



Conception et mise en œuvre d'une solution logicielle pour la gestion des expéditions de véhicules d'urgence et l'optimisation de leurs itinéraires

▪ **Objet du Hackathon :**

Alors que les villes et le nombre de véhicules sur les routes continuent de croître, le nombre d'accidents et de situations d'urgence augmente également, ce qui rend encore plus critique le problème de la répartition et du routage des véhicules d'urgence.

La répartition consiste à envoyer vers le lieu d'un incident les véhicules disponibles dans les endroits de dépôt les plus proches, tandis que le routage permet de sélectionner l'itinéraire optimal pour atteindre cette destination. Dans ce contexte, la priorité est donnée au temps de réponse à ces urgences, qui est considéré comme la principale quantité à optimiser. Ce temps de réponse est estimé en prenant en compte la distance de l'itinéraire et le nombre de virages, car ceux-ci ont tendance à ralentir la progression des véhicules d'urgences.

▪ **Le travail demandé :**

Dans ce Hackathon, vous êtes invités à concevoir et réaliser une solution web ou mobile, qui permet de gérer des alertes d'incident et générer une route optimale pour dispatcher des véhicules d'urgence convenable vers les points d'incidents. Le processus de déclaration d'incidents doit être pris en charge dans l'interface de l'application et nécessite la saisie d'une description de l'incident, sa localisation, son niveau sous forme de 3 catégories : incident élevé, moyen ou bien faible, et finalement son type. Le type et le niveau de l'incident détermine la nature et la capacité des véhicules d'urgence. Les véhicules d'urgence sont supposés toujours présent dans leurs endroits de dépôt et se caractérise par leur type, position et capacité. La tâche du routage vérifie les suppositions suivantes :

- Seule une requête à la fois est autorisée.
- Les sites d'urgence ont un "niveau d'urgence" associé et les véhicules d'urgence ont une "capacité" pour répondre à ces niveaux. Les sites d'urgence ne peuvent être satisfaits que par un véhicule d'une capacité égale ou supérieure, ou par plusieurs véhicules dont la somme atteint ou dépasse le niveau d'urgence du site.
- Les trois niveaux d'urgence sont élevés, moyens ou faibles.
- Tous les véhicules d'urgence sont initialisés à leur dépôt de départ.

Les véhicules sont répartis de la manière suivante :

- La liste de tous les véhicules est filtrée pour éliminer ceux qui ne correspondent pas au type d'urgence. Par exemple, les ambulances ne seront pas prises en compte lors d'un appel pour un incendie.
- La distance entre les positions des véhicules et l'emplacement d'urgence est déterminé calculer en utilisant la formule de **Haversine**, la liste est triée par ordre croissant.



- Les véhicules sont ajoutés à la liste de répartition jusqu'à ce que le niveau de gravité de l'urgence soit satisfait. Si le niveau final de la flotte répartie est supérieur à ce qui est requis, nous essaierons de supprimer les véhicules de faible niveau pour correspondre parfaitement au niveau d'urgence. Par exemple : S'il y a une urgence de niveau 3, et que les véhicules les plus proches sont :

1. Véhicule A (niveau 1) : à 2 km
2. Véhicule B (niveau 1) : à 4 km
3. Véhicule C (niveau 3) : à 6 km

L'algorithme de répartition sélectionnera les trois véhicules (A, B, C), mais supprimera ensuite A et B car ils ne sont pas nécessaires pour répondre à l'urgence.

Il est recommandé de créer pour votre ville choisie, une base de données d'incidents et de véhicules. Néanmoins, Un jeu de donnée est disponible à votre convenance pour vous faciliter la tâche de test de votre application.

1. Cinquante endroits ont été sélectionnés comme "points d'incident" dans la ville de Khouribga pour servir comme destinations. (Fichier : site_incident.csv)
2. Les emplacements réels de casernes de pompiers, de postes de police et d'hôpitaux de cette zone ont été sélectionnés comme dépôts de véhicules pour différents types de véhicules d'urgence (camions de pompiers, voitures de police, ambulances). (Fichier : vehicule_urgence_localisation.csv)

Pour chaque incident saisi, l'application va sélectionner les véhicules convenablement placés dans la ville et va générer la route optimale reliant ces véhicules avec l'incident en minimisant au plus le temps de réponse à l'incident. Ce temps de réponse est estimé en prenant en compte la distance et le nombre de virages dans l'itinéraire, car ceux-ci ont tendance à ralentir la progression des véhicules d'urgences.

