

République Algérienne Démocratique
et Populaire Université Abou Bekr
Belkaid– Tlemcen
Faculté des Sciences
Département d'Informatique

Mémoire de fin d'études

Pour l'obtention du diplôme de Master en Informatique

Option : Génie Logiciel (G.L)

*Conception et réalisation d'une plateforme de gestion des stocks et
de production pour une usine de chewing-gum 'SweetStock'*

Thème

Réalisé par :

- Kamli Kamilya.

Présenté le 23 juin 2026 devant le jury composé de :

- | | |
|---------------------------------|-------------------|
| - Mme. El Yebdri Zeyneb | (Présidente) |
| - Mme. Marouf Radja | (Examinatrice) |
| - Mme. Benmansour Fassila Lemya | (Encadrante) |
| - M. Bouabdellah Karim | (Maître de stage) |

Année universitaire : 2025/2026

Remerciements

Tout d'abord, je tiens à exprimer ma profonde gratitude à ALLAH Tout-Puissant, qui m'a accordé la force, la patience et la persévérance nécessaires pour la réalisation de ce mémoire.

Mes sincères remerciements et toute ma gratitude vont à mon encadrante de mémoire, Mme BENMANSOUR Fassila Lemya, qui est à la fois une excellente encadrante et une personne merveilleuse. L'aboutissement de ce mémoire n'aurait pas été possible sans ses orientations et ses conseils, qui m'ont accompagnée tout au long des différentes étapes de sa rédaction.

J'adresse également mes sincères remerciements à M. BOUABDELLAH Karim, maître de stage, pour son accueil au sein de l'entreprise « Naltis Communication » et pour la confiance qu'il m'a accordée en me confiant ce projet. Mes remerciements vont aussi à mes encadrantes de stage, Mme DJELLOULI Soraya et Mme BAGHDADLI Chahred, pour leur disponibilité, leur accompagnement et leurs précieux conseils tout au long de ma période de stage.

Mes sincères remerciements vont également aux membres du jury : Mme EL YEBDRI Zeyneb et Mme MAROUF Radja, qui ont accepté de lire et d'examiner mon travail, et qui ont consacré leur temps et leur savoir à évaluer la pertinence et la crédibilité de cette recherche.

Je remercie également tous les enseignants du département d'informatique pour leurs efforts et les savoirs qu'ils m'ont transmis tout au long de mes études universitaires.

Dédicaces

Je dédie ce travail à moi-même

À ma famille

Et à mes amis.

Table des matières

<i>Liste des figures</i>	i
<i>Liste des tableaux</i>	ii
<i>Liste des acronymes</i>	iii
Introduction générale	1
Contexte du projet	1
Présentation de l'entreprise d'accueil	1
Problématique	2
Objectifs du projet	2
Plan du mémoire	3
Chapitre I : Analyse des systèmes existants et étude comparative	4
I.1. Introduction	4
I.2. Contexte de transformation numérique industrielle	4
I.3. Présentation du secteur de la confiserie en Algérie	5
I.4. Notion d'ERP (Enterprise Resource Planning)	5
I.4.1. Définition d'ERP	5
I.4.2. Historique d'ERP	6
I.4.3. Rôle d'ERP	7
I.5. Présentation générale des systèmes existants	8
I.5.1. Logiciels généralistes	8
I.5.1.1. SYNAPSE ERP	8
I.5.1.2. GestiumERP	10
I.5.2. Logiciels dédiés	12
I.5.2.1. Konnect'AGRO	12
I.5.2.2. Sage X3 Food & Beverage	14
I.5.3. Analyse critique des application existantes	16
I.6. Étude comparative détaillée des logiciels existants	17
I.7. Positionnement de la solution proposée SweetStock	19
I.8. Conclusion	20
Chapitre II : Spécification des besoins et exigences de l'application « SweetStock »	21
II.1. Introduction	21
II.2. Identification des acteurs	21

II.3. Description des besoins (user stories).....	22
II.3.1. Besoins liés à l'administrateur	22
II.3.2. Besoins liés au magasinier	23
II.3.3. Besoins liés à l'atelier	23
II.4. Spécification des exigences fonctionnelles et non-fonctionnelles.....	24
II.4.1. Spécification des exigences fonctionnelles	24
II.4.2. Spécification des exigences non-fonctionnelles.....	28
II.5. Diagramme des cas d'utilisation.....	29
II.5.1. Diagramme de cas d'utilisation de l'administrateur	31
II.5.2. Diagramme de cas d'utilisation du magasinier	32
II.5.3. Diagramme de cas d'utilisation de l'atelier.....	33
II.6. Scénarios nominaux et alternatifs des cas d'utilisation	34
II.6.1. Scénario nominal et alternatif de validation d'un batch.....	34
II.6.1.1. Scénario nominal de validation d'un batch	35
II.6.1.2. Scénario alternatif de validation d'un batch	35
II.6.2. Scénario nominal et alternatif de réalisation d'un inventaire.....	36
II.6.2.1. Scénario nominal de réalisation d'un inventaire	36
II.6.2.2. Scénario alternatif de réalisation d'un inventaire	36
II.7. Diagramme de séquence	37
II.7.1. Diagramme de séquence de validation d'un batch.....	37
II.7.2. Diagramme de séquence de l'enregistrement d'une réception de commande fournisseur.....	39
II.8. Diagramme d'activités	41
II.9. Conclusion	43
Chapitre III : Conception de l'application « SweetStock ».....	44
III.1. Introduction	44
III.2. Architecture de l'application	44
III.3. Diagramme de classes	46
III.3.1. Diagramme de classes du module de gestion des stocks	46
III.3.2. Diagramme de classes du module de gestion de production	50
III.4. Modèle de données	54
III.4.1. Modèle de données des référentiels de base	54
III.4.2. Modèle de données du module de gestion des stocks.....	55
III.4.3. Modèle de données du module de gestion de production	57

III.5. Conclusion	58
Chapitre IV : Réalisation et implémentation de l'application « SweetStock »	59
IV.1. Introduction	59
IV.2. Processus du travail	59
IV.3. Technologies choisies	60
IV.4. Outils de développement	61
IV.5. Présentation des interfaces de l'application	62
IV.5.1. Interface de connexion	62
IV.5.2. Tableau de bord administrateur	63
IV.5.3. Logs du système	64
IV.5.4. Rôles du système	65
IV.5.5. Utilisateurs	65
IV.5.6. Fournisseurs	66
IV.5.7. Clients	67
IV.5.8. Dépôts et rayons	67
IV.5.9. Familles d'unités et les unités de mesure	68
IV.5.10. Exercices	69
IV.5.11. Gestion des produits	70
IV.5.11.1. Types produits	70
IV.5.11.2. Catégories de produits	70
IV.5.11.3. Produits	71
IV.5.11.4. Conditionnements	72
IV.5.12. Gestion des achats	73
IV.5.12.1. Commandes fournisseurs	73
IV.5.12.2. Réceptions	73
IV.5.12.3. Retours fournisseur	74
IV.5.13. Commandes clients	75
IV.5.14. Inventaires	75
IV.5.15. État des stocks	76
IV.5.16. Décharges	77
IV.5.17. Pertes de stock	78
IV.5.18. Transferts	78
IV.5.19. Gestion de production	79
IV.5.19.1. Recettes	79

IV.5.19.2. Batches.....	80
IV.5.19.3. Sessions de dragéification	81
IV.5.19.4. Ordres de conditionnement.....	81
IV.6. Conclusion	82
Conclusion générale & perspectives.....	83
Webographie.....	85

Liste des figures

Figure 1 : Historique d'ERP.....	6
Figure 2 : Propriétés ERP.....	7
Figure 3 : Le logo de l'entreprise algérienne MEGATHEQUE	9
Figure 4 : Le logo de l'entreprise algérienne CIRTASOFT	10
Figure 5 : Tableau de bord principal du logiciel GestiumERP.....	11
Figure 6 : Le logo de l'entreprise française AKANEA	12
Figure 7 : un aperçu du logiciel Konnect'AGRO	14
Figure 8 : Le logo du groupe Sage	14
Figure 9 : Tableau de bord du logiciel Sage X3	16
Figure 10 : Diagramme des cas d'utilisation de l'application « SweetStock ».....	30
Figure 11 : Diagramme de cas d'utilisation des tâches de l'administrateur.....	31
Figure 12 : Diagramme de cas d'utilisation des tâches du magasinier	32
Figure 13 : Diagramme de cas d'utilisation des tâches de l'atelier.....	33
Figure 14 : Diagramme de Séquence système de validation d'un batch.....	38
Figure 15 : Diagramme de Séquence système de l'enregistrement d'une réception de commande fournisseur	40
Figure 16 : Diagramme d'activités de validation d'un batch	42
Figure 17 : Illustration de l'architecture de l'application « SweetStock ».....	45
Figure 18 : Diagramme de classes du module de gestion des stocks.	47
Figure 19 : Diagramme de classes du module de production.	51
Figure 20 : Modèle de donnée des référentiels de base.....	55
Figure 21: Modèle de donnée du module de gestion des stocks.	56
Figure 22 : Modèle de données du module de gestion de production.....	57
Figure 23 : Tableau de bord Jira du projet « SweetStock ».....	59
Figure 24 : Interface de connexion.....	63
Figure 25 : Tableau de bord administrateur.....	64
Figure 26 : Logs du système	64
Figure 27 : Rôles du système	65
Figure 28 : Liste des utilisateurs du système.....	66
Figure 29 : Liste des fournisseurs	66
Figure 30 : Liste des clients.....	67

Figure 31 : Liste des dépôts de l'entreprise.....	68
Figure 32 : Liste des familles d'unités	68
Figure 33 : Liste des unités de mesure	69
Figure 34 : Liste des exercices	69
Figure 35 : Liste des types produits	70
Figure 36 : Liste des catégories de produits	71
Figure 37 : Liste des produits.....	72
Figure 38 : Liste des conditionnements.....	72
Figure 39 : Liste des commandes fournisseurs	73
Figure 40 : Liste des réceptions.....	74
Figure 41 : Liste des retours fournisseurs	74
Figure 42 : Liste des commandes clients	75
Figure 43 : Liste des inventaires	76
Figure 44 : État des stocks.....	77
Figure 45 : L'historique des mouvements d'un lot.....	77
Figure 46 : Liste des décharges	78
Figure 47 : Liste des pertes de stock	78
Figure 48 : Liste des transferts	79
Figure 49 : Liste des recettes.....	80
Figure 50 : Liste des batches	80
Figure 51 : Liste des sessions de dragéification	81
Figure 52 : Liste des ordres de conditionnement	82

Liste des tableaux

Tableau 1 : Tableau comparatif des logiciels de gestion	18
---	----

Liste des acronymes

Acronyme	Signification
CRM	Gestion de la Relation Client
DLC	Dates Limite de Consommation
ERP	Planification des Ressources de l'Entreprise
FEFO	Premier Expiré, Premier Sorti (First Expired, First Out)
GPAO	Gestion de la Production Assistée par Ordinateur
IDE	Environnement de Développement Intégré (Integrated Development Environment)
KPI	Indicateur Clé de Performance
MRP	Planification des besoins en matières
PGI	Progiciel de Gestion Intégré
PME	Petite et Moyenne Entreprise
RH	Ressources Humaines
SCF	Système Comptable Financier
SIG	Système d'Information Géographique
SQL	Structured Query Language (Langage de Requêtes Structurées)
TAP	Taxe sur l'Activité Professionnelle
TVA	Taxe sur la Valeur Ajoutée
UML	Langage de Modélisation Unifié (Unified Modeling Language)

Introduction générale

Contexte du projet

Dans un contexte économique où l'optimisation des ressources constitue un facteur clé de compétitivité, les entreprises et les usines industrielles, en particulier dans le secteur agroalimentaire, sont confrontées à une pression croissante pour rester compétitives et efficaces, notamment à la une de la transformation numérique. Par conséquent, elles recherchent des solutions pour optimiser leurs processus de stockage et de production, ainsi que leurs opérations d'achat et de vente.

Aujourd'hui, la numérisation des opérations est un outil crucial qui renforce le suivi des matières premières et des produits finaux, permet d'améliorer la traçabilité des stocks et la planification de la production, la gestion des fournisseurs et assure la gestion des facteurs d'achat et des activités commerciales. Dans ce cadre, la gestion des stocks est essentielle. Elle régule la relation entre la logistique, la production et la distribution des produits. Cette notion de stock regroupe : les marchandises, les matières premières, les matières d'emballage, les produits finis et les produits en cours de transformation (semi-finis), etc.

C'est dans cette dynamique que s'inscrit notre projet, réalisé au sein de l'entreprise « Naltis », dans le cadre de notre stage de fin d'études. L'entreprise nous a confié la réalisation d'une solution qui répond à une demande concrète de ses clients, qui manquaient d'un outil adapté aux spécificités de leur secteur. Le système développé, que nous avons baptisé « SweetStock », est destiné à une usine de confiserie de « chewing-gum ». Il intègre la gestion des stocks, de la production, des fournisseurs et des clients, ainsi que des commandes, des réceptions et des retours de fournisseurs. Ce système vise à automatiser, optimiser et centraliser les processus afin de minimiser les erreurs et les interruptions, tant dans les processus de production que dans le stock.

Présentation de l'entreprise d'accueil

Fondée en 2005 à Tlemcen par deux jeunes Algériens, « Naltis Communication » a débuté dans le domaine du web avant d'évoluer progressivement avec les nouvelles technologies, en les intégrant pleinement dans son activité. L'entreprise se distingue par sa capacité à concevoir des plateformes informatiques sur des supports modernes. Au fil des années, l'entreprise n'a pas seulement accompagné l'évolution des technologies émergentes, mais elle s'est également engagée activement dans leur développement. Elle propose deux types de services [\[1\]](#) :

- Le conseil en communication d'entreprise pour un marketing efficace et ciblé.
- Le développement de logiciels personnalisés, fiables et faciles à utiliser, s'appuyant sur

une solide maîtrise des langages de programmation.

Problématique

Actuellement, la gestion se fait via des fichiers Excel complexes, ce qui entraîne des risques d'erreurs de saisie, des difficultés de traçabilité et un manque de visibilité en temps réel sur les coûts.

Ainsi, la problématique principale de ce mémoire est la suivante : comment concevoir un système intégré de gestion opérationnelle qui optimise la gestion des stocks, de la production, des commandes, des réceptions et des transferts, tout en s'adaptant aux besoins et exigences de l'usine ?

Objectifs du projet

L'objectif principal de ce mémoire est de présenter le cycle de vie de ce projet, qui comprend les différentes étapes d'analyse, de conception et de développement de l'application « SweetStock ». Il met en avant les choix techniques et fonctionnels réalisés et explique comment cette solution répond aux besoins spécifiques d'une usine de confiserie, en matière de gestion des stocks, de traçabilité et de coordination entre les dépôts. Conçue pour une usine cliente, l'application peut aussi être adaptée à d'autres usines de confiserie qui souhaitent centraliser leurs données de stock et de production, tout en assurant une traçabilité complète pour l'ensemble des modules du système. Cette solution vise à offrir :

- Une gestion centralisée des données de stock et de production.
- Assurer la traçabilité complète des matières et des lots.
- Contrôler la disponibilité des ingrédients avant la production.
- Concevoir une interface simple, moderne et accessible à tous types d'utilisateurs.
- Faciliter les inventaires et les audits.
- Suivre les flux de production, de l'achat des matières premières (sucre, etc.) au conditionnement final, en passant par le suivi des étapes de fabrication.
- Réduire les erreurs manuelles et les interruptions des processus de production.

Ces objectifs partagent un désir commun : fournir à l'entreprise un outil de gestion fiable, centralisé et adapté à ses besoins opérationnels. Notre application s'engage pleinement dans la transformation numérique afin d'améliorer les performances globales et opérationnelles de l'entreprise.

Plan du mémoire

Ce mémoire est structuré en quatre chapitres distincts, précédé d'une introduction générale et suivi d'une conclusion générale :

- Dans **le chapitre 01**, intitulé « **Analyse des systèmes existants et étude comparative** », nous présentons un aperçu du concept d'ERP et de ses principales catégories, puis nous effectuons une étude comparative des logiciels existants de ces deux catégories. En mettant en évidence leurs points forts et leurs points faibles, nous pourrions tirer des conclusions importantes pour notre projet. Enfin, ce chapitre se conclura par une synthèse des points principales.
- Dans **le chapitre 02**, intitulé « **Analyse des besoins et spécification des exigences de l'application SweetStock** », nous identifions les principaux acteurs du système et exprimons leurs besoins fonctionnels. Ensuite, nous définissons les exigences fonctionnelles et non fonctionnelles, ainsi que les scénarios nominaux et alternatifs des cas d'utilisation les plus importants. Comme nous présentons quelques diagrammes UML pour définir le comportement du système : des diagrammes de cas d'utilisation, de séquence et d'activité. Enfin, nous concluons ce chapitre par un résumé des points importants et des avantages de notre analyse des spécifications du système.
- Dans **le chapitre 03**, intitulé « **Conception de l'application SweetStock** », nous nous concentrons sur la conception et l'architecture du système. Nous commençons par définir l'architecture globale de l'application, en précisant les principaux composants et leurs interactions. Nous présentons le diagramme de classe, puis le modèle de données pour définir la structure de la base de données. Enfin, nous concluons ce chapitre en résumant brièvement son contenu.
- Dans **le chapitre 04**, intitulé « **Réalisation et implémentation de l'application SweetStock** », nous exposons le processus de travail suivi pendant l'implémentation de ce projet, justifions les choix technologiques effectués et décrivons les outils utilisés. Ensuite, nous illustrons les différentes interfaces de l'application « SweetStock » à l'aide d'une description des fonctionnalités implémentées dans chaque interface.

Chapitre I : Analyse des systèmes existants et étude comparative

I.1. Introduction

Que ce soit dans le contexte économique ou organisationnel, la transformation numérique a remis en question beaucoup de choses ces dernières années. Et dans un domaine en constante évolution comme l'est la production industrielle. Aujourd'hui, les PME n'ont d'autre choix que de s'adapter au plus vite à cette révolution numérique pour rester compétitives. Il convient, aussi, de mentionner que cette dernière n'est pas l'apanage des grandes entreprises seulement ; bien au contraire, les PME ont beaucoup à y gagner : elle leur permettra en effet d'automatiser et d'optimiser leurs processus de gestion des stocks, de production, de logistique et de maintenance, et d'améliorer l'efficacité opérationnelle du système.

Dans ce chapitre, nous proposons une analyse des systèmes existants, tels que les ERP et les logiciels de gestion opérationnelle, en nous concentrant sur leurs apports et leurs limites. Cette analyse permettra d'identifier les opportunités pour notre système et de définir son positionnement.

I.2. Contexte de transformation numérique industrielle

La transformation numérique industrielle fait partie de l'industrie 4.0 : une nouvelle phase de modernisation des systèmes de production. Ce terme décrit le processus par lequel les entreprises du secteur industriel intègrent la technologie numérique à tous les niveaux de leurs opérations, entraînant des changements fondamentaux dans la manière dont la valeur et les services sont fournis.

Avec la digitalisation, les processus industriels deviennent de plus en plus automatisés et optimisent la chaîne d'approvisionnement. Cela permet aux entreprises de réduire les coûts d'exploitation et de leur donner un avantage concurrentiel en leur permettant de réagir rapidement aux changements du marché. Cette numérisation facilite la collecte de données, ce qui est essentiel pour l'analyse des données et la création de nouveaux modèles d'entreprise qui conviennent à l'ère numérique [2].

Toutefois, l'intégration de ces technologies de numérisation nécessite la mise en œuvre de solutions logicielles adaptées, capables de répondre aux besoins de chaque secteur industriel.

I.3. Présentation du secteur de la confiserie en Algérie

En Algérie, la confiserie relève du secteur agroalimentaire. Il constitue un élément important de l'économie nationale. Ce secteur regroupe de nombreuses entreprises et usines spécialisées dans la production de confiseries, notamment de chewing-gum et de chocolat.

Le marché algérien de la confiserie est segmenté selon le type de produit, le type d'emballage et le canal de distribution.

De manière générale, le secteur industriel dépend des matières premières, de la traçabilité logistique, de la production et du respect des normes sanitaires. À cet égard, une gestion efficace des différents modules au sein de l'entreprise est essentielle pour assurer leur productivité et éviter les pertes [3].

Cependant, les entreprises dépendent encore de fichiers Excel et d'outils traditionnels pour la gestion, ce qui complique le pilotage des processus. Par conséquent, l'implantation d'une solution numérique adaptée est cruciale pour améliorer la performance et la compétitivité des entreprises de confiserie algériennes.

I.4. Notion d'ERP (Enterprise Resource Planning)

Dans cette partie, nous allons définir la notion d'ERP, expliquer les rôles au sein de l'entreprise, ainsi que ses principales caractéristiques.

I.4.1. Définition d'ERP

Un système ERP, ou, en d'autres termes, progiciel de gestion intégré (PGI), est un type de logiciel que les entreprises utilisent pour gérer leurs principaux processus métiers, notamment ceux des fonctions telles que la gestion financière, logistique, production, ressources humaines, ventes et achats, etc. L'ERP relie de nombreux processus qui offrent une vision globale de l'activité et simplifient le flux d'échange des données entre eux ; cela permet de réduire les erreurs et d'augmenter la prise de décision [4].

L'ERP est constitué d'une architecture qui partage et unifie la base de données entre tous les modules, ce qui aide à centraliser les données, garantissant ainsi la cohérence et la fiabilité des informations à l'échelle de l'entreprise.

I.4.2. Historique d'ERP

Les systèmes ERP ont commencé comme de simples systèmes papier de planification de la production. Au début des années 1960, les entreprises manufacturières ont commencé à adopter des applications informatiques pour leurs avantages (plus rapides et plus précises que le travail manuel), malgré leur coût élevé, leurs fonctionnalités limitées et leur relative lenteur [5].

L'évolution des systèmes d'information a conduit à la transformation de ces applications distinctes (par exemple : CRM, MRP, etc.) en un système intégré unique appelé ERP. Dans les années 1990, l'ERP a transformé le secteur technologique en prenant en charge un plus grand panel d'activités dans de nombreuses industries [5].

L'accélération d'Internet et les nouveaux outils de développement logiciel ont davantage révolutionné les suites ERP. L'introduction d'un logiciel basé sur un navigateur a ouvert la voie à l'ERP cloud, ce qui a permis d'élargir l'adoption, la portée et l'éventail des fonctionnalités des solutions ERP.

Les systèmes ERP modernes intègrent des technologies d'intelligence artificielle (IA), etc. Ces fonctionnalités aident les entreprises à transformer leurs opérations et à rester compétitives en automatisant les tâches, en anticipant les résultats et en facilitant des prises de décision plus éclairées dans le domaine de la finance, entre autres [5]. La figure 1 illustre l'évolution historique des systèmes ERP.

The history of ERP

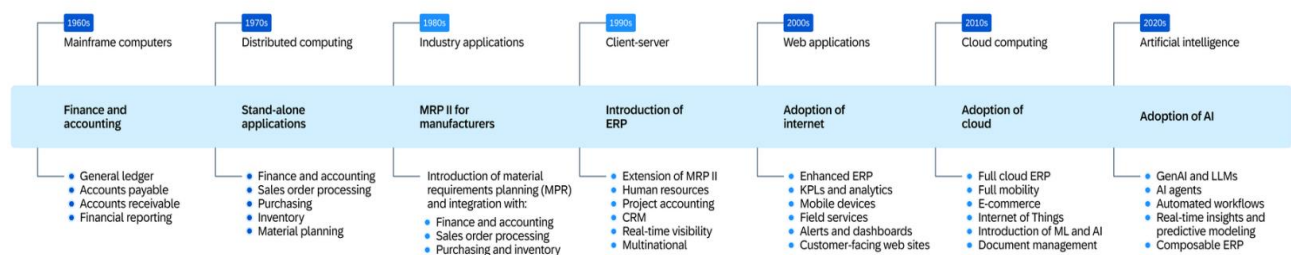


Figure 1 : Historique d'ERP [5]

I.4.3. Rôle d'ERP

Les ERP jouent un rôle important pour gérer l'ensemble des activités d'entreprises, ce qui peut être résumé ainsi [6] :

- Centralisation des données.
- Automatisation des processus métier.
- Intégration transparente des processus et des workflows.
- Amélioration de la communication entre les services.
- Meilleure coordination des services et donc meilleur suivi des processus
- Aide à la prise de décision.
- Aider à la gestion des ressources et des modules de l'entreprise.
- La réduction des coûts et des erreurs.

La figure 2 présente les principaux modules d'un système ERP et illustre l'importance dans la centralisation des activités de l'entreprise. L'ERP occupe une position centrale, couvrant la production, la comptabilité, la distribution, les ressources humaines, la gestion des inventaires et des stocks, ainsi que l'achat et la vente.



Figure 2 : Propriétés ERP [7]

I.5. Présentation générale des systèmes existants

Avant de concevoir une solution comme « SweetStock », il est essentiel de présenter les logiciels existants. Cette phase nous aide à définir les points faibles que « SweetStock » vise à résoudre. Les systèmes de gestion disponibles sur le marché peuvent être classés en deux grandes catégories.

La première catégorie comprend les logiciels généralistes qui peuvent être utilisés dans différents secteurs (industrie, services, etc.) mais nécessitent généralement des modifications personnalisées coûteuses pour répondre à des besoins spécifiques. La deuxième catégorie représente les logiciels dédiés à un secteur précis (agroalimentaire, logistique, etc.) et il n'y a pas besoin de modifier, car ils proposent des fonctionnalités prêtes à l'usage mais limitées à un secteur spécifique.

Dans ce qui suit, nous allons présenter deux logiciels déjà présents sur le marché pour chaque catégorie. Pour la première catégorie, il existe « SYNAPSE ERP » et « Gestium ERP » sur le marché algérien, ainsi que deux logiciels dédiés au secteur agroalimentaire international, à savoir « Kconnect'AGRO » et « Sage X3 Food & Beverage ».

I.5.1. Logiciels généralistes

Les logiciels généralistes, également appelés ERP généralistes, sont le modèle le plus standard d'ERP. Ces logiciels sont des PGI conçus pour piloter l'ensemble des processus de l'entreprise. Ce type de logiciel est une solution qui peut être adaptée à une variété de secteurs et d'entreprises grâce à sa capacité de paramétrage particulièrement développée [8]. Il propose des modules standards pour des fonctions communes telles que la gestion des finances, les ressources humaines (RH), la gestion des stocks, etc.

Parmi les logiciels généralistes disponibles sur le marché algérien, nous avons retenu « SYNAPSE ERP » et « Gestium ERP » comme logiciels représentatifs de cette catégorie.

I.5.1.1. SYNAPSE ERP

SYNAPSE ERP est un logiciel développé par la société algérienne « MEGATHEQUE ». La figure 3 présente le logo de cette société qui a développé ce logiciel.



Figure 3 : Le logo de l'entreprise algérienne MEGATHEQUE [9]

a) Définition :

L'ERP SYNAPSE intègre un ensemble de modules nécessaires à la gestion complète des ressources de l'entreprise, de la gestion commerciale au CRM, en passant par la chaîne d'approvisionnement et la production. Le tout dans un environnement intégré, avec une base de données unique partagée entre tous les modules d'application, ce qui permet d'assurer la cohérence de l'information [10].

b) Fonctionnalités :

SYNAPSE ERP offre un ensemble de modules couvrant les principaux besoins opérationnels d'une entreprise. Les fonctionnalités sont offertes comme suit [11] :

- Gestion commerciale complète : achats, ventes, suivi des devis et facturation.
- Gestion de la relation client (CRM) : suivi des clients et de leurs interactions.
- Gestion des stocks : permet la gestion de mono- et multi-dépôts, ainsi que des mouvements de stock en temps réel.
- Gestion de Ressources humaines (RH) et Paie.
- Gestion de la production.

c) Limites :

Bien que SYNAPSE ERP soit une solution locale, adaptée au contexte algérien, elle présente plusieurs limites pour une entreprise du secteur de la confiserie. Ces limitations sont comme suit :

- Absence de spécialisation métier : SYNAPSE ERP est un logiciel généraliste et ne dispose ni de module dédié à l'industrie agroalimentaire ni à la confiserie.
- Absence de traçabilité des lots par recette et de suivi des matières premières.
- Absence de gestion des dates limites de consommation (DLC), ce qui constitue un module essentiel dans le secteur agroalimentaire.
- L'adaptation personnalisée aux besoins spécifiques du secteur de la confiserie est coûteuse.

I.5.1.2. GestiumERP

GestiumERP est un logiciel développé 100% en Algérie par la société algérienne « CIRTASOFT ». La figure 4 présente le logo de la société algérienne « CIRTASOFT ».



Figure 4 : Le logo de l'entreprise algérienne CIRTASOFT [12]

a) Définition :

GestiumERP est une solution intégrée et complète, développée avec une large palette de modules métiers et adaptée aux particularités locales. Il est conçu pour répondre aux besoins réels des entreprises tout en respectant les normes du Système Comptable Financier (SCF) algérien [13]. Cette solution centralise l'ensemble des processus de l'entreprise, tels que la production, le stock, les achats, les ventes et la comptabilité. Au sein d'une base de données unique et partagée entre tous les modules.

b) Fonctionnalités :

GestiumERP offre de nombreux modules qui répondent aux besoins opérationnels essentiels d'une entreprise. Ses fonctionnalités incluent les éléments suivants [13] :

- Tableaux de bord interactifs, qui présentent des rapports personnalisés, des statistiques approfondies et des alertes pour l'aide à la prise de décision en temps réel.
- Gestion des ressources humaines (RH).
- Gestion de la distribution et de la livraison en temps réel.
- Gestion de la comptabilité et de la finance.
- Gestion de production (GPAO) : support la planification des ordres de fabrication et suivi des machines et optimisation des capacités de production.
- Gestion des stocks et de la logistique : suivi en temps réel des mouvements de stock, gestion multi-dépôt et prise en charge des alertes de rupture.
- Appuie les ventes et la facturation.

c) Limites :

Malgré toutes ces fonctionnalités, « GestiumERP » présente encore certaines limites dans le secteur de la confiserie. Ces limites sont comme suit :

- Absence de spécialisation en confiserie ; il ne dispose pas de modules métier spécifiques à la confiserie, tels que la gestion des variantes de conditionnement.
- Son adaptation complète à ce secteur nécessite un développement supplémentaire, ce qui entraîne des coûts supplémentaires.

La figure 5 présente le tableau de bord du logiciel « GestiumERP ». Il offre une vue centralisée des divers indicateurs de performance (Key Performance Indicators, KPI) : chiffre d'affaires, achats, règlement des clients, charges, avec des statistiques fournissant un résumé en temps réel des activités.

