



Lyon 1

**CAHIER DES
CHARGES**

SAÉ GRAMA

Bénéficiaire :

IUT Lyon 1 - Département Informatique

92 boulevard Niels Bohr
69100 VILLEURBANNE

iutdoua.info@univ-lyon1.fr

Informations :

Auteurs :

- François GRAUX
- Elliot CURVAT

Date de conception : 7 février 2022

Table des matières

Présentations.....	3
Présentation de l'entreprise.....	3
Présentation du projet.....	3
Contraintes	4
Principes	4
Objectifs.....	5
Rendu	7
Identité visuelle	7
Organisation des affichages.....	7
Caractéristiques visuelles concrètes.....	7
Planning	8
Généralités	8
Rendus intermédiaires	8

Présentations

Présentation de l'entreprise

L'Institut Universitaire de Technologie Lyon 1 est divisé en plusieurs départements répartis sur 3 zones géographiques. L'IUT emploie 1 300 personnes pour encadrer et enseigner à plus de 5 000 étudiants (alternants compris).

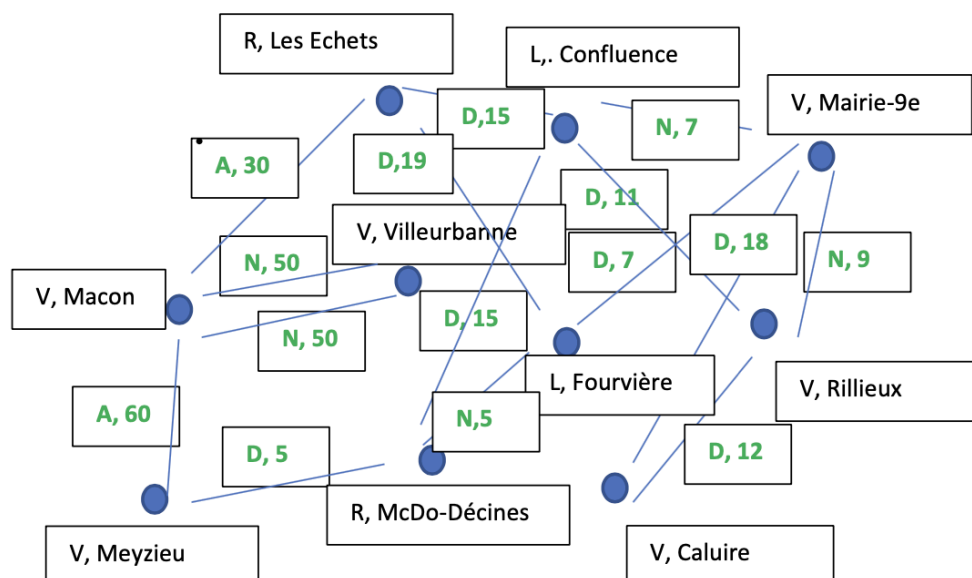
Le département informatique fait partie des 15 présents à l'IUT. Ce seul département enseigne à 250 personnes chaque année.

Présentation du projet

L'IUT Lyon 1 sollicite notre groupe pour réaliser une application informatique. Cette dernière a pour but d'analyser des graphes représentant des villes, restaurants et lieux de loisir. La finalité du projet est de modéliser une carte, un plan, représenté par un graphe et de le manipuler à travers de multiples fonctionnalités, pouvoir en tirer des chiffres, des chemins... L'analyse se fera à travers une IHM (interface homme-machine).

Dans notre projet, un sommet pourra représenter une ville « V » (Villeurbanne par exemple), un lieu de loisir « L » (Fourvière par exemple) ou encore un restaurant « R » (McDonald's par exemple).

Les arêtes entre les sommets représentent les routes. Elles sont pondérées en fonction de la distance en kilomètres séparant 2 sommets. Il existe trois types de routes pour notre graphe : les autoroutes « A » (entre *Macon* et le restaurant *Les Echets* par exemple), les routes nationales « N » (entre *Macon* et *Villeurbanne* par exemple) et les routes départementales « D » (entre *Caluire* et *Rillieux*).



