

RAPPORT DU PROJET

Tracking des Livreurs



Préparé Par :

Fennassi Oussama

Encadrant :

Pr. EL AACHAK LOTFI

Pr. EI MOKHTAR EN-NAIMI

PLAN DU TRAVAIL :

- 1- Introduction
- 2- Présentation de l'environnement de développement et des technologies utilisées.
- 3- Description des fonctionnalités du système de Tracking des livreurs.
- 4- Conception.
- 5- Réalisation.
- 6- Conclusion.

1- Introduction

Dans le cadre de mes études en informatique, j'ai décidé de mettre en pratique mes connaissances en Java en développant une application desktop pour la gestion des livreurs, des commandes et des produits à livrer. Le but de ce projet était de créer une application fonctionnelle et efficace qui répond aux besoins des entreprises en matière de suivi des livraisons.

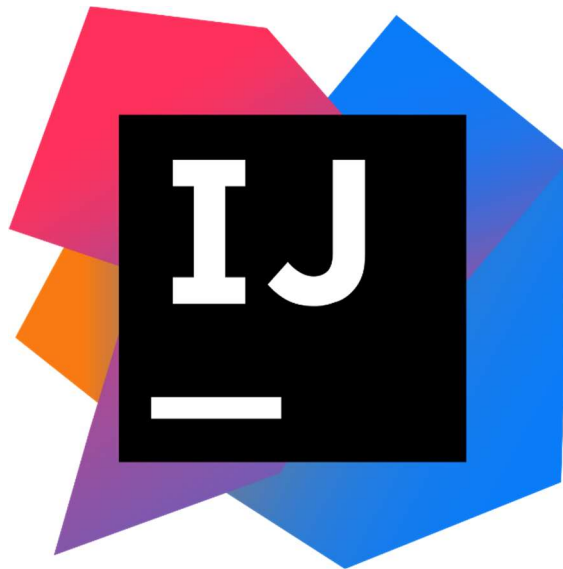
Au cours de ce projet, j'ai utilisé JavaFX pour l'interface utilisateur, JDBC pour la communication avec la base de données et MySQL pour la gestion des données. J'ai également inclus un tableau de bord générique pour une vue d'ensemble de toutes les opérations de livraison en cours.

Dans les sections suivantes, je vais décrire les fonctionnalités de l'application, l'architecture logicielle, les outils et les technologies utilisées, ainsi que les tests et les résultats obtenus. Je vais également évaluer les résultats obtenus et identifier les limites de l'application.

2- Présentation de l'environnement de développement et des technologies utilisées :

