

FIRST[®] Tech Challenge 2023-2024

Guide de Survie FTC Master

Version TETRIS[®] FIRST[®] Tech Challenge

Competition Set

Table des matières

- 1 Introduction : qu'est-ce que la FTC et à quoi sert ce guide ? 4
 - 1.1 La FIRST® Tech Challenge, c'est quoi ? 4
 - 1.2 Professionnalisme coopératif® 4
 - 1.3 A quoi sert ce guide ? 4
- 2 Qu'avez-vous besoin pour construire le robot ? 5
 - 2.1 Kits de pièces 5
 - 2.2 Logiciel de CAO 5
 - 2.3 Logiciel de Programmation 7
- 3 A quoi va ressembler le robot final ? 8
- 4 Assembler le robot pièce par pièce sur votre logiciel de CAO 10
 - 4.1 Téléchargez les pièces du kit dans votre logiciel de CAO 10
 - 4.2 Assembler des mécanismes du robot 10
- 5 Monter le robot en vrai, mécanisme par mécanisme 11
- 6 Installer les systèmes de contrôle 13
 - 6.1 Le Control Hub (microcontrôleur situé sur le robot) 13
 - 6.2 Le Driver Hub 14
 - 6.3 Les moteurs et les servo-moteurs 15
 - 6.4 Les capteurs 17
 - 6.5 Câbler le robot 21
 - 6.6 La batterie 22
 - 6.7 Les moteurs 23
 - 6.8 Les Servo-Moteurs 24
- 7 Configurer le robot 25
 - 7.1 Configurer le Driver Hub 25
 - 7.2 Configurer le Control Hub 25
 - 7.3 Configurer la connexion entre le Driver Hub et le Control Hub 25
 - 7.4 Configurer les moteurs et les servomoteurs 25
 - 7.5 Configurer les capteurs (seulement celui du kit de démarrage) 28
 - 7.7 Sauvegarder une configuration 29
- 8 Créer son premier programme en Java 30

8.1 Mieux comprendre le programme.....	30
8.2 Modifier son programme pour actionner un moteur	30
8.3 Connecter une manette	32
8.4 Modifier son programme pour actionner un servo moteur	34
8.5 Plus d'informations.....	34
9 Créer son premier programme en Block (style Scratch).....	34
9.1 Mieux comprendre le programme.....	35
9.2 Modifier son programme pour actionner un moteur	35
9.3 Connecter une manette	36
9.4 Modifier son programme pour actionner un servo moteur	38
9.5 Plus d'informations.....	38
10 Conclusion	41

1 Introduction : qu'est-ce que la FTC et à quoi sert ce guide ?

1.1 La FIRST® Tech Challenge, c'est quoi ?

La FIRST® Tech Challenge est un programme centré sur l'élève qui vise à offrir aux élèves une expérience unique, stimulante et enrichissante. Chaque année, les équipes se lancent dans un nouveau jeu où elles conçoivent, construisent, testent et programment des robots autonomes et pilotés qui doivent effectuer une série de tâches lors des matchs. Pour en savoir plus sur la FIRST® Tech Challenge et les autres programmes FIRST®, visitez le site <https://robotiquefirstfrance.org/>

1.2 Professionnalisme coopératif®

FIRST® utilise ce terme pour décrire l'intention derrière nos programmes.

Le Professionnalisme coopératif® est une façon de faire les choses qui encourage le travail de haute qualité, et qui met l'accent sur le respect des individus et de la communauté.

Le Dr Woodie Flowers explique le professionnalisme coopératif dans cette [vidéo \(voa\)](#).

1.3 A quoi sert ce guide ?

Ce document suggère une liste d'étapes que les équipes peuvent suivre pour construire un robot. Il vous permettra de vous débrouiller entièrement en partant de de zéro, de la réception du kit de pièces à la programmation.

Ce document se lit comme on lirait une recette de cuisine ; il faut suivre les étapes dans l'ordre. Cependant, dans certaines parties il vous sera peut-être nécessaire de cliquer sur les liens externes qui vous dirigeront vers d'autres ressources plus spécifiques.

Ce guide a pour vocation d'introduire les équipes aux concepts de la construction d'un robot et de les initier aux différentes techniques de l'ingénierie. Le robot final sera fonctionnel mais pas nécessairement très compétitif en match ; pour remporter une régionale il faudra aller plus loin.

2 Qu'avez-vous besoin pour construire le robot ?

2.1 Kits de pièces

Dans ce guide, nous allons utiliser deux kits de pièces :

- Le TETRIX® FIRST® Tech Challenge Competition Set ([voir lien](#)). Ce kit contient toutes les ressources matérielles nécessaires pour construire les différents éléments qui composent un robot de FTC. Il contient par exemple des moteurs, des roues, des vis, des plaques de métal, etc. Ce kit ne contient pas d'élément de contrôle (câbles, contrôleurs, capteurs).



- Le Kit de Control Robotique FIRST France. Ce kit contient tout ce qui sert à contrôler le robot et le programmer : contrôleurs, câbles, manette.



2.2 Logiciel de CAO

- La conception assistée par ordinateur ou CAO (ou CAD en anglais), est un concept d'ingénierie qui permet à une personne de dessiner, mesurer, standardiser et créer des plans afin de fabriquer des pièces à partir

de machines comme une imprimante 3D par exemple. La CAO se pratique à l'aide d'un logiciel de CAO qui permet de réaliser toutes les actions citées ci-dessus.

- Simplement, la CAO vous permet de rassembler vos idées pour vous éviter de commettre des erreurs pendant que vous fabriquez vos pièces.
- Il existe plusieurs licences disponibles gratuitement pour faire de la CAO. Voici un lien pour vous renseigner sur tous les logiciels et la procédure pour les avoir gratuitement : https://ftc-docs.firstinspires.org/cad_resources/index.html
- Pour commencer à les utiliser, voici des liens vers des tutoriels pour débiter :
 - Solidworks:
https://www.youtube.com/watch?v=UIttc_2p4DY&list=PLrOFa8sDv6jcp8E3ayUFZ4iNI8uuPjXHe
 - Autodesk Inventor:
https://www.youtube.com/watch?v=KKbwf2a53bA&list=PLRhna5_X7uWvjFID3uU0vKxpiuw8XvF35
 - Tinker CAD:
https://www.youtube.com/watch?v=2JFxtUIOnEI&list=PL90LC6zq_Lzf9tHyFPzX_90A35BFTfEBs
 - FreeCAD:
<https://www.youtube.com/watch?v=NXN7TOg3kj4&list=PLWuyJLVUNtc0UuszswD0oD5q4VeWTrK7JC>
 - Autodesk Fusion 360 :
<https://www.youtube.com/watch?v=vEIU3k25kJY&list=PLrOFa8sDv6jKx9poMArMUV2MGbZoXrCT>
 - OnShape : <https://www.youtube.com/watch?v=pMWnsHpDIQE&list=PLxmrkna-ixrIQmsPR3MITi4Ru1bnMH4-l>
 - Creo Parametric:
https://www.youtube.com/watch?v=sVWsUS_7V6s&list=PLrOFa8sDv6jVMccV28fssFut0EG0N-Nb6

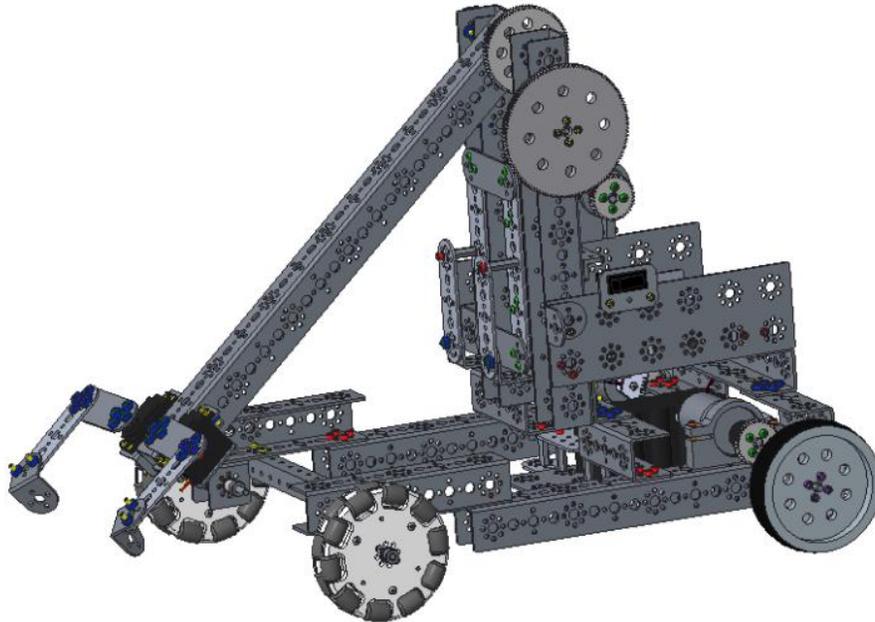
2.3 Logiciel de Programmation

- Dans le cadre de la FTC, la programmation sert à envoyer des instructions au robot pour qu'il effectue des tâches.
- Exemple de tâches que le programmeur peut ordonner au robot d'effectuer :
- Lorsque le pilote appuie sur le joystick gauche vers l'avant, fais tourner les deux moteurs des roues du robot pour avancer.
- Pour pouvoir donner une série d'instructions au robot, il faut un logiciel de programmation et un langage. Pour un robot FTC, vous pouvez programmer en blocks ou en Java.
- Dans ce guide, il vous sera expliqué la façon de programmer en Java et en Block.
- D'abord il vous sera expliqué la programmation en Block puis en Java.

