

---

PROJET DE FIN D'ETUDES

**Institut de Recherche Biomédicale des Armées**

Ministère de la Défense

Effectué par :

**Abbas RAMADAN**

ENSTA Bretagne, CI 2019, Option SPID, Profil Robotique

Date :

Juin - Novembre 2019

Sous la direction de :

**M. Fabien ROYNARD**

Ingénieur de recherche, Institut de Recherche Biomédicale des Armées

**Dr. Ouamar FERHANI**

Docteur en optronique et traitement d'image, Institut de Recherche Biomédicale des Armées

Tuteur école :

**M. Benoit ZERR**

Enseignant Chercheur, ENSTA Bretagne

**Développement d'une application de contrôle robotique**

---



Ce présent rapport intermédiaire fait état du travail effectué dès le début de stage de fin d'études jusqu'à la date actuelle.



## Remerciement

Je tiens à remercier,

Madame le Médecin Général **Inspecteur SAILLIOL** de m'avoir accordé sa confiance et de m'avoir donné l'opportunité de faire mon stage de fin d'études au sein de l'Institut de Recherche Biomédicale des Armées.

**M. Ouamar FERHANI**, chef de l'unité Prototypage, Instrumentation et Calcul Scientifique (U.PICS), et **M. Fabien ROYNARD**, d'avoir accepté de m'intégrer au sein de leur unité et de m'avoir aidé dans la rédaction de mon rapport de stage. Je les remercie également de leur disponibilité durant le stage.

**M. Benoit ZERR**, mon tuteur de stage à l'ENSTA Bretagne, pour sa disponibilité et la sollicitude dont il a fait preuve quant au bon déroulement du stage.

Mes remerciements vont également à l'ensemble de l'équipe de l'unité Prototypage, Instrumentation et Calcul Scientifique, qui m'ont si bien intégré dans l'équipe et qui m'ont fait partager leurs expériences professionnelles au cours de ce stage.



## Abstract

This report describes the internship I spent at the French Armed Biomedical Research Institute in France, with a view to obtaining a diploma in engineering studies and techniques from ENSTA Bretagne.

The Army Biomedical Research Institute is a public institution under the Ministry of the Armed Forces. Its main mission is to respond to the needs expressed by the Military Staffs in terms of combatant protection.

The objective of this internship is to develop a control application for a humanoid robot of the PEPPER type (manufacturer Softbank Robotics). It is part of a project to develop a platform based on a humanoid robot to enable prevention and medical monitoring. This platform would make it possible to learn about the interactions between the robot and military personnel who have suffered physical or psychological traumatization.

The work is divided into two parts; the first part consists of creating an Android application for manual robot control. It is used to display different information on a tablet integrated on the robot's chest. This part was developed in Android studio (JAVA - XML). The second consists in developing a program to control the robot autonomously in order to perform many tasks.

## Résumé

Ce rapport rend compte du travail effectué dans le cadre d'un projet de fin d'études au sein de l'Institut de Recherche Biomédicale des Armées à Brétigny-sur-Orge, France, en vue de l'obtention du diplôme d'ingénieur des études et Techniques de l'ENSTA Bretagne.

L'Institut de Recherche Biomédicale des Armées est un établissement public dépendant du Ministère des Armées. Sa mission principale est de répondre aux besoins exprimés par les Etats-majors en matière de protection du combattant.

Ce stage a pour objectif de développer une application de contrôle d'un robot humanoïde de type PEPPER. Il s'intègre dans un projet de développement d'une plateforme s'appuyant sur un robot humanoïde et permettant d'assurer des fonctions de prévention et de surveillance médicales. Cette plateforme permettrait d'étudier les interactions entre le robot et les militaires ayant subi un traumatisme physique ou psychique.

Le travail se divise en deux parties. La première consiste à développer un programme permettant le contrôle autonome du robot pour effectuer plusieurs actions. La seconde consiste à créer une application Android de contrôle manuel du robot. Elle sert à afficher différentes informations sur une tablette intégrée à la poitrine du robot. Cette partie a été développée en Android studio (JAVA – XML).

## Table des matières

Introduction.....	1
1. Organisme d'accueil .....	2
1.1. Présentation .....	2
1.2. Organigramme.....	2
1.3. L'Unité Prototypage Instrumentation et Calcul Scientifique (U. PICS).....	3
2. Problématique .....	4
2.1. Mise en contexte .....	4
2.2. Problématique de l'unité.....	5
2.3. Travaux à réaliser .....	6
3. État de l'art.....	7
3.1. Présentation du robot .....	7
3.1.1. Architecture physique .....	7
3.1.2. Fonctionnement du robot .....	9
3.2. Méthode Ingénierie Système (IS) .....	9
3.2.1. Analyse .....	10
3.2.2. Identification des exigences .....	11
3.2.3. Langage d'implantation .....	12
4. Application technique et résultats .....	15
4.1. Python Naoqi.....	15
4.2. Android SDK.....	19
5. Avancement de travail .....	22
Conclusion .....	23
Bibliographie.....	25
Annexe I - Planning des activités .....	27
Annexe II - Dialogue de base : "Basic Channel" .....	28

## Liste des tableaux et figures

Tableau 1 : Synthèse des exigences .....	12
Tableau 2 : Langages de programmation pris en charge .....	13
Tableau 3 : Liste d'avancement des différentes tâches avec les durées.....	22
Tableau 4 : Diagramme de Gantt .....	27
Figure 1 : Organigramme de l'IRBA .....	3
Figure 2 : Organisation de l'unité PICS .....	4
Figure 3 : Schémas plateforme.....	6
Figure 4 : Robot Pepper.....	7
Figure 5 : Les degrés de liberté du robot .....	8
Figure 6 : Capteurs présents sur le robot Pepper .....	9
Figure 7 : Active ou désactive Autonomous Life .....	9
Figure 8 : Diagramme d'ordonnancement des tâches .....	10
Figure 9 : Exemple de conversation Robot – Patient développée .....	11
Figure 10 : Diagramme « Chorégraphe » qui fait déplacer le robot et lever le bras.....	13
Figure 11 : Emulateur du robot et de sa une tablette virtuel .....	14
Figure 12 : Fonctions prédéfinies du module QI .....	16
Figure 13 : Diagramme relationnel des classes de mon programme.....	17
Figure 14 : Lancement du Plugin Pepper dans Android Studio .....	20
Figure 15 : Diagramme représentant le flux de conversation.....	21



## Introduction

Dans le cadre de mon projet de fin d'études de mon cycle d'ingénieur, j'effectue un stage au sein de l'Institut de Recherche Biomédicale des Armées du 03 juin au 30 novembre 2019. L'objectif est d'aborder concrètement les fonctions d'ingénieur en éprouvant nos capacités scientifiques, techniques et organisationnelles. Ce stage est une réelle opportunité de traiter un dossier technique et mettre en application mes connaissances théoriques. Par ailleurs, j'ai eu un grand plaisir à découvrir le milieu de la recherche et me permettre ainsi de valider mon projet professionnel.

Le Trouble de Stress Post-Traumatique (TSPT) est devenu une priorité pour le ministère des armées. Reconnu comme blessure de guerre depuis 1992, il fait l'objet d'un plan d'action depuis 2011. L'Institut de Recherche Biomédical des Armées a donc été sollicité afin de mener des recherches sur l'amélioration de la prise en charge des patients atteints de TSPT. Les différents projets auxquels l'U.PICS a participé ont amené l'unité à s'interroger sur les applications émergentes de la robotique et des systèmes intelligents dans le domaine des sciences cognitives.

Pour répondre à un besoin thérapeutique, l'U.PICS propose d'explorer/étudier une solution technologique sous la forme d'une plateforme s'appuyant sur un robot humanoïde et permettant d'assurer des fonctions de prévention et de surveillance du sujet. Cette plateforme permettrait d'étudier les interactions entre le robot et les militaires ayant subi un traumatisme physique ou psychique. Dans un premier temps, celle-ci serait déployée dans un environnement contrôlé, de type appartement témoin, et dans un second temps, au domicile du patient. Les fonctionnalités envisagées pour cette plateforme seraient :

- La surveillance du patient via le suivi de signaux électro-physiologiques tels que l'ECG ;
- La détection d'une possible « crise » de stress des militaires atteints de cette pathologie ;
- Une intervention du robot lors de la phase de stress ;
- Au travers des capacités d'interaction du robot, proposer des exercices permettant d'améliorer la prise en charge du patient.

Ma mission consiste à développer et tester deux applications de contrôle des activités du robot. Cela implique diverses tâches préliminaires, parmi lesquelles : l'étude de l'ingénierie système pour analyser les besoins du projet, l'étude de l'état de l'art et l'environnement de programmation du robot. Une fois ces tâches réalisées, j'ai développé les deux applications. La première consiste à réaliser un programme sous Python permettant un contrôle autonome du robot afin d'effectuer différentes tâches préprogrammées. La seconde consiste à développer une application Android de contrôle du robot par la tablette. Elle permet ainsi au patient d'activer certaines fonctions.

Ce rapport intermédiaire présente mon travail effectué durant les 3 premiers mois de stage. Au préalable, il apparaît logique de présenter l'environnement du stage, puis de définir le contexte du projet et les objectifs du stage. Le chapitre trois sera consacré à l'état de l'art. Je préciserai dans le chapitre quatre, les différentes missions et tâches que j'ai pu réaliser. Enfin, je conclurai mon rapport par un bilan personnel ainsi que les actions réalisées et les perspectives pour les 3 prochains mois de stage.

## 1. Organisme d'accueil

Ce premier chapitre est une courte présentation de l'IRBA, de son organigramme, et de ma place dans le centre de recherche. Elle se termine par la présentation des deux projets sur lesquels j'ai travaillé.

### 1.1. Présentation

Implanté en Île-de-France depuis le 1er mars 2009 à Brétigny-sur-Orge, l'Institut de Recherche Biomédicale des Armées (IRBA) est l'établissement du Service de Santé des Armées (SSA) spécifiquement dédié à la recherche. Il est issu du regroupement de quatre établissements de recherche du SSA dans une perspective de concentration des moyens de recherche dédiés à l'amélioration de la santé des militaires.

L'IRBA a une double mission. La première est de répondre aux besoins exprimés par les États-majors et le SSA en matière de protection du personnel militaire dans un contexte opérationnel marqué par un environnement hostile et des engagements sous menace NRBC (Nucléaires, Radiologiques, Biologiques, Chimiques). La seconde consiste à anticiper les besoins pour renforcer les contre-mesures médicales face à des contraintes opérationnelles variées.

L'IRBA agit dans trois domaines étroitement liés : la recherche, l'expertise et la formation. Il s'appuie également sur des partenariats civils ou militaires.

Les recherches visent à anticiper et prévenir les effets délétères associés à certaines situations opérationnelles, à améliorer la prise en charge et le traitement des malades et des blessés et à réduire les séquelles à court, moyen ou long terme qui pourraient en résulter.

