



Rapport Projet Java

Astrid BROUCAS

Jade CANO

Rémi MACHINAL

Table des matières

Introduction	3
I. Présentation de la structure de notre projet.....	4
a) Les modifications apportées à notre UML.....	4
b) L'organisation de nos packages avec le logiciel Eclipse	4
II. L'interface pour le ménage	5
a) Description des différentes actions réalisables via l'interface du ménage	5
b) Présentation des différentes scènes de notre interface.....	7
III. L'interface pour le centre de tri	14
a) Description des différentes actions réalisables via l'interface du centre de tri	14
b) Présentation des différentes scènes de notre interface.....	15
Conclusion.....	23

Introduction

Afin de valider notre première année de cycle ingénieur, il nous est demandé de réaliser un programme de gestion de tri sélectif avec le langage Java. Le but de ce projet est de nous initier à la programmation orientée objet avec un sujet assez complexe pour couvrir toutes les bases de ce domaine.

Ayant déjà travaillé ensemble, nous connaissions notre efficacité lors de travaux de groupe. Nous avons décidé de renouveler cette collaboration pour réaliser ce projet. Chacun d'entre nous apporte une vision, une réflexion et des connaissances uniques au groupe ce qui contribue à un avancement efficace et coordonné du projet.

Ce projet comporte trois rendus, le premier étant un diagramme de classe UML couvrant l'entièreté du projet, le second une implémentation d'une partie du code avec toutes les fonctions de test et le dernier un IHM permettant la liaison entre l'humain et la machine. Ce rapport concerne le rendu final du projet.

Dans ce dernier rendu, nous allons réaliser l'interface Homme-machine (IHM) correspondant à notre projet. Tout d'abord, nous aborderons les quelques modifications apportées à notre UML, ainsi que l'organisation de nos packages avec le logiciel Eclipse. Ensuite, nous vous présenterons les deux interfaces que nous avons conçues : l'une pour le centre de tri et l'autre pour les ménages.

I. Présentation de la structure de notre projet

a) Les modifications apportées à notre UML

Afin de pouvoir créer une interface Homme-machine qui soit fonctionnelle, nous avons dû apporter quelques modifications à notre UML. Avant de commencer à coder notre IHM, nous avons défini les actions qui étaient réalisables à partir de l'interface du centre de tri et à partir de l'interface du ménage. Ainsi, nous nous sommes rendu compte qu'il fallait rajouter la méthode `utiliserBonAchat` à la classe « Compte ». Cette méthode permet d'utiliser un bon d'achat en retournant un code généré aléatoirement pour cette utilisation.

De plus, notre IHM est composée de deux interfaces : l'une pour le centre de tri et l'autre pour le ménage. Ce qui signifie qu'un centre de tri doit pouvoir se connecter sur notre interface, tout comme le ménage. Dans nos deux UML précédents, nous avons pensé à définir un identifiant et un code pour chaque ménage (i.e. pour chaque élément de la classe « Compte »). Cependant, nous n'avions pas pensé à ajouter ces deux attributs pour la classe « CentreTri ». Dans la version finale de notre UML, nous avons donc ajouté les attributs `code` et `id` à la classe « CentreTri ». Nous avons également rajouté l'attribut `nom` à cette même classe, afin de faciliter l'identification d'un centre de tri par un ménage.

b) L'organisation de nos packages avec le logiciel Eclipse

Nous avons créé un projet Java nommé `Projet`. Puis, dans le dossier `src`, nous avons créé six packages :

1. `com.cytech.java.projet`,
2. `com.cytech.java.projet.Acteur`,
3. `com.cytech.java.projet.Observable`,
4. `com.cytech.java.projet.Poubelle`,
5. `com.cytech.java.projet.View`,
6. et `com.cytech.java.projet.Utilisateur`.

Le package `com.cytech.java.projet.Acteur` contient les classes « CentreTri », « Commerce », « Contrat » et « Produit ». Les deux classes d'énumération « Couleur » et « TypeDechet » sont également contenues dans ce package, au même titre que la classe « StaticVar ».

Le package `com.cytech.java.projet.projet` contient la classe « MainApp » qui va nous permettre d'exécuter l'IHM que nous avons conçue. Chacune de nos classes d'affichages vont contenir une référence à la classe « MainApp ».

Le package `com.cytech.java.projet.Poubelle` contient les classes relatives aux poubelles, c'est-à-dire « `PoubelleBleue` », « `PoubelleIntelligente` », « `PoubelleJaune` », « `PoubelleNoire` » et « `PoubelleVerte` ».

Le package `com.cytech.java.projet.Utilisateur` contient les classes « `BonReduction` », « `Compte` », « `Dechet` » et « `Depot` ».

Le package `com.cytech.java.projet.Observable` contient six classes : « `BonReductionObservable` », « `CommerceObservable` », « `ContratObservable` », « `DechetObservable` », « `DepotObservable` » et « `PoubelleIntelligenteObservable` ». Ces classes vont permettre de transformer les objets des classes correspondantes en éléments observables. Par exemple, la classe « `PoubelleIntelligenteObservable` » va permettre de représenter une poubelle intelligente en utilisant des propriétés observables. Ces propriétés observables vont nous aider à faire la liaison entre les différents éléments graphiques de notre IHM et les valeurs prises pour chaque attribut de la poubelle intelligente. Cela va permettre de mettre à jour automatiquement l'IHM lorsque la valeur d'un attribut est modifiée. Pour rendre un attribut observable, nous créons une variable de type `String` pour chaque attribut, afin d'avoir une liste d'éléments observables dans les classes contrôleurs. Ces listes d'éléments observables pourront donc être affichées.

Le package `com.cytech.java.projet.View` contient toutes les pages FXML avec leurs contrôleurs. Les fichiers FXML vont venir décrire la structure et le graphisme de chaque scène de notre IHM ainsi que les propriétés graphiques de chaque composant de ces scènes. Les contrôleurs de ces fichiers vont permettre de manipuler les différents composants définis dans le fichier FXML, d'y accéder et de modifier leur contenu en fonction des instructions de l'utilisateur. La classe « `FenetreRacineController` » ainsi que le fichier `FenetreRacine.fxml` vont nous permettre de définir le menu situé en haut de notre interface graphique. Ce menu est composé de deux onglets : Fichier et Aide. L'onglet Fichier va permettre à l'utilisateur, quel qu'il soit, de sauvegarder ses données, de retourner à l'écran de connexion ou encore de sauvegarder ses données avant de se déconnecter.

II. L'interface pour le ménage

a) Description des différentes actions réalisables via l'interface du ménage

Tout d'abord, le ménage doit pouvoir s'inscrire ou se connecter à l'interface adaptée. Lors de l'exécution de notre IHM, l'utilisateur arrive sur la page d'accueil, qui va lui permettre de s'identifier en tant que ménage. Puis il va devoir décider s'il souhaite s'inscrire ou se connecter.

