

Diagramme UML

1. Introduction Générale [🔗](#)

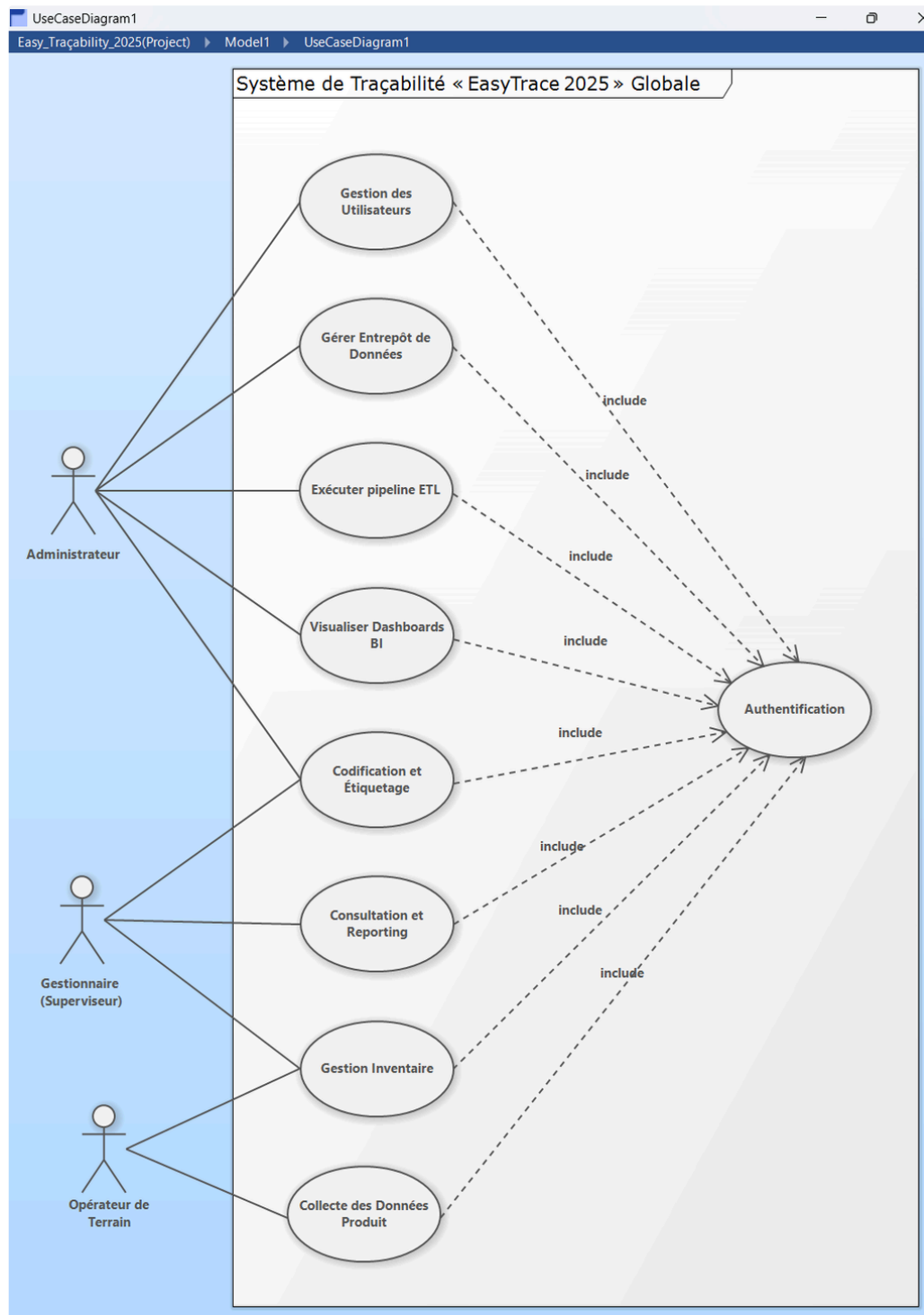
Dans un contexte de concurrence accrue, la gestion efficace des stocks et la traçabilité des produits sont devenues des enjeux stratégiques majeurs pour les commerçants, quel que soit leur secteur d'activité. L'adoption de solutions numériques intégrant des technologies comme les QR codes, les étiquettes RFID/NFC et les bases décisionnelles permet non seulement de garantir la conformité réglementaire, mais aussi de renforcer la transparence, la qualité de service et la satisfaction des clients.

Ce projet de fin d'études s'inscrit dans cette dynamique, en proposant une application web de codification et de gestion des produits, destinée à faciliter la traçabilité et la gestion des stocks pour les petites et moyennes entreprises (PME). L'objectif est de permettre aux commerçants de générer des étiquettes intelligentes pour chaque produit, de suivre le parcours de ceux-ci à travers la chaîne logistique et d'en extraire des indicateurs décisionnels pertinents via une base de données dédiée.

La solution repose sur l'intégration de plusieurs technologies complémentaires : un système de génération d'étiquettes QR/NFC, un terminal mobile pour la lecture et l'enregistrement des événements, ainsi qu'un entrepôt de données (data warehouse) permettant d'assurer une exploitation intelligente de l'information. Ce projet s'appuie également sur des méthodes de Business Intelligence (BI) et d'ETL (Extract, Transform, Load) pour transformer les données brutes en informations utiles à la prise de décision.

Ce rapport est structuré en plusieurs chapitres. Le premier chapitre présente le cadre général du projet, l'entreprise d'accueil, l'état de l'art des technologies mobilisées ainsi que la méthodologie de travail adoptée. Le deuxième chapitre détaillera les étapes de conception et d'implémentation de la solution proposée. Le dernier chapitre exposera les résultats obtenus, les difficultés rencontrées ainsi que les perspectives d'évolution de cette solution.