Contraintes

Une contrainte du projet est que le graphe utilisé devra contenir un minimum de 30 sommets avec les proportions suivantes :

3/5 de villes	1/5 de restaurants	1/5 de centres de loisirs
---------------	--------------------	---------------------------

Ensuite, pour ce qui est des arêtes, on va supporter la possibilité qu'il y ait plusieurs arêtes entre deux mêmes villes. Par ailleurs, on souhaite qu'il y ait toujours un certain quota pour le type des routes :

1,5/5 d'autoroutes	1,5/5 de routes nationales	2/5 de routes départementales
--------------------	----------------------------	-------------------------------

Le graphe présenté doit être connexe et non orienté.

Principes

Dans le cadre de ce projet, nous allons utiliser différents principes et outils, expliqués ci-dessous.

Les graphes

La théorie des graphes apparaît au 18e siècle après les travaux d'Euler. Un graphe est un modèle abstrait montrant des arêtes entre objets définis. Par convention, notre graphe sera non-orienté, c'est à dire qu'il n'y a pas de notion de sens entre nos différents sommets.

Java :

Java est un langage de programmation apparu en 1991 et aujourd'hui majoritaire tant dans le monde privé que public. Celui-ci est orienté objet, organisé en paquets et sécurisé.

Dans le cadre de notre projet, Java est la solution la plus pratique pour nous. Le langage est régulièrement mis à jour, et s'adapte parfaitement avec nos applications actuelles. De plus, un langage objet nous permet de facilement définir chaque élément que nous allons manipuler dans le programme par exemple, définir un objet « Sommet » pour manipuler de manière simple les attributs qui définissent ce dernier et de définir les différentes méthodes pour l'utiliser en fonction de nos besoins.

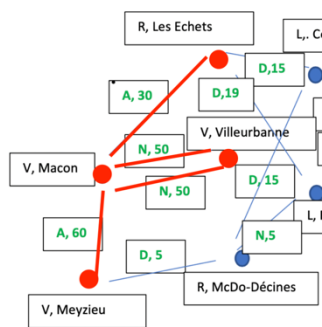
CSV :

CSV ou Comma Separated Values est un type de fichier qui, comme son nom l'indique, représente des données sous forme de lignes dont les valeurs sont séparées par des virgules. L'avantage principal de ce type de fichier est qu'il est universellement compréhensible et que beaucoup de logiciels le supportent.

Pour notre application, nous souhaitons pouvoir manipuler n'importe quel graphe à partir d'un fichier CSV correctement constitué.

Formatage des données du graphe :

Chaque ligne du fichier CSV décrira un sommet et toutes les arêtes qui partent de ce sommet.



Par exemple, une ligne du CSV concernant Mâcon sera :
V,Macon:A,30|R,Les
Echets;N,50|V,Villeurbanne;A,60|V,Meyzieu

Objectifs

L'objectif est d'interroger le graphe construit pour le manipuler selon différentes fonctionnalités

Primaire

0-afficher :

En ce qui concerne l'affichage, nous devons faire en sorte de donner la possibilité à l'utilisateur de faire apparaître à l'écran les informations qu'il souhaite en manipulant le graphe qu'il aura importé dans l'application. Les informations qu'il pourra afficher sont :

- Les villes
- Les lieux de loisir
- Les restaurants
- Les autoroutes
- Les routes nationales
- Les routes départementales

1-lister et interroger :

L'utilisateur aura la possibilité de lister/trier les sommets ainsi que les arêtes en fonction de leur catégorie respective (ville, lieu de loisir, restaurant et autoroute, route nationale, route départementale).