### 1.2. Organigramme

L'IRBA est l'une des composantes du SSA placé sous l'autorité du Ministère des Armées. La recherche au sein de l'IRBA est organisée selon 3 divisions (Cf. Figure 1) :

- La division Soutien Médical en Opération (SMO) étudie l'adaptation du combattant aux contraintes opérationnelles : environnements extrêmes (milieu aéronautique, climat contraignant), prévention des conséquences du stress ou de la fatigue du combattant, développement de contre-mesures médicales face à des traumatismes physiques (ex : effet du blast) ... ;
- La division Défense-NRBC étudie les effets biologiques face aux menaces nucléaires, radiologiques, biologiques et chimiques. Elle développe également des contre-mesures adaptées à ces menaces ;
- La division Appui à la Recherche (DAR) contient toutes les unités de soutien nécessaires aux recherches de l'IRBA (gestion des d'infrastructures, gestion des installations confinés (laboratoire NRBC)).

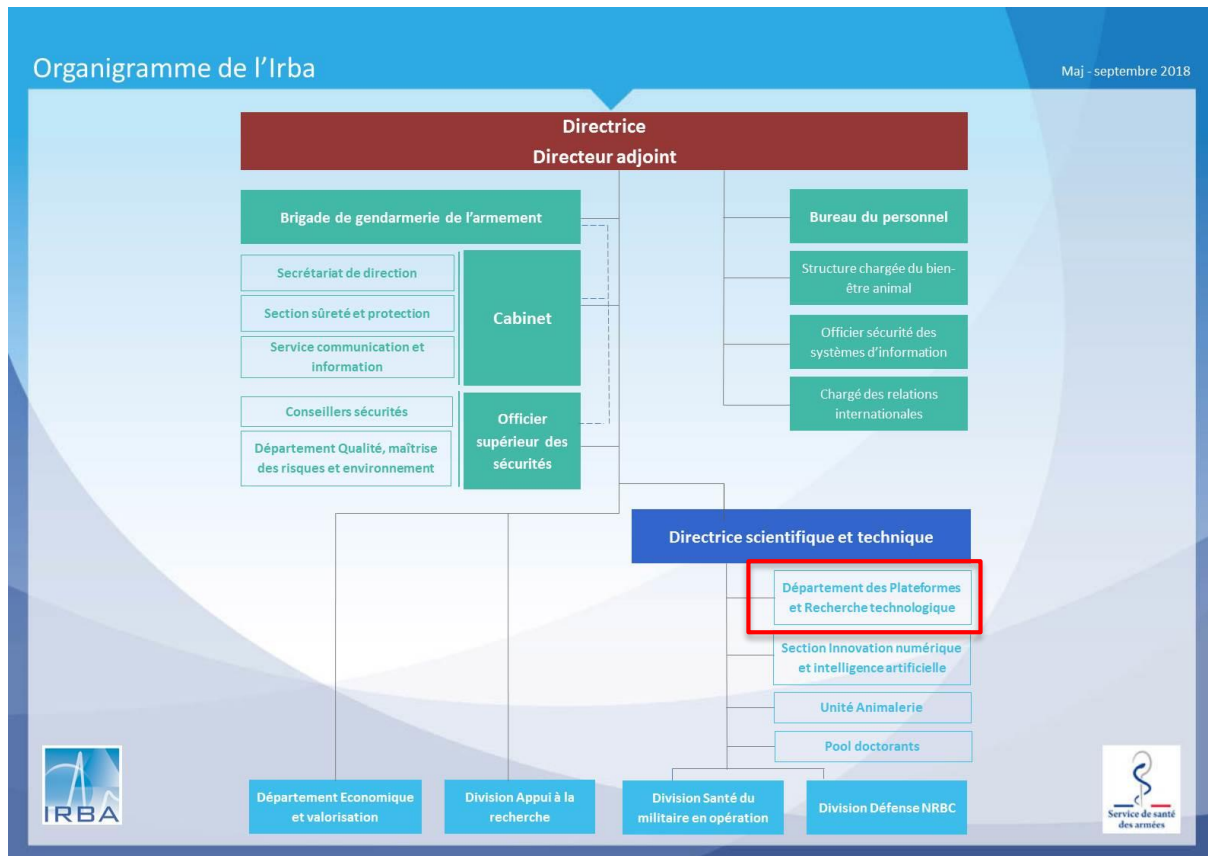


Figure 1 : Organigramme de l'IRBA

L'unité à laquelle je suis rattaché appartient au département des Plateformes et Recherches technologiques (D.PRT, Cf. Figure 1). Celle-ci met à disposition des équipes de recherche différentes plateformes telles que des plateformes d'imagerie ou d'analyses biologiques et propose différentes prestations de soutiens scientifiques et techniques. Ce département est directement subordonné à la direction scientifique et technique.

### 1.3. L'Unité Prototypage Instrumentation et Calcul Scientifique (U. PICS)

L'unité dans laquelle je travaille se nomme l'U.PICS (unité de prototypage instrumentation et calcul scientifique). Elle est constituée de chercheurs, d'ingénieurs, de techniciens et d'ouvriers d'état. Cette unité propose la réalisation de prototypes afin de mener à bien des expérimentations scientifiques. Elle est similaire à un bureau d'études en R&D, avec la capacité de créer et développer des outils technologiques ou scientifiques sur mesure. Ses principales missions sont la conception, la réalisation et le suivi de prototypes innovants destinés aux unités de recherche. Ses domaines de compétence recouvrent la mécanique de précision, l'instrumentation et l'électronique, la physique des capteurs et la métrologie. L'unité s'appuie également sur des compétences en calculs scientifiques et développe des outils informatiques d'acquisition, de traitement et d'analyse de signaux. Toutes ces compétences sont réparties sur quatre équipes (Cf. Figure 2).

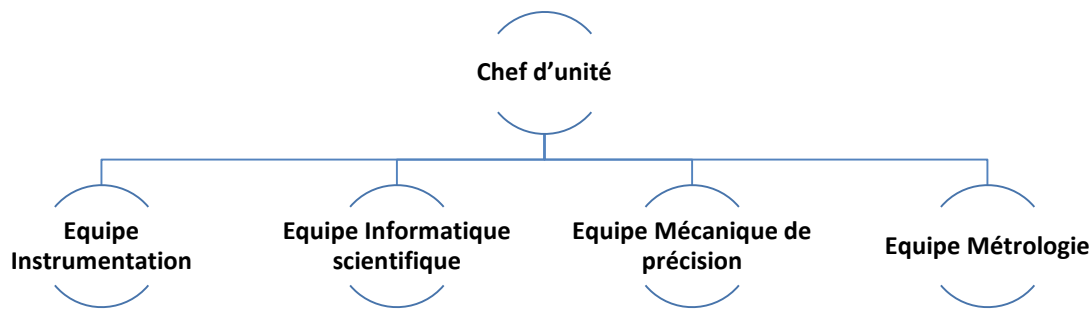


Figure 2 : Organisation de l'unité PICS

L'U.PICS appuie les unités de recherche dans la mise en place de plateaux d'expérimentation et développe des outils d'analyses de données. Elle assure également une veille technologique afin de proposer des solutions adaptées aux besoins futurs des unités. Ces solutions reposent, très souvent, sur l'emploi des Nouvelles Technologies de l'Information et de la Communication (NTIC).

## 2. Problématique

### 2.1. Mise en contexte

L'engagement militaire de la France durant ces quinze dernières années a conduit au déploiement des forces sur de nombreux théâtres extérieurs de forte intensité. Ces conflits armés soumettent les soldats à de nombreuses agressions pouvant conduire à des traumatismes physiques et psychologiques. L'engagement français en Afghanistan a notamment précipité les choses, 80 % des pathologies psycho-traumatiques dans l'armée surviennent après une mission dans ce pays.

L'amélioration de la prise en charge des blessés atteints de trouble stress post-traumatique (TSPT) est devenu un enjeu majeur pour les forces. Cette pathologie désigne un type de trouble anxieux sévère qui se manifeste à la suite d'une expérience vécue comme traumatisante avec une confrontation à des idées de mort.

La recherche de nouvelles solutions thérapeutiques est devenue multidisciplinaire notamment afin d'améliorer l'autonomie des blessés. En tant qu'établissement de recherche du SSA, l'IRBA a donc été mandaté afin de répondre à cet enjeu.

Au sein de l'IRBA, l'U.PICS est sollicitée pour ses diverses compétences dans la recherche de solutions technologiques et scientifiques. Ces solutions reposent, très souvent, sur l'emploi des Nouvelles Technologies de l'Information et de la Communication (NTIC). Les différents projets auxquels l'U.PICS a participé ont amené l'unité à s'interroger sur les applications émergentes de la robotique et des systèmes intelligents dans les domaines des sciences cognitives. Il a alors initié un programme de recherche et développement visant à améliorer le suivi des patients atteints de TSPT et d'alerter les professionnels de santé en cas de manifestation d'une « crise » de TSPT. Par ailleurs, l'unité propose d'explorer les capacités

d'interaction des robots dits « *sociaux* » afin de diminuer l'isolement des patients et faciliter leur réinsertion.

Pour répondre au besoin, l'U.PICS décide donc de proposer un dispositif de suivi médical et d'assistance en temps réel faisant appel à robot humanoïde compagnon capable :

- D'assurer des fonctions de surveillance au domicile de la personne ;
- D'assurer une fonction de prévention et de faciliter les actions du robot compagnon en lors situation de surcharge mentale du patient ;
- De participer à une action de soin et de faciliter la réappropriation de capacité chez des patients présentant des déficiences psychologiques, cognitives et/ou motrices.

Une première application sera d'assurer l'accompagnement de patients atteints de Troubles de Stress Post- Traumatique (TSPT), la finalité étant d'apporter une preuve de concept pour appliquer les résultats à d'autres pathologies, notamment les pathologies à forte prévalence en milieu militaire tel que le traumatisme crânien.

## 2.2.Problématique de l'unité

Le dispositif proposé par mon unité (Cf. Figure 3) doit répondre à deux types de problématiques:

- **Technologique** : le système doit pouvoir surveiller et accompagner au quotidien la personne afin de pouvoir détecter d'éventuelles crises de stress. Les données récoltées, doivent être accessibles, d'une part, par le médecin mais aussi par le chercheur avec différents niveaux d'accès.
- **Scientifique** : Améliorer la compréhension des interactivités d'un système robotique avec l'humain. Dans le cadre de cette étude, évaluer l'utilisabilité, l'acceptabilité et les capacités d'interactions avec des patients souffrant de TSPT.