2. Diagrammes de Cas d'Utilisation et leurs Raffinements



2.1 Vue Globale et Identification des Acteurs

Le diagramme de cas d'utilisation global présente la vision d'ensemble du système de traçabilité « Easy Trace 2025 ». On y identifie trois acteurs principaux :

- **Administrateur**
- **Gestionnaire (Superviseur)**

- **Opérateur de Terrain**

Chaque acteur a des responsabilités et des accès différents, en fonction du rôle et des tâches à réaliser dans le processus global de gestion des stocks, de collecte des données et de reporting.

2.2 Découpage des Cas d’Utilisation et Relations [↗](#)

Le diagramme global illustre de manière synthétique plusieurs cas d'utilisation majeurs, notamment :

- **Gestion des Utilisateurs**
Permet à l'administrateur de gérer les comptes ainsi que les permissions attribuées aux différents utilisateurs.
- **Gérer Entrepôt de Données**
Offre à l'administrateur la possibilité d'organiser et de superviser le stockage des données dans le système.
- **Exécuter pipeline ETL**
Autorise l'administrateur à lancer des processus d'extraction, de transformation et de chargement (ETL) des informations dans l'entrepôt de données.
- **Visualiser Dashboards BI**
Permet à l'administrateur d'accéder à des tableaux de bord interactifs pour analyser les données et obtenir des indicateurs de performance.
- **Codification et Étiquetage**
Concerne le gestionnaire (superviseur) qui peut effectuer le codage et l'étiquetage des produits pour assurer leur traçabilité.
- **Consultation et Reporting**
Offre au gestionnaire la possibilité de consulter les données collectées et de générer des rapports pour le suivi et la prise de décisions.
- **Gestion Inventaire**
Permet également au gestionnaire de surveiller et de gérer l'inventaire des produits liés à la chaîne de traçabilité.
- **Collecte des Données Produit**
C'est la fonctionnalité dédiée à l'opérateur de terrain qui collecte directement les informations sur les produits, sur le terrain.

Un point transversale et fondamental du système est le cas d'utilisation **"Authentication"**, qui est inclus dans tous les autres cas pour garantir l'accès sécurisé aux différentes fonctionnalités.

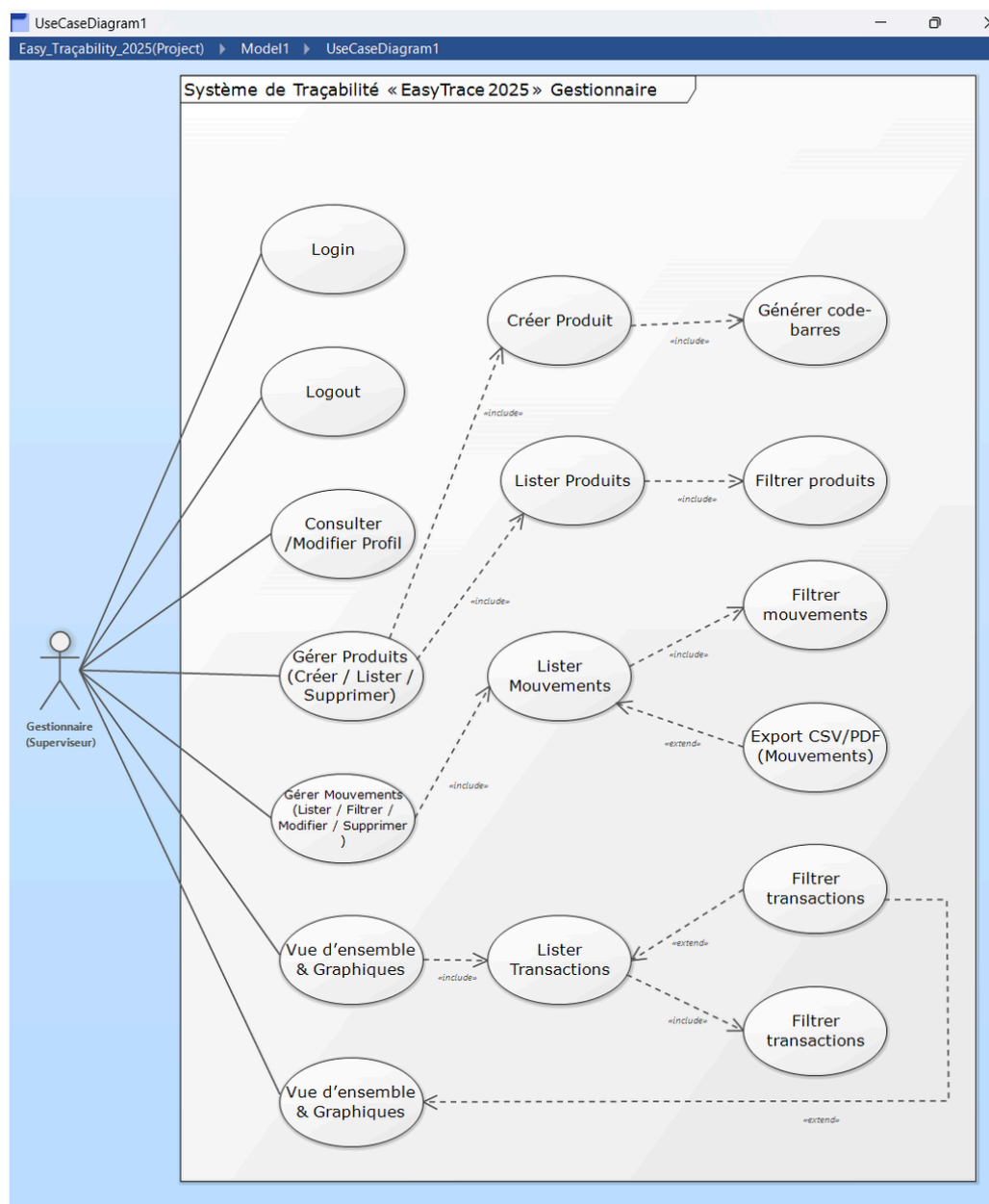
Tableau explicatif des cas d’utilisation [↗](#)

