

Mini Projet 3 : Trouver son chemin en vélo

1 Cahier des charges

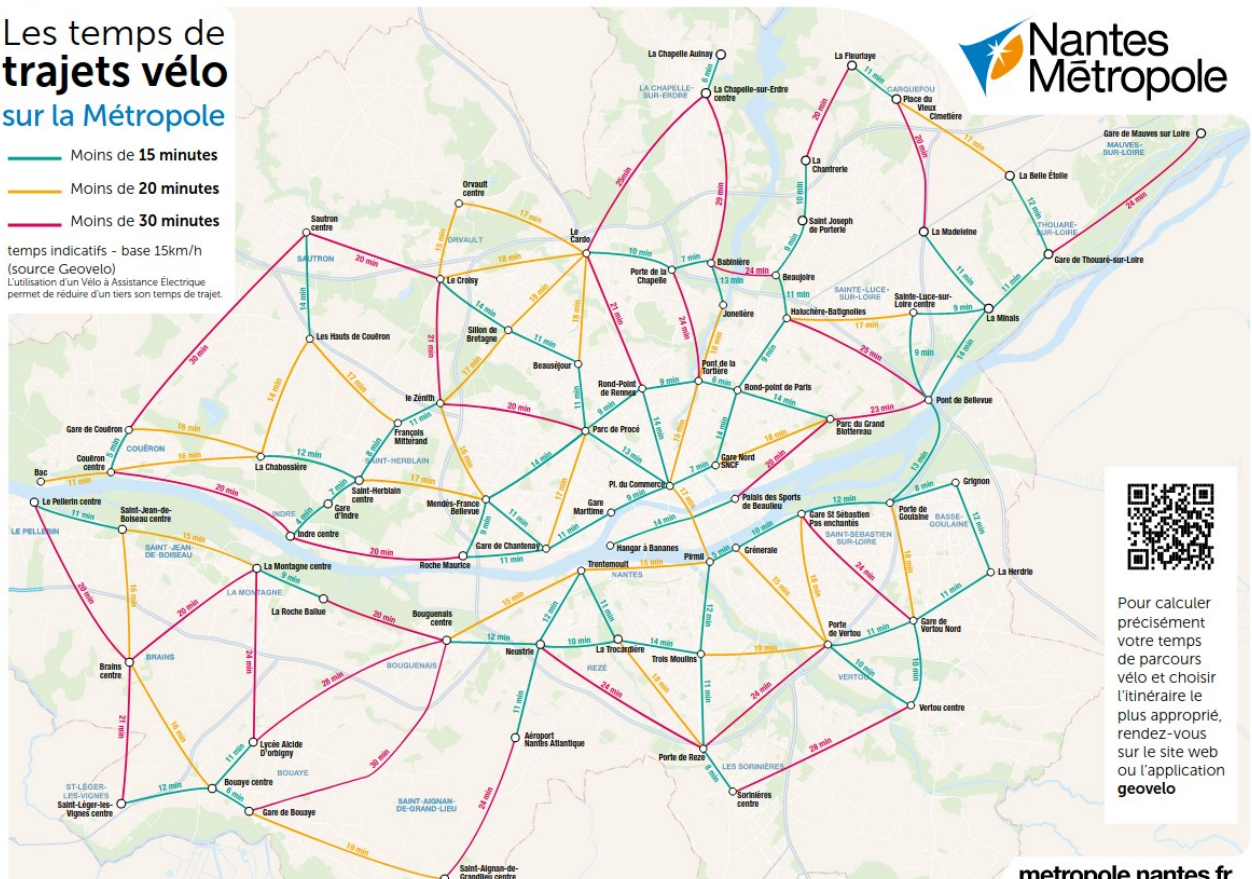
Objectif :

A partir d'une carte fournie par Nantes métropole (voir ci-dessous), créer une application Python qui calcule le meilleur trajet en vélo (le plus rapide !) entre deux lieux.

