



08/10/2023

# Plan de Développement projet Intelligence Artificielle

## Plan de Développement

BONNARDEL Adrien 12106324

DUEZ Valentine 12107343

ROMANET Clément 12213747

ROUME Emma 12125449

<b>1 - Introduction.....</b>	<b>3</b>
1.1 Objet.....	3
1.2 Terminologie et sigles utilisés.....	3
1.3 Documents de référence.....	3
<b>2 - Organisation du projet.....</b>	<b>3</b>
<b>3 - Méthode de développement et calendrier directeur.....</b>	<b>4</b>
3.1 Méthode de développement.....	4
3.2 Objectif des étapes.....	4
3.3 Fournitures:.....	5
3.4 Calendriers :.....	5
<b>4 - Plans d'étapes:.....</b>	<b>5</b>
4.1 Plan pour l'étape d'analyse du besoin:.....	5
4.1.1 Tâches et calendrier.....	5
<b>5 - Méthodes, techniques et outils:.....</b>	<b>6</b>
<b>6 - Infrastructures et moyens:.....</b>	<b>6</b>
<b>ANNEXES.....</b>	<b>6</b>
ANNEXE 1.....	7
ANNEXE 2.....	8
ANNEXE 3.....	9
ANNEXE 4.....	10

# 1 - Introduction

## 1.1 Objet

Ce document a pour but de planifier le développement de Pablo. Robot créé pour le projet d'initiation à l'IA. Pablo le robot devra être autonome lors de l'évaluation.

## 1.2 Terminologie et sigles utilisés

Cahier des charges : CdC

Intelligence Artificielle : IA

Gestion de projet : GdP

Plan de Développement : PdD

Work Breakdown Structure : WBS

## 1.3 Documents de référence

Nos documents de références sont le descriptif présent sur le site du [cours](#), la librairie utilisée est [LeJOS, Java for Lego Mindstorms](#) et notre cahier des charges.

# 2 - Organisation du projet

Nous avons décidé de réaliser une organisation de projet active qui permet à chaque membre d'explorer un rôle différent. L'annexe n°1 montre le tableau récapitulatif des rôles de chacun.e en fonction des semaines.

Description des rôles :

- Chef de projet :
  - Coordonner les activités de l'équipe
  - Gère les réunions (temps, décisions prises)
  - Gère le planning
- Responsable conception et développement :
  - Supervise le codage et la programmation du robot
  - Gère la documentation de conception et de développement
- Responsable des tests :
  - Élabore un plan de test (y compris ceux d'intégration et de recette)
  - S'occupe de la gestion de la qualité du code source
  - Effectue les tests rigoureux pour garantir le bon fonctionnement de l'automate

- Responsable de la communication :
  - Organise les réunions de l'équipe
  - Tient à jour les documents de projet, y compris les rapports de réunions et les plannings
  - Gère la bonne communication au sein de l'équipe

## 3 - Méthode de développement et calendrier directeur

### 3.1 Méthode de développement

Nous avons opté pour une approche agile à la fois en interne et en externe pour notre projet. Nous avons la chance de pouvoir nous entretenir avec le client chaque semaine, ce qui rend crucial de documenter nos progrès pour les présenter régulièrement. Notre objectif est de nous assurer que nous respectons les exigences du projet et d'obtenir ainsi la validation du client. Cette communication hebdomadaire avec le client est essentielle pour maintenir l'alignement et s'assurer que nous sommes sur la bonne voie.

En interne, nous allons utiliser des boucles itératives courtes, nous permettant d'obtenir des micro-retours réguliers. Cette approche nous offre la flexibilité nécessaire pour ajuster nos réalisations en fonction des retours obtenus. Ces ajustements continus sont essentiels pour garantir que notre travail interne reste en phase avec les attentes du client.

De plus, nous avons décidé de répartir les rôles en fonction des semaines. Cette rotation de rôles nous offre une meilleure polyvalence, tout en nous permettant d'approfondir notre expertise dans nos domaines respectifs. Cette polyvalence au sein de l'équipe favorise également une meilleure compréhension mutuelle des responsabilités, renforçant ainsi la collaboration.

