**UNIVERSITE DE FIANARANTSOA**

**ECOLE NATIONALE D’INFORMATIQUE**

**PROJET JAKARTA**

**Mention :** Informatique

**Parcours :** Administration des Systèmes et Réseaux

**Niveau :** Troisième année de licence professionnelle

CONCEPTION ET LA REALISATION D’UNE APPLICATION POUR LA GESTION DE STOCK DES MATERIELS

**Présentés par:**

* RAKOTOMALALA Faneva (2513)
* TOJONIRINA Vicah (2481)
* RAZAFINDRAVELO Tantelison Odilon (2484)
* RAFANOMEZANTSOA Nantenaina Antoine (2708)
* RAKOTONAMBININA Elysé (2776)

**Encadreur:** Gilante Goazafy, Assistant de l’Enseignement Supérieur et de recherche

**Année Universitaire**: 2023-2024

# **SOMMAIRE**

[**SOMMAIRE** I](#_Toc174682652)

[**REMERCIEMENTS** II](#_Toc174682653)

[**LISTE DES FIGURES** III](#_Toc174682654)

[**LISTE DES TABLEAUX** IV](#_Toc174682655)

[**LISTE DES ABBREVIATION** V](#_Toc174682656)

[**INTRODUCTION GENERALE** 1](#_Toc174682657)

[**Partie 1: DESCRIPTION DU PROJET** 2](#_Toc174682658)

[**1.** **Formulation** 3](#_Toc174682659)

[**2.** **Objectif et besoins de l’utilisateur** 3](#_Toc174682660)

[**3.** **Moyens nécessaire à la réalisation du projet** 3](#_Toc174682661)

[**4.** **Résultat attendus** 4](#_Toc174682662)

[**Partie 2: ANALYSE ET CONCEPTION** 5](#_Toc174682663)

[**CONCEPTION DETAILLEE** 6](#_Toc174682664)

[**Partie 3: REALISATION** 7](#_Toc174682665)

[**Chapitre 2 : Développement de l’application** 10](#_Toc174682666)

[Chapitre 3 : Présentation de l’application 20](#_Toc174682667)

[**CONCLUSION** 23](#_Toc174682668)

[**BIBLIOGRAPHIE** VI](#_Toc174682669)

[**WEBOGRAPHIE** VII](#_Toc174682670)

[**RESUME** VIII](#_Toc174682671)

[**ABSTRACT** VIII](#_Toc174682672)

# **REMERCIEMENTS**

Tout d’abord, nous tenons à remercier Dieu pour sa bénédiction tout au long de ce projet. Ensuite, nous apportons nos sincères remerciements à ces personnes :

* Monsieur HAJALALAINA Aimé Richard, Docteur HDR, Président de l’Université de Fianarantsoa, pour nous avoir accepté dans l’établissement ;
* Monsieur MAHATODY Thomas, Docteur HDR, Directeur de l’Ecole Nationale d’Informatique, qui nous a permis de réaliser ce projet ;
* Monsieur Gilante Goazafy, Encadreur pédagogique, pour son enseignement sur la technologie java permettant de réaliser ce travail ;
* Tous les personnels de l’Ecole Nationale d’Informatique, pour leur appui sur tous les plans possibles ;
* Nos familles, pour leur soutien financier et moral.

# **LISTE DES FIGURES**

[Figure 1:Diagramme de classe 6](#_Toc174682519)

[Figure 2:Configuration du fichier pom.xml 8](#_Toc174682520)

[Figure 3: Configuration du fichier persistance.xml 9](#_Toc174682521)

[Figure 4: La classe DAO de employer 10](#_Toc174682522)

[Figure 5: La classe DAO pour le produit 10](#_Toc174682523)

[Figure 6: La classe DAO pour la partie transaction 11](#_Toc174682524)

[Figure 7: La classe regroupant des méthodes propre de l'employer 12](#_Toc174682525)

[Figure 8: La classe regroupant des méthodes propres de la transaction 13](#_Toc174682526)

[Figure 9: La classe regroupant des méthodes propre des produits 14](#_Toc174682527)

[Figure 10: Implementation de la classe AddServlet 15](#_Toc174682528)

[Figure 11: Implementation de la classe AjoutServlet 16](#_Toc174682529)

[Figure 12: Implémentation de la classe ModifServlet 18](#_Toc174682530)

[Figure 13: Implémentation de la classe SupprimerServlet 19](#_Toc174682531)

[Figure 14: Page de connexion 20](#_Toc174682532)

[Figure 15: Page de connexion 20](#_Toc174682533)

[Figure 16: Ajouts de produit 21](#_Toc174682534)

[Figure 17: Modification du produit 21](#_Toc174682535)

[Figure 18: Suppressions du produit 22](#_Toc174682536)

[Figure 19: Page d' historique 22](#_Toc174682537)

# **LISTE DES TABLEAUX**

[Tableau 1:Caractéristiques du matériel utilize 3](#_Toc174682818)

# **LISTE DES ABBREVIATION**

**DAO:** Data Access Objects

**MVC**: Model View Controller

**ORM:** Object Relational Mapping

**SGBD**: Système de Gestion de Base de donnée

**SQL**: Structured Query Language

**UML**: Unified Modeling Language

# **INTRODUCTION GENERALE**

Actuellement, la gestion des stocks est une composante cruciale pour toutes types entreprise ou société souhaitant optimiser ses opérations et minimiser ses manœuvres d’erreur. Avec l’acheminement des applications web qui fournit, il est devenu simple de développer des solutions simple, pratique en même temps accessible partout avec toutes type de matériels.

Notre travail consiste alors à mettre en place une application pour la gestion des stocks des matériels. Ainsi on a choisi le thème : « CONCEPTION ET LA REALISATION D’UNE APPLICATION POUR LA GESTION DE STOCK DES MATERIELS ».

Pour mieux entrer dans le vif du sujet, nous allons diviser notre travail en trois grandes parties, tout d’abord la première partie où on va évoquer la description du projet ensuite la seconde partie où nous allons effectuer une analyse et conception et enfin pour terminer, nous verrons dans la troisième et dernière partie, la mise en place l’environnement de développement de l’application et enfin nous allons la développer.

# **Partie 1: DESCRIPTION DU PROJET**

## **Formulation**

L’amélioration du mode de travail est actuellement devenue très favorable grâce à l’innovation apportée par l’Informatique. Mais actuellement, la plupart des tâches se fait encore manuellement via une note. Dans cette image que nous avons eu l’idée de propose une application de gestion de stock.