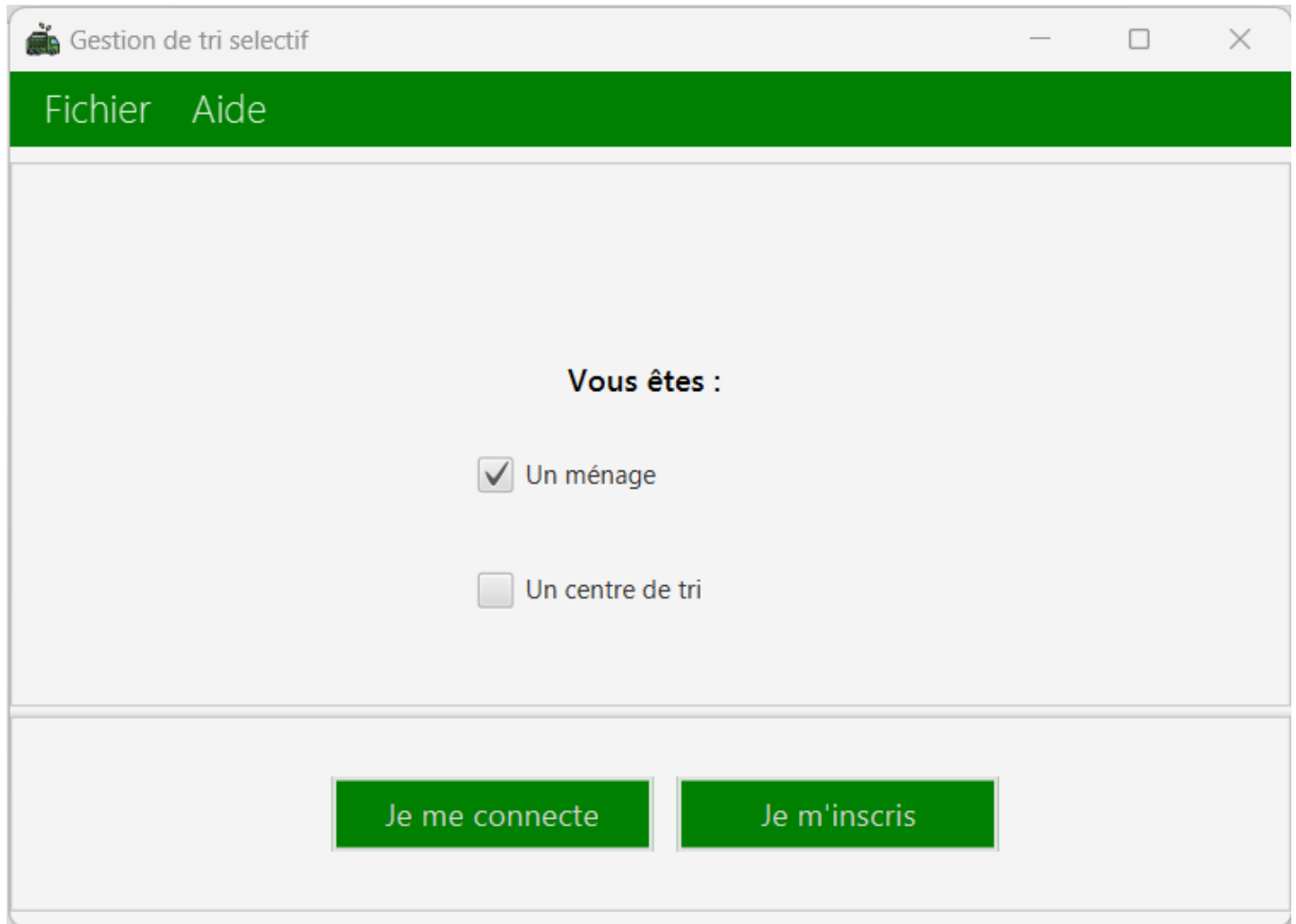
S'il décide de s'inscrire, le ménage sera redirigé vers une page d'inscription qui va lui permettre de renseigner différents champs. Si ces différents champs correspondent aux critères que l'on a fixés (champs non nuls et mot de passe composé uniquement de chiffres), alors l'utilisateur se voit attribuer un identifiant. Pour que le ménage puisse s'inscrire correctement, nous lui demandons de confirmer le mot de passe. Si les deux mots de passe rentrés ne correspondent pas alors un message d'erreur s'affiche, invitant ainsi l'utilisateur à corriger son erreur.

S'il décide de se connecter, le ménage sera redirigé vers une page de connexion. Cette page de connexion est identique pour le ménage et le centre de tri. Cependant, afin de pouvoir rediriger l'utilisateur vers l'interface qui lui correspond, nous avons créé un booléen `isConnexionCompte` dans la classe « `PageConnexionController` » qui va prendre la valeur `True` si c'est un ménage qui se connecte et `False` s'il s'agit d'un centre de tri. Pour se connecter, un ménage aura besoin de son id, de son nom et du mot de passe qu'il a lui-même défini.

Une fois qu'il a pu accéder à l'interface qui lui est attribuée, le ménage peut visualiser son nombre de points de fidélité ainsi que la liste des bons d'achats dont il dispose. Tous ces éléments se trouvent sur la gauche de notre interface graphique. Il peut également convertir ses points de fidélité en bons d'achats. Pour cela, il doit disposer d'au moins 200 points. Si cette condition est remplie, alors le ménage pourra convertir 200 points en un bon d'achat de 10% sur la catégorie de produit qu'il souhaite. Puis s'il décide d'utiliser un bon d'achat, alors il devra le sélectionner dans la liste de ses bons d'achat avant de cliquer sur le bouton « Utiliser un bon d'achat ».

Depuis son interface, le ménage peut réaliser un dépôt. Pour cela, il devra renseigner l'identifiant de la poubelle dans laquelle il souhaite faire son dépôt ainsi que l'identifiant et le nom du centre de tri auquel la poubelle est rattachée. Nous avons choisi de penser que ces informations seraient indiquées sur chaque poubelle intelligente, et donc facilement lisible par le ménage. Puis, le ménage peut ajouter des déchets à son dépôt en cliquant sur le bouton « Ajouter un déchet à mon dépôt ». En cliquant sur ce dépôt, le ménage voit apparaître un pop-up qui lui permet de renseigner le type et le poids (en grammes) de son dépôt. Pour indiquer le type du déchet qu'il souhaite ajouter à son dépôt, le ménage devra sélectionner le bon type dans un menu déroulant. Les éléments de ce menu déroulant sont ceux de la classe d'énumération nommée « `TypeDechet` ». Lorsque l'utilisateur va cliquer sur le bouton « Réaliser mon dépôt », toutes les informations qu'il a renseignées concernant les déchets constituant son dépôt seront enregistrées dans l'historique de ses dépôts. Pour consulter l'historique de ses dépôts, le ménage peut cliquer sur le bouton « Consulter l'historique de mes dépôts ». En cliquant sur ce bouton, le ménage va déclencher l'apparition d'une fenêtre pop-up. Sur la gauche de cette fenêtre se trouvent les différents dépôts réalisés par le ménage. Lorsqu'il sélectionne un dépôt dans la liste, il peut afficher les différents déchets qui le composent ainsi que leurs caractéristiques. Pour afficher les dépôts ainsi que les déchets, nous utilisons dû utiliser un tableau `TableView` afin d'afficher les éléments observables (ici les dépôts et les déchets) contenus dans une liste d'éléments observables.

b) Présentation des différentes scènes de notre interface



Dans un premier temps, l'utilisateur s'identifie en tant que ménage. S'il sélectionne le bouton « Je m'inscris », il sera invité à renseigner diverses informations sur la page Inscription ménage sur laquelle il sera redirigé.

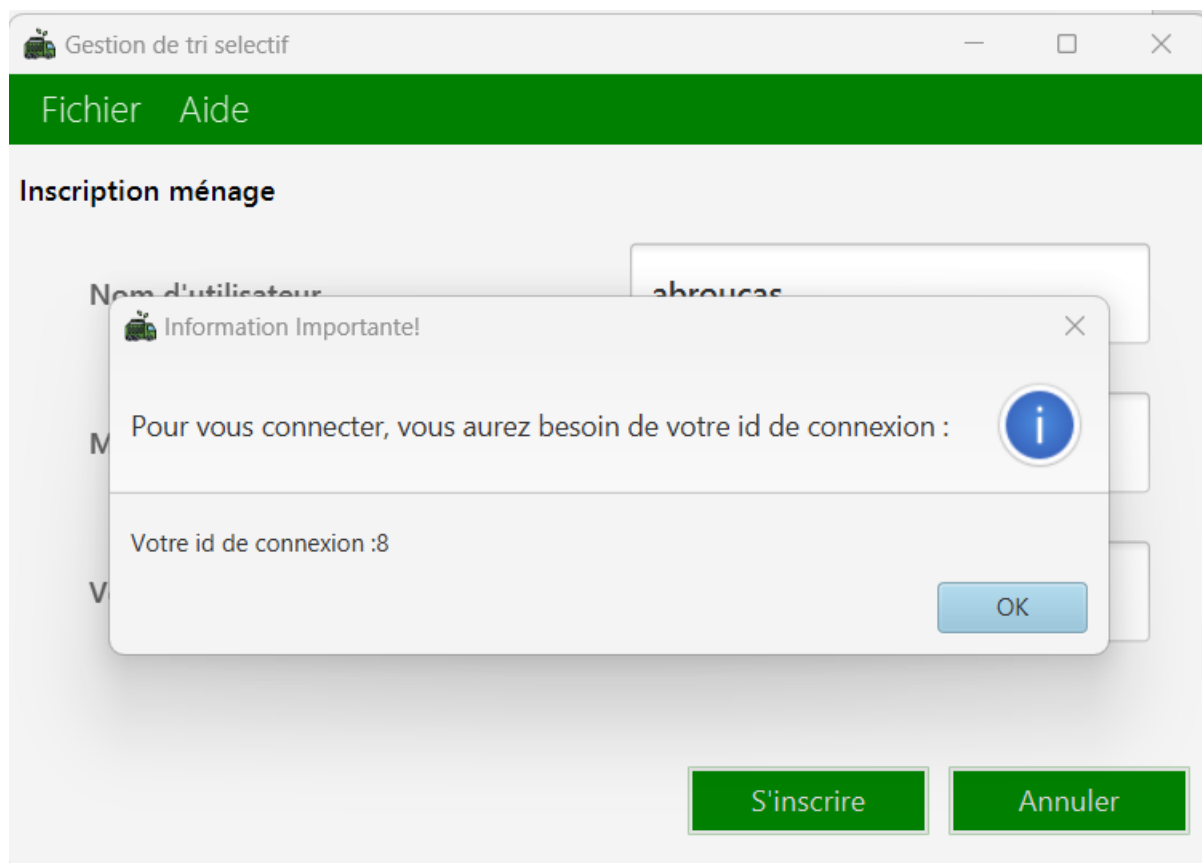
The image displays two screenshots of a Java Swing window titled "Gestion de tri selectif". The window has a green title bar with "Fichier" and "Aide" menus. The main content area is titled "Inscription ménage" and contains three text input fields: "Nom d'utilisateur", "Mot de passe", and "Vérifier mot de passe". At the bottom right, there are two green buttons: "S'inscrire" and "Annuler".

