Cahier des charges IOTIAQUARIUM

Expression du besoin :

IOTIAQUARIUM a pour but d’aider à prendre soin de vos poissons en automatisant les tâches de contrôle et de maintenance. L’objectif est de communiquer des données liées à la survie de la faune et de la flore à une application qui génèrera des alertes et permettra d’intervenir sur les actionneurs. Contrairement à la nature, un aquarium n’est pas un environnement autonome et a besoin d’une assistance constante, ces tâches sont le plus souvent assurées manuellement et donc susceptible d’erreurs, le système devra donc reproduire au mieux un milieu de vie viable avec un minimum d’intervention humaine.

Problématiques :

* Alimenter les poissons.
* Allumer/éteindre la lumière.
* Entretien de l’eau.
* Hyper-vision de l’environnement.

Solution proposée :

Six capteurs sont utilisés dans la plate-forme pour mesurer les paramètres clés de l'aquarium, tels que la température, le pH, et d'autres pour contrôler l'état de l'aquarium (niveau et fuites, luminosité). De plus, il y a 3 actionneurs différents pour automatiser des tâches telles que nourrir les poissons, l'activation des pompes pour le changement d'eau, le contrôle de l'intensité de la lumière pour simuler les cycles jour/nuit et créer un réflexe de Pavlov pour l’alimentation des poissons. Une API complète sera créée pour contrôler facilement le système via un Raspberry. Nous prévoyons également de concevoir une application web qui permettra de stocker dans une base de données les informations recueillies et de les visualiser à partir d'un navigateur et d'appareils iPhone/Android.

Fonctionnalités :

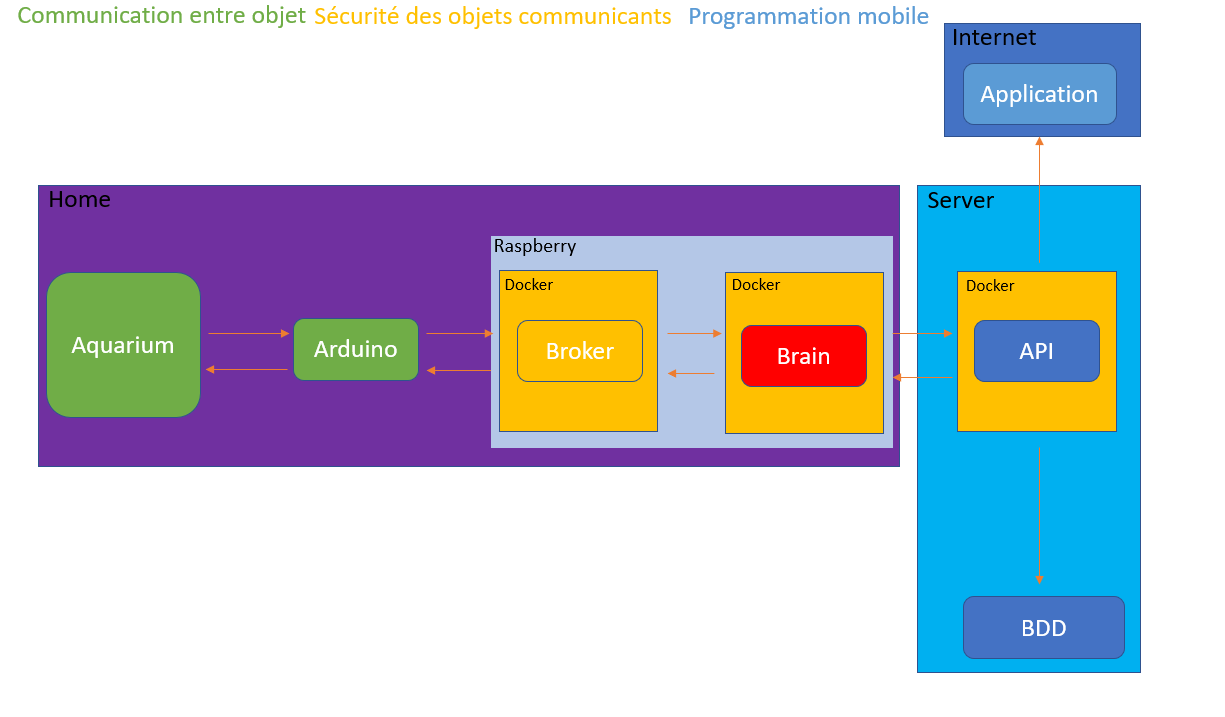
|  |  |
| --- | --- |
| * Fonction : Alimenter les poissons | |
| Objectif | Alimenter les poissons de manière automatique. |
| Description | Un moteur associé à un distributeur de nourriture, permet de gérer l’ouverture et la fermeture de la boite pendant une durée donnée qui permette de délivrer la quantité nécessaire de nourriture ( de l’ordre de la seconde) |
| Contraintes / règles de gestion | Données la quantité nécessaire de nourriture aux poissons. |
| Niveau de priorité | Priorité haute |

