

Ref : 2020/N°.....

République Tunisienne

ESIP-GAFSA



Etablissement d'Enseignement
Supérieur Privé Agréé par l'Etat
sous N° 05-2013

Ecole Supérieure d'Ingénieurs Privée de Gafsa
ESIP-Gafsa

Rapport de stage Technicien

Spécialité: (Logiciel et Systèmes d'Information)

Par

Trabelsi Oussama

**Conception et réalisation d'une application de gestion de stock en
temps réel avec code à barres**

De 15/07/2019 jusqu'à 30/08/2019 :

Encadrant industriel : Mr.Mehdi Baccar

Organisme : AZUR CONSULTING



Année Universitaire: 2018/2019



AZUR Consulting

ATTESTATION DE STAGE

Nous soussignés, la société AZUR CONSULTING, domiciliée au 12, Passage de l'information Cité El-Habib, EZZAHRA 2034 – Ben Arous, attestons par la présente que :

MR. OUSSAMA TRABELSI, a effectué un stage technicien au sein de notre société AZUR CONSULTING du 15/07/2019 au 30/08/2019 et a réalisé une application « conception et réalisation d'une application de gestion de stock en temps réel avec code à barre ».

Cette attestation est délivrée à l'intéressé(e) pour servir et valoir ce que de droit

Fait à Tunis le 31/08/2019

Le maître de stage

Mehdi Baccar

AZUR Consulting
12, Passage de l'Information
Cité El Habib - Ezzahra - 2034
MF: 1447765/C/A/M/000
Tél: 79 486 007 - GSM: 52 374 535

Remerciement

Je profite par le biais de ce rapport, pour exprimer mes vifs remerciements à toutes personnes contribuant de près ou de loin à la réalisation de ce travail.

En hommage à leur sympathie, je tiens à remercier vivement tous les responsables de Azur Consulting pour leur chaleureux accueil et pour leurs multitudes d'aides.

Je remercie tout spécialement mes enseignants au sein de l'ESIP Gafsa et mon encadreur professionnel Mr. Mehdi Baccar pour leur encadrement tout au long de ce projet, leur patience, leur disponibilité et leurs conseils toujours avisés.

J'ai eu l'honneur et le privilège de travailler sous leur aide. Je tiens à remercier aussi tous ceux qui m'ont tendu une main affectueuse et encourageante pour l'élaboration de ce travail.

Table des matières

1	Cadre du projet	2
1.1	Introduction	2
1.2	Cadre du travail	3
1.3	Présentation de l'organisme d'accueil	3
1.4	Étude de l'existant	4
	1.4.1 Présentation du progiciel OCTAL	4
	1.4.2 Critique du système existant	7
1.5	Solution proposée	8
1.6	Approches du travail	9
1.7	Méthodologie et formalismes adoptés	9
1.8	Conclusion	10
2	Analyse et spécification des besoins	11
2.1	Introduction	12
2.2	Analyse globale de l'application	12
	2.2.1 Description des acteurs	13
	2.2.2 Spécification des besoins fonctionnels	13
	2.2.3 Spécification des besoins non fonctionnels	14
2.3	Diagramme des cas d'utilisation général	16
2.4	Affectation des priorités	17
2.5	Raffinement des cas d'utilisation	17
	2.5.1 Raffinement du cas d'utilisation «S'authentifier»	17
	2.5.2 Raffinement du cas d'utilisation «Changer l'emplacement d'un article»	18
	2.5.3 Raffinement du cas d'utilisation «Mettre à jour la quantité des articles en stock»	19
	2.5.4 Raffinement du cas d'utilisation «Gérer l'inventaire de stock»	20
	2.5.5 Raffinement du cas d'utilisation «Enregistrer une fin fabrication»	21

Table des matières

2.5.6	Raffinement du cas d'utilisation «Demander article composant»	22
2.5.7	Raffinement du cas d'utilisation «Gérer les bons de livraison»	23
2.6	Diagramme des cas d'utilisation détaillé	25
2.7	Diagramme de séquence système	26
2.7.1	Définition	26
2.7.2	Diagramme de séquence système relatif à «S'authentifier»	26
2.7.3	Diagramme de séquence système relatif à «Changer l'emplacement d'un article»	28
2.7.4	Diagramme de séquence système relatif à «Changer quantité sortie en stock»	29
2.7.5	Diagramme de séquence système relatif à «Changer quantité entrée en stock»	30
2.7.6	Diagramme de séquence système relatif à «Mettre à jour la quantité dans l'inventaire»	31
2.7.7	Diagramme de séquence système relatif à «Vérifier la quantité dans l'inventaire»	32
2.7.8	Diagramme de séquence système relatif à «Enregistrer une fin de fabrication»	33
2.7.9	Diagramme de séquence système relatif à «Demander article composant»	34
2.7.10	Diagramme de séquence système relatif à «Créer un bon de livraison»	35
2.7.11	Diagramme de séquence système relatif à «Modifier un bon de livraison»	36
2.8	Conclusion	37
3	Conception de la solution	38
3.1	Introduction	38
3.2	Diagramme de séquence	39
3.2.1	Diagramme de séquence «S'authentifier»	40
3.2.2	Diagramme de séquence «Changer l'emplacement d'un article»	41
3.2.3	Diagramme de séquence «Mettre à jour la quantité des articles en stock»	42
3.2.4	Diagramme de séquence «Enregistrer une fin de fabrication»	43
3.3	Modèle relationnel	44
3.4	Architecture de l'application	44
3.4.1	Architecture matérielle	44
3.4.2	Architecture logicielle	47
3.5	Conclusion	49

Table des matières

4	Réalisation	50
4.1	Introduction.....	50
4.2	Environnement de travail.....	51
4.2.1	Environnement matériel.....	51
4.2.2	Les outils de travail.....	51
4.3	Principales interfaces graphiques.....	59
4.3.1	Principales interfaces de l'application web.....	59
4.3.2	Principales interfaces de l'application mobile.....	68
4.4	Chronogramme de travail.....	72
4.5	Problèmes rencontrés.....	73
4.5.1	Installation de la base de donnée.....	73
4.5.2	Problème avec l'émulateur de Windows Phone.....	74
4.6	Conclusion.....	75

Table des figures

1.1	Écran d'accueil OCTAL.....	5
1.2	Le configurateur avec ses composants.....	5
1.3	La planification et l'analyse de la charge.....	6
1.4	Le module du stock.....	7
1.5	Les phases d'un processus unifié.....	10
2.1	Le diagramme de cas d'utilisation général.....	16
2.2	Le diagramme de cas d'utilisation «S'authentifier».....	17
2.3	Le diagramme de cas d'utilisation «Changer l'emplacement d'un article»	18
2.4	Le diagramme de cas d'utilisation «Mettre à jour la quantité des articles en stock».....	19
2.5	Le diagramme de cas d'utilisation «Gérer l'inventaire de stock».....	20
2.6	Le diagramme de cas d'utilisation «Enregistrer une fin fabrication»	21
2.7	Le diagramme de cas d'utilisation «Demander article composant».....	22
2.8	Le diagramme de cas d'utilisation «Gérer les bons de livraison».....	23
2.9	Le diagramme de cas d'utilisation général.....	25
2.10	Le diagramme de séquence système «S'authentifier».....	27
2.11	Le diagramme de séquence système «Changer l'emplacement d'un article»	28
2.12	Le diagramme de séquence système «Changer quantité sortie en stock»	29
2.13	Le diagramme de séquence système «Changer quantité entrée en stock»	30
2.14	Le diagramme de séquence système «Mettre à jour la quantité dans l'inventaire».....	31
2.15	Le diagramme de séquence système «Vérifier la quantité dans l'inventaire»	32
2.16	Le diagramme de séquence système «Vérifier la quantité dans l'inventaire»	33
2.17	Le diagramme de séquence système «Demander article composant»	34

Table des figures

2.18	Le diagramme de séquence système «Créer un bon de livraison» . . .	35
2.19	Le diagramme de séquence système «Modifier un bon de livraison» . . .	36
3.1	Le diagramme de séquence «S'authentifier»	40
3.2	Le diagramme de séquence «Changer l'emplacement d'un article»	41
3.3	Le diagramme de séquence «Mettre à jour la quantité des articles en stock» ..	42
3.4	Le diagramme de séquence «Enregistrer une fin de fabrication».....	43
3.5	Schéma de la base de données.....	44
3.6	Lecteur code à barres Motorola MC40-HC	45
3.7	Architecture matérielle de l'application web	46
3.8	Architecture matérielle de l'application mobile.....	47
3.9	Architecture logicielle de l'application mobile.....	48
3.10	Architecture logicielle de l'application mobile.....	48
4.1	Oracle.....	51
4.2	TOAD	52
4.3	Power AMC.....	53
4.4	Visual Studio.....	54
4.5	Windows Phone SDK	54
4.6	LATEX.....	55
4.7	AngularJS.....	57
4.8	Visual C#	58
4.9	Interface d'authentification	59
4.10	Interface du menu	60
4.11	Interface du Side menu	61
4.12	Interface changer emplacement article.....	62
4.13	Interface mettre à jour la quantité des articles en stock.....	63
4.14	Interface gérer l'inventaire.....	64
4.15	Interface demander article composant	65
4.16	Interface enregistrer une fin fabrication.....	66
4.17	Interface gérer les bons de livraison	67
4.18	Interface mobile d'authentification.....	68
4.19	Interface mobile du menu.....	69
4.20	Interface mobile changer emplacement article.....	70
4.21	Interface mobile mettre à jour la quantité des articles en stock	71
4.22	Interface mobile enregistrer une fin fabrication.....	72
4.23	Le diagramme de Gantt.....	73
4.24	Problème de l'installation de la base de données	74

Table des figures

4.25 Problème de l'émulateur Windows Phone..... 75

Liste des tableaux

2.1	Tableau d'affectation des priorités.....	17
2.2	Description textuelle du cas d'utilisation «S'authentifier»	18
2.3	Description textuelle du cas d'utilisation «Changer l'emplacement d'un article» 19	
2.4	Description textuelle du cas d'utilisation «Mettre à jour la quantité des articles en stock»	20
2.5	Description textuelle du cas d'utilisation «Gérer l'inventaire de stock»	21
2.6	Description textuelle du cas d'utilisation «Enregistrer une fin fabrication» ..	22
2.7	Description textuelle du cas d'utilisation «Demander article composant»	23
2.8	Description textuelle du cas d'utilisation «Gérer les bons de livraison»	24
4.1	Oracle.....	52
4.2	Toad	52
4.3	Power AMC.....	53
4.4	Visual Studio	54
4.5	Windows Phone SDK.....	55
4.6	AngularJS.....	57
4.7	Visual C#	58

Introduction générale

Dans la plupart des entreprises, les stocks de produits constituent une source d'information indispensable à la bonne gestion puisqu'ils sont considérés comme valeurs d'exploitation qui représentent généralement une part importante de l'actif.

De nos jours, tant d'entreprises sont confrontées à des problèmes liés à la gestion des stocks. La nécessité d'une solution informatique permettant de diminuer voir même d'éliminer les pertes du temps et la difficulté de gérer les stocks devient un besoin urgent. La puissance de l'informatique offre des possibilités d'automatiser des procédures afin de gagner du temps et de diminuer le risque d'erreur. L'une des solutions recommandées à ce genre de problème est d'utiliser un système de gestion avec code à barres.

En effet, le travail demandé consiste à concevoir et réaliser une application web et mobile de gestion de stock en temps réel en utilisant le code à barres ayant pour objectif d'offrir une rapidité et portabilité de traitement des stocks. Pour relater le travail réalisé tout au long de la période de notre stage, nous avons opté pour diviser le présent rapport en quatre chapitres.

Dans le premier chapitre, nous introduisons le cadre dans lequel nous avons effectué notre projet. Nous énonçons ainsi la description du système actuel, l'étude de l'existant et enfin les solutions proposées.

Nous avons ensuite consacré le deuxième chapitre à l'analyse des différents besoins fonctionnels et non fonctionnels ainsi que la modélisation des diagrammes de cas d'utilisation et les diagrammes de séquences système de notre application.

Dans le troisième chapitre, nous avons présenté la conception, dans laquelle nous avons présenté les diagrammes de séquences détaillés et le schéma des tables de la base de données.

Dans le quatrième chapitre, nous expliquons la réalisation de toutes les fonctionnalités nécessaires pour notre application, la définition des différents outils utilisés ainsi que la présentation des interfaces proposées.

Enfin, nous récapitulons dans la conclusion le fruit de notre travail.

Chapitre 1 Cadre du projet

Sommaire

1.1	Introduction	2
1.2	Cadre du travail	3
1.3	Présentation de l'organisme d'accueil	3
1.4	Étude de l'existant.....	4
1.4.1	Présentation du progiciel OCTAL	4
1.4.2	Critique du système existant	7
1.5	Solution proposée.....	8
1.6	Approches du travail	9
1.7	Méthodologie et formalismes adoptés.....	9
1.8	Conclusion.....	10

1.1 Introduction

Ce premier chapitre est consacré à la présentation générale du cadre de notre projet. Dans une première partie, nous allons se concentrer sur la présentation de l'organisme d'accueil, à savoir Azur consulting. Par la suite, nous allons décrire l'existant en détaillant les problèmes rencontrés au niveau de la société Azur consulting. Finalement, nous allons introduire notre solution ainsi que les fonctionnalités attendues de notre nouveau système.

1.2 Cadre du travail

Ce stage s'inscrit dans le cadre d'un projet de fin d'études pour l'obtention d'un diplôme de licence appliquée en informatique de gestion de l'école supérieure des sciences économiques et commerciales de Tunis. Notre stage a été effectué au sein d'une société de conseil en système d'information « Azur Consulting ». Le sujet est intitulé "Conception et réalisation d'une application web et mobile de gestion de stock avec code à barres".

1.3 Présentation de l'organisme d'accueil

Dans cette section, nous présentons Azur Consulting au sein de laquelle le stage de 3 mois a été passé.



Date de création : 1995

Siège : 4 bis Rue Hassen Ennouri el Menzah V 2037 Ariana

Direction : Sami Gabsi

Activité : Management et systèmes d'information

Téléphone : 71 230 615 / 71 230 447 Fax : 71 230 507

Azur Consulting est un cabinet de conseil en management et en système d'information. Cet organisme propose aux entreprises industrielles à travers ses consultants

tunisiens et étrangers, le conseil et l'assistance dans les domaines suivants :

- Diagnostique stratégique, préalable à la " Mise à Niveau ".
- La mise en œuvre du " Plan de Mise à Niveau ".
- L'assistance à la mise en place de votre futur Système d'information.
- La mise en place de système qualité ISO 9000.

1.4 Étude de l'existant

1.4.1 Présentation du progiciel OCTAL

Le système existant, sur lequel la société Azur Consulting repose, se base sur un progiciel de gestion intégré nommé OCTAL. Il est parmi les progiciels les plus diffusés et installés sur le marché ORACLE français, ce qui est un gage de fiabilité, de pérennité et d'évolutivité de celui-ci.

Le progiciel de gestion intégré OCTAL a une couverture fonctionnelle très large qui est décrite comme suit :

- La figure 1.1 présente l'écran d'accueil du progiciel OCTAL paramétré selon le profil utilisateur.

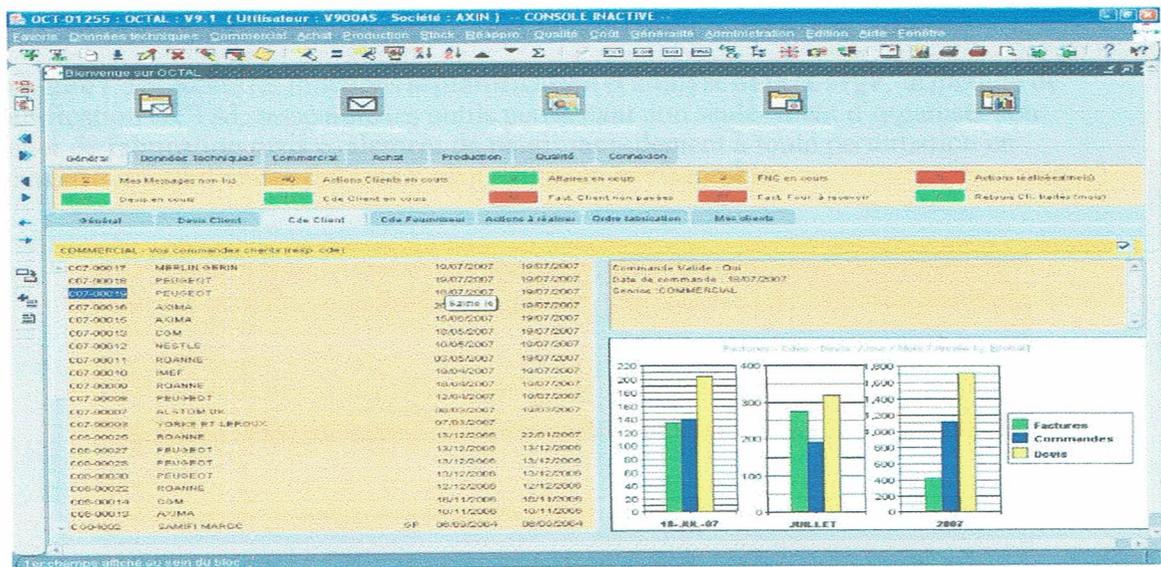


FIGURE 1.1 – Écran d'accueil OCTAL

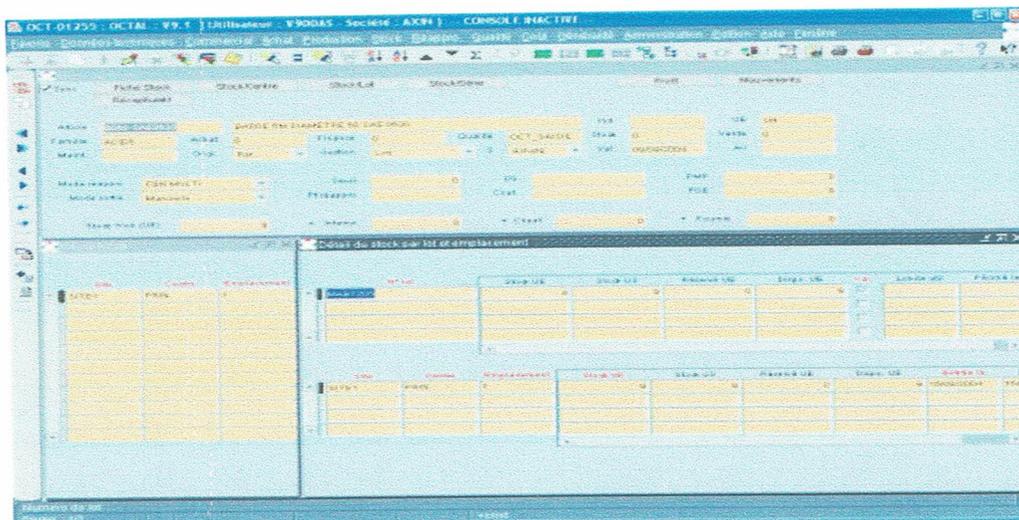


FIGURE 1.4 – Le module du stock

- **Achat** : La constitution du catalogue fournisseur est facilitée par l'utilisation d'outils de génération de consultations en masse à partir des données techniques et/ou des demandes d'achat.