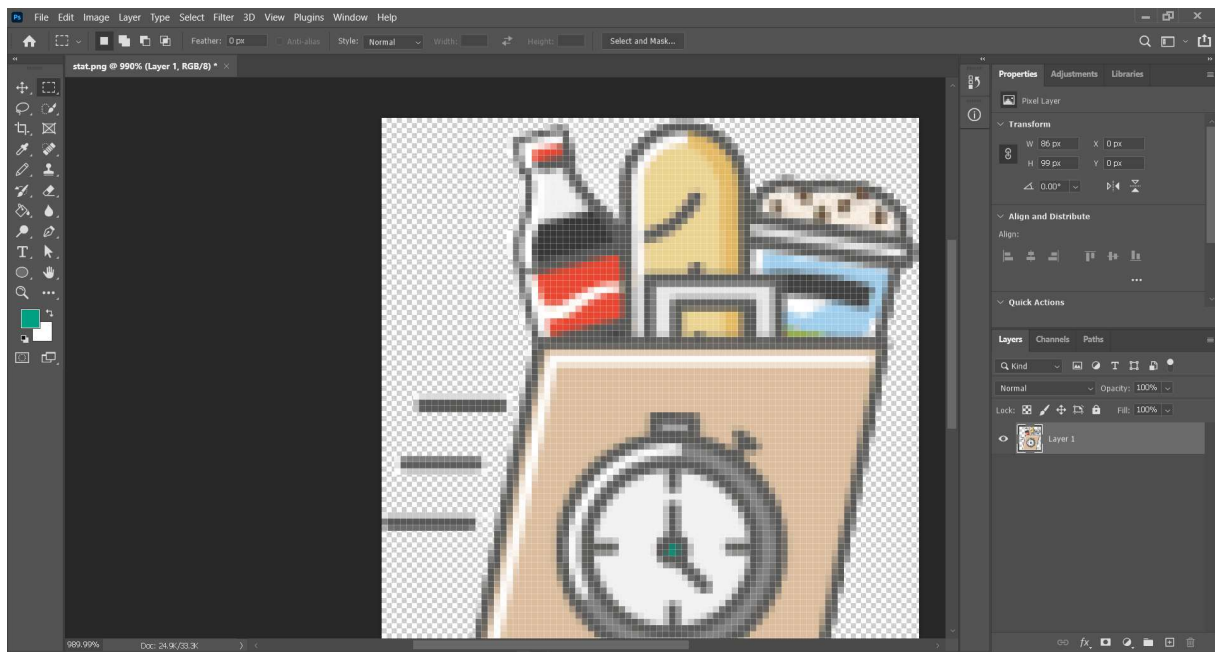
L'environnement de développement que nous avons choisi pour ce projet est IntelliJ IDEA, un environnement de développement intégré (IDE) pour Java. IntelliJ IDEA est une plateforme très populaire pour le développement Java et fournit un grand nombre de fonctionnalités pour accélérer le développement d'applications Java. Nous avons également utilisé Scene Builder, un outil de conception graphique pour JavaFX qui permet de créer des interfaces utilisateur rapidement et facilement.

Nous avons utilisé plusieurs technologies pour développer l'application de tracking des livreurs, notamment JavaFX pour l'interface utilisateur et JDBC pour la communication avec la base de données MySQL. JavaFX est un framework d'interface utilisateur Java qui permet de créer des applications de bureau riches et interactives. JDBC est une API Java qui permet aux applications Java d'accéder à des bases de données relationnelles.



Nous avons choisi MySQL comme système de gestion de base de données (SGBD) pour stocker et gérer les données de l'application. MySQL est un SGBD open-source très populaire qui offre des performances élevées, une stabilité et une fiabilité.

Pour la création des éléments graphiques de l'application, nous avons utilisé divers outils tels que Photoshop, un logiciel de retouche d'image populaire et efficace qui nous a permis de créer des images de haute qualité pour notre interface utilisateur.



En résumé, pour le développement de notre application de tracking des livreurs, nous avons utilisé l'environnement de développement IntelliJ IDEA, le framework JavaFX pour l'interface utilisateur, l'API JDBC pour la communication avec la base de données MySQL, et le SGBD MySQL pour stocker et gérer les données et photoshop pour la création des éléments graphiques.

3- Description des fonctionnalités du système de Tracking des livreurs

Authentification : Cette application démarre par une page d'authentification qui permet aux administrateurs et aux utilisateurs de se connecter. Les administrateurs ont accès à un dashboard avec des informations importantes telles que le nombre de livreurs en cours de livraison et les revenus du jour.

Gestion des produits et des livreurs : L'administrateur dispose de plusieurs fonctionnalités sur l'application. Il peut ajouter, modifier et supprimer des produits de la base de données et également ajouter, modifier et supprimer des livreurs. De plus, l'administrateur peut également voir le nombre de livreurs en cours de livraison ainsi que le revenu total et le revenu du jour grâce à un tableau de bord spécifique. Enfin, l'administrateur peut changer le statut d'un livreur de "non disponible" à "disponible" une fois qu'il a effectué une livraison.