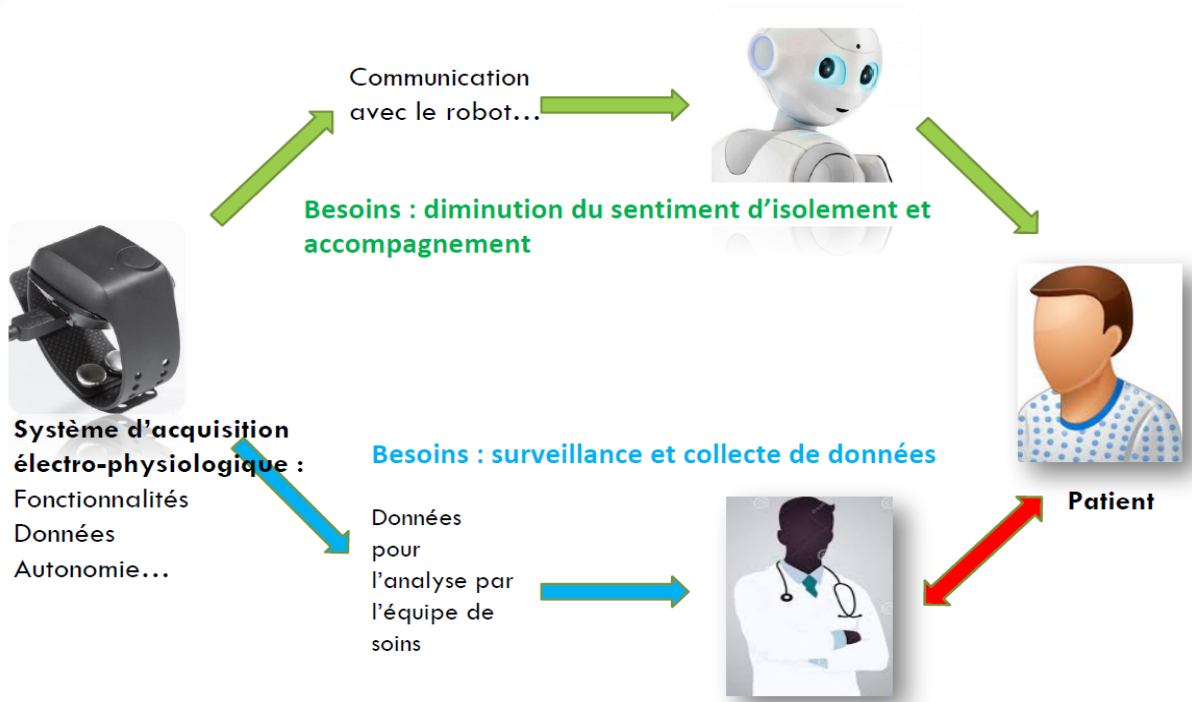


Figure 3 : Schémas plateforme

### 2.3. Travaux à réaliser

Le projet de l'unité consiste au développement d'un dispositif de suivi médical et d'assistance en temps réel faisant appel à un robot humanoïde à destination de personnes isolées présentant des déficiences psychologiques, cognitives et/ ou motrices. Le robot est un candidat pertinent en ce sens qu'il peut assurer des fonctions de surveillance, de suivi et d'aide à la pratique quotidienne de séances de régulation cognitive et émotionnelle chez les patients atteints d'un TSPT.

En pratique, le robot « d'assistance médicale » doit être capable de se déplacer vers le patient afin de maintenir des capacités d'interaction avec le patient (écouter et répondre), de proposer d'appeler une personne au téléphone ou de déclencher un appel d'urgence. Le robot doit également proposer une interface applicative robot-patient qui sera présentée sur l'écran tactile (environnement des applications thérapeutiques (gestion du stress ...)).

Au cours de ce stage, les actions suivantes sont attendues :

- Identifier les besoins et les exigences des fonctions du robot au regard des tâches définies dans le paragraphe précédent ;
- Développer un programme autonome de contrôle du robot en utilisant :
  - Le SDK fournis par le fabricant du Pepper (Logiciel Chorégraphe),
  - Le Langage : Python, C++ ;
- Développer une application Android pour contrôler le robot manuellement, afficher des informations et accéder aux applications thérapeutiques. Les contraintes seront les suivantes :
  - Développement d'un code Back-End => JAVA
  - Développement d'un code Front-End => XML

- Tester l'application sur un robot réel ;
- Rédiger une documentation au profit de l'unité afin qu'elle soit en mesure de poursuivre mon travail de stage.

### 3. État de l'art

Dans le cadre de l'ingénierie robotique, il est important de mettre en œuvre des procédures systématiques qui permettent de faciliter les évolutions futures du projet et de s'assurer du respect des contraintes imposées par le client. Cela englobe l'ensemble des activités de conception, de mise en œuvre des produits et des procédures tendant à rationaliser la production et son suivi. L'une de ses missions essentielles est d'appliquer la notion d'ingénierie système pour spécifier le code source et les outils spécifiés puis de les mettre en place.

#### 3.1. Présentation du robot

Le robot utilisé lors du stage est le robot Pepper, développé par SoftBank Robotics, Aldebaran. Pepper est un robot humanoïde conçu pour guider, accueillir, renseigner et discuter avec des personnes. Il est le premier robot humanoïde au monde capable d'identifier les visages et les principales émotions humaines. Pepper a été conçu pour interagir avec les humains de la façon la plus naturelle possible à travers le dialogue et son écran tactile.



Figure 4 : Robot Pepper

##### 3.1.1. Architecture physique

Le robot a les caractéristiques techniques suivantes :

- 1,21 m de haut ;
- Poids : 28 kg ;
- Une autonomie de douze heures au moins en utilisation ;
- 20 degrés de liberté (cf. Figure 5) ;
- 3 roues omnidirectionnelles, qui en font une plateforme holonome, disposée à chaque extrémité d'une base triangulaire.



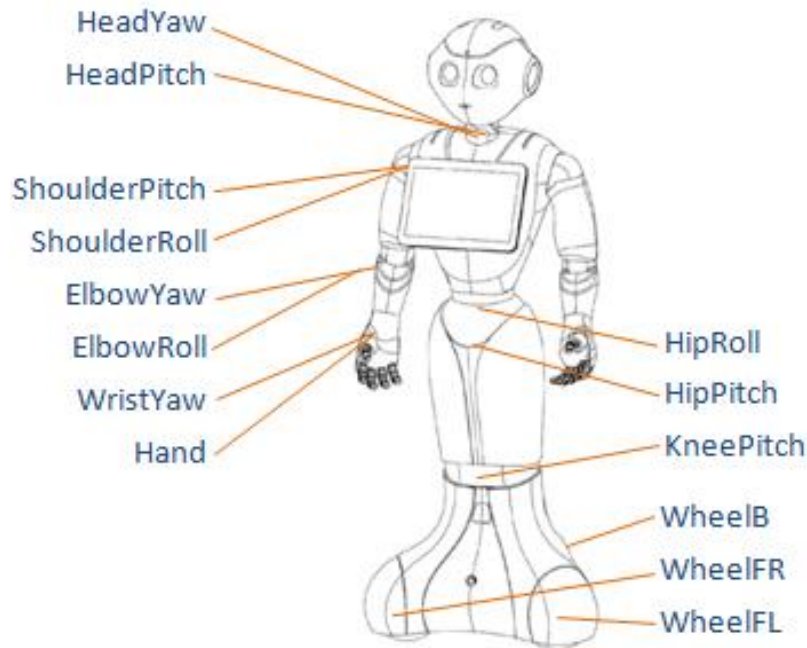


Figure 5 : Les degrés de liberté du robot

Pepper possède des capteurs sensorimétriques, la position de ces capteurs est indiquée dans la Figure 6, on trouve :

- Des capteurs tactiles situés sur le haut de la tête et les mains ;
- Des détecteurs de collision situés dans la base par-dessus les roues du robot ;
- Une tablette tactile 10 pouces fixée sur son torse (la tablette n'est pas amovible), qui possède son propre processeur mais peut interagir avec le système d'exploitation Naoqi ;
- Deux caméras couleur Omni Vision OV5640 situées en haut et en bas de la tête du robot ;
- Une caméra 3D Asus Xtion, de résolution maximale 320x240, avec une portée théorique de 0,8 m – 3,5 m et un champ de vue horizontal de 60°. La caméra est située derrière les yeux de Pepper (l'émetteur derrière un œil, le récepteur derrière l'autre) ;
- Deux sonars situés à l'avant et à l'arrière ; des émetteurs laser combinés avec des caméras infra-rouges pour la détection d'obstacles au sol.



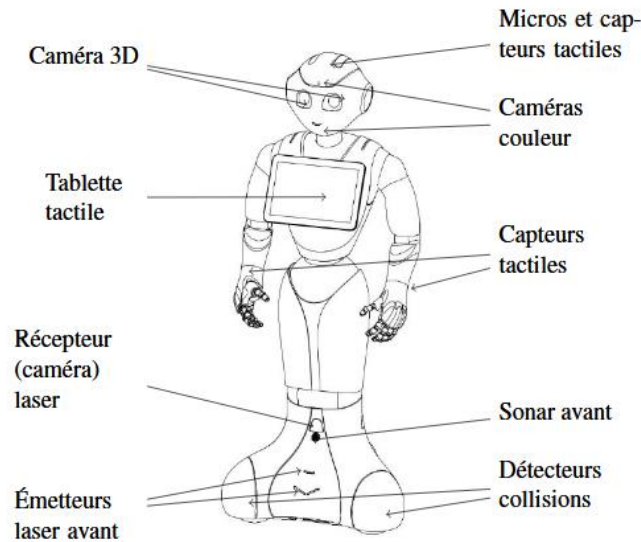


Figure 6 : Capteurs présents sur le robot Pepper

### 3.1.2. Fonctionnement du robot

Pepper est doté d'un mode "**Autonomous Life**" que l'on peut activer ou désactiver (Figure 7). Ce mode donne des capacités au robot afin de le rendre vivant et réactif. Dans ce mode, le robot peut déclencher des actions de base, de façon autonome, lorsqu'il détecte la présence d'une personne. Quand il entend ou voit l'individu, il tourne la tête ou son corps pour être face à la personne. Quand il a détecté un visage, les leds de ses yeux sont éclairées en rose. Il réagit également à d'autres stimulus via : la tablette, les capteurs sur la tête, les pare-chocs/boutons. Quand on parle au robot, il a des mouvements de tête et/ou de bras. De base, ce mode ne permet pas au robot de parler.

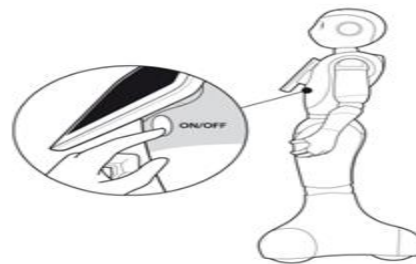


Figure 7 : Active ou désactive Autonomous Life

Le "**Basic Channel**" donne des capacités de discours au robot. Le robot peut répondre à des questions de base (Cf. Annexe II)

Avant de parler au robot, il faut que celui-ci ait détecté un visage (leds des yeux roses) et qu'il soit en mode écoute (leds des yeux tournent en bleu). Quand Pepper parle, il a des mouvements non-verbaux qui rendent l'interaction plus réaliste.

