

TECHNOLOGIE



Projet de 3ème

Groupe 6 : Mia, Charlotte, Lucie, Matteo G., Noam, Emma, Juliette J.

Présentation du projet

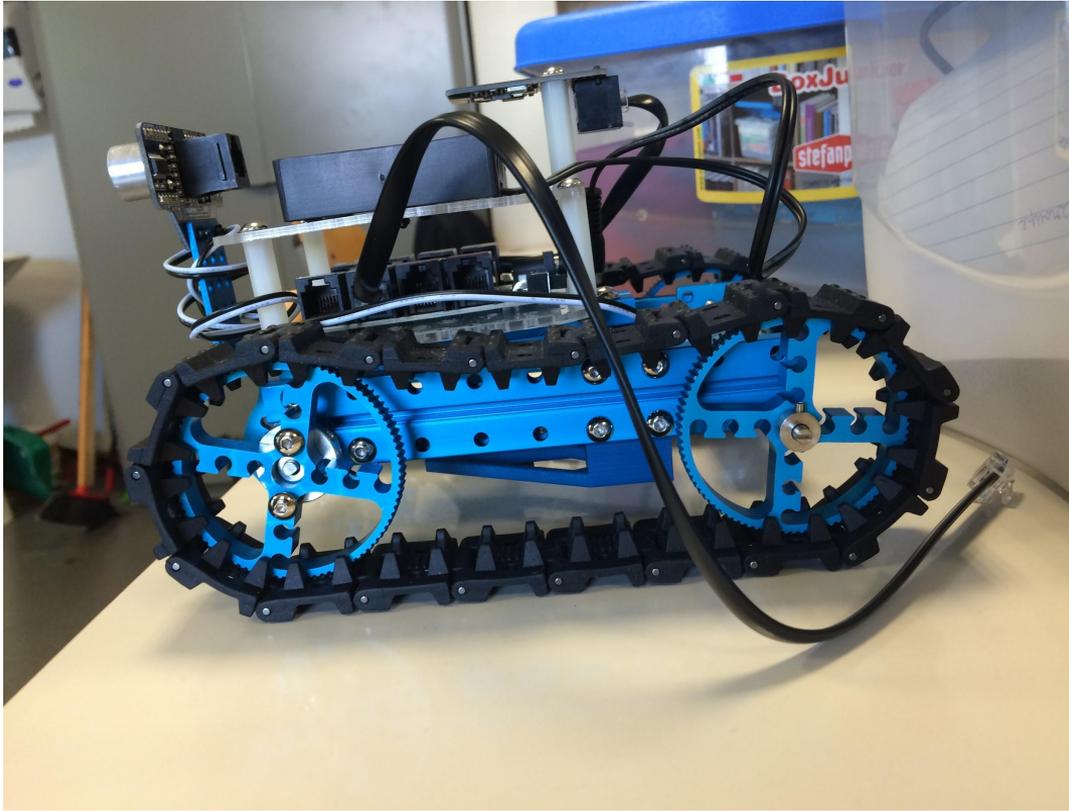
Nous avons tous réalisés pendant plusieurs mois une voiture ayant certaines capacités.

Nous avons plusieurs objectifs que nous allons ensuite vous expliquer, au fur et à mesure de l'exposé.