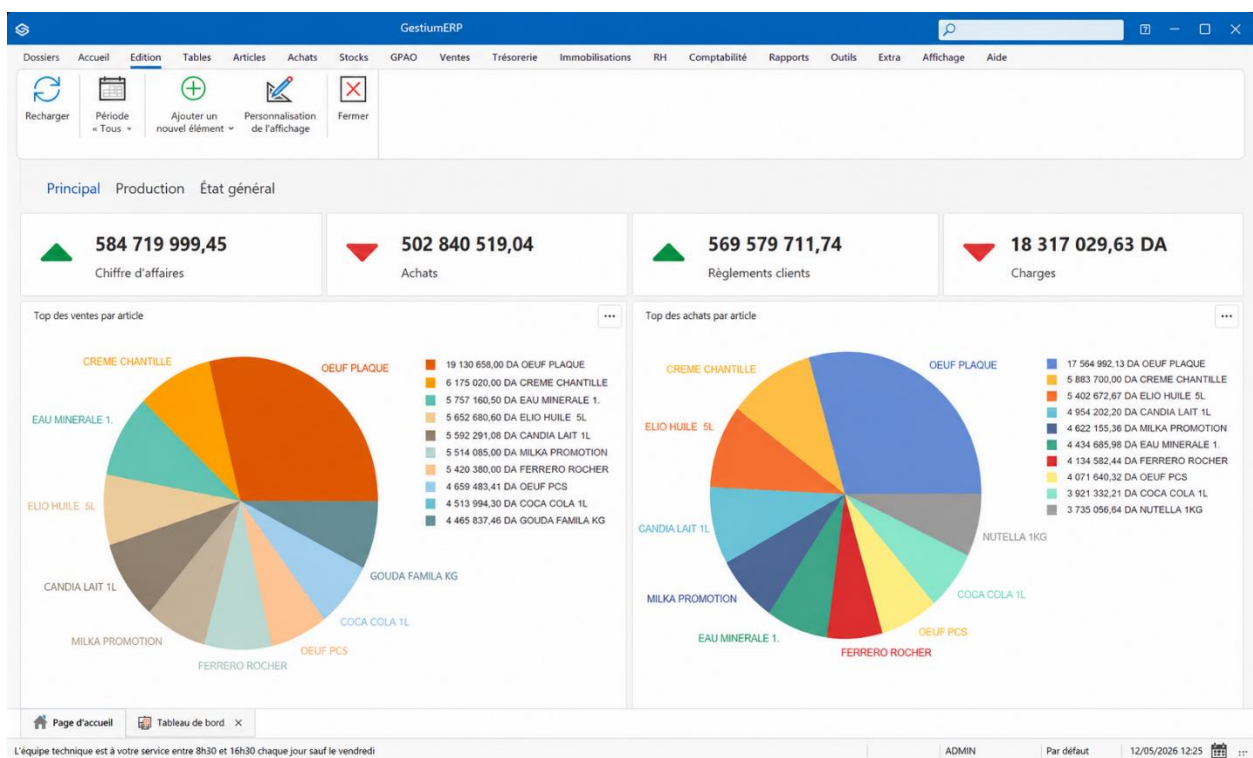


Figure 5 : Tableau de bord principal du logiciel GestiumERP [14]

I.5.2. Logiciels dédiés

Les logiciels dédiés, également appelés ERP métiers ou ERP spécialisés, sont conçus pour un secteur ou une activité spécifique, comme l'industrie pharmaceutique, l'agroalimentaire, etc. Il intègre les mêmes fonctionnalités et processus métier que les logiciels généralistes, mais prédéfinis et adaptés aux particularités du secteur et aux besoins opérationnels que ces derniers ne peuvent tout simplement pas fournir [15]. Par exemple, un logiciel dédié au secteur agroalimentaire propose des modules spécifiques tels que la traçabilité des lots, la gestion des dates limites de consommation (DLC) et la gestion des recettes de fabrication, des fonctionnalités manquantes ou incomplètes dans un ERP généraliste.

Parmi les logiciels disponibles à l'échelle internationale, nous avons choisi « Konnect'AGRO » et « Sage X3 Food & Beverage » comme représentatifs de cette catégorie.

I.5.2.1. Konnect'AGRO

Konnect'AGRO, développé par la société française « AKANEA », est un ERP spécialisé dans l'industrie agroalimentaire. La figure 6 présente le logo de la société « AKANEA ».



Figure 6 : Le logo de l'entreprise française AKANEA [16]

a) Définition :

Konnect'AGRO est un ERP spécialisé dans le secteur agroalimentaire qui centralise et pilote les activités suivantes : gestion de la fabrication et de la production, gestion des commandes, réceptions, transport de marchandises et gestion commerciale, afin de répondre aux enjeux des entreprises, avec une traçabilité fiable intégrée [17]. Ce logiciel vise à répondre à l'ensemble des besoins et des contraintes des professionnels agroalimentaires, tout en optimisant et en simplifiant chaque étape des processus de production, de transformation et de distribution.

b) Fonctionnalités :

Les principales fonctionnalités du logiciel « Konnect'AGRO » sont les suivantes [17] :

- Établir une traçabilité fiable en suivant les lots et les flux du processus de production.
- Améliorer les processus de production avec planification des ressources, gestion des recettes et réduction des pertes.
- Assurer une gestion des stocks, gestion des approvisionnements, contrôle de la qualité des matières premières et des produits finis. Et gestion DLC.
- Gestion financière par le suivi des coûts, l'analyse des marges et l'aide à la prise de décisions.

c) Limites :

Konnect'AGRO est une solution très performante pour le secteur agroalimentaire, mais elle présente plusieurs limites importantes surtout pour des PME algériennes :

- Les coûts élevés de mise en œuvre du système, notamment les licences et le déploiement, sont souvent incompatibles avec les capacités financières des PME en Algérie.
- Ce système n'est pas disponible sur le marché algérien, tel qu'aucun intégrateur ni support technique n'est présent en Algérie, rendant son installation et sa maintenance pratiquement impossibles pour une entreprise locale.
- Ce système n'est pas conforme à la réglementation algérienne, comme le système comptable financier algérien (SCF), la Taxe sur la Valeur Ajoutée (TVA), la Taxe sur l'Activité Professionnelle (TAP), etc., ce qui nécessite des coûts additionnels.

La figure 7 montre un aperçu du logiciel « Konnect'AGRO ». Illustrant ses principales fonctionnalités de gestion



Figure 7 : un aperçu du logiciel Konnect'AGRO [17]

I.5.2.2. Sage X3 Food & Beverage

Sage X3 Food & Beverage est une solution ERP développée par l'éditeur britannique Sage Group. La figure 8 présente le logo de l'ERP Sage X3.



Figure 8 : Le logo du groupe Sage [18]

a) Définition :

Sage X3 Food & Beverage est un ERP spécialisé dans le secteur agroalimentaire. C'est un logiciel qui gère l'ensemble des opérations métier de l'entreprise et propose des fonctions performantes de gestion de la qualité, qui permettent aux entreprises de l'agroalimentaire de surveiller et de contrôler la qualité tout au long du processus de production. Ainsi, il contribue à assurer la conformité, la traçabilité et le respect de la réglementation du secteur [19].

b) Fonctionnalités :

Sage X3 offre un ensemble de fonctionnalités spécialisées pour les besoins spécifiques de l'industrie agroalimentaire. Ces fonctionnalités sont les suivantes [\[19\]](#) :

- Gestion de la chaîne d'approvisionnement, collaboration et assurance de la conformité dans le travail avec les fournisseurs.
- Gestion des entrepôts et stocks, traçabilité des stocks alimentaires et rentabilité des produits.
- Gestion commerciale et marketing : combinaison du CRM avec la puissance des applications mobiles pour libérer le potentiel des activités commerciales et marketing.
- Support des solutions pour la sécurité alimentaire, la traçabilité, la gestion des rappels.
- Gestion avancée de la production, du contrôle qualité et du suivi des lots.

c) Limites :

Il existe des contraintes qui rendent l'adoption de Sage X3 par les PME algériennes difficile. Parmi ces limitations, on peut citer :

- Le coût très élevé et ce niveau d'investissement sont largement incompatibles avec les capacités financières des PME algériennes.
- Complexité de mise en œuvre et d'utilisation.
- Sage X3 ne se conforme pas au cadre réglementaire algérien.
- Facturation en devises étrangères, ce qui pose des difficultés aux entreprises algériennes soumises aux restrictions de change imposées par la Banque d'Algérie.

La figure 9 présente le tableau de bord du logiciel Sage X3, qui offre une vue d'ensemble des modules ERP, des commandes, de l'inventaire, de la production, des ventes et des activités commerciales

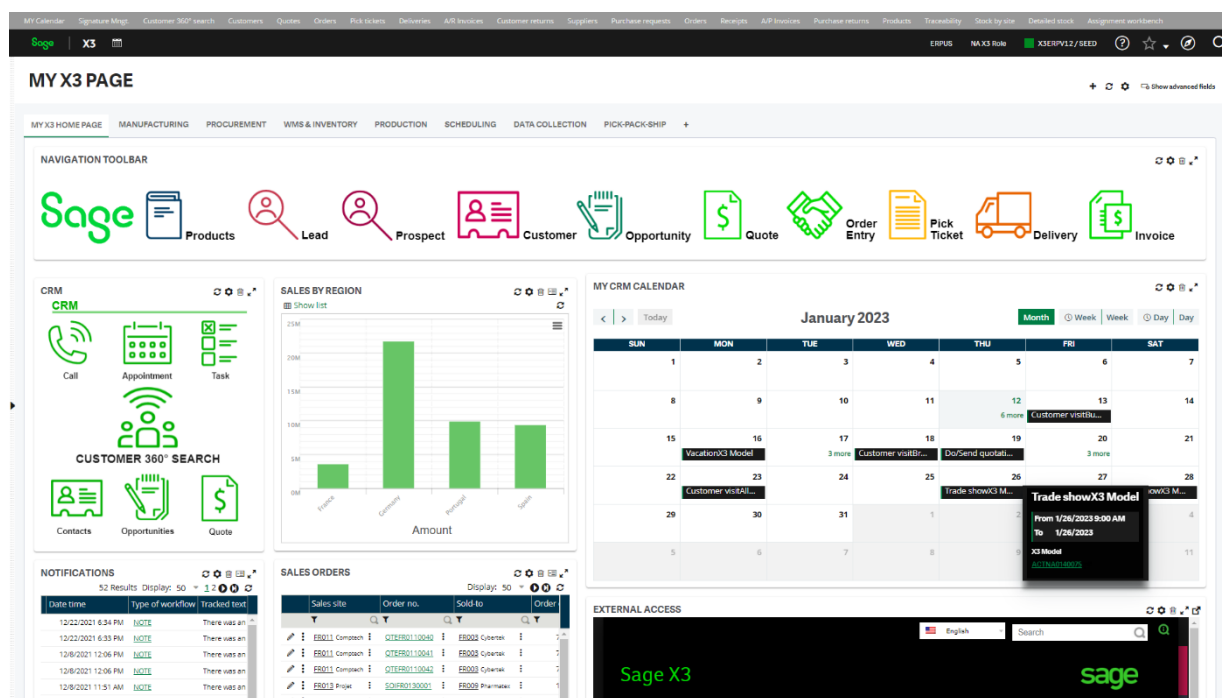


Figure 9 : Tableau de bord du logiciel Sage X3 [20]

I.5.3. Analyse critique des application existantes

À partir de l'analyse des systèmes existants sur le marché, nous avons identifié et étudié quatre logiciels, répartis en deux catégories. Les deux premiers logiciels « SYNAPSE ERP » et « GestiumERP », sont des solutions disponibles localement sur le marché algérien et développées par des sociétés algériennes ce qui est un avantage. Mais leur inconvénient majeur est leur caractère généraliste : ils ne possèdent pas de modules spécifiques adaptés au secteur de la confiserie, ce qui nécessite des personnalisations et engendre des coûts supplémentaires en termes de budget et de temps pour répondre aux besoins spécifiques de ce secteur.

Les deux derniers logiciels, « Konnect'AGRO » et « Sage X3 Food & Beverage », appartenant à la deuxième catégorie, à savoir les logiciels dédiés à l'industrie agroalimentaire, sont des solutions internationales. Ils offrent des fonctionnalités avancées, adaptées au secteur de l'agroalimentaire. Mais il existe un obstacle majeur : leur adaptation et leur disponibilité en Algérie. Le logiciel « Konnect'AGRO » est totalement absent du marché algérien. Par contre, le logiciel « Sage X3 » est partiellement accessible via l'intégrateur officiel SIG [21], mais son coût d'implémentation demeure élevé. En outre, ces deux solutions ne sont pas conformes au cadre réglementaire algérien, ce qui entraîne des coûts supplémentaires pour s'adapter à ces exigences nationales.

Pour résumer, il n'existe aucune solution qui allie la spécialisation dans le secteur de la confiserie, la disponibilité locale, la conformité réglementaire et un coût abordable pour les PME. C'est pourquoi nous avons conçu notre solution « SweetStock » pour combler ces lacunes.

I.6. Étude comparative détaillée des logiciels existants

Le tableau 1 présente un tableau comparatif mettant en évidence les différences entre les logiciels suivants : « SYNAPSE ERP », « GestiumERP », « Konnect'AGRO », « Sage X3 » et le logiciel développé dans ce projet, « SweetStock ». Il est possible de comparer ces solutions existantes selon les critères suivants : type, gestion des stocks, gestion de la production, traçabilité des batches, gestion des achats/fournisseurs, gestion des dates limites de consommation (DLC), gestion des recettes, ergonomie, coût, complexité, conformité aux systèmes comptables financiers (SCF) algériens, adaptabilité aux petites et moyennes entreprises (PME), qui représentent les besoins fonctionnels des entreprises.

Critère	SYNAPSE ERP	GestiumERP	Konnect'AGRO	Sage X3	SweetStock
Type	Généraliste algérien		Dédié agroalimentaire		Dédié confiserie DZ
Gestion des Stocks	✓	✓	✓	✓	✓
Gestion de production	✓	✓	✓	✓	✓ Dédiée
Traçabilité des batches	✓ Partielle	✓	✓	✓	✓
Gestion des achats /fournisseurs	✓	✓	✓	✓	✓
Gestion DLC	✗	✓	✓	✓ Via module	✓

Gestion des recettes	✗	✗	✓	✓	✓ Confiserie
Ergonomie	✓ Moyenne	✓ Moyenne	✓ Moderne	✓ Moderne	✓ Simple
Coût	✓ Abordable	✓ Abordable	✗ élevé	✗ Très élevé	✓ Faible
Complexité	✓ Faible	✓ Faible	✓ Moyenne	✗ élevé	✓ Faible
Conformité SCF Algérien	✓	✓	✗	✗	✓
Adaptabilité PME	✓	✓	✗	✓ Partiellement	✓

Tableau 1 : Tableau comparatif des logiciels de gestion

D'après le tableau comparatif qui met en évidence les différences entre les logiciels suivants : « SYNAPSE ERP », « GestiumERP », « Konnect'AGRO », « Sage X3 » et « SweetStock ». On peut analyser de manière critique les solutions existantes au regard des critères de ce tableau.

Pour les fonctionnalités métier en général, tous les logiciels analysés répondent aux besoins de base suivants : la gestion des stocks, la planification de la production, les inventaires et la gestion des achats et des fournisseurs. Bien qu'il y ait des différences qui apparaissent selon des critères métier spécifiques. La gestion des DLC est absente de « SYNAPSE ERP », tandis que « GestiumERP », « Konnect'AGRO » et « Sage X3 » la proposent nativement. La gestion des recettes de fabrication, indispensable dans le secteur de la confiserie, n'est disponible que dans « Konnect'AGRO », « Sage X3 » et « SweetStock ».

Pour la gestion des stocks, toutes les solutions permettent un suivi en temps réel des matières premières et des produits finis. « SYNAPSE ERP » et « GestiumERP » proposent un suivi avancé incluant la gestion de plusieurs sites, tandis que « Konnect'AGRO » et « Sage X » offrent une traçabilité avancée des lots, avec gestion des DLC. Pour « Sweetstock », il intègre également une gestion de multi-dépôts (Oran, Tlemcen) avec un suivi en temps réel adapté aux besoins d'une PME de confiserie algérienne par rapport aux autres solutions.

En ce qui concerne l'ergonomie et la complexité, critères non fonctionnels, les trois logiciels « SYNAPSE ERP », « GestiumERP » et « Konnect'AGRO » se distinguent par une interface moderne et intuitive, ainsi qu'une ergonomie moyenne à moderne, tandis que « Sage X3 » présente une interface complexe à maîtriser, avec une ergonomie moderne. Et pour « SweetStock », simple à utiliser et rapide à prendre en main.

Pour la disponibilité en Algérie, « SYNAPSE ERP » et « GestiumERP » sont des solutions 100% algériennes, disponibles sur le marché algérien, accessibles et conformes au système SCF algérien. En revanche, le logiciel « Konnect'AGRO » est absent du marché algérien, ce qui constitue un obstacle majeur. Sage X3 est partiellement accessible auprès de l'intégrateur Système d'Information Géographique (SIG) [\[21\]](#), mais demeure coûteux.

En bref, cette analyse montre que les logiciels ERP actuels répondent aux principaux besoins. Mais ils sont loin de proposer des solutions adaptables et flexibles à faible coût. C'est ce qui permet la création de nouveaux systèmes plus efficaces capables de répondre aux besoins des PME.

I.7. Positionnement de la solution proposée SweetStock

À partir de l'analyse comparative des systèmes existants que nous avons réalisée, ceux-ci offrent des fonctionnalités avancées, mais présentent certaines limites, notamment en termes de coût et d'adaptabilité aux besoins des entreprises. Ainsi, cette situation démontre l'urgence de trouver une solution simple, plus flexible et adaptable.

Dans ce contexte, « SweetStock » va, contrairement aux ERP existants, viser à résoudre ces limitations, à proposer une solution plus adaptée et développée pour répondre aux besoins réels et concrets d'une usine de confiserie industrielle spécialisée dans la fabrication de « chewing-gum », cliente de l'entreprise Naltis. La valeur ajoutée par notre solution réside donc dans sa facilité d'utilisation, son faible coût et son adaptation aux besoins spécifiques des PME industrielles du secteur agroalimentaire.

I.8. Conclusion

Dans ce chapitre, nous avons défini la transformation numérique et en avons souligné l'importance. Nous avons également présenté la notion d'ERP, en distinguant les logiciels généralistes (les ERP généralistes) des logiciels dédiés (les ERP spécialistes) à un secteur. Ensuite, nous avons listé quatre logiciels existants : « SYNAPSE ERP », « GestiumERP », « Sage X3 Food & Beverage » et « Konnect'AGRO », et réalisé une analyse comparative de ces systèmes, ce qui nous a permis d'identifier leurs points forts et leurs points faibles. Cela nous a permis de montrer qu'aucune de ces solutions ne répond pleinement aux besoins d'une PME de confiserie en Algérie. Les logiciels généralistes présentent un manque de spécialisation métier, tandis que les solutions dédiées, qui sont internationales, ne sont pas totalement accessibles. Cela nous encourage à développer notre solution locale « SweetStock » conçue pour le secteur de la confiserie en Algérie.

Dans le prochain chapitre, nous allons définir en détail les acteurs de notre système, leurs besoins, ainsi que la spécification des exigences fonctionnelles et non fonctionnelles de notre système proposé. Cette étape nous permettra de mettre en œuvre des diagrammes et de concevoir solidement notre système.

Chapitre II : Spécification des besoins et exigences de l'application « SweetStock »

II.1. Introduction

L'analyse des besoins et des exigences constitue l'étape clé du cycle de vie du développement logiciel. Elle permet de traduire les besoins et d'identifier clairement les fonctionnalités du système, ses utilisateurs et les conditions de fonctionnement, afin de prévenir les risques courants liés aux projets, tels que le manque de ressources, le dépassement du budget ou la sous-estimation des délais, et d'améliorer la prise de décision, le flux de travail ainsi que les performances des employés. De plus, investir du temps dans cette phase augmente significativement les chances de succès du projet [\[22\]](#).

Dans ce chapitre, nous allons procéder à une analyse approfondie des besoins des utilisateurs et à la spécification des exigences du système SweetStock. Nous identifions d'abord les différents acteurs interagissant avec le système et définissons les « User Stories » pour chaque acteur, qui exprime ses besoins fonctionnels. Ensuite, nous détaillons les spécifications des exigences fonctionnelles et non fonctionnelles. Enfin, nous modélisons ces besoins à l'aide de diagrammes UML ; nous avons adopté une approche basée sur quatre diagrammes UML : les diagrammes de cas d'utilisation, de séquence et d'activité sont présentés dans ce chapitre, tandis que le diagramme de classe concerne la modélisation de la structure statique du système, sera présenté dans le prochain chapitre. Cette approche étend l'approche du processus unifié (UP) qui utilise trois diagrammes seulement, en ajoutant le diagramme d'activités pour illustrer les flux complexes dans le système afin de poser les bases solides de la conception de notre système.

II.2. Identification des acteurs

Un acteur désigne toute entité extérieure au système modélisé qui interagit directement avec ce système [\[23\]](#). Que ce soit un utilisateur humain ou un système tiers, les principaux acteurs sont les utilisateurs du système. L'identification des acteurs est une phase cruciale, car elle permet de définir le champ fonctionnel du système et d'attribuer les droits d'accès et les responsabilités à chaque entité [\[24\]](#).

Les acteurs de l'application « SweetStock » sont :

➤ L'administrateur

L'administrateur est responsable de la gestion globale du système. C'est lui qui gère les utilisateurs, les droits d'accès et les référentiels de base, les recettes, supervise l'ensemble des opérations et consulte les statistiques via le tableau de bord. Il dispose d'un accès complet à tous les modules du système via une interface web complète.

➤ **Le magasinier**

Le magasinier est également le gestionnaire des stocks. C'est lui qui assure la gestion physique des produits, des inventaires, des entrées et des sorties de stock, le suivi des niveaux de stock et des DLC, la gestion des commandes fournisseurs et des réceptions, ainsi que des retours fournisseurs. Il dispose d'un accès dédié au stock et à la logistique.

➤ **L'atelier**

L'atelier ou l'opérateur de production. Il est responsable du cycle de production ; il assure la gestion des batches, des sessions de dragéification et du conditionnement, facilite la gestion des flux de travail entre les étapes de production et contribue à garantir une traçabilité complète des produits.

II.3. Description des besoins (user stories)

À partir des besoins de l'usine destinataire, et identifiés lors de nos échanges avec l'entreprise « Naltis », nous avons rédigé les user stories suivantes, qui serviront de base à l'identification des exigences fonctionnelle dans l'étape suivante.

Les user stories sont rédigées du point de vue de l'utilisateur final, et non de l'équipe de développement. Cela permet de se concentrer sur les besoins et les objectifs de l'utilisateur. Par conséquent, les développeurs peuvent fournir les fonctionnalités nécessaires pour répondre à ces besoins et offrir la meilleure expérience utilisateur possible.

Les user stories sont exprimées selon le format standard : « En tant que [rôle], je veux [action] afin de [bénéfice ou résultat] » [\[25\]](#).

II.3.1. Besoins liés à l'administrateur

Les besoins liés à l'administrateur sous forme de « user stories » sont les suivants :

- En tant qu'administrateur, je veux gérer les utilisateurs afin de contrôler l'accès au système.
- En tant qu'administrateur, je veux gérer les droits d'accès par rôle afin de sécuriser les opérations du système.
- En tant qu'administrateur, je veux gérer les référentiels de base (types de produits, etc.) du système afin d'adapter l'application.
- En tant qu'administrateur, je veux consulter le tableau de bord financier afin de piloter la rentabilité de la production.
- En tant qu'administrateur, je veux consulter les clients et gérer leurs commandes afin de piloter les ventes.

II.3.2. Besoins liés au magasinier

Les besoins liés au magasinier sous forme de « user stories » sont les suivants :

- En tant que magasinier, je veux gérer et consulter les produits afin d'assurer une gestion fiable des stocks.
- En tant que magasinier, je veux gérer les commandes fournisseurs afin d'assurer l'approvisionnement en matières premières.
- En tant que magasinier, je veux enregistrer les réceptions fournisseurs (entrée de stock) afin de mettre à jour le stock.
- En tant que magasinier, je veux enregistrer les sorties de stock (production, pertes, périmés, etc.) afin de mettre à jour et de tracer le stock.
- En tant que magasinier, je veux consulter les niveaux de stock en temps réel afin d'éviter les ruptures de stock.
- En tant que magasinier, je veux suivre les dates de péremption des produits (DLC) afin de réduire les pertes.
- En tant que magasinier, je veux gérer les fournisseurs et les retours fournisseurs afin de corriger les anomalies de réception.
- En tant que magasinier, je veux effectuer des inventaires afin de comparer les stocks réels et théoriques.

II.3.3. Besoins liés à l'atelier

Les besoins liés à l'atelier sous forme de « user stories » sont les suivants :

- En tant qu'atelier, je veux consulter les recettes afin de connaître les ingrédients et les quantités nécessaires avant de lancer un batch.
- En tant qu'atelier, je veux gérer et lancer les batches afin de démarrer la production.
- En tant qu'atelier, je veux vérifier la disponibilité des ingrédients avant de lancer un batch afin d'éviter toute interruption.
- En tant qu'atelier, je veux mettre à jour le statut d'un batch afin de suivre l'avancement de la production en temps réel.
- En tant qu'atelier, je veux que la validation d'un batch génère automatiquement les mouvements de consommation des matières premières afin d'assurer la traçabilité.
- En tant qu'atelier, je veux créer une session de dragéification liée à un batch afin d'enregistrer les passages.
- En tant qu'atelier, je veux gérer le conditionnement des produits afin de finaliser le cycle de fabrication.

II.4. Spécification des exigences fonctionnelles et non-fonctionnelles

Dans le cycle de développement logiciel, les exigences constituent la base de tout le système. Elles définissent les spécifications qui guident les développeurs dans la création d'un produit répondant aux besoins des utilisateurs et aux objectifs métier. Ces exigences sont généralement divisées en deux catégories : fonctionnelles et non fonctionnelles [26].

Nous nous sommes appuyés sur le cahier des charges fourni par l'entreprise « NALTIS » pour identifier les exigences fonctionnelles et non fonctionnelles décrivant les fonctionnalités, les qualités et les contraintes de notre système.

II.4.1. Spécification des exigences fonctionnelles

Les exigences fonctionnelles (FR, signifie en anglais Functional Requirements) constituent un ensemble de spécifications qui définissent ce qu'est un système. Une application logicielle doit répondre aux besoins et aux attentes de ses utilisateurs. Ces exigences décrivent le comportement, les caractéristiques et les fonctionnalités du système.

Nous allons identifier les exigences fonctionnelles décrivant les fonctionnalités de notre système. Elles sont organisées par modules.

➤ **Module A : Gestion des utilisateurs**

Au début, l'application doit permettre l'authentification des utilisateurs pour accéder aux différentes fonctionnalités. Elle doit également :

- FR-01 : supporter le contrôle d'accès basé sur les rôles.
- FR-02 : permettre à l'administrateur de créer, modifier, activer et désactiver les comptes des utilisateurs.
- FR-03 : permettre à l'administrateur de configurer des droits d'accès spécifiques pour chaque rôle.
- FR-04 : journaliser toutes les actions des utilisateurs dans un journal d'audit.

➤ **Module B : Gestion des produits**

Le système doit permettre de :

- FR-05 : Consulter, créer, modifier et archiver (activer et désactiver) des produits.
- FR-06 : Associer chaque produit à un type (matière première, semi-fini, fini), à une catégorie et à une unité de mesure.

- FR-07 : Gérer et associer un conditionnement à un produit.
- FR-08 : Activer ou désactiver la gestion de la date de péremption par produit (DLC).
- FR-09 : Rechercher et filtrer des produits par type, catégorie, statut et gestion de la date de péremption (activer ou désactiver).

➤ **Module C : Gestion des stocks**

Le système doit permettre de :

- FR-10 : Calculer automatiquement le stock réel à partir des mouvements de stock.
- FR-11 : Consulter et gérer les niveaux de stock en temps réel par produit et par emplacement de stockage.
- FR-12 : Gérer le stock multi-dépôts (Tlemcen, Oran).

➤ **Module D : Gestion des mouvements de stock**

Le système doit permettre de :

- FR-13 : Enregistrer automatiquement les entrées de stock.
- FR-14 : Enregistrer automatiquement les sorties de stock (production, pertes, décharges, etc.).
- FR-15 : Calculer automatiquement et représenter les mouvements de stock à travers une vue consolidée basée sur les données des différentes opérations (réceptions, batches, décharges, etc.).
- FR-16 : Assurer la traçabilité des mouvements de stock (date, type, produit, quantité).

➤ **Module E : Gestion des fournisseurs**

Le système doit permettre de :

- FR-17 : Consulter, créer, modifier, activer et désactiver les fournisseurs.
- FR-18 : Associer les fournisseurs aux commandes.

➤ **Module F : Gestion des commandes et réceptions fournisseurs**

Le système doit permettre de :

- FR-19 : Consulter, créer et modifier des commandes fournisseurs avec un ou plusieurs articles.
- FR-20 : Consulter, créer et modifier des réceptions fournisseurs avec un ou plusieurs articles (complètes ou partielles).

- FR-21 : Chaque réception doit être associée à une commande fournisseur.

➤ **Module G : Gestion des recettes**

Le système doit permettre de :

- FR-22 : Créer, modifier, consulter et désactiver (ou activer) des recettes.
- FR-23 : Associer une recette à un produit semi-fini et définir la quantité attendue.
- FR-24 : Rechercher et filtrer des recettes par type (batch ou dragéification).

➤ **Module H : Gestion des batches**

Ce module présente la première étape du processus de production. Un batch, également appelé préparation, consiste à mélanger les ingrédients (matières premières). Le système doit permettre de :

- FR-25 : Créer un batch à partir d'une recette de type batch, avec un code unique (auto-généré) et des dates de début et de fin.
- FR-26 : Vérifier la disponibilité et la date de péremption des matières premières avant de lancer un batch, en suivant l'algorithme FEFO (First Expire, First Out), soit « Premier Expiré, Premier Sorti » en français, qui consiste à consommer en priorité les lots (les matières premières réceptionnées ensemble partageant la même date de péremption) dont la date de péremption est la plus proche afin de minimiser les pertes liées à l'expiration des stocks.
- FR-27 : Suivre le statut des batches (planifiés, en cours, terminés, validés).
- FR-28 : Mise à jour automatique du stock en temps réel à chaque consommation ou retour.

➤ **Module I : Inventaires**

Le système doit permettre de :

- FR-29 : Créer et modifier des inventaires par emplacement.
- FR-30 : Comparer le stock théorique et le stock réel, puis calculer l'écart (quantité réelle – quantité théorique).

➤ **Module K : Dragéification (par Turbine)**

Ce module présente la deuxième étape du processus de production. La phase de dragéification est une technique d'enrobage consistant à recouvrir un noyau (le produit de type semi-fini), il est réalisé dans une machine appelée turbine. Cette opération comprend : l'enrobage, la coloration et le suivi climatique du produit. Le système doit permettre de :

- FR-31 : Créer, modifier et annuler une session de dragéification liée à une recette de type dragéification et à un batch spécifique.
- FR-32 : Chaque session doit permettre d'enregistrer plusieurs passages avec les mesures climatiques, les poids de noyau et les dates de début et de fin.
- FR-33 : Chaque passage doit permettre d'enregistrer plusieurs consommations des matières premières d'enrobage.

➤ **Module L : Conditionnement et Topping**

Ce module présente la dernière étape du processus de production : la phase de conditionnement, qui regroupe l'ensemble des opérations visant à envelopper, stocker et transporter un produit. Le système doit permettre de :

- FR-34 : Créer, modifier et annuler un ordre de conditionnement lié à une session de turbine terminée.
- FR-35 : L'atelier doit pouvoir enregistrer les résultats finaux : la quantité de boîtes produites, de dragées consommées et de déchets.
- FR-36 : calculer automatiquement le rendement de l'opération de conditionnement.

➤ **Module M : Gestion des clients et leurs commandes**

Le système doit permettre de :

- FR-37 : Créer, modifier et désactiver les clients.
- FR-38 : Associer les clients aux commandes.
- FR-39 : Consulter, créer et modifier des commandes des clients avec un ou plusieurs articles.

➤ **Module N : Traçabilité et historique**

Le système doit permettre de :

- FR-40 : assurer une traçabilité complète par batch, par produit et par date.
- FR-41 : Consulter l'historique des mouvements depuis une vue SQL unifiée et cohérente.
- FR-42 : maintenir un journal (Log) des actions des utilisateurs.

