

République Algérienne Démocratique et Populaire
Ministre de l'Enseignement Supérieur et de la Recherche Scientifique
Université Abderrahmane Mira-Bejaia

Faculté des Sciences Exactes
Département Informatique



Projet fin de Cycle

SPÉCIALITÉ : MASTER PROFESSIONNELLE GÉNIE LOGICIEL

T H È M E

Gestion des mouvements des bobines papiers et leurs traçabilité via des terminaux mobiles

Présenté par :

- IDIR TAALDJET

- MAKHLOUF BAHIA

Les jury :

- PRÉSIDENT : M^r MOUHAMMEDI MOHAMED
M.C.A UNIVERSITÉ DE BÉJAIA.

- EXAMINATRICE : M^{me} LAHLAH SOUAD
M.C.B UNIVERSITÉ DE BÉJAIA.

- ENCADRANT : M^{me} AIT ABDELOUHAB KARIMA
M.C.B UNIVERSITÉ DE BÉJAIA.

Année universitaire : 2021/2022

Dédicaces

Au mon dieu le clément et le miséricordieux louange à ALLAH le tout puissant.

Je dédie ce mémoire en premier lieu à mes parents, qui m'ont toujours soutenu et encouragé à suivre mes aspirations, mes passions et à atteindre mes objectifs.

À mon père, a celui qui m'a aidé à découvrir le " savoir " le trésor inépuisable. De tous les pères, tu as été le meilleur, tu as su m'entourer d'attention, m'inculquer les valeurs nobles de la vie, m'apprendre le sens du travail, de l'honnêteté et de la responsabilité. Tu as été et tu seras toujours un exemple à suivre pour tes qualités humaines, ta persévérance et ton perfectionnisme. Des mots ne pourront jamais exprimer la profondeur de mon respect, ma considération, ma reconnaissance et mon amour éternel.

À ma mère qui œuvre à ma réussite, de par son éducation, sa présence, son appui et son amour. A la plus douce et la plus merveilleuse de toutes les mamans. À une personne qui m'a tout donné sans compter. Aucun hommage ne saurait transmettre à sa juste valeur ; l'amour, le dévouement et le respect que je porte pour toi.

Que dieu leur procure bonne santé et longue vie.

*À mes chère sœur **Rima**, pour son encouragement permanente, et son soutien moral,*

*À mes chers frères **Riad, Bilal** pour leurs appuis et leurs encouragements,*

Et à tous ceux que ma réussite leur tient à cœur,

*À mon binôme **Bahia**, je ne vais jamais oublier nos souvenirs et tous les moments qu'on a passé ensemble, merci pour tout, que dieu te bénisse et te protège.*

*À mes très chères copines **Dalila, Amel, Nardjes, Ferial, Nadjia, Zakia, Nabila**, dont nous avons partagé les meilleurs et les plus agréables moments.*

Que ce travail soit l'accomplissement de vos vœux tant allégés, et le fruit de votre soutien infailible, Merci d'être toujours là pour moi.

Idir Taaldjet.

Au mon dieu le clément et le miséricordieux louange à ALLAH le tout puissant.

Je dédie ce mémoire en premier lieu à mes parents, qui m'ont toujours soutenu et encouragé à suivre mes aspirations, mes passions et à atteindre mes objectifs.

*À mon père "**Rachid**", à celui qui m'a aidé à découvrir le " savoir " le trésor inépuisable. Des mots ne pourront jamais exprimer la profondeur de mon respect, ma considération, ma reconnaissance et mon amour éternel.*

*À ma mère "**Nassima**" qui œuvre à ma réussite, de par son éducation, sa présence, son appui et son amour. À la plus douce et la plus merveilleuse de toutes les mamans.*

Que dieu leur procure bonne santé et longue vie.

*À mon chère frère **Yassin**, pour son encouragement permanent,*

*À mes chers sœurs **Djoudja**, **Wissam** pour leurs appuis et leurs encouragements,*

*À toute ma famille **MAKHOLOUF** et **KHELIFA**,*

*À mon fiancé **Kamel**, pour son appuis et son encouragements,*

Et à tous ceux que ma réussite leur tient à cœur,

*À mes très chères copines **Fahima**, **Wahiba**, **Meriem**, dont nous avons partagé les meilleurs et les plus agréables moments.*

*À ma binôme **Alja**, je ne vais jamais oublier nos souvenirs et tous les moments qu'on a passé ensemble, merci pour tout, que dieu te bénisse et te protège.*

Que ce travail soit l'accomplissement de vos vœux tant allégés, et le fruit de votre soutien infailible,

Merci d'être toujours là pour moi.

Makhlouf Bahia.

Remerciements

Nous tenons tout d'abord à remercier Dieu le tout puissant et miséricordieux, qui nous a donné la force et la patience d'accomplir ce Modeste travail.

*En second lieu, nous tenons à remercier notre encadrant **M^{me} Ait Abdelouhab Karima**, pour sa disponibilité, sa patience et son précieux suivi tout au long de la réalisation de ce travail.*

Nous remercions les membres de jury qui nous font l'honneur de présider et d'examiner ce modeste travail.

Nous tenons également à remercier du département d'Informatique de l'Université Abderrahmane Mira de Bejaïa qui nous ont prodigué durant les cinq ans de notre formation.

*Nos sincères remerciements vont à tous les membres de l'entreprise « **Général Emballage** », en particulier : **Mr. Chikhi Hakim** directeur des systèmes d'information, et **Mr. Mohand Arezki Lahlou** chef de département informatique pour leurs suivi et leurs soutiens au cours de ce stage notamment pour l'effort fournis pour aboutir à notre objectif et **Mr. Ben Maamer Zahir** responsable de gestion.*

Enfin, nous tenons également à remercier toutes les personnes qui ont participé de près ou de loin à la réalisation de ce travail

Table des matières

| | |
|---|------|
| TABLE DES FIGURES | VIII |
| LISTE DES TABLEAUX | IX |
| INTRODUCTION GÉNÉRALE | 1 |
| I PRÉSENTATION DE L'ORGANISME D'ACCUEIL ET ÉTUDE PRÉLIMINAIRE | 3 |
| I.1 INTRODUCTION | 3 |
| I.2 PRÉSENTATION DE L'ENTREPRISE GÉNÉRAL EMBALLAGE | 3 |
| I.2.1 Historique | 4 |
| I.2.2 Domaine d'activité de l'entreprise à l'international | 5 |
| I.2.3 La matière première de Général Emballage | 6 |
| I.2.4 Organigramme Général de L'entreprise | 6 |
| I.2.5 Organisation de service de stock bobines de General Emballage | 8 |
| I.3 PROBLÉMATIQUE | 9 |
| I.4 SOLUTION PROPOSÉE | 9 |
| I.5 DÉFINITIONS RELATIVES AUX STOCKS | 10 |
| I.6 GÉNÉRALITÉS RELATIVES AUX CODES-BARRES | 10 |
| I.6.1 Définition du code-barres | 11 |
| I.6.2 Types des codes-barres | 11 |
| I.6.3 Appareil de lecture de code-barres | 12 |
| I.6.4 Avantages du code-barres | 14 |

| | | |
|--------|---|----|
| I.6.5 | Type de code-barres utiliser dans l'implémentation | 14 |
| I.7 | DÉMARCHE DE DÉVELOPPEMENT | 15 |
| I.7.1 | Langage de modélisation du processus unifié | 15 |
| I.7.2 | Définition du Processus Unifié | 16 |
| I.8 | MAQUETTE | 17 |
| I.8.1 | Les principales interfaces (maquettes) | 18 |
| I.9 | CONCLUSION | 23 |
| II | ANALYSE ET SPÉCIFICATION DES BESOINS | 24 |
| II.1 | INTRODUCTION | 24 |
| II.2 | SPÉCIFICATION DES BESOINS | 24 |
| II.2.1 | Besoins fonctionnels | 25 |
| II.2.2 | Besoins non fonctionnels | 25 |
| II.3 | IDENTIFICATION DES ACTEURS | 26 |
| II.4 | DIAGRAMME DE CONTEXTE | 26 |
| II.5 | IDENTIFICATION DES CAS D'UTILISATIONS | 28 |
| II.6 | DIAGRAMME DE CAS D'UTILISATION | 28 |
| II.7 | DESCRIPTIONS TEXTUELLE DES CAS D'UTILISATION | 30 |
| II.8 | LES DIAGRAMMES DE SÉQUENCES | 41 |
| II.8.1 | Diagramme de séquence « Authentification » | 42 |
| II.8.2 | Diagramme de séquence « Ajouter Bobine » | 43 |
| II.8.3 | Diagramme de séquence « Modifier bobine » | 44 |
| II.8.4 | Diagramme de séquence « Recherche bobine » | 45 |
| II.8.5 | Diagramme de séquence « Gérer bobine » | 46 |
| II.8.6 | Diagramme de séquence « Etablir un bon de réception » | 47 |
| II.8.7 | Diagramme de séquence « Imprimer bon de réception » | 48 |
| II.8.8 | Diagramme de séquence « Réintégrer les bobine » | 49 |
| II.8.9 | Diagramme de séquence « Créer un code-à-barre » | 50 |
| II.9 | MODÈLE DE DOMAINE | 51 |
| II.10 | CONCLUSION | 51 |

| | | |
|---------|--|----|
| III | CONCEPTION | 52 |
| III.1 | INTRODUCTION | 52 |
| III.2 | DIAGRAMME D'INTERACTION | 52 |
| III.2.1 | Diagramme d'interaction du cas d'utilisation « Authentification » ... | 53 |
| III.2.2 | Diagramme d'interaction du cas d'utilisation « Ajouter bobine » | 54 |
| III.2.3 | Diagramme d'interaction du cas d'utilisation « Modifier Bobine » | 55 |
| III.2.4 | Diagramme d'interaction du cas d'utilisation « Rechercher Bobine » .. | 56 |
| III.2.5 | Diagramme d'interaction du cas d'utilisation « Etablir un bon de réception » | 57 |
| III.2.6 | Diagramme d'interaction du cas d'utilisation « Imprimer bon de réception » | 58 |
| III.2.7 | Diagramme d'interaction du cas d'utilisation « Réintégrer les bobines » | 59 |
| III.3 | DIAGRAMME DE CLASSE | 59 |
| III.4 | DICTIONNAIRE DE DONNÉES | 61 |
| III.5 | COMPRÉHENSION DES RÈGLES | 64 |
| III.6 | RÈGLES DE PASSAGE AU MODÈLE RELATIONNELLE | 65 |
| III.7 | MODÈLE RELATIONNELLE | 65 |
| III.8 | CONCLUSION | 65 |
| IV | IMPLÉMENTATION | 66 |
| IV.1 | INTRODUCTION | 66 |
| IV.2 | ENVIRONNEMENT ET OUTILS DE DÉVELOPPEMENT | 66 |
| IV.3 | L'ORGANIGRAMME DE L'APPLICATION | 70 |
| IV.4 | PRÉSENTATION DES INTERFACES | 70 |
| IV.4.1 | Les interfaces dédiées au navigateur web | 70 |
| IV.4.2 | Les interfaces dédié au Terminal mobile « Zebra MC33 » | 74 |
| IV.5 | CONCLUSION | 78 |
| | CONCLUSION GÉNÉRALE ET PERSPECTIVES | 79 |
| | BIBLIOGRAPHIE | 80 |

Table des figures

| | | |
|------|--|----|
| I.1 | Historique de l'entreprise. | 4 |
| I.2 | La Bobine. [29] | 6 |
| I.3 | Organigramme de l'entreprise.[29] | 7 |
| I.4 | Organisation de service de stock. | 8 |
| I.5 | Code-barres 1D. | 11 |
| I.6 | Code-barres 2D | 12 |
| I.7 | Zebra MC33 | 14 |
| I.8 | Code-barres bobine | 14 |
| I.9 | Les phases et les activités du processus unifié. | 16 |
| I.10 | Maquette ' Page d'accueil'. | 18 |
| I.11 | Maquette 'page d'authentification'. | 18 |
| I.12 | Maquette 'Espace Gestionnaire de stock'. | 19 |
| I.13 | Maquette 'Gestion Bobines'. | 20 |
| I.14 | Maquette 'Bon de réception'. | 21 |
| I.15 | Maquette 'Créer code-à-barre'. | 21 |
| I.16 | Maquette 'Espace Magasinier'. | 22 |
| I.17 | Maquette 'Espace Administrateur'. | 22 |
| I.18 | Maquette 'Espace Responsable dépôt'. | 23 |
| II.1 | Diagramme de contexte de notre système. | 26 |
| II.2 | Diagramme de cas d'utilisations. | 29 |
| II.3 | Diagramme de séquence système du cas d'utilisation « Authentification ». | 42 |
| II.4 | Diagramme de séquence système du cas d'utilisation « Ajouter bobine ». | 43 |
| II.5 | Diagramme de séquence système du cas d'utilisation « Modifier bobine ». | 44 |
| II.6 | Diagramme de séquence système du cas d'utilisation « Recherche bobine ». | 45 |
| II.7 | Diagramme de séquence système du cas d'utilisation « Gérer les bobines ». | 46 |
| II.8 | Diagramme de séquence système du cas d'utilisation « Etablir bon de réception ». | 47 |

| | |
|--|----|
| II.9 Diagramme de séquence système du cas d'utilisation « Imprimer bon de réception ». | 48 |
| II.10 Diagramme de séquence système du cas d'utilisation « Réintégrer les bobine ». | 49 |
| II.11 Diagramme de séquence système du cas d'utilisation « Créer un code-à-barre ». | 50 |
| II.12 Le modèle de domaine. | 51 |
| III.1 Diagramme d'interaction du cas d'utilisation « Authentification ». | 53 |
| III.2 Diagramme d'interaction du cas d'utilisation « Ajouter bobine ». | 54 |
| III.3 Diagramme d'interaction du cas d'utilisation « Modifier bobine ». | 55 |
| III.4 Diagramme d'interaction du cas d'utilisation « Rechercher bobine ». | 56 |
| III.5 Diagramme d'interaction du cas d'utilisation « Etablir un bon de réception ». | 57 |
| III.6 Diagramme d'interaction du cas d'utilisation « Imprimer bon de Réception ». | 58 |
| III.7 Diagramme d'interaction du cas d'utilisation « Réintégration des bobines ». | 59 |
| III.8 Diagramme de Classe. | 60 |
| IV.1 Organigramme de l'application | 70 |
| IV.2 Interface d'accueil.[29] | 70 |
| IV.3 Interface d'authentification. | 71 |
| IV.4 Espace Gestionnaire. | 71 |
| IV.5 Interface Gestion Bobine. | 72 |
| IV.6 Interface de création de code-à-barre. | 72 |
| IV.7 Interface de création de bon de sortie. | 73 |
| IV.8 Bon de sortie imprimable. | 73 |
| IV.9 Interface de réintégration des bobines. | 74 |
| IV.10 Interface d'accueil. | 74 |
| IV.11 Interface d'authentification. | 75 |
| IV.12 Espace gestionnaire. | 75 |
| IV.13 Espace Gestion Bobine. | 76 |
| IV.14 Espace de création de code-à-barre. | 76 |
| IV.15 Espace de création de bon de sortie. | 77 |
| IV.16 Espace de réintégration des bobines. | 77 |

Liste des tableaux

| | | |
|-------|---|----|
| I.1 | Marchés ciblés. | 5 |
| I.2 | Statistiques de l'entreprise. | 5 |
| I.3 | représente les caractéristiques de terminal mobile Zebra MC33 | 13 |
| II.1 | Messages échangés entre les acteurs et le système. | 27 |
| II.2 | Identification des cas d'utilisations. | 28 |
| II.3 | Description textuelle du cas d'utilisation « Authentification ». | 30 |
| II.4 | Description textuelle du cas d'utilisation « Gérer bobines ». | 31 |
| II.5 | Description textuelle du cas d'utilisation « Gérer types papier ». | 32 |
| II.6 | Description textuelle du cas d'utilisation « Etablir un bon de réception ». | 33 |
| II.7 | Description textuelle du cas d'utilisation « Demande bobine ». | 34 |
| II.8 | Description textuelle du cas d'utilisation « Réintégration des bobines ». | 35 |
| II.9 | Description textuelle du cas d'utilisation « Etablir le bon de sortie ». | 36 |
| II.10 | Description textuelle du cas d'utilisation « Etablir le bon d'entrée ». | 37 |
| II.11 | Description textuelle du cas d'utilisation « Consulter le stock bobines ». | 38 |
| II.12 | Description textuelle du cas d'utilisation « Gérer E/S ». | 39 |
| II.13 | Description textuelle du cas d'utilisation « Créer un code-à-barre ». | 40 |
| II.14 | Description textuelle du cas d'utilisation « Imprimer ». | 41 |
| III.1 | Dictionnaire de données | 63 |

Introduction générale

L'informatique, cette science de travail de l'information est considérée comme le support des connaissances dans les domaines scientifiques, économiques, sociaux et industriels notamment à l'aide des machines automatiques.

Le monde connaît une avancée technologique considérable dans tous les secteurs qui étudie les techniques de traitement automatique de l'information de l'entreprise et d'autres établissements.

En effet, l'informatique s'est imposée d'une manière très impressionnante dans les entreprises, cela est dû à son apport extraordinaire dans le domaine de gestion des bases de données.

Vu les avantages d'automatisations des systèmes d'information, l'exploitation des techniques avancés de traitement de l'information est devenue une nécessité dans la gestion des entreprises, et cela pour assurer une bonne démarche et une gestion efficace de leurs ressources par l'utilisation d'un logiciel.

Les logiciels de gestions de stock ont grandement facilité le travail des entreprises, et c'est dans ce cadre précis que s'inscrit notre étude. En effet, il s'agit de mettre en œuvre une application de gestion de stock pour l'entreprise "Général Emballage-Akbou", une entreprise spécialisée dans la fabrication et la transformation du carton ondulé.

Durant la période de notre stage au sein de l'entreprise General Emballage, nous avons pu identifier certains problèmes liés à la gestion de stock des bobines papier nous ont permis de déceler les insuffisantes suivantes : une lourdeur dans la gestion du stock lors le saisie manuel et une perte du temps une telle gestion conduit souvent à des erreurs de saisie.

L'objectif visé par cette application est de permettre entre autres la gestion des mouvements des bobines papier, l'attribution des codes à barres pour cette dernière, et l'impression des différents documents (bons de commande, réception, etc.).

Pour aboutir à cet objectif, notre mémoire est organisé en quatre chapitres ;

Le premier chapitre intitulé « Présentation de l'organisme d'accueil et Étude préliminaire » a pour objectif de présenter l'entreprise « Général Emballage », cerner les problèmes auxquels

l'entreprise est confrontée. Ensuite, nous présentons la méthode de travail qui est le processus unifié ainsi que le langage de modélisation UML.

Le second chapitre intitulé « analyse et spécification des besoins » consiste à collecter, et définir les besoins fonctionnels et non fonctionnels, identifier les acteurs du système et les cas d'utilisation, et analyser les besoins en mettant l'accent sur les interactions entre les acteurs et le système.

Le troisième chapitre intitulé « Conception » est consacré à la conception de l'application et présente les différents aspects conceptuels du projet.

Le dernier chapitre intitulé « Réalisation » a pour objectif de présenter les différents outils ainsi que les langages de programmation utilisés. Nous terminons ce chapitre avec l'arborescence de l'application et quelques interfaces de celle-ci.

Et pour finir, nous allons clôturer ce mémoire avec une conclusion générale résumant les connaissances acquises durant la réalisation du projet et quelques perspectives.

Présentation de l'organisme d'accueil et étude préliminaire

I.1 Introduction

Dans ce chapitre, nous allons présenter d'abord l'entreprise général emballage, son organigramme, son domaine d'activité à l'international. Ensuite, nous allons définir la problématique et déterminer les objectifs de notre application. Par la suite, nous allons donner quelques généralités sur la gestion de stock et les codes-barres. Puis, nous allons définir le processus unifié qui est notre choix pour développer notre application. Enfin, nous allons terminer ce chapitre par une conclusion.

I.2 Présentation de l'entreprise Général Emballage



Général Emballage est leader en Algérie de l'industrie du carton ondulé. C'est une entreprise papetière spécialisée dans la fabrication et la transformation de carton ondulé. En fait, l'entreprise a été créée en 2000, par Ramdane Batouche qui assure aujourd'hui la présidence du Conseil d'administration de la SPA (Société Par Actions). Général Emballage est le plus grand producteur de carton ondulé en Afrique.

I.2.1 Historique

La figure I.1 représente l'historique de entreprise :

| | |
|------|---|
| 2000 | La création de l'entreprise « General Emballage » avec un capital de 32 million DZD dans la Zone d'activités de Taharacht à Akbou dans la wilaya de Béjaia. |
| 2002 | L'entrée en production de l'usine avec un effectif de 83 employés. |
| 2006 | Le capital est porté à 150 millions DZD avec un effectif de 318 employés |
| 2007 | L'entrée en production de l'unité de Sétif, le capital est porté à 1,23 milliard DZD avec un effectif de 425 employés. |
| 2008 | Le début d'exportation vers la Tunisie et entrée en exploitation de l'unité d'Oran. |
| 2009 | Une augmentation du capital à 2 milliards DZD et entrée de Maghreb Invest avec une participation de 40 %. L'effectif est de 597 employés. |
| 2010 | Son chiffre d'affaires et son résultat net sont respectivement de 38,053 millions de dollars et 2,071 millions de dollars, ce qui la place au 50e rang des entreprises algériennes. |
| 2011 | Les trois usines de production d'Akbou, Oran et Sétif cumulaient une capacité de production de 130 000 tonnes équivalent à 80 % de la consommation algérienne. Toujours dans la même année, l'effectif de 699 employés. |
| 2012 | L'usine d'Oran est transférée à la Zone Industrielle de Hassi-Ameur. Signature d'une Convention cadre de partenariat avec l'Université de Béjaia. |
| 2013 | Général Emballage reçoit la certification ISO 9001 :2008 |
| 2014 | Début des exportations vers la Libye. |
| 2015 | L'entreprise reçoit le Prix d'encouragement du trophée Export 2014 (World Trade Center). |
| 2016 | La sortie de Maghreb Private Equity Fund et entrée de Développement Partners International (DPI) à hauteur de 49 % du capital social. |

Figure I.1 – Historique de l'entreprise.

- **Marchés ciblés :**

Le tableau I.1 représente les marchés ciblés de l'entreprise.

| Marché nord-africain | Marché européen |
|--|--------------------------------------|
| - Tunisie. - Maroc. - Lybie. - Mali. - Niger. - Mauritanie. | - France. - Espagne. - Italie. |

Table I.1 – Marchés ciblés.

- **Statistiques de l'entreprise :**

Le tableau I.2 représente les statistiques de l'entreprise.

| Effectif | Sites de production | Chiffre d'affaire en 2018 |
|------------------------|----------------------------|--|
| Plus de 1200 employés. | 3. | Plus de 20 milliards de DA (hausse de plus de 20% par rapport à 2017). |

Table I.2 – Statistiques de l'entreprise.

I.2.2 Domaine d'activité de l'entreprise à l'international

- Export de produits dans le domaine du carton ondulé (Plaque, Caisses, Barquettes, etc.) majoritairement en : Tunisie, Maroc, Lybie, Mali, Mauritanie, Espagne, Italie, France, etc.
- Par l'exportation directe : l'entreprise commercialise elle-même ses produits à l'étranger à travers une force de vente située à l'étranger, un bureau en Tunisie et de partenaires en Europe.
- L'entreprise participe très souvent à de nombreux salons afin de marquer sa présence dans l'international et ainsi gagner de nouveaux marchés.