|  |  |
| --- | --- |
| Fonction : Allumer/Eteindre la lumière | |
| Objectif | Allumer ou éteindre la lumière selon le niveau de luminosité |
| Description | Un capteur de lumière permet de déterminer si l’aquarium a besoin de lumière ou pas. |
| Contraintes / règles de gestion | Allumer la lumière quand cela est nécessaire. |
| Niveau de priorité | Priorité haute |

|  |  |
| --- | --- |
| Fonction : Entretien de l’eau | |
| Objectif | S’assurer que le ph et que le niveau d’eau sont correctes. |
| Description | Un capteur de niveau d’eau et un capteur de Ph, permettent de s’assurer que ces mesures sont correctes. |
| Contraintes / règles de gestion | Rajouter automatiquement de l’eau si le niveau d’eau ne sont pas correctes, sans dépasser le seuil maximum.  Si le PH varit de 0.2 autour de 7, on déclenche une demande d’entretien de l’aquarium ->Test de safety : Ajuster le ph en ajoutant de l’acide ou de la soude. |
| Niveau de priorité | Priorité haute |

|  |  |
| --- | --- |
| Fonction : Hypervision de l’environnement | |
| Objectif | Manager l’aquarium via l’application. |
| Description | Une application permet de superviser l’aquarium à distance |
| Contraintes / règles de gestion | L’application doit pourvoir afficher toutes les données des capteurs (courbes de PH température, luminosité et proposer des aménagements par rapport à des consignes et des espèces de poissons |
| Niveau de priorité | Priorité haute |

Schéma du projet :



Notre but est de mettre en avant l’hétérogénéité de nos compétences, en répartissant le travail en fonction des différentes matières concernant le projet.

Acteurs :

Olivier Durand, Raphaël Betti, Dieunelson Dorcelus, Eric Harkat.

Calendrier :

