

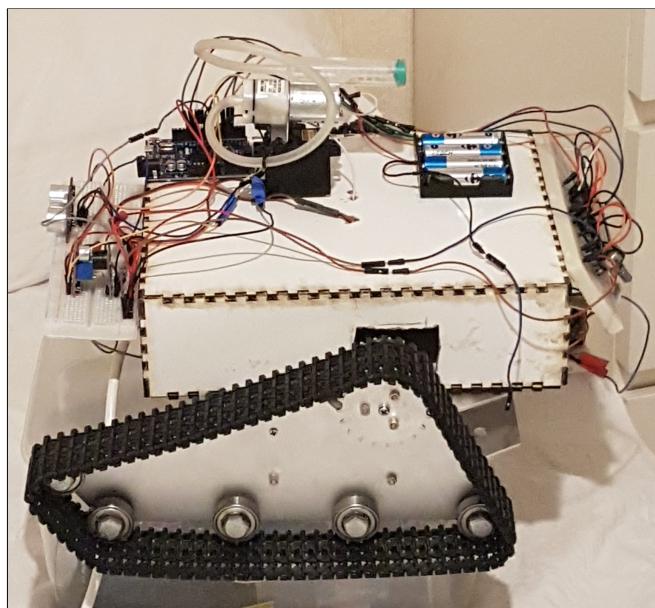
# Projet ARDUINO PeiP2

Année Scolaire 2021-2022

## Mini-tank télécommandé : Char Polytech FT-17 à tourelle pivotante

Étudiants : Allan CLERC, Jimmy VU

Encadrants : Pascal MASSON, Christian PETER, Amina BENOUAKTA



### I. Objectifs du projet

L'objectif est de créer un mini-tank qui puisse être contrôlé à distance (contrôle des déplacements).

Nous avons réfléchi à d'autres fonctionnalités qui peuvent rendre le tank plus réaliste et plus facile à utiliser.

Tout d'abord, nous avons voulu ajouter un canon qui puisse tirer sur commande.

Ensuite, nous avons pensé à un radar de recul, qui sonne quand on s'approche trop près d'un obstacle.

Nous voulions qu'il y ait des LEDs sur l'arrière du tank pour avertir de ses déplacements, une LED verte qui s'allume quand le tank est en mouvement, et une rouge qui s'éclaire quand il est à l'arrêt.

Enfin la présence d'un écran nous semblait pratiquement indispensable, nous avions beaucoup d'idées sur les données à afficher comme le nombre de projectiles restants dans le cas où nous aurions un système de recharge automatique du canon, la batterie restante si le tank a une batterie portable ou alors la distance entre le tank et un obstacle à l'arrière.

## II. Plannings

Ci-dessous le planning initial et le planning final.

Tâches	Semaine 1	Semaine 2	Semaine 3	Semaine 4	Semaine 5	Semaine 6	Semaine 7	Semaine 8
Montage de la base du véhicule (sans la partie pour accueillir le canon)								
Programmation des directions du véhicule (sans télécommande)								
Montage et programmation des LEDs								
Structure supérieure du tank								
Plateforme de rotation pour le canon								
Télécommande : partie déplacements								
Télécommande : partie écran LCD (batterie restante)								
Construction du pistolet à air comprimé (avec rechargeement facile)								
Signal sonore lorsque le tank recule trop près d'un obstacle								
Télécommande : partie écran LCD (munitions restantes)								
Microphone sur le tank								
Microphone sur la télécommande								
Assemblage du tank et du pistolet								
Télécommande : partie tir								
Construction de la télécommande : assemblage des parties LCD et déplacements								
Tests et corrections des problèmes								
Construction de la télécommande : assemblage de la partie tir								
Construction de la télécommande : assemblage partie microphone								

