

SIMULATEUR DE LIVRAISON

DOCUMENTATION



March 7, 2025

ALGORITHMIQUE ET STRUCTURE DE DONNEES

GROUPE 2

**MEMBRES DU GROUPE 2:**

MOUNRE Maxime

AYODILEKI Maléki Solange

AMANA Armel

IGBOKWE-AKOMAS Wisdom

CAPO Kalé Sabrina

SOMMAIRE

[1. Présentation du projet 2](#_Toc192888343)

[2. Fonctionnalités principales 2](#_Toc192888344)

[🔹 Gestion des commandes 2](#_Toc192888345)

[🔹 Modélisation du réseau routier 2](#_Toc192888346)

[🔹 Gestion des livreurs 3](#_Toc192888347)

[3. Structure du projet 3](#_Toc192888348)

[4. Fonctionnement global 4](#_Toc192888349)

[5. Interface Graphique 4](#_Toc192888350)

[6. Technologies utilisées 6](#_Toc192888351)

[7. Instructions d’exécution 6](#_Toc192888352)

[8. Conclusion 7](#_Toc192888353)

[9. Améliorations possibles 7](#_Toc192888354)

## 1. Présentation du projet

Ce projet est un simulateur de livraison qui modélise un réseau routier et gère un ensemble de commandes à livrer par des livreurs. Il simule le processus de prise en charge des commandes, de calcul d’itinéraires optimisés et de suivi des livraisons.

## 2. Fonctionnalités principales

### 🔹 Gestion des commandes

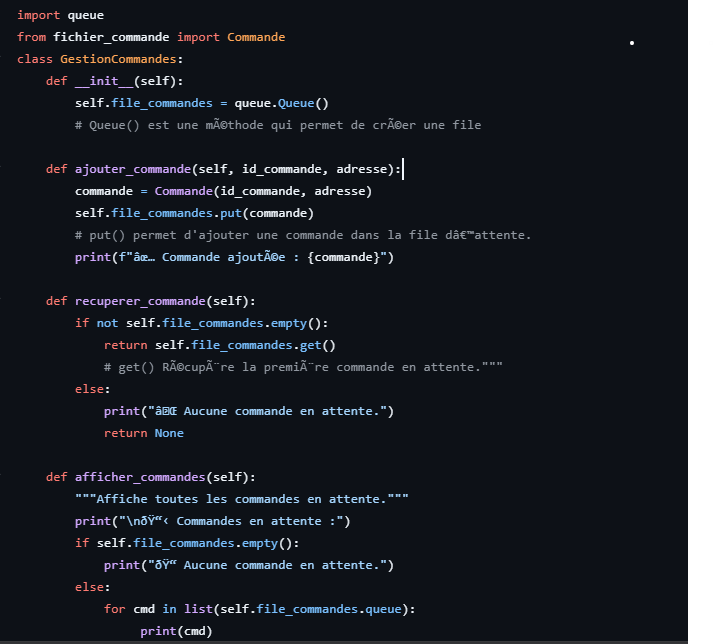
• Ajout de commandes (identifiant et adresse de livraison).

• Stockage des commandes dans une file d’attente FIFO.

• Affichage des commandes en attente.

• Récupération de la prochaine commande disponible.

Voici un extrait du code de la gestion des commandes :



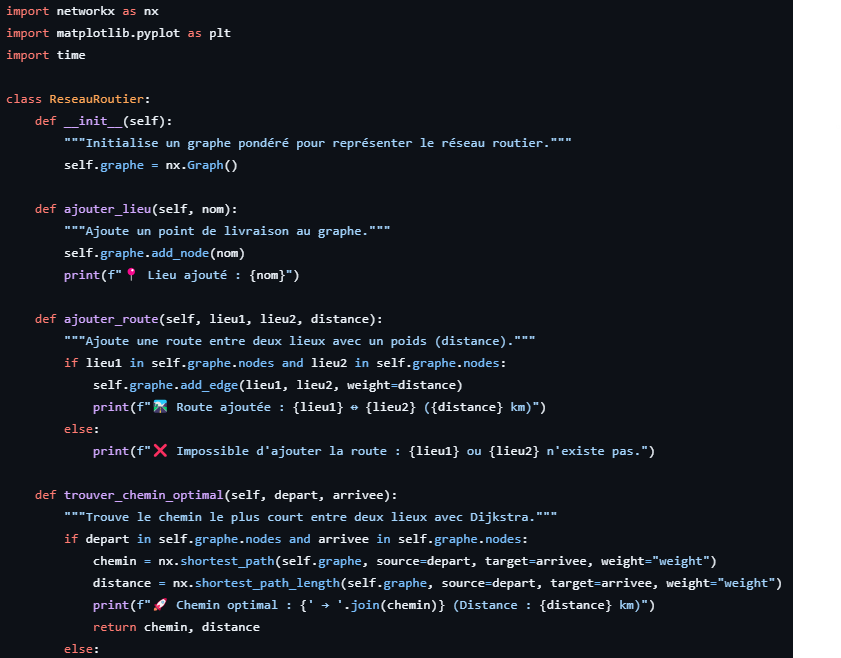
### 🔹 Modélisation du réseau routier

• Création d’un graphe pondéré représentant les lieux et les routes.

