

Cahier des Charges -

Team

Nathan Chevalier

Sachintha Mallawa Tantirige Dias

InfoSpe Epita



Table des matières

1	Introduction	3
1.1	Présentation du projet	3
1.2	Edition des cartes	4
1.3	Intérêts et enjeux	5
2	Etat de l’art	6
2.1	Applications/Editeurs d’itinéraire	6
3	Organisation pratique	10
3.1	Choix Techniques	10
3.2	Répartition des tâches	10
3.3	Planning prévisionnel	10
4	Conclusion	11

1 Introduction

1.1 Présentation du projet

Le but de ce projet est de créer une application de planification des trajets du métro parisien. L'application devra trouver une ou plusieurs alternatives pour rejoindre un point d'arrivée depuis un point de départ. La principale difficulté sera de pondérer les arrêtes en fonction des temps de trajet ainsi que de prendre en compte toutes les stations où des changements de lignes peuvent être effectués.

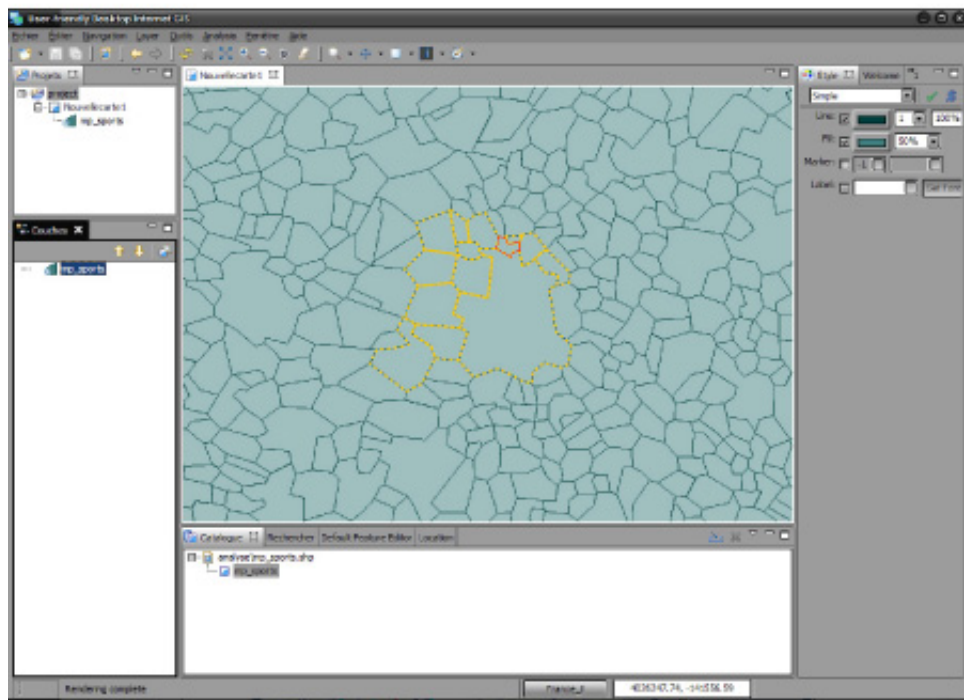
Les nombreuses lignes de métro et les interconnexions augmente fortement le nombre de possibilités pour passer d'un point A à un point B, mais une seule est la plus rapide. Le but d'une telle application étant de faire gagner du temps, l'algorithme devra privilégier des trajets qui pourrait sembler peu logique en premier abord.



1.2 Edition des cartes

La plus-value de cette application est l'intégration d'un système d'édition des plans de transports en communs. En effet, pour pouvoir s'adapter à n'importe quelle ville, une photo de la carte des transports d'une ville inconnue peut-être importée. A partir de cette image, l'utilisateur peut créer la version numérique de ce plan de transports en ajoutant des stations et des points d'arrêts, en connectant deux stations...