Gestion des commandes : Cette application permet aux administrateurs de gérer les commandes des clients, de l'affectation des commandes à des livreurs jusqu'à leur livraison. Les administrateurs peuvent assigner des commandes à des livreurs en fonction de leur proximité et de leur disponibilité.

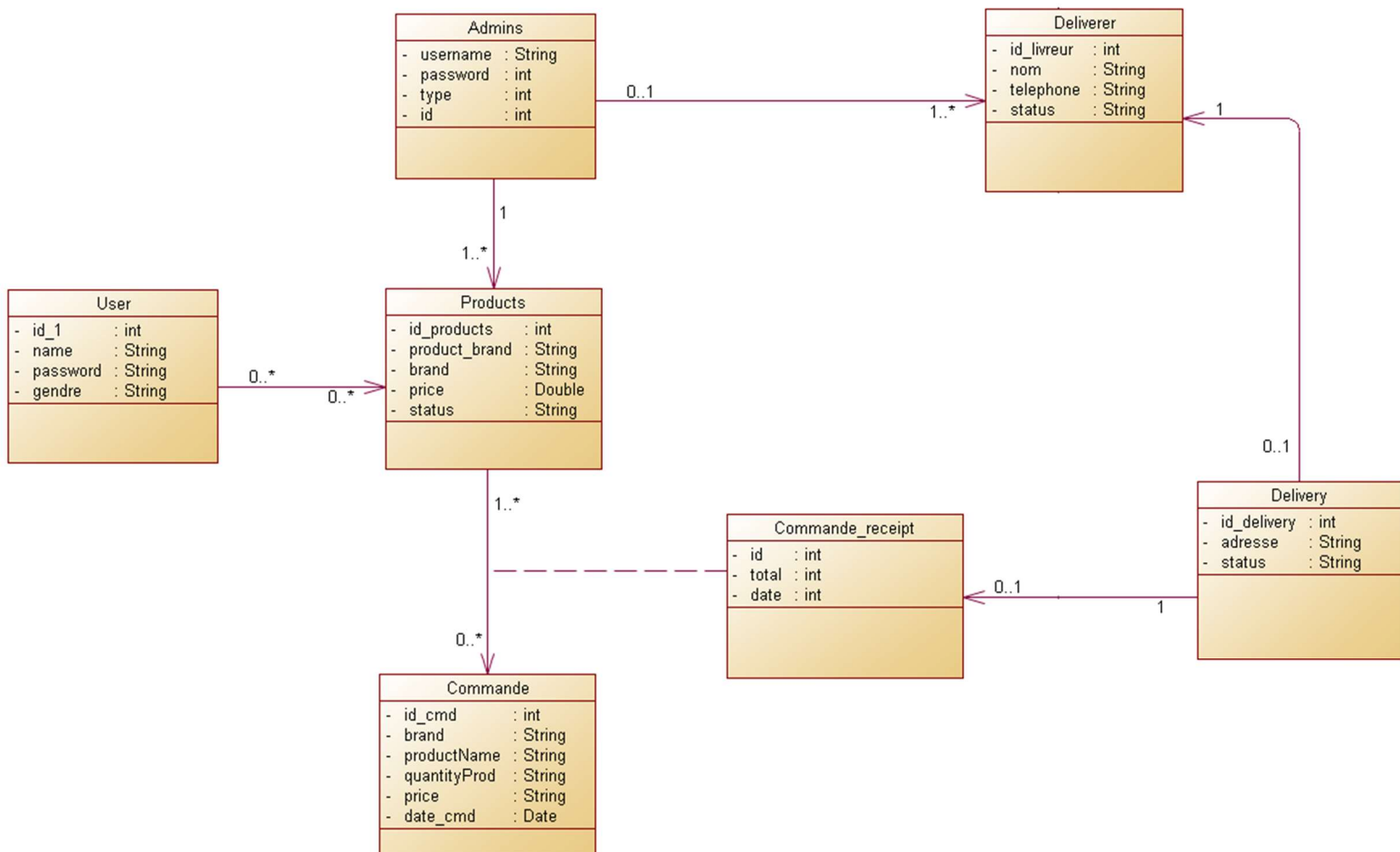
Statut de livraison : Cette application permet de suivre le statut de livraison de chaque commande. Les administrateurs peuvent savoir si une commande a été livrée ou non, et les livreurs peuvent mettre à jour le statut de livraison en temps réel.

