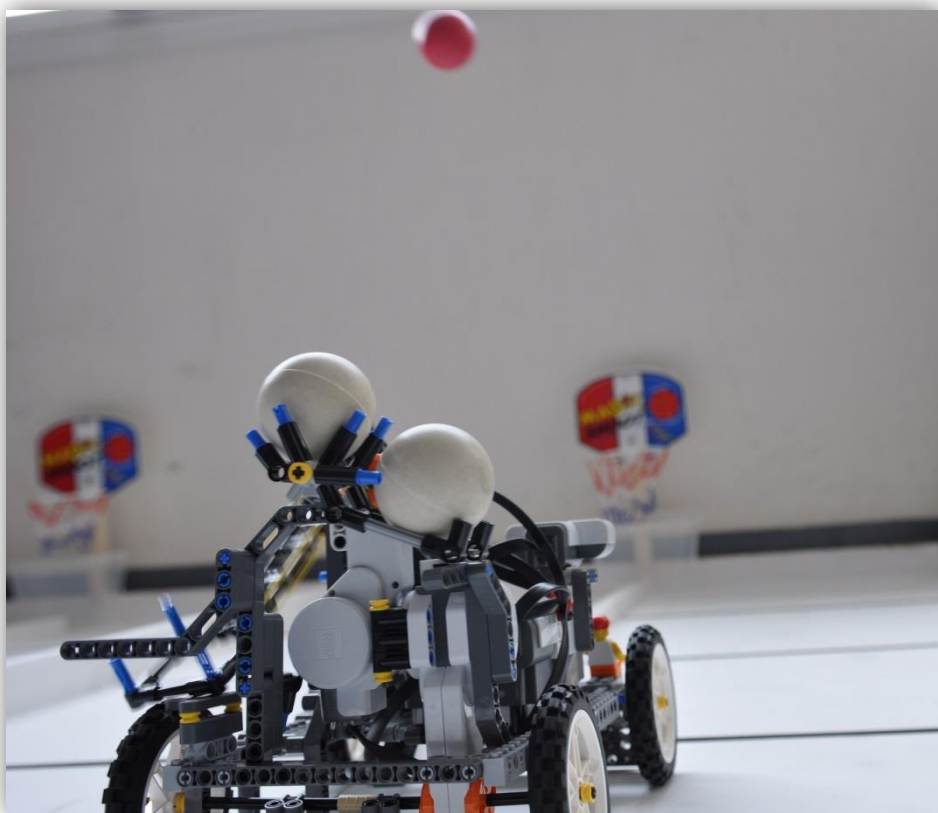


Groupe 9

# Challenge LEGO

*Rapport de notre projet*



*Notre robot en action...*

CAILLAUD Alain  
JAFFRE Vincent  
LESSEAU Quentin  
RUELLE Brandon

# Sommaire

---

Introduction .....	2
Notre organisation .....	3
Stratégie choisie .....	6
Réalisation du robot et problèmes rencontrés .....	6
La programmation .....	13
Retour sur la compétition .....	15
Apports techniques et humains .....	16
Conclusion .....	17
Annexes : le programme .....	18

# Introduction

---

Le challenge LEGO consiste à développer un robot afin de se confronter à une situation qui pourrait être rencontrée en entreprise. Ce dossier sera essentiellement orienté vers l'aspect gestion de ce challenge.

Cette semaine de projet avait trois objectifs principaux. Tout d'abord, elle avait pour but de faciliter l'intégration entre les nouveaux et les anciens étudiants de l'ISTIA. Ensuite, ce projet était un moyen de fournir une première approche à la mécatronique, un des enseignements principaux lors de cette année scolaire. Enfin, c'était aussi un moyen de nous confronter à une situation d'autogestion, de planification et d'organisation tel qu'on se retrouve en entreprise avec des contraintes également du monde de l'entreprise tel qu'un cahier des charges et des délais à respecter.

Premièrement, nous expliquerons l'organisation et le déroulement de la semaine. Nous verrons ensuite les problèmes que nous avons rencontrés et la manière dont nous les avons résolus. Enfin, nous présenterons les apports techniques et humains de ce projet.

