



CY Tech - 2024/2025

Analyse et programmation orienté objet - Java - Rendu 3

DABOUT Mattéo,
DOSSANTOS Emma,
EISH Aicha,
ROBARD Baptiste,
YAZIDI Asma

*Première année d'ingénieur
Génie mathématique apprentissage*

*Analyse et programmation orienté objet
Verin Renaud*

Table des matières

1	Introduction	2
1.1	Contexte du projet	2
1.2	Objectifs et enjeux du rendu 3	2
2	Organisation du Projet	2
2.1	Package application :	3
2.2	Package classe :	3
2.3	Package DAO	3
2.4	Package FXML :	3
2.5	Package IHM :	3
2.6	Package test	3
2.7	Package MainTest	3
3	Fonctionnalités Administrateur	4
3.1	Super Admin	4
3.1.1	Connexion	4
3.1.2	Gestion des centres de tri	4
3.2	Admin	4
3.3	Gestion des poubelles du centre de tri	4
3.4	Gestion des utilisateurs	4
3.5	Statistiques	4
3.6	Gestion des partenariats	5
3.7	Gestion des catégories de produit	5
4	Parcours Utilisateur	5
4.1	Connexion et inscription	5
4.2	Tableau de bord	5
4.3	Déposer un déchet	5
4.4	Acheter un bon avec ses points	6
4.5	Consulter son historique	6
5	Difficultés rencontrées	6
5.1	Ajouter les interfaces à notre projet	6
5.2	Naviguer entre les interfaces sans se perdre	6
5.3	Une étape difficile, mais super formatrice	6
6	Bilan et perspectives	7
6.1	Bilan général du rendu 3	7
6.2	Pistes d'amélioration	7
7	Annexe :	8

1. Introduction

• Contexte du projet

Dans le cadre de notre cursus nous devons ce semestre réaliser un projet en groupe dans la matière analyse et programmation orienté objet. Notre groupe est composé de Mattéo Dabout, Asma Yazidi, Baptiste Robard, Aicha Eish et Emma Dos Santos.

Le thème de ce projet porte sur la gestion intelligente du tri sélectif des déchets. L'idée est de développer une application simulant le fonctionnement d'un système dans lequel des ménages déposent leurs déchets dans des poubelles connectées. Ces dernières sont capables d'identifier l'utilisateur, de mesurer la quantité et la nature des déchets déposés, de vérifier la conformité du tri, et d'envoyer des alertes au centre de tri lorsqu'elles sont pleines. En contrepartie d'un tri bien fait, les ménages reçoivent des points de fidélité qu'ils peuvent échanger contre des réductions chez des commerces partenaires.

Le projet se déroule en trois rendus :

- Une modélisation UML du système,
- Une implémentation Java en mode texte,
- L'implémentation de l'IHM.

• Objectifs et enjeux du rendu 3

Pour ce troisième rendu, il nous a été demandé de rendre l'application utilisable en ajoutant un IHM à notre précédent travail. L'idée, c'est que tout soit simple et concret pour l'utilisateur. Il doit pouvoir se connecter à son compte, jeter ses déchets dans les poubelles connectées, voir son historique, consulter ses points de fidélité et profiter des réductions proposées par les commerçants partenaires. Nous avons également intégré un espace pour les administrateurs où ils pourront gérer les poubelles, suivre les statistiques et s'occuper des partenariats avec les commerces.

Ce rendu constitue une étape essentiel dans la progression du projet. Techniquement, il faut réussir à établir une bonne connexion entre JavaFX et la logique que nous avons déjà programmée, ainsi qu'avec la base de données MySQL. Il est également nécessaire que l'application soit plaisante à utiliser et qu'elle puisse représenter les diverses fonctionnalités élaborées jusqu'à maintenant.

2. Organisation du Projet

Pour ce troisième rendu, nous avons gardé la même organisation du code que dans les rendus précédents, tout en y intégrant l'interface graphique. Le projet est divisé en plusieurs packages pour séparer clairement les différentes parties de l'application : la logique métier, l'accès aux données, l'interface utilisateur, et les tests.

- **Package application :**

Ce package contient le point d'entrée de l'application. C'est ici que se trouve la classe principale permettant de lancer l'application JavaFX.

- **Package classe :**

Ce package contient toutes les classes qui correspondent directement aux entités qu'on a pu définir dans le diagramme UML. Il sert à définir les attributs et les méthodes de chaque entité tout en prenant en compte les relations entre eux.

- **Package DAO**

Ce package rassemble toutes les classes DAO (Data Access Object) qui servent de pont entre les objets Java et la base de données MySQL `tri_selectif`. On y implémente les opérations CRUD (Create, Read, Update, Delete) pour chaque entité avec chaque classe qui est dédiée à une table de la base.

- **Package FXML :**

Il contient les fichiers FXML générés avec Scene Builder. Chaque fichier correspond à une vue spécifique de l'application. Cela permet de construire l'interface graphique de façon claire et structurée.

- **Package IHM :**

Ce package regroupe les contrôleurs associés aux vues JavaFX. Ces classes gèrent les actions de l'utilisateur (clics, saisies, etc.), font appel aux classes métier ou aux DAO lorsque c'est nécessaire, et mettent à jour l'interface en conséquence.

- **Package test**

Ce package contient au moins un test spécifique pour chaque classe métier afin de valider leur comportement logique mais également un test DAO pour chaque classe qui s'assure que les opérations de chaque entité sur la base de données MySQL se passent bien. Ceci permet donc de tester séparément : la logique métier sans dépendre de la base de données et la logique en utilisant la base de données via les DAO.

- **Package MainTest**

Ce package contient les classes principales utilisées pour lancer des tests globaux sur toutes les classes avec et sans DAO.

3. Fonctionnalités Administrateur

- **Super Admin**

- **Connexion**

On a tout d'abord le super admin qui se connecte grâce à l'adresse mail `admin@tri.fr` et le mot de passe : `admin`.

- **Gestion des centres de tri**

Une fois connecté, il peut accéder à sa page d'accueil de gestion du réseau de centres de tri (cf. figure 1), où il peut voir tous les centres de tri disponibles avec des infos comme leur nom, leur adresse, l'ID de leur admin associé, le nombre de dépôts qui ont été faits dans ces centres et la quantité totale.