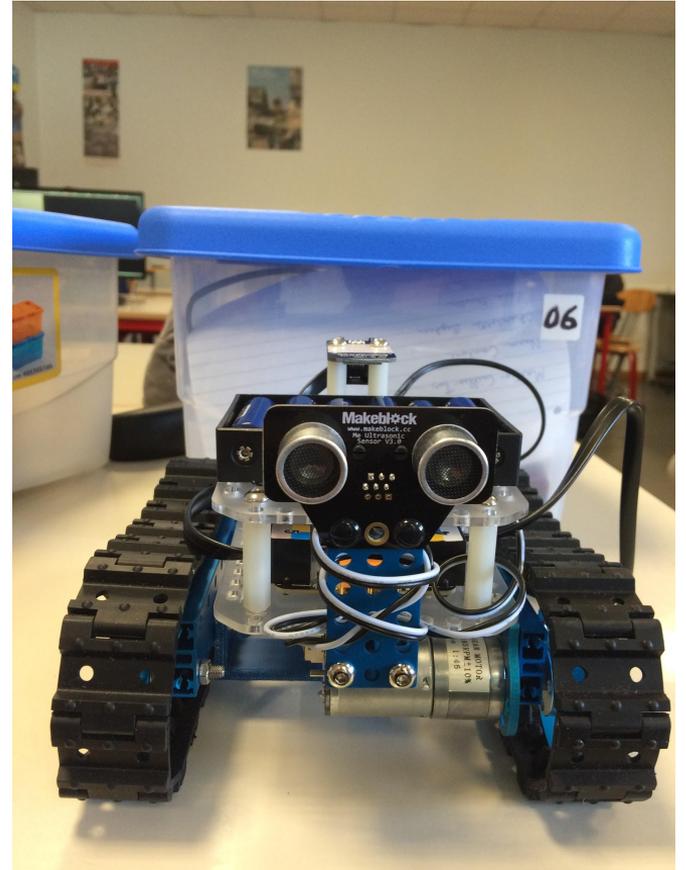
Chacun avait une tâche différente, Matteo et Charlotte s'occupaient de Sketchup pour pouvoir dessiner et construire les pièces à rajouter sur notre voiture, ensuite Mia et Lucie se sont occupés de App Inventor pour lancer les programmes pour la voiture, Juliette et Emma travaillaient sur MBlock pour créer les programmes et les indications que la voiture devait suivre et Noam c'est occupé de App Inventor et MBlock en aidant les personnes concernées et en travaillant sur son projet personnel.

Au fil de ces derniers mois nous avons réussi à créer différentes pièces et différents programmes afin que la voiture fonctionne comme il se doit, nous vous laissons alors avec des explications plus approfondies.