I.2.3 La matière première de Général Emballage

La figure I.2 représente la matière première de cette entreprise et ses caractéristiques :

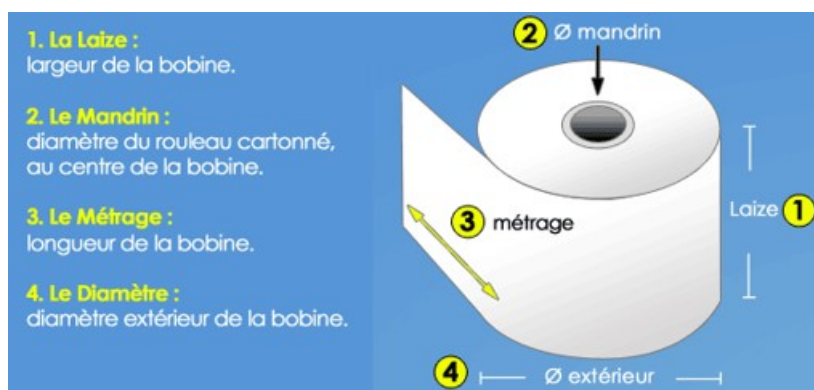


Figure I.2 – La Bobine. [29]

I.2.4 Organigramme Général de L'entreprise

Le diagramme de la figure I.3, donne une vue générale sur les différents organes constituant le complexe Général Emballage :

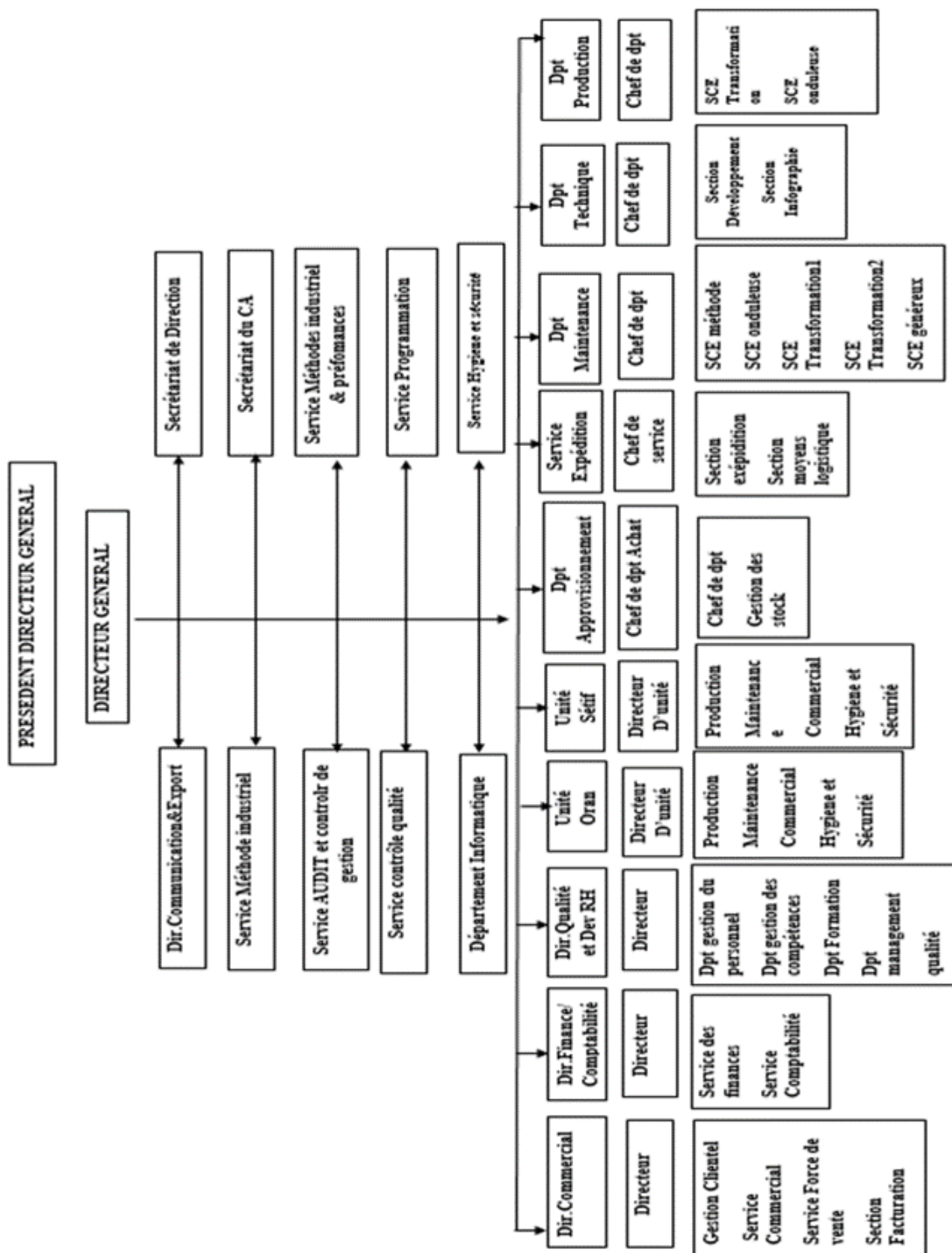


Figure I.3 – Organigramme de l'entreprise.[29]

I.2.5 Organisation de service de stock bobines de General Emballage

Le diagramme de la figure ci-dessous, représente les différents dépotes de stockage du la matière première :

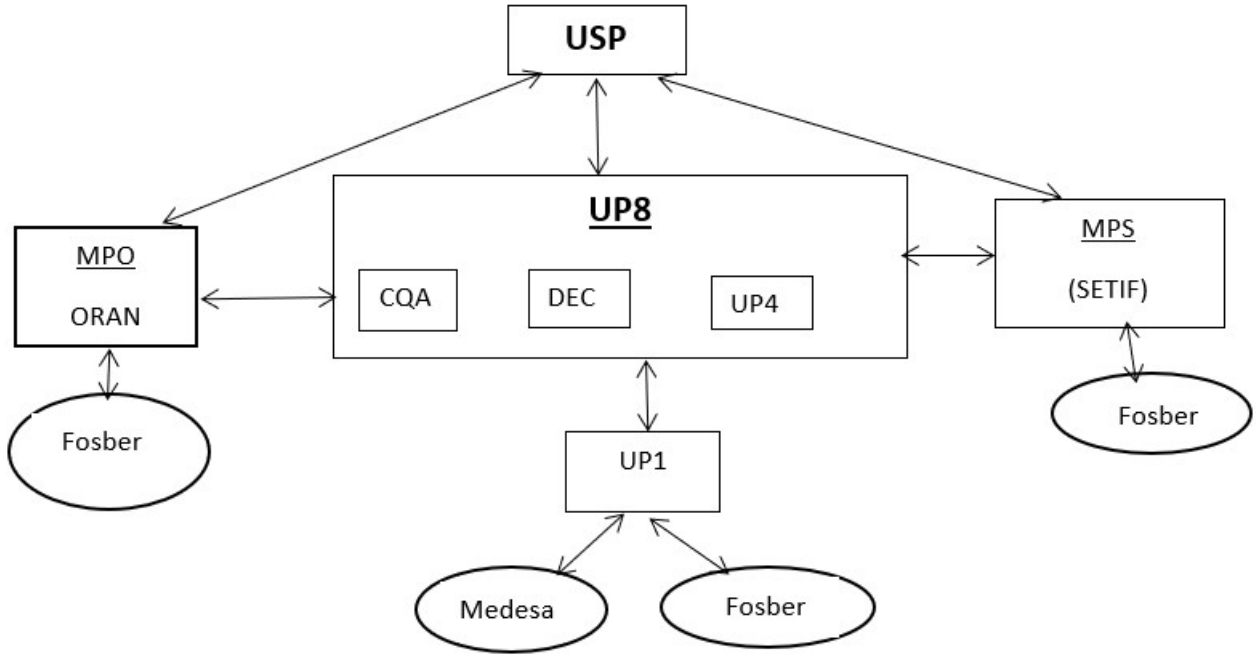


Figure I.4 – Organisation de service de stock.

BR : Réception des bobines papier(UP8).

TR : Transfère Bobines entre dépôt (UP8, UP1, MPS et MPO).

CM : Consommation Bobines (UP1, MPS et MPO).

RT : Réintégration Bobines (UP1, MPS et MPO).

USP : Espace de stockage dans le port.

UP8 : Dépôt d'Akbou (Stock principal).

MPO : Dépôt Oran.

MPS : Dépôt Sétif.

CQA : Dépôt pour les bobines à contrôler.

DEC : Dépôt pour les bobines déclassées.

UP4 : Dépôt pour les bobines mouillées.

UP1 : Dépôt pour les bobines consom-

mables.

I.3 Problématique

Durant la période de notre stage au sein de l'entreprise General Emballage, nous avons pu identifier certains problèmes liés à la gestion de stock des bobines papier.

Des entretiens répétés avec le responsable du service de gestion des stocks bobines nous font permis de déceler les insuffisantes suivantes :

- Une lourdeur dans la gestion du stock lors de la réception, la réintégration, et le transfères des bobines entre dépôts en utilisant des documents papiers.
- Une telle gestion conduit souvent à des erreurs de saisie et une perte de temps.
- Une faible sécurité des informations concernant les documents.

Afin de faciliter le suivi de stock, ils ont décidé d'inclure l'utilisation d'un terminal mobile pour lire les codes-à-barres, et pour pouvoir utiliser ce terminal, il faut exploiter le code à barres de la bobine à son arrivé, et ce que nous avons constaté que :

- Ce n'est pas toutes les bobines qui ont un code à barres ;
- Les codes à barres ne sont pas standardisés (c'est a dire on a plusieurs type de codes-à-barres(code128,code QR,...)).

I.4 Solution proposée

Le but de notre projet est de parvenir à une solution pour résoudre les problèmes cités précédemment, et pour ce faire nous proposons de développer une application de gestion pour suivre les mouvements des bobines, à l'aide des terminaux mobiles. Cette application doit pouvoir assurer les fonctionnalités suivantes :

- Gestion des bobines ;
- Gestion des types papiers ;
- Etablir des bons ;
- Attribuer un code à barres propre pour chaque bobine afin de les standardisés ;
- Assurer une traçabilité optimale ;
- Suivre le stock plus rapidement et plus précisément ;
- Éviter les erreurs liées à la saisie manuelle.

I.5 Définitions relatives aux stocks

Dans ce qui suit, nous allons donner quelques définitions relatives aux stocks.

- ✓ **La gestion** : la mise en œuvre des ressources pour atteindre des objectifs dans un cadre bien déterminé [1].
- ✓ **Le stock** : représente les biens achetés, transformés ou à vendre dans une entreprise à un moment donné. Il est ainsi constitué de matières premières, matières consommables, composants achetés ou fabriqués, sous-ensembles achetés ou fabriqués, articles sous-traités, produits finis, articles de rechange, articles défectueux à retoucher, articles obsolètes, emballages, etc. [2].
- ✓ **Les types de stock** : Dans le processus de production, les stocks sont utilisés à plusieurs niveaux. On distingue différents types [2].
 - Le stock de matières premières ;
 - Le stock des produits en cours de fabrication ;
 - Le stock des produits finis ;
 - Le stock des marchandises.
- ✓ **Objectifs de la gestion de stock** : permet de diminuer les coûts de stockage, tout en disposant d'un niveau de stock nécessaire permettant d'éviter les ruptures du stock [2].

I.6 Généralités relatives aux codes-barres

Le premier brevet d'un code-barres a été déposé le 20 octobre 1949 par deux étudiants du *Drexel Institute of Technology* : **Joseph Woodland** et **Bernard Silver**. Ce brevet concernait non seulement un code à lignes verticales, mais aussi un code composé de cercles concentrique qui n'a jamais trouvé d'application pratique.

En 1970, un comité fut créé aux Etat-Unit afin de définir une codification adaptée à la grande distribution. Le 7 octobre 1970, **George Laurer**, ingénieur chez IBM proposa le code UPC (Universal Product Code). Contrairement à ce que son nom suggère, ce code à barres n'a pas été universellement adopté mais été largement utilisé aux Etat-Unit et au Canada. IL a par la suite été remplacé par un Standard international ; Le code barre EAN (European Article Numbering).

La codification EAN 13 a été adoptée par les européens en 1977, il existe aujourd'hui des codes EAN 8 et EAN 13, composés respectivement de 8 ou 13 chiffres ; le premier est employé sur des produit de petite taille tandis que le second apparait sur tous les autre Produits [27].

I.6.1 Définition du code-barres

Un code-barres est une série de barres et d'espaces d'épaisseur variable qui représente une suite de caractères numérique ou alphanumériques et qui peut être interprétée automatiquement par un lecteur optique. Ce code, qui permet une acquisition rapide et automatique de données, est utilisé dans de nombreux domaines de l'industrie à la grande distribution. Il est très simple et économique d'imprimer des codes-barres, à l'aide d'une imprimante ou par transfert thermique. Pour lire ces codes-barres, on utilise ensuite un lecteur optique qui envoie sur le code un faisceau lumineux, en générale un laser de très faible puissance [27].

I.6.2 Types des codes-barres

Il existe deux types généraux de code-barres :

- a. **Code-barres unidimensionnel (1D)** : Représentés par une série de lignes parallèles d'épaisseur variable. Selon la technologie de lecteur utilisée, le décodage pourra se faire de façons unidirectionnelle ou bidirectionnelle afin de confirmer le premier décodage (Figure I.5).



Figure I.5 – Code-barres 1D.

Quelques types de code-barres unidimensionnelles (1D) [27] :

- Les codes-barres EAN : EAN8, EAN 13, Code Universel des produits(CUP) ;
- Le code 11 : Format obsolète utilisé pour la téléphonie ;
- Le code 39 : pour le marquage des médicaments en pharmacie ;
- Le code 128 : permet de coder les 128 caractères de la table ASCII ;
- Le code Plessey et le code MSI : utilisé pour les magasins à rayons, les interpots et les inventaires ;
- Le code GS1 et ses variantes GS1-128, GS1 Data Bar et ITF-14 ;
- Le code PostBar : utilisé par les Postes Canadiennes ;
- Le code Postnet : utilisé par la poste américaine.

b. Code-barres bidimensionnels (2D) sont des codes qui utilisent une variété de symboles (rectangles, point, hexagones et autres formes géométriques). Cette forme permet d'enregistrer davantage d'information Figure (1.4), On trouve deux familles de codes 2D :

- ✓ **Les codes empilés** : sont des codes 1D empilés (Code 16K, PDF417, etc.). Ces codes peuvent être lus par les lecteurs 1D en faisant un balayage du code.
- ✓ **Les codes bidimensionnels** : sont des codes dont les motifs constituent une forme souvent rectangulaire ou carrée qui ne peuvent être lus que par des technologies de prise de photos.

Quelques types de codes-barres bidimensionnels (2D) [27] :

- Le code 1 : de domaine public, utilisé pour les étiquettes médicales et les l'industrie du recyclage ;
- Le code PDF-417 : Portable Data File, code avec une grande capacité de stockage ;
- Le DataMatrix surnommé Tag pour certaines applications ;
- Le DPM : Direct Part Mark, code DataMatrix gravé dans la matière comme le métal, le verre, le caoutchouc, ... ;
- Le flashcode : spécification issue de DataMatrix ;
- Le MaxiCode : de domaine public, utilisée par United Parcel Service ;
- Le Bokode : code expérimental, prévu pour contenir beaucoup plus d'informations que les autres codes-barres tout en étant plus petit ;
- Le code QR : Quick Response, conçu pour être décodé rapidement, stocker une grande quantité d'informations et être lu par plusieurs types d'appareil ;
- Le High CapacityColorBarcode : créé par Microsoft ;
- Le MMCC : Mobile Multi-Coloured Composite.



Figure I.6 – Code-barres 2D

I.6.3 Appareil de lecture de code-barres

Un lecteur de code-barres est un appareil électronique servant à lire les informations stockées sous forme de code-barres.

Dans notre projet, nous avons utilisé l'appareil électronique « ZEBRA MC33 » pour la lecture des code-à-barre. Le tableau I.3 représente les caractéristiques de terminal mobile ZEBRA.

➤ **Définition Le Terminal Mobile Zebra MC33**

Le terminal Zebra MC33 est un ordinateur compact capable de capturer dans différents position grâce à sa tête pivotante des codes-barres 1D, il offre un double mode de saisie de données à l'écran numérique. Son point plume et son ergonomie font partie de ses meilleurs à toute pour une adoption rapide des utilisateurs finaux (Figure I.7) [3].

| | |
|----------------------------------|--|
| Système d'exploitation | Android 7.0 |
| Processeur (CPU) | Qualcomm 8056 1.8 GHz hexa-core 64 bit |
| Taille | Brique et Tête 45° : 202,2 mm (H) x 74,7 mm (L) x 34,5 mm (P) Tête rotative : 223,4 mm (H) x 74,7 mm (L) x 34,5 mm (P) Gun : 202,6 mm (H) x 74,7 cm (L) x 163,9 mm (P) |
| Poids | Brique ° : 375g - Tête rotative : 377g - Tête 45° : 382g - Gun : 505g |
| Mémoire interne | Brique ° : 375g - Tête rotative : 377g - Tête 45° : 382g - Gun : 505g Mémoire interne : Standard : 2Go RAM / 16Go Flash Premium : 4Go RAM / 16Go Flash Premium Plus : 4Go RAM / 32Go Flash |
| Écran | 4 pouces WVGA ; couleur, antireflets rétroéclairé LED et tactile capacitif |
| Clavier | Ecran tactile capacitif Clavier rétroéclairé 29 touches (numériques), 38 touches (numérique avec touches de fonctions) ou 48 touches (alphanumériques) |
| Alimentation | Batterie Li-ion standard : 2700 mAh (Brique, tête rotative, tête 45°) Batterie Li-ion haute capacité : 5200 mAh (Tous les modèles) |
| Communication | (:) Wi-Fi 802.11 a/b/g/n/ac/d/h/i/k/r/w Bluetooth classe 4.1 ou 2,1 avec EDR |
| Lecteurs de code à barres | (:) Laser 1D : Brique, tête rotative, gun Imageur 1D/2D : Brique, tête 45, gun° Imageur 1D/2D ERI : Brique, gun |

Table I.3 – représente les caractéristiques de terminal mobile Zebra MC33



Figure I.7 – Zebra MC33

I.6.4 Avantages du code-barres

Les codes-barres permettent d'assurer :

- ❖ Une meilleure traçabilité des produits commercialisés ;
- ❖ Une meilleure gestion de stock ;
- ❖ Un meilleur contrôle sur la distribution des produits ;
- ❖ Élimine la saisie manuelle des données ;
- ❖ Réduire les erreurs humaines et augmente l'efficacité ;
- ❖ Une connaissance plus rapide des ventes réalisées pour un produit par tous ses partenaires commerciaux. [27]

I.6.5 Type de code-barres utiliser dans l'implémentation

Le code-barres utilisé dans notre travail est de type unidimensionnel (la code 128), qui contient les informations sur la bobine papier à savoir : le code papier, le Poids, la laize, l'abréviation papier.

La figure I.8, représente un exemple de code-barres bobine.



Figure I.8 – Code-barres bobine

I.7 Démarche de développement

Pour le développement de notre projet, nous avons opté d'utiliser le processus unifié (UP) qui est un cadre général très complet du processus de développement. En effet, le processus unifié est une solution de développement logiciel à adapter à tout type de projet.

Nous avons également choisi d'utiliser le langage de modélisation UML (Unified Modeling Language) qui est une partie intégrante de la démarche UP. Ces diagrammes sont largement utilisés dans chaque étape et phase de ce processus de développement.

I.7.1 Langage de modélisation du processus unifié

UML est un langage de modélisation graphique et textuelle destiné à comprendre et décrire des besoins d'une entreprise ou d'une société, spécifier et documenter des systèmes, concevoir des solutions et communiquer des points de vue. Ses deux principaux objectifs sont la modélisation de systèmes utilisant les techniques orientées objet, depuis la conception jusqu'à la maintenance, et la création d'un langage abstrait compréhensible par l'homme et interprétable par les machines. Il est divisé en deux catégories [4] :

- **Diagrammes statiques** : composés de diagrammes de classes, d'objets, des composants et de cas d'utilisation.
- **Diagrammes dynamiques** : composés de diagrammes d'activité, de séquence, d'états-transitions, de contexte système et de diagramme de collaboration.

Dans notre cas, nous allons utiliser les diagrammes UML suivants :

- **Diagramme de cas d'utilisation** : Il montre les interactions fonctionnelles entre les acteurs et le système à l'étude [4] ;
- **Diagramme de séquence système** : Il montre la séquence verticale des messages passés entre objets au sein d'une interaction [4] ;
- **Diagramme d'interaction** : représenter le comportement interactif d'un système, se concentrent sur la description du flux de messages au sein d'un système [4] ;
- **Diagramme de classe** : montre les briques de base statiques : classes, associations, interfaces, attributs, opérations, généralisations, etc [4].

I.7.2 Définition du Processus Unifié

Le Processus Unifié (UP, pour Unified Process) est un processus de développement logiciel « itératif et incrémental, centré sur l'architecture, conduit par les cas d'utilisation et piloté par les risques » [4].

Le processus unifié se déroule en quatre phases, inception, élaboration, construction et transition. Et chaque itération est composée de cinq activités : capture des besoins, analyse, conception, implémentation et test.

La figure I.9 décrit les différentes phases et activités du processus unifié :

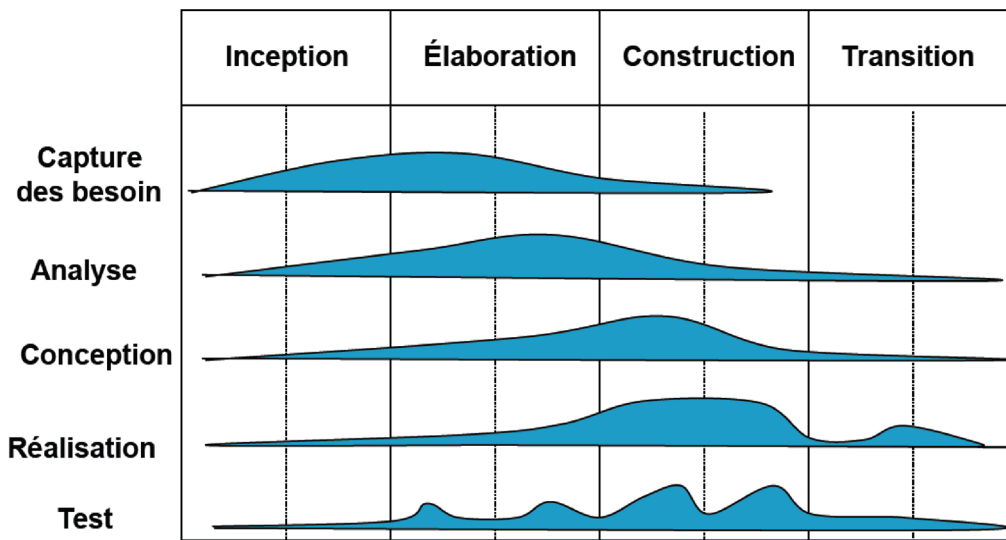


Figure I.9 – Les phases et les activités du processus unifié.

a. Les phases [4]

- La phase d'inception (ou pré-étude) conduit à définir la « vision » du projet, sa portée, sa faisabilité, son business case, afin de pouvoir décider au mieux de sa poursuite ou de son arrêt ;
- La phase d'élaboration poursuit trois objectifs principaux en parallèle :
 - Identifier et décrire la majeure partie des besoins des utilisateurs,
 - Construire (et pas seulement décrire dans un document !) l'architecture de base du système,
 - Lever les risques majeurs du projet.
- La phase de construction consiste surtout à concevoir et implémenter l'ensemble des éléments opérationnels (autres que ceux de l'architecture de base). C'est la phase la plus consommatrice en ressources et en effort ;
- La phase de transition permet de faire passer le système informatique des mains des développeurs à celles des utilisateurs finaux.

b. Les activités [4]

1. **Capture des besoins** : l'expression des besoins comme son nom l'indique, permet de définir les différents besoins :
 - Capturer les besoins principaux et fournir une liste de leurs fonctions ;
 - Recenser les besoins fonctionnels (du point de vue de l'utilisateur) qui conduisent à l'élaboration des modèles de cas d'utilisation.
2. **Analyse** : l'objectif de l'analyse est d'accéder à une compréhension des besoins et des exigences du client.
3. **Conception** : décrit les différentes vues (fonctionnelles, dynamiques et statiques) d'une architecture.
4. **Implémentation** : c'est le résultat de la conception pour implémenter le système sous formes de composants. On planifie les intégrations des composants pour chaque itération, on produit les classes et les sous-systèmes sous formes de codes sources.
5. **Test** : les tests permettent de vérifier les résultats de l'implémentation en testant la construction. Pour mener à bien ces tests, il faut les planifier pour chaque itération, les implémenter, effectuer ces tests et prendre en compte le résultat de chacune.

I.8 Maquette

Une maquette est une représentation partielle ou complète d'un système ou d'un objet afin d'en tester et valider certains aspects et/ou le comportement, ou simplement à des fins ludiques ou informatives [5].

En informatique, le maquettage est une méthode de conception d'interface qui nous permet de proposer aux clients des interfaces conformes à leurs attentes et besoins. Elle permet également de s'assurer que les besoins du client sont adaptés ou non au projet [5].

Pour réaliser nos maquettes, nous avons utilisé un logiciel de création de maquette qui est « Balsamiq Mockups ».

Dans ce qui suit, nous allons présenter les différentes interfaces de notre application à développer.

I.8.1 Les principales interfaces (maquettes)

- La maquette de la page d'accueil

La Figure I.10 ci-dessous représente la maquette de la page d'accueil.

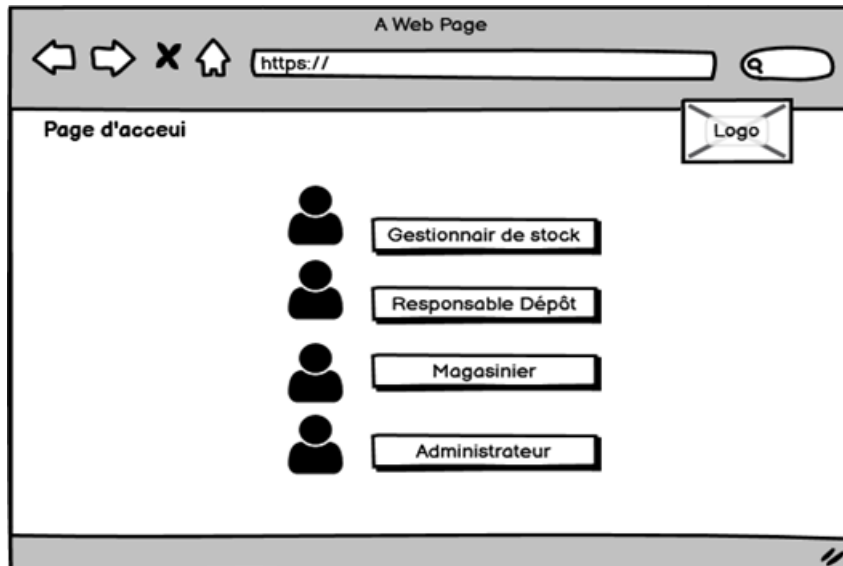


Figure I.10 – Maquette ‘ Page d’accueil’.

