

cole supérieure d'ingénieurs Léonard-de-Vinci



ÉCOLE
D'INGÉNIEURS
PARIS-LA DÉFENSE



Cahier des charges

Equipe CFR-ESILV 2017

Paris La Défense, le 4 novembre 2016

Table des matières

1	Prsentation	3
2	Projet	5
2.1	La coupe de France de robotique	5
2.2	Contraintes respecter	5
2.3	Epreuves choisies	6
2.4	Organisation interne	6
3	Les outils utilis	7
3.1	Un systme d'exploitation pour les robots	7
3.2	Architecture de nos robots	8
3.3	La table de jeu	9
3.4	Un travail collaboratif	10
3.5	Une documentation de qualit	10
3.6	De la conception la ralisation	11
4	Calendrier	12
4.1	Date - Coupe de France de Robotique	12
4.2	Date - Ecole	12

1 | Présentation

Notre quipe pour la Coupe de France de Robotique (CFR) est constitue uniquement d'tudiants de l'ESILV (cole Suprieure d'Ingnieurs Leonard de Vinci). Pour bnficier d'une certaine exprience dans ce projet, trois tudians de l'anne 3 sont cette anne avec nous :

- POUSSEUR Hugo (chef de projet)
- BRINGER Hippolyte
- FRADET Guillaume
- HOUAOUI Rabah
- L'OUJDI Sofiane
- PIERRE Hugo
- SAM Medhi

Et bien sr la majorit de l'quipe est forme par des tudians de deuximes annes :

- JONAH HARISON Johan
- POUSSEUR Hugo
- PIERRE Hugo
- FRADET Guillaume
- ALBIZZATI Gregoire
- MARTIN Jeremy
- DIAZ PEREZ Anais
- BESSILA Rania
- RAMOS Miguel

Un mentor nous a t assign pour nous suivre tout au long de la ralisation de notre projet :

- DJERROUD Halim

l'ESILV a dj particip il y a deux ans et galement l'anne dernire. Une association de Robotique "DAVIN-CIBOT" a d'ailleurs t cre l'an pass par GREGORI Anthony. Nous voulons obtenir un statut d'association afin de bnficier d'aide financiere de la part de la COFA¹ et aussi auprs de sponsors extrieurs.

En effet, la ralisation de robots demande un certain budget pour l'achat des differents composants (cartes Arduino, cartes Raspberry PI, corps des robots ...).

1. COMmission du Financement des Associations

Cette année l'association a changé de bureau :

Président : JONAH HARISON Johan

Trésorier : POUSSEUR Hugo

GREGORI Anthony, ancien président de l'association, se charge de la communication web ; c'est à dire la gestion de notre site "davincibot.fr", de la page facebook "Davincibot" et du compte twitter "@DavincibotEsilv".

Au niveau de l'école, la réalisation de ce projet s'inscrit dans le PIX 2 (Projet d'Imagination et d'Exploration année 2).

C'est un projet dont le but est :

- d'appliquer ce qu'on apprend en cours
- de travailler en groupe
- de découvrir d'autres outils (en plus des cours)
- de construire un produit
- d'apprendre à gérer un long projet avec un budget

Le thème de cette année est "Ville Intelligente et Collaborative". *

- Comment la conception d'un robot peut répondre à cette problématique ?

Nous construisons ce qu'on appelle un automate, c'est à dire un robot capable de se déplacer et réaliser des actions. Nous avons l'un robot qui pourrait opérer dans une maison (comme une aide aux tâches ménagères par exemple) ou à plus grande échelle dans une ville.

2 | Projet

2.1 La coupe de France de robotique

La coupe de France de robotique a été créée en 1994 et se déroulait jusqu'en 2014 à la Fert-Bernard, dans le département de la Sarthe. Elle a lieu tous les ans à la fin mai et réunit environ 200 équipes de toute la France venues affronter leurs robots sur une table. Le règlement change tous les ans en fonction de la preuve proposée. Cette année, le thème est la plage et comporte plusieurs preuves qui rapportent chacune un nombre de points bien défini. Les preuves se déroulent donc sur une table avec deux concurrents, chacun ayant ces preuves indépendantes de l'autre.