Screenshot 1 (Top): The input fields are empty.

Screenshot 2 (Bottom): The input fields contain the following text:

- Nom d'utilisateur: abroucas
- Mot de passe: 0710
- Vérifier mot de passe: 0710

Une fois que le ménage a rempli les différents champs en respectant les critères fixés et sélectionné le bouton « S'inscrire », un id de connexion lui est attribué.



Celui-ci a alors accès à l'interface lui permettant de visualiser son nombre de points ou encore ses bons d'achats. Depuis cette interface, il peut également réaliser d'autres actions telles que réaliser un dépôt ou consulter l'historique de ses dépôts.

Gestion de tri selectif

Fichier Aide

Mes Points : 0

Mes Bons d'achats

No content in table

Réaliser un dépôt :

Identifiant de la poubelle :

Identifiant et nom du centre de tri :

Ajouter un déchet à mon dépôt

Réaliser mon dépôt

Consulter l'historique de mes dépôts

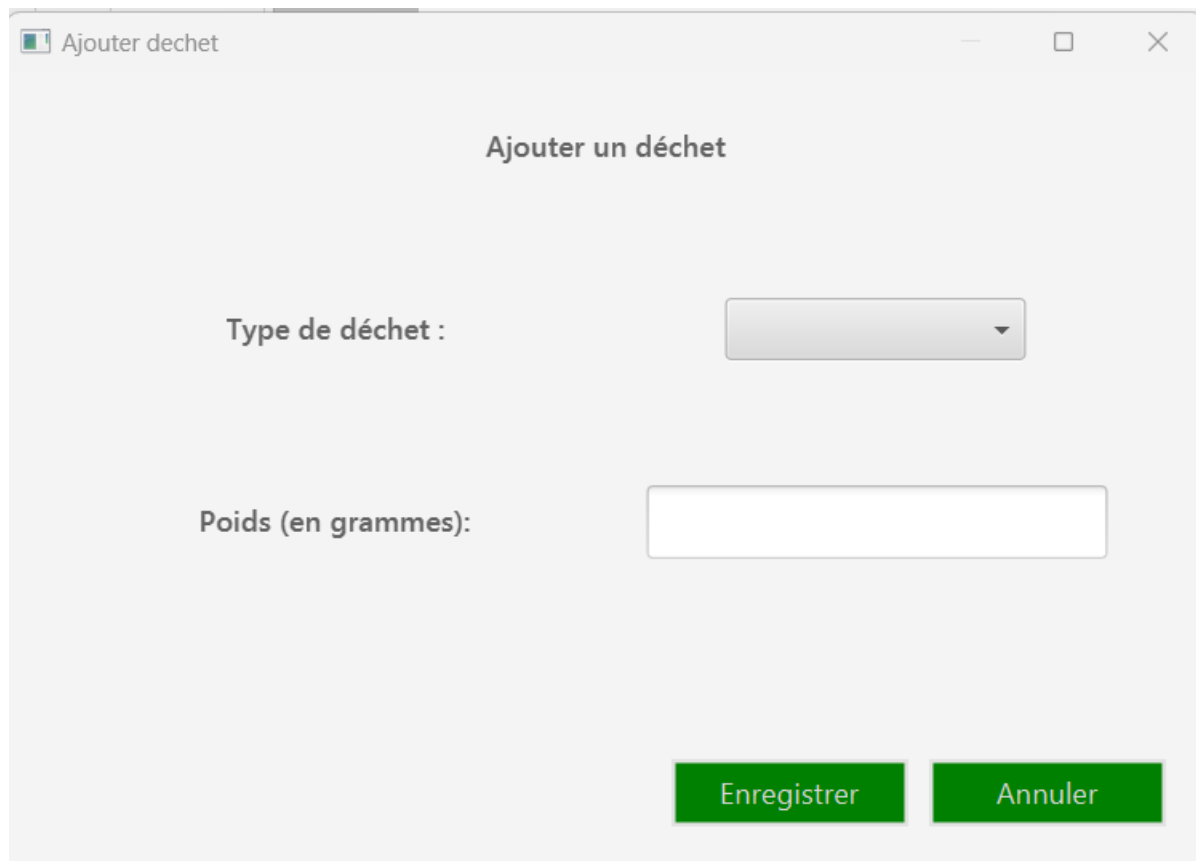
Convertir mes points de fidélité en bon d'achat :

Vous pouvez convertir vos points de fidélité en bons d'achats. 200 points vous permettent de générer un bon d'achat de 10% sur une certaine catégorie de produits.

Convertir en bon d'achat

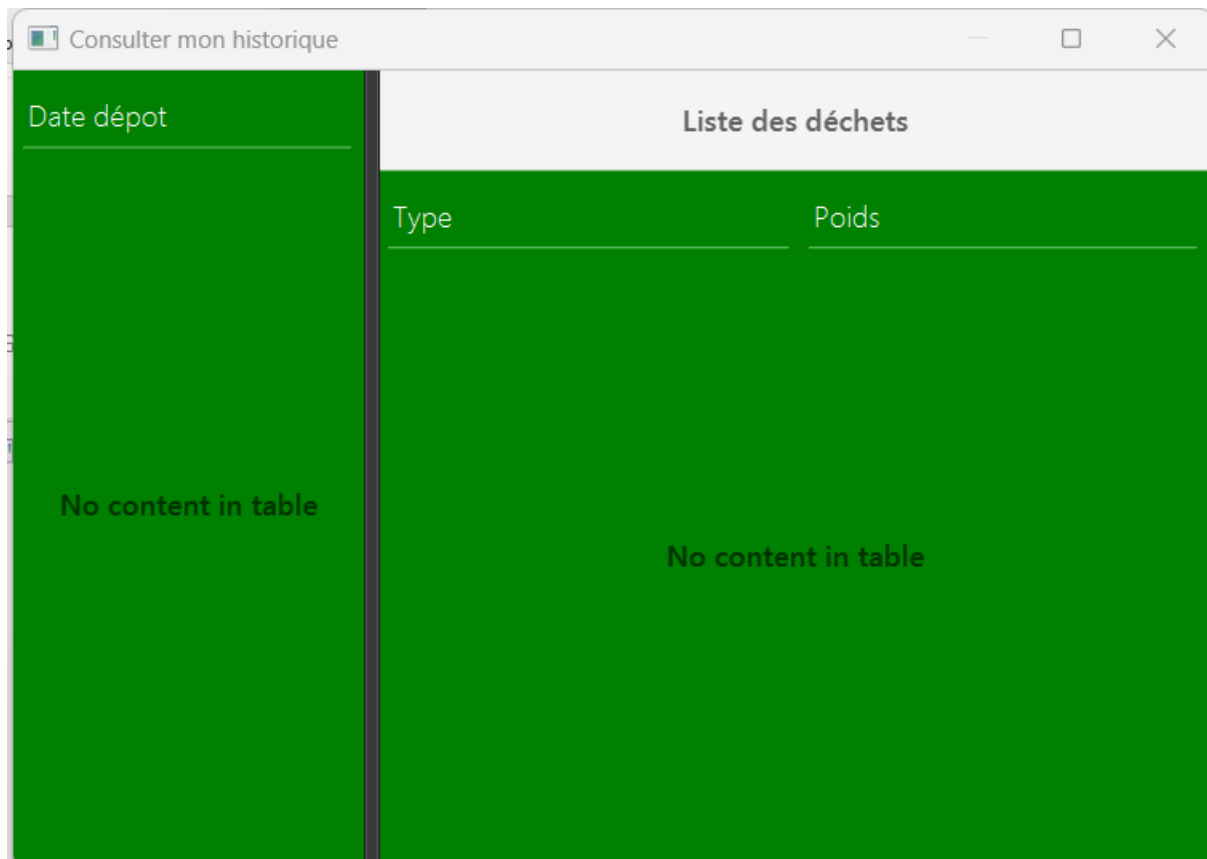
Utiliser un bon d'achat

En choisissant par exemple d'ajouter un déchet à un dépôt, il est alors invité à choisir le type de déchet sur le menu déroulant puis à entrer son poids avant d'enregistrer.