Depuis cette page, il peut soit ajouter un centre, soit en supprimer un. Pour en ajouter un, grâce à la page de création d'un centre de tri, il doit indiquer le nom, l'adresse, l'e-mail de l'admin et le mot de passe associé (cf. figure 2).

S'il souhaite supprimer un centre, il a le choix du mode de suppression : ou bien il supprime également tous les utilisateurs de ce centre, ou bien il peut les transférer vers un autre centre (cf. figure 3), qu'il doit ensuite sélectionner (cf. figure 4).

- **Admin**

On a ensuite les admins "simple" qui sont chacun lié à un centre de tri qu'ils peuvent gérer (figure 5).

- **Gestion des poubelles du centre de tri**

Ils peuvent notamment gérer les poubelles du centre (figure 6) : ils peuvent ajouter une nouvelle poubelle en indiquant le type de déchet que doit recevoir la poubelle, sa capacité max et son emplacement (figure 7). Ils peuvent également modifier une poubelle (figure 8), en supprimer une ou encore vider une poubelle lorsque celle-ci a atteint sa capacité max (figure 9), ce qui permet de remettre à 0 la capacité de la poubelle (figure 10).

- **Gestion des utilisateurs**

L'admin peut également gérer les utilisateurs de son centre de tri (figure 11) : il peut créer des comptes utilisateurs ou admin associés au centre de tri (figure 12), les modifier (figure 13), les supprimer et les transférer vers un autre centre de tri.

- **Statistiques**

Il peut également accéder aux statistiques (figure 14) de son centre de tri, notamment la quantité totale de déchets qui ont été déposés dans son centre et la quantité moyenne par dépôt.

- **Gestion des partenariats**

Enfin, il peut gérer les partenariats du centre (figure 15). En sélectionnant un commerce, il peut alors accéder au contrat associé avec celui-ci (figure 16) qui indique notamment l’ID de ce contrat, sa date de début et sa date de fin. Il peut également ajouter un nouveau commerce avec lequel il va devoir spécifier les infos du contrat, notamment les dates de début et de fin et le taux de réduction, en plus du nom et de l’emplacement (figure 17).

Une fois créé, on peut modifier que le nom et l’emplacement, pas les informations liées au contrat (figure 18). On peut également le supprimer.

- **Gestion des catégories de produit**

On peut gérer les catégories de produits que va proposer le commerce (figure 19) : notamment créer une nouvelle catégorie de produit en indiquant le nom, le taux de réduction associé à la catégorie, et les points nécessaires qui permettront à l’utilisateur de débloquent un bon de réduction dans cette catégorie de produit (figure 20). On peut également la modifier.

Une fois créée, on peut également indiquer les produits que contient la catégorie sélectionnée (figure 21). On peut en ajouter (figure 22), en indiquant leur nom et leur prix ainsi que la date où le commerce les a reçus, les modifier, et en supprimer.

4. Parcours Utilisateur

- **Connexion et inscription**

L’utilisateur commence par l’interface de connexion (figure 23). Il doit y renseigner son adresse mail et son mot de passe. S’il entre des identifiants valides, il est redirigé vers son tableau de bord personnel (figure 24). En cas d’erreur, un message lui indique que la connexion a échoué (figure 25). S’il ne possède pas encore de compte, il peut en créer un via l’interface d’inscription (figure 26).

- **Tableau de bord**

Une fois connecté, l’utilisateur accède à son tableau de bord (figure 24). Il peut y voir ses informations personnelles, le centre de tri auquel il est rattaché, son solde de points, et la liste des poubelles accessibles. Pour chaque poubelle, sont indiqués le type de déchet attendu, la capacité maximale et l’emplacement. Trois boutons lui permettent ensuite d’accéder directement aux fonctionnalités principales : déposer un déchet, faire un achat, ou consulter son historique.

- **Déposer un déchet**

En déposant, il accède à l’interface dédiée au dépôt de déchets (figure 27). Il choisit alors le type de déchet qu’il veut jeter, et indique son poids. L’application vérifie automatiquement si le tri est correct :

- Si le type est incorrect, un message d’erreur s’affiche (figure 28) ;
- Si le déchet est trop volumineux, une autre alerte apparaît (figure 29) ;

- Si tout est bon, une confirmation est affichée (figure 30) et des points sont ajoutés à son solde.

- **Acheter un bon avec ses points**

Depuis le tableau de bord, l'utilisateur peut aussi faire un achat dans un commerce (figure 31). Depuis l'interface d'achat, il retrouve les commerces partenaires de son centre, avec les bons actuellement disponibles. S'il souhaite consulter les offres d'un commerce (figure 32), il y voit la liste des catégories de produits proposées, les réductions associées, et le nombre de points nécessaires.

Si l'utilisateur a assez de points, il peut acheter un bon (figure 33). Une fois le bon obtenu, il peut l'utiliser. Il doit alors choisir sur quel produit de la catégorie concernée il souhaite appliquer le bon (figure 34), car les bons sont liés à une catégorie précise.

- **Consulter son historique**

Enfin, l'utilisateur peut voir son historique de dépôts pour accéder à la page dédiée (figure 35). Cette interface affiche l'ensemble de ses dépôts : type de déchet, date, points gagnés, et achats effectués avec des bons.

5. Difficultés rencontrées

- **Ajouter les interfaces à notre projet**

Ce troisième rendu a été le plus complexe à réaliser, car il fallait enfin tout rassembler : la logique Java, la base de données, et l'interface graphique avec JavaFX. La première vraie difficulté a été de réussir à faire communiquer les vues avec notre code Java déjà en place. On ne savait pas toujours comment passer les bonnes données d'une vue à l'autre, ni comment déclencher les actions depuis un simple clic sur un bouton. Il a fallu pas mal d'essais et de recherches pour comprendre comment structurer tout ça correctement. On a du également modifier de nombreux éléments du code ce qui pouvait être parfois un peu frustrant vis-à-vis de ce qu'on pensait abouti.

- **Naviguer entre les interfaces sans se perdre**

Faire passer l'utilisateur d'un écran à un autre, tout en gardant en mémoire des informations comme l'utilisateur connecté ou la poubelle sélectionnée, a aussi posé pas mal de problèmes. On a du faire beaucoup de tests pour réussir à produire quelque chose de logique.