Le rendu final des résultats produite par l'application est préférablement illustrée sur une carte de navigation réelle de la ville sélectionnée. La présentation du travail effectué doit aborder les points suivants :

1. Les choix de conception de l'architecture de l'application de gestion des alertes d'incidents et du dispatching de véhicules d'urgences. L'architecture doit permettre l'ajout éventuel de d'autre types d'alerte, véhicules, et la vérification de la validité des positions de ces derniers dans la ville en question.
2. La formulation de la problématique et le choix de l'algorithme d'optimisation, du langage de programmation choisi et des bibliothèques utilisées. Une argumentation des différents choix de bibliothèque et des algorithmes d'optimisation est vivement recommandée.
3. La réalisation et le test sur un jeu de données de votre choix (voir les fichiers CSV fournis).

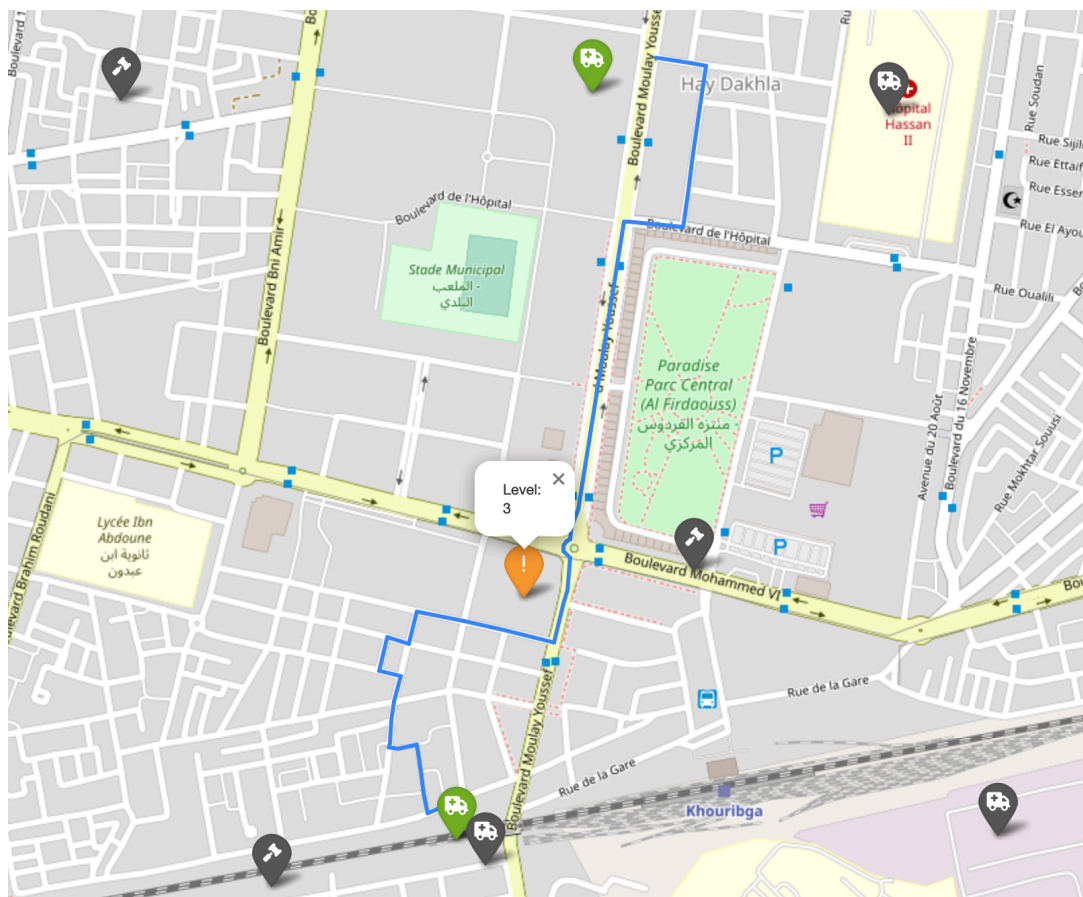
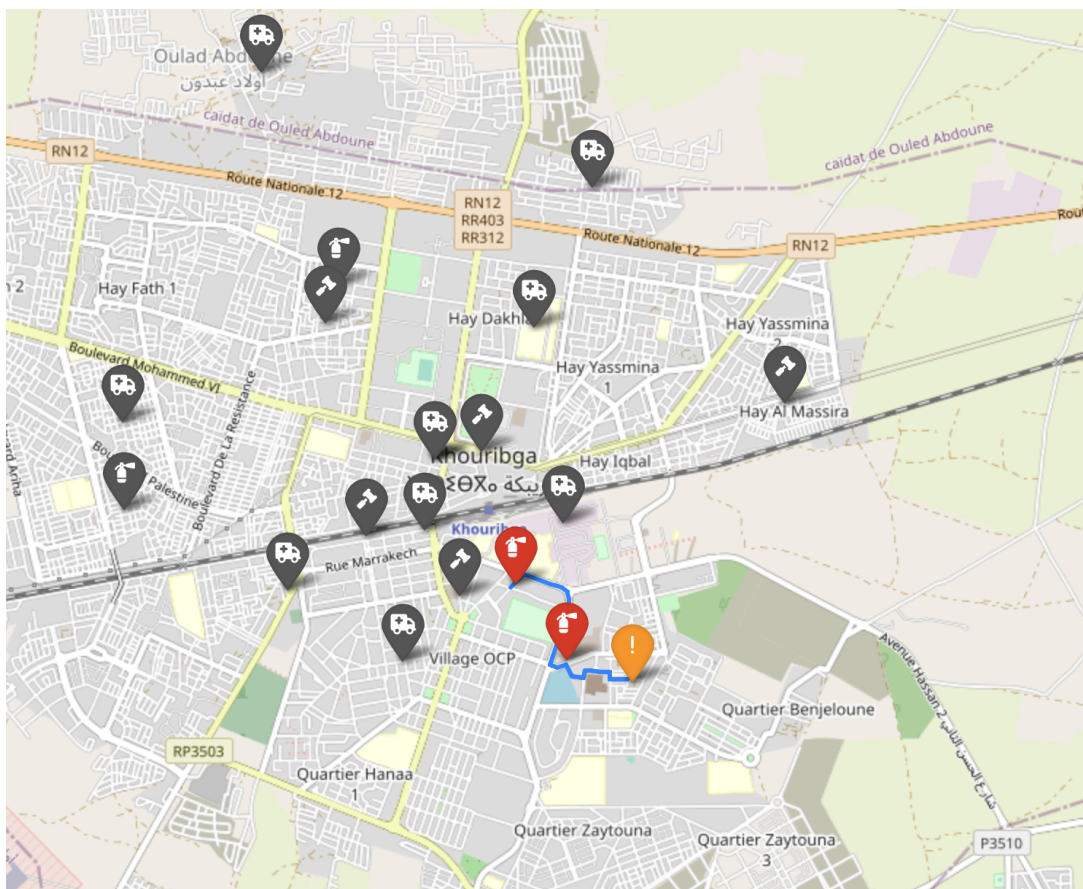
Pour résoudre ce problème, vous pouvez opter pour une implémentation utilisant les principes de programmation orientée objet et utiliser des algorithmes Meta-heuristique (ou autre algorithme heuristique ou évolutionnaire) optimisé en cascade. Vous êtes libre d'utiliser des librairies open source adéquates pour faire la tâche d'optimisation, routage, visualisation, etc.