Ces preuves sont :

- Hisser un drapeau en fermant une porte
- Pêcher des poissons
- Construire un château de sable
- Collecter des coquillages
- Ouvrir un parasol intégré au robot (Funny Action)

Bien sûr, toutes les preuves ne sont pas obligatoires et peuvent être effectuées dans l'ordre que l'on souhaite.

2.2 Contraintes à respecter

Il y a donc, comme dans toutes les compétitions, des contraintes à respecter. Tout d'abord, un robot ne peut pas empêcher l'adversaire de marquer des points, ou gêner de quelque façon la progression de son adversaire. Ensuite viennent les contraintes sur le dimensionnement des robots. Le robot principal ne doit pas excéder 1200 mm de largeur au départ (donc non déployé), et 1500 mm de largeur lorsqu'il est déployé. Pour le second robot, il ne doit pas dépasser 700 mm au départ et 900 mm déployé. Ils ne doivent pas dépasser 350 mm de hauteur. Le robot doit être autonome autant au niveau de sa progression que de son alimentation.

Il doit y avoir deux espaces rectangulaires de 100x70 mm laissés libres sur au moins deux faces du robot pour les autocollants de l'organisation.

Les robots doivent être dotés avec un dispositif composé d'un cordon d'au moins 500 mm de long qui ne restera pas sur le robot après le départ. Ils doivent avoir un bouton d'arrêt d'urgence facilement accessible sur le dessus du robot, et aussi d'un système permettant leurs arrêts automatiques au bout de 90 secondes. Ils seront composés aussi d'un système permettant d'éviter les autres robots et d'un support pouvant accueillir une balise de détection de l'équipe adverse.

Les matchs durent 90 secondes au-delà desquelles les robots ne peuvent plus se déplacer, mais possèdent encore 5 secondes pour faire la Funny Action.

2.3 Epreuves choisies

Nous avons donc choisi, en fonction de toutes ces contraintes de réaliser les preuves suivantes :

- Drapeaux (10 points par porte)
- Pêche (5 points hors de l'eau et 10 points dans le filet)
- Chateau de sable (2 points par bloc dans la zone, 2 points supplémentaires par bloc empilé et 12 points supplémentaires si le modèle du chateau est respecté)
- La Funny Action (20 points si le parasol est valide)

Nous avons choisi ces preuves car elles nous paraissent faisables et plutôt faciles à réaliser. Pour le chateau de sable nous avons juste décidé de pousser les blocs présents au niveau de la table dans la zone, car il y en a en hauteur. Pour les coquillages nous avons décidé que cela allait nous prendre beaucoup de temps à réaliser car ils sont de couleurs différentes et certains appartiennent à l'autre équipe. Nous nous sommes dit que nous allions prendre trop de risque pour le peu de points que nous pourrions gagner sur cette preuve.

Nous pensons pouvoir gagner environ 60 à 100 points avec la réalisation de toutes ces preuves.

Epreuve	Points
Drapeaux	10 par porte
Pêche en mer	5 hors de l'eau - 10 filet
Chateau de sable	2 bloc dans la zone - 2 bloc empilé - 12 chateau
Coquillages	2 sur la serviette
Funny action	20

2.4 Organisation interne

Pour mener bien ce projet, nous avons dû nous organiser de manière spécifique. Du fait de la taille de l'équipe. Nous utilisons une méthode de travail dite par XP (Extreme programming).

Cette méthode de travail inventée en 1999 se base notamment sur la communication et la simplicité.

Elle est très utilisée dans le milieu informatique pour favoriser la communication entre l'équipe projet, le client et les membres au sein de l'équipe.

Cette méthode est bien adaptée à notre situation, nous allons travailler par binômes sur des tâches définies et en volant sur les différents postes au sein de l'équipe. Ceci nous permettra de toucher toutes les facettes du projet, et d'être au courant de l'avancement de tous les objectifs.

Ainsi un membre de l'équipe ne fera pas toujours la même chose et chaque membre apportera sa part à l'ensemble du projet.