- **Une étape difficile, mais super formatrice**

Malgré tous ces obstacles, cette dernière partie du projet a été la plus satisfaisante. C'est là qu'on a vraiment vu notre application prendre vie, avec des fonctionnalités concrètes qu'on pouvait utiliser. C'était dur, mais très enrichissant.

6. Bilan et perspectives

• Bilan général du rendu 3

Ce troisième rendu a été le plus concret et le plus abouti du projet. Après avoir travaillé dans les rendus précédents sur la modélisation UML et la logique métier en Java, nous avons cette fois intégré l'interface graphique pour vraiment donner vie à notre application.

Nous avons pu construire une application fonctionnelle et exploitable, aussi bien du côté utilisateur que du côté administrateur.

Contrairement aux rendus précédents où nous partagions nos fichiers manuellement, nous avons cette fois mis en place un dépôt Git pour travailler plus efficacement. Cela nous a permis de partager notre code en temps réel, de suivre les modifications, et de mieux coordonner notre travail en équipe.

Cependant, il est vrai que ce rendu est tombé à une période assez chargée : la fin du semestre, les révisions pour les partiels, d'autres projets à rendre. . . Tout cela a limité le temps que nous aurions aimé consacrer à ce projet. Malgré cela, nous avons fait de notre mieux pour livrer une version stable, complète et cohérente.

Ce rendu a été à la fois exigeant et formateur. Il nous a permis de consolider nos compétences sur la construction d'une vraie application Java connectée à une base de données. Voir notre application fonctionner « comme une vraie » a été une vraie source de satisfaction.

• Pistes d'amélioration

Même si le projet est globalement abouti, il reste de nombreuses pistes intéressantes pour aller plus loin. On pourrait par exemple améliorer l'interface graphique en ajoutant du CSS pour un meilleur rendu visuel. Certaines fonctionnalités pourraient également être renforcées, comme l'application d'un taux de réduction spécifiques en fonction des contrats.

Il serait aussi pertinent d'ajouter de nouvelles contraintes, par exemple la suppression automatique des produits une fois qu'ils sont périmés, ou encore l'ajout de règles plus strictes concernant la validité ou la durée des contrats.

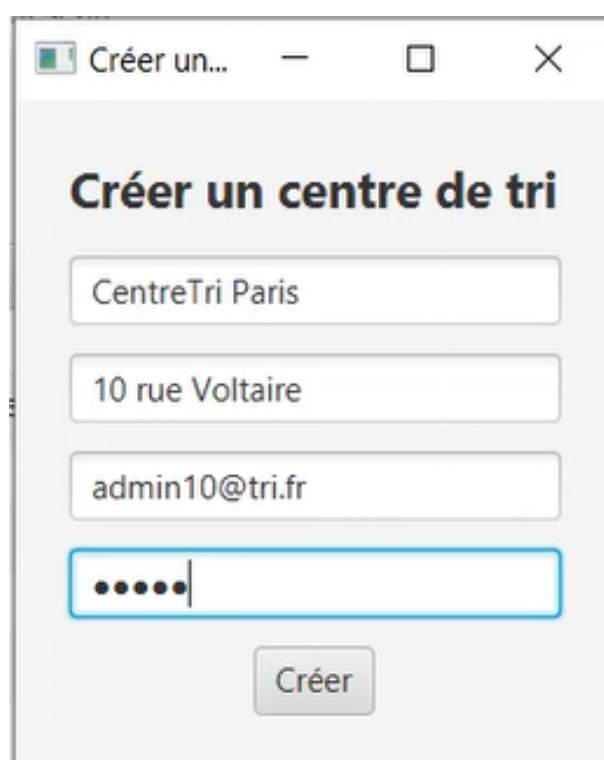
7. Annexe :



Nom du centre	Adresse	Admin associé	Nb dépôts	Quantité totale (kg)
CentreTri1	Adresse1	Admin ID : 101	0	0.0
CentriTri Montreuil	10 rue de la forêt	Admin ID : 97537	0	0.0

Ajouter un centre Supprimer le centre sélectionné Retour à l'accueil

FIGURE 1 – Interface de gestion du réseau de centres de tri par le super admin.



Créer un centre de tri

CentreTri Paris

10 rue Voltaire

admin10@tri.fr

•••••

Créer

FIGURE 2 – Formulaire de création d'un centre de tri.

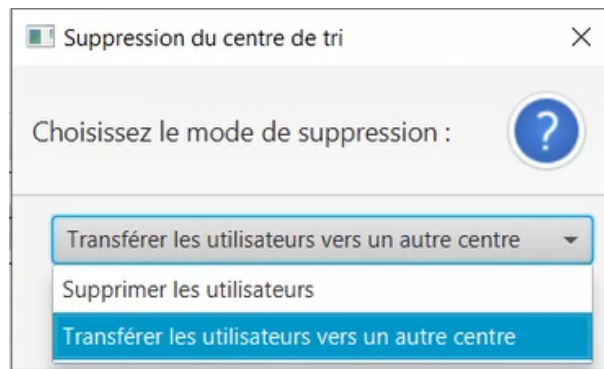


FIGURE 3 – Options disponibles lors de la suppression d'un centre de tri.

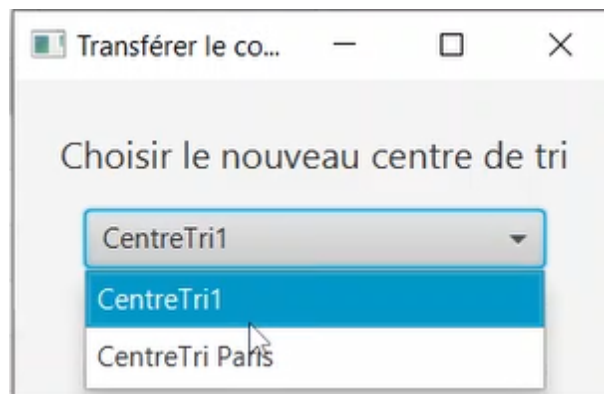


FIGURE 4 – Options disponibles pour transférer les utilisateurs.

