

REPUBLIQUE DU CAMEROUN  
Paix – Travail – Patrie

UNIVERSITÉ DE YAOUNDÉ I  
Faculté des Sciences  
Département d'Informatique  
B. P 812 Yaoundé

PUBLIC OF CAMEROON  
Peace – Work – Fatherland

UNIVERSITY OF YAOUNDÉ I  
Faculté des Sciences  
Department of Computer Science  
P.O. Box 812 Yaoundé



**RAPPORT DE PROJET**  
**THÈME: CALCUL D'UN TRAJET DE TRANSPORT**  
**PUBLIC**

*Membres du groupe*

- ★ TIENTCHEU TAKOU MAURICE DONALD  
21Q2439
- ★ ESSEME ONGBAGNAK MICHELINE AUDREY  
21L2309
- ★ HAPPI NOUWE MOREL ALBAN  
21Q2523
- ★ FAISSAL SALEH  
21Q2377
- ★ JULIE FON TSA DIANA  
21Q2356
- ★ MACHE VOUTSA STEVIE  
21T2654
- ★ TALLA DONALD RODRIGUE  
21U2344
- ★ BELA ONANA SABINE LAETICIA  
21T2410
- ★ DJIELE NKAMGA JAMES LEEMAN  
21O0444
- ★ EBENGUE DIMENE ANNICK CAROLE  
23V2654

## Sommaire

<b>1. INTRODUCTION .....</b>	<b>1</b>
<b>2. OBJECTIFS DU PROJET.....</b>	<b>2</b>
<b>3. CONCEPTION ET CHOIX D'IMPLÉMENTATION.....</b>	<b>3</b>
<b>4. DESCRIPTION DES FONCTIONNALITÉS.....</b>	<b>4</b>
<b>5. DÉTAILS DE L'IMPLÉMENTATION.....</b>	<b>5</b>
<b>6. CAS DE TESTS ET RÉSULTATS.....</b>	<b>6</b>
<b>7. CONCLUSION .....</b>	<b>7</b>
<b>8. ANNEXE : COMMANDES POUR L'EXÉCUTION DU PROGRAMME</b>	
<b>9. ANNEXE RAPPORT DE PARTICIPATION</b>	

# INTRODUCTION

L'intelligence artificielle est un outil énormément prisé de nos jours ce qui entraîne une grande utilisation de celle-ci dans la vie de tous les jours. Le processus de création de celui-ci peut être particulièrement complexe. Dans l'objectif de parfaire notre formation en ICT4D Solution Provider, il nous a été donné, nous étudiants de concevoir et de mettre sur pieds un système de calcul de trajet de transport public en ce qui concerne l'unité d'enseignement ICT 302, qui s'intitule Introduction à l'intelligence artificielle. Le présent document servira donc de rapport de fin de projet, dans lequel seront détaillés les points concernant l'objectif du projet, sa conception, les fonctionnalités qui la composent, des détails de l'implémentation ensuite les résultats après tests, et enfin les difficultés rencontrées au cours du projet.

## I. OBJECTIFS

“ITINERAIRE” est un projet qui a pour but premièrement de fournir des informations précises sur les déplacements qu'un utilisateur peut effectuer dans la ville de Yaoundé. Celui fournit des informations comme les quartiers que nous devons traverser et quel est le meilleur chemin selon le temps pour y arriver, et toutes les options en ce qui concerne les chemins à suivre pour parvenir à notre objectif.

Le second objectif concernant ce projet sera aussi de tester nos connaissances afin que nous apprenants puissions aisément exposer ce que nous avons au cours de cette unité d'enseignement, de façon pratique.

## II. CONCEPTION ET CHOIX D'IMPLÉMENTATION

### **PRESENTATION DE L'ALGORITHME UTILISER**

Notre algorithme est simple, nous enregistrons des stations et des liens entre différentes stations ainsi que la distance en kilomètre entre ces stations, nous avons maintenant une sorte de graphe virtuelle entre des stations et distances en kilomètres entre ces derniers. Maintenant entre deux chemins on peut déterminer le chemin le plus court en fonction de la distance en kilomètre et le nombre de stations par lequel on peut passer.