- La maquette de la page d'authentification

À partir de cette interface, l'utilisateur remplit les deux champs (email et le password). Ensuite, il clique sur le bouton connexion afin qu'il accède à son espace (Figure I.11).

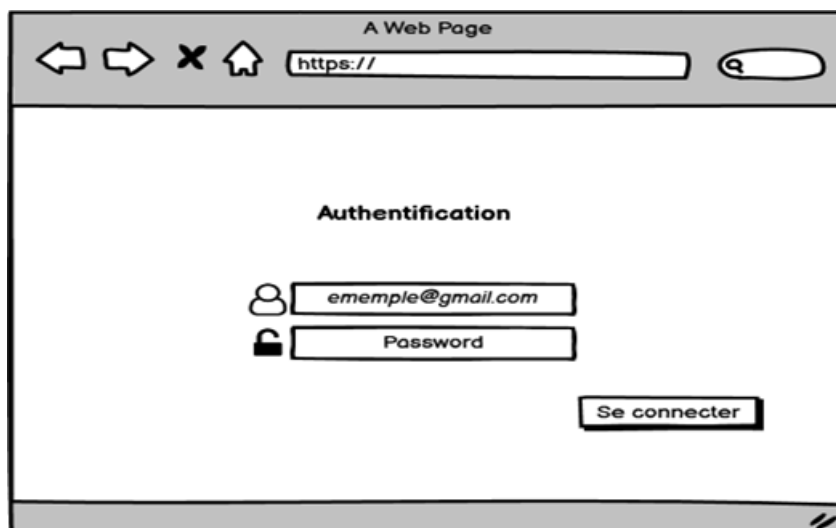


Figure I.11 – Maquette ‘page d’authentification’.

- La maquette de l'espace de gestionnaire de stock

Après l'authentification du gestionnaire de stock, il accède à son espace décrit dans la maquette suivante, dont il pourra gérer les bobines, gérer les types papiers, créer un bon de réception, créer un cde-à-barre pour la bobine (Figure I.12).

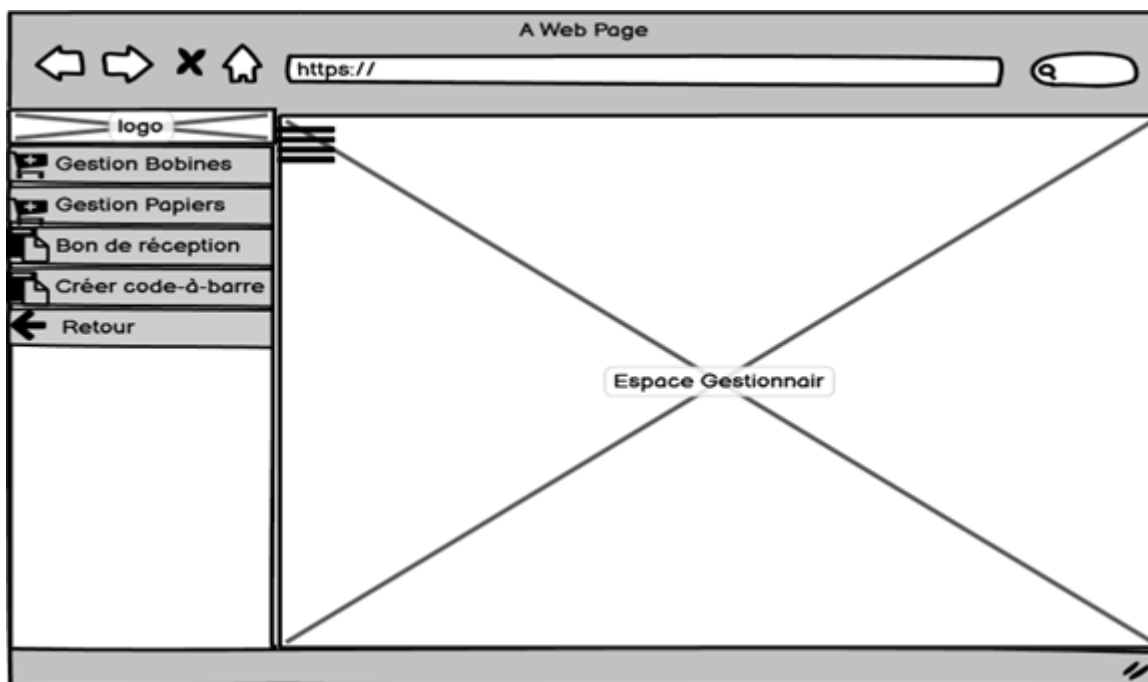


Figure I.12 – Maquette 'Espace Gestionnaire de stock'.

- La maquette de Gestion Bobines

La Figure I.13 représente la maquette qui permet au gestionnaire de gérer les bobines papiers.

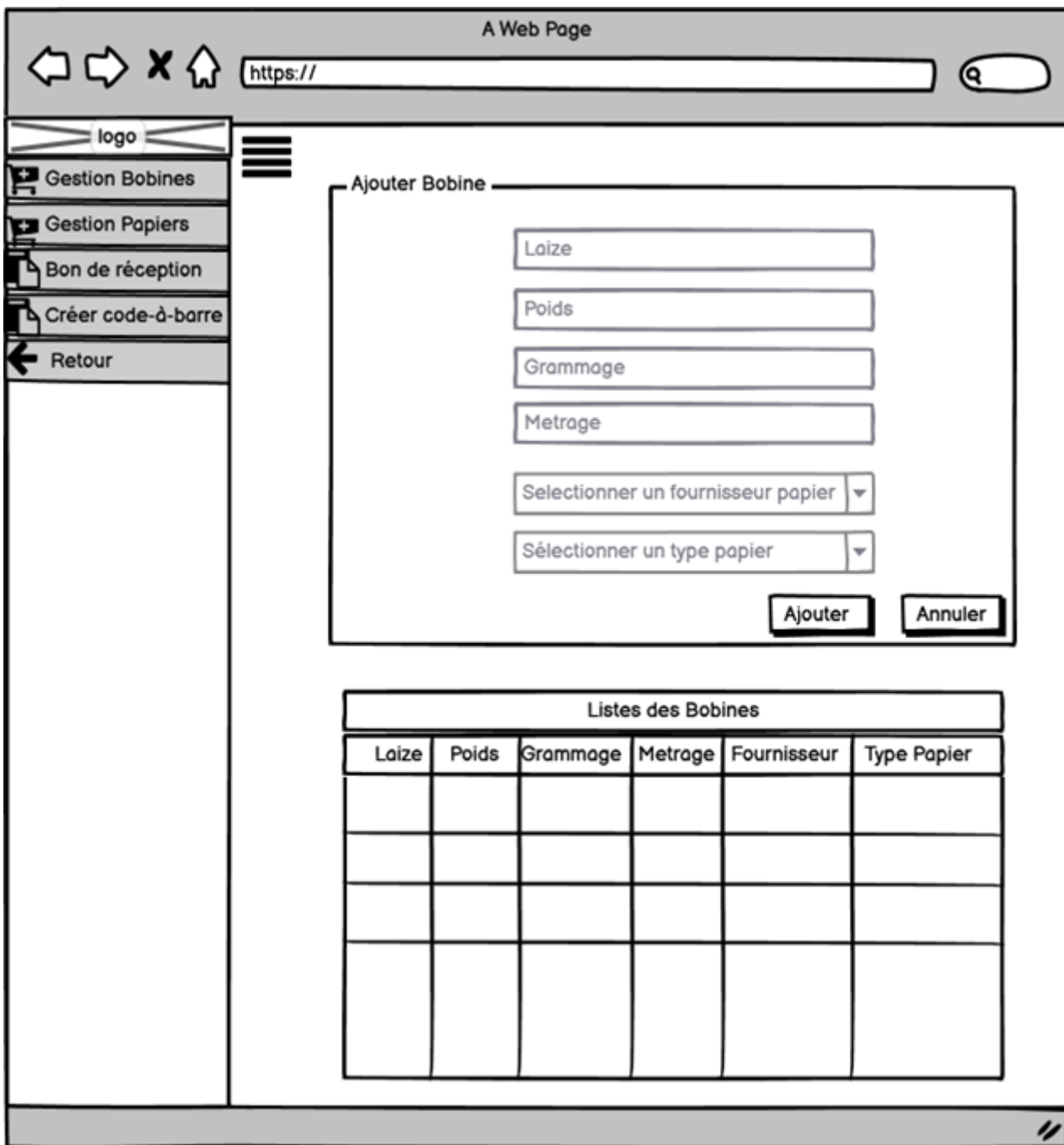


Figure I.13 – Maquette ‘Gestion Bobines’.

- La maquette de création de bon

La Figure I.14 représente la maquette de créations de bon de réceptions des bobines.

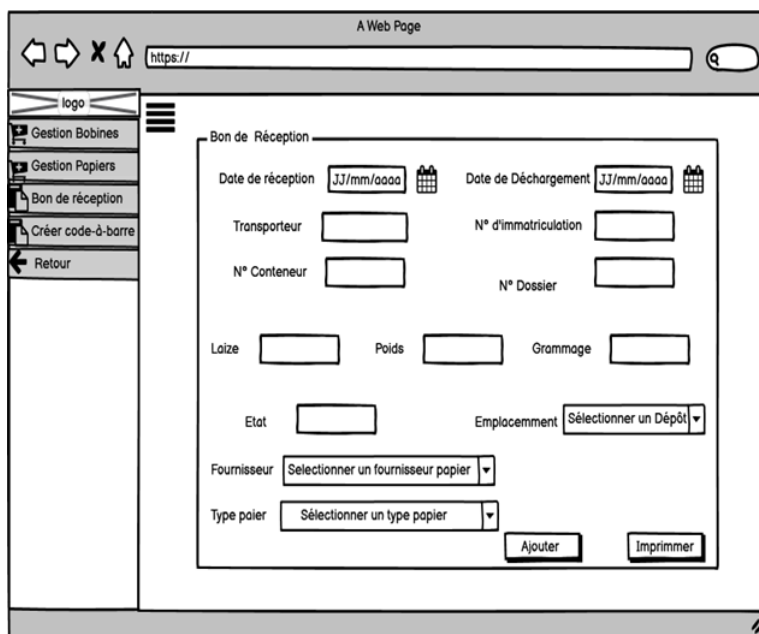


Figure I.14 – Maquette ‘Bon de réception’.

- La maquette de création de code-à-barre

La Figure I.15 représente la maquette de création de code-à-barre bobine.

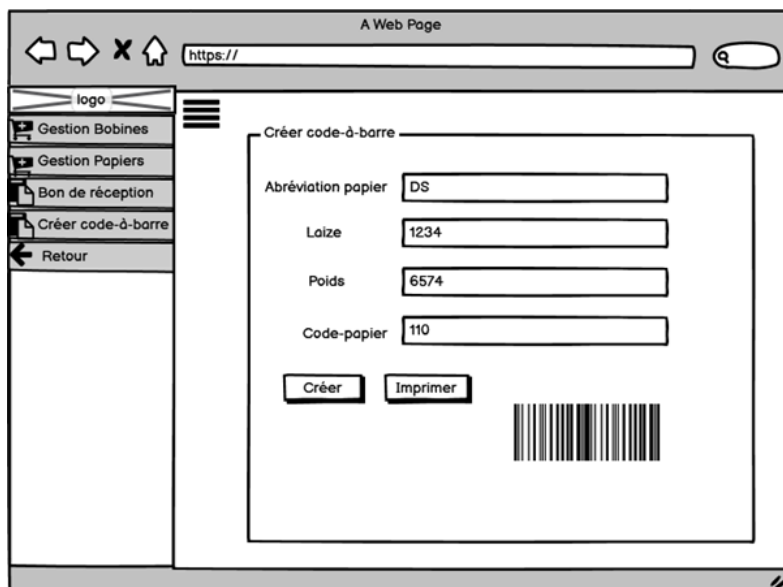


Figure I.15 – Maquette ‘Créer code-à-barre’.

- **La maquette de l'espace de Magasinier**

Après l'authentification du Magasinier, il accède à son espace décrit dans la maquette suivante, dont il pourra gérer les entrées, gérer les sortis, créer des bon entrée/sortis (Figure I.16).

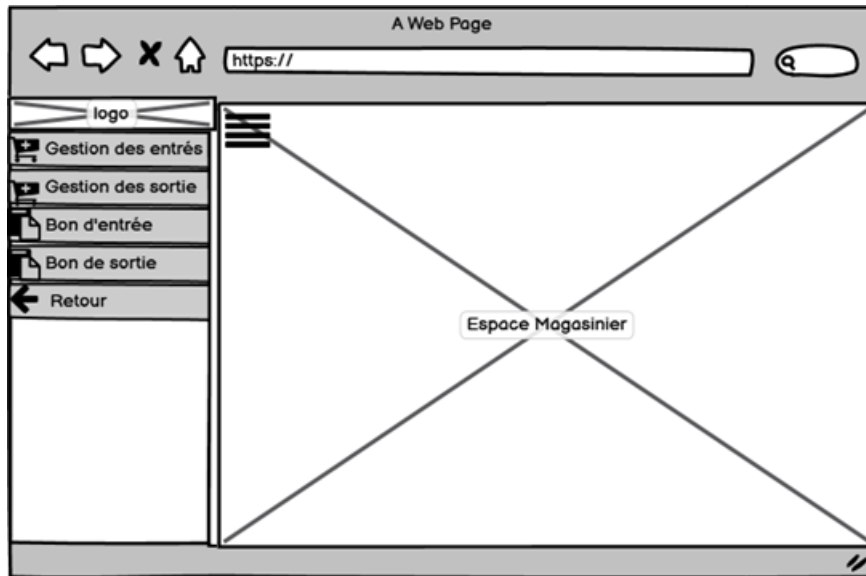


Figure I.16 – Maquette 'Espace Magasinier'.

- **La maquette de l'espace Administrateur**

Après l'authentification du l'admin, il accède à son espace décrit dans la maquette suivante, dont il pourra gérer les utilisateurs à savoir le gestionnaire, le Magasinier, et le responsable dépôt (Figure I.17).

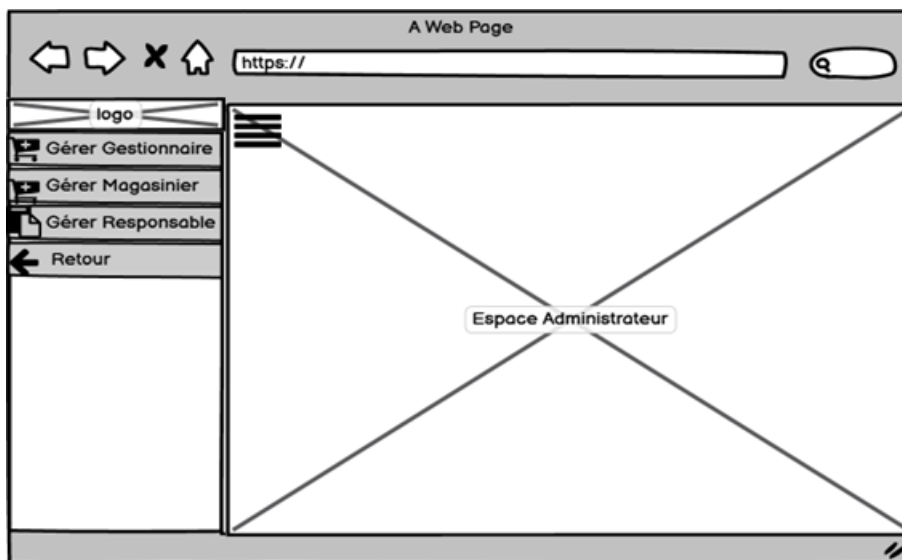


Figure I.17 – Maquette 'Espace Administrateur'.

- **La maquette de l'espace de responsable de dépôt**

Après l'authentification du responsable de dépôt, il accède à son espace décrit dans la maquette suivante, dont il pourra réintégrer les bobines, créer un bon de commande papier (Figure I.18).

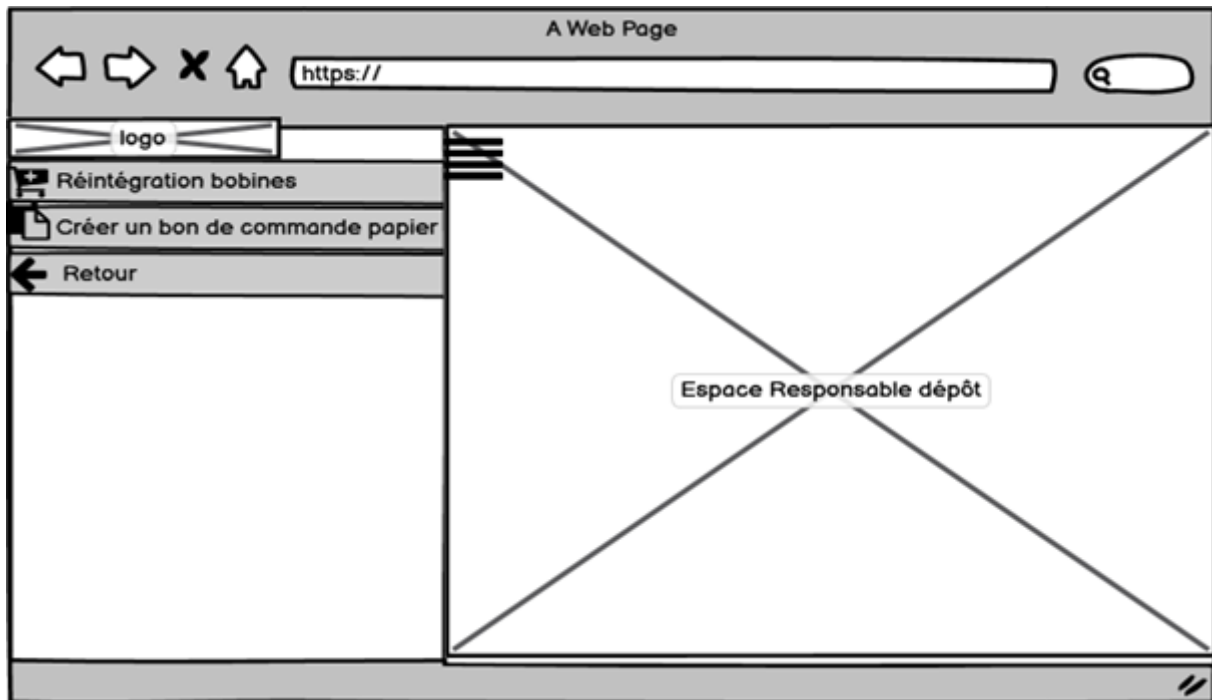


Figure I.18 – Maquette ‘Espace Responsable dépôt’.

I.9 conclusion

Au cours de ce chapitre, nous avons présenté d'une manière générale l'entreprise Général Emballage ainsi que son organisation. Nous avons également présenté quelques généralités sur le stock et les codes-barres. Ensuite, nous avons défini le problème auquel est confrontée cette entreprise ainsi que proposé une solution pour améliorer sa productivité. Après, nous avons choisi et justifié notre processus de développement qui est UP. Enfin, nous avons complété ce chapitre avec des maquettes que nous avons réalisées avec Balsamiq Mockups. Dans le chapitre suivant, nous allons analyser et spécifier les besoins utilisateurs.

Analyse et spécification des besoins

II.1 Introduction

Dans ce chapitre, nous nous focalisons sur les fonctionnalités offertes par notre application, cela en recensant les différents besoins fonctionnels et non fonctionnels qu'elle doit satisfaire, les motivations et objectifs du développement de l'application, les acteurs qui interagissent avec le système, et en présentant à l'aide du formalisme UML (Unified Modeling Language) les diagrammes de modélisation tout en expliquant leurs rôles. Nous allons principalement présenter le diagramme de contexte dynamique, le diagramme des cas d'utilisation, les diagrammes de séquences et le modèle de domaine. Enfin, nous terminons ce chapitre par une conclusion.

II.2 Spécification des besoins

La spécification des besoins constitue la phase de départ de toute application à développer. Elle consiste à comprendre le contexte du système. Il s'agit de déterminer les fonctionnalités, les acteurs les plus pertinents, et d'identifier les cas d'utilisation. Pour concevoir le système, il nous était indispensable de collecter les informations nécessaires auprès de l'entreprise Générale Emballage (Akbou).

II.2.1 Besoins fonctionnels

Notre application doit répondre aux exigences fonctionnelles suivantes :

1. **Côté Administrateur (Admin) :**

- Gérer les utilisateurs. (Gestionnaire Magasinier, Responsable dépôt)

2. **Côté Gestionnaire de stock (Gest) :**

- Gérer les bobines (Ajouter, modifier, rechercher) ;
- Créer un code à barres pour chaque bobine ;
- Gérer les types de papiers (Ajouter, modifier, rechercher) ;
- Etablir le bon de réception des bobines.

3. **Côté Responsable dépôt (RespD) :**

- Etablir un bon de commande bobine ;
- Réintégrer des bobines : les bobines qui ne sont pas complètement utiliser lors de la production ;

4. **Côté Magasinier (Mag) :**

- Gérer les entrées/sorties ;
- Consulter le stock bobines ;
- Etablir les bons (bon de sortie/bon d'entrée).

II.2.2 Besoins non fonctionnels

Afin d'assurer un bon fonctionnement de l'application et de garantir la satisfaction de l'utilisateur, des contraintes doivent être prises en compte tout au long du développement du projet. Pour notre application, nous nous sommes basés sur les points suivants :

1. **La performance** : un logiciel doit être avant tout performant, c'est à -dire à travers ses fonctionnalités, réponds à toutes les exigences de l'entreprise d'une manière optimale ;
2. **La convivialité** : l'application doit fournir des interfaces simples et élégantes pour l'utilisateur afin de faciliter l'exploitation des services de l'application ;
3. **Robustesse et Sureté** : (dysfonctionnements ou ne plante pas) : le logiciel fonctionne raisonnablement en toutes circonstances, rien de catastrophique ne peut survenir, même en dehors des conditions d'utilisation prévues ;
4. **Fiabilité** : (correction, justesse et conformité) : le logiciel est conforme à ses spéciations, les résultats sont ceux attendus ;
5. **Sécurité** : c'est la sûreté (assurance) et la garantie offerte par un logiciel, ou l'absence du danger lors de l'exploitation du logiciel ;

- 6. **La portabilité** : Un même logiciel doit pouvoir fonctionner sur plusieurs machines ainsi le rendre indépendant de son environnement d'exécution ;
- 7. **Traçabilité** : c'est la possibilité de suivre un produit aux différents stades de sa production, de sa transformation et de sa commercialisation [7].

II.3 Identification des acteurs

Un acteur représente un rôle joué par une entité externe (utilisateur humain, dispositif matériel ou autre système) qui interagit directement avec le système étudié. Un acteur peut consulter et /ou modifier directement l'état du système, en émettant et/ou en recevant des messages susceptibles d'être porteurs de données [8].

Dans notre système, nous avons identifié les trois acteurs principaux :

- 1. L'administrateur système : Admin ;
- 2. Le responsable dépôts : RespD ;
- 3. Le gestionnaire : Gest ;
- 4. Magasinier : Mag.

II.4 Diagramme de contexte

Le diagramme de contexte dynamique délimite le domaine d'étude en précisant :

- Ce qui est à la charge du system,
- En identifiant l'environnement extérieur au système étudié avec lequel ce dernier communique [15].

La figure II.1 représente le diagramme de contexte de notre système :

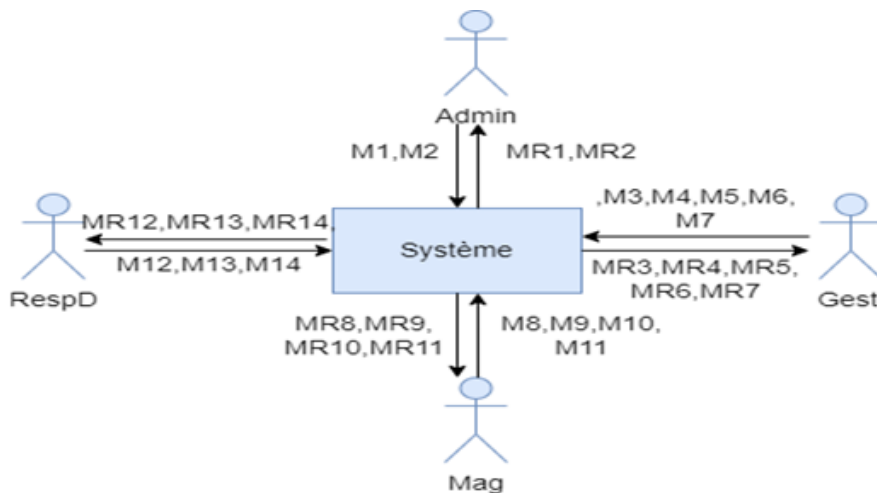


Figure II.1 – Diagramme de contexte de notre système.