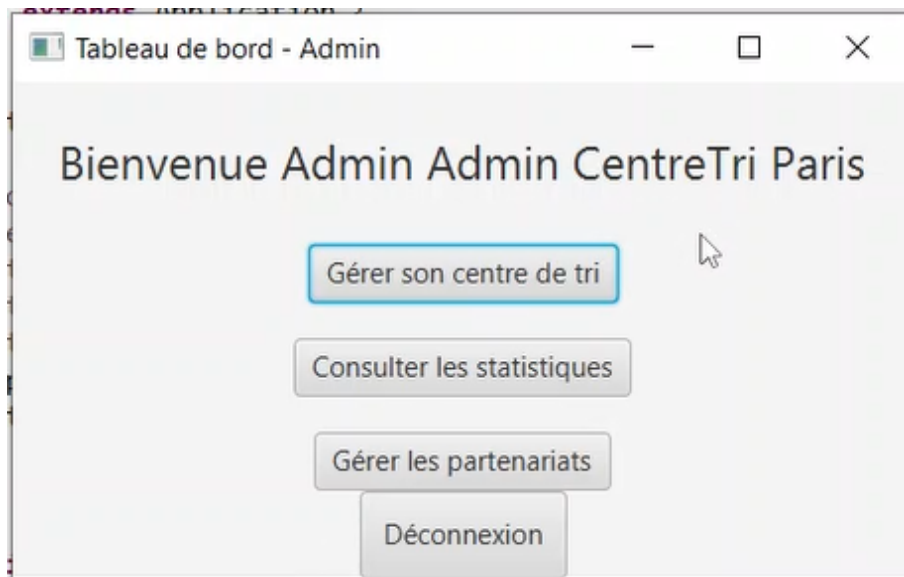


FIGURE 5 – Tableau de bord d'un administrateur de centre de tri

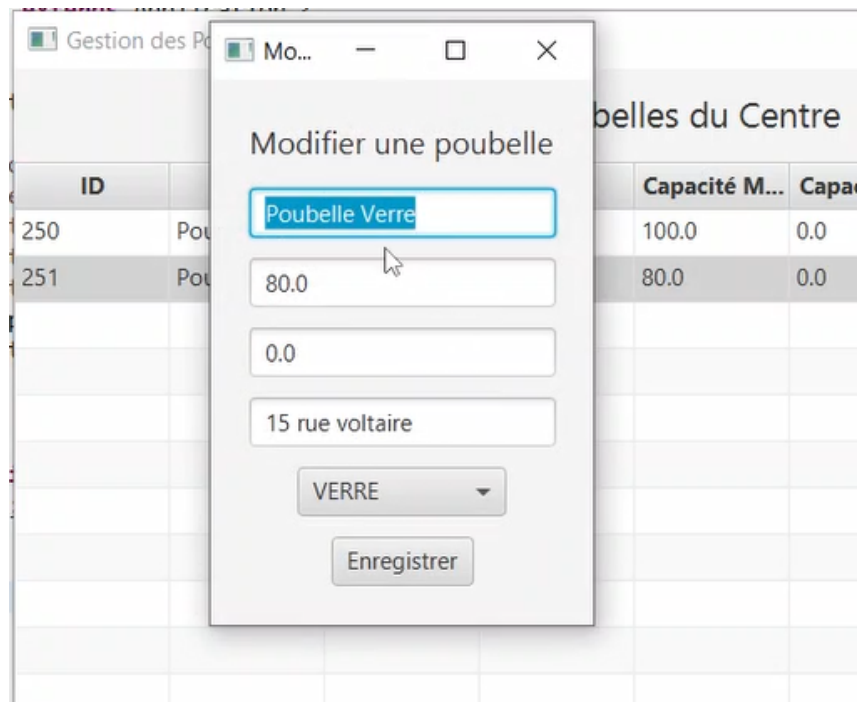


FIGURE 8 – Modification d'une poubelle

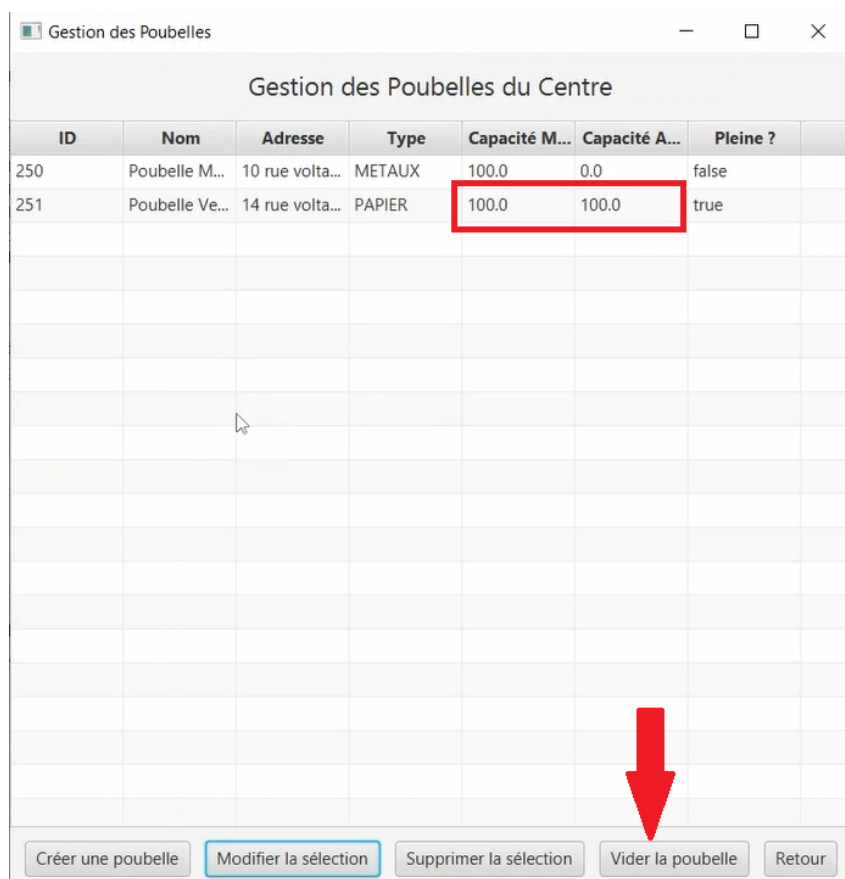


FIGURE 9 – Vidage manuel d'une poubelle pleine

Gestion des Poubelles

Gestion des Poubelles du Centre

ID	Nom	Adresse	Type	Capacité M...	Capacité A...	Plaine ?
250	Poubelle M...	10 rue volta...	METAUX	100.0	0.0	false
251	Poubelle Ve...	14 rue volta...	PAPIER	100.0	0.0	false

FIGURE 10 – Capacité de la poubelle remise à zéro après vidage

[illegible]

FIGURE 11 – Liste des utilisateurs du centre de tri

Créer un compte

Créer un compte

Jacob

jacob@gmail.com

.....

test123

Type (user/admin)

user

admin

FIGURE 12 – Création d'un nouveau compte utilisateur ou admin

Modi...