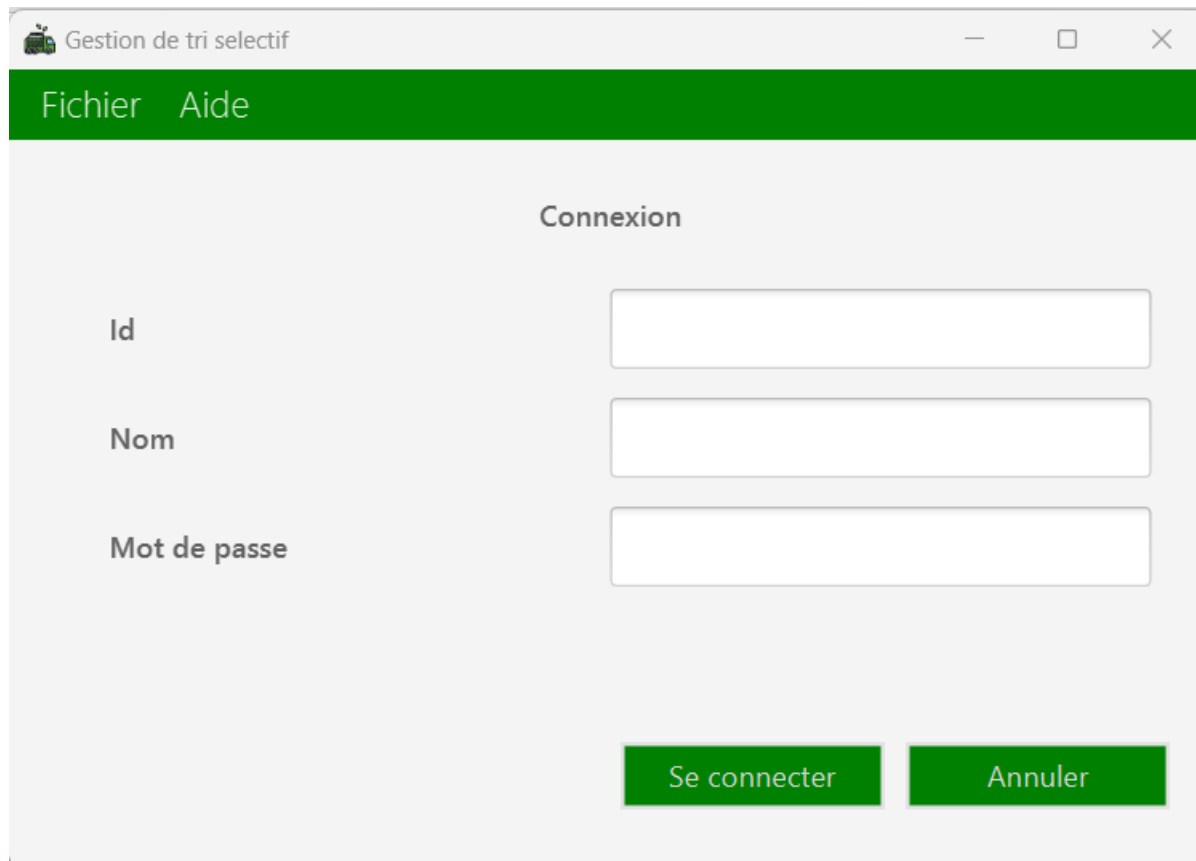


The screenshot shows a Java Swing window titled "Ajouter dechet". The window has a light gray background and standard window controls (minimize, maximize, close) in the top right corner. The title bar also contains a small icon of a trash can. The main content area is titled "Ajouter un déchet" in bold black text. Below the title, there are two labels: "Type de déchet :" and "Poids (en grammes):". The "Type de déchet :" label is positioned to the left of a gray dropdown menu with a small downward arrow. The "Poids (en grammes):" label is positioned to the left of a white text input field with a gray border. At the bottom right of the window, there are two green buttons with white text: "Enregistrer" and "Annuler".

S'il choisit de consulter l'historique de ses dépôts, un pop-up lui permettra de visualiser ses anciens dépôts et notamment de consulter la liste des déchets contenus dans celui qu'il sélectionnera.



Le ménage et le centre de tri sont redirigés vers une même page de connexion.



The screenshot shows a Java application window titled "Gestion de tri selectif". The window has a green title bar with the text "Fichier" and "Aide" on the left, and standard window control buttons (minimize, maximize, close) on the right. The main content area is light gray and contains a section titled "Connexion". Below this title, there are three input fields: "Id", "Nom", and "Mot de passe". At the bottom right of the form, there are two green buttons: "Se connecter" and "Annuler".

Connexion	
Id	<input type="text"/>
Nom	<input type="text"/>
Mot de passe	<input type="password"/>
<div><input type="button" value="Se connecter"/> <input type="button" value="Annuler"/></div>	

III. L'interface pour le centre de tri

a) Description des différentes actions réalisables via l'interface du centre de tri

Tout d'abord, le centre de tri doit pouvoir s'inscrire ou se connecter à l'interface adaptée. Lors de l'exécution de notre IHM, l'utilisateur arrive sur la page d'accueil, qui va lui permettre de s'identifier en tant que centre de tri. Puis il va devoir décider s'il souhaite s'inscrire ou se connecter.

S'il décide de s'inscrire, le centre de tri sera redirigé vers une page d'inscription qui va lui permettre de renseigner différents champs. Si ces différents champs correspondent aux critères que l'on a fixés (champs non nuls, mot de passe composé uniquement de chiffres...), alors l'utilisateur se voit attribuer un identifiant. Pour que le centre de tri puisse s'inscrire correctement, nous lui demandons de confirmer le mot de passe. Si les deux mots de passe rentrés ne correspondent pas alors un message d'erreur s'affiche, invitant ainsi l'utilisateur à corriger son erreur.

S'il décide de se connecter, le centre de tri sera redirigé vers une page de connexion. Cette page de connexion est identique pour le ménage et le centre de tri. Cependant, afin de pouvoir rediriger l'utilisateur vers l'interface qui lui correspond, nous avons créé un booléen `isConnexionCompte` dans la classe « `PageConnexionController` » qui va prendre la valeur `True` si c'est un ménage qui se connecte et `False` s'il s'agit d'un centre de tri. Pour se connecter, un centre de tri aura besoin de son id, de son nom et du mot de passe qu'il a lui-même défini.

Une fois qu'il accède à l'interface qui lui est dédiée, le centre de tri peut choisir d'ajouter une poubelle intelligente. Pour cela, il va devoir renseigner différents champs. Pour indiquer la couleur de la poubelle qu'il souhaite ajouter, le centre de tri doit sélectionner un élément dans un menu déroulant, contenant les différentes couleurs possibles définies dans la classe d'énumération « `Couleur` ». Une fois qu'il a renseigné tous les champs demandés, le centre de tri peut cliquer sur le bouton « Ajouter la poubelle ». S'il n'y a pas d'erreur alors il recevra un message de confirmation lui indiquant que la poubelle a bien été ajoutée à la liste des poubelles. Dans le cas contraire, il recevra un message d'erreur l'invitant à modifier sa saisie.

Le centre de tri peut également choisir d'organiser une collecte de déchets en cliquant sur le bouton « Collecter tous les déchets ». S'il n'y a pas d'erreur alors il reçoit un message de confirmation lui indiquant que tous les déchets ont bien été collectés.

Le centre de tri peut également décider de générer un contrat. Pour cela, il va devoir remplir différents champs avant de cliquer sur le bouton « Générer ». Afin de définir quel commerce est lié au centre de tri par ce contrat, il doit sélectionner un commerce dans un menu déroulant qui contient tous les éléments de la liste de commerces. Afin d'éviter des erreurs inutiles, nous avons choisi de préciser le format sous lequel il fallait renseigner la date de début du contrat ainsi