• Recherche du chemin le plus court avec l’algorithme de Dijkstra.

• Affichage graphique du réseau routier et animation des livraisons.

Voici un extrait du code de la modelisation du réseau routier :



### 🔹 Gestion des livreurs

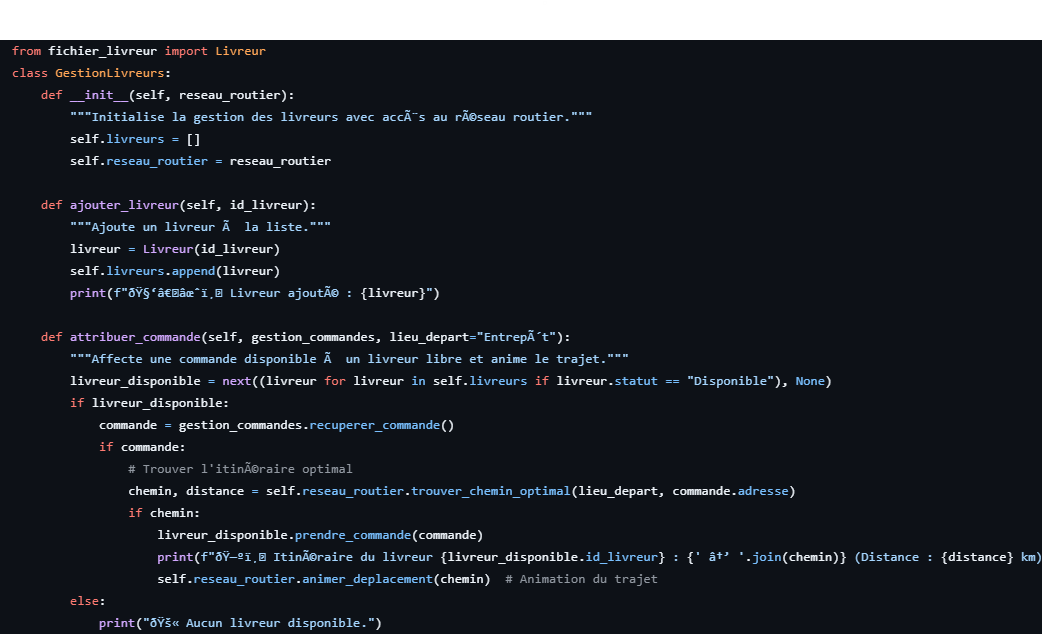
• Attribution automatique des commandes aux livreurs disponibles.

• Suivi de la position et de l’état de chaque livreur.

• Mise à jour de la commande en cours.

• Mise à jour de l’achèvement des livraisons.

Voici un extrait du code de la gestion des livreurs



## 3. Structure du projet

Organisation des classes et modules principaux :

|  |  |
| --- | --- |
| Classe | Description |
| Commande | Représente une commande avec identifiant, adresse et statut. |
| GestionCommandes | Gère la file de commandes (ajout, récupération, affichage). |
| Livreur | Représente un livreur (nom, position, commande en cours). |
| ReseauRoutier | Modélise le réseau routier sous forme de graphe pondéré. |
| GestionLivreurs | Gère la flotte de livreurs et l’attribution des commandes. |
| InterfacApplication | Implémentation de l’interface graphique |

## 4. Fonctionnement global

1. Création du réseau routier (ajout de lieux et routes entre eux).

2. Création de la flotte de livreurs.

3. Ajout de commandes à livrer.

4. Attribution des commandes aux livreurs disponibles.

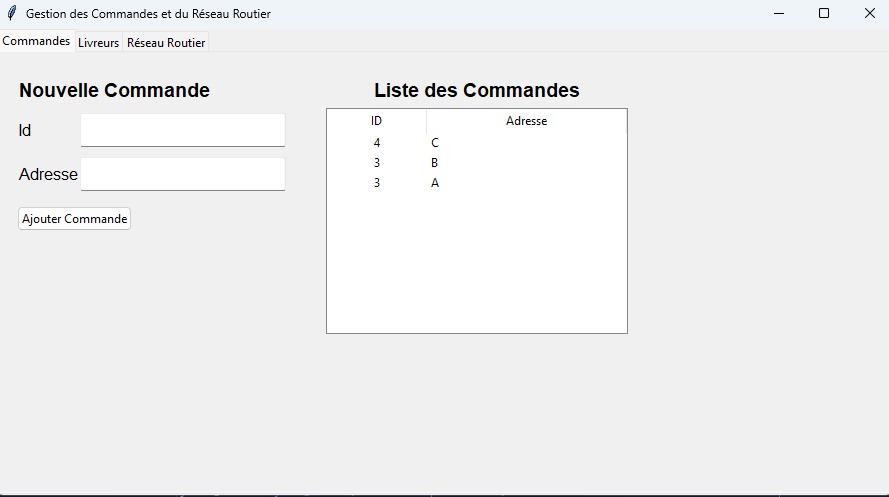
5. Calcul du chemin optimal entre le lieu actuel du livreur et la destination.