La conception de notre programme s'est effectuée en deux phases. Nous avons premièrement identifier les données qui nous seront utilisées dans notre conception, ensuite nous avons échafaudé un principe de calcul de chemin optimal, et de calcul de tous les chemins possibles afin d'arriver à nos différents résultats.

Les différentes données que nous utiliserons pendant la conception de notre programme sont:

- La station de départ
- La station d'arrivée
- Mode de transport
- La durée du trajet entre deux stations
- Les stations qui sont reliées entre elles.

La conception des différents trajets s'est faite en plusieurs étapes qui sont les suivantes:

- Détermination du départ et de l'arrivée parmi les quartiers dans notre base de données.
- calcul de l'arbre graphique à suivre afin d'arriver à la destination.
- calcul des distances entre toutes les stations intermédiaires dans le trajet.

Lors de notre implémentation, nous avons utilisé le système d'exploitation **Ubuntu**, pour sa robustesse. Ensuite nous avons installé le logiciel swi prolog la version suivante **SWI-Prolog version 9.2.5**.

### **III. DESCRIPTION DES FONCTIONNALITÉS**

Les fonctionnalités qui suivent permettent à l'utilisateur de planifier un trajet en fonction des stations de départ et d'arrivée, de choisir le mode de transport, et de voir les différents itinéraires possibles avec les coûts, durées et distances associés y compris les chemin direct et court. L'utilisateur peut également choisir un itinéraire optimal basé sur le coût ou la durée.

Nous avons:

### **1. Définition des stations et horaires de départ :**

- Les stations (château, emia, messa, mokolo, total\_melen, elig\_efa) sont définies.
- Les horaires de départ pour chaque station sont spécifiés.

### **2. Modes de transport avec coûts et vitesses :**

- Les modes de transport (pieds, taxi, moto) sont définis avec leur coût par kilomètre et leur vitesse en km/h.

### **3. Recherche d'itinéraires entre deux stations :**

- ``recherche_itineraire/3`` : Trouve un itinéraire entre deux stations en utilisant un prédicat auxiliaire ``trouve_chemin/4``.
- ``trouve_chemin/4`` : Recherche récursive d'un chemin entre deux stations, en explorant les stations suivantes et précédentes.

### **4. Définition des stations suivantes et précédentes :**

- Les relations entre les stations suivantes (``next_station/2``) et précédentes (``prev_station/2``) sont définies pour permettre la navigation entre elles.

### **5. Calcul du coût, de la durée et de la distance d'un trajet :**

**`calcule_cout_duree_distance(Trajet, Mode, Cout, Duree, Distance)`.**

- Trajet: Représente le trajet ou le voyage considéré.
- Mode: Représente le mode de transport utilisé, comme "voiture", "train" ou "bus".
- Cout: Une variable de sortie qui sera liée au coût total du voyage.
- Duree: Une variable de sortie qui sera liée à la durée totale du voyage en minutes.
- Distance: Une variable de sortie qui sera liée à la distance totale parcourue en kilomètres.

### **6. Distances entre les stations :**

- Les distances entre les stations sont spécifiées pour permettre le calcul des distances totales des itinéraires.

### **7. Recherche d'itinéraires optimaux :**

Ce prédicat trouve tous les itinéraires possibles entre deux stations, puis les classe en fonction du type spécifié (court ou long) pour optimiser soit le coût, soit la durée.

### **8. Comptage du nombre d'itinéraires :**

- `nombre\_itinerares/3` : Compte le nombre d'itinéraires uniques entre deux stations.

## **9. Interface utilisateur pour le calcul d'itinéraire :**

- `main/0` : Interface utilisateur principale pour saisir la station de départ, la station d'arrivée et le mode de transport, puis afficher les itinéraires possibles, permettant à l'utilisateur de choisir son itinéraire préféré.

## **10. Affichage des stations avec horaires et itinéraires :**

- `affiche\_stations\_avec\_horaires/0` : Affiche les stations avec leurs horaires de départ.
- `affiche\_horaires/1` : Affiche les horaires sous format HH:MM.
- `affiche\_stations\_intermediaires/1` : Affiche les stations intermédiaires dans l'itinéraire avec les distances entre elles.
- `affiche\_itinerares/2` : Affiche tous les itinéraires possibles sans coût, durée et distance.