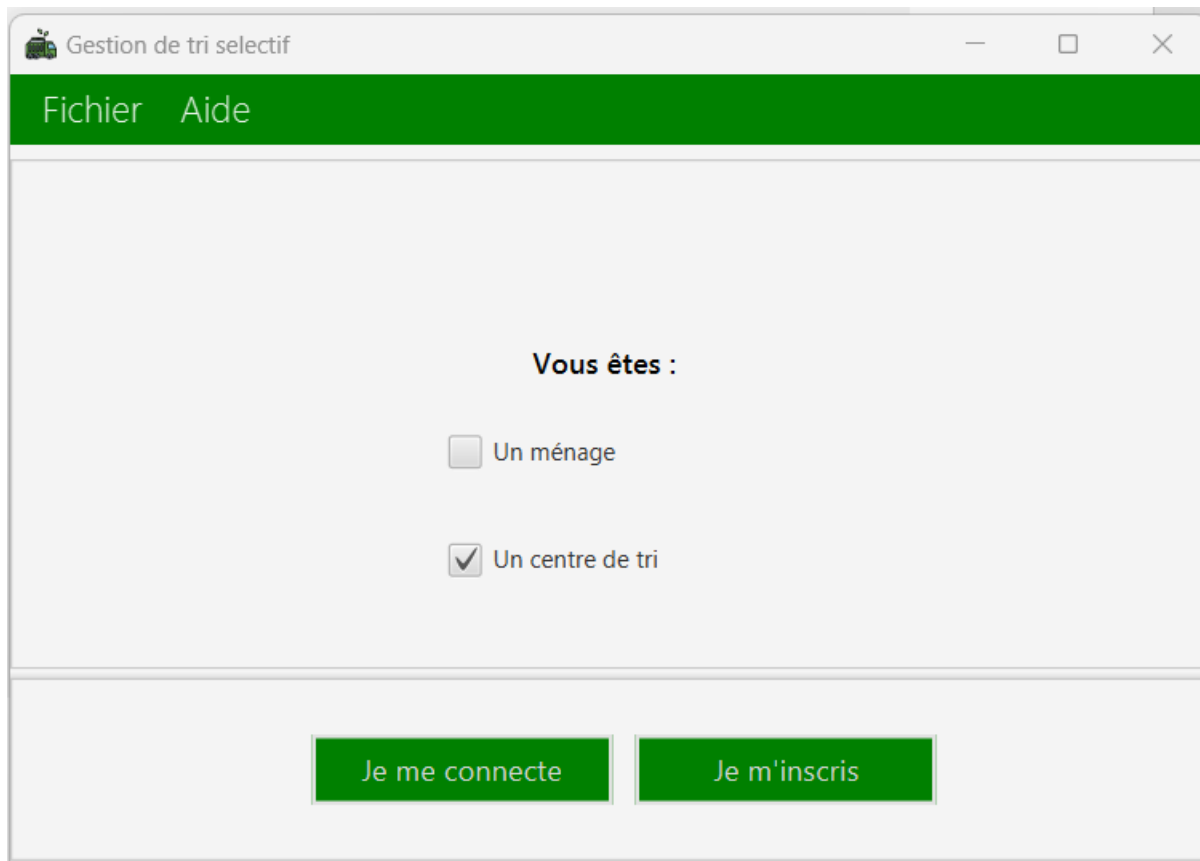
que la date de fin. Si le contrat a bien pu être généré un message de confirmation apparaît et le contrat apparaît alors dans la liste de contrat.

Le centre de tri peut aussi ajouter commerce en renseignant son nom avant de cliquer sur le bouton « Ajouter ». Si l'action a bien pu être réalisée alors un message de confirmation apparaît et le commerce apparaît dans la liste des commerces.

Le centre de tri peut aussi visualiser les statistiques associées aux dépôts qui ont été réalisés dans les poubelles intelligentes qu'il possède. Ils peuvent ainsi lire le nombre de collectes qu'ils ont organisées, le taux de déchets mis dans la bonne poubelle, le poids moyen d'un dépôt, ainsi que le nombre de points attribués par dépôt en moyenne. Il peut également visualiser, grâce à deux fenêtres pop-up, des statistiques en fonction des heures de dépôt et des types de déchets qui constituent les dits dépôts.

A gauche de l'interface graphique, se trouvent quatre boutons qui permettent de visualiser respectivement la liste des poubelles dont dispose le centre de tri, la liste des poubelles pleines, la liste des commerces qui sont partenaires du centre de tri et enfin la liste des contrats qui lient le centre de tri aux différents commerces. Lorsque l'on clique sur l'un des ces quatre boutons, une fenêtre pop-up apparaît. Ces fenêtres affichent des tableaux TableView contenant des éléments observables. Par exemple, le pop-up intitulé PopUpListePoubelle va contenir un tableau TableView composés de PoubelleObservable et chaque colonne de ce tableau contiendra les différentes valeurs des différents attributs de la classe « PoubelleObservable ». En sélectionnant une poubelle dans la liste des poubelles, il peut choisir de la retirer ou de la vider. Dans les deux cas, il recevra un message de confirmation si l'action choisie a bien pu être réalisée. Pour le pop-up intitulé PopUpListeContrat, il est possible de renouveler un contrat en le sélectionnant dans la liste puis en cliquant sur le bouton « Renouveler ». Un nouveau pop-up apparaît alors. Le centre de tri doit renseigner plusieurs champs avant de pouvoir enregistrer la modification. Cependant, le contrat après le renouvellement apparaîtra dans la liste des contrats comme étant un contrat différent du premier. Nous avons fait le choix de garder une trace de l'ancien contrat afin de faciliter la compréhension de l'historique de chaque contrat. Sur chacun des pop-ups, on trouve un bouton « Retour » qui nous ramène sur l'interface principale du centre de tri. Ces boutons ont pour but de faciliter la navigation sur notre interface.

b) Présentation des différentes scènes de notre interface



Gestion de tri selectif

Fichier Aide

Vous êtes :

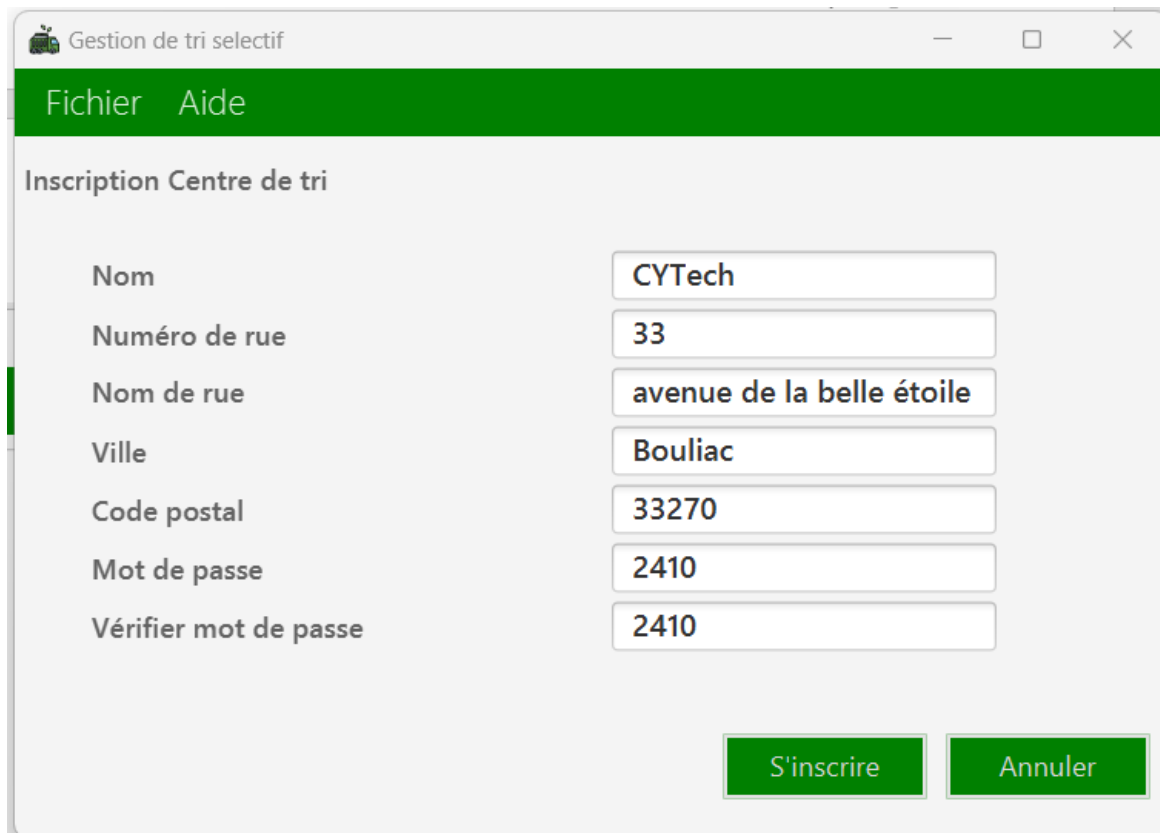
☐ Un ménage

☒ Un centre de tri

Je me connecte Je m'inscris

Dans un premier temps, l'utilisateur s'identifie en tant que centre de tri. S'il sélectionne le bouton « Je m'inscris », il sera invité à renseigner diverses informations sur la page Inscription Centre de tri sur laquelle il sera redirigé.