Modifier un compte

Jacob Rousseau

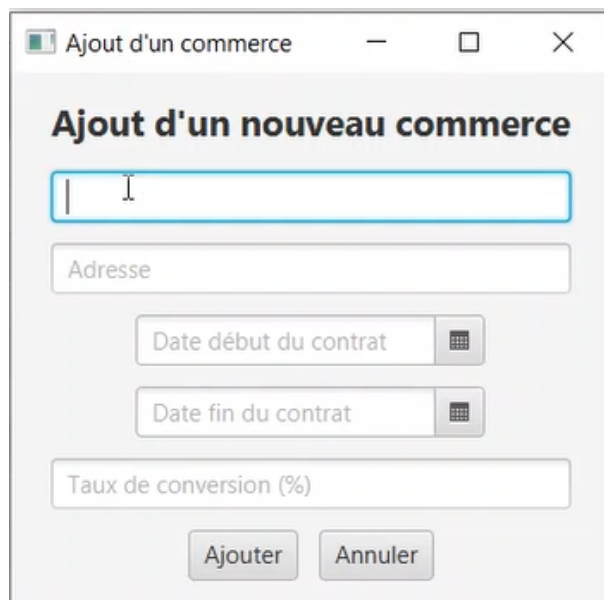
jacob@gmail.com

test123

user

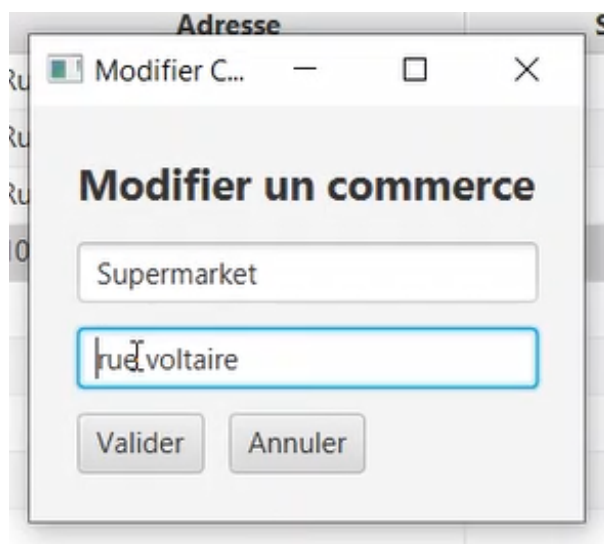
Enregistrer les modifications

FIGURE 13 – Modification d'un compte



A screenshot of a software window titled "Ajout d'un commerce". The window has a title bar with standard minimize, maximize, and close buttons. The main content area is titled "Ajout d'un nouveau commerce". It contains several input fields: a text field for the name (currently empty with a cursor), a text field for "Adresse", two date pickers for "Date début du contrat" and "Date fin du contrat", and a text field for "Taux de conversion (%)". At the bottom, there are two buttons: "Ajouter" and "Annuler".

FIGURE 17 – Ajout d'un commerce partenaire



A screenshot of a software window titled "Modifier C...". The window has a title bar with standard minimize, maximize, and close buttons. The main content area is titled "Modifier un commerce". It contains two text input fields: the first contains the word "Supermarket", and the second contains "rue voltaire" with a cursor. At the bottom, there are two buttons: "Valider" and "Annuler".