### 3.2 Objectif des étapes

La réalisation du code se divise en plusieurs étapes, nous avons déjà pu réaliser le cahier des charges, nous rentrons maintenant dans la phase de conception et de développement du projet. Cette étape consiste à réaliser le code qui permettra au robot d'atteindre l'objectif qui lui est donné. Nous aurons ensuite un plan de tests à rendre dans lequel nous inscrirons les tests réalisés permettant de justifier que notre logiciel respecte le cahier des charges. Le code source et la documentation interne seront l'avant dernier document à rendre, que nous réaliserons avec la javadoc en parallèle d'une description des classes (méthodes, attributs). La dernière étape est le rapport final du projet qui contiendra toutes les informations qui ne figurent pas sur les autres documents demandés.

### 3.3 Fournitures:

Documents à rendre au cours du projet :

Etape	Fourniture	Echéance
Analyse de la demande	Cahier des charges Plan de développement	3 semaines 5 semaines
Réalisation du code	Plan de tests Code source Documentation interne	10 semaines 11 semaines 11 semaines
Evaluation	Rapport final	12 semaines

### 3.4 Calendriers :

Nous avons décidé de réaliser un échéancier sous la forme d'un diagramme de Gantt (annexe ...). Cela nous permet d'effectuer un planning qui montrera l'avancement du projet. Il sera également le fil conducteur du projet.

Nous réalisons également un rétro-planning, il sera présent dans le rapport final et il montrera les étapes où nous avons pris un temps différent de celui prévu, voir que nous n'avions pas prévu une étape.

## 4 - Plans d'étapes:

### 4.1 Plan pour l'étape d'analyse du besoin:

#### 4.1.1 Tâches et calendrier

WBS	Tâches	Objet
1 1.1 1.2	Organisation du projet Gestion du projet Planification	Définition des outils à utiliser Mise en place d'outils permettant la planification (échéancier, moyen de communication)
2	Spécification	

2.1	Analyse de la demande	Collecte des exigences du client, des besoins technique et fonctionnels
2.2	Rédaction du CdC	Rédaction des différentes parties du CdC
3	Conception	
3.1	Conception architecturale	Définition des classes ; méthodes ; attributs ; packages
3.2	Développement du PdD	Réalisation du plan de développement
4	Développement	Classes motor/sensor qu'on crée et qu'on ne touche plus après
4.1	Module 1 : Fonctions de base	Aller au 1er palet, l'amener dans le but puis se retourner
4.2	Module 2 : Stratégies simples	Construction de la partie autonome
4.3	Module 3 : Construction automate	Optimisation pour avoir le meilleur rendu
4.4	Module 4 : Optimisation des modules 2 et 3	Le plan de test est réalisé en parallèle de chaque module
4.5	Plan de tests	
5	Rapport final	
5.1	Compétition entre robots	Compétition qui a lieu le jour de rendu final du projet
5.2	Rédaction du rapport final	Rédaction du rapport final permettant d'implémenter les éléments non précisés auparavant

## 5 - Méthodes, techniques et outils:

Les principaux outils utilisés sont :

- Google Drive pour le partage de ressources et la rédaction commune
- Discord pour la communication
- GitHub / Git pour le partage du code
- Java, par l'IDE Eclipse, et la librairie LeJOS
- le robot fourni

Les méthodes que nous utilisons :

- Gestion agile du projet

## 6 - Infrastructures et moyens:

Nous sommes 4 à développer le robot. Nous utilisons nos ordinateurs personnels.