# Notre organisation

---

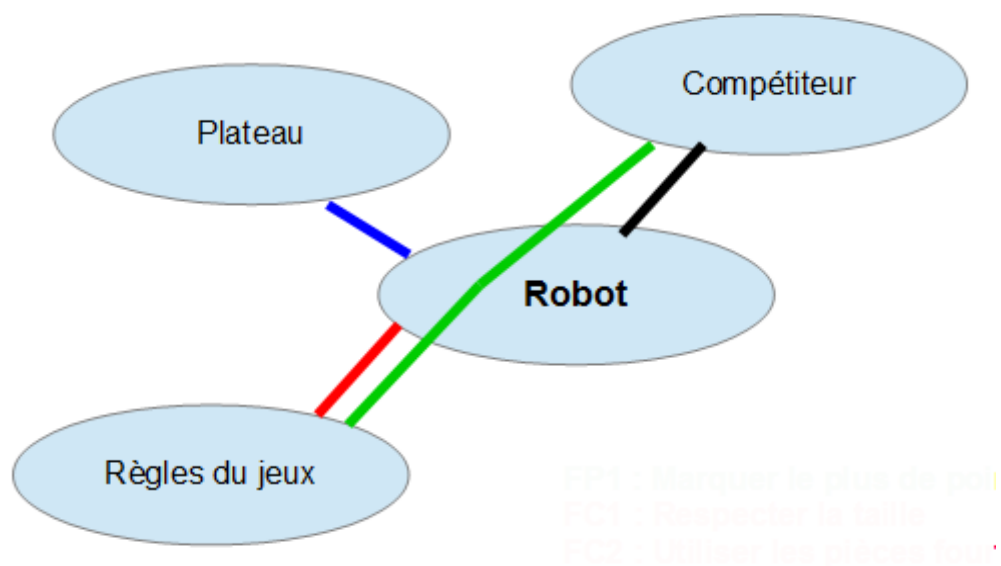
Lors de notre première séance de travail, nous ne formions pas réellement un groupe. En effet Vincent vient d'un IUT Génie Informatique et Industriel, Quentin d'un BTS Assistance Technique d'Ingénieur alors qu'Alain et Brandon viennent du cycle préparatoire. Aucun de nous ne savait à quoi s'attendre. Il est d'autant plus difficile de travailler avec des personnes qui nous sont inconnues.

C'est pour cette raison que nous avons consacré entre trente et quarante-cinq minutes au premier contact. Le but de cette manœuvre était de briser la glace pour éviter qu'une personne ou une autre se sente mise à part, ou tout simplement, n'ose pas émettre ses idées durant la totalité du projet pour diverses raisons.

Ce dialogue eut les conséquences souhaitées puisque les échanges d'idées se sont retrouvés facilités et les compétences de chacun ont pu être mises en avant. Nous avons observé que notre groupe était relativement polyvalent et hétérogène.

## *Lecture du cahier des charges*

Le but du projet est de réaliser un robot qui marque des paniers depuis plusieurs zones. L'analyse fonctionnelle externe suivante nous montre les principales fonctions à respecter afin de répondre aux exigences du cahier des charges.



## Répartition des rôles

Après avoir fait connaissance, nous savions que chaque membre du groupe était relativement polyvalent. Ainsi, chacun a pu assister les autres membres du groupe dans son travail.

Le seul contre-exemple fut la programmation. Alain et Brandon se sont en effet chargés de la totalité de celle-ci une fois la structure terminée.

## Brainstorming

A partir de cet instant, nous avons pu nous concentrer sur le lancement du projet. A l'unanimité nous avons pensé et décidé qu'un brainstorming serait la meilleure façon de commencer nos travaux. Nous avons trois systèmes principaux à mettre en œuvre pour notre robot : le système de déplacement, de chargement de balle et enfin de tir.

Nous avons donc mis sur feuille le maximum de mécanismes pour les différentes fonctions, peu importe la complexité de construction ou la pertinence de ceux-ci.

