Sommaire

1. Introduction........................................................................................................................1
2. Organisation du projet……………...........................................................................................2
   1. Objectifs……………………………………………………………………………………………..……………..……2
   2. Répartition des tâches………………………………………..…………………………………………...……..2
3. Technologie et outils utilisés……………..…..……………………………………………………………..………..3
   1. Leap Motion…………………………………………………………………………………………………………….3
      1. Qu’est-ce que le Leap Motion ?.........................................................................3
      2. Fonctionnement…………………………………………………………………………………………...3
      3. Quel place dans le projet ?................................................................................3
   2. Arduino.…………………….……….………………….………………..……………………………………..………3
      1. Arduino Orion……………………………………………………………………………………………….3
      2. Programmation Arduino………………………………………………………………………………..4
   3. Robot……………………………………………………………………………………………………………………...4
      1. Modèle ?...........................................................................................................4
      2. Fonctionnement……………………………………………………………………………………………4
      3. Quel place dans le projet ?.......……………………………………………………………………..4
   4. Choix technologiques………….……………………………………………………………………...…………..5
4. Guide d’utilisation…………..…….…………..………………………..….…………………………………………..…6
5. Conclusion…………………………………………………………………………………………..………………..……….8
6. Annexes……………………………………………………………………………………………………………….…...9-14
7. **Introduction**

Ce projet IOC (Interaction des Objets Communicants) est un projet réalisé par Loïc MONOT et Pierre-Antoine CHARPENTIER et dirigé par Madame BELLEUDY, intervenante de l’université de Nice.

L’IOC est un domaine vaste qui nous entoure. En effet il concerne l’ensemble des appareils électriques, électroniques et microélectroniques capables d'interagir avec au moins une source d’informations. Nous avons choisi de mettre en application cette technologie en utilisant un Leap Motion communicant par bluetooth avec un bras robotisé sur chenilles.

Cette application permettra d’analyser et de comprendre le transfert d’informations entre deux objets. L’interaction s’est faite dans un premier temps par câble afin d’assurer la mobilité de base du bras robotisé sur chenilles, puis nous avons implémenté la connectivité bluetooth.

Nous allons donc voir quelles sont les différentes étapes du projet en allant des objectifs jusqu’à l’explication de l’utilisation du robot, en passant notamment par la répartition des tâches et les outils utilisés.

(Codes utilisés disponibles sur Github: loicphenix, rubrique LeapArm)

1. **Organisation du projet**
   1. **Objectifs** L’objectif de ce projet est de réussir à faire se mouvoir à distance via bluetooth un bras robotisé sur chenilles piloté grâce à un Leap Motion analysant la position des mains.
   2. **Répartition des tâches** Deux personnes travaillant sur ce projet, il fallait répartir les tâches en fonction des capacités et compétences de chacun. Ainsi, la répartition est la suivante :

* Loïc Monot:
  + Programmation Java
  + Programmation Arduino
  + Implémentation Leap Motion
* Pierre-Antoine:
  + Construction du robot
  + Rédaction du rapport
  + Power Point
  + Schémas explicatifs

1. **Technologie et outils utilisés**
   1. **Leap Motion**
      1. **Qu’est-ce que le Leap Motion**

Le Leap Motion est une sorte de mini caméra utilisable uniquement avec les mains et avec une précision détectant les doigts de celles-ci. D’une longueur de 8 cm, c’est un capteur analysant la position de nos mains selon des axes X, Y et Z (fig. 1).

* + 1. **Fonctionnement**

Le boîtier comprend une caméra style webcam ainsi que 3 LED infrarouges lui permettant une vue en relief. Les données collectées sont ainsi transmises par câble USB au pc. Un programme adéquat pourra lire les données, les traduire et les envoyer au bras robotisé par bluetooth.

* + 1. **Quel place dans le projet ?**

Le Leap Motion occupe une place majeure dans le projet. En effet, c’est l’objet communicant qui nous permet de récolter des données, il est donc indispensable dans notre application.  
 Nous avons choisi d’utiliser le Leap Motion afin de créer une réelle application au projet utilisant différents objets communicants, et d’apporter une certaine difficulté en sachant que c’est la première fois que nous faisons ce type de projet avec cette technologie.

