



Rapport S8 APP-VOLET

MEYER Johan
DESAGE Hippolyte
MILOME Arnaud
YOVODEVI Zaide
ERUAM Hervé

Table des matières

1.	Remerciements	3
2.	Présentation et objectifs du semestre	4
	• Introduction au projet	4
	• Résumé du semestre précédent :	5
	• Fonctionnement global :	5
	• Schéma globale :	6
	• Objectifs pour ce semestre :	6
3.	Gestion des séances	7
	• Affectation des rôles	7
	• Gestion du temps	8
4.	Partie électronique	10
	• Introduction	10
	• Schéma du câblage réalisé au S7	10
	• Montage finale sur la maquette	11
	• Problèmes rencontrés	11
5.	Partie programmation	13
	• Site de maintenance	13
	Introduction	13
	Fonctionnement :	13
	• Raspberry Pi : Utilisation de home assistant	15
	• Automatisations dans home Assistant :	17
	• Programmation embarquée de l'ESP-32 :	18
	Introduction :	18
	Les autres classes :	20
	SETUP & LOOP :	21
6.	Conclusion et ouverture sur le prochain semestre	22
7.	Apports personnels	23
	• Hippolyte	23
	• Johan	23
	• Arnaud	23
	• Zaïde:	23
	• Hervé :	24

1. REMERCIEMENTS

Nous voulons tout d'abord remercier l'université Savoie Mont Blanc ainsi que l'école Polytech Annecy-Chambéry pour nous avoir fourni un cadre d'étude privilégié, et ce, tout au long de nos séances d'apprentissage par problèmes et par projet du semestre 8.

Nous remercions également Mr Lionel VALET pour l'encadrement, l'aide et les conseils qu'il nous a apporté dans ce projet.

Nous voulons également remercier les experts pour leurs aides et pour les informations qu'ils ont pu nous donner et qui nous ont permis de mieux comprendre le fonctionnement de certaines parties de notre maquette.

2. PRESENTATION ET OBJECTIFS DU SEMESTRE

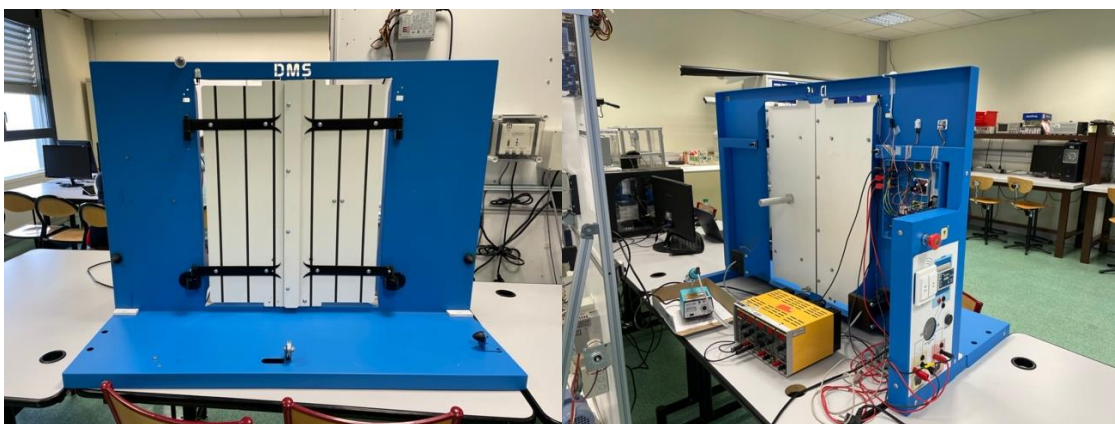
- **Introduction au projet**

Les Apprentissages par projet commencent au semestre 6 et se déroulent jusqu'à la fin de notre cursus d'ingénieur. Ils s'organisent en groupe d'étudiants. Le module d'APP vise à développer des compétences telles que des compétences techniques particulières dans un domaine mais aussi des compétences transversales d'ingénieur comme le travail en équipe (responsabilités, animation de réunion, organisation, rédaction de comptes-rendus de réunion, communication interne et bien d'autres). Les dernières compétences enseignées à travers ce module sont toutes celles liées à la gestion de projet c'est-à-dire la définition d'un cahier des charges, l'organisation, la répartition, la planification et le suivi des tâches, la consultation d'experts et l'acquisition de connaissances.

Il est important, avant toutes choses, d'apporter quelques éléments introductifs permettant de positionner notre projet dans les nombreux domaines pouvant composer notre formation. En quelques mots, le but de notre projet est de rendre automatique un volet battant à l'aide de différents capteurs et actionneurs.

Ce projet s'inscrit dans le domaine de la domotique, soit des techniques pour l'automatisation de l'habitat qui est une technologie en pleine essor dans la construction ou la rénovation de bâtiment car elle apporte confort et simplicité. De plus, de nombreuses grandes entreprises comme Samsung, Apple ou encore Amazon se lancent dans ce domaine avec des objets connectés de plus en plus performant et capable de réaliser de plus en plus de choses de notre quotidien. C'est donc pour ces raisons que nous avons choisi ce projet.

Dans notre cas, nous avons à notre disposition une maquette échelle réduite de volet battant que nous pouvons modifier à notre guise afin de répondre à un cahier des charges que nous allons nous fixer. Que ce soit l'optimisation de la gestion de l'énergie, la sécurité des personnes et des biens, la facilitation de la vie dans l'habitat, la domotique et donc notre projet sera la réponse à tous ces besoins.



Concernant la réalisation du projet, il se déroulera durant les semestres 6, 7, 8 et 9 de notre formation. Nous consacrerons chacun des semestres à une étape précise de la conception du produit fini. Le semestre actuel (semestre 8) est consacré à la réalisation concrète de tous les éléments que nous avons mis en place aux semestres derniers au cours de la mise en place des objectifs et du cahier des charges. Ce semestre est le plus conséquent en termes de nombre d'heures et est la concrétisation des deux derniers

semestres. Il est, pour les membres du groupe qui seront en mobilité l'an prochain, le dernier semestre d'APP.

- **Résumé du semestre précédent :**

Après avoir pris en main la maquette au semestre 6, nous avons mis en commun les objectifs que nous aurions aimé atteindre. C'est de ce travail préliminaire que nous sommes partis pour mettre en place notre cahier des charges durant le semestre 7.

Il a en effet fallu mettre nos idées au clair et définir exactement les différents objectifs que nous voulions atteindre pour la fin du semestre 8, afin de gagner du temps et d'être sûr que tout le groupe avançait bien dans la même direction. Ce travail a été important pour le suivi de nos objectifs. Nos objectifs seront susceptibles d'être modifiés en fonction des différents problèmes que nous rencontrerons. Nous pourrions donc comparer le cahier des charges avec le produit final.

Durant le semestre dernier, nous avons également caractérisé les différents éléments de notre maquette (capteurs et moteurs) ce qui va nous aider pour la mise en place des codes. Le fait de bien connaître notre maquette nous permettra plus aisément de mettre en place notre code qui s'appuiera directement sur les valeurs renvoyées par les capteurs et qui piloteront nos moteurs.

