

**Département des PEIP**

**PROJETS – PEIP2**

***Année scolaire 2018-2019***



***BARDUINO***

**Etudiants : Armel Guinard,**

**Quentin Hauquin**

**Encadrants : Mr. MASSON,**

**Mr. ABDERHAMANE**

Ecole Polytechnique Universitaire de Nice Sophia-Antipolis, Département électronique

1645 route des Lucioles, Parc de Sophia Antipolis, 06410 BIOT

**REMERCIEMENTS :**

Pour leurs conseils avisés :

* Mr Masson
* Mr ABDERHAMANE

Pour avoir fabriqué la première version du Barduino :

* Germain Valentin
* Lupi Marjorie

Pour leur joie de vivre

* Victor Girard
* Benjamin Ferrer

**SOMMAIRE** :

Introduction 3

I.1. Définition du projet 4

I.2. Problématique 4

I.3. Schémas et fonctions 5

I.3.1. Algorigramme 5

I.3.2. Les fonctions 6

I.4 Cahier des charges 7

I.5 Planning 8

I.6 Evolution du projet 9

I.7 Matériel 11

I.8 Problèmes 12

I.9 Conclusion……………………………………………………………………………………………………..14

**Introduction :**

**Le Barduino :**

Le BARDUINO est un projet consistant à servir automatiquement différents cocktails depuis une commande vocal située sur le téléphone.

**Notre problématique a été :**

Comment **améliorer** le Barduino en **vitesse** et en **ergonomie** tout en gardant un maximum de **précision** et de **fiabilité** ?

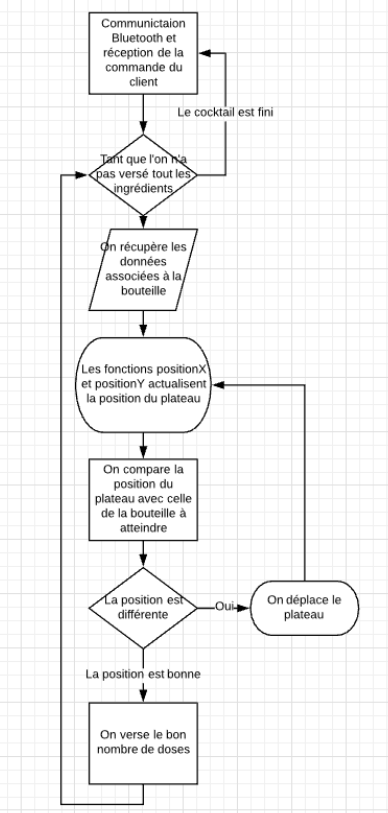
Nos **motivations** ont étaient premièrement d’avoir un projet concret qui nous permet d’appliquer les connaissances acquises lors de ces deux ans d’électronique au sein de Polytech de plus nous avons rarement l’occasion d’avoir un projet à aussi longue durée dans le temps.

