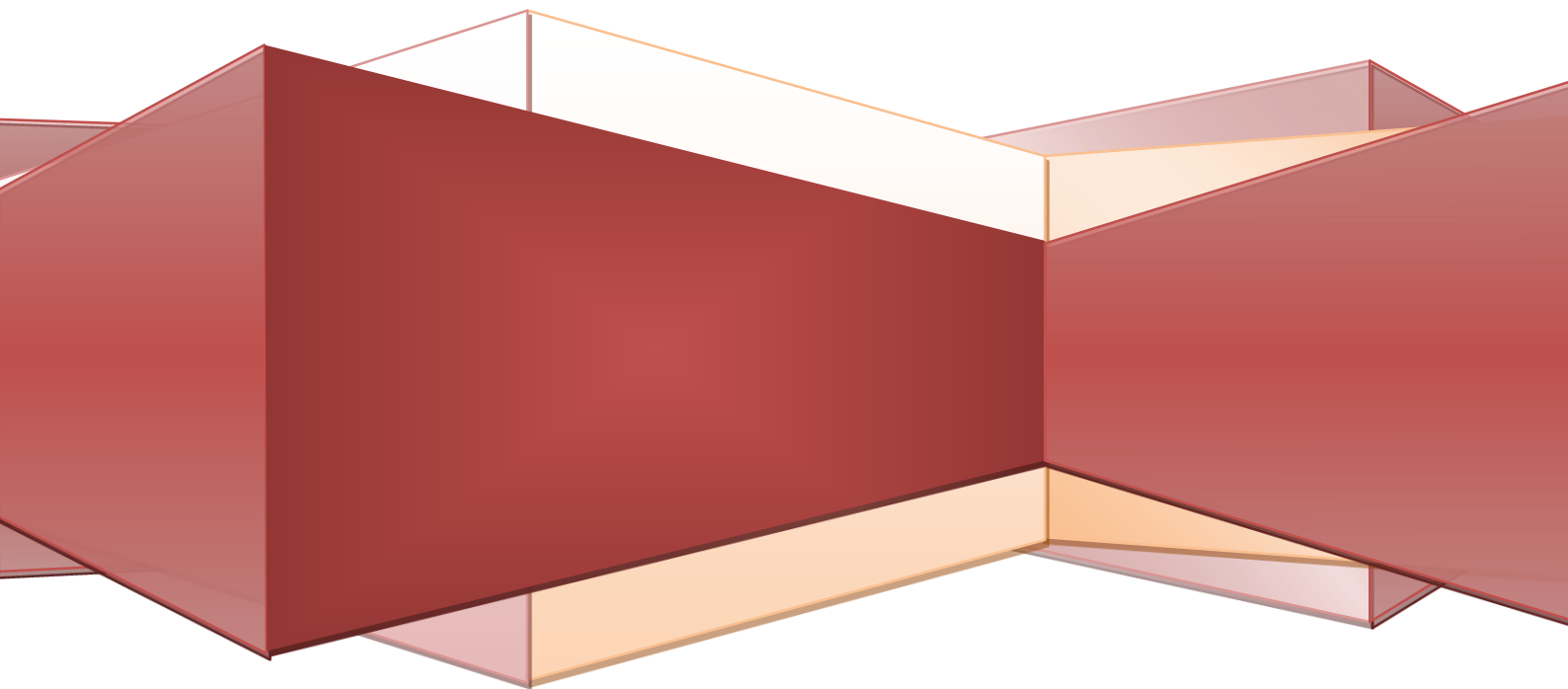




**PROJET:**  
**ROBOT R2D2**

**ISEN** | école  
d'ingénieurs

# Rapport de projet : Robot R2d2





## Table des matières

<b>Présentation du projet.....</b>	<b>3</b>
<b>Gestion de projet.....</b>	<b>4</b>
<b>Étude de l'existant.....</b>	<b>4</b>
<b>Objectifs fixés.....</b>	<b>5</b>
Principaux.....	5
Optionnels.....	6
Formalisation .....	6
<b>Étude du projet .....</b>	<b>6</b>
.....	25
<b>Choix matériels .....</b>	<b>32</b>
Choix des batteries.....	33
Choix des matériels divers .....	35
<b>Choix des technologies et des ports .....</b>	<b>35</b>
Technologies .....	35
Ports utilisés.....	36
<b>Chiffrage du projet .....</b>	<b>36</b>
<b>Planning prévisionnel .....</b>	<b>37</b>
<b>Annexes.....</b>	<b>37</b>
<b>Cahier des charges.....</b>	<b>Erreur ! Signet non défini.</b>



**PROJET:**  
**ROBOT R2D2**

RP

**ISEN** | école  
d'ingénieurs

## Rapport de projet

### Présentation du projet

L'an dernier un projet M1 a consisté à développer un robot similaire à R2D2 de la saga Star Wars pour participer à la promotion de l'école (journées portes ouvertes, salons) en attirant l'attention du public. Le projet a permis de créer le robot et de développer une partie « sons et lumières ».

Le but du projet de cette année est de s'occuper du déplacement du robot (de façon totalement séparée de l'existant). Le développement s'effectuera sur une carte BeagleBone. La partie robotique sera en partie fournie. Le travail demandé consiste à actualiser la partie robotique, faire la liaison entre la carte BeagleBone et la partie robotique et à faire se déplacer R2D2.



**PROJET:**  
**ROBOT R2D2**

RP

**ISEN** | école  
d'ingénieurs

## Rapport de projet

### Gestion de projet

Dans cette partie figurera l'intégralité des recherches qui ont été réalisées pour permettre une totale compréhension du projet et de la problématique fixée, ainsi, les différentes études menées nous ont permis de définir le cadre et les objectifs à remplir pour permettre de remplir toutes les attentes concernant le projet du robot R2D2.

### Étude de l'existant

#### Le fonctionnement existant :

Un robot R2D2 a été créé l'année dernière afin de servir de promotion sur les forums post-bac. Il possède actuellement une partie Son & Lumière commandée par une carte Beaglebone.

Le projet consiste à ajouter une partie Déplacement totalement indépendante de la partie Son & Lumière visant à permettre de contrôler le déplacement du robot via ordinateur avec un câble ou via wifi/bluetooth avec un smartphone.



### Le périmètre du projet :

Le robot R2D2 actuel est constitué d'une carcasse en cuivre qui contient la carte beaglebone ainsi que diverses LED et buzzers.

Le mode de déplacement retenu consiste en une plateforme de mobilité constituée d'un support physique et de 4 roues dont 2 seront propulsées par des moteurs, plateforme qui sera fixée sur le bas du robot.

La carte Beaglebone contient une distribution Debian spécialement adaptée. Les communications avec les moteurs se feront au travers d'un port série de la beaglebone qui sera relié au contrôleur des moteurs.

Les utilisateurs finaux du robot seront essentiellement les professeurs et ambassadeurs chargés d'aller aux forums de recrutement.

### Objectifs fixés

Le robot actuel servant pour les portes ouvertes de l'établissement, la carte BeagleBone ainsi que les câbles servant à la transmission de données devront être remis avant chaque porte ouverte.

Des réunions d'avancement devront également avoir lieu de façon hebdomadaire, afin de tenir informés les clients de l'évolution du projet et des éventuels problèmes rencontrés.

### Principaux

L'objectif principal est le mouvement du robot R2D2. Ce mouvement devra être contrôlable depuis un ordinateur branché directement en USB et par le biais d'une application smartphone qui sera développé pour également contrôler le robot. Le robot doit pouvoir être évolutif.



## Optionnels

Si les objectifs principaux seront atteints (réalisés, testés, validés). On pourra se tourner vers des objectifs plus optionnels comme un mode autonome et/ou intelligent pour le robot R2D2.