II.4.2. Spécification des exigences non-fonctionnelles

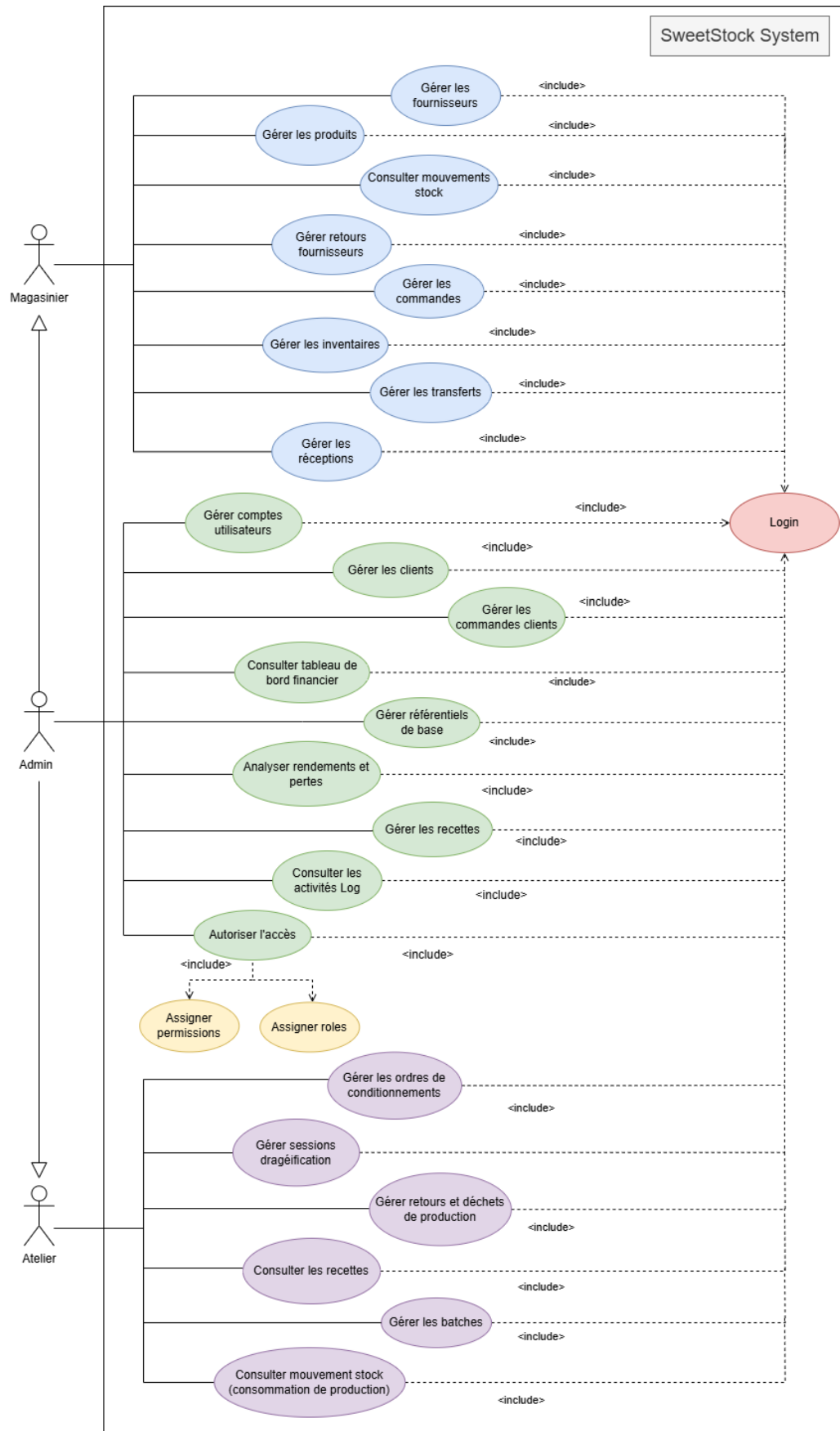
Les exigences non fonctionnelles (NFR, signifie en anglais Non-Functional Requirements) spécifient les normes et les qualités qu'un système doit respecter pour fonctionner efficacement, en se concentrant sur « comment » un système fonctionne plutôt que sur ce que nous faisons [26].

Les exigences non fonctionnelles de notre logiciel « SweetStock » se concentrent sur la performance, la sécurité, la disponibilité, l'ergonomie, la fiabilité, l'utilisabilité, la maintenabilité, l'évolutivité, l'intégration et la traçabilité afin d'assurer une expérience utilisateur optimale. Ces exigences sont expliquées dans ce qui suit :

- **Performance** : le logiciel doit garantir des temps de réponse rapides (< 3 s), réactifs et optimaux pour les interactions des utilisateurs, et être capable de supporter la charge liée aux accès simultanés des utilisateurs aux différents modules de gestion du système. Cela inclut le chargement rapide des pages, la récupération rapide des données, etc.
- **Sécurité et fiabilité** : le logiciel doit être sécurisé par une authentification forte, une gestion fine des droits et le chiffrement des données sensibles. En plus, il doit garantir la cohérence des données (par exemple : pas de modification directe des mouvements de stock).
- **Disponibilité** : le logiciel doit être accessible 24/7 et assurer une tolérance aux pannes afin de garantir la continuité des opérations sans perte de données.
- **Ergonomie et utilisabilité** : l'application doit offrir une interface utilisateur simple, fluide, intuitive et ergonomique pour faciliter la navigation et les interactions.
- **Maintenabilité et évolutivité** : le système doit être conçu de manière modulaire afin de faciliter l'ajout de nouvelles fonctionnalités ou de nouveaux modules. En plus, il doit être désigné pour faciliter l'entretien ou la maintenance en bon état de fonctionnement.
- **Traçabilité** : Retracer les exigences et les activités ou tâches au sein du système.

II.5. Diagramme des cas d'utilisation

Le diagramme de cas d'utilisation est un outil de modélisation appartenant aux diagrammes comportementaux de l'UML (Unified Modeling Language) [27]. Ce diagramme illustre les fonctionnalités proposées par le système sous forme de cas d'utilisation et il décrit comment les différents acteurs (humains ou non humains) interagissent avec le système [28]. La figure 10 ci-dessous présente le diagramme de cas d'utilisation de notre application « SweetStock » offrant un aperçu complet de sa structure, de ses fonctionnalités principales et de ses acteurs. Il met en lumière les interactions entre les acteurs et le système, tout en détaillant les fonctionnalités accessibles à chacun selon son rôle. Cela permet de visualiser explicitement l'organisation du système.

Figure 10 : Diagramme des cas d'utilisation de l'application « SweetStock » [Lien](#)

II.5.1. Diagramme de cas d'utilisation de l'administrateur

Dans cette partie, nous allons mettre en évidence les fonctionnalités de l'administrateur dans le diagramme de cas d'utilisation. La figure 11 ci-dessous présente la partie du diagramme de cas d'utilisation des fonctionnalités de l'administrateur.

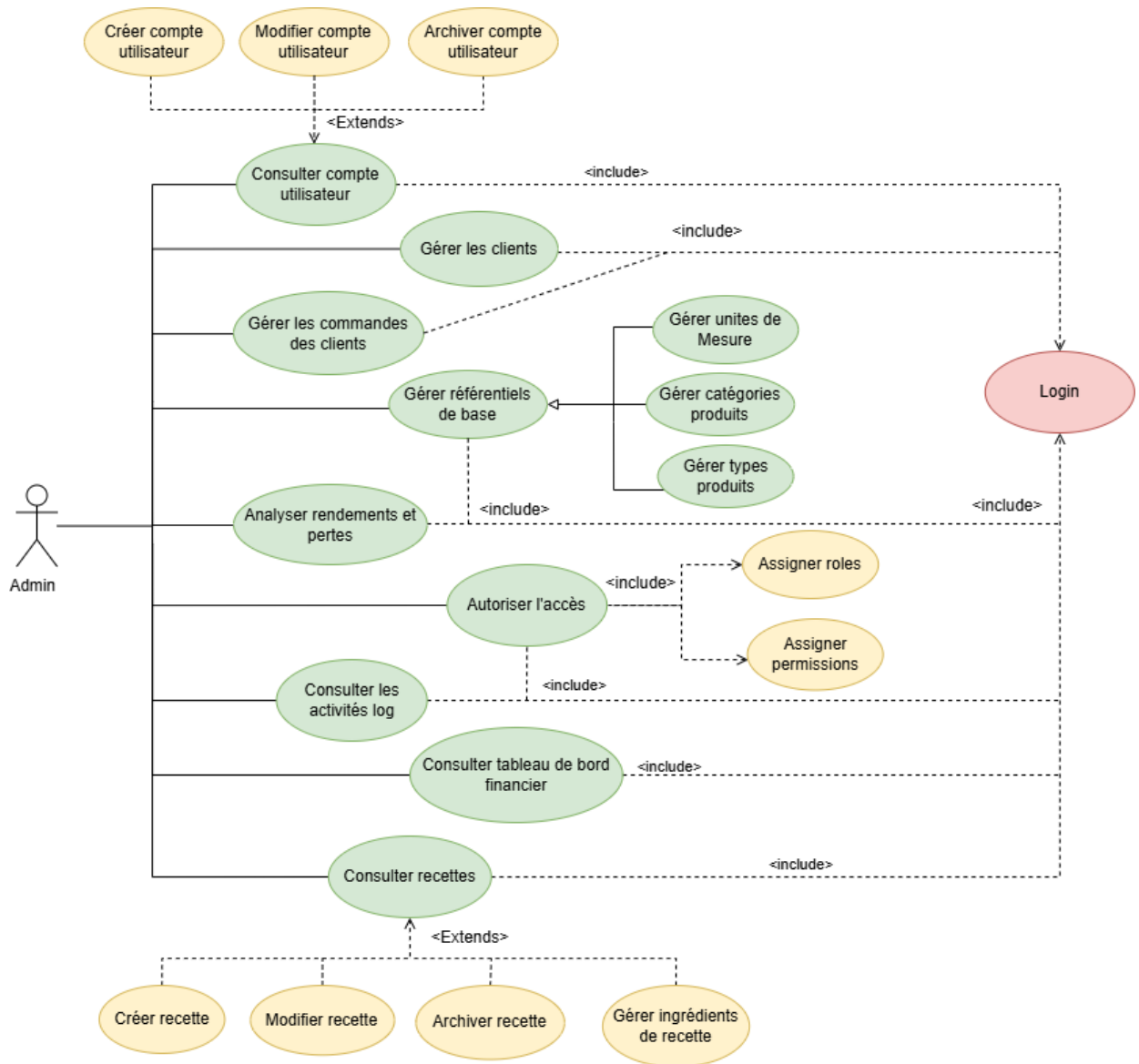


Figure 11 : Diagramme de cas d'utilisation des tâches de l'administrateur

L'administrateur dispose de droits étendus lui permettant de gérer les utilisateurs. Il peut créer, modifier et activer ou désactiver des comptes d'utilisateurs, assigner des rôles (magasinier ou atelier), ainsi qu'attribuer des autorisations en assignant les permissions à chaque rôle afin d'assurer un contrôle d'accès complet. Il peut consulter le tableau de bord, les statistiques et les logs d'activité, et gérer les référentiels de base du système : les unités de mesure et leurs familles de mesure, les conditionnements, les emplacements de stockage (les dépôts et les rayons), les types

et les catégories de produits, ainsi que les recettes (recettes batch et de dragéification). Toutes ces opérations nécessitent l'authentification de l'administrateur

II.5.2. Diagramme de cas d'utilisation du magasinier

Maintenant, nous portons notre attention sur le rôle clé du magasinier dans le diagramme de cas d'utilisation. La figure 12 ci-dessous présente le diagramme de cas d'utilisation des fonctionnalités du magasinier.

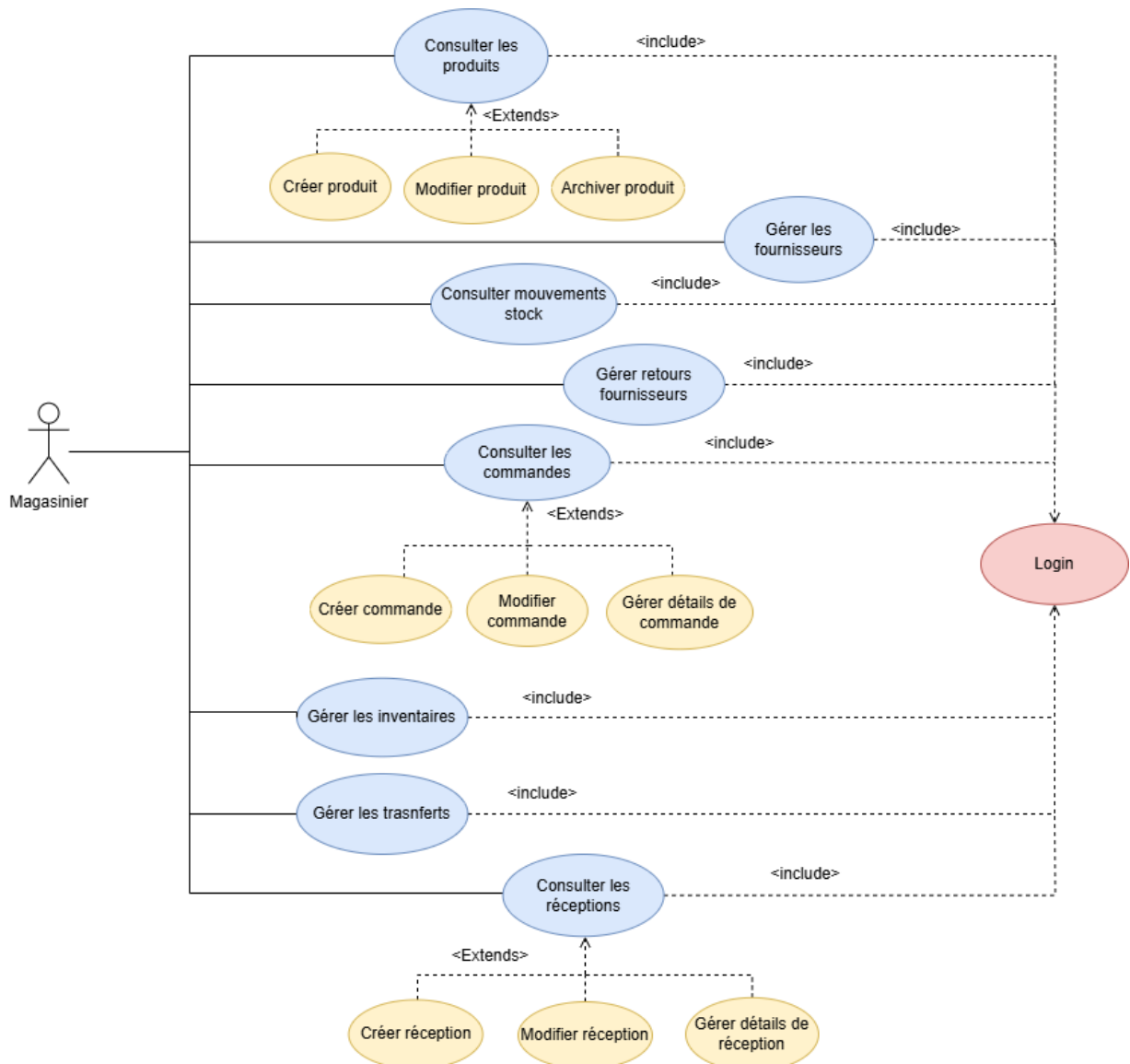


Figure 12 : Diagramme de cas d'utilisation des tâches du magasinier

Le magasinier supervise la gestion des produits ; il peut créer, modifier et consulter la liste de produits du système ; il attribue à chaque produit un type, une catégorie, une unité de mesure, divers conditionnements, ainsi que la gestion des fournisseurs, des commandes, des réceptions, des retours fournisseurs et toutes les opérations d'inventaire, et gère les transferts, les décharges

internes et externes et consulte des mouvements de stock, etc. Tout cela pour assurer une traçabilité complète des produits. Toutes ces opérations nécessitent l'authentification du magasinier.

II.5.3. Diagramme de cas d'utilisation de l'atelier

Pour le dernier acteur, qui est l'atelier, également responsable de la production, la figure 13 ci-dessous présente le diagramme de cas d'utilisation des fonctionnalités de l'atelier.

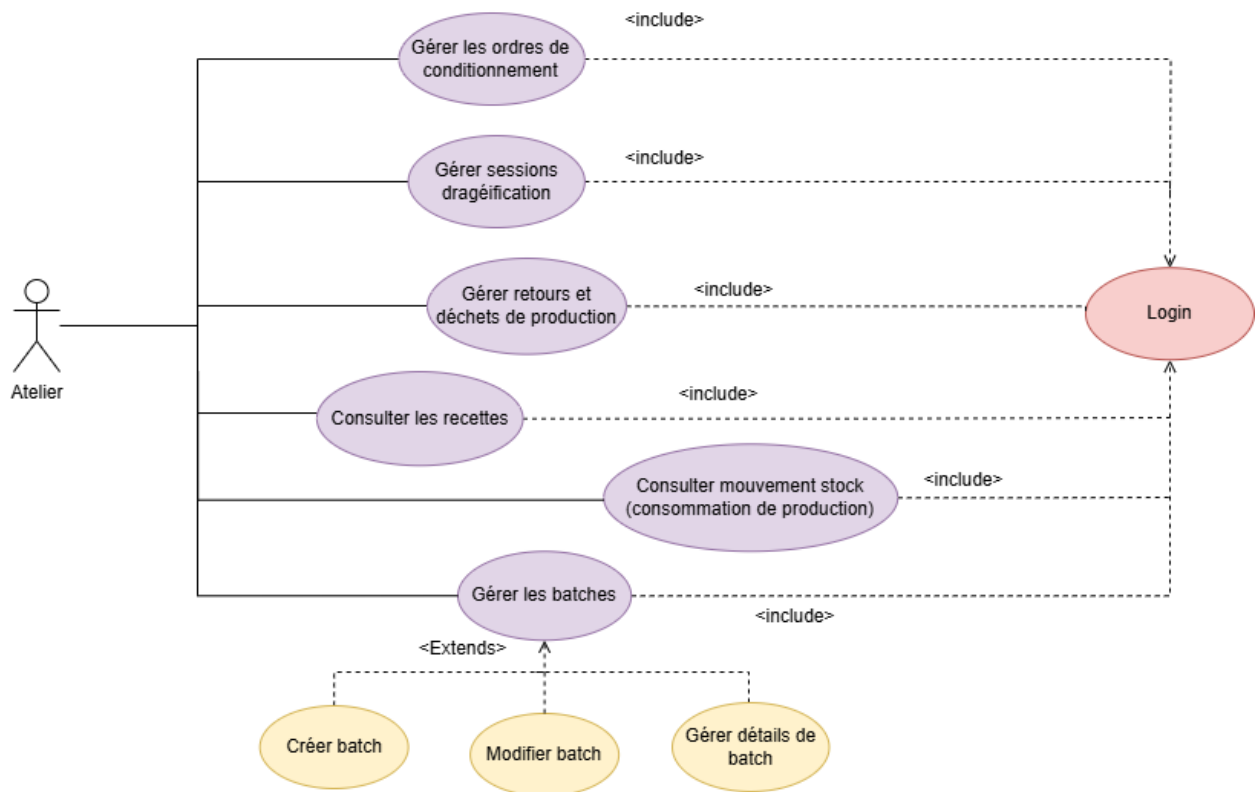


Figure 13 : Diagramme de cas d'utilisation des tâches de l'atelier

L'atelier est responsable du cycle de production. Il peut consulter la liste des recettes (il n'a pas le droit de les modifier ni de les créer), et il est chargé de la création, de la modification et du lancement des batches, ainsi que de la gestion des sessions de dragéification, ce qui nécessite de gérer les passages et les consommations de chaque passage, jusqu'à la phase de conditionnement, afin d'obtenir des produits finis. De même, il peut consulter les mouvements de stock liés aux processus de production et de consommation : la déduction des ingrédients dans les batches, ainsi que les entrées de produits sortant de chaque phase de production, afin d'assurer la traçabilité et la cohérence du cycle de production. Toutes ces opérations nécessitent l'authentification de l'atelier.

II.6. Scénarios nominaux et alternatifs des cas d'utilisation

Un scénario de cas d'utilisation est une description détaillée d'un cas d'utilisation spécifique, qui définit le comportement du système de manière séquentielle et fournit une description claire et concise de l'interaction entre l'acteur et le système pour atteindre un objectif précis. L'objectif est de garantir que toutes les personnes impliquées dans le développement et le test du système comprennent les exigences et le comportement attendu du système [29]. Il existe principalement trois scénarios : le scénario nominal, les scénarios alternatifs et les scénarios d'exception. Nous nous concentrerons sur deux types de scénarios : les scénarios nominaux, qui présentent le déroulement idéal ou le plus courant, et les scénarios alternatifs, qui incluent toutes les variations du chemin principal, ainsi que les erreurs et les exceptions [30].

Dans cette partie, nous présenterons les deux scénarios, à la fois nominaux et alternatifs, pour deux cas d'utilisation clés : la validation d'un batch et l'enregistrement d'une réception de commande fournisseur.

II.6.1. Scénario nominal et alternatif de validation d'un batch

Le cas d'utilisation de validation d'un batch permet à l'opérateur de vérifier sa conformité avant de passer à l'étape suivante de la production et de l'intégrer au système. Ce processus consiste à contrôler les informations de batch : les matières premières et leurs quantités, les quantités produites ainsi que les dates de production. Le batch est validé lorsque ces informations sont validées et conformes. Nous présenterons les deux scénarios pour ce cas d'utilisation dans les flux normal et alternatif, qui couvrent l'ensemble des situations possibles lors de l'exécution de processus.

- **Acteur principal :** L'atelier (ou l'opérateur de production).
- **Pré-conditions :**
 - Toutes les données de production : les quantités, les dates, les lots de consommation du stock sont saisis.
- **Post-conditions :**
 - Le batch passe au statut « validé ».

II.6.1.1. Scénario nominal de validation d'un batch

Ce scénario décrit le processus de validation d'un batch par l'atelier, dans le déroulement normal et sans erreur, depuis la création et le lancement du batch jusqu'à la fin de l'opération avec succès, lorsque toutes les conditions sont réunies (la recette est disponible, le stock est suffisant et non périmé).

- **Étape 1** : au début, l'atelier accède au module batch et clique sur « Nouveau batch ». Le formulaire de création s'affiche.
- **Étape 2** : l'atelier saisit les informations du batch et sélectionne une recette existante. Le système crée le batch avec statut « planifié » et charge automatiquement les ingrédients et les quantités théoriques associée à la recette.
- **Étape 3** : l'atelier clique sur « Lancer batch » et le système lance d'abord une vérification de la disponibilité et de la non-péremption des ingrédients.
- **Étape 4** : après vérification, s'il est valide, le système lance le batch et passe le statut à « en cours ».
- **Étape 5** : L'atelier saisit les quantités réelles prélevées sur le stocke pour chaque ingrédient. Le stock est mis à jour en temps réel.
- **Étape 6** : L'atelier saisit les retours et les déchets éventuels.
- **Étape 7** : L'atelier clôture le batch et saisit la quantité réelle du produit sortant. Le système met à jour le stock en conséquence et passe le statut à « terminé ».
- **Étape 8** : l'atelier valide le batch ; le statut devient « validé » pour passer à la phase suivante de production.

II.6.1.2. Scénario alternatif de validation d'un batch

Ce scénario couvre les déviations à chaque étape du processus de validation d'un batch. Le système signale alors les erreurs à l'atelier, qui doivent être corrigées avant de relancer la validation.

- **A1.** Le stock est insuffisant ou périmé ; le système bloque le lancement, affiche un message d'erreur pour informer l'atelier et le statut du batch reste à « planifié ».
- **A2.** L'opérateur repère une erreur de saisie des données ou un oubli à l'**étape 2**. Il clique sur « Modifier » pour modifier les données du batch, puis il enregistre les corrections, et le système valide la cohérence des nouvelles données.

II.6.2. Scénario nominal et alternatif de réalisation d'un inventaire

Ce cas d'utilisation permet au magasinier de réaliser un inventaire physique du stock d'un dépôt. Le système génère automatiquement les lignes d'inventaire à partir du stock théorique en temps réel, permettant au magasinier de saisir les quantités réellement comptées. Nous présenterons les deux scénarios, à la fois nominal et alternatif :

- **Acteur principal :** Magasinier (Gestionnaire de stock).
- **Pré-conditions :**
 - La dépôt et l'exercice en cours sont configurés dans le système.
- **Post-conditions :**
 - Les écarts entre le stock théorique et réel sont enregistrés et inventaire est « validé ».

II.6.2.1. Scénario nominal de réalisation d'un inventaire

Nous présentons le scénario nominal de réalisation d'un inventaire qui décrit l'enchaînement idéal de ce processus sans aucune anomalie.

- **Étape 1 :** Le magasinier accède au module Inventaires, clique sur « Créer un inventaire », sélectionne le dépôt et l'exercice en cours, puis enregistre. L'inventaire est alors créé avec le statut « brouillon ».
- **Étape 2 :** Le magasinier lance le comptage. Le système interroge le stock en temps réel et génère automatiquement les lignes d'inventaire avec les quantités théoriques. Le statut passe à « en cours ».
- **Étape 3 :** Ensuite, le magasinier saisit la quantité réelle pour chaque produit.
- **Étape 4 :** Le magasinier clôture le comptage. Le système vérifie que toutes les lignes sont complétées, calcule les écarts et met le statut à « terminé »
- **Étape 5 :** Le magasinier examine les écarts détectés, les analyse et valide l'inventaire. Le statut passe à « validé ».

II.6.2.2. Scénario alternatif de réalisation d'un inventaire

Ce scénario décrit les différents cas de réalisation d'un inventaire qui identifie les anomalies logistiques.

- **A1.** A l'étape 4, une ou plusieurs lignes n'ont pas de quantité réelle saisie. Le système bloque la clôture et affiche une alerte indiquant le nombre de lignes manquantes. Le magasinier doit compléter les saisies manquantes avant de relancer la clôture.

- **A2.** À l'étape 4, après la clôture du comptage, le système calcule automatiquement les écarts et affiche le nombre d'écarts détectés en rouge. Pour chaque ligne en écart, le champ « Observation » devient visible et obligatoire. Le responsable renseigne une explication puis valide l'inventaire.

II.7. Diagramme de séquence

Un diagramme de séquence UML est un outil de modélisation orienté objet qui décrit les interactions et les comportements dynamiques entre différents composants ou objets d'un système sur une période donnée. Il fournit une séquence des messages échangés entre les objets, ainsi que l'ordre de ces interactions, à l'aide de lignes de vie verticales et de flèches horizontales [31]. Ce type de diagramme permet de visualiser le comportement dynamique et le flux d'un système.

Pour ce projet, nous présentons en détail deux diagrammes de séquence UML illustrant deux fonctionnalités clés : la validation d'un batch et l'enregistrement d'une réception de commande fournisseur, correspondant aux deux principaux flux du système : la gestion de production et la gestion des entrées de stock. Ces scénarios mettent en évidence les étapes principales de chaque processus, en mettant l'accent sur l'interaction entre les acteurs (atelier ou magasinier) et le système. Ces diagrammes de séquence permettent de visualiser en détail les différentes actions et les messages échangés entre les acteurs et le système, offrant ainsi une compréhension claire de sa dynamique dans ces situations spécifiques. Les deux scénarios présentés sont :

II.7.1. Diagramme de séquence de validation d'un batch

Le processus débute lorsque l'atelier crée et lance un batch à partir d'une recette prédéfinie. Le système vérifie d'abord la disponibilité et la non-péréemption de tous les ingrédients nécessaires ; si le stock est insuffisant, le batch est automatiquement bloqué et un message est affiché pour informer l'atelier. Sinon, le batch est lancé avec succès ; l'opérateur saisit les quantités réelles de chaque ligne du batch (ingrédient, retour, déchet) ; le système met automatiquement à jour les stocks en temps réel. Le batch passe alors par les statuts « en cours », puis « terminé ». Le système va enregistrer l'entrée du produit sortant, augmenter sa quantité en stock, et l'atelier pourra passer le statut du batch à « validé ». La figure 14 ci-dessous présente le diagramme de séquence pour le scénario complet de validation d'un batch :

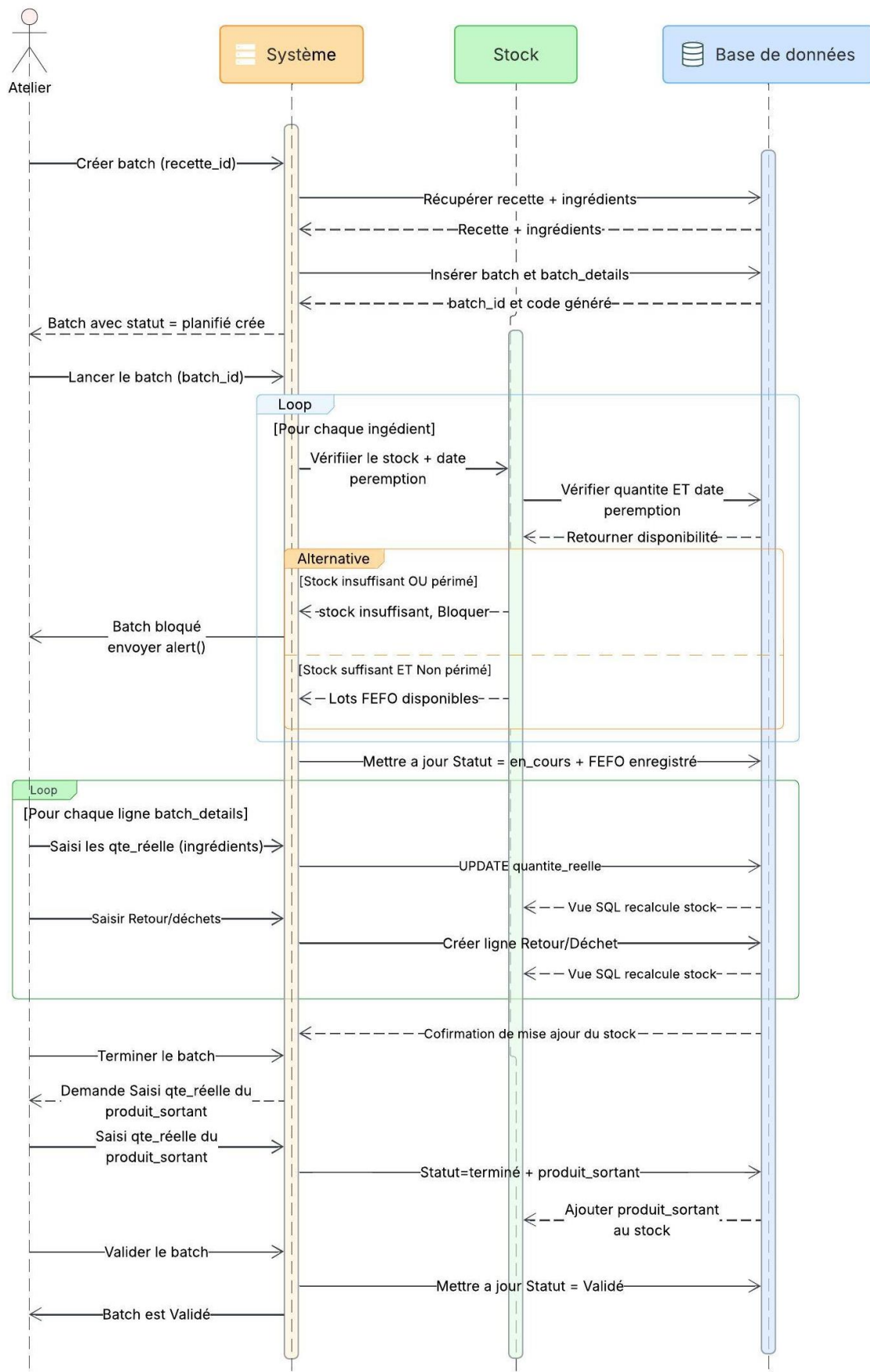


Figure 14 : Diagramme de Séquence système de validation d'un batch

Ce diagramme de séquence illustre l'ensemble des interactions entre les quatre acteurs essentiels du processus de Validation d'un batch : l'Atelier, le Système, le module Stock et la Base de données.