## **11. Nombre d'itinéraires possibles**

Ce prédicat compte le nombre total d'itinéraires possibles entre deux stations en utilisant *findall/3* pour collecter tous les chemins uniques.

## **12. Calcul de l'heure d'arrivée**

Ce prédicat convertit l'heure de départ et calcule l'heure d'arrivée en ajoutant la durée du trajet.

## **13. Prédicat principal itineraire**

Ce prédicat gère l'interface utilisateur pour saisir la station de départ, la station d'arrivée, l'heure de départ et le mode de transport. Il utilise les prédicats précédents pour calculer et afficher l'itinéraire optimal en fonction du mode et du critère spécifié (court ou long).

## **14. Affichage des stations intermédiaires (affiche\_stations\_intermediaires/1)**

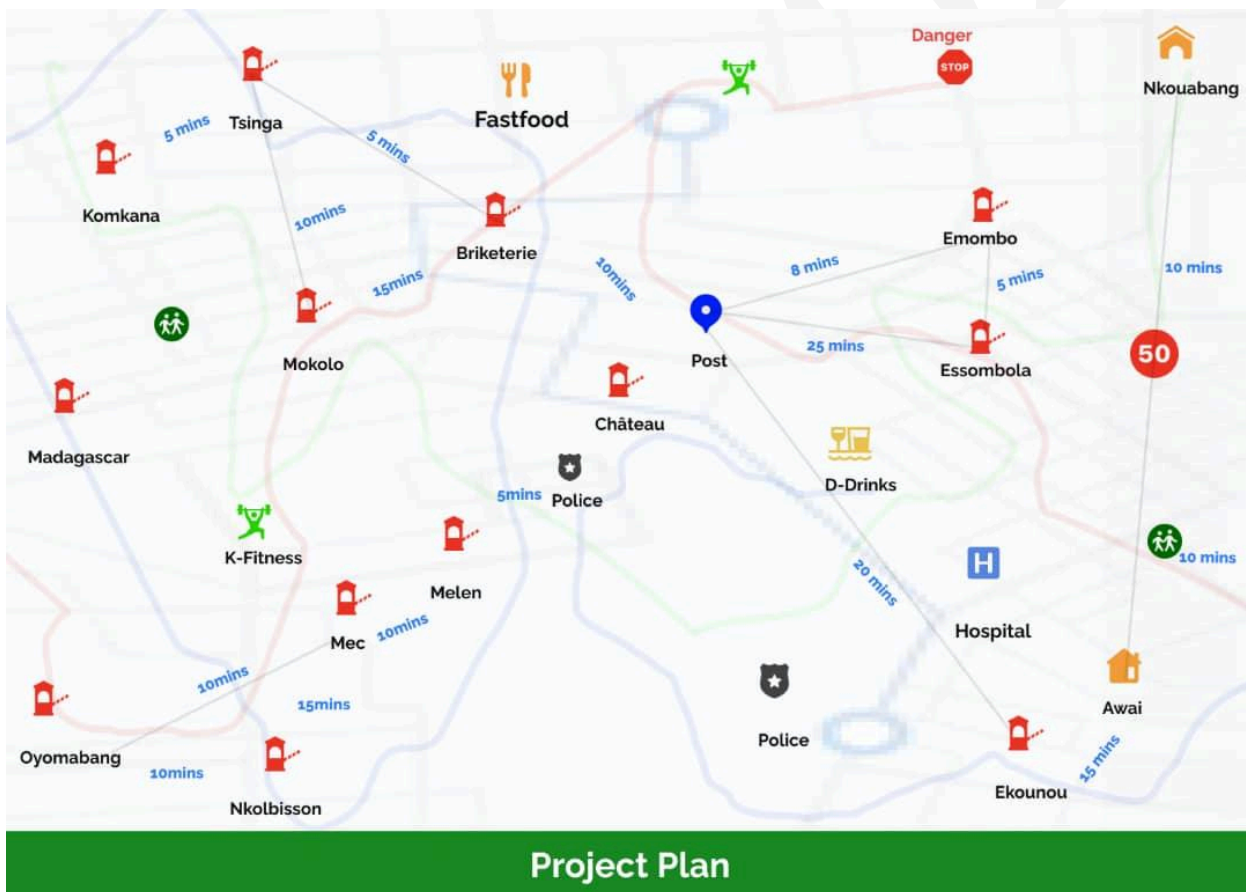
Ce prédicat affiche les stations intermédiaires dans l'itinéraire, en listant les transitions entre chaque paire de stations avec leur distance.

