilité : De modifier un ou plusieurs compte. Pour cela, il demande de modifier un compte, le système lui renvoie un formulaire a remplir et une fois terminé, il valide : si aucune erreur n'est commise la modification sera fait sinon le système lui renvoie un message d'erreur.

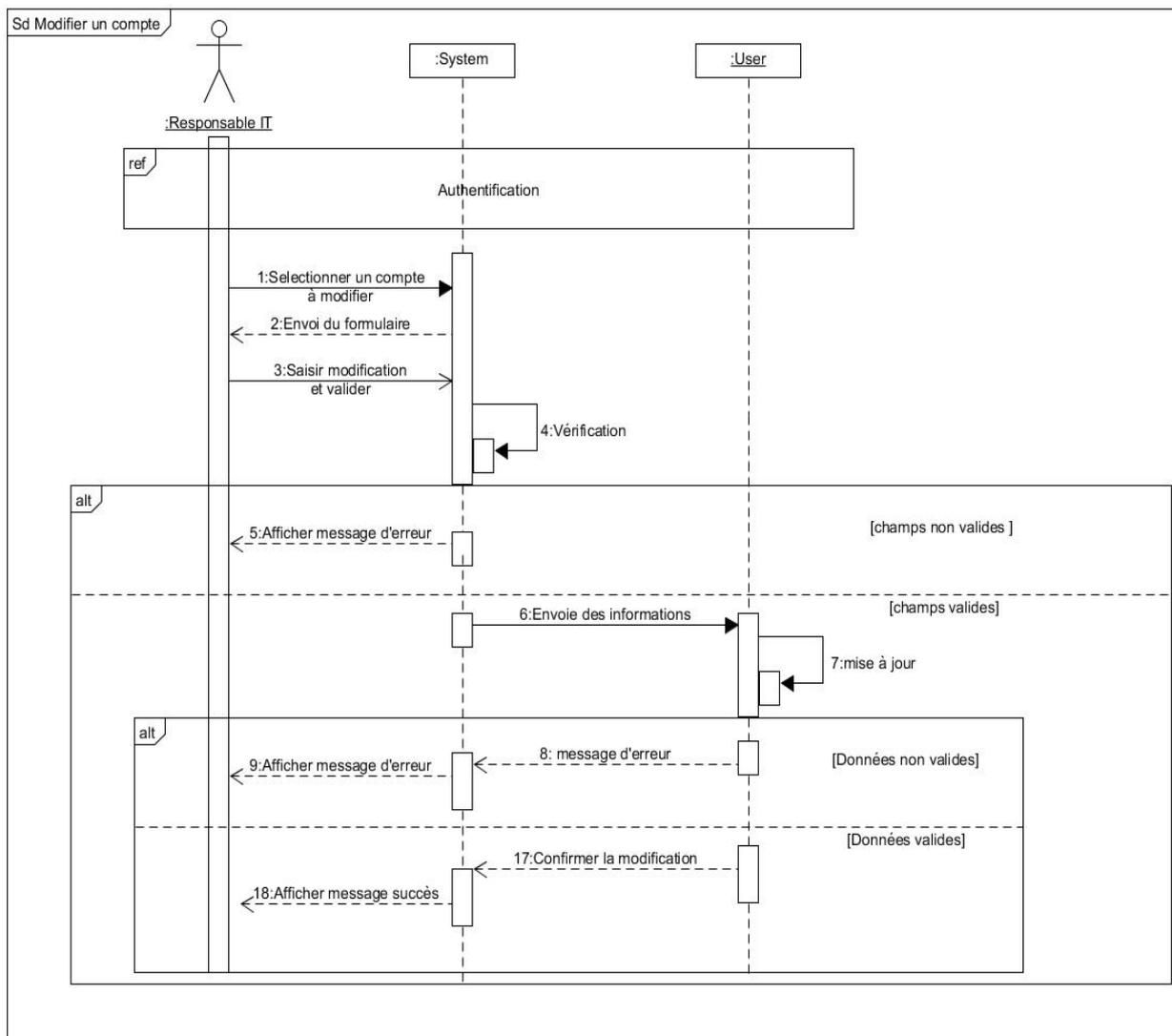


FIGURE II.12 – Diagramme de séquence du cas d'utilisation «Modifier un compte».

II.6.4 Diagramme de séquence du cas d'utilisation «Supprimer un compte»

Une fois authentifié, le responsable IT a la possibilité : De supprimer un ou plusieurs compte.

Pour cela, il demande de supprimer un compte, le système vérifie son existence dans la base de données : si le compte existe la suppression sera appliquée sinon un message d'erreur sera affiché.

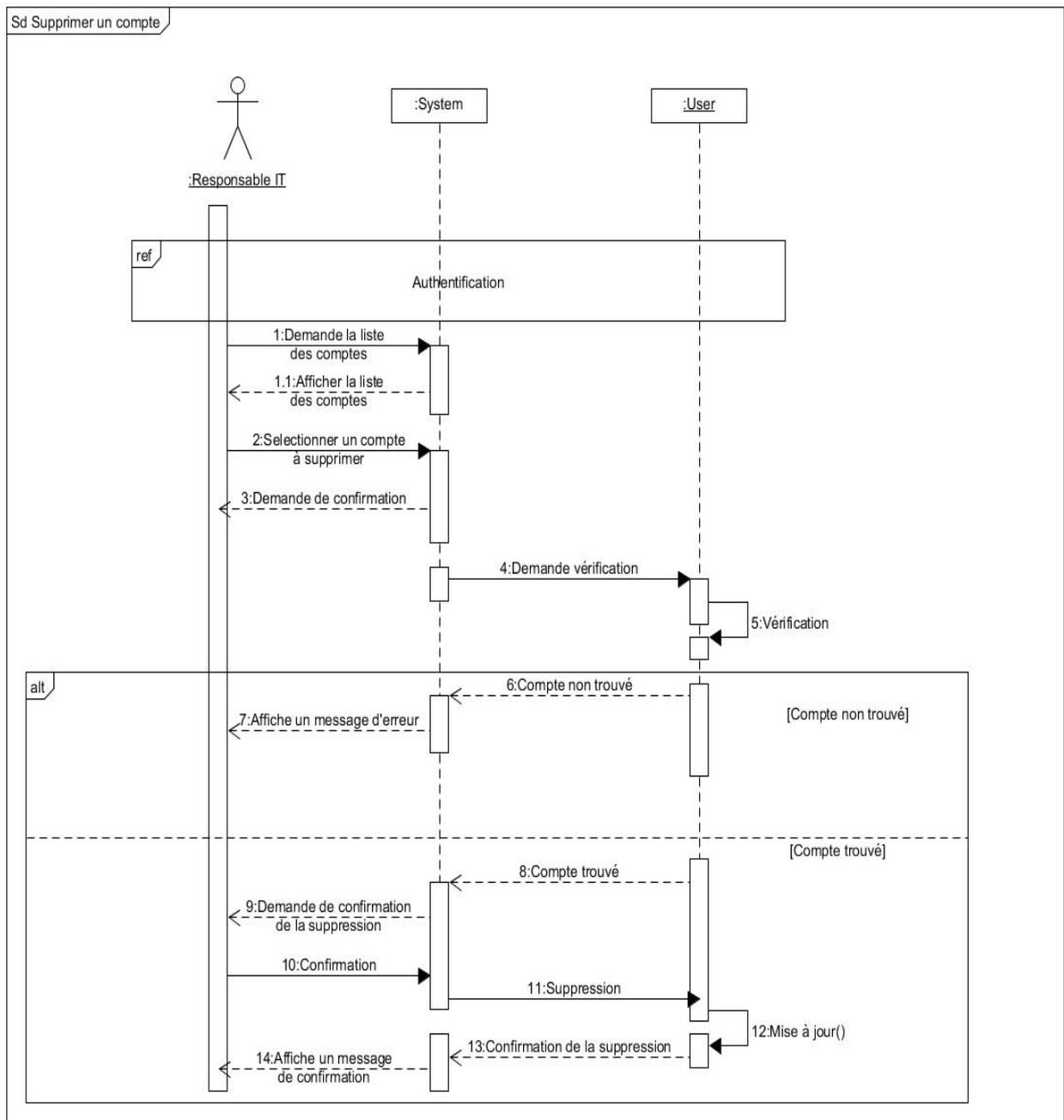


FIGURE II.13 – Diagramme de séquence du cas d'utilisation «Supprimer un compte».

II.6.5 Diagramme de séquence du cas d'utilisation «Ajouter un ticket»

Une fois authentifié, Le demandeur a la possibilité d'ajouter un ou plusieurs ticket. Pour cela, il demande d'ajouter un ticket, le système lui renvoie un formulaire a remplir. Une fois terminé, il valide : si aucune erreur n'est commise l'ajout sera fait sinon le système lui renvoie un message d'erreur.

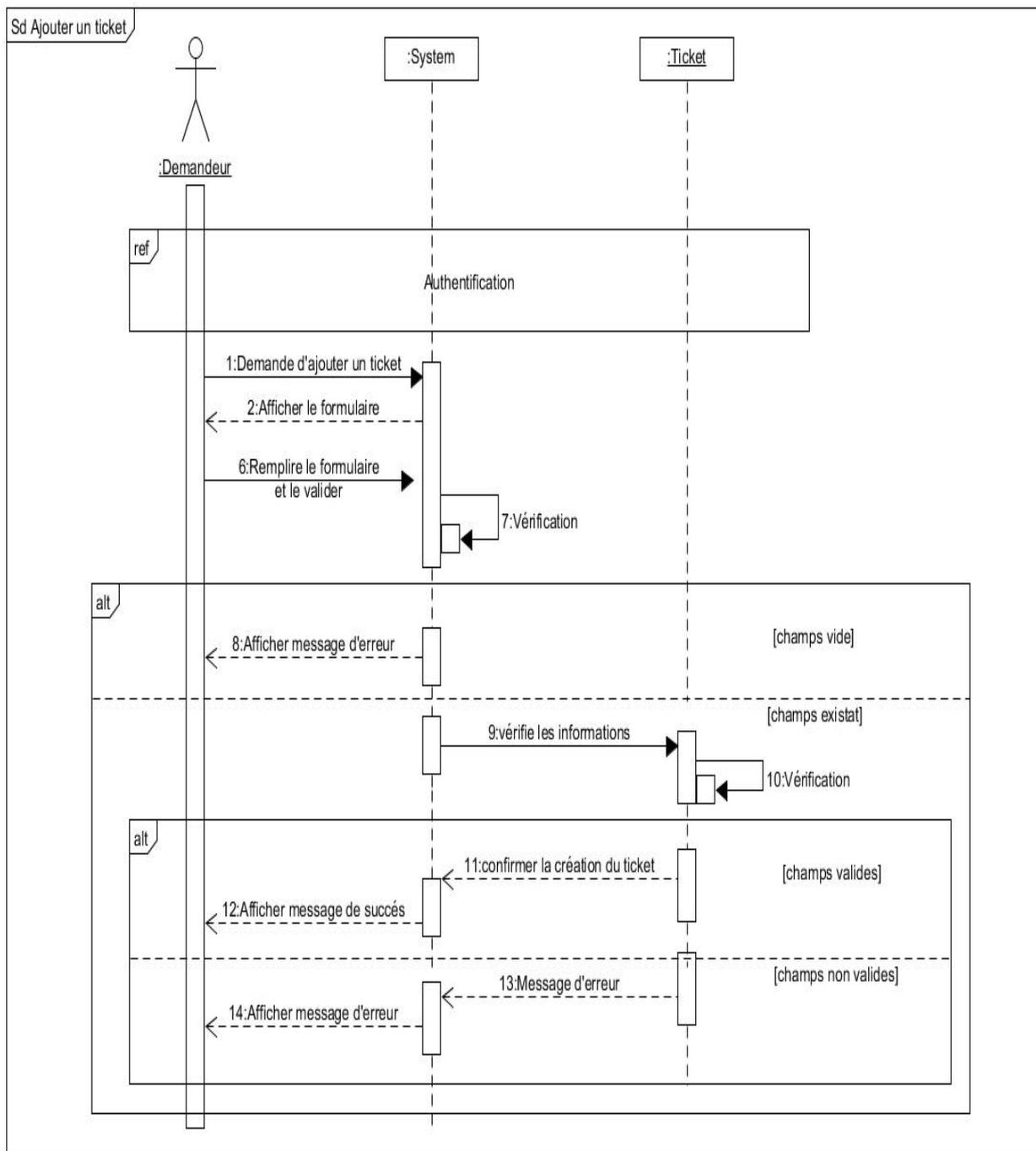


FIGURE II.14 – Diagramme de séquence du cas d'utilisation «Ajouter un ticket».