Cette édition doit pouvoir se faire dans une interface fluide permettant le déplacement et l'agrandissement afin d'avoir la possibilité d'éditer des plans complexes. Une fois l'édition terminée, ce plan doit être enregistré sous forme de graphe dans un format standardisé pour pouvoir être par la suite parcouru afin de trouver le plus court chemin.



La principale difficulté de cette partie sera de faire déplacer tous les points sur l'interface utilisateur quand ce dernier souhaite se déplacer sur la plan. Cela va nécessiter d'utiliser des positionnements relatifs et non absolus. A l'heure actuelle nous n'avons pas fais le choix de la bibliothèque GUI que nous utiliserons pour répondre au mieux à cette problématique.

1.3 Intérêts et enjeux

Ce projet fait suite à une question originale : quel trajet permet de parcourir les 304 stations du métro parisien le plus vite possible. Nous pouvons également imaginer dans un second temps ajout une note d'intérêt à chaque station tel que les lieux touristiques à proximité, les commerces et services à proximité ou bien encore les meilleurs endroits pour faire de la photo à proximité. Ainsi l'application pourrait proposer une alternative un peu plus longue mais plus intéressante en fonction de votre profil.

Pour répondre à ces questions, l'enjeu principal sera de trouver une implémentation optimale du système de transport qui rendra plus facile le travail des algorithmes de recherche par la suite. Cette implémentation pourrait également se baser sur un format de fichier standart permettant de modifier aisément les informations sur les stations, d'ajouter de nouvelles stations de nouvelles lignes (notamment celle de la ligne 14 qui desservira notre magnifique campus Epita Villejuif) ou bien même de changer totalement de ville.

2 Etat de l'art

2.1 Applications/Editeurs d'itinéraire

Voici une liste des applications qui permettent de calculer les itinéraires en fonction de la position GPS.

Citymapper :

Citymapper est une application connue sur Paris pour avoir la capacité de planifier les itinéraires en temps réel avec une grande variété de transports (Train, metro, bus, taxi, velo, et bien plus) ce qui permet pour les utilisateurs d'avoir un très grand choix d'itinéraire de déplacement.



Google maps :

C'est une application très basique, un calculateur d'itinéraire classique qui permet d'avoir des cartes précises avec des informations sur le trafic (que ce soit dans le metro ou sur les routes/autoroutes) en temps réel.

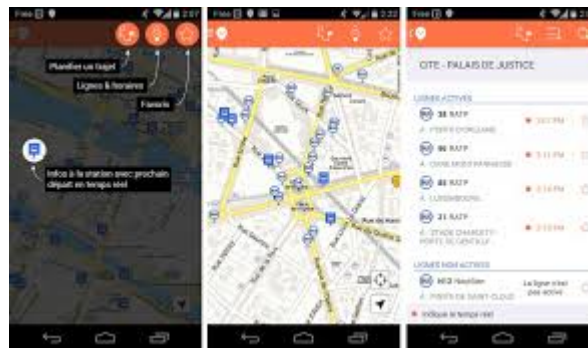
Metro01 :

C'est l'une des seules applications/systemes(apk) pour utilisateurs qui permet de calculer l'itinéraire d'un point A à un point B sans avoir besoin d'internet ce qui peut sauver le temps des utilisateurs s'il se retrouvent sans services. Mais reste un calculateur classique avec des options qui s'ajoute avec l'accès à internet(GPS etc...)



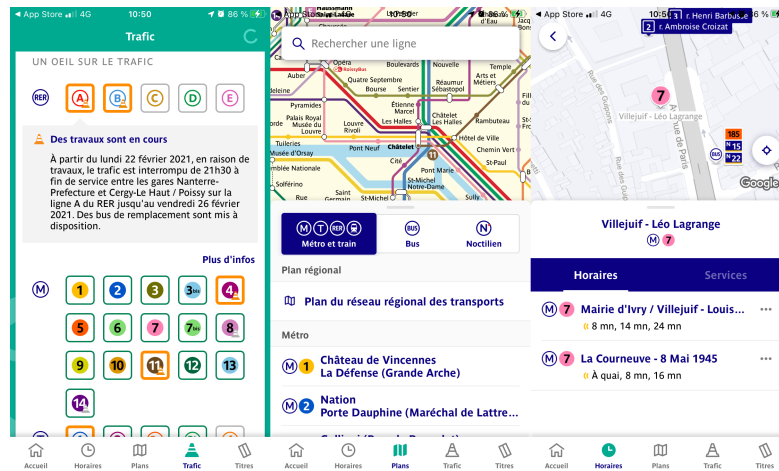
Moovit :