Le processus commence lorsque l'Atelier envoie une requête de création d'un batch avec la recette existante « recette_id ». Le système récupère la recette et ses ingrédients depuis la base de données, crée le batch et ses détails, puis renvoie le batch_id ainsi que le code généré automatiquement. Enfin, il informe l'atelier que le batch a été créé avec le statut « planifié ».

L'atelier lance le batch et le système déclenche une vérification pour chaque ligne du batch (ingrédient). Le système vérifie la quantité disponible et la date de péremption. Si le stock est insuffisant ou l'ingrédient est périmé, le batch est bloqué et une alerte est envoyée à l'atelier. Sinon, un message est affiché pour confirmer la disponibilité et le système fournit les lots FEFO disponibles.

Après validation de tous les ingrédients, le système met à jour le statut du batch en « en_cours » et enregistre les lots FEFO. Pour chaque ligne de batch_details, l'atelier saisit les quantités réellement prélevées (qte_réelles) du rayon (stock). Le système met également à jour la base de données et, via une vue SQL, recalcule le stock en temps réel. L'atelier saisit ensuite les retours et les déchets, s'ils existent. Le système crée les lignes correspondantes et le stock est recalculé.

Une fois le batch terminé, l'atelier envoie une demande de clôture du batch. Le système demande de saisir la quantité réelle produite dans ce batch pour le produit sortant, que l'atelier saisit. Le système passe le batch au statut « terminé » et ajoute la quantité du produit sortant au stock. Pour clôturer définitivement le batch, l'atelier le valide ; le système met à jour le statut en « validé » dans la base de données et confirme la validation à l'atelier.

II.7.2. Diagramme de séquence de l'enregistrement d'une réception de commande fournisseur

Le magasinier initie l'enregistrement d'une réception de commande fournisseur depuis la page dédiée aux réceptions. Pour créer une réception, le magasinier doit d'abord sélectionner une commande fournisseur existante, dont les lignes sont automatiquement récupérées et pré remplies dans le formulaire, avec possibilité de modification. Il peut également ajouter des lignes pour des produits absents ou supplémentaires. Lors de la validation, le système mettra à jour le stock réel en temps réel et enregistrera l'opération dans le journal d'audit. La figure 15 présente le diagramme de séquence pour cette fonctionnalité :

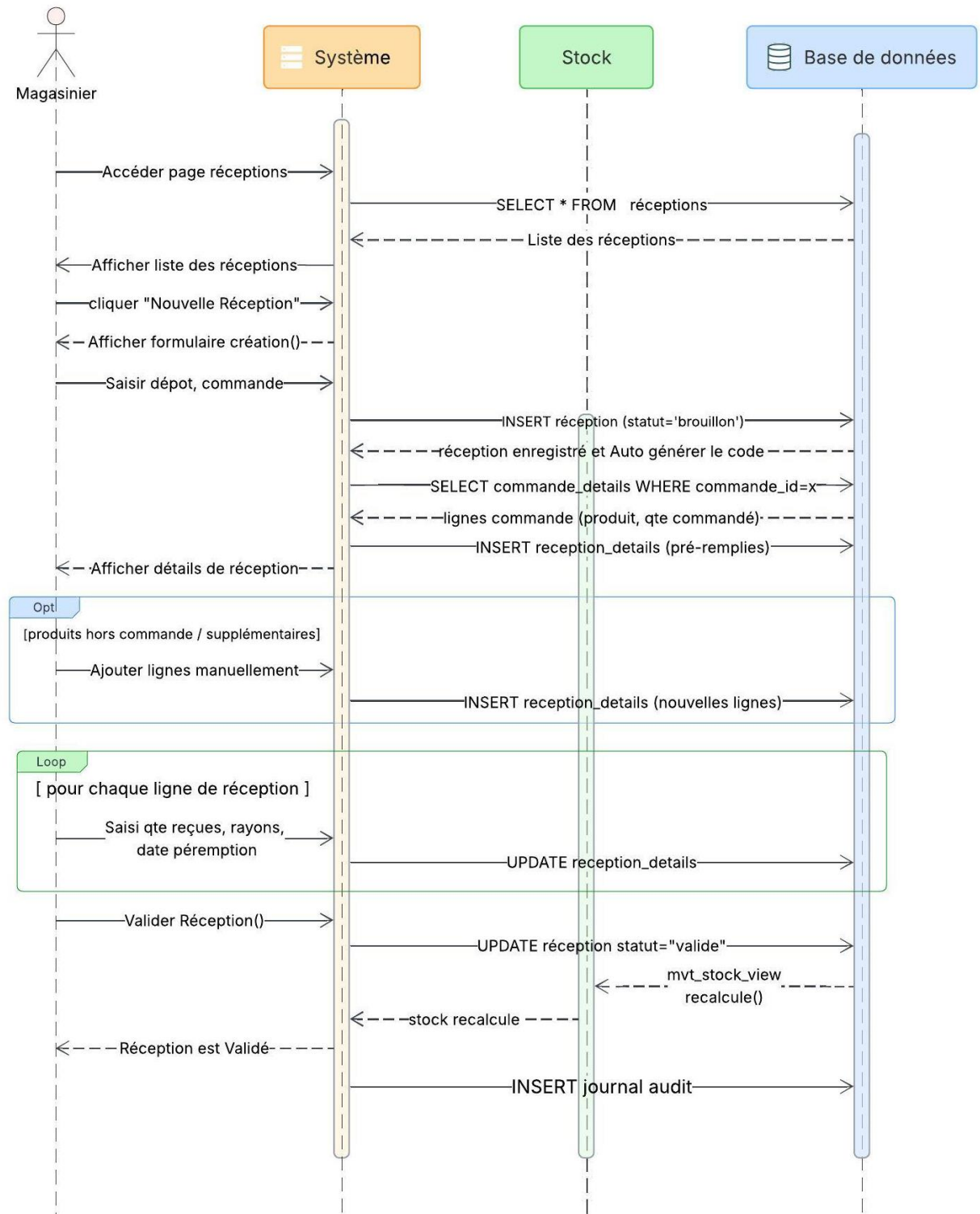


Figure 15 : Diagramme de Séquence système de l'enregistrement d'une réception de commande fournisseur

Ce deuxième diagramme présente le processus de réception d'une commande fournisseur et inclut quatre acteurs : le magasinier (utilisateur), le système, le module Stock et la base de données.

Le processus commence quand le magasinier accède à la page des réceptions. Le système renvoie automatiquement toutes les réceptions de la base de données et affiche la liste des réceptions existantes. Le magasinier envoie une requête pour créer une nouvelle réception, le système affiche le formulaire de création, puis le magasinier saisit les informations : le dépôt et la commande concernés.

Le système crée une nouvelle réception dans la base de données avec le statut « brouillon » par défaut, génère automatiquement un code de réception, puis récupère les lignes de la commande associée afin d'obtenir les produits et les quantités commandés. Ensuite, il insère les lignes de réception (réception_details) à partir de ces données et affiche les détails de la réception. S'il y a des produits hors commande ou supplémentaires, le magasinier peut ajouter manuellement de nouvelles lignes à ces réceptions, et le système les insère dans réception_details. Pour chaque ligne de réception, le magasinier saisit les quantités réellement reçues, les rayons et les dates de péremption. Le système met ensuite à jour les reception_details dans la base de données.

Pour clore le processus, le magasinier valide la réception et le système met à jour le statut en « validé ». La vue SQL mvt_stock_view recalcule automatiquement les stocks. Enfin, un enregistrement est ajouté au journal d'audit afin de tracer l'opération.

II.8. Diagramme d'activités

Un diagramme d'activités offre une représentation du comportement d'un système en détaillant la séquence d'actions d'un processus. Il ressemble à un organigramme de traitement de l'information. Il décrit les différentes activités, actions et décisions menées pour réaliser une fonctionnalité précise, ainsi que les transitions entre ces étapes. Les diagrammes d'activités peuvent, cependant, également montrer les flux parallèles simultanés et les flux de remplacement. Dans les diagrammes d'activités, on utilise des nœuds et des arcs d'activité pour modéliser le flux de commandes et de données entre les actions [\[32\]](#).

Pour notre projet « SweetStock », la figure 16 présente le diagramme d'activités du processus de validation d'un batch de production.

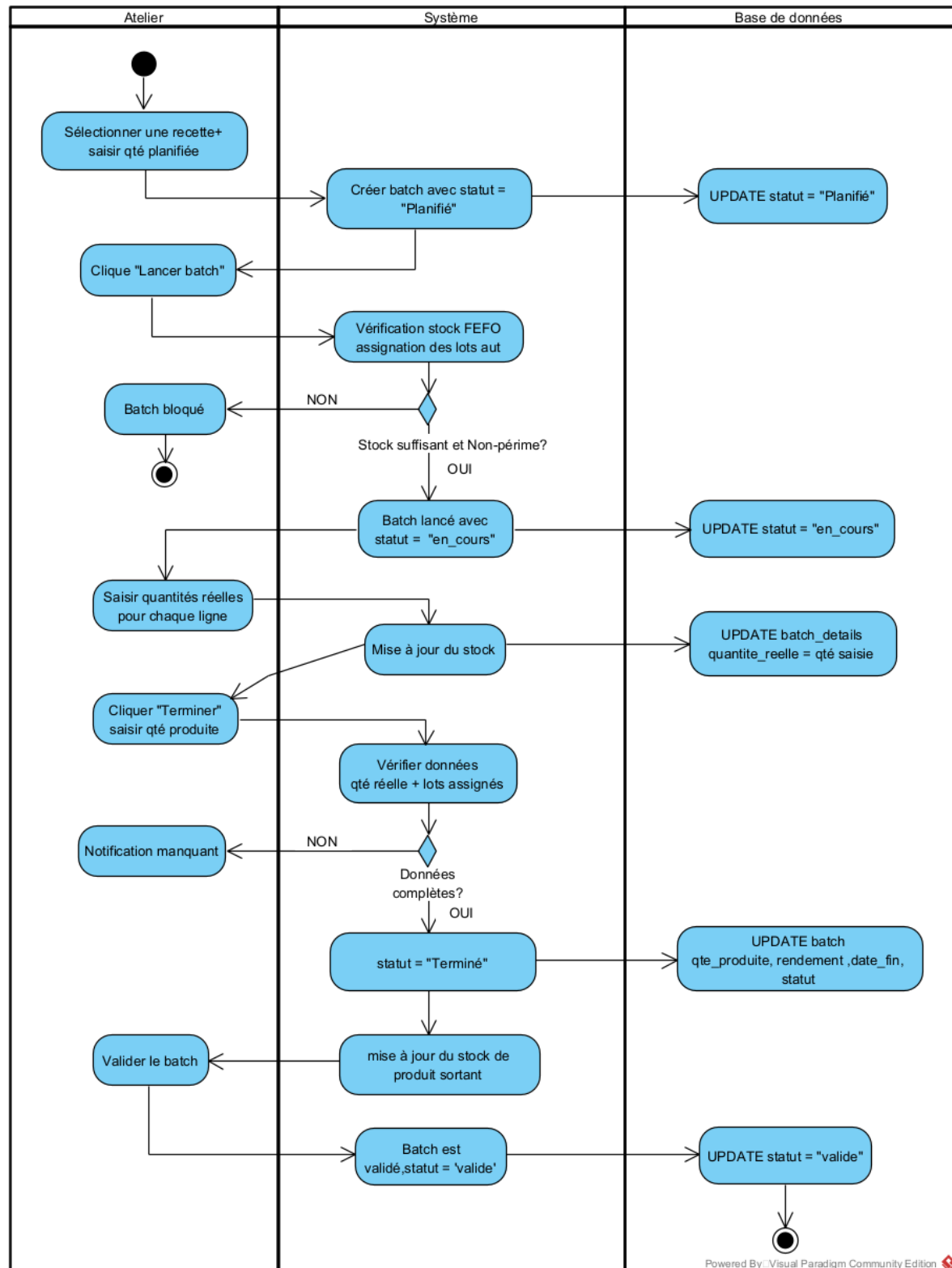


Figure 16 : Diagramme d'activités de validation d'un batch

Ce diagramme d'activités illustre le déroulement du processus de validation d'un batch, en mettant en évidence les flux de décisions et les transitions de statut.

Le processus commence par la sélection d'une recette et la saisie de la quantité planifiée. Le système crée alors un batch avec le statut « planifié ». L'étape suivante consiste à lancer le batch, ce qui déclenche une vérification automatique de la disponibilité et de la non-péréemption des ingrédients par le système. En cas d'échec de la vérification, le batch est bloqué ; dans le cas contraire, le statut passe à « en cours ».

L'atelier saisit les quantités réelles pour chaque ingrédient et le stock est mis à jour en conséquence. Ensuite, l'atelier clique sur « Terminer » et saisit la quantité réelle du produit fini. Si des données sont manquantes, le système les demande ; sinon, le statut passe à « terminé » et le stock du produit fini est ajusté. Enfin, l'atelier valide le lot, ce qui fait passer le statut à « validé ».

En bref, le diagramme d'activités illustre le flux de travail des états pour le traitement de validation d'un batch, il montre que le batch passe par quatre statuts : **planifié, en cours, terminé, valide**. Avec 2 points de contrôle (nœuds de décision) qui décident de bloquer ou de corriger le processus avant de l'avancer.

II.9. Conclusion

Dans ce chapitre, nous nous sommes concentrés sur l'analyse des besoins et la spécification des exigences du système. D'abord, nous avons identifié les acteurs principaux du système : l'administrateur, le magasinier et l'atelier, ainsi que leurs rôles. Ensuite, nous avons exprimé les besoins fonctionnels sous forme d'user stories afin d'identifier les exigences fonctionnelles et non-fonctionnelles. Par la suite, nous avons présenté les diagrammes de cas d'utilisation et détaillé les scénarios nominaux et alternatifs des cas d'utilisation les plus importants ainsi que les diagrammes de séquence et d'activités.

Cette analyse constitue une base solide pour la prochaine étape, qui portera sur la conception de l'architecture du système et de la base de données, ainsi que sur la modélisation de ses composants.

Chapitre III : Conception de l'application « SweetStock »

III.1. Introduction

La phase de conception est une étape essentielle du cycle de vie de développement d'un projet. Elle permet de définir l'architecture globale du système, ses composants et la structure des données. Dans cette phase, nous allons approfondir la conception de notre application « SweetStock ».

Pour notre projet « SweetStock », la phase d'analyse des besoins et spécification des exigences dans le chapitre précédant a clarifié les fonctionnalités attendues et les contraintes du système. Ce chapitre présente une vue d'ensemble complète de la conception de l'application et vise à assurer une mise en œuvre efficace et cohérente du système et offrir une base solide pour le développement, les tests et la maintenance futurs.

Dans ce chapitre, nous commençons par présenter l'architecture de l'application, qui montre comment les principaux composants de l'application sont organisés et interagissent. Ensuite, nous détaillons le diagramme de classe qui décrit la structure des données et les relations entre les différentes entités du système, puis nous définissons le modèle de données qui traduit ce diagramme en un schéma relationnel.

III.2. Architecture de l'application

L'architecture de notre application est fondée sur une organisation bien structurée, garantissant la modularité, maintenabilité et performance pour le système. Elle est conçue selon 3 tiers. Cette organisation permet une séparation claire entre les interfaces (les vues) et la logique métier et la gestion des données. La figure 17 ci-dessous présente une illustration de l'architecture de notre application.

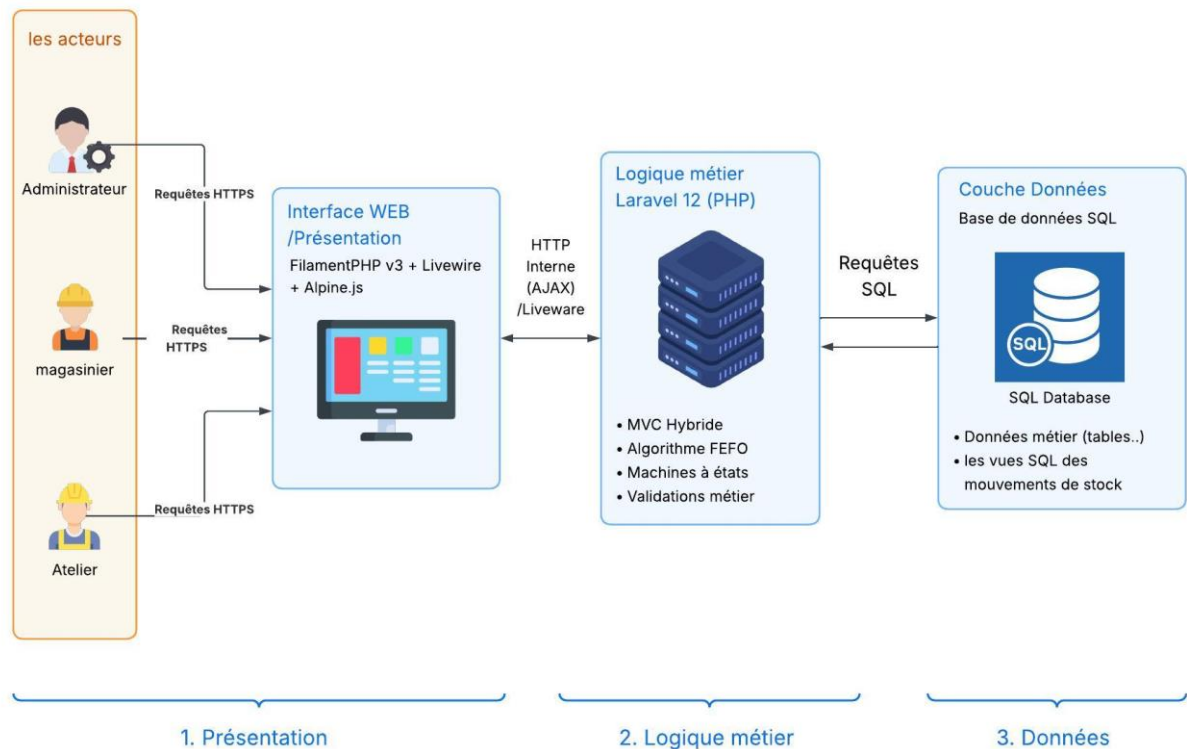


Figure 17 : Illustration de l'architecture de l'application « SweetStock »

Cette figure illustre l'architecture globale de notre application « SweetStock », nous avons conçu et déployé une architecture multi-support robuste et cohérente, accessible via une application mobile et une interface web. Cette architecture est basée sur plusieurs composants qui fonctionnent et assurent l'efficacité et la traçabilité de la gestion de production ainsi que la fluidité de l'expérience utilisateur. Cette architecture repose sur trois couches principales :

La première couche est la **présentation**, qui comprend une interface web et multi-support développée avec FilamentPHP v3 couplée à Livewire et Alpine.js, l'interface s'adapte à tous les types d'écrans. Elle assure une réactivité et une performance optimale grâce à des formulaires et des vues dynamiques et ergonomiques sans rechargement de page.

Le deuxième est la couche de **logique métier** basé sur Laravel 12. Elle est structurée selon le modèle MVC (Modèle-Vue-Contrôleur) Hybride ; combine le modèle MVC classique avec la réactivité en temps réel de ressources Filament et Livewire pour éviter de devoir créer des API séparées et complexes. Cette couche est responsable de toute la logique de l'application : elle gère le stock et les flux ses opérations de production via des transitions d'état qui assure un déroulement optimal et sécurisé. Elle applique l'algorithme FEFO pour l'allocation automatique des lots et utilise les coefficients de conditionnement, saisis par l'administrateur, pour calculer

automatiquement les conversions d'unités de mesure afin de garantir la traçabilité et d'éviter les erreurs lors de la manipulation de quantités exprimées dans des unités différentes.

La dernière couche, qui constitue la **base de données relationnelle**, organise l'ensemble des données : les référentiels, les stocks, les données de production, avec des vues SQL centralisées `mvt_stock_view`, qui recalcule les niveaux de stock à chaque mouvement, et `historique_mvt_stock_view`, qui présente l'historique de tous les mouvements de stock physiques de l'usine, toutes en temps réel.

III.3. Diagramme de classes

Le diagramme de classes est un type de diagramme structurel du langage UML. Il joue un rôle important dans le processus de modélisation des objets et représente la structure statique d'un système. Un seul diagramme de classes peut suffire à modéliser l'ensemble du système, ou plusieurs diagrammes distincts peuvent être utilisés pour modéliser ses différents composants.

Les diagrammes de classes servent de plans pour un système. Ils modélisent les entités (également les classes) du système, représentent leurs relations et décrivent leurs fonctions ainsi que les services fournis [\[33\]](#). Cette étape facilite la conception de la base de données et l'identification des composants du système, afin d'offrir une meilleure compréhension de l'aperçu général des schémas d'une application.

Le système repose sur deux fonctions principales de gestion : la gestion des stocks et la gestion de la production. Afin d'en faciliter la lecture, le diagramme de classes est divisé en deux sous-diagrammes.

III.3.1. Diagramme de classes du module de gestion des stocks

La figure 18, ci-dessous, illustre la première partie du diagramme de classes de gestion des stocks qui identifie les principales entités de la logistique ainsi que leurs relations.

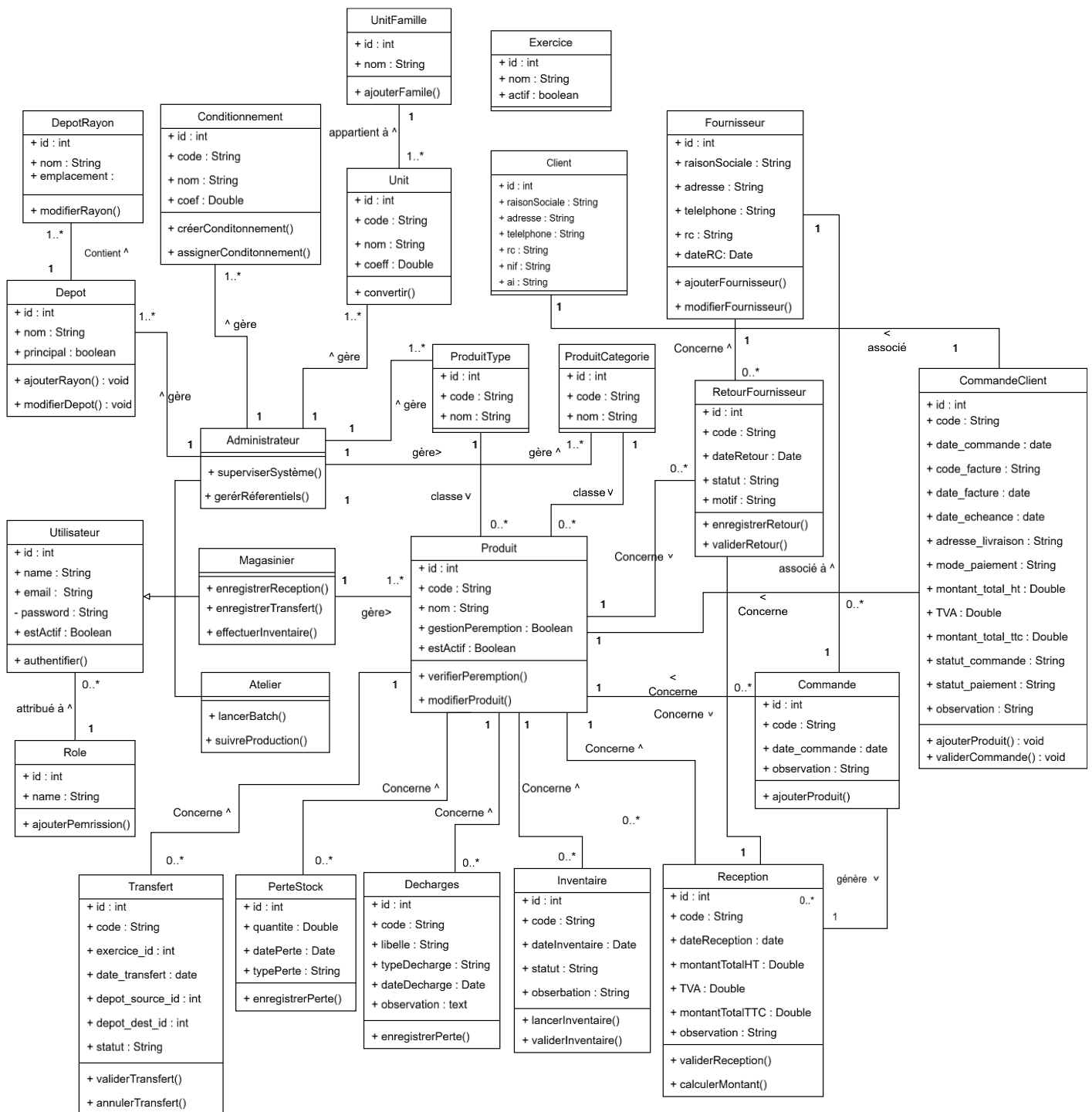


Figure 18 : Diagramme de classes du module de gestion des stocks. [Lien](#)

Cette partie du diagramme de classes est concentrée sur la gestion des stocks ainsi que sur les référentiels de base qui assurent la gestion efficace de la logistique. Voici une description des principales classes de cette partie :

a) Utilisateurs et rôles

La classe utilisateur est une classe parent qui contient les informations communes à tous les acteurs du système : nom, email, mot de passe, estActif (état du compte : active ou non) ainsi qu'une méthode authentifier() qui permet la connexion au système. Chaque utilisateur est lié à un rôle ; cette classe permet de définir les rôles attribués aux utilisateurs afin de gérer les droits d'accès. Les classes qui héritent cette classe utilisateur :

- **Administrateur** : supervise le système et gère les référentiels de base : unités de mesure, les familles des unités, conditionnements, exercices, les dépôts et les rayons ainsi que les types et les catégories de produit.
- **Magasinier** : gère les produits, les fournisseurs, les inventaires, les commandes, les réceptions, les transferts et les retour fournisseurs.
- **Atelier** : gère les batchs, les sessions de dragéification, les ordres de conditionnements (emballage).

b) Référentiels de base

Les référentiels de base présentent l'ensemble des données fondamentales utilisées par le système pour fonctionner correctement : les types de produits, les catégories de produits, les unités de mesure, les familles d'unités, les conditionnements, les dépôts et les rayons.

- **Classe « ProduitType »** : présente les types de produits avec un code unique et nom (Matière premiers, produit semi-fini, etc.). Chaque produit appartient à un seul type.
- **Classe « ProduitCategorie »** : présente les catégories de produits avec un code unique et nom (Matière batch, matière emballage, emballage primaire, etc.). Chaque produit appartient à une seule catégorie.
- **Classe « Unite »** : présente les unités de mesure utilisées dans le système avec un code et nom et coefficient ainsi que méthode convertir permet la conversion entre les unités (Kg, g, L, etc.). Chaque produit appartient à une seule catégorie.
- **Classe « UnitFamille »** : regroupe les unités appartenant à la même famille de mesure (famille masse, volume, etc.). Une famille peut contenir plusieurs unités.
- **Classe « Conditionnement »** : représente les différents conditionnements des produits (sachet, carton, sac, etc.). Chaque conditionnement possède code unique, nom et un coefficient utilisé pour les calculs de quantité.
- **Classe « Dépôt »** : représente le lieu de stockage de l'entreprise pour stocker, organiser des stocks, il possède nom et champ principale afin d'assurer gestion efficace des multi

dépôts. Un dépôt peut contenir plusieurs rayons.

- **Classe « DepotRayon »** : représente les rayon ou zones de stockage dans un dépôt : nom et emplacement pour décrire localisation de rayon.
- **Classe « Exercice »** : représente des années ou des périodes comptables utilisés afin d'organiser les opérations et d'archiver selon une période donnée. Il possède : id, nom et champ actif pour activer or désactiver.

c) Gestion des stocks

Le module de gestion des stocks constitue une position centrale de notre système. Il permet de suivre et gérer les mouvements des produits pour optimiser l'organisation et la gestion des produits afin de garantir une traçabilité des opérations sur le stock. Nous présenterons les classes du diagramme de classes qui font partie de ce module :

- **Classe « Produit »** : représente les produits gérés dans le système, les principaux attributs sont : code unique, nom, gestionPeremption indique si le produit possède une gestion de péremption ou non et estActif pour l'activation du produit. Cette classe constitue l'entité centrale du diagramme car elle intervient dans la plupart des opérations métier du système : les commandes, les réceptions, etc.
- **Classe « Fournisseur »** : représente les fournisseurs de l'entreprise auprès desquels l'entreprise commande ses produits : id, raison sociale du fournisseur, adresse, téléphone, rc (registre du commerce) et la date du registre du commerce.
- **Classe « Commande »** : représente les commandes passées auprès des fournisseurs, il possède : id, code unique, date de commande et observations s'il y a une remarque liée à la commande. Une commande peut concerner plusieurs produits et réceptions (partielles).
- **Classe « Réceptions »** : représente les réceptions des produits commandés auprès des fournisseurs, il possède : id, code unique, date de réception, montant total HT, TVA, montant total TTC et observation.
- **Classe « RetourFournisseur »** : gère les retours de produits aux fournisseurs en cas d'anomalies ou de défauts trouvés à la réception, il détient : id, code unique, date de retour, statut (état du retour : brouillon, validé, annulé) et un motif pour mentionner la raison du retour.
- **Classe « Client »** : représente les clients de l'entreprise soit une personne ou entreprise qui passe des commandes à l'entreprise : id, raison sociale du client, adresse, téléphone, rc (registre du commerce), adresse, téléphone, nif (Numéro d'Identification Fiscale) et ai (Article d'Imposition).

- **Classe « CommandeClient »** : présente une commande de client pour acheter un ou plusieurs produits. Il possède : code, date de commande, code de facture, date d'échéance, adresse de livraison, mode de paiement, montant total HT, montant total TTC, TVA, statut de commande, statut de paiement et observation.
- **Classe « Inventaire »** : représente les opérations d'inventaire pour vérifier les quantités réelles des produits stockés. Il possède : id, code unique, date d'inventaire, statut (brouillon, en cours, terminé, validé, annulé) ainsi qu'observation.
- **Classe « Transfert »** : représente les mouvements de produits entre les différents dépôts et rayons : id, code unique, date de transfert, et statut (brouillon, validé, annulé). Il est lié à un dépôt source, à un dépôt destination et à un exercice.
- **Classe « PerteStock »** : cette classe permet d'enregistrer les pertes de produits dues à la péremption, ou les déchets pendant la production ou à d'autres anomalies. Il détient : id, quantité, date de perte et le type de perte pour définir la nature ou la raison de la perte.
- **Classe « Décharges »** : cette classe présente une sortie de stock interne ou externe afin de tracer qui a sorti quoi, quand et la raison. Il possède : id, code, libelle, type de décharge, date de décharge et observation.