II.6.6 Diagramme de séquence du cas d'utilisation «Modifier un ticket»

Une fois authentifié :
 l'utilisateur a la possibilité de modifier son ticket.
 Le responsable IT peut aussi modifier des tickets.
 le système lui renvoie un formulaire a remplir, une fois terminé, il valide, si aucune erreur n'est commise, la modification sera fait, sinon le système lui renvoie un message d'erreur.

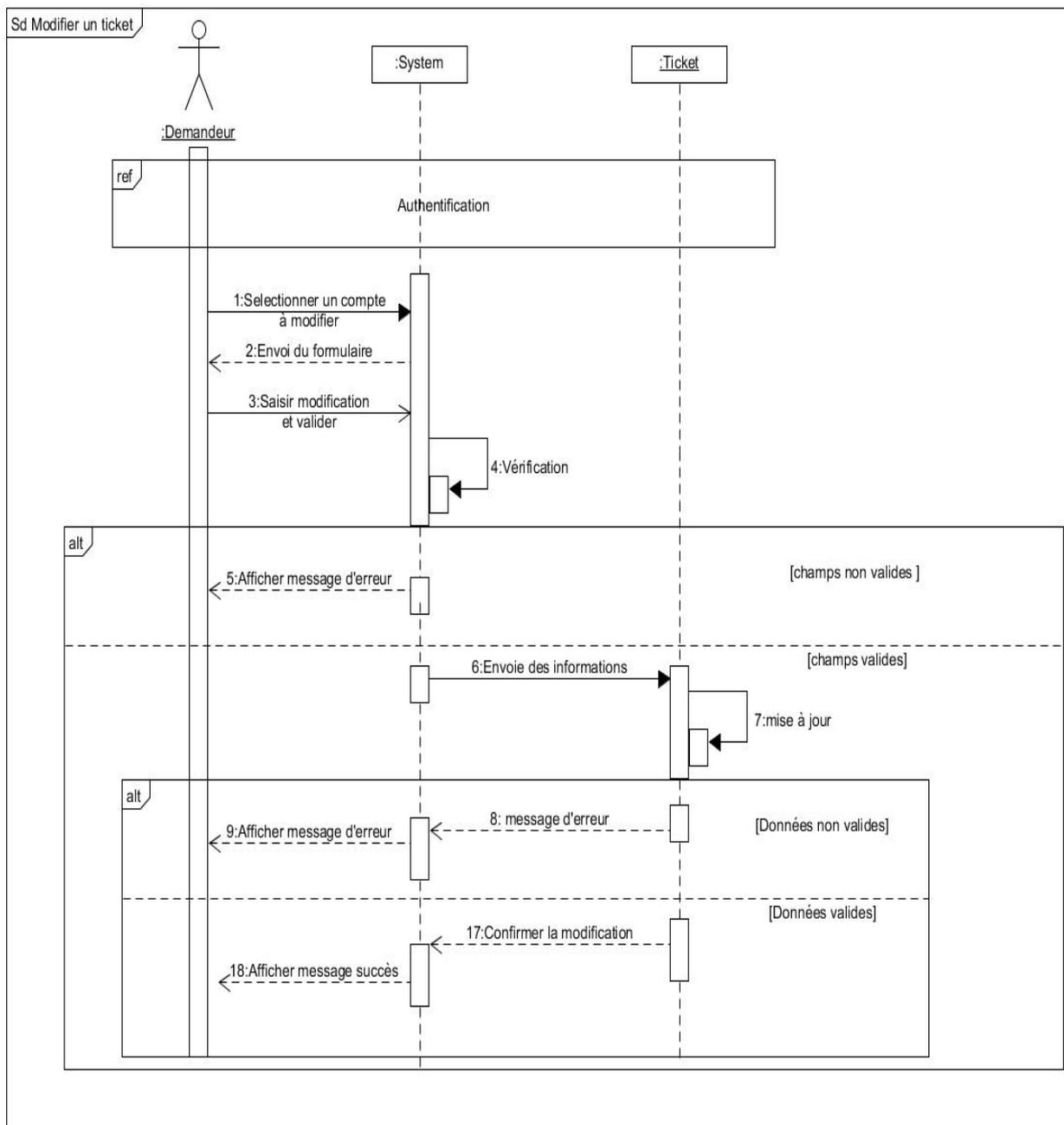


FIGURE II.15 – Diagramme de séquence du cas d'utilisation «Modifier un ticket».

II.6.7 Diagramme de séquence du cas d'utilisation «Supprimer un ticket»

Une fois authentifié :
 l'utilisateur a la possibilité : De supprimer son ticket.
 Le responsable IT a la possibilité de supprimer des tickets. Pour cela, ils demandent de supprimer un ticket, le système vérifie son existence dans la base de données : si le ticket existe la suppression sera appliquée sinon un message d'erreur sera affiché.

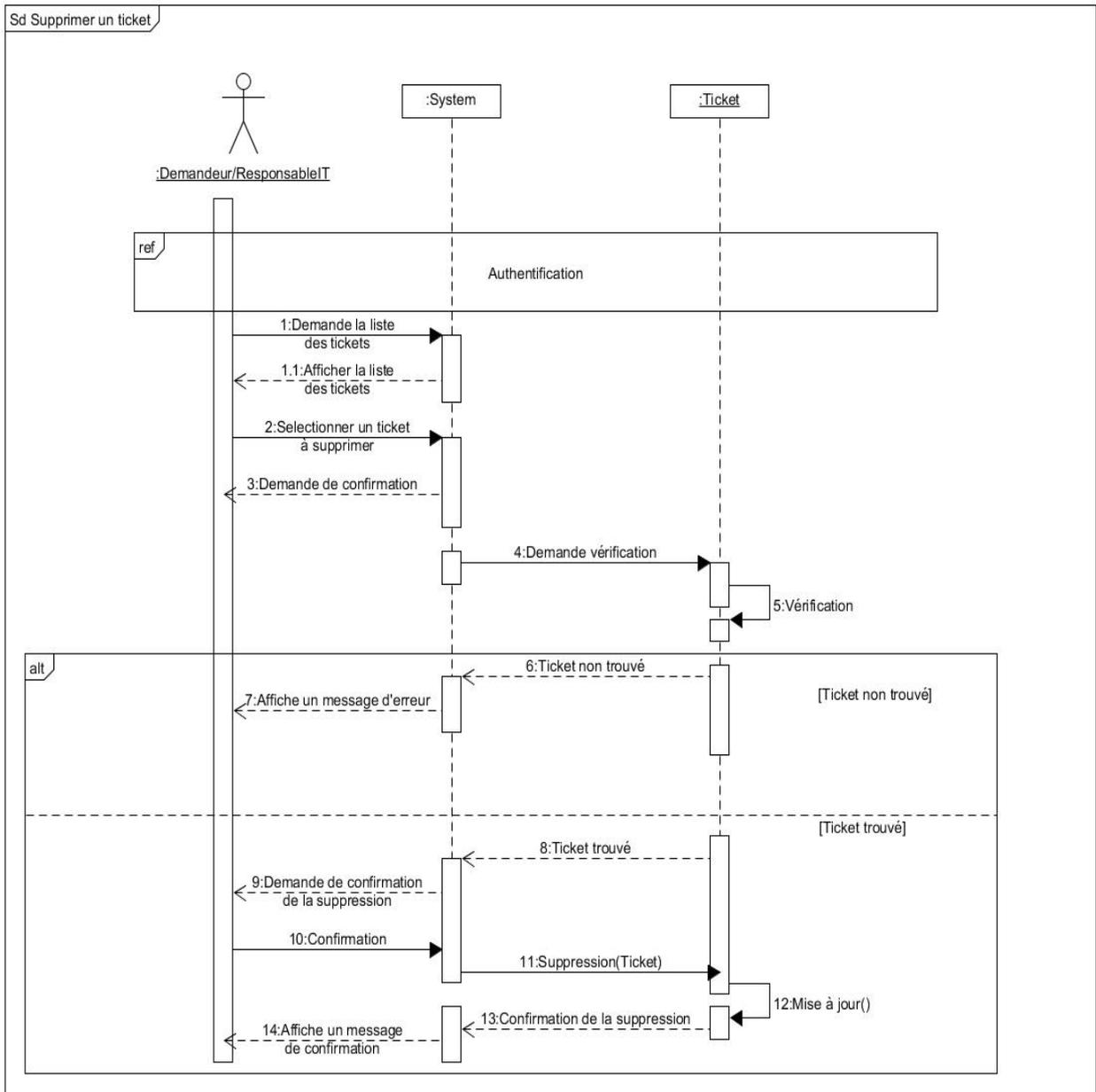


FIGURE II.16 – Diagramme de séquence du cas d'utilisation «Supprimer un ticket».

II.6.8 Diagramme de séquence du cas d'utilisation «Traiter un ticket»

Pour traiter un ticket, le technicien doit d'abord s'authentifier puis accéder a la liste des tickets.

Il sélectionne le ticket qui veut traiter, puis le système lui affiche le formulaire du ticket et il modifie. Une fois terminé, il valide : si aucune erreur ni commise, le traitement sera fait sinon il envoi un message d'erreur.