FIGURE 18 – Modification d'un commerce partenaire



FIGURE 19 – Gestion des catégories de produits du commerce

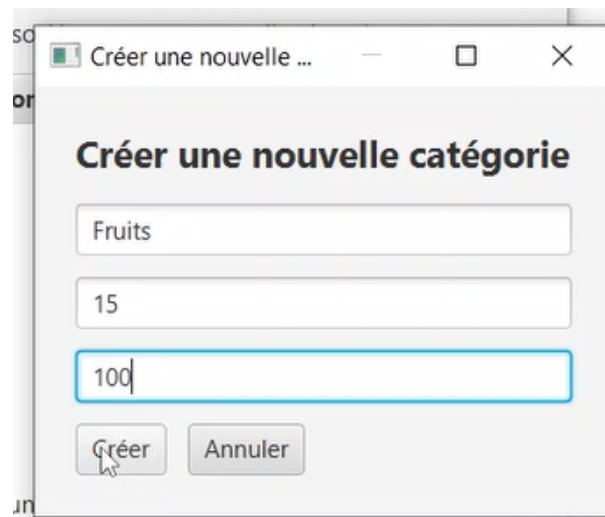


FIGURE 20 – Création d'une nouvelle catégorie

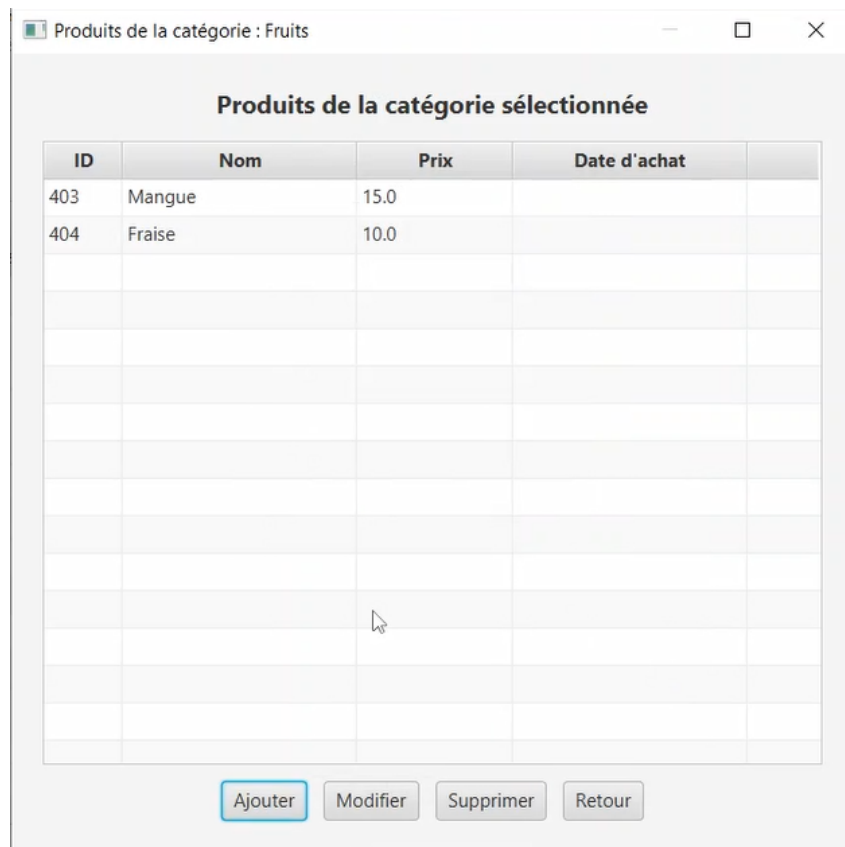


FIGURE 21 – Affichage des produits d’une catégorie

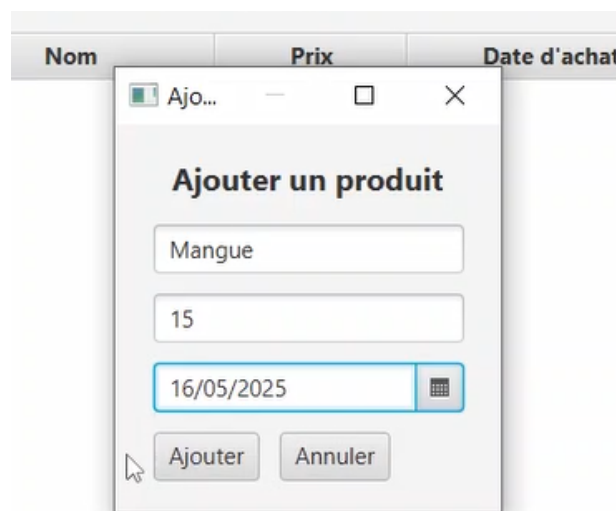


FIGURE 22 – Ajout d’un produit à une catégorie

Connexion Utilisateur

Connexion à la plateforme

Mot de passe

Se connecter

Créer un compte

FIGURE 23 – Page de connexion

Tableau de bord - Utilisateur

Bienvenue Jack Nicole

Adresse : 10 rue jacobin
Points de fidélité : 0.0
Centre assigné : CentreTri Paris

Vos poubelles autorisées

ID	Type	Capacité	Emplacement	Action	
250	METAUX	100.0	10 rue voltaire	Déposer	
252	PAPIER	150.0	10 rue voltaire	Déposer	

Faire un achat Voir l'historique des dépôts Se déconnecter

FIGURE 24 – Tableau de bord de l'utilisateur

Connexion Utilisateur

Connexion à la plateforme

testfaux

.....

Se connecter

Créer un compte

Email ou mot de passe incorrect.

FIGURE 25 – Page de connexion : erreur

Créer un compte

Mot de passe

Choisissez votre centre de tri

Créer le compte

FIGURE 26 – Inscription d'un utilisateur

Tableau de bord - Utilisateur

Déposer un déchet

Type de déchet :

Masse (en kg) :