III.3.2. Diagramme de classes du module de gestion de production

La figure 19 ci-dessous illustre la deuxième partie du diagramme de classes de gestion de production qui identifie les principales entités de ce processus ainsi que leurs relations.

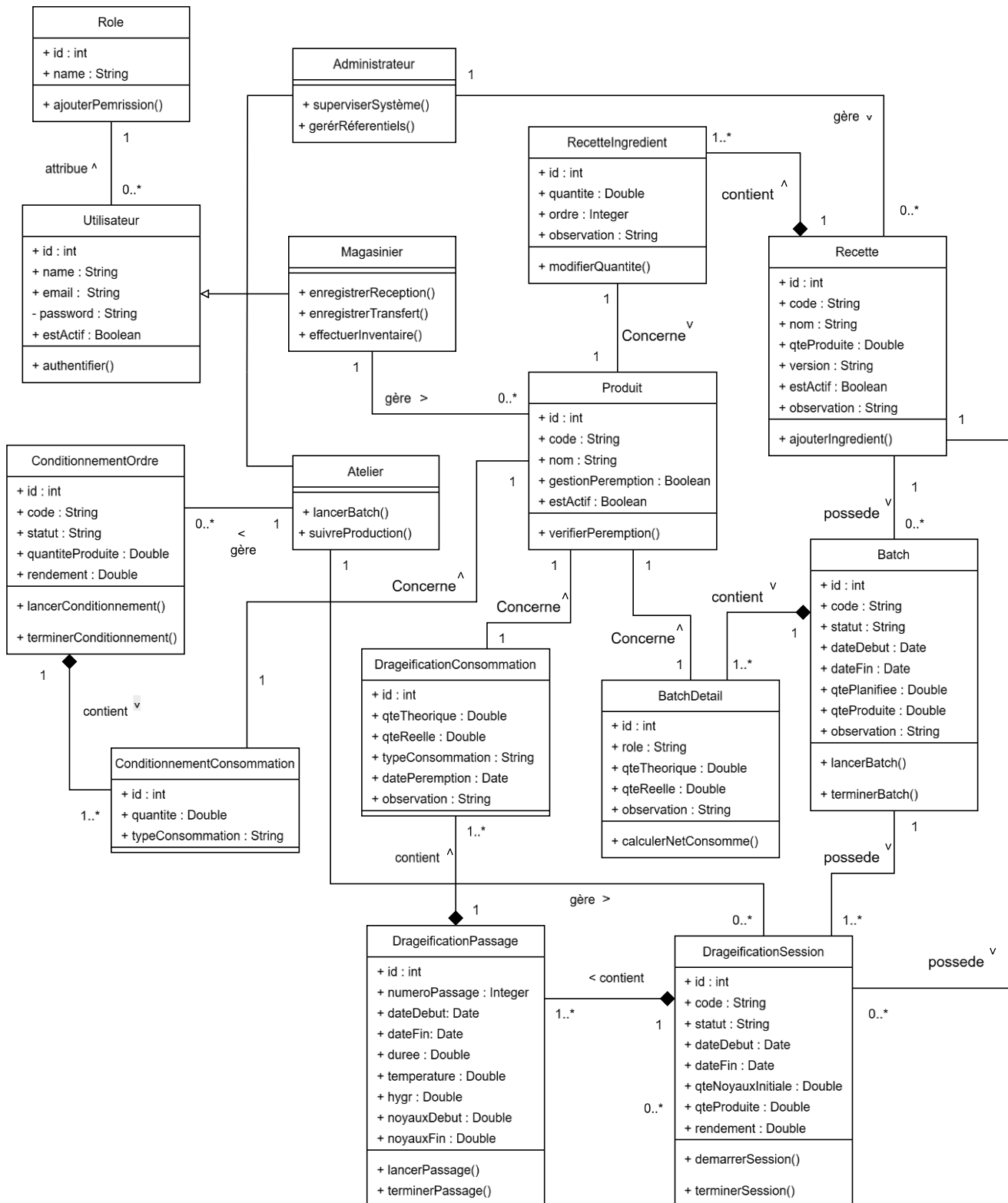


Figure 19 : Diagramme de classes du module de production. [Lien](#)

Cette partie de diagramme de classe met en évidence les principales entités du processus de production ainsi leurs relations. Voici une description détaillée de ces classes :

a) Gestion de production

Cette partie traite des entités qui constituent le processus de production depuis la planification des lots de production (batches) jusqu'à l'étape de conditionnement (emballage). Nous présenterons les classes du diagramme de classes qui font partie de ce module :

- **Classe « Recette »** : définit la formule de fabrication d'un produit ; elle contient des ingrédients via la classe « RecetteIngrédients ». Il possède un type : batch ou dragéification et peut être versionné. Ces attributs : id, code unique, nom, qteProduite, version, estActif pour activer ou désactiver ainsi qu'observation.
- **Classe « RecetteIngredient »** : cette classe représente un ingrédient au sein d'une recette, également une ligne dans une recette : id, quantité, ordre pour définir l'étape et l'observation. La méthode modifierQuantite () permet d'ajuster la quantité d'un ingrédient dans la recette. Chaque ligne est liée à un produit.
- **Classe « Batch »** : le batch représente un lot de fabrication. Il est défini par un id, un code unique, un statut (planifié, en cours, terminé, validé, annulé), des dates de début et de fin, une qtePlanifiee pour la quantité prévue, une qteProduite pour la quantité fabriquée ainsi qu'une observation. La méthode lancerBatch () permet de lancer le processus de production. Un batch est lié à une recette de type batch et il possède un ou plusieurs batch Details (batchDetails) ainsi que les sessions de dragéification (DrageificationSessions).
- **Classe « BatchDetail »** : cette classe représente les lignes ou les détails d'un batch ; chaque détail est lié à un produit afin de définir le produit consommé ou utilisé dans un batch de production. Il contient : un id, un rôle (ingrédient, produit sortant, retour ou déchet), une qteTheorique pour la quantité à consommer ou à utiliser pour ce produit ; il est calculé à partir l'ingrédient de la recette, une qteReelle également quantité prélève de stock et une observation. La méthode calculerNetConsomme () calcule la consommation nette entre la quantité théorique, réelle, de retour et de déchet.
- **Classe « DrageificationSession »** : représente la session de dragéification, qui regroupe des passages d'un cycle de dragéification. Elle contient : un id, un code unique, un statut (planifié, en cours, terminé, validé, annulé), des dates de début et fin, qteNoyauxInitiale, une qteProduite et un rendement qui mesure l'efficacité de la session. La méthode demarrerSession () permet de lancer le processus de dragéification. Une session est liée à une recette de type dragéification.

- **Classe « DrageificationPassage »** : représente une étape ou phase du processus de dragéification. Il décrit par un id, un numéro de Passage, des dates début et fin, durée, une température, un hyg pour l'hygrométrie, ainsi que noyauxDebut et noyauxFin mesurent la quantité de noyau au début et fin de passage. Avec des méthodes pour lancer et terminer un passage : lancerPassag () et terminerPassage (). Un passage possède un ou plusieurs consommations.
- **Classe « DrageificationConsommation »** : cette classe permet d'enregistrer la consommation de matières et produits pendant le processus de dragéification. Elle comprend : un id, type de consommation, qteTheorique et qteReelle, et une date de péremption pour assurer la traçabilité, ainsi qu'une observation.
- **Classe « ConditionnementOrdre »** : cette classe représente un ordre de conditionnement généré par l'atelier. Elle comprend un id, un code unique, un statut (planifie, en cours, terminé, validé, annulé), une quantité produite, et un rendement. Les méthodes lancerConditionnement () et terminerConditionnement () assurent la gestion du cycle de vie de l'ordre. Un ordre possède un plusieurs consommations.
- **Classe « ConditionnementConsommation »** : cette classe permet de registrer des consommations effectuées dans un ordre. Elle détient : id, quantité, type de consommation qui précise la ressource consommée (matière premier, noyau semi-fini).

Le diagramme de classe de « SweetStock » présente la structure statique du système ainsi que les relations entre les principales entités de l'application. Cette modélisation facilite la compréhension de la structure et de l'organisation du système et en simplifie la conception.

III.4. Modèle de données

La modélisation des données consiste à créer une représentation visuelle d'un système d'information, ou de certaines de ses parties, afin de clarifier les relations entre les points de données et les structures. Cette modélisation vise à illustrer les types de données utilisés et stockés dans le système, les relations entre ces types, les méthodes de regroupement et d'organisation, ainsi que les attributs associés. Les modèles de données sont des documents vivants, qui évoluent en fonction des besoins de l'entreprise. Ils jouent un rôle important dans l'optimisation des processus métier et la planification de l'architecture et de la stratégie informatiques. Les modèles de données peuvent être partagés avec les fournisseurs, les partenaires, etc. [\[34\]](#).

Pour notre système « SweetStock », en raison de la taille et de la complexité du modèle de données, nous avons décidé de le diviser en trois sections distinctes afin de faciliter sa compréhension et son analyse. Nous commençons par la première section, qui présente les référentiels de base regroupant l'ensemble des entités utilisées par le système pour fonctionner correctement. Puis, nous présentons la deuxième section qui couvre le module de gestion du stockage, et enfin, nous passons à la dernière section du module de production.

III.4.1. Modèle de données des référentiels de base

Cette section du modèle de données illustre les entités appartenant au référentiel de base qui détermine le fonctionnement du système. Ces entités ne sont pas liées à aucun module particulier. Elles sont utilisées et référencées par tous les modules du système (le stock et la production), tels que les types et les catégories de produits, les unités de mesure, les familles d'unités, les conditionnements, les dépôts et les rayons, ainsi que les exercices comptables, qui sont également représentés, car ils définissent le cadre temporel auquel toutes les opérations du système sont associées. Les entités utilisateurs, fournisseurs et clients ont été ajoutées à ce diagramme afin d'en améliorer la lisibilité. La figure 20 ci-dessous présente la section de modèle de données des référentiels de base.

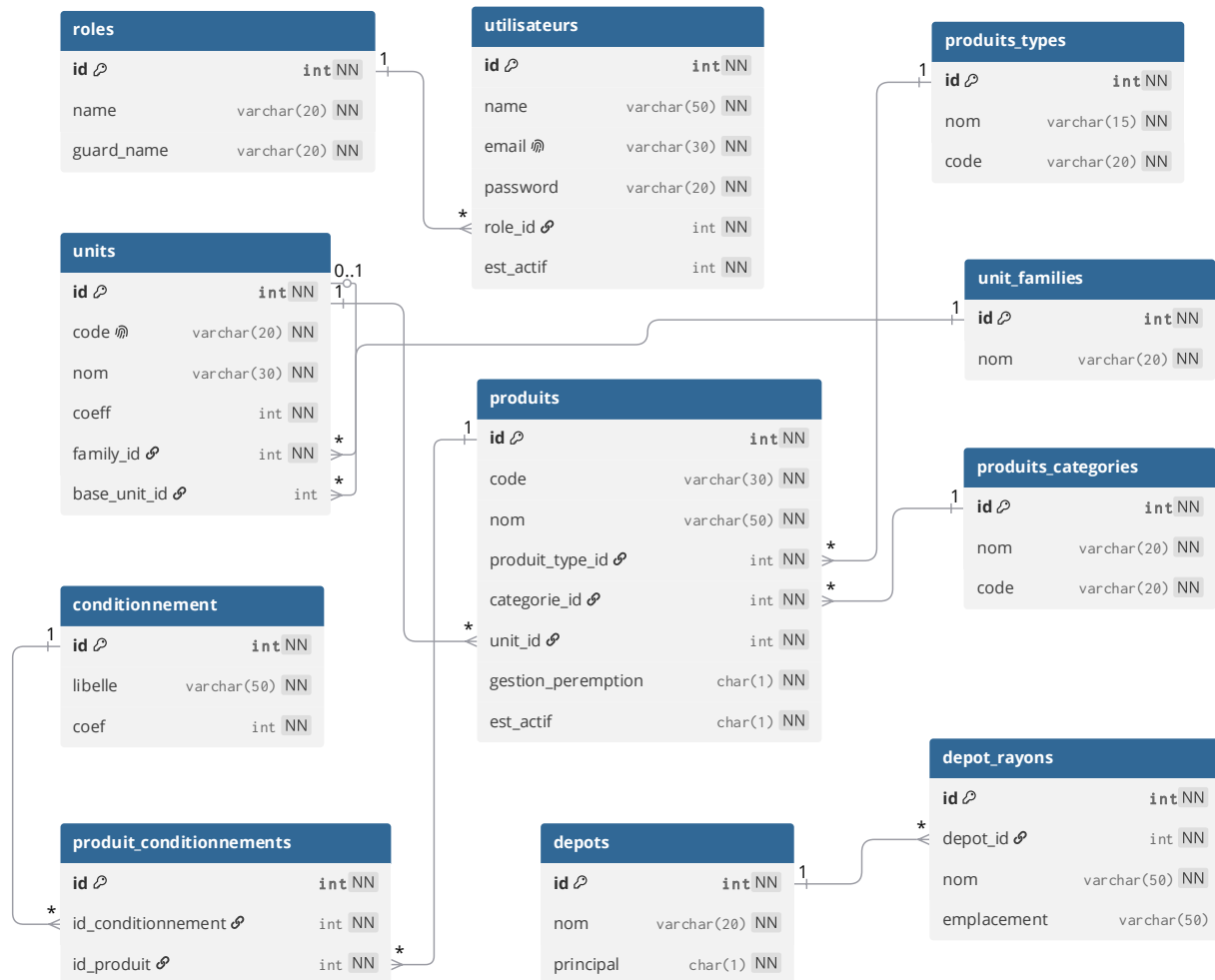


Figure 20 : Modèle de donnée des référentiels de base.

III.4.2. Modèle de données du module de gestion des stocks

Cette partie du modèle relationnel présente le module de gestion des stocks. Il couvre l'ensemble des entités qui impactent le stock de l'usine. Il inclut les flux d'approvisionnement tels que les commandes, les réceptions et les retours fournisseurs, ainsi que les différentes formes de sorties de stock : décharges, pertes, transferts entre dépôts et inventaires. Nous avons également intégré le module commercial à travers les entités commandes clients et leurs lignes de détail. Chaque opération est structurée en une entité d'en-tête et une ou plusieurs lignes de détail, selon le modèle Master-Détail dans lequel une entité principale (master) regroupe les informations générales d'une opération, tandis que ses entités filles (détails) en décrivent les lignes [35]. La figure 21 présente le modèle de données du module de gestion des stocks.

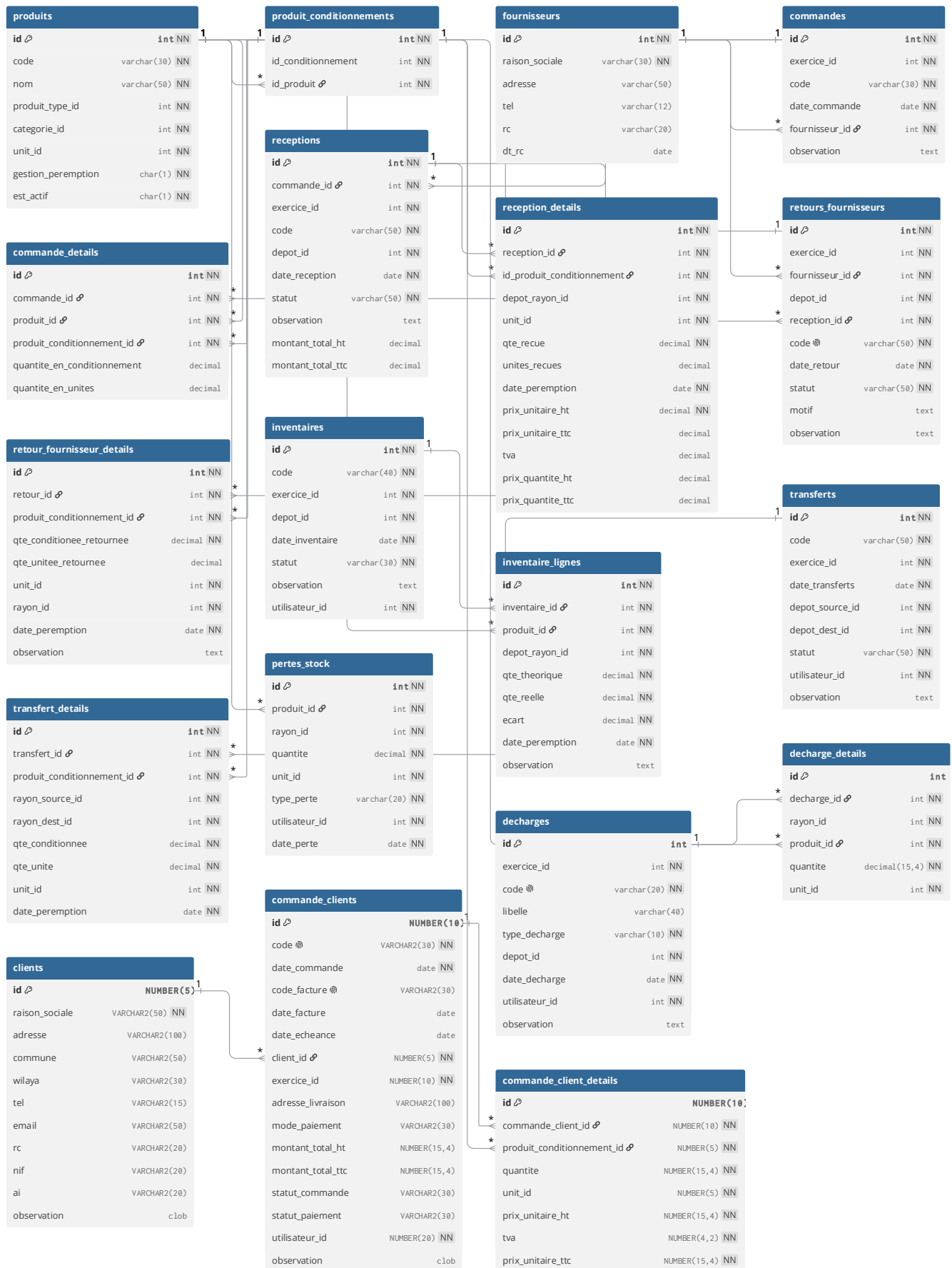


Figure 21: Modèle de données du module de gestion des stocks.

III.4.3. Modèle de données du module de gestion de production

Cette section présente le modèle de données du module de production. Il montre l'ensemble des entités qui affectent la cycle de fabrication tels que : des recettes et ingrédients, des batches, sessions de dragéification et ordres de conditionnement. Chaque étape est liée aux produits consommés et fabriqués afin de garantir une traçabilité complète sur l'ensemble de la chaîne de transformation. La figure 22 ci-dessous présente la partie de modèle de processus de production.

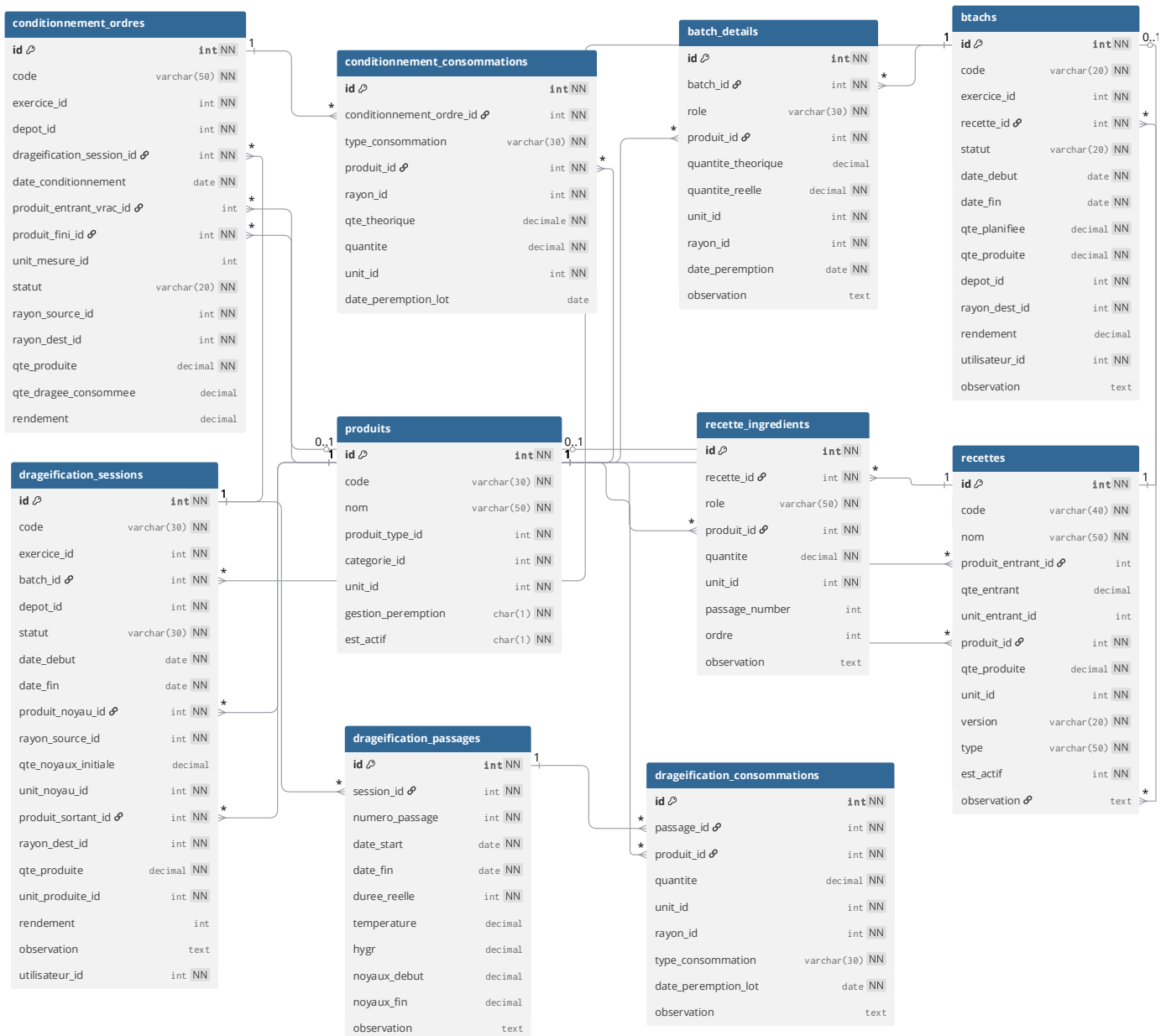


Figure 22 : Modèle de données du module de gestion de production.

III.5. Conclusion

Dans ce chapitre, nous avons examiné en détail la conception de notre application en analysant plusieurs aspects fondamentaux. Ces aspects incluent l'architecture globale, le diagramme de classes et les modèles de données. L'objectif principal de ce chapitre était de présenter une vue d'ensemble complète de la conception de notre application, en mettant en avant les aspects techniques et les choix effectués pour garantir une expérience utilisateur optimale.

L'architecture de l'application a été définie selon les exigences fonctionnelles et non fonctionnelles. Nous avons utilisé le diagramme de classes pour représenter les classes de notre application, ainsi que leurs attributs et leurs relations. Cette approche nous a permis d'obtenir une vision précise de la structure et des interactions entre les classes. Enfin, pour la modélisation des données, nous avons identifié les entités principales ainsi que leurs relations, ce qui a permis d'élaborer un schéma de base de données cohérent et efficace.

Dans le prochain chapitre, nous nous concentrerons sur la mise en œuvre de notre solution « SweetStock » à partir de la conception que nous avons élaborée au cours de ce chapitre.

Chapitre IV : Réalisation et implémentation de l'application « SweetStock »

IV.1. Introduction

La phase de réalisation d'un projet, également appelée l'implémentation de la conception de projet, est une étape essentielle qui vise à transformer les spécifications fonctionnelles en une solution opérationnelle. Cette phase se compose deux étapes principales : le lancement de la réalisation qui inclut le développement, les tests et la livraison du projet ou bien le déploiement.

Dans ce chapitre, nous présenterons les choix techniques utilisés dans ce projet, l'environnement de développement ainsi que la carte graphique de notre application « SweetStock ». Le but est de mettre en évidence tout le travail effectué, depuis le lancement de la réalisation jusqu'au déploiement de l'application.

IV.2. Processus du travail

Nous avons utilisé l'outil « Jira » pour optimiser la gestion de notre projet. Généralement cet outil est utilisé pour le travail d'équipe, mais cela a prouvé qu'il est également efficace pour le travail individuel, car il a aidé à clarifier les priorités des tâches et à maintenir un rythme de développement constant. « Jira » nous a aidés à gérer les différentes tâches et à suivre l'avancement. Cet outil nous a permis d'adopter une méthode de travail structurée, en planifiant les fonctionnalités à développer sous forme de tâches et en suivant leur progression. La figure 15 présente le tableau de bord Jira du projet « SweetStock ».

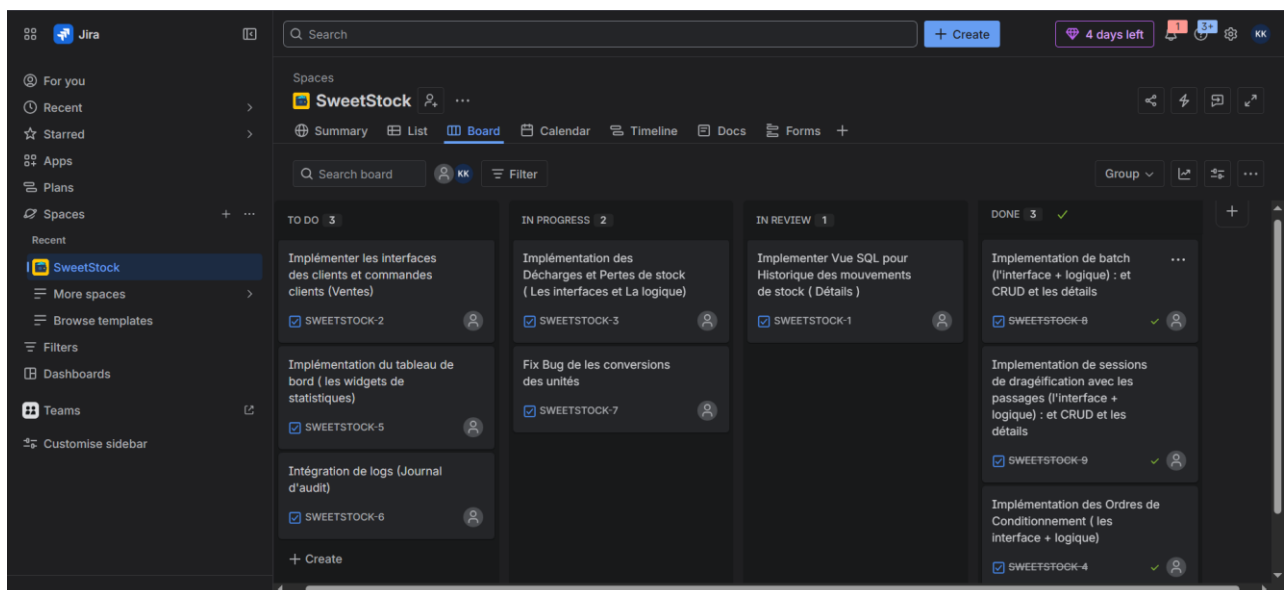


Figure 23 : Tableau de bord Jira du projet « SweetStock ».

IV.3. Technologies choisies

Le choix des technologies pour un projet informatique constitue une décision stratégique et une étape clé, impactant directement sur le développement, la maintenance du projet. Pour notre projet, nous avons choisi les technologies selon deux critères. L'entreprise a demandé d'utiliser Laravel et SQL pour la base de données, afin d'assurer la cohérence. De notre côté, nous avons choisi Filament pour créer l'interface d'administration. Les technologies choisies et utilisées dans notre projet sont les suivantes :

a) Laravel (Framework PHP)

Laravel est un framework PHP open source, reconnu pour son expressivité et son intuitivité, créé en 2011 par « Taylor Otwell ». Il est conçu pour faciliter le développement web. Son caractère intuitif permet aux développeurs d'anticiper son fonctionnement, ce qui facilite son apprentissage et son utilisation. Grâce à une syntaxe élégante et à des fondations robustes, Laravel favorise la création d'applications sécurisées et performantes. Ce framework suit le modèle architectural MVC (Modèle, Vue, Contrôleur), permettant une séparation nette entre la logique métier, la présentation et la gestion des interactions des utilisateurs. Il intègre également un ORM (Object-Relational Mapping), éloquent, qui permet de travailler avec la base de données de manière orientée objet. Le framework propose de nombreuses fonctionnalités intégrées, telles que le routage, l'authentification, et intègre plusieurs fonctionnalités de sécurité (protection CSRF (Cross-Site Request Forgery), prévention des attaques XSS (Cross-Site Scripting)), ainsi qu'un support natif pour la création d'API RESTful. Laravel dispose d'un écosystème riche ainsi que d'une expérience de développement optimisée pour les développeurs PHP avec des outils comme Artisan, une interface de ligne de commande (CLI), qui accélèrent et simplifient le développement et la maintenance du projet [\[36\]](#).

b) SQL (Structured Query Language)

Nous avons choisi SQL comme langage de gestion de base de données pour notre projet. SQL (Structured Query Language), également appelé langage de requête structuré, est un langage de programmation universel et spécialisé permettant d'interagir avec des systèmes de gestion de bases de données relationnelles (SGBDR) tels que MySQL. Ce langage a été créé au début des années 1970 par IBM pour simplifier l'accès aux systèmes de base de données relationnels et faciliter le traitement des informations [\[37\]](#). Dans le cadre de notre projet, SQL nous a permis de gérer les relations complexes entre les tables et d'effectuer des opérations précises sur nos données, telles que l'interrogation, l'insertion, la mise à jour ou la suppression d'enregistrements, afin d'assurer la cohérence et la qualité des données stockées. Cela en faisait la meilleure solution pour la gestion

de la base de données de « SweetStock ».

c) Filament (Interface d'administration)

Pour notre projet, nous avons utilisé Filament pour développer l'interface de gestion de l'application. Filament est une bibliothèque open source populaire de panneaux d'administration et d'interfaces utilisateur (back-office) reconçue pour Laravel. Elle offre un ensemble de fonctionnalités et de composants prêts à l'emploi pour simplifier le développement des panneaux d'administration. Elle est construite sur des technologies modernes : la TALL stack pour Tailwind, Alpine, Laravel et Livewire. Filament combine des technologies frontend modernes à la puissance de Laravel [38]. Il propose une expérience de développement efficace, flexible et agréable, que vous créiez des applications web complexes axées sur les données ou de simples interfaces d'administration.