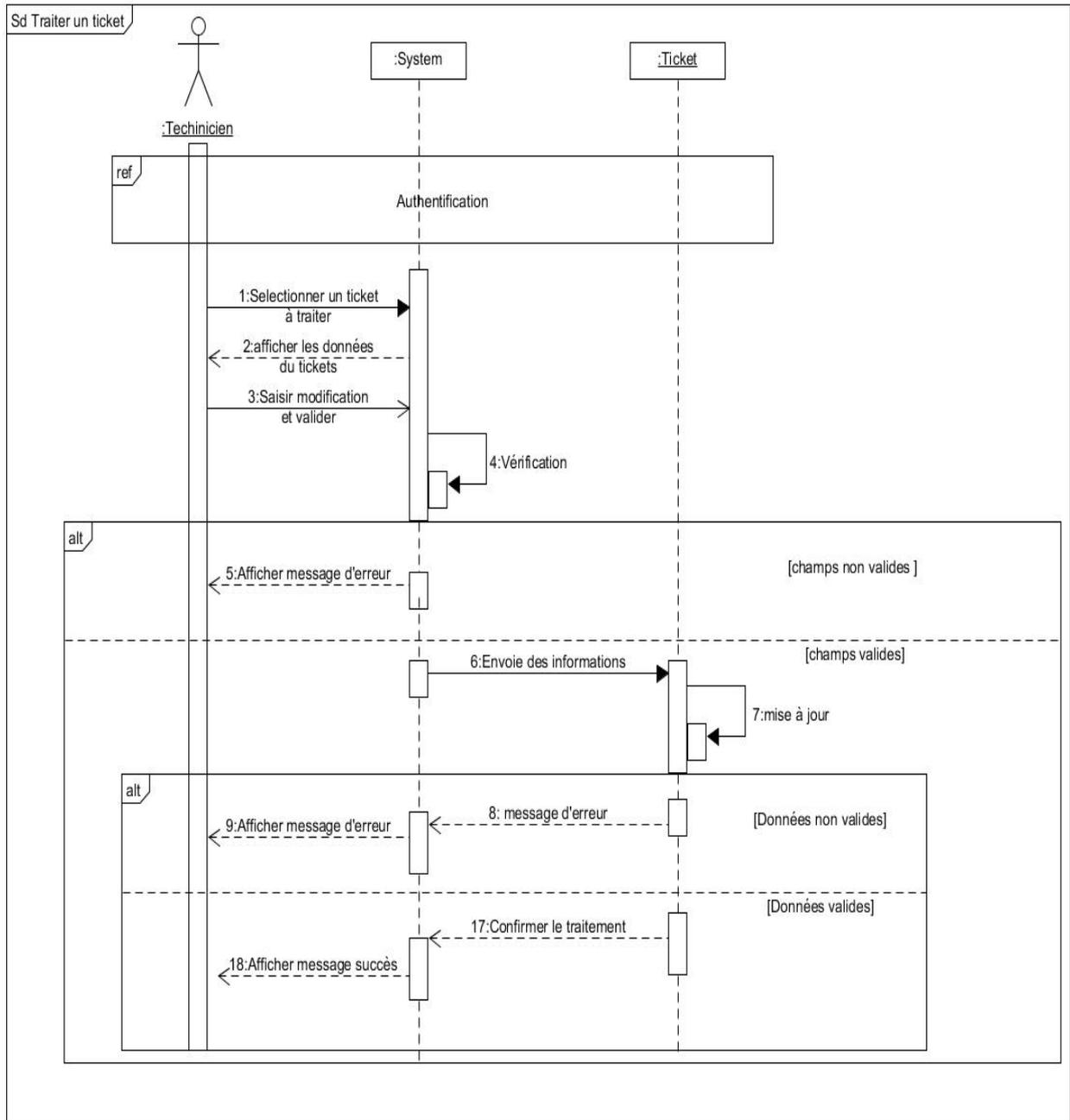


FIGURE II.17 – Diagramme de séquence du cas d'utilisation «Traiter un ticket».

II.6.9 Diagramme de séquence du cas d'utilisation «Affecter un ticket»

Une fois authentifié, le responsable IT a la possibilité d'affecter les tickets, le système lui renvoie un formulaire a remplir. Une fois terminé, il valide : si aucune erreur n'est commise l'affectation sera faite sinon le système lui renvoie un message d'erreur.

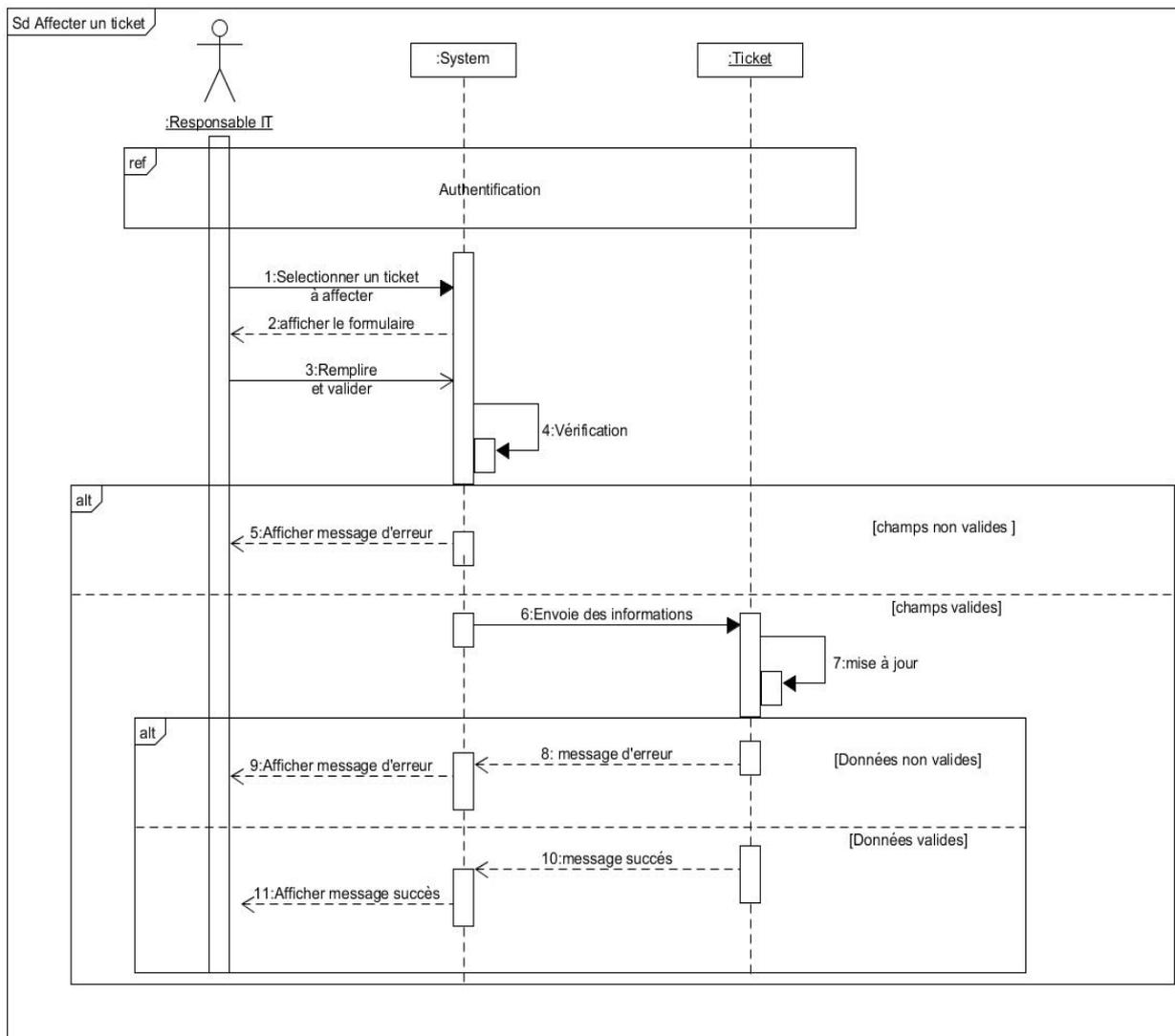


FIGURE II.18 – Diagramme de séquence du cas d'utilisation «Affecter un ticket».

II.6.10 Diagramme de séquence du cas d'utilisation «Ajouter une catégorie»

Une fois authentifié, le responsable métier a la possibilité d'ajouter une ou plusieurs catégories.

Pour cela, il demande d'ajouter une catégorie, le système lui renvoie un formulaire a remplir. Une fois terminé, il valide : si aucune erreur n'est commise l'ajout sera fait sinon le système lui renvoie un message d'erreur.

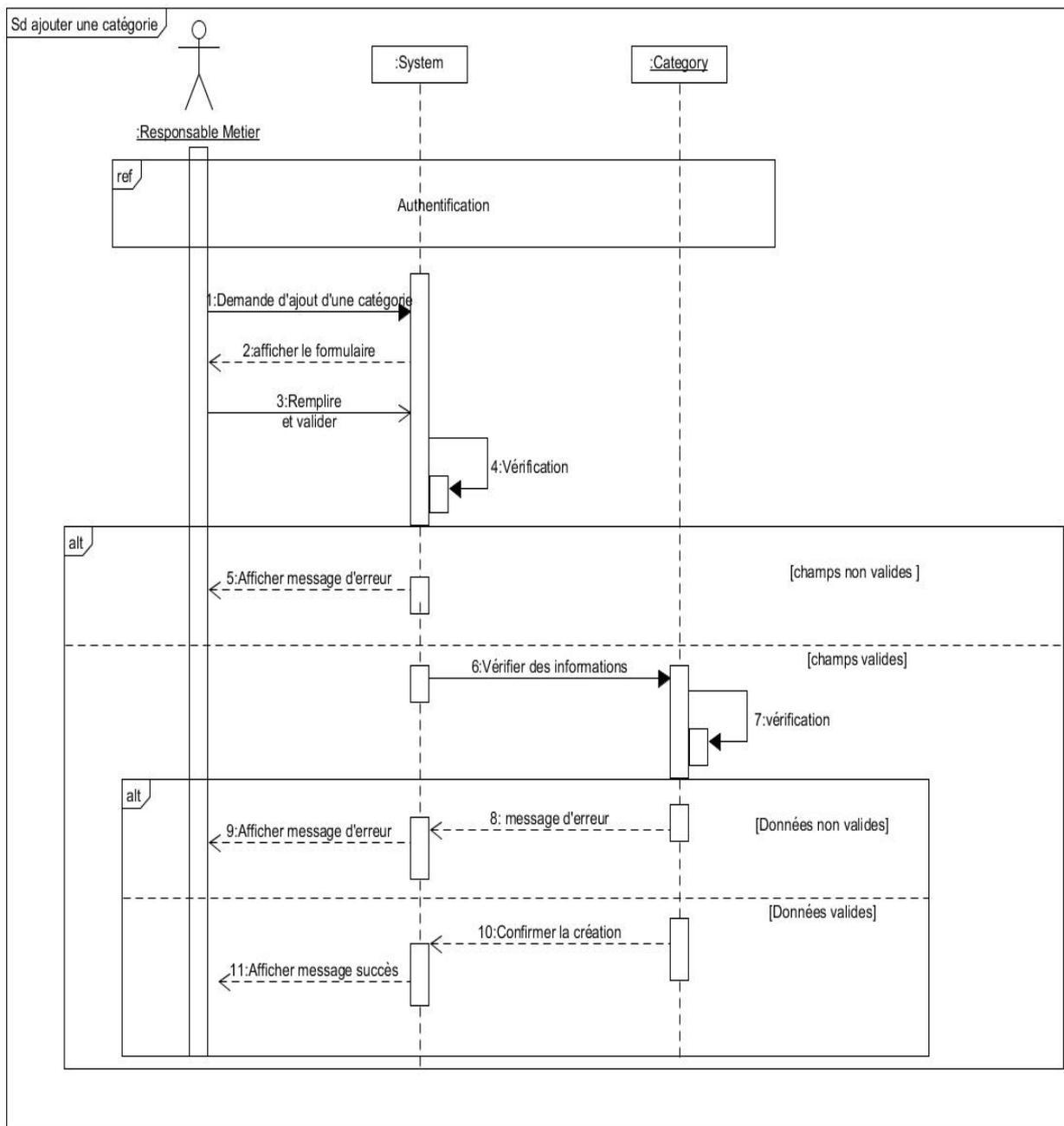


FIGURE II.19 – Diagramme de séquence du cas d'utilisation «Ajouter une catégorie».