Les temps de trajets vélo sur la Métropole

- Moins de 15 minutes
- Moins de 20 minutes
- Moins de 30 minutes

temps indicatifs - base 15km/h
(source Geovelo)
L'utilisation d'un Vélo à Assistance Électrique permet de réduire d'un tiers son temps de trajet.



Fichier complet disponible ici :

<https://metropole.nantes.fr/files/images/deplacement-stationnement/NMMAJBasecarteVectParcoursEXE4web.pdf>

2 Les attendus

A rendre :

Les documents à rendre sont :

- 1 Fichier texte contenant :
 - l'explication du fonctionnement de l'application pour l'utilisateur
 - le schéma du graphe utilisé
- 1 Dossier zippé contenant le(s) programme(s) Python(s) et les données nécessaires au fonctionnement de l'application

Fonctionnalités demandées :

Les principales fonctionnalités de l'application sont :

- Saisir le lieu de départ et le lieu d'arrivée
- Calculer le plus court chemin
- Afficher l'itinéraire correspondant et le temps total

On pourra imaginer en bonus :

- Une interface graphique (tkinter ou graphviz)
- Un historique des itinéraires

+ 2

+ 1

Les compétences évaluées :

	A	B	C	D
APP /6	<ul style="list-style-type: none"> • Mobiliser les concepts et les technologies adaptés au problème 			
	<i>Implémentation efficace du graphe. Appropriation et implémentation réussie de l'algorithme de Disjktra.</i>	<i>Implémentation efficace du graphe. Appropriation de l'algorithme de Disjktra, mais implémentation difficile (AIDE).</i>	<i>Implémentation du graphe. Pas d'utilisation de l'algo de Dijkstra.</i>	<i>Pas d'implémentation du graphe.</i>
	<ul style="list-style-type: none"> • Rechercher l'information utile à l'aide de sources fiables 			
	<i>Recherches précises disponibles, dans les docs et sur le web. Codes adaptés pour répondre au problème.</i>	<i>Recherches effectives dans les docs à disposition. Les codes utilisés ne sont pas toujours adaptés.</i>	<i>Utilisations des exemples, avec une ré-appropriation manifeste. Codes externes peu ou mal utilisés.</i>	<i>Pas de recherche d'informations.</i>
REA /6	<ul style="list-style-type: none"> • Concevoir une solution algorithmique en réponse à un problème • Imaginer et concevoir une solution, décomposer en blocs, se ramener à des sous-problèmes simples et indépendants, adopter une stratégie appropriée 			
	APP FONCTIONNELLE Saisie le lieu de départ et le lieu d'arrivée. Calcul le plus court chemin Affichage de l'itinéraire correspondant et le temps total.	APP PARTIELLEMENT FONCTIONNELLE Saisie le lieu de départ et le lieu d'arrivée. Affichage d'un itinéraire et du temps total correspondant. Pas de calcul le plus court chemin.	APP NE FONCTIONNE PAS Saisie le lieu de départ et le lieu d'arrivée. OU Affichage d'un itinéraire et du temps total correspondant. Pas de calcul le plus court chemin.	APP NE FONCTIONNE PAS Aucune des fonctionnalités n'est remplie.
COM /6	<ul style="list-style-type: none"> • Communiquer à l'écrit en utilisant un langage rigoureux et des outils pertinents 			
	Explications claires dans le fichier texte pour utilisateur et graphe présent. Commentaires pertinents et détaillés.	Explications confuses dans le fichier texte, mais graphe présent OU Commentaires partiels ou lacunaires.	Explications confuses dans le fichier texte. ET Commentaires partiels ou lacunaires.	Pas de commentaires, ni explications pour l'utilisateur.
	<ul style="list-style-type: none"> • Travail d'équipe efficace 			
	Travail d'équipe clairement répartis, rôles bien définis. Suivi efficace de l'ensemble.	Travail d'équipe mal réparti. Suivi de l'avancée par tous.	Travail d'équipe mal réparti. Pas de suivi global.	Pas de travail d'équipe