**WomenShop - Stock Management System**

**Rapport de Projet**

**Cours**: Advanced Object-Oriented Programming  
**Projet**: JavaFX Desktop Application with MySQL Database  
**Team Members**: Uliana Chernysheva, Joss Develter

**Résumé**

WomenShop ("Boutique") est une application desktop complète développée avec JavaFX pour gérer l'inventaire d'une boutique de mode féminine. L'application implémente avec succès toutes les fonctionnalités requises incluant la gestion des produits sur trois catégories (Vêtements, Chaussures, Accessoires), le contrôle du stock via les transactions d'achat et de vente, un système de remise flexible, le suivi financier en temps réel, et une validation complète des données. Le système respecte les patterns de conception MVC et DAO avec une base de données MySQL bien structurée.

**1. Features Successfully Implemented**

**1.1 Product Display & Management**

*Status: Fully Implemented*

L'application affiche tous les produits dans un tableau complet avec les informations suivantes:

* Nom du produit
* Catégorie avec badges colorés
* Détails spécifiques au produit (taille, couleur, matériau/marque)
* Prix d'achat, prix de vente et prix final
* Quantité en stock avec indicateurs de couleur (rouge : rupture, orange : stock faible, vert : stock correct)
* Statut de remise avec pourcentage affiché

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer screen

AI-generated content may be incorrect.**

**1.2 Category-Based Navigation**

**Status**: **Fully Implemented**

Leftside barre avec boutons de catégories :

* **Toutes les Collections** : Affiche tous les produits
* **👗 Prêt-à-Porter** : Filtre uniquement les vêtements
* **👠 Chaussures** : Filtre uniquement les chaussures
* **👜 Accessoires** : Filtre uniquement les accessoires

A screenshot of a website

AI-generated content may be incorrect.

La catégorie active est mise en surbrillance. La catégorie sélectionnée est également affichée en haut du catalogue de produits.

A screenshot of a computer

AI-generated content may be incorrect.

**1.3 Advanced Sorting Functionality**

**Status**: **Fully Implemented**

Les produits peuvent être triés selon plusieurs critères via menu déroulant :

* **Ordre par Défaut** : Ordre original de la base de données
* **Prix : Croissant** : Par prix final croissant
* **Prix : Décroissant** : Par prix final décroissant
* **Stock : Croissant** : Produits avec le moins de stock en premier
* **Stock : Décroissant** : Produits avec le plus de stock en premier
* **Nom : A-Z** : Ordre alphabétique croissant
* **Nom : Z-A** : Ordre alphabétique décroissant

Le tri fonctionne parfaitement avec le filtrage par catégorie.A screenshot of a computer

AI-generated content may be incorrect.

Exemple: Price Low to HIghA screenshot of a computer screen

AI-generated content may be incorrect.

Exemple: Name A-Z

A screenshot of a computer

AI-generated content may be incorrect.

**1.4 Real-Time Search Functionality**

**Status**: **Fully Implemented**

Barre de recherche intégrée permettant de rechercher les produits par :

* Nom du produit
* Détails du produit (taille, couleur, type, marque, matériau)

La recherche fonctionne en temps réel pendant la saisie et se combine avec le filtrage par catégorie et le tri.

Exemple: search “heels”

A screenshot of a computer

AI-generated content may be incorrect.

Exemple: trouver un produit de couleur noirA screenshot of a computer

AI-generated content may be incorrect.

**1.5 Add New Products**

**Status**: **Fully Implemented**

Formulaire élégant pour ajouter de nouveaux produits avec :

* **Section Informations de Base** : Nom, catégorie, prix d'achat, prix de vente
* **Section Spécifique à la Catégorie** : Change dynamiquement selon la catégorie sélectionnée
  + **Vêtements** : Taille, Couleur, Matériau
  + **Chaussures** : Pointure, Couleur, Marque
  + **Accessoires** : Type, Couleur, Matériau
* Sections colorées selon le thème de la catégorie
* Stock initial à 0 par défaut
* Validation complète avant sauvegarde
* Gestion des exeptions et des erreurs

A screenshot of a web page

AI-generated content may be incorrect.A screenshot of a computer screen

AI-generated content may be incorrect.

Exemple: ajouter un t-shirt

A screenshot of a web page

AI-generated content may be incorrect.A screen shot of a computer

AI-generated content may be incorrect.