1.4.2 Critique du système existant

Le système existant dans la société Azur Consulting se caractérise par :

- La difficulté de manipulation par l'utilisateur. Par exemple il y a une difficulté au court de la connexion, au court de la récupération des informations, etc.
- L'ambiguïté d'informations.
- La complexité de chargement de la base de données : à chaque connexion il faut charger la base de données à nouveau qui est une phase couteuse en terme de temps et qui réduit la performance.
- La mauvaise présentation des interfaces : interface classique et elle peut être un peu complexe pour un nouveau utilisateur.
- La longue durée sur les traitements de gestion (sortie sur stock, sortie sur ordre de fabrication, etc).
- L'absence d'outil de gestion avec code à barre qui peut réduire le temps du traitement d'une tâche comme la sortie sur stock en effectuant une saisie rapide des codes des articles .

- L'absence d'une solution portable et à temps réel pour une utilisation mobile pour l'utilisateur qui se limite à effectuer les tâches nécessaires sur une machine fixe. Donc il y a perte de temps pendant les déplacements.
- L'utilisation d'une grande masse de mémoire inutile à cause du chargement des modules et données qui ne semblent pas nécessaires pour certains traitement.

1.5 Solution proposée

Notre projet consiste à concevoir et à mettre en place une application web et mobile de gestion avec code à barre basée sur un ERP. Cette application comporte des différents modules permettant de réaliser des opérations de gestion.

- Un module **transfert** : qui permet de modifier l'emplacement d'un article ou plusieurs, d'une source à une destination. La source et la destination sont caractérisés par l'emplacement et le centre. Le code d'article est saisie à travers un scan de son code à barres.
- Un module **fin fabrication** : pour déclarer qu'un ou plusieurs articles ont terminé leurs processus de fabrication. Ce processus est caractérisé par son numéro d'ordre de fabrication qui peut être scanné.
- Un module **entrée ou sortie stock** : l'utilisateur peut scanné le code d'un article, vérifier sa désignation, choisir la quantité et le type du transaction (soit sortie, soit entrée du stock).
- Un module **sortie sur ordre de fabrication** : pour déclarer la sortie d'une ou plusieurs composantes de stock qui concerne un ordre de fabrication particulier. L'utilisateur scanne le code de l'article composant et saisie l'ordre de fabrication correspondant.
- Un module **livraison** : pour créer ou modifier des bons de livraisons des articles pour des clients.
- Un module **inventaire** : afin de vérifier l'égalité de la quantité physique et la quantité informatique des articles. En cas de différence, l'utilisateur mettra à jour cette valeur.

La mise à jour au niveau de la base de données se fait en temps réel (à chaque opération les tables se mettent à jour).

1.6 Approches du travail

Pour l'atteinte des objectifs désirés, notre travail comportait deux grandes phases :

Phase de recherche

C'est l'étape qui inclut l'étude bibliographique, dans laquelle nous devons saisir les différentes notions et technologies à utiliser dans le projet, les architectures, etc. Aussi, elle renferme les tests des différentes solutions mises en hypothèse pour réaliser l'application, et nous fixons les outils nécessaires pour la réalisation du projet.

Phase de conception et de développement

C'est une étape, dans laquelle, nous spécifions les besoins fonctionnels et nous modélisons le système à réaliser pour clarifier les tâches à accomplir dans la partie développement. Cette phase se termine par une partie qui comprend la programmation et les tests de validation.

1.7 Méthodologie et formalismes adoptés

Pour assurer un bon rendement de développement en termes de qualité et de productivité, le choix de la méthodologie en informatique est primordial. Pour ce projet nous allons adopter la méthode : Rational Unified Process comme un processus de développement logiciel.

Nous avons opté pour ce choix dû à ses caractéristiques car il est :

- Itératif
- Guidé par les besoins de l'utilisateur
- Centré sur l'architecture
- Piloté par des cas d'utilisation et orienté vers la diminution des risques.

La figure 1.5 représente les phases d'un processus unifié.

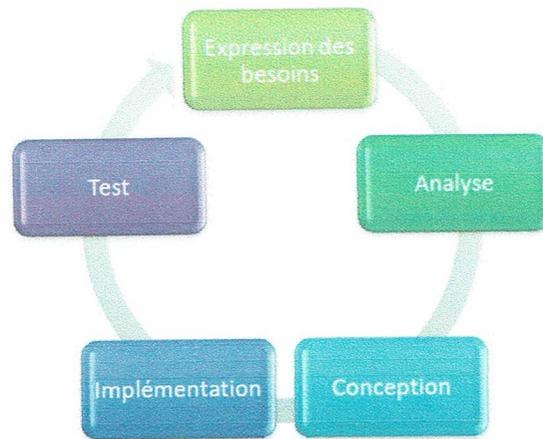


FIGURE 1.5 – Les phases d'un processus unifié

1.8 Conclusion

Dans ce chapitre introductif, nous avons présenté l'organisme d'accueil Azur Consulting dans lequel nous avons effectué notre projet. Ensuite, nous avons énoncé la description du système actuel, l'étude de l'existant et la solution proposées. Le chapitre suivant sera consacré pour présenter la phase de l'analyse et de spécification des besoins fonctionnels et non fonctionnel de ce projet.

Chapitre 2

Analyse et spécification des besoins

Sommaire

2.1	Introduction	12
2.2	Analyse globale de l'application.....	12
2.2.1	Description des acteurs.....	13
2.2.2	Spécification des besoins fonctionnels	13
2.2.3	Spécification des besoins non fonctionnels.....	14
2.3	Diagramme des cas d'utilisation général.....	16
2.4	Affectation des priorités	17
2.5	Raffinement des cas d'utilisation.....	17
2.5.1	Raffinement du cas d'utilisation «S'authentifier»	17
2.5.2	Raffinement du cas d'utilisation «Changer l'emplacement d'un article»	18
2.5.3	Raffinement du cas d'utilisation «Mettre à jour la quantité des articles en stock»	19
2.5.4	Raffinement du cas d'utilisation «Gérer l'inventaire de stock»	20
2.5.5	Raffinement du cas d'utilisation «Enregistrer une fin fabri- cation»	21
2.5.6	Raffinement du cas d'utilisation «Demander article com- posant»	22
2.5.7	Raffinement du cas d'utilisation «Gérer les bons de livraison»	23
2.6	Diagramme des cas d'utilisation détaillé.....	25
2.7	Diagramme de séquence système	26

2.7.1	Définition.....	26
2.7.2	Diagramme de séquence système relatif à «S'authentifier»	26
2.7.3	Diagramme de séquence système relatif à «Changer l'emplacement d'un article»	28
2.7.4	Diagramme de séquence système relatif à «Changer quantité sortie en stock»	29
2.7.5	Diagramme de séquence système relatif à «Changer quantité entrée en stock»	30
2.7.6	Diagramme de séquence système relatif à «Mettre à jour la quantité dans l'inventaire»	31
2.7.7	Diagramme de séquence système relatif à «Vérifier la quantité dans l'inventaire»	32
2.7.8	Diagramme de séquence système relatif à «Enregistrer une fin de fabrication»	33
2.7.9	Diagramme de séquence système relatif à «Demander article composant»	34
2.7.10	Diagramme de séquence système relatif à «Créer un bon de livraison»	35
2.7.11	Diagramme de séquence système relatif à «Modifier un bon de livraison»	36
2.8	Conclusion.....	37

2.1 Introduction

La phase d'analyse et spécification des besoins présente une étape primordiale dans le cycle de développement d'un projet. En effet, elle permet de mieux comprendre le travail demandé en dégagant les besoins des différents utilisateurs que le système doit accomplir.

Dans ce chapitre, nous allons décrire les acteurs et spécifier les besoins fonctionnels et non-fonctionnels auxquels doit répondre le système. Ensuite, nous allons présenter les diagrammes de cas d'utilisation, leurs descriptions textuelles et les diagrammes de séquence système.

2.2 Analyse globale de l'application

L'objectif de notre travail est de concevoir et développer une application web et mobile pour la gestion avec code à barres qui permettra à l'utilisateur connecté à

partir de l'application d'exécuter plusieurs opérations de gestion pour le stock.

2.2.1 Description des acteurs

Cette section a pour objectif de présenter les acteurs et leurs fonctionnalités auxquelles doit répondre notre application. Nous commençons notre analyse par identifier les acteurs qui agissent sur notre système à savoir :

Le magasinier : c'est un acteur principal qui interagit avec notre application. Cette personne bénéficie des fonctionnalités de l'application qui concerne son type d'activité comme le transfert du stock, l'entrée ou la sortie du stock et la vérification de l'inventaire.

Le responsable de production : C'est un acteur principal qui intervient seulement dans les modules qui concernent ses activités, comme la déclaration d'une fin de fabrication ou bien la sortie du stock pour une consommation de production.

Le responsable commercial : il est un acteur principal qui intervient seulement pour réaliser ou modifier les bons de livraison pour leurs traitement ultérieur par le magasinier.

2.2.2 Spécification des besoins fonctionnels

Les besoins fonctionnels sont ceux qui concernent les fonctions fournies par le projet et que l'utilisateur va s'en servir pour réaliser ses tâches du travail.

Le magasinier : il accède à l'application pour :

- Scanner ou saisir un code d'article et choisir la quantité pour transférer les articles d'une source à une destination.
- Scanner ou saisir le code d'article et choisir la quantité pour déclarer une entrée de stock ou bien une sortie de stock.
- Scanner ou saisir un code article pour vérifier si la quantité saisie dans le système est égale à la quantité physique de cette article. Dans le cas d'une différence de valeur, l'acteur fait la mise à jour de cette valeur.

Le responsable de production : Il se connecte sur l'application pour :

- Vérifier les quantités à produire et déjà produites d'un article après avoir saisi le numéro d'ordre de fabrication.
- Scanner ou saisir un numéro d'ordre de fabrication afin de déclarer la fin de production d'un ou plusieurs articles.

Le responsable commercial : Il utilise l'application afin de :

- Créer un bon de livraison.
- Modifier un bon de livraison.

2.2.3 Spécification des besoins non fonctionnels

Après avoir déterminé les besoins fonctionnels, nous présentons ci-dessous l'ensemble des contraintes à respecter pour garantir la performance du système, donc de fournir un produit performant qui respecte les exigences de l'utilisateur et qui peut faire face à des risques de panne ou de non fonctionnement.

La performance :

Afin d'être acceptée par l'utilisateur, notre application doit respecter ce critère tout en assurant un temps de réponse minimum et des fonctionnalités répondant aux besoins.

La simplicité :

Un visiteur assez modeste pourra utiliser un tel service de façon intuitive.

L'ergonomie de l'interface :

Les interfaces doivent être simple et conviviale : On doit essayer le maximum d'éliminer l'encombrement.

L'extensibilité :

dans le cadre de ce travail, l'application devra être extensible, c'est-à-dire qu'il pourra y avoir une possibilité d'ajouter ou de modifier de nouvelles fonctionnalités.

La modularité de l'application :

Avoir un code simple facile à maintenir et à comprendre en cas de besoin.

L'adaptativité du design :

Offrant toujours une meilleure navigation et structure des pages, le design adaptatif permet d'afficher l'application selon les différentes résolutions d'écran des appareils utilisés (téléphones, tablettes, écran d'ordinateur).

2.3 Diagramme des cas d'utilisation général

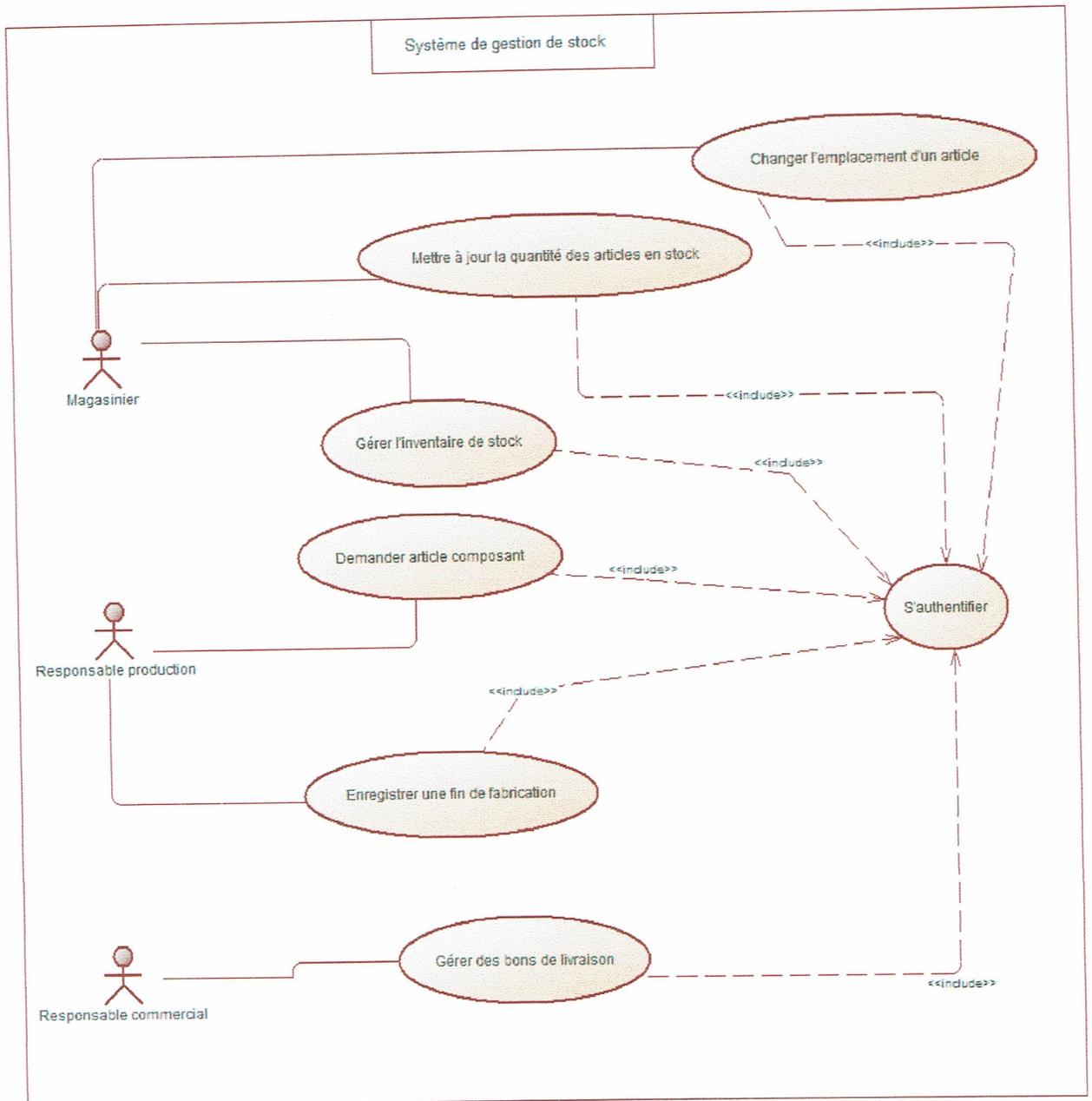


FIGURE 2.1 – Le diagramme de cas d'utilisation général

2.4 Affectation des priorités

A cette étape de cycle de développement, il est nécessaire de déterminer les cas d'utilisation les plus prioritaires. Le choix repose sur la dépendance des cas d'utilisation, de l'activité de l'entreprise et des choix des utilisateurs.

TABLE 2.1 – Tableau d'affectation des priorités

Cas d'utilisation	Acteur	priorité
S'authentifier	tout les acteurs	2
Changer l'emplacement d'un article	Magasinier	1
Mettre à jour la quantité des articles en stock	Magasinier	1
Gérer l'inventaire de stock	Magasinier	1
Demander article composant	Responsable production	1
Enregistrer une fin de fabrication	Responsable production	1
Gestion des bons de livraison	Responsable commercial	1

2.5 Raffinement des cas d'utilisation

2.5.1 Raffinement du cas d'utilisation «S'authentifier»

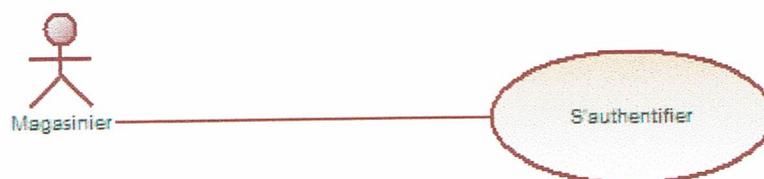


FIGURE 2.2 – Le diagramme de cas d'utilisation «S'authentifier»

TABLE 2.2 – Description textuelle du cas d'utilisation «S'authentifier»

Cas d'utilisation	S'authentifier
Acteur principale	Tout les utilisateurs du système
Objectif	Accéder au système
Pré condition	Utilisateur non authentifié
Post condition	Utilisateur authentifié
Scénario nominal	<ol style="list-style-type: none"> 1. Utilisateur entre son login et mot de passe. 2. Utilisateur valide l'opération. 3. Le système vérifie l'opération. 4. Le système affiche l'interface adéquate à l'utilisateur.
Scénario alternative	<ol style="list-style-type: none"> 3.1. Login ou mot de passe incorrect. 3.2. Retour à l'étape 1. 2.1. champs incomplets. 2.2. Retour à l'étape 1.

2.5.2 Raffinement du cas d'utilisation «Changer l'emplacement d'un article»

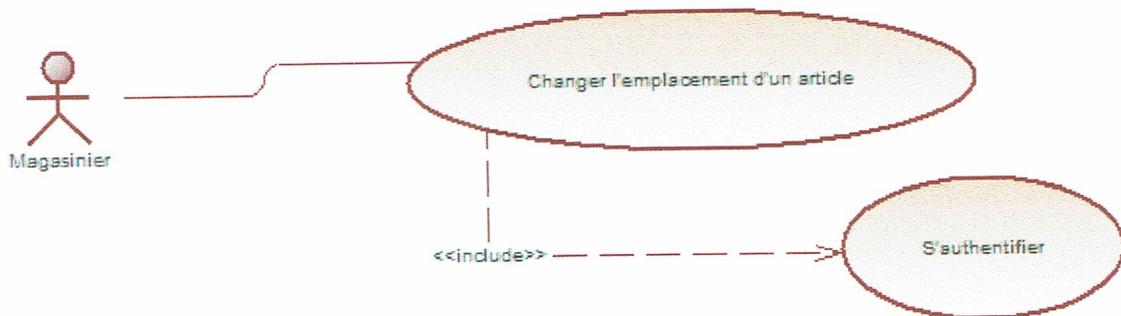


FIGURE 2.3 – Le diagramme de cas d'utilisation «Changer l'emplacement d'un article»

TABLE 2.3 – Description textuelle du cas d'utilisation «Changer l'emplacement d'un article»

Cas d'utilisation	Changer l'emplacement d'un article
Acteur principale	Magasinier
Objectif	Modifier le centre et l'emplacement d'un article
Pré condition	Magasinier authentifié
Post condition	Événement géré
Scénario nominal	1. Le magasinier introduit le code de l'article par clavier ou bien par scan de son code à barres et les valeurs à changer. 2. Le magasinier valide la modification. 3. Le système confirme la modification.
Scénario alternative	1.1. Champs obligatoire no rempli. Retour à l'étape 1.