## Formalisation

Pour permettre une meilleure compréhension de la part des étudiants (Mingam Pierre-yves et Toquer François) ainsi que des professeurs référents (Mme Leonard et M. Aubry), nous mettrons en place un système de documents type pour les comptes rendu de réunions par exemple.

## Étude du projet

Pour commencer l'étude, nous avons réalisé le « QQQQCP ». Ce document nous paraissait le plus adapté pour dégrossir le projet et obtenir une entrevue du projet.

### QQQQCP :

#### Quoi :

- Gérer le déplacement d'un robot R2D2 via une plateforme de mobilité (roues + moteurs) ordonnée via ordinateur et/ou smartphone

#### Qui :



## Rapport de projet

- Utilisateur lambda (professeurs + ambassadeur)

==> Nécessité d'un système d'authentification

### Où :

- Support physique plat (sol habituel : lino, parquet, carrelage) dans un environnement contrôlé (forums ou portes ouvertes)

### Quand :

- Robot doit être accessible 24h/24h

==> nécessité d'un OS temps réel/longue durée de fonctionnement

### Comment:

- Application C/C++ pour programmation Beaglebone (filaire et Wifi/bluetooth)
- Application Java pour Smartphone (communication Wifi/Bluetooth)

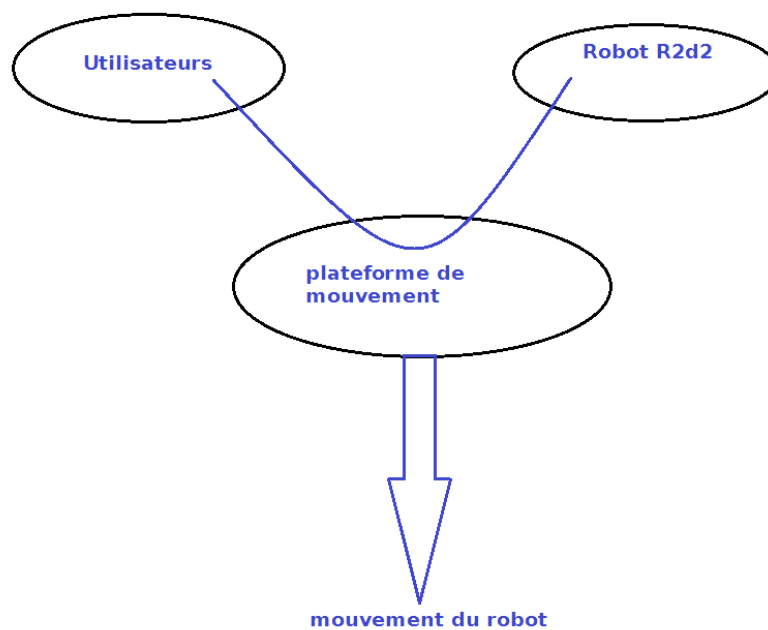
==> Voir connexion entre application Beaglebone et smartphone via ROS (Robot Operating System)

### Pourquoi :

- Permettre une plus grande maniabilité et un plus grand contrôle sur le robot

Suite à ce premier document, nous avons réalisé un schéma de « bête à corne » qui nous a permis de visualiser le dispositif général du système.

### Bête à corne :



**Figure 1 : Bête à Corne**

Après avoir étudié le système général, nous avons opté pour la réalisation du « diagramme de pieuvre » global du robot, qui nous sert à appréhender le contexte général du système.

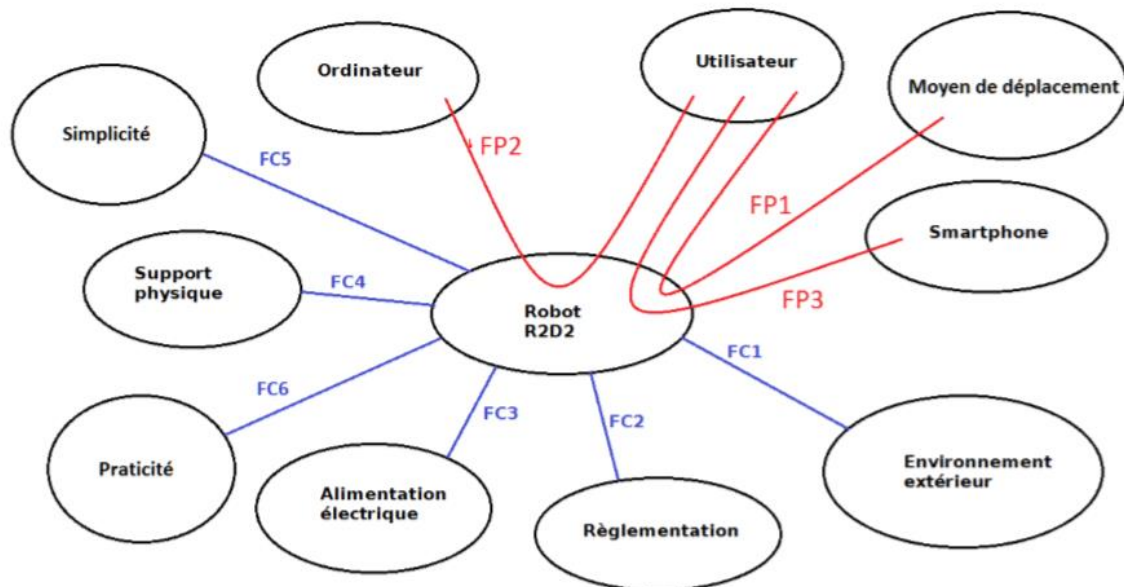




**PROJET:**  
**ROBOT R2D2**

RP

## Rapport de projet



FP1 : L'utilisateur peut déplacer le robot

FP2 : L'utilisateur peut contrôler le robot via ordinateur

FP3 : L'utilisateur peut contrôler le robot via smartphone

FC1 : Le robot (et son moyen de déplacement inclus) doit résister à l'environnement extérieur

FC2 : Le robot final doit suivre les réglementations et normes françaises

FC3 : Le robot (et son moyen de déplacement inclus) doit être autonome électriquement

FC4 : Le robot (à l'aide de son moyen de déplacement inclus) doit pouvoir gravir de très légères pentes

FC5 : L'assemblage final robot + moyen de déplacement doit être simple de conception

FC6 : L'utilisation et la manipulation du robot doivent être pratiques au quotidien

FP : Fonction principale



**PROJET:**  
**ROBOT R2D2**

RP

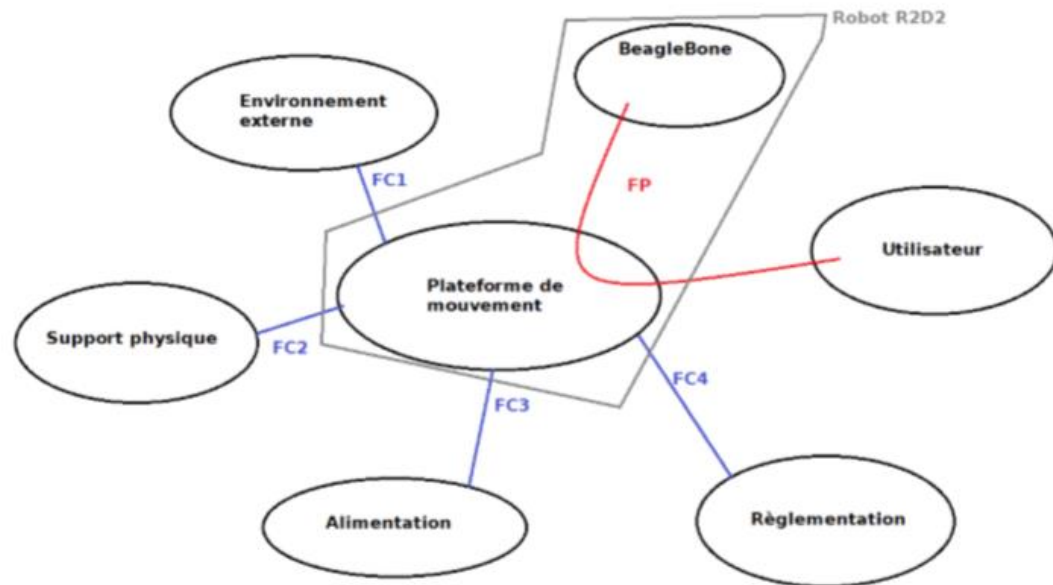
**ISEN** | école  
d'ingénieurs

## Rapport de projet

FC : Fonction de contrainte (contrainte appliquée sur le robot)

### Figure 2 : Diagramme de pieuvre global du robot

Suite au diagramme de pieuvre général, nous avons réalisé un diagramme de pieuvre centré sur la plateforme de mobilité.