Gestion des livreurs : Votre application permet aux administrateurs de gérer les informations des livreurs, telles que leur disponibilité. Il est également possible de changer le statut des livreurs de "indisponible" à "disponible" s'ils ont terminé leur mission.

Analyse des données : Cette application permet de collecter des données sur les performances des livreurs et sur les commandes, et de les analyser pour améliorer la gestion des livraisons. Les administrateurs peuvent ainsi utiliser ces données pour prendre des décisions éclairées et optimiser leur processus de livraison.

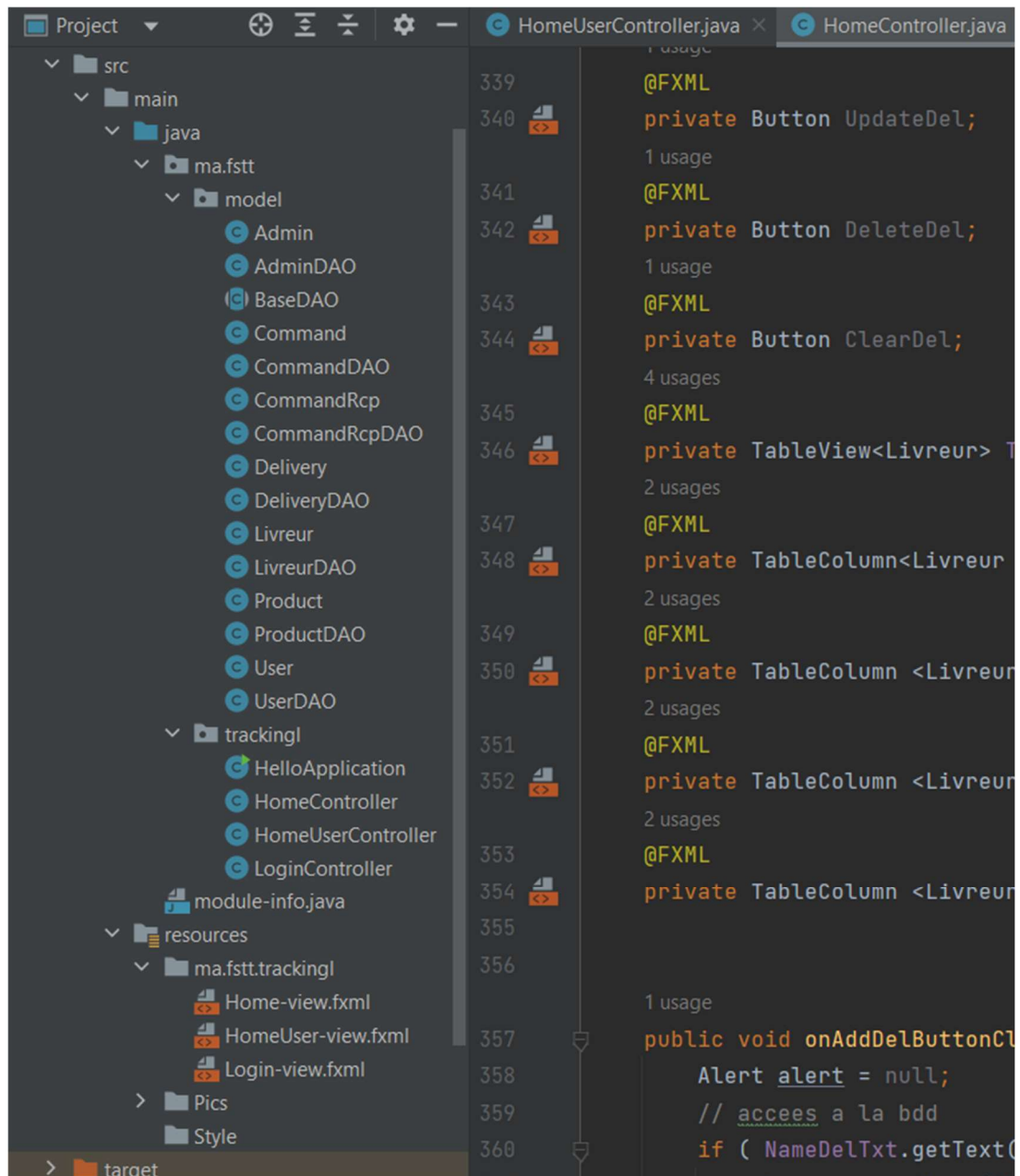
4- Conception :