IV.4. Outils de développement

Pour ce projet, nous avons choisi les outils de développement pour l'édition de code, la gestion de la base de données, la conception et la modélisation ainsi que le design des interfaces. Nous listons ces outils en détail.

a) Visual Studio Code

Visual Studio Code (VS Code) est un éditeur de code source gratuit, léger et multiplateforme, développé par « Microsoft ». Il offre un environnement de développement moderne et personnalisable, compatible avec de nombreux langages de programmation. Son objectif est de fournir aux utilisateurs un espace de travail ergonomique, puissant et extensible, adapté à divers types de projets, notamment le développement web, logiciel, ainsi que la gestion de scripts. Il se distingue par une architecture modulaire enrichie par des extensions, ce qui permet d'augmenter les fonctionnalités selon des besoins spécifiques de chaque utilisateur [39].

b) MySQL Workbench

MySQL Workbench est un outil graphique officiel et gratuit fourni par Oracle pour la gestion des bases de données MySQL. Il permet de concevoir, modéliser, générer et gérer visuellement des bases de données. Parmi ses fonctionnalités : la modélisation entité relation (ER), la rédaction et l'exécution de requêtes SQL, ainsi que la gestion des connexions et des utilisateurs. Enfin, MySQL Workbench propose un environnement intégré pour le développement et la maintenance des bases de données relationnelles [40].

c) Github

GitHub est une plateforme de développement collaboratif basée sur le système de contrôle de version Git. Elle facilite la collaboration entre les développeurs et la gestion des projets logiciels. GitHub offre un hébergement de code source et la gestion des versions de projet [41]. GitHub présente un outil essentiel pour assurer la traçabilité du développement logiciel.

d) Draw.io

Draw.io est une solution accessible en ligne et gratuite qui permet de créer des diagrammes et des schémas et bien plus encore. Il permet de concevoir divers types de diagrammes, notamment des diagrammes de flux UML. Avec une interface simple et facile à utiliser [42].

e) Dbdiagram.io

Dbdiagram.io est un outil simple et efficace conçu pour créer et gérer les diagrammes de base de données. Il permet de dessiner des diagrammes de relation d'entité (ERD) à l'aide d'un langage simple spécifique au domaine (DSL), ce qui facilite la création et la visualisation des structures de base de données et il est bien adapté à la phase de modélisation des données [43].

f) Lucidchart

Lucidchart est un outil de modélisation de diagrammes en ligne qui permet de créer des schémas UML, des organigrammes, des diagrammes de flux et des modèles de bases de données de manière collaborative et intuitive, directement depuis un navigateur web, sans nécessiter d'installation locale [44].

g) Jira

Jira est un outil de gestion de projet et de suivi des tâches développé par l'éditeur « Atlassian ». Il prend en charge toute méthodologie agile et permet de planifier, suivre et organiser le travail. Cette plateforme offre la possibilité de créer des tâches, de les attribuer aux membres de l'équipe, de suivre leur avancement [45].

IV.5. Présentation des interfaces de l'application

Dans cette section, nous présentons les interfaces de l'application à l'aide de captures d'écran. Chaque fonctionnalité implémentée est illustrée par une image avec une description détaillée. Cette présentation offre une vue claire et exhaustive de l'application développée.

IV.5.1. Interface de connexion

La page de connexion est le point d'entrée principal de l'application. Elle permet aux utilisateurs

de saisir leur identifiant de connexion (l'adresse e-mail et mot de passe) qui sont des champs obligatoires. Un lien de « Mot de passe oublié ? » permet la réinitialisation du mot de passe. Il y a une case à cocher « Se souvenir de moi ? » pour garder la session active. Enfin, le bouton « Connexion » valide l'authentification et dirige vers l'interface utilisateur. La figure 24 représente l'interface de connexion.

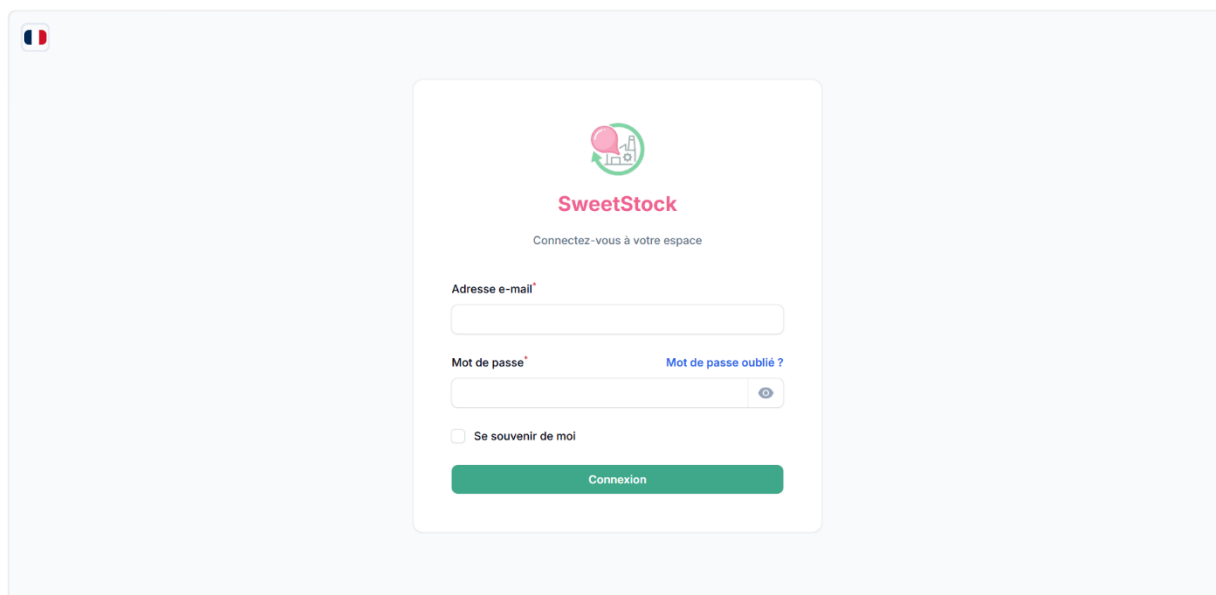


Figure 24 : Interface de connexion

IV.5.2. Tableau de bord administrateur

La figure 25 présente la page de tableau de bord administratif qui est également la page d'accueil principale après authentification. Elle centralise l'ensemble des modules de l'application offrant une vue globale des opérations du système.

Le tableau de bord administratif est conçu pour offrir une vue exhaustive des opérations du système. Tout d'abord, il y a un message de bienvenue précisant le rôle de l'utilisateur connecté. Il présente des indicateurs clés de performance comme les valeurs des achats et des ventes (HT) du mois en cours. Il présente aussi un flux qui permet le suivi et l'avancement des quatre phases de la chaîne de production. Une section illustre un graphique en barres qui retrace l'évolution des achats et des ventes HT sur les trois derniers mois, permettant à l'administrateur de comparer les tendances financières d'un mois à l'autre et de voir la différence entre les dépenses d'approvisionnement et les recettes commerciales de l'usine. Par ailleurs, Un élément de suivi des expirations liste les produits dont la date de péremption approche dans les 90 prochains jours. Enfin, un panneau d'alertes opérationnelles signale en temps réel les anomalies détectées dans le système, un panneau de batchs du mois présente la répartition des batchs du mois selon leur statut,

tandis qu'un journal d'activité récente retrace les dernières opérations validées à travers les différents modules. Tout cela afin de soutenir la prise de décision.

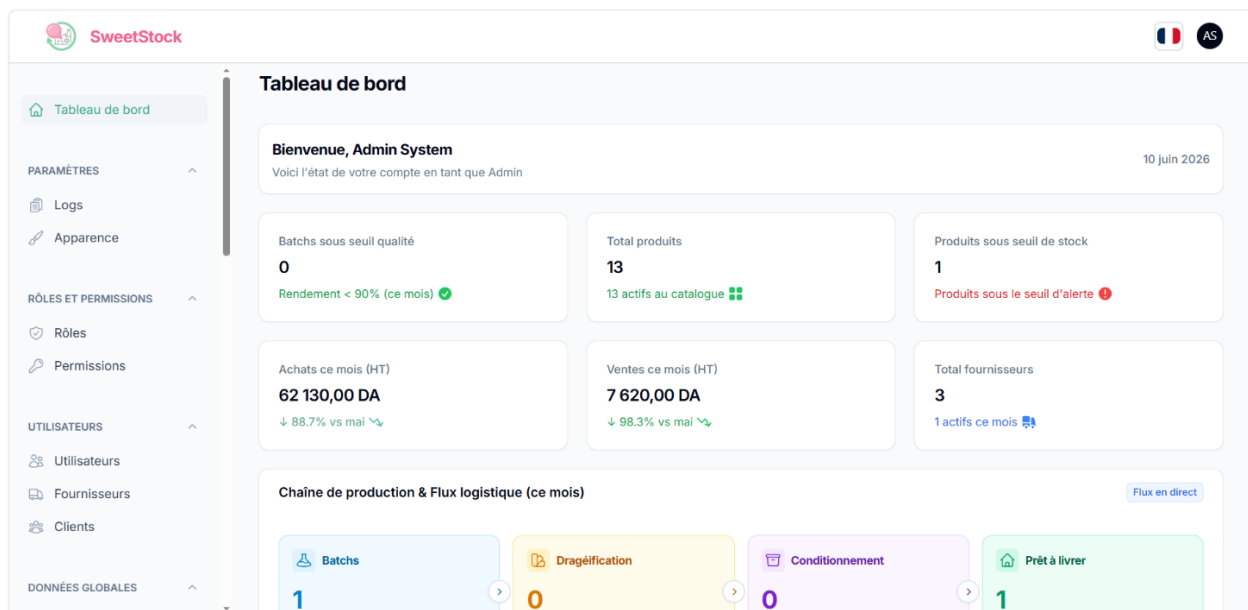


Figure 25 : Tableau de bord administrateur

IV.5.3. Logs du système

Le log, ou journal d'activité, présente l'historique complet des actions effectuées dans l'application. Il comprend cinq colonnes : Type qui indique le module concerné, événement (action), le sujet qui identifie l'enregistrement par un code avec le module, l'utilisateur, les propriétés et la date d'enregistrement. La figure 26 présente l'interface du journal d'activité.

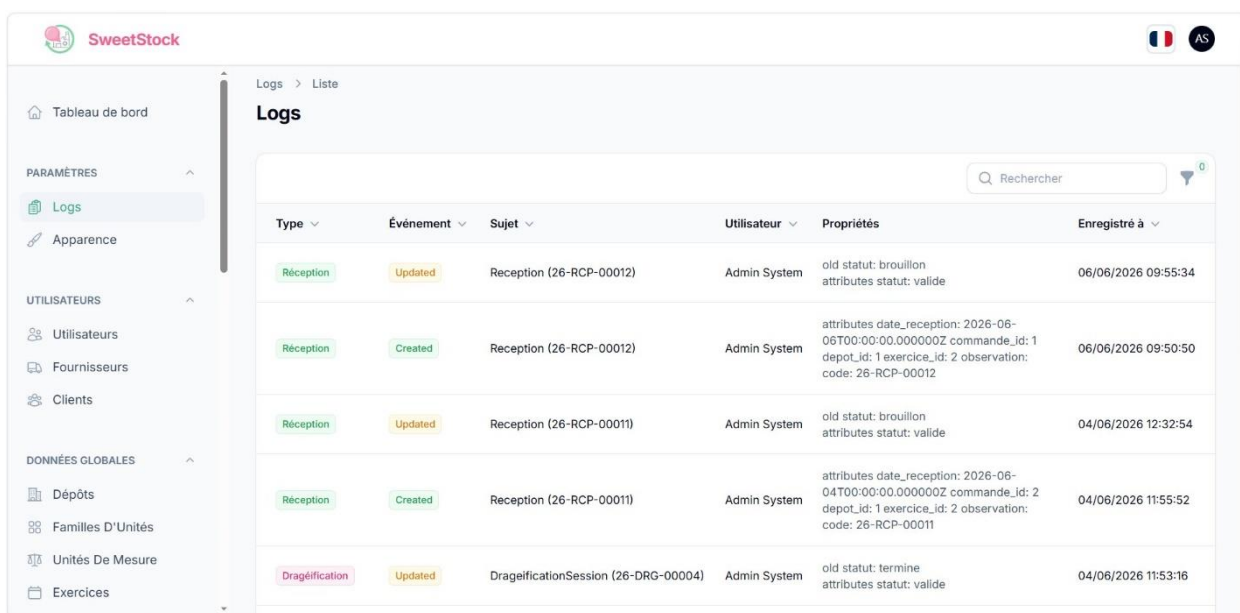


Figure 26 : Logs du système

IV.5.4. Rôles du système

La figure 27 présente l'interface de la gestion des rôles qui définit les différents niveaux d'accès dans le système, structurée de façon claire et efficiente. Pour chaque rôle, on attribue les permissions associées via l'action de modifier.

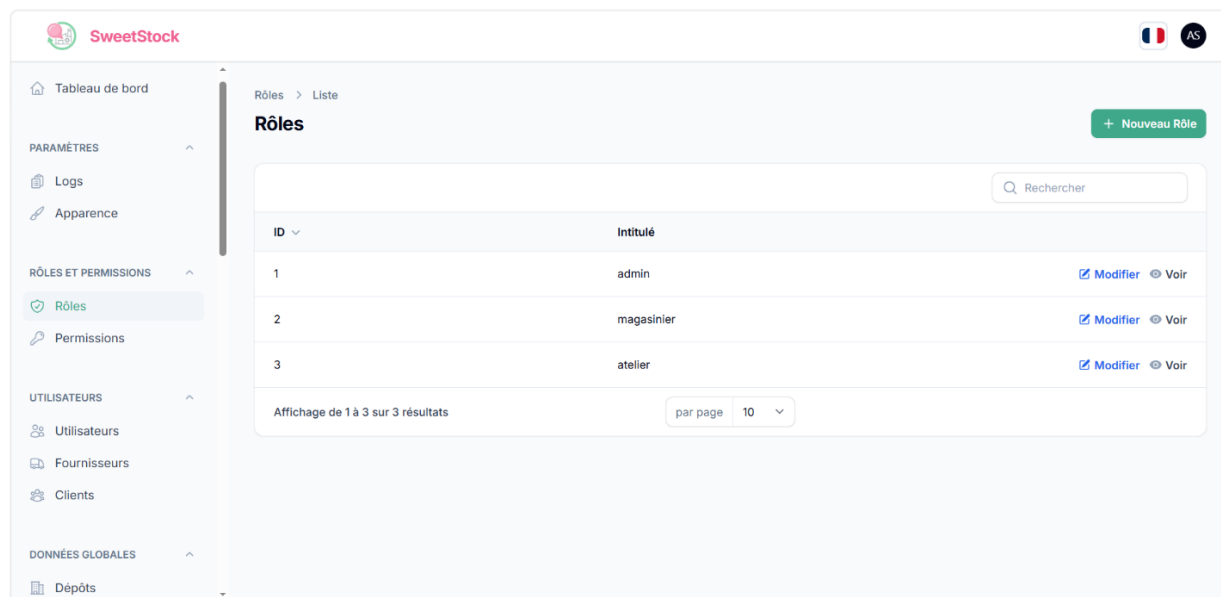


Figure 27 : Rôles du système

IV.5.5. Utilisateurs

La figure 28 présente l'interface de gestion des utilisateurs au sein du système. La page affiche la liste des utilisateurs. Chaque ligne permet de modifier rapidement les informations de l'utilisateur via l'action « Modifier ». Un champ de recherche facilite la localisation d'un utilisateur, avec la possibilité d'ajouter de nouveaux utilisateurs grâce à l'action « Nouvel utilisateur ». Cette structure avec une répartition claire des rôles, assure un contrôle précis des accès et simplifie l'administration du système.

Utilisateurs > Liste

Utilisateurs

+ Nouvel utilisateur

Rechercher

☐	Nom complet	Adresse email	Rôles	Actif	Date création	
☐	Admin System	admin@sweetstock.com	admin	✓	07/05/2026	Modifier
☐	test Magasinier	magasinier@sweetstock.com	magasinier	✓	07/05/2026	Modifier
☐	test Atelier	atelier@sweetstock.com	atelier	✓	07/05/2026	Modifier

Affichage de 1 à 3 sur 3 résultats

par page 10

Figure 28 : Liste des utilisateurs du système

IV.5.6. Fournisseurs

La figure 29 présente l'interface de gestion des fournisseurs. Elle permet de centraliser et de gérer l'ensemble des fournisseurs. Il offre la possibilité d'enregistrer un nouveau fournisseur en renseignant ses informations : raison sociale, adresse, numéro de téléphone et numéro du registre du commerce, ainsi que de consulter ou de modifier les données d'un fournisseur existant. Un indicateur d'activation permet de distinguer les fournisseurs actifs de ceux qui ne sont plus en activité. Le module dispose également d'une fonction de recherche facilitant la localisation rapide d'un fournisseur.

Fournisseurs > Liste

Fournisseurs

+ Nouveau fournisseur

Rechercher

	Raison Sociale	Adresse	Téléphone	N° RC	Date RC	Actif	
	SUCRERIES DU NORD	12 Rue Industrielle, Alger	0541122041	RC-ALG-001	02/12/2014	✓	Voir Modifier
	EMBALLAGES MODERNES	Rue Commerce, Constantine	0741152078	RC-CST-012	02/01/2012	✓	Voir Modifier
	Test	TEST	0541122041	RC-TST-012	02/02/2021	✓	Voir Modifier

Affichage de 1 à 3 sur 3 résultats

par page 10

Figure 29 : Liste des fournisseurs

IV.5.7. Clients

La figure 30 présente la page de gestion des clients. Elle permet de gérer les clients de l'usine ainsi que d'enregistrer un nouveau client en saisissant ses coordonnées : raison sociale, adresse, commune, wilaya, numéro de téléphone, adresse email, numéro de registre commerce (RC), le Numéro d'Identification Fiscale (NIF) et Article d'Imposition (AI). Elle permet d'ajouter, de consulter ou de modifier les informations d'un client, ainsi que de gérer son statut d'activation et de le localiser rapidement grâce à la fonction de recherche.

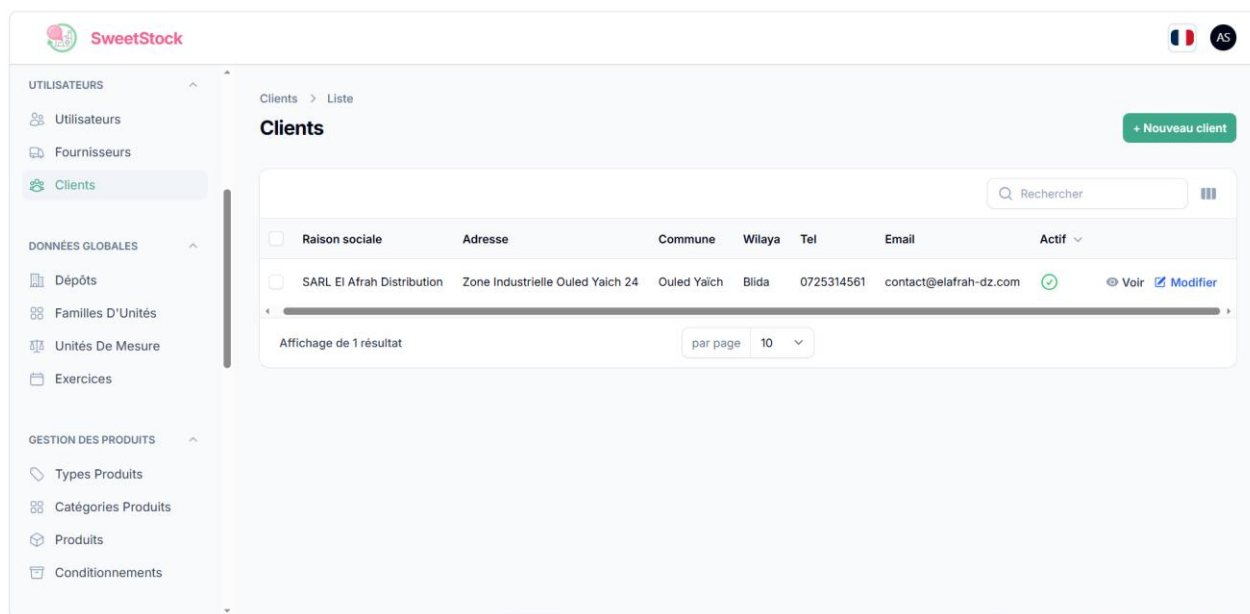


Figure 30 : Liste des clients

IV.5.8. Dépôts et rayons

La figure 31 présente l'interface de gestion des dépôts de l'entreprise, elle optimise la gestion des entrepôts de l'usine en enregistrant les informations essentielles de chaque emplacement et en identifiant le dépôt principal. La fonctionnalité « Gérer rayons » permet de définir les emplacement internes du dépôt afin d'affiner la localisation des produits lors des opérations de stock. Cette organisation assure une traçabilité des mouvements de stock.

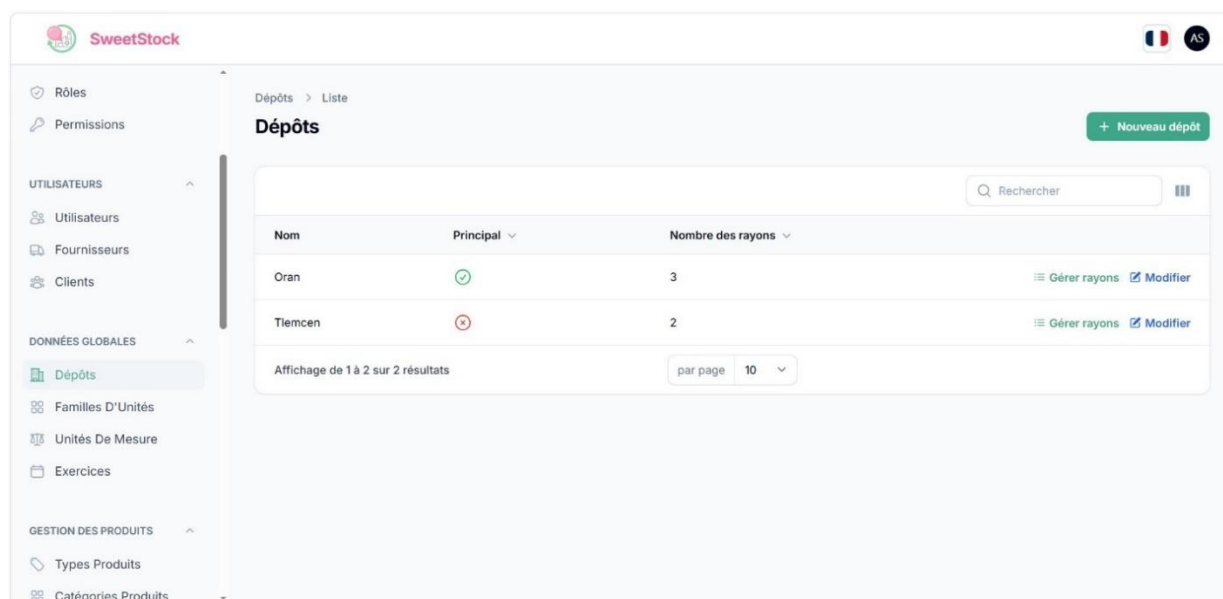


Figure 31 : Liste des dépôts de l'entreprise

IV.5.9. Familles d'unités et les unités de mesure

Les familles d'unités et unités de mesure forment le référentiel de mesure du système. Une famille d'unités regroupe les unités par catégorie, telles que la masse, le volume, etc. Chaque unité de mesure est définie par son code, son nom et sa famille d'appartenance. Une unité peut être définie comme unité de base. Lorsqu'une unité est dérivée d'une unité de base, un coefficient de conversion lui est associé, ce qui permet au système d'effectuer automatiquement les conversions nécessaires lors des opérations de stock et de production. La figure 32 présente la page de gestion des familles d'unités ainsi que la figure 33 montre la page des unités de mesure.

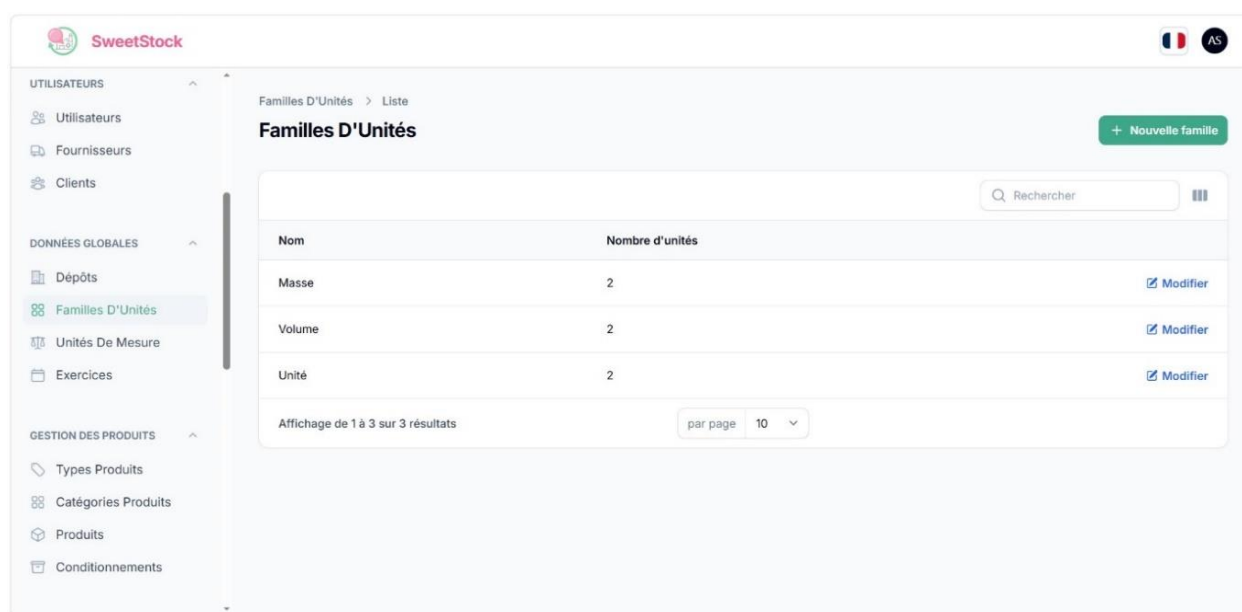


Figure 32 : Liste des familles d'unités

Code	Nom	Famille	Unité de base	Coeff	
Kg	Kilogramme	Masse	N/A	1	Modifier Supprimer
g	gramme	Masse	Kilogramme	0,001	Modifier Supprimer
L	Litre	Volume	N/A	1	Modifier Supprimer
ML	Millilitre	Volume	Litre	0,001	Modifier Supprimer
Psc	Pièce	Unité	N/A	1	Modifier Supprimer
Rl	Rouleau	Unité	N/A	1	Modifier Supprimer

Figure 33 : Liste des unités de mesure

IV.5.10. Exercices

La figure 34 présente la page des exercices qui permet de gérer les périodes comptables annuelles du système. Chaque exercice correspond à une année et un seul exercice peut être actif à la fois, ce qui définit la période courante utilisée pour les opérations du système. La clôture d'un exercice se traduit par la désactivation de son statut, ce qui archive ses données, qui restent stockées en base, mais ne s'affichent plus dans les interfaces opérationnelles. Un nouvel exercice peut ensuite être créé pour commencer un nouveau cycle, tout en conservant la possibilité de consulter les données historiques grâce au filtrage par exercice.

Nom	Actif	
2026		Modifier
2025		Modifier

Figure 34 : Liste des exercices

IV.5.11. Gestion des produits

Cette partie décrit les interfaces de la gestion des produits. Elle expose l'organisation des produits qui composent des types de produits, des catégories et des fiches produits détaillées.

IV.5.11.1. Types produits

Les types de produits définissent les grandes familles de classification. Chaque type a un code unique et nom. Cette classification détermine le comportement du produit dans les différents modules du système, notamment en production et en stock. La figure 35 présente la page de gestion des types produits.

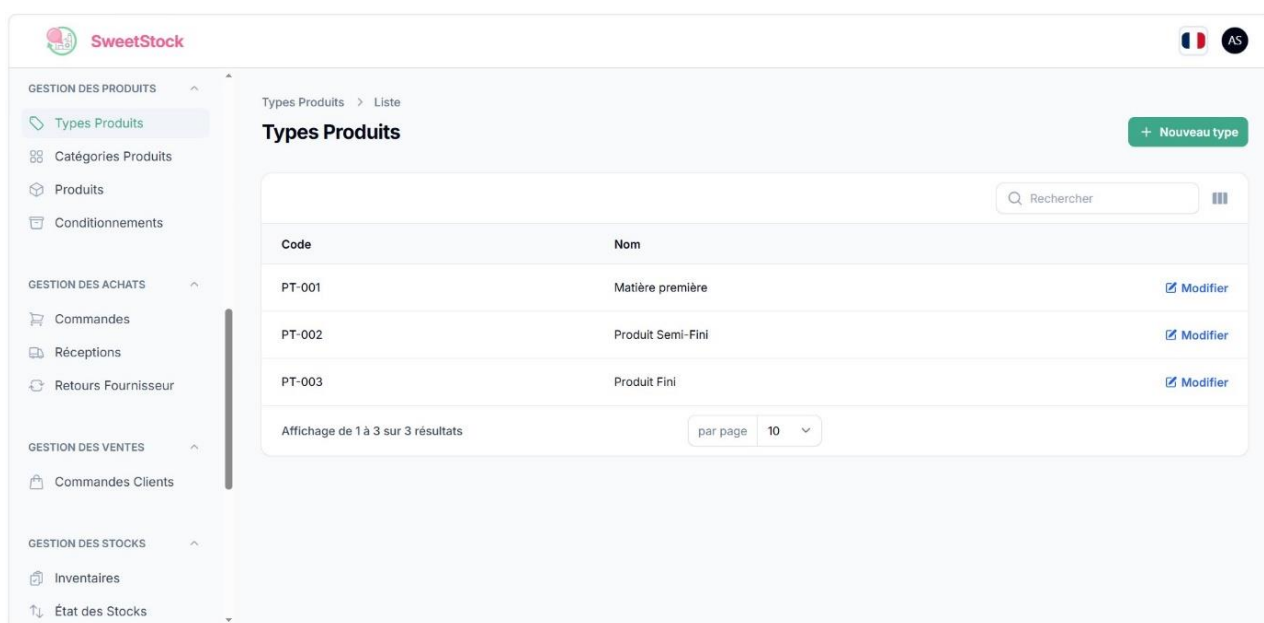
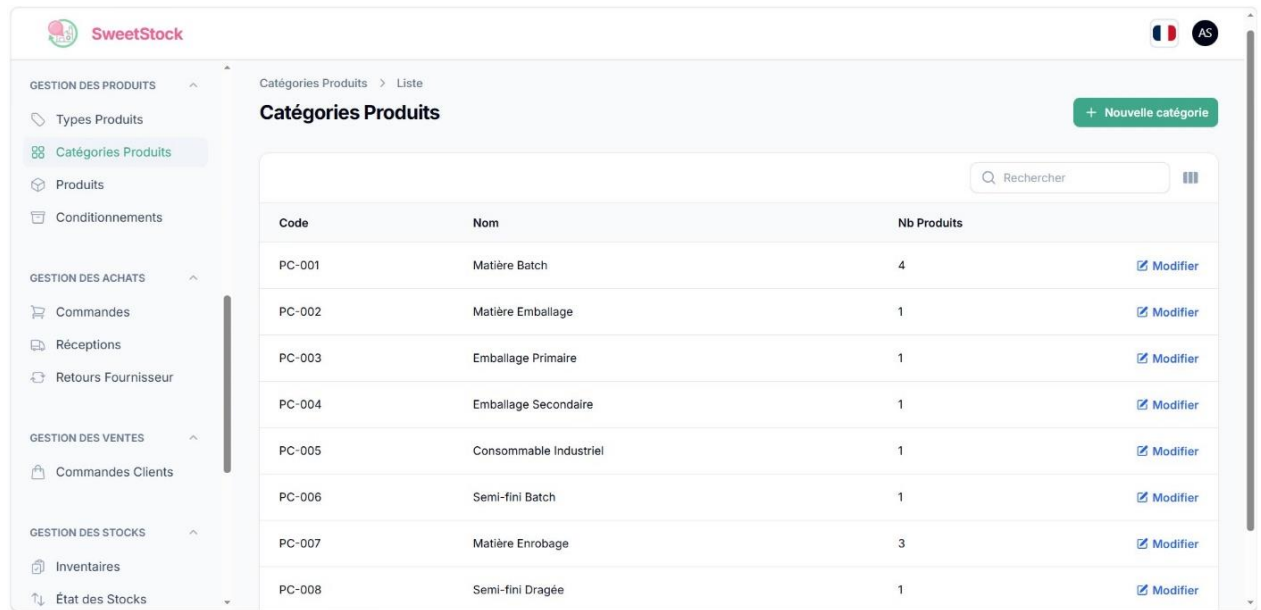


Figure 35 : Liste des types produits

IV.5.11.2. Catégories de produits

Les produits sont classés en catégories, telles que les matières batch, les emballages primaires et secondaires. Chaque catégorie est identifiée par un code unique et un nom, et indique le nombre de produits qui lui sont rattachés. Cette organisation facilite la recherche, le filtrage et la gestion efficace des produits. La figure 36 présente la page de gestion des catégories de produits.