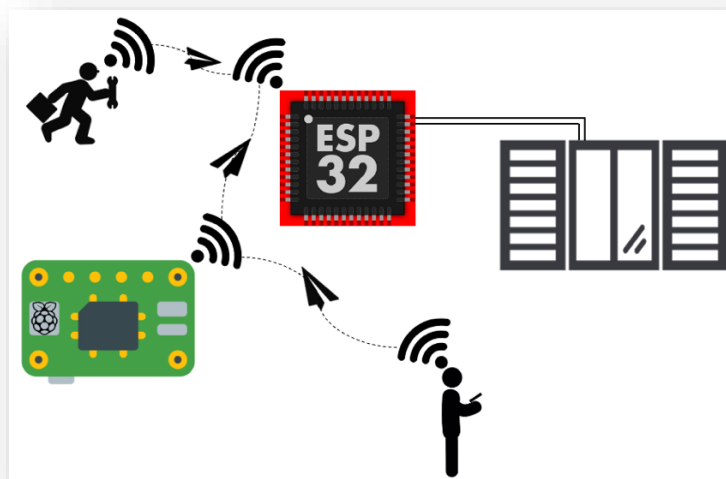
- **Fonctionnement global :**

Notre volet battant automatisé et connecté peut être contrôlé à distance via un serveur Raspberry Pi et un module ESP 32 qui assurent la commande du volet. L'utilisation de capteurs de courant permet de détecter les fins de course et de stopper le mouvement du volet une fois qu'il a atteint sa position maximale ou minimale. Le système peut également être équipé de nombreux autres capteurs tels que des capteurs de température, d'humidité ou encore de luminosité. Ces capteurs permettent au système de détecter les conditions météorologiques et d'agir en conséquence pour maintenir un niveau de confort optimal à l'intérieur de la maison.

Par exemple, si la température intérieure est basse, le système peut détecter cette variation et ouvrir les volets pour permettre aux rayons du soleil de réchauffer l'intérieur. Si la luminosité extérieure est faible, le système peut fermer automatiquement les volets pour maintenir la privacité de l'habitant et réduire la consommation d'énergie.

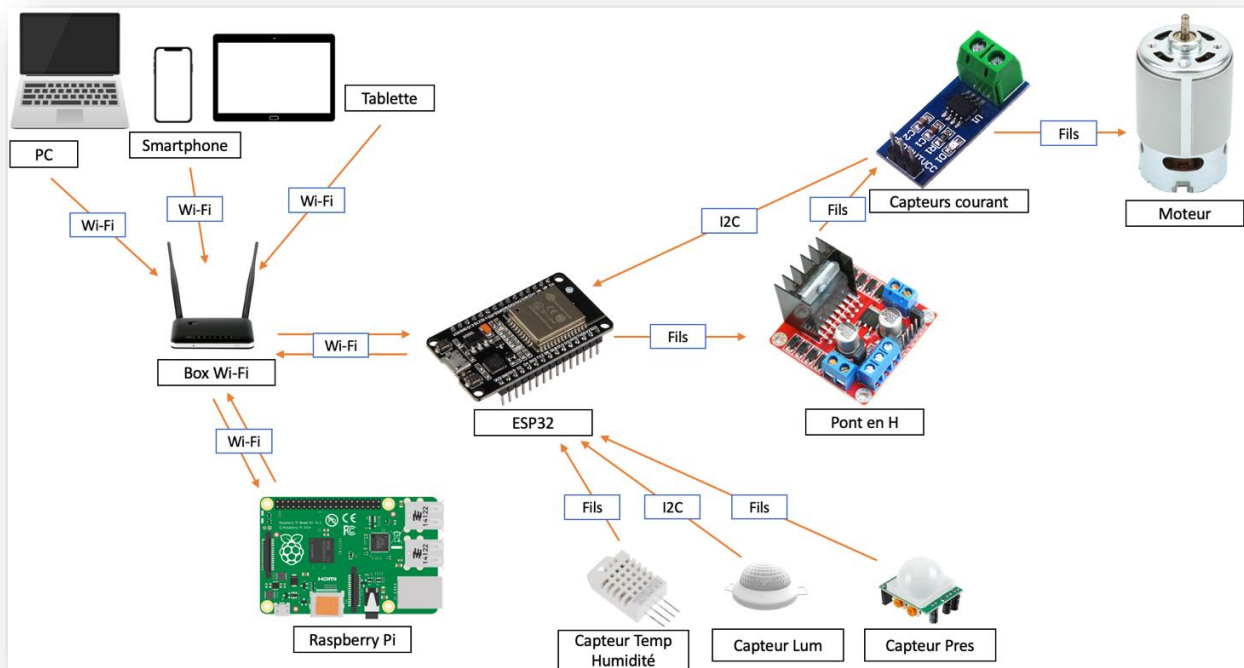
Le système est également équipé d'une interface utilisateur, telle qu'une application mobile ou un site web, qui permet à l'utilisateur de surveiller et de contrôler le volet à distance, ainsi que de régler les paramètres en fonction de ses préférences.

En somme, un volet battant automatisé et connecté peut être un ajout pratique et intelligent à toute maison, offrant un contrôle automatisé et personnalisé du volet pour répondre aux conditions météorologiques et aux préférences de l'utilisateur.



• Schéma globale :

Sur le schéma ci-dessous, on peut retrouver tous les composants de notre système ainsi que leur moyen de communication :



• Objectifs pour ce semestre :

Nous avons donc plusieurs objectifs de fonctionnalités pour la fin du semestre :

Fonctionnalité 1 : Un moyen de contrôle physique, intuitif et proche du volet permettant un réglage manuel à tout moment.

Fonctionnalité 2 : Suivi de la puissance développée par le volet lors des mouvements avec la possibilité d'arrêter rapidement le mouvement et revenir en arrière pour libérer l'obstacle (exemple : le cas d'un doigt ou main coincé entre le volet et le mur).

Fonctionnalité 3 : Un mode automatique qui ouvre et ferme les volets en fonction de l'heure, et des mouvements dans la pièce.

Sous-fonctionnalité 4.1 : Possibilité de configurer via une interface des routines en fonction du jour et de l'heure.

Sous-fonctionnalité 4.2 : Routines avancées en fonction du jour, de l'heure, des températures(int/ext), luminosité...

Fonctionnalité 4 : Proposer une interface de contrôle sur un site web (local).

Sous-fonctionnalité 5.1 : Proposer une interface de contrôle via une application.

Sous-fonctionnalité 5.2 : Rendre ces interfaces accessibles depuis internet (avec sécurité nécessaire).

Fonctionnalité 5 : Proposer un mode Régulation de la température qui se base sur les températures intérieures et extérieures, l'ensoleillement et la luminosité intérieure.

3. GESTION DES SEANCES

• Affectation des rôles

La bonne organisation du projet est passé par une affectation de rôles à chaque séance pour les membres de l'équipes. En effet, afin que les séances soit les plus productives possible, il est important que chacun joue un rôle pour optimiser les séances. Ces rôles changeaient à chaque séance sauf pour le chef de projet. Nous avons fait tourner les rôles afin que tout le monde puisse expérimenter différentes positions et ainsi en tirer le maximum de manière individuelle, nous avons donc chaque semaine :

- Un responsable projet supervisant le bon déroulé des séances et la cohérence globale du projet sur le semestre.
- Un secrétaire de séances gardant une trace écrite de ce qui est fait durant la séance mais également des objectifs pour les séances suivantes.
- Un animateur coordonnant les tâches effectuées par tous les membres du groupe durant la séance.
- Un responsable sécurité s'assurant du bon déroulé des manipulations afin de garantir le respect du matériel et la sécurité de tous.