## Planning

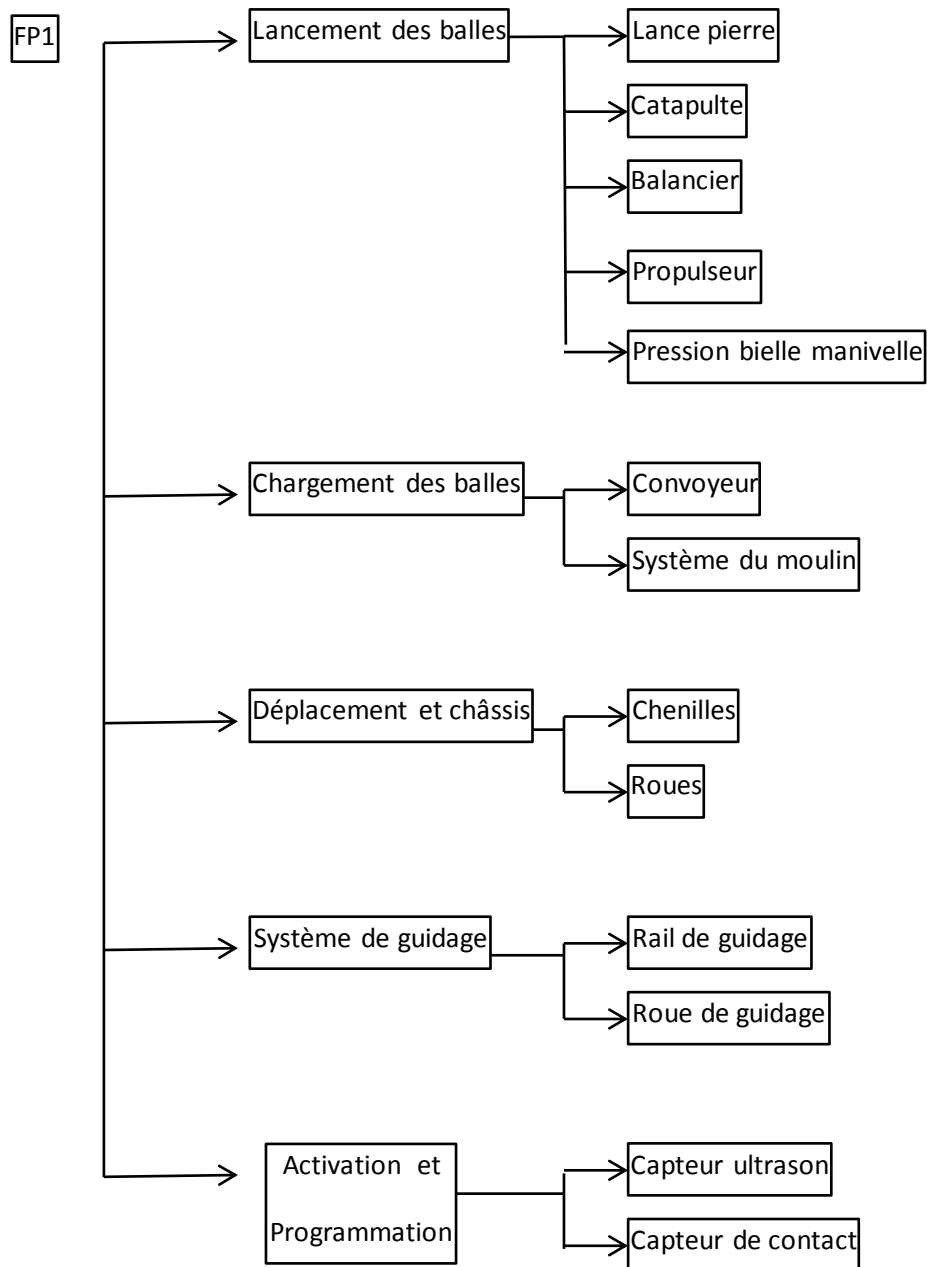
Le planning est une tâche très importante puisqu'il consiste à déterminer et à ordonnancer les tâches à effectuer. L'objectif est de déterminer si les tâches sont réalisées en temps et en heure ou non, et surtout de communiquer en fonction de l'avancement du projet.

Il nous a donc semblé indispensable de réaliser ce planning dès le lundi matin, après avoir pris connaissance de cahier des charges. Ainsi, nous avons déterminé, en fonction des capacités de chacun, les différentes tâches à réaliser, leur répartition au sein du groupe et le temps alloué à chacune d'elle.

Lundi	Prise de contact entre les différents membres du groupe Lecture du cahier des charges Brainstorming Mise au point d'une stratégie Réalisation du planning Découverte du matériel et du logiciel Construction des différents lanceurs
Mardi	Construction du châssis et du chargeur de balle
Mercredi	Construction de système de guidage et des capteurs Diaporama pour l'homologation
Jeudi	Dernière vérification structurelle Programmation du robot Homologation Tests avec le robot
Vendredi	Challenge

### Diagramme FAST «Function Analysis System Technique»

Le diagramme FAST, établi après notre brainstorming, nous a permis de répertorier hiérarchiquement toutes les technologies que nous pouvons utiliser pour chaque fonction du robot.



Ces différentes technologies constituent alors un ensemble de données essentielles permettant de trouver les meilleures options pour la construction du robot.

# Stratégie choisie

---

Lors de la lecture du cahier des charges, plusieurs options de stratégie étaient envisagées :

- Marquer dix points : marquer tous les paniers dont un dans le panier adverse et avoir le point bonus
- Marquer neuf points : marquer deux paniers depuis la zone à trois points puis un depuis la zone à deux points et avoir le point bonus
- Marquer six points : marquer deux paniers depuis la zone à deux points puis un depuis la zone à un point et avoir le point bonus

Pour notre part, nous avons choisi de ne pas tirer dans le panier adverse afin de conserver toute la rapidité de notre robot. Ainsi, nous pourrions obtenir le deuxième point bonus, celui de rapidité.

L'ensemble de la structure s'est donc fait selon cette stratégie. Pour obtenir un robot rapide, il nous fallait donc un robot léger. Le plus simple était de réduire au maximum le nombre de pièces utilisées en réalisant un ensemble le plus compacte possible.

Pour ce qui est de la programmation, nous avons décidé d'écrire deux programmes correspondant à deux tactiques différentes et adaptées à l'adversaire :

- 2 tirs à 3 points et un tir à 2 points
- 2 tirs à 2 points et un tir à 1 point

Le programme choisi dépendait des performances du robot adverse.

# Réalisation du robot et problèmes rencontrés

---

Nous avons commencé par installer un petit atelier de fabrication de prototypes, chacun tentant de rendre ces derniers optimaux.