Planning initial

Tâches	Semaine 1	Semaine 2	Semaine 3	Semaine 4	Semaine 5	Semaine 6	Semaine 7	Semaine 8
Programmation des directions du véhicule (sans télécommande)								
Montage et programmation des LEDs								
Résolution de problèmes liés au code des moteurs et aux LED								
Début de l'écran LCD, solidification des fils								
Ecran LCD et construction du canon à air comprimé								
Début sur le module SR-04 et le buzzer et résolution des problèmes liés au code des moteurs								
Programmation du canon et de l'écran LCD, résolution de problèmes de faux contacts								
Installation de boutons poussoirs pour simuler le Bluetooth et faux contacts + L293N ne fonctionnait plus								
Début sur le Bluetooth et faux contact (un fil relié à la masse était cassé mais de l'intérieur)								
Ecran LCD qui marche en tant que voltmètre, tank pilotable grâce au Bluetooth, retrait des boutons poussoirs, solidification des câbles								
Canon télécommandable, tank pilotable depuis le bluetooth, écran LCD dysfonctionnel, microphone branché au tank								
Plateforme de rotation pour le canon (servomoteur), assemblage du tank et tests finaux								
Modélisation de la structure supérieure en bois et tests finaux								

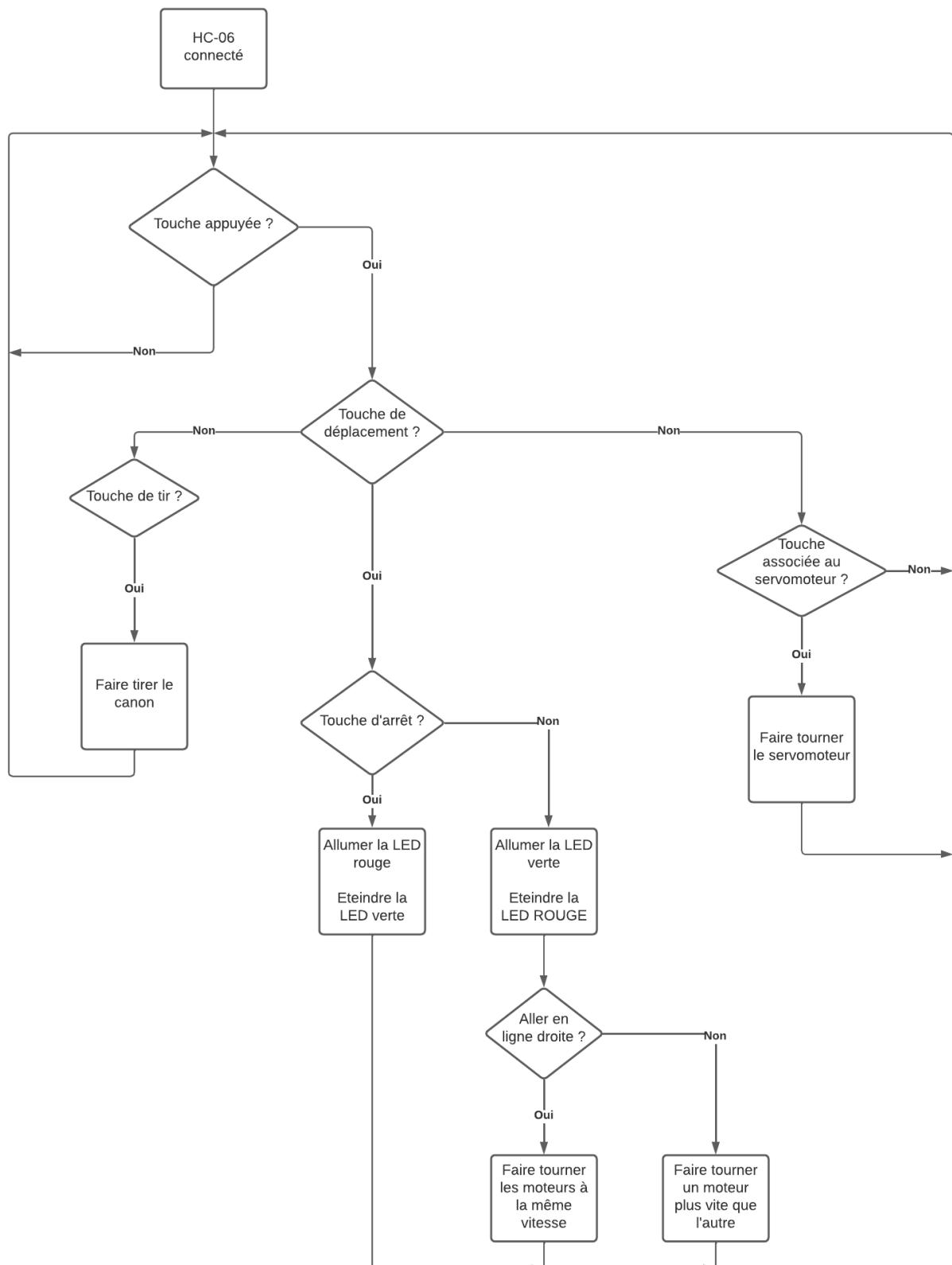
Planning final

### III. Algorithmes

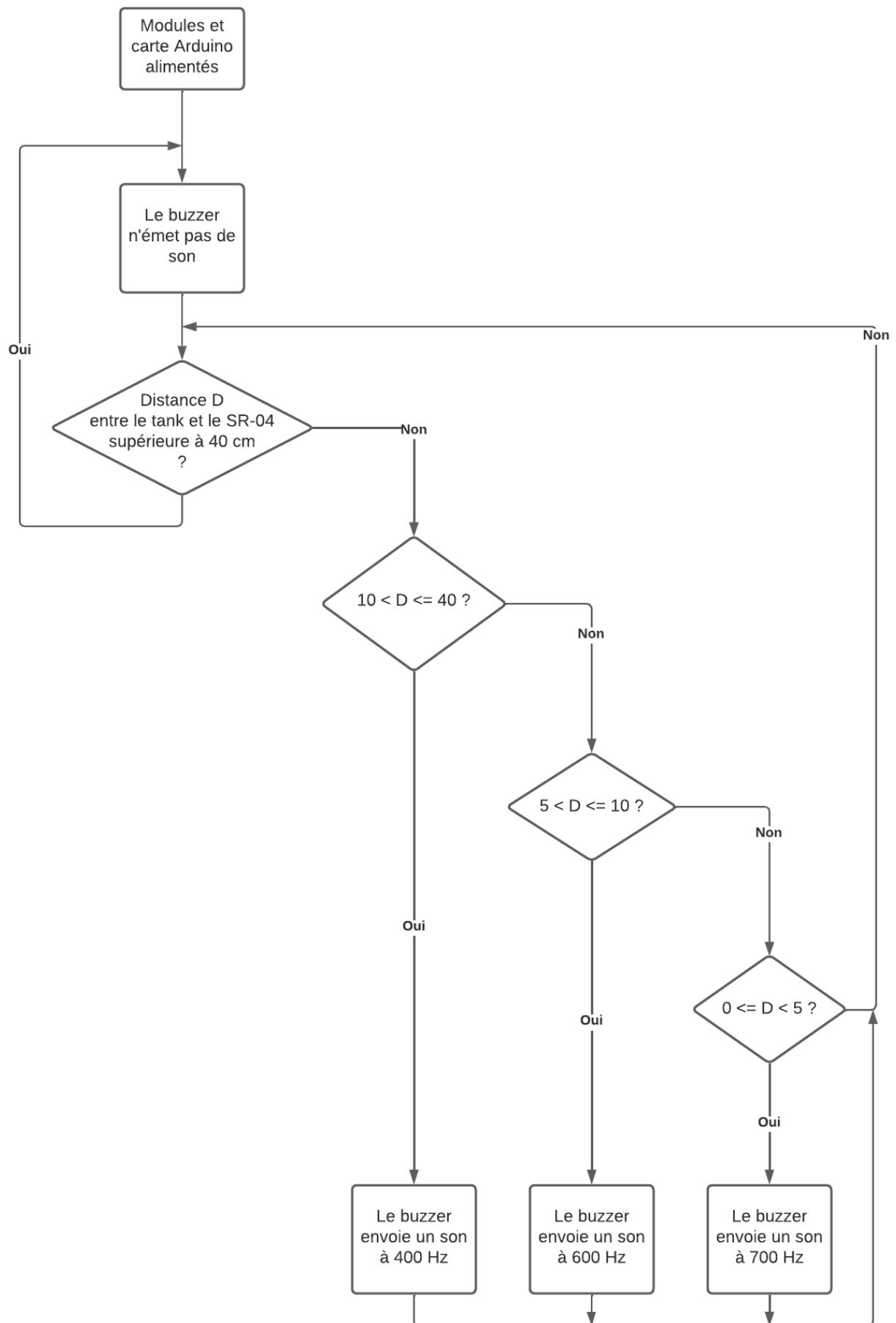
Le tank est composé de deux cartes Arduino qui ne sont pas directement reliées entre elles.

