**Cahier des charges**

*Thomas Paredes*

*Benoît Cavallo*

*LP IOTIA - 2019/20*

A- Description générale du projet

Le projet mBot consiste à programmer un robot de type mBlock pour intervenir dans des zones difficiles (incendie) et prélever certaines données.

Les données prélevées seront des mesures de températures et la position GPS de celle-ci.

# B - Schéma d’architecture : schémas matériels et logiciels

**Matériels**

Robot « Mbot » de MBlock.

Capteur Ultrason.

Carte Arduino mega 2560.

Carte LinkItOne.

Capteur de température.

Module GPS.

**Logiciels**

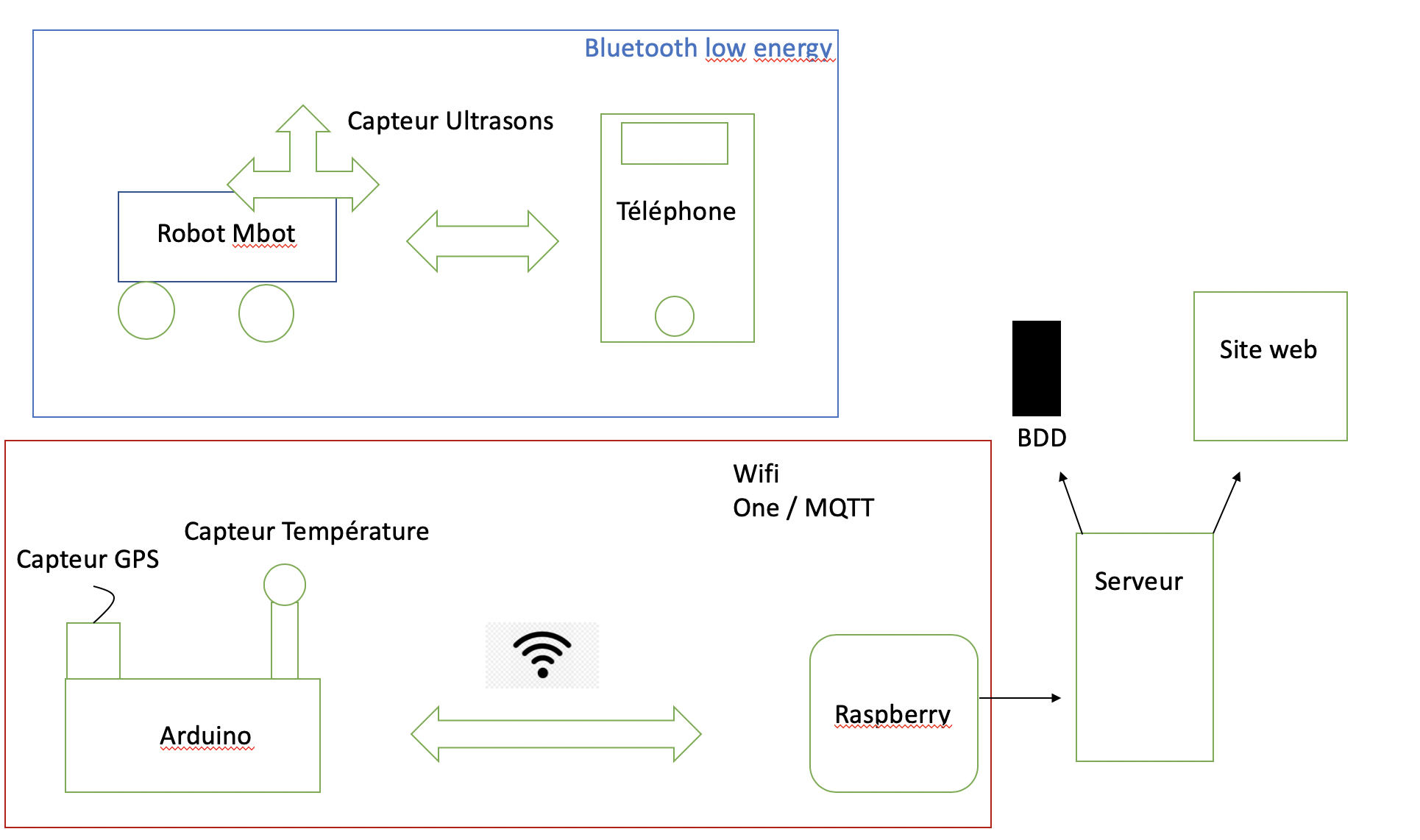
MBlock (IDE)

Arduino (IDE)

Application téléphone MBlock

Site web (Local)

BDD (MySql)



C - Décomposition des différentes fonctionnalités

L’application permettra d’illustrer les 3 modules de l’UE 63 :

Module “Connexion entre Objets” de Cécile Belleudy :

Pour l’utilisation du robot nous allons programmer les déplacements du robot avec l’application mobile et le capteur à ultrasons pour se déplacer dans l’environnement. Le capteur de température et de position (GPS) transmettent les données par la LinkItOne en wifi.

