JULIEN VADEBONCOEUR  
MARC-ANDRÉ COUTURIER  
GUIDE TECHNIQUE  
420-6E9-LL, groupe 00001

GUIDE TECHNIQUE DU ROBOT MINDSTORM

Travail présenté à  
Serge Lévesque  
Département d’informatique

Cégep de Lévis  
23 novembre 2021

# Tank Mindstorm Guide technique

## Démarrage :

* Le Lego Mindstorm doit être démarré manuellement.
* Au démarrage du Raspberry pi, un script est lancé et attend qu’un tag RFID soit scanné. Après avoir scanné le tag RFID il sera validé. Si le tag est valide le serveur apache sera lancé et la connexion Bluetooth au Lego Mindstorm sera établie. Pour ajouter un tag RFID à la liste des tags valides, il faut lancer l’utilitaire RFID en python. Un mot de passe sera demandé pour l’ajout d’une carte RFID valide.
* Le panneau de contrôle et le tableau de bord seront alors accessibles sur l’adresse IP du Raspberry pi sur le port 80. Le Panneau de contrôle sera accessible directement à la racine du serveur et le tableau de bord sera accessible à l’index du dossier « /TableauBord ».

## Classe Ev3Controller :

### Variables globales :

* device : objet qui contient l’instance ev3
* MAC : constante qui contient l’adresse mac du Mindstorm
* FRONT\_MOTORS : tableau contenant les valeurs des deux moteurs avant
* BACK\_MOTORS : tableau contenant les valeurs des deux moteurs arrière
* LINUX : Constante qui dit si Le système d’exploitation est linux

### Constructeur :

Le constructeur prend seulement l’adresse mac sous forme de string

### Fonction :

**Start : Démarre les moteurs choisis avec la vitesse choisie**

Paramètres :

* Speed (0 à 100) : Vitesse des moteurs
* Motorlist : tableau contant les valeurs des moteurs à avancer

**Stop : Stop les moteurs choisis**

Paramètres :

* Motorlist : tableau contenant les valeurs des moteurs à arrêter

**Move : Fait avancer le véhicule en ligne droite**

Paramètres :

* Speed (0 à 100) : Vitesse des moteurs (Si la vitesse est négative le véhicule reculera)

**StopAll : Arrête tous les moteurs**

Aucun paramètre

**TurnLeft : Fait tourner légèrement le véhicule sur la gauche**

**(Seules les roues du côté droit vont tourner)**

Paramètres :

* Speed (0 à 100) : Vitesse des moteurs

**TurnRight : Fait tourner légèrement le véhicule su la droite**

**(Seules les roues du côté gauche vont tourner)**

Paramètres :

* Speed (0 à 100) : Vitesse des moteurs

**TurnRightOnHimself : Fait tourner le véhicule sur lui-même**

**(Les roues du côté gauche vont tourner vers l’avant et les roues du côté droit vont tourner vers l’arrière)**

Paramètres :

* Speed (0 à 100) : Vitesse des moteurs

**TurnLeftOnHimself : Fait tourner le véhicule sur lui-même**

**(Les roues du côté droit vont tourner vers l’avant et les roues du côté gauche vont tourner vers l’arrière)**

Paramètres :

* Speed (0 à 100) : Vitesse des moteurs

**Stream : Démarre le Stream de son HLS du micro de la caméra**

Aucun paramètre

**Sound : Joue un son présent sur le Lego Mindstorm**

Paramètres :

* Which : le nom du fichier sonore à lire sur le Mindstorm

**Speak : Fait jouer une voix sur le Mindstorm à partir d’un texte et d’un accent**

Paramètres :

* Text : La phrase à fait dire au Lego Mindstorm
* Language (« en » par défaut) : La langue dans laquelle la phrase est écrit pour avoir le bon accent

**GetBattery : Retourne le niveau de batterie du Lego Mindstorm**

Aucun Paramètre

## Contrôleur Principal :

Le contrôleur principal en écrit en python. Il sert à faire la liaison entre tous les éléments du projet.