6. Déplacement des livreurs étape par étape.

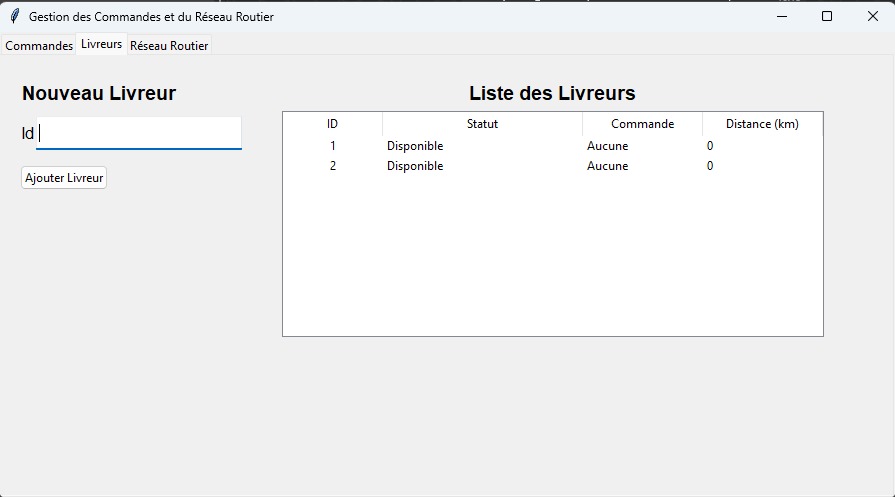
7. Mise à jour de l’état des livreurs et des commandes.

## Interface Graphique

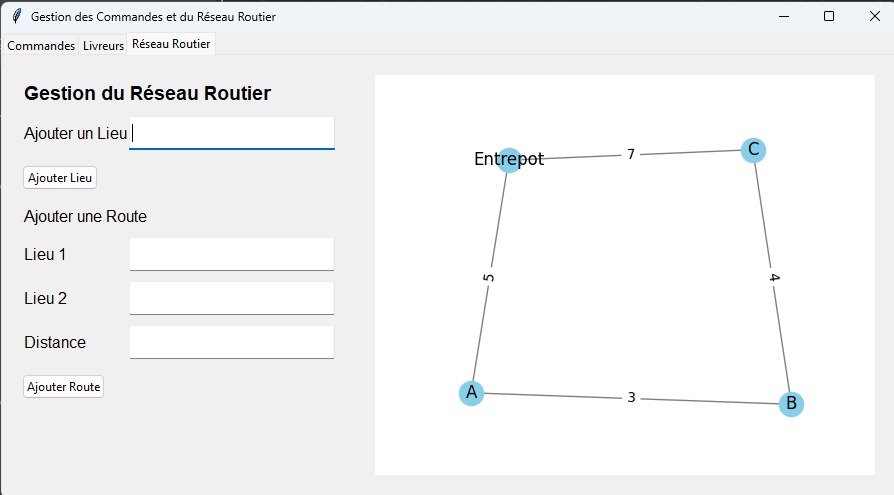
* Gestion des commandes:



* Gestion des livreurs



* Gestion du Réseau Routier



## 6. Technologies utilisées

|  |  |
| --- | --- |
| Outil / Bibliothèque | Rôle |
| Python 3.x | Langage principal. |
| queue | Gestion de la file de commandes (FIFO). |
| networkx | Modélisation et calcul de plus court chemin dans le graphe. |
| matplotlib | Affichage et animation graphique du réseau. |
| tkinter | Interface graphique |
| time | Gestion de l’animation en temps réel. |

## 7. Instructions d’exécution

Prérequis :

pip install networkx matplotlib

Commande pour lancer la simulation dans un terminal :

Python nom\_du\_ficher.py

## 8. Conclusion

Ce simulateur de livraison a permis de modéliser un processus logistique complet, de la gestion des commandes à l’optimisation des trajets des livreurs. En utilisant une file d’attente pour les commandes et un graphe routier pour les itinéraires, il met en pratique plusieurs concepts :

• Gestion d’une file de commandes.

• Optimisation des trajets de livraison avec un graphe pondéré.

• Visualisation dynamique du réseau routier.

## 9. Améliorations possibles

• Ajout d’une interface graphique complète (tkinter ou customtkinter).

• Gestion des pannes de livreurs.

• Gestion de la capacité de charge par livreur.

• Priorisation des commandes urgentes.

• Enregistrement de l’historique des livraisons.