2.5.3 Raffinement du cas d'utilisation «Mettre à jour la quantité des articles en stock»

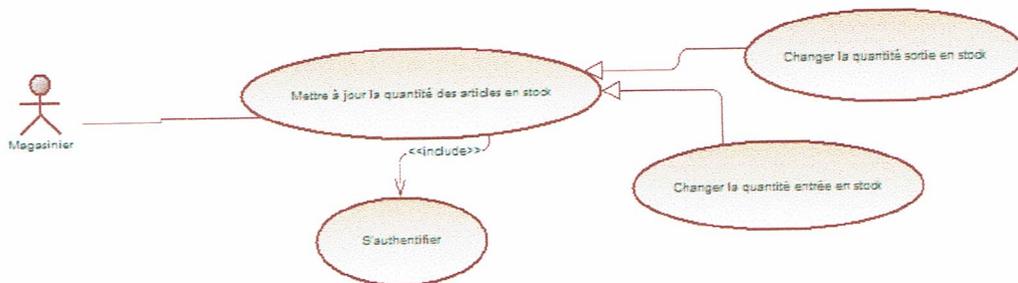


FIGURE 2.4 – Le diagramme de cas d'utilisation «Mettre à jour la quantité des articles en stock»

TABLE 2.4 – Description textuelle du cas d'utilisation «Mettre à jour la quantité des articles en stock»

Cas d'utilisation	Mettre à jour la quantité des articles en stock
Acteur principale	Magasinier
Objectif	Modifier la quantité entrante ou sortante d'un article
Pré condition	Magasinier authentifié
Post condition	Mise à jour gérée de la quantité
Scénario nominal	<p>I. Cas de l'entrée du stock</p> <ol style="list-style-type: none"> 1. Le magasinier scanne le code à barres de l'article du stock à entrer et saisit les autres données . 2. Le magasinier valide la mise à jour. 3. Le système confirme la mise à jour. <p>II. Cas de la sortie du stock</p> <ol style="list-style-type: none"> 1. Le magasinier scanne le code à barres de l'article du stock à sortir et saisit les autres données . 2. Le magasinier valide la mise à jour. 3. Le système confirme la mise à jour.
Scénario alternative	1.1. Champs obligatoire non rempli. Retour à l'étape 1.

2.5.4 Raffinement du cas d'utilisation «Gérer l'inventaire de stock»

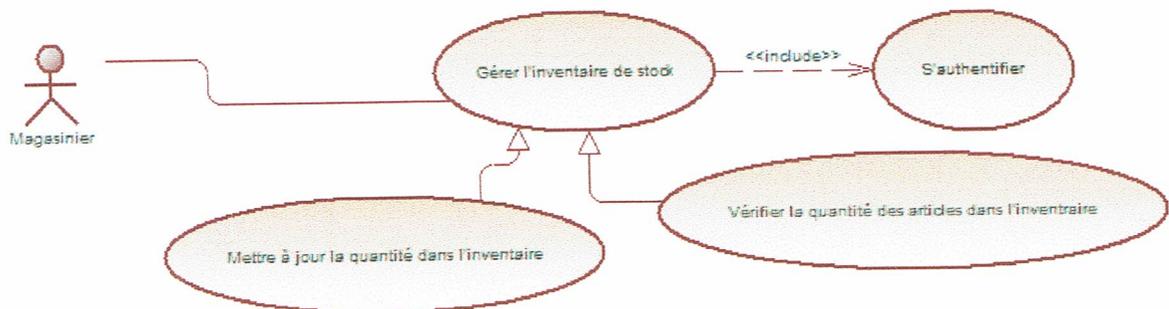


FIGURE 2.5 – Le diagramme de cas d'utilisation «Gérer l'inventaire de stock»

TABLE 2.5 – Description textuelle du cas d'utilisation «Gérer l'inventaire de stock»

Cas d'utilisation	Gérer l'inventaire de stock
Acteur principale	Magasinier
Objectif	Vérifier ou modifier la quantité des articles dans l'inventaire
Pré condition	Magasinier authentifié
Post condition	Mise à jour gérée de la quantité
Scénario nominal	<p>I. Vérifier la quantité</p> <ol style="list-style-type: none"> 1. Le magasinier scanne le code à barres de l'article de l'inventaire . 2. Le magasinier lit la valeur de la quantité de l'article. <p>II. Cas de la modification de la quantité dans l'inventaire</p> <ol style="list-style-type: none"> 1. Le magasinier scanne le code à barres de l'article du stock à sortir et saisit les valeurs à modifier . 2. Le magasinier valide la modification. 3. Le système confirme la modification.
Scénario alternative	1.1. Champs obligatoire non rempli. Retour à l'étape 1.

2.5.5 Raffinement du cas d'utilisation «Enregistrer une fin fabrication»



FIGURE 2.6 – Le diagramme de cas d'utilisation «Enregistrer une fin fabrication»

TABLE 2.6 – Description textuelle du cas d'utilisation «Enregistrer une fin fabrication»

Cas d'utilisation	Enregistrer une fin fabrication
Acteur principale	Responsable production
Objectif	Enregistrer une quantité produite d'articles
Pré condition	Responsable production authentifié
Post condition	Enregistrement de la quantité fabriquée des articles
Scénario nominal	<ol style="list-style-type: none"> 1. Le responsable production saisit les données des articles pour déclarer leur fin de fabrication. 2. Le responsable de production valide l'enregistrement. 3. Le système confirme l'enregistrement.
Scénario alternative	<ol style="list-style-type: none"> 1.1. Champs obligatoire non rempli. Retour à l'étape 1.

2.5.6 Raffinement du cas d'utilisation «Demander article composant»

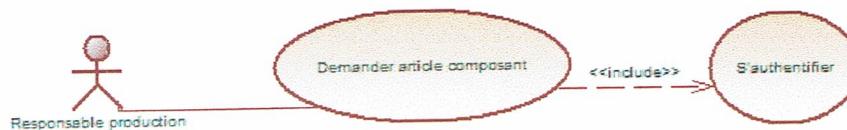


FIGURE 2.7 – Le diagramme de cas d'utilisation «Demander article composant»

TABLE 2.7 – Description textuelle du cas d'utilisation «Demander article composant»

Cas d'utilisation	Demander article composant
Acteur principale	Responsable production
Objectif	Sortir une quantité d'article pour une fabrication
Pré condition	Responsable production authentifié
Post condition	Enregistrement de la quantité nécessaire
Scénario nominal	<ol style="list-style-type: none"> 1. Le responsable production saisit les données des articles demandés pour la fabrication. 2. Le responsable de production valide l'enregistrement. 3. Le système confirme l'enregistrement.
Scénario alternative	<ol style="list-style-type: none"> 1.1. Champs obligatoire non rempli. Retour à l'étape 1. 2.1 La quantité demandée est supérieure à la quantité disponible de l'article Retour à l'étape 1

2.5.7 Raffinement du cas d'utilisation «Gérer les bons de livraison»

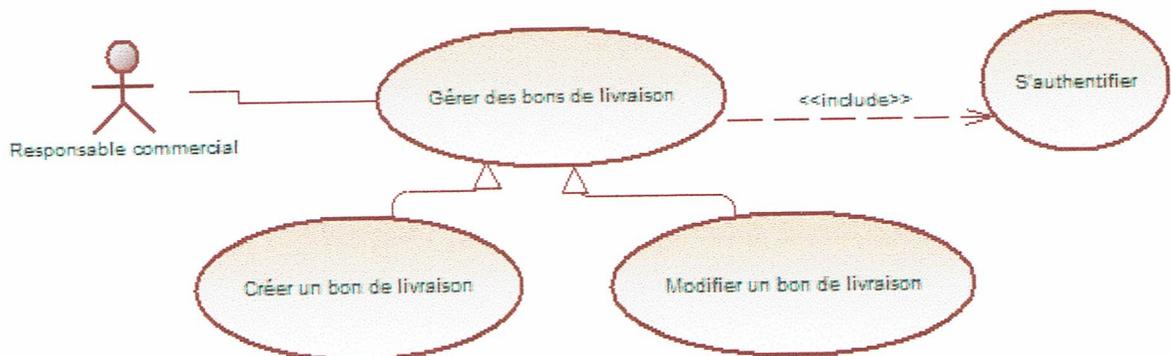


FIGURE 2.8 – Le diagramme de cas d'utilisation «Gérer les bons de livraison»

TABLE 2.8 – Description textuelle du cas d'utilisation «Gérer les bons de livraison»

Cas d'utilisation	Gérer les bons de livraison
Acteur principale	Responsable commercial
Objectif	Créer ou modifier des bons de livraison
Pré condition	Responsable commercial authentifié
Post condition	Bon de livraison créé ou modifié
Scénario nominal	<p>I. Cas de la Création d'un bon de livraison</p> <ol style="list-style-type: none"> 1. Le responsable commercial introduit les données de l'article à livrer . 2. Le responsable commercial valide la création du bon de livraison. 3. Le système confirme la création du bon de livraison. <p>II. Cas de la modification du bon de livraison</p> <ol style="list-style-type: none"> 1. Le responsable commercial saisit les données du bon de livraison à modifier . 2. Le responsable commercial valide la modification. 3. Le système confirme la modification.
Scénario alternative	<p>1.1. Champs obligatoire non rempli. Retour à l'étape 1.</p> <p>2.1 La quantité saisie à livrée est supérieure à la quantité disponible Retour à l'étape 1.</p>

2.6 Diagramme des cas d'utilisation détaillé

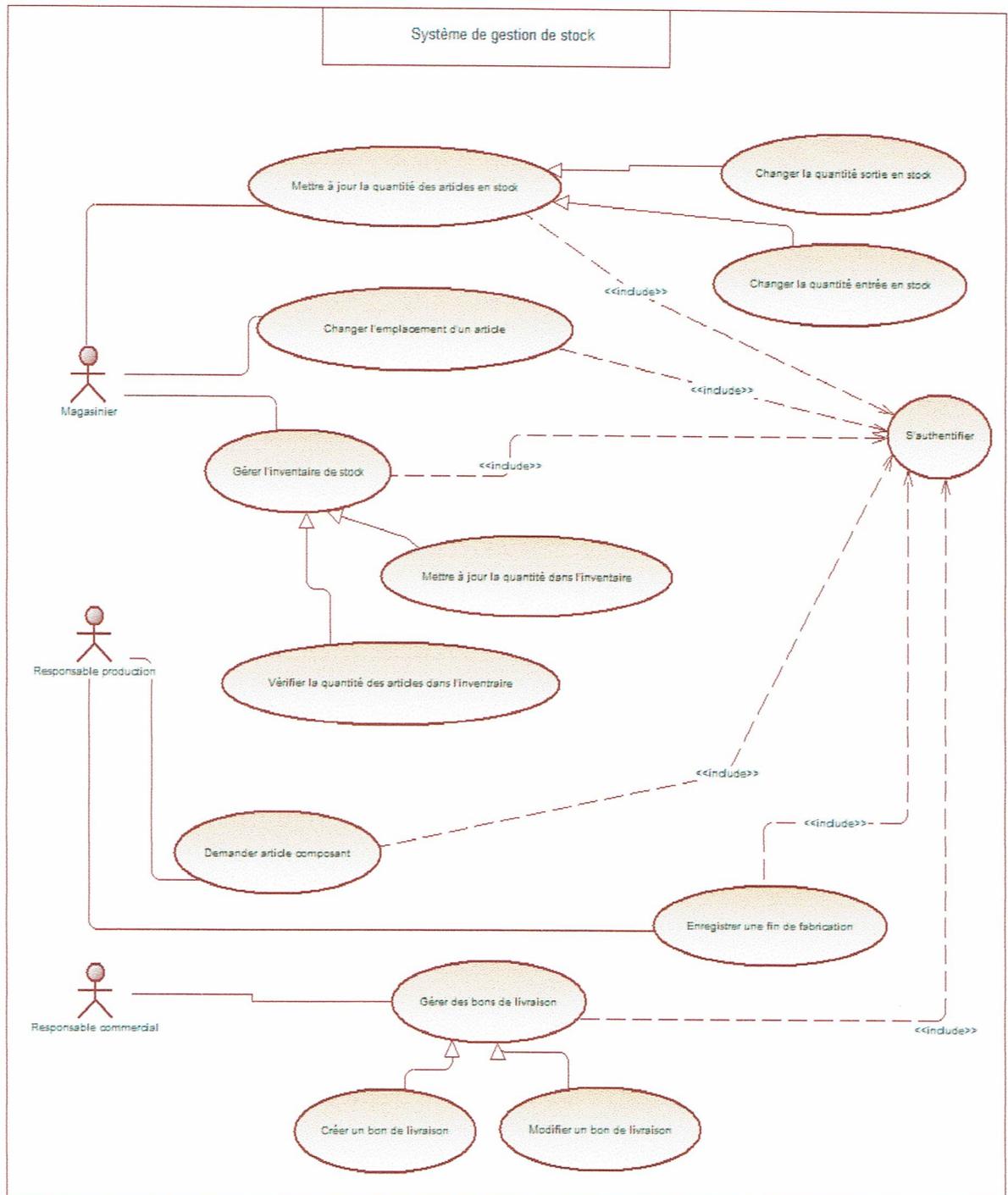


FIGURE 2.9 – Le diagramme de cas d'utilisation général

2.7 Diagramme de séquence système

2.7.1 Définition

C'est un diagramme d'interaction qui représente les objets participant à une interaction particulière et les messages qu'ils échangent organisé en séquences horaires. Axé sur ce que fait un système et non sur la manière dont il le fait, un diagramme de séquence définit la logique d'une instance particulière d'un cas d'utilisation.

2.7.2 Diagramme de séquence système relatif à «S'authentifier»

Dans la figure ci-dessous, nous avons illustré le diagramme de séquence système relatif au cas d'utilisation «S'authentifier»

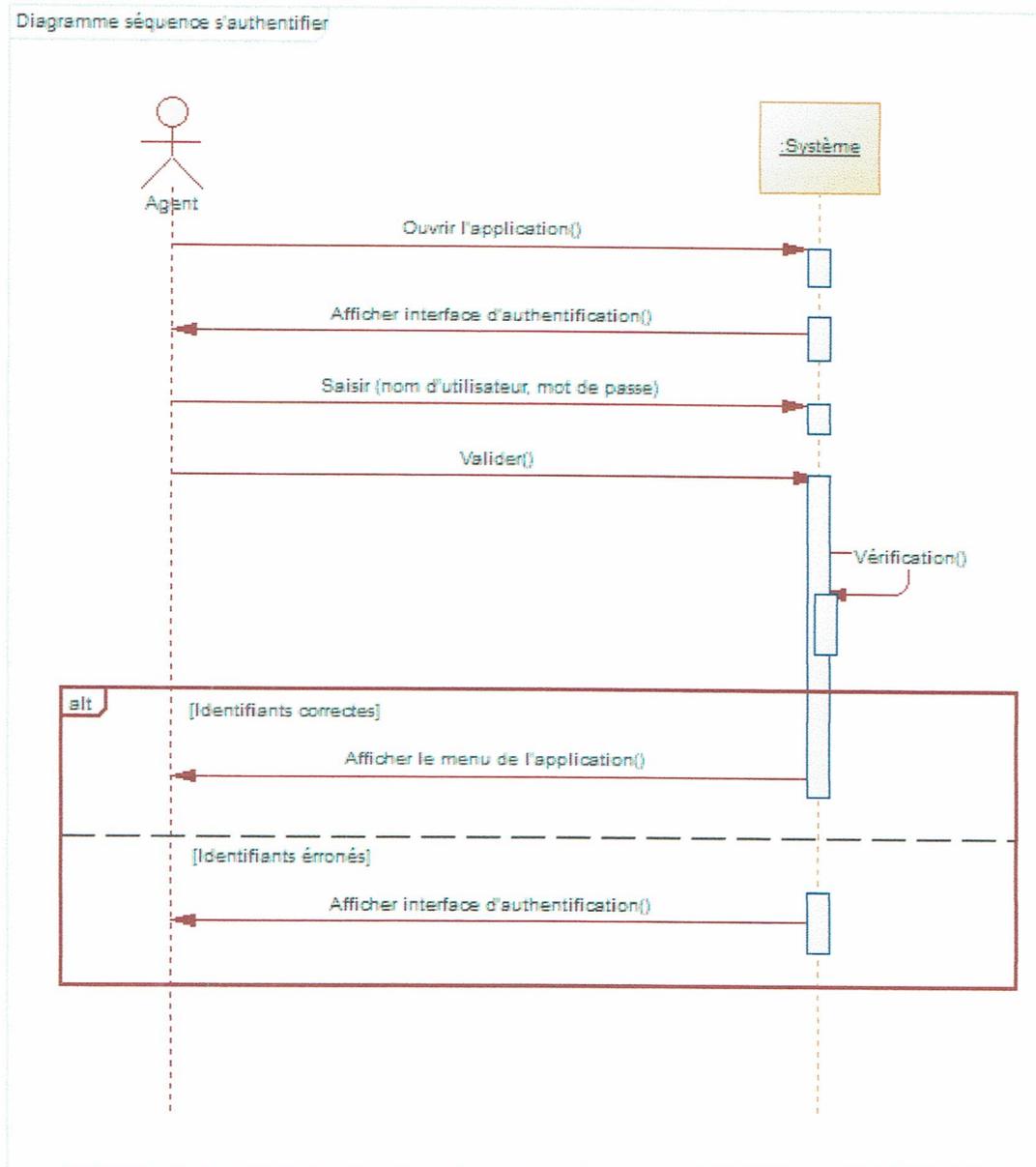


FIGURE 2.10 – Le diagramme de séquence système «S'authentifier»