Pour mener bien ce projet, nous nous organisons comme suit :

- Une réunion hebdomadaire le lundi après midi
- Nous nous voyons aussi en dehors de la réunion du lundi pour pouvoir en faire plus
- Compte rendu pour chaque réunion

3 | Les outils utilis

3.1 Un systme d'exploitation pour les robots

Les preuves que nous devons affronter imposent l'excution de plusieurs tches en parallles, par exemple faire avancer le robot tout en ajustant sa position ou tout simplement excuter diffrentes tches afin de respecter les contraintes temporelles imposes par le rglement. Ce mode de fonctionnement impose une programmation parallles, ce qui n'est pas possible avec les microcontrleurs de type Arduino¹. Pour ces raisons nous avons dcid d'utiliser une autre technologie base aussi sur des microcontrleurs plus puissant, savoir la Raspberry Pi².

La programmation parallle pose son tour un problme. Il est ncessaire d'avoir une connaissance approfondie d'un langage de programmation spcialis tel que C ou C++. Afin de pallier ce problme, nous avons choisi d'utiliser un systme d'exploitation spcialis dans la programmation des robots. Ce dernier tant conu dans le but de faciliter la programmation des robots. Chaque concepteur de robot passait un temps non ngligeable concevoir matriellement son robot ainsi que le logiciel embarqu associ. Cela demandait des comptences en mcanique, lectronique et programmation embarque. Gnralement, les programmes ainsi conus correspondaient plus de la programmation embarque, proche de l'lectronique, qu' de la robotique proprement dite, telle que nous pouvons la rencontrer aujourd'hui dans la robotique. La rutilisation des programmes tait non triviale car fortement lie au matriel sous-jacent.

Comme son nom l'indique, ROS (Robot Operating System) est un systme d'exploitation pour robots. De mme que les systmes d'exploitation pour les ordinateurs. Il fournit des services proches d'un systme d'exploitation (abstraction du matriel, gestion de la concurrence, des processus ?) mais aussi des fonctionnalits de haut niveau (appels asynchrones, appels synchrones, base centralise de donnes, systme de paramtrage du robot ?). L'ide principale d'un OS robotique est d'viter de rinventer la roue chaque fois et de proposer des fonctionnalits standardises faisant abstraction du matriel.

Le principe de base d'un OS robotique est de faire fonctionner en parallle un grand nombre d'excutables qui doivent pouvoir changer de l'information de manire simple, l'aide de messages textes. Par exemple, interroger une frquence dfinie les capteurs du robot (capteur de distance ultrasons ou infrarouge, capteur de pression, capteur de temprature, gyroscope, acclromtre, camras, microphones ?), rcuprer ces informations, les traiter (faire ce que l'on appelle la fusion de donnes), les passer des algorithmes de traitement (traitement de la parole, vision artificielle, localisation et cartographie simultane ?) et enfin contrler les moteurs en retour. Tout ce processus s'effectue en continu et en parallle. D'autre part, l'OS robotique doit assurer la gestion de la concurrence afin d'assurer l'accs efficace aux ressources du robot.

1. Arduino est un circuit imprim en matriel libre sur lequel se trouve un microcontrleur

2. Le Raspberry Pi est un nano-ordinateur monocarte processeur ARM

3.2 Architecture de nos robots

Comme nous l'avons annonc  s le dbut de ce rapport nous avons dcid   de construire deux robots. Dans le but d'optimiser le processus de construction nous avons adopt   la m  me architecture pour les deux robots, nous avons utilis   pour chaque robot une carte RaspberryPi2 et une carte Arduino avec un shield pour contr  ler les moteurs qui font avancer le robot.



FIGURE 3.1 – RaspberryPi2 , carte Arduino avec un shield et moteur

Dans la figure suivante nous allons montrer l'architecture gn  rale qui d  crit l'agencement des ces diff  rents modules.

Une carte RaspberryPi2 sur laquelle nous avons embarqu   l'OS ROS et d  velopp   un petit module. Celui-ci permet de contr  ler les moteurs qui font avancer le robot. Cette carte n'est pas assez puissante pour alimenter directement les moteurs, pour cela nous avons intercal   entre cette dernire et les moteurs une carte arduino avec un shield pr  vu pour contr  ler les moteurs. Sur la carte arduino aussi, un petit code a t   embarqu   pour communiquer avec la carte RaspberryPi2.