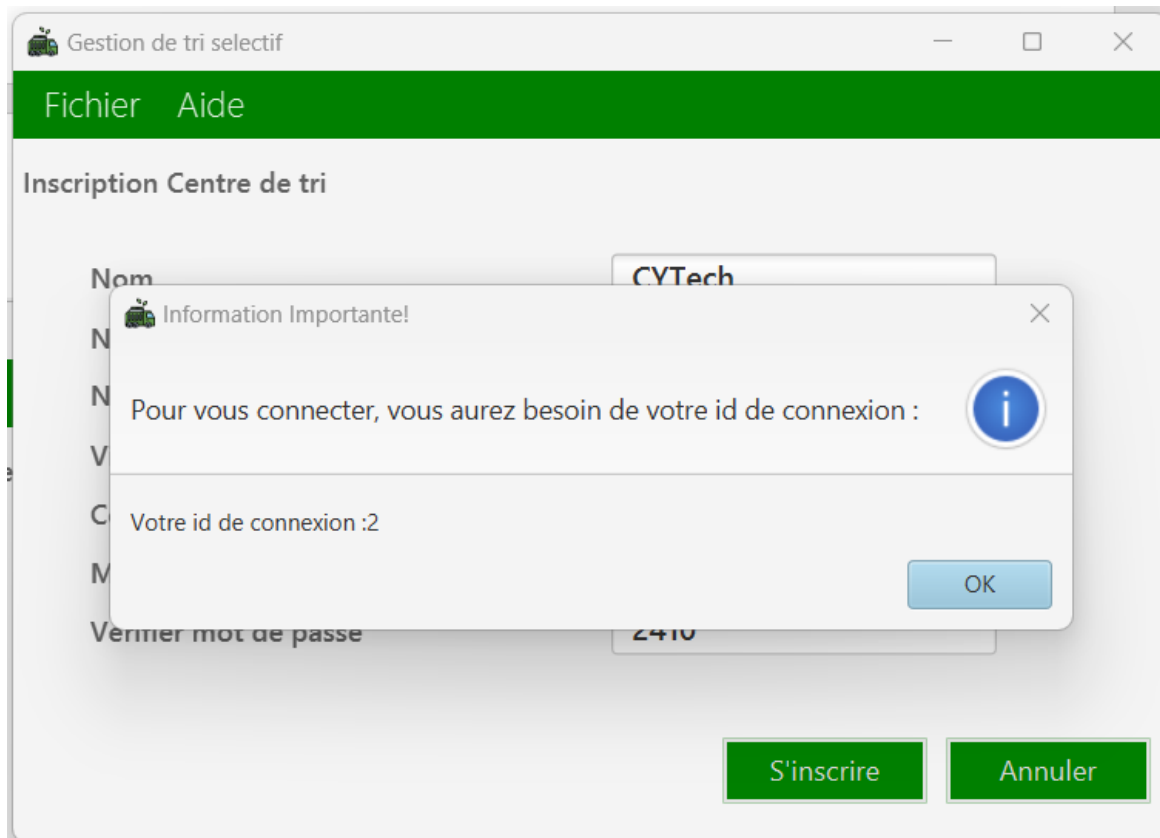
Une fois que le centre de tri a rempli les différents champs en respectant les critères fixés et sélectionné le bouton « S'inscrire », un id de connexion lui est attribué.



The screenshot shows a Java Swing window titled "Gestion de tri selectif". It has a green header bar with "Fichier" and "Aide" menus. The main content area is titled "Inscription Centre de tri". It contains a registration form with the following fields and values:

Label	Value
Nom	CYTech
Numéro de rue	33
Nom de rue	avenue de la belle étoile
Ville	Bouliac
Code postal	33270
Mot de passe	2410
Vérifier mot de passe	2410

At the bottom right, there are two green buttons: "S'inscrire" and "Annuler".



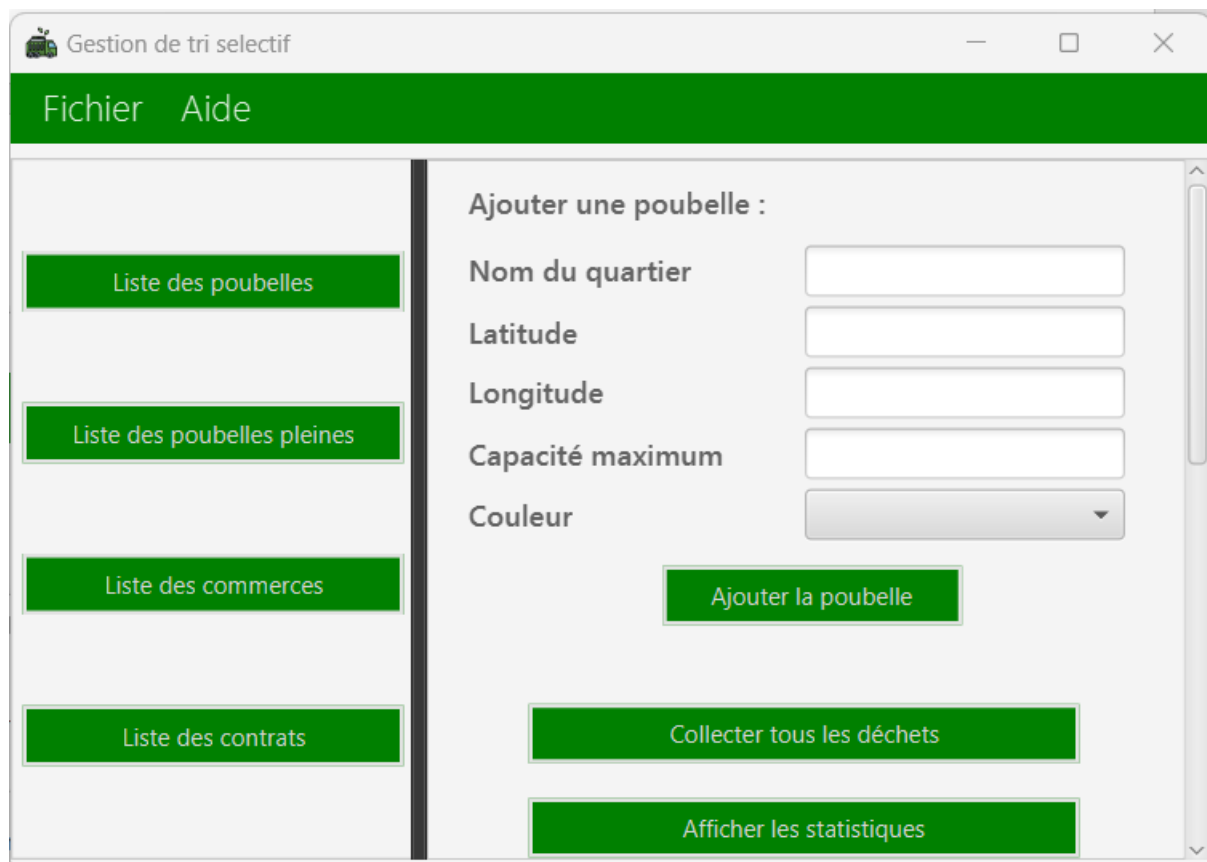
This screenshot shows the same application window as above, but with an "Information Importante!" dialog box overlaid. The dialog box has a title bar with a close button (X) and a blue information icon (i). The text inside the dialog reads:

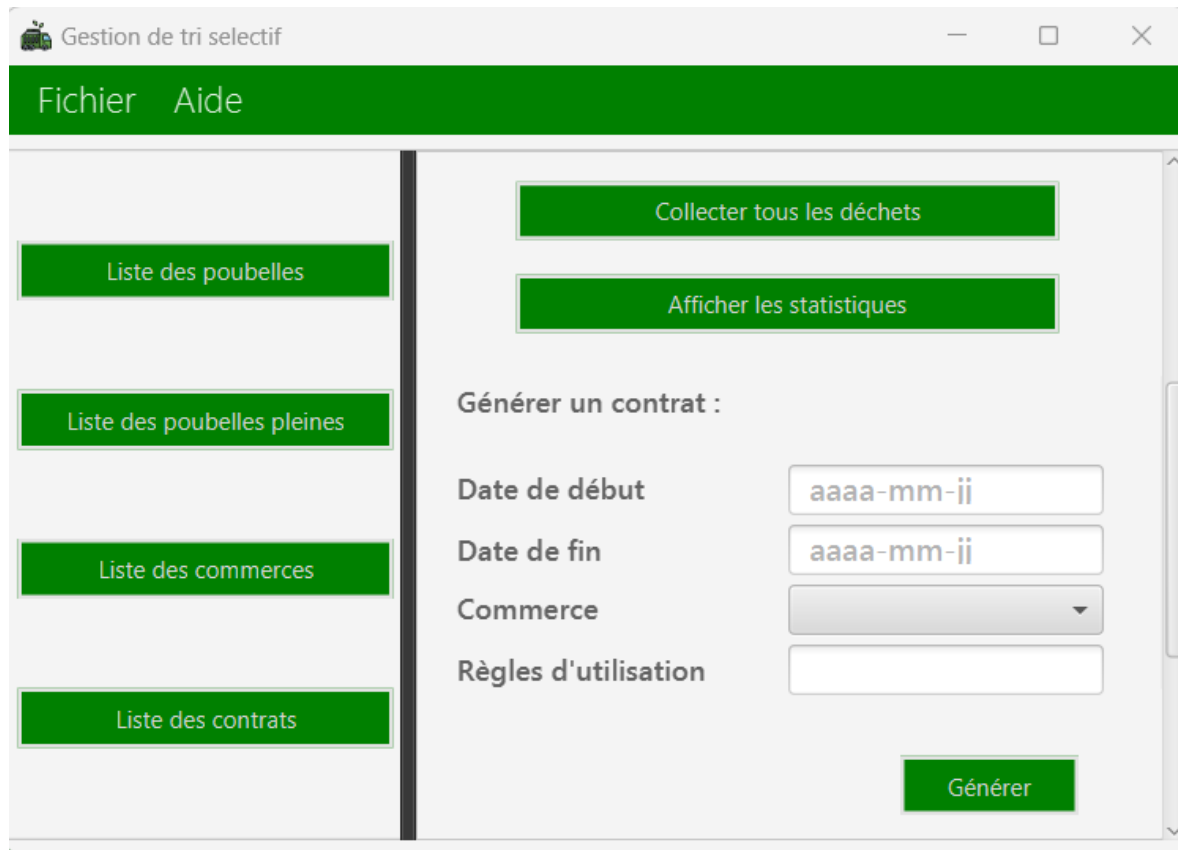
Pour vous connecter, vous aurez besoin de votre id de connexion :

