

MASTERMIND ARDUINO

RAPPORT DE SÉANCE : Semaine du 17 février

Objectif de la séance :

- Réalisation boîte en bois
- Programmation du bluetooth
- Finalisation de l'application en ligne

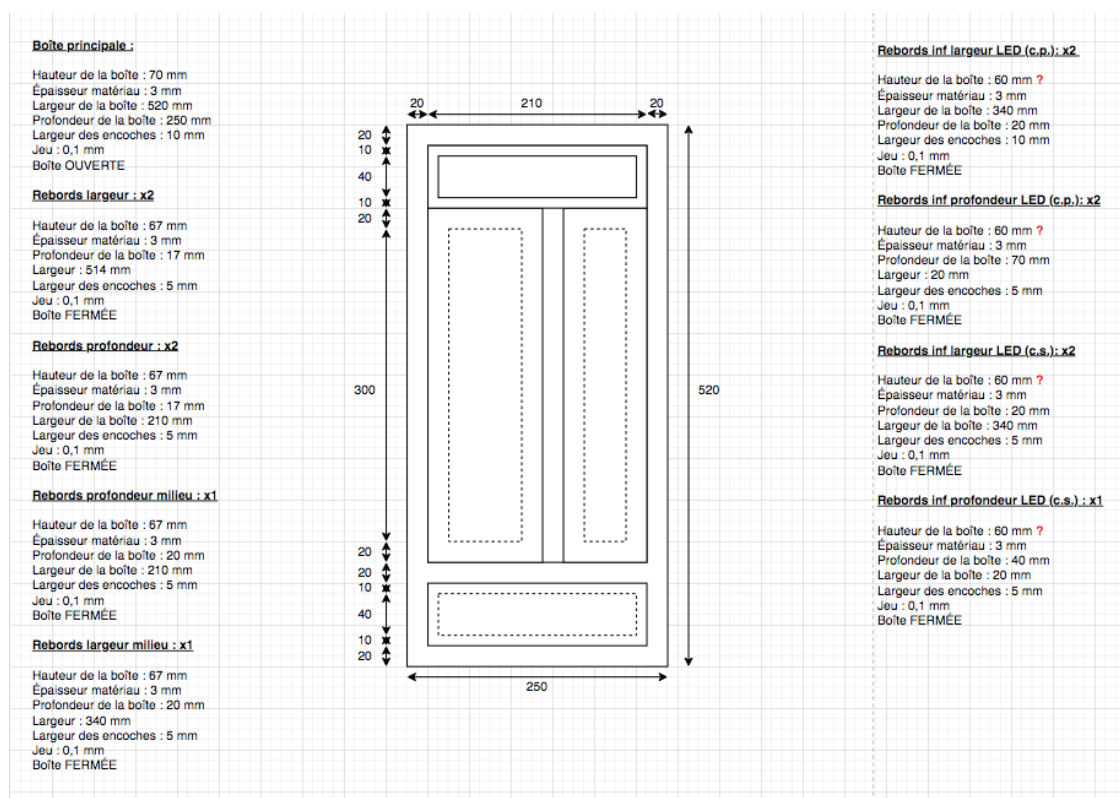
1) RÉALISATION BOÎTE EN BOIS

Notre projet final sera réalisé sous forme d'une **boîte en bois qui contiendra tous les éléments du jeu** (interface visible par le joueur et composants Arduino et électronique cachés à l'intérieur de la caisse).