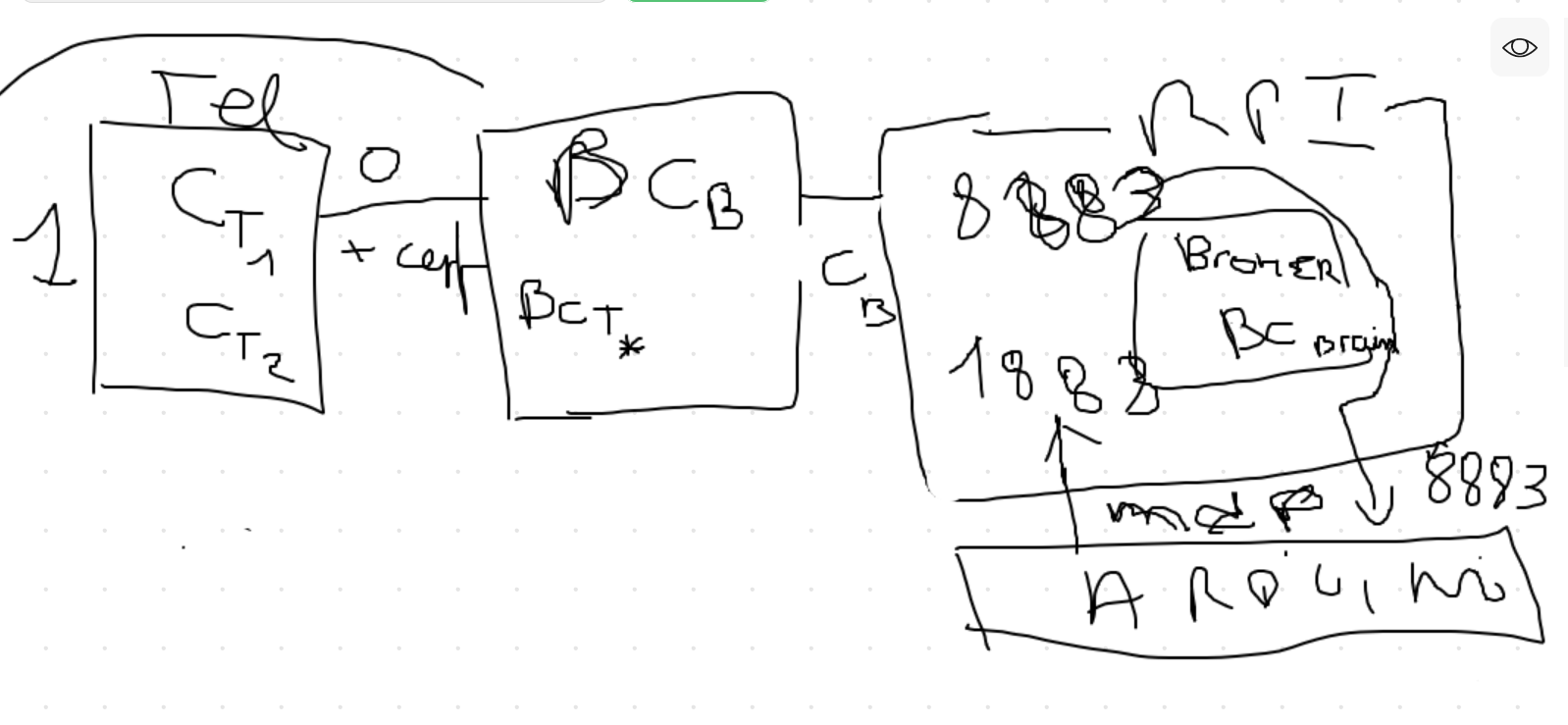
|  |  |  |
| --- | --- | --- |
| Date | Tâche | Avancement |
| 13/03/2020 | * Cahier des charges | 0% |
| 19/03/2020 | * Livraison du Brain | 10% |
| 20/03/2020 | * Livraison du cœur de l’API * Intégration entre le Brain et l’API * Configuration du Broker | 20%  30%  35% |
| 26/03/2020 | * Livraison de la partie Arduino * Intégration entre la partie Arduino et Brain via Broker | 40%  50% |
| 27/03/2020 | * Livraison des vues de l’application mobile | 70% |
| 02/04/2020 | * Intégration entre l’application et l’API | 75% |
| 03/04/2020 | * Livraison de l’application mobile | 85% |
| 09/04/2020 | * Préparation de la livraison finale | 95% |
| 10/04/2020 | * Livraison finale | 100% |

Sous-équipes :

|  |  |  |
| --- | --- | --- |
| Partie : Circuit et arduino | Partie : virtualisation et communication | Partie : web et mobile |
| Raphael Betti (leader) | Olivier Durand (leader) | Dieunelson Dorcelus (leader) |
| Eric Harkat | Eric Harkat | Eric Harkat |
| Olivier Durand | Dieunelson Dorcelus | Raphael Betti |

Circuit et arduino : Capteurs/ actionneurs testés lecture en mode stream avec sépartion par des ; avec une périodicité paramètrable pour les capteurs Prévoir une alerte sur seuil de niveau d’eau ou de température

Virtualisation et communication : le broker est prêt. Un mosquitto sécurisé avec mot de passe et certificat sécurisé . C’est fait avec docker compose. Vous allez mettre dans le github une doc qui décrit ce qui sera automatique et manuel. Ca permet une connexion au broker dans l’intranet avec mot de passe et depuis l’extérieur utiliser un certificat



Partie : web et mobile

Web api route ?

Brain capable d’écouter sur un port et retransmettre l’info par l’arduino Pour le moment le réseu de neurones est staique et les neurone composant sont écrasés à chaque stream de données.

Code est dans