Le tableau II.1 résume les différents les messages échangés entre les acteurs et le système :

| Acteur | Messages entrants | Messages sortants |
|----------------------|---|---|
| Admin | <p>M1 : Demande d'authentification.</p> <p>M2 : Gérer les utilisateurs.</p> | <p>MR1 : Affiche l'interface d'authentification.</p> <p>MR2 : Affiche l'interface pour la gestion des utilisateurs.</p> |
| Gestionnaire | <p>M3 : Demande d'authentification.</p> <p>M4 : Gérer les bobines.</p> <p>M5 : Gérer les types papiers.</p> <p>M6 : Créer Code barre.</p> <p>M7 : Etablir un bon de réception.</p> | <p>MR3 : Affiche l'interface d'authentification.</p> <p>MR4 : Affiche l'interface pour la gestion des bobines.</p> <p>MR5 : Affiche l'interface pour la gestion des types papiers.</p> <p>MR6 : Affiche l'interface pour la création de code-a-barre.</p> <p>MR7 : Affiche l'interface pour remplir le bon de réception.</p> |
| Magasinier | <p>M8 : Demande d'authentification.</p> <p>M9 : Consulter le stock bobines.</p> <p>M10 : Gérer les entrées/sorties.</p> <p>M11 : Etablir un bon de (sortie/entrée).</p> | <p>MR8 : Affiche l'interface d'authentification.</p> <p>MR9 : Affiche l'état de stock.</p> <p>MR10 : Afficher l'interface de gestion des E/S.</p> <p>MR11 : Affiche l'interface pour remplir un bon de (sortie/entée).</p> |
| Responsable de dépôt | <p>M12 : Demande d'authentification.</p> <p>M13 : Etablir un bon de commande bobine.</p> <p>M14 : Réintégrer les bobines.</p> | <p>MR12 : Affiche l'interface d'authentification.</p> <p>MR13 : Affiche le formulaire pour Remplir le bon de commande bobine.</p> <p>MR14 : Affiche le formulaire de réintégration.</p> |

Table II.1 – Messages échangés entre les acteurs et le système.

II.5 Identification des cas d'utilisations

Un cas d'utilisation représente une unité discrète d'interaction entre un utilisateur et un system. Il est une entité significative de travail [9].

Dans le système que nous allons réaliser, les cas d'utilisation sont illustrés dans le tableau II.2.

| Num | Cas d'utilisation | Acteur |
|-----|-----------------------------------|------------------|
| 1 | Authentification | Tous les acteurs |
| 2 | Gérer les bobines papiers | Gest |
| 3 | Gérer type papier | |
| 4 | Etablir le bon de réception | |
| 5 | Etablir un bon de commande bobine | RespD |
| 6 | Réintégrer les bobines | Mag |
| 7 | Consulter le stock bobines | |
| 8 | Gérer les E/S | |
| 9 | Etablir le bon d'E/S | Gest |
| 10 | Créer un code-à-barre | |

Table II.2 – Identification des cas d'utilisations.

II.6 Diagramme de cas d'utilisation

Les diagrammes de cas d'utilisation sont utiles dans les situations suivantes :

- Avant de commencer un projet, vous pouvez créer des diagrammes de cas d'utilisation pour modéliser une entreprise, afin que tous les participants au projet visualisent bien les travailleurs, clients et activités de l'entreprise.
- Lors du recueil des exigences, vous pouvez créer des diagrammes de cas d'utilisation pour capturer les exigences du système et pour présenter aux autres ce que le système doit faire.
- Lors des phases d'analyse et de conception, vous pouvez utiliser les cas d'utilisation et les acteurs de vos diagrammes de cas d'utilisation pour identifier les classes nécessaires.
- Lors de la phase de test, vous pouvez utiliser les diagrammes de cas d'utilisation pour identifier les tests à réaliser pour le système [10].

Dans la Figure II.2, nous présentons le diagramme des cas d'utilisation. Il s'agit de tout ce que les acteurs peuvent faire comme fonctionnalités sur notre application.

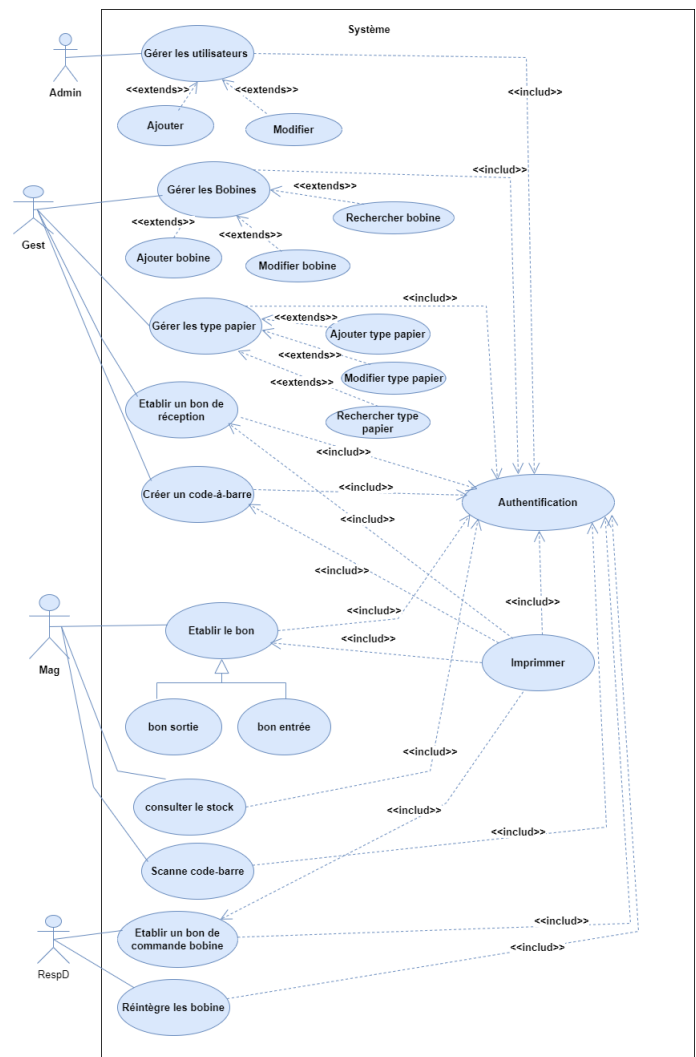


Figure II.2 – Diagramme de cas d'utilisations.

II.7 Descriptions textuelle des cas d'utilisation

Dans cette section, nous allons décrire les différents cas d'utilisation.

- **Le cas d'utilisation « Authentification »**

La description textuelle du cas d'utilisation « Authentications » est donnée par le tableau II.3 ci-dessous :

| Cas d'utilisation N° 1 | S'authentifier |
|------------------------|--|
| Résumé | Ce cas d'utilisation permet à un utilisateur de se connecter au système. |
| Acteurs | Tous les acteurs du système. |
| Précondition | Application accessible. |
| Scénario nominal | <ol style="list-style-type: none"> 1. L'utilisateur accède à la page d'authentification. 2. Le système affiche le formulaire d'authentification. 3. L'utilisateur saisit son email et son mot de passe. 4. Le système vérifie la validation des champs. 5. Le système renvoie l'interface correspondante. |
| Alternative | Dans le cas où les informations fournies sont incorrectes, le système renvoie un message d'erreur, réaffiche le formulaire d'authentification et attend que l'utilisateur ressaisisse ses informations. |
| Post condition | L'utilisateur se connecte au système et peut accéder à son espace. |

Table II.3 – Description textuelle du cas d'utilisation « Authentification ».

• **Le cas d'utilisation « Gérer les bobines »**

La description textuelle du cas d'utilisation « Gérer les bobines » est donnée par le tableau II.4 ci-dessous.

| | |
|-------------------------------|---|
| Cas d'utilisation N° 2 | Gérer les bobines |
| Résumé | Ce cas d'utilisation permet à l'utilisateur de gérer les bobines (matière première). |
| Acteurs | Gestionnaire. |
| Précondition | L'utilisateur doit s'authentifier. |
| Scénario nominal | <ol style="list-style-type: none"> 1. L'utilisateur demande le formulaire de gestion des bobines (ajout, modification). 2. Le système affiche le formulaire de gestion des bobines. <ol style="list-style-type: none"> (a) L'utilisateur saisit les informations relatives aux nouvelles bobines en cas d'ajout puis clique sur Ajouter. (b) Dans le cas de modification, l'utilisateur choisit la bobine à modifier, puis effectue les modifications souhaitées, puis enregistre. |
| Alternative | Le système affiche un message d'erreur si un champ obligatoire est vide ou contient des informations invalides. |
| Post condition | Mise à jour effectuée. |

Table II.4 – Description textuelle du cas d'utilisation « Gérer bobines ».

• **Le cas d'utilisation « Gérer type papier »**

La description textuelle du cas d'utilisation « Gérer type papier » est donnée par le tableau II.5 ci-dessous :

| | |
|-------------------------------|--|
| Cas d'utilisation N° 3 | Gérer les types de papier |
| Résumé | Ce cas d'utilisation permet à l'utilisateur de gérer les types de papier. |
| Acteurs | Gestionnaire. |
| Précondition | L'utilisateur doit s'authentifier. |
| Scénario nominal | <ol style="list-style-type: none"> 1. L'utilisateur demande le formulaire de gestion des types papier (ajout, modification). 2. Le système affiche le formulaire de gestion des papiers. 3. L'utilisateur saisit les informations relatives aux nouvelles bobines en cas d'ajout puis clique sur Ajouter. 4. Dans le cas de modification l'utilisateur choisit la bobine à modifier puis effectue les modifications souhaitées, puis enregistre. |
| Alternative | Le système affiche un message d'erreur si un champ obligatoire est vide ou contient des informations invalides. |
| Post condition | Mise à jour effectuée. |

Table II.5 – Description textuelle du cas d'utilisation « Gérer types papier ».

• **Le cas d'utilisation « Etablir un bon de Réception »**

La description textuelle du cas d'utilisation « Etablir un bon de réception » est donnée par le tableau II.6 ci-dessous :

| | |
|-------------------------------|---|
| Cas d'utilisation N° 4 | Etablir un bon de réception |
| Résumé | Ce cas d'utilisation permet à l'utilisateur d'établir le bon de réception. |
| Acteurs | Gestionnaire. |
| Précondition | L'utilisateur doit s'authentifier. |
| Scénario nominal | <ol style="list-style-type: none"> 1. L'utilisateur demande d'établir un bon de réception. 2. Le système affiche l'interface du bon de réception. 3. L'utilisateur remplit les champs du bon de réception. 4. L'utilisateur valide le bon de réception. 5. Le système affiche un message qui affirme que le bon de réception a été établi. 6. L'utilisateur choisit l'option « imprimer » pour imprimer le bon de réception. 7. Le système affiche la fiche d'état imprimable du bon de réception. |
| Alternative | Si un champ d'information obligatoire est incomplet le système affiche un message d'erreur. |
| Post condition | Le bon de réception est établi. |

Table II.6 – Description textuelle du cas d'utilisation « Etablir un bon de réception ».

• **Le cas d'utilisation « Etablir un bon de commande bobine »**

La description textuelle du cas d'utilisation « Etablir une demande bobines » est donnée par le tableau II.7 ci-dessous :

| | |
|-------------------------------|---|
| Cas d'utilisation N° 5 | Etablir un bon de commande bobine |
| Résumé | Ce cas d'utilisation permet au responsable de dépôt (dépôt des bobines consommable) de faire une commande de bobines pour l'utilisation. |
| Acteurs | RespD. |
| Précondition | L'utilisateur doit s'authentifier. |
| Scénario nominal | <ol style="list-style-type: none"> 1. L'utilisateur demande d'accéder à l'interface de bon de commande. 2. Le système affiche l'interface sous forme d'un formulaire. 3. L'utilisateur remplit les informations et valide. 4. Le système confirme la création du bon. 5. L'utilisateur choisit l'option « imprimer » pour imprimer le bon. |
| Alternative | Le système affiche un message d'erreur si un champ obligatoire est vide ou contient des informations invalides. |
| Post condition | Le bon est créée et imprimer. |

Table II.7 – Description textuelle du cas d'utilisation « Demande bobine ».

• Le cas d'utilisation « Réintégrer des bobines »

La description textuelle du cas d'utilisation « réintégrer bobines » est donnée par le tableau II.8.

| | |
|-------------------------------|---|
| Cas d'utilisation N° 6 | Réintégrer les bobines |
| Résumé | Ce cas d'utilisation permet de réintégrer les bobines qui ne sont pas complètement utilisées. |
| Acteurs | RespD. |
| Précondition | Le RespD doit s'authentifier. |
| Scénario nominal | <ol style="list-style-type: none"> 1. L'utilisateur demande d'accéder à l'interface de « Réintégration des bobines ». 2. Le système affiche l'interface de réintégration. 3. L'utilisateur remplit les nouvelles informations relatives à la bobine et valide. 4. Le système enregistre les informations. |
| Alternative | Le système affiche un message d'erreur si un champ obligatoire est vide ou contient des informations invalides. |
| Post condition | Réintégration effectuée. |

Table II.8 – Description textuelle du cas d'utilisation « Réintégration des bobines ».

• **Le cas d'utilisation « Etablir le bon de sortie »**

La description textuelle du cas d'utilisation « Etablir le bon de sortie » est donnée par le tableau II.9.

| | |
|-------------------------------|--|
| Cas d'utilisation N° 7 | Etablir le bon de sortie |
| Résumé | Ce cas d'utilisation permet d'établir un bon de sortie des bobines |
| Acteurs | Le magasinier. |
| Précondition | Le magasinier doit s'authentifier. |
| Scénario nominal | <ol style="list-style-type: none"> 1. Le magasinier demande d'établir le bon de sortie. 2. Le système affiche l'interface de bon de sortie. 3. L'utilisateur remplit les champs de bon et valide. 4. Le système confirme la création de bon. 5. L'utilisateur choisit l'option « imprimer » pour imprimer le bon. |
| Alternative | Le système affiche un message d'erreur si un champ obligatoire est vide ou contient des informations invalides. |
| Post condition | Le bon de sortie est éablit et imprimer. |

Table II.9 – Description textuelle du cas d'utilisation « Etablir le bon de sortie ».

• **Le cas d'utilisation « Etablir le bon d'entrée »**

La description textuelle du cas d'utilisation « Etablir le bon d'entrée » est donnée par le tableau II.10.

| | |
|-------------------------------|--|
| Cas d'utilisation N° 8 | Etablir le bon d'entrée |
| Résumé | Ce cas d'utilisation permet d'établir un bon d'entrée des bobines. |
| Acteurs | Le magasinier. |
| Précondition | Le magasinier doit s'authentifier. |
| Scénario nominal | <ol style="list-style-type: none"> 1. Le magasinier demande d'établir le bon d'entrée. 2. Le système affiche l'interface de bon d'entrée. 3. L'utilisateur remplit les champs de bon et valide. 4. Le système confirme la création de bon d'entrée. 5. L'utilisateur choisit l'option « imprimer » pour imprimer le bon. 6. Le système affiche le menu d'impression. 7. L'utilisateur saisi les informations pour l'impression et valide. |
| Alternative | Le système affiche un message d'erreur si un champ obligatoire est vide ou contient des informations invalides. |
| Post condition | Le bon d'entrée est établi et imprimer. |

Table II.10 – Description textuelle du cas d'utilisation « Etablir le bon d'entrée ».

- **Le cas d'utilisation « Consulter le stock bobines »**

La description textuelle du cas d'utilisation « consulter le stock bobines » est donnée par le tableau II.11.

| | |
|-------------------------------|---|
| Cas d'utilisation N° 9 | Consulter le stock bobines |
| Résumé | Ce cas d'utilisation permet de consulter les bobines en stock. |
| Acteurs | Le magasinier. |
| Précondition | Le magasinier doit s'authentifier. |
| Scénario nominal | <ol style="list-style-type: none"> 1. L'utilisateur demande l'état de stock. 2. Le système affiche l'état de stock. |
| Post condition | L'utilisateur consulte le stock. |

Table II.11 – Description textuelle du cas d'utilisation « Consulter le stock bobines ».

• Le cas d'utilisation « Gérer les E/S »

La description textuelle du cas d'utilisation « Gérer les E/S » est donnée par le tableau II.12.

| | |
|--------------------------------|---|
| Cas d'utilisation N° 10 | Gérer les E/S |
| Résumé | Ce cas d'utilisation permet au magasinier de gérer les entrées et les sorties des bobines. |
| Acteurs | Le magasinier. |
| Précondition | Le magasinier doit s'authentifier. |
| Scénario nominal | <ol style="list-style-type: none"> 1. Le magasinier demande l'interface de gestion des entrées/ sorties. 2. Le système affiche l'interface. <ol style="list-style-type: none"> (a) Dans le cas où il s'agit d'une sortie, le magasinier scanne le code-barres de la bobine et l'enregistre. (b) Dans le cas où il s'agit d'une entrées, le magasinier scanne le code-barres de la bobine et l'enregistre. 3. le système enregistre la bobine soit comme une entrée où une sortie. |
| Alternative | Si le code-barres est mal scanné, le système affiche un message d'erreur. |
| Post condition | Le mouvement est effectué. |

Table II.12 – Description textuelle du cas d'utilisation « Gérer E/S ».

• **Le cas d'utilisation « Créer un code-à-barre »**

La description textuelle du cas d'utilisation «Créer un code-à-barre » est donnée par le tableau II.13.

| | |
|--------------------------------|---|
| Cas d'utilisation N° 11 | Créer un code-à-barre |
| Résumé | Ce cas d'utilisation permet au gestionnaire e créer un code-à-barre pour chaque bobine. |
| Acteurs | Le gestionnaire. |
| Précondition | Le gestionnaire doit s'authentifier. |
| Scénario nominal | <ol style="list-style-type: none"> 1. Le gestionnaire demande de créer un code-à-barre ; 2. Le système affiche l'interface de création de code-à-barre ; 3. Le gestionnaire remplit les champs puis clique sur créer code-à-barre ; 4. Le système vérifier les champs et renvoie le code-à-barre ; 5. L'utilisateur choisit l'option « imprimer » pour imprimer le code-à-barre ; 6. Le système affiche la fiche d'état imprimable du code-à-barre. |
| Alternative | Le système affiche un message d'erreur si un champ obligatoire est vide ou contient des informations invalides. |
| Post condition | Le Code-à-barre est créer. |

Table II.13 – Description textuelle du cas d'utilisation « Créer un code-à-barre ».

- Le cas d'utilisation « Imprimer »

Le tableau II.14 représente la description du cas d'utilisation « Imprimer » (Imprimer les bon).

| | |
|--------------------------------|---|
| Cas d'utilisation N° 12 | Imprimer |
| Résumé | Ce cas d'utilisation permet aux utilisateurs d'imprimer les bon (bon de réception, bon de commande bobine, bon E/S). |
| Acteurs | Gest, Mag, RespD. |
| Précondition | L'utilisateur doit s'authentifier. |
| Scénario nominal | <ol style="list-style-type: none"> 1. L'utilisateur remplit les champs puis clique sur imprimer. 2. Le système vérifie les champs et affiche l'interface du bon imprimable. |
| Alternative | Le système affiche un message d'erreur si un champ obligatoire est vide ou contient des informations invalides. |
| Post condition | Le bon est imprimé. |

Table II.14 – Description textuelle du cas d'utilisation « Imprimer ».

II.8 Les diagrammes de séquences

Chaque cas d'utilisation peut donner lieu à un (ou plusieurs) diagramme de séquence simple représentant graphiquement la chronologie des interactions entre les acteurs et le système. Il montre également les événements systèmes déclenchés par les acteurs. L'ordre chronologique des messages doit suivre la séquence décrite dans le cas d'utilisation [11].

Un message est une transmission d'informations unidirectionnelle entre deux objets, l'objet émetteur et l'objet récepteur [12].

Dans un diagramme de séquence, deux types de messages peuvent être distingués, des messages synchrones et asynchrones.

- a) Message synchrone : dans ce cas l'émetteur reste en attente de la réponse de son message avant de poursuivre ses actions. La flèche avec une extrémité pleine symbolise ce type de message [13] (voir le message 1 de la figure II.3).
- b) Message asynchrone : dans ce cas l'émetteur n'attend pas de réponse à son message, il poursuit l'exécution de ses opérations, c'est une extrémité non pleine qui symbolise ce type de message [13] (voir le message 3 de la figure II.10).

Dans ce qui suit, nous présenterons les diagrammes de séquences des cas d'utilisation les plus pertinents de notre système.

II.8.1 Diagramme de séquence « Authentification »

Le diagramme de la figure II.3 représente le scénario du cas d'utilisation « Authentification ».

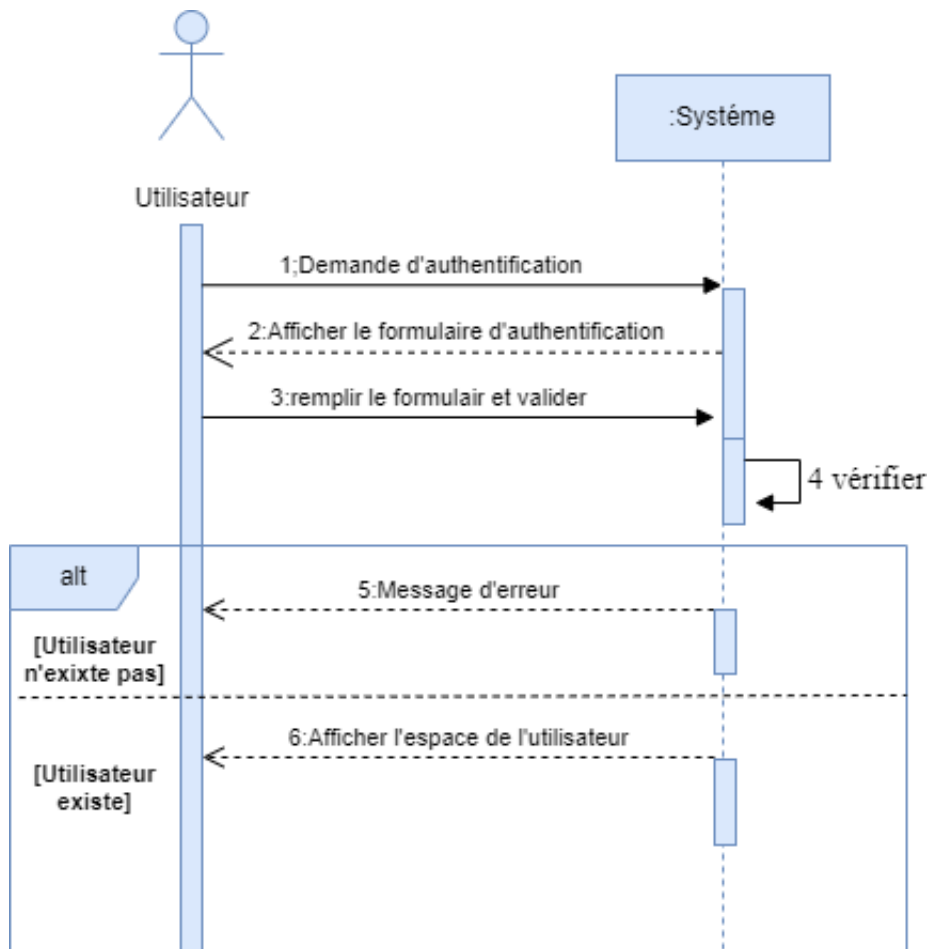


Figure II.3 – Diagramme de séquence système du cas d'utilisation « Authentification ».

II.8.2 Diagramme de séquence « Ajouter Bobine »

Le diagramme de la figure II.4 représente le scénario lorsque l'utilisateur ajoute une bobine.

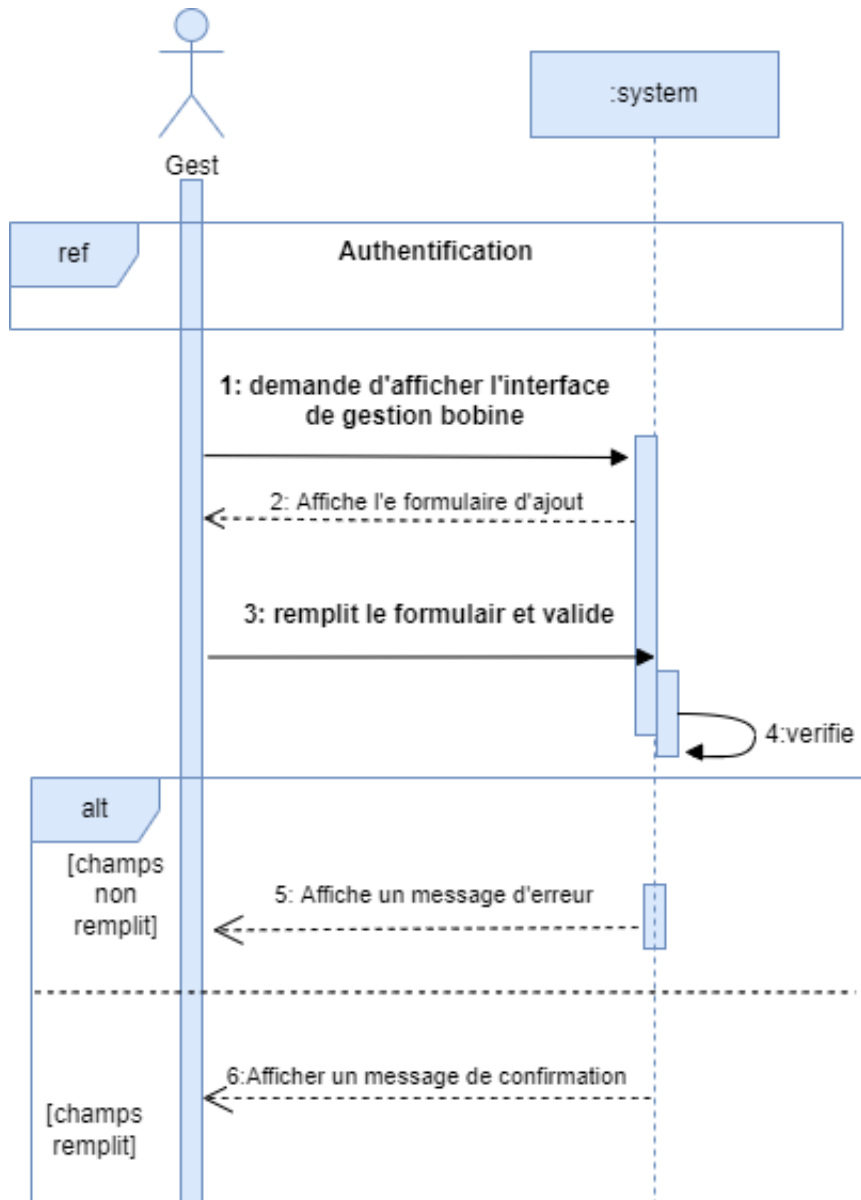


Figure II.4 – Diagramme de séquence système du cas d'utilisation « Ajouter bobine ».