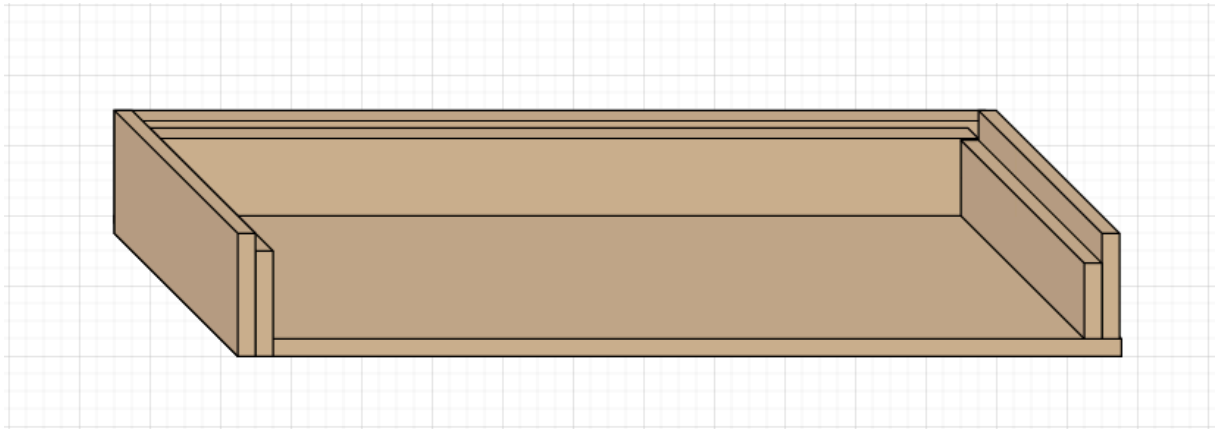
Nous nous sommes rendus au FABLAB pour débiter la fabrication de la caisse finale, mais pour cela il fallait **remesurer toutes nos composantes afin d'avoir les bonnes dimensions et les découpes les plus précises possibles.**

On avait initialement prévu **une caisse principale de 2cm d'épaisseur**, sauf que le seul bois disponible au FABLAB était d'une épaisseur de 3mm, il a donc fallu **fabriquer des boîtes de 1,7cm de profondeur que l'on va venir coller dans notre boîte pour obtenir notre résultat d'une boîte de 2cm d'épaisseur.**

Cela rajoute beaucoup de travail car chaque face de la boîte et des composantes représente de nouvelles boîtes « rebords » à construire. On réalise un schéma précis reprenant toutes les mesures pour faciliter les calculs et les fabrications des boîtes en bois :

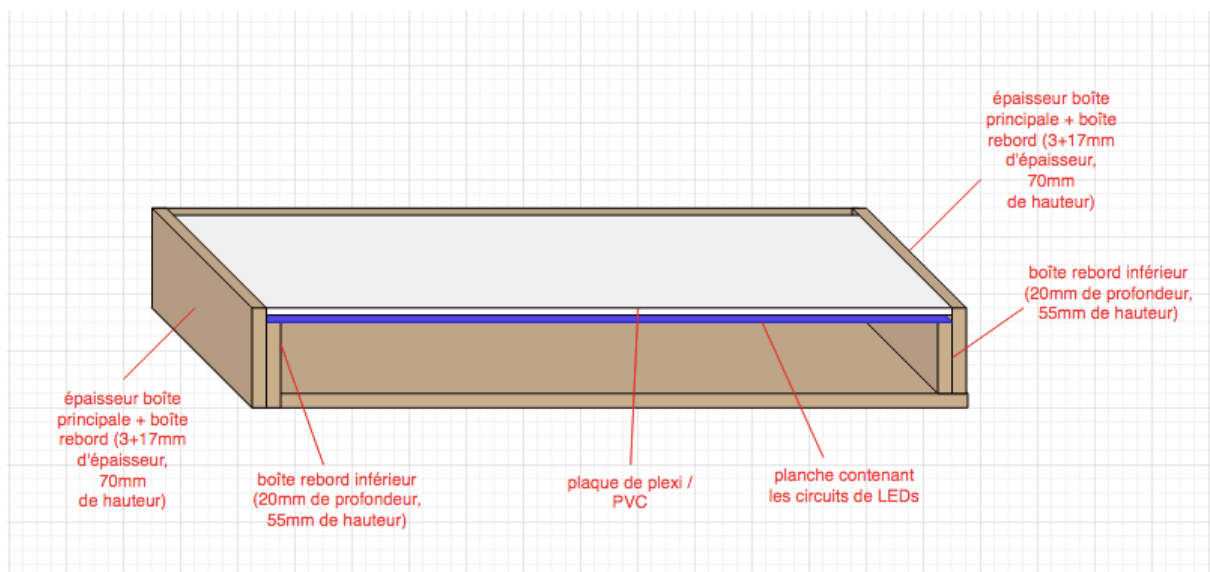


Les traits représentés en pointillés sur le schéma ci-dessus sont en fait des boîtes « rebords » de hauteur légèrement plus petite que les rebords de la boîte principale (55mm de hauteur contre 70mm),