a. Modélisation du système de Tracking des livreurs :



5- Réalisation :

→ Modèle (Modèle-Vue-Contrôleur)



Le modèle MVC (Modèle-Vue-Contrôleur) est un modèle d'architecture logicielle qui divise une application en trois parties distinctes : le modèle, la vue et le contrôleur. Cette architecture est utilisée pour faciliter le développement de logiciels évolutifs et réutilisables.

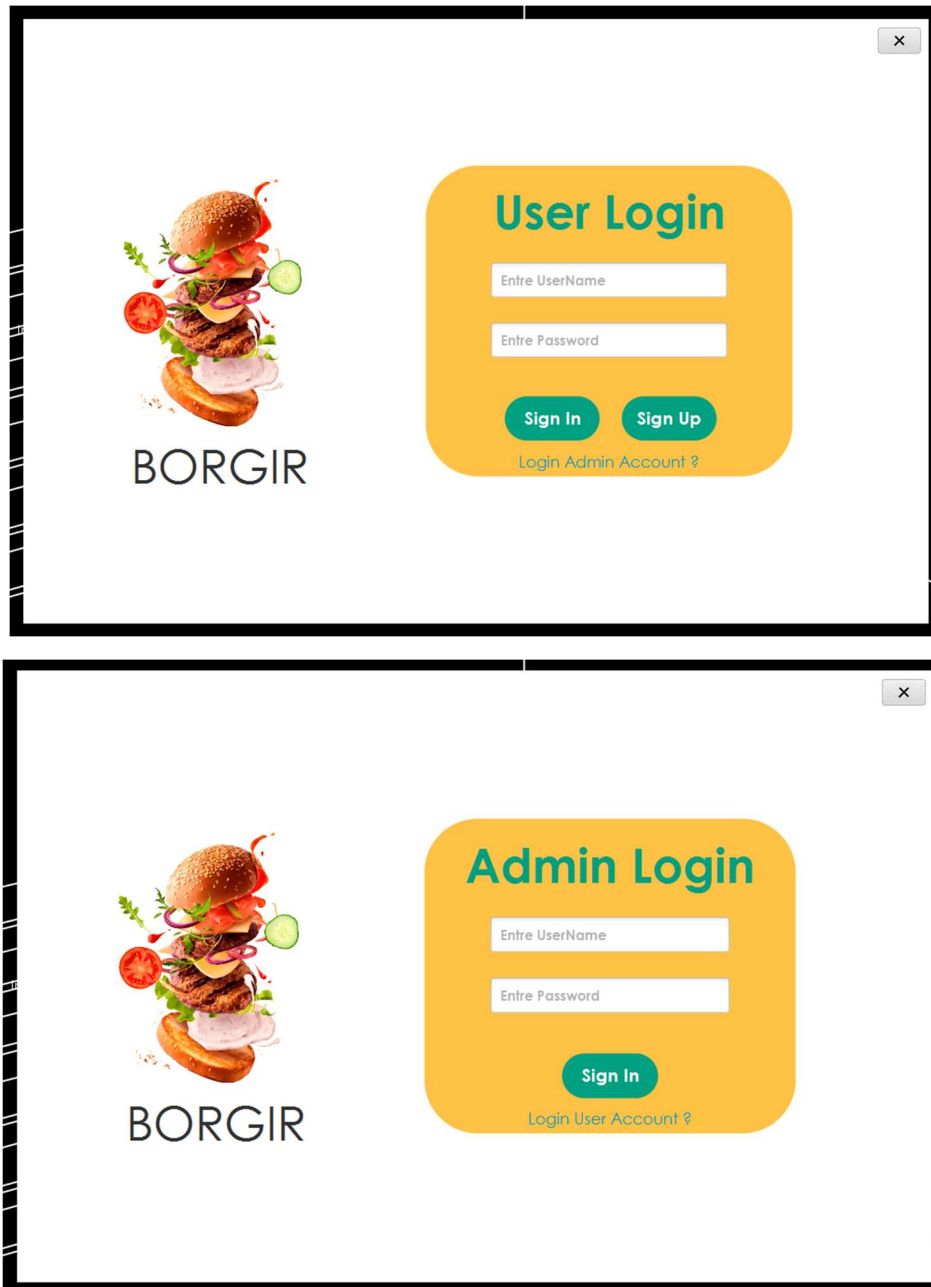
Dans IntelliJ, vous pouvez implémenter l'architecture MVC en utilisant les fonctionnalités fournies par le framework JavaFX. Voici comment les trois parties du modèle MVC sont représentées dans IntelliJ :