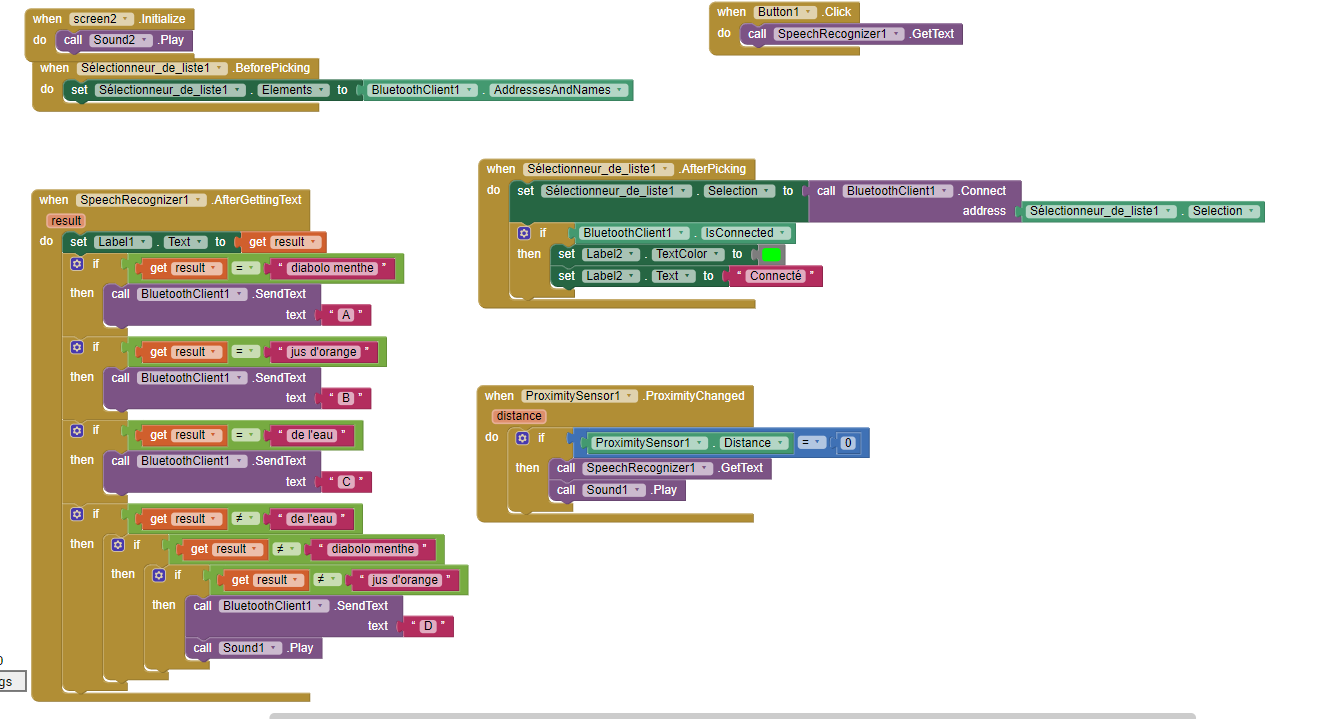
Ce projet était aussi un avant-gout du travail sur les projets que les entreprises attendront de nous cela nous a donc permis de voire nos progressions possibles et nos points forts lors de ce projet.

**Schémas et fonctions :**

Voici l’algorigramme qui représente le fonctionnement du Barduino détaillé ci-contre.

Tout d’abord, lorsqu’aucun cocktail n’est en cours de préparation, la carte Arduino communique en Bluetooth à l’aide du module HC06 avec le téléphone. Dès que le client commande un cocktail, cette communication est coupée pour laisser place au déroulement du reste du code. Tout d’abord, en fonction du cocktail choisi les ingrédients du mélange ne sont pas les mêmes. Nous récupérons donc les numéros des bouteilles et le nombre de doses associées à chaque bouteille du cocktail choisi. Par la suite, nous appelons la première fonction: déplacer. Cette fonction va aller chercher la position de la bouteille qui lui est passée en paramètre dans des listes prédéfinies et va appeler les fonctions positionX et positionY pour connaître la position du plateau en temps réel. Ces fonctions servent juste à renvoyer les valeurs des capteurs de distance. La fonction déplacer va ensuite comparer la position du plateau avec celle de la bouteille à atteindre et en déduire le sens de déplacement des deux moteurs. Ensuite déplacer va appeler les fonctions moteurX et moteurY qui vont faire tourner le moteur de l’axe X et de l’axe Y dans un sens défini par un booléen passé en paramètre. Pendant ce déplacement, la fonction déplacer continue de comparer la position de la bouteille à atteindre avec la position du plateau en appelant les fonctions positionX et positionY. Si la position est dépassée sur un axe, grâce aux fonctions moteurX et moteurY, le moteur concerné change de sens de rotation. Une fois la position atteinte sur un axe, le moteur concerné s’arrête. Lorsque les deux moteurs ont atteint la bonne position, la fonction déplacer s’arrête. La fonction doser est ensuite lancée. Cette fonction fait pivoter le servomoteur jusqu’à un angle de 90 degrés pour actionner la dose, attend deux secondes que la dose coule dans le verre, refait pivoter le servomoteur jusqu’à un angle de 0 degrés, attend deux secondes que la dose se remplisse et recommence un nombre de fois passé en paramètre. Une fois toutes les doses versées, la fonction doser est terminée. Ensuite on recommence la fonction déplacer et doser pour chaque bouteille du cocktail. Une fois chaque ingrédient versé, le cocktail est terminé et on attend une nouvelle commande du client en rouvrant la communication Bluetooth avec le téléphone. Les fonctions sont en annexes.