Nous pouvons décomposer le tank en deux parties ayant chacune leur algorithme : la partie Bluetooth et la partie détection d'obstacle.

#### 1. Bluetooth



## 2. Détection d'obstacle



## IV. Problèmes et solutions

De multiples problèmes sont survenus pendant le projet.

Tout d'abord, des erreurs de manipulation qui ont conduit à la destruction de plusieurs cartes Arduino.

Ensuite, certains fils se sont cassés de l'intérieur, ce qui est très difficile à repérer. Les faux contacts étaient aussi très nombreux, rendant les tests et l'assemblage du tank plus complexes.

Un problème mineur : les moteurs ne tournent pas à la même vitesse, il faut compenser cette différence de vitesse dans le code.

La carte L298N était défectueuse au bout de plusieurs semaines de marche. Elle ne faisait plus la liaison entre les moteurs et la carte Arduino. Elle a été remplacée par une autre L298N.

Les cartes Arduino et L298N ne doivent pas toucher la base du tank qui est en métal. Pour isoler les cartes, nous avons placé du carton entre le tank et les cartes, puis nous avons remplacé le carton par une plaque de bois plus solide, plus pratique et plus ergonomique.

Le programme du servomoteur MG995 était difficile à écrire. Le servomoteur en lui-même est fonctionnel, mais lorsqu'il est branché et programmé avec les bibliothèques Servo ou ServoTimer2, il n'est pas compatible avec les bibliothèques SoftwareSerial ou AltSoftSerial.

Le module Bluetooth HC-06 avait du mal à rester alimenté, les câbles n'étaient pas défectueux et il n'y avait pas de faux contacts. De plus, lorsque beaucoup de personnes utilisent du Bluetooth près du module, ce dernier perd sa connexion entre lui et le téléphone, et l'utilisateur en est informé 2 secondes après la perte de connexion. Pendant ce laps de temps, on ne peut pas contrôler le tank : il continue de rouler s'il était en train de rouler. La seule solution pour l'arrêter est de couper l'alimentation du tank.

Parfois, le module Bluetooth et le servomoteur ne pouvaient pas fonctionner en même temps. Si on les branchait ensemble, seul un composant parmi les deux était alimenté pour une raison inconnue. Le problème était indépendant du code. Notre branchement et notre simulation sur Tinkercad étaient corrects. On a tout de même réussi à les faire fonctionner ensemble, sans savoir réellement comment et pourquoi le problème persistait.

En fonction du bruit ambiant, le module KY-038 est censé renvoyer un nombre entre 0 et 1023 sur la sortie analogique auquel il est relié. Plus le bruit est élevé, plus la valeur est élevée. Cependant il renvoie toujours un nombre constant quelque soit le bruit ambiant et quelque soit le pin sur lequel il est branché. Nous n'avons pas trouvé de solution.

Nous avons décidé de retirer l'idée de l'écran LCD pour la afficher la batterie restante du tank car nous avons un câble muni d'un adaptateur 12V relié au secteur. L'idée de la batterie restante ne faisait plus sens. A la place, nous avons décidé d'afficher la distance entre le tank et l'obstacle sur l'écran LCD.

Le bloc à piles 6V n'alimente pas correctement la carte Arduino qui contrôle les modules sonores. Le buzzer jouer un son discontinu au lieu de jouer un son continu. En revanche, lorsque la carte était alimentée grâce au 5V fourni par l'ordinateur portable, le son était continu. Malgré ce problème, nous avons décidé de garder le bloc à piles 6V.

Pour faire tenir les breadboards à l'extérieur, nous avons utilisé des équerres. Le seul petit problème est la colle qui peut ne pas tenir et lâcher au bout d'un moment.

## V. Résultat final

La grande majorité des objectifs que nous nous sommes fixés ont été atteints.

Tout d'abord le tank est commandable à distance par Bluetooth. Il peut se diriger dans les quatre directions, en avançant et en reculant, et en tournant à droite ou à gauche. Pour le contrôler, on utilise l'application Bluetooth Electronics sur notre Smartphone. C'est depuis cette même application que nous contrôlons le tir.