On note que malgré ces rôles, chacun effectuait les tâches prévues durant la séance, tâches également réparties à chaque début de séances.

YOVODEVI Zaïde

ERUAM Hervé

MILOME Arnaud

MEYER Johan

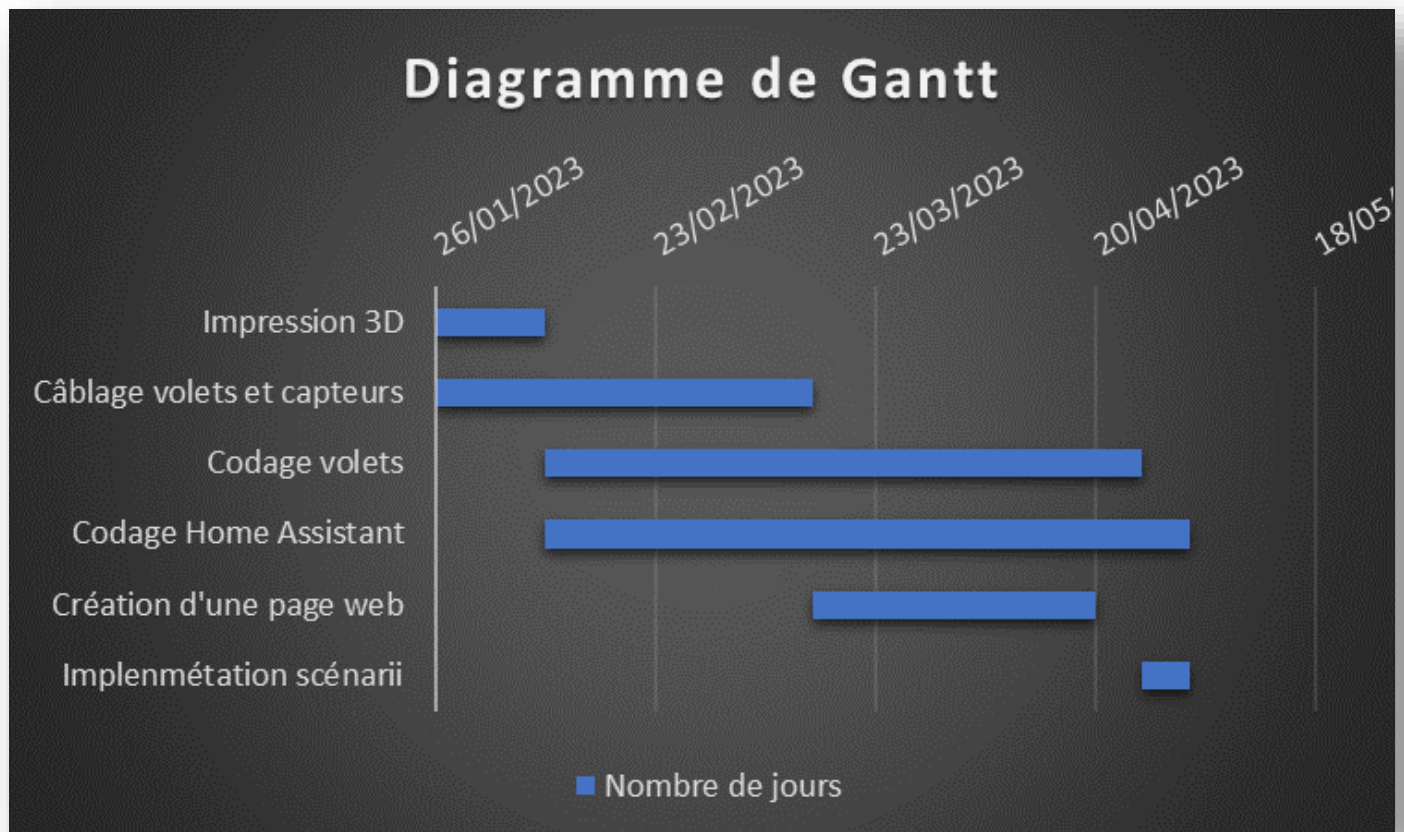
DESAGE
Hippolyte

26/01/2023	Responsable sécurité	Secrétaire	Animateur	Responsable de projet
09/02/2023	Secrétaire	Animateur	Responsable sécurité	Responsable de projet
23/02/2023	Responsable sécurité	Animateur	Secrétaire	Responsable de projet
01/03/2023		Responsable sécurité	Animateur	Responsable de projet
09/03/2023		Secrétaire	Responsable sécurité	Responsable de projet
15/03/2023	Secrétaire	Animateur	Responsable sécurité	Responsable de projet
24/03/2023		Responsable sécurité	Secrétaire	Responsable de projet
31/03/2023	Responsable sécurité	Animateur	Secrétaire	Responsable de projet
18/04/2023	Responsable sécurité	Animateur		Responsable de projet/ Secrétaire
20/04/2023	Animateur		Responsable sécurité	Responsable de projet/secrétaire
26/04/2023	Secrétaire	Animateur	Responsable sécurité	Responsable de projet
02/05/2023	Responsable sécurité		Secrétaire	Responsable de projet
03/05/2023				Responsable de projet

Ces rôles nous ont apporté beaucoup que ce soit en productivité mais aussi en termes d'apports personnels. En effet ces projets nous apprennent bien entendu à travailler en groupe mais ces rôles nous ont permis d'acquérir des compétences en leadership par exemple.

- **Gestion du temps**

Pour pouvoir avancer plus efficacement lors de ce semestre, nous avons mis en place un diagramme de Gantt qui nous permet de visualiser l'avancement des différentes tâches



Ci-dessous se trouve un tableau récapitulatif des séances qui nous permet d'avoir une vue globale de l'avancement du projet séances par séances. Ce tableau nous aura aussi permis de quantifier combien de séances il nous reste jusqu'à la fin du semestre pour effectuer nos différentes tâches prévues lors de celui-ci. On notera que chaque séance était suivie d'un compte rendu qui nous a permis de ne pas oublier le contenu des séances mais aussi les différents problèmes et objectifs de chaque séance.

<i>Numéro de la séance</i>	<i>Date</i>	<i>Principales réalisations durant la séance</i>
<i>Séance 1</i>	26/01/2023	Envoie des pièces pour les imprimer Recâblage du circuit électronique Résolutions de dysfonctionnements du code sur l'ESP
<i>Séance 2</i>	09/02/2023	Mise en place des capteurs sur la maquette Prise en main de Home Assistant
<i>Séance 3</i>	01/03/2023	Mise en place du circuit électronique complet avec le dessoudage des broches Début du code sur Home Assistant
<i>Séance 4</i>	24/03/2023	Codage du site de maintenance Résolutions de bugs de librairie sur Home Assistant Résolution de bugs sur la récupération de valeurs de capteurs
<i>Séance 5</i>	18/04/2023	Début du compte-rendu Établissement de la connexion de l'ESP au serveur MQTT
<i>Séance 6</i>	20/04/2023	Connexion ESP Raspberry opérationnelle Interface Home Assistant fonctionnelle
<i>Séance 7</i>	26/04/2023	Début de la mise en place des scénarios Mise en place de la sécurité d'obstacle sur l'ESP
<i>Séance 8</i>	02/05/2023	
<i>Séance 9</i>	03/05/2023	

4. PARTIE ELECTRONIQUE

• Introduction

La partie électronique a déjà été bien avancée au semestre 7 dans le sens où nous avons mis en place un schéma complet du circuit. L'objectif est dans un premier temps de mettre en place le circuit, puis de le fixer dans une petite boîte se trouvant au dos de la maquette qui est prévu à cet effet. Nous rappelons le schéma électrique complet qui est le suivant.