# ANNEXES

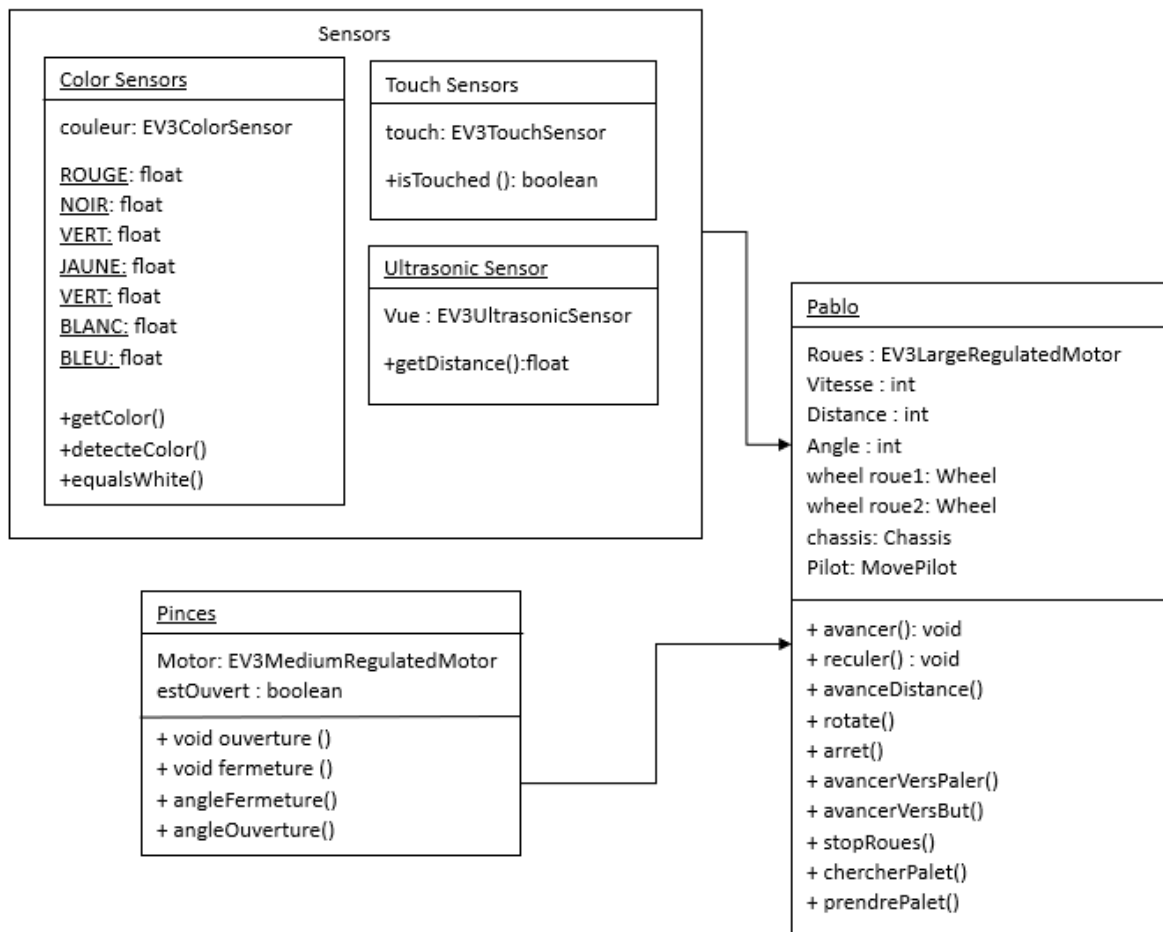
## ANNEXE 1

Planning des rôles au sein du projet :

	Chef de projet	Responsable conception et développement	Responsable des tests	Responsable communication
semaine 4	Valentine	Emma	Clément	Adrien
semaine 5	Adrien	Valentine	Emma	Clément
semaine 6	Clément	Adrien	Valentine	Emma
semaine 7	Emma	Clément	Adrien	Valentine
semaine 8	Valentine	Emma	Clément	Adrien
semaine 9	Adrien	Valentine	Emma	Clément
semaine 10	Clément	Adrien	Valentine	Emma
semaine 11	Emma	Clément	Adrien	Valentine
semaine 12	Valentine	Emma	Clément	Adrien