## IV. DÉTAILS DE L'IMPLEMENTATION

Pour l'implémentation, nous avons utilisé l'environnement UBUNTU et le langage Prolog avec la version SWI-Prolog 7.6.4, choisis pour leur robustesse et leur adaptabilité à notre projet. Le code source complet est disponible dans le fichier .pl fourni avec ce rapport.

## V. CAS DE TESTS ET RÉSULTATS

### PLAN DE LA VILLE





?- itineraire.

Bienvenue dans l'application de calcul d'itinéraire.

Saisissez la station de départ: nkoabang.

Saisissez la station d'arrivée: |: mokolo.

Saisissez l'heure de départ (HH:MM): |: "15:00".

Choisissez votre mode de transport (car, voiture, moto) : |: moto.

Itinéraire le plus court :

De nkoabang à essomba : 1.5 km

De essomba à poste : 3.75 km

De poste à mokolo : 1.5 km

Station de départ: nkoabang

Heure de départ: 15:00

Durée du trajet: 5.0625 minutes

Coût du trajet: 675.0 francs

Distance totale: 6.75 km

Station d'arrivée: mokolo

Heure d'arrivée: 15:5

true .

?- itineraire.

Bienvenue dans l'application de calcul d'itinéraire.

Saisissez la station de départ: emonbo.

Saisissez la station d'arrivée: |: tsinga.

Saisissez l'heure de départ (HH:MM): |: "10:00"

|: .

Choisissez votre mode de transport (car, voiture, moto) : |: voiture.

Itinéraire le plus court :

De emonbo à belibi : 0.75 km

De belibi à poste : 2 km

De poste à briquetterie : 1.5 km

De briquetterie à tsinga : 0.75 km

Station de départ: emonbo

Heure de départ: 10:00

Durée du trajet: 5.0 minutes

Coût du trajet: 375.0 francs

Distance totale: 5.0 km

Station d'arrivée: tsinga

Heure d'arrivée: 10:5

true .

## **POUR AFFICHER LA LISTE DES STATIONS**

?- stations.

Liste des stations disponibles:

- chateau
  - poste
  - belibi
  - emonbo
  - briquetterie
  - tsinga
  - nkomkana
  - melen
  - mecc
  - mokolo
  - nkolbison
  - oyomabang
  - ekounou
  - awai
  - essomba
  - nkoabang
- true.

## **VI. DIFFICULTES RENCONTREES ET SOLUTIONS APPORTÉES**

Nous avons eu des difficultés parce que nous n'avions jamais codé dans un langage comme celui-ci alors nous avons effectué des recherches pour mieux comprendre le langage.

## **VII. CONCLUSION**

Nous avons pu réaliser ce projet, grâce à l'encadrement de notre professeur, et cela a pu nous faire apprendre la base de l'intelligence artificielle. Pendant le projet, nous avons eu des incompréhensions et des challenges, mais nous avons pu nous en sortir grâce à notre détermination et la multitude d'informations disponibles sur le sujet.

## **ANNEXE : COMMANDES POUR L'EXÉCUTION DU PROGRAMME**

- ❖ Assurez-vous que SWI-Prolog est installé sur votre système.
- ❖ Téléchargez et extrayez le fichier .pl et les fichiers associés.
- ❖ Ouvrez SWI-Prolog et chargez le fichier .pl en utilisant la commande `consult('main.pl')`.
- ❖ Pour prendre connaissance des stations en entrant la commande `'station.'` dans l'interpréteur prolog.
- ❖ Pour exécuter le programme, appelez le prédicat `main/0` en tapant `'itinéraire.'` dans l'interpréteur Prolog.
- ❖ Renseignez la station de départ suivie d'un point et exécuter.
- ❖ Renseignez la station d'arrivée suivie d'un point et exécuter.
- ❖ Renseignez l'heure de départ suivie(dans les doubles cotes) d'un point et exécuter.
- ❖ Renseignez le mode de transport suivi d'un point et exécuter.