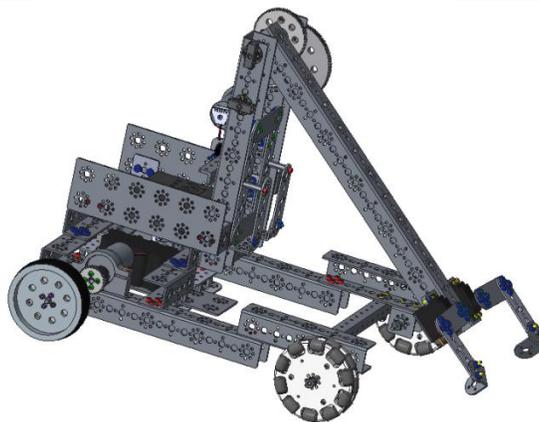
- Pour la programmation par Blocks, voici un tutoriel pour vous aider à démarrer :
<https://www.youtube.com/watch?v=sb-wF35TuvQ&list=PLlryJer4FuggBT5-4ZDTcYivs7kJHowXb>
- Pour la programmation en Java, voici un tutoriel pour vous aider à démarrer :
<https://openclassrooms.com/fr/courses/6173501-apprenez-a-programmer-en-java>

3 A quoi va ressembler le robot final ?

Voici le robot final en modélisation 3D !

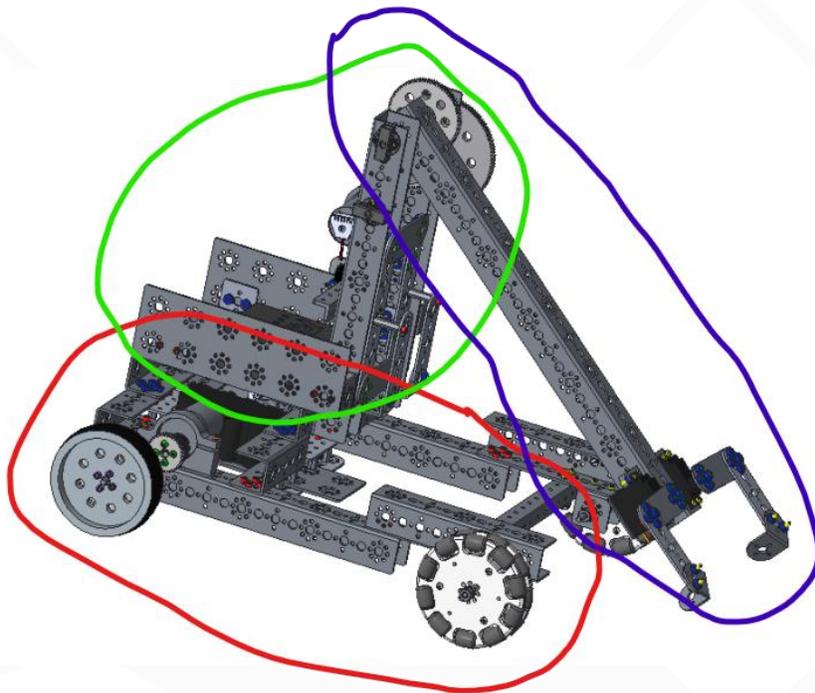


Le robot peut se déplacer sur le terrain, récupérer des cônes au sol avec une modification de la pince, les faire monter à l'aide d'un bras rotatif, et les relâcher dans les piquets. Voici un lien vidéo pour le voir en action : [Vidéo PushBot du FIRST Tech Challenge - YouTube](#)



Il se compose de plusieurs grandes "parties" ou mécanismes :

- Le châssis qui est en fait une base roulante sur laquelle reposent les autres mécanismes. Il se compose de 4 roues actionnées par 2 moteurs.
- Une pince qui permet d'attraper les cônes sur le terrain et de les expulser pour les enfiler dans les piquets. La pince est actionnée par deux servomoteur : il s'agit d'un moteur qui va très doucement est qui n'est pas fait pour effectuer des tours sur lui-même.
- Un bras rotatif au bout duquel est accroché la pince. Le bras tourne à l'aide d'un moteur.



4 Assembler le robot pièce par pièce sur votre logiciel de CAO

Cette étape est cruciale car elle vous permettra de visualiser le robot en 3D avant de l'assembler en vrai. Vous pourrez ainsi faire des modifications à votre guise, puis pas la suite améliorer les mécanismes ou bien en concevoir des nouveaux de toute pièce.

L'objectif de la compétition est que vous soyez les plus autonomes possibles dans la réalisation du robot, il est donc conseillé d'essayer d'assembler le robot par soi-même, sans regarder les guides détaillés disponibles sur internet. Cependant, si vous êtes bloqués, vous pouvez consulter les ressources suivantes :

- Le [lien CAO](#) (téléchargeable en STEP qui est l'un des formats de fichier 3D les plus communs pour la CAO) du robot monté au complet est mis à disposition par FIRST France. Ce lien vous emmène vers un modèle OnShape que vous pouvez télécharger en STEP.
- Un guide très détaillé du montage du robot. Il s'agit du guide pour monter le robot "en vrai" mais il vous sera très utile pour l'assemblage 3D. Ce guide est très similaire à une notice Lego.

4.1 Téléchargez les pièces du kit dans votre logiciel de CAO

Une fois le fichier STEP du robot téléchargé, il faudra importer le modèle dans votre logiciel de CAO. Lorsque le modèle STEP importé et convertit, vous pouvez le sauvegarder dans le format que vous voulez.

Dans la suite de ce guide, il vous est expliqué comment assembler chaque pièce modélisée pour recréer le modèle que vous venez de télécharger.

4.2 Assembler des mécanismes du robot

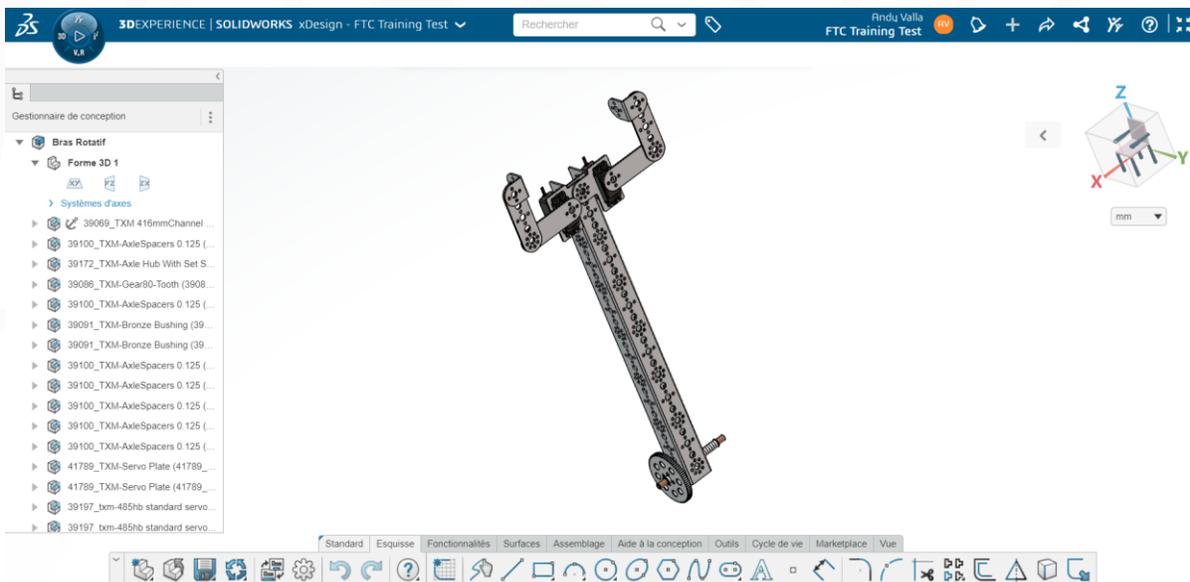
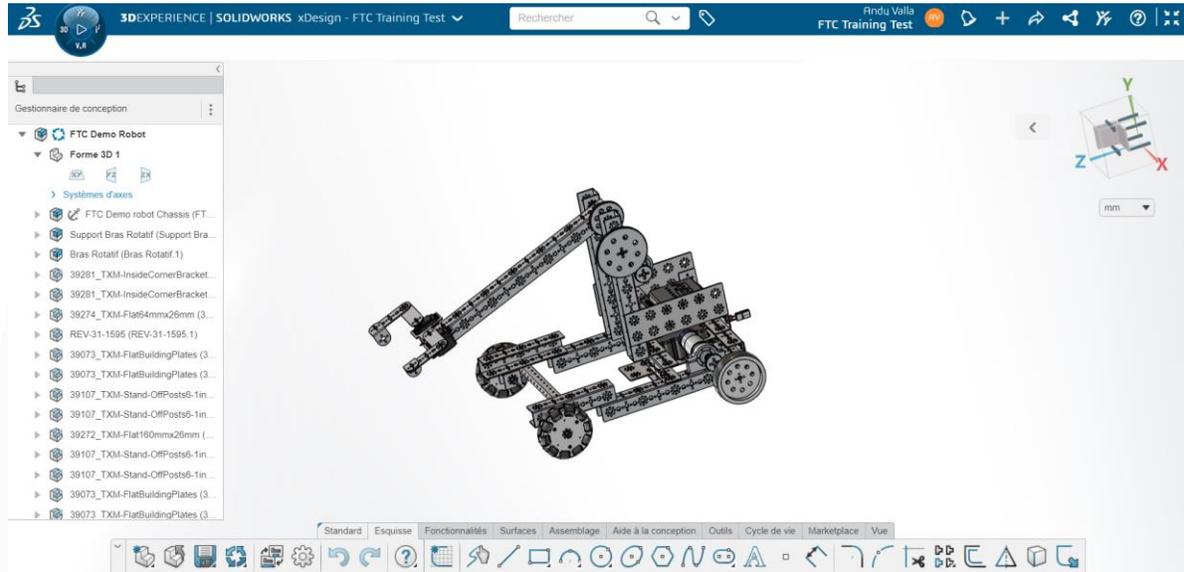
Voici une vidéo montrant un assemblage de deux pièces ensemble, puis un assemblage de deux mécanismes ensemble : <https://www.youtube.com/watch?v=Yj6oYVqbB7g>

Votre Conseiller technique Robotique FIRST France est à votre disposition pour vous aider en termes de formation, d'aide et de ressources pour vous aider à progresser.