Ces constructions peuvent sembler superflues voir inutiles, mais ce fut un moyen très efficace de prendre des pièces de formes variées et assez inhabituelles en mains. De plus, il n'y a rien de plus parlant qu'un système physique pour voir l'efficacité d'une idée. C'est notamment grâce à cela que nous avons pu remarquer si la puissance du moteur ou si les pièces intérieures (pièces du mécanisme) ou extérieures (balles) au système permettaient le bon fonctionnement de l'ensemble.

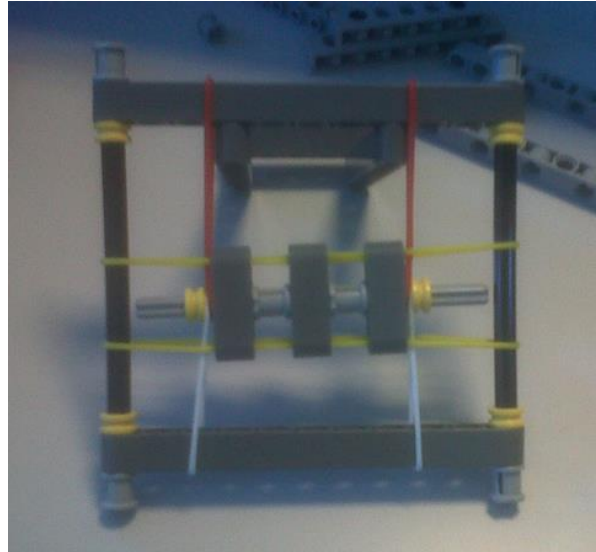
C'est ensuite avec l'ensemble de ces prototypes sous les yeux que nous avons fait nos choix. Les plus inefficaces ont ainsi été éliminés.

## *Système de lancement des balles*

Dès notre premier jour, nous avons réfléchi à comment lancer une balle à une distance d'environ un mètre tout en étant précis. C'est un des éléments les plus important du robot et aussi celui qui nous a posé le plus de travail.

### Le lance-pierre

Nous avons tout d'abord pensé à un système de lance-pierre comme ci-contre. Ce système, s'il fonctionnait lorsque nous l'utilisions à la main (sans moteur) était difficile à utiliser avec un moteur, il aurait fallu un mécanisme qui permette de faire tendre les élastiques puis de les libérer rapidement, ce qui n'était pas simple à mettre en œuvre. Nous avons donc fait le choix de ne pas garder ce système



### Le balancier

Nous avons donc ensuite imaginé un système de balancier. On s'était dit que si on faisait un balancier avec un poids important, que l'on faisait tourner ce balancier de manière à le relever de 90°, celui-ci, lorsqu'il serait libéré tournerait avec une force assez importante.

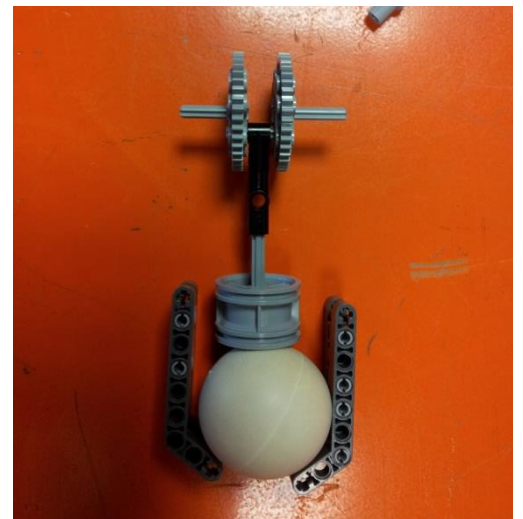
Ce système fonctionnait, cependant, il n'avait pas assez de puissance pour qu'on puisse l'utiliser, en effet, il envoyait la balle tout au plus à quelques dizaines de centimètres.



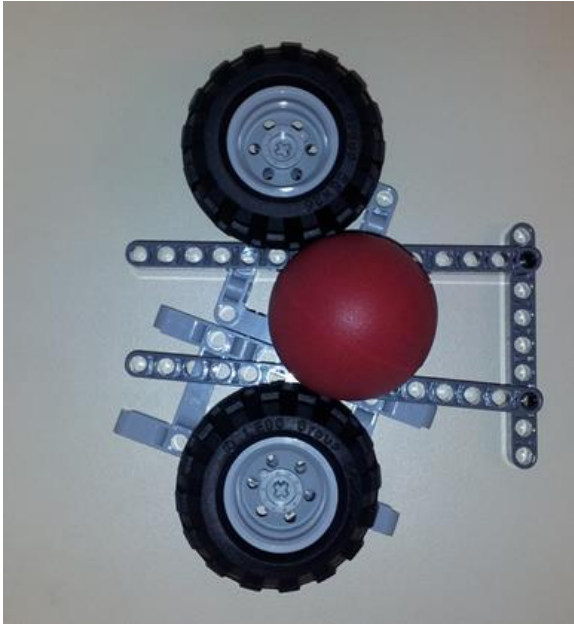
### Pression avec une bielle-manivelle

Nous avons ensuite imaginé réaliser un système de bielle-manivelle pour faire pression sur une balle coincée entre deux éléments. Une fois la pression suffisamment importante, la balle passe entre les deux éléments (à condition tout de même que ceux-ci ne soient pas trop serrés).

