

Université Claude Bernard



Lyon 1

## Rapport de projet

# M1IF20 : Projet Transversal Innovant

### Equipe :

MBOUP Mouhamadou Lamine 11922789

BARRY Abdoul Rahime 11712599

DERBAL Younes 11809208

BARRY Aliou Sadio 11809163

## *Table des matières*

Introduction	2
Organisation du projet	3
Structure du code	3
Détection des couleurs	4
Mise en place de la caméra	4
Conclusion	5

# I. Introduction

Dans le cadre de notre semestre 2 du master 1 informatique, nous avons pour objectif de réaliser un projet pour l'UE MIF20 nommé *Projet Transversal Innovant*.

L'objectif de ce projet était de combiner nos connaissances théoriques et pratiques acquises dans les autres UE. Tout ceci autour d'un projet en lien avec les technologies innovantes.

Afin de fusionner nos compétences autour de ce projet, nous avons eu à notre disposition un robot Thymio II, un Raspberry Pi et deux caméras et un circuit avec des codes couleurs et des priorités marqués dessus.

A la fin de ce projet nous devrions avoir codé (en python) un robot capable rouler en autonomie sur le circuit en respectant le code de la circulation (sens interdit, priorité à droite, cédez le passage etc...)

## II. Organisation du projet

### Phase 1 :

Nous avons commencé par l'installation du Raspberry pi sur notre carte et ensuite, nous avons configuré le système pour qu'on puisse y accéder directement via nos machines en SSH ainsi l'installation des différents outils pour exécuter notre code.

### Phase 2 :

Avec l'utilisation du simulateur de Thymio Suite, nous avons compris le fonctionnement du robot ainsi que son architecture. C'est-à-dire l'ensemble des variables qu'il possède afin de pouvoir les utiliser dans notre programme.

### Phase 3:

Mettre en place un script qui nous permettra de se connecter avec le robot en utilisant *asebamedulla* côté robot et la bibliothèque Python *D-bus*, ensuite faire marcher le robot en suivant la ligne grise. Donc il marche tout droit sans lire les codes.

### Phase 4:

L'implémentation du code permettant de faire fonctionner la caméra tout en prenant des captures d'images des différents codes couleurs sur le circuit afin de les analyser, les traiter pour les injecter dans le code du robot pour que ce dernier puisse tourner à gauche, à droite ou s'arrêter si nécessaire en fonction du code couleur capturé. (bug voir le fichier *test\_camera.py*).

### III. Structure du code

Notre code est composé de trois fichiers :

- *aseba.py* contient une classe Aseba qui gère la liaison entre le robot et notre code en utilisant la bibliothèque D-bus et la classe AsebaException pour gérer les exceptions.
- Le fichier *main.py* contient la classe Thymio qui commence par initialiser les différents paramètres du robot tels que : les capteurs horizontaux, les capteurs de sol, la vitesse, ses statuts (WAIT\_CODE, RIDE, TURN, STOP,..) pour bien manipuler le robot. La méthode principale main utilise aussi autre fonction telle que la méthode *followTrack* permettant au robot de suivre la ligne, et comme nous n'avons pas fini la synthèse d'image avec la caméra, nous avons tenté d'utiliser un système d'étalonnage et détection des cases avec un des deux capteurs en bas du robot qui se charge de détecter la ligne tout en utilisant le principe des statuts.
- Et enfin le fichier *test\_camera.py* qui permet de capturer les codes couleurs et ensuite les analyser en utilisant la bibliothèque CV2. Mais nous n'avons pas pu l'intégrer dans notre classe principale pour que le code du robot fonctionne en même temps que la caméra. C'est pour cela que nous avons choisi d'utiliser les capteurs du robot comme expliqué ci-dessus.

### IV. Détection des couleurs

Le traitement des images pour la détection des couleurs via la caméra nous ayant posé trop de problème et vu le peu de temps qui nous restait nous avons utilisé comme solution palliative les capteurs du raspberry pi pour détecter les codes couleurs sur la map.

La partie droite du robot détecte la ligne du circuit et partie gauche les codes couleurs ( exemple noir-noir-blanc-blanc signifie stop ou encore rouge-rouge-blanc-blanc signifie virage serré à gauche etc...) et c'est ainsi, de cette manière nous avons pu gérer toutes les contraintes de priorité sur le circuit.

### V. Mise en place de la caméra

Pour mettre en place la caméra sur le Raspberry Pi et permettre à ce dernier de prendre des photos ou enregistrer des vidéos, nous avons utilisé le module caméra de Raspberry Pi.

Nous avons commencé à nous documenter pour savoir comment contrôler les deux caméras depuis une ligne de commande.

- *raspistill -o Desktop/image-test.jpg -w 640 -h 480* : pour une image
- *raspivid -o Desktop/video.h264* : pour un enregistrement vidéo

Ensuite, comprendre comment contrôler le lancement et la vision de la caméra avec un programme Python.

L'objectif de cette caméra permet à notre programme de détecter les codes couleurs présents sur la route afin que le programme puisse savoir quelle fonctionnalité doit être exécutée. En plus, elle nous permet de visualiser en live les couleurs détectées sur la vidéo.

Cependant nous avons utilisé les bibliothèques *CV2* et *Numpy* pour tester notre caméra.

Le problème est que notre caméra détecte des carrés avec un contour noir un peu de partout. De ce fait, il génère plusieurs valeurs dans le terminal. Ce qui nous empêche de prendre que les codes couleurs en carré.

## VI. Conclusion

Le but de ce projet était de coder un robot capable de circuler en autonomie sur un circuit avec d'autres robots tout en respectant les règles de la circulation soigneusement établie par le professeur responsable de la matière.

Grâce à ce projet, nous avons acquis énormément de connaissances techniques, beaucoup plus que ce qu'on pouvait imaginer avant le début de cette UE. Nous avons appris de nouveaux outils et technologies tels que : Thymio , *AsebaMedulla* , la technologie du Raspberry Pi et des nouvelles bibliothèques comme D-bus et CV2 pour la gestion d'images.

Par dessus tout, nous avons appris à travailler en équipe pendant ces moments difficiles de pandémie bien sûr avec la supervision de monsieur GUILLOU Erwan qui n'a ménagé aucun effort pour nous accompagner durant tout le long de ce projet

Globalement on est très satisfait du travail qu'on a réalisé même si nous sommes conscient qu'il y a des choses à améliorer surtout au niveau de la détection des codes couleurs via la caméra.