Pour les capteurs et les effecteurs qui ne demandent pas une grande quantit   d'  nergie de fonctionnement nous les avons directement connects sur la carte RaspberryPi2.

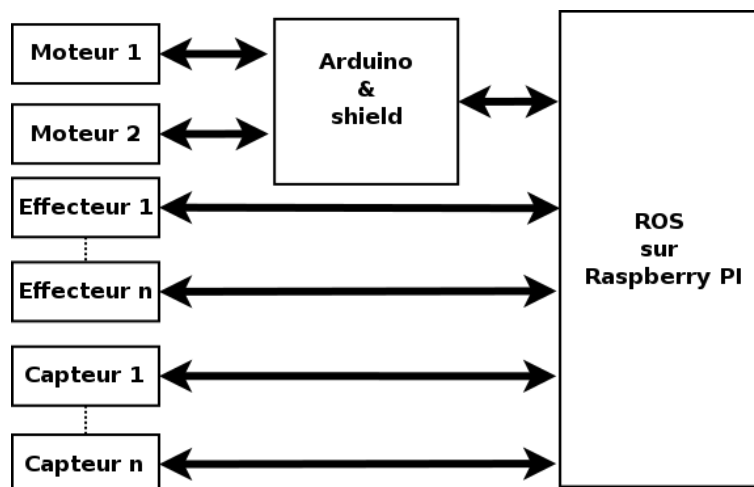


FIGURE 3.2 – Architecture gn  rale

Pour de valider notre architecture nous avons fait quelques essais avec un petit robot improvis, ce dernier a pour but d'interprter des commandes envoyes par l'utilisateur en utilisant les touches flches directionnelles du clavier.

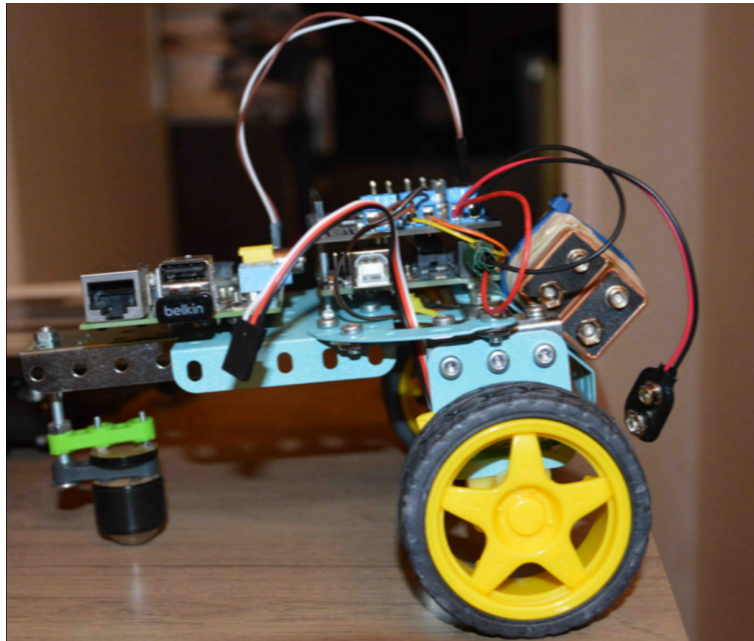


FIGURE 3.3 – tt

Le code embarqu sur ROS

3.3 La table de jeu

Les deux robots voluent donc sur une table de jeux possdant des obstacles et des endroits de diffrentes couleurs. La table est de 3000 mm par 2000 mm avec des bordures de chaque ct. Chaque quipe possde une zone de dpart matrialise par une zone peinte la couleur de l'quipe. Toutes les preuves sont prsentes sur la table : des drapeaux avec la porte qu'il faut fermer, ensuite les bacs d'eau avec les poissons pcher sur le ct de la table. Pour le chteau de sable, il y a une zone o doivent tre mis les blocs pour gagner des points. Ces blocs sont parpillés sur l'aire de jeu, avec une place bien dfinie pour chacun. Ensuite, il y a les coquillages (preuve que nous ne raliserons pas) qui sont rpartis alatoirement sur la table de jeu, et des places bien prcises sur le n rocher z.