Nous avons abandonné ce système car il était difficile à mettre en œuvre.







### Le propulseur

Après nous être rendus comptes que les idées précédentes n'étaient pas fonctionnelles, nous avons donc du imaginer d'autres systèmes. Nous avons pensé par exemple à utiliser deux roues pour propulser une balle située sur une sorte de rail tel que visible ci-contre.

Ce système fonctionnait, mais ne permettait d'envoyer la balle qu'à quelques dizaines de centimètres ce qui n'était pas suffisant pour notre besoin.

### La catapulte

Enfin, après avoir dû abandonner toutes les idées précédentes, nous nous sommes donc penchés sur une catapulte. Après avoir effectué quelques tests, il s'est avéré que la catapulte était le seul système réellement efficace parmi ceux que nous avons, nous avons donc fait le choix de l'appliquer à notre robot. Ci-dessous, deux catapultes imaginées, nous avons abandonné la première à gauche qui était relativement lourde pour finalement utiliser celle de droite qui était bien plus légère.



## *Le chargeur de balles*

Afin de pouvoir embarquer les 3 balles et pouvoir les placer automatiquement dans la catapulte durant le match, il nous fallait un chargeur de balles. Pour, cela, nous avons imaginé seulement deux systèmes :

### Le convoyeur

L'idée à la base de ce système n'était pas un chargeur de balles. En effet, nous avons pensé pendant un temps utiliser un système de chenilles pour propulser une balle guidée par un rail, cependant, ce système était beaucoup trop lent pour lancer une balle à un mètre, par contre il s'est avéré qu'il pouvait libérer une ou plusieurs balles de manière assez précise, c'était une idée de chargeur intéressante.



### Le moulin

Nous avons aussi pensé à un autre système de chargeur, un mécanisme en forme de moulin dont l'axe est relié sur un moteur.

En tournant le moteur d'un angle déterminé, on libérait la première ou la seconde balle, c'est un autre système dont les tests nous satisfaisaient.

Contrairement au propulseur où nous avons plusieurs idées mais une seule de réellement fonctionnelle, pour le chargeur nous avons donc deux idées qui se révélaient intéressantes l'une comme l'autre. Nous devons donc faire un choix, nous nous sommes naturellement tournés vers le moulin pour la simple raison qu'il était moins volumineux et qu'il nous laisserait donc plus de libertés dans la construction du robot.

## *Déplacement et châssis*

Nous avons commencé à travailler sur la base de notre robot, c'est-à-dire sur son déplacement et sur le châssis le plus adapté. Nous avons tout d'abord travaillé chacun de notre côté pour ensuite confronter les différentes idées. Les deux châssis les plus intéressants étaient un châssis à chenille ainsi qu'un châssis à roues. Le châssis à chenille a du couple mais il n'est pas aussi rapide que le châssis à roues et l'espace dont nous disposions était insuffisant pour les différentes parties de notre robot (catapulte, moteurs, chargeurs ...).



Système de guidage



Ayant un rail sur le parcours pour guider le robot, nous avons tout d'abord pensé utiliser un système avec des roues qui auraient été placées à l'horizontal le long du rail. Lorsque nous les avons fixés au châssis, le robot ne respectait plus une contrainte celle qu'il ne devait pas dépasser une feuille A4, du coup cette idée a dû être abandonnée.

Nous sommes repartis sur une idée beaucoup plus simple avec un système de glissière en installant sous le châssis deux pièces qui permettraient de caler le robot sur le rail dès le départ.

## *Activation du robot*

Le robot devait démarrer après que la feuille en papier disposée devant lui ne soit enlevée. Nous avons d'abord pensé à un système avec capteur de contact mais le problème en le regardant fonctionner c'est qu'il fallait un objet en dur devant et la contrainte stipuler était une feuille en papier. Nous avons repensé l'idée avec le capteur à ultrason et nous l'avons réglé avec une distance minimale, du coup avec la feuille en papier devant il ne pouvait démarrer mais une fois enlevé aucun soucis.

Nous pensions le réutiliser pour arrêter le robot mais ayant une catapulte sur le côté nous ne pouvions nous servir du panier comme point d'arrêt. Nous nous sommes servi du capteur de contact qui lorsqu'il touchait le panier devait s'arrêter.