Module “Développement Web Service” de Jean-Michel Bruneau :

Les données collectées seront stockées dans une BDD et visibles via un le site web dont l’accès sera sécurisé en https et/ou login mot de passe.

Module “Internet des objets” de Marie-Agnès Peraldi : Mise en place d’une infrastructure MQTT.

## Fonctionnalités

Cette partie regroupe les plus grosses fonctionnalités de notre projet à développer.

*Fonction 1* : contrôler les déplacements du robot à l’aide d’une application sur l’arduino (direction avant, arrière, gauche et droite). Le code est en arduino C. Les communications se font via bluetooth low energy.

→ Test Fonction 1 : définir un motif de déplacement codé en dur sur l’arduino et contrôler le robot avec cela.

*Fonction 2* : une application mobile pourra contrôler le robot

→ Test Fonction 2 : l’app peut envoyer des ordres à l’arduino et permet au robot d’être en mouvement.

*Fonction 3* : collecter les données températures et gps et envoies en wifi au serveur qui sont stockées dans la base de données.

→ Test Fonction 3 : Envoyer des données à stocker dans la BDD, et vérifier si la sauvegarde est bien effective.

*Fonction 4* : consulter les données sur le site web et application mobile qui seront présent sur un raspberry.

→ Test Fonction 4 : accèder aux relevées des données sur le site Web.

*Fonction 5* : Accès depuis l’internet : Serveur avec port ouvert pour accessibilité depuis l'extérieur.

→ Test Fonction 5 : depuis son téléphone ou un ordinateur, accèder au site Web.

*Fonction 6* : Installation et utilisation de Mosquitto. Il est disponible sur toutes les plateformes et s’installe en quelques minutes. Comme MQTT est un standard, nous pourrons communiquer avec nos objets connectés de la même façon. Publication (envoie) d’un message MQTT depuis le terminal.Recevoir (souscrire) des messages. Et sécurisation avec un mot de passes.

→ Test Fonction 6 :Communication et affichage de la température et données gps

*Fonction 7* : Module de connexion pour sécuriser l’accès au site Web avec un mot de passe haché.

→ Test fonction 7 : Essayer de se connecter au site sans identifiants ou mot de passe.

## D - Scénario/acteur/rôles

Scénario 1 :

Fonctions associées : F1 et F2

Description du scénario : Un utilisateur peut au travers du téléphone en local faire avancer le robot.

Scénario 2 :

Fonctions associées : F3

Description du scénario : Un utilisateur collecte les données de température associé à leurs positions GPS respectives.

Scénario 3 :

Fonctions associées : F4

Description du scénario : Un utilisateur accède au site Web en local et visualise les données de température associé à leurs positions GPS respectives.

Scénario 4 :

Fonctions associées : F5

Description du scénario : Un utilisateur accède au site Web depuis n’importe où.

Scénario 5:

Fonction associées : F6

Description du scénario: Visualisation des messages (échange) sur Mosquitto, depuis n’importe quelle plateforme

Scénario 6:

Fonction associées : F7

Description du scénario: une personne mal intentionné veut se connecter au site, mais il est empêché par le module de connexion

E - Répartition du travail

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Tâches | Etudiant | Dates | Etat d’avancement |
| Réalisation du CDC | Benoît/ Thomas | 13/03/2020 |  |
| Fonctionnement du robot Mbot, avancement, recul (utilisation du capteur ultrason) | Benoît/ Thomas | 13/03/2020 jusqu’au 21/03/2020 | En utilisant le langage dédié à l’équipement on arrive à déplacer le robot. |
| Communication entre arduino et raspberry( récupération des différentes donnés) | Benoît | démarrage le 19/03/2020 | Travail à démarrer le 3/4 |
| Mise en place de l’appli web avec visualisation des données | Thomas | démarrage le 19/03/2020 | Thomas à développer un board avec les températures et une map avec des coordonnées de la position GPS du robot. Le site web et la BD sont sur le RPI . La BD sera en mysql |
| Mise en place d’un module de connexion au site sécurisé hachage des données | Thomas | démarrage le 26/03/2020 | Fait en tp |
| Adaptation de la visualisation en mobile | Thomas | démarrage le 03/04/2020 |  |
| Insertion dans une plateforme MQTT mosquitto installer sur raspberry | Benoît/ Thomas | démarrage le 03/04/2020 | Sur l’arduino (client mqtt) sur l’arduino et grace à une librairie :  On enregistrera (publish) le capteur GPS et le capteur de température chqaue fois quíl y aura une lecture de ces deux capteurs.  La publication se fait auprès du broker installé sur le RPI ( accès securisé ou pas).  Sur le RPI il faut un code qui subscribe au broker de ces deux types de données et à chaque échantillon reçu il envoie sur la BD ave la date |