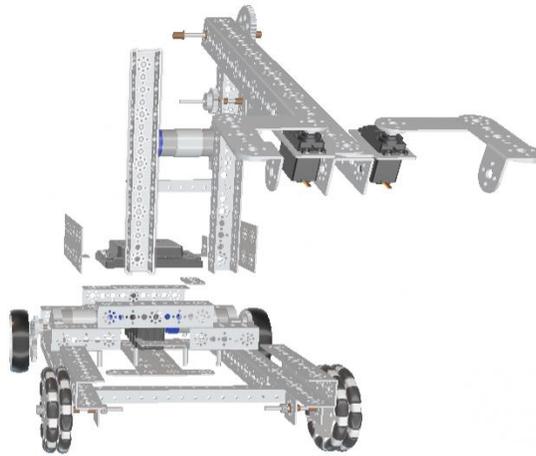
5 Monter le robot en vrai, mécanisme par mécanisme

Une fois que vous avez finalisé la CAO du robot, vous pouvez utiliser le logiciel pour vous aider dans le montage. Vous commencez mécanisme par mécanisme. N'hésitez pas à aller au détail pour être sûr de ne pas se tromper. Par exemple, regardons ensemble l'assemblage de la pince :

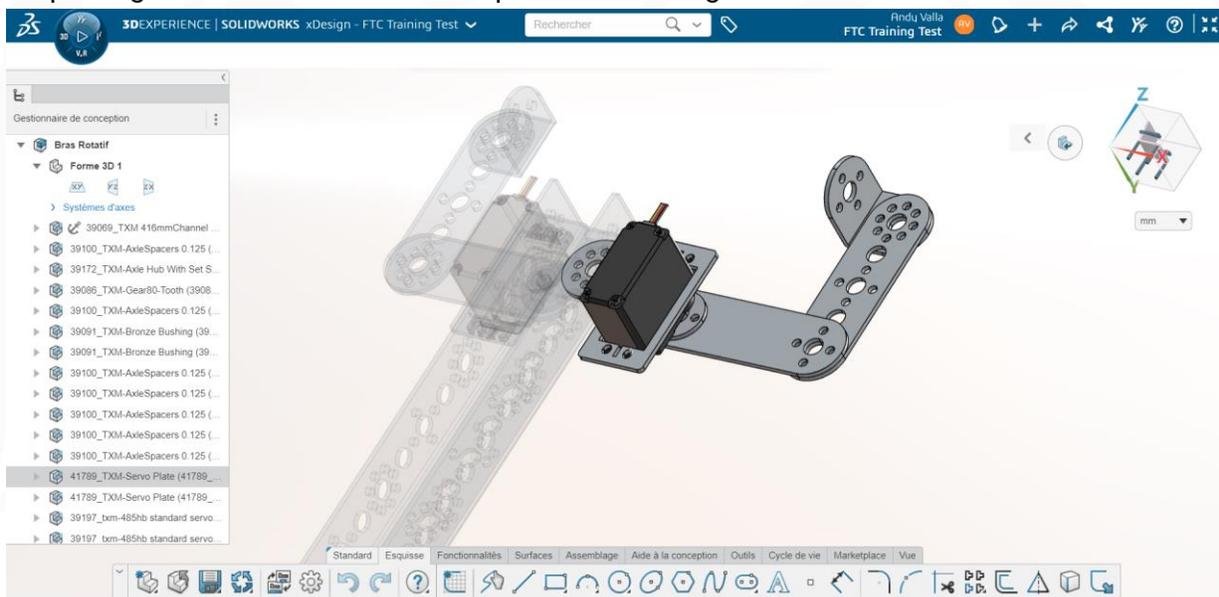
- Je commence par isoler la pince dans mon logiciel de CAO :



- Ensuite on peut faire une vue éclatée ou explosée. C'est une vue qui permet à tous les composants d'être séparé pour mieux comprendre l'ordre de montage.



- On peut également avoir le détail très précis du montage en sélectionnant les éléments voulus :



Utiliser la CAO pour assembler les pièces est le meilleur moyen d'être sûr de monter le mécanisme dans le bon ordre et surtout de ne pas oublier des pièces.

6 Installer les systèmes de contrôle

6.1 Le Control Hub (microcontrôleur situé sur le robot)

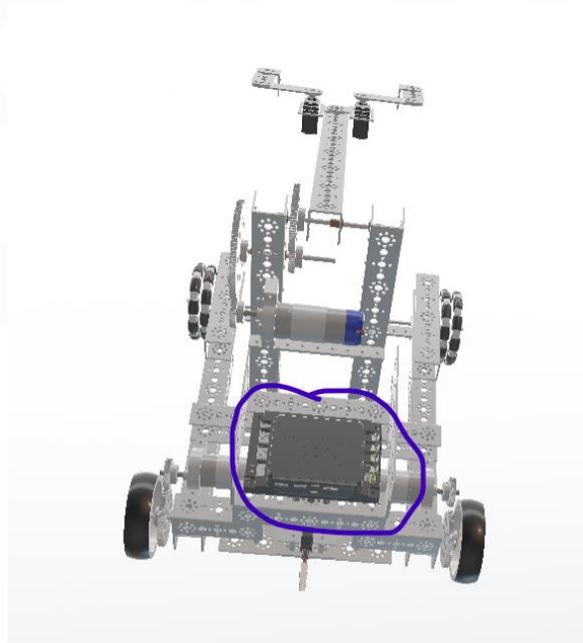


Le Microcontrôleur est le cerveau du robot, il permet d'établir des liens de communication entre les capteurs, les moteurs et la manette du pilote. C'est sur celle-ci qu'est téléversé le code du robot (programme en Java).

En FTC, le Control Hub fait office de microcontrôleur. Voici le lien pour voir la documentation du Control Hub : <https://www.revrobotics.com/rev-31-1595/>

Pour l'installer sur le robot, vous avez quatre emplacements pour des vis. Faites attention qu'il soit bien attaché pour éviter de le casser en cas de chocs.

Faites attention qu'il ne soit pas trop loin de la batterie pour pouvoir le connecter facilement.



6.2 Le Driver Hub



Le Driver Hub est un petit ordinateur qui sert d'interface entre le pilote et le robot. Voici un lien pour voir la documentation du Driver Hub : <https://www.revrobotics.com/rev-31-1596/>

Pour piloter le robot, il suffira de brancher une manette sur un des ports USB du Driver Hub et de connecter la manette. Ceci vous sera montré plus tard dans le guide.

Le Driver Hub n'est pas à installer sur le robot. C'est une station de pilotage qui permet de piloter le robot à distance.



6.3 Les moteurs et les servo-moteurs

Il existe plusieurs moteurs pour permettre à votre robot de réaliser des actions. Ce sont tous des moteurs à courant continu. Pour comprendre ce qu'est un moteur à courant continu, voici une courte vidéo explicative : <https://www.youtube.com/watch?v=A3b3Km5KVXs>

Motors

TETRIX 12V DC Motor 	AndyMark 12V DC Motor 	MATRIX 12V DC Motors 
TETRIX 12V TorqueNADO DC Motor	AndyMark NeveRest series 12V DC Motors	Modern Robotics/MATRIX 12V DC Motors
REV 12V DC Motor 	REV 12V DC Motor 	
REV Robotics HD Hex 12V DC Motor	REV Robotics Core Hex 12V DC Motor	

Pour plus d'information sur chaque moteur :

- TETRIX 12V TorqueNADO DC Motor : <https://www.pitsco.com/TETRIX-MAX-TorqueNADO-Motor-with-Encoder>
- AndyMark NeveRest series 12V DC Motors: <https://www.andymark.com/products/neverest-series-motor-only>
- Modern Robotics/MATRIX 12V DC Motors: <https://www.gobilda.com/modern-robotics-matrix-12vdc-motor/>
- REV Robotics HD Hex 12V DC Motor: <https://www.revrobotics.com/rev-41-1301/>
- REV Robotics Core Hex 12V DC Motor: <https://www.revrobotics.com/rev-41-1300/>

Dans la FTC, il y a aussi des servo-moteurs. Ce sont de plus petits moteurs qui fonctionnent pas-à-pas. Voici une vidéo explicative sur les servo-moteurs : <https://www.youtube.com/watch?v=fb8ZSe8fVIQ>

Voici les servo-moteurs utilisés dans la compétition :

Servos

REV-41-1097	GB 2000-0025-0504	W39197
		
REV-41-1097	GB 2000-0025-0504	W39197

Pour plus d'information sur chaque servo-moteur :

- REV-41-1097: <https://www.revrobotics.com/rev-41-1097/>
- GB 2000-0025-0504: <https://www.gobilda.com/2000-series-5-turn-dual-mode-servo-25-4-super-speed/>
- W39197: <https://www.pitsco.com/TETRIX-MAX-Standard-Scale-Servo-Motor>

6.4 Les capteurs

Un capteur est un instrument de mesure qui permet à votre robot de sonder le monde autour de lui. La récupération de ces données permet au robot de prendre une décision. Par exemple, lorsque vous vous apprêtez à traverser la route, vous regardez le feu piéton. Si le feu piéton est rouge vous vous arrêtez, si le feu piéton est vert, vous traversez. Les yeux sont vos capteurs, votre cerveau prend la décision de s'arrêter ou d'avancer en fonction de ce que vos yeux transmettent à votre cerveau.