Comme citymapper, c'est un calculateur d'itinéraire qui à une très grande variété de modes de transports. Mais qui a pour particularité de regrouper l'itinéraire de plus de 2000 villes du monde qui permet un choix d'itinéraire très varié en temps réel. Que ce soit sur Paris ou ailleurs c'est une application qui est très utilisée.



RATP :

C'est le classique pour ce qui est du déplacement dans le metro parisien car il regroupe toutes les informations sur le plan des metros, les horaires, et le trafic en temps réel qui permet de prévoir des changements de ligne en cas de problèmes.



NOUVEAU

AllyApp :

Il permet aux utilisateurs de l'application d'alimenter de manière collaborative les données et offre une multitude d'API. Il dispose également d'une fonctionnalité de calcul d'itinéraire.

Fulcrum :

Il permet les collectes de données géo-référencées sur téléphone et peut être modifié. Et l'utilisation de la cartographie des transports avec le GPS.

JungleBus :

Cette application Android gratuite et open-source est concentrée exclusivement sur la collecte d'informations sur les arrêts de bus, dont les données géographiques/de localisation.



OSM tracker :

Cet outil offre un suivi GPS, associé à un formulaire d'enquête basé sur une application du type « build-it-yourself », permettant de collecter des informations sur les arrêts et itinéraires. Les données peuvent également être directement synchronisées avec OpenStreetMap.

3 Organisation pratique

3.1 Choix Techniques

Ce projet sera l'occasion de mettre en pratique les connaissances acquises sur certains types de structures complexes tels que les graphes. Afin de respecter les contraintes fixées dans le cadre du projet, l'application sera développée à l'aide du langage C sous Unix. Nous devons donc dans un premier temps créer des outils pour utiliser les structures de travail (ie les graphes).

Enfin, pour le site web dédié au jeu, nous utiliserons les langages HTML, CSS et JS obligatoires en front-end et ferons le choix de PHP pour le back-end et MySQL pour un éventuel système de base de données. Tous ces choix techniques n'engendrent pas de coût, puisque certains membres du groupe peuvent utiliser leurs serveurs web personnels.

Comme dit plus haut, nous n'avons pas fixé notre choix de bibliothèque de GUI qui permettra de créer l'éditeur de plan. Nous aimerions trouver une alternative à SDL.

3.2 Répartition des tâches

Tâche à effectuer	Référent + Suppléant
Implémentation des structures	Nathan + Sachintha
Algorithme de chemins	Nathan + Sachintha
Editeur de plans	Nathan + Sachintha
Saisie des données	Nathan + Sachintha
Interface	Nathan + Sachintha
Site et docu	Nathan + Sachintha

3.3 Planning prévisionnel

Tâches à effectuer	Soutenance 1	Soutenance 2	Soutenance 3
Implémentation des structures	100%	100%	100%
Algorithme de chemins	0%	70%	100%
Editeur de plans	40%	70%	100%
Saisie des données	50%	100%	100%
Interface	20%	40%	100%
Options	0%	0%	100%
Site et docu	20%	60%	100%

4 Conclusion

Au-delà des aspects techniques, la première condition pour la réussite d'un tel projet est la gestion des ressources humaines. Nous avons la particularité de n'être que deux pour ce projet ce qui pourra être une difficulté mais également une simplification car nous aurons peu de chance d'avoir des problèmes de groupe. La gestion du temps, et donc un travail régulier sera une clé pour l'aboutissement du projet. Il faudra également prendre du recul pour s'assurer au fur et à mesure de respecter les objectifs fixés