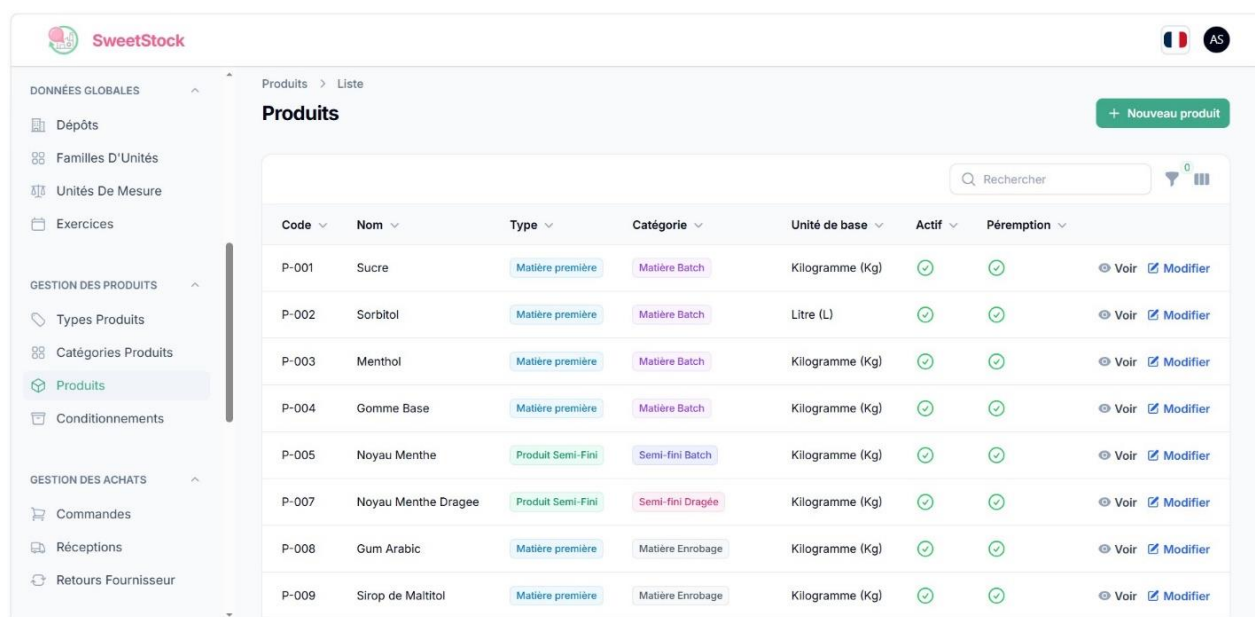


Code	Nom	Nb Produits	
PC-001	Matière Batch	4	Modifier
PC-002	Matière Emballage	1	Modifier
PC-003	Emballage Primaire	1	Modifier
PC-004	Emballage Secondaire	1	Modifier
PC-005	Consommable Industriel	1	Modifier
PC-006	Semi-fini Batch	1	Modifier
PC-007	Matière Enrobage	3	Modifier
PC-008	Semi-fini Dragée	1	Modifier

Figure 36 : Liste des catégories de produits

IV.5.11.3. Produits

Les produits servent de catalogue central du système. Il permet d'enregistrer et de gérer tous les produits de la fabrique, en leur attribuant un code unique, un type, une catégorie, une unité de base et un statut d'activation. La colonne péremption précise si le suivi de la date d'expiration est activé pour chaque produit, ce qui détermine l'utilisation de l'algorithme FEFO lors des mouvements de stock. Chaque produit peut être consulté ou modifié. Le module propose également des filtres par type et par catégorie pour faciliter la navigation dans le catalogue. La figure 37 présente la page de gestion des produits.



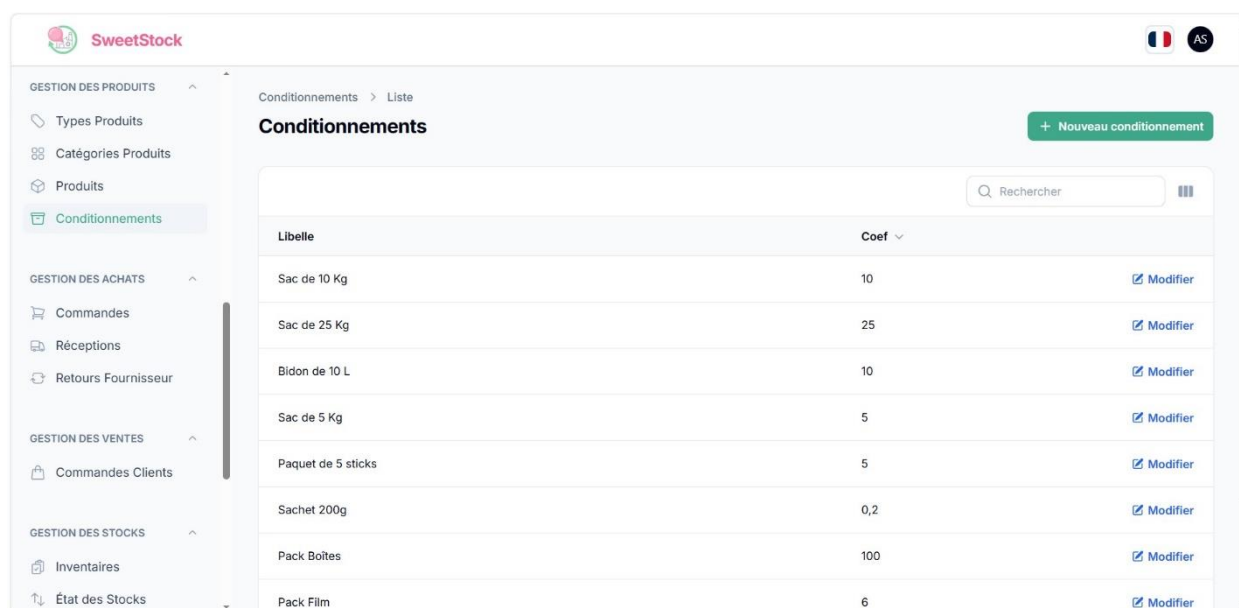
Produits

Code	Nom	Type	Catégorie	Unité de base	Actif	Péréemption	
P-001	Sucre	Matière première	Matière Batch	Kilogramme (Kg)	✓	✓	Voir Modifier
P-002	Sorbitol	Matière première	Matière Batch	Litre (L)	✓	✓	Voir Modifier
P-003	Menthol	Matière première	Matière Batch	Kilogramme (Kg)	✓	✓	Voir Modifier
P-004	Gomme Base	Matière première	Matière Batch	Kilogramme (Kg)	✓	✓	Voir Modifier
P-005	Noyau Menthe	Produit Semi-Fini	Semi-fini Batch	Kilogramme (Kg)	✓	✓	Voir Modifier
P-007	Noyau Menthe Dragee	Produit Semi-Fini	Semi-fini Dragée	Kilogramme (Kg)	✓	✓	Voir Modifier
P-008	Gum Arabic	Matière première	Matière Enrobage	Kilogramme (Kg)	✓	✓	Voir Modifier
P-009	Sirop de Maltitol	Matière première	Matière Enrobage	Kilogramme (Kg)	✓	✓	Voir Modifier

Figure 37 : Liste des produits

IV.5.11.4. Conditionnements

Les conditionnements permettent de spécifier les formats d'emballage utilisés par l'usine pour conditionner ses produits. Chaque format est défini par un libellé et un coefficient indiquant la quantité contenue dans ce conditionnement. Par exemple, un sac de 25 kg a un coefficient de 25. Ce coefficient est utilisé par le système pour convertir les unités de conditionnement en unités de base lors des opérations de réception, de production et de vente. La figure 38 présente la page de gestion des conditionnements.



Conditionnements

Libelle	Coef	
Sac de 10 Kg	10	Modifier
Sac de 25 Kg	25	Modifier
Bidon de 10 L	10	Modifier
Sac de 5 Kg	5	Modifier
Paquet de 5 sticks	5	Modifier
Sachet 200g	0,2	Modifier
Pack Boîtes	100	Modifier
Pack Film	6	Modifier

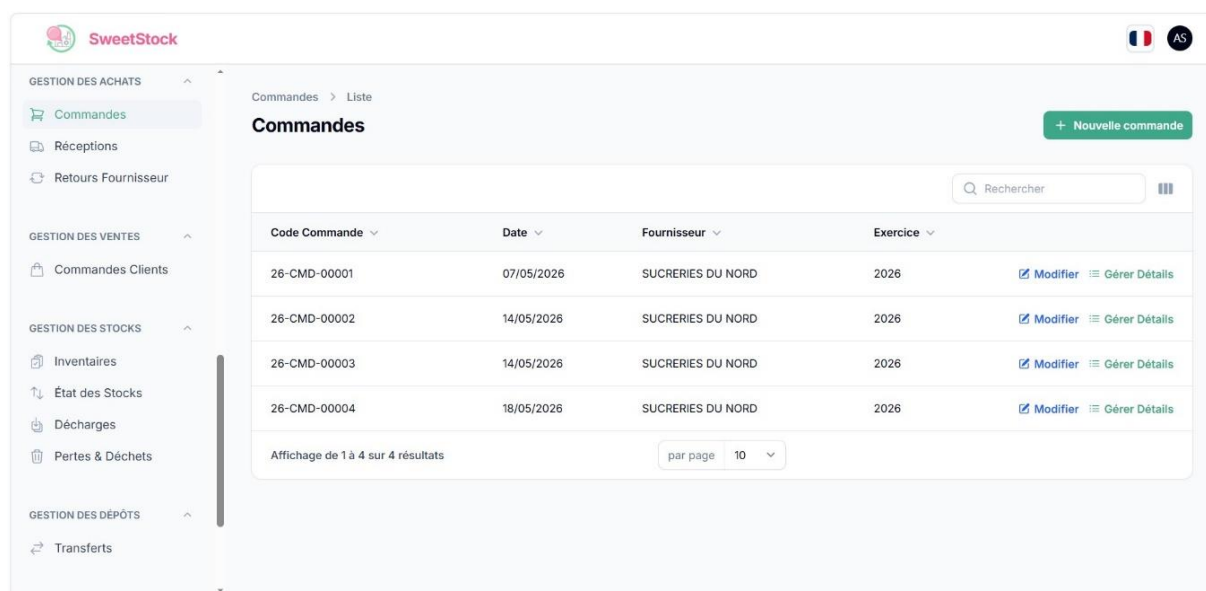
Figure 38 : Liste des conditionnements

IV.5.12. Gestion des achats

Cette section présente les interfaces de gestion des achats qui regroupe et décrit le processus d'approvisionnement : Commandes fournisseurs, Réceptions et les retours fournisseurs.

IV.5.12.1. Commandes fournisseurs

Les commandes permettent de gérer des bons de commande envoyés aux fournisseurs. Chaque commande est identifiée par un code généré automatiquement et associée à un fournisseur ainsi qu'à un exercice. La fonctionnalité « Gérer Détails » permet d'accéder aux lignes de commande afin de renseigner les produits commandés et leurs quantités. La commande constitue la base du processus de réception, assurant la traçabilité entre la commande adressée au fournisseur et les marchandises effectivement reçues en stock. La figure 39 présente l'interface de gestion des commandes fournisseurs.

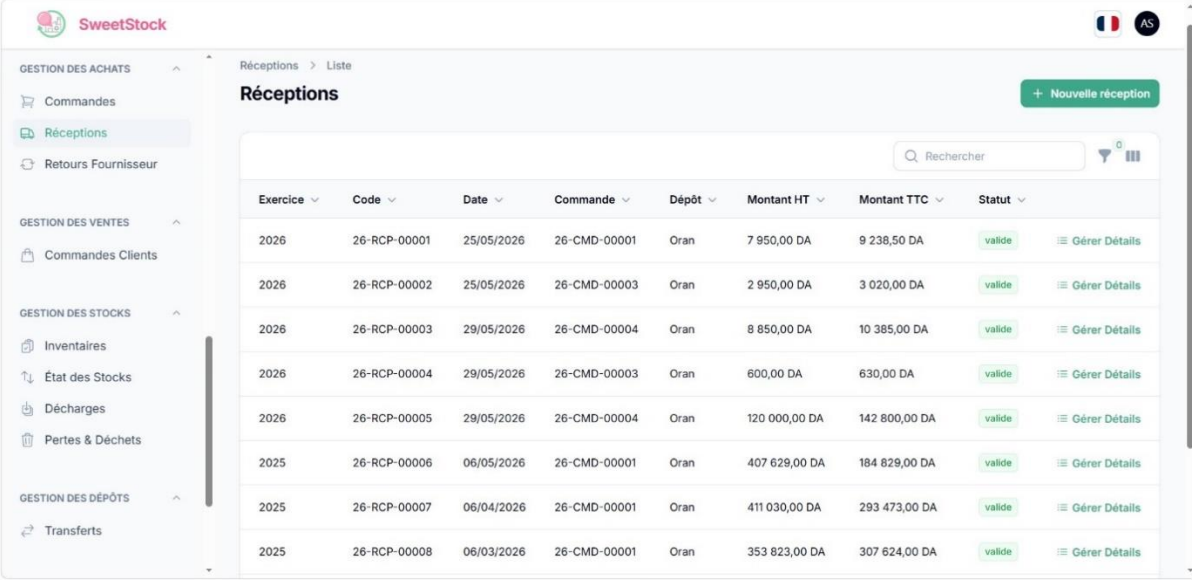


Code Commande	Date	Fournisseur	Exercice	
26-CMD-00001	07/05/2026	SUCRERIES DU NORD	2026	Modifier Gérer Détails
26-CMD-00002	14/05/2026	SUCRERIES DU NORD	2026	Modifier Gérer Détails
26-CMD-00003	14/05/2026	SUCRERIES DU NORD	2026	Modifier Gérer Détails
26-CMD-00004	18/05/2026	SUCRERIES DU NORD	2026	Modifier Gérer Détails

Figure 39 : Liste des commandes fournisseurs

IV.5.12.2. Réceptions

Les réceptions permettent d'enregistrer l'entrée en stock des marchandises à partir d'une commande fournisseur. Chaque réception est liée à une commande existante, attribuée à un dépôt de destination et associée à un exercice. Les montants hors taxes (HT) et toutes taxes comprises (TTC) sont affichés. La validation de réception est indiquée par un statut qui garantit que les quantités ne sont stockées qu'après validation de la réception. La fonctionnalité « Gérer Détails » permet d'accéder aux lignes de réception pour saisir les produits reçus, les quantités, les prix unitaires et les dates de péremption des produits soumis au suivi FEFO. La figure 40 présente la page de gestion des réceptions.

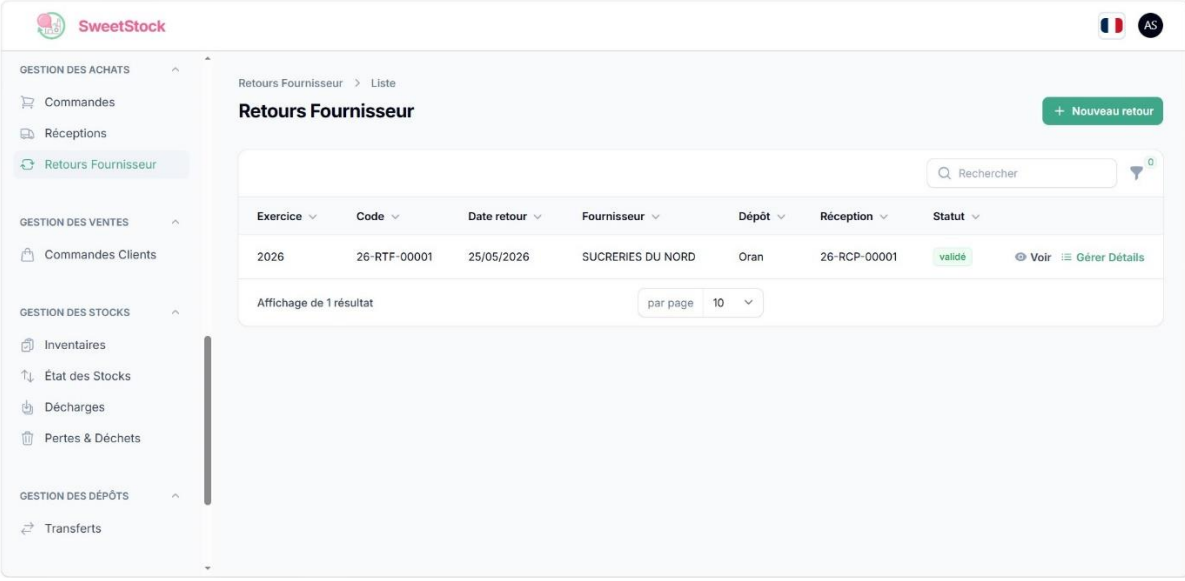


Exercice	Code	Date	Commande	Dépôt	Montant HT	Montant TTC	Statut	
2026	26-RCP-00001	25/05/2026	26-CMD-00001	Oran	7 950,00 DA	9 238,50 DA	valide	Gérer Détails
2026	26-RCP-00002	25/05/2026	26-CMD-00003	Oran	2 950,00 DA	3 020,00 DA	valide	Gérer Détails
2026	26-RCP-00003	29/05/2026	26-CMD-00004	Oran	8 850,00 DA	10 385,00 DA	valide	Gérer Détails
2026	26-RCP-00004	29/05/2026	26-CMD-00003	Oran	600,00 DA	630,00 DA	valide	Gérer Détails
2026	26-RCP-00005	29/05/2026	26-CMD-00004	Oran	120 000,00 DA	142 800,00 DA	valide	Gérer Détails
2025	26-RCP-00006	06/05/2026	26-CMD-00001	Oran	407 629,00 DA	184 829,00 DA	valide	Gérer Détails
2025	26-RCP-00007	06/04/2026	26-CMD-00001	Oran	411 030,00 DA	293 473,00 DA	valide	Gérer Détails
2025	26-RCP-00008	06/03/2026	26-CMD-00001	Oran	353 823,00 DA	307 624,00 DA	valide	Gérer Détails

Figure 40 : Liste des réceptions

IV.5.12.3. Retours fournisseur

Les retours fournisseurs facilitent la gestion des renvois de produits vers les fournisseurs, en cas de réception non conforme ou défectueuse. Chaque retour est associé à une réception source, à un fournisseur et à un dépôt, ce qui permet au système de déduire automatiquement les quantités retournées du stock. Les retours suivent le même cycle de validation que les autres modules afin de s'assurer que la sortie de stock n'est effectuée que si le retour est validé. L'option « Gérer Détails » permet d'accéder aux lignes du retour avec les produits et leurs quantités. La figure 41 présente la page de gestion des retours fournisseurs.



Exercice	Code	Date retour	Fournisseur	Dépôt	Réception	Statut	
2026	26-RTF-00001	25/05/2026	SUCRERIES DU NORD	Oran	26-RCP-00001	validé	Voir Gérer Détails

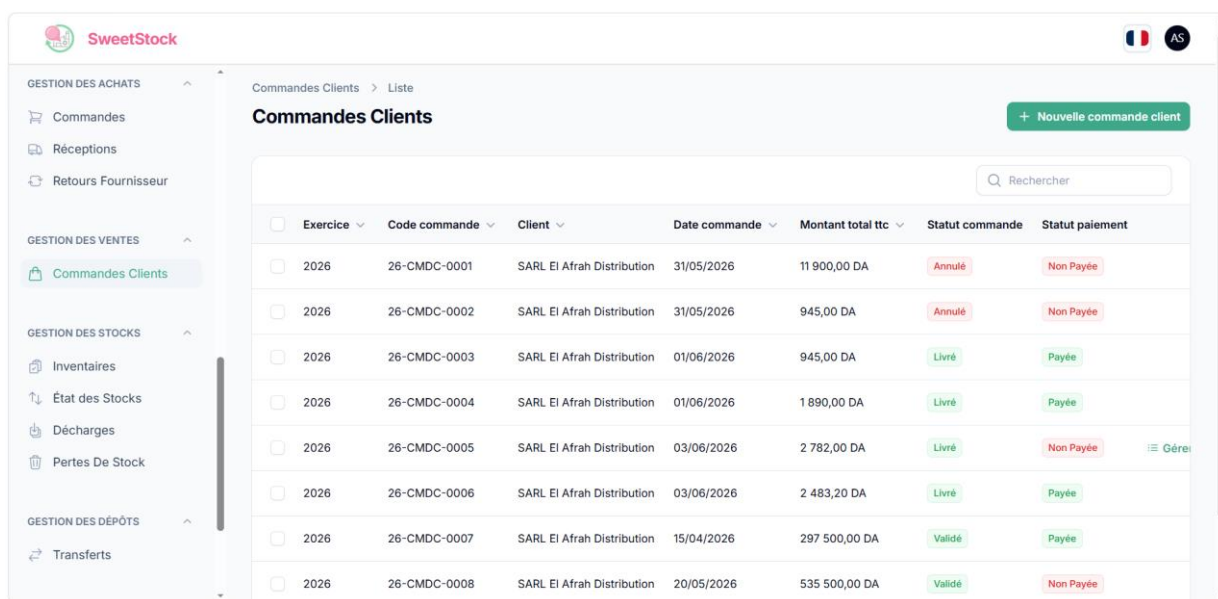
Affichage de 1 résultat

par page 10

Figure 41 : Liste des retours fournisseurs

IV.5.13. Commandes clients

Le module commandes clients permet de gérer les ventes de produits finis aux clients de l'usine. Chaque commande est identifiée par un code unique généré automatiquement et rattaché à un client, un exercice, et un statut de la commande (validée, livrée ou annulée), qui montre l'avancement de la livraison, et le statut de paiement (payée ou non payée), qui permet de suivre la situation financière de chaque commande, quel que soit son état de livraison. La fonctionnalité « Gérer Détails » offre un accès aux lignes de commande afin de renseigner les produits, les quantités et les prix, et permet de générer la facture correspondante après validation de la commande. La figure 42 présente l'interface de gestion des commandes clients.



☐	Exercice	Code commande	Client	Date commande	Montant total ttc	Statut commande	Statut paiement
☐	2026	26-CMDC-0001	SARL EI Afrah Distribution	31/05/2026	11 900,00 DA	Annulé	Non Payée
☐	2026	26-CMDC-0002	SARL EI Afrah Distribution	31/05/2026	945,00 DA	Annulé	Non Payée
☐	2026	26-CMDC-0003	SARL EI Afrah Distribution	01/06/2026	945,00 DA	Livré	Payée
☐	2026	26-CMDC-0004	SARL EI Afrah Distribution	01/06/2026	1 890,00 DA	Livré	Payée
☐	2026	26-CMDC-0005	SARL EI Afrah Distribution	03/06/2026	2 782,00 DA	Livré	Non Payée
☐	2026	26-CMDC-0006	SARL EI Afrah Distribution	03/06/2026	2 483,20 DA	Livré	Payée
☐	2026	26-CMDC-0007	SARL EI Afrah Distribution	15/04/2026	297 500,00 DA	Validé	Payée
☐	2026	26-CMDC-0008	SARL EI Afrah Distribution	20/05/2026	535 500,00 DA	Validé	Non Payée

Figure 42 : Liste des commandes clients

IV.5.14. Inventaires

Le module Inventaires permet d'effectuer des comptages physiques des stocks pour vérifier la concordance entre les quantités réelles en dépôt et celles enregistrées dans le système. Chaque inventaire est associé à un dépôt et à un exercice, et indique le nombre de lignes comptées ainsi que le nombre d'écarts constatés entre le stock théorique et le stock physique. La fonctionnalité « Gérer lignes » permet de saisir les quantités comptées pour chaque ligne. En haut de page, quatre indicateurs présentent une synthèse du total des inventaires, des inventaires en cours, de ceux avec des écarts et de ceux validés. La figure 43 affiche l'interface des inventaires.

SweetStock

GESTION DES VENTES

- Commandes Clients

GESTION DES STOCKS

- Inventaires**
- État des Stocks
- Décharges
- Pertes De Stock

GESTION DES DÉPÔTS

- Transferts

GESTION DE PRODUCTION

- Batches
- Ordres De Conditionne...
- Sessions de Dragéifica...
- Recettes

Inventaires > Liste

Inventaires

+ Nouvel inventaire

Total
2
Exercice 2026

En cours
0
Comptage actif

Écarts détectés
0
À valider

Validés
1
Ce mois

Rechercher

Code	Dépôt	Date	Exercice	Lignes	Écarts	Statut	
26-INV-00001	Tiemcen	06/06/2026	2026	0 / 0	0	annulé	Gérer lignes
26-INV-00002	Tiemcen	06/06/2026	2026	1 / 1	1	validé	Gérer lignes

Affichage de 1 à 2 sur 2 résultats

par page 10

Figure 43 : Liste des inventaires

IV.5.15. État des stocks

La page de l'état des stocks fournit une vue en temps réel des quantités disponibles par produit, lot (selon la date de péremption) et emplacement. Chaque ligne dans la liste est un lot distinct, identifié par sa date de péremption, son dépôt et son rayon, afin d'assurer une traçabilité précise au niveau du lot. L'action « Mouvements » permet de consulter l'historique des mouvements d'un produit précis (lot) concerné. La page présente d'abord une carte récapitulative des informations du produit : nom, code, catégorie, dépôt, rayon, stock actuel et date de péremption. Cette interface repose sur deux vues SQL complémentaires : (mvt_stock_view), qui agrège l'ensemble des mouvements issus des différents modules du système pour calculer le stock disponible en temps réel, et (mvt_stock_historique_view) regroupe et classe chronologiquement toutes les opérations ayant affecté un lot (ligne de stock) donné. La figure 44, État des stocks, présente le stock disponible, et la figure 45 présente l'historique des mouvements d'une ligne dans la liste des états des stocks.

Code produit	Produit	Catégorie	Type	Dépôt	Rayon	Date péremption	Stock calculé	Mouvements
P-001	Sucre	Matière Batch	Matière première	Oran	Rayon Matières Premières	01/01/2028	35,00 Kg	Mouvements
P-001	Sucre	Matière Batch	Matière première	Oran	Rayon Matières Premières	12/05/2028	100,00 Kg	Mouvements
P-001	Sucre	Matière Batch	Matière première	Oran	Rayon Matières Premières	01/03/2029	50,00 Kg	Mouvements
P-001	Sucre	Matière Batch	Matière première	Tlemcen	Rayon Matières Premières	01/03/2029	70,00 Kg	Mouvements
P-002	Sorbitol	Matière Batch	Matière première	Oran	Rayon Matières Premières	02/12/2027	100,00 L	Mouvements
P-002	Sorbitol	Matière Batch	Matière première	Oran	Rayon Matières Premières	02/02/2028	10,00 L	Mouvements
P-003	Menthol	Matière Batch	Matière première	Oran	Rayon Matières Premières	12/06/2026	20,00 Kg	Mouvements
P-003	Menthol	Matière Batch	Matière première	Oran	Rayon Matières Premières	16/05/2027	50,00 Kg	Mouvements

Figure 44 : État des stocks

Date	Sens	Type	Dépôt	Rayon	Lot (Péremption)	Quantité
03/06/2026 09:05	↓ Sortie	Decharge EXTERNE	Oran	Rayon Matières Premières	01/01/2028	- 5,00 Kg
02/06/2026 20:41	↓ Sortie	Perte (Manuelle)	Oran	Rayon Matières Premières	01/01/2028	- 0,80 Kg
29/05/2026 10:55	↓ Sortie	Batch (Consommation)	Oran	Rayon Matières Premières	01/01/2028	- 40,00 Kg
29/05/2026 10:55	↑ Entrée	Batch (Retour Ingrédient)	Oran	Rayon Matières Premières	01/01/2028	+ 0,80 Kg

Figure 45 : L'historique des mouvements d'un lot

IV.5.16. Décharges

Les décharges permettent d'enregistrer les sorties de stock non associées à une vente ni à une production. Chaque décharge est définie par un type (interne ou externe), un dépôt, un motif exprime la raison et un statut qui indique la validation. La fonctionnalité « Gérer détails » permet de saisir les produits et leurs quantités. Après validation, la décharge génère automatiquement les mouvements de sortie. De plus, il est possible de créer ou de modifier une décharge. La figure 46 présente la page des décharges.

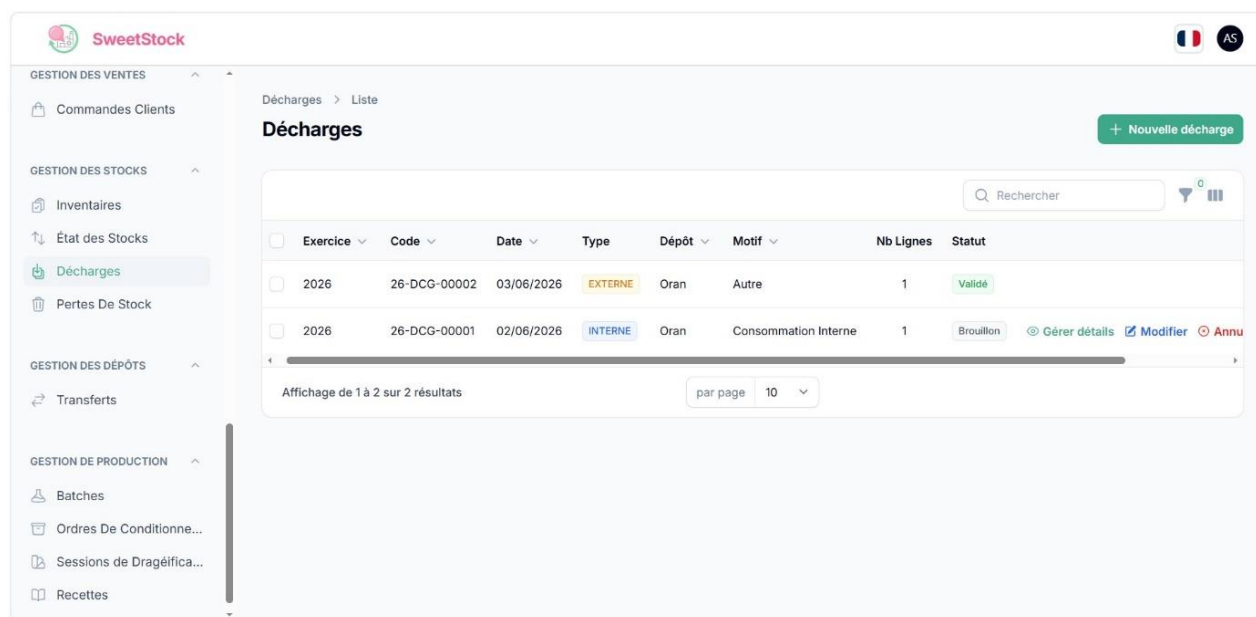


Figure 46 : Liste des décharges

IV.5.17. Pertes de stock

L'interface des pertes de stock permet la déclaration des quantités de produits perdues ou détériorées dans un dépôt. Chaque perte est enregistrée, indiquant le produit, le lot concerné, la quantité et le dépôt. Le système déduit automatiquement les quantités correspondantes du stock. La figure 47 présente la page des pertes de stock.