L'utilisateur aura également la possibilité d'interroger le graphe via l'application, c'est-à-dire qu'il pourra connaître :

- Les voisins directs d'un sommet donné (lieux reliés au sommet donné par une route)
- Les sommets reliés par une arête donnée (les lieux qui sont les extrémités d'une route choisie)

2-donner :

L'utilisateur pourra connaître :

- Le nombre de sommets de la catégorie choisie au préalable
- Le nombre de routes de la catégorie choisie au préalable

Il pourra, soit choisir de connaître le nombre pour une seule catégorie soit pour toutes les catégories.

3-vérifier :

L'utilisateur devra avoir la possibilité de vérifier :

Si deux sommets donnés sont reliés par une arête (s'il existe une route entre les deux sommets)

Si une ville choisie par l'utilisateur possède plus ou moins de chemins (composés par 2 arêtes exactement) la reliant à un autre lieu (ville, lieu de loisir, restaurant) qu'un deuxième

sommet lui aussi choisi par l'utilisateur. C'est-à-dire qu'il pourra comparer plusieurs villes pour savoir lesquelles ont plus ou moins de sommets directement liés.

Secondaire

4-calculer :

Le calcul consisterait à donner la possibilité à l'utilisateur de connaître la distance exacte entre deux lieux qu'il aurait choisi en additionnant les distances des routes nécessaires pour lier les deux lieux choisis.

4-trouver :

L'utilisateur aurait la possibilité de connaître un chemin à emprunter en fonction de certains critères comme :

- Un chemin traversant, entre deux sommets donnés, un troisième donné
- Un chemin traversant, entre deux sommets donnés, un troisième donné... et un x-ième donné

C'est-à-dire qu'il pourrait décider de trouver un itinéraire de Lyon à Paris en passant par des lieux spécifiques comme un restaurant et un lieu de loisir.

On pourra s'intéresser à la mise en place des deux fonctionnalités secondaires si le budget et le planning s'y prêtent.

Rendu

Identité visuelle

Organisation des affichages

Dans le cadre du projet nous voulons que le design ainsi que le couleur du logiciel soit simple et épuré.

Nous avons convenu que dans le cadre de ce projet chaque écran disponible sur l'application correspondra à une fonctionnalité :

Affichage	Contenu
Accueil	Choix des fonctionnalités
Écran 0	Fonctionnalité 0
Écran 1	Fonctionnalité 1
Écran 2	Fonctionnalité 2
Écran 3	Fonctionnalité 3
Écran 4 (objectif secondaire)	Fonctionnalité 4

Exemple : l'écran 0 ne correspondrait qu'à la fonctionnalité d'affichage des informations.

Caractéristiques visuelles concrètes

Nous avons décidé que pour ce projet, l'application devra être la plus simple et épurée possible. Elle devra aussi respecter la contrainte ci-dessous :

- Couleurs pastel comme le bleu et le vert

En matière d'identité visuelle, seuls ce critère est imposé. Le reste sera décidé par l'entreprise.

Planning

Généralités

L'IUT Lyon 1 demande que le projet, dans sa version finale, soit rendue au plus tard le 15 juin.

Rendus intermédiaires

En parallèle, l'entreprise demande à ce que l'avancement du projet soit prouvé au travers de plusieurs rendus intermédiaires. Le but est de vérifier la compatibilité des fonctionnalités entre elles et d'avoir une application finale stable correspondant aux attentes du maître d'ouvrage.

Entre autres, le planning pourrait ressembler à cela :

0 %	Début : 31/01/2022	Début du projet
20 %	J+27	Fonctionnalité 0 complétée
40 %	J+54	Fonctionnalité 1 complétée et intégrée
60 %	J+81	Fonctionnalité 2 complétée et intégrée
80 %	J+108	Fonctionnalité 3 complétée et intégrée
100 %	Fin : 16/06/2022	Rendu final

Si une fonctionnalité se révèle plus courte à réaliser, ou plus simple à intégrer, on pourra travailler sur les fonctionnalités secondaires jusqu'à la fin de la période en question, et les reprendre s'il nous reste du temps avant la fin de la date butoir.