Cas d'utilisation	Description
Gestion des Utilisateurs	Permet à l'administrateur de créer, modifier ou supprimer les comptes utilisateurs et de définir les droits d'accès.
Gérer Entrepôt de Données	Offre à l'administrateur la possibilité d'organiser le stockage, la structuration et la maintenance des données du système.
Exécuter pipeline ETL	Autorise l'administrateur à déclencher les processus d'extraction, transformation et chargement afin d'intégrer les données.
Visualiser Dashboards BI	Permet à l'administrateur de consulter des tableaux de bord interactifs et de suivre des indicateurs clés de performance.
Codification et Étiquetage	Permet au gestionnaire d'attribuer des codes et des étiquettes aux produits, facilitant ainsi leur identification et leur suivi.
Consultation et Reporting	Permet au gestionnaire de consulter les données enregistrées et de générer des rapports pour l'analyse et la prise de

	décision.
Gestion Inventaire	Permet au gestionnaire de suivre et de gérer l'inventaire des produits, assurant un contrôle précis des stocks.
Collecte des Données Produit	Permet à l'opérateur de terrain de collecter directement sur le terrain les informations relatives aux produits.
Authentification	Processus de vérification d'identité nécessaire pour accéder à l'ensemble des autres fonctionnalités du système.

Les raffinements apportés aux diagrammes (dédiés séparément pour l'Administrateur, le Gestionnaire et l'Opérateur de Terrain) précisent les spécificités de chacun :

- **Pour le Gestionnaire (Superviseur) :**



Description Textuelle Détaillée du Raffinement pour le Manager

Dans ce raffinement, le Manager est le garant de la supervision opérationnelle et stratégique du système. Son objectif est de s'assurer de la qualité des données et de la cohérence des opérations en matière de traçabilité. Pour cela, il interagit avec plusieurs cas d'utilisation spécifiques dès qu'il s'identifie via un processus d'**Authentification** sécurisé.

Une fois authentifié, le Manager accède aux fonctionnalités clés suivantes :

1. Codification et Étiquetage

Le Manager supervise la mise en place et la validation des codes et des étiquettes attribués aux produits ou lots. Cela permet d'assurer une identification unique et une traçabilité précise, facilitant le suivi des produits dans l'ensemble du système.

2. Consultation et Reporting

À travers cette fonctionnalité, le Manager peut consulter les données consolidées provenant des différentes sources du système. Il génère ensuite des rapports détaillés pour analyser la performance, détecter des anomalies et orienter la prise de décision stratégique.

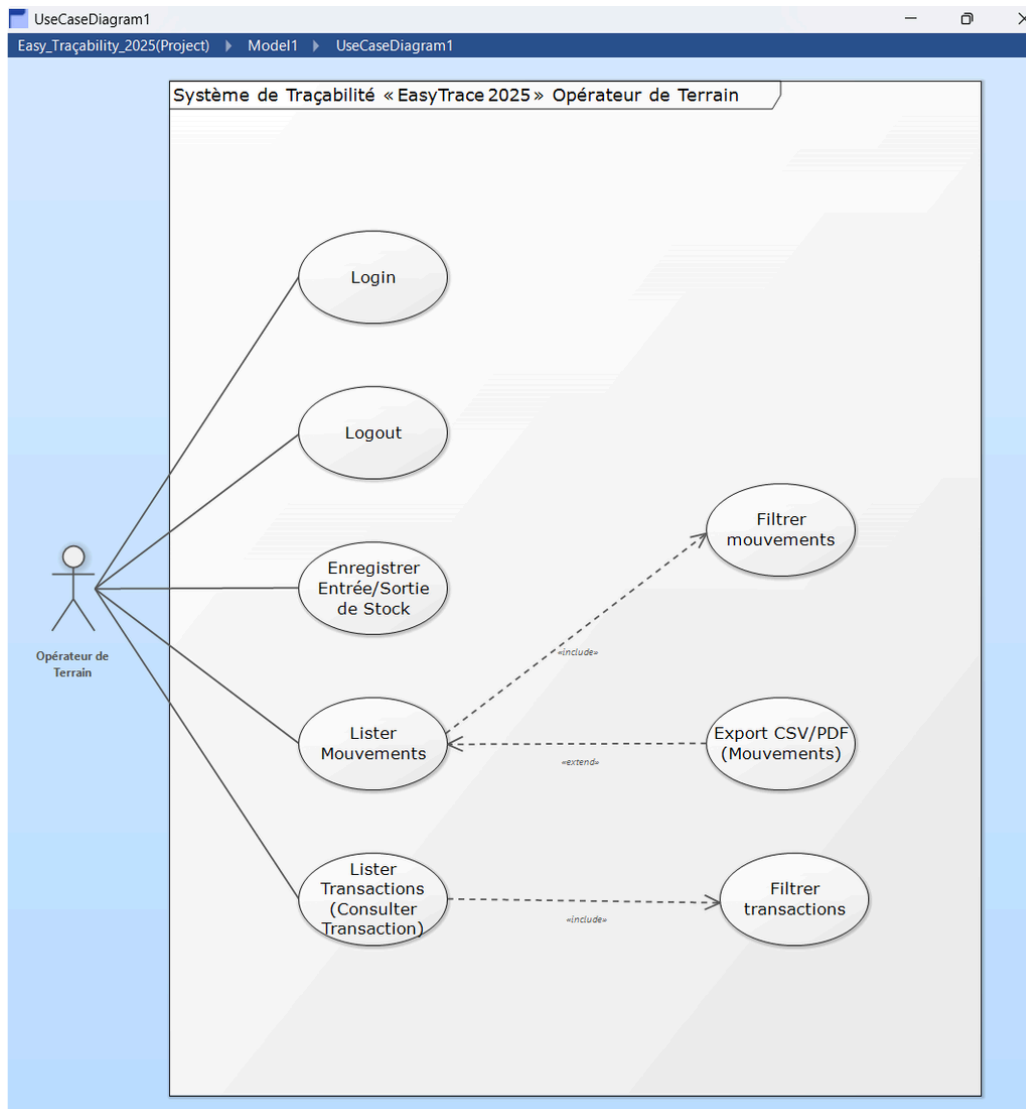
3. Gestion de l'Inventaire

Le Manager surveille activement l'état des stocks en temps réel. Ce processus inclut la vérification des mouvements d'entrées et de sorties, garantissant ainsi la correspondance entre le stock physique et les enregistrements numériques. Cette supervision permet d'anticiper les besoins et d'éviter les ruptures.