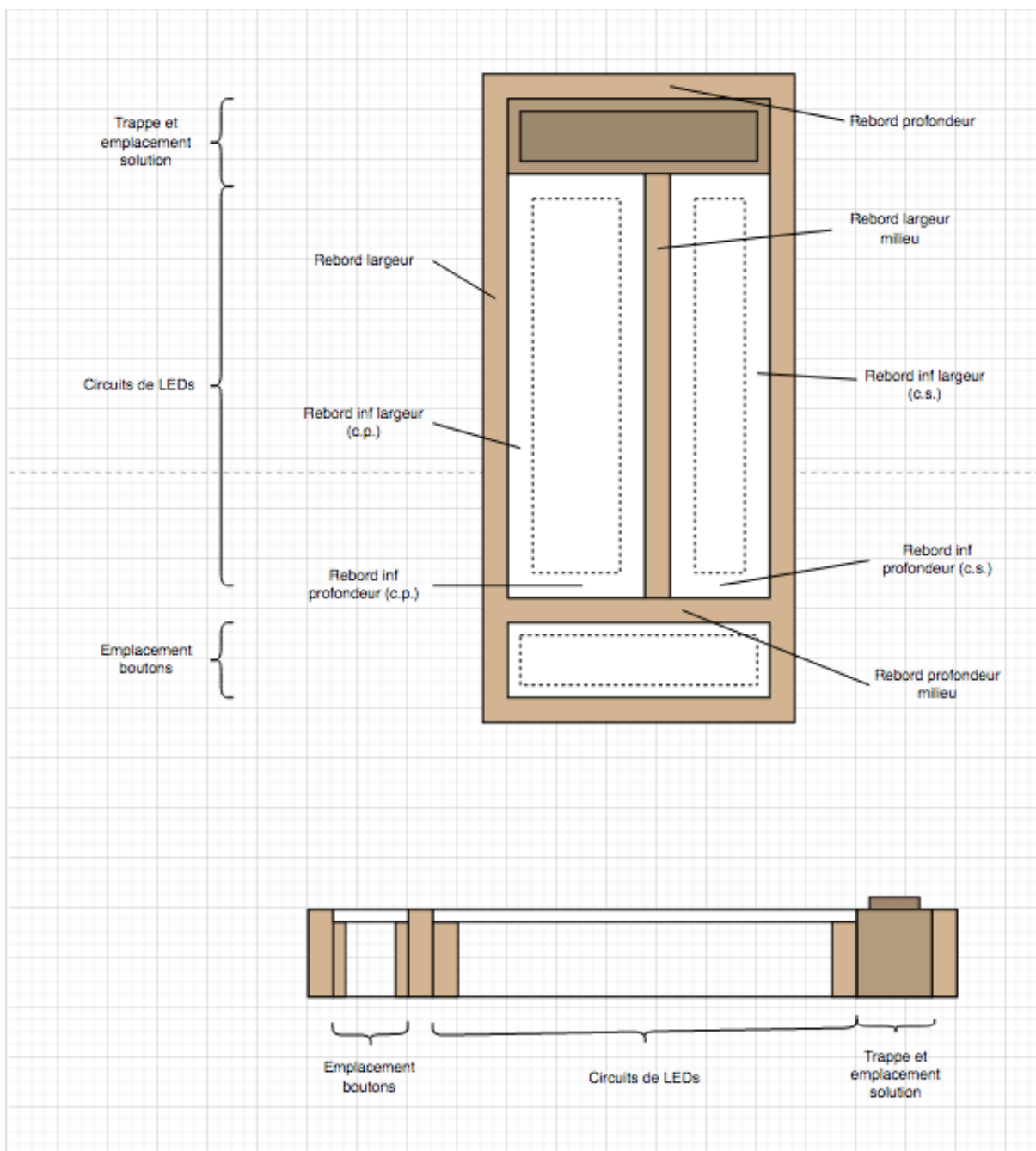
Ils permettront de simplement venir poser la plaque en bois contenant les LEDs ainsi que la plaque de PVC qui recouvre les LEDs,