Plastique

Verre

Papier

Métaux

Biodéchet

Valider le dépôt

FIGURE 27 – Dépôt d'un déchet

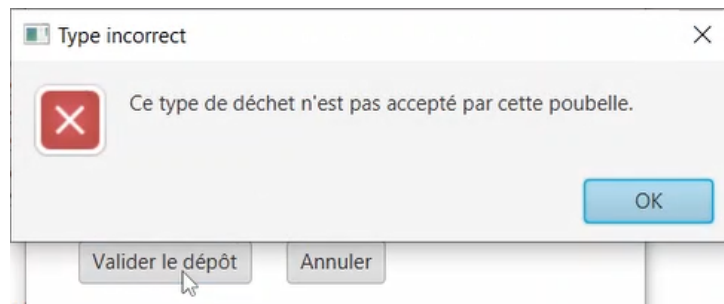


FIGURE 28 – Erreur : type de déchet incorrect

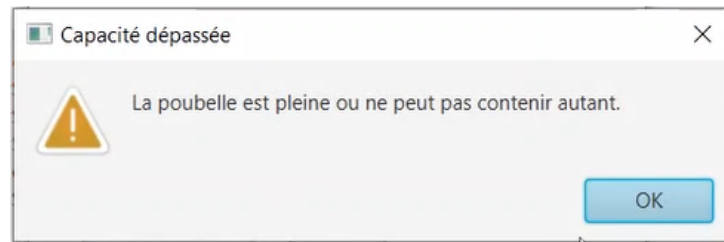


FIGURE 29 – Erreur : quantité de déchet trop élevée

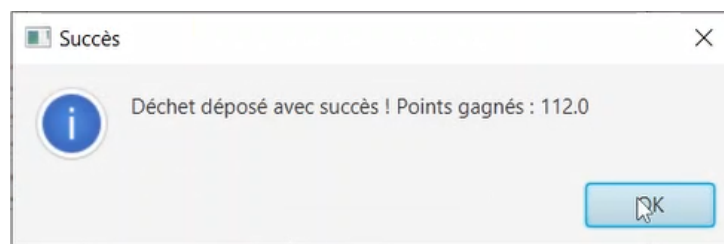


FIGURE 30 – Dépôt correct avec points

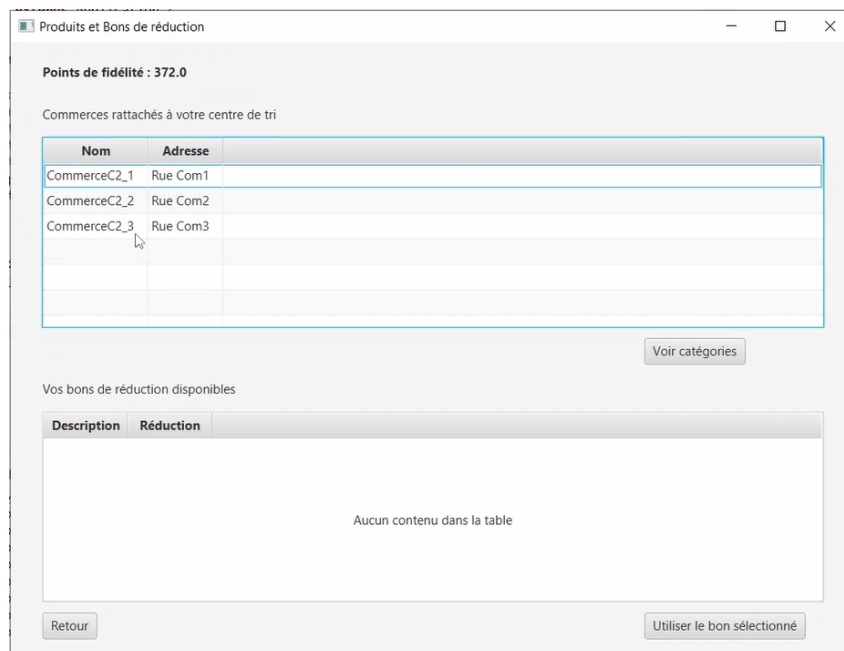


FIGURE 31 – Interface d'achat

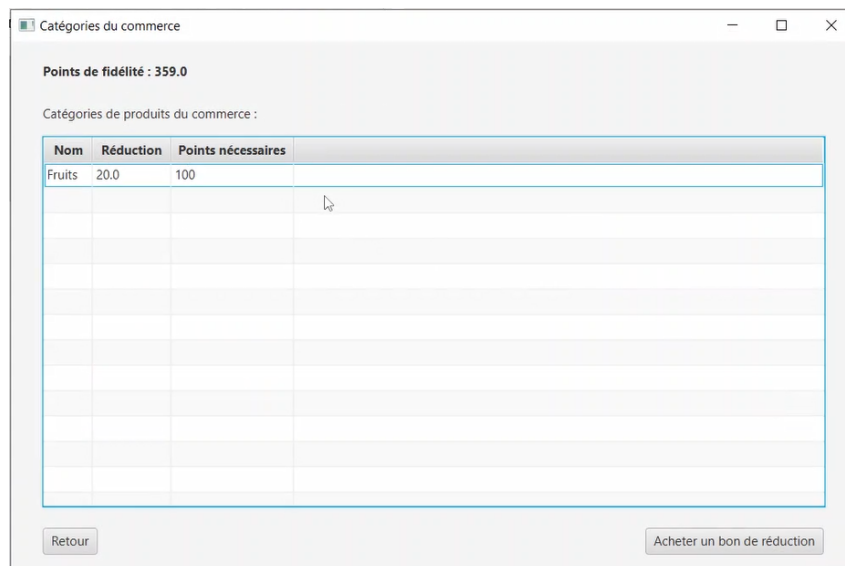


FIGURE 32 – Catégories de produits

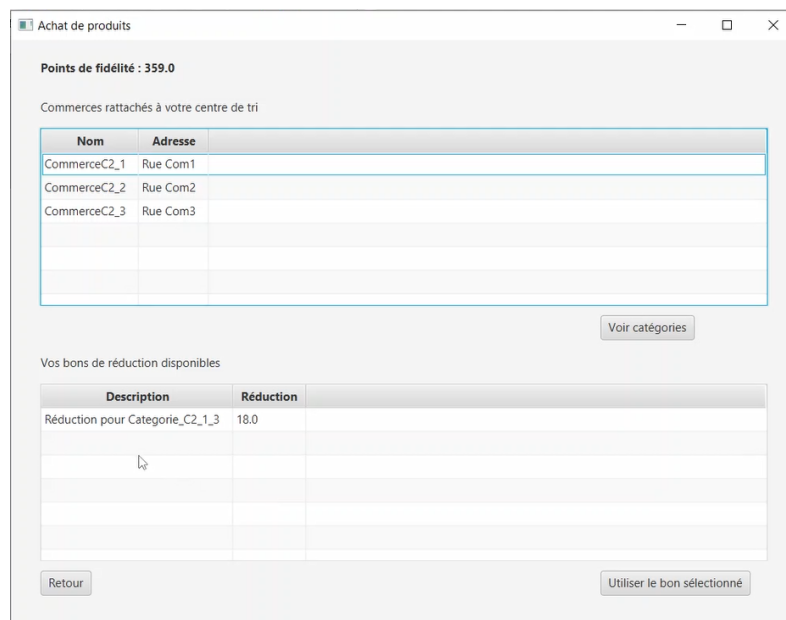


FIGURE 33 – Apparition du bon après achat

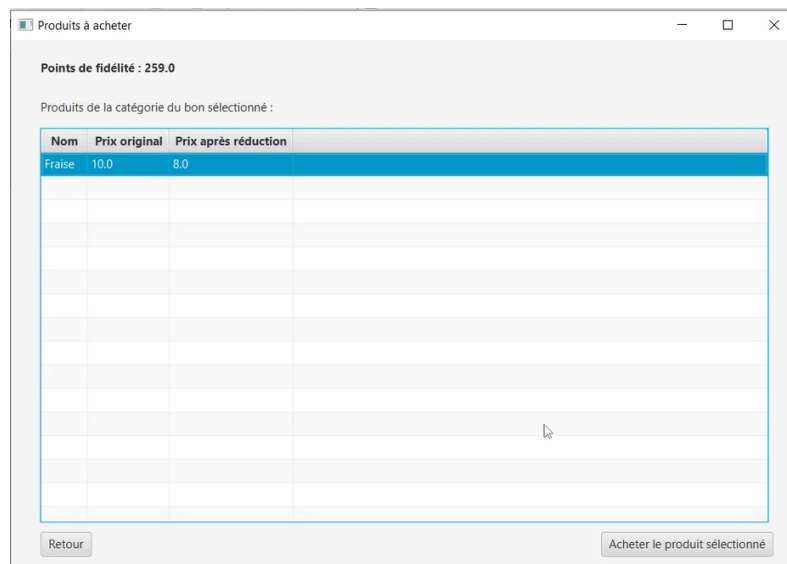


FIGURE 34 – Utilisation d'un bon sur un produit appartenant à sa catégorie

[illegible]