## **Objectif et besoins de l’utilisateur**

Ce projet consiste à créer une application destinée à gérer des stocks de produit par les employés via une application web. Les fonctionnalités qui vont être développées dans cette application sont simplement pour les employés comme

* Consultation des produits
* Ajouts de produits
* Suppression de produits
* Modification de produits

## **Moyens nécessaire à la réalisation du projet**

### Moyens matériels

Le tableau 5 illustre les caractéristiques des matériels utilisés.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Nom de l’ordinateur | Marque | Processeur | Disque dur | RAM | | OS | |
| Ragnar | HP notebook G6 | Intel Corei5 | 750 Go | 8 Go | Debian 12 | |  |

Tableau 1:Caractéristiques du matériel utilize

### 3.2. Moyens Humains

Les personnes contribuant à ce projet sont tous étudiants de l’Ecole Nationale Informatique.

### Moyens logiciels

Nous avons besoin des logiciels suivants pour l’élaboration du projet :

* IntellijIdea : Pour le développement de l’application
* Mysql serveur: Pour l’administration de la base de donné
* Firefox: Pour le test

## **Résultat attendus**

Le résultat escompté à la fin du projet est de produire une application qui va rendre facile la gestion des stocks de produit.

L’application va être également :

* Parfaitement sécurisée, seuls ceux qui ont les droits pourront y accéder.
* Facile à manipuler pour que les utilisateurs soient à l’aise dans leur travail.
* Optimale pour le gain de temps.
* Et surtout, facilement maintenable.

# **Partie 2: ANALYSE ET CONCEPTION**

## **CONCEPTION DETAILLEE**

Dans cette projet, nous allons utiliser une structure particulière de répertoires, une bibliothèque qui utilise une architecture proche de MVC. MVC signifie « Modèle / Vue / Contrôleur ». C'est un découpage très répandu pour développer les sites internet, car il sépare les couches selon leur logique propre. Le Contrôleur (ou Controller) : son rôle est de générer la réponse à la requête demandée par notre visiteur. Il est la couche qui se charge d'analyser et de traiter la requête de l'utilisateur. Le contrôleur contient la logique de notre site internet et va se contenter « d'utiliser » les autres composants : les modèles et les vues Le Modèle (ou Model) : son rôle est de gérer les données et le contenu. La Vue (ou View) : son rôle est d'afficher les pages. Ce n'est pas le contrôleur qui affiche le formulaire, il ne fait qu'appeler la bonne vue. En pratique, c'est le designer d'un projet qui travaille sur les vues. Séparer vues et contrôleurs permet aux designers et développeurs JAVA de travailler ensemble sans se marcher dessus. Mais pour notre part nous allons l’implémenté d’un repository regroupant tous les classes et ses méthodes

Voici un diagramme de classe qui va illustrer notre projet

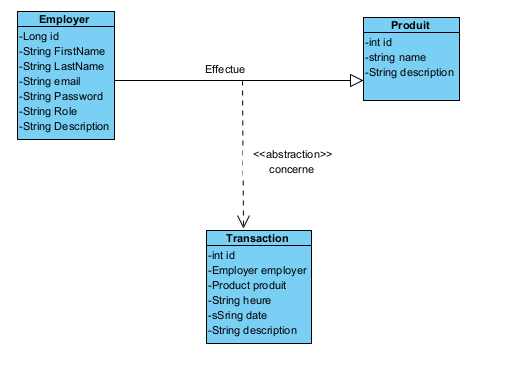


Figure 1:Diagramme de classe

# **Partie 3: REALISATION**

**Chapitre 1 : Mise en place de l’environnement**

### Création de la base de donnée

La création du base de donnée se fasse via le commande SQL suivante dans notre serveur MySQL:

CREATE DATABASE Gestion\_stock;

### Configuration du fichier pom.xml pour les dépendances

La configuration du fichier pom.xml se fasse via l’ajout de dépendance de hibernate et du mysql-connector



Figure 2:Configuration du fichier pom.xml

### Configuration du fichier persistance.xml

La configuration du fichier persistance.xml se fasse via l’ajout d’information de connexion

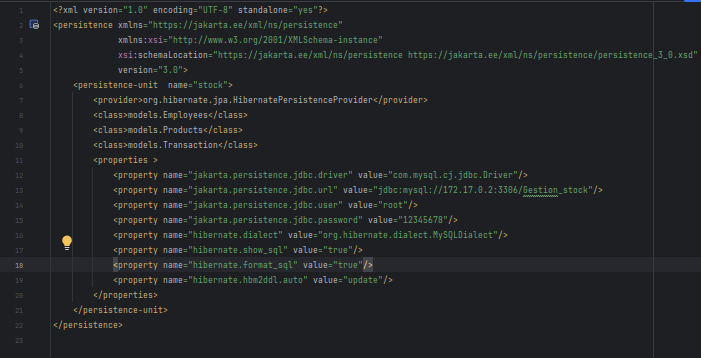


Figure 3: Configuration du fichier persistance.xml

## **Chapitre 2 : Développement de l’application**

### Les classe DAO

DAO ou Data Access Objects sont des motifs de conception utilisé pour encapsuler l’accès à la base de donnée. Pour notre part on a créé 3 fichier représentants chacun l’accès à une table de notre base de donnée de manière suivante.