Votre id de connexion :2

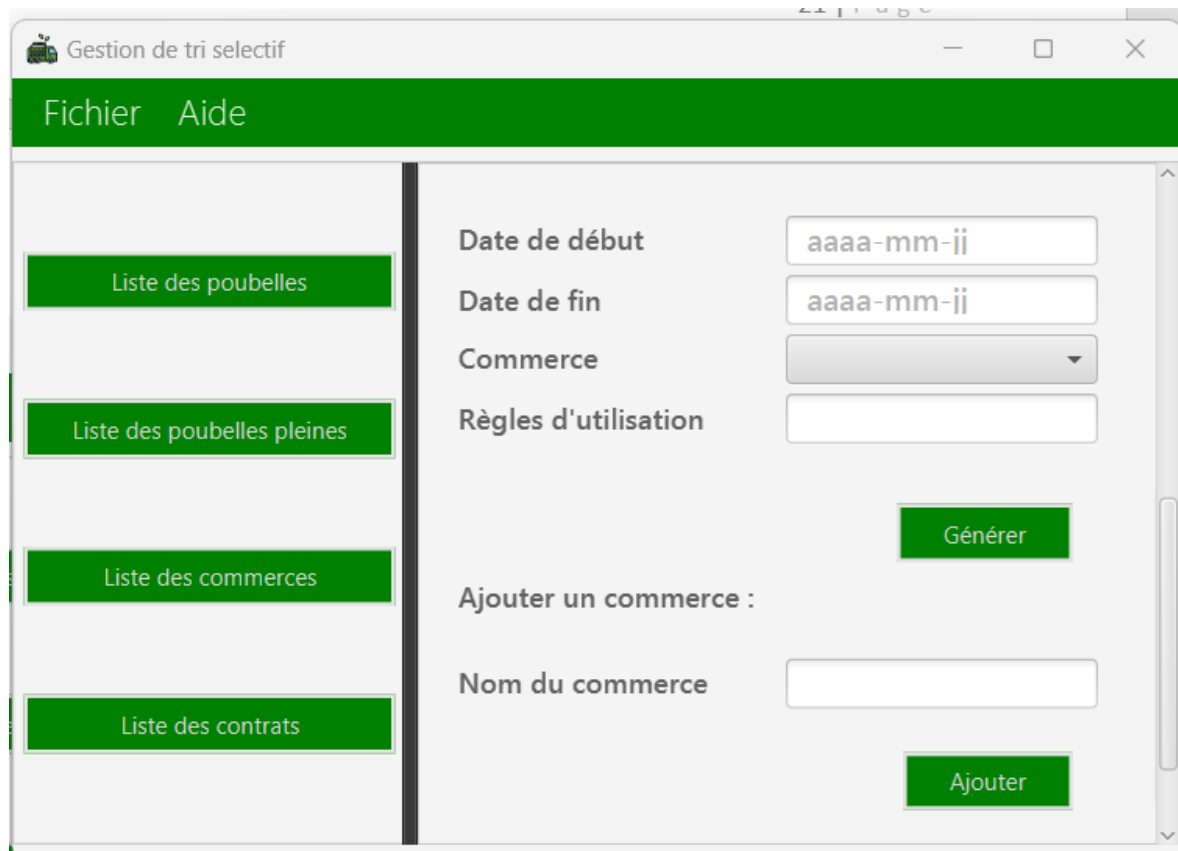
At the bottom right of the dialog is an "OK" button. The background registration form is partially visible behind the dialog.

Celui-ci a alors accès à l'interface lui permettant notamment de visualiser la liste des poubelles ou la liste des poubelles pleines par exemple. Depuis cette interface, il peut également réaliser d'autres actions telles qu'ajouter une poubelle ou un commerce, organiser une collecte de déchets ou encore générer un contrat.





The screenshot shows a Java application window titled "Gestion de tri selectif". It has a green header bar with "Fichier" and "Aide" menus. On the left is a sidebar with five green buttons: "Liste des poubelles", "Liste des poubelles pleines", "Liste des commerces", and "Liste des contrats". The main area contains a form for generating a contract. At the top are two green buttons: "Collecter tous les déchets" and "Afficher les statistiques". Below them is the section "Générer un contrat :". It includes four fields: "Date de début" and "Date de fin" (both with "aaaa-mm-jj" placeholder text), a "Commerce" dropdown menu, and a "Règles d'utilisation" text field. A green "Générer" button is at the bottom right of the form.



The screenshot shows the same application window, but the "Ajouter un commerce" section is visible. The sidebar remains the same. The main area now shows the "Date de début", "Date de fin", and "Commerce" fields, with the "Règles d'utilisation" field and "Générer" button hidden. Below these is the "Ajouter un commerce :" section, which includes a "Nom du commerce" text field and a green "Ajouter" button.

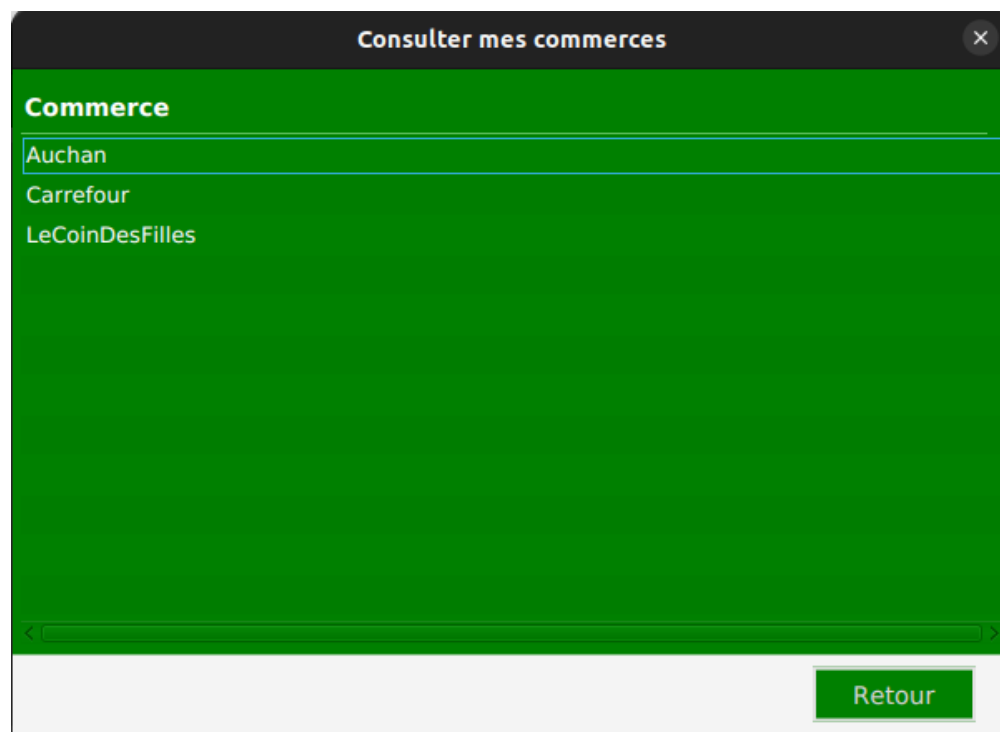
Une fenêtre pop-up apparaît si le centre de tri décide de consulter ses poubelles, ses commerces, ou encore ses contrats.