**Le code de l’application sur le téléphone :**

Ce code réalisé sur <http://ai2.appinventor.mit.edu/>.

Cette application permet avec des blocs de fonctions préfabriqués de coder une application.

Cette application est divisée en 3 parties principales.

La première permet d’activer le Bluetooth et de lister les Bluetooth que capte le portable.

La seconde permet de se connecter à un des Bluetooth listés.

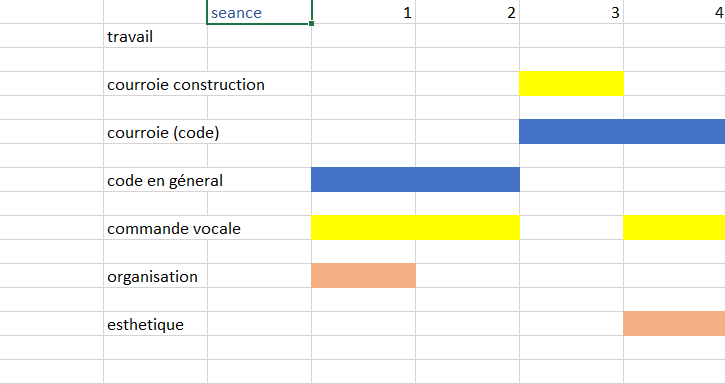
La troisième permet de transformer en chaine de caractère ce que la commande vocale reçoit et de le transmettre par Bluetooth à l’Arduino.

