|  |  |
| --- | --- |
| Référence du document  Document d'architecture general.docx | Type de document : |
| Qualification  V1 | Auteur(s) : A.SANNIER  A.PERRIN |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Modifications apportées à ce document : | | | | |
| **Version** | **Date** | **Auteur** | **Action** | **Modifications** |
| V1 | 20 mars 2020 | A.SANNIER  A.PERRIN | Création |  |

Table des matières :

[1. INTRODUCTION : 3](#_Toc35510366)

[1.1 Contexte : 3](#_Toc35510367)

[1.2 Présentation de l’équipe : 3](#_Toc35510368)

[2. PRESENTATION DU PROJET : 3](#_Toc35510369)

[2.1 Expression du besoin : 3](#_Toc35510370)

[2.2 Exigences 3](#_Toc35510371)

# INTRODUCTION :

## Contexte :

Nous avons dû réaliser ce projet dans un cadre pédagogique. Dans la dominante ISE OC que nous avons choisi, nous avons consacré les heures de bus de communication ainsi que les heures de MOQL pour le réaliser.

N’ayant pas pu nous rendre dans les salles TP prévues à cet effet pour cause de fermeture de l’école, nous avons dû réaliser ce projet en Visio conférence, chacun chez nous.

## Présentation de l’équipe :

Alexandre SANNIER : Employé par Transdev ROUEN en tant que Agent de maitrise dans l’informatique industriel et l’électronique

Arthur PERRIN : Je sais pas trop ce qui fou , mais il est dans l’oise avec plein de malade du cococorona

Nous sommes actuellement tous les deux en deuxième année du cycle ingénieur à l’esigelec, en alternance.

# PRESENTATION DU PROJET :

## Expression du besoin :

Nous devons pouvoir contrôler un robot à distance afin que ce dernier puisse se déplacer un espace délimité. Ce dernier doit pouvoir détecter les obstacles, à l’aide d’un capteur pouvant pivoter, les signaler et s’arrêter au besoin.

## Exigences logiciels

* **SYS\_0010**:

Nom : CommandeRobot

Description : Le SAMBOT doit répondre aux commandes envoyées par l’utilisateur.

* **SYS\_0020 :**

Nom : DetectionObstacle

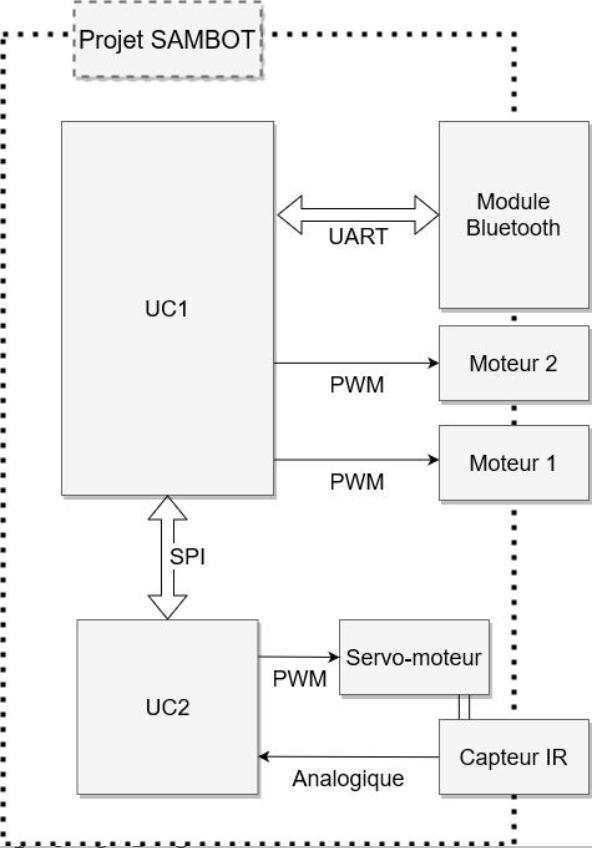
Description : Le SAMBOT doit pouvoir détecter un obstacle et s’arrêter lors de cette détection. Il doit également envoyé une notification à l’utilisateur de cette détection.

* **SYS\_0030 :**

Nom : InterfaceBluetooth

Description : Les commandes doivent pouvoir être envoyées via une interface Bluetooth ainsi que les notifications d’obstacles.

## Solutions techniques



|  |  |  |
| --- | --- | --- |
|  | Arthur PERRIN | Alexandre SANNIER |
| UC1 | Atemga328p (atmel studio) |  |
| UC2 | Attiny85 (atmel studio) |  |
| Servo-moteur | Similaire ESIGELEC |  |
| Capteur IR | Similaire ESIGELEC |  |
| 2 Moteurs | 2 LEDs |  |
| Module Bluetooth | Similaire ESIGELEC/TTL converter |  |

# Architecture Logiciel

Nous allons ici lister exigences architectural logiciels. Elles seront en accord avec les exigences logicielles citées plus haut.

Elles seront nommées HLR\_XXXX

* HLR\_0100

Nom : MouvementRobot

Description : Lorsqu’une commande est passée le robot doit réagir en fonction de cette commande.

Couvre : SYS\_0010

Module : Robot

* HLR\_0200

Nom : InterfaceUtilisateur

Description : Permet de passer une commande au robot via une interface utilisateur distante.

Couvre : SYS\_0020

Module : main

* HLR\_0300

Nom : MouvementCapteur

Description : Permet de commander la direction du capteur d’obstacle.

Couvre : SYS\_0010

Module : capteur

* HLR\_0400

Nom : DetectionObstacle

Description : Permet de détecter un obstacle situé à une distance donnée du capteur

Couvre : SYS\_0020

Module : capteur

* HLR\_0500

Nom : EnvoiNotification

Description : Envoi une notification à l’utilisateur lorsqu’un obstacle est détecté

Couvre : SYS\_0020 , SYS\_0030

Module : main

* HLR\_0600

Nom : CommunicationUtilisateur

Description : Toutes les communications avec l’utilisateur doivent se faire par Bluetooth

Couvre : SYS\_0030

Module : com

# Architecture Logiciel détaillée

Nous allons ici lister exigences architectural détaillées logiciels. Elles seront en accord avec les exigences architectural citées plus haut.

Elles seront nommées DDR\_XXXX

* DDR\_0100

Nom : Mouvement Robot

Description : Le robot doit bouger dans la direction passée en paramètre. Si aucune direction il s’arrete

Couvre : HLR\_0100

Fonction : void mvtRobot(char sens)

* DDR\_0200

Nom : Stop Robot

Description : Le robot doit s’arrêter sur commande

Couvre : HLR\_0100

Fonction : void stopRobot()