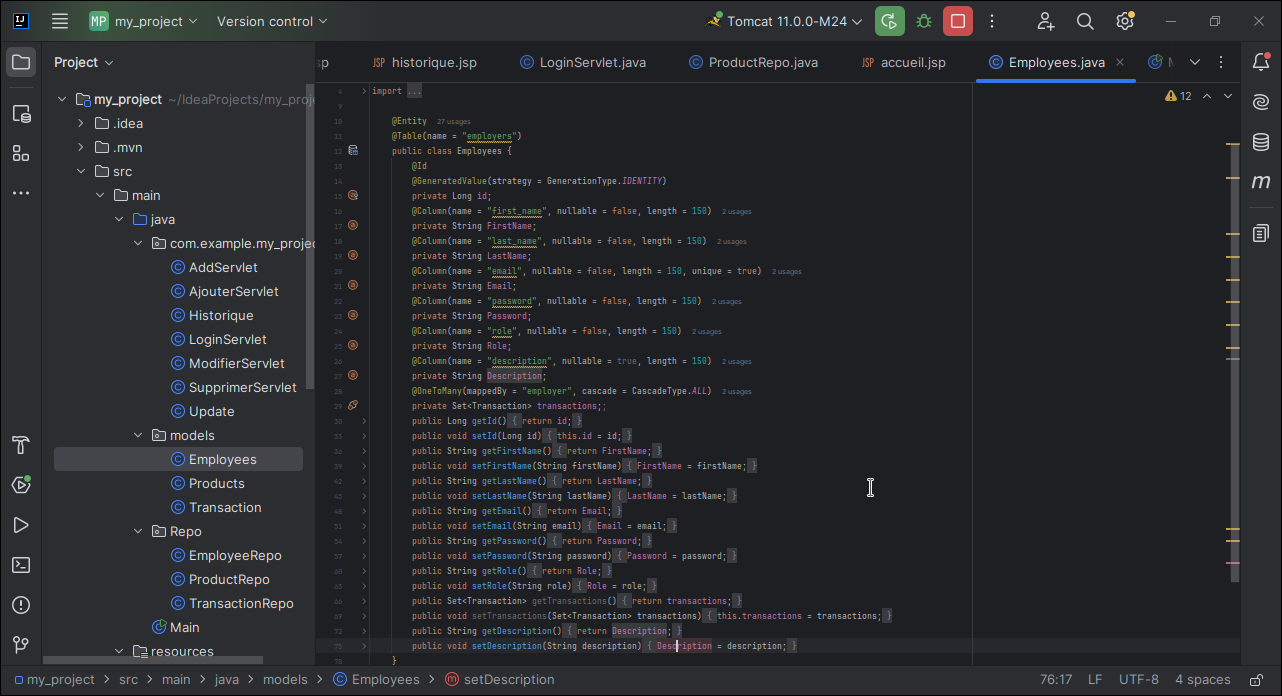


Figure 4: La classe DAO de employer

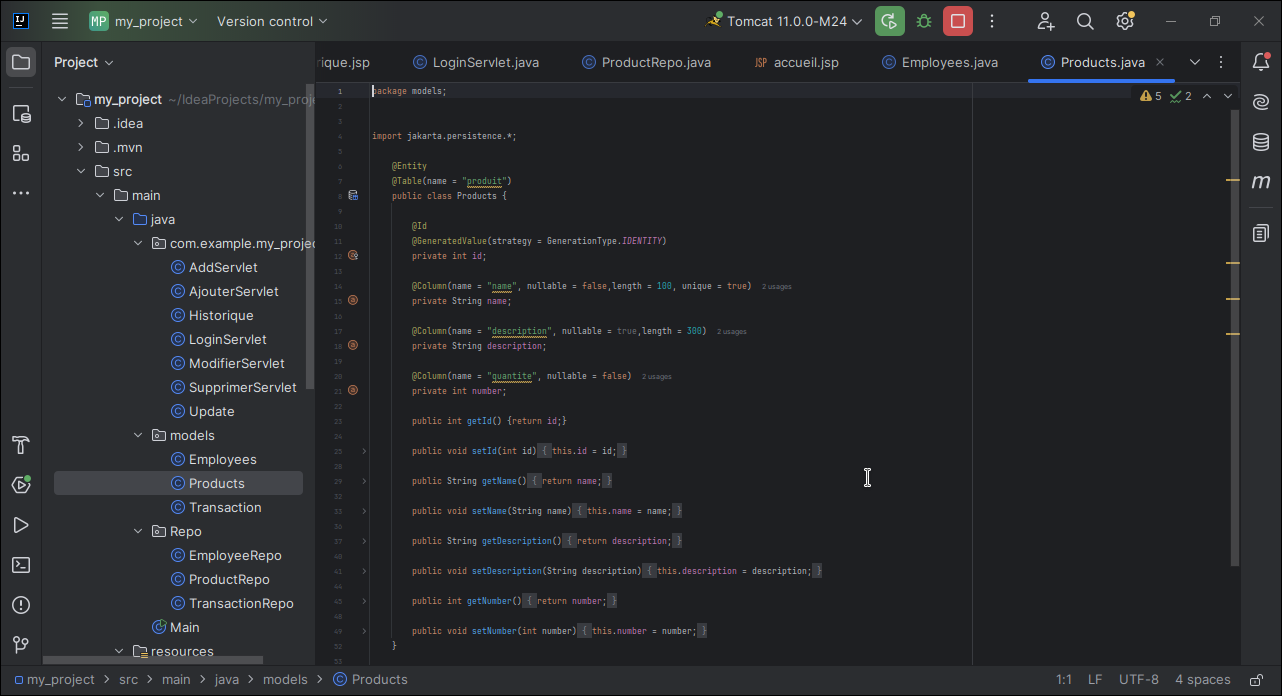


Figure 5: La classe DAO pour le produit

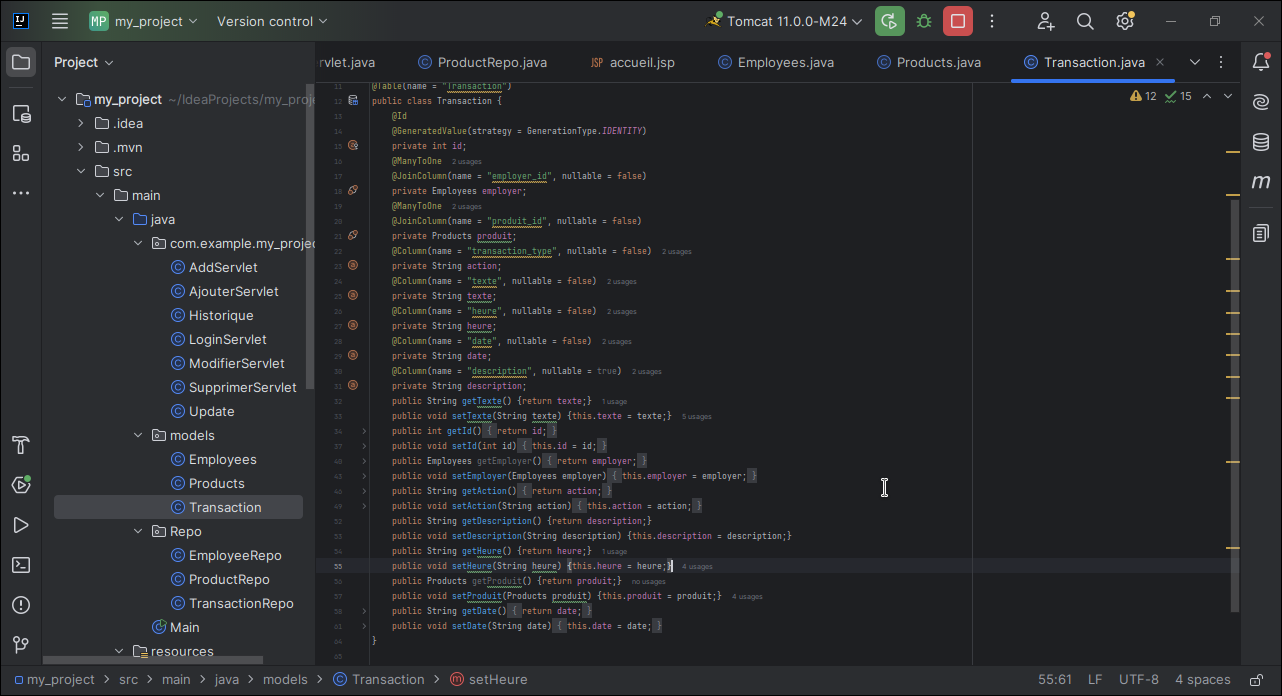
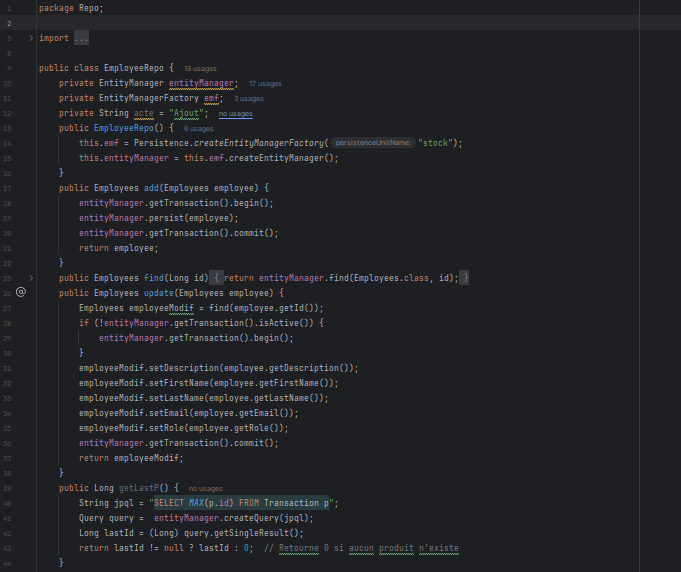