**1.6 Modify/Edit Products**

**Status**: **Fully Implemented**

Fonctionnalité de modification avec :

* Formulaire pré-rempli avec les données existantes du produit
* **Le champ catégorie est désactivé** pour maintenir l'intégrité des données
* Tous les autres champs peuvent être modifiés
* Même validation que l'ajout de produits
* Accessible via le bouton "Modifier" sur chaque ligne

A screenshot of a website

AI-generated content may be incorrect.

**1.7 Delete Products**

**Status**: **Fully Implemented**

Fonctionnalité de suppression avec :

* Dialogue de confirmation pour éviter les suppressions accidentelles
* Affiche le nom du produit à supprimer
* Suppression en cascade de toutes les tables liées (products, clothes/shoes/accessories)
* Rafraîchissement automatique du tableau après suppression
* Accessible via le bouton "Supprimer"

A screenshot of a computer

AI-generated content may be incorrect.

Avant suppresion du “Summer Floral Dress” :

A screenshot of a computer

AI-generated content may be incorrect.

Aprés:

A screenshot of a screen

AI-generated content may be incorrect.

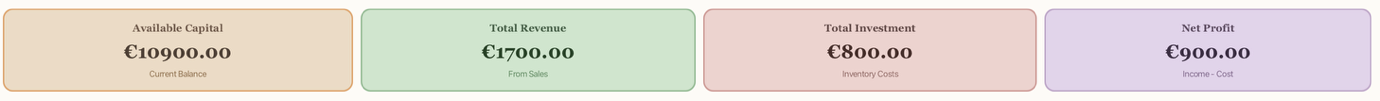
**1.8 Financial Dashboard**

**Status**: **Fully Implemented**

Suivi financier avec 4 cards:

* **Available Capital**: Current balance
* **Total Revenue**: Sum of all sales
* **Total Investment**: Sum of all purchases
* **Net Profit**: Revenue - Investment

Formule impéméntée : Current Capital = Initial Capital + Total Income - Total Cost

Toutes les valeurs se mettent à jour en temps réel après chaque transaction.

**1.9 Purchase Items (Stock Replenishment)**

**Status**: **Fully Implemented**

Fonctionnalité d'achat incluant :

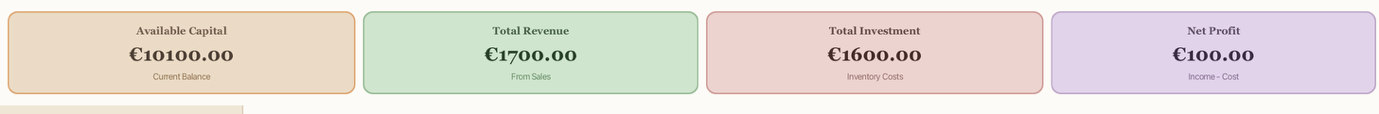
* Dialogue de saisie pour la quantité
* Affiche le prix unitaire (prix d'achat)
* **Dialogue de confirmation** affichant :
  + Nom du produit
  + Quantité
  + Prix unitaire
  + Coût total
  + Invite "Procéder à l'achat ?"
* **Validation** : Vérifie si le capital disponible est suffisant
* **Gestion des erreurs** : Affiche "Capital insuffisant" si le budget est dépassé
* **Mises à jour** :
  + La quantité en stock augmente
  + Le coût total augmente
  + Le capital actuel diminue
  + Transaction enregistrée dans la base de données

A screen shot of a purchase

AI-generated content may be incorrect.A screenshot of a purchase form

AI-generated content may be incorrect.

A screen shot of a computer

AI-generated content may be incorrect.

**1.10 Sell Items**

**Status**: **Fully Implemented**

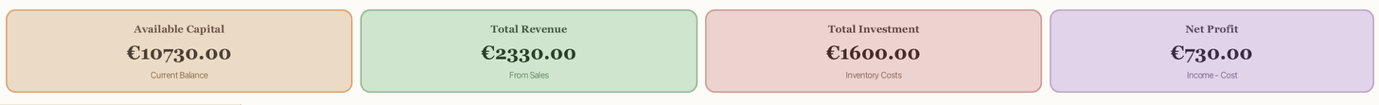
Fonctionnalité de vente incluant :