Tableau Explicatif des Cas d'Utilisation pour le Manager

Cas d'utilisation	Description
Authentification	Processus de vérification de l'identité qui assure que le Manager accède de manière sécurisée aux fonctionnalités qui lui sont réservées.
Codification et Étiquetage	Supervision et validation de l'attribution des codes et étiquettes aux produits, garantissant une identification claire et une traçabilité optimale.
Consultation et Reporting	Consultation des données consolidées et génération de rapports analytiques détaillés pour suivre la performance et détecter des anomalies critiques.
Gestion de l'Inventaire	Suivi en temps réel des stocks, vérification des mouvements et via une gestion rigoureuse pour assurer la cohérence entre les données enregistrées et le stock physique.

- Pour l'Opérateur de Terrain :



Description Textuelle Détaillée du Raffinement pour l'Opérateur [🔗](#)

L'Opérateur de Terrain intervient directement sur le terrain pour recueillir et enregistrer les données essentielles au système de traçabilité. Dès l'ouverture de session via une authentification sécurisée, ses actions se concentrent sur la mise à jour immédiate des informations relatives aux flux de produits. Ses principales fonctionnalités sont :

1. Enregistrement des Mouvements de Stock

L'opérateur saisit en temps réel les entrées et sorties de stock. Ce processus permet d'actualiser immédiatement la disponibilité des produits et d'assurer que chaque mouvement est correctement consigné.

2. Consultation et Filtrage des Mouvements

Après l'enregistrement, l'opérateur peut consulter la liste des mouvements récents. Il a la possibilité de filtrer ces informations selon des critères précis (par exemple, par date ou par type de mouvement) pour identifier rapidement des irrégularités ou des données spécifiques.

3. Exportation des Données

Pour faciliter le partage ou l'archivage, l'opérateur peut exporter sous format CSV ou PDF les listes de mouvements. Cela permet d'assurer une communication fluide des informations collectées avec d'autres parties prenantes du système (par exemple, pour des audits ou des analyses ultérieures).

4. Consultation et Filtrage des Transactions

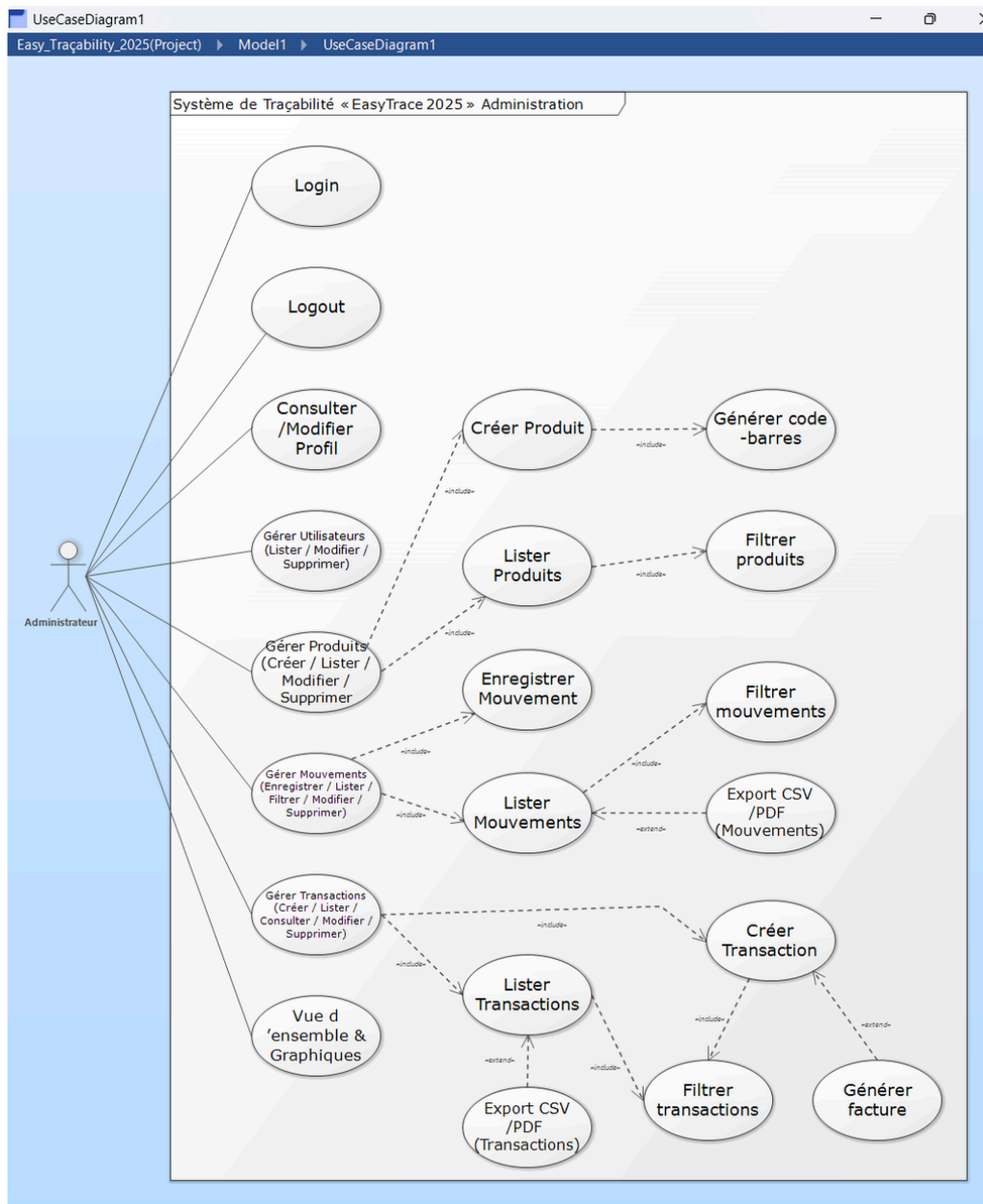
Parallèlement à l'enregistrement des mouvements, l'opérateur peut également accéder aux transactions déjà enregistrées dans le système. Une fonctionnalité de filtrage lui permet de restreindre l'affichage aux transactions pertinentes selon