### 3.2. Méthode Ingénierie Système (IS)

Cette partie présente la manière dont ces concepts ont été appliqués à notre sujet d'étude par des méthodes et outils standards (Outils Ingénierie systèmes et Modélisation Logiciel)

pour permettre l'obtention d'une vue globale du système et de son comportement fonctionnel. Ce travail sert ensuite à faciliter l'application technique de l'étude d'une manière organisée et programmable. Ainsi, elle conduit à l'obtention d'un produit performant et facile à maintenir.

### 3.2.1. Analyse

La Figure ci-dessous montre d'une manière générale ce que le système doit offrir aux utilisateurs.

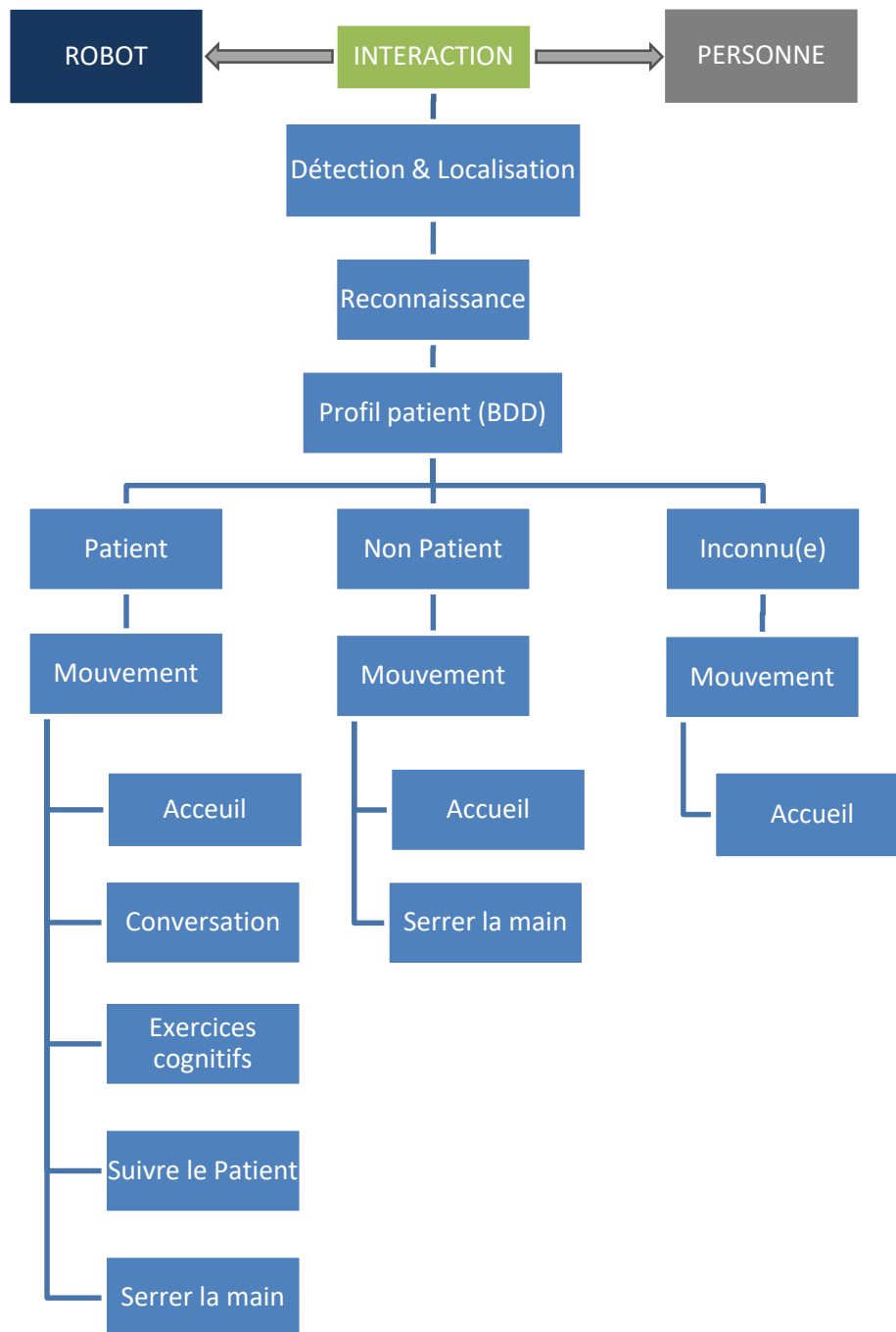


Figure 8 : Diagramme d'ordonnancement des tâches

La fonctionnalité principale du système est l'interaction entre le robot et les personnes. La première fonction afin de pouvoir interagir est la détection. Ensuite, le robot spécifie le profil de la personne détectée en trois catégories (patient, non patient et inconnus) grâce à une fonction de reconnaissance. Enfin, le robot réagit avec la personne (propose des exercices cognitifs, conversation, serrer la main, accueil, ...) selon son profil (Cf. Figure 8).

Pour la fonction de conversation, le vocabulaire auquel Pepper réagit est donc limité au "Basic Channel". Pour étoffer son langage, il est possible d'ajouter une conversation. Cependant, il est nécessaire de créer un script de questions et de réponses possible de l'utilisateur. Le chemin ci-dessous montre un exemple de plan de conversation (questions et réponses) que j'ai programmé entre le robot et le patient :

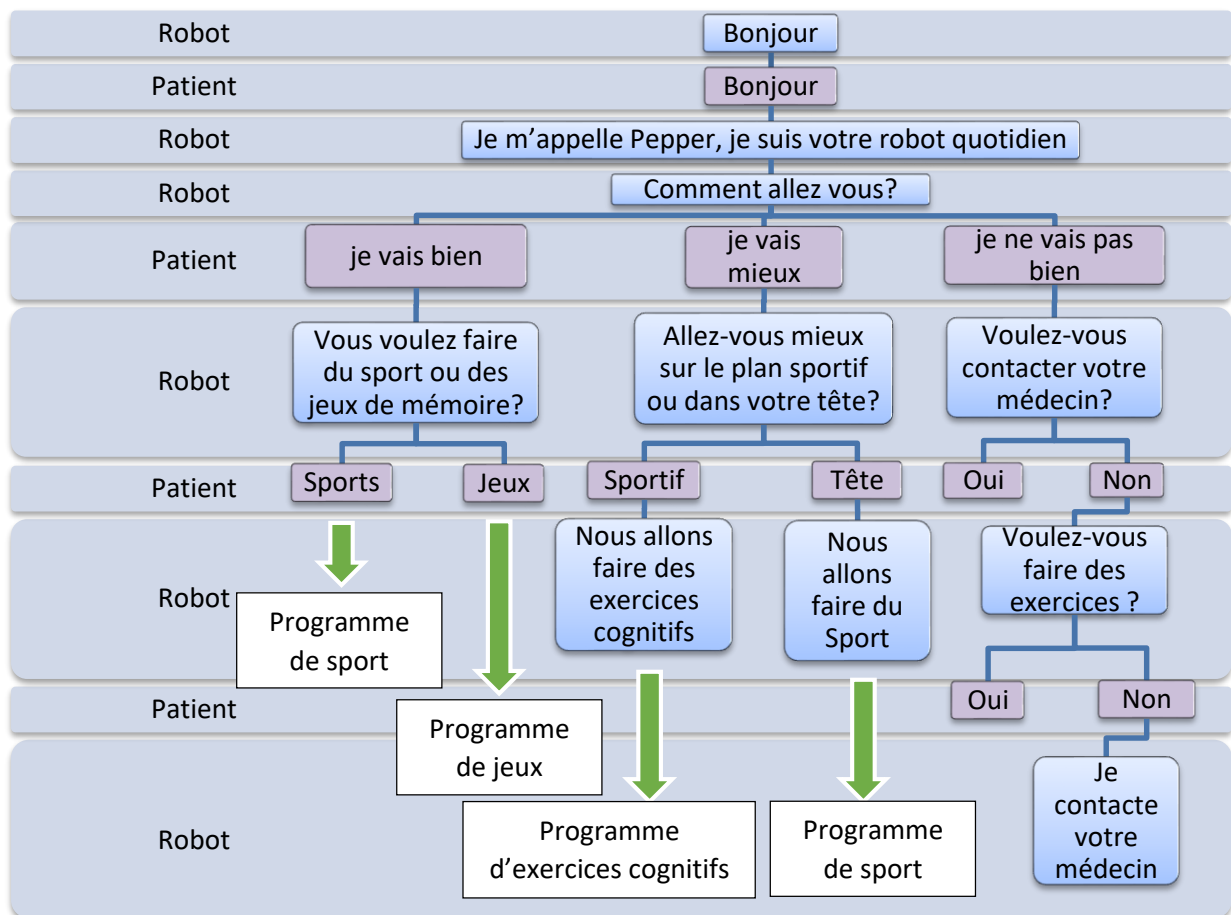


Figure 9 : Exemple de conversation Robot – Patient développée

### 3.2.2. Identification des exigences

En ingénierie, les exigences sont l'expression d'un besoin documenté sur ce qu'un produit ou un service particulier devraient être ou faire. Il faut différencier entre une exigence (système) et un besoin (client). Pour le développeur, l'exigence traduit le besoin qui doit être satisfait au sein du système.

En se basant sur les fonctionnalités listées dans la section 3.2.1 et aux caractéristiques du robot, je peux maintenant identifier les exigences de haut niveau du système. Le Tableau ci-dessous récapitule les exigences de notre système.

Identifiant	Exigences	Type	Priorité	Raison d'être	Exigences liées
EX01	Le robot doit être autonome	Fonction principale, contrainte	1	Pour faire une mission	EX03 EX04 EX05 EX06 EX07
EX02	Le robot doit être contrôlable par une application Android	Fonction principale, contrainte	1	Pour contrôler le robot manuel	EX03 EX04 EX05 EX06 EX07
EX03	Le robot doit se déplacer	Fonction principale	1	Pour déplacer vers le patient	EX04
EX04	Le robot doit détecter des personnes	Fonction principale, contrainte	1	Pour avancer vers le patient et lui discuter	EX03 EX05
EX05	Le robot doit bouger ses membres	Fonction principale	1	Pour effectuer plusieurs actions	EX03
EX06	Le robot doit écouter et comprendre la parole	Fonction principale	1	Pour discuter avec le patient	EX04 EX07
EX07	Le robot doit Parler et répondre aux questions	Fonction principale	1	Pour discuter avec le patient	EX04 EX06
EX08	Le robot doit rechercher des réponses sur internet	Fonction secondaire	2	Pour répondre au patient	EX06 EX07
EX9	Le robot doit reconnaître le patient après la détection	Fonction secondaire	2	Pour sortir et enregistrer des informations propres au patient	EX04 EX06
EX10	Le robot doit proposer des solutions au patient	Fonction principale	1	Pour faire baisser le stress	EX06 EX07

*Tableau 1 : Synthèse des exigences*

### 3.2.3. Langage d'implantation

Il est possible de programmer le robot via trois outils : le logiciel Chorégraphe, Naoqi API et Android SDK. Cette section est consacrée à la présentation de ces outils, de leurs avantages et de leurs limitations.