Nous avons donc dcid de reproduire l'identique la table pour nous aider dans la ralisation des robots et nous permettre de faire des rglages prcis. Pour raliser notre table, nous possdons dj la table des annes prcdentes. Nous allons donc reproduire le dcor de la table de cette anne. Pour se faire nous avons recouvert de papiers adhsifs colors la table, pour ne pas avoir la peindre. Pour se procurer les diffrentes pices ncessaires, la Coupe de France a fourni sur son site internet les diffrents plans des pices et les fichiers pour les impressions 3D de ces dernires. La table sera accessible dans notre salle et nous aider dans la ralisation du robot.



FIGURE 3.4 – Table

3.4 Un travail collaboratif

Note qu'une équipe est composée de quatorze membres, le travail en équipe devient très difficile à partir du moment où on travaille sur les mêmes tâches. Pour résoudre ce problème, nous avons opté pour l'utilisation d'un système de versioning GIT. Ce dernier est un système qui enregistre l'évolution d'un fichier ou d'un ensemble de fichiers au cours du temps de manière ce qu'on puisse rappeler une version antérieure d'un fichier tout moment. De plus, il permet aussi de gérer les conflits qui peuvent survenir dans le cas où plusieurs utilisateurs travaillent sur la même portion de code.

Le logiciel git fonctionne en mode décentralisé, c'est-à-dire qu'aucun serveur n'est requis pour fonctionner, mais ce mode de fonctionnement est difficile à mettre en place. Nous avons opté pour une utilisation du mode centralisé de git, ce mode permet de centraliser le code sur un serveur et chaque utilisateur pourra consulter et effectuer des modifications.

Nous avons choisi d'utiliser la plate-forme GitHub. Elle présente l'avantage d'être gratuite et offre une version web qui permet de consulter directement les modifications en ligne. Nous avons créé un compte pour chaque membre de l'équipe et nous avons créé deux départements principaux, le premier permet de gérer nos documents de travail : le planning, le cahier des charges, les documents de conception etc, un deuxième département est réservé strictement au code. Ces derniers sont consultables sur les deux URL suivantes :

- <https://github.com/ESILV-CFR-2016/test>
- https://github.com/ESILV-CFR-2016/ros_raspi_pkg

3.5 Une documentation de qualité

Afin de produire une documentation de qualité nous avons décidé d'utiliser \LaTeX . Beaucoup de personnes se demandent pourquoi nous utilisons un traitement de texte comme \LaTeX si si barbant alors qu'il existe des traitements de texte WYSIWYG³ de plus en plus performants avec des correcteurs orthographiques et grammaticaux, avec la création de tableaux par simples clics de souris, avec des dizaines de polices de caractères différentes, avec des éditeurs de formules WYSIWYG gaisement. La réponse est simple \LaTeX est une vraie merveille si on doit rédiger un document avec de nombreuses pages car la mise en forme de celles-ci est très soignée. De même, la création et la modification de celles-ci sont relativement simples.

3. ³ *what you see is what you get*, signifiant littéralement en français *ce que vous voyez est ce que vous obtenez*

LaTeX permet l'utilisateur de se concentrer sur le contenu du document sans se soucier de la mise en forme qui sera effectuée automatiquement. Par exemple, on change de taille de caractères relativement une taille par défaut précise au début du document. Donc si on modifie la taille par défaut, les tailles relatives sont modifiées également et il n'est pas nécessaire de mettre à nouveau tout le document en page.

Nous avons choisi ce mode d'édition pour l'ensemble de nos documents ainsi, d'un côté cela nous permettra de nous entraîner à l'utilisation de LaTeX et d'un autre côté vu que LaTeX utilise du texte cela facilite la rédaction collaborative grâce à Git.