2.7.3 Diagramme de séquence système relatif à «Changer l'emplacement d'un article»

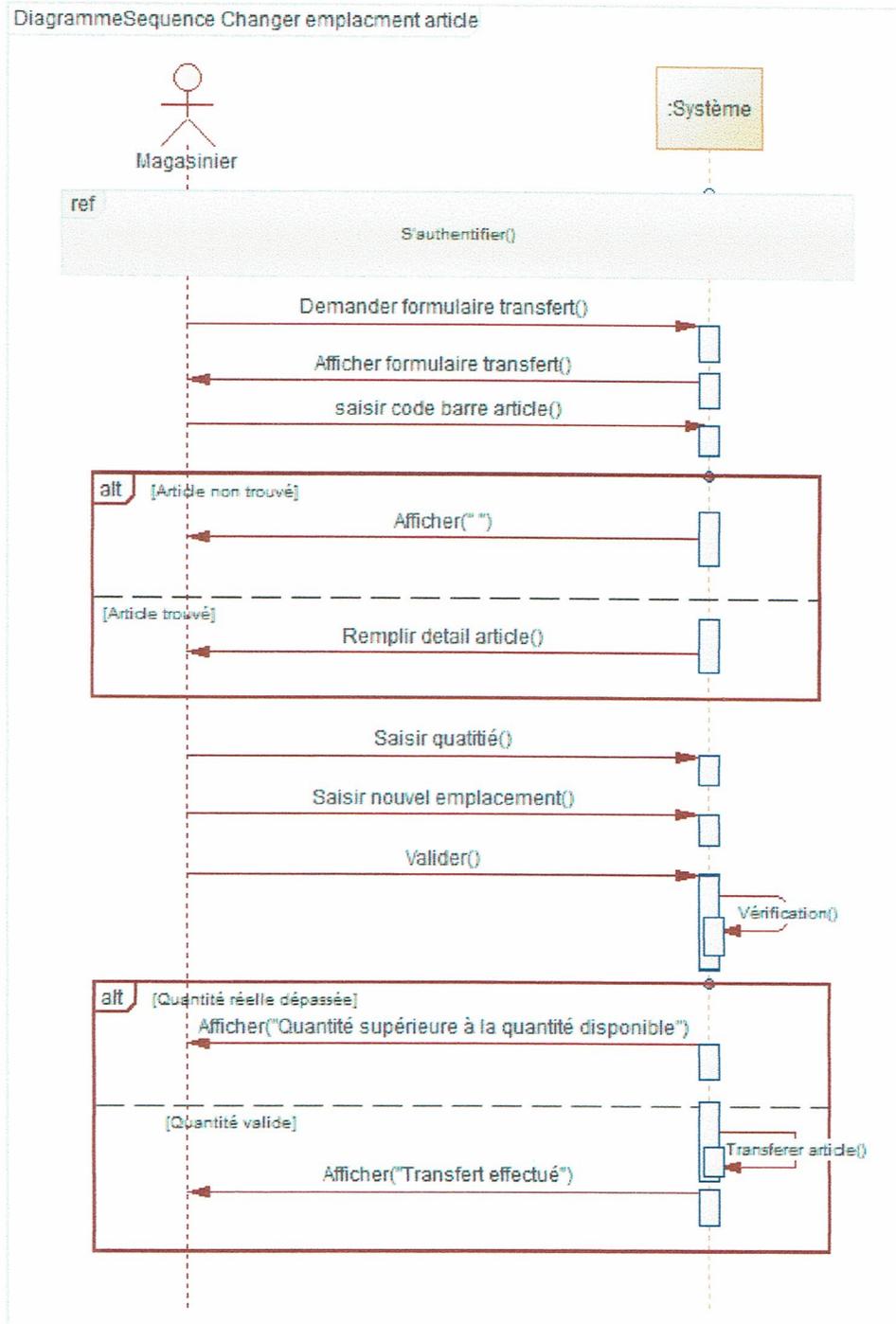


FIGURE 2.11 – Le diagramme de séquence système «Changer l'emplacement d'un article»

2.7.4 Diagramme de séquence système relatif à «Changer quantité sortie en stock»

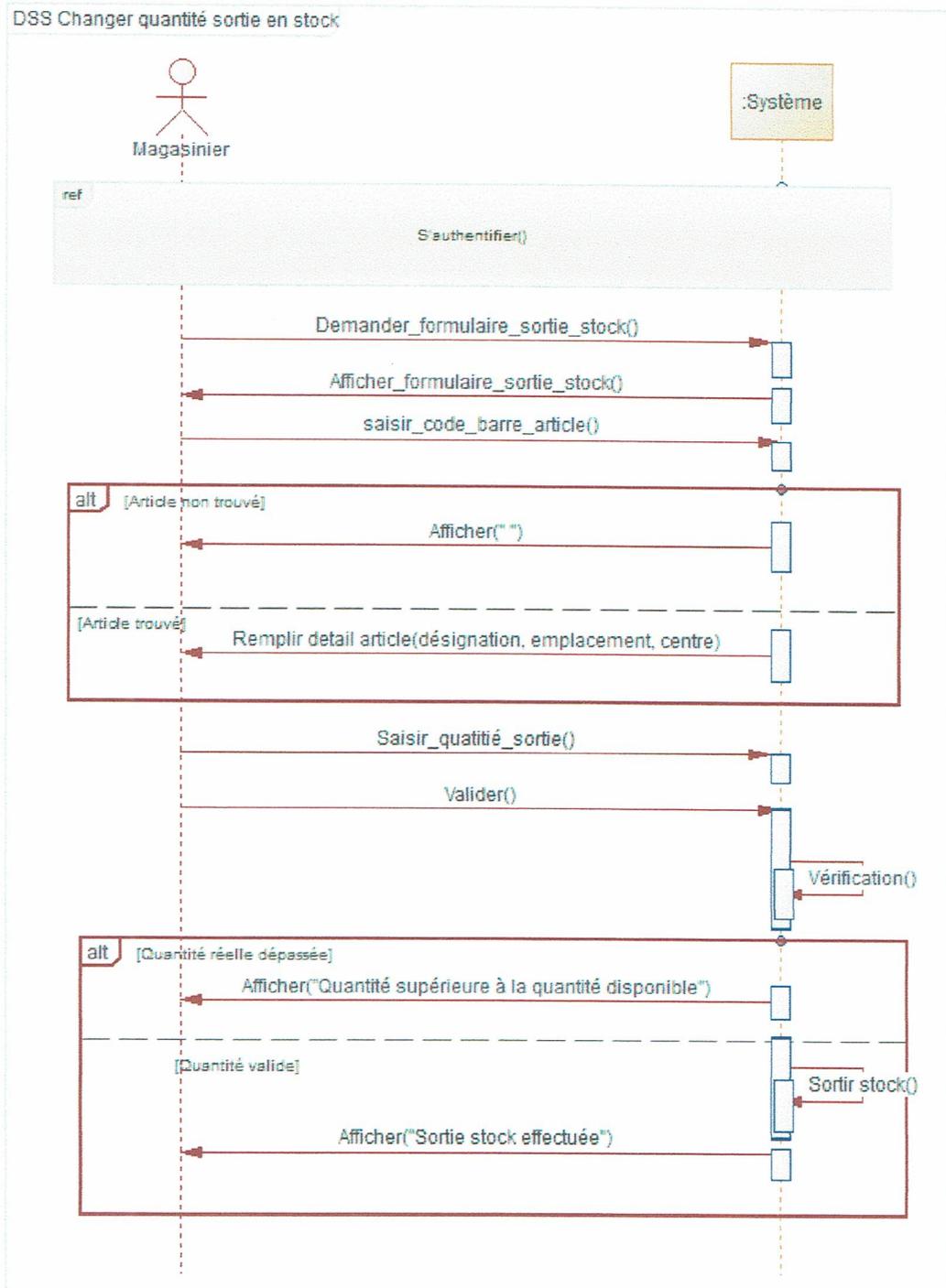


FIGURE 2.12 – Le diagramme de séquence système «Changer quantité sortie en stock»

2.7.5 Diagramme de séquence système relatif à «Changer quantité entrée en stock»

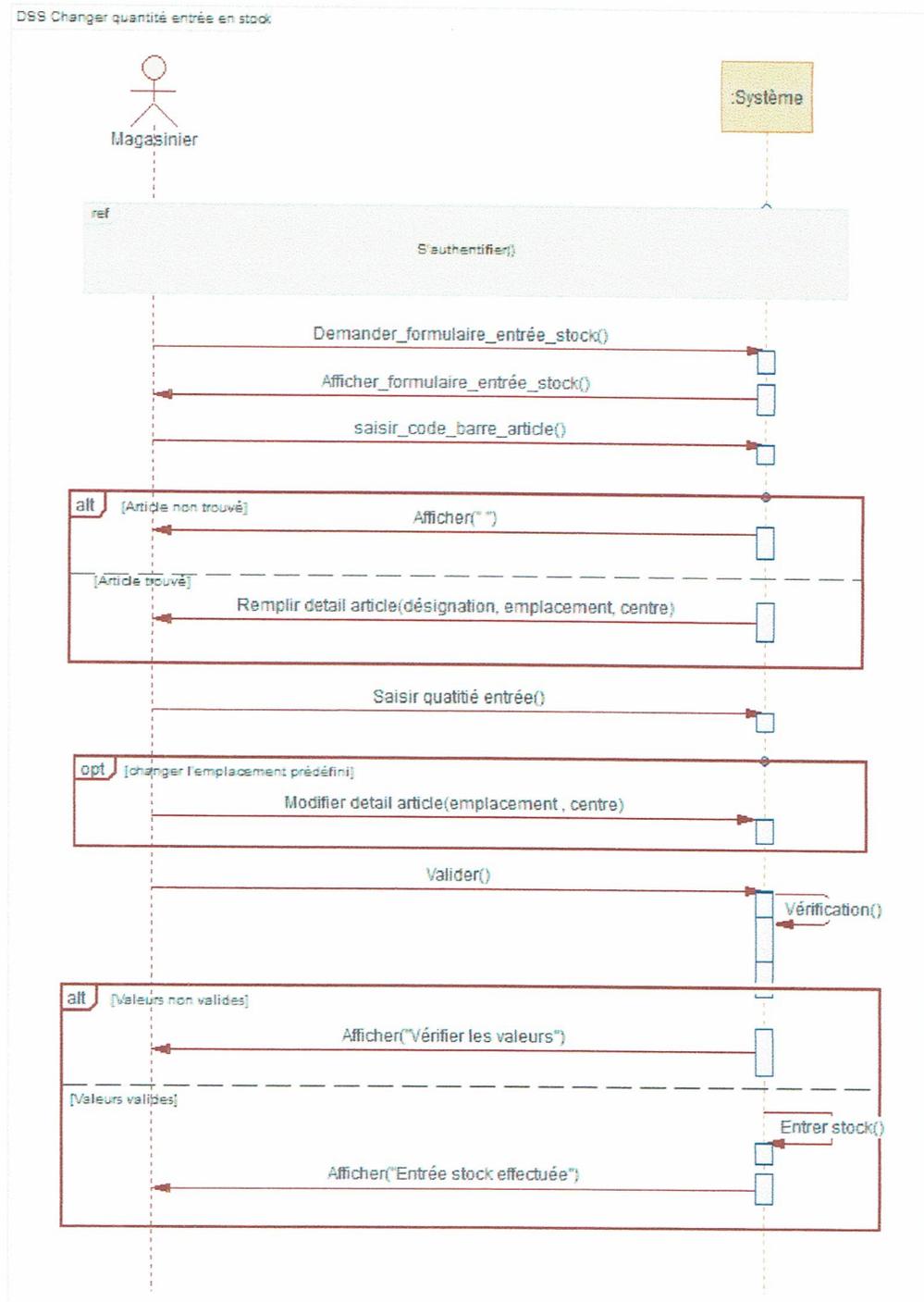


FIGURE 2.13 – Le diagramme de séquence système «Changer quantité entrée en stock»

2.7.6 Diagramme de séquence système relatif à «Mettre à jour la quantité dans l'inventaire»

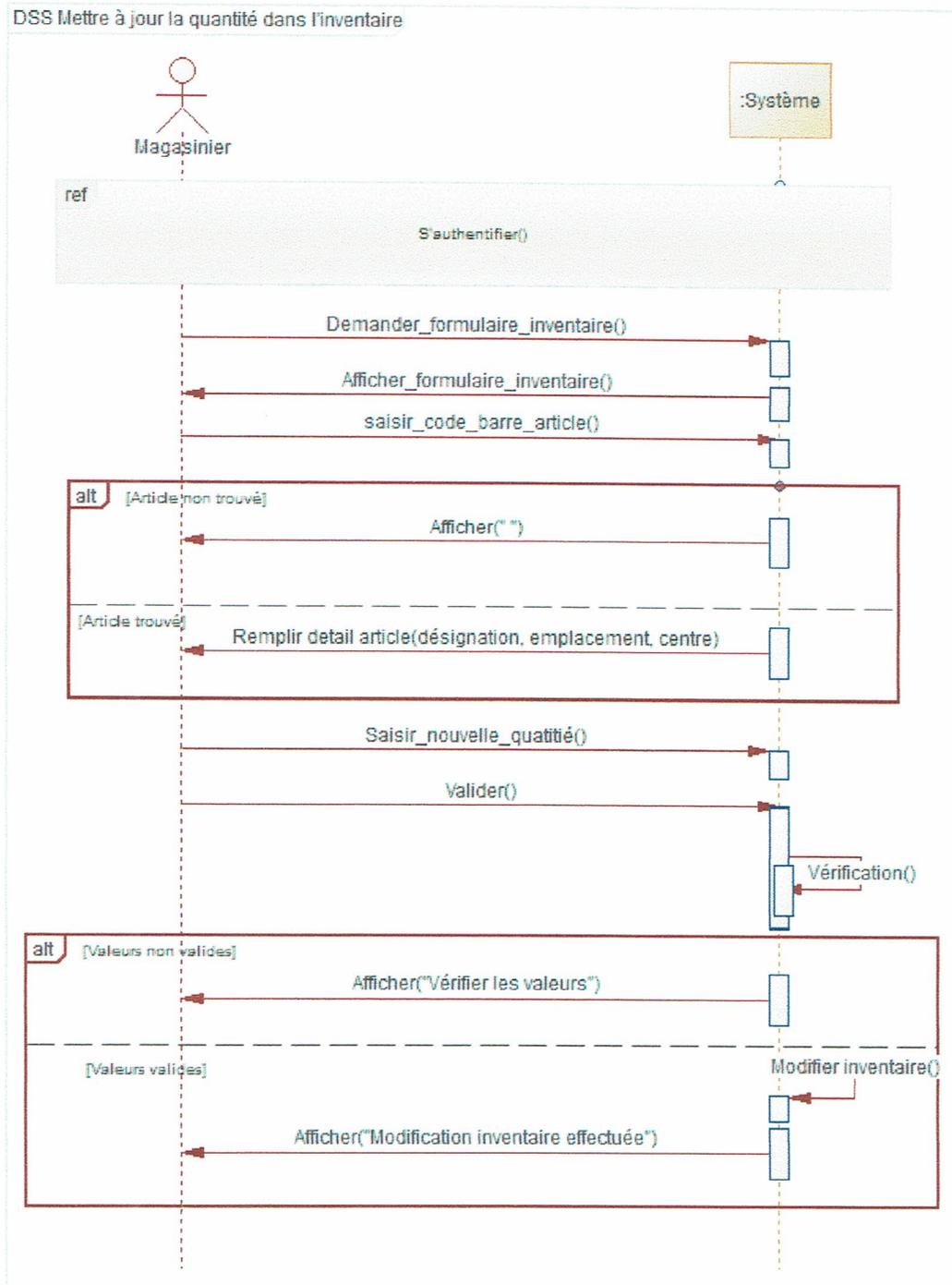


FIGURE 2.14 – Le diagramme de séquence système «Mettre à jour la quantité dans l'inventaire»

2.7.7 Diagramme de séquence système relatif à «Vérifier la quantité dans l'inventaire»

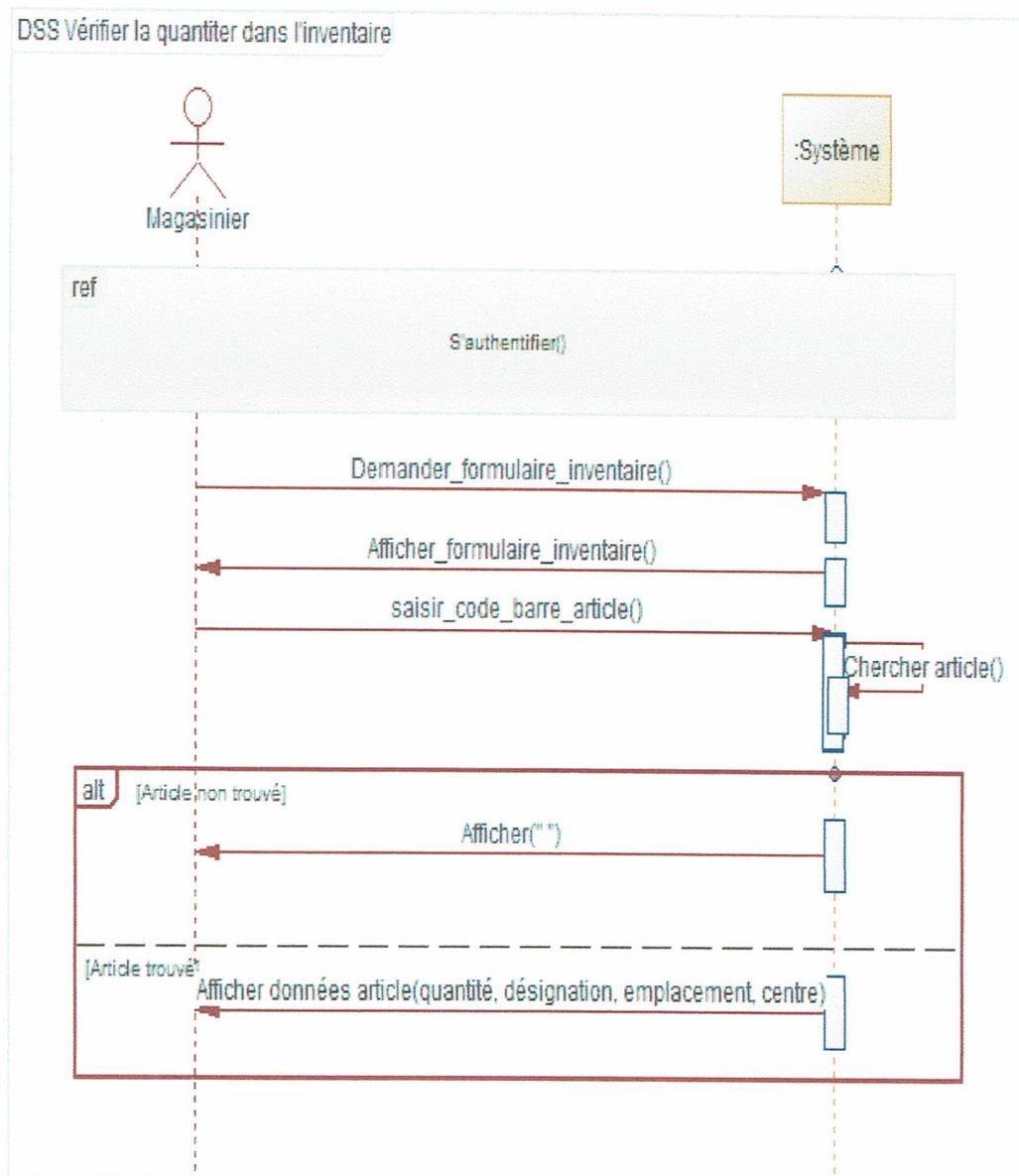


FIGURE 2.15 – Le diagramme de séquence système «Vérifier la quantité dans l'inventaire»