II.6.11 Diagramme de séquence du cas d'utilisation «Modifier une catégorie»

Une fois authentifié, le responsable métier a la possibilité de modifier les catégories, le système lui renvoie un formulaire à remplir. Une fois terminé, il valide : si aucune erreur n'est commise la modification sera fait sinon le système lui renvoie un message d'erreur.

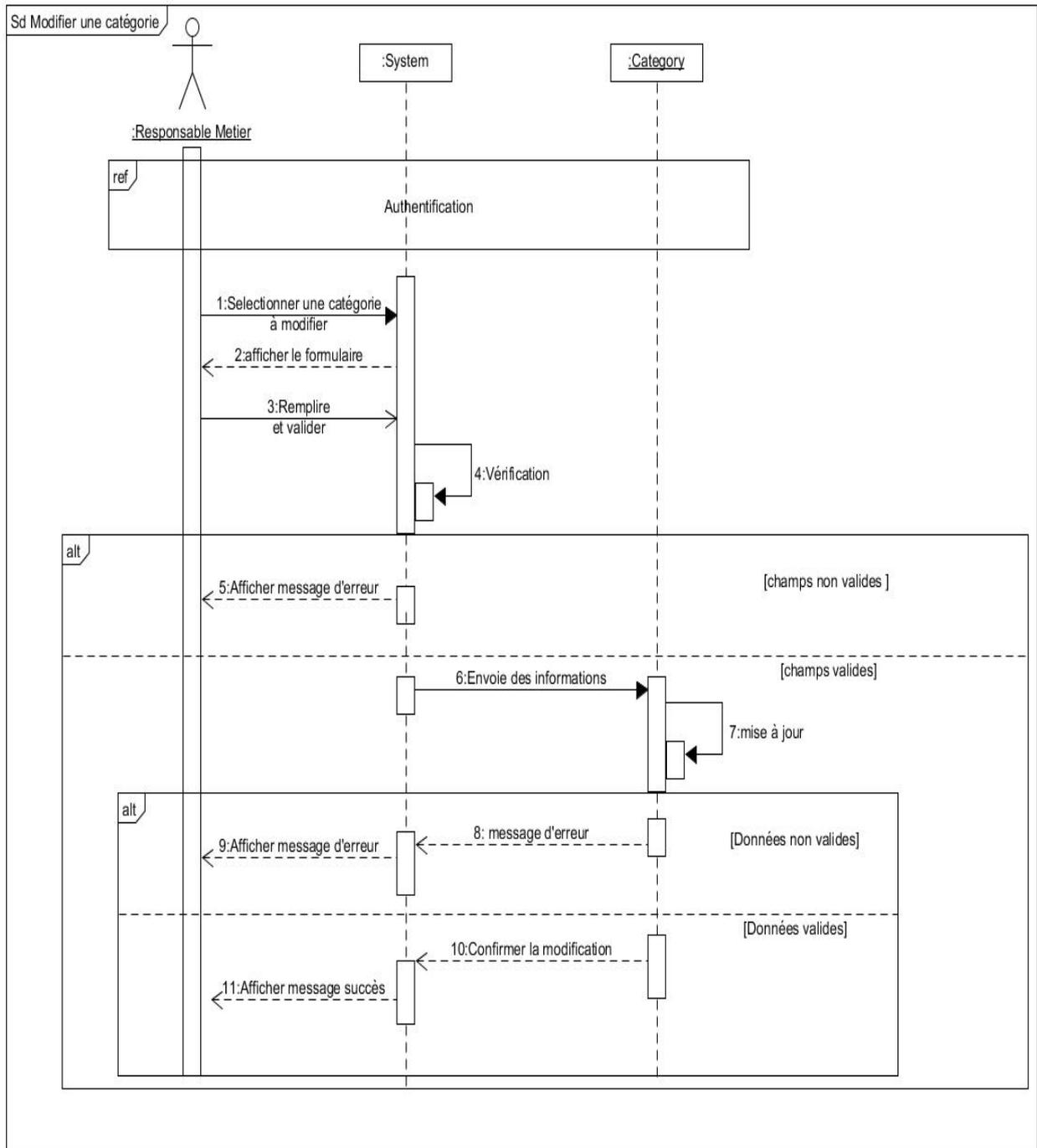


FIGURE II.20 – Diagramme de séquence du cas d'utilisation «Modifier une catégorie».

II.6.12 Diagramme de séquence du cas d'utilisation «Supprimer une catégorie»

Une fois authentifié, le responsable métier a la possibilité de supprimer une catégorie. Pour se faire, le responsable métier demande de supprimer une catégorie, il lui affiche un message de confirmation :
Si il valide alors la requête est envoyée à la base de données et la catégorie sera supprimé sinon la requête sera abandonnée.

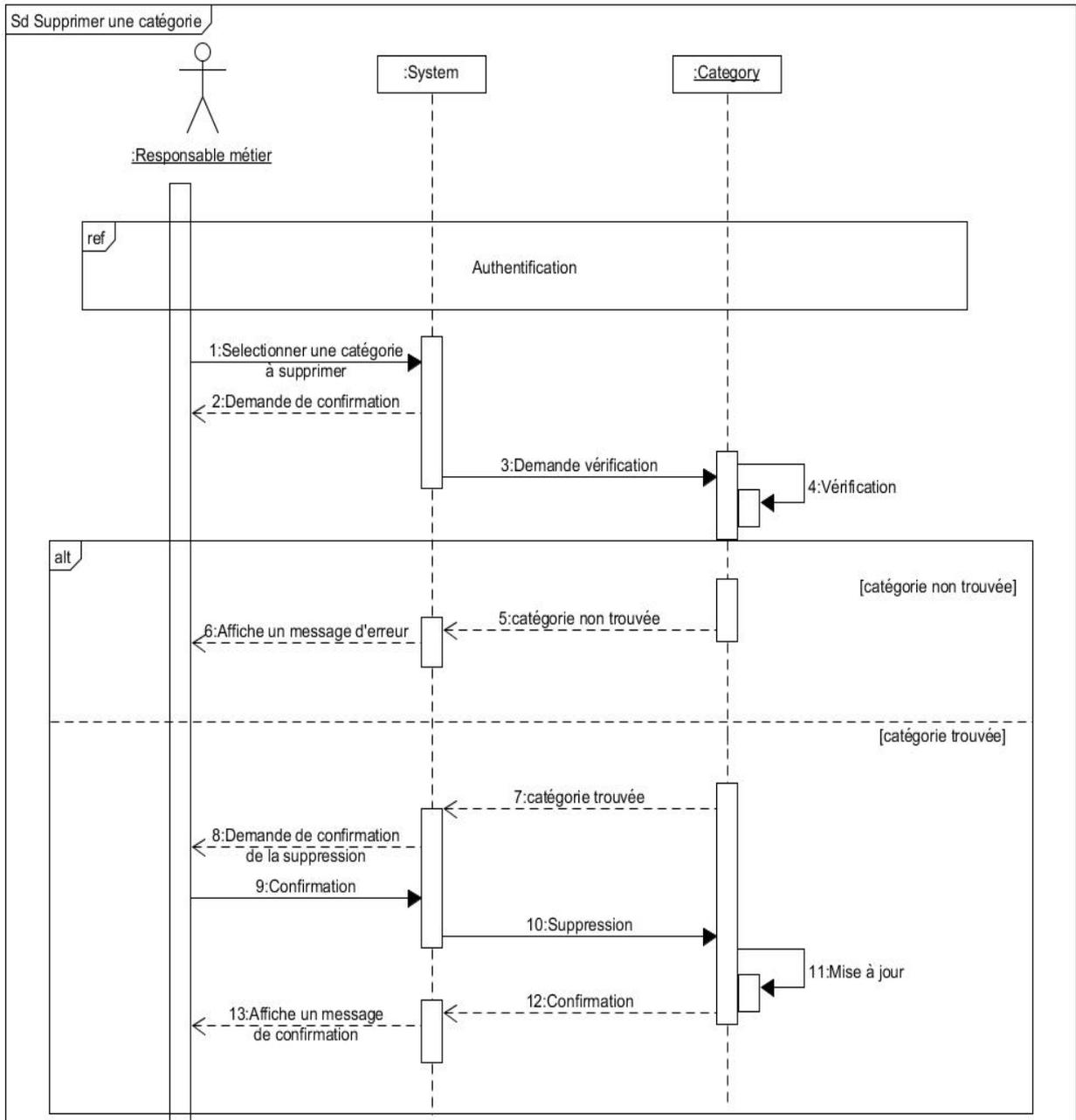


FIGURE II.21 – Diagramme de séquence du cas d'utilisation «Supprimer une catégorie».

II.7 Conclusion

Ce chapitre nous a permis dans un premier lieu de spécifier les besoins de notre application en définissant les différents besoins fonctionnels et non fonctionnels, et de les analyser dans un deuxième lieu.

Cette analyse s'est faite en décrivant le contexte de notre projet, nous avons présenté les diagrammes qui décrivent les fonctionnalités de notre application comme les diagrammes de séquence et cas d'utilisation.

Dans ce qui suit nous entamerons la conception de notre projet en présentant le diagramme de BPMN et le diagramme de classe.

Chapitre III

Conception

III.1 Introduction

Après avoir citer l'étape analyse et spécification des besoins de notre projet dans le deuxième chapitre, on va entamer la phase de conception. Puis nous détaillerons la conception de notre application en présentant le diagramme BPMN et le diagramme de classe.

III.1.1 BPMN (Business Process Model and Notation)

1. Bref historique

La norme BPMN a été développé au départ en 2004 par la Business Process Management Initiative (BPMI). Et depuis 2008 par l'OMG (Object Management Group) qui est l'organisme chargé de normaliser la représentation des modèles de systèmes d'information [14].

2. Definition

La norme BPMN est une norme de notation pour la modélisation de processus. Elle définit une manière de représenter les processus graphiquement pour les rendre clairs et compréhensibles pour toutes les personnes sensibles (intervenants externes, collaborateurs).Il permet de dessiner des diagrammes représentant le déroulement et l'enchaînement de tâches [14].

A l'aide de différents éléments standards :

(a) Les objets de flux :

- **Événement** : Définit une action qui déclenche, il initie un point de départ, une étape intermédiaire ou un point final d'un processus particulier.
- **Activités** : Les activités sont des rectangles arrondis, illustrant une tâche spécifique effectuée par une personne ou un système.

Il existe de nombreuses activités que les concepteurs de processus peuvent créer telles que :

- Sous-processus : un ensemble de tâches supplémentaires classées ensemble.
- Tache : L'action précise qui doit être effectuée.

- **Passerelles** : sont des symboles en forme de losange qui cartographie les points de décision.

(b) **Les objets de connexion** :

Sont les différentes pièces reliées entre elles.

Il existe trois types :

- **Séquence** : Cartographie du flux séquentiel des objets.
- **Association** : Représente la relation entre les différentes données et objets.
- **Message** : Indique un message envoyé entre les différents participants au flux de travail.

(c) **Couloirs** : Piscine ou couloir [14].