FP1 : La plateforme de mouvement doit pouvoir être commandé via beaglebone par l'utilisateur (communication via ordinateur et/ou smartphone)

FC1 : La plateforme de mobilité doit pouvoir résister à l'environnement extérieur

FC2 : La plateforme de mobilité doit pouvoir tenir sur un support physique plat ou légèrement incliné

FC3 : La plateforme de mobilité doit être autonome électriquement

FC4 : La plateforme de mobilité et le robot en général doivent suivre les lois et réglementations françaises.

### **Figure 3 : Diagramme de pieuvre centré sur la plateforme de mobilité du robot**

La réalisation de ces deux diagrammes nous a permis de continuer à dégrossir le système général et permettre ainsi une approche optimale du projet.

L'étape suivante est l'étude des voies technologiques, cette étude permet de définir les fonctions présentes dans le diagramme de pieuvre global du robot.



But à atteindre	Fonction technique	Solutions
FP 1 : Déplacement du robot	<ul style="list-style-type: none"><li>Déplacement terrestre</li><li>Déplacement aérien</li><li>Déplacement maritime</li></ul>	<ul style="list-style-type: none"><li>Roues</li><li>Chenilles</li><li>Pattes</li><li>Jetpack</li><li>Fusées</li><li>Flotteurs</li></ul>
FP 2 : Déplacement via ordinateur	<ul style="list-style-type: none"><li>Application et contrôle via ordinateur externe et transmission des commandes via moyen de transmission</li><li>Robot directement pilotable via la carte</li></ul>	<ul style="list-style-type: none"><li>Filaire</li><li>Transmission Wifi</li><li>Transmission Bluetooth</li><li>Clavier branché</li><li>Joystick</li></ul>
FP 3 : Déplacement via smartphone	<ul style="list-style-type: none"><li>Application sur smartphone + moyen de transmission</li></ul>	<ul style="list-style-type: none"><li>Filaire</li><li>Transmission Wifi</li><li>Transmission Bluetooth</li></ul>

**Figure 4 : Étude des voies technologiques**

Après cette réflexion et énumération des choix possibles pour la réalisation des différentes fonctions énumérées, nous avons réalisé un tableau permettant de vérifier et de définir quelle solution est la plus adaptée pour chaque fonction.



**PROJET:**  
**ROBOT R2D2**

RP

**ISEN** | école  
d'ingénieurs

## Rapport de projet



	FP1	FP2	FP3	FC1	FC2	FC3	FC4	FC5	FC6
<i>Déplacement du robot</i>									
Roues									
Chenilles									
Pattes									
Jetpack									
Fusées									
Flotteurs									
<i>Contrôle via ordinateur</i>									
Filaire									
Wifi									
Bluetooth									
Clavier branché									
Joystick									
<i>Contrôle via smartphone</i>									
Filaire									
Wifi									
Bluetooth									

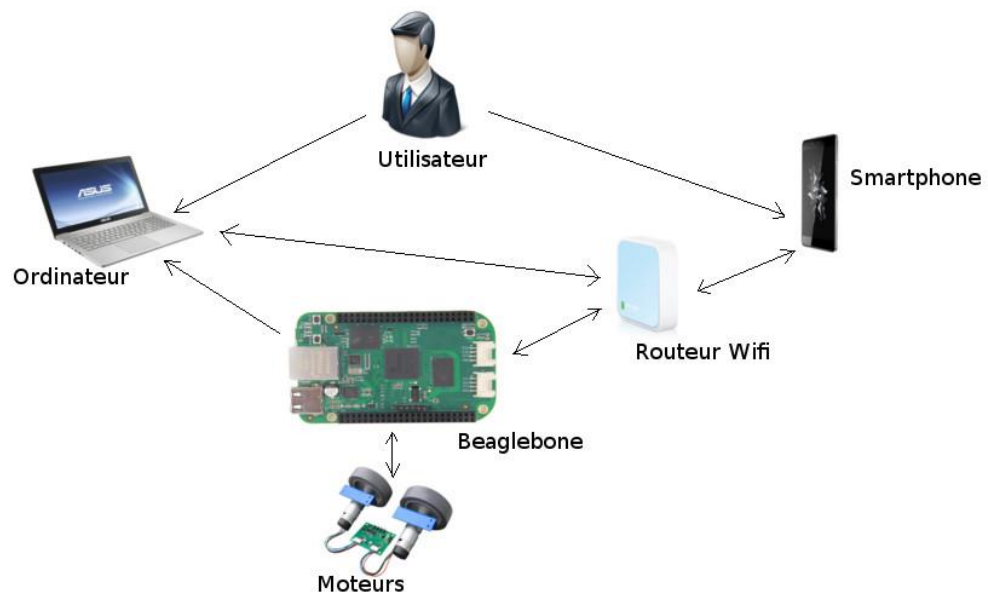
**Figure 5 : Tableau de choix des solutions techniques**



Compte tenu de l'existant et des contraintes de simplicité et de praticité :

- La solution retenue avec le client pour le déplacement du robot sera le déplacement terrestre via des roues. Les déplacements aériens et maritimes ne sont pas demandés et présentent des risques beaucoup plus importants.
- Les solutions retenues pour le contrôle de ce déplacement sont la transmission via Wifi, simple à mettre en place, et la transmission filaire pour le déplacement via ordinateur, si la liaison wifi venait à échouer ou être indisponible.

Après avoir étudié le système dans sa généralité nous pouvons en déduire le schéma suivant :



**Figure 6 : Vue générale du système**



Nous avons voulu déterminer un cycle « classique » de marche du robot. Le résultat obtenu est le suivant :

### Cycle de marche du robot :

- Initialisation du robot
- Mise en place des liaisons (démarrage routage par Wifi)
- Lancement du programme Son/Lumières/Écran LCD (indépendant de notre projet)
- Lancement du programme permettant la mise en mouvement
- Attente des instructions (écoute du port série du contrôleur)
- Si commande reçue, démarrage des moteurs et effectuation de la commande
- Si inactivité détectée, mise en veille des moteurs

Après cette étude, nous avons travaillé sur un diagramme de changement d'états. Ce diagramme nous a permis d'étudier les cas d'utilisation futur du robot R2D2.