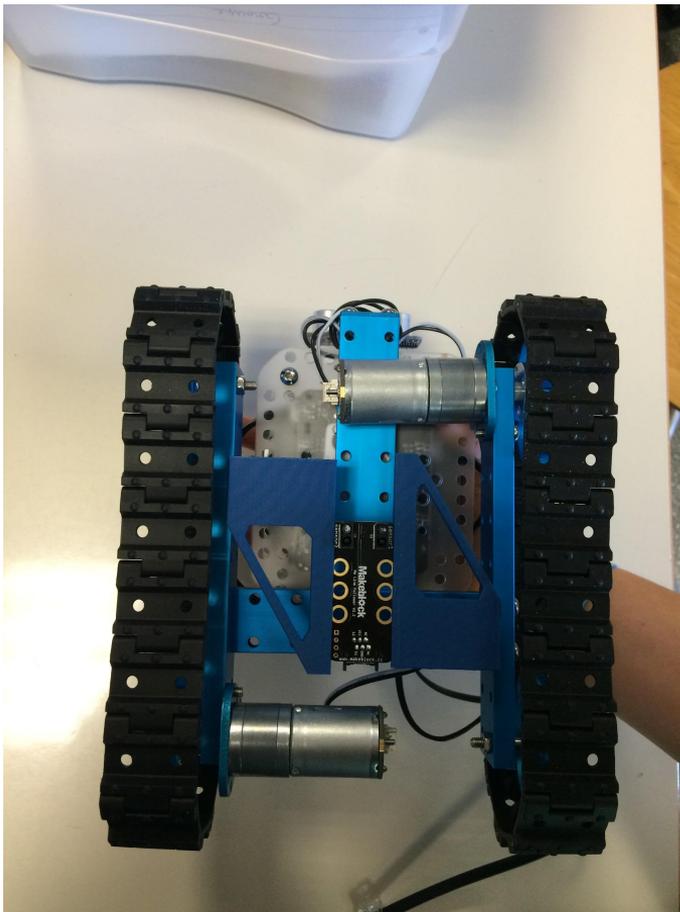
Quelques images de notre voiture



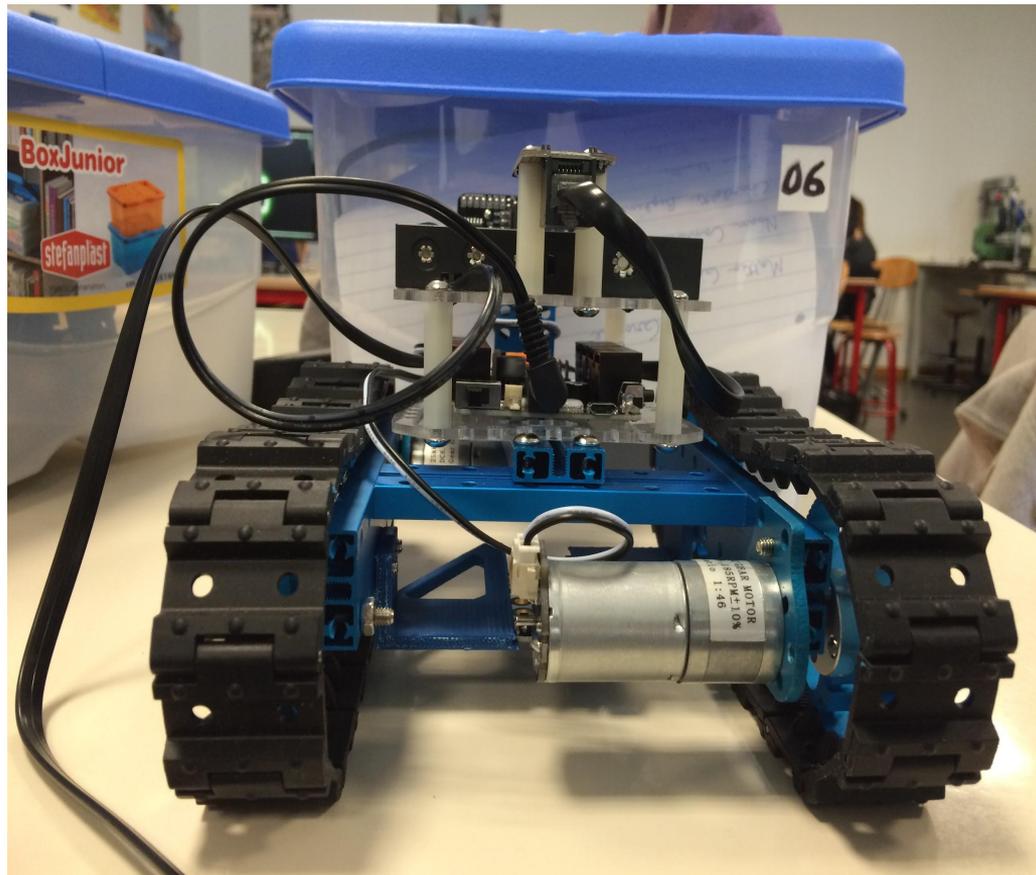
Vue de côté



Vue de devant



Vue d'en dessous



Vue de derrière

APPINVENTOR

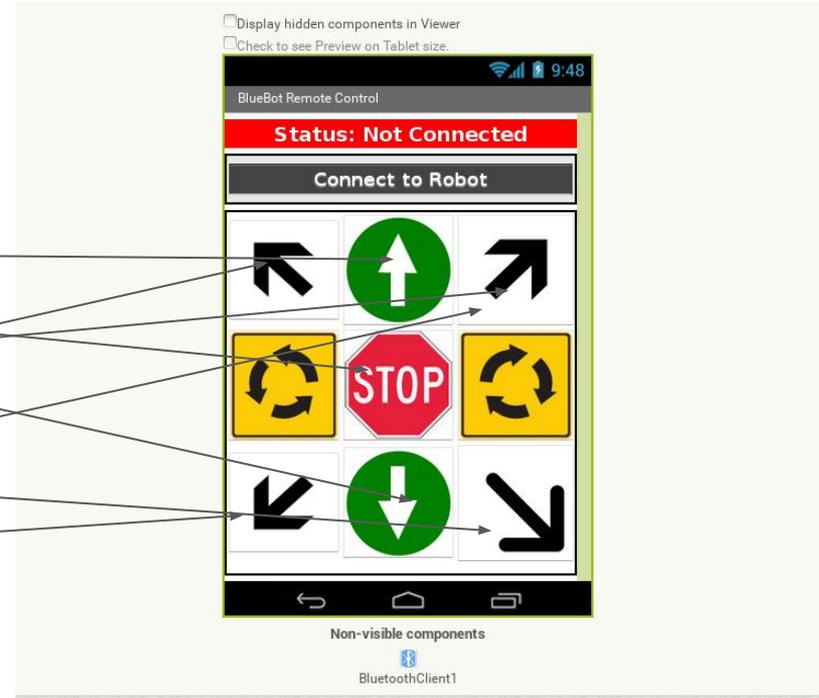
Objectifs atteint:

Ajouter la possibilité de déplacer le véhicule seulement si le doigt est maintenu sur le bouton correspondant, et s'arrêter dès que le doigt est levé,

- **ajouter dans les quatre angles de l'interface des boutons supplémentaires**
- flèche pour avancer en tournant à gauche -> renvoyer le nombre 6 à la voiture
 - flèche pour avancer en tournant à droite -> renvoyer le nombre 7 à la voiture
 - flèche pour reculer en tournant à gauche -> renvoyer le nombre 8 à la voiture
 - flèche pour reculer en tournant à droite -> renvoyer le nombre 9 à la voiture

APPINVENTOR

```
when btnStop .Click
do
  call BluetoothClient1 .Disconnect
  set lblConnectionStatus .Text to " Status:Connected "
  set lstPairedBluetoothDevices .Visible to true
  set lstPairedBluetoothDevices .Enabled to true
  set btnForward .Enabled to false
  set btnStop .Enabled to false
  set btnReverse .Enabled to false
  set btnRight .Enabled to false
  set btnLeft .Enabled to false
  set ul .Enabled to false
  set dr .Enabled to false
  set ur .Enabled to false
  set dl .Enabled to false
```



APPIVENTOR

CE QU'ON A AJOUTÉ:

On a fini de programmer les deux modes de déconnection de la liaison Bluetooth :

- Ajouter la possibilité de déconnecter le Bluetooth, pour pouvoir ensuite se reconnecter.
- Faire en sorte que lorsque l'on ne fait rien pendant plus de 30 secondes, la voiture se déconnecte.

On a créé un nouveau programme:

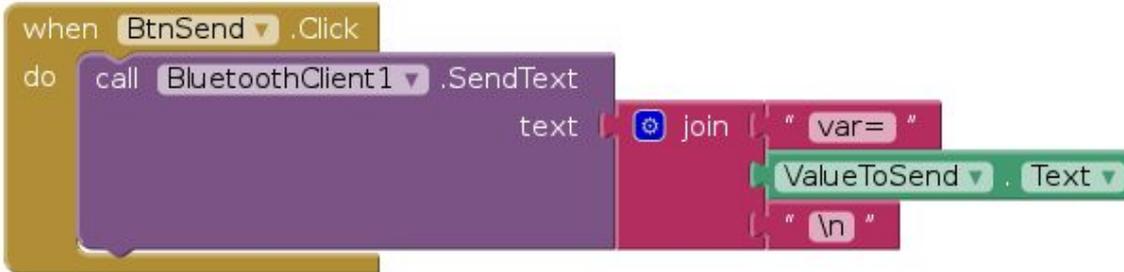
Le programme de base vous est fourni :

- Une liste de bouton de connexion - déconnexion,
- Une case de saisie de valeur,
- Un bouton "Envoi d'une valeur",
- Un label d'affichage de la valeur reçue.

Il faut implémenter les fonctions suivantes :

- > envoyer la valeur saisie (ou sinon rien) lorsque l'on appuie sur le bouton
- > recevoir une valeur toutes les secondes, en testant d'abord si l'on a reçu quelque chose !

NOUVEAU PROGRAMME



MBlock

Objectifs:

- dans le programme Projet_Demo.sb2, implémenter pour les valeurs de rx suivantes le déplacement du tank indiqué :
 - 6 -> avancer en tournant à gauche
 - 7 -> avancer en tournant à droite
 - 8 -> reculer en tournant à gauche
 - 9 -> reculer en tournant à droite
- intégrer la prise en charge du capteur d'ultrasons, pour que la voiture s'arrête d'avancer **lorsqu'un obstacle se trouve à moins de 40 cm (elle ne peut alors que reculer !)**
- ensuite faire en sorte que la voiture ne puisse à nouveau avancer que **lorsqu'il n'y a pas d'obstacle à moins de 50cm (cela crée une "hystérésis" avec les 40cm)**
- **si l'obstacle se trouve à moins de 20 cm**, la voiture recule pour que **l'obstacle se trouve à 40 cm** (si l'obstacle disparaît, elle repart)
- afficher dans l'interface AppInventor la distance lue par le sonar du véhicule
- envoyer un nombre toutes les 2 secondes : compteur (envoyer 1 puis 2 puis 3 puis...)
- si l'on reçoit un nombre d'Appinventor2, le mettre dans le compteur (le programme se met à compter à partir de ce nombre) !
- Faire un programme très simple qui avance doucement, et va droit, tourne à droite ou à gauche selon ce que voit la carte de suivi de ligne