### *Quels sont les limites de détection du capteur ?*

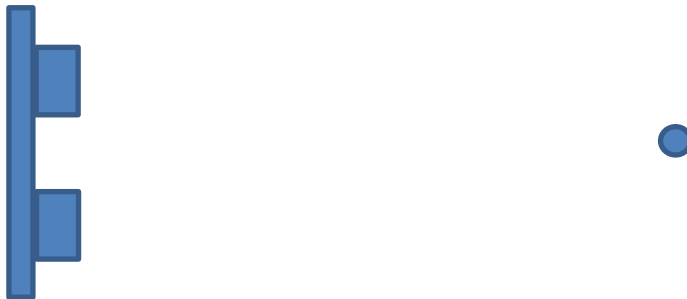
Pour fonctionner, le capteur envoie une onde sonore avec sa partie gauche, cette onde est normalement réfléchiée dès qu'elle rencontre un obstacle jusqu'à revenir vers le capteur, où elle est détectée par la partie droite. Le temps que met cette onde à revenir, est proportionnel à la distance, ce qui nous permet de connaître la distance entre le capteur et l'objet.

En travaillant sur le capteur, on a remarqué qu'il fonctionnait mieux dans certains cas que d'autres. Voici les cas où le capteur fonctionne mal :

- Objet trop près :

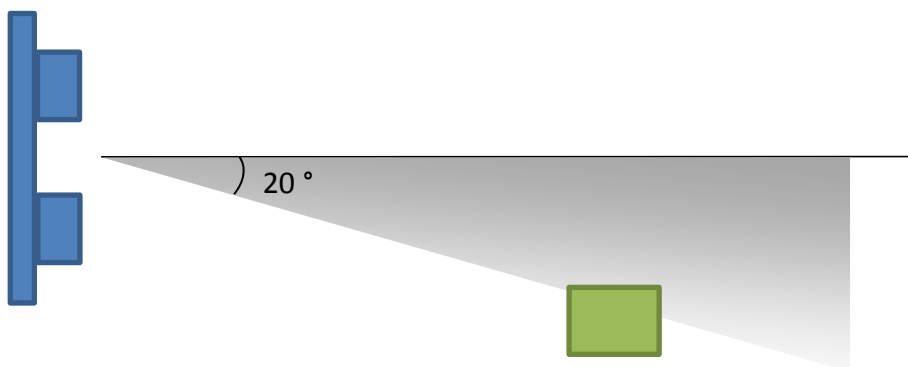
Si un objet est trop près du capteur, celui-ci ne voit pas l'objet, il est « aveuglé ». En effet, si un objet est collé au capteur, l'onde émise par la partie gauche n'arrivera pas à revenir à la partie droite de celui-ci, le capteur ne mesure rien.

- Objet trop petit :



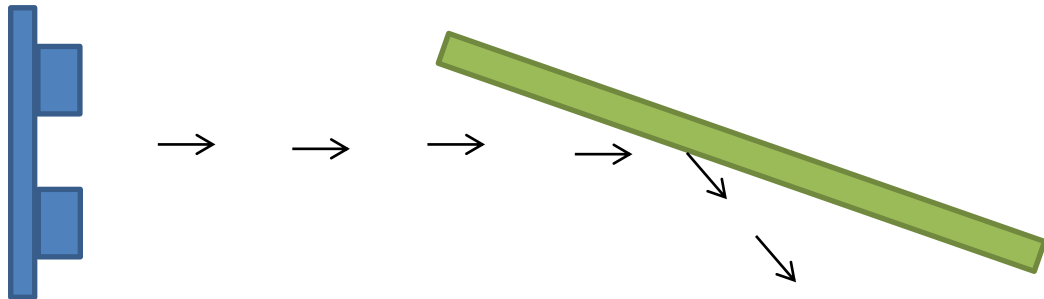
On remarque que lorsque l'objet est petit, celui-ci n'est pas forcément détecté par le capteur. Pour nous, c'était parfois le cas dans les salles de TPs où il y a beaucoup de câbles au sol pour les automates ou autre, ces câbles n'étaient pas forcément détectés. Ceci dit, en général dès que l'objet est un peu plus gros comme du diamètre d'un crayon par exemple, généralement le capteur fonctionne bien.

- Limite angle du capteur :



Lorsque l'objet est situé à la limite de l'angle de détection comme sur le schéma ci-dessus, le capteur a du mal à calculer la distance. Soit il ne voit pas l'objet, soit il le voit mais évalue mal sa distance, en le voyant beaucoup plus loin qu'il ne l'est réellement.

- Surface de l'obstacle inclinée :



Lorsque la surface de l'obstacle fait un angle important par rapport à la surface du capteur, celui-ci a beaucoup de mal à estimer la distance de l'obstacle. Lors de nos tests, nous avons remarqué qu'il indiquait souvent une distance différente de la distance réelle et en plus il n'indiquait pas toujours la même distance bien que nous ne touchions à rien.