3.6 De la conception à la réalisation

Pour la partie conception mécanique nous avons choisi le logiciel Solidworks. Il permet de modéliser des objets en 3D. Grâce au FabLab de l'école, nous allons imprimer ces pièces en 3D. Il serait notamment souhaitable d'imprimer la base roulante du robot, et les petits accessoires du plateau (les coquillages par exemples...).

Solidworks est un logiciel de modélisation, le but de ce logiciel est de créer des plans de la pièce modélisée. Ces plans peuvent être utilisés par un centre d'usinage afin de réaliser le produit ou de support pour une présentation du projet. À l'heure où l'imprimante 3D devient de plus en plus courante, il est désormais possible d'imprimer directement la pièce à partir de sa modélisation. Cette technique est intéressante pour la réalisation de pièces complexes et son faible coût (aucune main d'œuvre). Nous avons la chance d'avoir accès au FabLab du pôle Léonard de Vinci composé de plusieurs imprimantes 3D.

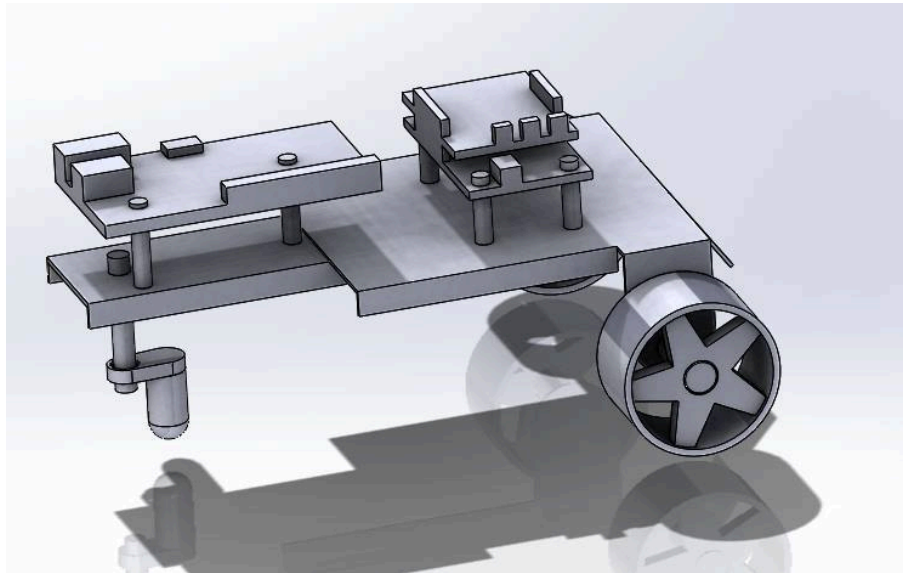


FIGURE 3.5 – Robot

4 | Calendrier

4.1 Date - Coupe de France de Robotique

La Coupe de France de Robotique nous impose quelques deadlines telles que :

- 31/12 : (Deadline) Inscription sur le site Poolzor. (<http://www.planete-sciences.org/robot/poolzor/>)
- En Mai 2016, sur une dure d'une semaine se droule la Coupe de France, la date exacte et le lieu n'ont pas encore t preiss.

4.2 Date - Ecole

La participation la Coupe de France de Robotique s'inscrit dans le cadre d'un projet donn par l'cole, de ce fait nous devons respecter plusieurs deadlines qui permettent l'avancement de notre projet :

- 21/11 : Rendu du pitch, film (45 60 secondes face camra pour "vendre" notre projet).
- 22/11 : Rendu du cahier des charges du projet (1-2 pages) + Dfinition et rpartition des tches (2-3 pages).
- 16/12 : Evaluation des membres du projet.
- 10/01 : Premire commande de matriel.
- 11/01 : Premire soutenance (psentation et avancement).
- 01/02 : Seconde commande de matriel.
- 04/04 : Premier prototype.
- 22/05 : Slides de psentation pour la soutenance finale.
- 24/05 : Soutenance finale et rendu d'un rapport du projet de 6 8 pages .
- 29/05 : Vido projet (1 2 minutes) avec un rsum de notre projet, ses avancements et un visuel de nos robots.
- 02/06 : Journe projets (psentation des projets de toutes les quipes).