- ➔ Le modèle : cette partie représente la logique métier de l'application et traite les données. Dans IntelliJ, vous pouvez implémenter le modèle en créant des classes Java qui représentent les objets de l'application, tels que les produits, les commandes et les utilisateurs.
- ➔ La vue : cette partie représente l'interface utilisateur de l'application et permet aux utilisateurs d'interagir avec l'application. Dans IntelliJ, vous pouvez implémenter la vue en créant des fichiers FXML qui définissent l'apparence de votre interface utilisateur.
- ➔ Le contrôleur : cette partie gère les interactions entre le modèle et la vue et coordonne le fonctionnement global de l'application. Dans IntelliJ, vous pouvez implémenter le contrôleur en créant des classes JavaFX qui réagissent aux événements déclenchés par l'utilisateur et mettent à jour le modèle ou la vue en conséquence.

En utilisant l'architecture MVC, vous pouvez séparer les différentes parties de votre application en modules distincts et faciliter la maintenance et l'évolutivité de votre code.

→ Présentation de l'application :

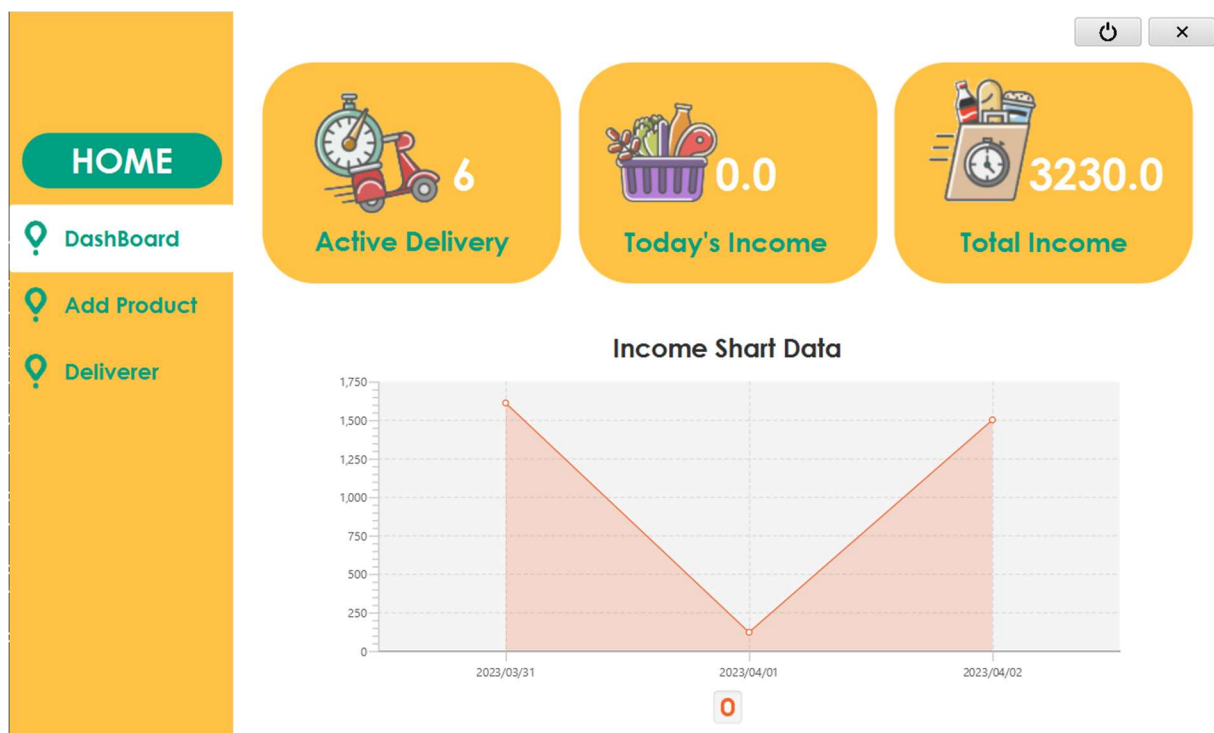
○ Login :



- 7- J'ai mis en place une interface graphique de connexion nommée "Login-view.fxml" en utilisant le Scene Builder du JavaFx. Cette interface comporte deux formulaires distincts, l'un pour les utilisateurs et l'autre pour les administrateurs. Il est possible de basculer entre les deux en cliquant sur le lien "Login User Account ?". Afin de connecter ces formulaires à la base de données, j'ai développé quatre classes : "User" et "Admin" pour stocker les informations relatives aux utilisateurs et aux administrateurs, ainsi que "UserDAO" et "AdminDAO" pour les méthodes de connexion. En outre, j'ai créé un contrôleur pour gérer les données saisies sur les formulaires.

○ Admin Home

- La page d'administration se compose de trois pages : le tableau de bord, la page "Ajouter un produit" pour la gestion des produits et la page "Livreur" pour la gestion des livreurs.



- La page d'accueil de l'administrateur contient un dashboard qui affiche le nombre de livreurs actifs, les revenus du jour et les revenus totaux, ainsi qu'un graphique qui trace l'évolution des revenus au fil des jours.

HOME

DashBoard

Add Product

Deliverer

Products

Name

Brand

Price

Status

Search

ID	Name	Brand	Status	Price
1	Pizza	PizzaHat	Available	50.0
7	Hamburger	Macdonald	Available	20.0
8	Limonad	CocaCola	Available	3.0
9	Nuggets	KFC	Available	30.0
11	Pasta	FastFood	Not Available	30.0
12	IceCream	Macdonald	Not Available	13.0

ADD
Update

Clear
Delete

→ La page "Add Product" permet la gestion des produits, où il est possible d'ajouter, modifier ou supprimer un produit en remplissant les champs de texte correspondants et en cliquant sur le bouton approprié. Les produits sont affichés sur une table view, triés par identifiant, nom, marque, statut (disponible ou non) et prix.