• Schéma du câblage réalisé au S7

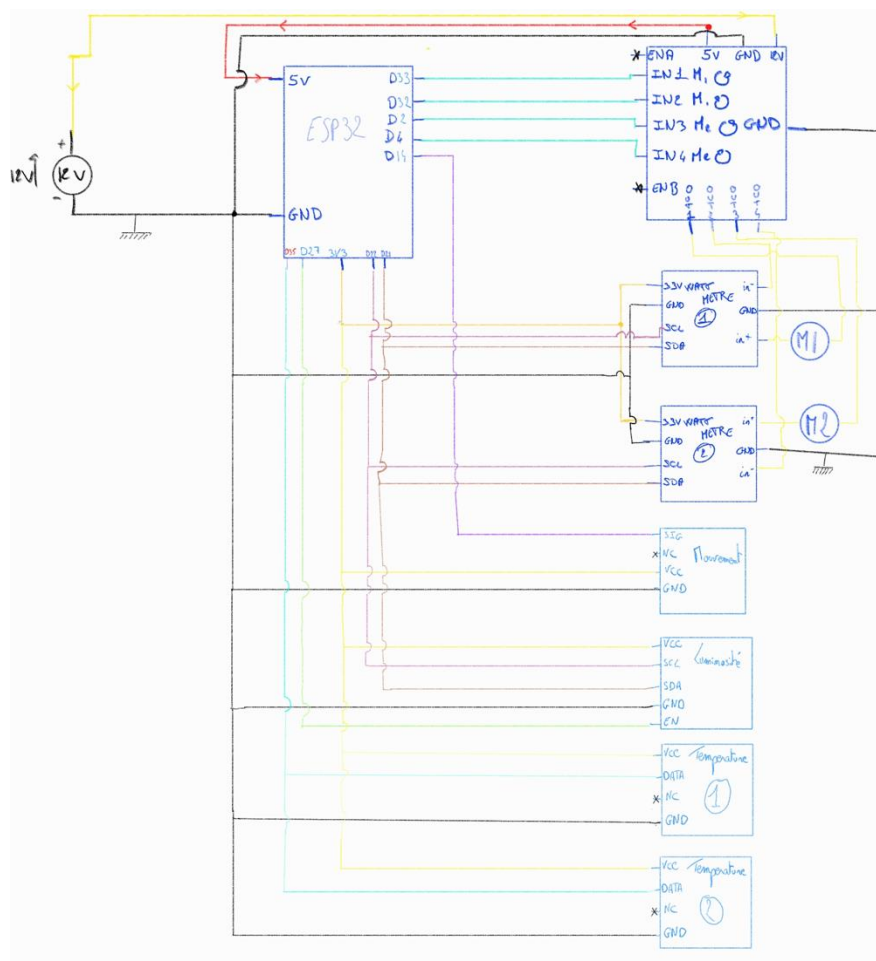


Figure 1 : Schéma électronique complet de notre système

Nous avons donc notre partie directement rattachée au volet. On retrouve notre microcontrôleur (ESP32) central dans ce schéma, il sera connecté à tous les autres éléments :

- 2 capteurs de puissances (pour les butées)
- 1 capteur de mouvement, 1 capteur de luminosité, 2 capteurs de température (intérieur et extérieur)
- 1 pont en H (pour modifier le sens de rotation de notre volet)

• Montage finale sur la maquette

Nous avons donc réalisé ce schéma à côté de la maquette pour vérifier son bon fonctionnement. Une fois que nous avons vérifié le bon fonctionnement de tous les éléments un à un, puis de l'ensemble du circuit avec un code simple nous avons pu l'intégrer à la maquette. On rappelle que nous avons réalisé des supports à l'imprimante 3D pour les capteurs de mouvement, courant, luminosité et température ainsi que pour l'ESP32 et pour le pont en H. Ces supports nous ont permis de pouvoir fixer les éléments à la maquette sans que les broches touchent la maquette et pour ainsi éviter les courts-circuits. Il y avait un espace prévu au dos de la maquette pour y intégrer notre circuit nous avons donc utilisé des plaquettes LABDEC autocollantes. Nous sommes donc arrivés à ce schéma :

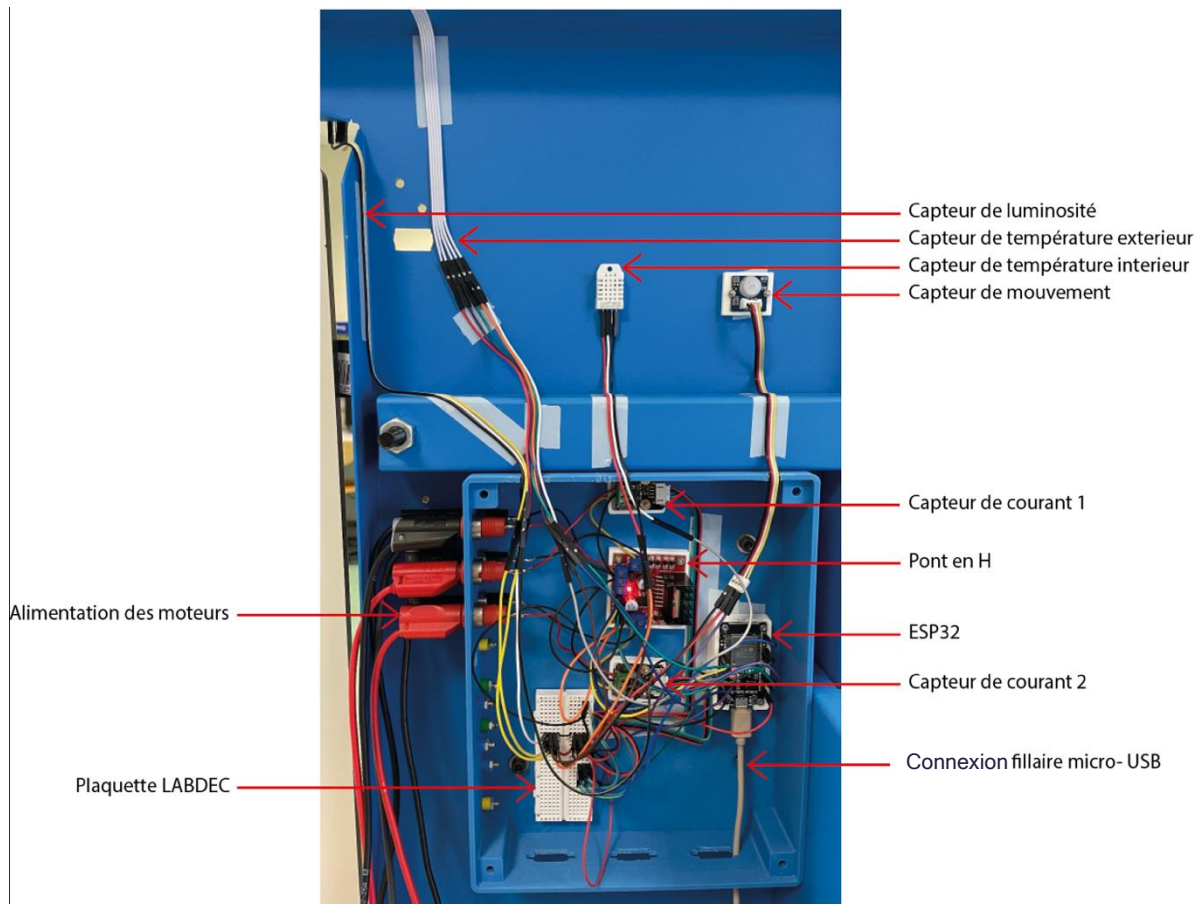


Figure 2 : Câblage finale sur la maquette

On notera que ce boîtier peut se couvrir afin d'avoir de ne pas avoir le montage à la vue de tous. Ainsi nous avons une entrée pour se connecter à l'ESP32 en filaire et les entrées de l'alimentation ainsi que des capteurs extérieurs.