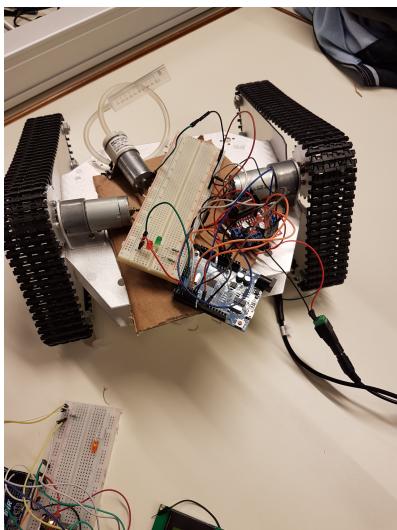
Le canon est opérationnel si on le contrôle sans passer par le Bluetooth c'est-à-dire juste en connectant le + et le -. Le projectile est envoyé à quelques mètres, et il est possible de faire tourner le canon sur 180° grâce à un servomoteur, qui est lui aussi contrôlé par Bluetooth.

Cependant, lorsqu'il y a beaucoup d'appareils utilisant le Bluetooth dans la même pièce, le module HC-06 est perturbé et il devient impossible de contrôler le tank à moins de se reconnecter.

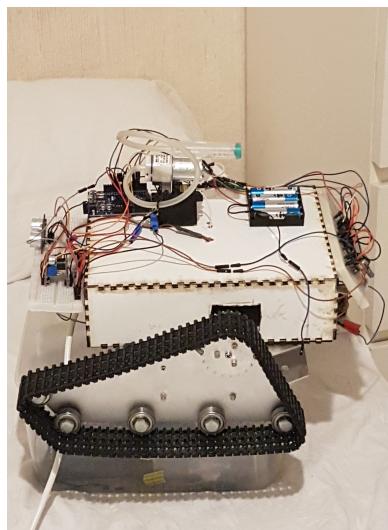
Les LEDs s'allument et informent correctement l'utilisateur sur les déplacements du tank.

Quand le tank se trouve trop proche d'un objet, le radar de recul le détecte et un buzzer émet un bruit, puis l'écran LCD affiche la distance qui sépare le tank de l'obstacle.

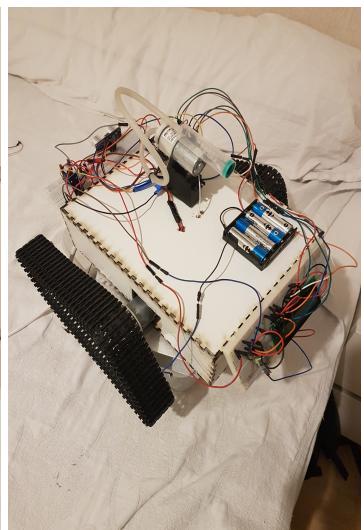
Finalement, l'écran n'affiche pas la batterie restante car le tank doit rester branché au secteur pour fonctionner.



Le tank au départ



Le tank à la fin



## VI. Conclusion

Finalement, nous avons un tank télécommandé commandé depuis notre smartphone. Nous pourrions améliorer le projectile en utilisant des billes en caoutchouc avec la bonne circonférence, ainsi que rajouter une détente afin de pouvoir faire monter la pression plus haut dans la seringue, et donc faire partir le projectile plus loin.

En ce qui concerne le Bluetooth, nous aurions pu le remplacer par un module wifi pour limiter les déconnexions. Nous aurions également pu ajouter un bouton poussoir qui fasse office de bouton d'arrêt d'urgence pour arrêter le tank sans le débrancher.

Nous voulions que le tank puisse se déplacer sans être branché pour qu'il ne roule pas sur son câble de connexion. Faire en sorte que le tank ait une batterie serait alors une priorité si on devait continuer le projet.

## Bibliographie :

- Code du voltmètre avec l'écran LCD :  
<https://www.allaboutcircuits.com/projects/make-a-digital-voltmeter-using-the-arduino/>
- Code du microphone :  
<https://www.aranacorp.com/en/using-a-microphone-with-arduino/#:~:text=The%20microphone%20requires%20low%20power,to%20adjust%20the%20microphone%20sensitivity>.
- Forums Arduino pour trouver une solution à beaucoup de problèmes : <https://forum.arduino.cc>