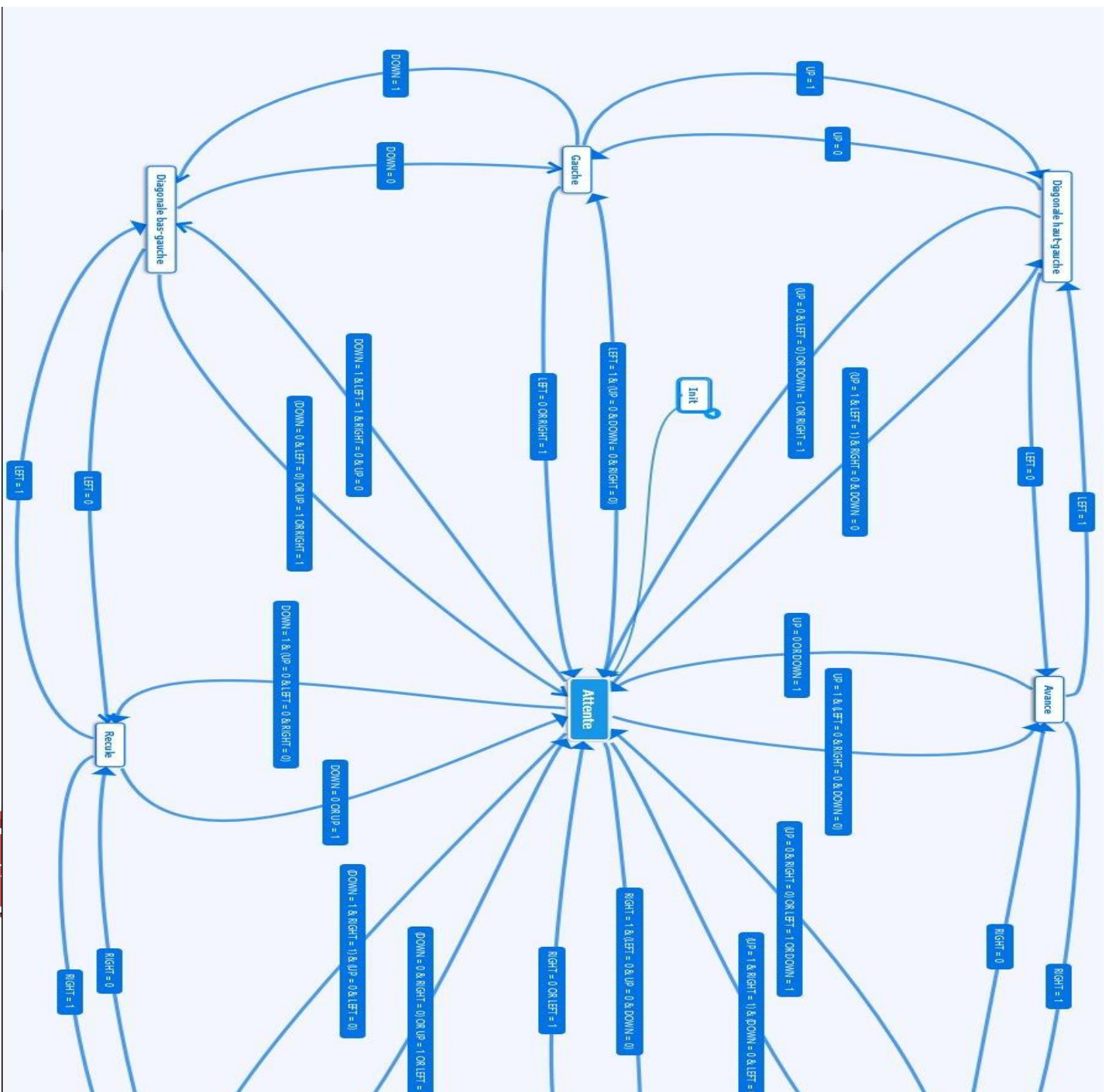


**PROJET:**  
**ROBOT R2D2**

RP

**ISEN** | école  
d'ingénieurs

## Rapport de projet





## Figure 7 : Diagramme de changements d'état simpliste

Suite à la sélection des différentes fonctions, il nous est apparu évident de réaliser une analyse fonctionnelle du besoin. Voici ci-dessous l'analyse fonctionnelle du besoin effectuée. Sont présentées ici les fonctions que doit remplir le système, les critères sur lesquels sont jugées ces fonctions, les niveaux attendus, ainsi que la flexibilité maximale autorisée pour chacun de ces critères.



Fonction	Critère d'appréciation	Niveau	Flexibilité
Déclencher les moteurs du robot suivant les commandes reçues	- Vitesse de déclenchement - Accélération	- Instantanée - 6 secondes d'accélération maximal	- +/- 1 seconde - +/- 2 secondes
Afficher le niveau de batterie du robot	- Vitesse de remontée des informations - Vérité des informations	- Instantanée - Informations vraies	- +/- 1 seconde - Erreurs mineures
Recevoir des commandes via ordinateur	- Vitesse de transmission	- Moins d'une seconde entre l'envoi et la réception de la commande	- Aucune
Recevoir des commandes via smartphone	- Vitesse de transmission	- Moins d'une seconde entre l'envoi et la réception de la commande	- Aucune
S'adapter au support physique	- Angle de la pente du support physique - Arrêt si détection d'obstacles	- 5 degré de pente maximal - 1 seconde maximale de temps de réaction	- +/- 5 degrés de pente - aucune
Arrêt complet en cas de perte du signal ou des commandes ou via bouton	- Temps d'arrêt du robot	- instantané	- +/- 1 seconde
Veille automatique des moteurs si niveau de batterie de la carte faible	- Temps de mise en veille - Message d'avertissement	- Moins d'une minute pour la mise en veille après sauvegarde des programmes - Affichage automatique et durée de 15 secondes pour le message d'avertissement	- 1 minute maxi - Durée de +/- 10 secondes

**Figure 8 : Analyse fonctionnelle du besoin**

Cette analyse est utile car celle-ci nous permet de commencer à évaluer les spécifications réelles du robot R2D2 et ensuite nous pourrons nous baser sur cette analyse pour effectuer des tests de validation.

En continuant dans cette optique de création d'outils pour la validation de tests nous avons créé un tableau des exigences. Celui-ci sera à remplir lors de chaque validation des différentes fonctions.

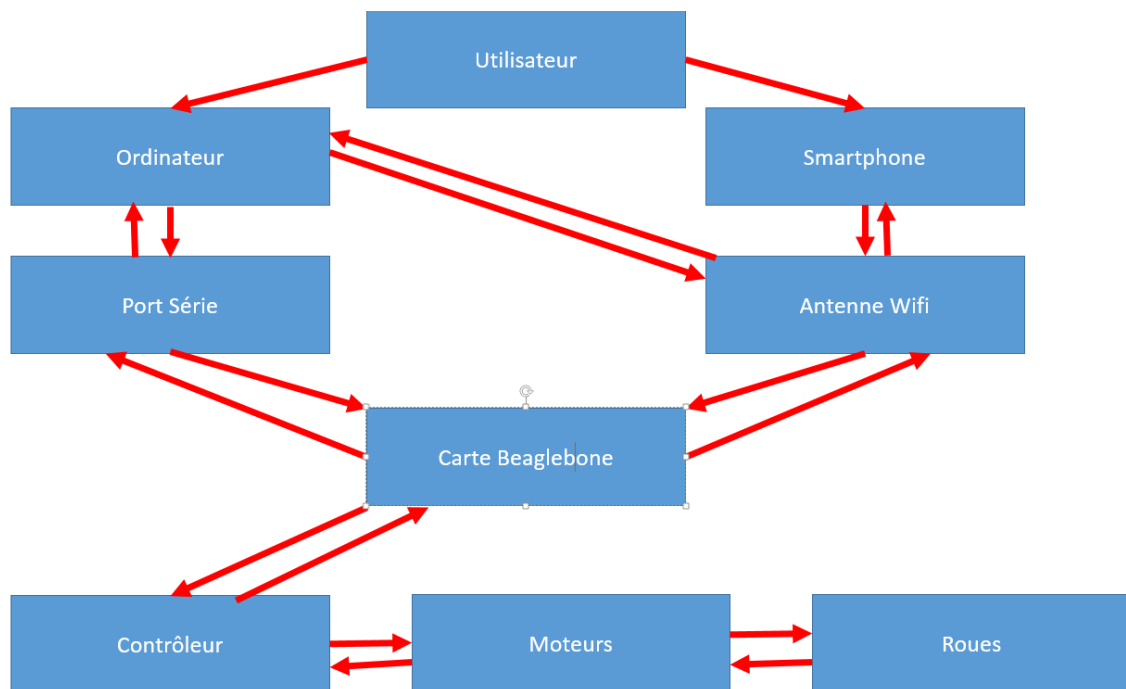


*Nota : - FAT: Factory Acceptance Test (tests effectués sur chaque exemplaire du matériel et dont les résultats conditionnent l'acceptation du produit par le client)*

- *I : Inspection (vérification visuelle)*
- *A : Analyse (vérification s'appuyant sur des preuves obtenues par calculs)*
- *D : Démonstration (vérification visuelle)*

### Figure 9: Tableau des exigences

Dans la continuité de la caractérisation des attentes du client, nous avons réalisé un synoptique fonctionnel. Celui-ci nous a permis de continuer à étudier le cycle « normal » d'utilisation du robot.



