

MAIGA Abdoulaye

TIROGO Hyacinthe

1. Présentation générale

Inventory est une application web de **gestion de stock** destinée à une petite ou moyenne organisation.

Elle permet de gérer :

- les **produits**
- les **catégories**
- les **fournisseurs**
- les **mouvements de stock** (entrées / sorties)
- les **utilisateurs** avec rôles
- un **tableau de bord décisionnel**

L'application est sécurisée par authentification et repose sur une architecture **client-serveur**.

2. Fonctionnalités principales

2.1 Authentification et sécurité

- Page de **connexion sécurisée**
- Génération de **JWT** après authentification
- Gestion des rôles :
 - ADMIN
 - EMPLOYEE
- Redirection automatique vers le **dashboard** après connexion

2.2 Gestion des produits

- Création, modification, suppression des produits
- Attributs :
 - SKU (unique)
 - Nom
 - Prix
 - Quantité en stock
 - Catégorie
 - Fournisseur
 - Description
- Liaison directe avec les catégories et fournisseurs

2.3 Gestion des catégories

- CRUD complet des catégories
- Association des produits à une catégorie
- Recherche des produits par catégorie

2.4 Gestion des fournisseurs

- CRUD des fournisseurs
- Association fournisseur ↔ produits

2.5 Mouvements de stock

- Enregistrement des **entrées** et **sorties**
- Mise à jour automatique du stock produit
- Historique des mouvements
- Champ note pour justification

2.6 Tableau de bord (Dashboard)

Le tableau de bord affiche en temps réel :

- **Stock total** (quantité globale)
- **Valeur totale du stock**
→ calculée par :
 $\text{SUM}(\text{stockQuantity} \times \text{price})$
- **Top 5 produits** (par quantité)
- **Produits en rupture** (stock = 0)
- Boutons de navigation vers toutes les pages

3. Architecture technique

3.1 Backend

- **Java 17**
- **Spring Boot 3**
- **Spring Data JPA**
- **Spring Security**
- **JWT (Json Web Token)**
- **Hibernate**
- **PostgreSQL**

Organisation :

- **entity** : entités JPA
- **repository** : accès base de données
- **service** : logique métier
- **controller** : API REST
- **security** : configuration JWT & rôles

3.2 Frontend

- **HTML5**
- **CSS3**
- **JavaScript Vanilla**
- Communication via **API REST**
- Stockage du token JWT côté navigateur
- Interface simple, claire et intuitive

3.3 Base de données

- **PostgreSQL**
- Relations :
 - Produit ↔ Catégorie
 - Produit ↔ Fournisseur
 - Produit ↔ Mouvements de stock
- Contraintes d'intégrité (clé unique, clés étrangères)

3.4 Conteneurisation

- **Docker**
- **Docker Compose**
- Conteneurs :
 - inventory-app (Spring Boot)
 - inventory-db (PostgreSQL)
- Volumes persistants pour les données

4. Installation et exécution

4.1 Prérequis

- Docker & Docker Compose
- Java 17 (si exécution hors Docker)
- Git

4.2 Installation avec Docker (recommandée)

```
git clone https://github.com/votre-compte/inventory.git
```

```
cd inventory
```

```
docker-compose up --build
```

Accès :

- Application : <http://localhost:9090>
- API backend : <http://localhost:8080>

4.3 Compte administrateur par défaut

Créé automatiquement au démarrage (idempotent) :

Email : admin@azakary.com

Password : admin123

Rôle : ADMIN