- **Chorégraphe**

Le logiciel Chorégraphe est un environnement de programmation graphique. La programmation se fait grâce à des boîtes de fonctions qu'il faut relier entre elles (Cf. Figure 10). Il est également possible de créer de nouvelles boîtes dans Chorégraphe en utilisant le langage Python ou C++.

Un point fort du logiciel Chorégraphe est la possibilité de tester rapidement les capacités du robot telles que la parole, la reconnaissance de mots, les mouvements et la visualisation des

caméras. Ce logiciel offre également des outils de réglages tels que le niveau de son, le niveau de batterie, l'activation et désactivation des moteurs. L'autre point fort du logiciel est la possibilité de simuler les actions du robot connecté sur des adresses IP et ports virtuels. Il est alors possible de visualiser les mouvements sur un robot virtuel. Cela permet d'effectuer des tests avant d'utiliser le robot réel au risque de l'endommager.

L'inconvénient principal de cet outil est qu'il est mal adapté pour développer des actions complexes et personnalisées.

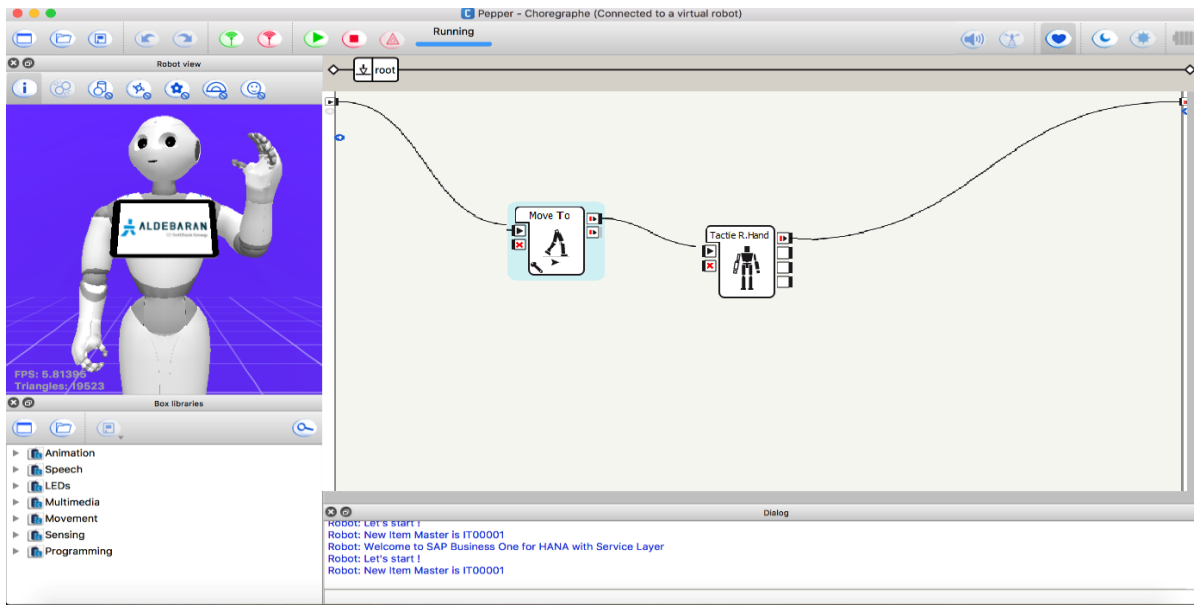


Figure 10 : Diagramme « Chorégraphe » qui fait déplacer le robot et lever le bras

- **Naoqi API**

Naoqi est une Interface de Programmation Applicative (API) qui permet aux utilisateurs de programmer et de commander le robot Pepper via des lignes de codes/commandes. Les API sont disponibles pour différents langages de programmation, tels que Python, C++ et JavaScript.

Programming Languages	Bindings running on		Choregraphe support	
	Computer	Robot	Build Apps	Edit code
Python	✓	✓	✓	✓
C++	✓	✓	⊘	⊘
JavaScript	✓	✓	✓	⊘

✓	OK
⊘	Not available

Tableau 2 : Langages de programmation pris en charge

Cet outil permet d'exploiter la totalité des capacités du robot. Néanmoins, il doit être couplé au logiciel Chorégraphe afin de bénéficier de l'outil de simulation.

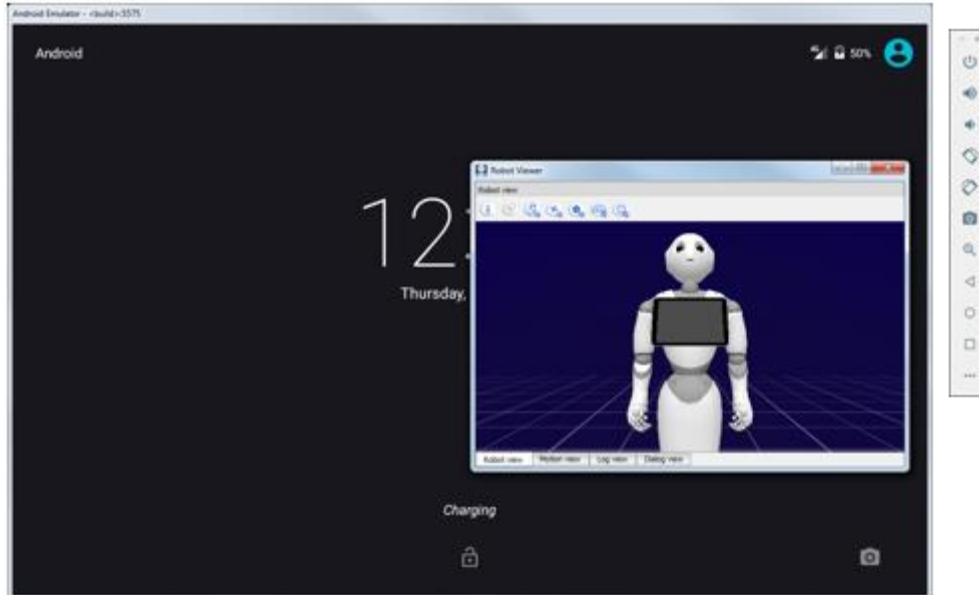
Plusieurs raisons m'ont conduit à retenir le langage Python afin de programmer le robot. La première est la compatibilité du Python avec Naoqi et logiciel Chorégraphe. Nous bénéficions de l'outil de simulation. Par ailleurs, le Python est utilisé par une communauté scientifique très active. Il est donc possible d'accéder à de nombreux forums pour résoudre certaines difficultés. Néanmoins, bien que ce langage soit facile à appréhender, la programmation du robot n'est pas aussi aisée que sous le logiciel Chorégraphe.

- **Android SDK**

Ce module est une bibliothèque qui offre la possibilité de créer des applications robotiques dans Android Studio. Il est mis à disposition par la société SoftBank Robotics et permet de contrôler le robot Pepper ou d'afficher des informations sur la tablette via une application Android. Ces applications peuvent être développées sous Java et XML. Il est également possible d'accéder facilement à des exemples d'animations que l'on peut modifier afin de développer des animations personnalisées.

Android SDK propose également un émulateur de périphérique virtuel Android (Cf. Figure 11) pour afficher un Pepper virtuel permettant la programmation et le mode débogage.

Après analyse ce module, destiné à un contrôle du robot via la tablette, n'est pas adapté au développement d'un comportement autonome du robot.



*Figure 11 : Emulateur du robot et de sa une tablette virtuel*

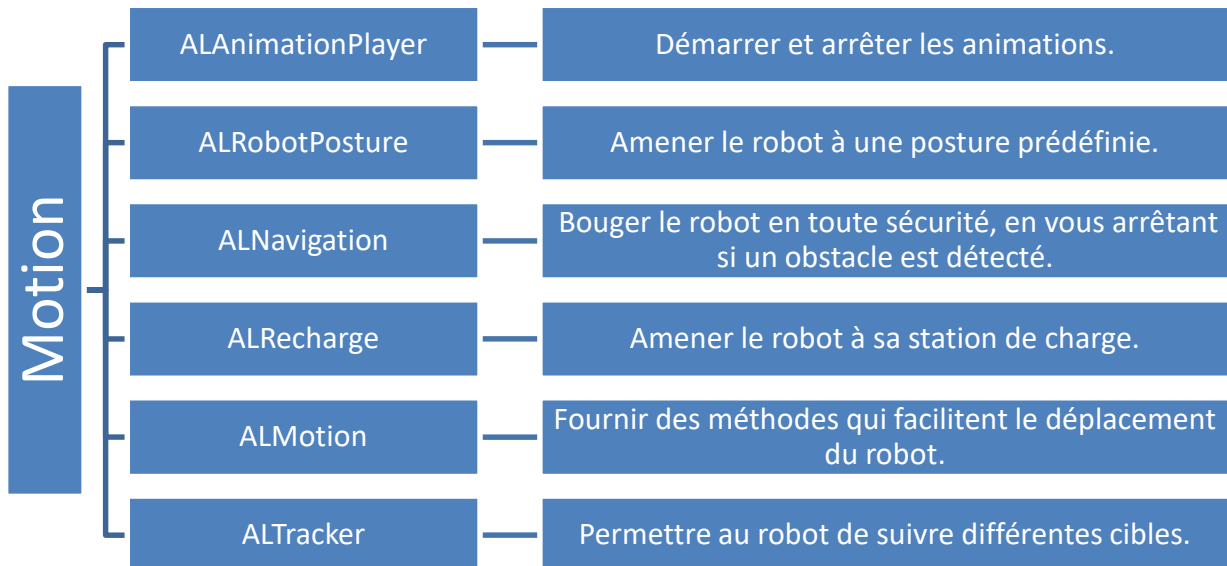
## 4. Application technique et résultats

Cette partie présente la concrétisation des exigences fonctionnelles, définies au Tableau 1, en solutions techniques. Cela se fait par l'implémentation du code physique qui assure les fonctionnalités du système.

### 4.1. Python Naoqi

**Qi** est un module libre qui permet d'intégrer la bibliothèque **Naoqi** (développée en C++) au langage Python. Le principal bénéfice de ce module est la possibilité de développer des applications pour contrôler le robot virtuel compatible avec le robot réel. En fonction de l'avancée du projet, **Qi** offre ainsi au développeur une grande flexibilité de communication sur des adresses IP et ports virtuels ou réels.