**Figure 10 : Synoptique fonctionnel**

L'étude suivante concerne le fonctionnement interne du robot R2D2. Nous avons réalisé un organigramme de fonctionnement qui nous a permis de développer un fonctionnement théorique du système.

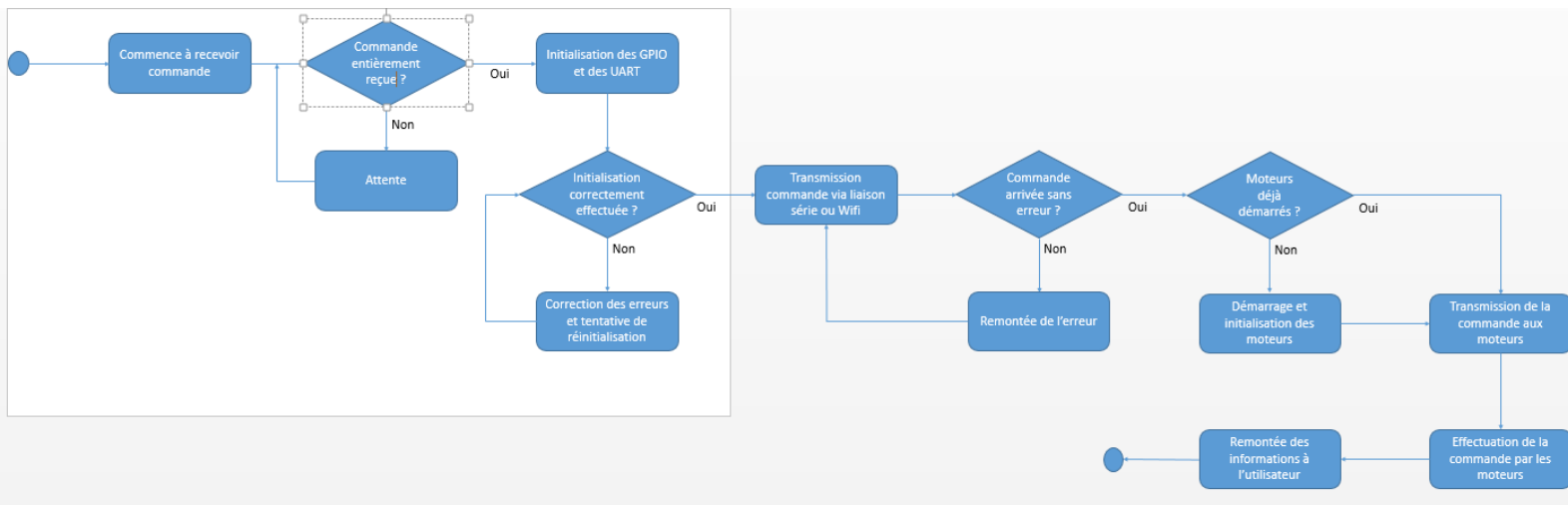


Figure 11 : Organigramme de fonctionnement

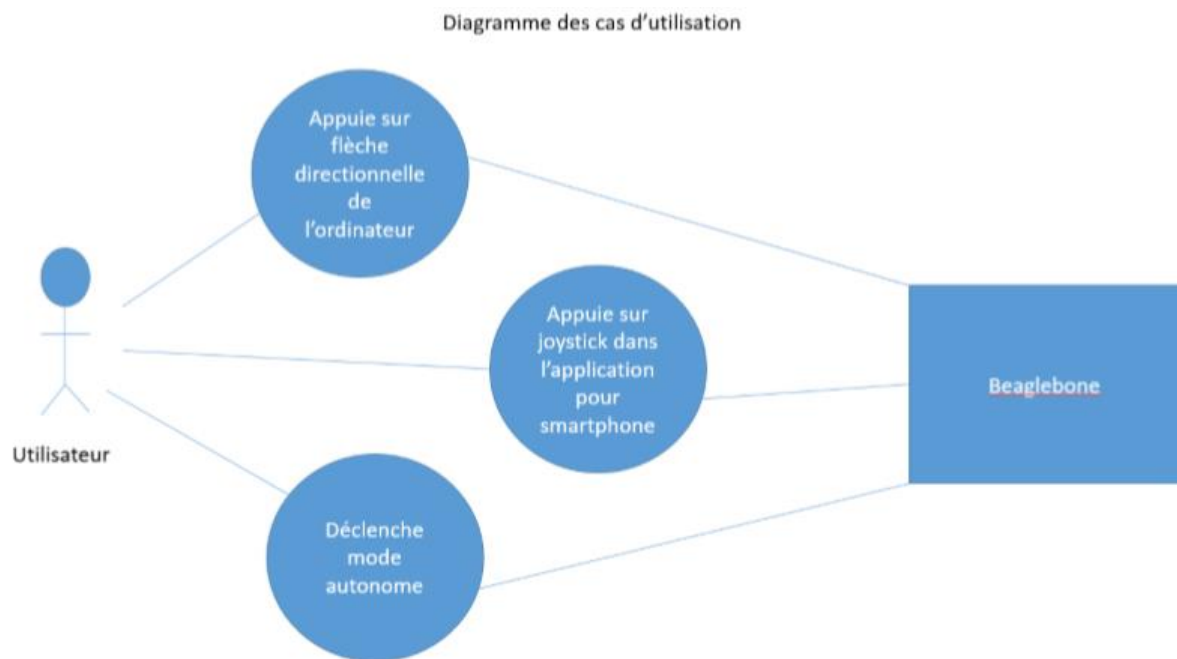
Après toutes les études générales faites, nous avons modélisé des diagrammes UML permettant une compréhension plus spécifique du système « robot R2D2 », allant encore du diagramme dit « simpliste », au diagramme des « cas d'utilisation », au diagramme de « composants », au diagramme de « déploiement » et enfin le diagramme de « séquence ».