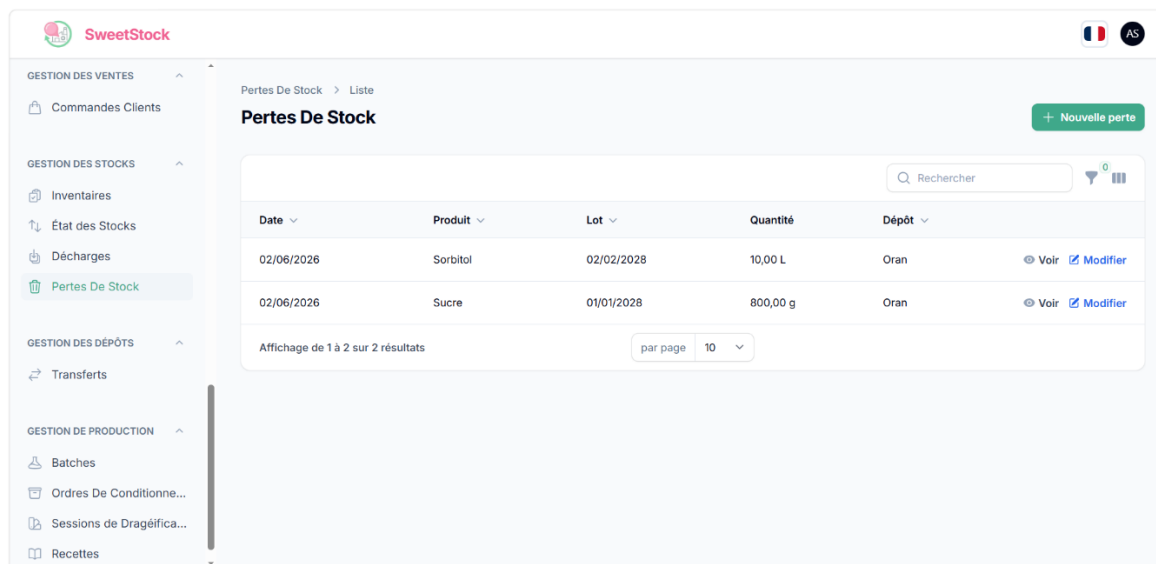


Figure 47 : Liste des pertes de stock

IV.5.18. Transferts

Les transferts permettent de gérer les mouvements de stock entre les dépôts. Chaque transfert est défini par un dépôt source et un dépôt destination et suit un cycle de validation. La fonctionnalité « Gérer Détails » permet de spécifier les produits, les lots et les quantités à transférer.

Un transfert non validé peut être modifier ou supprimé. La figure 48 présente l'interface des transferts.

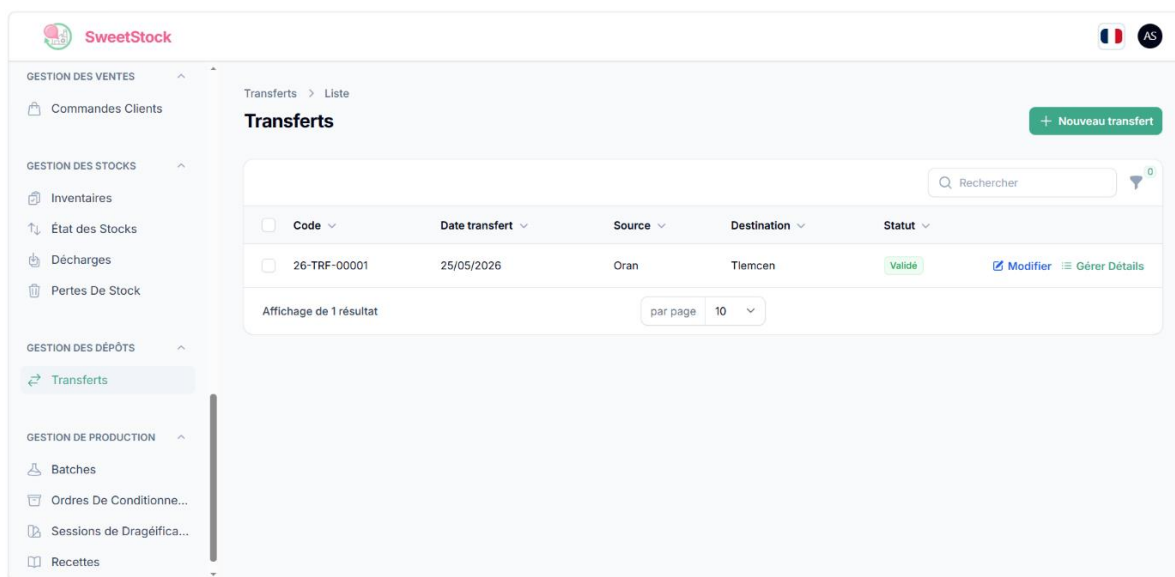


Figure 48 : Liste des transferts

IV.5.19. Gestion de production

Cette section présente les interfaces de gestion de production qui regroupe et décrit le processus de fabrication des produits finis : Recettes, batches, sessions de dragéification et les ordres de conditionnements.

IV.5.19.1. Recettes

Les recettes permettent de définir les formules de fabrication utilisées lors de la production. Chaque recette a un type (batch ou dragéification), un produit entrant (pour une recette de type dragéification), un produit sortant et une quantité de référence produite. Cette recette permet au système de calculer automatiquement les quantités de matières premières à utiliser lors du lancement d'un batch. La figure 49 présente la page des recettes.

Nom de la recette	Type	Produit Entrant (Noyau)	Produit Sortant	Qté produite	Actif	
Noyau Chewing gum Menthe 100Kg	Recette batch	-	Noyau Menthe P-005	100,00 Kg	✓	Modifier Gérer Détails
Chewing gum Menthe Dragée 100Kg	Recette dragéification	Noyau Menthe P-005	Noyau Menthe Dragée P-007	100,00 Kg	✓	Modifier Gérer Détails

Affichage de 1 à 2 sur 2 résultats

par page 10

Figure 49 : Liste des recettes

IV.5.19.2. Batches

Le module batches gère les ordres de fabrication. Chaque batch repose sur une recette de type batch, est affecté à un dépôt et suit un cycle de vie (planifié, en cours, terminé et validé). Au lancement, le système consomme automatiquement les matières premières requises selon la recette, en appliquant l'algorithme FEFO pour sélectionner les lots les plus proches de leur date de péremption. Une fois le batch terminé et la quantité produite saisie, le système calcule le rendement en comparant la production réelle à la quantité planifiée. La validation finale enregistre l'entrée du produit sortant en stock. La figure 50 présente la page de gestion des batches de production.

Recette	Dépôt	Qté Planifiée	Qté Produite	Rendement	Statut	
Noyau Chewing gum Menthe 100Kg	Oran	100,00 Kg	99,00 Kg	99,0 %	validé	Gérer Détails
Noyau Chewing gum Menthe 100Kg	Oran	100,00 Kg	-	0,0 %	planifié	Modifier Gérer Détails Annuler
Noyau Chewing gum Menthe 100Kg	Oran	40,00 Kg	-	0,0 %	en_cours	Gérer Détails Annuler

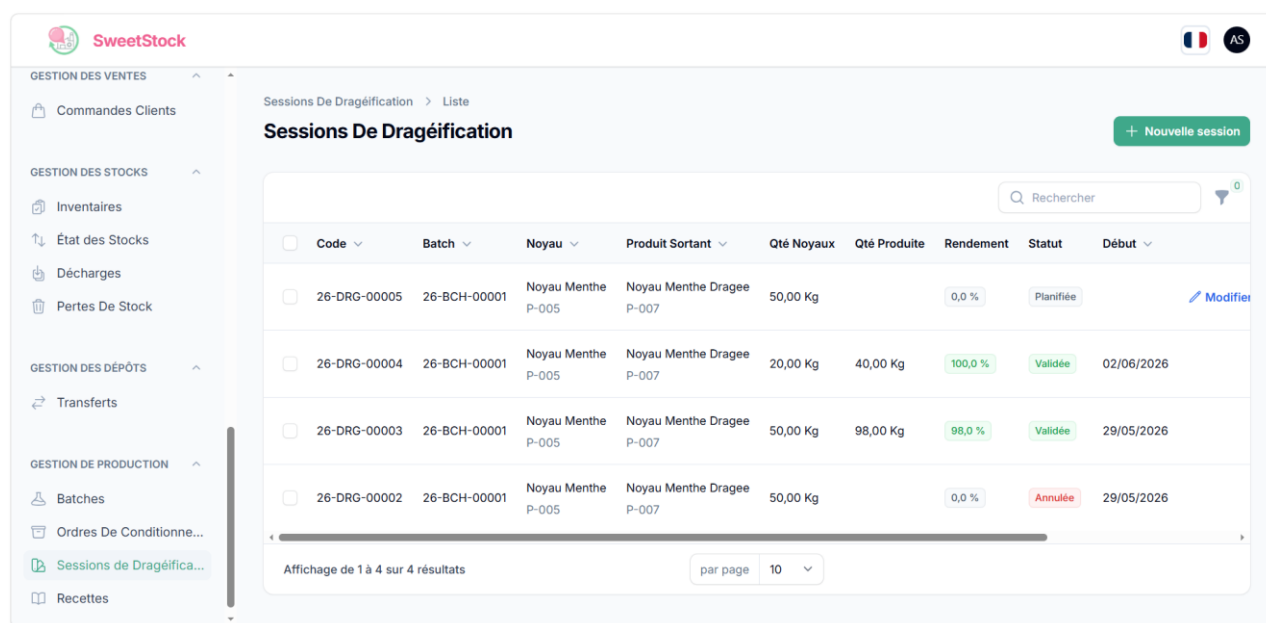
Affichage de 1 à 3 sur 3 résultats

par page 10

Figure 50 : Liste des batches

IV.5.19.3. Sessions de dragéification

Le module Sessions de Dragéification gère la phase d'enrobage des noyaux de chewing-gum. Chaque session précise le noyau utilisé en entrée et le produit dragéifié obtenu en sortie, et suit le même cycle de statuts (planifiée, en cours, validée ou annulée). Après saisie de la quantité produite, le système calcule automatiquement le rendement de la session. La validation enregistre l'entrée du produit dragéifié en stock, garantissant la traçabilité sur l'ensemble de la chaîne de production. La figure 51 présente l'interface de gestion des sessions de processus de dragéification.

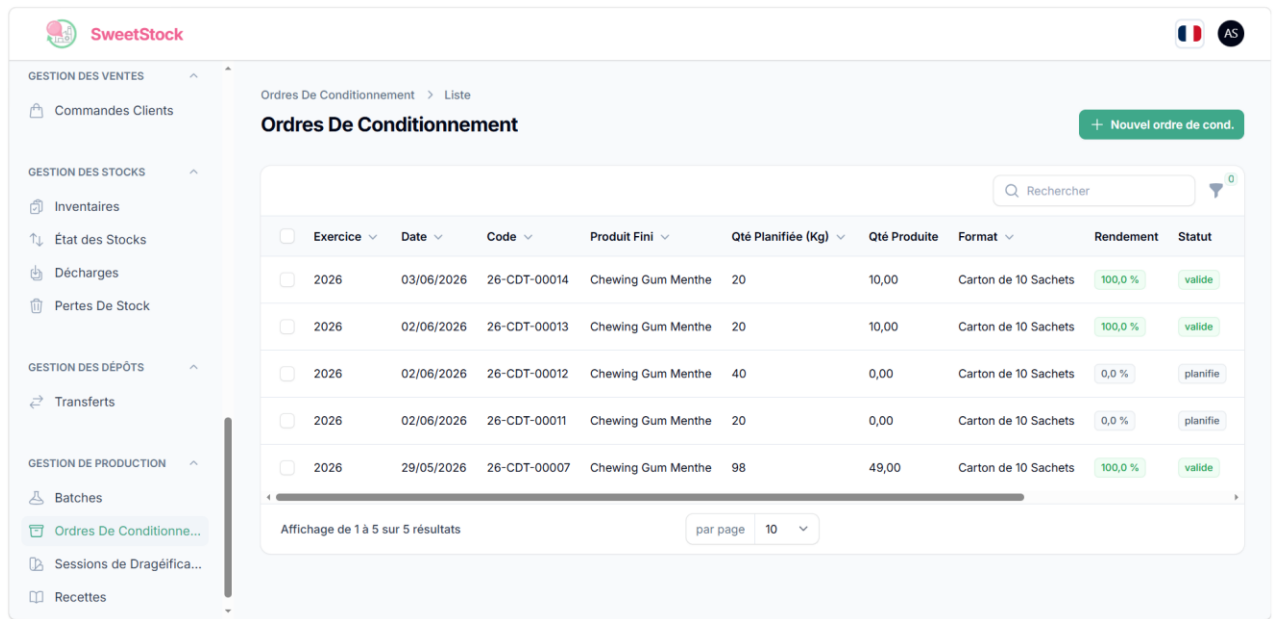


Code	Batch	Noyau	Produit Sortant	Qté Noyaux	Qté Produite	Rendement	Statut	Début
26-DRG-00005	26-BCH-00001	Noyau Menthe P-005	Noyau Menthe Dragee P-007	50,00 Kg		0,0 %	Planifiée	
26-DRG-00004	26-BCH-00001	Noyau Menthe P-005	Noyau Menthe Dragee P-007	20,00 Kg	40,00 Kg	100,0 %	Validée	02/06/2026
26-DRG-00003	26-BCH-00001	Noyau Menthe P-005	Noyau Menthe Dragee P-007	50,00 Kg	98,00 Kg	98,0 %	Validée	29/05/2026
26-DRG-00002	26-BCH-00001	Noyau Menthe P-005	Noyau Menthe Dragee P-007	50,00 Kg		0,0 %	Annulée	29/05/2026

Figure 51 : Liste des sessions de dragéification

IV.5.19.4. Ordres de conditionnement

Les ordres de conditionnement consistent à emballer le produit fini dans son conditionnement commercial. Chaque ordre spécifie le produit fini à conditionner, la quantité planifiée et le format d'emballage, et suit le même cycle de statuts que les modules précédents. Après saisie de la quantité produite, le système calcule automatiquement le rendement de l'opération. La validation de l'ordre clôt le cycle de fabrication et rend le produit disponible à la vente. La figure 52 présente l'interface de phase de conditionnement.



The screenshot shows the 'Ordres De Conditionnement' (Conditioning Orders) list in the SweetStock application. The interface includes a sidebar with navigation menus for Sales, Stocks, Depots, and Production. The main area displays a table of conditioning orders with columns for Exercise, Date, Code, Product, Planned Quantity, Produced Quantity, Format, Yield, and Status. A search bar and a 'Nouvel ordre de cond.' button are also visible.

Exercice	Date	Code	Produit Fini	Qté Planifiée (Kg)	Qté Produite	Format	Rendement	Statut
2026	03/06/2026	26-CDT-00014	Chewing Gum Menthe	20	10,00	Carton de 10 Sachets	100,0 %	valide
2026	02/06/2026	26-CDT-00013	Chewing Gum Menthe	20	10,00	Carton de 10 Sachets	100,0 %	valide
2026	02/06/2026	26-CDT-00012	Chewing Gum Menthe	40	0,00	Carton de 10 Sachets	0,0 %	planifie
2026	02/06/2026	26-CDT-00011	Chewing Gum Menthe	20	0,00	Carton de 10 Sachets	0,0 %	planifie
2026	29/05/2026	26-CDT-00007	Chewing Gum Menthe	98	49,00	Carton de 10 Sachets	100,0 %	valide

Figure 52 : Liste des ordres de conditionnement

IV.6. Conclusion

Dans ce chapitre, nous avons examiné les outils essentiels utilisés dans ce projet et la phase de mise en œuvre de notre application. Les technologies utilisées (Laravel, MySQL, Filament) nous ont fourni une base solide pour développer les fonctionnalités de l'application. Pour l'environnement de développement, nous avons choisi des outils adaptés à nos langages de programmation. Pour conclure, ce chapitre présente le travail accompli de la conception à la mise en œuvre pour ce projet.

Conclusion générale & perspectives

Dans le cadre de notre projet de fin d'études, nous avons effectué un stage au sein de l'entreprise « Naltis Communication », qui nous a confié le développement de « SweetStock », une application web de gestion des stocks et de la production dédiée à une usine spécialisée dans la fabrication de « chewing-gum ». Notre objectif principal était d'équiper l'usine d'un outil de gestion opérationnelle adapté, en réponse à l'absence de solutions spécialisées dans le secteur permettant à la fois le suivi de production par lots, la gestion des stocks de matières premières et de produits finis, ainsi que la traçabilité complète des mouvements de stock.

L'analyse comparative initiale a montré qu'aucune solution logicielle existante ne répondait aux besoins d'une PME algérienne de confiserie. Les ERP généralistes locaux manquaient de spécialisation métier, tandis que les solutions internationales dédiées faisaient face à des obstacles réglementaires, financiers et techniques en Algérie. Ce constat justifie pleinement le développement d'une solution sur mesure.

La phase d'analyse et de spécification a permis de définir les besoins fonctionnels et non fonctionnels du système, en impliquant les acteurs concernés et en formalisant leurs interactions à l'aide de diagrammes UML. La phase de conception a ensuite conduit à la mise en place d'une architecture technique adaptée et à une modélisation des données axée sur la traçabilité des mouvements de stock grâce à des vues SQL dynamiques. Pour réaliser ce projet, nous avons eu recours à un ensemble d'outils et de technologies complémentaires : Laravel et la bibliothèque FilamentPHP, MySQL pour la gestion de la base de données, Visual Studio Code comme environnement de développement principal, ainsi que des outils de conception et de modélisation.

Grâce à l'implémentation de SweetStock, nous avons développé une application web qui répond aux besoins essentiels d'une usine de confiserie. Elle propose la gestion des réceptions fournisseurs, la traçabilité des lots avec l'algorithme FEFO, la gestion de plusieurs dépôts et des tableaux de bord en temps réel. Toutes ces fonctionnalités respectent la réglementation algérienne et s'adaptent aux moyens financiers des PME locales. À la fin du projet, l'application a été déployée sur l'environnement de production de l'usine de « chewing-gum », cliente de « Naltis Communication », et est désormais pleinement opérationnelle, permettant à ses équipes de gérer quotidiennement leurs stocks, leur production et leurs commandes via l'application « SweetStock ».

En conclusion, ce projet montre l'importance d'une méthode structurée, depuis l'analyse des besoins jusqu'à la livraison du produit, pour concevoir des solutions logicielles adaptées aux entreprises. Nous espérons que les résultats de ce mémoire aideront, même modestement, à améliorer la gestion des stocks dans les petites et moyennes entreprises.

Cependant, comme tout projet informatique, l'application SweetStock ouvre la voie à plusieurs perspectives d'évolution. Parmi celles-ci, l'ajout d'un système de notifications automatiques en temps réel permettrait d'alerter les responsables dès que les seuils critiques de stock ou de qualité seront atteints. Développer une application mobile complémentaire permettant aux utilisateurs de consulter les indicateurs clés et l'état des stocks en temps réel depuis le terrain. Ainsi que l'ajout d'un module de prévision de la demande basé sur l'historique des ventes.

Nous sommes fiers du travail accompli tout au long de ce projet et confiants que « SweetStock » continuera à évoluer pour mieux répondre aux exigences opérationnelles d'une industrie de confiserie en constante croissance.

Webographie

- [1] « Présentation de l'entreprise », NALTIS. Disponible sur : <https://www.naltis.com/presentation.php>. Consultée le 14 juin 2026.
- [2] « La transformation numérique dans l'industrie », Nortek. Disponible sur : [La transformation numérique dans l'industrie - Nortek](#) Page consultée le 05/04/2026.
- [3] « Marché de la confiserie en Algérie 2026-2031 », StrategyHelix. Disponible sur: <https://strategyh.com/fr/report/confectionery-market-in-algeria/> Page consultée le 05/04/2026.
- [4] « Qu'est-ce qu'un ERP? », SAP. Disponible sur : <https://www.sap.com/france/products/erp/what-is-erp.html> Page consultée le 05/04/2026.
- [5] « Historique et évolution de l'ERP », SAP. Disponible sur : <https://www.sap.com/france/resources/what-is-erp#historique-et-%C3%A9volution-de-lerp> Page consultée le 05/04/2026.
- [6] « Fonctionnalités d'un ERP », Generix. Disponible sur : https://www.generixgroup.com/fr/glossaire/erp-pgi-definition?utm_source=chatgpt.com. Page consultée le 05/04/2026.
- [7] « Qu'est-ce qu'un logiciel ERP ? », Enodea. Disponible sur : <https://www.enodea.com/definition-logiciel-erp/> Page consultée le 05/04/2026.
- [8] « ERP généraliste VS ERP métier : lequel choisir ? », ORISHA Distribution. Disponible sur : <https://distribution.orisha.com/blog/erp/erp-generaliste-vs-erp-metier-lequel-choisir/> Page consultée le 26/04/2026.
- [9] « Megatheque Paiesoft », comparili. Disponible sur: [Megatheque Paiesoft - Comparili.net](#) Page consultée le 26/04/2026.
- [10] « SYNAPSE : un ERP complet et performant pour une gestion harmonieuse d'une entreprise », Megatheque. Disponible sur: <https://megatheque.net/erp-algerie/> Page consultée le 26/04/2026.
- [11] « Megatheque SYNAPSE », Comparili. Disponible sur: https://comparili.net/produit/megatheque-synapse-ensemble-erp-logiciel-algerie-comparili/#google_vignette Page consultée le 26/04/2026.
- [12] « CIRTASOFT » CIRTASOFT. Disponible sur : <https://www.cirtasoft.com/>. Page consultée le 26/04/2026.
- [13] « Gérez votre entreprise avec précision & efficacité » CIRTASOFT. Disponible sur: <https://www.cirtasoft.com/logiciel-erp-algerie/>. Page consultée le 26/04/2026.
- [14] Société GestiumERP, Capture d'écran du tableau de bord du logiciel GestiumERP, document interne transmis par communication personnelle, le 12 avril 2026.
- [15] « ERP généraliste vs ERP dédié aux entreprises industrielles », ABAS FORTERRO. Disponible sur <https://abas-erp.com/fr/blog/erp-generaliste-dedie-pme-industrielles>. Page

consultée le 26/04/2026.

[16] « AKANEA » AKANEA. Disponible sur : <https://contact.akanea.com/5-tendances-sc-2023/>
Page consultée le 26/04/2026.

[17] «ERP KONNECT'AGRO pour l'industrie agroalimentaire », AKANEA.
Disponible sur : <https://akanea.com/agroalimentaire/konnectagro/> Page consultée le 26 avril 2026.

[18] «Sage - Solution de gestion et de comptabilité pour les entreprises», turnk. Disponible sur :
<https://www.turnk.co/outils-nocode/sage> consultée le 26 avril 2026.

[19] « Sage X3 : bien plus qu'un ERP », Sage. Disponible sur <https://www.sage.com/fr-fr/sage-business-cloud/sage-x3/> consultée le 26 avril 2026.

[20] « SAP S/4HANA vs Sage X3 : quel ERP choisir pour votre entreprise en 2025 ? ». APPVIZE.
Disponible sur : <https://www.appvizer.fr/magazine/operations/erp/saps4hana-vs-sagex3> consultée le 08 avril 2026.

[21] « Déclaration d'accessibilité pour Sage X », Sage. Disponible sur <https://www.sage.com/fr-ca/accessibilite/declarations/sage-x3/> consultée le 26 avril 2026.

[22] « Requirements analysis in systems engineering with IBM Engineering Systems Design Rhapsody », IBM. Disponible sur <https://www.ibm.com/docs/fr/engineering-lifecycle-management-suite/design-rhapsody/10.0.2?topic=rhapsody-requirements-analysis> Page consultée le 04 mai 2026.

[23] « Acteur (UML) », Wikipédia. Disponible sur :
https://fr.wikipedia.org/wiki/Acteur_%28UML%29. Page consultée le 04 mai 2026.

[24] Fien VAN DER HEYDE et Laurent DEBRAUWER, « La modélisation des exigences », dans *UML 2.5 : Initiation, exemples et exercices corrigés (5e édition)*. Disponible sur :
<https://www.editions-eni.fr/livre/uml-2-5-initiation-exemples-et-exercices-corriges-5e-edition-9782409024085/la-modelisation-des-exigences>. Page consultée le 04 mai 2026.

[25] « User Stories », IBM Documentation. Disponible sur :
<https://www.ibm.com/docs/en/targetprocess/atp/saas?topic=model-user-stories>. Page Consultée le 08 juin 2026.

[26] « Différence entre les exigences fonctionnelles et non fonctionnelles », Prodiffs.
Disponible sur : <https://fr.prodiffs.com/article/difference-between-functional-and-non-functional-requirements>. Page consultée le 09 mai 2026.

[27] « Le diagramme de cas d'utilisation : structure et fonction ». IONOS. Disponible sur :
<https://www.ionos.fr/digitalguide/sites-internet/developpement-web/diagramme-de-cas-de-utilisation/>. Page consultée le 08 juin 2026.

- [28] « UML - Diagrammes de cas d'utilisation », Tutoriels.edu.lat. Disponible sur : <https://tutoriels.edu.lat/pub/uml/uml-use-case-diagram/uml-diagrammes-de-cas-d-utilisation>. Page consultée le 09 mai 2026.
- [29] « Comprendre les scénarios de cas d'utilisation : modélisation du comportement du système avec des parcours normaux, d'exception et alternatifs », Visual Paradigm. Disponible sur : <https://guides.visual-paradigm.com/fr/understanding-use-case-scenarios-modeling-system-behavior-with-normal-exception-and-alternative-paths/>. Page consultée le 09 mai 2026.
- [30] « Tutoriel UML – Cas d'utilisation », HELMo. Disponible sur : <https://dartagnan.cg.helmo.be/~p150107/tutoriels/uml-uc/>. Page consultée le 09 mai 2026.
- [31] « Understanding Sequence Diagram Notation in UML », Visual Paradigm, Disponible sur: <https://guides.visual-paradigm.com/fr/understanding-sequence-diagram-notation-in-uml/> Page consultée le 09 mai 2026.
- [32] « Diagrammes d'activité », IBM. Disponible sur : <https://www.ibm.com/docs/fr/rsas/7.5.0?topic=diagrams-activity>. Page consultée le 10 mai 2026.
- [33] « Diagrammes de classes ». IBM. Disponible sur : [IBM Docs – Diagrammes de classes](#). Consulté le 15 mai 2026.
- [34] « Qu'est-ce que la modélisation des données ? ». IBM. Disponible sur : <https://www.ibm.com/fr-fr/think/topics/data-modeling>. Consulté le 15 mai 2026.
- [35] « Guide pratique pour utiliser le modèle Master-Detail avec des données hiérarchiques ». Microsoft. Disponible sur : <https://learn.microsoft.com/fr-fr/dotnet/desktop/wpf/data/how-to-use-the-master-detail-pattern-with-hierarchical-data>. Consulté le 11 juin 2026.
- [36] « Support de cours Laravel ». Brosseau. Disponible sur : <https://cours.brosseau.ovh/tp/laravel/support.html>. Consulté le 05 juin 2026.
- [37] « Qu'est-ce que le SQL (Structured Query Language) ? ». IBM. Disponible sur : <https://www.ibm.com/fr-fr/think/topics/structured-query-language>. Consulté le 05 juin 2026.
- [38] « What is Filament? ». Filament Documentation, Disponible sur : <https://filamentphp.com/docs/5.x/introduction/overview>. Consulté le 05 juin 2026.
- [39] « Visual Studio Code (VSCODE) : qu'est-ce que c'est ? ». Futura Sciences. Disponible sur : <https://www.futura-sciences.com/tech/definitions/logiciels-visual-studio-code-vscode-27671/>. Consulté le 05 juin 2026.
- [40] « MySQL Workbench Reference Manual ». Oracle. Disponible sur

:<https://dev.mysql.com/doc/workbench/en/>. Consulté le 05 juin 2026.

[41] « À propos de GitHub et Git ». GitHub. Disponible sur : <https://docs.github.com/fr/get-started/start-your-journey/about-github-and-git>. Consulté le 05 juin 2026.

[42] « draw.io : logiciel de diagrammes et de modélisation ». Appvizer. Disponible sur : <https://www.appvizer.fr/services-informatiques/diagramme/drawio>. Consulté le 05 juin 2026.

[43] « dbdiagram.io : outil de conception et de visualisation de bases de données ». WebCatalog. Disponible sur : <https://webcatalog.io/fr/apps/dbdiagram-io>. Consulté le 05 juin 2026.

[44] « Logiciel de gestion de projet en ligne ». Lucidchart. Disponible sur : <https://www.lucidchart.com/pages/fr/exemple/logiciel-gestion-de-projet>. Consultée le 09 juin 2026.

[45] « Outils Agile destinés aux équipes de développement ». Jira. Disponible sur : <https://www.atlassian.com/fr/software/jira/agile>. Consultée le 11 juin 2026.

Résumé

Nous avons conçu et développé une application web « SweetStock », destinée à digitaliser et centraliser la gestion des stocks et le processus de production d'une usine de confiserie en Algérie. Elle couvre l'ensemble du flux de fabrication, depuis l'achat des matières premières jusqu'au conditionnement final des produits finis, avec une traçabilité complète des lots et une vérification automatique des stocks avant chaque lancement de production. Le système utilise aussi un algorithme FEFO pour optimiser l'utilisation des ingrédients, fournit des indicateurs de performance en temps réel et gère plusieurs dépôts. Cela donne à l'entreprise une vue d'ensemble de ses opérations et réduit les risques d'erreurs liés aux fichiers Excel.

Mots clés : Application web, gestion des stocks, production, traçabilité, gestion des produits, suivi des ateliers de fabrication, gestion multi-dépôts.

Abstract

We designed and developed a web application « SweetStock », to digitize and centralize inventory management and the production process for a confectionery factory in Algeria. It covers the entire manufacturing flow, from raw material procurement to final product packaging, with complete batch traceability and automatic stock verification before each production launch. The system also uses a FEFO algorithm to optimize ingredient usage, provides real-time performance indicators, and manages multiple warehouses. This gives the company a comprehensive overview of its operations and reduces the risk of errors associated with Excel files.

Keywords: Web application, inventory management, production, traceability, products management, manufacturing workshop monitoring, multi-warehouse management.

ملخص

صممنا وطورنا تطبيقًا إلكترونيًا «SweetStock» ، لرقمنة وتوحيد إدارة المخزون وعملية الإنتاج في مصنع حلويات بالجزائر. يغطي التطبيق كامل عملية التصنيع، بدءًا من شراء المواد الخام وصولًا إلى تغليف المنتج النهائي، مع إمكانية تتبع كاملة للدفعات والتحقق التلقائي من المخزون قبل كل عملية إنتاج. كما يستخدم النظام خوارزمية (FEFO) لتحسين استخدام المكونات، ويوفر مؤشرات أداء فورية، ويدير عدة مستودعات. يمنح هذا الشركة نظرة شاملة على عملياتها ويقلل من مخاطر الأخطاء المرتبطة بملفات إكسل.

الكلمات المفتاحية: تطبيق ويب، إدارة المخزون، الإنتاج، التتبع، إدارة المواد الخام، مراقبة خط الإنتاج، تحسين المخزون، إدارة المستودعات المتعددة.