Cette forme d'escalier permet d'avoir un résultat stable sans avoir à fixer la planche des LEDs+PVC au reste de la boîte (plus facile à manipuler et à ouvrir en cas de réglage à faire à l'intérieur de la boîte puisque le couvercle sera enclenché dans la boîte sans être fixé)



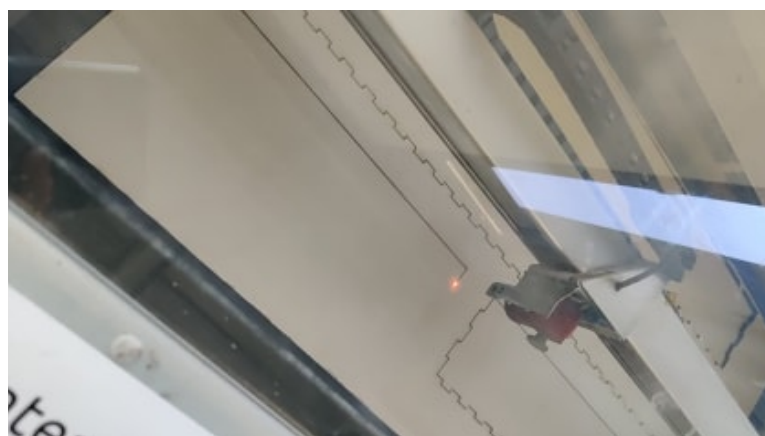
On doit pour cela se rendre au FABLAB (créneau réservé le 21 février) pour finir de construire toutes ces boîtes et pouvoir les assembler pour obtenir le résultat final, il faut, avant de s'y rendre, créer toutes les boîtes au préalable pour n'avoir qu'à les découper pendant le créneau réservé.

On utilise le site http://carrefour-numerique.cite-sciences.fr/fablab/wiki/doku.php?id=projets:generateur_de_boites pour les créer

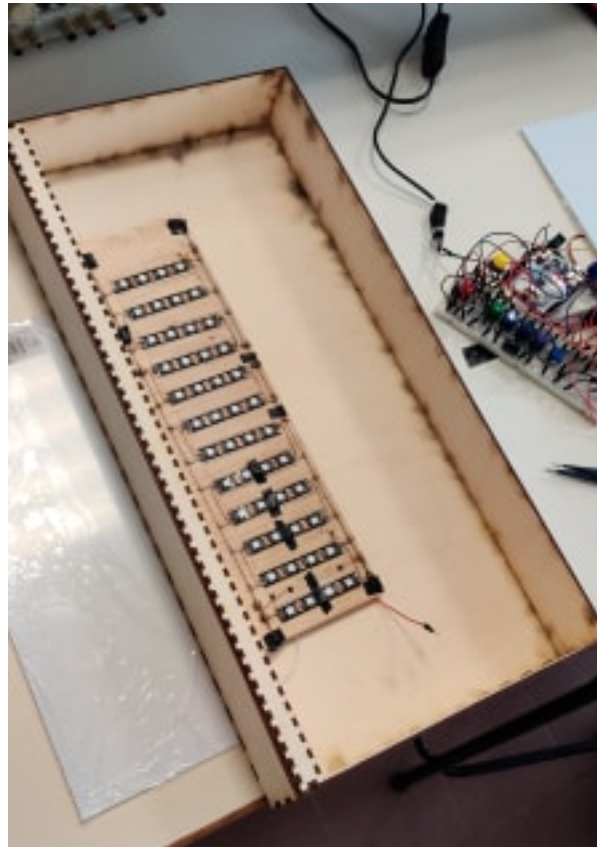


(c.p.) = corps principal (contient les LEDs que le joueur va utiliser pour trouver la bonne solution)
(c.s.) = corps secondaire (contient les LEDs analyseuse)