MBlock - Premier programme

```
Orion - générer le code
répéter indéfiniment
mettre sonar à distance mesurée par le capteur ultrasons du Port3
si distance mesurée par le capteur ultrasons du Port3 < 20 alors
  activer le moteur M1 à la puissance -100
  activer le moteur M2 à la puissance -100
attendre jusqu'à distance mesurée par le capteur ultrasons du Port3 > 40
  activer le moteur M1 à la puissance 0
  activer le moteur M2 à la puissance 0
si sonar < 40 et sonar > 0 alors
  si rx = 1 ou rx = 6 ou rx = 7 alors
    activer le moteur M1 à la puissance 0
    activer le moteur M2 à la puissance 0
  si undefined alors
    mettre rx à undefined var
    si rx = 1 alors
      activer le moteur M1 à la puissance 100
      activer le moteur M2 à la puissance 100
    sinon
      si rx = 2 alors
        activer le moteur M1 à la puissance -100
        activer le moteur M2 à la puissance -100
    sinon
```

Reculer lorsque ...

S'arrêter lorsque ...

S'arrêter lorsque ...

Avancer

Reculer

Tourner à gauche

Tourner à droite

S'arrêter

Tourner à droite en avançant

Tourner à gauche en avançant

Tourner à gauche en reculant

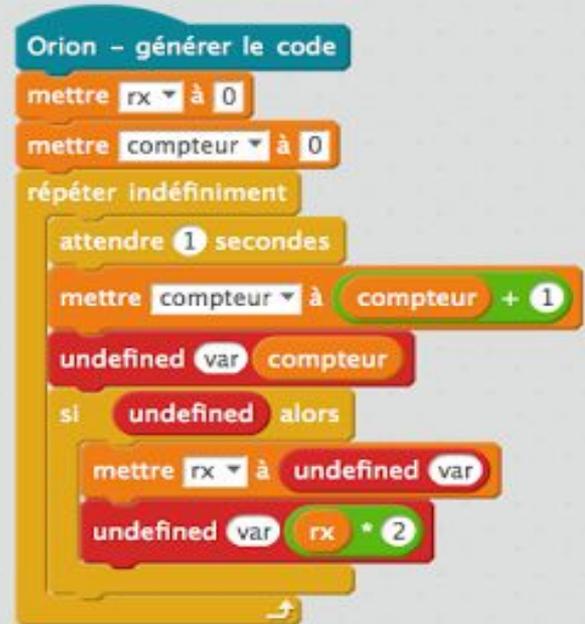
Tourner à droite en reculant

```
si rx = 3 alors
  activer le moteur M1 à la puissance -100
  activer le moteur M2 à la puissance 100
sinon
  si rx = 4 alors
    activer le moteur M1 à la puissance 100
    activer le moteur M2 à la puissance -100
  sinon
    si rx = 5 alors
      activer le moteur M1 à la puissance 0
      activer le moteur M2 à la puissance 0
    si rx = 6 alors
      activer le moteur M1 à la puissance 100
      activer le moteur M2 à la puissance 50
    si rx = 7 alors
      activer le moteur M1 à la puissance 50
      activer le moteur M2 à la puissance 100
    si rx = 8 alors
      activer le moteur M1 à la puissance -100
      activer le moteur M2 à la puissance -50
    si rx = 9 alors
      activer le moteur M1 à la puissance -50
      activer le moteur M2 à la puissance -100
```