Ce module possède également certains modules et fonctions qui sont prédéfinis et prêts à être utilisés pour contrôler le robot. Ces modules sont divisés en plusieurs catégories :



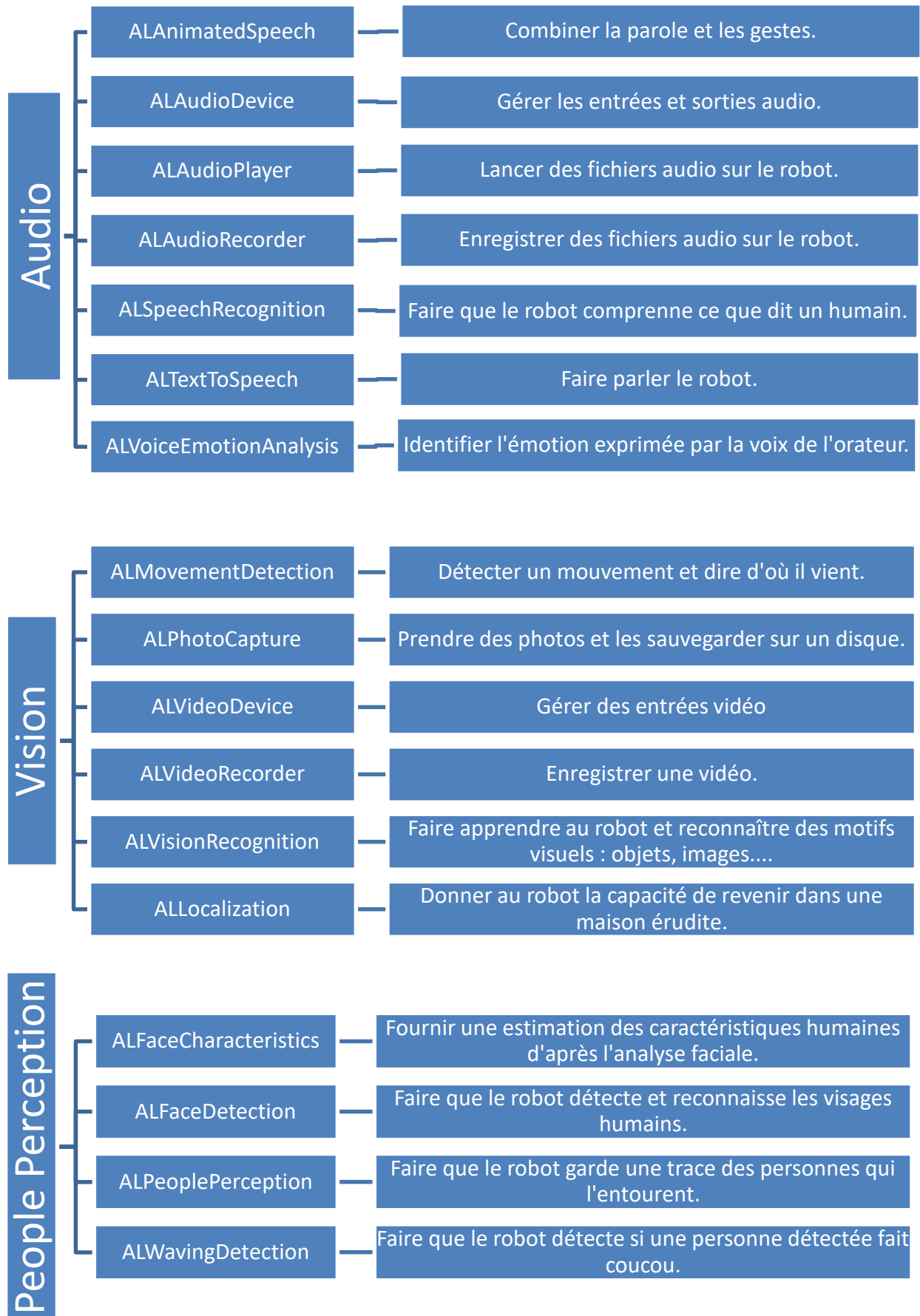


Figure 12 : Fonctions prédéfinies du module QI



Les différentes tâches du robot sont regroupées par famille d'action : dialogue, contrôle des mouvements, vision... Dans le langage python, les familles d'action sont modélisées par des classes. Une classe est l'unique moyen pour définir un nouveau type de fonctions et de variables propre au développeur.

Pour mon travail, J'ai donc développé quatre classes en langage python permettant de piloter les actions du robot. La Figure 13 illustre l'architecture et les relations entre ces quatre classes. Chacune de ces classes contient des fonctions (fonctions encadrées en orange) que j'ai développées et qui s'appuient sur des fonctions **Naoqi** présentées dans la Figure 12.

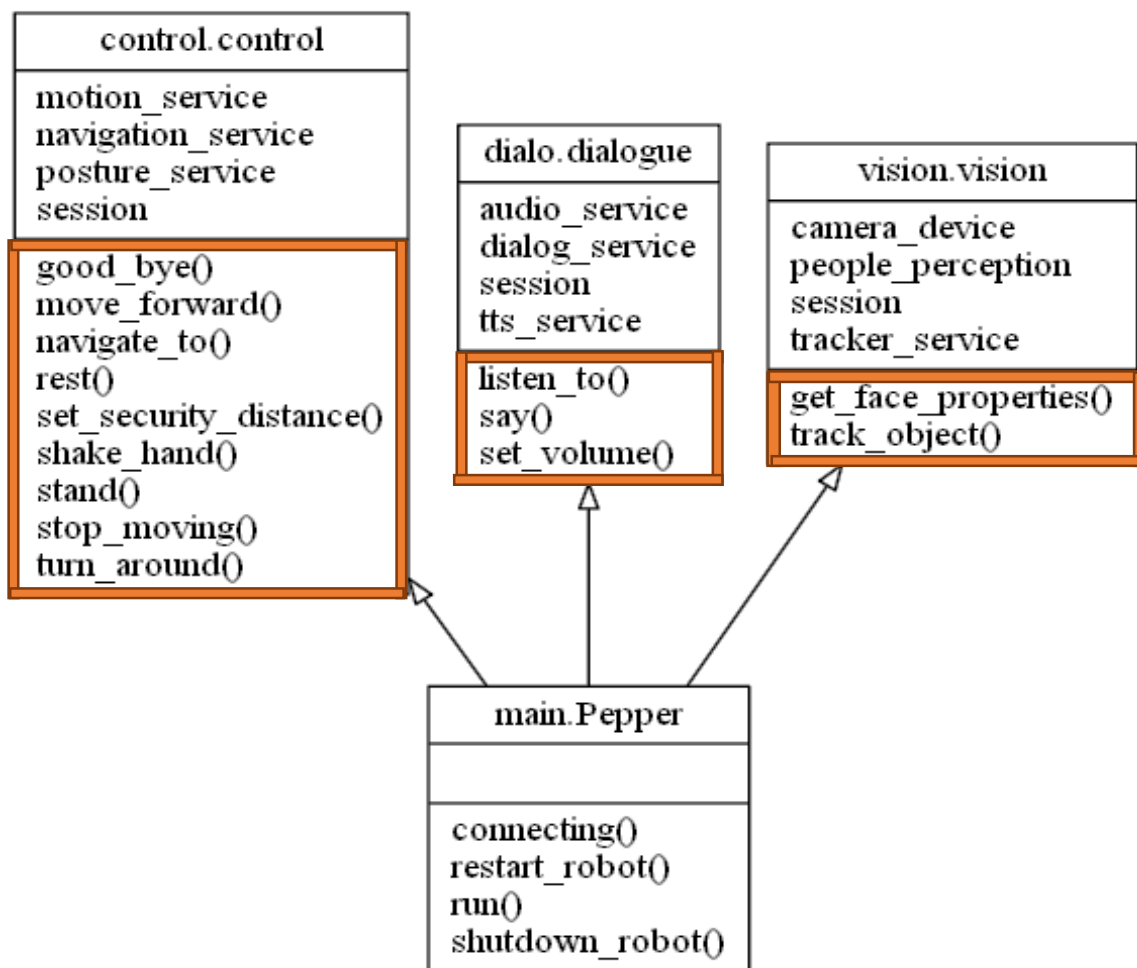


Figure 13 : Diagramme relationnel des classes de mon programme

### Classe Control :

Cette classe sert à contrôler les mouvements du robot (déplacements ou mouvements des membres du robot...). Elle répond aux besoins des exigences EX03 et EX05. Elle utilise les modules **Motion**, **Navigation**, **Posture** qui ont des fonctions prédéfinis. Elle comprend les fonctions suivantes :

- **Stand** : Mettre le robot en position debout par défaut ;
- **Set\_security\_distance** : Définir la distance de sécurité paramétrée en mètres ;

- **Navigate\_to** : Faire naviguer le robot vers l'emplacement spécifié sous forme d'une coordonnée (x, y) ;
- **Move\_forward** : Faire avancer le robot en face avec une vitesse définie ;
- **Turn\_around** : Faire tourner le robot autour d'un axe Z à une vitesse définie ;
- **Shake\_hand** : le robot serre la main de l'utilisateur, en ajustant les angles de ses membres en fonction du temps. Les angles des membres mobilisés sont :
  - « Roll (rotation) » du coude,
  - « Pitch (élongation) » de l'épaule,
  - « Pitch » de la hanche.

A la fin de cette fonction, le robot se remet à l'état de repos.

- **Good\_bye** : Dans cette fonction, le robot fait l'animation de « au revoir » en ajustant les angles de ses membres en fonction du temps. Les angles des membres mobilisés sont :
  - Roll de l'épaule,
  - Pitch de l'épaule,
  - Roll du coude.

A la fin de cette fonction, le robot se remet à l'état de repos.

### Classe vision

Cette classe comprend des fonctions de traitement d'image. Elle répond aux besoins des exigences EX04 et EX09. Elle permet de faire du traitement automatique après une phase d'apprentissage d'objets dans son environnement. Elle contient les fonctions suivantes :

- **Tracker\_object** : permet au robot de suivre une cible désignée (balle rouge, visage, point de repère, etc.) en mobilisant différents membres (tête seule, corps entier, mouvement, etc.). Cette fonction est paramétrée par deux variables liées à la cible et aux membres mobilisés.
- **Get\_face\_properties** : récupère les propriétés du visage de la personne détectée. Les différentes informations « évaluées » dans le visage sont :
  - Les motions (neutre, joyeux, surpris, en colère et triste),
  - Âge,
  - Sexe.

### Classe dialogue

Cette classe sert à faire le dialogue entre le robot et l'utilisateur. Elle répond aux besoins des exigences EX06 et EX07. Elle utilise les modules et fonctions définies par SoftBank pour créer mes propres fonctions :

- **Listen\_to** : Cette fonction contrôle la fonctionnalité « écoute » du robot sur des questions prédéfinies. La réponse vocale du patient est alors comparée à une liste de mots prédéfinis dans les paramètres de la fonction **Listen\_to**.
- **Say** : Cette fonction permet au robot de « parler ». Il exprime un texte défini dans les paramètres de la fonction Say.

- **Set\_volume** : Réglage du volume de la voie définie en pourcentage du volume maximal.

**Classe Pepper** : La classe Pepper est la classe principale du programme. Toutes les sous-classes (contrôle, dialogue, vision ...) héritent des fonctions de la classe « main.Pepper ». Elle fait la connexion avec le robot (virtuel ou réel). Ensuite, elle utilise les fonctions déjà définies dans les autres classes pour créer la mission.