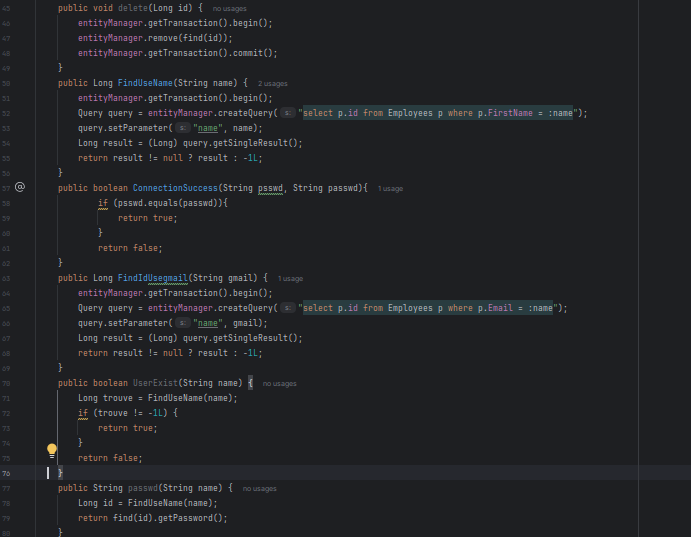
Figure 6: La classe DAO pour la partie transaction

### Les classes de repository ou de stockage

Ce sont des classes regroupant les fonctions sous forme de méthode qui permet de mieux élaborer le code.

Pour la partie employer :





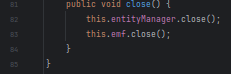
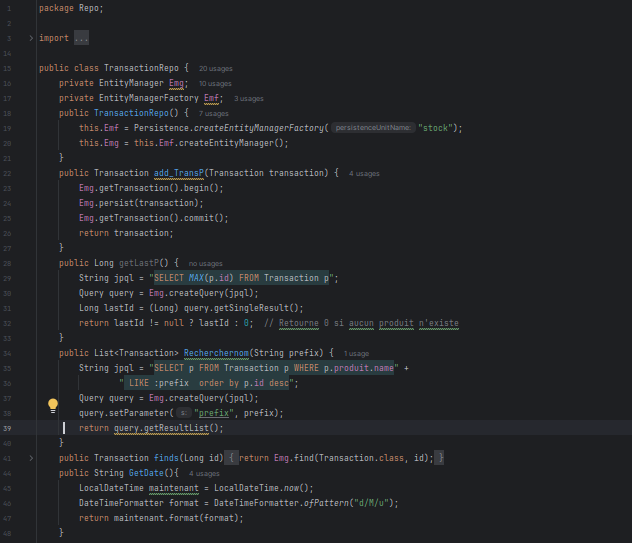


Figure 7: La classe regroupant des méthodes propre de l'employer

Pour la partie transaction :



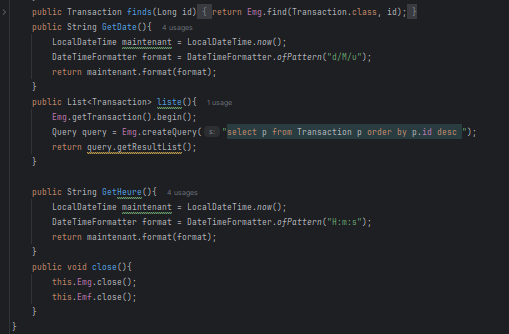
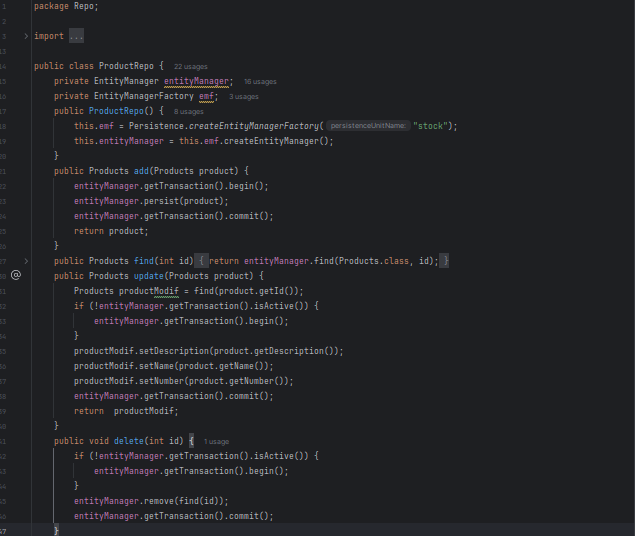


Figure 8: La classe regroupant des méthodes propres de la transaction

Pour la partie produit :