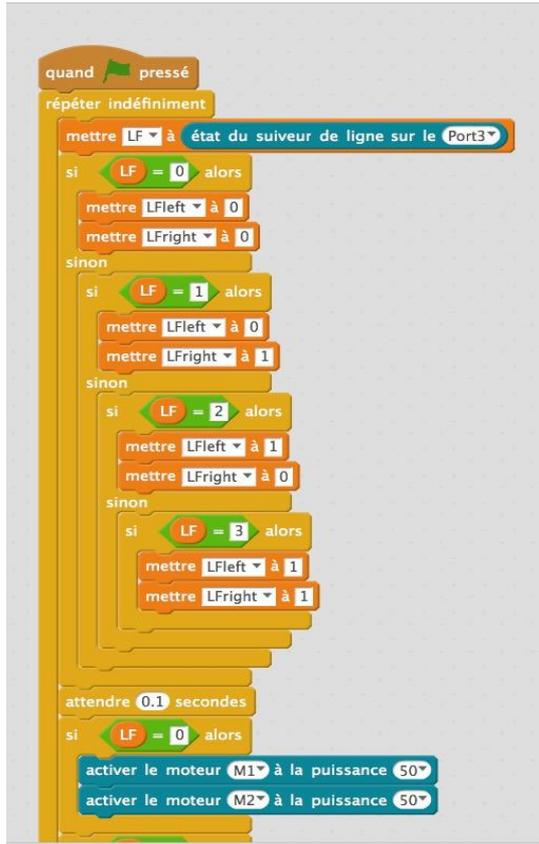
Programme envoyant les données à Appinventor par Bluetooth

Ce programme sert envoyer et récupérer des données en Bluetooth.

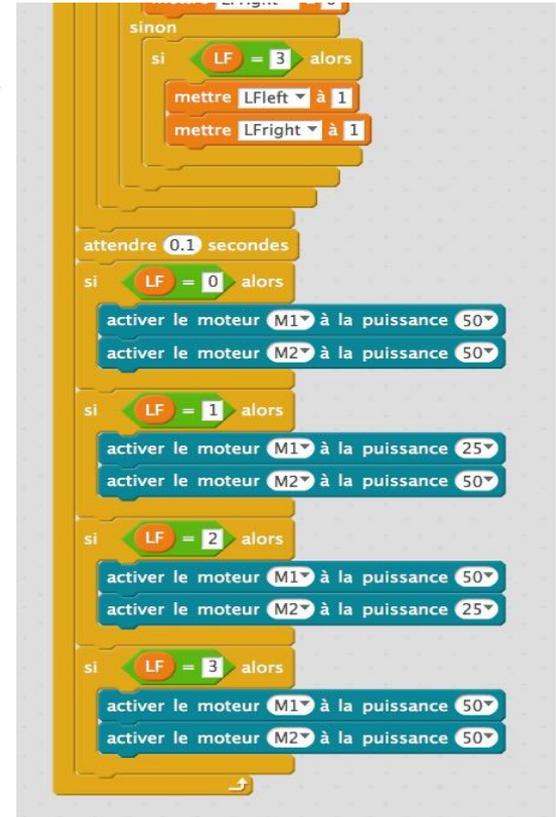
Il envoie chaque 2 secondes un nombre à travers le compteur. Ce nombre est vu sur la tablette. Si l'on reçoit un nombre d'Appinventor, il faut que le compteur se mette à jour et affiche ce nombre-là pour après repartir du nombre.



MBlock - Deuxième programme

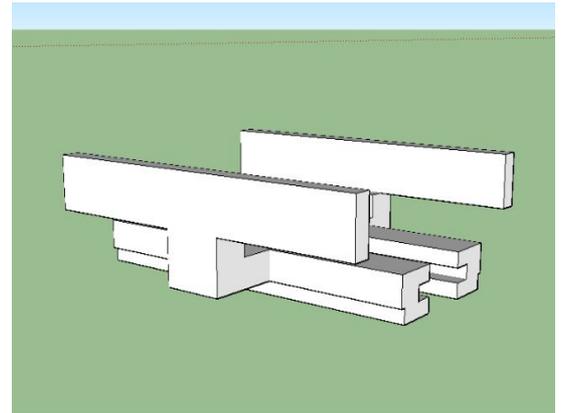
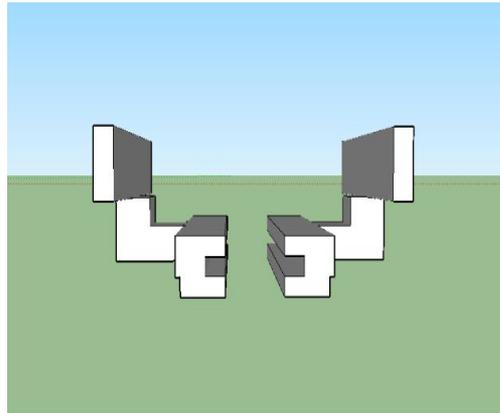
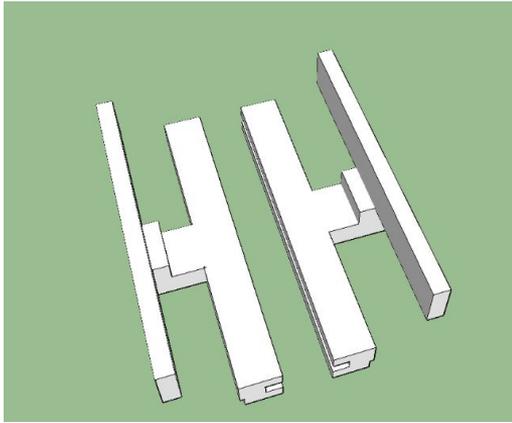