Dans un match de FTC, un capteur peut être utilisé pour détecter la distance entre le robot et un mur du terrain. En fonction de cette distance, le robot peut s'arrêter ou continuer d'avancer.

Voici la liste des différents capteurs :

Distance Sensor (Ultrasonic)

MB1242



MB1242

RCWL-1601



RCWL-1601

Un capteur de distance à ultrasons est un dispositif capable de mesurer la distance entre un objet et le capteur. Pour ce faire, il émet une onde sonore et mesure le temps que met l'onde à se déplacer jusqu'à l'objet puis à revenir. En utilisant ce temps et la vitesse du son, la distance peut être calculée.

Distance Sensor (Optical)



Un capteur optique de distance est un dispositif capable de mesurer la distance entre un objet et le capteur. Pour ce faire, il envoie un faisceau lumineux et mesure le temps que met le faisceau pour aller jusqu'à l'objet et revenir. En utilisant ce temps et la vitesse connue de la lumière, la distance peut être calculée. Il faut savoir que la façon dont l'objet en question interagit avec la lumière peut modifier la précision de la mesure de la distance. Un objet transparent, tel qu'un panneau de terrain, fournira souvent des mesures imprécises.

Color Sensor



Un capteur de couleurs est un dispositif de sortie numérique capable de mesurer la couleur d'un objet. La plupart des capteurs de couleurs exigent que l'objet en question soit relativement proche du capteur.

Touch Sensor

REV-31-1425



REV-31-1425

Un capteur tactile est un dispositif de sortie numérique qui détecte l'activation d'un bouton. Il peut être utilisé comme interrupteur de fin de course, pour limiter l'amplitude de mouvement d'un mécanisme.

Magnetic Limit Switch

REV-31-1462

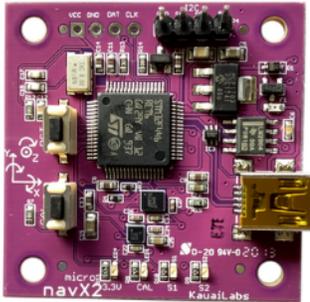


REV-31-1462

Un interrupteur de fin de course magnétique est utilisé pour détecter la présence d'un aimant à proximité. Il est généralement utilisé pour limiter la plage de mouvement d'un mécanisme qui pourrait être endommagé s'il dépassait cette limite. Pour ce faire, on place un aimant sur le mécanisme en question, ce qui déclenche l'activation de l'interrupteur de fin de course.

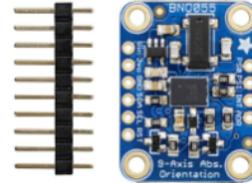
IMU

NAV X2 Micro



NAV X2 Micro

BNO055



BNO055

Une unité de mesure interne (IMU) est un capteur qui combine un gyroscope, un accéléromètre et un magnétomètre. Un gyroscope est un dispositif qui indique l'orientation angulaire d'un objet en trois dimensions. Un accéléromètre est un dispositif qui indique l'accélération d'un objet en trois dimensions. L'accélération peut être considérée comme le taux de variation de la vitesse à un moment donné. Un magnétomètre est un appareil qui mesure l'intensité des champs magnétiques sur trois axes. Il peut être utilisé comme une boussole pour obtenir l'orientation d'un robot par rapport aux pôles de la Terre.

Pour avoir plus d'information concernant ces capteurs :

- MB124: https://www.maxbotix.com/ultrasonic_sensors/mb1242.htm
- RCWL-1601: <https://www.adafruit.com/product/4007>
- REV-31-1505: <https://www.revrobotics.com/rev-31-1505/>
- REV-31-1557: <https://www.revrobotics.com/rev-31-1557/>
- MR-45-2018: <https://modernroboticsinc.com/product/color-sensor/>
- REV-31-1425: <https://www.revrobotics.com/rev-31-1425/>
- REV-31-1462: <https://www.revrobotics.com/rev-31-1462/>
- NAV X2 Micro : <https://www.andymark.com/products/navx2-micro-navigation-sensor-bundle>
- BNO055: <https://learn.adafruit.com/adafruit-bno055-absolute-orientation-sensor>

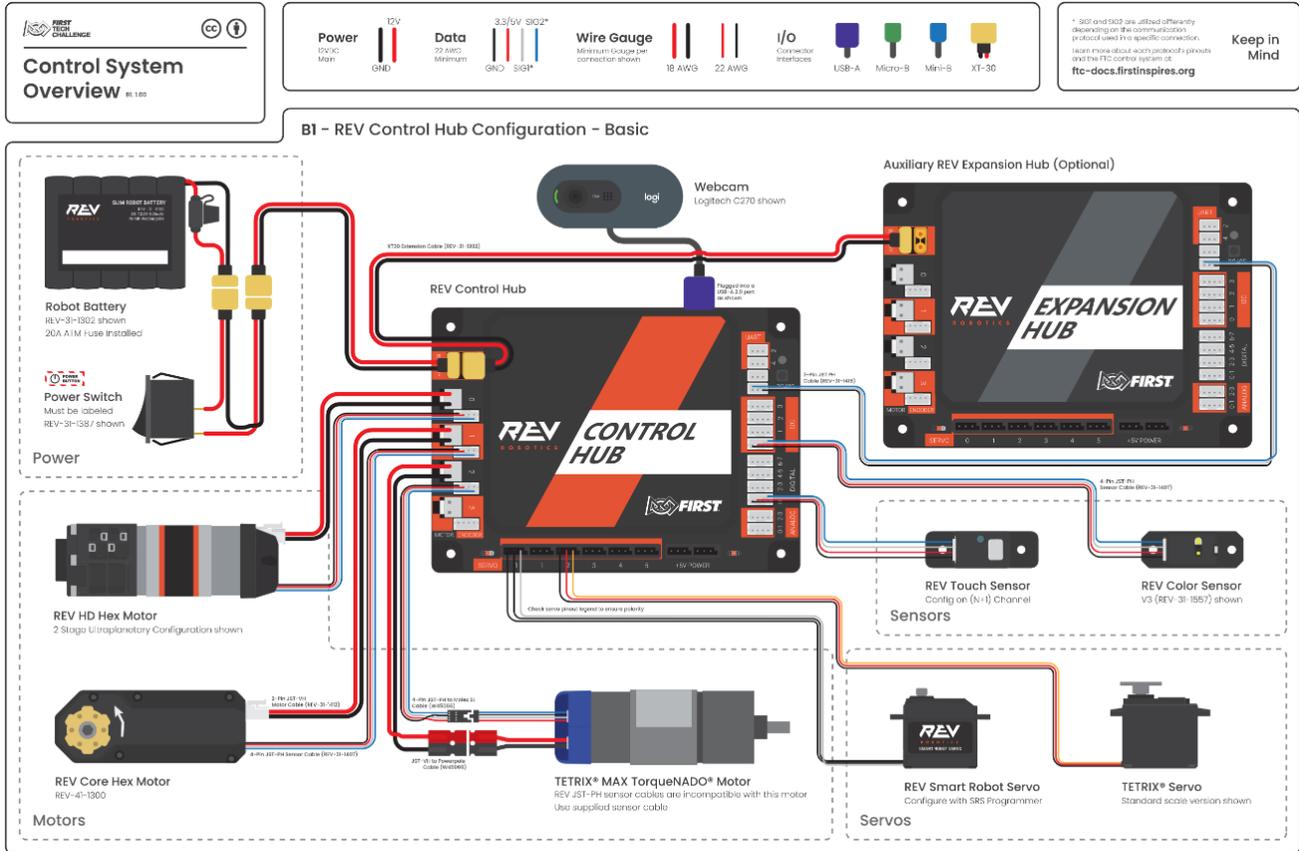
Dans le kit de démarrage de Robotique FIRST France, vous avez 1 capteur :

- Un IMU dans le Control Hub

A vous de voir si vous voulez l'utiliser.

6.5 Câbler le robot

Pour câbler le robot, il faut se référer au schéma suivant :



Dans le kit de démarrage, vous n'avez pas l'Expansion Hub, ni le REV Touch Sensor, ni le REV Color Sensor, ni de Webcam.