différents critères (date, nature de la transaction, etc.), garantissant ainsi une vérification de la cohérence des données saisies.

Tableau Explicatif des Cas d'Utilisation pour l'Opérateur [↗](#)

Cas d'utilisation	Description
Authentification	Processus de vérification qui permet à l'opérateur de se connecter de manière sécurisée afin d'accéder aux outils de saisie et de consultation.
Enregistrer Entrée/Sortie de Stock	Action de saisir en temps réel les mouvements de stock (entrées et sorties), mettant à jour l'inventaire dès qu'un produit est réceptionné ou expédié.
Lister Mouvements	Consultation de la liste des mouvements enregistrés, permettant de visualiser les actions récentes sur le stock.
Filtrer Mouvements	Fonction intégrée à la consultation des mouvements, permettant de trier les données par date, type ou autre critère spécifique pour affiner la recherche.
Exporter CSV/PDF (Mouvements)	Option d'exportation des données de mouvements dans un format CSV ou PDF, facilitant l'archivage, le partage et l'analyse externe.
Lister Transactions (Consulter Transactions)	Permet de visualiser l'ensemble des transactions enregistrées, offrant un aperçu détaillé et validé des actions passées.
Filtrer Transactions	Fonction de filtrage appliquée à la liste des transactions pour cibler des périodes ou des types spécifiques de transactions, permettant ainsi une vérification fine.

- Pour l'Administrateur :



Dans ce raffinement, le système se concentre sur les fonctionnalités auxquelles l'administrateur a accès. Tout commence par un processus d'**authentification** obligatoire qui garantit, de manière sécurisée, que seul un administrateur correctement vérifié peut accéder aux fonctionnalités de gestion du système.

Une fois authentifié, l'administrateur interagit avec plusieurs cas d'utilisation essentiels :

1. Gestion des Utilisateurs

L'administrateur peut créer, modifier ou supprimer les comptes. Cela comprend la définition des droits d'accès et des permissions pour chaque utilisateur, permettant ainsi de contrôler qui peut interagir avec les autres modules du système.

2. Gestion de l'Entrepôt de Données

Cette fonctionnalité offre la capacité d'organiser, de structurer et de maintenir le stockage centralisé des données.

L'administrateur peut ainsi configurer les schémas de données, gérer les sauvegardes et assurer la cohérence et l'intégrité de l'information dans le système.

3. Exécution du Pipeline ETL

L'administrateur est responsable du lancement et de la supervision des processus d'**Extraction**, de **Transformation** et de **Chargement** des données. Ce pipeline permet d'intégrer de manière régulière et fiable les données provenant de diverses sources dans l'entrepôt, garantissant des informations à jour pour les analyses ultérieures.

4. Visualisation des Dashboards BI

Grâce à cette fonctionnalité, l'administrateur peut accéder à des tableaux de bord interactifs. Ces dashboards offrent une analyse visuelle des indicateurs clés de performance, facilitant la prise de décision et le suivi de l'efficacité du système.

Ce raffinement met en avant la centralité du rôle de l'administrateur dans la configuration, la sécurisation et la surveillance globale du système de traçabilité. Chaque processus est précédé par l'authentification, garantissant une gestion sécurisée et rigoureuse des données et des opérations.