* Dialogue de saisie pour la quantité
* Affiche le prix final (utilise le prix réduit si remise active)
* Affiche le stock disponible
* **Dialogue de confirmation** affichant :
  + Nom du produit
  + Quantité
  + Prix unitaire (avec note de remise si applicable)
  + Revenu total
  + Invite "Procéder à la vente ?"
* **Validation** : Vérifie si le stock disponible est suffisant
* **Gestion des erreurs** : Affiche "Stock insuffisant" si tentative de vendre plus que disponible
* **Mises à jour** :
  + La quantité en stock diminue
  + Le revenu total augmente
  + Le capital actuel augmente
  + Transaction enregistrée dans la base de données

A screenshot of a computer screen

AI-generated content may be incorrect.A screenshot of a computer screen

AI-generated content may be incorrect.



**1.11 Apply Discounts**

**Status**: **Fully Implemented**

* **👗 Clothes**: 30% discount
* **👠 Shoes**: 20% discount
* **👜 Accessories**: 50% discount

Fonctionnalités :

* Dialogue d'information montrant les taux de remise pour toutes les catégories
* Menu déroulant de sélection de catégorie
* Applique la remise à TOUS les produits de la catégorie sélectionnée
* **Indicateurs visuels** :
  + La colonne prix final montre le montant réduit
  + La colonne remise affiche le pourcentage (ex : "30% OFF") avec badge doré
  + Lors de la vente d'articles remisés, la vente utilise le prix réduit

A screenshot of a screen

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.

**1.12 Stop/Remove Discounts**

**Status**: **Fully Implemented**

Fonctionnalités:

* Dialogue de sélection de catégorie
* Retire la remise de TOUS les produits de la catégorie sélectionnée
* **Mises à jour visuelles :**
  + Le prix final revient au prix de vente original
  + La colonne remise passe de "30% OFF" à "No discount"
  + Le style du badge de remise est retiré

**A screenshot of a discount

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.**

**1.13 Comprehensive Data Validation**

**Status**: **Fully Implemented**

Système de validation complet avec messages d'erreur:

**Validation de Saisie Produit** :

* Le nom du produit ne peut pas être vide
* Les prix doivent être des nombres positifs
* Le prix de vente doit être supérieur au prix d'achat
* Les champs spécifiques à la catégorie sont obligatoires (taille, couleur, etc.)
* Les champs numériques sont validés (la pointure doit être un nombre)

**Validation de Transaction** :

* La quantité doit être un entier positif
* Impossible de vendre plus que le stock disponible
* Impossible d'acheter sans capital suffisant
* Détection de format de nombre invalide

**Messages d'Erreur :**

* Descriptions d'erreur
* Alertes dialogue avec icônes appropriées
* Empêche la soumission jusqu'à correction des erreurs

Exemples des messages d’errreurs:

**A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a error message

AI-generated content may be incorrect.**

**1.14 Design & User Experience**

**Status**:  **Fully Implemented**

* **Hiérarchie Visuelle** : Sections claires avec bordures et arrière-plans subtils
* **Éléments Interactifs** : Effets de survol sur les boutons
* **Codage Couleur** :
  + Badges de catégorie avec couleurs distinctes
  + Indicateurs de stock (rouge/orange/vert)
  + Badges de remise
  + Cartes financières avec dégradés thématiques

A screenshot of a computer

AI-generated content may be incorrect.

**1.15 Additional Features Implemented**

**Status**: **Bonus Features**

* **Affichage du Nombre de Produits** : Montre le total de produits dans la vue actuelle
* **Indicateur de Catégorie Actuelle** : Affiche la catégorie active en haut du catalogue
* **Horodatage de Dernière Mise à Jour** : Montre quand les données ont été rafraîchies
* **Animations de Succès** : Animations de fondu pour les messages de succès
* **Colonne Détails Produit** : Vue rapide des spécifications du produit dans le tableau

**2. Technical Architecture**

**2.1 MVC Design Pattern**

**Successfully Implemented**

Séparation claire des responsabilités :

* **Modèle** : Product (abstrait), Clothes, Shoes, Accessories, Transaction, AppSettings, ProductCategory, TransactionType
* **Vue** : hello-view.fxml
* **Contrôleur** : HelloController (logique principale), ProductDialogController (dialogues ajout/modification)

**2.2 DAO Design Pattern**

**Successfully Implemented**

Couche d'accès aux données complète :