Vous pouvez retrouver ce schéma en suivant ce lien : https://ftc-docs.firstinspires.org/control_hard_compon/rc_components/index.html

6.6 La batterie



Voici un lien qui vous explique comment brancher la batterie : [Connecting 12V Power to the Hub – FIRST Tech Challenge Docs 0.1 documentation \(firstinspires.org\)](https://www.firstinspires.org/first-tech-challenge/docs/0.1-documentation)

6.7 Les moteurs



Voici un lien qui vous explique comment brancher un moteur : [Connecting a Motor to the Hub – FIRST Tech Challenge Docs 0.1 documentation \(firstinspires.org\)](https://firstinspires.org/first-tech-challenge/docs/0.1-documentation)

7 Configurer le robot

Pour commencer à utiliser votre robot, il faut configurer certains éléments. Commencez par configurer le Driver Hub.

Pour configurer et programmer le robot, il vous est conseillé d'utiliser l'interface OnBot. Pour cela deux solutions :

- Vous pouvez télécharger un logiciel appelé REV Hardware Client. Ce logiciel vous permet de vous connecter au robot, d'écrire des séries d'instructions et de les envoyer dans le robot. Voici le lien d'installation : [REV Hardware Client Overview - REV Hardware Client \(revrobotics.com\)](#)
- Vous pouvez utiliser l'interface web en vous connectant au robot directement. Voici un lien qui vous explique comment procéder : [Control Hub : Getting Started using a Web Browser - YouTube](#)

7.1 Configurer le Driver Hub

Pour ce faire, il suffit de regarder les deux vidéos YouTube suivantes :

- La première vidéo présente le Driver Hub : [Introducing the REV Driver Hub - YouTube](#)
- La deuxième vidéo explique les étapes de la configuration : [Driver Hub: Getting Started - YouTube](#)

7.2 Configurer le Control Hub

Pour ce faire, il suffit de regarder l'une des deux vidéos suivantes :

- Si vous avez téléchargé le client Hardware REV regardez cette vidéo : [Control Hub: Getting Started using the REV Hardware Client - YouTube](#)
- Si vous avez opté pour la version web, regardez cette vidéo : [Control Hub: Getting Started using a Web Browser - YouTube](#)

7.3 Configurer la connexion entre le Driver Hub et le Control Hub

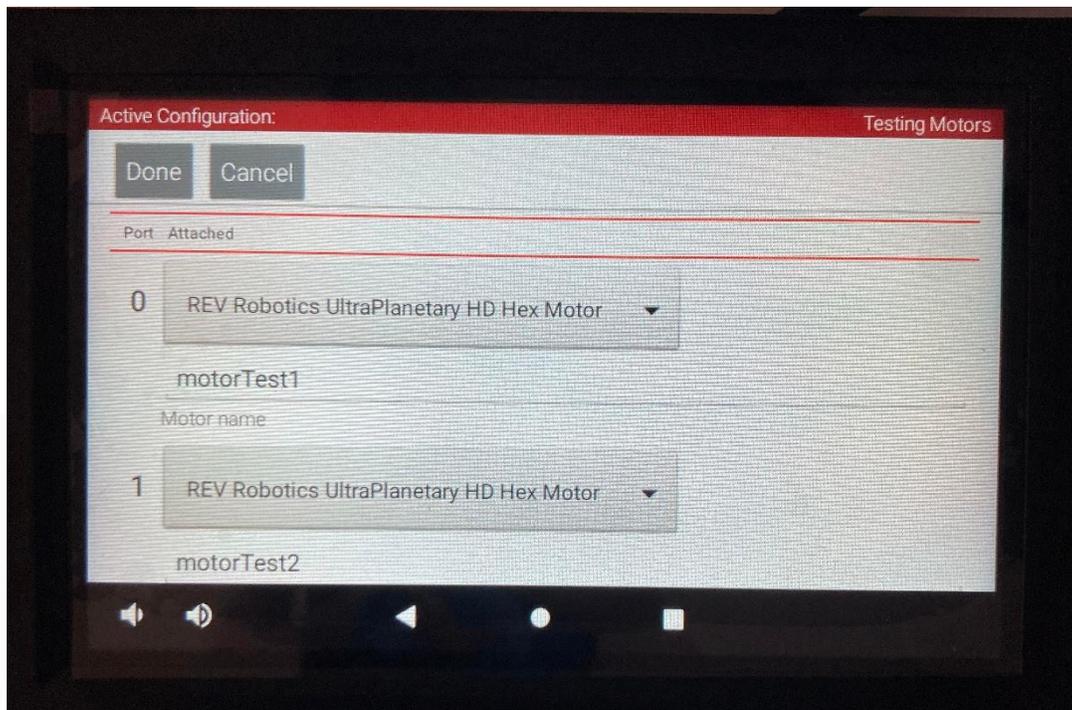
Pour ce faire, il suffit de regarder cette vidéo explicative : [Connecting a Driver Hub to a Control Hub - YouTube](#)

7.4 Configurer les moteurs et les servomoteurs

Pour ce faire, il faut créer un fichier de configuration. Voici les étapes à suivre pour créer un fichier de configuration : [Getting Started – FIRST Tech Challenge Docs 0.1 documentation \(firstinspires.org\)](#)

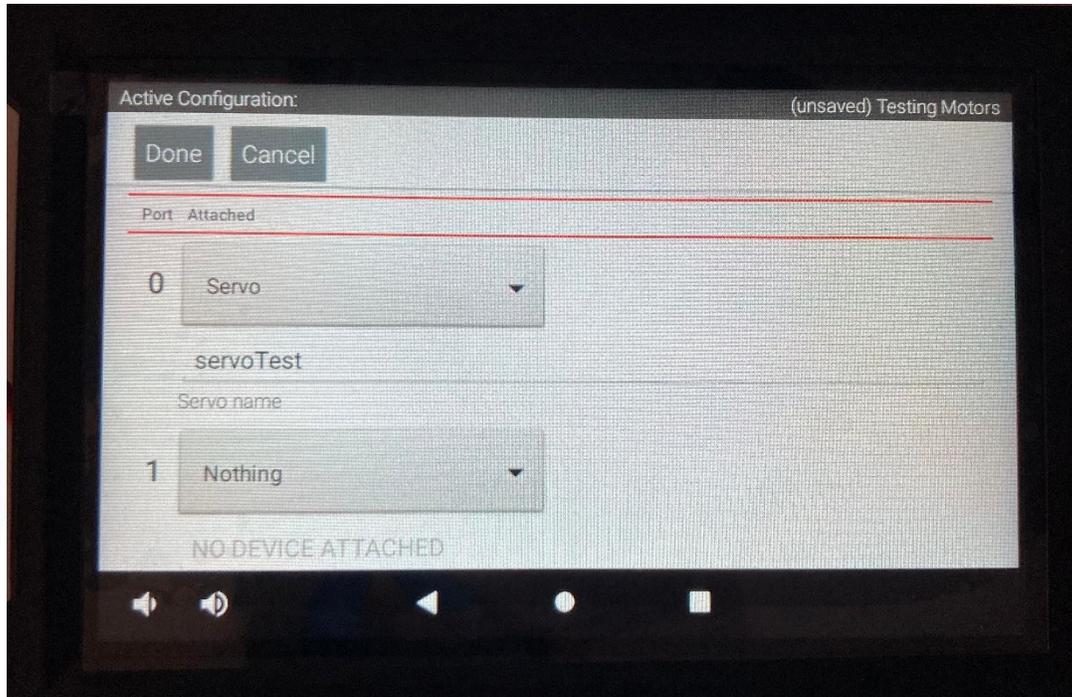
Pour les moteurs, il faut suivre ces étapes : [Configuring a DC Motor – FIRST Tech Challenge Docs 0.1 documentation \(firstinspires.org\)](#)

Voici deux moteur Rev Robotics sur le port 0 et 1 sur l'écran de Driver Hub (à titre d'exemple)



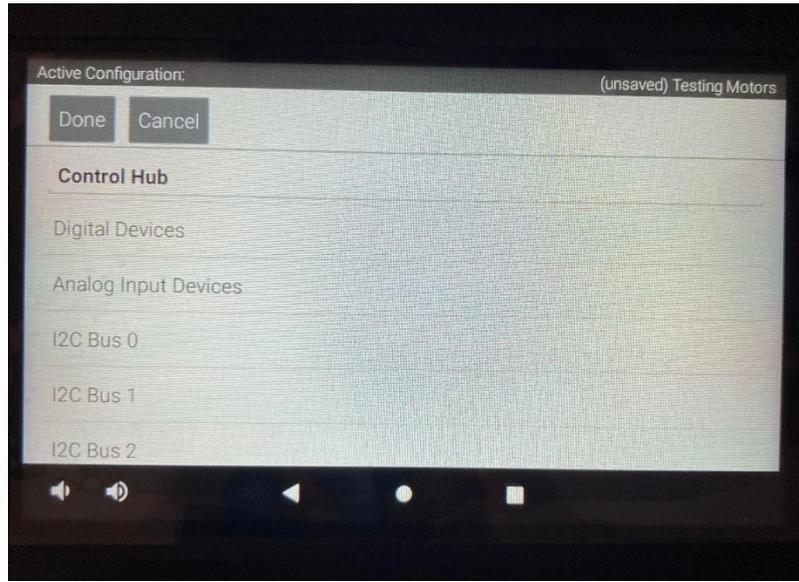
Pour un servo moteur, il faut suivre ces étapes : [Configuring a Servo – FIRST Tech Challenge Docs 0.1 documentation \(firstinspires.org\)](https://firstinspires.org/documentation)