Diagramme de BPMN «Gérer les tickets»

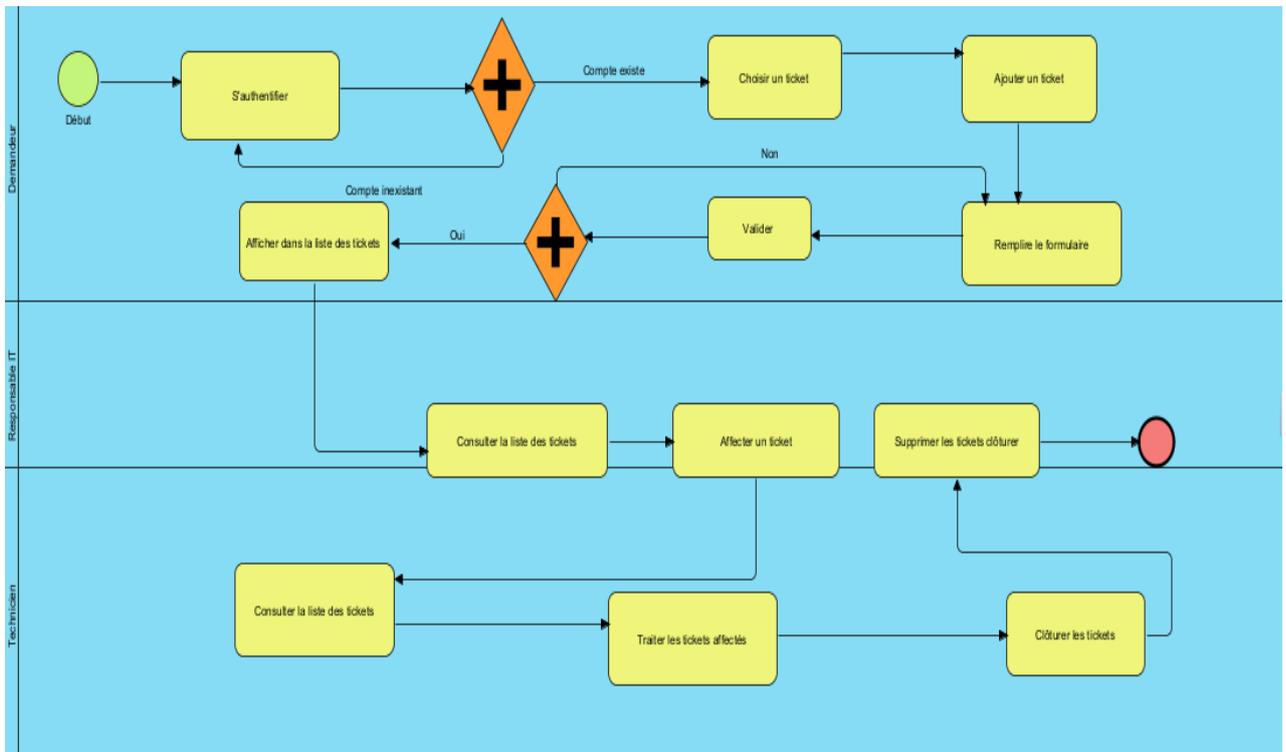


FIGURE III.1 – Diagramme BPMN.

III.2 Diagramme de classe

Le diagramme de classe est généralement considéré comme le plus important dans la conception d'une application.

Ce diagramme est une représentation statique qui permet de définir la structure interne de système ainsi les différentes relations entre les classes et les interfaces .

Il constitue l'un des pivots essentiels de la modélisation [15].

Ci-dessous nous présenterons le diagramme de classe associé à notre application.

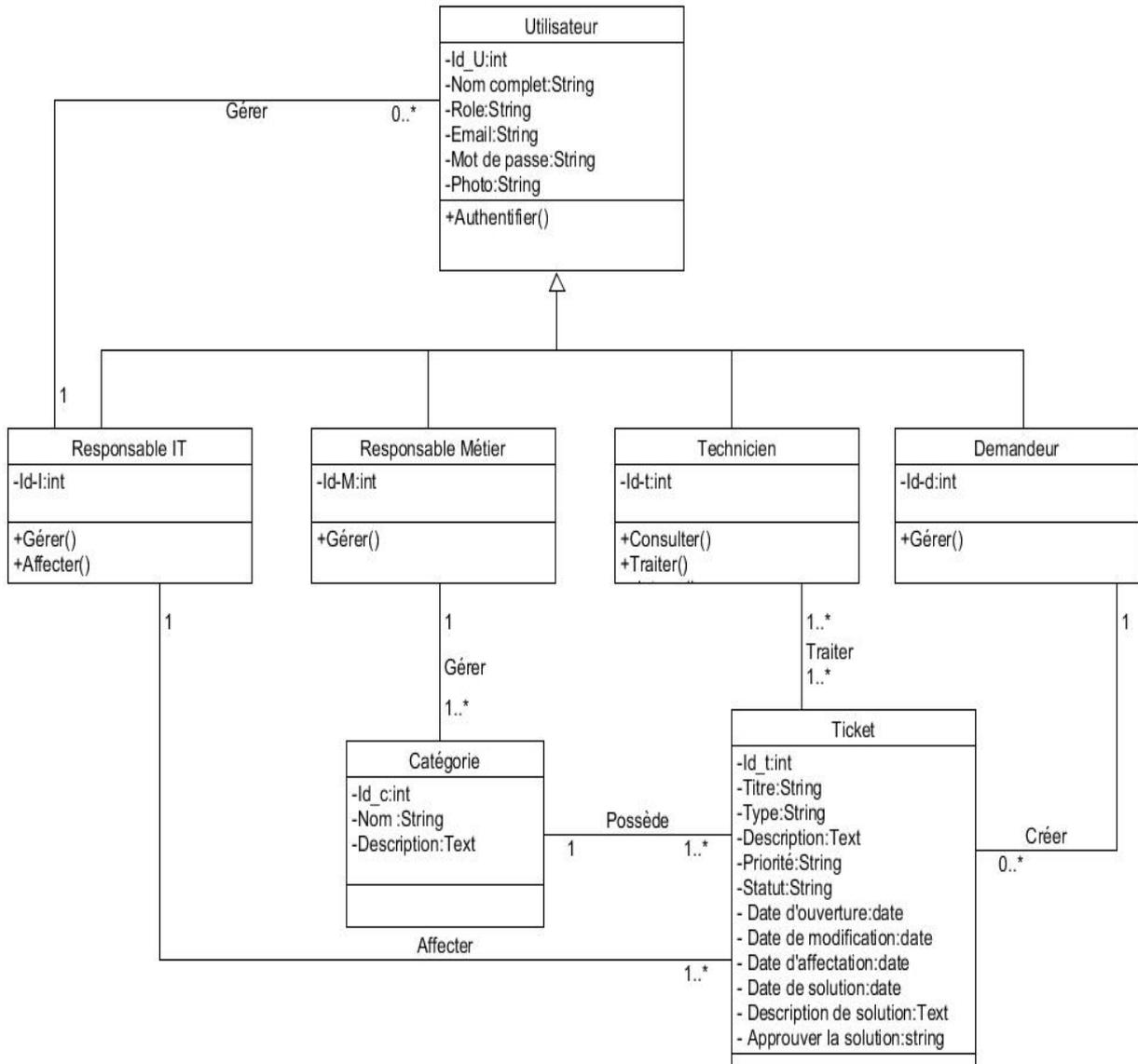


FIGURE III.2 – Diagramme de classe.

III.3 Modèle relationnel

Après l'application des règles de passages du diagramme de classe au modèle relationnel, nous présentons ci-dessous toutes les relations obtenues :

Technicien(id_t , id_u, nom complet, role, email, mot de passe, photo)

Ticket (id_tic ,titre, description, priorité, statut, type, date d'ouverture , date de modification, date de solution, description de la solution, approuver la solution, #id_i, #id_c, #id_d)

Catégorie (id_c, nom, description, id_m)

Traiter(id_t, id_tic)

Responsable IT(id_i , id_u, nom complet, role, email, mot de passe, photo)

Responsable Metier(id_m , id_u, nom complet, role, email, mot de passe, photo)

Demandeur(id_d , id_u, nom complet, role, email, mot de passe, photo)

III.4 Conclusion

Au cours de ce chapitre, nous avons présenté les diagrammes qui décrivent les fonctionnalités de notre application comme les diagrammes de séquence.

Dans le chapitre suivant la réalisation, nous allons formaliser ce qu'on vient de décrire au cours de ce chapitre conception.

Chapitre IV

Réalisation

IV.1 Introduction

Après avoir affecter l'étude et la conception de notre application, nous passons à la phase d'implémentions.

Ce chapitre présente le résultat du travail effectuée durant ce projet de fin d'études.

Nous allons présenter aussi les environnements et les outils de développement utilisé.

Nous clôturons ce chapitre par quelques captures d'écran montrant les fonctionnalités de notre application.

IV.2 Langages et environnements de développement

IV.2.1 Langage de développement

1. Le modèle MVC

Le modèle MVC est un moyen de concevoir une application informatique en la divisant en trois sous-parties : Modèle, vue et contrôleur.

Il aide à gérer et à organiser correctement le code source, dans le but de diviser la logique du code en trois parties dans des fichiers séparés.

- **Modèle** : Cette partie gère les données de votre site. Son rôle est d'aller récupérer les informations brutes dans la base de données, de les organiser et de les assembler pour qu'elles puissent ensuite être traitées par le contrôleur. On y trouve donc entre autres les requêtes SQL.
- **Vue** : cette partie se concentre sur l'affichage. Elle ne fait presque aucun calcul et se contente de récupérer des variables pour savoir ce qu'elle doit afficher. On y trouve essentiellement du code HTML mais aussi quelques boucles et conditions PHP très simples, pour afficher par exemple une liste de messages.
- **Contrôleur** : Cette partie gère la logique du code qui prend des décisions. C'est en quelque sorte l'intermédiaire entre le modèle et la vue. Le contrôleur va demander au modèle les données, les analyser et prendre des décisions et renvoyer le texte à afficher à la vue.

Le contrôleur contient exclusivement du PHP. C'est notamment lui qui détermine si l'utilisateur a le droit de voir la page ou non (gestion des droits d'accès) [16].

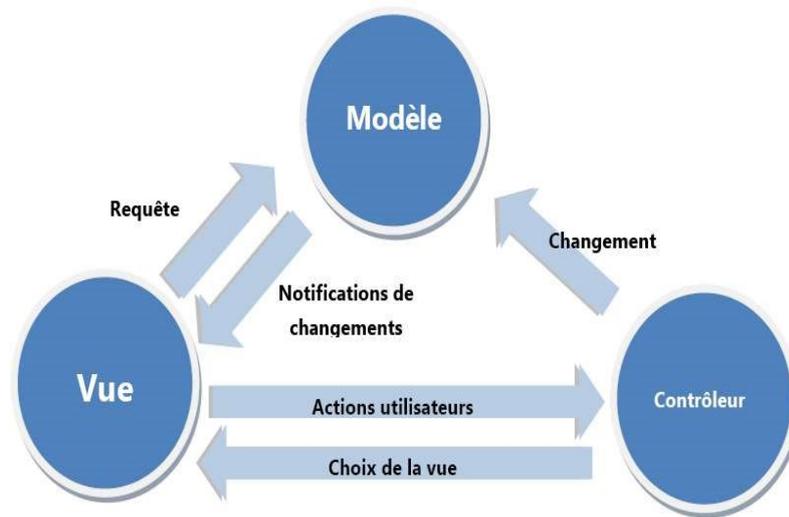


FIGURE IV.1 – Achitecture MVC [22].

