



« supporté » = l'arduino est capable de gérer ce genre de requêtes, c'ad que celles-ci correspondent à une fonctionnalité qu'il offre

#### Fonctions de manipulation des différents senseurs/valve

Temperature
Lit la tension au borne de la thermistance, la transforme en °C et retourne ce résultat

Presence
Récupère la donnée du capteur infrarouge, renvoie un booléen

Set valve
Prend une valeur entre 0 et 100, la transforme linéairement pour rentrer dans la marge 0-5V

Get valve
Lit la tension au borne de la valve, la transforme en pour-cent et renvoie ce résultat

Start

Setup

Connexion au réseau wifi THRANH201 (et donc obtention d'une adresse ip)

Tentative d'enregistrement au près du serveur (RPi)

Réponse 'OK 200' de la part du serveur?

Oui

Setup du serveur (sur le arduino cette fois-ci) et configuration pour écouter sur port spécifié lors de l'enregistrement au près du Pi

On attend 30 sec (+ ou -) et on recommence

loop

En attente de requêtes

Une requête arrive

Analyse de la requête

Le type, le chemin spécifié sont supportés

Non

Envoi de 400

Oui

Le contenu, les headers sont supportés

Non

Envoi de 400

Oui

Détermination de la fonction répondant à la requête et appel de celle-ci

Récupération du résultat de la fonction, envoi de celui-ci dans une réponse http au pi dans le format correct (défini dans la cahier des charges pg 13)

#### Fonctions d'analyse des données

Pas encore conçu. L'idée est d'implémenter différents algorithmes (du plus simple vers le plus complexe et intelligent) qui en entrée prennent l'ensemble des données collectées sur un certain laps de temps et en sortie indiquent à quel moment et à quel point il faut ouvrir/fermer les vannes.

#### Enregistrement de nouveaux senseurs

Démarrage du RPi

Reçois automatiquement 192.168.10.1 et émet TRANH200

Commence à écouter pour des requêtes d'enregistrement sur le port 8080

S'associe à une base de données

Commence la récolte périodique des mesures sur tous les devices connectés et leur stockage dans la bd

Reçois requête PUT sur '/register'

Requête correcte

c'ad si content-type = json, request type = PUT, content = [json avec ip, port et type]

Non

Répond 400 Bad request

Oui

Enregistre device

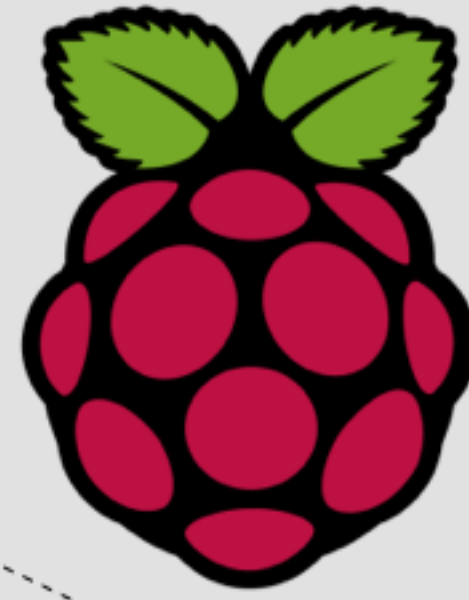
faillure

Répond 500 Int Error

succes

Répond 200 OK

C'ad crée un nouvel objet de la classe RemoteDevice (d'un type correct), l'inscrit dans la base de données et l'ajoute à la liste des devices connectés



Les fonctions présentées ici sont les fonctions de base demandées dans le cahier des charges. Le code est fait tel que l'implémentation de nouvelles fonctions (du genre mesure d'humidité, de luminosité, ...) soient (très) facilement implémentable. Les diagrammes de classes montrent plus en détails cette implémentation.

#### Différents types de senseur actuellement supportés

Temperature (Sensor)
Envoie une GET sur '/temperature'

Presence (Sensor)
Envoie une GET sur '/presence'

Valve (InterctaveSensor)
Envoie une GET sur '/valve'

Valve (InterctaveSensor)
Envoie une PUT sur '/valve' content-type='text/plain' content = [0-100]

#### Programme « principal »

Start

Démarre le serveur et le prépare à gérer les enregistrements des senseurs

Tous les x temps, récupère toutes les mesures/informations offertes par chacun des senseurs connectés

Enregistre ces données

Transmettre la modification d'état de la vane (au bon moment)

Récupérer un ordre d'ouverture/fermeture de vane

A chaque fois qu'il faut prédire l'ouverture/fermeture des vannes (dépend du fonctionnement des algorithmes développés), transmettre l'ensemble des données collectées les x derniers temps aux algos

#### Fonctions de gestion des données

Enregistrement des éventuelles information d'un remote device (son nom, son adresse ip?)

La base de données spécifiée existe-t-elle?

Non

Création d'une nouvelles base de données

Oui

Récupération de la structure de la base de données existante.

Enregistrement/Récupération d'un lot de mesures associé à remote device

Le module data et la structure des données n'est pas définitive. Ce comportement est néanmoins, vue d'un haut niveau d'abstraction, commun au deux structures/implémentation.