* **DatabaseConnection** : Pattern Singleton pour gestion de connexion MySQL
* **ProductDAO** : Opérations CRUD pour tous types de produits
* **TransactionDAO** : Gestion de l'historique des transactions
* **AppSettingsDAO** : Données financières et traitement des transactions

Fonctionnalités :

* Instructions préparées pour prévenir l'injection SQL
* Support des transactions avec capacité de rollback
* Opérations en cascade pour intégrité des données

**2.3 Database Design**

**Successfully Implemented - Third Normal Form (3NF)**

Base de données relationnelle bien structurée :

* **products** : Table de base avec attributs communs
* **clothes** : Attributs spécifiques aux vêtements (relation 1:1)
* **shoes** : Attributs spécifiques aux chaussures (relation 1:1)
* **accessories** : Attributs spécifiques aux accessoires (relation 1:1)
* **transactions** : Historique complet des transactions
* **app\_settings** : Suivi financier

Fonctionnalités :

* Contraintes de clés étrangères
* Opérations de suppression en cascade
* Colonnes indexées pour performance
* Types de données et contraintes appropriés

**3. Données d'Exemple**

La base de données inclut **22 produits d'exemple** pour les tests :

* **8 articles de vêtements** : Robes, vestes, jeans, t-shirts, cardigans, etc.
* **6 articles de chaussures** : Baskets, talons, bottes, sandales, etc.
* **8 accessoires** : Sacs à main, bijoux, foulards, montres, etc.

Capital initial fixé à **€10,000.00**

**5. Limitations Connues**

1. **Taux de Remise Fixes** : Les remises sont fixées à 30% (vêtements), 20% (chaussures), 50% (accessoires). Impossible de définir des remises personnalisées par produit individuel.
2. **Pas d'Authentification Utilisateur** : Système mono-utilisateur. Tout utilisateur ayant accès à l'application a tous les droits administrateur.
3. **Pas de Vue d'Historique des Transactions** : Bien que les transactions soient enregistrées dans la base de données, il n'y a pas d'interface pour visualiser l'historique.
4. **Devise Unique** : Supporte uniquement l'Euro (€). Pas de support multi-devises.
5. **Pas de Gestion des Fournisseurs** : Le prix d'achat est fixe par produit. Impossible de suivre différents fournisseurs ou variations de prix.
6. **Pas de Gestion des Clients** : Le système se concentre sur la gestion d'inventaire. Les données clients ne sont pas suivies.

**6. Résultats des Tests**

Toutes les fonctionnalités testées avec succès :

* ✅ Opérations CRUD sur les produits (Create, Read, Update, Delete)
* ✅ Filtrage par catégorie et navigation
* ✅ Fonctionnalité de recherche
* ✅ Options de tri multiples
* ✅ Transactions d'achat avec validation du capital
* ✅ Transactions de vente avec validation du stock
* ✅ Application et retrait des remises
* ✅ Précision des calculs financiers
* ✅ Validation des données et gestion des erreurs
* ✅ Intégrité de la base de données (suppressions en cascade, clés étrangères)
* ✅ Réactivité et mise en page de l'interface

**7. Conclusion**

Le projet WomenShop implémente avec succès toutes les fonctionnalités requises pour un système complet de gestion de stock. L'application démontre :

**Couverture Complète des Fonctionnalités** : Les 12 exigences principales implémentées   
**Architecture Solide** : Patterns MVC et DAO appropriés  
**Intégrité des Données** : Base de données 3NF bien conçue  
**Validation Robuste** : Gestion complète des erreurs  
**Prêt pour Production** : Entièrement fonctionnel et testé

Le système fournit une base solide pour gérer une boutique de mode féminine avec possibilité d'améliorations futures telles que rapports, analyses et gestion des clients.

**8. Installation & Setup**

**Prerequisites**:

* Java JDK 17 or higher
* MySQL Server 8.0 or higher
* Maven 3.6 or higher

**Setup Steps**:

1. Create MySQL database using provided SQL script
2. Configure database credentials in DatabaseConnection.java
3. Run mvn clean install
4. Execute mvn javafx:run

**Database Configuration**:

* Database: womenshop
* Default User: root
* Tables: 6 (products, clothes, shoes, accessories, transactions, app\_settings)
* Sample Data: 22 products pre-loaded