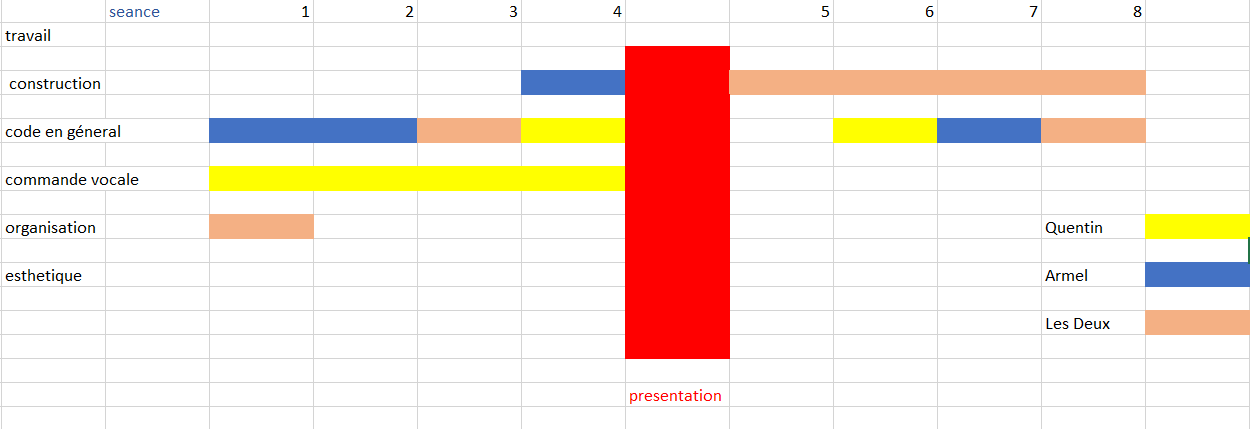
**Cahier des charges**

* **Nos attentes :**
  + - Nous souhaitions que le temps mît pour réaliser un cocktail soit plus court que ce que réalisait la version précédente du Barduino.
    - Nous voulions intégrer une commande vocale au Barduino pour commander les cocktails.
    - Nous souhaitions garder le même nombre de bouteille que la version précédente du Barduino.
    - Nous désirions pouvoirs servir différents cocktails de plusieurs boissons
* **Résultats** :
  + **Points positifs :**
    - En effet notre Barduino est bien plus rapide que l’ancienne version du Barduino qui était extrêmement lente avec comme moyen de déplacement une vis sans fin.
    - Nous avons bel et bien une commande vocale qui fonctionne parfaitement sur le téléphone
  + **Points négatifs :**
    - Pour gagner en vitesse nous avons donc changés de technologie en optant pour des moteurs à courant continue et des courroies. Cependant ces moteurs restent plus large que les précédents l’armature étant resté la même ils empêchent l’accès à la bouteille extérieur gauche et d’autres installations empêchent l’accès à la bouteille extérieur droite.
    - Néanmoins, nous gardons l’utilisation de cinq bouteilles différentes.

**Les plannings**

Le planning attendu :



Le planning réalisé :

Le planning de départ était divisé en trois parties, le code avec une grande partie déplacement, la commande vocale et la construction. Nous pensions finir la construction beaucoup plus rapidement qu’en réalité.

Une fois la commande vocale fonctionnelle, nous avons tous les deux travaillés sur la construction du Barduino. Ensuite à chacun des nombreux inattendus nous devions adapter la structure et le code en conséquence. Notre projet fut donc un jeu de va et vient entre code et construction.

**Evolution du projet**

* Séance 1 :

Tout d’abord, nous avons défini le travail de chacun à chaque séance.

Pour cette séance, l’un rechercha un modèle de commande vocale déjà existant tandis que l’autre débranchait les câblages et défaisait une grande partie du Barduino pour retirer un moteur pas à pas.

* Séance 2 :

Toujours à la recherche d’un modèle de commande vocal existant nous n’arrivons pas à comprendre le fonctionnement des autres modèles qui sont en langue étrangère (Turque).

En parallèle, des tests ont étés faits sur les moteurs à courant continu (moteurs CC) et les moteurs pas à pas ont étés effectués pour arriver à la conclusion qu’il était préférable d’utiliser des moteurs CC.

* Séance 3 :

D’un côté, depuis l'application MIT App Inventor Quentin tente de créer une application communiquant avec un module Bluetooth depuis un téléphone avec comme avantage de pouvoir personnaliser l'application et principalement, d'être la seule alternative trouvée pour transférer une chaine de caractère issue d'une voix vers un module Bluetooth.

De l’autre côté, le code du déplacement du plateau a été effectué, le deuxième moteur pas à pas démonté, le premier moteur vissé et la première courroie vissée. Ensuite, Armel a fixé une vis dans une pièce de bois pour faire tourner la poulie de la courroie.

* Séance 4 :

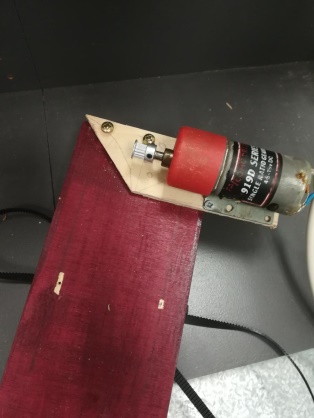
Le changement de screen sur l’application déconnecte le module Bluetooth ce qui pose un petit peu problème donc plusieurs solutions, supprimer un screen et tout faire sur un seul screen ou se débrouiller pour que l’appli se reconnecte automatiquement.