II.8.3 Diagramme de séquence « Modifier bobine »

Le diagramme de la figure II.5 représente le scénario de modification d'une bobine.

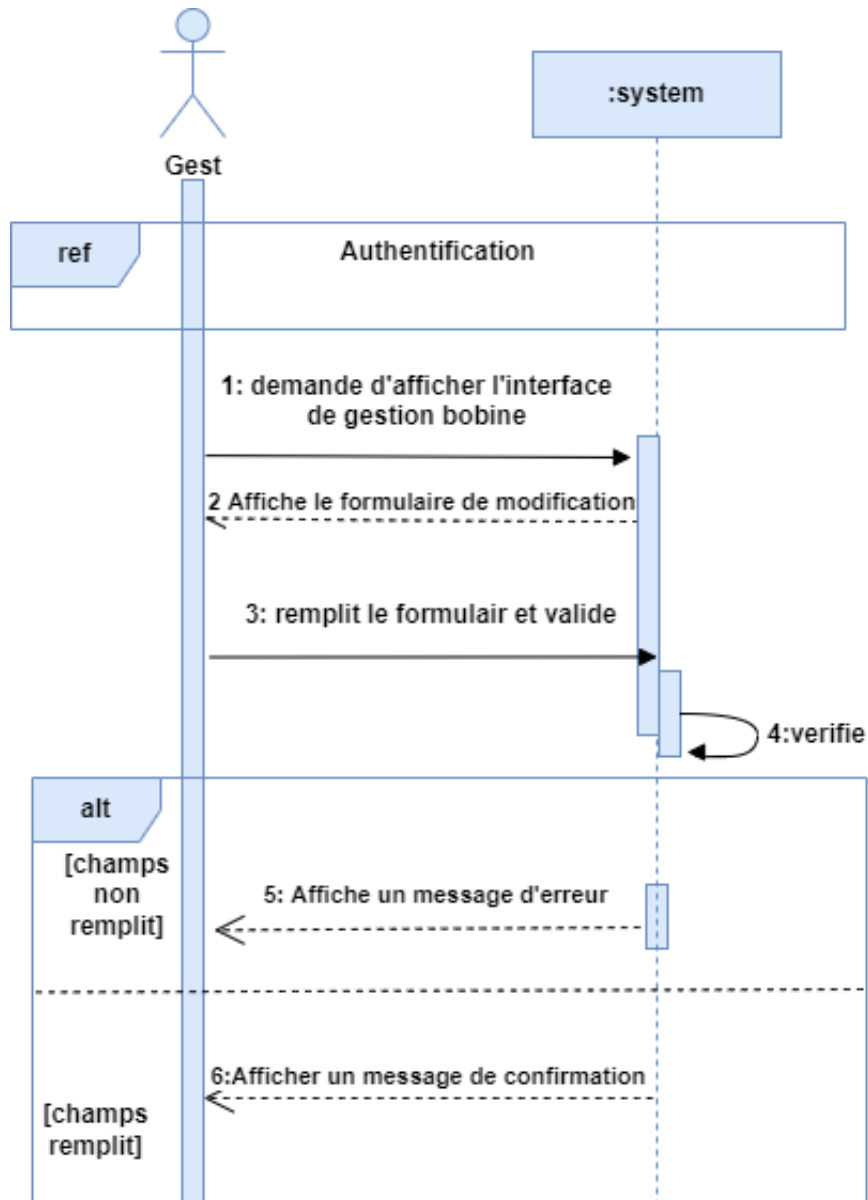


Figure II.5 – Diagramme de séquence système du cas d'utilisation « Modifier bobine ».

II.8.4 Diagramme de séquence « Recherche bobine »

Le diagramme de la figure II.6 représente le scénario de recherche d'une bobine.

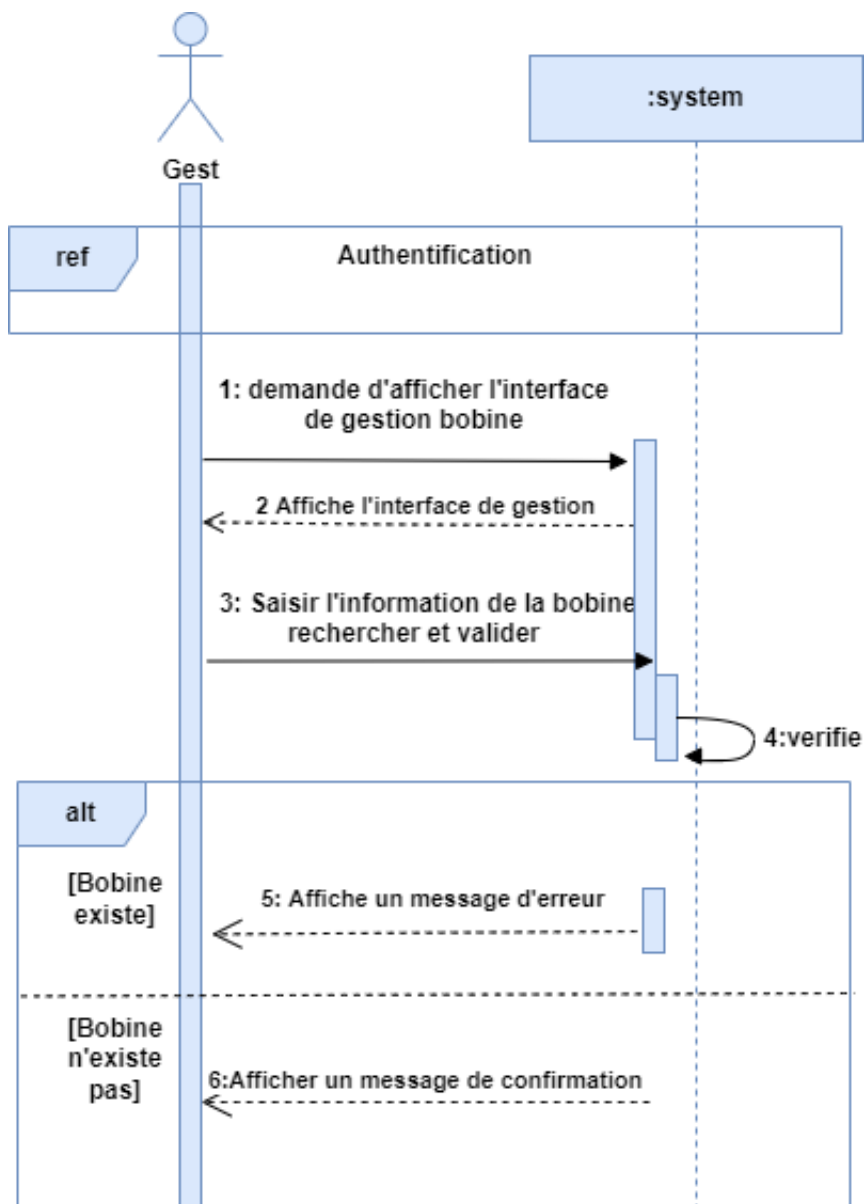


Figure II.6 – Diagramme de séquence système du cas d'utilisation « Recherche bobine ».

II.8.5 Diagramme de séquence « Gérer bobine »

Le diagramme de la figure II.7 représente le scénario de gestion des bobines.

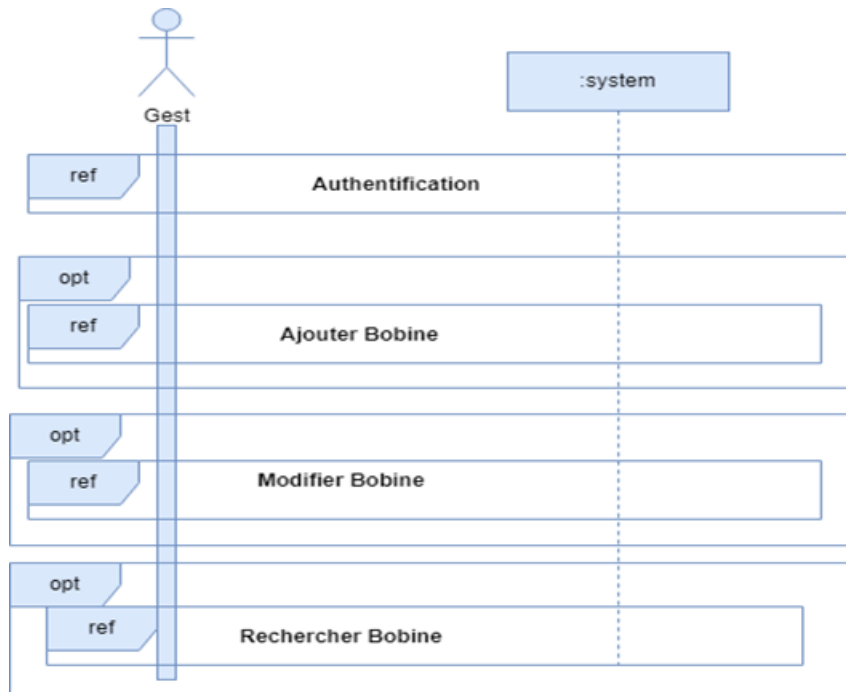


Figure II.7 – Diagramme de séquence système du cas d'utilisation « Gérer les bobines ».

II.8.6 Diagramme de séquence « Etablir un bon de réception »

Le diagramme de la figure II.8 représente le scénario d'établissement du bon de réception des bobines.

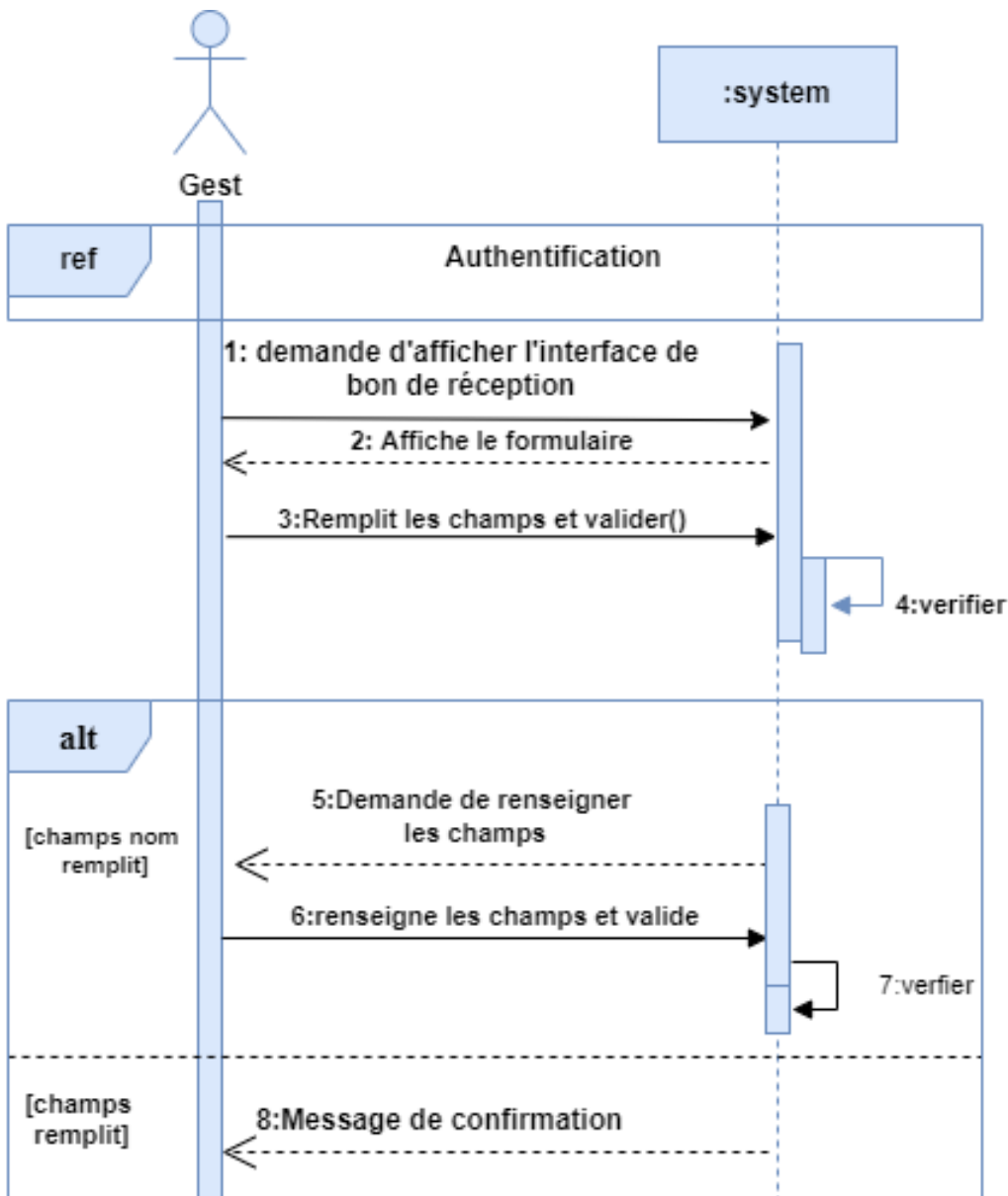


Figure II.8 – Diagramme de séquence système du cas d'utilisation « Etablir bon de réception ».

II.8.7 Diagramme de séquence « Imprimer bon de réception »

Le diagramme de la figure II.9 représente le scénario d'impression du bon de réception bobines.

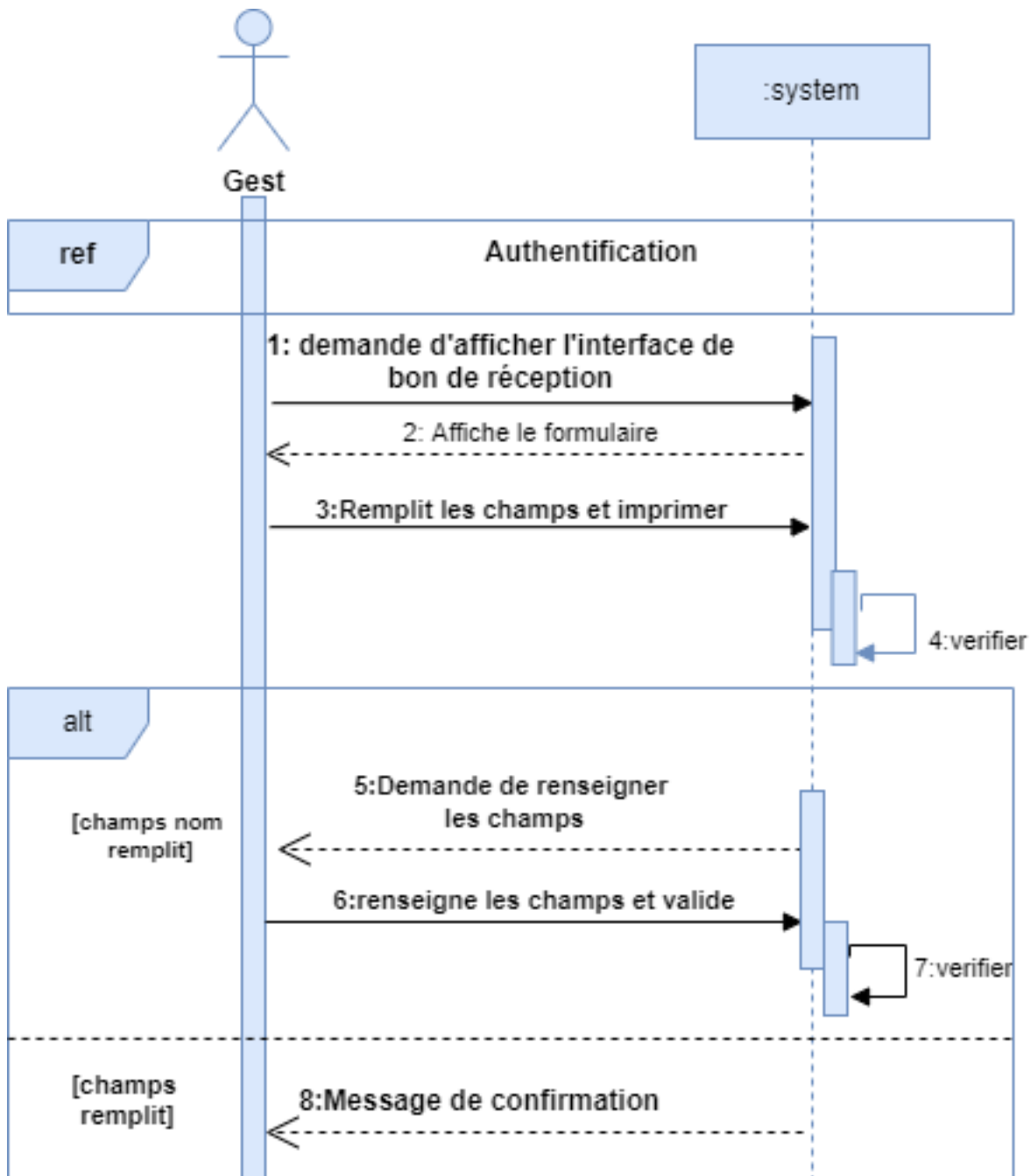


Figure II.9 – Diagramme de séquence système du cas d'utilisation « Imprimer bon de réception ».

II.8.8 Diagramme de séquence « Réintégrer les bobine »

Le diagramme de la figure II.10 représente le scénario réintégration d'une bobine.

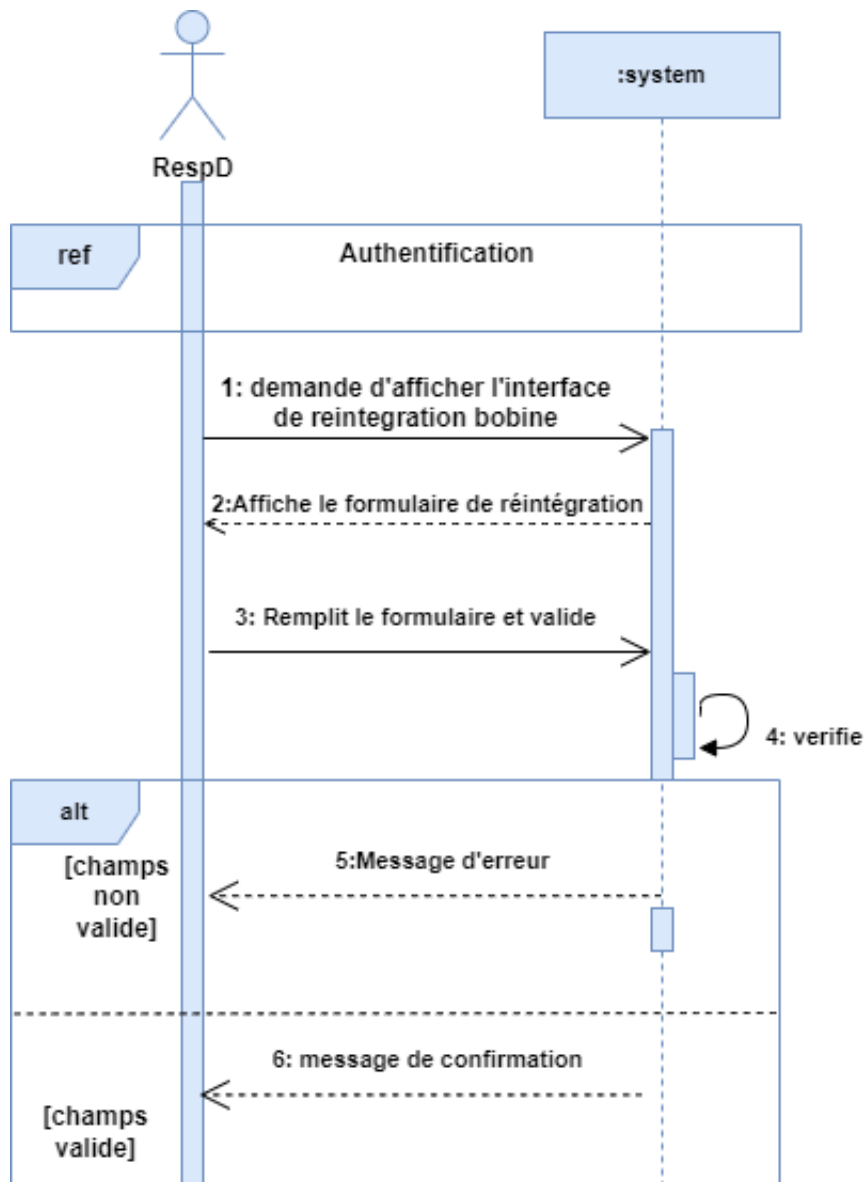


Figure II.10 – Diagramme de séquence système du cas d'utilisation « Réintégrer les bobine ».

II.8.9 Diagramme de séquence « Créer un code-à-barre »

Le diagramme de la figure II.11 représente le scénario de création de code-à-barre.

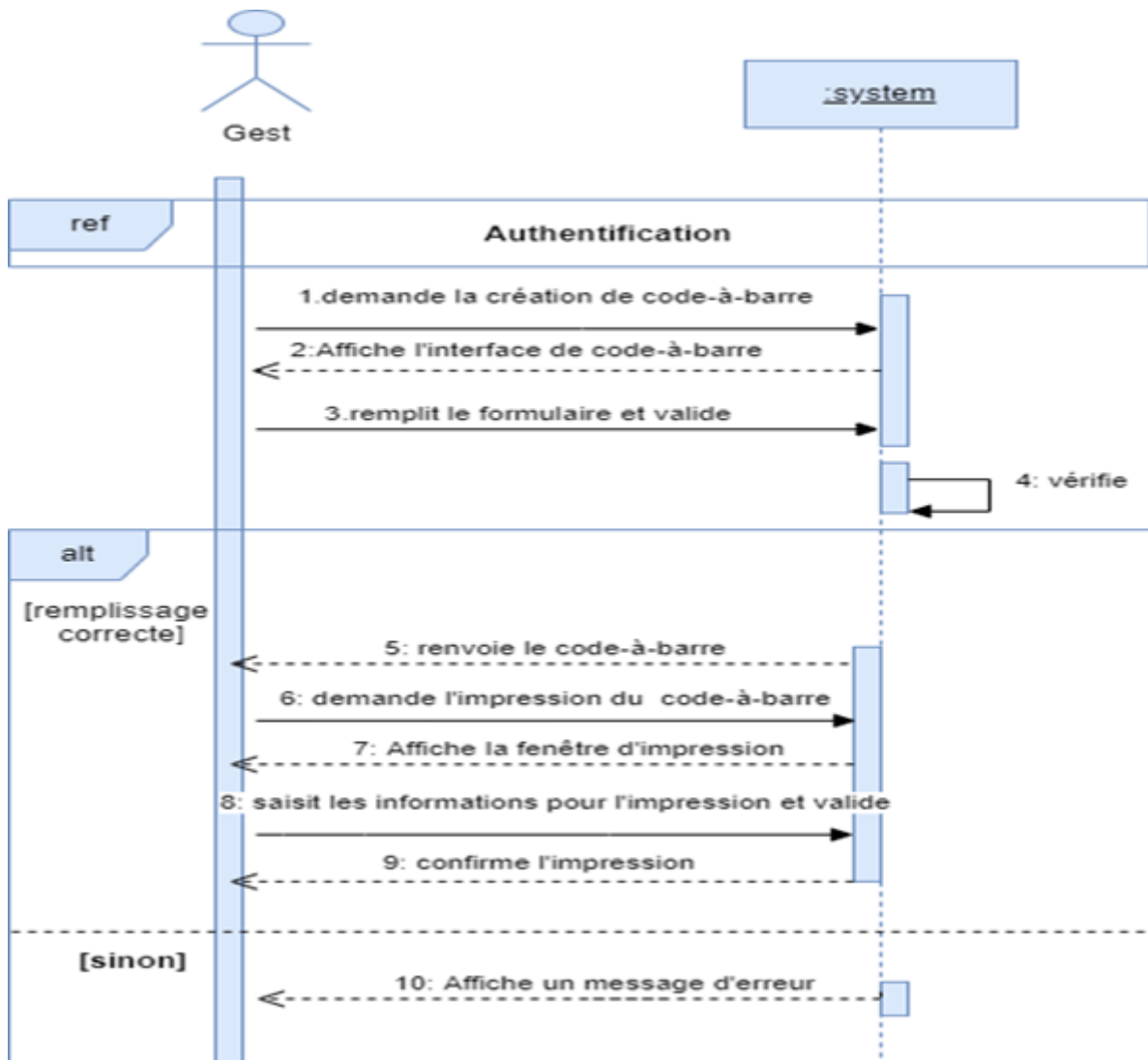


Figure II.11 – Diagramme de séquence système du cas d'utilisation « Créer un code-à-barre ».