Voici un servo moteur sur le port 0

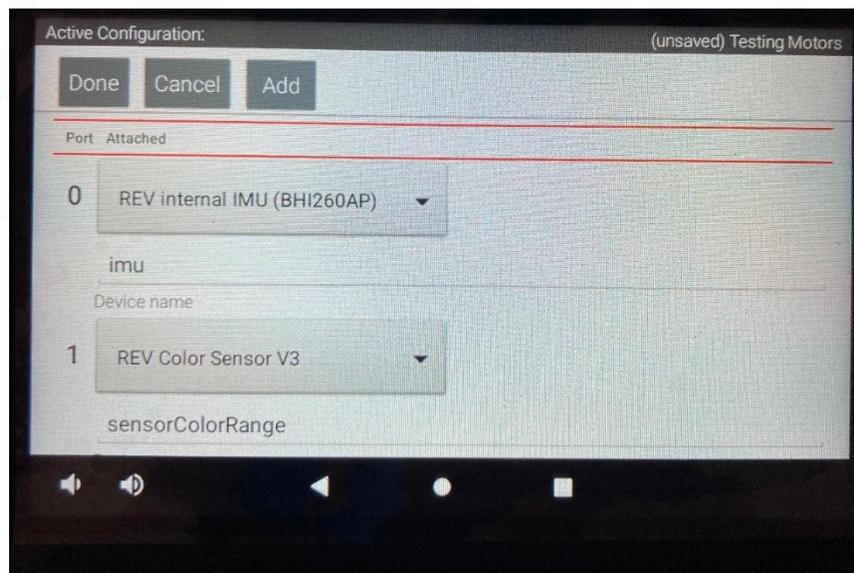


7.5 Configurer les capteurs (seulement celui du kit de démarrage)

Pour ce faire, il faut créer un fichier de configuration comme pour les moteurs si ce n'est pas déjà fait. Si vous n'avez pas créé de fichier de configuration, voici les étapes : [Getting Started – FIRST Tech Challenge Docs 0.1 documentation \(firstinspires.org\)](https://www.firstinspires.org/robotics/first-tech-challenge/docs/0.1-getting-started)



Il faut faire attention de bien aller dans I2C Bus 0



Voici l'IMU sur le port 0

7.7 Sauvegarder une configuration

Après avoir configuré vos éléments, il faut sauvegarder la configuration. Pour ce faire, il faut suivre ces étapes : [Saving the Configuration Information – FIRST Tech Challenge Docs 0.1 documentation \(firstinspires.org\)](https://firstinspires.org/docs/0.1/Saving_the_Configuration_Information)



8 Créer son premier programme en Java

Pour les utilisateurs qui sont sur le client web : il suffit de se connecter à son Control Hub (Wi-Fi).

Pour les utilisateurs de client hardware REV : il suffit d'ouvrir le client et ensuite de se connecter à son Control Hub (Wi-Fi). Puis, vous retournez sur le client et cliquez sur le Control Hub qui devrait apparaître.

Pour la suite des étapes, c'est les mêmes pour tous les types d'utilisateur :

- Cliquez sur l'onglet "OnBotJava »
- Cliquez sur « Program and Manage.»
- Cliquez sur "+"
- Pour ce premier programme, appelez le « MyFIRSTJavaOpMode »
- Dans « sample » sélectionnez « BlankLinearOpMode »
- Cliquez sur Téléop et sur « Setup Code for Configured Hardware »
- Cliquez sur OK

Ce programme ne fait rien pour l'instant mais nous allons le modifier ensemble.

8.1 Mieux comprendre le programme

Pour mieux comprendre ce programme, le lien suivant détaille ligne par ligne la signification du code : [Creating and Running an Op Mode \(OnBot Java\) – FIRST Tech Challenge Docs 0.1 documentation \(firstinspires.org\)](https://www.firstinspires.org/tech-challenge/docs/0.1/creating-and-running-an-op-mode-onbot-java)

8.2 Modifier son programme pour actionner un moteur

Pour actionner un moteur, il faut modifier notre premier programme en suivant les étapes présentées dans ce lien : [Creating and Running an Op Mode \(OnBot Java\) – FIRST Tech Challenge Docs 0.1 documentation \(firstinspires.org\)](https://www.firstinspires.org/tech-challenge/docs/0.1/creating-and-running-an-op-mode-onbot-java)

Program non modifié (ceci servira de base pour la suite du guide) :

```
1  /*
2  Copyright 2023 FIRST Tech Challenge Team FTC
3
4  Permission is hereby granted, free of charge, to any person obtaining a copy of th
5  associated documentation files (the "Software"), to deal in the Software without r
6  including without limitation the rights to use, copy, modify, merge, publish, dist
7  sublicense, and/or sell copies of the Software, and to permit persons to whom the
8  furnished to do so, subject to the following conditions:
9
10 The above copyright notice and this permission notice shall be included in all cop
11 portions of the Software.
12
13 THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED
14 NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
15 NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR
16 DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, A
17 OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFT
18 */
```

```
19 package org.firstinspires.ftc.teamcode;
20
21 import com.qualcomm.robotcore.eventloop.opmode.LinearOpMode;
22 import com.qualcomm.robotcore.hardware.Blinker;
23 import com.qualcomm.robotcore.hardware.DigitalChannel;
24 import com.qualcomm.robotcore.hardware.IMU;
25 import com.qualcomm.robotcore.hardware.ColorSensor;
26 import com.qualcomm.robotcore.hardware.Servo;
27 import com.qualcomm.robotcore.eventloop.opmode.TeleOp;
28 import com.qualcomm.robotcore.eventloop.opmode.Disabled;
29 import com.qualcomm.robotcore.hardware.DcMotor;
30 import com.qualcomm.robotcore.hardware.DcMotorSimple;
31 import com.qualcomm.robotcore.util.ElapsedTime;
32
33 /**
34  * This file contains an minimal example of a Linear "OpMode". An OpMode is a 'program' that runs in either
35  * the autonomous or the teleop period of an FTC match. The names of OpModes appear on the menu
36  * of the FTC Driver Station. When an selection is made from the menu, the corresponding OpMode
37  * class is instantiated on the Robot Controller and executed.
38  *
39  * This particular OpMode just executes a basic Tank Drive Teleop for a PushBot
40  * It includes all the skeletal structure that all linear OpModes contain.
41  *
42  * Remove a @Disabled the on the next line or two (if present) to add this opmode to the Driver Station OpMode list
43  * or add a @Disabled annotation to prevent this OpMode from being added to the Driver Station
44  */
45 @TeleOp
46
47 public class Test extends LinearOpMode {
48     private Blinker control_Hub;
49     private DigitalChannel digitalTouch;
50     private IMU imu;
51     private DcMotor motorTest1;
52     private DcMotor motorTest2;
53     private ColorSensor sensorColorRange;
54     private Servo servoTest;
55
56
57     @Override
58     public void runOpMode() {
59         control_Hub = hardwareMap.get(Blinker.class, "Control Hub");
60         digitalTouch = hardwareMap.get(DigitalChannel.class, "digitalTouch");
61         imu = hardwareMap.get(IMU.class, "imu");
62         motorTest1 = hardwareMap.get(DcMotor.class, "motorTest1");
63         motorTest2 = hardwareMap.get(DcMotor.class, "motorTest2");
64         sensorColorRange = hardwareMap.get(ColorSensor.class, "sensorColorRange");
65         servoTest = hardwareMap.get(Servo.class, "servoTest");
66
67         telemetry.addData("Status", "Initialized");
68         telemetry.update();
69         // Wait for the game to start (driver presses PLAY)
70         waitForStart();
71
72         // run until the end of the match (driver presses STOP)
73         while (opModeIsActive()) {
74             telemetry.addData("Status", "Running");
75             telemetry.update();
76
77         }
78     }
79 }
80
```



Modification à réaliser pour actionner un moteur :

```
// run until the end of the match (driver presses STOP)
double tgtPower = 0;
while (opModeIsActive()) {
    tgtPower = -this.gamepad1.left_stick_y;
    motorTest.setPower(tgtPower);
    telemetry.addData("Target Power", tgtPower);
    telemetry.addData("Motor Power", motorTest.getPower());
    telemetry.addData("Status", "Running");
    telemetry.update();
}
```

8.3 Connecter une manette

Pour connecter une manette afin de contrôler le robot, il faut suivre les étapes présentées dans ce lien : [Creating and Running an Op Mode \(OnBot Java\) – FIRST Tech Challenge Docs 0.1 documentation \(firstinspires.org\)](https://www.firstinspires.org/robotics/first/docs/0.1/creating-and-running-an-op-mode-on-bot-java)

Pour résumer le contenu de ce lien :

- Il faut brancher la manette par un câble USB au Driver Hub
- Pour la première manette : Il faut appuyer sur le bouton start et le bouton A simultanément
- Pour le deuxième pilote : Il faut appuyer sur le bouton start et le bouton B simultanément.