Vous trouvez ci-dessous quelques visualisations du résultat finale de routage :



جامعة السلطان مولاي سليمان
ⵜⴰⵎⴰⵎⴰⵔⵜ ⵏ ⵓⵎⵓⵔⵓⵔⵓⵙ ⵏ ⵓⵔⵓⵔⵓⵔⵓⵙ
Université Sultan Moulay Slimane

ENSA
KHOUREBGA
ÉCOLE NATIONALE
DES SCIENCES
APPLIQUÉES





- L'évaluation : Vous serez évalués sur la base des points suivants :
 - Compréhension de la problématique et analyse justifiée : votre travail doit démontrer que vous avez une compréhension claire de la problématique posée, que vous avez identifié les enjeux et les contraintes, et que vous avez justifié vos choix de conception en fonction de ces éléments.
 - Originalité de la solution et créativité de l'approche : votre travail doit témoigner d'une approche innovante pour résoudre la problématique posée, que ce soit dans la conception de l'architecture logicielle, le choix de l'algorithme d'optimisation, ou encore dans la façon de traiter les données. Vous serez évalué sur votre capacité à apporter une réelle valeur ajoutée par rapport aux solutions existantes.
 - Illustration de la conception et optimisation de la solution proposée : vous devez présenter de manière claire et détaillée votre conception de l'application, en justifiant les choix techniques effectués. Vous devez également montrer que vous avez optimisé votre solution pour atteindre les objectifs définis, en utilisant des outils ou des méthodes spécifiques pour améliorer les performances ou réduire les coûts.
 - Qualité de l'application développée : vous serez évalué sur la qualité de votre application. Vous devez montrer que votre application est fonctionnelle et qu'elle répond aux exigences du cahier des charges.
 - Présentation et réponses aux questions : vous devez présenter votre travail de manière claire, organisée et professionnelle, en utilisant des supports appropriés.

À la fin de ce document, voici optionnellement quelques bibliothèques intéressantes, libres et gratuites dont vous pouvez éventuellement faire usage dans votre travail :

- Osmnx : un module python qui vous permet de télécharger des données géospatiales à partir d'OpenStreetMap et de modéliser, projeter, visualiser et analyser des réseaux routiers réels.
- Folium : un module puissant qui permet de faire de la visualisation de cartes interactives avec librairie JavaScript leaflet.
- Pymoo : un framework open-source qui offre des algorithmes à un et plusieurs objectifs de pointe et de nombreuses autres fonctionnalités liées à l'optimisation, la visualisation et la prise de décision.

Bonne chance à toutes les équipes participantes.