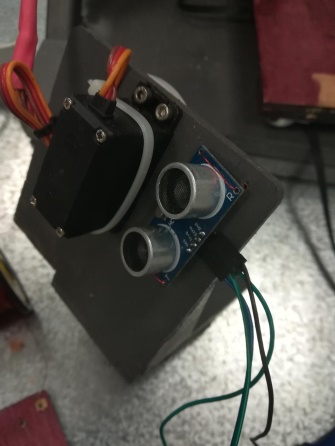
En parallèle, la pièce de bois supportant la poulie a été collée au socle et un dispositif permettant au petit plateau de se déplacer grâce à la vis sans fin à dû être retiré. De plus, les fils gainés nécessaires pour les moteurs et capteurs ont étés choisis et dénudés.

* Séance 5 :

Pendant la séance 5 nous avons réuni les deux codes Arduino de Armel et Quentin permettant ainsi de donner des ordres depuis l'application mobile au moteur à courant continu.

Ensuite, nous avons fini de visser la première courroie au plateau, découpé et vissé une planche qui sert d’extension du plateau et de support pour le deuxième moteur, repositionnés les roues du plateau suite à un problème d’équilibre, fait une encoche dans le petit plateau suite à un problème de taille du moteur, et décollé, scié et recollé la pièce de bois supportant la poulie de la première courroie suite au même problème de dimension du moteur.



* Séance 6 :

Nous avons fixé un des capteurs de distance au petit plateau avec du fil de fer, découpé et collé une pièce de bois supportant la poulie de la deuxième courroie identique à celle supportant la poulie de la première courroie et fixé une planche de bois sur l’un des côtés du plateau servant à réfracter les ondes de l’un des capteurs de distance.

* Séance 7 :

Nous avons essayés de faire fonctionner le servomoteur qui ne semblait marcher que dans un seul sens. Nous avons pu fixer la deuxième courroie que nous n'avions pas pu mettre la séance précédente car la colle n'avait pas encore fixé le support de sa poulie. Malheureusement cette courroie ne suffit pas à maintenir le petit plateau droit. Il faudra donc trouver une solution lors de la séance suivante.

* Séance 8 :

Pour la dernière séance nous avons donc mis en place des glissières sur lequel le chariot circule afin de le consolider et de supporter le petit plateau. Deux pièces de bois se sont décollées dû à une trop forte tension exercée par les courroies. Nous les avons donc vissées au socle. L'une directement au travers du socle et l'autre en rajoutant une pièce de bois sur le côté. Nous avons modifiés quelques parties du code suite à plusieurs essais et tout semble fonctionnel.

**Matériel**

Nous avons utilisé plusieurs composant tout au long de ce projet.



1. Module Bluetooth :

Pour la communication entre l’Arduino et le téléphone

1. Moteur courant continu :

Pour les déplacements avec les courroies ayant pour but de faire tourner la courroie.

1. Servomoteur :

Pour lever un disque qui appuie sur les verses-doses permettant ainsi le versage des doses de cocktail.

1. Sonar :

Deux sonars nous permettant de savoir en continue la position du chariot sur un plan en deux dimension.

**Problèmes**

Nous avons donc eu de nombreux problèmes tout au long de notre projet.

**Les capteurs de distance :**

Nous nous sommes rendu compte que nos capteurs de distance même en parfait état de marche renvoyaient des valeurs fausses de temps à autre. Cela empêchait le bon déroulement du code. Nous avons donc dû ignorer les valeurs zéro et celles supérieures à cent centimètre renvoyées par les capteurs de distance. Les valeurs zéros correspondent à une non-réception de l’onde et pour ce qui est des valeurs supérieures à cent, le Barduino ne fait que soixante centimètres dans la largeur.