Ce programme sert à faire avancer, tourner à droite et tourner à gauche la voiture en fonction de si elle voit du blanc ou du noir. En effet, la première partie du programme consiste dans le fait qu'elle "voie" du blanc ou du noir. La deuxième partie sert justement à la faire tourner à droite ou à gauche afin qu'elle reste sur le noir. Donc, si elle voit du blanc, elle va tourner dans un des deux sens pour que ses capteurs ne voient plus de blanc.



Sketchup

Pour la partie Sketchup nous avons avant tout réalisé des rails qui permettent à la carte de suivie de ligne de pouvoir coulisser.



Ensuite nous avons réfléchis à un moyen pour protéger la voiture. Notre première idée fut le couvre-chenilles, mais après les avoir terminés et après quelques moments de réflexion nous nous sommes rendu compte qu'il lui fallait une armure intégrale.

En étant en collaboration avec Noam j'ai du dessiner un support pour hélices qui permettait à la voiture de voler.

L'application de Noam.

Ce que j'ai ajouté

1-Un système de mot de passe: J'ai intégré un système d'authentification d'identité, pour éviter que vous, petits farfadets maléfiques des autres groupes, accédiez à l'application ou à la voiture. Le mot de passe de l'application est sur le premier écran, il y a aussi quelque conditions présentées sous formes de cases à cocher. Si une case n'est pas cochée et donc une condition non respectée ou le mot de passe est erroné il s'en suivra une sanction pour vos tympans, maintenant vous ne pouvez pas dire que l'on ne vous a point avertis. Le mot de passe de la voiture est un code envoyé par l'application lorsque le bouton décoré d'une icône plate de cadenas est pressé. Si ce code n'est pas reçu, la voiture ignorera toute les commandes ou informations reçues par la suite .

2- Une organisation et un aspect différent : En voulant garder plus d'options pour contrôler la voiture j'ai fait un autre écran. Bien que les boutons aient les même images que ceux l'autre écrans avec des couleurs moins lourdes, les deux flèches verticales activent d'autre commandes soit l'accélération et la décélération. Chaque fois que vous cliquerez sur la flèche avant, soit l'accélération, la voiture ira plus vite de $\frac{1}{5}$ soit du 20%, ce qui veut dire que vous devrez cliquer 5 fois sur la flèche pour arriver au maximum de la vitesse. La flèche arrière soit la décélération fait baisser la vitesse de $\frac{1}{5}$ si l'on est à 0 il faudra appuyer 5 fois pour arriver au maximum de la marche arrière. Cet écran va être doté de la vue du robot une fois que je trouverais la caméra appropriée.

J'ai ajouté a l'écran dont je parlais avant une barre vitesse qui augmente et baisse avec la vitesse de la voiture. J'ai opté pour un aspect de l'application plus sobre et plus professionnelle j'ai donc remplacé toute les icônes d'avant par des icônes plates et en noir, gris, blanc et seulement pour la barre de vitesse, bleue. J'ai fait un quatrième écran, celui de l'intelligence artificielle, sur cet écran j'y ai mis le boutons activant le suivis de ligne et le bouton activant l'autogestion.

Ce que je vais faire :

- Changer quelques composants de la voiture
- Rajouter des hélices (peut-être)
- Créer une boîte de dialogue avec le robot
- Faire les tutoriels
- Ajouter un turbo à la voiture
- Ajouter des microphones à la voiture

Merci de votre
écoute et de votre
attention!