Tableau Explicatif des Cas d'Utilisation pour l'Administrateur

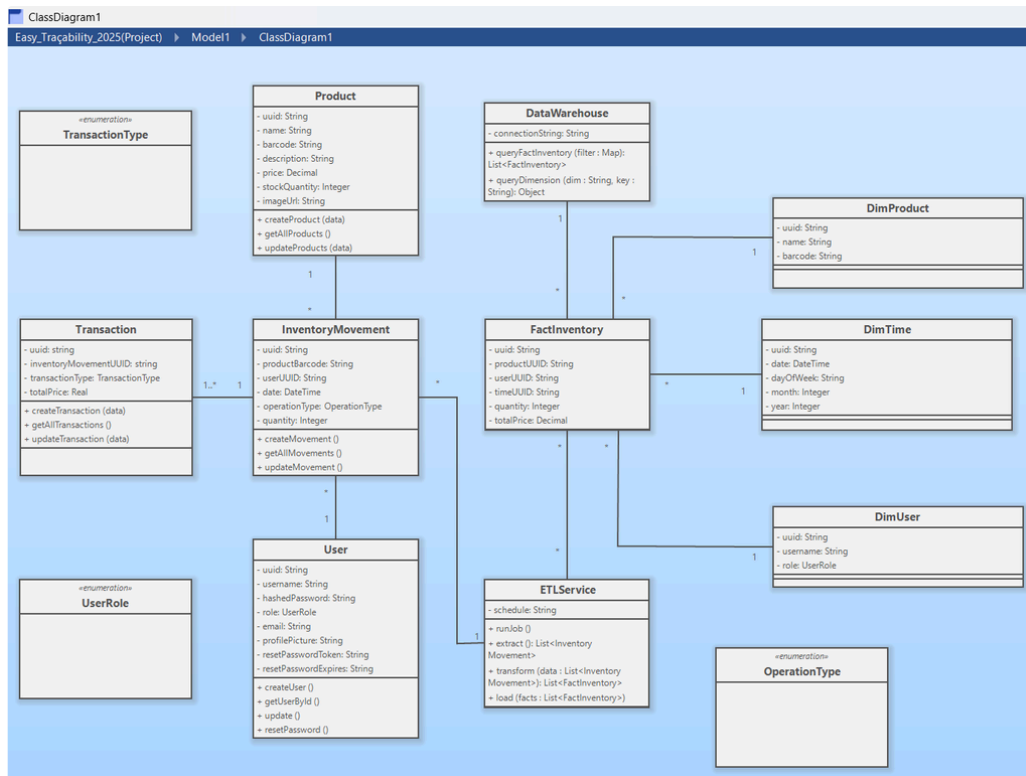
Cas d'utilisation	Description
Authentification	Processus de vérification de l'identité qui sécurise l'accès aux fonctionnalités, assurant que seuls les administrateurs validés peuvent opérer.
Gestion des Utilisateurs	Permet de créer, modifier et supprimer les comptes utilisateurs ainsi que de gérer les droits d'accès, formant le socle de la sécurité du système.
Gestion de l'Entrepôt de Données	Organisation, structuration et maintenance du stockage des données. L'administrateur configure, gère les sauvegardes et veille à l'intégrité des données.
Exécution du Pipeline ETL	Lancement et supervision des processus d'extraction, transformation et chargement, essentiels pour intégrer et mettre à jour les données dans l'entrepôt.
Visualisation des Dashboards BI	Accès à des tableaux de bord interactifs qui offrent une analyse en temps réel des indicateurs clés, permettant une prise de décision éclairée.

3. Diagramme de Classes

3.1 Présentation Générale et Rôle dans l'Architecture

Le diagramme de classes offre une représentation statique de la structure logicielle du système Easy Traçability. Il traduit le modèle objet en définissant les classes principales qui constituent le domaine du projet : **Product**, **Transaction**, **InventoryMovement** et **User**. Ce diagramme illustre à la fois les attributs (représentant l'état) et les méthodes (décrivant les opérations possibles sur ces états).

3.2 Description des Classes et Attributs



Description Textuelle du Diagramme de Classes

Le diagramme de classes structure la logique statique et la répartition des responsabilités dans le système. Il se fonde sur la décomposition des rôles et des opérations issues du diagramme de cas d'utilisation en différentes entités logicielles qui interagissent pour fournir les fonctionnalités attendues.

1. Classe Abstraite Utilisateur

Au cœur du modèle se trouve la classe abstraite `Utilisateur` qui définit les attributs et méthodes communs à tous les profils utilisateurs. Celle-ci encapsule les informations génériques telles que l'identifiant, le nom, le mot de passe et l'email, ainsi qu'une méthode d'authentification :

- **Attributs communs** : `idUtilisateur`, `nom`, `motDePasse`, `email`.
- **Méthode commune** : `authentifier()`, utilisée pour vérifier les identifiants lors de la connexion.

2. Spécialisation par Héritage

Les différents rôles du système sont dérivés de la classe `Utilisateur` grâce à l'héritage :

• Administrateur

Hérite de `Utilisateur` et ajoute des opérations spécifiques liées à la gestion du système. Ses méthodes incluent :

- `gererUtilisateurs()`
- `gererEntrepotDonnees()`
- `executerPipelineETL()`
- `visualiserDashboardsBI()`

• Gestionnaire (Manager)

Également dérivé de `Utilisateur`, le Gestionnaire se concentre sur la supervision opérationnelle et analytique. Il possède des méthodes telles que :

- `codifierProduit()` (pour la gestion des codes et étiquettes)
- `consulterReporting()` / `genererRapport()`
- `gererInventaire()`

- **Opérateur de Terrain**

Spécialisé pour la collecte de données, l'Opérateur hérite également de `Utilisateur`, et se focalise sur des actions en temps réel sur le terrain :

- `collecterDonneesProduit()`
- `enregistrerMouvementStock()`

3. Classes Fonctionnelles et de Service

Au-delà des utilisateurs, le système décrit plusieurs modules opérationnels qui implémentent les fonctionnalités spécifiques issues des cas d'utilisation :

- **Authentification**

Plutôt qu'être une méthode unique de la classe `Utilisateur`, le mécanisme d'authentification peut être représenté sous forme de service ou de classe dédiée, possédant une méthode telle que :

- `verifierCredentials()`

- **EntrepotDonnees**

Cette classe gère le stockage centralisé et la structuration des données. Elle comporte :

- Des attributs indiquant la connexion à la base (ex. : `connexion`, `donneesStockees`)
- Des méthodes comme `stockerDonnees()`, `recupererDonnees()` et `backup()`

- **PipelineETL**

Responsable de l'extraction, transformation et chargement des données, cette classe comporte des méthodes telles que :

- `executer()`
- `transformerDonnees()`
- `chargerDonnees()`

- **DashboardBI**

Représente le module de Business Intelligence qui permet la visualisation interactive des indicateurs de performance.

- Attributs: `indicateurs`, `configurationGraphiques`
- Méthodes: `genererGraphiques()` et `actualiserIndicateurs()`

- **Rapport / ConsultationReporting**

Pour répondre aux besoins de consultation et génération de rapports, cette classe offre des fonctionnalités analytiques :

- Méthodes: `genererRapport()` et `filtrerDonnees()`

- **Inventaire**

Cette entité est responsable du suivi en temps réel des stocks.