## 4.2. Android SDK

Le robot humanoïde incorpore une tablette sur sa poitrine. Dans les précédentes versions software du robot, cette tablette servait uniquement d'interface de communication entre le robot et l'utilisateur. Cette tablette ne permettait pas un contrôle du robot. A présent, la tablette est devenue une interface de contrôle de l'ensemble du corps.

A l'issue de ma première moitié de stage, j'ai pu appréhender les principes de bases de création d'une application. Dans cette section, je présente un aperçu rapide des étapes nécessaires à la création d'une application pour le robot humanoïde. Les étapes sont les suivantes :

1. Installer « Android Studio » et le « SDK Android ».
2. Téléchargez le plugin « Pepper SDK pour Android Studio ».
3. Installer le « Qi SDK ».
4. Créer un nouveau projet dans « Android Studio ».
5. Créer une application robot.

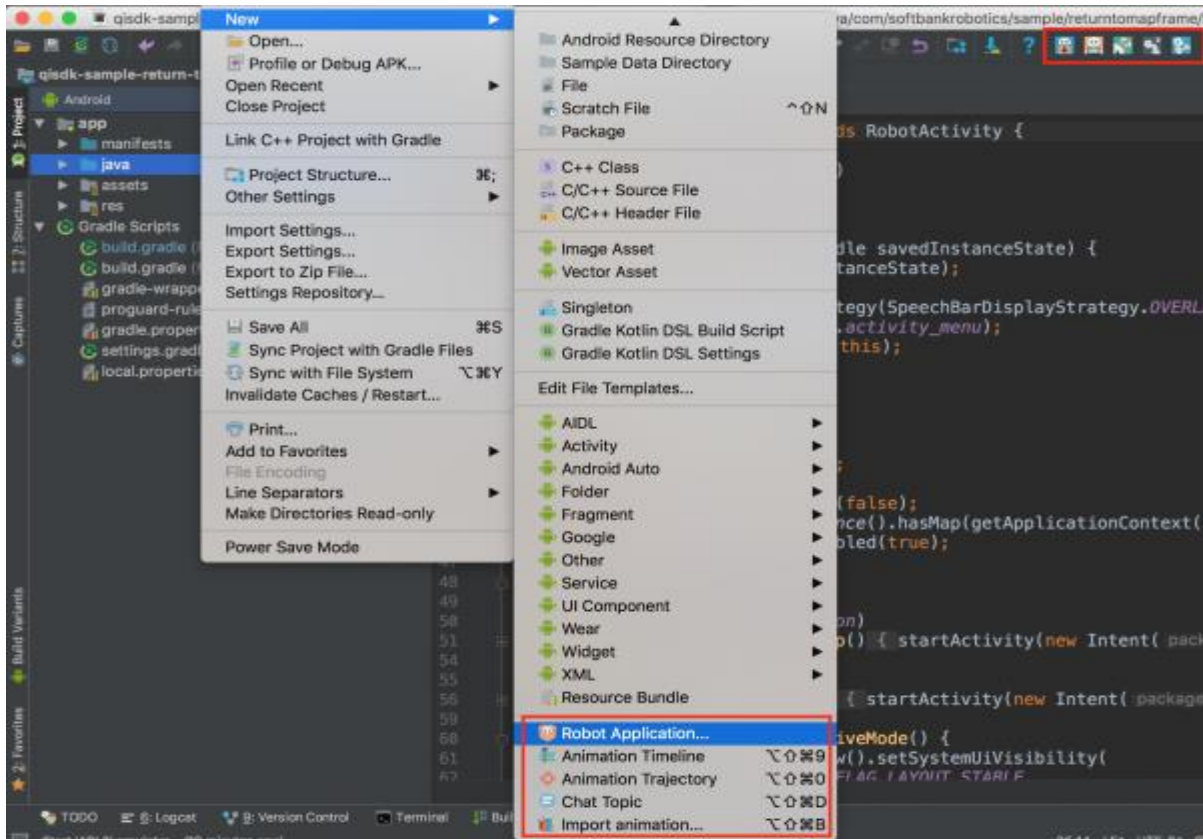


Figure 14 : Lancement du Plugin Pepper dans Android Studio

Sachant que le simulateur virtuel du robot possède un émulateur de conversation, la première tâche qui m’a été confiée a été de faire une analyse des capacités de dialogue entre un Patient et un Robot proposées par l’outil Android SDK. Pour cet objectif, le SDK Pepper possède un **Chatbot** qui est un objet permettant de faire dialoguer le robot selon des phrases humaines. La réaction à ce dialogue peut prendre différentes formes (verbale, mouvement ...) ou un mélange de celles-ci.

Ce **Chatbot** est utilisable dans les trois cas suivant :

- *Cas 1* : pour gérer une conversation de type questions/réponses avec l’utilisateur sur quelques questions génériques présent par défaut dans le robot ;
- *Cas 2* : QiChatbot, pour gérer et définir une base de donnée « questions /réponses » personnalisée,
- *Cas 3* : pour gérer des questions appelant Wikipédia.

Dans notre cas, le robot doit proposer une liste de dialogues telle que présentée dans la Figure 9. Pour cela, j’ai choisi le **QiChatbot** qui me permet de personnaliser les questions. Les principales caractéristiques de QiChatbot sont :

- La reconnaissance vocale intégrée ;
- La reconnaissance vocale à distance qui permet d’améliorer la performance de la reconnaissance via un accès à internet. Je n’ai pas encore identifié comment la performance était augmentée.

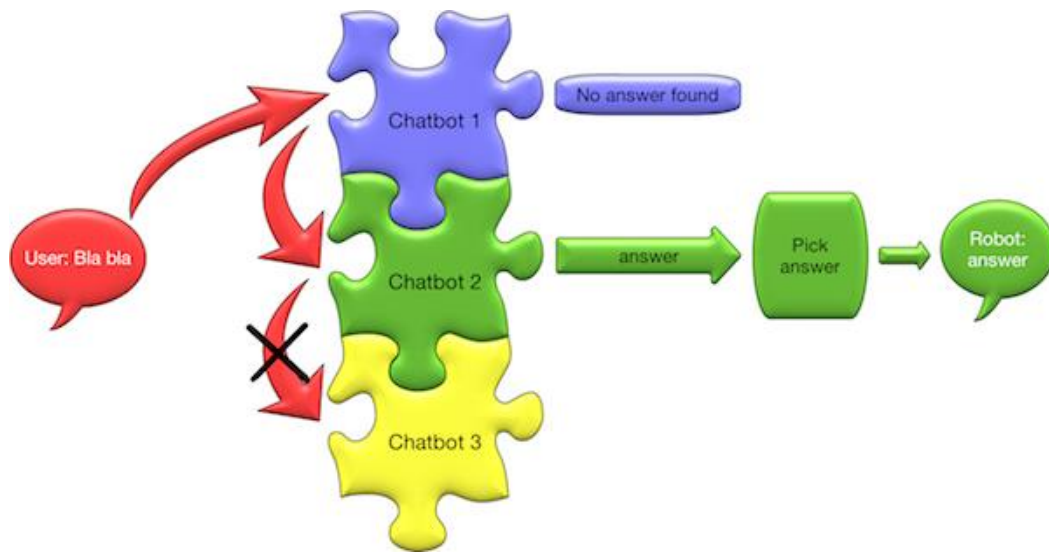


Figure 15 : Diagramme représentant le flux de conversation

## 5. Avancement de travail

Le tableau ci-dessous, montre mon avancement de travail selon les tâches réalisées et les tâches à venir.

	<i>Taches</i>	<i>Date de début</i>	<i>Durée(jour)</i>	<i>Date de fin</i>	<i>Réalisé</i>
<b>1</b>	<b>Etude avant-projet</b>				
1.1	Problématique	04/06/2019	4	07/06/2019	100 %
1.2	Etude ingénierie systèmes	08/06/2019	7	14/06/2019	100 %
1.3	Modélisation de système	15/06/2019	6	20/09/2019	100 %
1.4	Etude de l'état de l'art	21/06/2019	4	24/06/2019	100 %
<b>2</b>	<b>Réalisation et implémentation</b>	<b>25/06/2016</b>		<b>30/11/2019</b>	
2.1	Installation des logiciels	25/06/2019	1	26/06/2019	100%
2.2	Tutorial sur les logiciels	26/06/2019	4	30/06/2019	100%
2.3	Connexion Naoqi avec le robot	30/06/2019	1	01/07/2019	100%
2.4	Test de plusieurs fonctions basiques fournies avec le robot	01/07/2019	1	02/07/2019	90%
2.5	Déplacement du robot	02/07/2019	2	04/07/2019	100%
2.6	Mouvement des membres	04/07/2019	3	07/07/2019	100%
2.7	Création de la fonction « serre la main »	07/07/2019	2	09/07/2019	90%
2.8	Modélisation du dialogue	09/07/2019	1	10/07/2019	70%
2.9	Implémentation code Python dialogue	10/07/2019	2	12/07/2019	50%
2.10	Modélisation de la partie vision	12/07/2019	1	13/07/2019	50%
2.11	Implémentation code Détection	13/07/2019	2	15/07/2019	50%
2.12	Installation environnement Android	16/07/2019	1	17/07/2019	100%
2.13	Création application de Test de l'environnement	17/07/2019	1	18/07/2019	100%
2.14	Comprend le code tutorial fourni par SoftBank	19/07/2019	3	22/07/2019	70%
2.15	Création d'une application Android de dialogue	22/07/2019	20	10/08/2019	60%
2.16	Ecriture du rapport de stage	22/07/2019	20	10/08/2019	100%
2.17	Amélioration de l'application Android (dialogues et pilotage manuel)	10/08/2019	30	10/09/2019	10%
2.18	Amélioration des codes en fonction des tests sur le robot réel	10/09/2019	60	10/11/2019	0%
2.19	Rédaction des documentations	10/11/2019	20	30/11/2019	0%

*Tableau 3 : Liste d'avancement des différentes tâches avec les durées*

## Conclusion

Ce stage de fin d'études réalisé au sein de l'Institut de recherche biomédicale des armées a constitué pour moi une expérience très enrichissante sur les plans professionnel, intellectuel et personnel. Grâce à cette expérience, j'ai pu approfondir et élargir mes compétences techniques et organisationnelles dans le cadre du développement logiciel et robotique.

Le bilan intermédiaire est très positif. Le premier objectif qui consistait à prendre en main et implémenter des fonctions de base a été atteint. A cet effet, j'ai testé différents outils de contrôle dans un environnement virtuel. Sur la base de modules présents dans Naoqi, j'ai développé plusieurs fonctions de commande, en langage Python. Elles permettent ainsi au robot d'effectuer certaines tâches simples en autonomie. Par ailleurs, j'ai initié le développement d'une application Android. Elle permet d'interagir avec le robot via sa tablette intégrée.