Consulter mes poubelles					
Id	Capacit...	Capacité Oc...	Longit...	Latitu...	Quartier
2	500	500	16.0	8.0	Saint-Germain-De..
3	200	100	12.0	6.0	Jean-Jaures

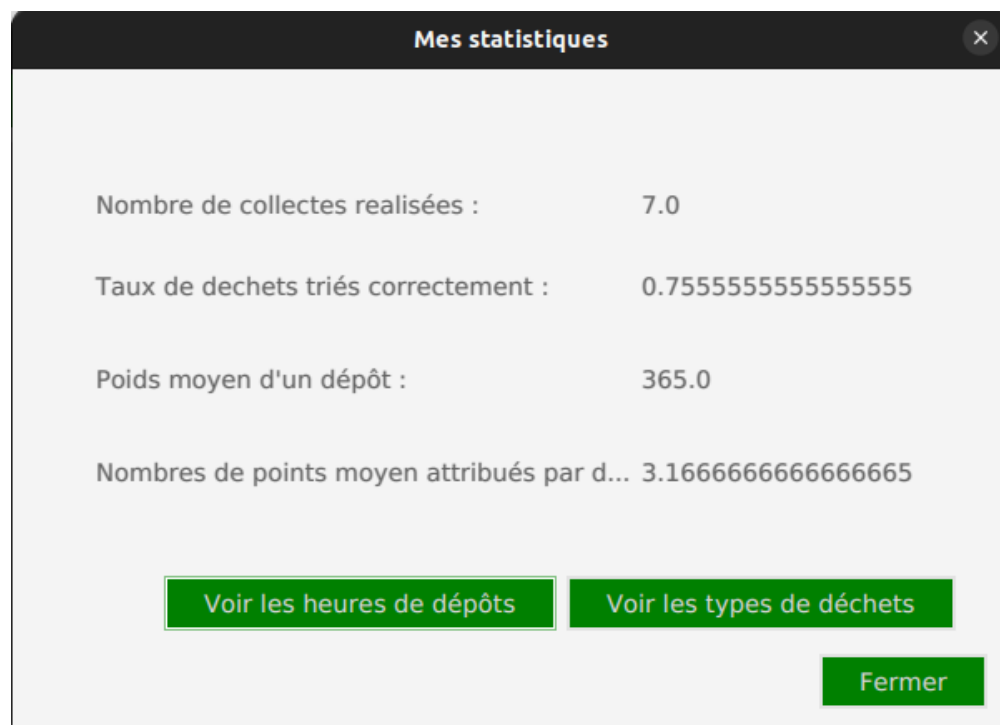
ViderRetirerRetour

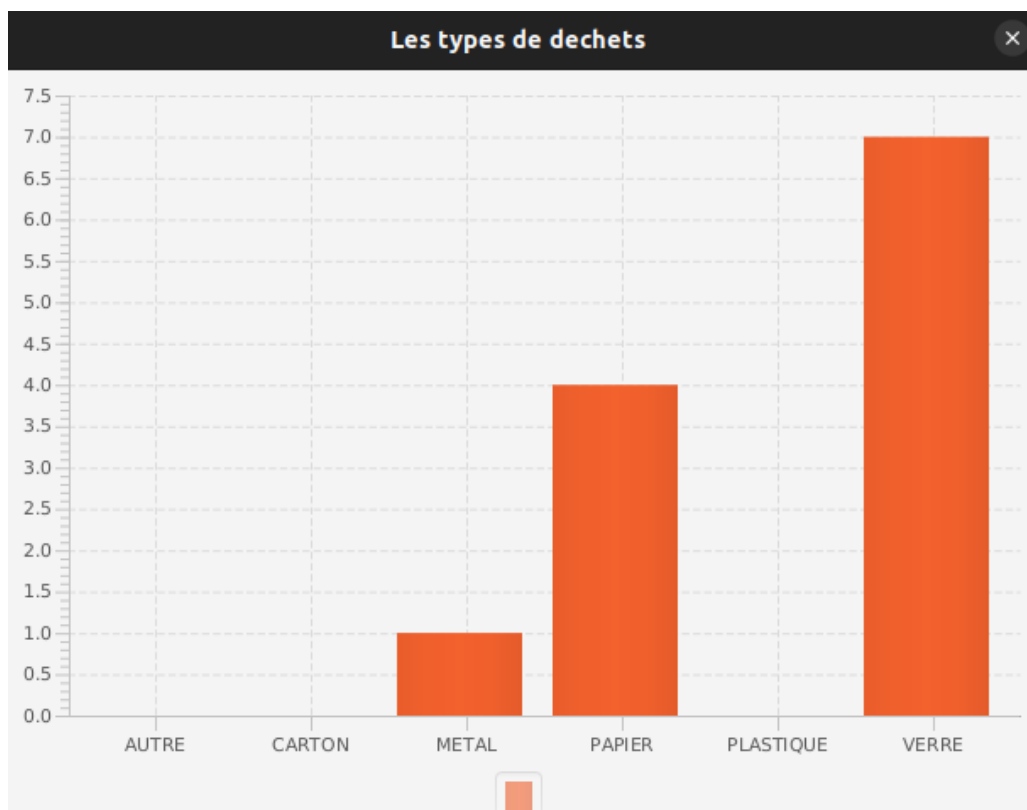
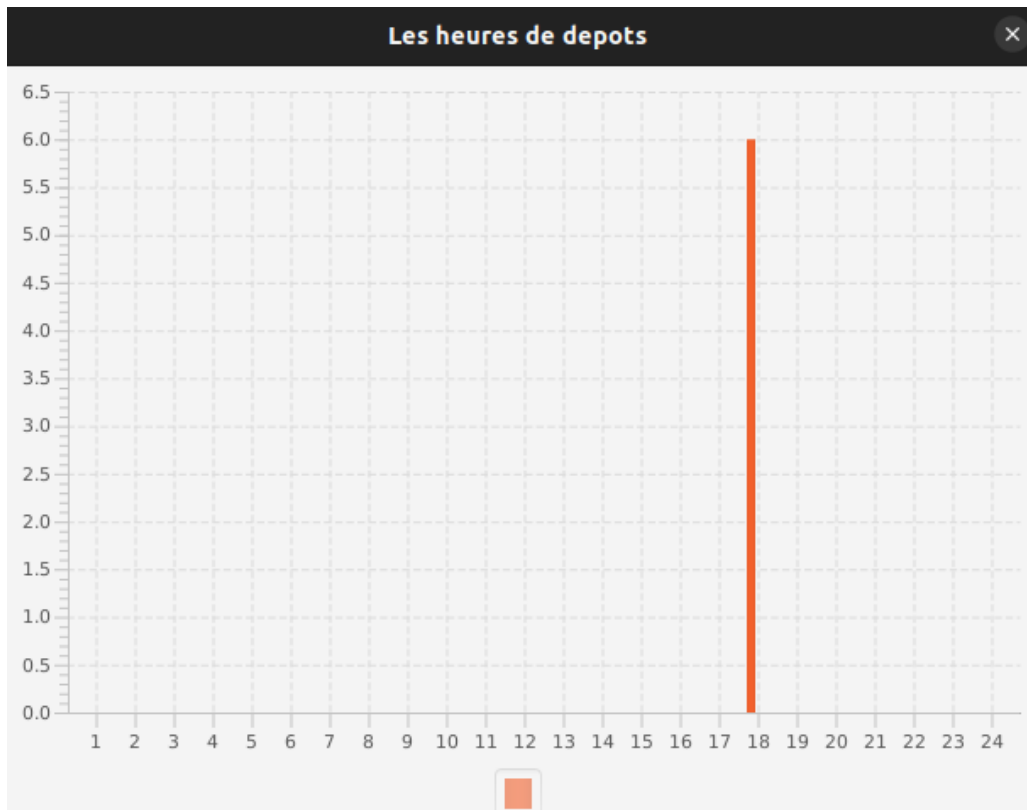
Consulter mes contrats			
Date de d...	Date d...	Règles d'utilisation	Commerce
2023-05-01	2024-05-01	Bon à utiliser dans les 2 mois suivan...	LeCoinDesFilles
2023-05-01	2025-05-01	Bon à utiliser dans les 2 mois suivan...	LeCoinDesFilles

RenouvelerRetour



Une fenêtre pop-up présente les statistiques puis permet l’affichage de deux graphiques.





Conclusion

Nous avons donc achevé la dernière partie de ce projet de tri sélectif en Java. Malgré une courte deadline, cette partie a pu être réalisée efficacement grâce à la motivation de tous les membres du groupe. Notre aptitude à travailler en groupe a été grandement renforcée ces dernières années grâce aux nombreux projets réalisés et cette dernière partie témoigne, une fois encore, des progrès individuels de chacun concernant cette compétence.

En conclusion, notre projet de groupe a été une expérience enrichissante et passionnante. Nous avons travaillé ensemble pour développer une solution complète pour la gestion des déchets dans une ville. Les trois rendus du projet ont été progressivement développés pour aboutir à une interface Homme-Machine (IHM) fonctionnelle et ergonomique.

Ce projet nous a permis de mettre en pratique les compétences acquises lors de notre formation et de développer de nouvelles compétences en travaillant en équipe sur un projet complexe. Nous sommes reconnaissants envers tous les membres de l'équipe pour leur engagement et leur contribution tout au long du projet.