## ANNEXE 2

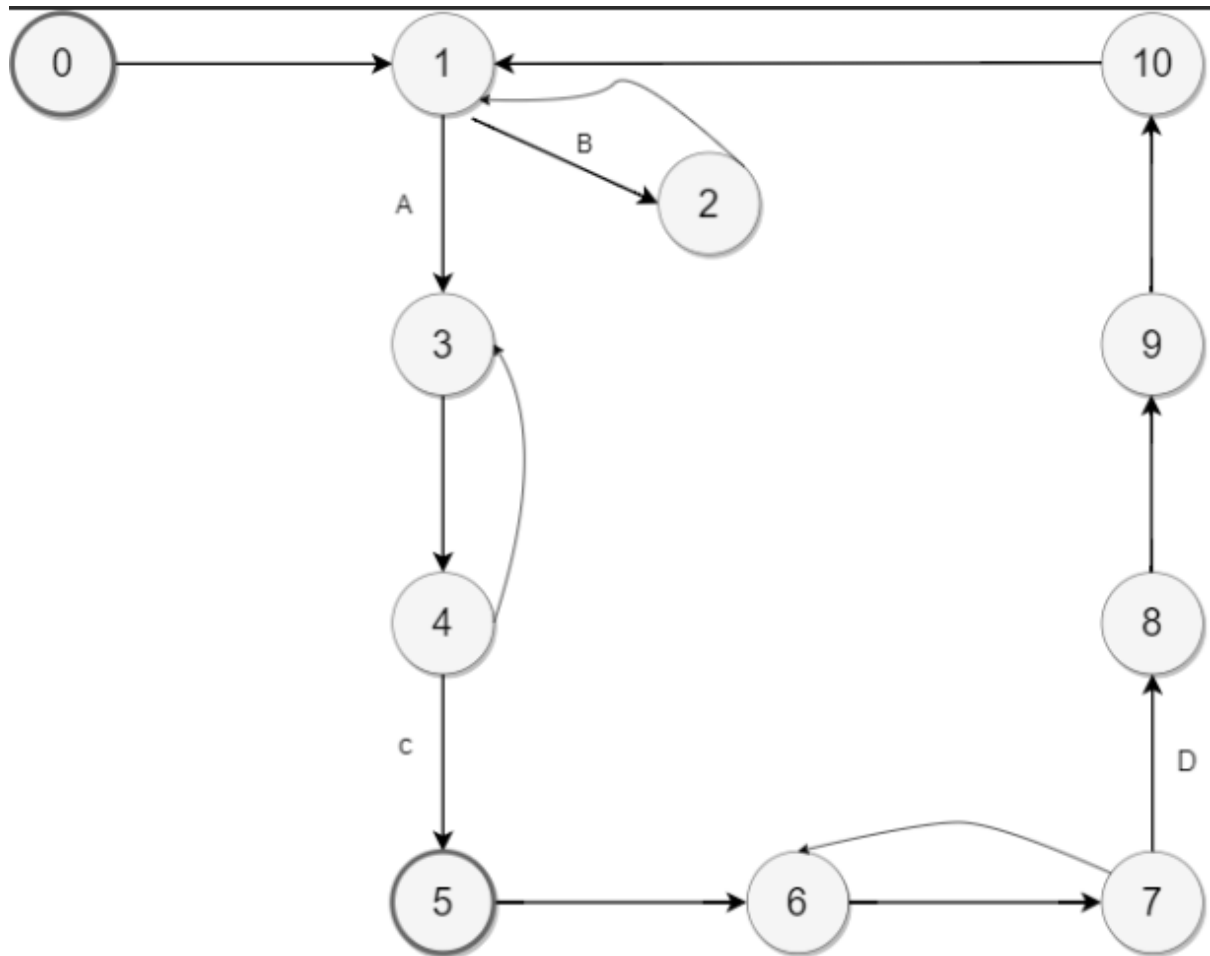
Voici la première version de notre diagramme des classes, que nous modifierons au cours des semaines à venir.





## ANNEXE 3

Diagramme à état fini :



0 : Départ  
1 : cherche palet  
2 : tourne  
3 : va vers palet  
4 : arrêt devant palet  
5 : ferme les pinces  
6 : se tourne vers le but  
7 : va vers le but  
8 : ouvre les pinces  
9 : recule  
10 : demi tour

A : détecte un palet  
B : ne détecte pas de palet  
C : sensor touch détecte un palet  
D : dépasse la ligne blanche du but adverse

## ANNEXE 4

[illegible]