### Voici les différents diagrammes :

Diagramme de classe simpliste

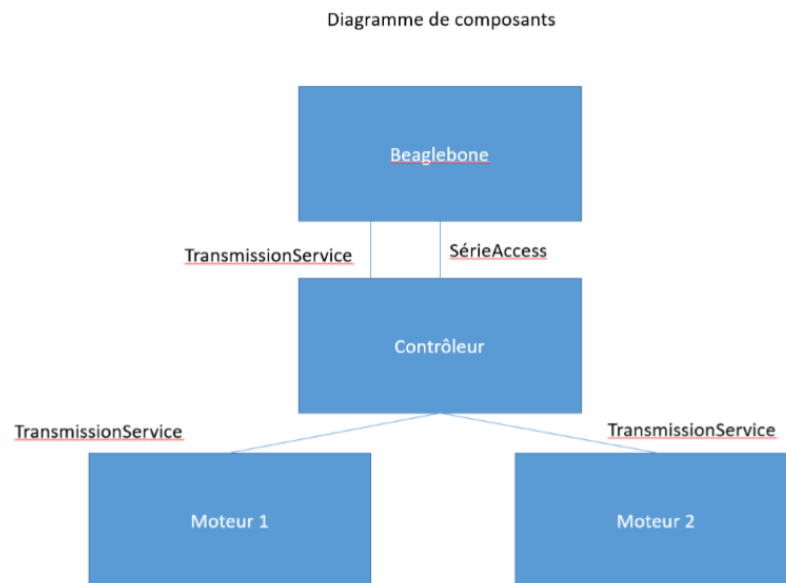


Figure 12 : Diagramme UML de classe



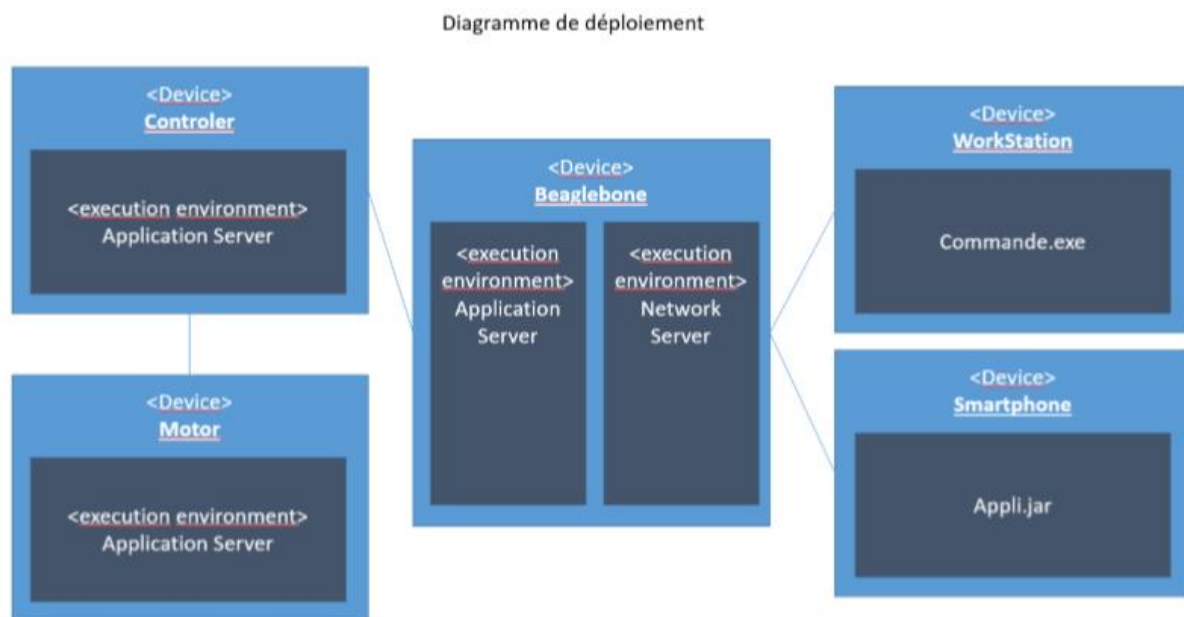
**Figure 13 : Diagramme UML des cas d'utilisation**



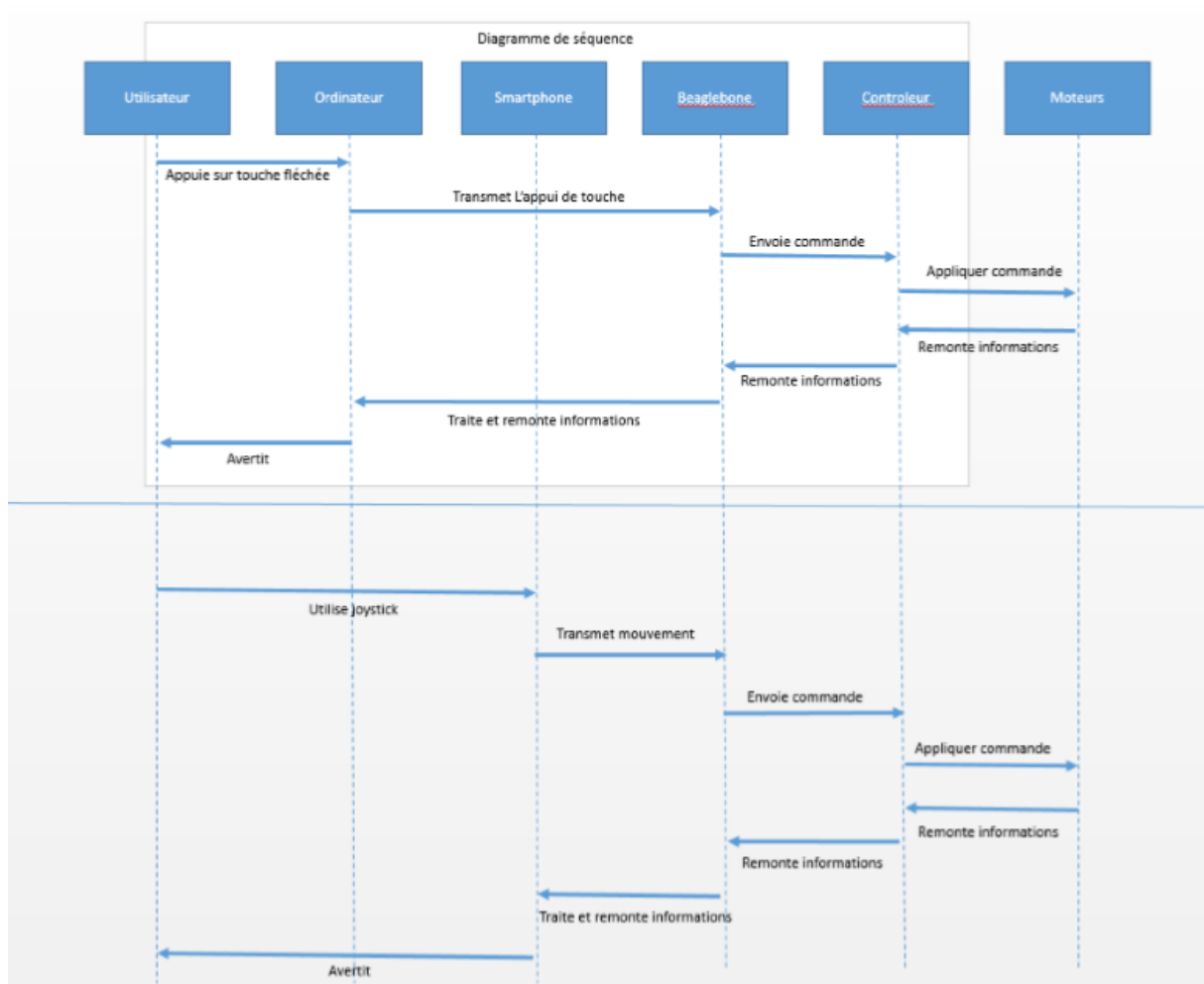


**Figure 14 : Diagramme UML des composants**





**Figure 15 : Diagramme UML de déploiement**



**Figure 16 : Diagramme UML de séquence**

L'intégralité de ces diagrammes sont ici dans l'optique de continuer à formaliser les outils générés dans la gestion de projet pour être compris de tous. C'est pour cela que pour les derniers outils nous avons opté pour l'UML.



**PROJET:  
ROBOT R2D2**

RP

Application smartphone :

Dans le cadre de l'étude du projet, nous avons pu réaliser une maquette de l'application qui permettra le contrôle du robot R2D2.