On s'était déjà rendu au FABLAB une première fois pour découper les planches qui servent de boîte principale + rebords :



On a ensuite assemblé les boîtes « rebords » à la boîte principale pour obtenir un résultat final de boîte de 2cm d'épaisseur (comme face gauche de la boîte principale ci-dessous) :



On a ensuite découpé des plaques de PVC à la bonne dimension que l'on vient superposer aux planches de LEDs (*principale et secondaire*) pour atténuer la luminosité des LEDs du jeu.

2) PROGRAMMATION DU BLUETOOTH

On commence par configurer le module Bluetooth :

```
Bluetooth_test | Arduino 1.8.9

Bonjour, prêt pour les commandes AT
OKOKsetnameOKsetPIN

Bluetooth_test
#include <SoftwareSerial.h>
#define RX 10
#define TX 11
SoftwareSerial BlueT(RX,TX);

void setup() {
  Serial.begin(9600);
  delay(500);
  Serial.println("Bonjour, prêt pour les commandes AT");
  BlueT.begin(9600);
  delay(500);
}

void loop() {
  while (BlueT.available()){
    Serial.print(char(BlueT.read()));
  }
  while (Serial.available()) {
    BlueT.write(char(Serial.read()));
  }
}

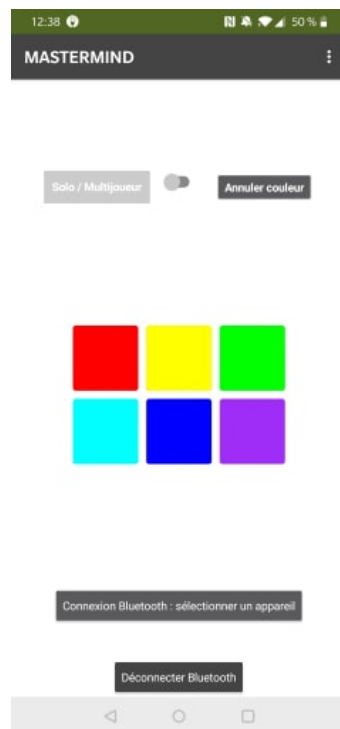
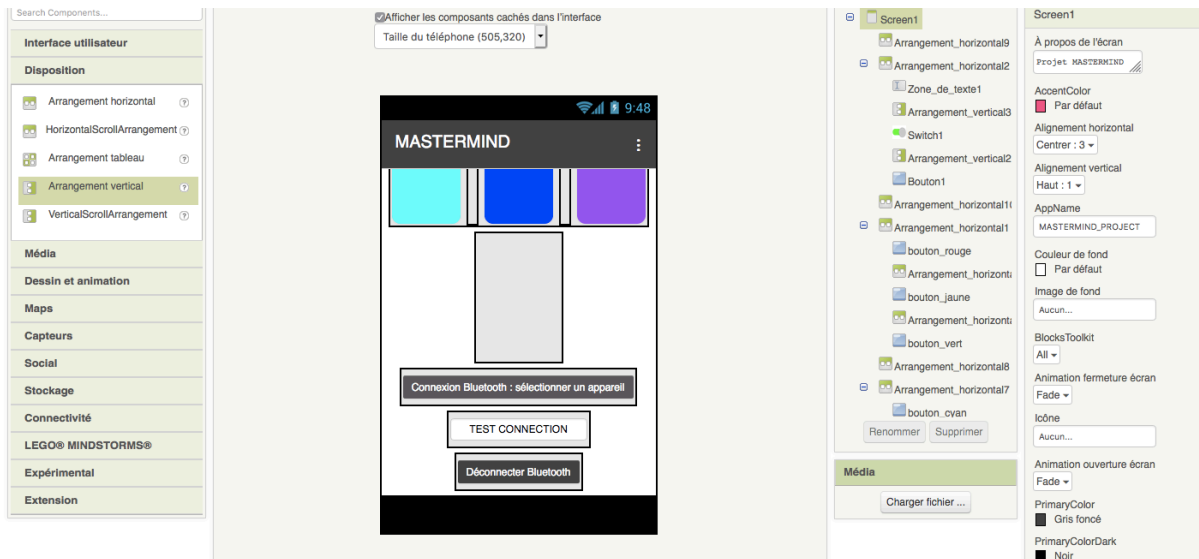
Téléversement terminé
mmes. Le maximum est de 30720 octets.
a, ce qui laisse 1705 octets pour les variables locales. Le maximum es

22 Arduino Nano sur /dev/cu.usbmodem1412
```