Exemple, lors d'un test, on a mis un obstacle incliné à une distance d'environ 30 cm du capteur. Notre programme indiquait toute les secondes la distance mesurée sur l'ordinateur. Celui-ci indiquait une distance qui variait entre 80 cm et 250 cm bien que nous ne déplaçons ni le capteur ni l'obstacle. Pour résumer si on place un obstacle incliné devant le capteur, non seulement celui-ci détecte une valeur fausse mais en plus il ne donne pas la même valeur au cours du temps.

### *L'assemblage*

Une fois les solutions aux fonctions trouvées, le travail d'assemblage a donc pu commencer. Nous nous sommes alors attelés à la tâche en respectant deux consignes : le cahier des charges et nos objectifs personnels cités précédemment (la stratégie). Une fois l'ensemble réalisé, nous avons entamé la phase de programmation puis des tests.

# La programmation

---

Lors de nos recherches le premier jour de la semaine, nous nous sommes rendu comptes qu'il était possible de donner au moteur une indication en degrés ou en nombre de tours plutôt qu'un temps ce qui nous as tout de suite paru un avantage important. En effet, si on demande au moteur de tourner pendant un certain temps, nous ne pouvions pas être sûrs qu'il reproduira à chaque fois la même trajectoire, une batterie plus ou moins chargée influe sur la vitesse du moteur. Or, en lui indiquant un angle en degré ou un nombre de tours, nous sommes surs que celui-ci reproduira à chaque fois la même trajectoire indépendamment de la charge de la batterie.

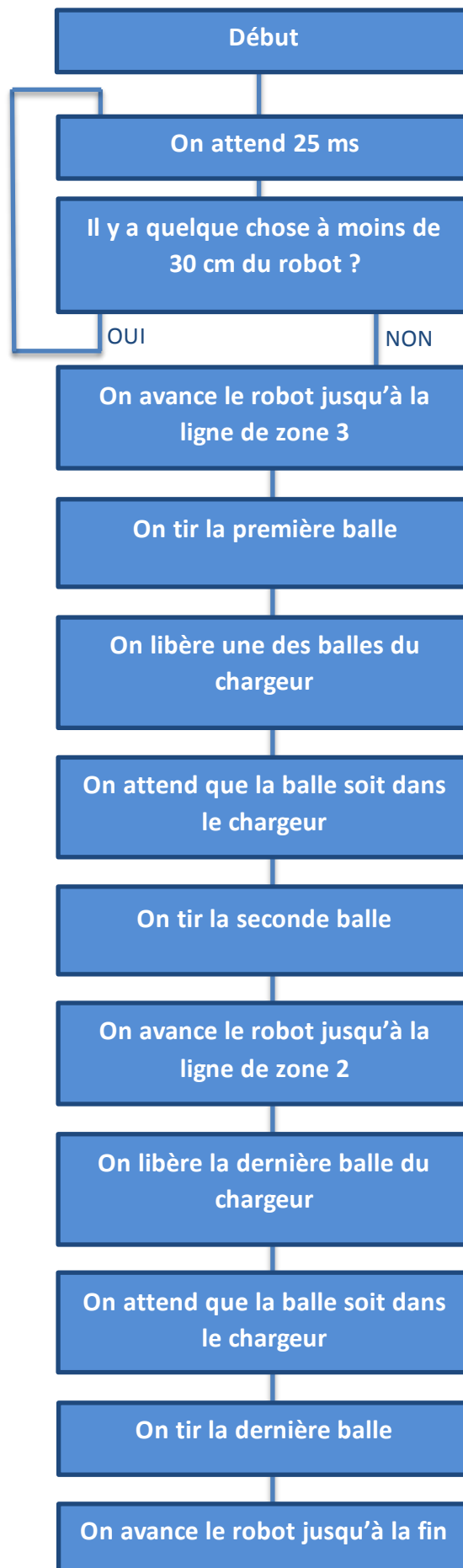
De même, grâce à ça, nous pouvions le faire avancer juste devant une ligne tout en étant sûr qu'il ne la dépasse pas, sans avoir à utiliser de capteur de luminosité ou autre, en effet, une fois que nous avons indiqué dans le programme l'angle correct d'avancement, si nous mettions toujours le robot à la même position initiale, celui-ci s'arrêtait toujours devant la ligne souhaitée.

## *Les tests*

Les tests ont consisté en premier lieu en des réglages à l'aveugle. Ceux-ci nous ont permis d'observer le comportement des moteurs en fonction des valeurs que nous leur attribuons : Puissance, nombre ou angle de rotations...

Ces tests nous ont permis d'observer des défauts dans la structure de notre robot, dont un en particulier : notre système de lancement, une catapulte, s'est révélé trop lourd pour la puissance de notre moteur. Elle a donc dû être changée pour répondre aux capacités du moteur.

C'est au bout de plusieurs tests et de problèmes résolus que nous avons obtenu nos programmes finaux.