2.7.8 Diagramme de séquence système relatif à «Enregistrer une fin de fabrication»

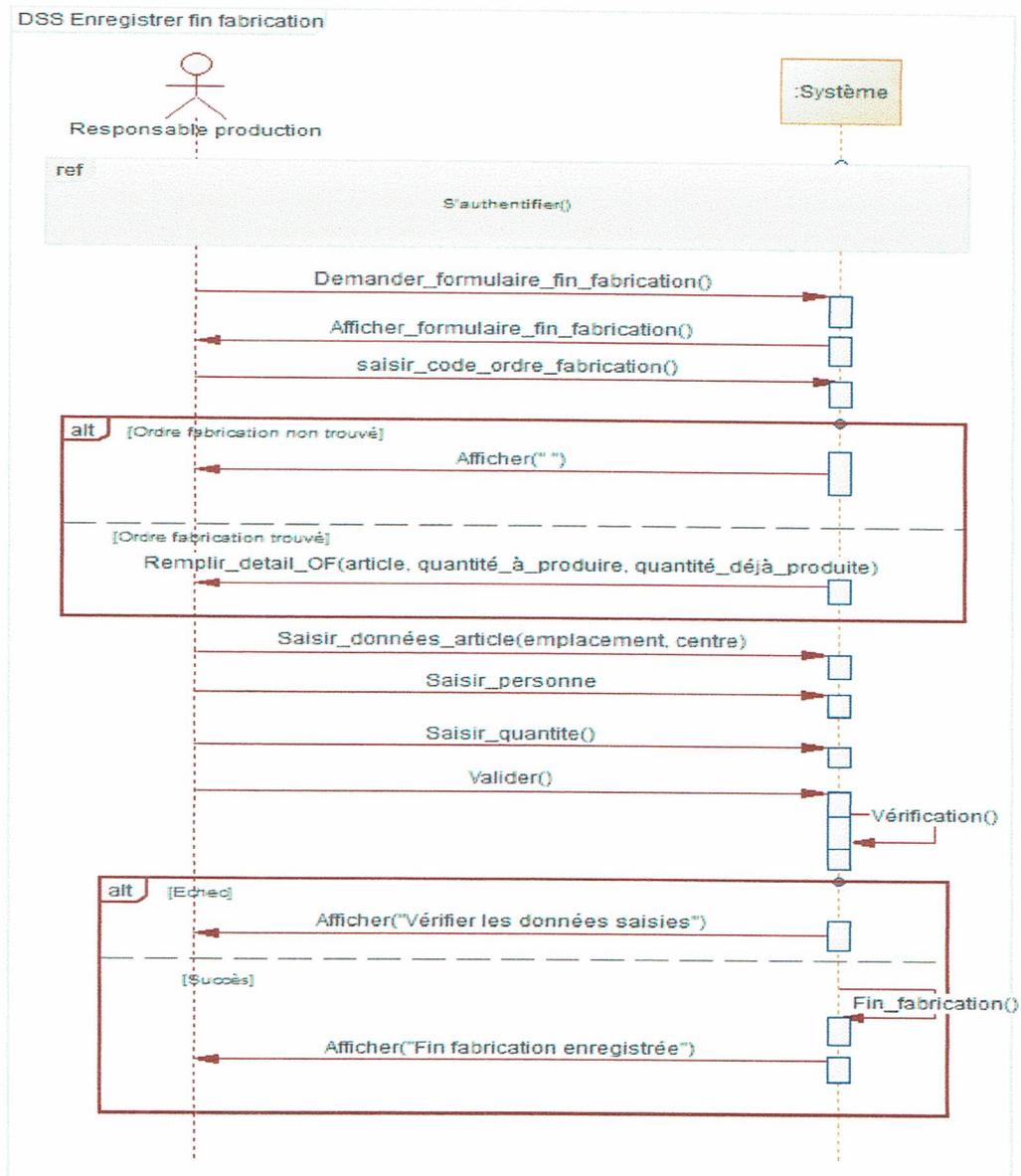


FIGURE 2.16 – Le diagramme de séquence système «Vérifier la quantité dans l'inventaire»

2.7.9 Diagramme de séquence système relatif à «Demander article composant»

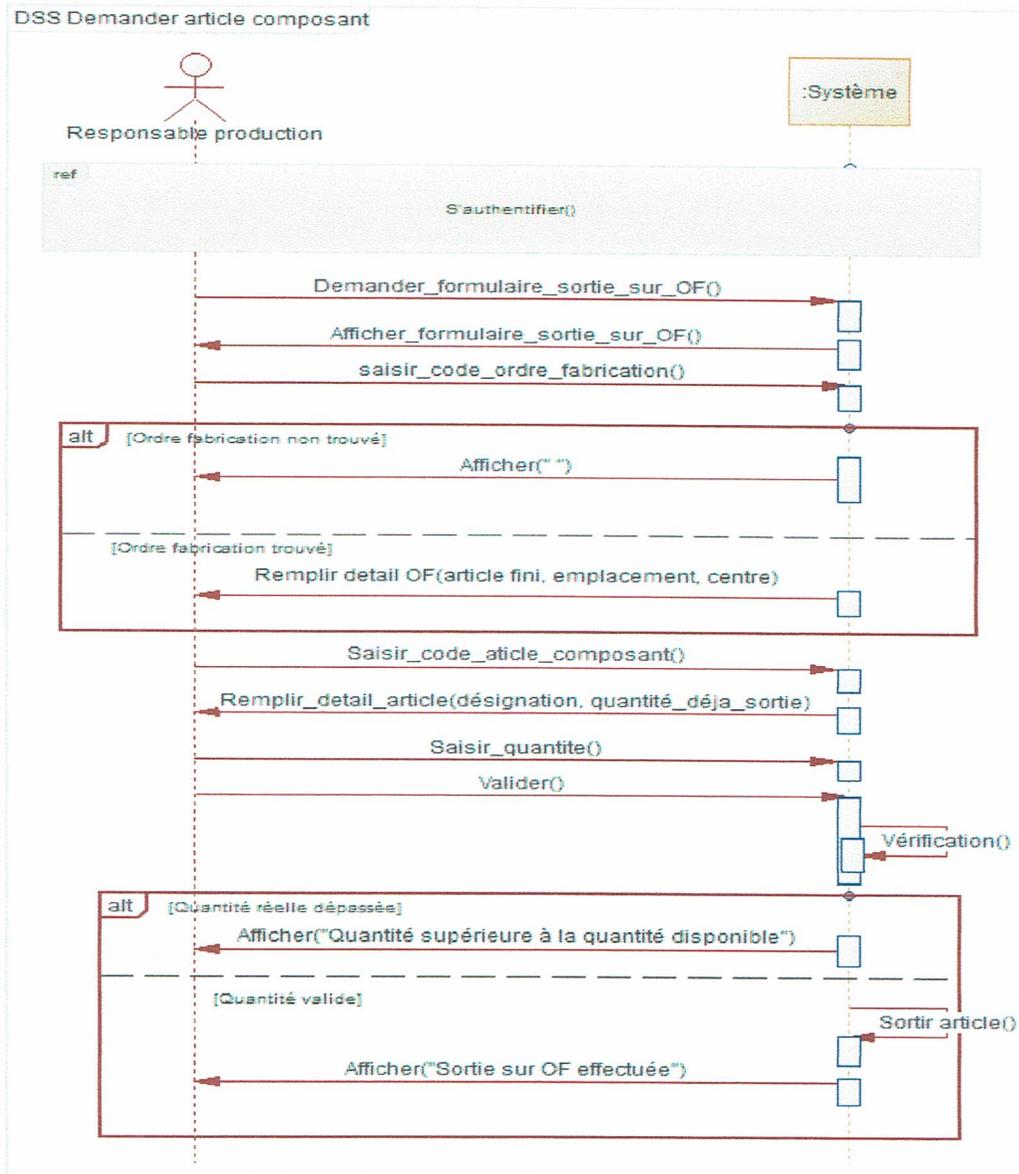


FIGURE 2.17 – Le diagramme de séquence système «Demander article composant»

2.7.10 Diagramme de séquence système relatif à «Créer un bon de livraison»

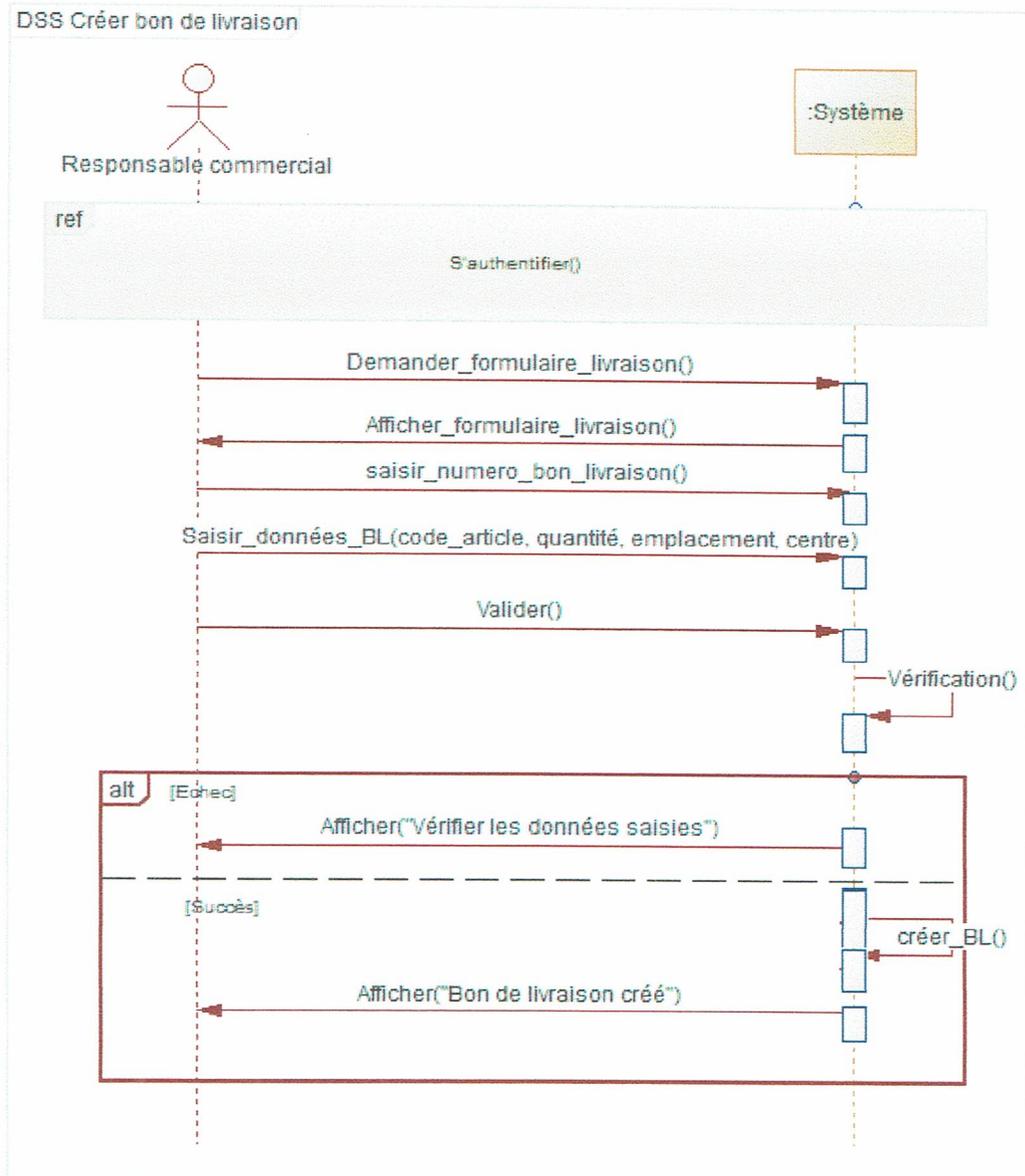


FIGURE 2.18 – Le diagramme de séquence système «Créer un bon de livraison»

2.7.11 Diagramme de séquence système relatif à «Modifier un bon de livraison»

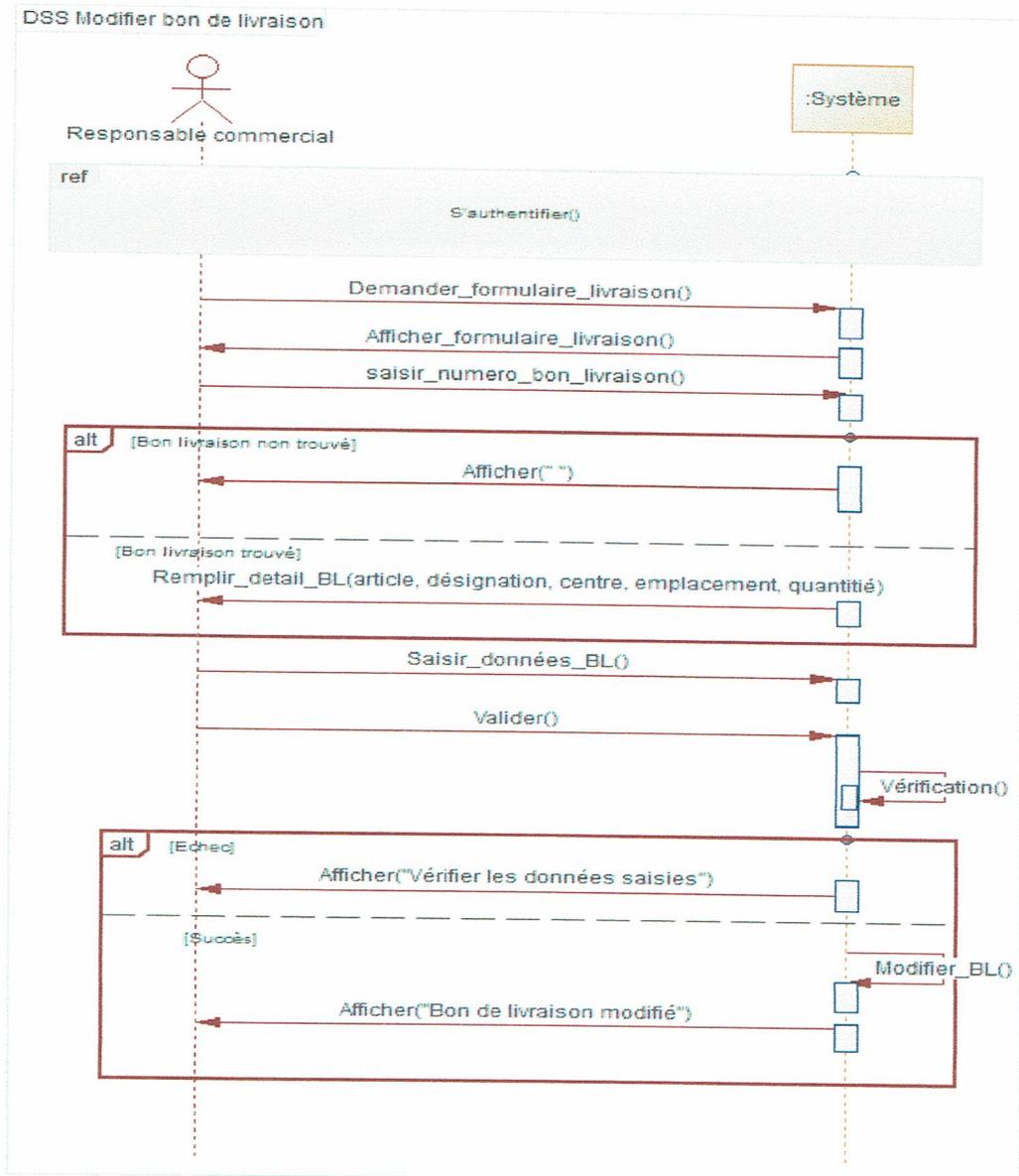


FIGURE 2.19 – Le diagramme de séquence système «Modifier un bon de livraison»

2.8 Conclusion

Dans ce chapitre, nous avons réalisé la spécification des besoins fonctionnels que le système doit les satisfaire. Nous avons aussi présenté avec détails les diagrammes de cas d'utilisation. Par la suite, nous avons effectué l'analyse de ces cas d'utilisation à travers les descriptions textuelles ainsi que les diagrammes de séquencesystème.

Chapitre 3

Conception de la solution

Sommaire

3.1	Introduction	38
3.2	Diagramme de séquence.....	39
3.2.1	Diagramme de séquence «S`authentifier».....	40
3.2.2	Diagramme de séquence «Changer l`emplacement d`un ar- ticle»..	41
3.2.3	Diagramme de séquence «Mettre à jour la quantité des articles en stock»	42
3.2.4	Diagramme de séquence «Enregistrer une fin de fabrication»	43
3.3	Modèle relationnel.....	44
3.4	Architecture de l`application	44
3.4.1	Architecture matérielle	44
3.4.2	Architecture logicielle	47
3.5	Conclusion.....	49

3.1 Introduction

Avant de commencer la réalisation de notre application, il faut d`abord le conce- voir pour bien expliquer le fonctionnement des nouveaux besoins. Pour cette raison, nous consacrons ce chapitre pour la conception de différentes parties de notre projet.

Ce chapitre commence par la présentation du diagramme de séquence. Par la suite nous allons schématiser les tables utilisées de la base de données. Enfin, nous

allons présenter l'architecture matérielle et logicielle de l'application web ainsi que de la version mobile.

3.2 Diagramme de séquence

Nous présentons dans cette partie les diagrammes de séquence de notre application qui sert à modéliser les interactions entre les objets situés dans un contexte temporel (décrire les scénarios de chaque cas d'utilisation en mettant l'accent sur la chronologie des opérations en interaction avec les objets).