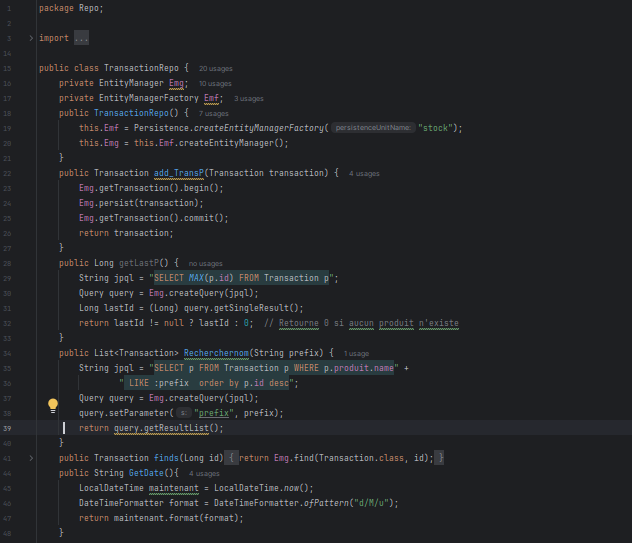
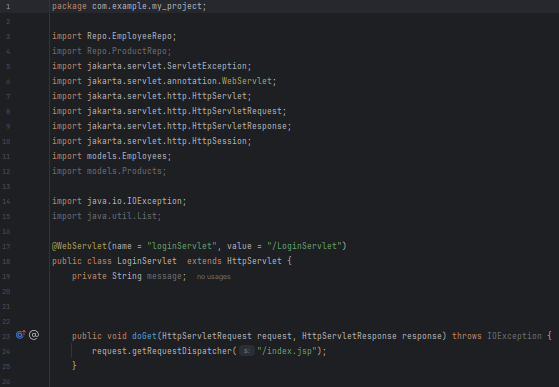


Figure 9: La classe regroupant des méthodes propre des produits

### Les classes servlets

Les classe servlets sont tous des classes héritières du classe HttpServlets permettant la gestion de la requête surtout la requête GET et POST :

Tout d’abord pour la requête permettant d’entrée dans l’application ou page de connexion

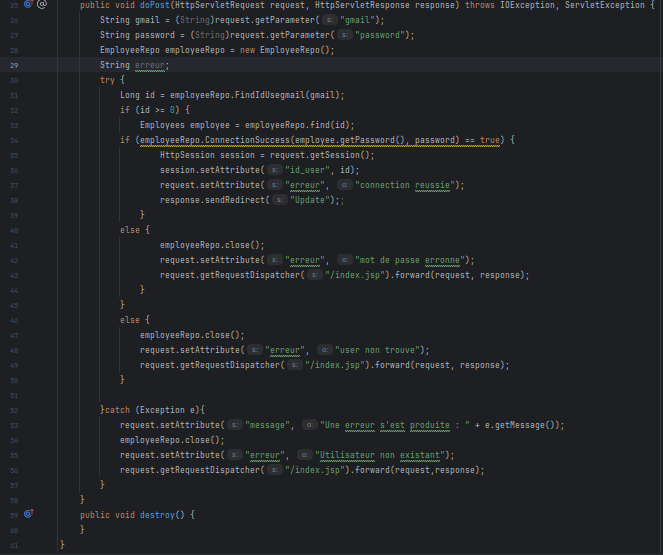
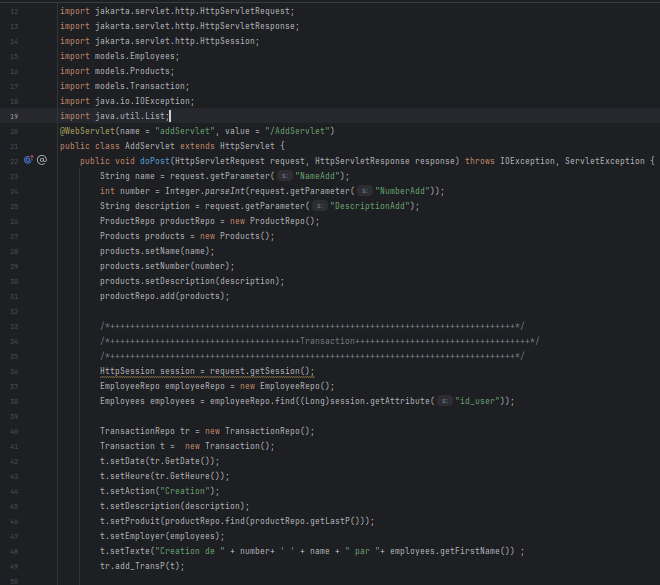


Figure 9: : Implémentation de la classe LoginServlet

Pour la partie création



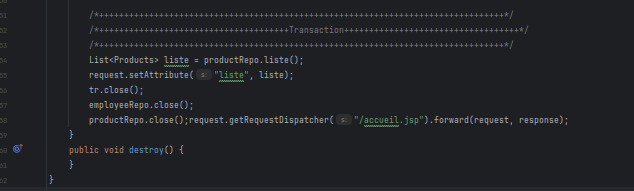
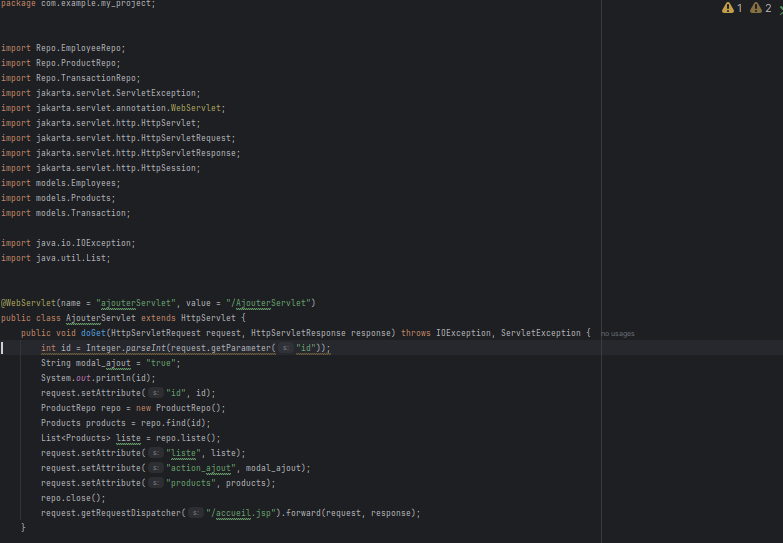


Figure 10: Implementation de la classe AddServlet

Servlet pour l’ajout:  