Panneau de contrôle ßà Contrôleur principal : Cette connexion est faite avec deux connexion web socket sur le même serveur web socket qui est sur le port 443. Le port 443 est utilisé pour éviter la détection et le blocage par le firewall du cégep. Le Panneau de contrôle est une page web servie par un serveur apache2 qui réside sur le Raspberry pi.

Tableau de bord ßàContrôleur principal : Cette connexion est faite avec une connexion socket entre le back end du tableau de bord et le contrôleur principal. Il est fait pour que le back end du tableau de bord puisse avoir accès à toutes les données qu’il a besoin.

Contrôleur principal ßà Lego Mindstorm : Cette connexion est faite en Bluetooth et elle est utilisée pour contrôler les moteurs du véhicule à partir du Raspberry pi, ainsi que pour envoyer les commandes sonores qui sont aussi possibles.

Contrôleur principal ßà Base de données : Cette connexion est faite en XBee. Le contrôleur envoie par XBee le type de requête voulu suivi des données nécessaires pour exécuter la requête.

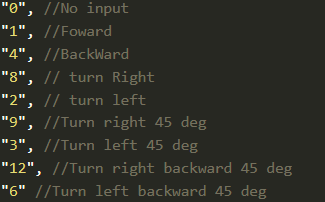
## Panneau de contrôle :

**Langages utilisés** : Html, javascript, css

**Protocole de communication** : http, web socket

Le panneau de contrôle est divisé en plusieurs parties, cela inclus un onglet d’état, le point vu de la caméra, deux consoles sur les états des sockets et un onglet de commandes audio.

À l’ouverture du panneau de contrôle deux connexions web socket sont établie entre le client web et Raspberry pi. Le site sera maintenant à l’écoute d’entées clavier ou d’une manette. Sur le backend un Stream vidéo est lancé. Sur le site web quand l’utilisateur exécute une commande, elle est envoyée au backend sous forme de code.



Une fois le code envoyé au backend, une commande est envoyée par Bluetooth au Mindstorm et est exécuter.

Le panneau de contrôle utilise deux web sockets sur le port 443 pour ne pas être bloqué par le firewall du cégep.

Le panneau de contrôle demande une image à l’ouverture du Web socket et lorsqu’il l’a reçu et affiché il en demande une nouvelle.

Lorsqu’une commande est utilisée sur le Panneau de contrôle, la commande est envoyée par le Web socket de contrôle et est traité par le contrôleur principal du Raspberry pi.

Le contrôleur utilise ensuite la classe Ev3Controller pour communiquer avec le Lego Mindstorm.

## Tableau de Bord :

Le tableau de bord est l’interface qui permet à l’utilisateur de surveiller l’état du Mindstorm.

Le tableau de bord demande en temps réel au Raspberry pi les dernières commandes exécutées, le niveau de la batterie et l’adresse IP du client connecté au panneau de contrôle. Ensuite il affiche toutes ces données à l’utilisateur.

Pour ce qui est du Backend en PHP, lorsqu’une commande AJAX est faite, le Backend va se connecter sur un socket du contrôleur principal du Raspberry pi. Ensuite le Raspberry pi renvoie les informations qui sont conservé dans un tableau. Pour finir, le tableau est réinitialisé et la connexion est fermée. Cette opération est répétée à chaque seconde et garde le tableau de bord à jour.

## Base de données :

Base de données PostgresSql qui contient un historique des commandes envoyées au Mindstorm

La base de données réside sur un ordinateur externe et est accéder par XBee. Elle contient une table commandes et cette table a les colonnes suivantes: id, code, direction, temps. Un script python qui utilise le module psycopg s’occupe d’insérer les commandes dans la BD.

## Contrôle local:

Le fichier LocalControl.py contient le code nécessaire pour contrôler le Lego Mindstorm localement pour faire des tests. Il utilise la classe Ev3Controller pour communiquer avec le Lego Mindstorm de la même façon que le fait le contrôleur principal.