• Problèmes rencontrés

Durant la mise en place du schéma à l'extérieur de la maquette tout s'est relativement bien passé, en revanche lorsque nous avons déplacé le circuit, nous avons rencontré des problèmes que nous avons dû résoudre.

- Certains supports étaient un peu petits dû à l'incertitude de l'impression il a donc fallu les limer pour y faire rentrer les capteurs.
- La boîte prévue pour le circuit ne nous permettait pas de faire sortir les fils des capteurs nous avons donc dû y créer des espaces pour y faire passer les fils.

- Les broches de l'ESP étaient vers le bas mais il nous fallait qu'elles soient dans l'autre sens pour y connecter les fils. Nous avons donc dessoudé toutes les broches et nous avons soudé les broches dont nous avons besoin vers le haut pour y faire nos connexions.

Lorsque tout a été mis en place nous avons donc un montage fixe fonctionnel et nous pouvions alors utiliser le montage complet en récupérant les données des capteurs pour commander les fermetures et ouvertures de volets en fonctionnement normal.

5. PARTIE PROGRAMMATION

- Site de maintenance

Introduction

Il nous semble important de trouver une solution pour communiquer avec le volet même en cas de disfonctionnement du serveur Raspberry. En effet, dans l'optique où on voudrait effectuer de la maintenance ou une installation du système en s'affranchissant de la Raspberry il faut tout de même que nous soyons capables de piloter le volet.

Fonctionnement :

La solution que nous avons décidé de mettre en place est un site qui communiquera directement avec l'ESP32. En effet, le principe sera que des actions seront associées à des liens qui, lorsqu'ils seront cliqués, activeront différentes actions qui leurs seront assignés.

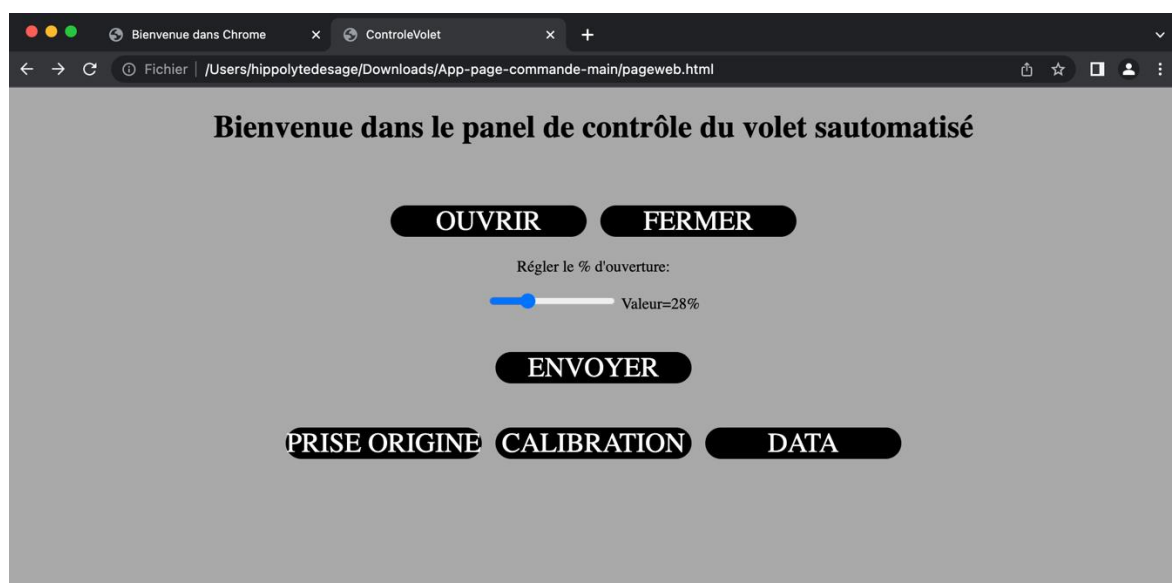


Figure 3 : Interface du panel de maintenance du volet

Par exemple, on choisit d'ouvrir le volet à 60%, on va alors régler le curseur à 60. Lorsqu'on cliquera sur le bouton "envoyer" (Qui est simplement un lien esthétiquement modifié avec le .css) le lien suivant sera donc actif :

[adresse/ouvrir?val=60](#)

L'adresse sera donc remplacée par l'adresse du serveur créé directement sur l'ESP, on lui indiquera donc une consigne d'ouverture et on lui transmettra la valeur de la variable "val". Ce fonctionnement est le même pour tous les autres liens. Nous pouvons aussi donner un exemple d'ouverture d'une des autres pages, par exemple la page DATA nous donnant la valeur des capteurs :



Figure 4 : Ouverture de la page correspondant au lien DATA

Explication du code correspondant :

Nous aurons donc une partie html et css (pour la mise en forme graphique), mais aussi une partie Java script qui va gérer le contenu mis à jour de façon dynamiquement. Nous en aurons besoin pour transmettre la valeur de la barre de progression.

```
<body>
  <h1>Bienvenue dans le panel de contrôle du volet automatisé</h1>

  <div id = "fond">

    <div id = "uno">

      <a href="#">OUVRIR</a>
      <a href="#">FERMER</a>

    </div>

    <p>Régler le % d'ouverture:</p>

    <input id="Range" type="range" step="1" value="0" min="0" max="100" onchange="AfficheRange()">
    <span id="AfficheValue">Valeur=0%</span>

    <script>
      function AfficheRange() {
        var pourcentage=document.getElementById("Range").value;
        document.getElementById("AfficheValue").innerHTML="Valeur="+pourcentage+"%";
        var envoyer = document.getElementById("envoyer");
        envoyer.href = "http://ADRESSE/ouvrir?val="+pourcentage;
      }
    </script>

    <div>
      <a id="envoyer" href ="http://ADRESSE/ouvrir?val=0">ENVOYER</a>
    </div>

    <div id = "scenario">
      <a href="#">SCENARIO 1</a>
      <a href="#">SCENARIO 2</a>
      <a href="#">SCENARIO 3</a>
    </div>

  </div>
</body>
```

Figure :5 Cade html et JavaScript de la page du panel de maintenance

Le programme ci-dessus est une page web HTML qui propose une interface utilisateur pour contrôler notre volet automatisé. La page comprend différents éléments tels que des liens permettant d'ouvrir ou de fermer le volet, un contrôle de saisie pour régler le pourcentage d'ouverture du volet et des boutons pour sélectionner différents scénarios. La partie centrale de la page (correspondant à l'encadré rouge du code) est constituée du contrôle de saisie qui permet de régler le pourcentage d'ouverture du volet. Lorsque la valeur du pourcentage est modifiée par l'utilisateur, une fonction JavaScript récupère cette valeur et passe par un lien pour l'envoyer au volet en passant par une adresse spécifique. En utilisant les liens "OUVRIR" et "FERMER", l'utilisateur peut déclencher l'ouverture ou la fermeture du volet. Les boutons "PRISE ORIGINE" et "CALIBRATION" permettent de réaliser les actions qui lui sont associés et la page "DATA" nous donne les valeurs des capteurs et les positions des battants.