2. PHP

Pre Hyertext Processor est un langage de programmation côté serveur.

Autrement dit c'est le serveur qui va interpréter du code PHP (langage de script). Il a été conçu pour permettre la création d'applications dynamiques, la génération du code (HTML, CSS, XHTML) et des données (JPEG,PNG) [17].

3. HTML

Hyper Text Markup Langage est un langage de balise utilisé pour structurer et donner du sens au contenu web.

Par exemple : Définir des paragraphes, titres et tables de données ou encore intégrer des images ou des vidéos dans une page[18] [19].

4. CSS

Les feuilles de style CSS sont utilisées pour gérer l'apparence des pages Web.

CSS est le langage de règle de style utilisé pour formater les fichiers HTML.

Exemple : Modifier la police, la taille, la couleur et l'alignement du texte [18] [19].

5. JavaScript

JavaScript est un langage de script orienté objet qui permet de réaliser des mécanismes complexes sur une page web interactif pour améliorer l'ergonomie de l'interface utilisateur et intégrer des effets esthétiques. Aussi il est chargé d'accélérer le chargement des pages Web [20] [21].

IV.2.2 Environnement de développement

1. PgAdmin

est un outil principal de gestion Open Source des bases de données PostgreSQL.

Il peut être exécuté en mode serveur web (web application) ou en mode bureau.

Il fournit une interface graphique pour la création, la maintenance et l'utilisation d'objets de base de données [22].

2. Visual Studio Code

est un éditeur de code Open Source développé par Microsoft supportant un très grand nombre de langages grâce à des extensions [23].

IV.2.3 Framework utilisé

1. Définition

Le framework est un espace de travail modulaire qui contient des composants autonomes qui permettent de faciliter le développement d'un site web ou d'une application. Ces composants résolvent des problèmes souvent rencontrés par les développeurs. Ces composants sont organisés pour une utilisation de manière interactive [24] [25].

Il existe différents types d'usage pour un framework, on distingue ainsi :

- **Les frameworks applicatifs pour le développement d'applications web** : Symfony, Django.
- **Les frameworks de présentation de contenu web ou Frameworks front-end** : Bootstrap, Semantic UI ou Foundation.
- **Les frameworks de développement de bureau** : Cocoa sur Mac ou Qt sur du multi-plateformes.
- **Les frameworks de logging** : Log4J en langage Java .
- **Les frameworks de persistance** : SQLAlchemy pour le langage Python[33].

Dans notre application nous avons utilisé deux framework : Symfony et Bootstrap.

(a) **Bootstrap**

Bootstrap est un framework développé par l'équipe du réseau social Twitter. Ce framework est Open Source (sous licence MIT) et utilise HTML, CSS et JavaScript.

Cela permet aux développeurs de créer facilement des sites Web et de fournir des outils pour tous les types d'écrans afin de développer des sites Web responsives design principalement adaptés aux smartphones. Il fournit des outils avec des styles existants tels que la typographie, les boutons et l'interface de navigation. Ce type de framework est appelé Front-End Framework [26].

(b) **Symfony**

est un ensemble de composants PHP ainsi qu'un framework MVC libre écrit en PHP et qui peuvent être pratiqué dans des projets web privés ou Open Source .

Il permet de acquitter des sites internet dynamiques de manière rapide, structurée avec un développement clair [27].

IV.2.4 Les Systèmes de gestion de base de données relationnelles (SGBDR)

1. PostgreSQL

Est un système de gestion de base de données relationnelle Open Source développé par une équipe internationale constituée de bénévoles.

Il permet de gérer les transactions, les sous-sélections, les déclencheurs, les vues et l'intégrité référentielle de clé étrangère, ainsi que le verrouillage évolué. Il s'exécute sur de nombreuses plateformes dont Linux, Windows [28].

IV.3 Captures d'écran de l'application

•La première page qui apparaît à l'utilisateur, elle sert à s'identifier avec un email et un mot de passe.

Dans la figure IV.2 ci-dessous, nous représentons l'interface d'authentification.

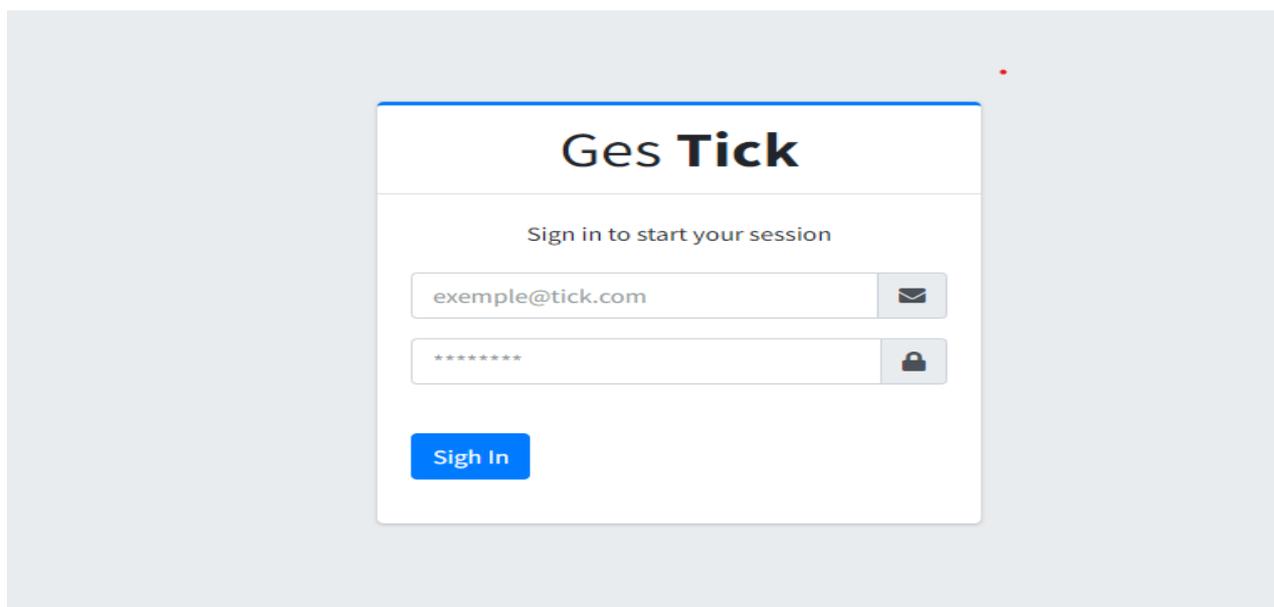


FIGURE IV.2 – Interface d'Authentification.

- Le technicien, une fois connecté est renvoyer vers cette page.

La figure IV.3 ci-dessous représente l'interface Technicien.

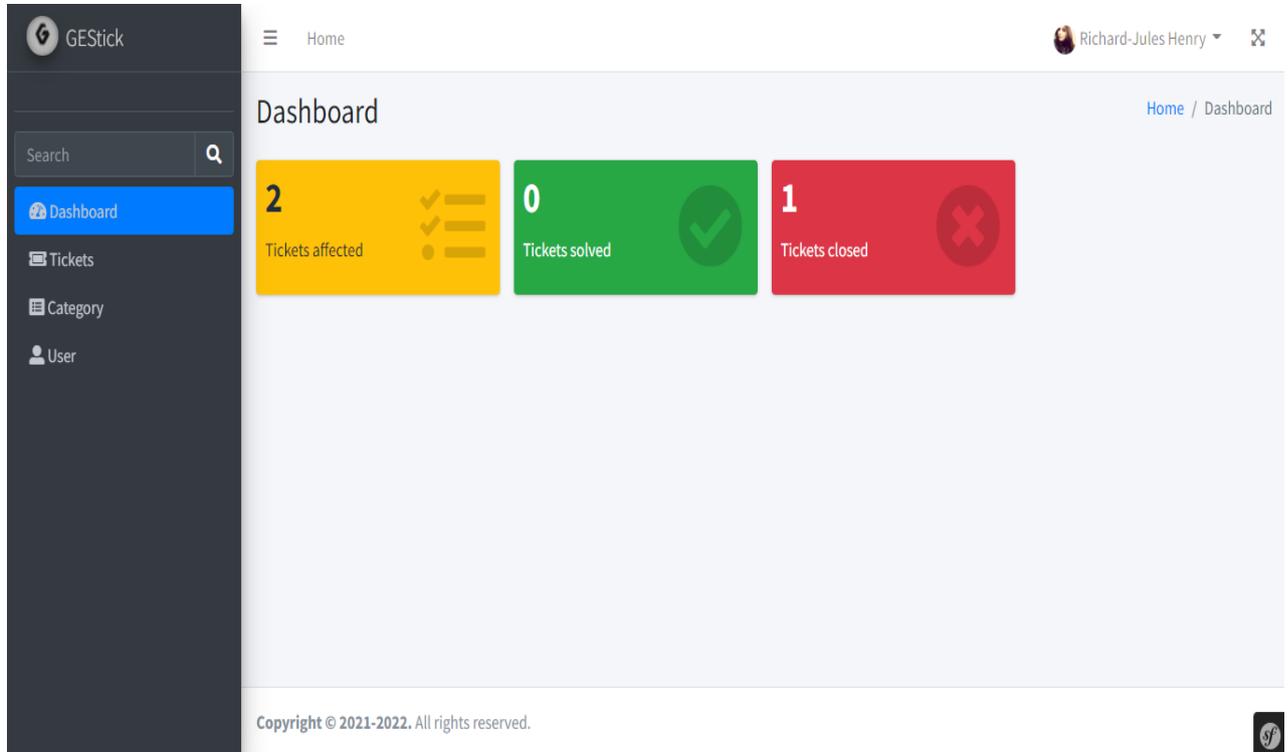


FIGURE IV.3 – Interface Technicien.

- Responsable IT, une fois connecté est renvoyé vers cette page, il peut consulter la liste des utilisateurs, ajouter, modifier des comptes et supprimer le compte de n'importe quel utilisateur.

La figure IV.4 ci-dessous représente l'interface Responsable IT.

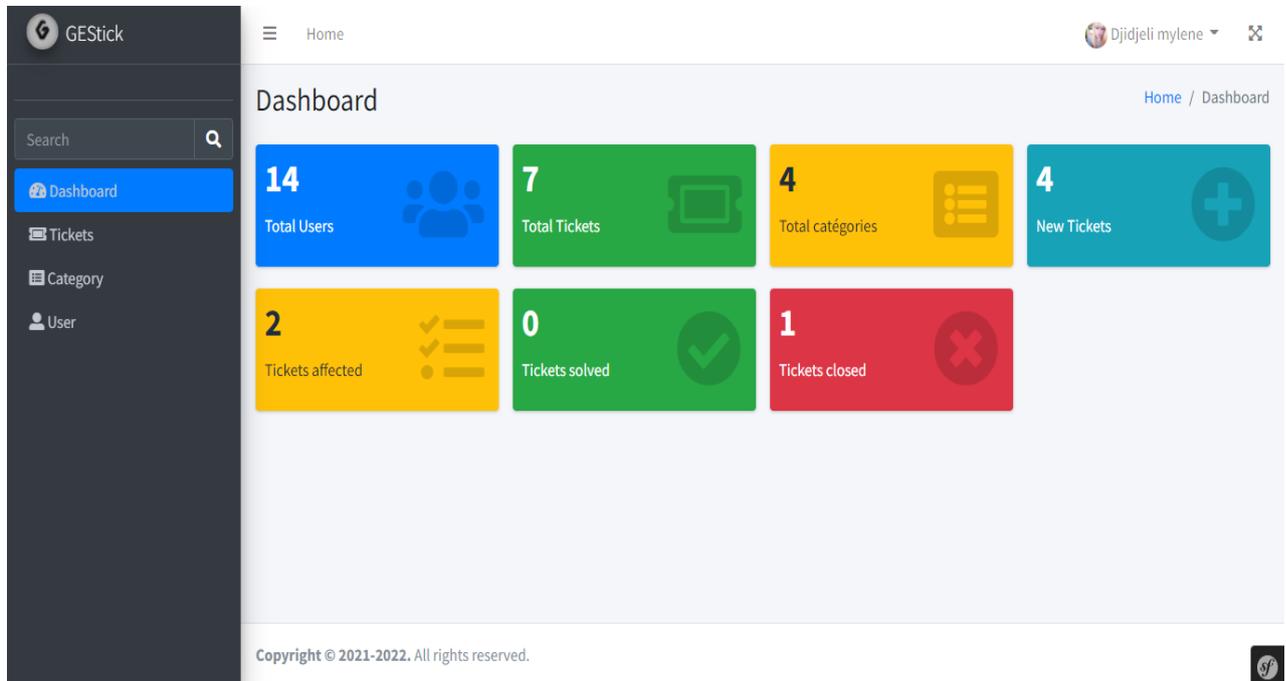


FIGURE IV.4 – Interface Responsable IT.