Maquette :

Vue 1:



**Figure 17 : Vue 1 application smartphone**



**PROJET:**  
**ROBOT R2D2**

RP

**ISEN** | école  
d'ingénieurs

## Rapport de projet

Vue 2

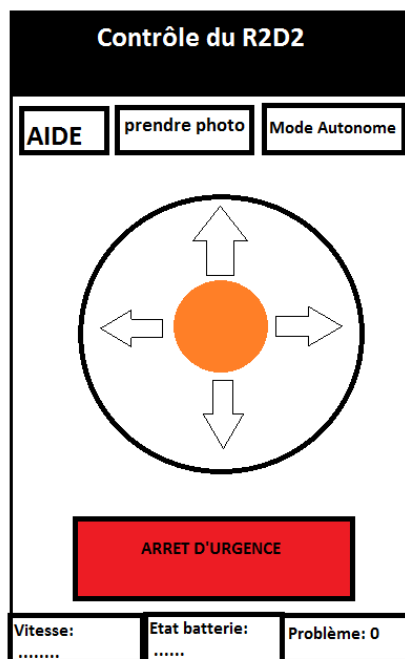
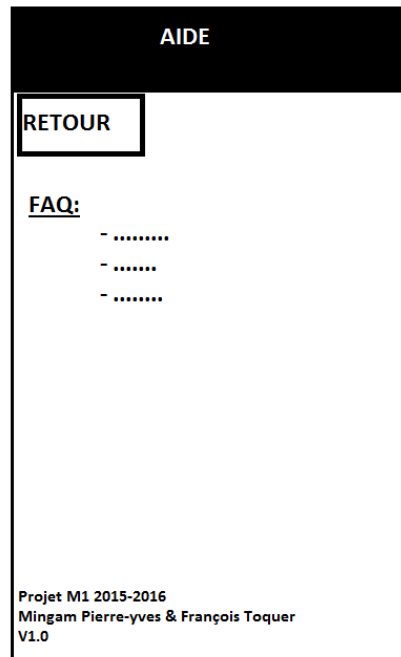


Figure 18 : Vue 2 application smartphone



Vue: 3



**Figure 19 : Vue 3 application smartphone**

Nous avons mis en place un dernier outil qui va nous permettre de valider le projet. Cet outil est le cahier des tests logiciels. Il sera sûrement modifié quand nous arriverons au terme du projet. Mais voici une ébauche de celui-ci :



Titre		Test du déclenchement du moteur		
Objectif		Vérifier que les cas d'erreurs possibles ont bien été prévus et que le moteur est bien déclenché à la bonne intensité à la réception de la commande		
Exigence du test		Tous les tests doivent passer pour certifier que ce module fonctionne correctement		
Exécutable		Commande.c		
Environnement		Debian,gcc		
ID	Démarche	Données entrées	Comportement attendu	Validation
1	Lancement de l'exécutable	/	Déclenchement des deux moteurs vers l'avant à mi-intensité	
2	Passer en paramètre à la fonction deux nombres entiers positifs différents supérieurs à 128	150,180	Déclenchement des deux moteurs vers l'avant à intensité différentes (le robot avance en diagonale)	
3	Passer en paramètre à la fonction deux nombres entiers positifs différents inférieurs à 128	40,80	Déclenchement des deux moteurs vers l'arrière à intensité différentes (le robot recule en diagonale)	
4	Passer en paramètre à la fonction deux fois le nombre 128	128,128	Le robot s'arrête (ou ne bouge pas s'il était déjà à l'arrêt)	
5	Passer en paramètre à la fonction deux nombres entiers positifs différents, l'un supérieur à 128, l'autre inférieur à 128	60,180	Le robot tourne sur lui-même	
6	Passer en paramètre à la fonction deux caractères quelconques	A, b	Aucune exécution et redemande des entrées valides	

**Figure 20 : Cahier de tests du bloc de déclenchement du moteur**



Titre		Test de réception des commandes		
Objectif		Vérifier que les commandes reçues correspondent bien à des commandes connues		
Exigence du test		Tous les tests doivent passer pour certifier que ce module fonctionne correctement		
Exécutable		reception.c		
Environnement		Debian,gcc		
ID	Démarche	Données entrées	Comportement attendu	Validation
1	Envoyer la commande pour récupérer le mode de fonctionnement	0x2B en liaison série	Affichage sur la console du mode de fonctionnement du contrôleur	
2	Envoyer la commande pour récupérer la vitesse du moteur 1	0x21 en liaison série	Affichage sur la console de la vitesse du moteur 1	
3	Envoyer la commande pour forcer la vitesse du moteur 1	0x31 en liaison série, suivi de 0xFF	Démarrage du moteur 1 à pleine puissance	
4	Envoyer la commande pour récupérer la vitesse du moteur 1	0x22 en liaison série	Affichage sur la console de la vitesse du moteur 2	
5	Envoyer la commande pour forcer la vitesse du moteur 1	0x312en liaison série, suivi de 0xFF	Démarrage du moteur 2 à pleine puissance	

**Figure 21 : Cahier de tests du bloc de réception des commandes moteurs**



Titre		Test d'envoi des commandes		
Objectif		Vérifier que les commandes prévues sont bien transmises		
Exigence du test		Tous les tests doivent passer pour certifier que ce module fonctionne correctement		
Exécutable		transmission.c		
Environnement		Debian,gcc		
ID	Démarche	Données entrées	Comportement attendu	Validation
1	Envoyer la commande pour récupérer le mode de fonctionnement	0x2B en liaison série	Récupération de 0x2b sur le scanalogic	
2	Envoyer la commande pour récupérer la vitesse du moteur 1	0x21 en liaison série	Récupération de 0x21 sur le scanalogic	
3	Envoyer la commande pour forcer la vitesse du moteur 1	0x31 en liaison série, suivi de 0xFF	Récupération de 0x31 et de 0xFF sur le scanalogic	
4	Envoyer la commande pour récupérer la vitesse du moteur 1	0x22 en liaison série	Récupération de 0x22 sur le scanalogic	
5	Envoyer la commande pour forcer la vitesse du moteur 1	0x32 en liaison série, suivi de 0xFF	Récupération de 0x32 et de 0xFF sur le scanalogic	

**Figure 22 : Cahier de tests du bloc d'envoi des commandes moteurs**

## Choix matériels

Grâce à notre étude du projet, nous avons pu optimiser le choix des différents matériels à utiliser. Les moteurs ainsi que la carte BeagleBone étaient imposés par le client. Nos différentes réflexions se sont tournées vers l'alimentation du système ainsi que vers la communication de la carte BeagleBone avec le smartphone.





### Choix des batteries

Une grande partie de notre gestion de projet a été orientée vers ce choix technique. Le client demande des batteries faciles d'utilisation, aussi bien pour le rechargement que pour la mise en place dans le système, avec une autonomie minimum de 4 heures. Nous avons réalisé un dimensionnement de la batterie nécessaire pour le système que nous voulions établir.

#### Dimensionnement de la batterie :

##### Partie BeagleBone :

Nom composants	Consommation:
Carte BeagleBone	210mA/h
Routeur wifi	250 mA/h
ecran lcd	240 mA + carte : 3 mA
led externe	20-25 mA
buzzer	+/-25mA/h
module de camera	100 mA
capteur de distance ultrason	15mA
<b>Total:</b>	<b>+/- 868mA/h max</b>

**Figure 23 : Tableau de dimensionnement de la batterie de la carte Beaglebone**

##### Partie moteurs :

Nom composants	Consommation:
Moteurs	2,5 A/h en bloqué / 500 mA/h en charge
<b>Total:</b>	<b>5A/h (2 moteurs) / 1 A/h (2 moteurs)</b>

**Figure 24 : Tableau de dimensionnement de la batterie des moteurs**



## Rapport de projet

Après cette étude, nous avons regardé quel type de batterie nous conviendrait le mieux. Cette partie a été sujette à débat, car il nous fallait une batterie permettant de fournir du 5V avec plus ou moins 1A/h pour la partie BeagleBone. Pour la partie motrice, il nous fallait une batterie permettant de fournir du 12V avec du 5A/h en blocage.