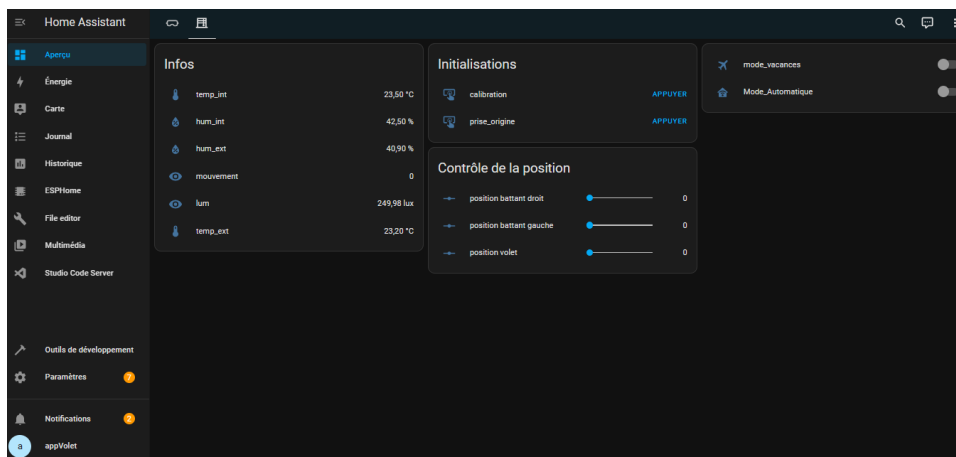
Le code HTML est structuré en différentes balises qui contiennent des informations sur la page, des liens vers des fichiers CSS pour la mise en forme de notre page. Globalement, ce programme offre une interface simple pour contrôler le volet automatisé en utilisant un contrôle de saisie et des liens pour des actions prédéfinies.

• Raspberry Pi : Utilisation de home assistant

Home Assistant est un logiciel d'automatisation et de gestion d'objets connectés. Il est gratuit, récent et populaire et est simple à mettre en œuvre. Il s'adapte également à tous les types de profil. C'est un logiciel open source et à héberger sur un réseau local. Il se couple parfaitement à un Raspberry Pi. Il possède en tout plus de 1300 composants lui permettant de se connecter à des objets et systèmes du marché. Il offre une interface utilisateur conviviale pour configurer et contrôler différents appareils, ainsi que pour créer des automatisations basées sur des règles ou des événements spécifiques.

Home Assistant prend en charge de nombreux protocoles de communication, tels que Zigbee, Z-Wave, MQTT, etc. Cela signifie qu'il peut communiquer avec une grande variété d'appareils, qu'ils soient connectés à Internet ou non.

Intégré à notre Raspberry Pi, nous pourrions donc établir une connexion à l'ESP32 via un point d'accès. Cette connexion nous permettra de recenser les données des capteurs et d'en envoyer en commande au besoin. L'interface de home assistant est similaire à l'image suivante :



L'avantage de cette méthode est de faciliter la connexion et la communication de plusieurs appareils connectés sur un point d'accès, étant donné que l'utilisation d'un serveur DNS est plutôt pratique sur l'OS de Home Assistant. Pour notre projet, les informations entre ces appareils et la Raspberry Pi se feront par le **protocole MQTT**. Notre Raspberry Pi pourra envoyer des commandes et recevoir des données reçues d'un capteur sur un topic. Chaque topic contient des informations précises et sont recensées dans un broker qui permet une distribution optimale de celle-ci.

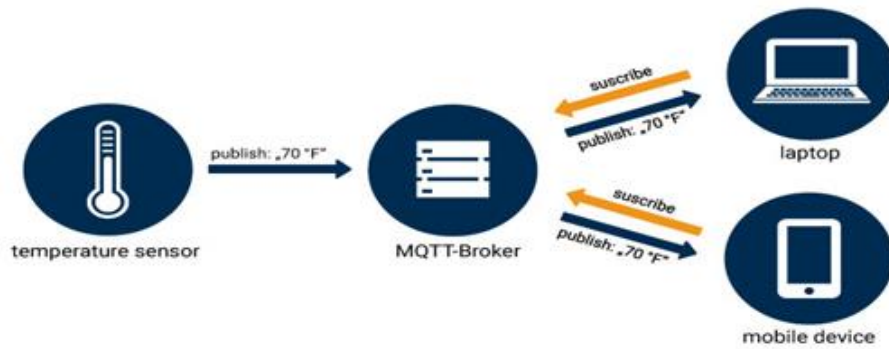


FIGURE 3 : ILLUSTRATION DU PROTOCOLE MQTT

Le protocole MQTT utilise un broker permettant de recenser l'ensemble des informations provenant d'un capteur pour les transmettre entre 2 parties : les subscribers et les publishers.

Par exemple si nous transmettons une information de température d'une ESP à une Raspberry, l'ESP est le publisher puisqu'il a l'information à la source et la Raspberry est le subscriber. Un autre avantage de Home Assistant est que ce dernier possède un broker facile à intégrer.

Pour notre projet, nous avons créé les topics suivants :

Pour récupérer les températures :

home/voletSalon/capteurs/temperatures/temp/(int ou ext)

Pour la luminosité :

home/voletSalon/capteurs/lum/ext

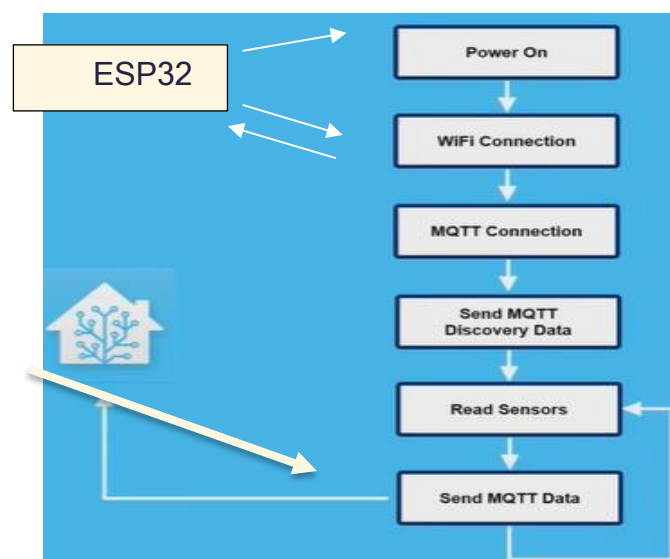
Pour la présence :

home/voletSalon/capteurs/mouv/pres/int

Pour l'humidité :

home/voletSalon/capteurs/hum/(int ou ext)