- Attributs: `listeProduits`, `quantiteStock`
- Méthode: `miseAJourInventaire()`

4. Relations entre les Classes

- **Héritage :**

La classe abstraite `Utilisateur` est étendue par `Administrateur`, `Gestionnaire` et `Opérateur de Terrain`, assurant la réutilisation des attributs et la spécialisation des comportements.

- **Associations :**

- L' `Administrateur` est en relation avec `EntrepotDonnees`, `PipelineETL` et `DashboardBI`, indiquant qu'il supervisera et déclenchera ces modules.
- Le `Gestionnaire` interagit avec `DashboardBI`, `Rapport` et `Inventaire` pour valider et analyser les données.
- L' `Opérateur` se connecte aux modules de collecte, et ses actions sont enregistrées en même temps que des instances de `MouvementStock` qui pourraient être traitées dans le cadre d'un suivi d'inventaire.

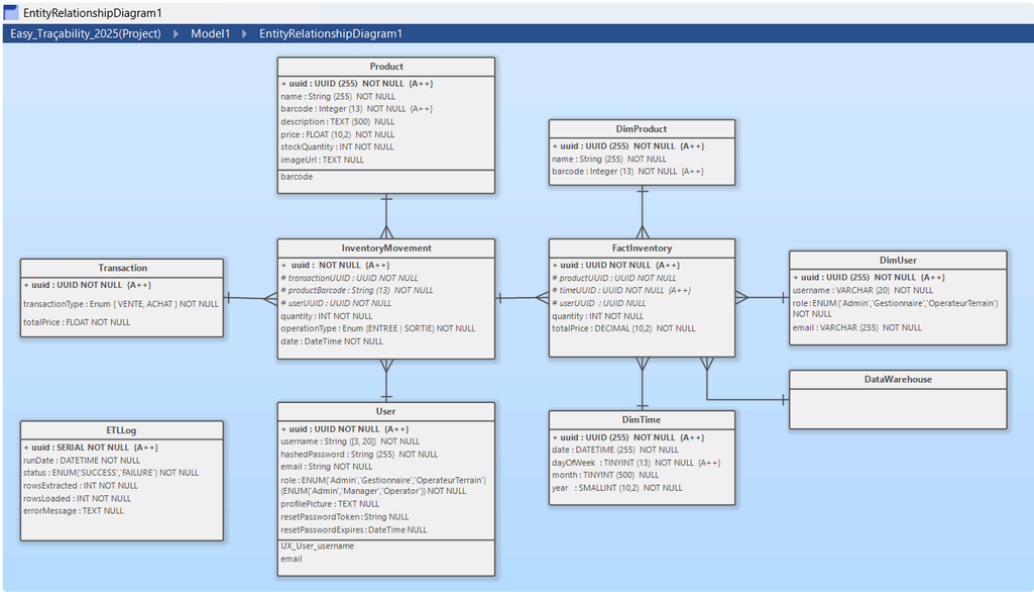
- **Dépendances :**

Certaines méthodes, comme l'exécution du pipeline ETL, dépendent d'un processus d'authentification préalable, illustrant une dépendance logique qui renforce la sécurité et l'intégrité des opérations.

Synthèse

Ce diagramme de classes reflète une conception modulaire et orientée objet, où les responsabilités sont réparties selon la nature des utilisateurs et les fonctionnalités du système. La séparation entre classes de type utilisateur et classes de service permet d'assurer une meilleure maintenabilité et une évolutivité du code, tout en renforçant la sécurité grâce à une étape d'authentification préalable. Les associations et dépendances entre les classes garantissent que chaque module collabore efficacement pour offrir une traçabilité fiable et réactive des opérations.

4. Diagramme Entité-Relation (ER)



Ce diagramme illustre dix tables, chacune définie avec ses attributs, types de données, et contraintes (certaines contraintes indiquées par « A++ » suggérant des propriétés particulières comme l'auto-incrémentation ou une contrainte d'unicité). Voici le détail de chaque table :

4.1. Table Product

Explication :

La table **Product** contient les informations principales sur les produits proposés dans le système. Elle stocke des attributs essentiels comme l'identifiant unique, le nom du produit, le code-barres, la description, le prix, la quantité en stock et une éventuelle URL d'image. Cette table sert de référence pour la gestion des articles en stock.

Attribut	Type	Contraintes / Description
uuid	UUID(255)	NOT NULL, A++ (identifiant unique)
name	VARCHAR(255)	NOT NULL (nom du produit)
barcode	INTEGER (13)	NOT NULL, A++ (code-barres)
description	TEXT (500)	Optionnel (description du produit)
price	FLOAT(10,2)	NOT NULL (prix du produit)
stockQuantity	INT	NOT NULL (quantité en stock)
imageUrl	TEXT	Optionnel (URL de l'image du produit)

4.2. Table DimProduct [🔗](#)

Explication :

La table **DimProduct** est une table de dimension qui répertorie les produits avec leurs attributs essentiels. Elle est utilisée dans des contextes analytiques pour faciliter les jointures avec d'autres tables de faits ou des opérations de reporting.

Attribut	Type	Contraintes / Description
uuid	UUID(255)	NOT NULL, A++ (identifiant unique)
name	VARCHAR(255)	NOT NULL (nom du produit)
barcode	INTEGER (13)	NOT NULL, A++ (code-barres)

4.3. Table Transaction [🔗](#)

Explication :

La table **Transaction** enregistre les opérations commerciales réalisées dans le système. Chaque enregistrement correspond à une transaction, identifiée de manière unique, et précise le type (vente ou achat) ainsi que le montant total lié à la transaction.