Une fois proprement configuré, vous verrez ce symbole :



Attention, ceci est une vue de téléphone

8.4 Modifier son programme pour actionner un servo moteur

Pour actionner un servo moteur, il faut modifier notre premier programme en suivant les étapes présentées dans ce lien : [Controlling a Servo \(OnBot Java\) – FIRST Tech Challenge Docs 0.1 documentation \(firstinspires.org\)](https://firstinspires.org/fr/robotics/first-tech-challenge/docs/0.1-documentation/controlling-a-servo-on-bot-java)

Modification à réaliser pour actionner un servo moteur :

```
// run until the end of the match (driver presses STOP)
double tgtPower = 0;
while (opModeIsActive()) {
    tgtPower = -this.gamepad1.left_stick_y;
    motorTest.setPower(tgtPower);
    // check to see if we need to move the servo.
    if(gamepad1.y) {
        // move to 0 degrees.
        servoTest.setPosition(0);
    } else if (gamepad1.x || gamepad1.b) {
        // move to 90 degrees.
        servoTest.setPosition(0.5);
    } else if (gamepad1.a) {
        // move to 180 degrees.
        servoTest.setPosition(1);
    }
    telemetry.addData("Servo Position", servoTest.getPosition());
    telemetry.addData("Target Power", tgtPower);
    telemetry.addData("Motor Power", motorTest.getPower());
    telemetry.addData("Status", "Running");
    telemetry.update();
}
```

Attention, si vous cumulez les moteurs et les servo-moteurs, faites attention à ne pas dupliquer le nom des variables (par exemple tgtPower).

8.5 Plus d'informations

Pour avoir plus d'informations sur l'outil OnBot et sur les classes et les fonctions communément utilisé pour la FIRST Tech Challenge, voici un lien avec de la documentation : [OnBot Java Reference Info – FIRST Tech Challenge Docs 0.1 documentation \(firstinspires.org\)](https://firstinspires.org/fr/robotics/first-tech-challenge/docs/0.1-documentation/on-bot-java-reference-info)

9 Créer son premier programme en Block (style Scratch)

Cette partie du guide est inspiré par ce tutoriel suivant : [Tutoriel original](#).

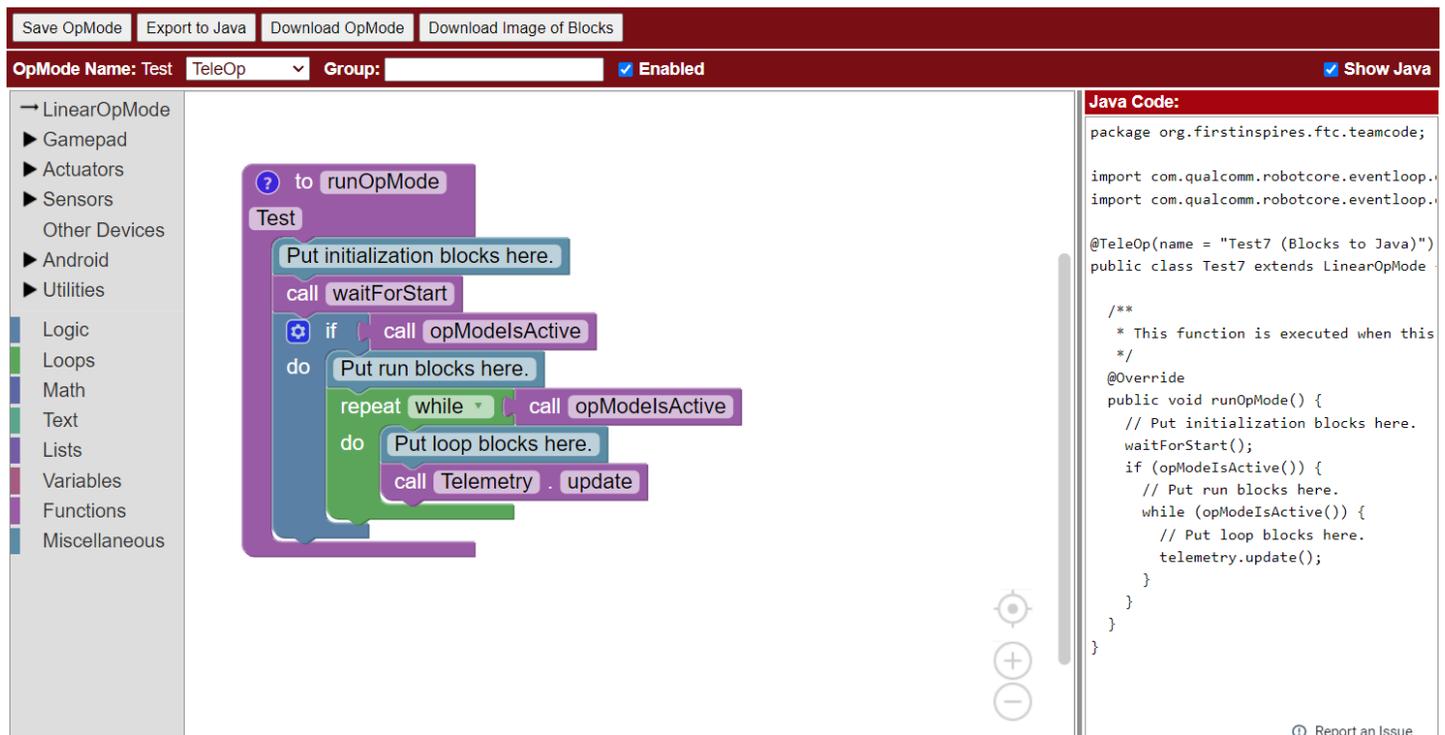
Pour les utilisateurs qui sont sur le client web : il suffit de se connecter à son Control Hub (Wi-Fi).

Pour les utilisateurs de client hardware REV : il suffit d'ouvrir le client et ensuite de se connecter à son Control Hub (Wi-Fi). Puis, vous retournez sur le client et cliquez sur le Control Hub qui devrait apparaître.

Pour la suite des étapes, c'est les mêmes pour tous les types d'utilisateur :

- Cliquez sur l'onglet "Blocks"
- Cliquez sur « Create New Op Mode »
- Pour ce premier programme, appelez le « MyFIRSTOpMode »
- Dans « sample », laissez vide
- Cliquez sur OK

Ce programme ne fait rien pour l'instant mais nous allons le modifier ensemble.



The screenshot shows the FTC Blocks IDE interface. At the top, there are buttons for 'Save OpMode', 'Export to Java', 'Download OpMode', and 'Download Image of Blocks'. Below that, the 'OpMode Name' is 'Test', the mode is 'TeleOp', and it is 'Enabled'. A sidebar on the left lists various block categories like LinearOpMode, Gamepad, Actuators, Sensors, etc. The main workspace contains a block-based program for 'runOpMode' with the following structure:

- Put initialization blocks here.
- call waitForStart
- if opModelsActive (true) do:
 - Put run blocks here.
 - repeat while opModelsActive (true) do:
 - Put loop blocks here.
 - call Telemetry . update

On the right, the 'Java Code' tab is active, showing the following code:

```
package org.firstinspires.ftc.teamcode;

import com.qualcomm.robotcore.eventloop.*;
import com.qualcomm.robotcore.eventloop.*;

@TeleOp(name = "Test7 (Blocks to Java)")
public class Test7 extends LinearOpMode {

    /**
     * This function is executed when this
     */
    @Override
    public void runOpMode() {
        // Put initialization blocks here.
        waitForStart();
        if (opModeIsActive()) {
            // Put run blocks here.
            while (opModeIsActive()) {
                // Put loop blocks here.
                telemetry.update();
            }
        }
    }
}
```