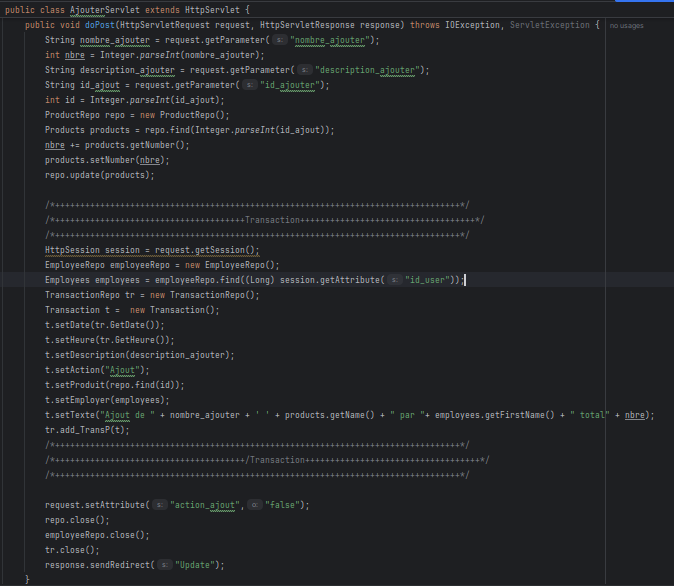
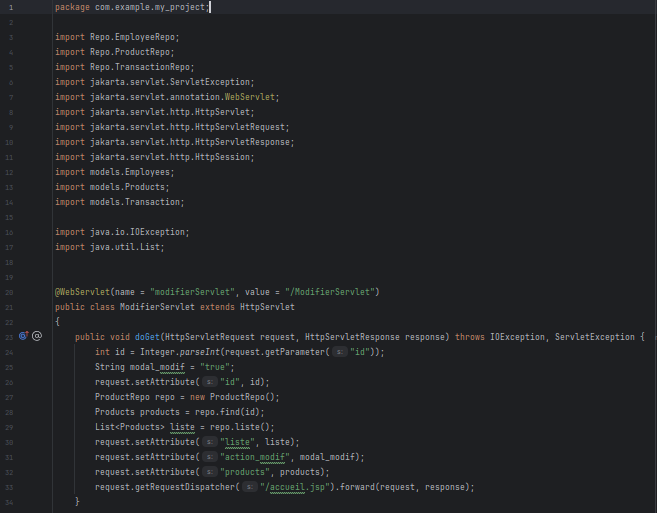
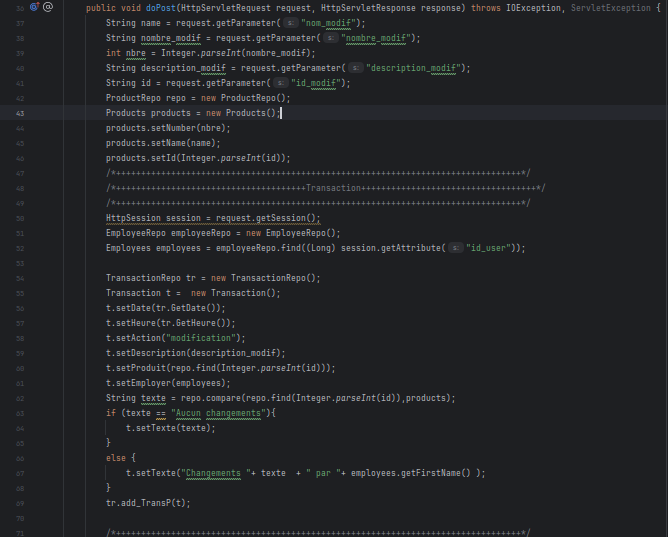


Figure 11: Implementation de la classe AjoutServlet

Servlet pour la partie modification





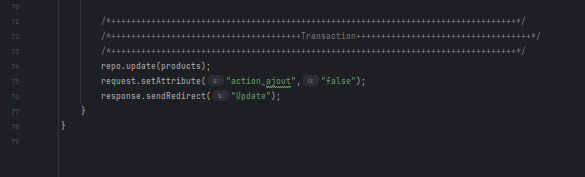
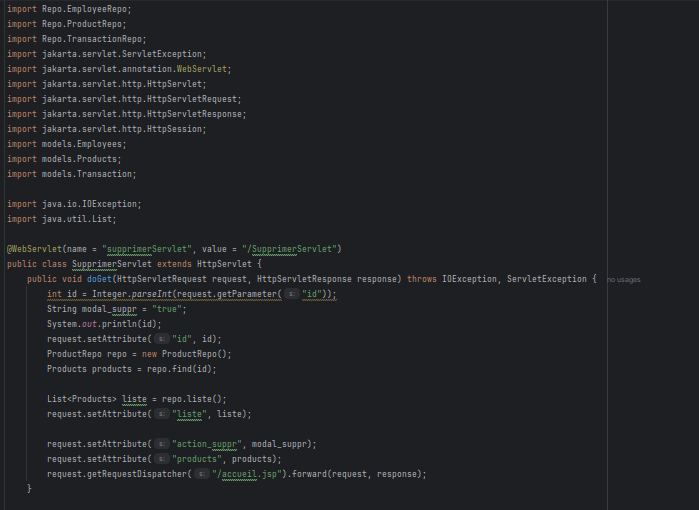
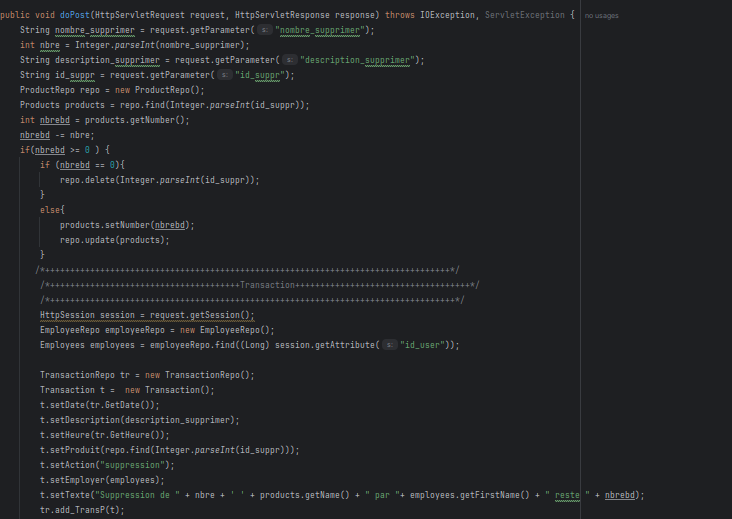