Ainsi grâce à ce protocole, on peut soit recevoir des données sur l'interface ou envoyer des commandes. D'un autre œil, on peut visualiser les connexions et les données de la manière suivante :

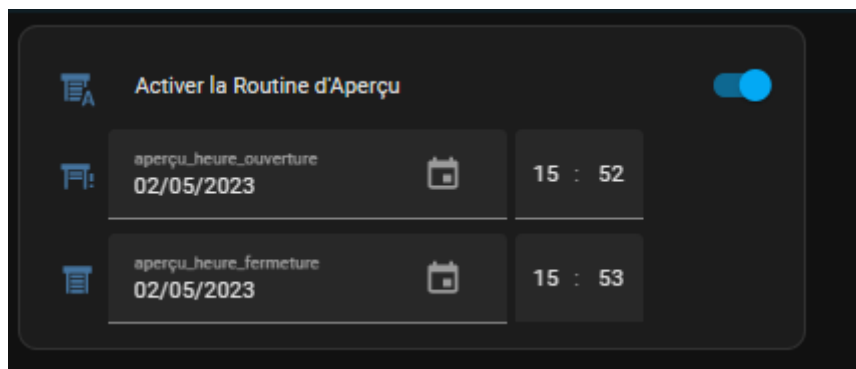
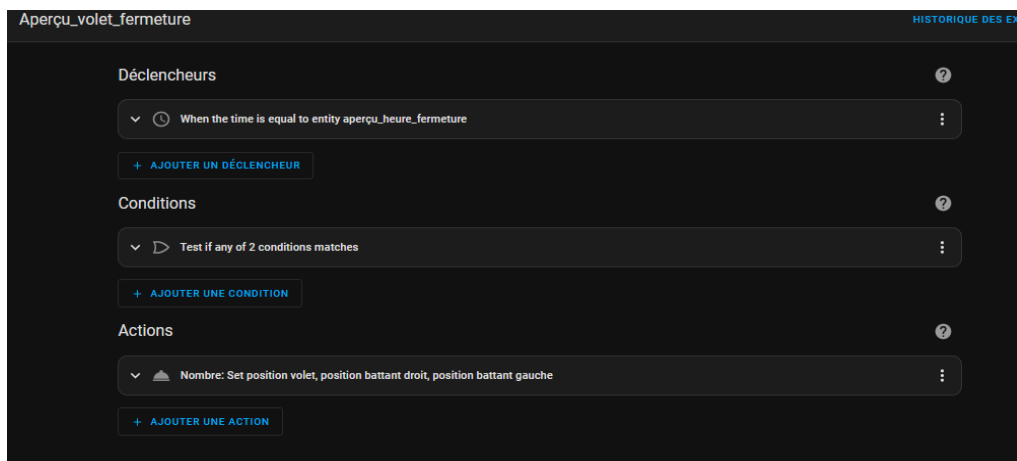


- **Automatisations dans home Assistant :**

Afin de garantir à l'utilisateur une expérience optimale, nous avons programmé des scénarios modifiables par l'utilisateur. Ceux-ci sont des modes automatiques qui ouvrent et ferment automatiquement les volets en fonction de la saison et des envies de l'utilisateur. Ces modes sont directement activables sur l'accueil de l'application mobile ou via le site internet.

Il existe deux manières de coder ces automatisations : une en YAML et une via l'interface utilisateur. Via l'interface utilisateur, il suffit d'entrer les différentes conditions nécessaires afin d'effectuer les actions prévues par le mode et définies par l'utilisateur.

Les conditions peuvent être définies à partir des déclencheurs, des états du volet ou des états de différents modes. Et quand toutes les conditions pour une action donnée seront validées, le mode effectuera celle-ci.



L'automatisation "Aperçu" présentée telle quelle nous permet de planifier n'importe quelle heure d'ouverture et de fermeture des volets. Cette automatisation a été implémentée pour prouver la faisabilité de nos automatisations. En bref, plusieurs automatisations sont possibles, la liste n'étant pas exhaustive. Toutefois, pour rendre nos automatisations de saison (hiver et été) efficaces, nous avons utilisé des paramètres telles que la température, la date, la présence et la luminosité pour ouvrir ou fermer nos volets. D'où l'utilité de nos capteurs en plus de fournir des informations à son utilisateur.

- **Programmation embarquée de l'ESP-32 :**

Introduction :

Pour pouvoir fonctionner, le volet est équipé d'un ESP-32. Il s'agit d'un microcontrôleur disposant d'entrées/sorties, d'une connexion wifi/Bluetooth, d'une connexion I2C. Ce microcontrôleur est compatible avec Arduino permettant ainsi une mise en service rapide ce qui est idéal dans notre cas pour la réalisation d'un prototype.

Ce microcontrôleur devra gérer les fonctions bas niveaux du volet de façon autonome :

- Détermination de la position des battants
 - Calibration
 - Prise d'origine
 - Traçage des déplacements
- Mise en position désirée avec système anticollision
- Connectivité avec le serveur (home Assistant)
- Accessibilité en mode maintenance
- Lecture des capteurs pour l'environnement extérieur

Toutes ces fonctions ont été implémentées en C/C++ (Arduino) avec une approche orientée objet. Nous avons divisé le code en 4 grandes classes :

- Volet : Fait le lien entre les deux battants et synchronise leurs fonctionnements tout en proposant une interface unique au reste du programme.
- Battant : Gère un battant avec la gestion de sa position et les sécurités.
- Capteurs : Une classe qui abstrait le fonctionnement des capteurs pour le reste du code.
- Mqtt_handler : Gère la communication MQTT avec le serveur home assistant.

La classe principale qui gère le programme dans son intégralité est substituée par les fonctions setup et loop fournit par Arduino.

Ci-dessous un diagramme de classe de notre programme. Certaines méthodes et variables ont été omises pour améliorer la lisibilité.