- La figure IV.5 ci-dessous représente l'interface liste des utilisateurs.

The screenshot displays the 'Users list' interface. On the left is a dark sidebar with the 'GEstick' logo and a search bar. Below the search bar are menu items: 'Dashboard', 'Tickets', 'Category', and 'User' (which is highlighted in blue). The main content area has a light gray header with 'Home' and a user profile 'Djadjeli mylene'. Below the header is the title 'Users list' and a breadcrumb 'Home / Users list'. A '+ Add User' button is located below the title. The main content is a table with the following data:

[#]	Id	FullName	Email	Adresse	picture
6022	Marc Lambert	imeyer@ifrance.com	Laetitia-Catherine Renaud	https://randomuser.me/api/portraits/women/61.jpg	
6023	Anouk Le Raymond	dlegendre@ferrand.fr	Monique Berthelot	https://randomuser.me/api/portraits/women/12.jpg	
6024	Zacharie de la Vincent	fblot@rocher.fr	Honoré-Thibault Laroche	https://randomuser.me/api/portraits/women/29.jpg	
6025	Timothée Legrand	gerard69@marty.fr	Gabrielle Godard	https://randomuser.me/api/portraits/women/59.jpg	
6026	Léon Lefebvre	noemi65@martinez.fr	Dorothée Gillet-Roux	https://randomuser.me/api/portraits/women/88.jpg	
6027	Manon de Potier	marine.potier@evcard.net	Marcel Jacob	https://randomuser.me/api/portraits/women/35.jpg	

FIGURE IV.5 – Interface liste des utilisateurs.

- La figure IV.6 ci-dessous représente le formulaire d'ajout d'un utilisateur.

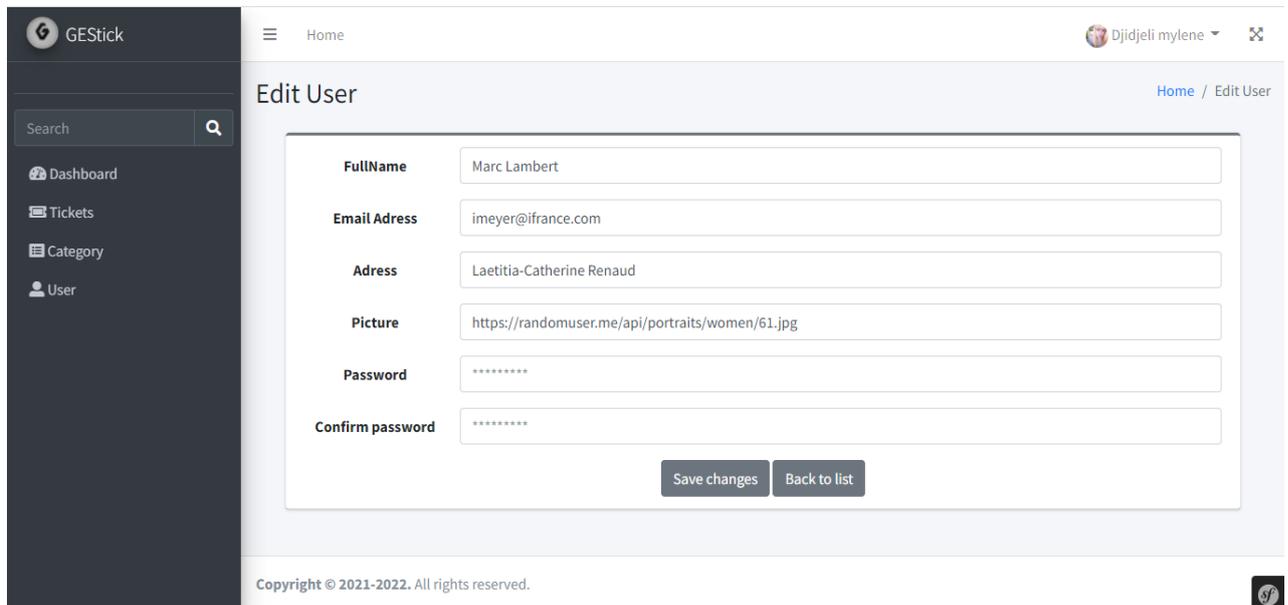
The screenshot shows the 'New User' form in the GEStick application. The interface includes a dark sidebar with navigation options: Dashboard, Tickets, Category, and User. The main content area is titled 'New' and contains a form with the following fields:

FullName	<input type="text" value="Name"/>
Email Address	<input type="text" value="exemple@gmail.com"/>
Adress	<input type="text" value="your adress"/>
Picture	<input type="text" value="your picture"/>
Password	<input type="password" value="*****"/>
Confirm password	<input type="password" value="*****"/>

At the bottom of the form, there are two buttons: 'Save' and 'Back to list'. The footer of the page contains the text 'Copyright © 2021-2022. All rights reserved.' and a small logo.

FIGURE IV.6 – Interface formulaire d'ajout.

- La figure IV.7 ci-dessous représente le formulaire qui permet de modifier une information donnée d'un utilisateur déjà existant.



The screenshot displays the 'Edit User' page in the GESTick application. On the left is a dark sidebar with a search bar and menu items: Dashboard, Tickets, Category, and User. The main content area has a light blue header with 'Home' and a user profile 'Djidjeli mylene'. Below the header, the page title is 'Edit User' with a breadcrumb 'Home / Edit User'. The form contains the following fields:

FullName	Marc Lambert
Email Adress	imeyer@ifrance.com
Adress	Laetitia-Catherine Renaud
Picture	https://randomuser.me/api/portraits/women/61.jpg
Password	*****
Confirm password	*****

At the bottom of the form are two buttons: 'Save changes' and 'Back to list'. The footer of the page includes the text 'Copyright © 2021-2022. All rights reserved.' and a small logo.

FIGURE IV.7 – Interface de modification.

- Responsable Métier, une fois connecté est renvoyer vers cette page, il peut ajouter, modifier et supprimer les catégories et consulter la liste des tickets .

La figure IV.8 ci-dessus représente l'interface Responsable Métier.

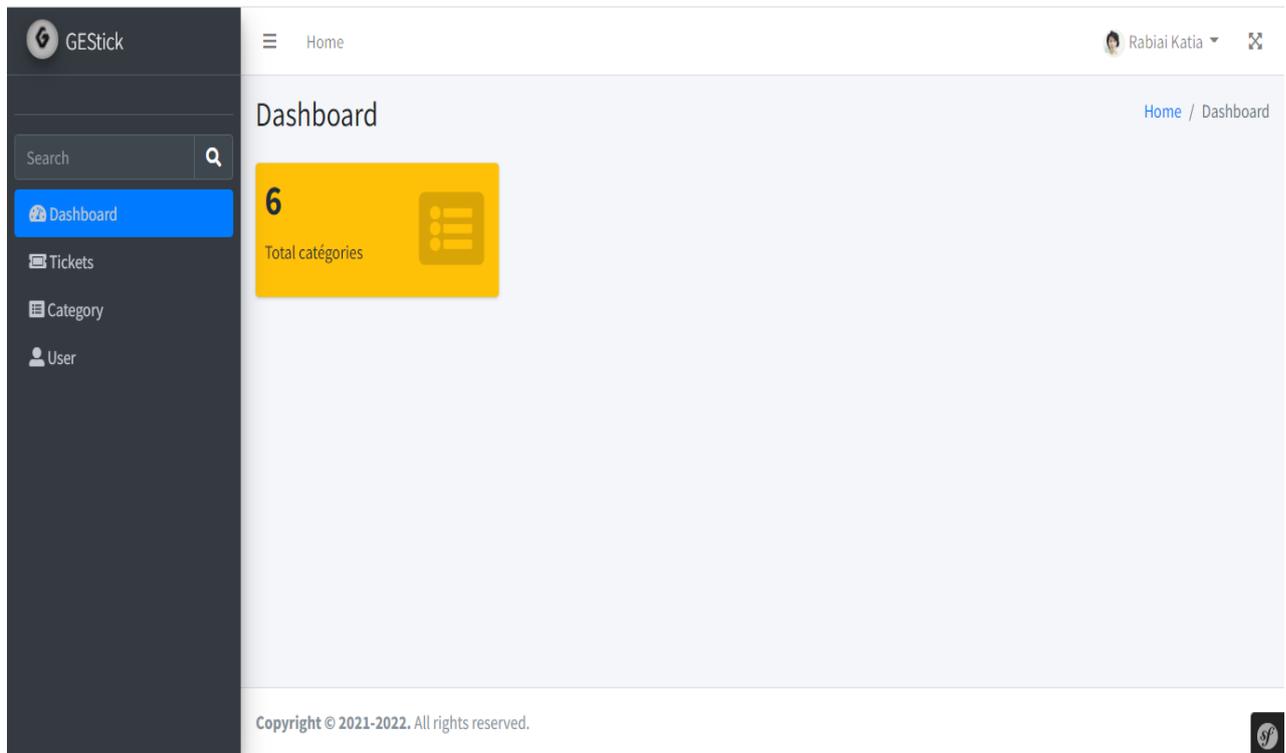


FIGURE IV.8 – Interface Responsable Métier.

- La figure IV.9 ci-dessous suivante représente la liste des tickets.

