

## TP9

Ce TP est à faire après avoir fini les TP1 à 8. Le but est de mettre en pratique toutes les connaissances des TP précédents.

Il n'est pas nécessaire de faire tous les points des consignes qui suivent pour obtenir des points bonus, en revanche, tout code écrit devra être documenté et expliqué (en commentaires ou sur un document supplémentaire). Un code non-documenté ne vaudra pas de points.

Le travail est à envoyer avant le 22/06 23:59, tout travail rendu en retard ne vaudra pas de points.

### Frontend

L'application devra permettre d'afficher sur une carte interactive le contenu de fichiers gpx ou kml (chemins, points). Elle devra permettre l'interopérabilité entre les deux formats notamment la conversion et l'exportation dans les deux formats devra être possible. L'utilisateur aura la possibilité de chercher un point donné avec son adresse. Les données d'OpenStreetMap seront utilisées pour localiser l'adresse (<https://nominatim.org/>). Finalement l'application devra permettre de manière visuelle de fusionner et découper des chemins contenus dans des fichiers gpx ou kml ou créés à la main par l'utilisateur soit en le dessinant soit en cherchant le chemin, avec un certain nombre de contraintes, entre deux points.

L'application devra être réalisée en utilisant le framework JavaScript vue.js et la bibliothèque leaflet.js sera utilisée pour le rendu de la carte. Le calcul de chemin entre deux points sera réalisé en backend

### Backend

L'application sera une API REST permettant le calcul de chemin entre deux points géographiques donnés. Les données des chemins existants devront être obtenus via les données d'OpenStreetMap soit en local soit en passant par l'API OverPass. En plus du calcul de chemin le plus court en suivant les routes, le calcul de chemin le plus court entre deux points pour des modes spécifiques de déplacement (piéton, vélo) devra être possible. L'utilisateur pourra également demander un calcul d'itinéraire avec des points intermédiaires. Finalement le calcul de chemin le plus court pourra également être étendu au chemin le plus rapide notamment en prenant en compte les différences de vitesse de déplacement en fonction des propriétés du chemin emprunté (type de route, inclinaison,...).

Le serveur sera réalisé avec nodejs et expressjs.