* 1. **Arduino**
     1. **Arduino Orion**

La carte Arduino Orion est une carte électronique imprimée contenant plusieurs ports et connectiques permettant une utilisation complète et multi usage. Elle est capable de d’alimenter jusqu’à 4 moteurs en courant continu simultanément. Simple d’utilisation et ergonomique, elle est compatible avec la plupart des outils de programmation Arduino. Ainsi, nous assurons la compatibilité entre le Leap Motion, l’ordinateur et le bras robotisé sur chenilles (fig. 2).

* + 1. **Programmation Arduino**

Le langage utilisé par le logiciel Arduino pour programmer le microcontrôleur est basé sur les langages C/C++. Le logiciel est un IDE simple et clair, facile d’utilisation et compatible sur les 3 OS les plus important (Windows, iOS, Linux). Avec cette programmation on peut envoyer des informations à une carte Arduino. Celle-ci est capable de lire les entrées et de les transformer en sorties.

* 1. **Bras robotisé sur chenilles**
     1. **Modèle**

Le robot vient de chez Makeblock, un site internet proposant un large panel de constructions en kit (fig. 3). Ici nous avons choisi d’utiliser le Robotic Arm Tank afin de pouvoir contrôler plusieurs mouvements à savoir:  
 -avancer, reculer et rotation du robot entier  
 -abaissement et élevage du bras  
 -serrer et desserrer la pince  
Le robot n’a pas de cerveau moteur et est donc composé 4 moteurs indépendants, d’un bras articulé et d’une pince grande ouverture capable de soulever une charge allant jusqu’à 600g.  
Le robot est à monter soi-même.

* + 1. **Fonctionnement**

La pince est fixée à l’extrémité du bras articulé, lui-même fixé au centre du robot. Équipé de chenilles crantées, le robot peut se mouvoir même sur les terrains avec un faible reliefs. Contrôlé par ses 4 moteurs indépendants, il offre une bonne disponibilité et maniabilité.

* + 1. **Quel place dans le projet ?**

Le robot est aussi un élément clé du projet. En effet sans lui aucune information ne pourrait être réceptionnée et lue. Aussi indispensable, il permet donc de capter les données envoyées via bluetooth par l’ordinateur, suite à l’enregistrement des positions de la main au-dessus du Leap Motion.  
 Nous avons choisi le robot comme projet car nous sommes personnellement intéressés par la robotique et le transfert de données.

* 1. **Choix technologiques**
* Eclipse Mars 2.0 - Langage Java
* Arduino 1.6.9 - Langage C
* librairie LeapJava (Leap Motion) et RXTXcomm 2.2 (port série)

1. **Guide d’utilisation**
   1. **Marche avant/arrière**

Le robot a la possibilité de se mouvoir de différentes façons. Les deux premières sont les deux mobilités les plus importantes. En effet le robot doit pouvoir avancer et reculer pour se déplacer. Une main fermée assure une sécurité qui permet de contrôler les déplacements du robot en l’arrêtant net quelque soit son déplacement et sa vitesse. Aussi, une “zone d’ombre” (déterminée par nos soins et propre à chaque nouvelle application) permet au robot ne pas être toujours en mouvement (fig. 4-5).   
 Pour avancer: Placer la main droite fermée au-dessus du Leap Motion, paume vers le bas. Puis l’ouvrir et l’avancer légèrement.  
 Pour reculer: Placer la main droite fermée au-dessus du Leap Motion, paume vers le bas. Puis l’ouvrir et la reculer légèrement.

* 1. **Rotation droite/gauche**

Afin de pouvoir optimiser les déplacements du robot, nous avons implémenté les mouvements de rotation. Ainsi, il peut se déplacer dans toutes les directions et même faire demi-tour sur place. Une main fermée assure une sécurité qui permet de contrôler les déplacements du robot en l’arrêtant net quelque soit son déplacement et sa vitesse. Aussi, une “zone d’ombre” (déterminée par nos soins et propre à chaque nouvelle application) permet au robot ne pas être toujours en mouvement (fig. 4-5).  
 Rotation à droite: Placer la main droite fermée au-dessus du Leap Motion, paume vers le bas. Puis l’ouvrir et la déplacer légèrement vers la droite.  
 Rotation à gauche: Placer la main droite fermée au-dessus du Leap Motion, paume vers le bas. Puis l’ouvrir et la déplacer légèrement vers la gauche.