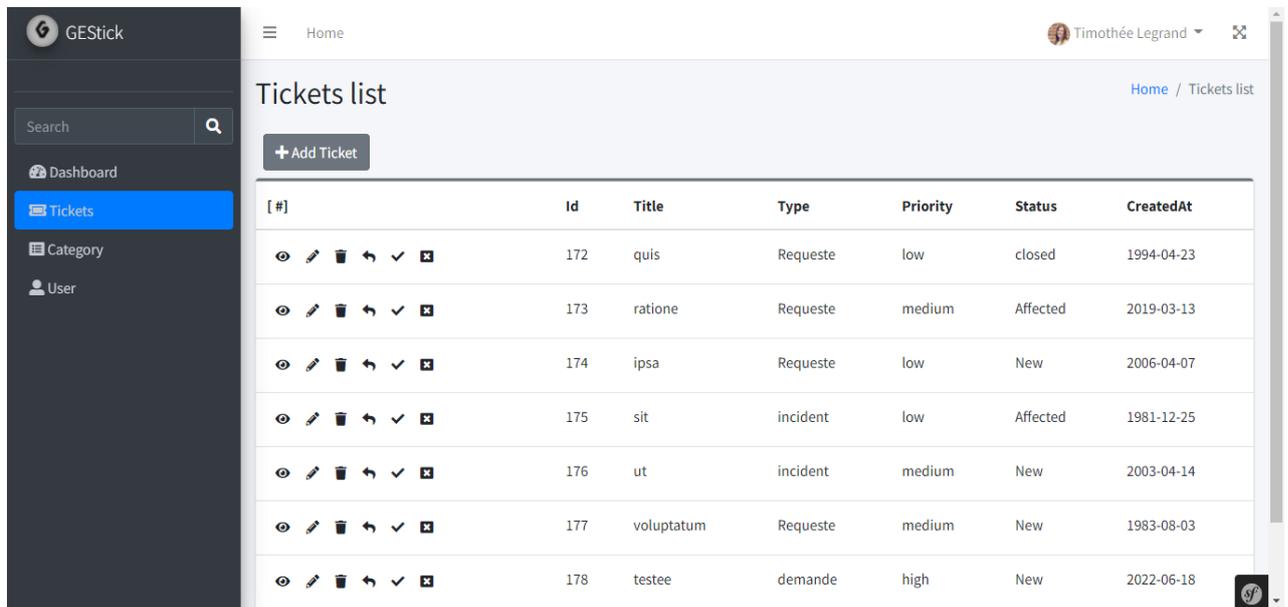


FIGURE IV.9 – Interface la liste des tickets.

- La figure IV.10 représente l'ensemble des informations propres à un utilisateur, où il a la possibilité de modifier ses informations personnelles.

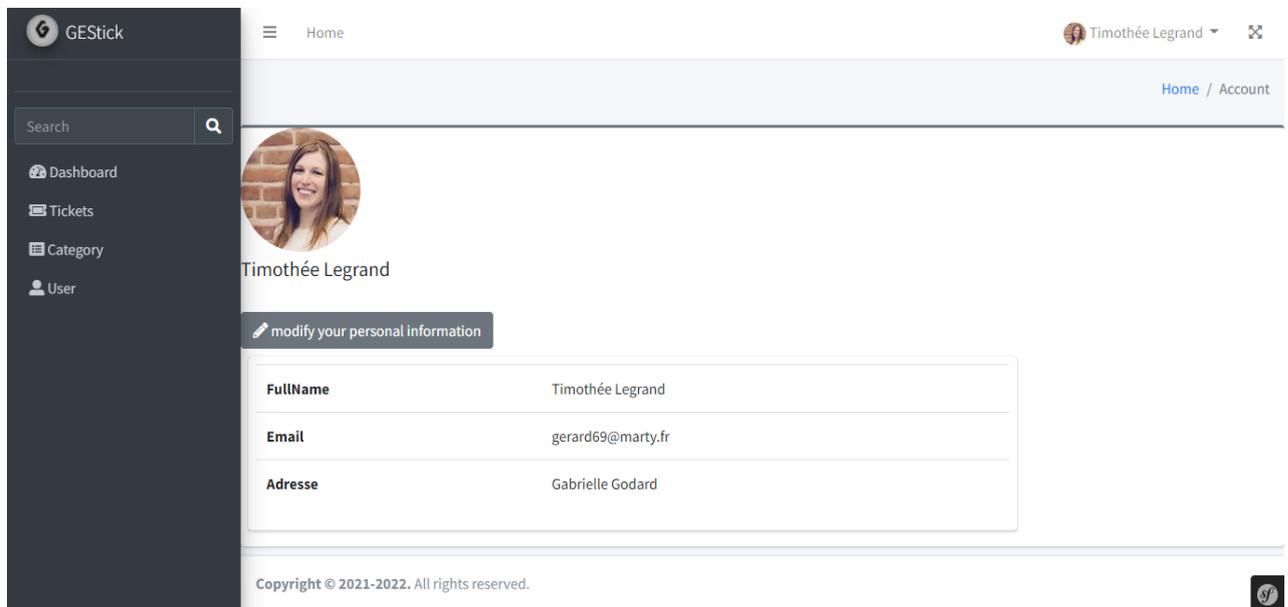


FIGURE IV.10 – Interface Modifier compte.

IV.4 Conclusion

Dans ce chapitre, nous avons commencé par définir l'environnement de développement de notre application et les langages que nous avons utilisé, puis nous avons consacré la fin de ce chapitre à exposer quelques interfaces de l'application.

Chapitre V

Conclusion générale et prespective

Dans ce mémoire, nous avons conçu et réalisé une plateforme de doléances d'un système de gestion de ticket. L'objectif de notre projet consiste à faciliter certaines tâches au profit des employés de l'entreprise. Nous nous sommes munis de plusieurs outils et langages à cette fin.

Pour ce faire nous avons implémenté les besoins de notre plateforme, nous nous sommes orientés vers la modélisation qui s'avère que c'est une étape importante pour un travail réussi.

Nous avons suivi la démarche de développement logiciel UP, le diagramme BPMN et aussi le langage de modélisation UML qui est un langage basé sur l'orienté objet et aussi pour l'élaboration des différents diagrammes à savoir :

Les diagrammes de cas d'utilisation, diagrammes de séquences et le digramme de classe.

Concernant la réalisation, nous avons choisi de programmer notre plateforme avec la collection d'outils design BOOTSTRAP qui contient des codes HTML et CSS et le langage PHP Objet avec SYMFONY comme framework et PostgreSQL comme SGBD.

Enfin, comme tout autre projet de développement logiciel non fini, des perspectives d'avenir s'imposent.

Dans notre cas on voudrait :

- Compléter la phase de réalisation de l'application selon autre conception.
- Améliorer l'interface pour répondre aux normes ergonomiques.
- Ajouter des nouvelles fonctionnalités.

Bibliographie

- [1] Michel Cotte, consulté le 10/04/2022
<https://journals.openedition.org/ephaistos/392>.
- [2] Mise en place d'une solution d'authentification RADIUS Cas Cevital de Bejaia.pdf, consulté le 10/04/2022.
- [3] Processus Unifié.pdf, consulté le 10/04/2022
<http://www-sop.inria.fr/members/Serena.Villata/Resourses/COO-s6-C02.pdf>
- [4] lucas-uzan, consulté le 22/04/2022
<https://www.lucas-uzan.fr/cest-quoi-le-processus-unifie>.
- [5] L'expérience _utilisateur, consulté le 20/04/2022
https://www.researchgate.net/publication/259602185_L'experience_utilisateur_un_concept_sans
- [6] Les 10 commandes de l'expérience Utilisateurs(UX), consulté le 11/04/2022
<https://www.optimisation-conversion.com/webdesign/les-10-commandements-de-lexperience-utilisateur-ux/>
- [7] ux_design, consulté le 22/04/2022
<https://www.verywell.digital/les-6-piliers-dune-strategie-dux-design>.
- [8] BENJAMINSCHEMIDT, consulté le 22/04/2022
<https://benjaminschemidt.com/accueil/la-demarche-ux>.
- [9] Pascal Roques,Franck Vallée,UML en action, 2ème édition2003, consulté le 22/04/2022.
- [10] *Caractéristiques des besoins fonctionnels*, consulté le 01/04/2022
https://www.researchgate.net/figure/Caracteristiques-des-besoins-fonctionnels-emotionnels-psychologiques-dans-le-cadre-de_fig3_346380300
- [11] Janiszek, *Cours de génie logiciel*, consulté le 23/04/2022.
- [12] Crozat et Stéphane, *Introduction à la modélisation conceptuelle de données avec UML*, consulté le 24/04/2022
<https://stph.scenari-community.org/bdd/mod1.pdf>
- [13] *UML les diagrammes de séquence*, consulté le 23/04/2022
<http://tvaira.free.fr/dev/uml/UML-DiagrammesDeSequence.pdf>
- [14] Erwan, *Modélisation de processus métiers avec BPMN*, consulté le 23/04/2022.
- [15] PIXEES, consulté le 24/04/2022
https://pixees.fr/informatiquelycee/n_site/snt_web_html.htm
- [16] Bekcaid,Mehdi, consulté le 26/04/2022
<https://medium.com/@belcaid.mehdi/larchitecture-logicielle-mvc-1a8bbb5cf6dc>.
- [17] Php, consulté le 21/04/2022
<https://www.php.net>.

- [18] Leoetlino et Tregagnon, *Guide de développement HTML*, consulté le 23/04/2022
<https://developer.mozilla.org/fr/docs/Web/Guide/HTML/HTML5>.
- [19] Romain.bohdanowicz et SphinxKnight, consulté le 23/04/2022
<https://developer.mozilla.org/fr/docs/Web/CSS>.
- [20] JDN, consulté le 24/04/2022
www.journaldunet.fr/web-tech/dictionnaire-du-webmastering/1203585-javascript/.
- [21] SphinxKnight, Arnaudettes et Tregagnon, *Guide JavaScript - Introduction*, consulté le 30/04/2022.
- [22] pgAdmin, consulté le 28/04/2022
<https://www.pgadmin.org/>.
- [23] Code Visual Studio, consulté le 21/04/2022
<https://edutechwiki.unige.ch/fr/Visual-studio-code>.
- [24] Arielle Moro, *Les frameworks au coeur des applications webs*, consulté le 30/04/2022
https://doc.rero.ch/record/21099/files/TDIG_48.pdf
- [25] *Les frameworks PHP les plus populaires à utiliser en 2022*, consulté le 01/05/2022
<https://kinsta.com/fr/blog/frameworks-php/>
- [26] Bootstrap, consulté le 29/04/2022
www.journaldunet.com/web-tech/developpeur/1159810-bootstrap-definition-tutoriels-astuces-pratiques/.
- [27] Potencier et Fabien, *en Route Pour Symfony 5*, consulté le 29/04/2022
<https://fr.scribd.com/document/451453016/Fabien-Potencier-En-Route-Pour-Symfony-5>.
- [28] Sébastien Namèche, *Support de cours PostgreSQL 8*, consulté le 02/05/2022
file:///C:/Users/BejaiaNet/Downloads/0233-formation-postgresql.pdf

Résumé

En vue de l'obtention du diplôme Master professionnel en génie logiciel, ce travail consiste à la réalisation d'une plateforme de doléances d'un système de gestion de ticket.

La modélisation de notre plateforme a été réalisée en suivant la démarche processus unifié UP et en utilisant le langage de modélisation UML. Concernant la réalisation de l'application, s'est faite à l'aide de l'environnement de développement Visual Studio Code, en utilisant le langage PHP sous le Framework SYMFONY, et enfin PostgreSQL comme gestionnaire de base de données.

Mots clés : UP, UML, MVC, PHP, SYMFONY, PostgreSQL.

Abstract

With a view to obtaining the professional Master's degree in software engineering, this work consists of to the realization of a complaint platform of a ticket management system. The modeling of our platform was carried out by following the UP unified process approach and using the language UML modeling. Regarding the realization of the application, made using the Visual Studio Code development environment, using the PHP language under the SYMFONY Framework, and finally PostgreSQL as the database manager data.

Keywords : UP, UML, MVC, PHP, SYMFONY, PostgreSQL.