Attribut	Type	Contraintes / Description
uuid	UUID	NOT NULL, A++ (identifiant unique)
transactionType	ENUM	NOT NULL, valeurs possibles : {VENTE, ACHAT}
totalPrice	FLOAT	NOT NULL (montant total de la transaction)

4.4. Table InventoryMovement [🔗](#)

Explication :

La table **InventoryMovement** est dédiée au suivi des mouvements d'inventaire. Elle enregistre chaque opération d'entrée ou de sortie de stock, en associant une transaction, un produit et (éventuellement) un utilisateur à l'opération effectuée, avec une date précise.

Attribut	Type	Contraintes / Description
uuid	UUID	NOT NULL, A++ (identifiant unique)
transactionUUID	UUID	NOT NULL (référence à une transaction)
productBarcode	CHAR(13)	NOT NULL (identifiant du produit)
userUUID	UUID	Optionnel (référence à l'utilisateur ayant initié l'opération)
operationType	ENUM	NOT NULL, valeurs possibles : {ENTREE, SORTIE}
date	DATETIME	NOT NULL (date de l'opération)

4.5. Table FactInventory [🔗](#)

Explication :

La table **FactInventory** regroupe les données factuelles relatives aux opérations d’inventaire. Elle sert à centraliser les mesures, telles que la quantité manipulée et le prix total associé à une opération donnée, facilitant ainsi l'analyse multidimensionnelle en lien avec d'autres dimensions comme le produit ou le temps.

Attribut	Type	Contraintes / Description
uuid	UUID	NOT NULL, A++ (identifiant unique)
productUUID	UUID	NOT NULL (référence au produit)
timeUUID	UUID	NOT NULL, A++ (référence à la dimension temporelle)
userUUID	UUID	Optionnel (référence à l'utilisateur)
quantity	INT	NOT NULL (quantité impliquée dans l'opération)
totalPrice	DECIMAL(10,2)	NOT NULL (prix total associé à l'opération)

4.6. Table DimUser [🔗](#)

Explication :

La table **DimUser** est une table de dimension dédiée aux utilisateurs. Elle contient des informations analytiques fondamentales comme le nom d'utilisateur, le rôle et l'e-mail, ce qui permet de lier les données utilisateurs aux analyses et rapports générés dans un contexte décisionnel.

Attribut	Type	Contraintes / Description
uuid	UUID(255)	NOT NULL, A++ (identifiant unique)
username	VARCHAR(20)	NOT NULL (nom d'utilisateur)
role	ENUM	NOT NULL, valeurs possibles : {Admin, Gestionnaire, OperateurTerrain}
email	VARCHAR(255)	NOT NULL (adresse e-mail)

4.7. Table User [🔗](#)

Explication :

La table **User** conserve les informations d'authentification et de profil des utilisateurs. Contrairement à **DimUser**, qui présente des données à des fins analytiques, cette table centralise les éléments relatifs à la connexion, au cryptage des mots de passe, et aux processus de réinitialisation de mot de passe.

Attribut	Type	Contraintes / Description
uuid	UUID	NOT NULL, A++ (identifiant unique)
username	VARCHAR(20)	NOT NULL (entre 3 et 20 caractères, nom de l'utilisateur)

hashedPassword	VARCHAR(255)	NOT NULL (mot de passe crypté)
email	VARCHAR(255)	NOT NULL (adresse e-mail)
role	ENUM	NOT NULL, valeurs possibles : {Admin, Gestionnaire, OperateurTerrain}
profilePicture	TEXT	Optionnel (URL ou référence vers l'image de profil)
resetPasswordToken	VARCHAR	Optionnel (token utilisé pour la réinitialisation du mot de passe)
resetPasswordExpires	DATETIME	Optionnel (date d'expiration du token de réinitialisation)

4.8. Table DimTime [↗](#)

Explication :

La table **DimTime** est une table de dimension pour la gestion du temps. Elle stocke des informations détaillées sur chaque date, comme le jour de la semaine, le mois et l'année. Ces données facilitent l'analyse temporelle dans les opérations de reporting et de suivi.

Attribut	Type	Contraintes / Description
uuid	UUID(255)	NOT NULL, A++ (identifiant unique)
date	DATETIME	NOT NULL (date complète)
dayOfWeek	TINYINT(1)	NOT NULL, A++ (jour de la semaine)
month	TINYINT	NOT NULL (mois – annotation « 500 » observée dans le diagramme)
year	SMALLINT	NOT NULL (année)

4.9. Table ETLog [↗](#)

Explication :

La table **ETLog** enregistre les informations relatives aux processus d'extraction, transformation et chargement (ETL). Elle permet de suivre l'état d'exécution des opérations ETL, en indiquant si elles ont réussi ou échoué, et en documentant le nombre de lignes traitées ainsi que d'éventuels messages d'erreur.

Attribut	Type	Contraintes / Description
uuid	SERIAL	NOT NULL, A++ (identifiant unique, auto-incrémenté)
runDate	DATETIME	NOT NULL (date d'exécution du processus ETL)
status	ENUM	NOT NULL, valeurs possibles : {SUCCESS, FAILURE}
rowsExtracted	INT	NOT NULL (nombre de lignes extraites)

rowsLoaded	INT	NOT NULL (nombre de lignes chargées)
errorMessage	TEXT	Optionnel (message d'erreur en cas d'échec)

4.10. Table DataWarehouse [↗](#)

Explication :

La table **DataWarehouse** représente un espace de stockage conçu pour regrouper des données agrégées. Elle ne comporte pas d'attributs spécifiques dans le diagramme, ce qui suggère qu'elle joue le rôle de conteneur pour des informations consolidées en vue d'analyses décisionnelles globales.

Attribut	Type	Contraintes / Description
<i>Aucun</i>	<i>N/A</i>	Cette table sert uniquement comme regroupement d'informations dans l'entrepôt de données.

Ces tableaux, accompagnés de leurs explications, fournissent une vue d'ensemble claire de la structure et du rôle de chaque table dans le système.

5. Conclusion [↗](#)