3.2.1 Diagramme de séquence «S'authentifier»

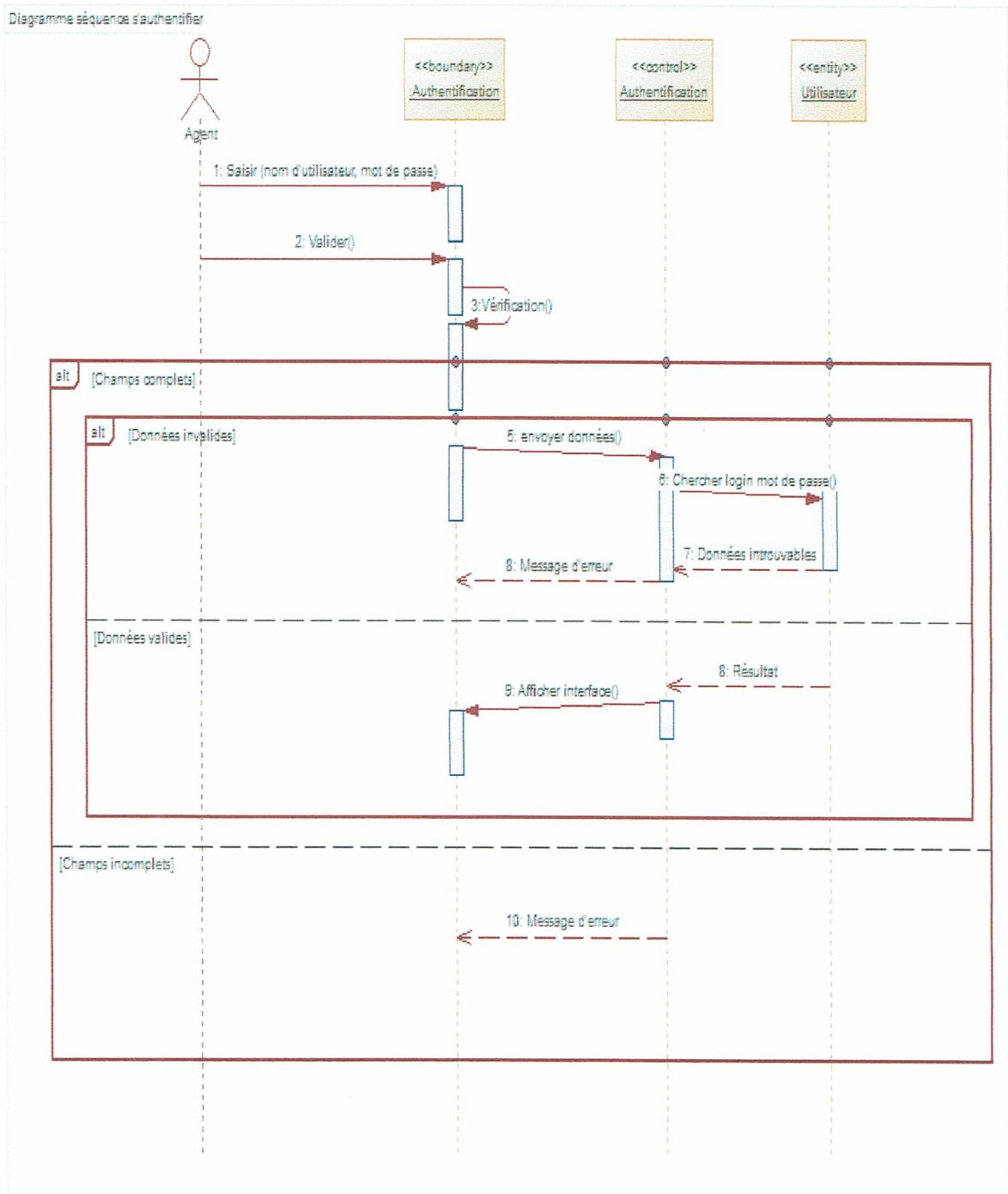


FIGURE 3.1 – Le diagramme de séquence «S'authentifier»

3.2.2 Diagramme de séquence «Changer l'emplacement d'un article»

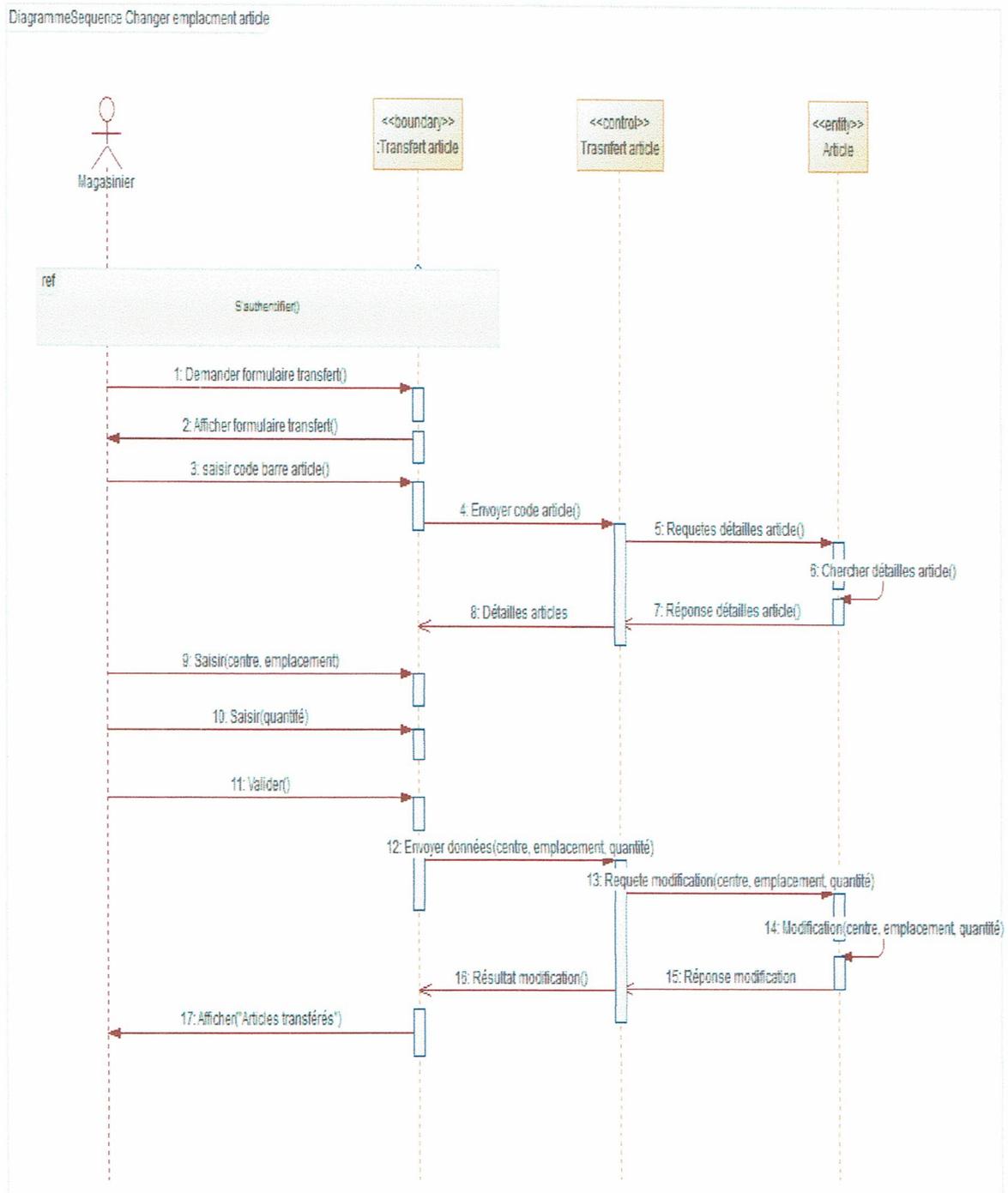


FIGURE 3.2 – Le diagramme de séquence «Changer l'emplacement d'un article»

3.2.3 Diagramme de séquence «Mettre à jour la quantité des articles en stock»

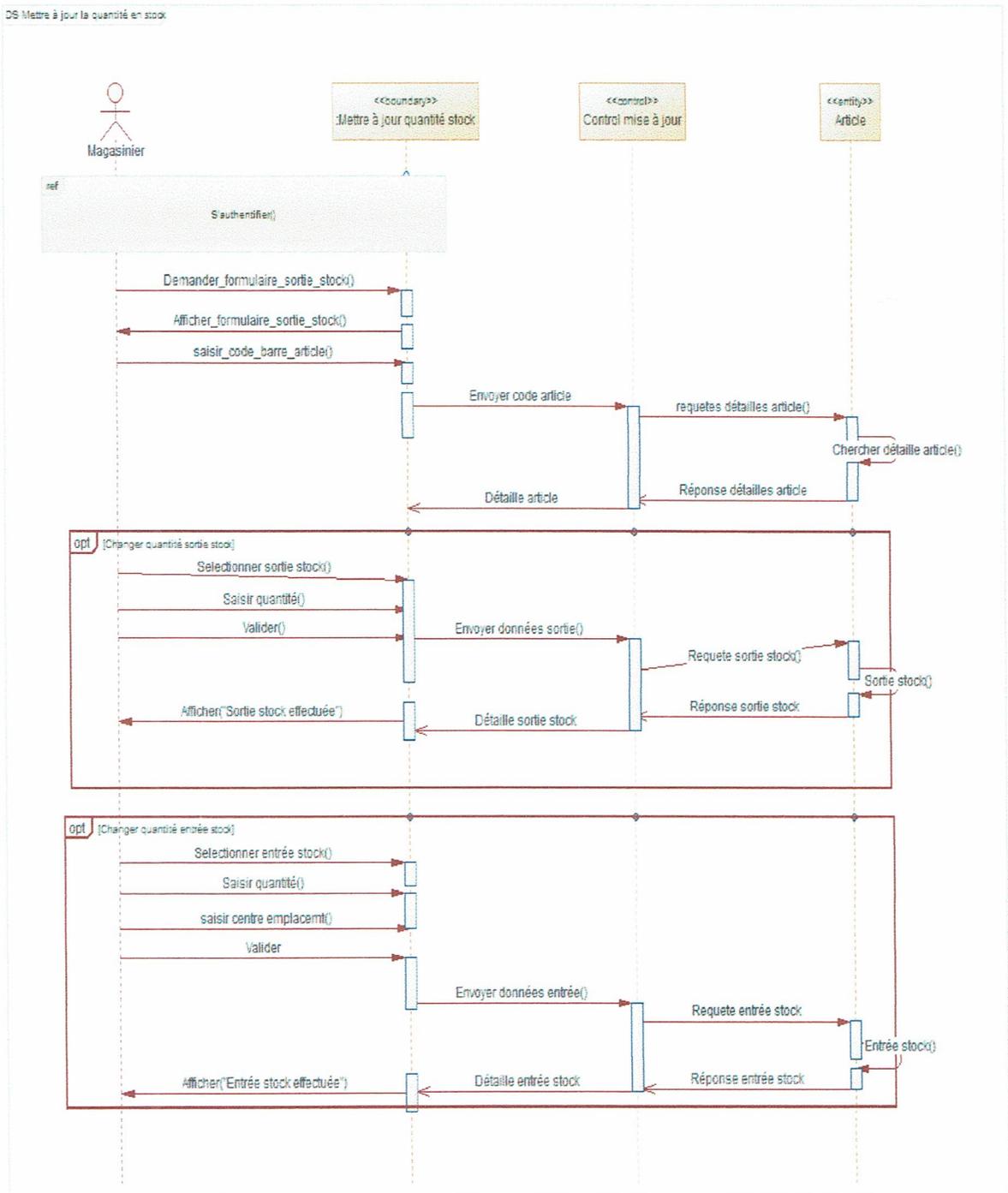


FIGURE 3.3 – Le diagramme de séquence «Mettre à jour la quantité des articles en stock»

3.2.4 Diagramme de séquence «Enregistrer une fin de fabrication»

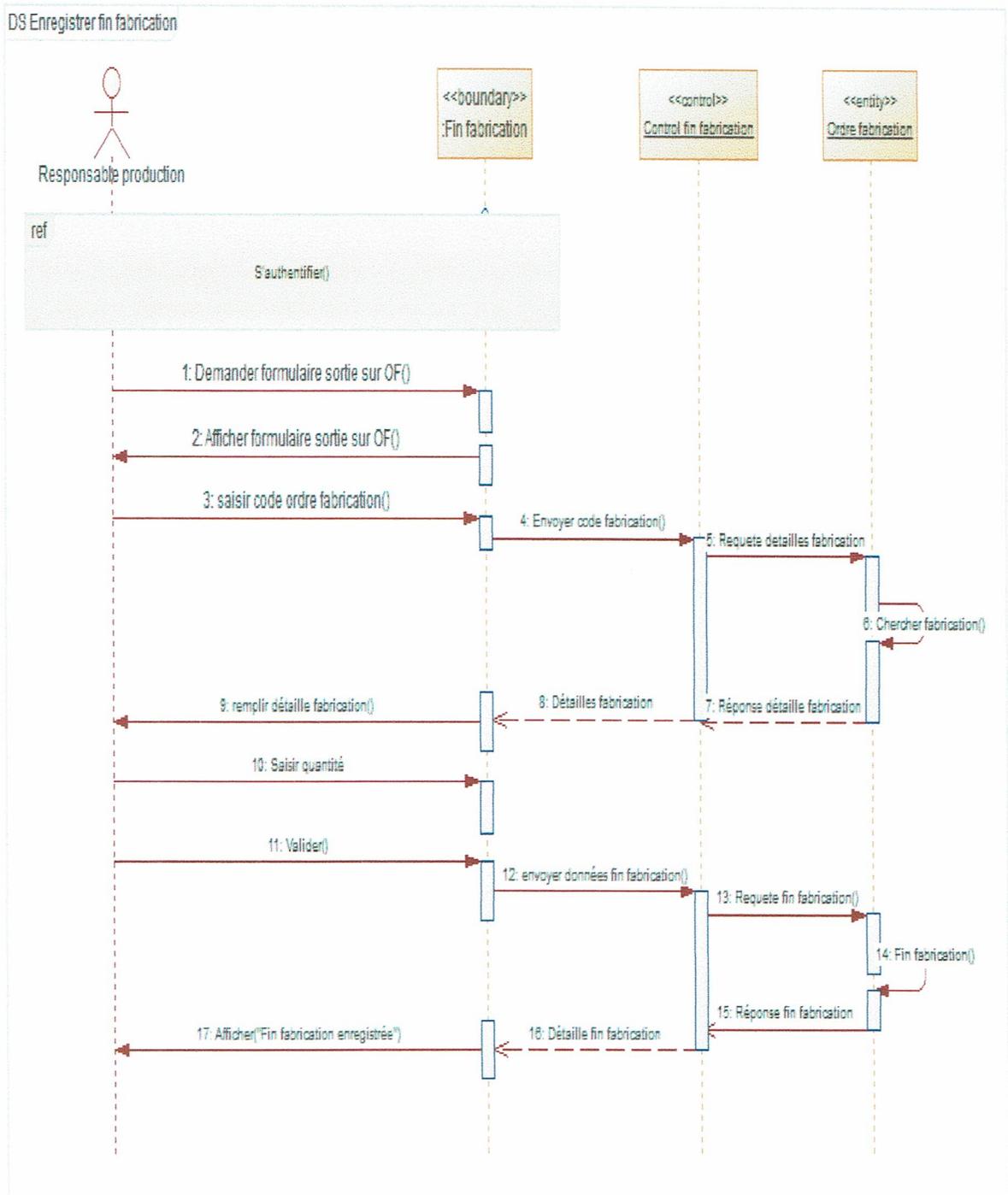


FIGURE 3.4 – Le diagramme de séquence «Enregistrer une fin de fabrication»

3.3 Modèle relationnel

La figure ci-dessous représente les tables utilisées dans notre base des données. Ces tables représentent l'unité de persistance de notre application.

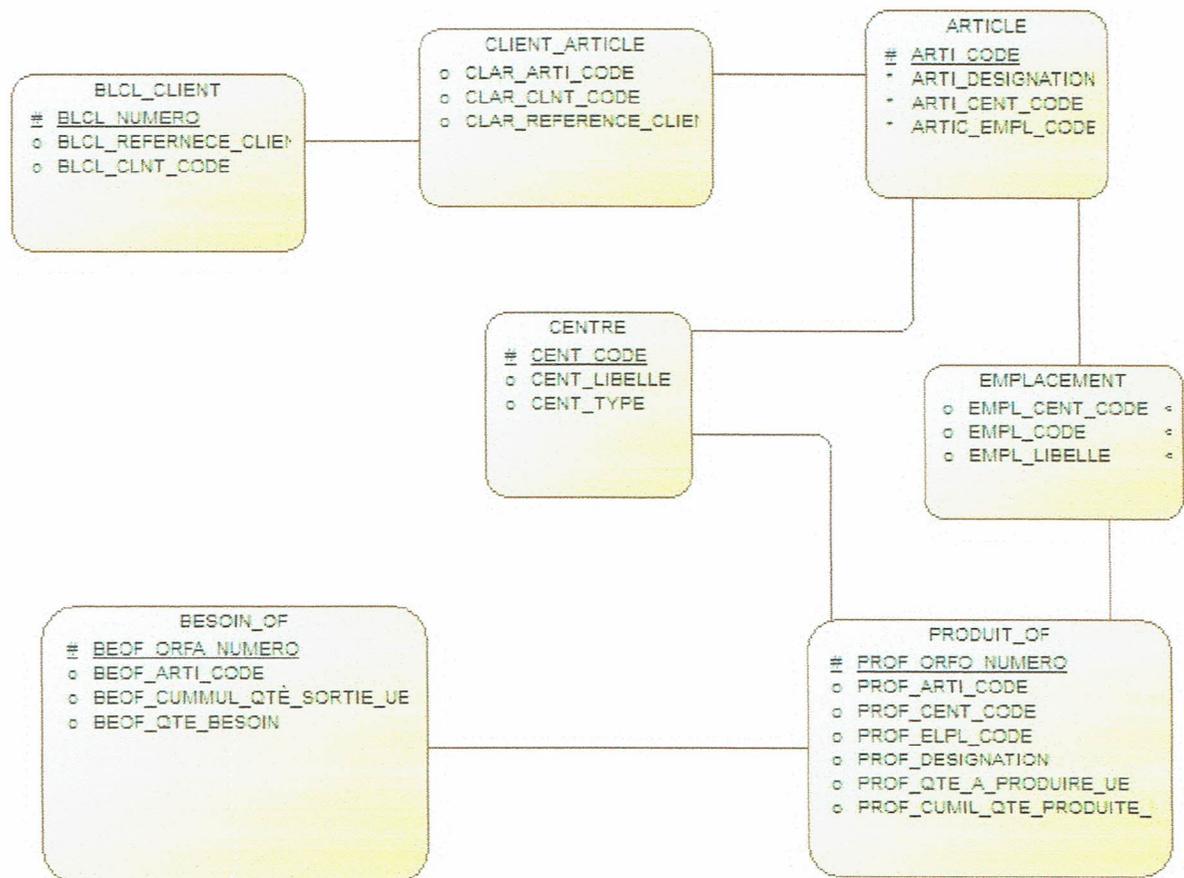


FIGURE 3.5 – Schéma de la base de données

3.4 Architecture de l'application

3.4.1 Architecture matérielle

Pour l'application web :

La machine de l'utilisateur est le lecteur code à barres mobile qui est lié par Wi-Fi

avec le serveur de la société qui communique avec la base de données.

Exemple de lecteur code à barres mobile :
Motorola MC40-HC, qui est caractérisé par un écran tactile de 4.3", un processeur dual-core 800Mhz et 1 Gb de mémoire ram. Ce lecteur marche sous Android 4.1 .



FIGURE 3.1 de à
barres Motorola MC40-HC

L'utilisateur peut aussi lancer l'application web à travers un navigateur web sous différents types dispositifs comme un smart-phone, une tablette ou un ordinateur, grâce au design adaptatif qui permet d'afficher l'interface de l'application sur différentes résolutions d'écran.

