

LO43 Flotte de véhicule autonomes

Buri Theo Florian Lacour

Table des matières

Introduction	3
I Présentation du sujet	4
0.1 Objectif	5
0.2 Reformulation du sujet	5
0.2.1 Plateau	5
0.2.2 Missions et contrôleur	5

Introduction

Dans le cadre de l'UV LO43 " Bases fondamentales de la programmation orientée objet", il nous a été demandé de réaliser un projet de groupe, afin de mettre en pratique les connaissances acquises lors des cours et TDs du semestre.

Trois sujet nous ont été présentés. Nous avons fait le choix de traiter le sujet de la "Flotte de véhicules autonomes", et ceci pour plusieurs raisons :

- (A TROUVER)
- N'étant que deux, sur un maximum de quatre étudiants par groupe autorisés, les autres sujets ne nous ont pas paru réalisables en temps et en heures et sans bugs majeurs...
- (A TROUVER)

Nous présenterons tout d'abord le sujet, ses contraintes, et les libertés prises par rapport à celles-ci. Par la suite, nous parlerons des différents diagrammes UML, et les expliquerons. Enfin, nous terminerons par l'implémentation en Java et l'interface graphique.

Première partie

Présentation du sujet

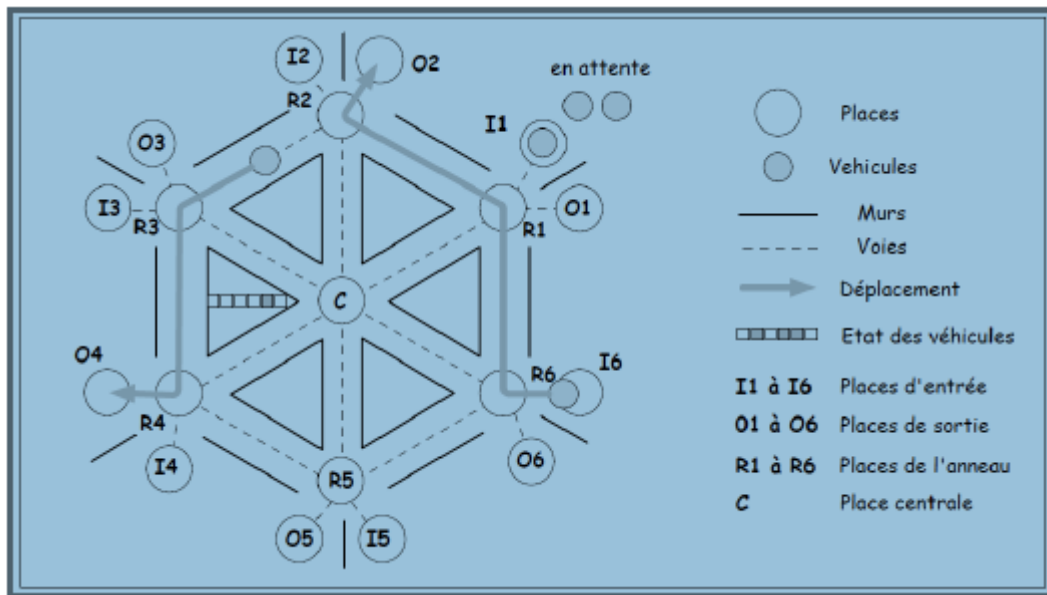
0.1 Objectif

Le programme à réaliser consistait en la modélisation d'une flotte de véhicules évoluant dans une infrastructure de circulation partagée. Pour cela, il a tout d'abord fallu modéliser la partie calculatoire à l'aide du langage UML, puis ensuite l'implémenter en Java et lui donner une interface graphique

0.2 Reformulation du sujet

0.2.1 Plateau

Le plateau donné par le sujet est le suivant :



Chaque place, exceptée celle du centre, est rattachée à une place de départ et une place de sortie. Les voitures disposant d'une mission partent depuis une place d'entrée et, à la fin de celle-ci, rejoignent une place de sortie. La place centrale n'est utilisée que lors d'un trajet reliant deux places opposées...

0.2.2 Missions et contrôleur

Chaque passager doit être transporté d'une place à une autre. Pour ce faire, ils envoient une requête au module qui se charge du contrôle du plateau : le contrôleur. Par la suite, celui-ci

De plus ce sont les voitures qui doivent décider du moment où elles partent, pour cela elles doivent envoyer une requête au contrôleur pour savoir si leur chemin est déjà réservé. Afin d'éviter toute erreur lors de la réservation nous avons défini un ordre tel que suit :

$$I1 < I2 < I3 < I4 < I5 < I6 < R1 < R2 < R3 < R4 < R5 < R6 < O1 < O2 < O3 < O4 < O5 < O6 < C$$

Par exemple pour une voiture qui désire aller de la place I1 à O2 la request map sera la suivante.

I1	I2	I3	I4	I5	I6	R1	R2	R3	R4	R5	R6	O1	O2	O3	O4	O5	O6	C
T	F	F	F	F	F	T	T	F	F	F	F	F	T	F	F	F	F	F

0.2.3 Contraintes et libertés prises sur le sujet

De plus les voitures ne peuvent prendre la place centrale qu'à la seule condition qu'elles doivent aller à la place en face de la leurs. Cette requête est implémentée sous forme de bitmap.