Figure 12: Implémentation de la classe ModifServlet

Servlet pour la partie modification





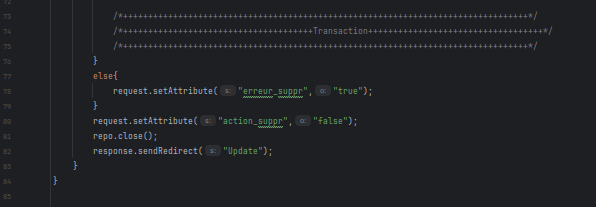


Figure 13: Implémentation de la classe SupprimerServlet

## **Chapitre 3 : Présentation de l’application**

### Le formulaire de connexion

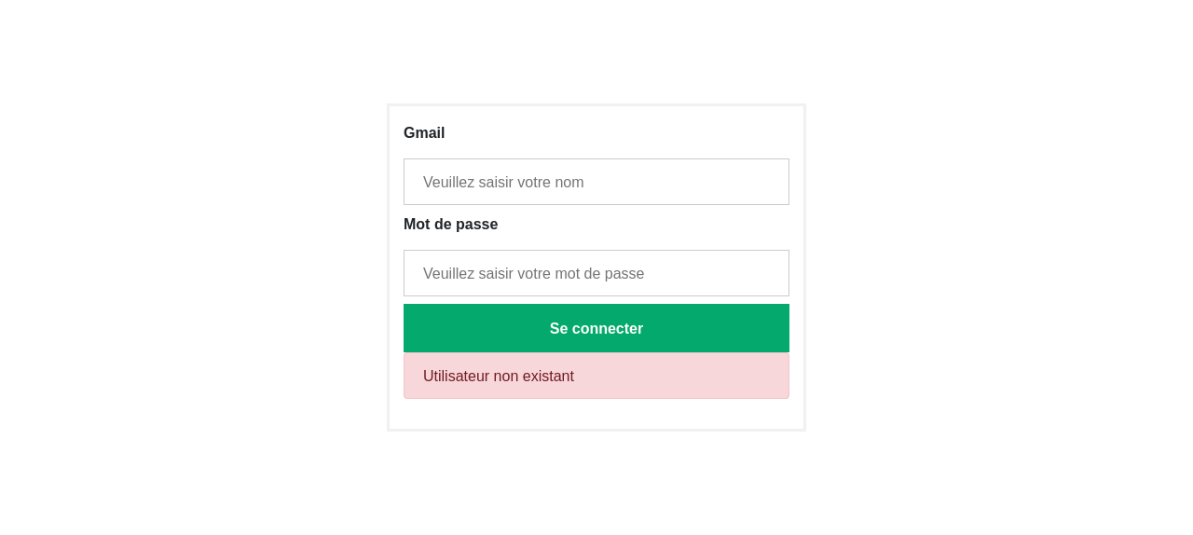


Figure 14: Page de connexion

### La page d’accueil

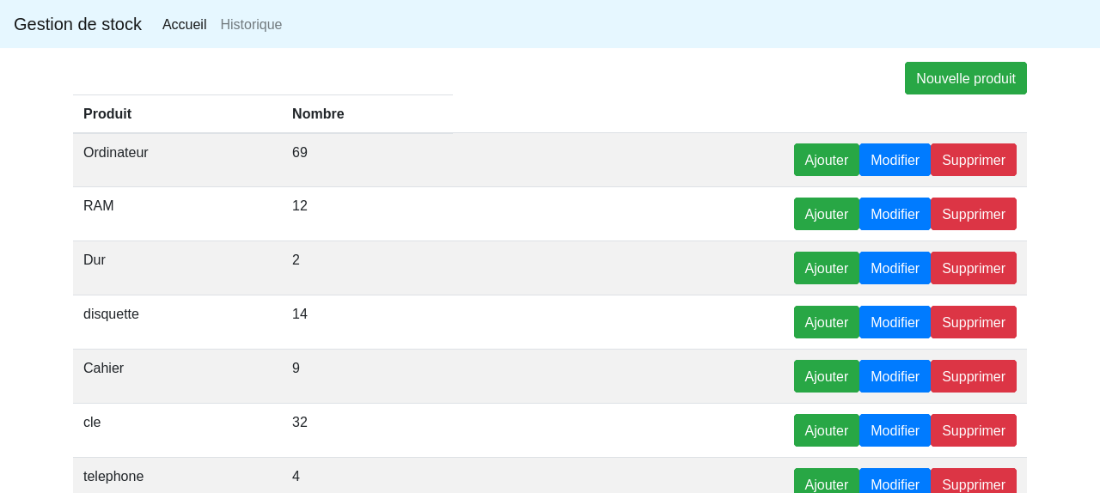


Figure 15: Page de connexion

### Création d’un nouveau produit :

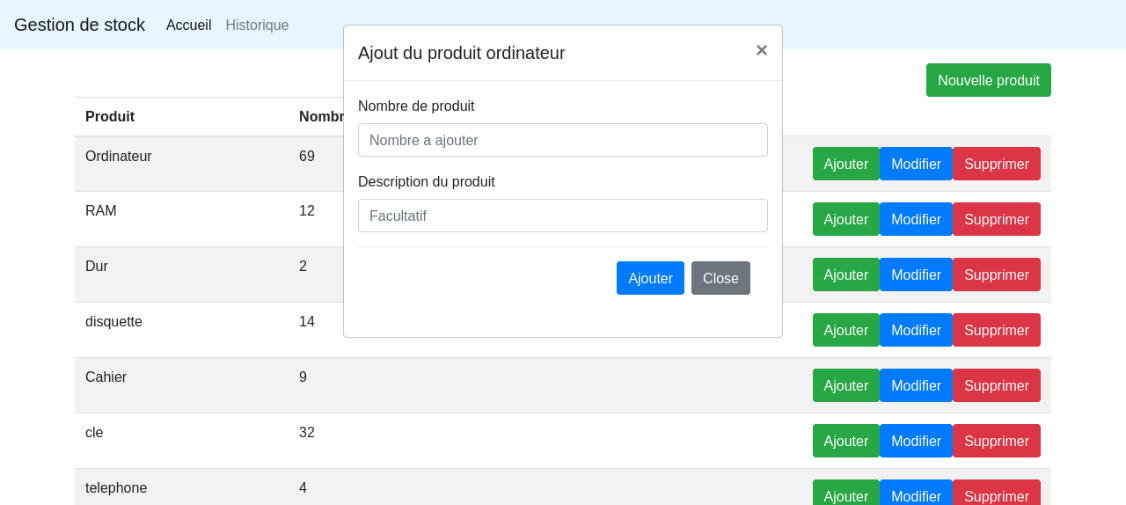


Figure 16: Ajouts de produit

### Modification du produit

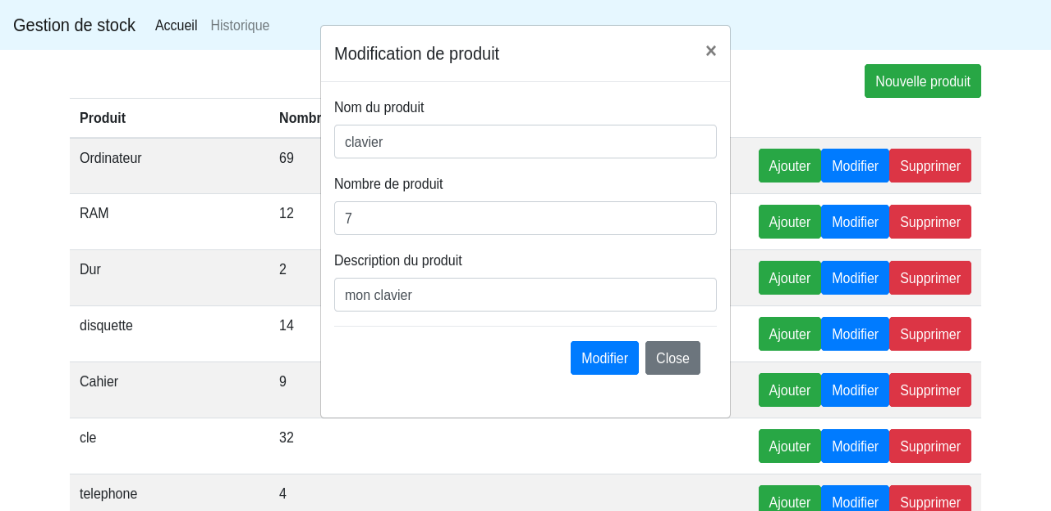


Figure 17: Modification du produit

### Suppression du produit

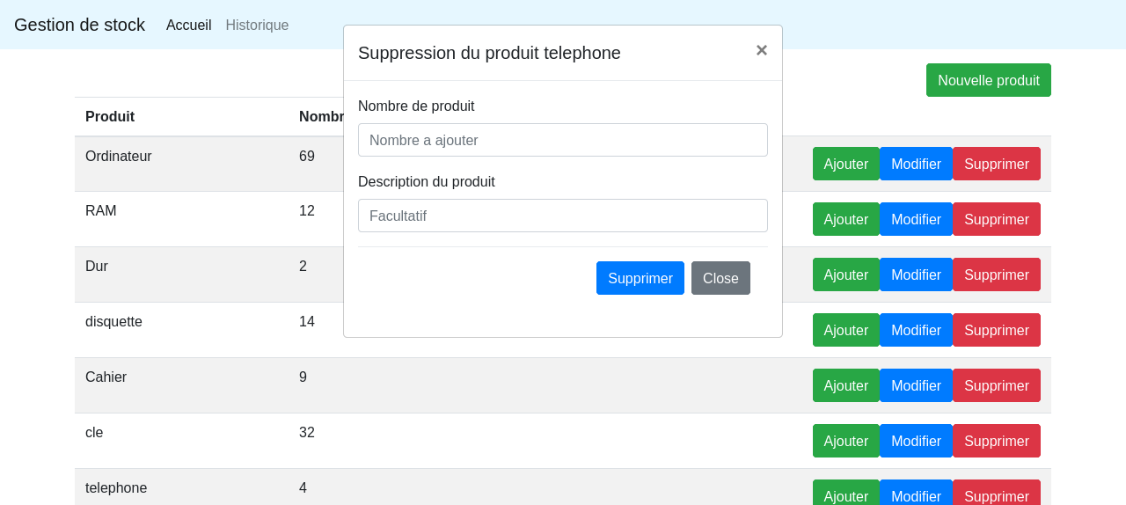


Figure 18: Suppressions du produit

### Historique



Figure 19: Page d' historique

# **CONCLUSION**

En entreprenant ce voyage à travers le développement d'une application de gestion de stock, nous avons exploré les multiples facettes des technologies Java et Jakarta. Ce projet nous a permis de comprendre comment ces outils puissants s'articulent pour donner naissance à une application robuste, scalable et maintenable.

Nous avons commencé par les bases, en construisant les fondations de notre application avec des concepts clés tels que les classes, les objets, et les relations entre entités. Progressivement, nous avons intégré des technologies plus avancées, telles que Jakarta EE pour le développement d'applications web et Hibernate pour la gestion de la persistance des données.

Tout au long de ce projet, vous avez acquis des compétences essentielles non seulement en Java, mais aussi en gestion de projet logiciel. Vous avez appris à concevoir des architectures applicatives, à implémenter des fonctionnalités critiques, et à optimiser la performance et la sécurité de votre application.

Mais au-delà des compétences techniques, ce projet a aussi été une occasion de développer un esprit d'ingénierie : une capacité à résoudre des problèmes complexes, à prendre des décisions architecturales judicieuses, et à créer des solutions élégantes et efficaces.

L'apprentissage ne s'arrête jamais dans le domaine du développement logiciel. Ce projet n'est qu'une étape dans votre parcours de développeur. En continuant à explorer de nouveaux outils, frameworks, et paradigmes, vous continuerez à renforcer votre expertise et à faire évoluer vos compétences.