Notre choix s'est finalement porté sur l'achat de deux batteries différentes, une pour la partie BeagleBone et une autre pour la partie motrice du système.

Pour pouvoir correspondre à la demande du client, nous avons rapidement opté pour une batterie dite « intelligente ». La batterie retenue est une batterie externe pour recharger divers appareils tel que des téléphones, ou tablettes. Cette batterie nous semble convenir car elle remplit les spécifications requises.

Le problème qui s'est imposé à nous fût le choix pour la batterie de la partie motrice du système. Cette partie demandait 12V et non 5V, nous devons donc sélectionner une batterie différente que celle sélectionnée précédemment. La deuxième partie du problème était les ampérages à fournir à la carte BeagleBone. En étudiant les spécifications, nous avons pu nous apercevoir qu'il était possible que chaque moteur puisse consommer jusqu'à 2.5A/h en situation de blocage. Notre première idée fut donc de trouver une batterie permettant de fournir assez de courant pour les deux moteurs en situation de blocage ainsi que du 12V. Nous avons vite remarqué que trouver une batterie dite « intelligente » comme la première batterie serait difficile, car ces batteries ne disposent souvent pas du courant et du voltage demandé. Nous avons ensuite pensé à une batterie au plomb, qui était le deuxième choix au niveau des technologies de batteries. Ce choix de technologie nous permettait de dimensionner une batterie comme nous le voulions. Mais cette technologie ne nous permettait pas de combler toutes les demandes du client.

Nous voulions donc retourner sur la technologie lithium ion utilisé par les batteries « intelligentes ». Quelques précisions concernant ces batteries dites « intelligentes », elles disposent de mini-contrôleurs intégrés chargés d'évaluer l'état de la batterie et une protection en cas de surcharge ou de court-circuit. Ces batteries sont équipées de fusibles internes qui permettent de couper l'alimentation si elles détectent une anomalie dans l'alimentation. Nous avons également vu que dans les spécifications des capteurs des moteurs, il y avait le



nécessaire pour contrôler le courant et le voltage en temps réel. Nous avons donc opté pour une solution software qui nous permettrait de sécuriser la batterie et les moteurs d'une éventuelle surtension. Grâce à cette solution, nous pouvions donc chercher une batterie de 12V avec moins de 5A/h. Nous sommes donc revenus sur une batterie dite « intelligente », cette fois ci pour la recharge d'ordinateur portable.

### Choix des matériels divers

Pour le bon déroulement du projet, il est nécessaire de réaliser une communication par wifi entre la carte BeagleBone et le smartphone. Pour se faire, nous pensions opter pour une simple antenne wifi et une utilisation du réseau wifi de l'ISEN. Mais le projet M1 précédent ayant réalisé une carcasse en cuivre, cela limiterait la transmission de données à cause de l'isolation induite par le cuivre. On pensait simplement utiliser une antenne wifi branché directement sur la BeagleBone et dépassant de la carcasse du robot. Le deuxième problème qui se posait était de savoir s'il y aurait des réseaux wifi disponibles dans les endroits où le robot R2D2 se déplacera. Ainsi que de savoir si les ports nécessaires pour la communication entre le smartphone et la beaglebone seraient ouverts. Nous étions dépendants de l'infrastructure réseaux qui serait hôte de la démonstration. Pour une sécurisation et une gestion totale du réseau sur lequel nous voulons réaliser la communication entre le système et le smartphone, nous avons retenu le choix de créer notre propre réseau wifi avec l'utilisation d'un routeur wifi qui sera branché sur la batterie de la carte BeagleBone et sur le port Ethernet de la carte BeagleBone.

Concernant le mode autonome, nous avons opté pour un capteur à ultrasons ainsi qu'une caméra, les deux équipements seront connectés à la carte BeagleBone.

### Choix des technologies et des ports

#### Technologies

- Pour coder l'application actionnant les moteurs la technologie utilisée sera du C/C++, ou Python.



**PROJET:**  
**ROBOT R2D2**

RP

## Rapport de projet

- L'application smartphone sera elle développée en JAVA.
- La liaison entre le système android et la carte BeagleBone sera faite avec la technologie ROS (Robot Operating System)
- Il y aura développement d'une application JAVA pour interagir avec la carte BeagleBone

### Ports utilisés

- Port série : Pour connexion contrôleur moteurs
- Port Ethernet : Pour le routeur wifi
- Port USB : Pour ordinateur
- Micro USB (beagleBone Green) ou port alimentation (BeagleBone Black) : Pour la batterie externe
- Port I2C : Pour la camera
- GPIO : Capteur ultrason

### Chiffrage du projet

- Distance de contrôle par téléphone : 15 mètres garantis avec plus ou moins 3 mètres
- Autonomie : 4 heures minimum
- Surface où le robot peut se déplacer : plane avec un dénivelé maximal de 10 degrés
- Sécurité d'accès : Intégration ENT ISEN, avec disponibilité de l'application pour enseignants et ambassadeurs
- Distance de sécurité avec l'ultrason (obstacle avant, trou dans le sol) : Distance de sécurité minimale 50 cm
- Poids total du système : environ 7,5Kilos
- Coût total (batterie, camera, capteur ultrason, plateforme en bois, moyen de fixation moteurs) :

Batterie externe pour alim moteurs	69.90 euros
Batterie externe pour alim carte	29.99 euros (en promotion)
Capteur ultrason	16.51 euros
Camera	31.92 euros



## PROJET: ROBOT R2D2

RP

**ISEN** | école  
d'ingénieurs

## Rapport de projet

Plateforme en bois	Gratuit
Routeur wifi	19.98 euros
<b>Total TTC :</b>	<b>168.30 euros</b>

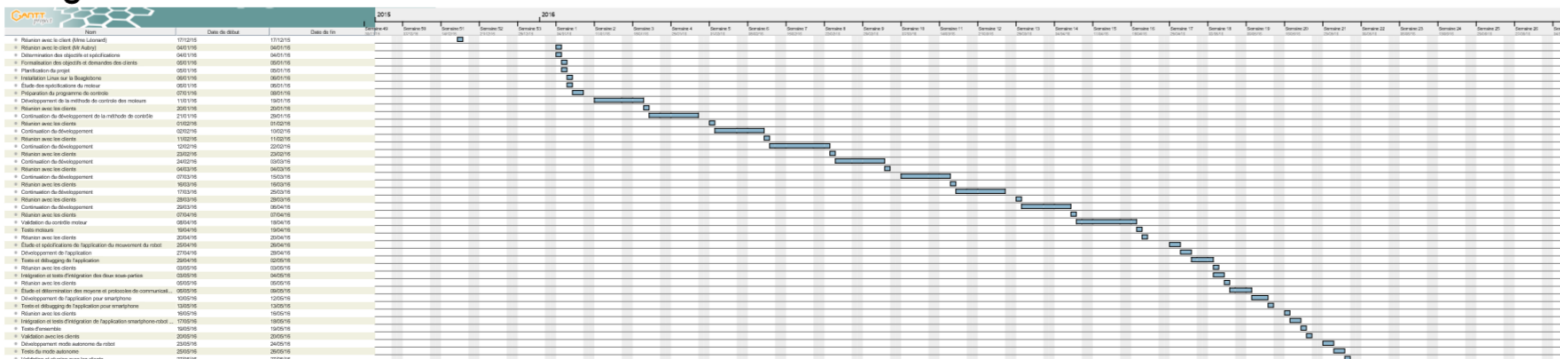
**Figure 25 : Chiffrage du projet**

## Planning prévisionnel

Le planning prévisionnel tient compte des deadline demandées et ne tient pas compte des objectifs optionnels. Il peut donc y avoir de grands changements dans les durées des tâches considérées.

## Diagramme de Gantt

5



**Figure 26: Planning prévisionnel du projet**

## Annexes