On configure le nom du module : **mastermind** et le pin : **0000** pour pouvoir par la suite faire interagir notre carte Arduino avec l'application MIT App Inventor

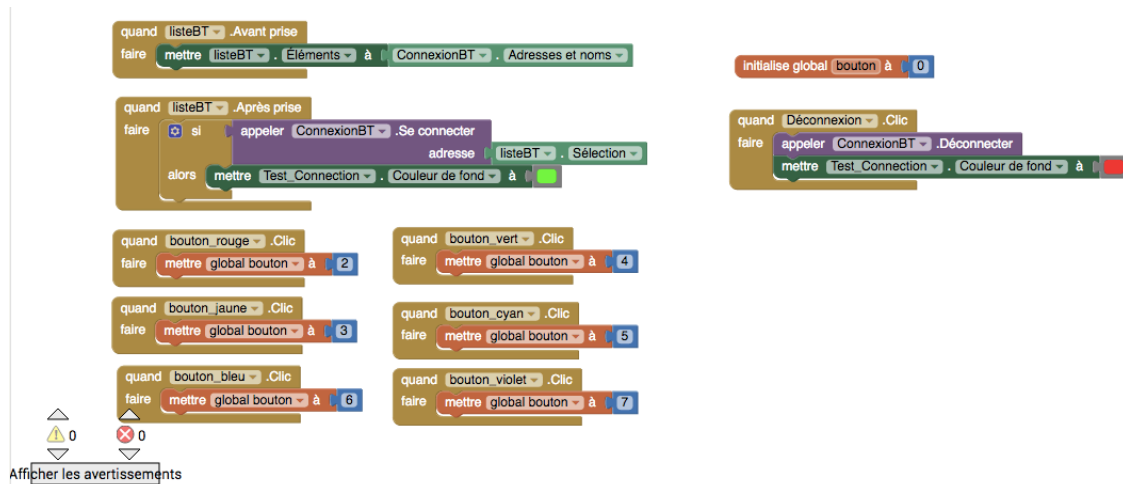
3) FINALISATION DE L'APPLICATION

On continue de configurer l'application sur le site MIT App Inventor :

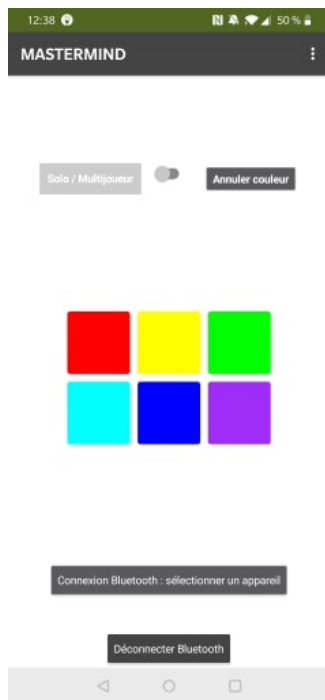


(Ci dessus : rendu sur téléphone après configuration de l'application, chaque bouton est fonctionnel et réagit lorsqu'il est sollicité)