En conclusion, la création de cette application de gestion de stock a été une démonstration pratique de la puissance de Java et de Jakarta. Que ce soit pour des projets personnels ou professionnels, vous avez maintenant une base solide sur laquelle bâtir des applications encore plus sophistiquées. L'avenir est entre vos mains, et avec ces connaissances, vous êtes bien équipé pour relever les défis du développement logiciel dans le monde dynamique et en constante évolution de la technologie.

# **BIBLIOGRAPHIE**

**Alur, Deepak, John Crupi, and Dan Malks.** Core J2EE Patterns: Best Practices and Design Strategies. Prentice Hall, 2003.

# **WEBOGRAPHIE**

<https://hibernate.org/documentation/>

[https://jakarta.](https://jakarta.ee/)ee

# **RESUME**

Dans ce projet, nous avons conçu et élaboré une application destinée à la gestion de stock. Nous avons mené une enquête dans ce domaine pour comprendre les attentes des utilisateurs et identifier les différents problèmes du système de gestion actuel, afin de proposer des solutions adéquates. Après avoir effectué cette enquête, nous avons procédé à une analyse conceptuelle et logique en utilisant la méthode du Processus Unifié et le langage UML, afin de concevoir une base de données et de gérer un système d’information. Pour manipuler les données dans la base de données ainsi créée, nous avons développé une application web en utilisant la technologie Jakarta, le serveur web Tomcat et l'ORM Hibernate, avec MySQL comme SGBD.

# **ABSTRACT**

In this project, we designed and developed an application for stock management. We conducted a survey in this field to understand user expectations and identify the various issues with the current management system in order to propose suitable solutions. After conducting this survey, we proceeded with a conceptual and logical analysis using the Unified Process method and the UML language, in order to design a database and manage an information system. To manipulate the data in the newly created database, we developed a web application using Jakarta technology, the Tomcat web server, and the Hibernate ORM, with MySQL as the DBMS.