* 1. **Mouvement bras/pince**

Après les déplacements principaux, nous avons pu mettre en application une gestion du robot avec la main gauche. Permettant une meilleure optimisation du robot, cela augmente également la difficulté de programmation et de contrôle. En effet lier les deux mains dans le code nous a permis d’apprendre de nouvelles fonctionnalités, et pour la personne contrôlant le robot, une bonne coordination de ses deux mains est primordiale. Une deuxième “zone d’ombre” (déterminée par nos soins et propre à chaque nouvelle application) permet au bras ne pas être toujours en mouvement. Par contre, les déplacements s’effectuent avec une main fermée comme ouverte (fig. 4-5).  
 Mouvement du bras vers le haut: Placer la main gauche au-dessus du Leap Motion, pouce vers le haut. Puis la déplacer légèrement vers le haut.  
 Mouvement du bras vers le bas: Placer la main gauche au-dessus du Leap Motion, pouce vers le haut. Puis la déplacer légèrement vers le bas.  
 Ouverture pince: Placer la main gauche au-dessus du Leap Motion, pouce vers le haut. Puis l’incliner légèrement vers la gauche.  
 Fermeture pince: Placer la main gauche au-dessus du Leap Motion, pouce vers le haut. Puis l’incliner légèrement vers la droite.

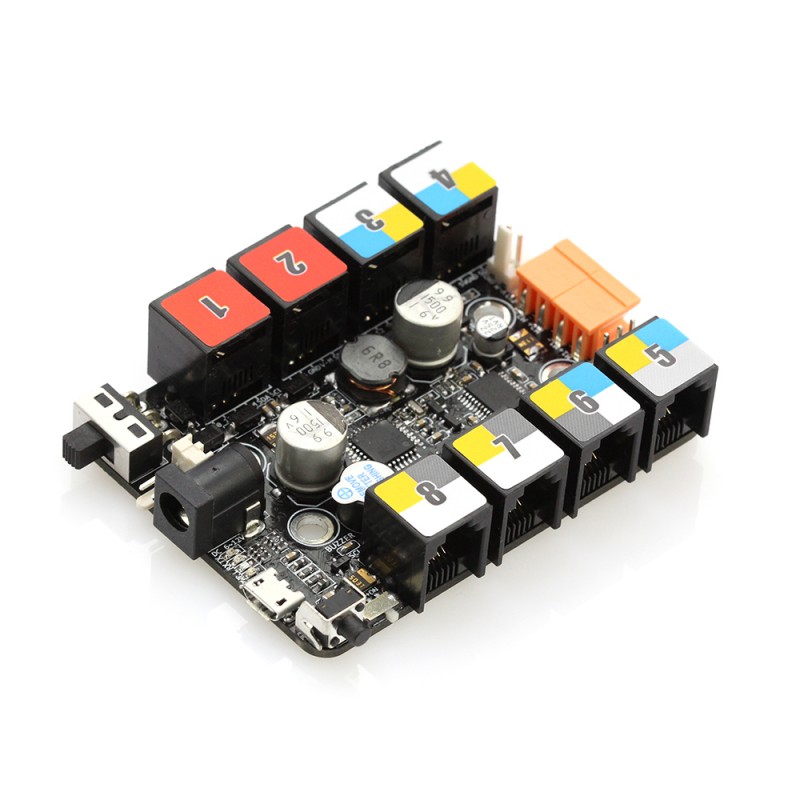
1. **Conclusion**

Au cours de ce projet nous avons pu nous familiariser avec la technologie Leap Motion et la lier avec la communication bluetooth avec un bras robotisé sur chenilles. Nous avons su créer un outil utile, pratique, amusant, accessible à tous et utilisable pour de nombreuses applications.  
 En effet on peut imaginer que le robot soit utilisable tant chez un particulier pour du loisir, dans une université pour des démonstrations ou même pour des professionnels tels que les militaires pour du déminage par exemple.

Comme propositions d’améliorations au projet, nous pourrions apporter des précision sur les mouvements et une rotation plus rapide. Nous pourrions ajouter une caméra embarquée ou encore des panneaux solaires afin d’optimiser l’utilisation du robot car il fonctionne avec piles et on est obligé d’avoir un oeil dessus afin de connaître ses déplacements et ses réactions par rapport aux positions des mains.