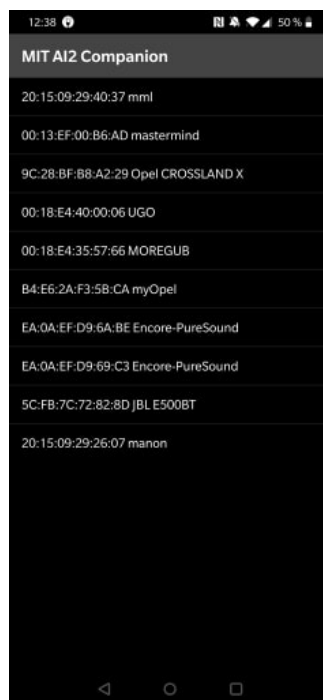
Puis on relie le module Bluetooth Arduino à l'application créée en utilisant les paramètres sur site MIT App Inventor :



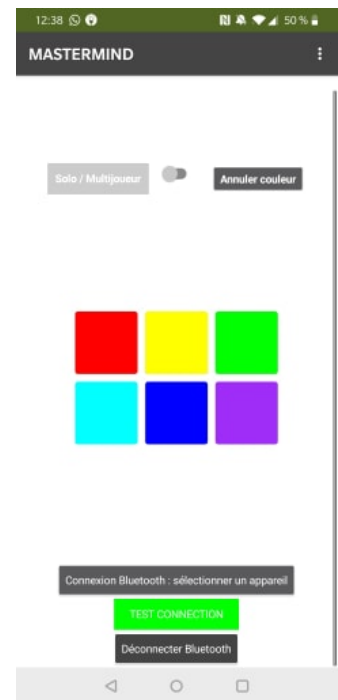
Explication du code rédigé ci-dessus :



Interface de base, apparaît lors du lancement de l'application



Après clic sur le bouton « Connexion bluetooth : sélectionner un appareil », la liste des appareils bluetooth à proximité s'affiche

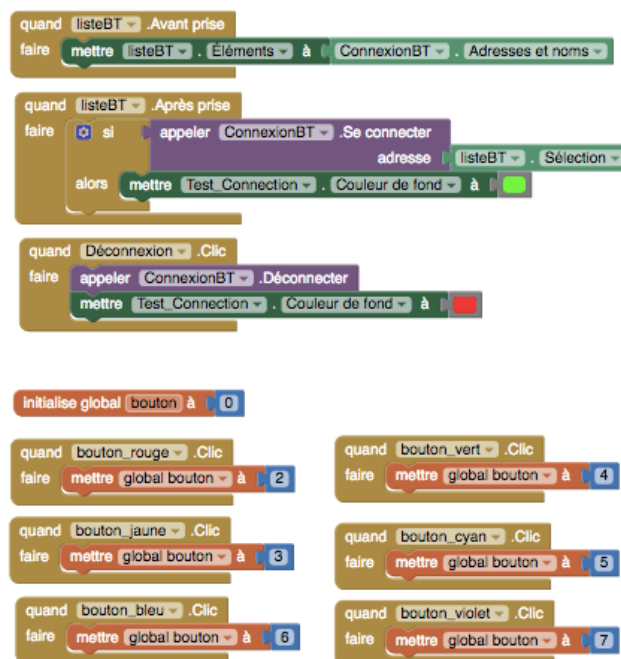


Après avoir sélectionné le module bluetooth configuré plus tôt : ici, « 00:13:EF:00:B6:AD mastermind »

Le bouton test de connexion s'allume en vert



Cette configuration se fait par les paramètres blocs du site MIT App Inventor



Elle permet de paramétrer les différents objets de l'application (boutons, switch, cadrans) et de transmettre des résultats à notre Arduino.

On commence par créer une **variable** « bouton » de **type int** qui, **selon la couleur du bouton enclenché, prend une certaine valeur (comprise entre 2 et 7)**

Ce résultat est **recupéré puis utilisé dans le programme principal pour remplacer le mode de jeu « normal »** (= *appui physique sur les boutons de couleurs*) sous forme vulgarisée d'un :
if (bouton==3) { led_jaune s'allume}

Ce qu'il reste à faire lors de la prochaine séance :

- Finir d'établir le lien entre l'application MIT App Inventor et le programme Arduino du jeu
- Construction boîte finale (à faire Vendredi 21/02 au FABLAB)
- Création boîte contenant les LEDs solution avec toit sous forme d'une trappe