II.9 Modèle de domaine

La modélisation de domaine est un moyen de décrire et de modéliser des entités du monde réel et les relations entre elles, qui décrivent collectivement l'espace du domaine du problème. Dérivée d'une compréhension des exigences au niveau du système, l'identification des entités de domaine et de leurs relations fournit une base efficace pour comprendre et aide les praticiens à concevoir des systèmes pour la maintenabilité, la testabilité et le développement incrémentiel [14].

La figure II.12 représente le modèle de domaine dans notre système :

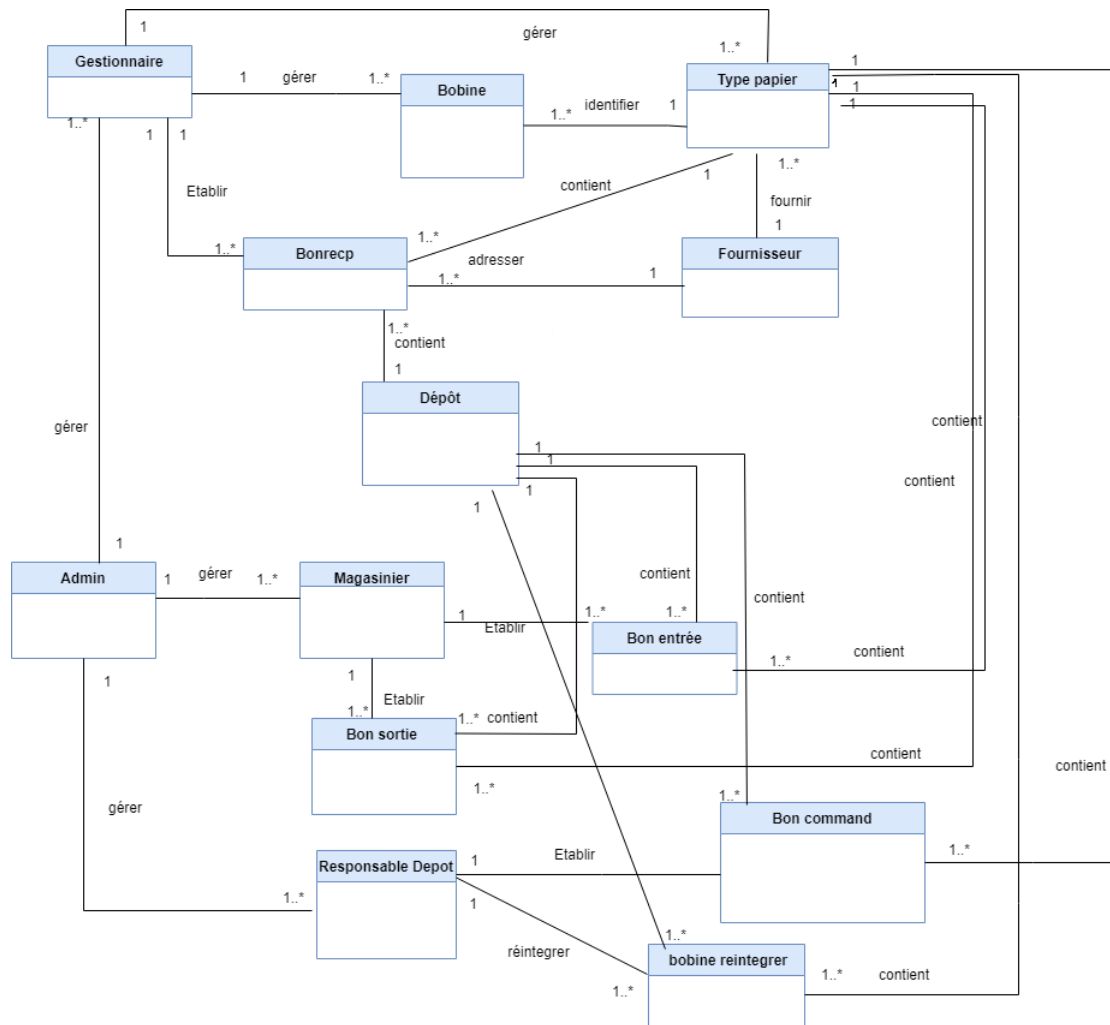


Figure II.12 – Le modèle de domaine.

II.10 Conclusion

Dans ce chapitre, nous avons procédé à l'analyse et la spécification des besoins, ce qui nous a aidé à identifier les principales fonctionnalités. Ce chapitre nous servira de point de départ à l'élaboration de la prochaine phase, qui est la conception.

Conception

III.1 Introduction

Après avoir identifier les différents acteurs et leurs besoins et élaborer les diagrammes de cas d'utilisation, nous nous intéressons dans ce chapitre à l'aspect architectural de l'application. En conséquence, nous allons créer les diagrammes d'interactions qui évoquent les interactions entre ces acteurs et les différents objets. Ensuite, nous allons réaliser le diagramme de classe. Nous allons également appliquer les règles de passage au relationnel ainsi que les règles de gestion de l'application. Enfin, nous terminons ce chapitre par une conclusion.

III.2 Diagramme d'interaction

Un diagramme d'interaction est un type de diagramme UML utilisé pour représenter le comportement interactif d'un système. Les diagrammes d'interaction se concentrent sur la description du flux de messages au sein d'un système, en fournissant du contexte pour une ou plusieurs lignes de vie. Ils peuvent également servir à illustrer des séquences ordonnées et permettre de visualiser des données en temps réel dans les modélisations de systèmes UML [4].

Dans ce qui suit, nous allons donner quelques diagrammes d'interaction des cas d'utilisation les plus importants.

III.2.1 Diagramme d'interaction du cas d'utilisation « Authentification »

Le diagramme d'interaction du cas d'utilisation «Authentification» est illustré par la figure III.1.

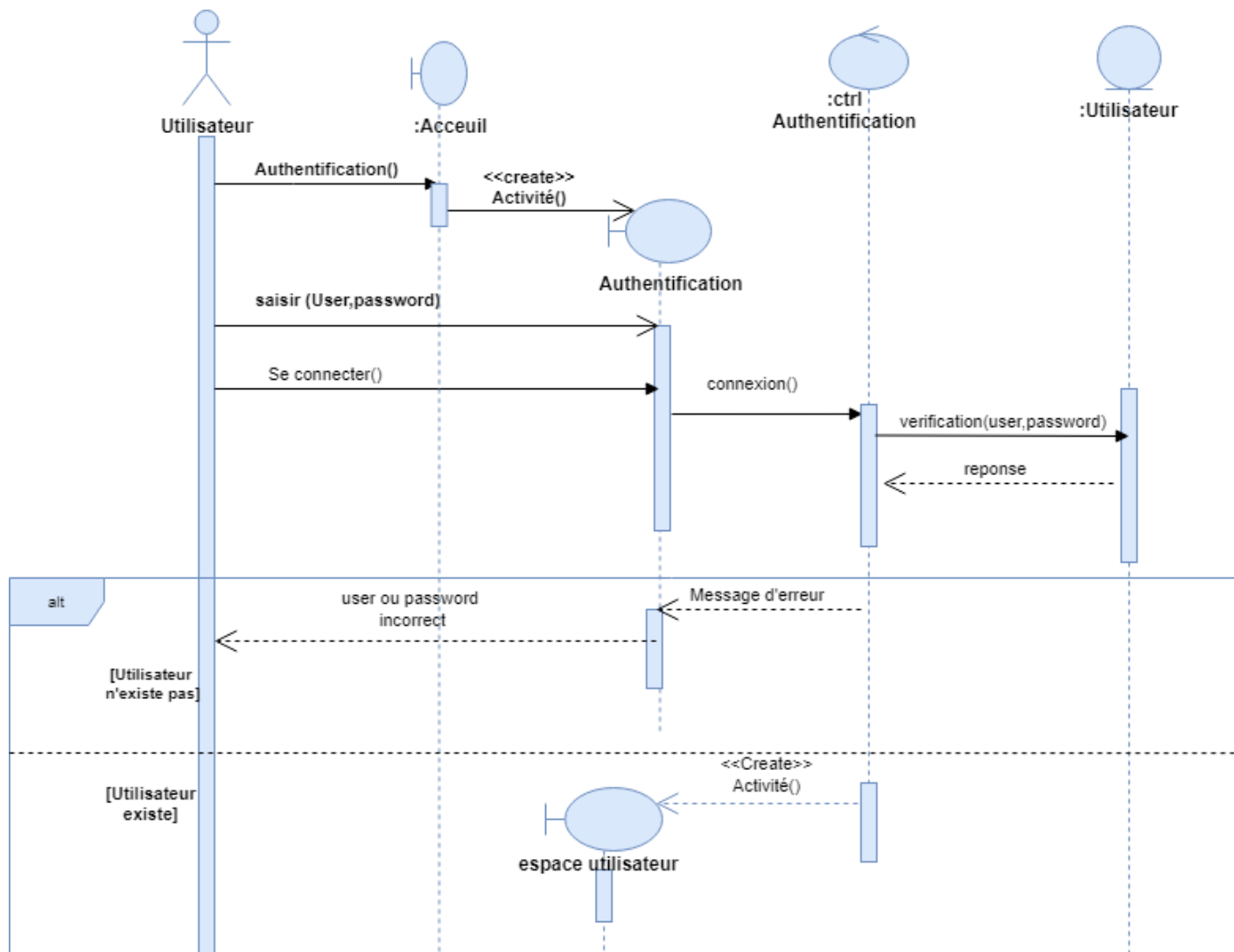


Figure III.1 – Diagramme d'interaction du cas d'utilisation « Authentification ».

III.2.2 Diagramme d'interaction du cas d'utilisation « Ajouter bobine »

Le diagramme d'interaction du cas d'utilisation « Ajouter bobine » est illustré par la figure III.2.

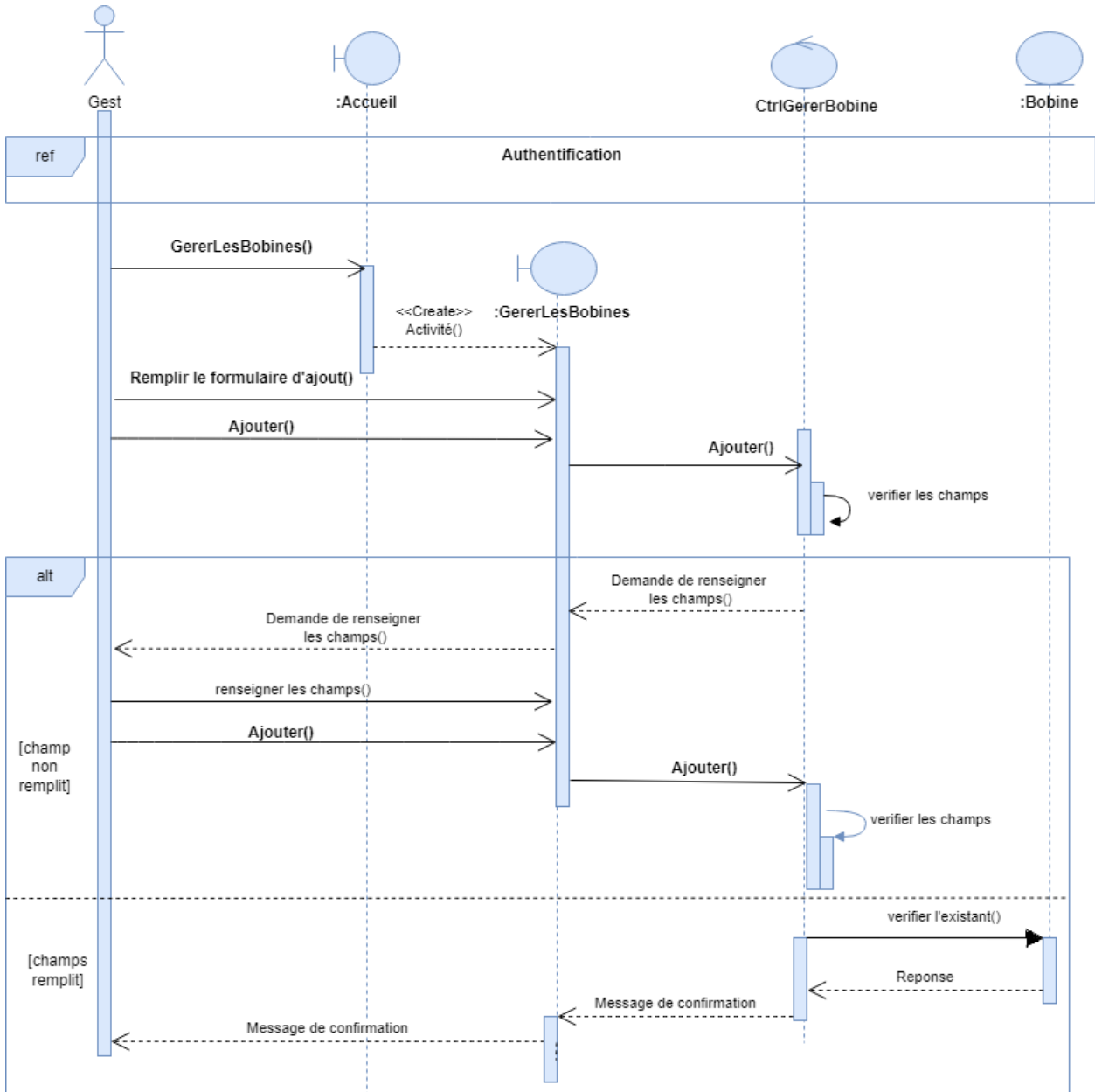


Figure III.2 – Diagramme d'interaction du cas d'utilisation « Ajouter bobine ».

III.2.3 Diagramme d'interaction du cas d'utilisation « Modifier Bobine »

Le diagramme d'interaction du cas d'utilisation «Modifier bobine » est illustré par la figure III.3.

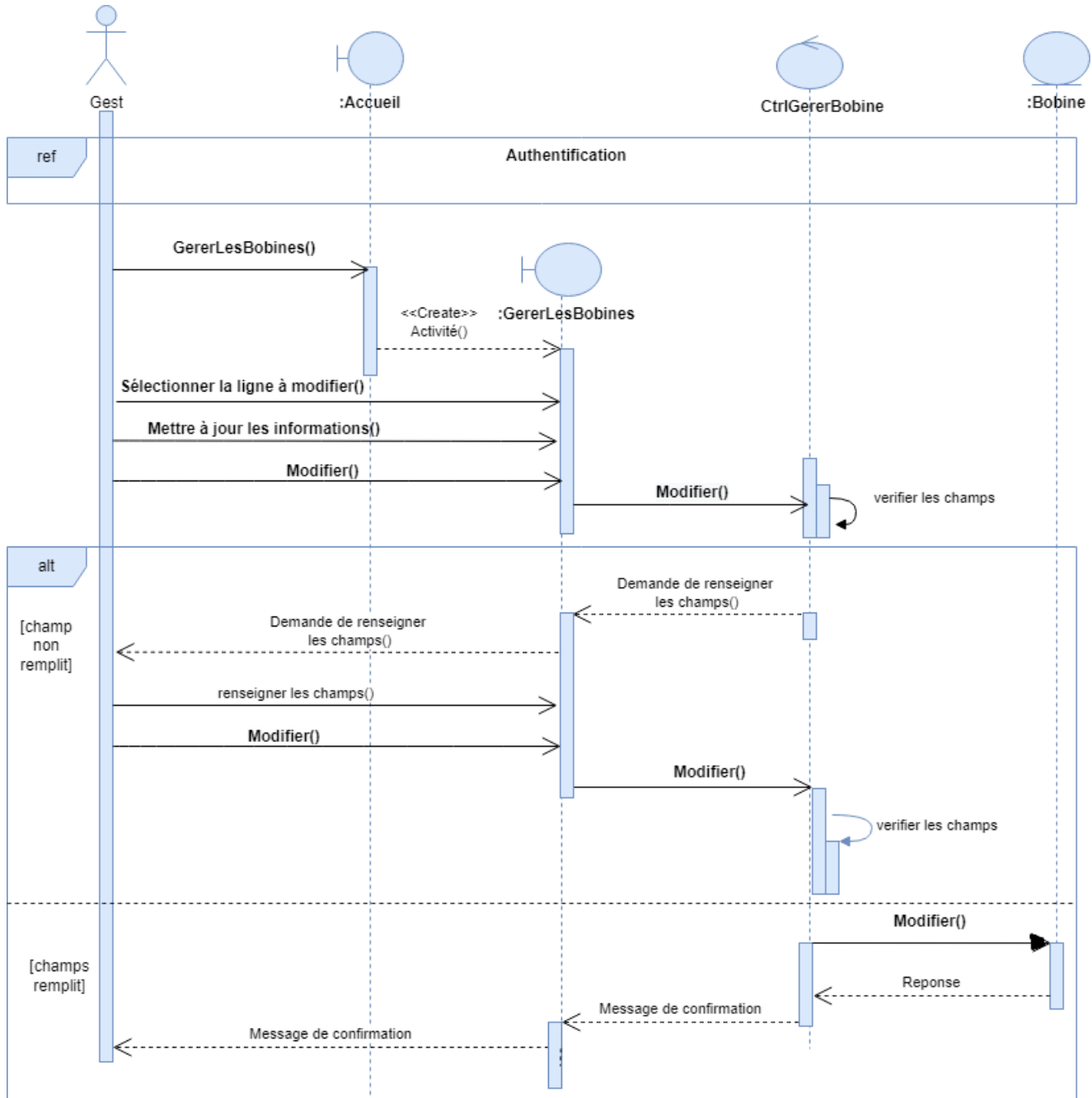


Figure III.3 – Diagramme d'interaction du cas d'utilisation «Modifier bobine ».

III.2.4 Diagramme d'interaction du cas d'utilisation « Rechercher Bobine »

Le diagramme d'interaction du cas d'utilisation «Rechercher bobine » est illustré par la figure III.4.

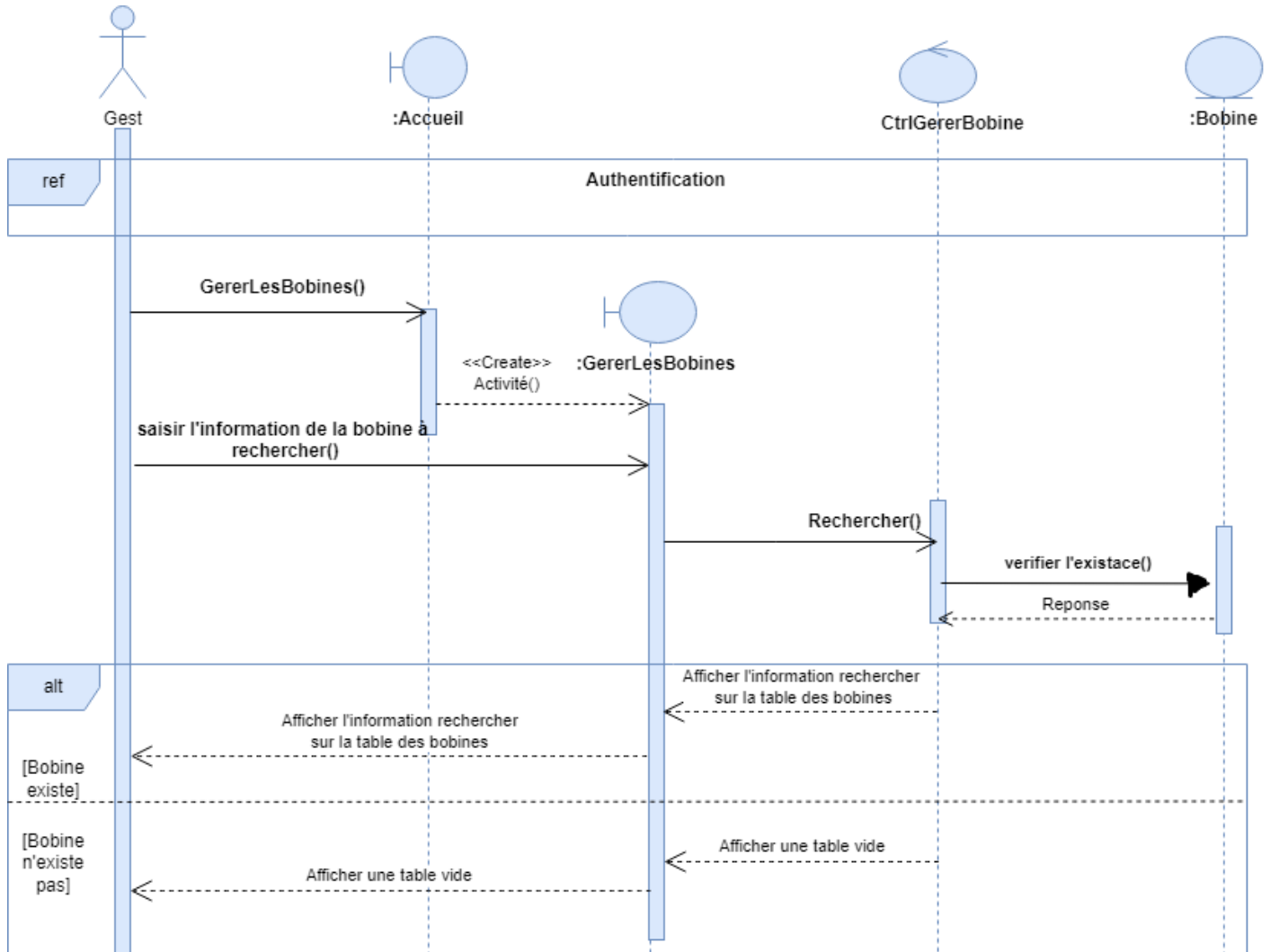


Figure III.4 – Diagramme d'interaction du cas d'utilisation «Rechercher bobine ».

III.2.5 Diagramme d'interaction du cas d'utilisation « Etablir un bon de réception »

Le diagramme d'interaction du cas d'utilisation « Etablir un bon réception » est illustrée par la figure III.5.

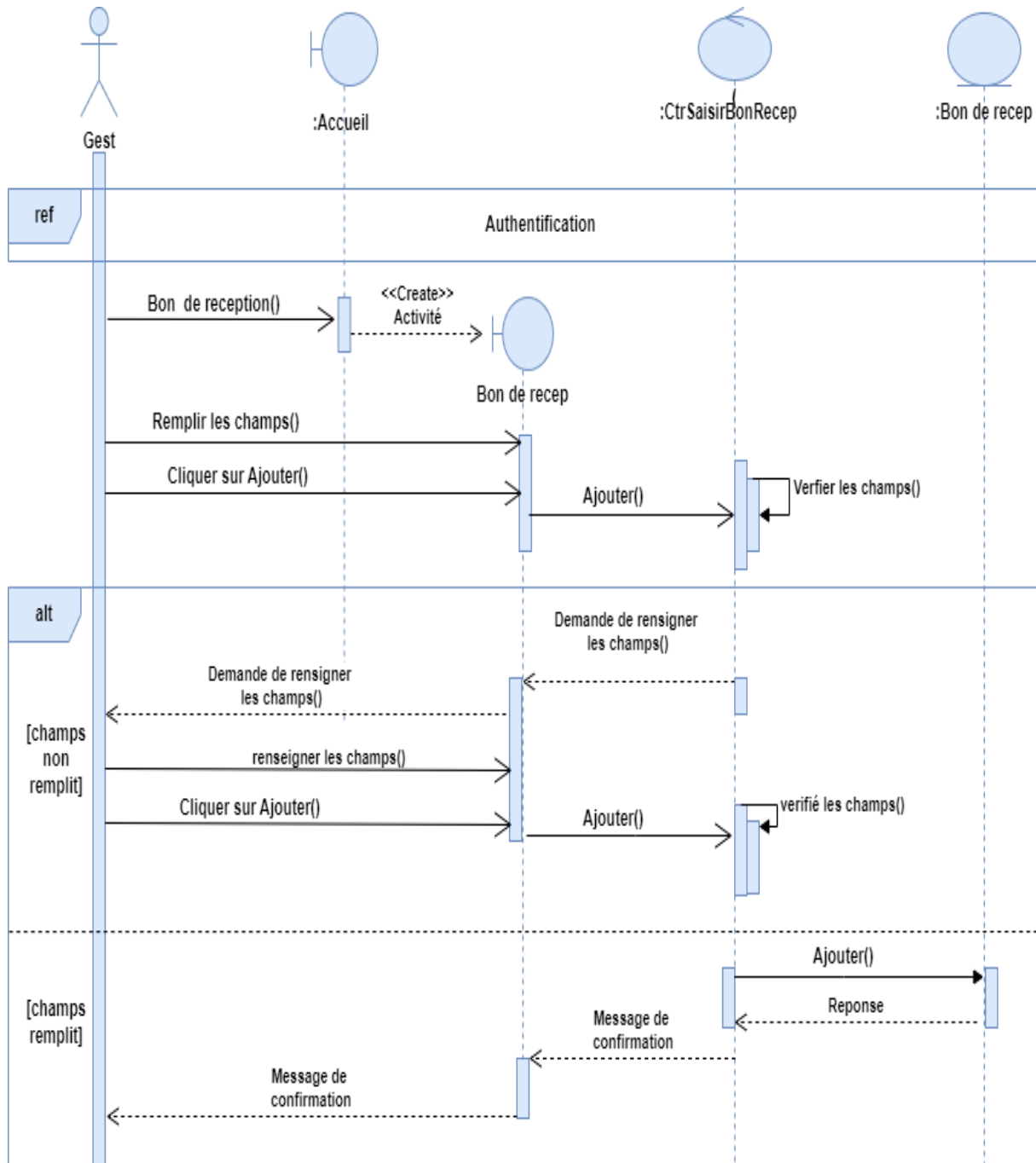


Figure III.5 – Diagramme d'interaction du cas d'utilisation « Etablir un bon de réception ».