HOME

DashBoard

Add Product

Deliverer

Deliverer

Name

Telph

ID	Name	Telephone	Status
13	Oussama	0676877687	Available
14	Amine	0678876545	Available
15	Anass	0787009988	Available
16	Hassan	0762633567	Available
17	Zineb	0677665543	Available
18	Salma	0765678765	Available

Save
Update
Delete
Clear

Delivery

Delivered

→ La page "Deliverer" de l'interface d'administration permet la gestion des livreurs, notamment l'ajout, la modification et la suppression de ces derniers. Chaque champ de saisie correspondant est rempli et la méthode souhaitée est sélectionnée via des boutons. En outre, la page comprend un bouton "Delivered" qui permet à l'administrateur de changer l'état d'un livreur de "not available" à "available" s'il a terminé sa tâche.

○ User Home

➔ La page d'utilisateur se compose de trois pages : FOOD, la page Purchase et la page "Delivery".

The screenshot displays the 'User Home' interface. On the left is a yellow sidebar with a green 'HELLO USER' button at the top. Below it are three navigation links: 'FOOD', 'Purchase', and 'Delivery', each preceded by a location pin icon. The main content area features a green 'Products List' button above a table. The table has three columns: 'Product Name', 'Brand', and 'Price'. It lists four items: Pizza (PizzaHat, 50.0), Hamburger (Macdonald, 20.0), Limonad (CocaCola, 3.0), and Nuggets (KFC, 30.0). Below the table is an empty grid for additional products. To the right of the table is a form with input fields for 'Brand', 'Product', and 'Quantity' (set to 0), and an 'ADD' button. At the top right of the main area are two small buttons: a power icon and a close icon (X).

Product Name	Brand	Price
Pizza	PizzaHat	50.0
Hamburger	Macdonald	20.0
Limonad	CocaCola	3.0
Nuggets	KFC	30.0

➔ Sur cette page, vous trouverez un tableau affichant les produits triés par nom, marque et prix. Vous pouvez sélectionner un produit en cliquant sur sa ligne, puis choisir la quantité souhaitée avant de cliquer sur le bouton "Ajouter" pour l'ajouter à votre panier. Le produit sera alors ajouté à la page "Purchase".

HELLO USER

FOOD

Purchase

Delivery

CART

Brand	ProductName	Quantity	Price	Date commande
Pizza	PizzaHat	2	100.0	2023-04-03

Total DH: 100.0

Adresse: Tanger Boukhalef

Clr

Pay

- La page "Purchase" présente un tableau représentant le panier de tous les produits sélectionnés depuis la page "Food". L'utilisateur peut payer ses commandes en saisissant une adresse puis en cliquant sur le bouton "Payer", ce qui vide le panier et crée une livraison qui s'affiche sur la page "Delivery".

HELLO USER

FOOD

Purchase

Delivery

Delivery Stat

Delivery Adresse	Deliverer	Telephone	Delivery Status
Tanger Boukhalef	Oussama	0676877687	In progress

Deliverer

Oussama

Status

In progress

Teleph

0676877687

Delivered

- Sur cette page, un tableau présente les livraisons en cours avec les informations relatives à l'adresse de livraison, au nom du livreur, à son numéro de téléphone et au statut de la livraison. L'utilisateur a la possibilité de supprimer une livraison en cliquant sur le bouton "delivered" si la livraison est complète. Par ailleurs, l'administrateur peut également supprimer une livraison depuis sa propre page.

6- Conclusion :

En conclusion, cette application de gestion de livraison est conçue pour simplifier le processus de livraison et de gestion des produits pour les utilisateurs et les administrateurs. Grâce à cette application, les utilisateurs peuvent facilement parcourir et sélectionner des produits, les ajouter à leur panier, payer et suivre leur livraison. Les administrateurs, quant à eux, peuvent gérer les produits, les livreurs, les livraisons et suivre les revenus en temps réel. Cette application est destinée à améliorer l'expérience de livraison pour les clients et à rationaliser les processus de gestion pour les administrateurs.