1. **Annexes**

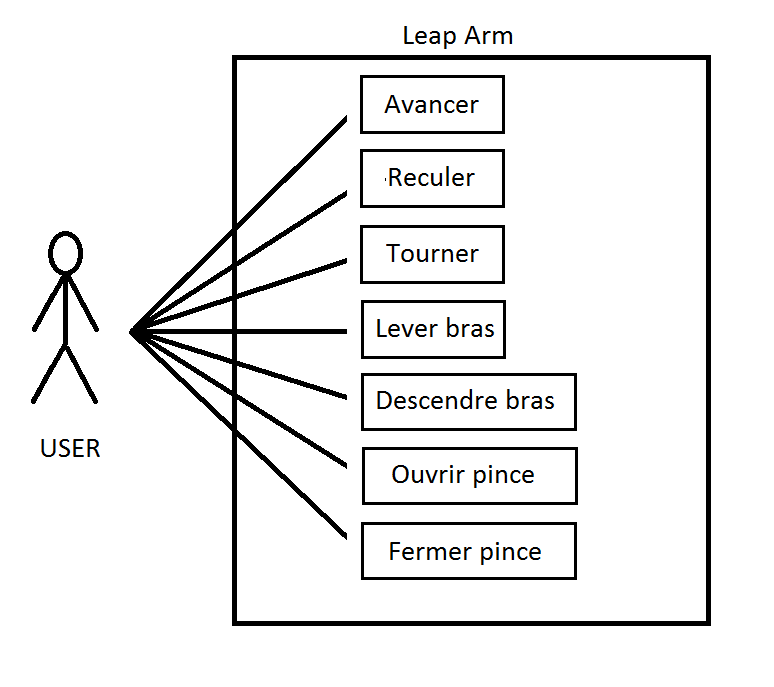
 **Représentation du Leap Motion avec positionnement des mains (fig.1)**

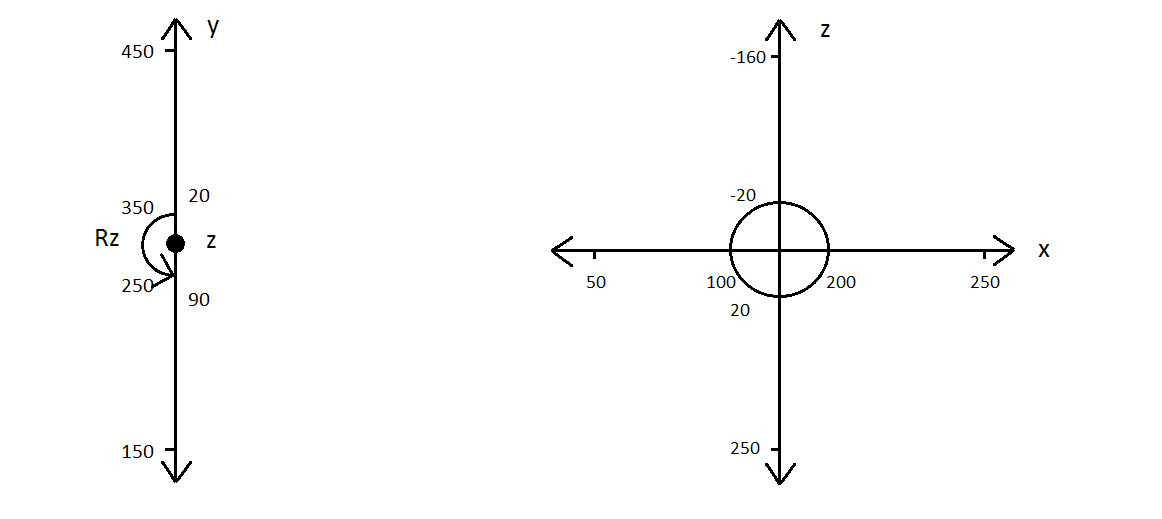


**Représentation de la carte Arduino Orion (fig. 2)**



**Représentation bras robotisé sur chenilles (fig. 3)**

 **Schéma UML représentatif des différentes actions possibles du robot (fig. 4)**

 **Schéma représentatif de la position des mains et des bornes utilisées (fig. 5)**