III.2.6 Diagramme d'interaction du cas d'utilisation « Imprimer bon de réception »

Le diagramme d'interaction du cas d'utilisation «Imprimer bon réception » est illustré par la figure III.6.

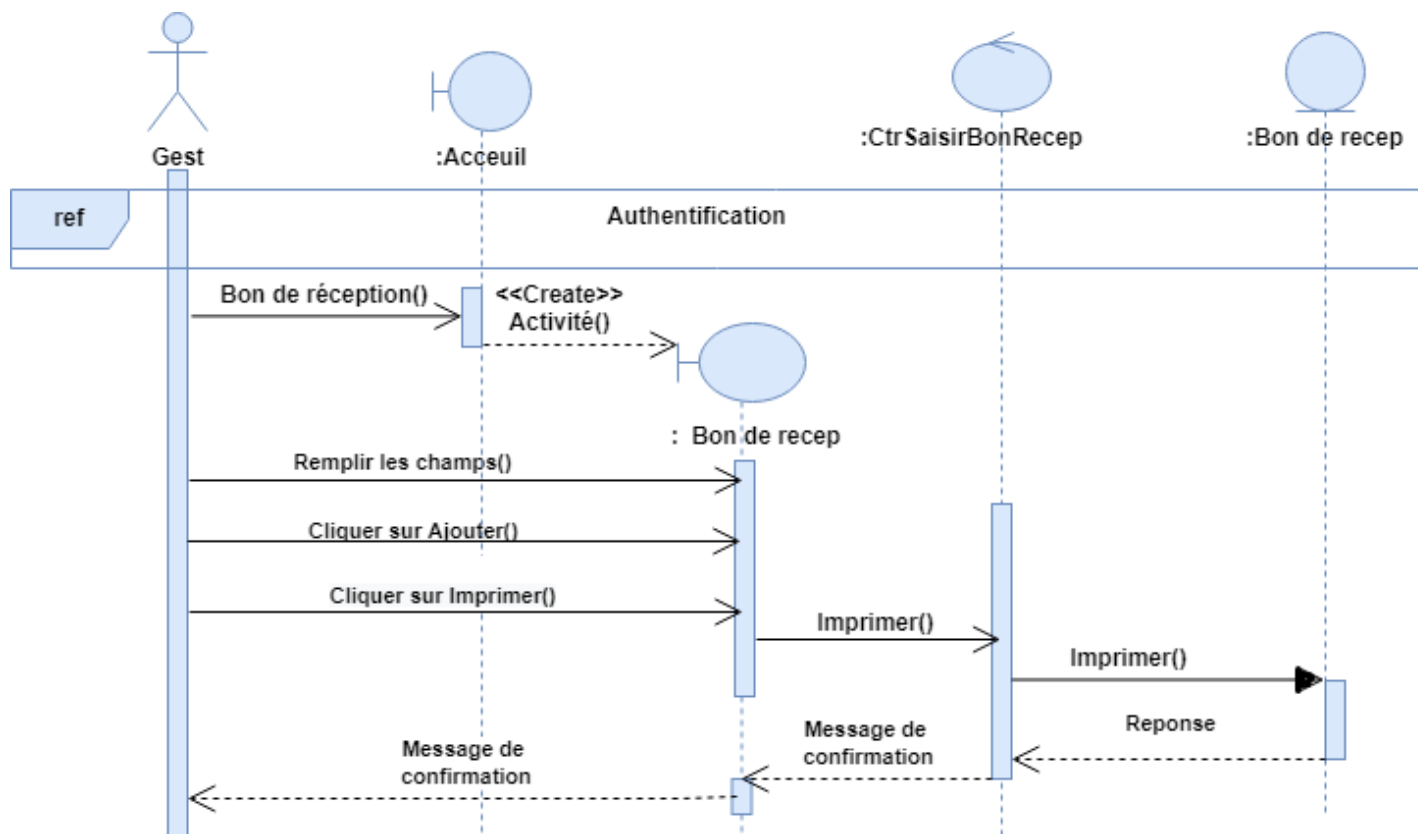


Figure III.6 – Diagramme d'interaction du cas d'utilisation « Imprimer bon de Réception ».

III.2.7 Diagramme d'interaction du cas d'utilisation « Réintégrer les bobines »

Le diagramme d'interaction du cas d'utilisation « Réintégrer les Bobines » est illustré par la figure III.7.

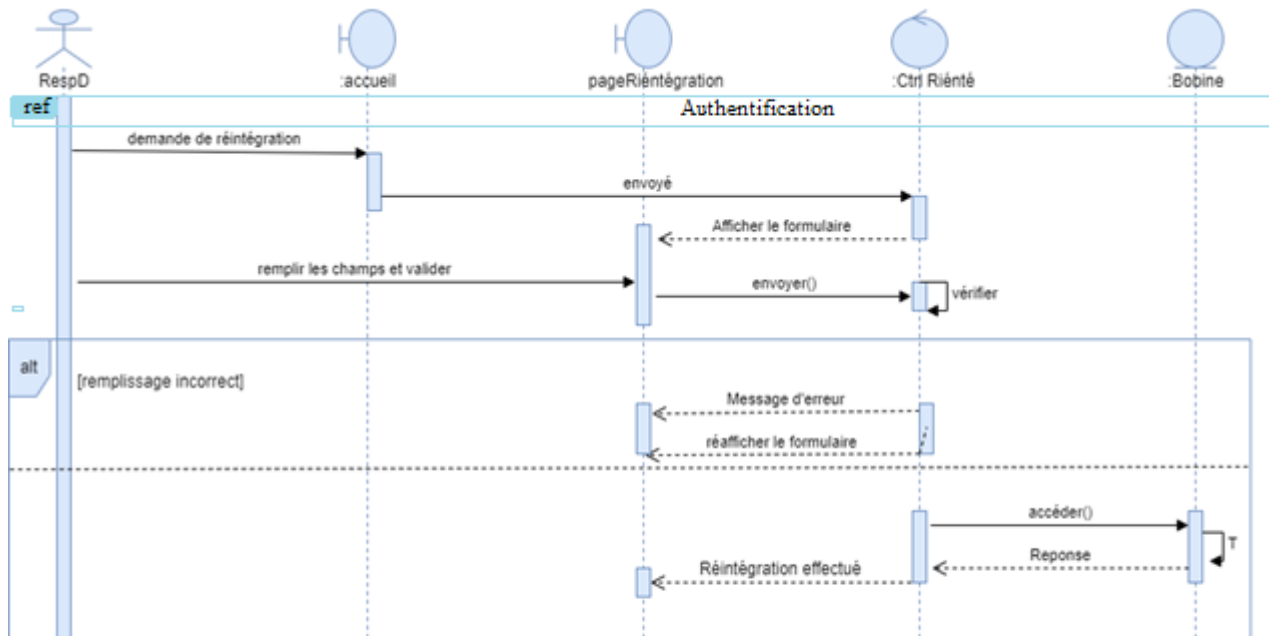


Figure III.7 – Diagramme d'interaction du cas d'utilisation « Réintégration des bobines ».

III.3 Diagramme de Classe

Le diagramme de classes représente le concept le plus important dans un développement orienté objet. Sur la branche fonctionnelle, ce diagramme traduit les besoins en spécifiant comment l'application pourra les satisfaire avant de procéder à sa réalisation. En effet, dans cette étape on va représenter l'architecture de l'application et le diagramme de classes [25].

Le diagramme de classes est défini par la figure III.8 :

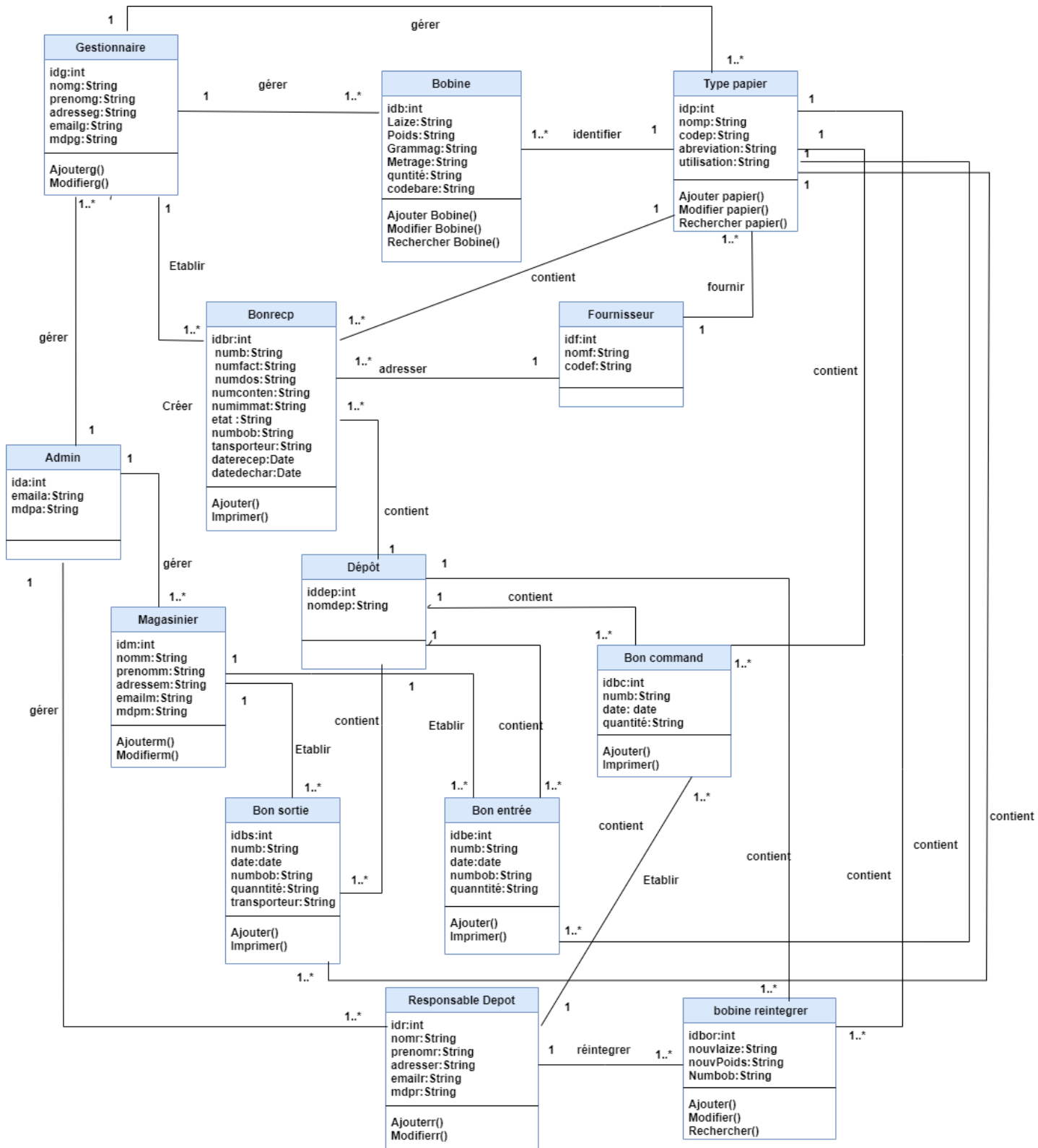


Figure III.8 – Diagramme de Classe.

III.4 Dictionnaire de données

Il s'agit d'un tableau recensant des informations rencontrées lors de l'analyse préalable. Le tableau III.1, présente le dictionnaire des données de notre application :

| Classe | Attribut | Signification | Type |
|--------------------------|-----------|-----------------------------------|---------|
| Gestionnaire | idg | Identifiant du gestionnaire | INT |
| | nomg | Nom du gestionnaire | VARCHAR |
| | prénom | Prénom du gestionnaire | VARCHAR |
| | adressesg | Adresse du gestionnaire | VARCHAR |
| | emailg | Email du gestionnaire | VARCHAR |
| | mdpg | Mot de passe du gestionnaire | VARCHAR |
| Magasinier | idm | Identifiant du magasinier | INT |
| | nomm | Nom du magasinier | VARCHAR |
| | prénom | Prénom de magasinier | VARCHAR |
| | adressesm | Adresse du magasinier | VARCHAR |
| | emailm | Email du magasinier | VARCHAR |
| | mdpm | Mot de passe du magasinier | VARCHAR |
| Responsable dépôt | idr | Identifiant du responsable dépôt | INT |
| | nomr | Nom du responsable dépôt | VARCHAR |
| | pre nomr | Prenom de responsable de dépôt | VARCHAR |
| | adressesr | Adresse de responsable dépôt | VARCHAR |
| | emailr | Email de responsable dépôt | VARCHAR |
| | mdpr | Mot de passe du responsable dépôt | VARCHAR |
| Fournisseur | idf | Identifiant du fournisseur | INT |
| | codef | Code de fournisseur | VARCHAR |
| | Nomf | Nom de fournisseur | VARCHAR |

| | | | |
|----------------------|--------------|------------------------------|---------|
| Bobine | idb | Identifiant du la bobine | INT |
| | Laize | La largeur de la bobine | VARCHAR |
| | Poids | Le poids de la bobine | VARCHAR |
| | Grammage | Le grammage de la bobine | VARCHAR |
| | Métrage | La longueur de la bobine | VARCHAR |
| | quantité | La quantité de bobines | VARCHAR |
| | Code barre | Le code-à-barre | VARCHAR |
| Type papier | idp | Identifiant de papier | INT |
| | nomp | Nom de papier | VARCHAR |
| | codep | Code de papier | VARCHAR |
| | abreviation | abreviation de papier | VARCHAR |
| | Utilisation | Utilisation de papier | VARCHAR |
| Bon Entrée | idbe | Identifiant du bon d'entrée | INT |
| | date | La date d'entrée des bobines | DATE |
| | numb | Numero de bon | VARCHAR |
| | numbob | Numero de la bobine | VARCHAR |
| | quantité | Quantité de bobines entrée | VARCHAR |
| Bon de sortie | idbs | Identifiant du bon de sortie | INT |
| | date | Date de sortie des bobines | Date |
| | numb | Numero de bon de sortie | VARCHAR |
| | numbob | Numero de la bobine | VARCHAR |
| | transporteur | transporteur | VARCHAR |
| | quantité | Quantité de bobines sortie | INT |

| | | | |
|--------------------------|--------------|-------------------------------------|---------|
| BonRecep | idbr | Identifiant du bon de réception | INT |
| | Numb | Numero d bon de réception | VARCHAR |
| | daterecep | La date de réception des bobine | DATE |
| | datedechar | Date de déchargement des bobines | Date |
| | numconten | Numero de conteneur | VARCHAR |
| | numdos | Numero de dossier | VARCHAR |
| | numfact | Numero de la facture | VARCHAR |
| | Transporteur | transporteur | VARCHAR |
| | numimmat | Numero d'immatriculation | VARCHAR |
| | Etas | Etats de la bobine | VARCHAR |
| | numbr | Le numéro du bon de réception | INT |
| Bon commande | idbc | Identifiant du bon de commande | INT |
| | numb | Numero de bon de commande | VARCHAR |
| | date | Date de commande | Date |
| | quantité | Quantité de bobine commandé | |
| Dépôt | iddep | Identifiant du dépôts | INT |
| | nomdep | Nom de dépôts | VARCHAR |
| Bobine réintégrer | idbor | Identifiant de la bobine réintégrer | INT |
| | nouvLaize | La nouvelle largeur de la bobine | VARCHAR |
| | nouvPoids | Le nouveau Poids de la bobine | VARCHAR |
| | Numbob | Numéro de la bobine | VARCHAR |
| Admin | ida | Identifiant de l'administrateur | INT |
| | emaila | Email de l'administrateur | VARCHAR |
| | mdpa | Mot de passe de l'administrateur | VARCHAR |

Table III.1 – Dictionnaire de données

III.5 Compréhension des règles

Les règles de gestion de notre application se traduisent comme suit :

- ❖ Une bobine est identifiée par un seul type papier ;
- ❖ Un type de papier identifie un ou plusieurs (bobines, bon de réception, bon d'entrée, bon de sortie, bon de commande) ;
- ❖ Le gestionnaire peut gérer un ou plusieurs (bobine, type papier) ;
- ❖ Bobine, type papier sont gérés par un seul gestionnaire ;
- ❖ Le magasinier consulte un ou plusieurs bobines ;
- ❖ La bobine est consultée par un seul magasinier ;
- ❖ Le magasinier établit un ou plusieurs bons d'entrée/sortie ;
- ❖ Le bon d'entrée/sortie est établi par seul magasinier ;
- ❖ Le fournisseur fournit un ou plusieurs type papier ;
- ❖ Un type papier est fourni par un seul fournisseur ;
- ❖ Le gestionnaire établit un ou plusieurs bons de réceptions ;
- ❖ Le bon de réception est établi par un seul gestionnaire ;
- ❖ Le bon de réception contient un seul dépôt, un seul type papier ;
- ❖ Le bon de réception est adressé par un seul fournisseur ;
- ❖ Le Responsable Dépôt établit un ou plusieurs bon de commande ;
- ❖ Le bon de commande est établi par un seul responsable de dépôts ;
- ❖ Le bon de commande contient un seul dépôt, un seul type papier ;
- ❖ Le Responsable Dépôt réintègre un ou plusieurs bobines ;
- ❖ La bobine est réintégrée par plusieurs responsables de dépôt ;
- ❖ La bobine réintégrée contient un seul dépôt, un seul type papier ;

III.6 Règles de passage au modèle Relationnelle

À partir du modèle de classes de notre application et pour obtenir le modèle relationnel, nous avons appliqué les règles de passage suivantes :

Règle 1 : Transformation des classes en relations Toute entité (classe) devient une relation ayant pour clé primaire son identifiant.

Règle 2 : Transformation des associations en relations

❖ **Association un-à-plusieurs**

Toute association hiérarchique de type (1, n) se traduit par une clé étrangère. La clé primaire correspondant à l'entité père (coté n) migre comme clé étrangère dans la relation correspondant à l'entité fille.

❖ **Association plusieurs-à-plusieurs**

Toute association hiérarchique de type (1, n) se traduit par une clé étrangère. La clé primaire correspondant à l'entité père (coté n) migre comme clé étrangère dans la relation correspondant à l'entité fille.

III.7 Modèle Relationnelle

Bobine (idb, Laize, Poids, Grammage, Métrage, quantité, #idg, #idp) ;

Type Papier (idp, nomp, codep, abreviation, utilisation, #idf, #idg) ;

Gestionnaire (idg, nomg, prenomg, adresseg, emailg, mdpg, #ida) ;

Magasinier (idm, nomm, prenomm, adressem, emailm, mdpm, #ida) ;

Responsable dépôt (idr, nomr, prenomr, adresser, emailr, mdpr, #ida) ;

Fournisseur (idf, nomf, codef) ;

Dépôt (iddep, nomdep) ;

BonRecep (idbr, numb, numdos, numconten, numimmat, numfact, etat, numbob, transporteur, datrecep, datedechar, #idg, #idp, #idf, #iddep) ;

Bon sortie (idbs, numb, numbob, date, transporteur, quantité, #idm, #idp, #iddep) ;

Bon entrée (idbe, numb, numbob, date, quantité, #idm, #idp, #iddep) ;

Bobine reintégré(idbor, nouvLaize, nouvPoids, Numbob, #idbr, #idp, #iddep) ;

Bon commande(idbc, numb, date, quantité, #idr, #idp, #iddep).

III.8 Conclusion

Dans ce chapitre, nous avons présenté les diagrammes d'interaction et le diagramme de classe qu'on a traduit en modèle relationnel afin de concevoir le schéma de la base de données.

Dans le prochain chapitre nous allons procéder à la réalisation et l'implémentation de notre application.

Implémentation

IV.1 Introduction

Après avoir effectué la conception de notre application, nous passons à présent à la phase d'implémentation. D'abord, nous allons présenter l'environnement et les outils de développement utilisés, ainsi que l'organigramme de notre l'application. Ensuite, nous allons présenter quelques interfaces de notre application et nous clôturons ce chapitre par une conclusion.

IV.2 Environnement et outils de développement

Pour développer notre système, nous avons utilisé des différents technologies notamment :

➤ **Eclipse :**

Comme un environnement de développement intégré (IDE : Integrated Development Environment) conçu pour le langage Java [16].



➤ **Oracle :**

Un système de gestion de base de données (SGBD). permettant d'assurer la définition, la manipulation, la cohérence, la confidentialité, l'intégrité, la sauvegarde et la restauration des données [28].



➤ **AJAX :**

(Asynchronous Javascript And XML) : technique qui permet de réaliser des requêtes au serveur Web pour obtenir des données ou effectuer des opérations à distance, et actualiser une partie de la page [17].



➤ **Java Enterprise Edition (Java EE) :**

Est une plateforme basée sur le langage Java. Elle est une extension de la plateforme Java SE (Java Standard Edition) à laquelle est ajouté un ensemble de classes, méthodes et des fonctions sont rassemblées dans une multitude d'interfaces de programmation (API) définissant la façon dont les composants informatiques s'invoquent l'un l'autre comme les servlets, les pages JSP et les beans [18].



➤ **JQuery :**

Est une bibliothèque de JavaScript utilisée pour automatiser les tâches courantes et simplifier les plus complexes Elle est donc utile pour la plupart des développements de scripts. En raison de sa nature extensible [19].



➤ **L'éditeur PDF :**

Permet de rendre les documents scannés consultables et éditables, également en mesure de reconnaître les formulaires éditables, de changer la taille et la police des caractères [20].



➤ **Balsamiq Mockups :**

Est un outil de conception d'interface utilisateur pour la création de wireframes (parfois appelés maquettes (Mockups) ou prototypes basse fidélité). Il peut être utilisé pour générer des croquis numériques des idées ou concept pour une application ou un site Web, pour faciliter la discussion et la compréhension avant l'écriture d'un code [6].



➤ **Apache Tomcat :**

Est un serveur http qui gère les servlets et JSP écrits en java. Grace à ce serveur, on peut héberger et déployer les servlets java d'un site internet [23].



➤ Draw.io :

Est une application web de création de diagramme compatible avec google drive et complètement gratuite, elle permet de dessiner des diagrammes UML en ligne via un navigateur (sou protocole http) [24].



➤ Modèle MVC :

Le pattern MVC permet de bien organiser son code source. Il va vous aider à savoir quels fichiers créer, mais surtout à définir leur rôle. Le but de MVC est justement de séparer la logique du code en trois parties que l'on retrouve dans des fichiers distincts, comme l'explique la description qui suit [26].

- **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.
- **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 très simples, pour afficher par exemple la liste des messages des forums.
- **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, prendre des décisions et renvoyer le texte à afficher à la vue.

➤ Langage HTML (HyperText Markup Language) :

Permettant de séparer efficacement la structure de la présentation, des feuilles de style qui donnent aux utilisateurs le contrôle de la présentation des documents (informations sur les polices de caractères, alignement, couleurs, etc.), le langage JavaScript qui permet d'écrire des scripts incorporés dans un document HTML [21].



➤ Java Script :

JavaScript est un langage relativement complexe et puissant, qui apporte des fonctions dynamiques à HTML dans les navigateurs. Malgré son nom, il est très différent du langage Java [4].



➤ **CSS (Cascading Style Sheets) :**

Le langage CSS met en forme le HTML mais ne structure pas la page web. Il permet de donner de la forme aux différents éléments de la page, en définissant par exemple une couleur de fond pour un bloc ou encore une taille de police pour un titre. Il permet également de dimensionner et positionner tous les éléments comme on le souhaite sur la page [4].



➤ **Bootstrap :**

Est une collection d'outils utiles à la création du design de sites et d'application web. C'est un ensemble qui contient des codes HTML et CSS, des formulaires, boutons, outils de navigation et autres éléments interactifs ,ainsi que des extensions JavaScript en option [4].



➤ **Java :**

Java est un langage de programmation informatique orienté objet. Java permet de développer des applications client-serveur. Côté client, les applets sont à l'origine de la notoriété du langage. C'est surtout côté serveur que Java s'est imposé dans le milieu de l'entreprise grâce aux servlets, et aux JSP (JavaServer Pages) [4].



➤ **SQL (Structured Query Language) :**

Est un langage universellement reconnu par MySQL et les autres bases de données et permettant d'interroger et de modifier le contenu d'une base de données [22].