**Les courroies :**

Nous pensions que les courroies allaient supporter le poids du chariot. Pour cela nous avons essayés de les tendre fortement. Malheureusement la tension exercée sur les poulies arracha les pièces de bois sur lesquelles elles étaient fixées pour les deux axes. Nous avons donc ajoutés une glissière à chacun des plateaux permettant de rectifier la trajectoire ainsi que de maintenir le petit plateau.

**Servomoteur :**

Lors des phases de tests nous nous sommes aperçue que chaque module en lui-même fonctionner cependant l’ensemble refusait de fonctionner et ce bloqué lors du fonctionnement du servomoteur il nous a donc fallu un peu de temps pour finalement comprendre qu’il y avait une perturbation de certains ports avec les servos et donc changer certains câbles de place tout en les laissant sur du PWM.

**Si nous devions recommencer du début, que ferions-nous autrement ?**

1. Nous commencerions la construction plus tôt.
2. Nous ne garderions que le socle du projet préexistant et reconstruirions le chariot de nous même afin de l’adapter à nos besoins.
3. Nous serions plus précis dans la pose des moteurs pour tenter de garder l’accès aux bouteilles extérieurs.
4. Nous commencerions les phases de test plus tôt afin d’éviter les surprises de dernière minute.
5. Nous communiquerions plus pour éviter les quiproquos et incompréhensions.

**Conclusion :**

Pour finir, nous avons rendu le Barduino plus rapide, il répond aux commandes vocales et sert les cocktails demandés. Cependant nous avons perdu l’accès aux deux bouteilles extérieures. De plus, nous rencontrons quelques problèmes avec la courroie. En effet, lorsque l’effort pour déplacer le plateau devient trop important la courroie se bloque alors que le moteur continue de tourner.

**Perspectives :**

Si nous devions continuer le projet nos objectifs seraient :

-Régler les problèmes mentionnés précédemment.

-Entamer la partie design en commençant par ajoutant des bandeaux de LED s’allumant au passage du chariot.

-Améliorer l’interface Bluetooth sur le téléphone pour la rendre plus accessible.

**Bibliographie**

* <https://www.instructables.com/>
* <https://www.youtube.com/watch?v=1wLd4qb22KM/>
* <https://www.lextronic.fr/shield-audio-video/20417-platine-shield-de-reconnaissance-vocale.html/>
* [http://ai2.appinventor.mit.edu/?locale=en#6049108077969408/](http://ai2.appinventor.mit.edu/?locale=en)
* <http://users.polytech.unice.fr/~pmasson/Enseignement-arduino.htm/>

**Annexe :**

Voici les différentes fonctions qui codent le Barduino :

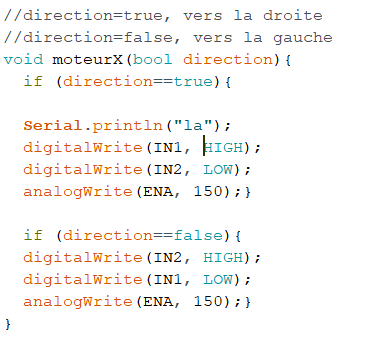
Déplacer :

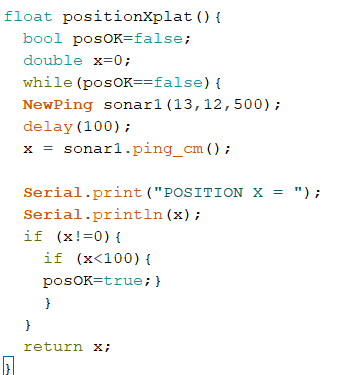


Dosage :

Les fonctions avec X dans le nom ont leur équivalent en Y.

MoteurX :



PositionXplat :

Le mot de la fin :

Ce projet a été une belle expérience malgré tous les problèmes que l’on a pu avoir. N’hésitez pas à essayer de le faire fonctionner ☺.