FIGURE 35 – Historique des dépôts et des bons utilisés

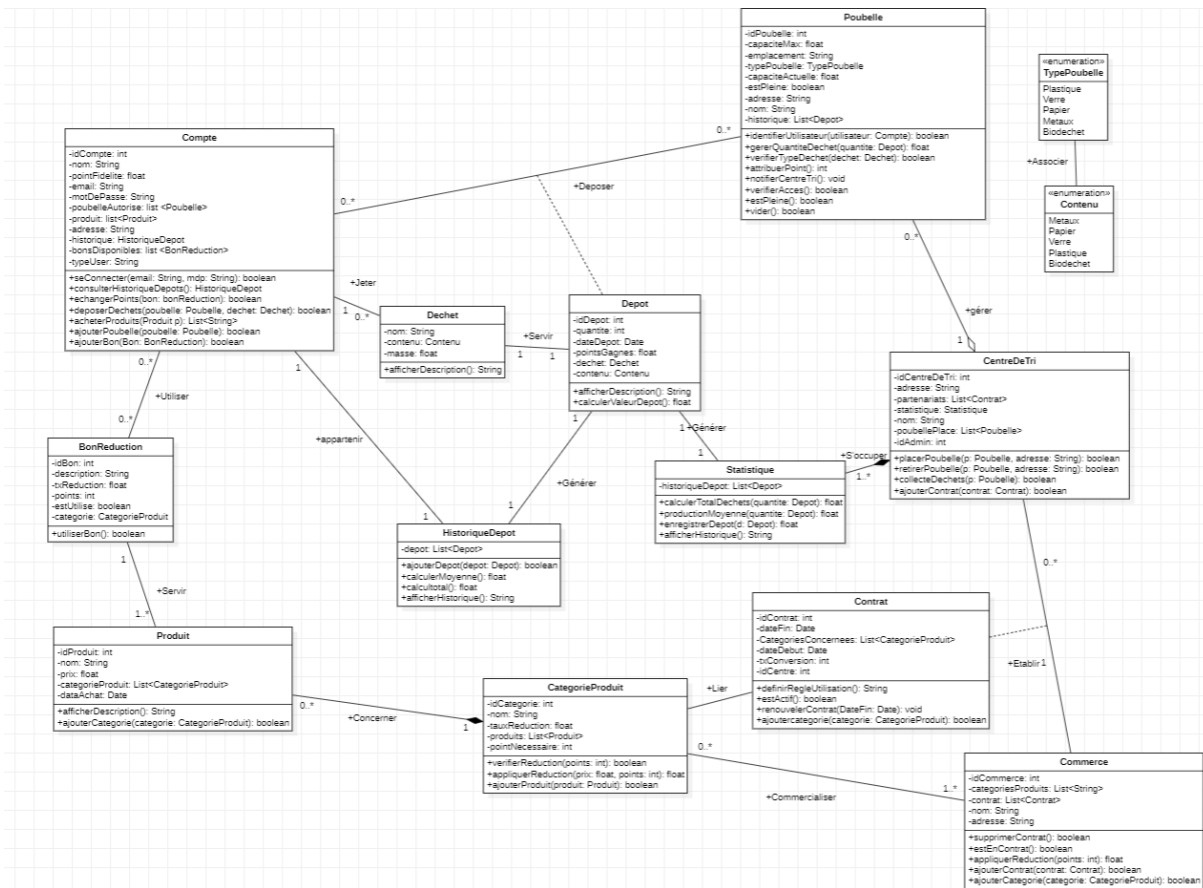


FIGURE 36 – UML

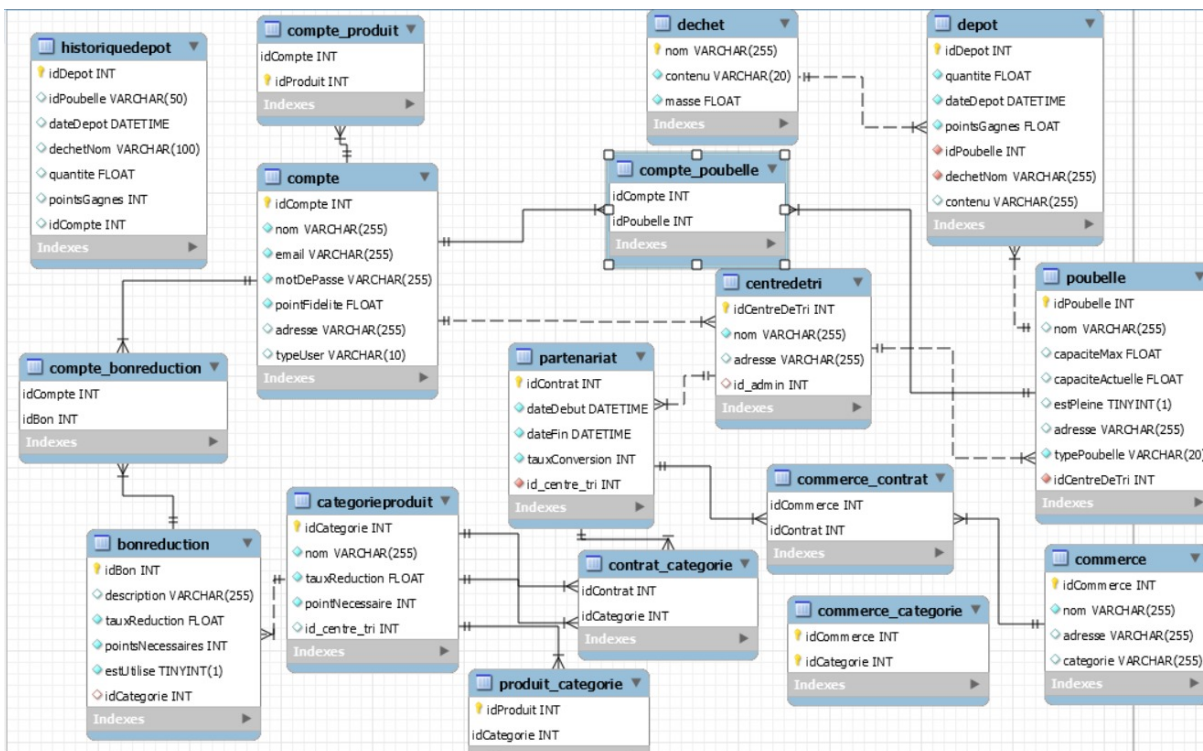


FIGURE 37 – MCD