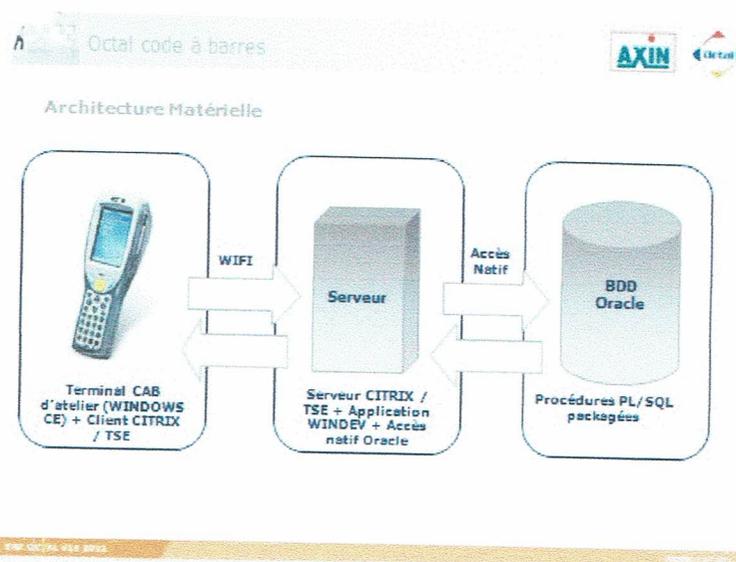


FIGURE 3.7 – Architecture matérielle de l'application web

Pour l'application mobile :

la machine doit exécuter un système d'exploitation Windows Phone (la version minimum est Windows Phone 8). Le smartphone est connecté au serveur de la société à travers la technologie Wi-Fi. Le serveur de la société contient un serveur web pour être accessible à distance et un serveur de base de données qui contient les tables.

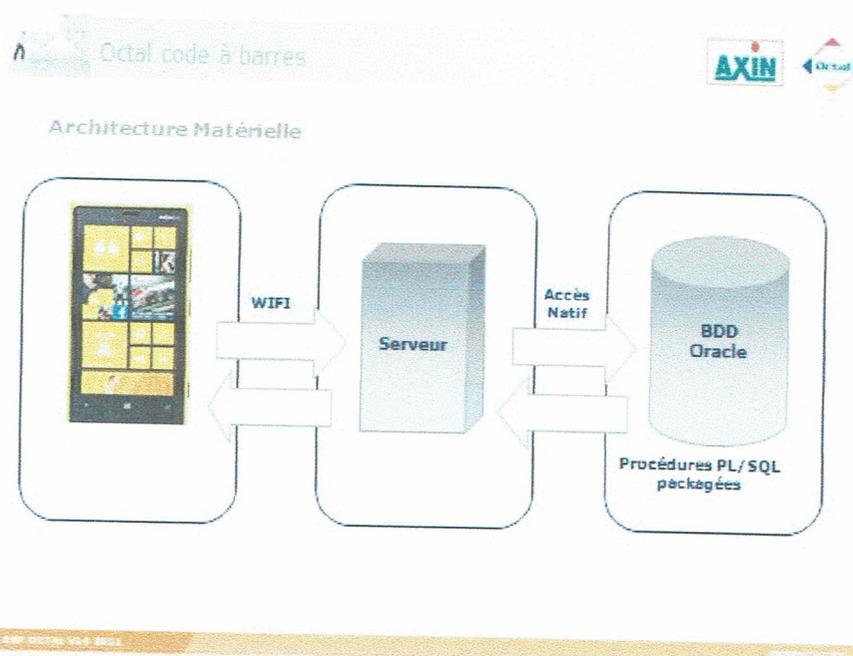


FIGURE 3.8 – Architecture matérielle de l'application mobile

3.4.2 Architecture logicielle

Pour l'application web :

Les interfaces des pages web sont réalisées par les technologies HTML, CSS et JavaScript. La communication entre le serveur et les pages web se fait à travers le langage PHP qui est un langage de script exécuté du côté serveur. Enfin pour l'envoi des requêtes et la réception des résultats, une extension OCI (Oracle Call Interface) est nécessaire.

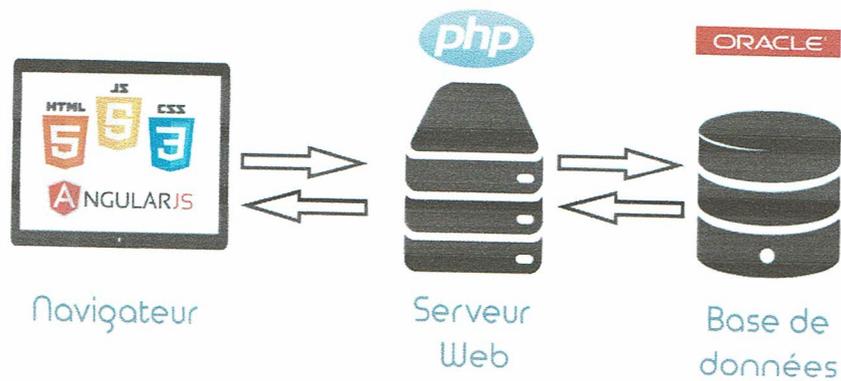


FIGURE 3.9 – Architecture logicielle de l'application mobile

Pour l'application mobile :

Les pages et le traitement de l'application Windows Phone sont créés avec le langage Extensible Application Markup Language (XAML) avec C# qui communiquent à l'aide des services web avec le serveur et la base de données.

La figure ci-dessous représente l'architecture d'application mobile réalisée.

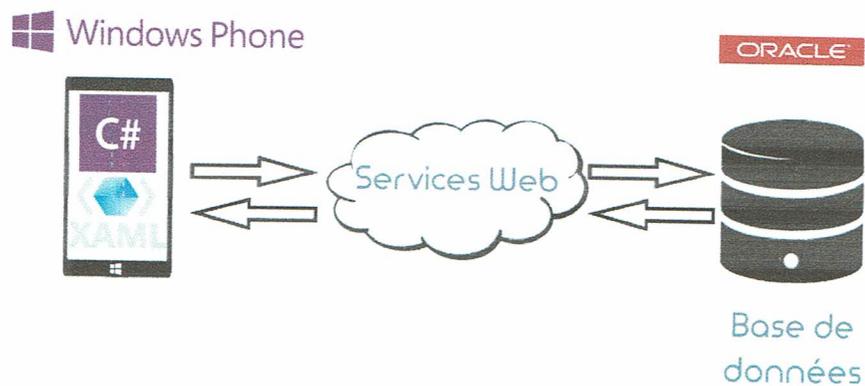


FIGURE 3.10 – Architecture logicielle de l'application mobile

3.5 Conclusion

Dans ce chapitre, nous avons accompli l'analyse des cas d'utilisation. Ensuite nous avons élaboré les diagrammes de séquence détaillés de chaque cas d'utilisation et le schéma de la base de données .

Dans le chapitre suivant, nous allons entamer la partie réalisation du projet.

Chapitre 4

Réalisation

Sommaire

4.1	Introduction	50
4.2	Environnement de travail	51
4.2.1	Environnement matériel	51
4.2.2	Les outils de travail	51
4.3	Principales interfaces graphiques	59
4.3.1	Principales interfaces de l'application web	59
4.3.2	Principales interfaces de l'application mobile	68
4.4	Chronogramme de travail	72
4.5	Problèmes rencontrés	73
4.5.1	Installation de la base de donnée	73
4.5.2	Problème avec l'émulateur de Windows Phone	74
4.6	Conclusion	75

4.1 Introduction

Tout au long de ce chapitre, nous mettrons en relief les détails techniques spécifiques à la réalisation du projet. Nous présenterons tout d'abord l'environnement matériel et logiciel de la réalisation de notre solution.

Enfin, nous finirons par présenter quelques interfaces de l'application.

4.2 Environnement de travail

4.2.1 Environnement matériel

Pour le développement, nous avons utilisé un ordinateur portable doté par les caractéristiques suivantes :

- Processeur : Intel (R) Core TM i5-3230M CPU @ 2.60 GHz 2.50 GHz
- Ram : 4.00 GB
- Disque Dur : 500 Go.

Pour la compilation de l'application mobile, nous avons utilisé un Smartphone Nokia Lumia 520 qui est caractérisé par un processeur 1 Ghz dual core, 512MB de Ram et qui marche sous Microsoft Windows Phone 8.

4.2.2 Les outils de travail

Pour réaliser notre travail, nous avons installé le système de gestion de base de données (SGBD) Oracle 10g, puis nous avons installé TOAD for Oracle pour qu'on puisse bien visualiser les schémas des tables de la base de données.

L'ensemble des logiciels et langages qui nous ont servi tout au long du travail à savoir :

- **Oracle 10g :**
Oracle Database est un système de gestion de base de données relationnel (SGBDR). Fourni par Oracle Corporation, il a été développé par Larry Ellison, accompagné d'autres personnes telles que Bob Miner et Ed Oates.

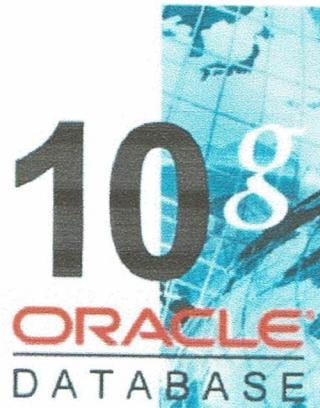


TABLE 4.1 – Oracle

Développeur	Oracle Corporation
Dernière version	12.1
Environnement	Multiple-forme
Type	SGBD
Licence	Commercial
Site web	http://www.oracle.com

— **Toad :**

Toad (Tool for Oracle Application Developers) est un logiciel de la société Quest Software qui permet de consulter et d'administrer une base de données. Il est utilisé en particulier par les développeurs Oracle et les administrateurs de bases de données.

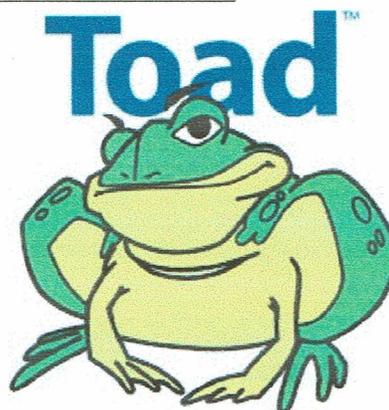


TABLE 4.2 – Toad

Développeur	Quest Software
Dernière version	11.5
Environnement	Microsoft Windows
Type	SGBD
Licence	Propriétaire
Site web	http://www.quest.com/toad

— **PowerAMC :**

Power AMC est un logiciel de modélisation. Il permet de modéliser les traitements informatiques et leurs bases de données associées. Créé par SDP sous le nom AMC*Designer, racheté par Powersoft, ce logiciel est produit par Sybase depuis le rachat par cet éditeur en 1995. Hors de France, la version internationale est commercialisée par Sybase sous la marque PowerDesigner. PowerAMC supporte différents types de modèles de bases de données.



FIGURE 4.3 – Power AMC

TABLE 4.3 – Power AMC

Développeur	SAP
Dernière version	16.5 SP05
Environnement	Microsoft Windows
Type	Outil UML
Licence	Propriétaire
Site web	http://www.sap.com/poweramc

— **Visual Studio 2015 :**

Visual Studio est un ensemble complet d'outils de développement permettant de générer des applications web ASP.NET, des services web XML, des applications bureautiques et des applications mobiles. Visual Basic, Visual C++, Visual C# utilisent tous le même environnement de développement intégré (IDE), qui leur permet de partager des outils et facilite la création de solutions faisant appel à plusieurs langages.



FIGURE 4.4 – Visual Studio

TABLE 4.4 – Visual Studio

Développeur	Microsoft
Dernière version	2015 (juillet 2015)
Écrit en	C++ et C#
Type	Environnement de développement intégré
Licence	Propriétaire
Site web	http://www.visualstudio.com

- **Windows Phone SDK 8.0 :** C'est un environnement de développement complet à utiliser afin de créer des applications et des jeux pour Windows Phone 8.0 et Windows Phone 7.5. Le kit de développement logiciel inclut de multiples émulateurs et des outils supplémentaires permettant de définir le profil et de tester l'application Windows Phone dans des conditions réelles d'utilisation.



FIGURE 4.5 – Windows Phone SDK

TABLE 4.5 – Windows Phone SDK

Développeur	Microsoft
Version	8.0
Date de publication	Octobre 2012
Type	Bibliothèque de développement
Licence	Propriétaire
Site web	dev.windowsphone.com/sl- si/downloadsdk

— **Texstudio (LATEX) :**

LATEX sert à produire des documents contenant du text. LATEX créé en 1977 par Donald Knuth (Version 3.14159 de 1982) LATEX créé en 1982 par Leslie Lamport (version 2e de 1994). Utilisations : principalement dans les milieux universitaire et enseignant (surtout scientifique).



FIGURE 4.6 – LATEX

Caractéristiques :

par rapport à Word/OpenOffice : in n'est pas de type wysiwyg (what you see is what you get). C'est un langage à balises, comme le html nécessite d'une étape de compilation pour passer du fichier source au document final. Il est très pratique pour composer des documents complexes : notes de bas de page, table de matières, des figures, bibliographie, index, mélange de plusieurs langues, équations mathématiques, etc...

Inconvénients : difficulté d'apprentissage, gestion des polices et des caractères spéciaux.

— **Le langage HTML :**

L'Hypertext Markup Language, généralement abrégé HTML, est le format de données conçu pour représenter les pages web. C'est un langage de balisage permettant d'écrire de l'hypertexte, d'où son nom. HTML permet également de structurer sémantiquement et de mettre en forme le contenu des pages, d'inclure des ressources multimédias dont des images, des formulaires de saisie, et des programmes informatiques. Il permet de créer des documents interopé-

rables avec des équipements très variés de manière conforme aux exigences de l'accessibilité du web. Il est souvent utilisé conjointement avec des langages de programmation (JavaScript) et des formats de présentation (feuilles de style en cascade). HTML est initialement dérivé du Standard Generalized Markup Language (SGML).

— **CSS :**

Les feuilles de style en cascade, généralement appelées CSS de l'anglais *Cascading Style Sheets*, forment un langage informatique qui décrit la présentation des documents HTML et XML. Les standards définissant CSS sont publiés par le World Wide Web Consortium (W3C). Introduit au milieu des années 1990, CSS devient couramment utilisé dans la conception des sites web et bien pris en charge par les navigateurs web dans les années 200.

— **JavaScript :**

JavaScript est un langage de programmation de scripts principalement employé dans les pages web interactives mais aussi pour les serveurs. C'est un langage orienté objet à prototype, c'est-à-dire que les bases du langage et ses principales interfaces sont fournies par des objets qui ne sont pas des instances de classes, mais qui sont chacun équipés de constructeurs permettant de créer leurs propriétés, et notamment une propriété de prototypage qui permet d'en créer des objets héritiers personnalisés. En outre, les fonctions sont des objets de première classe. Le langage a été créé en 1995 par Brendan Eich (Brendan Eich étant membre du conseil d'administration de la fondation Mozilla à cette époque).

— AngularJS :

AngularJS est un framework JavaScript libre et open-source développé par Google.

AngularJS est fondé sur l'extension du langage HTML par de nouvelles balises (tags) et attributs pour aboutir à une définition déclarative des pages web, par opposition à l'utilisation systématique de l'élément div et à la définition des éléments de présentation en javascript.



FIGURE 4.7 – AngularJS

TABLE 4.6 – AngularJS

Développeur	Google
Première version	2009
Dernière version	1.5.0 (février 2016)
Écrit en	JavaScript
Type	Framework JavaScript
Licence	Licence MIT
Site web	http://www.angularjs.org

— **Visual C# :**

est un langage de programmation orienté objet, commercialisé par Microsoft depuis 2002 et destiné à développer sur la plateforme Microsoft .NET. Il est dérivé du C++ et très proche du Java dont il reprend la syntaxe générale ainsi que les concepts, y ajoutant des notions telles que la surcharge des opérateurs, les indexeurs et les délégués. Il est utilisé notamment pour développer des applications web sur la plateforme ASP.NET.



FIGURE 4.6 Visual C#

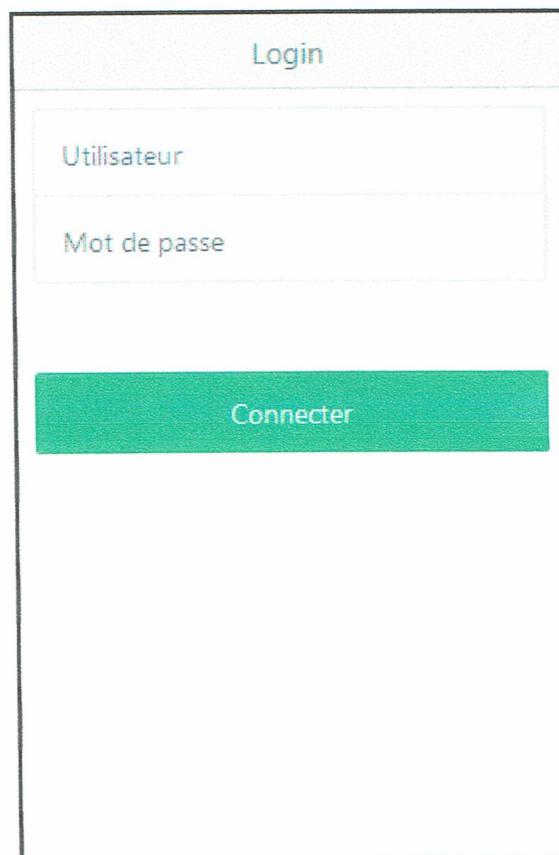
TABLE 4.7 – Visual C#

Développeur	Microsoft
Première version	2001
Dernière version	6.0 (Juillet 2015)
Famille	C
Licence	Propriétaire
Site web	msdn.microsoft.com/pt-br/vcsharp

4.3 Principales interfaces graphiques

4.3.1 Principales interfaces de l'application web

La figure 4.9 présente la page d'authentification de l'application.



The image shows a login form with a white background and a thin black border. At the top center, the word "Login" is written in a blue font. Below this, there are two input fields stacked vertically. The first field is labeled "Utilisateur" and the second is labeled "Mot de passe". Both fields have a light gray border and a white background. Below the input fields, there is a prominent green rectangular button with the word "Connecter" written in white text in the center.

FIGURE 4.9 – Interface d'authentification

La figure 4.10 représente l'interface correspondante au menu de sélection des modules.



FIGURE 4.10 – Interface du menu

La figure 4.11 s'agit du "Side menu" de l'application. Ce menu permet de naviguer entre les modules sans perdre les données saisies. Il contient aussi l'option de déconnexion.



FIGURE 4.11 – Interface du Side menu

La figure 4.12 s'agit de la page correspondante au module de transfert. L'utilisateur doit remplir les différents champs et valider afin de changer l'emplacement des articles.

The screenshot shows a mobile application interface for a transfer module. At the top, there is a navigation bar with a back arrow and the text 'Menu' on the left, and 'Transfert' on the right. Below this, the interface is divided into several sections, each with a header and a value:

- Article**: Code article 911060007
- Désignation**: Bec type 1
- Quantité**: 10 (with a spinner control)
- Source**: Centre d'origine Centre principal
- Emplacement origine**: 1
- Destination**: Centre destir MAGASIN REBUT (with a dropdown arrow)

FIGURE 4.12 – Interface changer emplacement article

La figure 4.13 présente l'interface pour faire entrer ou sortir des articles. Pour choisir entre l'entrée ou la sortie il suffit de choisir l'option désirée.