La suite de mon travail consiste à implémenter mes fonctions sur un robot réel et identifier tous les problèmes auxquels peut être confronté le robot (ex : difficulté de déplacement en fonction du revêtement du sol tel que la moquette). Je dois également poursuivre le programme d'interaction avec le patient sous environnement Android. Enfin, il me faudra rédiger une documentation précise sur les codes que j'ai développés ainsi qu'une notice d'utilisation dans le but d'une reprise de mon travail.

Sur un plan personnel, ce travail m'a permis d'obtenir une bonne expérience en développement d'application robotique. De plus, l'acquisition de compétences dans le développement d'applications Android pourra m'offrir des opportunités professionnelles dans les domaines de la robotique et du développement informatique. Par ailleurs, la mise en application de la modélisation logicielle et de l'ingénierie système m'a permis d'effectuer mes premiers pas dans le domaine de la résolution et de l'analyse des systèmes complexes. Enfin, ce projet au sein d'un institut de recherche m'a enrichi l'esprit à travers le travail en équipe et m'a permis de prendre conscience de mes capacités d'adaptation. Ces qualités sont aujourd'hui incontournables pour la suite de ma vie professionnelle en France.





## Bibliographie

- Alberto Giaretta, M. D. ( 4 Jul 2018). *Adding Sal tto Pepper, A Structured Security Assessment over a Humanoid Robot* . arXiv:1805.04101v2 [cs.CR] .
- Santano, S. (2018 , October 19). *When Pepper met Android*. Récupéré sur Inovex Blog:  
<https://www.inovex.de/blog/when-pepper-met-android/>
- SoftBank Robotics. (s.d.). *All Module NAOqi APIs*. Récupéré sur Aldebaran documentation :  
<http://doc.aldebaran.com/2-4/naoqi/index.html>
- SoftBank Robotics. (s.d.). *Android - QISDK - Chatbot*. Récupéré sur Aldebaran documentation :  
[https://android.aldebaran.com/sdk/doc/pepper-sdk/ch4\\_api/conversation/reference/chatbot.html](https://android.aldebaran.com/sdk/doc/pepper-sdk/ch4_api/conversation/reference/chatbot.html)
- SoftBank Robotics. (s.d.). *Android - QISDK - QiChatbot*. Récupéré sur Aldebaran documentation:  
[https://android.aldebaran.com/sdk/doc/pepper-sdk/ch4\\_api/conversation/reference/qichatbot.html](https://android.aldebaran.com/sdk/doc/pepper-sdk/ch4_api/conversation/reference/qichatbot.html)
- SoftBank Robotics. (s.d.). *Supported programming languages*. Récupéré sur Aldebaran documentation : [http://doc.aldebaran.com/2-4/dev/programming\\_index.html](http://doc.aldebaran.com/2-4/dev/programming_index.html)
- SoftBank Robotics. (s.d.). *using Python*. Récupéré sur Aldebaran documentation :  
[http://doc.aldebaran.com/2-4/getting\\_started/helloworld\\_python.html](http://doc.aldebaran.com/2-4/getting_started/helloworld_python.html)
- Trouble De Stress Post-Traumatique*. (2019, Avril 28). Récupéré sur Wikipedia:  
[fr.wikipedia.org/wiki/Trouble\\_de\\_stress\\_post-traumatique](fr.wikipedia.org/wiki/Trouble_de_stress_post-traumatique).
- Vaugrente, A. (2015, Février 4). *www.pourquoidocteur.fr*. Récupéré sur  
<https://www.pourquoidocteur.fr/Articles/Question-d-actu/9711-Stress-post-traumatique-dans-l-armee-une-epidemie-silencieuse>
- Wirbel, É. (2014). *Localisation et navigation d'un robot humanoïde en environnement domestique*. *Robotique[cs.RO]*. Paris: Ecole Nationale Supérieure des Mines de Paris.



## Annexe I - Planning des activités

Voici le diagramme de Gantt des travaux prévus jusqu'à présent :

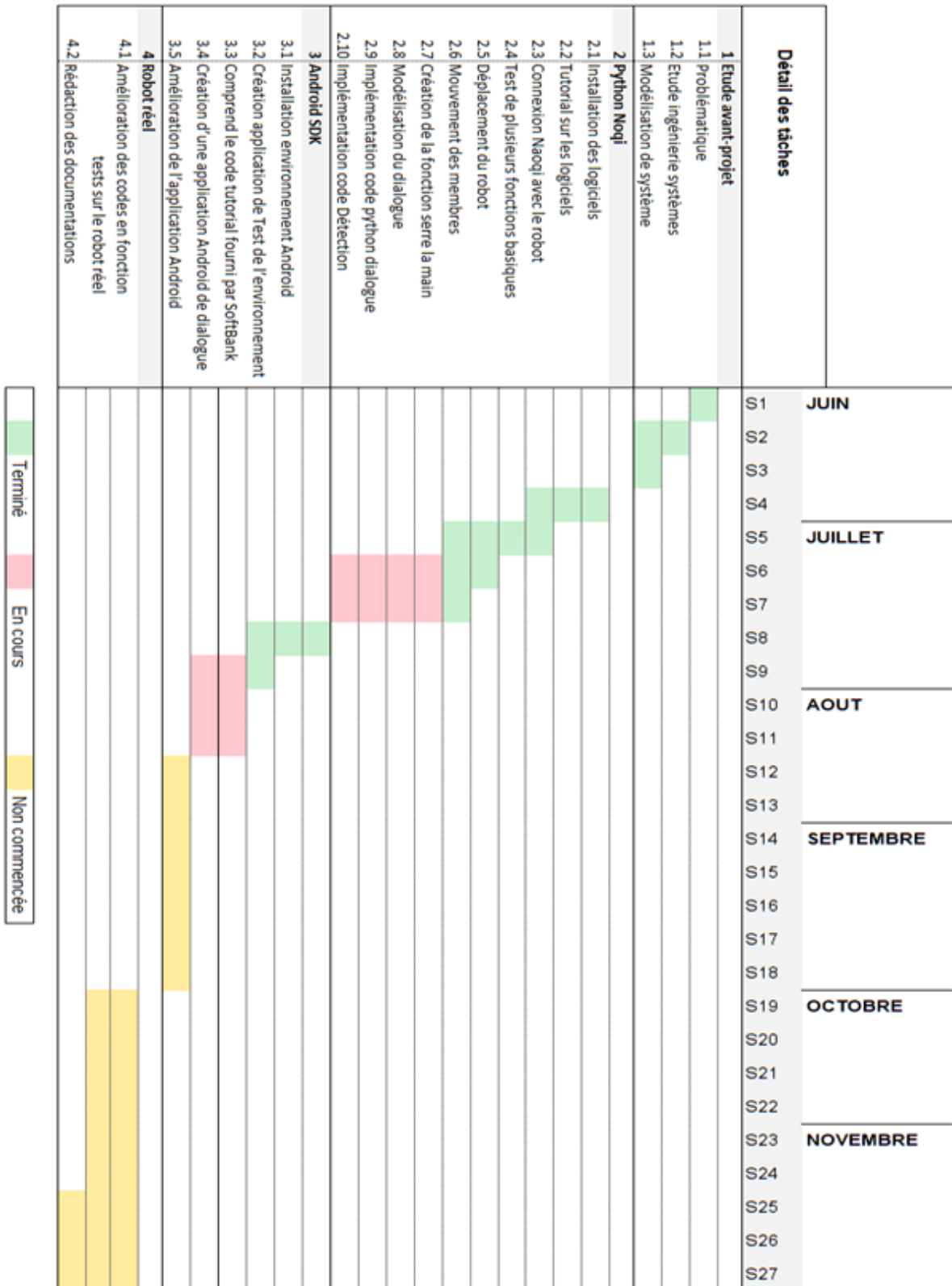


Tableau 4 : Diagramme de Gantt

## Annexe II - Dialogue de base : "Basic Channel"

Pepper a un dialogue de base : le "Basic Channel". La liste des questions de base ci-dessous n'est pas exhaustive. Pepper peut reconnaître les questions sous une autre forme. Par exemple, Pepper peut comprendre "Comment puis-je t'appeler ?" ou "Comment tu t'appelles ?". La liste correspond à la version Naoqi 2.5.

<b>Rencontre avec Pepper</b>	
Comment ça va?	Fais-moi coucou.
<b>A propos des applications</b>	
Que sais-tu faire?	Quelles sont tes applications?
<b>Répétition</b>	
Qu'est-ce que j'ai dit ?	Tu peux répéter?
<b>Niveau de batterie</b>	
Quelle est ton niveau de batterie?	Es-tu en charge?
<b>Connexion Internet</b>	
Quelle est ton adresse IP ?	Es-tu connecté à Internet?
<b>Date et temps</b>	
Quelle heure est-il?	Quel jour on est?
<b>Langues</b>	
Quelle langue parles-tu?	Est-ce que tu sais parler <nom d'une langue>?
Parle <nom d'une langue>.	
<b>Volume</b>	
Parle moins fort.	Parle plus fort.
<b>Mouvements de Pepper</b>	
Tends les bras (le bras gauche/droit).	Peux-tu ouvrir les mains (gauche/droite)?
Bouge la tête (à gauche/droite).	Regarde-moi.
Lève les bras (le bras gauche/droit).	Peux-tu fermer les mains (gauche/droite)?
Tourne la tête (à gauche/droite).	Regarde ailleurs.
Fais bouger tes mains (gauche/droite).	Mets tes mains (gauche/droite) en arrière.
Lève la tête.	Baisse les yeux (à gauche/droite). Mains en l'air!
<b>Caractéristiques de Pepper</b>	
Comment tu t'appelles?	Est-ce que je peux te faire confiance ?
Quel est ton surnom?	Es-tu humain?
Présente-toi.	Es-tu vivant?
Quelle taille fais-tu?	Es-tu conscient?
Combien pèses-tu?	Est-ce que tu peux penser?
Quel âge as-tu?	Peux-tu ressentir des émotions?
Quand est ton anniversaire?	Comment détectes-tu les émotions?
Es-tu un garçon ou une fille?	Est-ce que tu manges?
Tu es en quelle matière?	Est-ce que tu peux parler?
De quelle couleur es-tu?	Est-ce que tu peux me voir?
Pourquoi tu as une tablette?	Est-ce que tu peux m'entendre?

Es-tu gentil ?	Est-ce que tu peux me comprendre?
Es-tu cool?	Est-ce que tu peux me reconnaître?
Es-tu intelligent?	
<b>A propos des robots</b>	
Es-tu un robot?	Qui est-ce qui t'a appelé comme ça?
C'est quoi un robot?	Qui t'a fabriqué?
C'est quoi un robot humanoïde?	Où as-tu été fabriqué?
Est-ce que les robots vont remplacer les humains?	As-tu de la famille?
Connais-tu les lois de la robotique?	As-tu des amis?
Connais-tu Nao?	Comment s'appelle ton ami?
Connais-tu Roméo?	Je peux être ton ami?
Pourquoi tu t'appelles Pepper?	