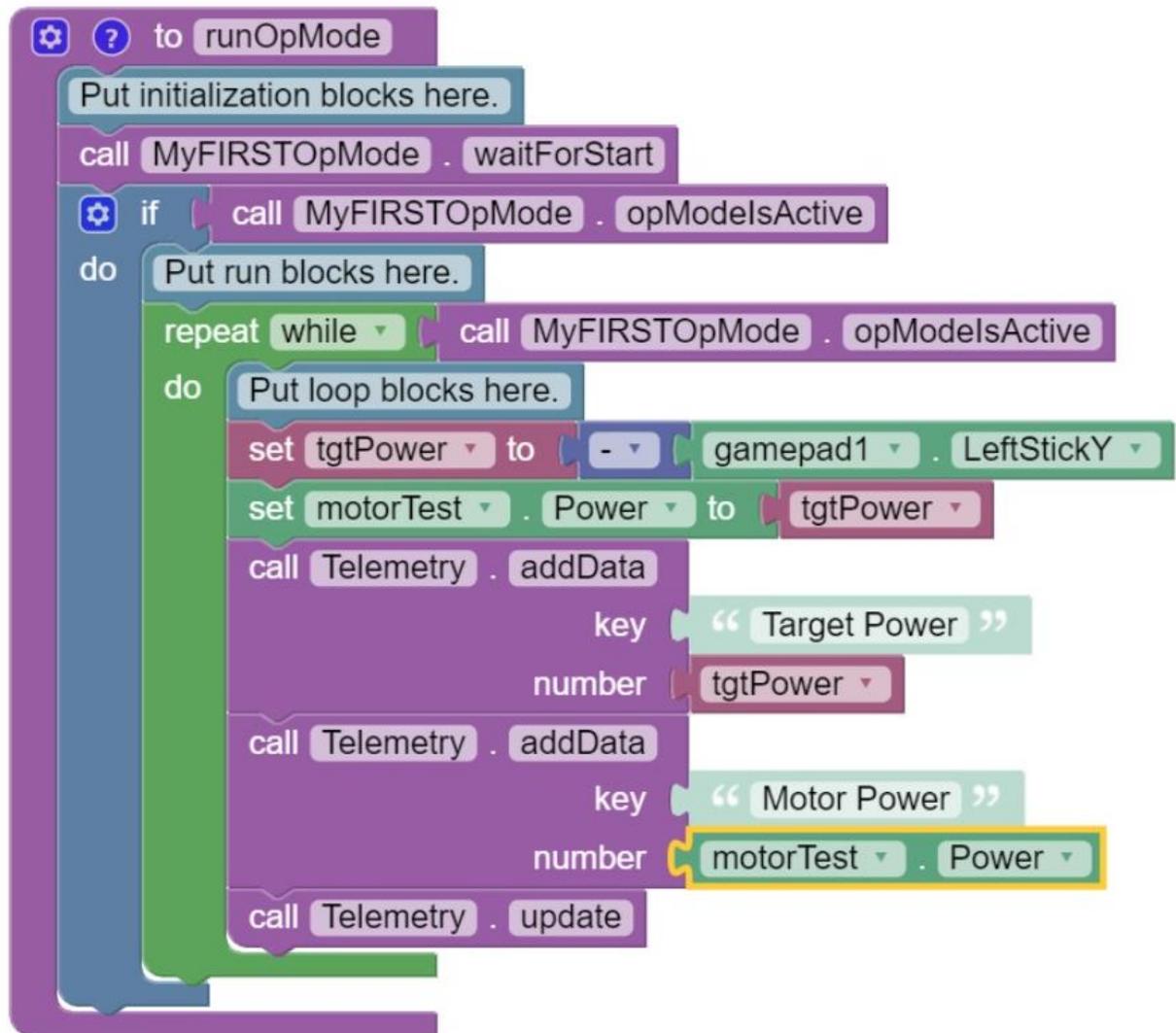
9.1 Mieux comprendre le programme

Pour mieux comprendre ce programme, le lien suivant détaille ligne par ligne la signification du code : https://ftc-docs.firstinspires.org/en/latest/programming_resources/tutorial_specific/blocks/creating_op_modes/Writing-an-Op-Mode-with-FTC-Blocks.html#examining-the-structure-of-your-op-mode

9.2 Modifier son programme pour actionner un moteur

Pour actionner un moteur, il faut modifier notre premier programme en suivant les étapes présentées dans ce lien : [Lien](#).

Programme avec la modification pour actionner un moteur (il servira de base pour la suite du guide) :



```
to runOpMode
  Put initialization blocks here.
  call MyFIRSTOpMode . waitForStart
  if call MyFIRSTOpMode . opModelsActive
  do
    Put run blocks here.
    repeat while call MyFIRSTOpMode . opModelsActive
    do
      Put loop blocks here.
      set tgtPower to - gamepad1 . LeftStickY
      set motorTest . Power to tgtPower
      call Telemetry . addData
        key "Target Power"
        number tgtPower
      call Telemetry . addData
        key "Motor Power"
        number motorTest . Power
      call Telemetry . update
```

9.3 Connecter une manette

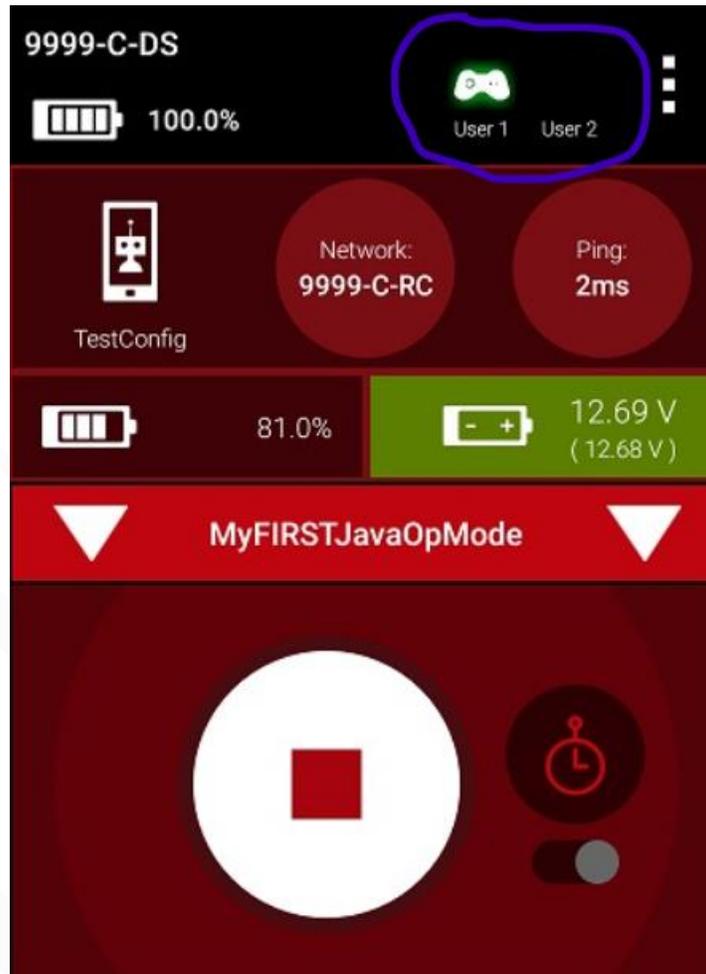
Pour connecter une manette afin de contrôler le robot, il faut suivre les étapes présentées dans ce lien : [Creating and Running an Op Mode \(OnBot Java\) – FIRST Tech Challenge Docs 0.1 documentation \(firstinspires.org\)](https://www.firstinspires.org/tech-challenge/docs/0.1/creating-and-running-an-op-mode-on-bot-java)

Pour résumer le contenu de ce lien :

- Il faut brancher la manette par un câble USB au Driver Hub
- Pour la première manette : Il faut appuyer sur le bouton start et le bouton A simultanément
- Pour le deuxième pilote : Il faut appuyer sur le bouton start et le bouton B simultanément.



Une fois proprement configuré, vous verrez ce symbole :



Attention, ceci est une vue de téléphone

9.4 Modifier son programme pour actionner un servo moteur

Pour actionner un servo moteur, il faut modifier notre premier programme en suivant les étapes présentées dans ce lien : [Lien](#).

Modification à réaliser pour actionner un servo moteur :

Attention, si vous cumulez les moteurs et les servo-moteurs, faites attention à ne pas dupliquer le nom des variables (par exemple tgtPower).

9.5 Plus d'informations

Pour avoir plus d'informations sur l'outil OnBot et sur les classes et les fonctions communément utilisé pour la FIRST Tech Challenge, voici un lien avec de la documentation : [Lien](#).

10 Programmation Virtuelle

Il existe la possibilité de programmer de manière virtuelle pour permettre à toute l'équipe de développer ses compétences simultanément.

La programmation virtuelle d'un robot FIRST Tech Challenge (FTC) se fait sur un site créé par FIRST Canada. Il suffit d'aller sur le site internet pour pouvoir commencer à programmer. Le Java et le block sont disponible.

Voici le lien : <https://ftcsim.org/>

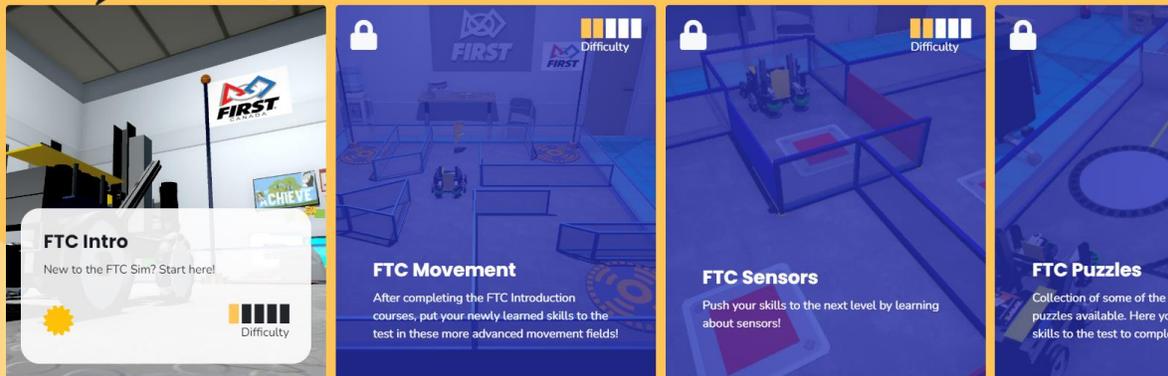
A l'heure actuelle ce site est proposé seulement en anglais.

Pour vous accompagner, il existe 5 cursus qui permettrons à l'équipe d'apprendre à programmer le robot virtuellement.



La page d'accueil du site internet

Start Your *FIRST* Robotics Adventure Now



Les différents cursus sont accessibles en bas de de la page d'accueil.

10 Conclusion

Une fois toutes les étapes de ce guide complété, vous serez passé d'un kit de démarrage dans une caisse (ou plusieurs caisses) à un robot fonctionnel qui se déplace sur le terrain et qui peut potentiellement marquer des points.

Ceci n'est pas la fin, loin de là. L'objectif est que vous puissiez utiliser ce guide comme d'une première marche à gravir pour construire le robot le plus performant possible.

Enfin, le but final est bien sûr de permettre aux jeunes de découvrir le monde des sciences et des technologies et de les inspirer à être curieux, repousser leurs limites et progresser.

N'hésitez surtout pas à contacter votre conseiller technique pour qu'il puisse travailler avec vous sur la compréhension de ce guide si besoin ou tout autres questions.

Toute l'équipe de Robotique FIRST France vous remercie d'avoir lu ce guide jusqu'à la fin.