IV.3 L’organigramme de l’application

La figure IV.1, récapitule l’organisation des pages de notre application :

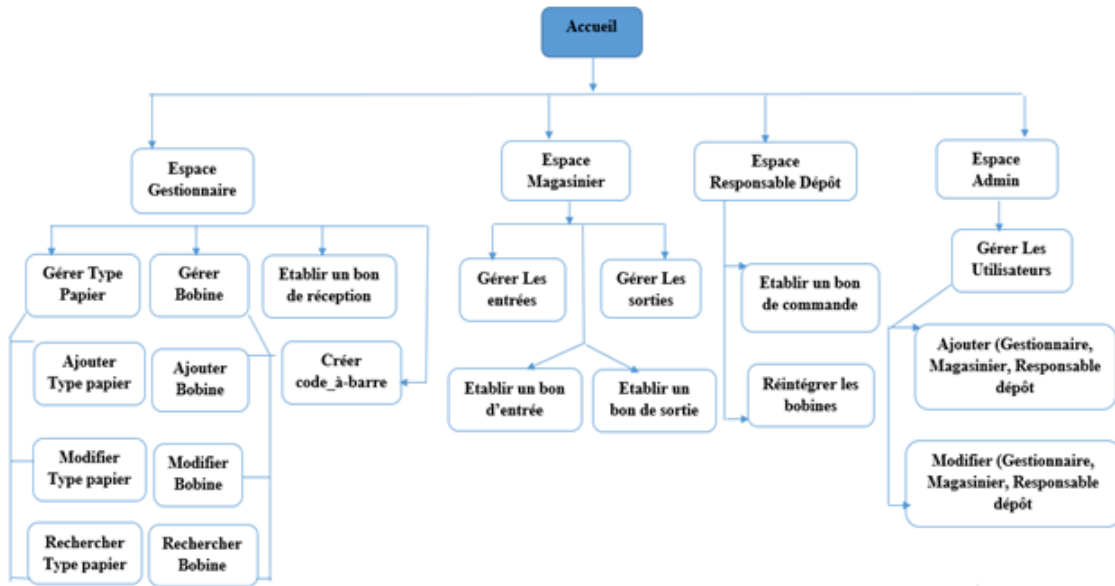


Figure IV.1 – Organigramme de l’application

IV.4 Présentation des interfaces

IV.4.1 Les interfaces dédiées au navigateur web

- Interface Accueil

La figure IV.2 représente la première interface de l’application.

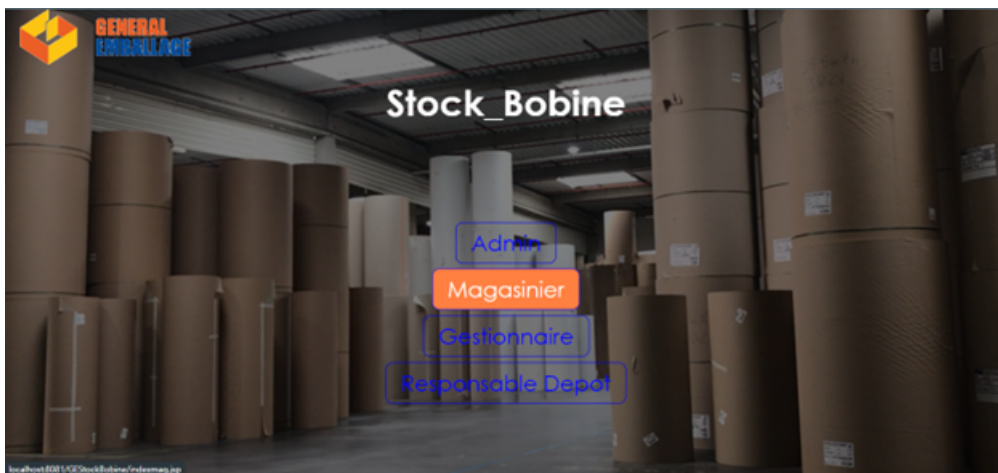


Figure IV.2 – Interface d’accueil.[29]

- **Interface d'authentification**

La figure IV.3 représente l'interface d'authentification, elle permet à l'utilisateur d'accéder à son espace après une validation de son email et son mot de passe.

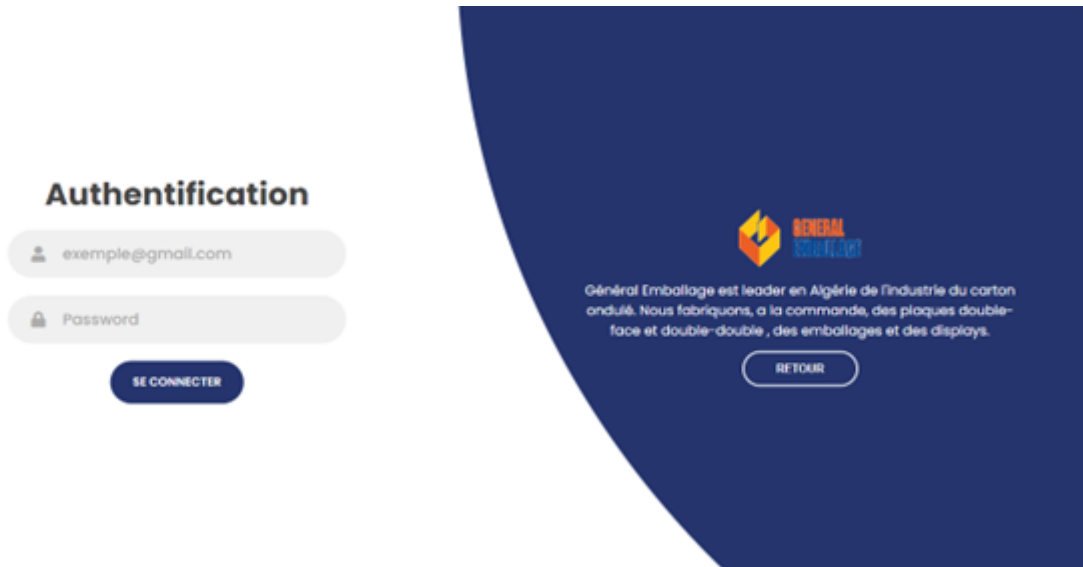


Figure IV.3 – Interface d'authentification.

- **Interface Gestionnaire**

La figure IV.4 représente l'espace de gestionnaire.

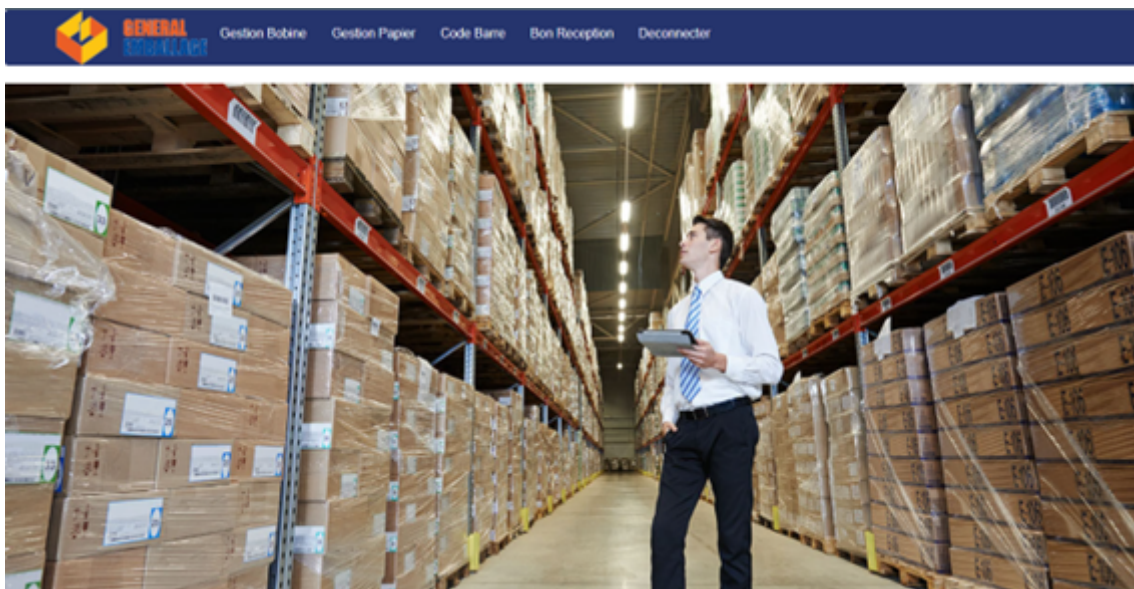


Figure IV.4 – Espace Gestionnaire.

- Interface de gestion des bobines

La figure IV.5 représente l'interface de gestion des bobines, elle permet au gestionnaire d'ajouter, modifier et rechercher les bobines.



Figure IV.5 – Interface Gestion Bobine.

- Interface de création de code-à-barre

La figure IV.6 représente l'interface de création de code-à-barre pour les bobines.

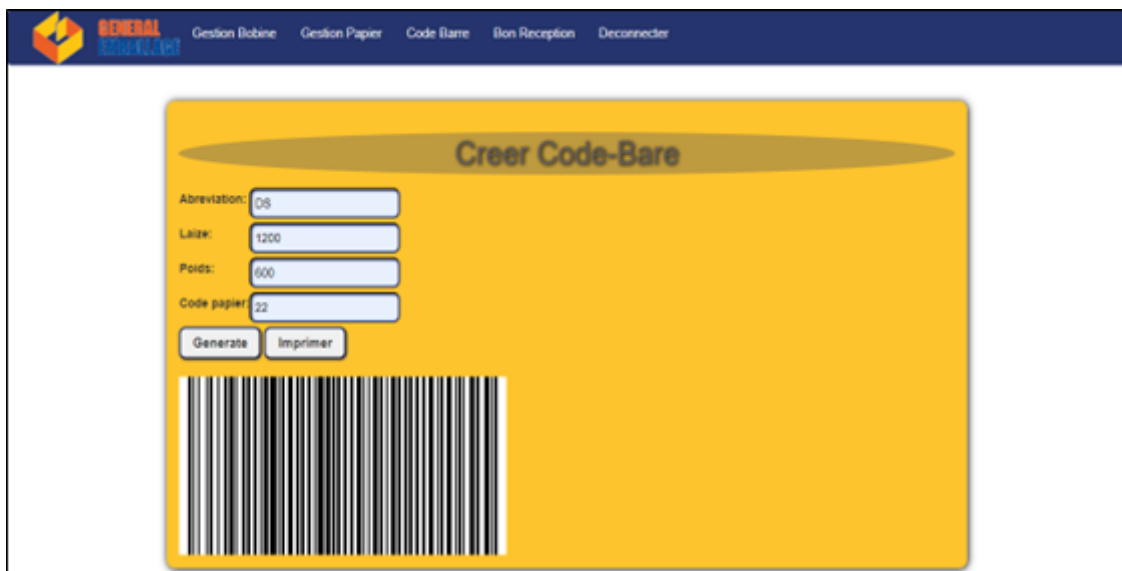


Figure IV.6 – Interface de création de code-à-barre.

- Interface de création de bon de sortie

La figure IV.7 représente l'interface de création de bon de sortie.



Figure IV.7 – Interface de création de bon de sortie.

- Exemple d'un bon de sortie imprimable

La figure IV.8 représente un bon de sortie imprimable.

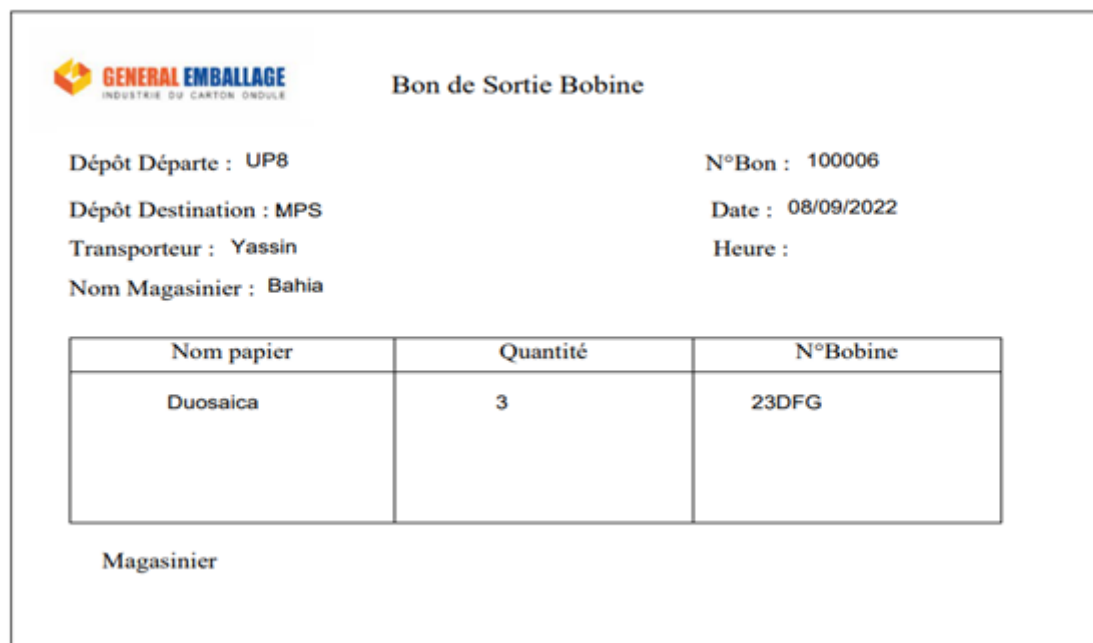


Figure IV.8 – Bon de sortie imprimable.

- **Interface de Réintégration des bobines**

La figure IV.9 représente l'interface de réintégration des bobines, elle permet au responsable de dépôt de réintégrer les bobines qui ne sont pas complètement utiliser lors de la production.

| N°Bobine | Nouvelle Laize | Nouvelle Poids | Nom papier | Emplacement |
|----------|----------------|----------------|------------|-------------|
| 234DERT | 1000 | 270 | Duosaka | MPS |

Figure IV.9 – Interface de réintégration des bobines.

IV.4.2 Les interfaces dédié au Terminal mobile « Zebra MC33 »

- **Interface d'accueil**

La figure IV.10 représente l'interface d'accueil sur le terminale mobile « Zebra MC33 ».

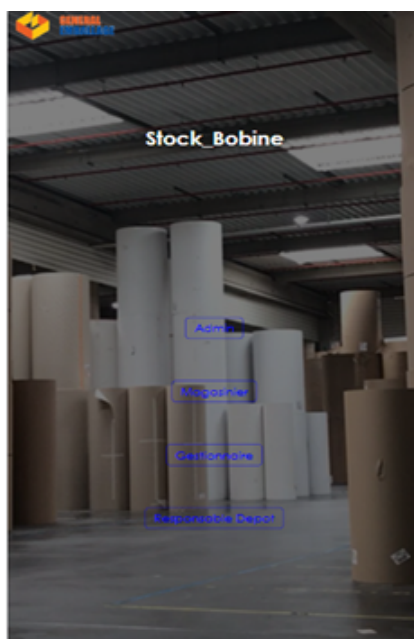


Figure IV.10 – Interface d'accueil.

- **Interface d'authentification**

La figure IV.11 représente l'interface d'authentification sur le terminal mobile « Zebra MC33 ».

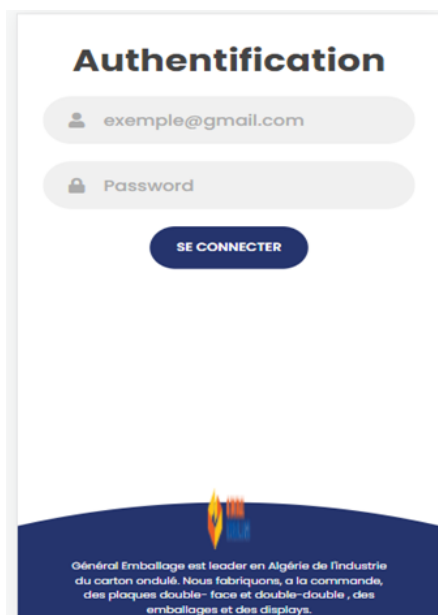


Figure IV.11 – Interface d'authentification.

- **Interface gestionnaire**

La figure IV.12 représente l'espace de gestionnaire sur le terminal mobile « Zebra MC33 ».



Figure IV.12 – Espace gestionnaire.

- Interface de gestion des bobines

La figure IV.13 représente l'interface de gestion des bobines sur le terminal mobile « Zebra MC33 ».

The screenshot shows a mobile application interface with a yellow background. At the top, there is a blue oval containing the text 'Ajoute Bobine'. Below this, there are several input fields: 'Laize:' with a text box, 'Poids:' with a text box, 'Grammage:' with a text box, 'Metrage:' with a text box, 'Nom papier:' with a dropdown menu showing 'Sélectionner un papier', and 'Quantité:' with a text box. Below these fields are two buttons: 'Ajouter' and 'Annuler'. At the bottom of the form area, there is a 'Show' dropdown menu set to '10' and the text 'entries', and a 'Search:' text box. At the very bottom, there is a yellow bar with the text 'Liste des Bobine'.

Figure IV.13 – Espace Gestion Bobine.

- Interface de création de code-à-barre

La figure IV.14 représente l'interface de création de code-à-barre pour les bobines sur le terminale mobile « Zebra MC33 ».

The screenshot shows a mobile application interface with a yellow background. At the top, there is a blue oval containing the text 'Creer Code-Bare'. Below this, there are several input fields: 'Abreviation:' with a text box containing 'FL', 'Laize:' with a text box containing '1200', 'Poids:' with a text box containing '600', and 'Code papier:' with a text box containing '44'. Below these fields are two buttons: 'Generate' and 'Imprimer'. At the bottom, there is a barcode.

Figure IV.14 – Espace de création de code-à-barre.

- **Interface de création de bon de sortie**

La figure IV.15 représente l'interface de création de bon de sortie sur le terminal mobile « Zebra MC33 ».



Figure IV.15 – Espace de création de bon de sortie.

- **Interface de réintégration des bobines**

La figure IV.16 représente l'interface de réintégration des bobines sur le terminale mobile « Zebra MC33 ».



Figure IV.16 – Espace de réintégration des bobines.

- **Choix de Couleurs**

on a choisit la couleurs des interfaces de notre application jaune et bleu a partir de logo de l'entreprise "Generale Emballage".

IV.5 Conclusion

La dernière partie de ce projet est dédiée à la présentation de notre application. Cette partie a pour objet de décrire l'environnement de développement et les différents outils que nous avons utilisés, ainsi que les bibliothèques et les librairies qui nous ont aidées à la réalisation de notre application. Nous avons par la suite présenté notre application à travers une arborescence des vues de cette dernière. Ensuite, nous avons exposé le résultat de développement à l'aide des captures écran.

Conclusion générale et perspectives

Tout au long de ce rapport, nous avons présenté les différentes étapes de réalisation de l'application au sein de l'entreprise « Général emballage ». L'application réalisée permet de garder une trace sur tous les mouvements effectués notamment la gestion des bobines, des types papiers, entrée et sortie des bobines dans le stock, et les transferts entre les différents dépôts, et de faciliter toutes sortes d'interaction avec l'utilisateur en matière de saisie, de mises à jour, et de génération des états imprimable tels que les bons de réception, de commande, d'entrée et sortie, ainsi que l'attribution des codes à barres pour les bobines afin de faciliter leur identification et leur traçabilité.

Pour le développement de cette application, nous avons suivi un processus de développement UP, avec le langage de modélisation UML, ce qui nous a permis de mener correctement les différentes phases de ce processus.

Pour la mise en œuvre de l'application, nous avons eu recours à la programmation avec l'environnement « Éclipse », la plateforme « Java EE », le serveur « Tomcat », le SGBDR 'Oracle'. Ces outils, nous ont permis de réaliser facilement la partie IHM de l'application, ainsi que la base de données.

Le stage que nous avons effectué au sein de l'entreprise « Général emballage » a été pour nous une expérience enrichissante, qui nous a permis de mettre en pratique et développer nos connaissances et compétences en termes de conception et de programmation que nous avons acquis tout au long de notre cursus universitaire.

Comme perspectives, le projet peut être amélioré, en ajoutant de nouveaux modules ou interfaces telles que le scanne de code a barre.

Bibliographie

- [1] M. Kabori, Problématique de la gestion des stocks dans les secteurs hôteliers. Cas de l'hôtel Lac Kivu Lodge de 2009 à 2011.
- [2] P. Nguka, Conception pour la gestion d'un système d'information pour la gestion des stocks dans une entreprise commerciale. Cas de l'entreprise Goshop.
- [3] <http://www.solushop.com/terminaux-industriels/355-zebra-mc3300-ordinateur-portable-de-poche.html>. Consulté le 22/08/2022.
- [4] Pascal Roques, « Les cahiers du programmeur UML2 : Modéliser une application web », Eyrolles, 4^{ème} édition.
- [5] Alphalives, La maquette du site web : la troisième étape d'un projet web, Paris, 2001- 2020, <https://www.alphalives.com/agence-web-paris/projet-web/maquette-web>. Consulté le 22/08/2022.
- [6] Balsamiq, Balsamiq for Desktop Documentation, Introduction to Balsamiq Wireframes for Desktop, 2008-2020, <https://balsamiq.com/wireframes/desktop/docs/intro/> Consulté le 22/08/2022.
- [7] Raphael Yende. SUPPORT DE COURS DE GENIE LOGICIEL. Licence. RDC (BÉNI), Congo Kinshasa. 2019. cel-01988734.
- [8] Claude BELLEIL, « UML2.0 Diagramme de contexte », Support de cours, Université de Nantes, France, 2003.
- [9] Atefsd.weebly, Diagramme de cas d'utilisation, http://atefsd.weebly.com/uploads/5/0/3/6/503639/csi_ Consulté le 24/08/2022.
- [10] IBM knowledge center, Conception applications system, https://www.ibm.com/support/knowledgecenter/fr/SS8PJ7_9.5.0/com.ibm.xtools.modeler.doc/topics Consulté le 24/08/2022.
- [11] Gilles Roy, Conception de base de données avec UML : PUQ,2007.

- [12] Djilil, DJ, gestion d'un cabinet médical, licence en informatique, université abou bekrbal-kaid, telmcen, 2014/2015.
- [13] Cabay D et Cabay J, UML2 analyse et conception, Dunnod.s.
- [14] <https://www.scaledagileframework.com/domaine-modeling>. Consulter le 05/06/2022.
- [15] UML, C. Johnen, UML : DIAGRAMME DE CONTEXTE STATIQUE, IUT de Bordeaux, V2.
- [16] <http://doc.ubuntu-fr.org/eclipse>. Consulter le 19/06/2022
- [17] AJAX : Développer pour le web 2.0, 2^{ème} édition.
- [18] Stephanie Bodoff. THE J2EE Tutorial. Addison-Wesley, Mars 2002.
- [19] J. Chaffer et K. Swedberg. jQuery Simplifiez et enrichissez vos développements JavaScript.PEARSON, 3^{ème} édition.
- [20] <https://www.futura-sciences.com/tech/questions-reponses/outils-pdf-sont-10-meilleurs-logiciels-edition-pdf-15682/>. Consulter le 20/06 /2022.
- [21] Fabrice Lemainque. HTML, XHTML, CSS, Scripts - Le Guide Complet. Micro Application, 2010.
- [22] Christian Soutou. UML 2 pour les bases de données. EYROLLES 2007.
- [23] A.Vukotic,J.Goodwil,Apache Tomcate 7,Apress.
- [24] <http://drawio-app.com>. Consulter le 25/06/2022.
- [25] GRAÏNE. S " UML2pour modélisation orientée objet et des exercices corrigés " Edition l'Abeille, 2009.
- [26] Mathieu Nebra, «Organiser son code selon l'architecture MVC». <http://openclassrooms.com/courses/concevez-votre-site-web-avec-php-et-mysql/organiseron-code-selon-l-architecture-mv>. Consulter le 02/04/2022.
- [27] T. BOUCHOUCHANE Y. TITOUAH, « Conception et réalisation d'une application de gestion de stock de ALL PLAST », Mémoire fin de cycle Master , Université A/Mira Bejaia, Algérie 2018.
- [28] <https://web.maths.unsw.edu.au/lafaye/CCM/oracle/oracintro.htm>
- [29] PrésentationGE.pdf

Résumé

Vu le volume important des informations liées à la gestion de stock dans les entreprises, l'utilisation d'un système de gestion de stock est indispensable. L'objectif de ce projet est de développer un système de gestion des mouvements des bobines papiers pour l'entreprise « Général Emballage » spécialisé dans la fabrication et la transformation du carton ondulé.

La modélisation de notre application a été faite par le langage UML en suivant des étapes de processus UP. L'implémentation de l'application a été réalisée en utilisant l'IDE « Eclipse », Le SGBDR « Oracle », le serveur « Tomcat », la plateforme « Java EE », avec l'architecture MVC.

Mots clés : UML, Java EE, Tomcat, Eclipse, MVC, Oracle, UP.

Abstract

Given the large volume of information related to stock management in companies, the use of an inventory management system is essential. The objective of this project is to develop an inventory management system for the company « Général Emballage » specialized in the manufacture and processing of corrugated cardboard.

The modelling of our system has been done by the UML language by following steps of the UP process. The implementation of the application was carried out using "Eclipse" software, the "Tomcat" server and the "Java EE" platform, with the MVC architecture.

Key words : UML, Java EE, Tomcat, Eclipse, MVC, Oracle, UP.