Menu Entrée/Sortie stock	
Centre	centre principal
Emplacement	emplacement automatique
Code article	911060007
Désignation	Bec type 1
Quantité	10
Entrée	<input checked="" type="checkbox"/>
Sortie	<input type="checkbox"/>

FIGURE 4.13 – Interface mettre à jour la quantité des articles en stock

La figure 4.14 s'agit de la page de gestion de l'inventaire de stock.

< Menu Inventaire	
Code article	
Désignation	
Centre	▼
Emplacement	▼
Quantité info	
Quantité réelle	

Valider Annuler

FIGURE 4.14 – Interface gérer l'inventaire

La figure 4.15 représente la page de sortie sur ordre de fabrication qui permet de demander des articles composants.



Sortie sur OF	
Centre	
Emplacement	
OF	
Article fini	
Code article	
Désignation	
Déjà sortie	
Quantité	<input type="text"/>

FIGURE 4.15 – Interface demander article composant

La figure 4.16 s'agit de la page du module de fin fabrication qui permet d'enregistrer la fin de fabrication des articles.

< Menu	Fin fabrication
OF	222
Article	KITCP/165108870
Désignation	KIT COUPE POUR FAISCE
Quantité	
A produire	10500
Déjà produites	10500
Centre	centre principal
Emplacement	1
Quantité	12 <input type="text"/>

FIGURE 4.16 – Interface enregistrer une fin fabrication

La figure 4.17 concerne l'interface du module livraison. Ce module permet de créer ou modifier des bons de livraison.

The image shows a mobile application interface for managing delivery orders. At the top, there is a navigation bar with a back arrow and the text 'Menu livraison'. Below this is a form with several input fields: 'Numero BL', 'Centre' (with a dropdown arrow), 'Emplacement' (with a dropdown arrow), 'Code article', 'Désignation', and 'Quantité' (with a numeric keypad icon). At the bottom of the form are two buttons: 'Valider' (highlighted in green) and 'Annuler'.

FIGURE 4.17 – Interface gérer les bons de livraison

4.3.2 Principales interfaces de l'application mobile

La figure 4.18 correspond à la page d'authentification de l'application. L'utilisateur doit aussi spécifier l'adresse IP du serveur pour le transfert des données.

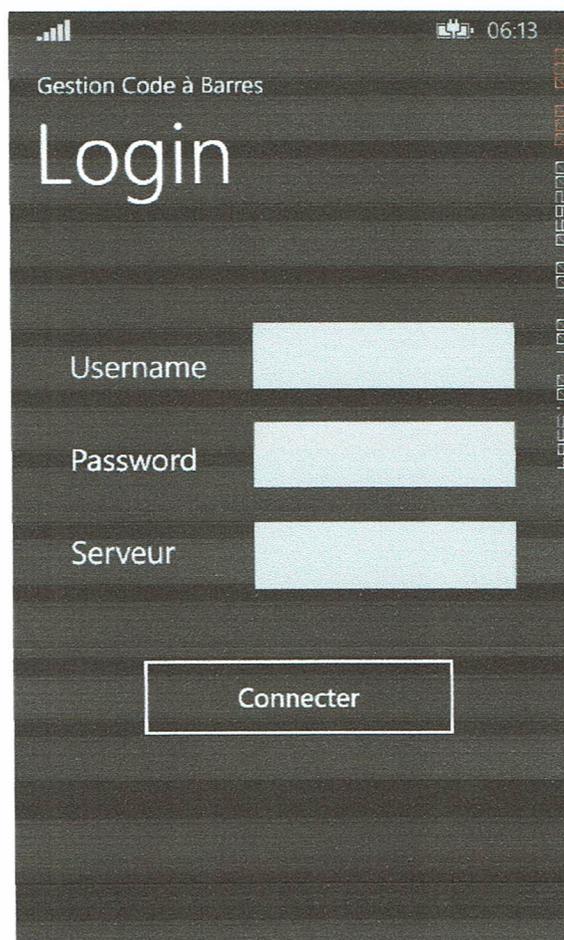


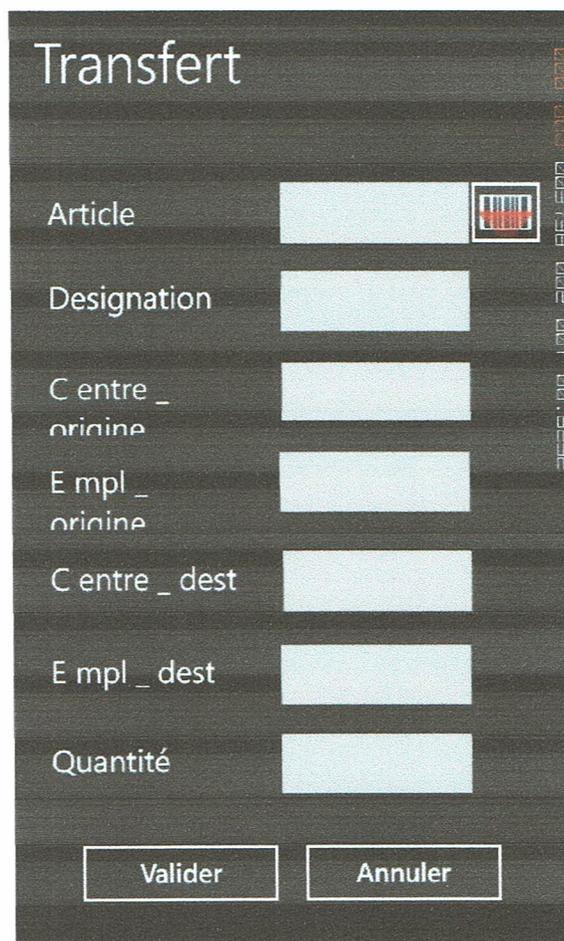
FIGURE 4.18 – Interface mobile d'authentification

La figure 4.19 représente le menu de l'application mobile qui donne la main à l'utilisateur de choisir le module désiré.



FIGURE 4.19 – Interface mobile du menu

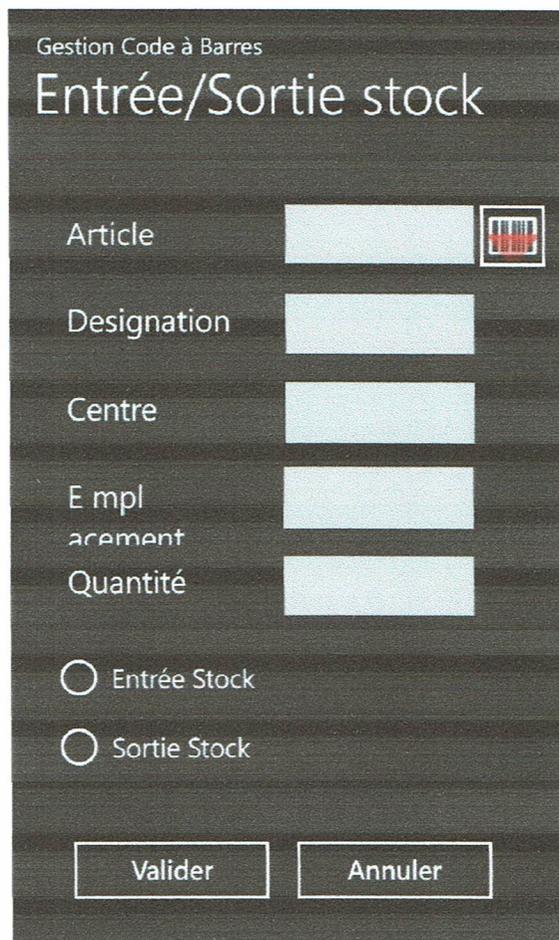
La figure 4.20 s'agit de la page du transfert. L'utilisateur peut scanner le code à barres de article en choisissant le bouton de scan. L'utilisateur peut aussi introduire le code manuellement.



The image shows a mobile application screen with a dark background and white text. The title 'Transfert' is at the top. Below it are seven input fields, each with a label to its left: 'Article', 'Designation', 'Centre _ origine', 'E mpl _ origine', 'Centre _ dest', 'E mpl _ dest', and 'Quantité'. To the right of the 'Article' field is a small icon of a barcode scanner. At the bottom of the screen are two buttons: 'Valider' on the left and 'Annuler' on the right. On the right edge of the screen, there is a vertical status bar with various icons and the time '10:00'.

FIGURE 4.20 – Interface mobile changer emplacement article

La figure 4.21 présente l'interface du module de stock. L'utilisateur peut scanner le code de l'article et remplir le formulaire pour sortir ou entrer des articles.



Gestion Code à Barres

Entrée/Sortie stock

Article 

Designation

Centre

Emplacement

Quantité

Entrée Stock

Sortie Stock

FIGURE 4.21 – Interface mobile mettre à jour la quantité des articles en stock

La figure 4.22 correspond à l'interface du module fin fabrication qui permet d'enregistrer des articles ayant fini leur fabrication.



The image shows a mobile application interface with a dark background. At the top, it says "MON APPLICATION" and "Fin fabrication". Below this, there are several input fields: "OF" (with a barcode icon), "Article", "Designation", "A produire", "D éjà produit", "Centre", "Emplacement", and "Quantité". At the bottom, there are two buttons: "Valider" and "Annuler".

FIGURE 4.22 – Interface mobile enregistrer une fin fabrication

4.4 Chronogramme de travail

La figure 2.23 présente le diagramme de Gantt de notre projet.

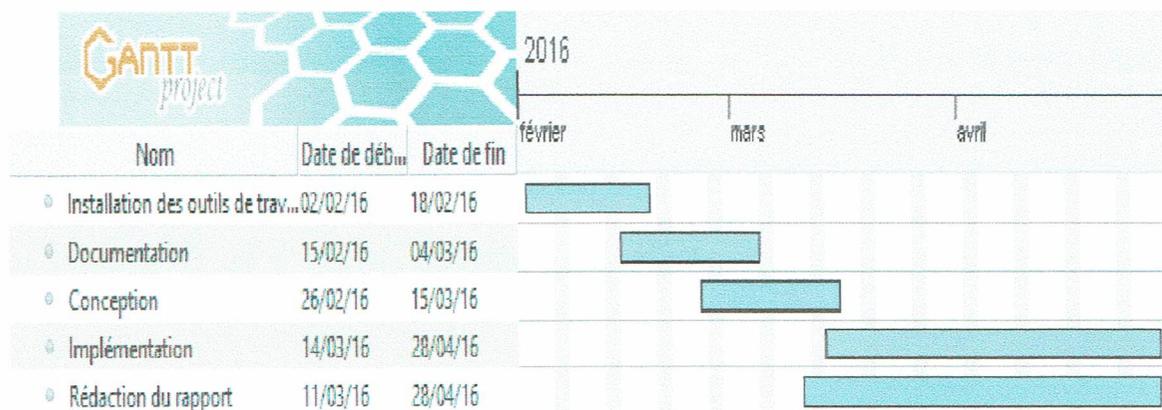
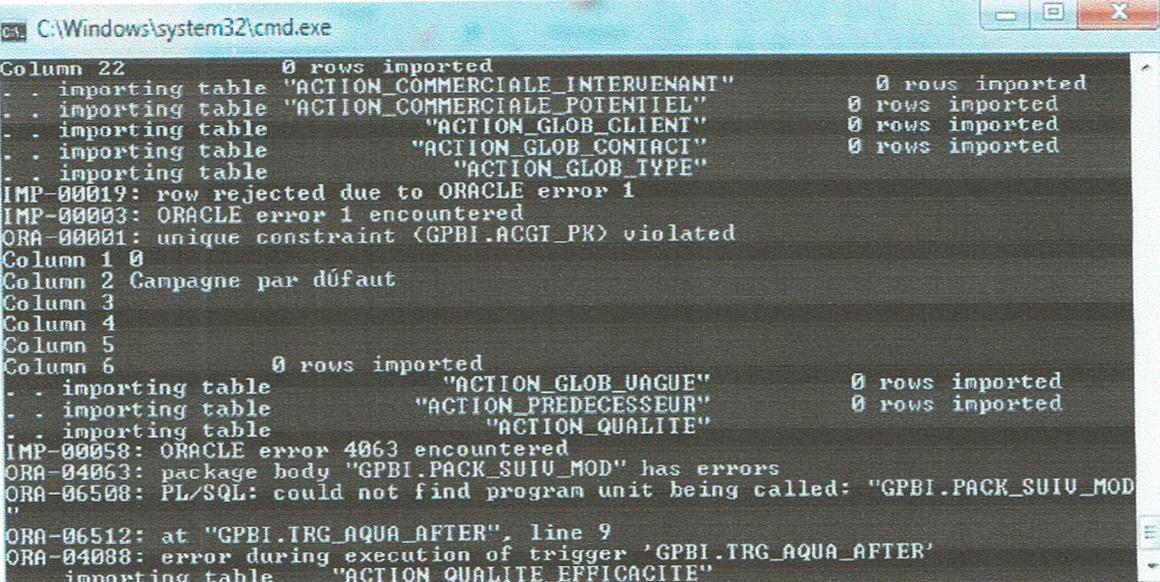


FIGURE 4.23 – Le diagramme de Gantt

4.5 Problèmes rencontrés

4.5.1 Installation de la base de donnée

La société nous a fourni la base de données sous forme d'un fichier script de commande Windows "imp". Nous avons rencontré une erreur lors de l'exécution de ce script, qui consiste à une malle formation des commandes.



```
C:\Windows\system32\cmd.exe
Column 22      0 rows imported
. . importing table "ACTION_COMMERCIALE_INTERUENANT"      0 rows imported
. . importing table "ACTION_COMMERCIALE_POTENTIEL"      0 rows imported
. . importing table "ACTION_GLOB_CLIENT"      0 rows imported
. . importing table "ACTION_GLOB_CONIACI"      0 rows imported
. . importing table "ACTION_GLOB_TYPE"
IMP-00019: row rejected due to ORACLE error 1
IMP-00003: ORACLE error 1 encountered
ORA-00001: unique constraint (GPBI.ACGI_PK) violated
Column 1 0
Column 2 Campagne par défaut
Column 3
Column 4
Column 5
Column 6      0 rows imported
. . importing table "ACTION_GLOB_UAGUE"      0 rows imported
. . importing table "ACTION_PREDECESSEUR"      0 rows imported
. . importing table "ACTION_QUALITE"
IMP-00058: ORACLE error 4063 encountered
ORA-04063: package body "GPBI.PACK_SUIV_MOD" has errors
ORA-06508: PL/SQL: could not find program unit being called: "GPBI.PACK_SUIV_MOD"
ORA-06512: at "GPBI.TRG_AQUA_AFTER", line 9
ORA-04088: error during execution of trigger 'GPBI.TRG_AQUA_AFTER'
. . importing table "ACTION_QUALITE_EFFICACITE"
```

FIGURE 4.24 – Problème de l'installation de la base de données

4.5.2 Problème avec l'émulateur de Windows Phone

Pour compiler l'application mobile, nous avons besoin d'un émulateur Windows Phone. Cet émulateur est limité de performance et était un peu lent au cours de l'exécution. Parfois il cesse de fonctionner.

De ce fait, nous avons utilisé un smartphone Windows Phone relié à la machine de développement pour compiler l'application.

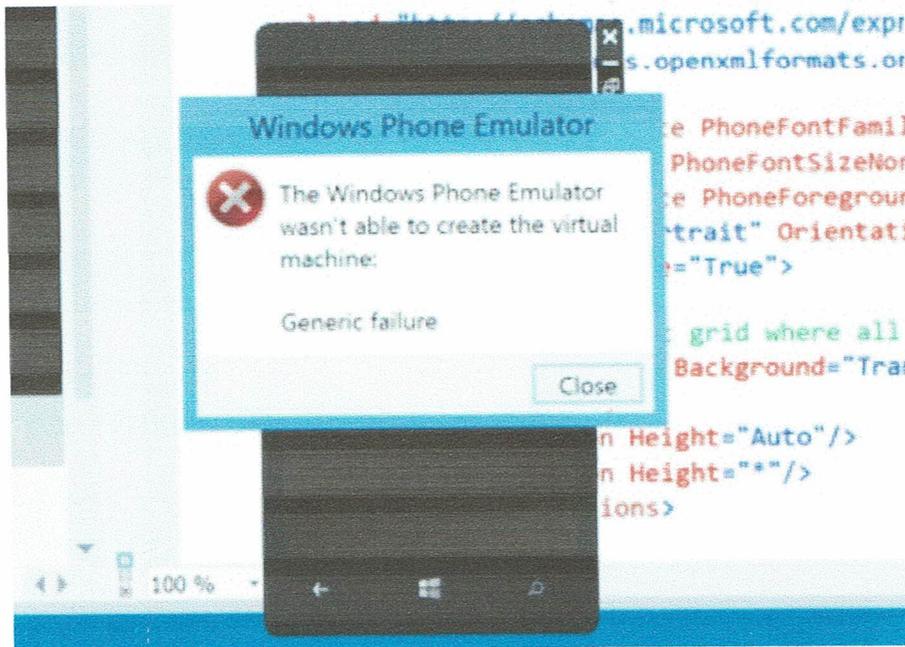


FIGURE 4.25 – Problème de l'émulateur Windows Phone

4.6 Conclusion

Dans ce chapitre nous nous sommes focalisés sur la description de l'environnement matériel et logiciel qui nous ont permis d'implanter notre application, du part et d'autre, nous avons mis en évidence les principales interfaces utilisateurs de cette dernière.

Conclusion générale

Dans le cadre du projet de fin d'étude pour l'obtention de la licence appliquée en informatique de gestion, nous avons l'occasion d'effectuer un stage au sein de Azur Consulting qui nous a permis de concrétiser nos connaissances acquises durant notre formation universitaire.

A l'occasion de ce projet, nous avons été amenés à créer une application web et mobile de gestion de stock en temps réel avec l'utilisation des codes à barres. Cela nous a permis non seulement de mettre en œuvre les connaissances acquises des trois années d'étude à l'ESSECT et d'intégrer le milieu professionnel, mais aussi d'approfondir nos connaissances par l'apprentissage et l'utilisation de nouveaux outils et technologies de développement.

Cette expérience nous a permis d'enrichir notre base de connaissance au niveau du système de gestion de base de données ORACLE et du langage UML ainsi que de profiter de l'environnement professionnel et des compétences de cette organisation par l'échange d'idées et d'informations.

De même, ce projet nous a offert l'opportunité de découvrir nos aptitudes à prendre en charge un problème ou une défaillance et de lui concevoir les solutions efficaces possibles présentant des chances nettes pour parvenir aux résultats adéquats.

Ce stage constitue une expérience professionnelle valorisante et encourageante pour l'avenir. Enfin, c'est satisfaisant d'avoir pu travailler dans des bonnes conditions et dans un environnement agréable.

Références

- (I) <http://www.google.com>
- (II) <http://fr.wikipedia.org>
- (III) <http://www.oracle.com>
- (IV) <http://msdn.microsoft.com>
- (V) <http://developpez.com>
- (VI) <http://commentcamarche.net>