# Retour sur la compétition

---

## *Observations*

Le jour du challenge, nous n'avons eu aucun problème à nous qualifier. Malheureusement, ce n'était pas grâce aux performances de notre appareil mais plutôt à la non performance de ceux des autres équipes.

Lors de nos tests, le robot marchait parfaitement malgré le peu de fiabilité de notre système de recharge de balle (qui était plus adapté à l'ancienne catapulte).

Nous tirions environ 10 cm plus loin avec les balles neuves du challenge qu'avec les balles de test, un problème qui aurait pu être résolu entre deux épreuves en modifiant rapidement le programme du robot. De plus, cet imprévu rendait inutile la création de deux programmes puisque les deux étaient rendus quasiment inutilisables.

## *Autres améliorations possibles*

Si nous nous étions rendu compte du problème de catapulte plus tôt, nous aurions pu perfectionner ou du moins adapter notre système de recharge et éviter quelques pertes de balles durant le challenge.

Le robot aurait également pu être plus rapide et donc satisfaire bien plus nos objectifs stratégiques à l'aide d'un système de transmission à engrenage entre le moteur et l'ensemble des roues.



# Apports techniques et humains

---

## *Apports Techniques*

- Planifier :  
Le déroulement du projet, de la conception en passant par la soutenance d'homologation et pour finir le rapport final du projet.
- Concevoir :  
Les différentes parties du robot ont été faites en équipe
- Communiquer :  
Les différentes idées qui pouvaient être mises en œuvre pour concevoir le robot tout en respectant les contraintes données. Donner son point de vue sur la manière de l'assembler ou simplement pour améliorer les pièces déjà assemblées.
- Programmer :  
Les différents capteurs et moteurs pour permettre à notre robot de marquer le plus de paniers sans échouer tout en allant assez vite, il faut passer par une programmation en C.

## *Apports Humains*

- Travail d'équipe : Apprendre à travailler avec des personnes que l'on ne connaît pas et en fonction des capacités de chacun.
- Pragmatisme :  
Conception et programmation

# Conclusion

---

A travers son apparence ludique, ce projet s'est révélé être un vaste support de réflexion, que ce soit en mécanique mais aussi en programmation. Il a été l'occasion pour nous d'apprendre à nous connaître, mais aussi d'apprendre à travailler avec des gens que nous ne connaissions pas sur un projet commun.

Ce projet a eu l'intérêt de pouvoir nous servir de base dans différentes matières que nous suivons actuellement, depuis le projet LEGO. En effet, nous avons pu découvrir et nous interroger sur différents mécanismes lors de la réalisation de ce challenge.

Cependant, outre les éléments mécanique et informatique, il y avait tous les éléments liés à la gestion et à l'amélioration du projet. En effet, lors de ce projet, comme nous l'avons vu, nous avons fait beaucoup de prototypes différents avant d'arriver à une solution valable, cela nous a donc permis de constater que bien souvent, il est très utile de toujours remettre en question ses réalisations et de ne pas hésiter à les démonter pour en refaire des mieux. En effet, en général, à chaque fois que nous tentions une modification nous avions quasiment à chaque fois réussi à améliorer certaines fonctionnalités par rapport au modèle précédent. C'est une des méthodes clés que nous retiendrons grâce à ce projet, à savoir de ne jamais rester sur un acquis et de toujours chercher à améliorer la réalisation.

C'était donc pour nous, un projet très original qui nous a permis d'apprendre à nous connaître mais aussi de nous apporter pleins de choses tant sur l'aspect technique qu'organisationnel.

# Annexes : le programme

---

```
task main()
{
    SetSensorLowSpeed(IN_4);

    do //Tant qu'il y a quelque chose à moins de 30 cm du capteur, on attends
    {
        Wait(25);
    }while(SensorUS(IN_4)<30);

    RotateMotor(OUT_A,100,500); //On avance le robot
    Wait(1000); //On s'assure de l'arrêt du robot avant de tirer

    RotateMotor(OUT_B,100,60); //Premier tir de catapulte
    RotateMotor(OUT_B,100,-60); // On remet la catapulte à sa position initiale

    RotateMotor(OUT_C,100,-45); //On tourne le chargeur de 45° => on libère une balle

    Wait(1000); //On attend que la balle soit dans la catapulte

    RotateMotor(OUT_B,100,60); //Second tir de catapulte
    RotateMotor(OUT_B,100,-40); //On descend un peu la catapulte pour la placer à hauteur de la 3ème balle

    RotateMotor(OUT_A,100,500); //On avance le robot

    RotateMotor(OUT_C,100,-45); //On tourne le chargeur de 90° => on libère l'autre balle

    Wait(1000); //On attend que la balle soit dans la catapulte

    RotateMotor(OUT_B,100,-20); // On remet la catapulte à sa position initiale
    RotateMotor(OUT_B,80,60); //Dernier tir de catapulte

    RotateMotor(OUT_A,100,500); //On avance le robot jusqu'au bout

    StopAllTasks(); //On arrête tout
} //fin du main
```