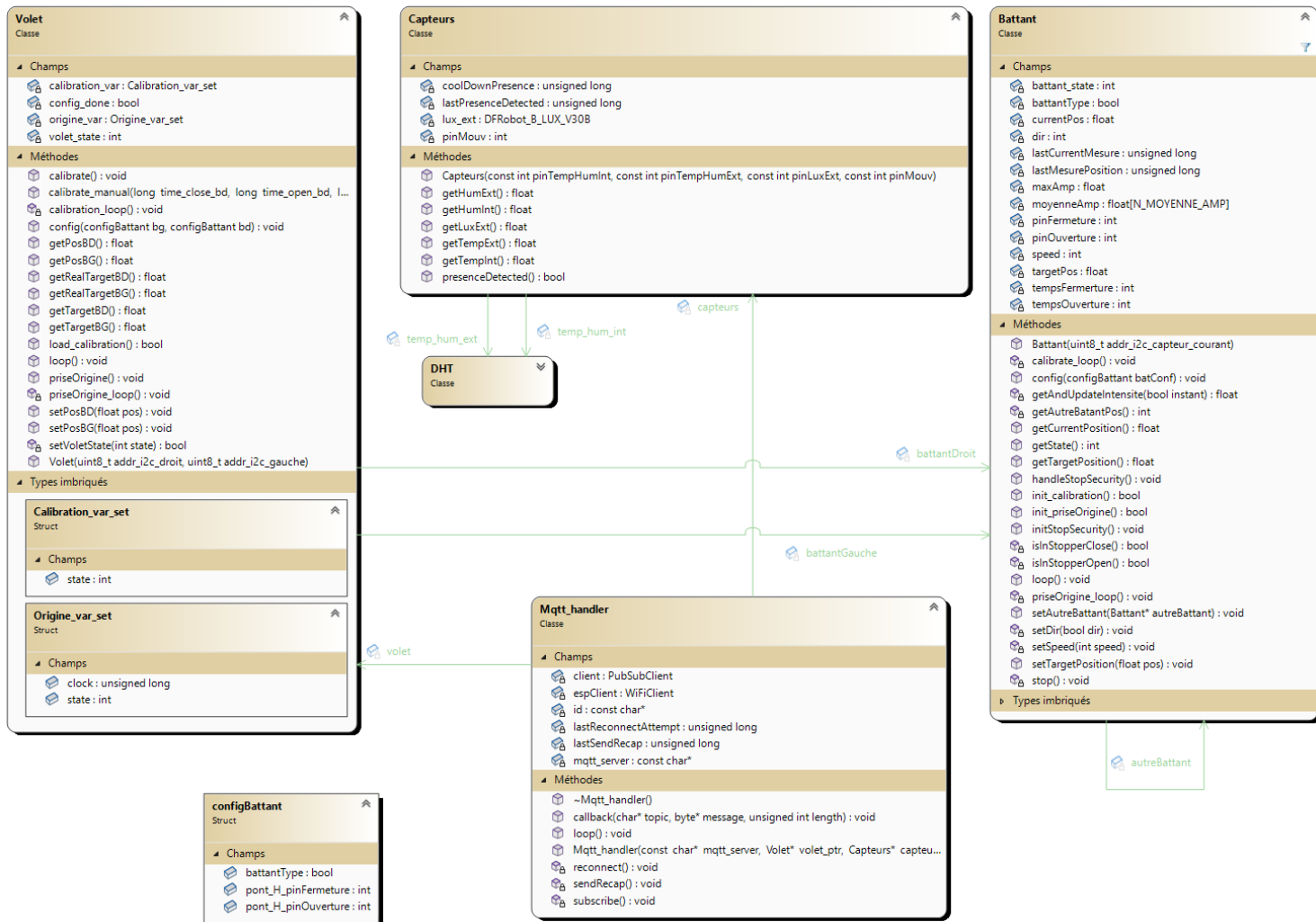


Figure 6 Diagramme de classe

Classe Battant :

La classe Battant est sans doute la plus importante du code embarqué dans le volet. Elle permet de réaliser toutes les actions utiles sur les battants tout en assurant la sécurité de son fonctionnement.

Détection de butée :

Pour la détection de butée, nous avons décidé de ne pas utiliser de capteur de fin de course classique mais plutôt de se baser sur la consommation des moteurs. En effet, lorsque le battant atteint sa position de butée (ouverte ou fermée), la consommation du moteur augmente significativement. Le code détecte cette augmentation de consommation et considère alors que le battant est en butée.

Estimation de la position :

L'estimation de la position du battant est un élément clé pour assurer un fonctionnement précis et sécurisé des volets. Cette estimation repose sur deux étapes importantes : la calibration et la prise d'origine.

1. Calibration :

La calibration est une étape cruciale pour déterminer les temps d'ouverture et de fermeture du battant. Elle est effectuée lors de la mise en service du volet ou lorsqu'une réinitialisation est nécessaire.

Au cours de la calibration, le battant est d'abord déplacé vers sa position ouverte maximale, puis vers sa position fermée maximale, en mesurant le temps nécessaire pour atteindre chaque butée. Le temps mesuré est ensuite enregistré et utilisé pour estimer la position du battant lors de son fonctionnement. Cette méthode permet d'obtenir une estimation précise du temps nécessaire pour ouvrir et fermer le battant.

2. Prise d'origine :

La prise d'origine permet de déterminer la position de départ du battant afin de s'assurer que l'estimation de la position est correcte tout au long de son mouvement. Elle est effectuée après la calibration et lors de chaque mise sous tension du volet.

Au cours de la prise d'origine, le battant est d'abord déplacé vers sa position fermée en utilisant la consommation du moteur pour détecter la butée lorsqu'il se ferme. Une fois la butée atteinte, la position actuelle du battant est considérée comme étant à 0% (position fermée). Cette position d'origine est alors utilisée comme point de référence pour les calculs de positions ultérieures.

La prise d'origine se fait également en continu « à la volée » lors de l'exécution du programme. Si l'utilisateur ferme ou ouvre le volet (à 0% ou 100%) le volet va aller jusqu'en butée et définir sa position comme étant 0 ou 100%.

3. Estimation de la position en temps réel :

Une fois la calibration et la prise d'origine effectuées, le code est capable d'estimer la position du battant en temps réel en fonction du temps écoulé, de la vitesse et de la direction du moteur.

Anti-chevauchement :

L'anti-chevauchement est une fonctionnalité importante pour assurer la sécurité de fonctionnement des volets. Elle permet d'éviter que les deux battants ne se chevauchent lors de leurs mouvements, ce qui pourrait endommager les volets ou causer des problèmes de fonctionnement. Pour mettre en œuvre l'anti-chevauchement, le code vérifie la position des deux battants et s'assure qu'ils ne sont pas trop proches l'un de l'autre. Si l'un des battants est dans une zone de chevauchement potentiel, le code ajuste la position cible de ce battant pour éviter le chevauchement. Cette vérification et cet ajustement sont réalisés en continu pendant le mouvement des battants.

En résumé, la classe Battant permet de gérer efficacement le mouvement des volets tout en assurant leur sécurité de fonctionnement. Elle utilise des méthodes intelligentes pour détecter les butées, estimer la position du battant et éviter le chevauchement, sans avoir besoin de capteurs supplémentaires.

Les autres classes :

Les autres classes sont moins complexes dans leurs fonctionnements. La classe volet permet de synchroniser les battants lors de la calibration et de la prise d'origine tout en fournissant une interface unique pour les deux battants.

La classe capteurs, s'occupe d'initialiser et de récupérer les valeurs des différents capteurs en fournissant une interface simple.

La classe Mqtt_handler s'occupe de maintenir la connexion ouverte avec le broker MQTT (dans la mesure du possible) et d'envoyer régulièrement les valeurs des capteurs mise à jour.

SETUP & LOOP :

Les fonctions setup et loop de notre programme gèrent la connexion au réseau wifi (avec la création d'un point d'accès wifi pour rétablir la connexion si le réseau habituel n'est pas disponible). C'est aussi dans ces fonctions que nous mettons en place le serveur web de maintenance. Le reste contient simplement l'appel au fonction loop des classes volet et Mqtt_handler.

6. CONCLUSION ET OUVERTURE SUR LE PROCHAIN SEMESTRE

Ce semestre aura donc été très productif et aura satisfait les objectifs que nous nous étions fixés au semestre précédent. En effet, nous avons pu répondre au cahier des charges que nous avons établis au semestre 7 qui lui était consacré. De plus, la finalité de notre projet est en accord avec les objectifs que nous nous étions fixés au début du semestre 6. Nous avons donc un projet finalement complet qui correspond aux envies premières du groupe. Nous aurions pu aussi nous écarter légèrement de notre sujet principal en intégrant un deuxième élément dans notre écosystème domotique comme une ampoule connecté par exemple. Cet ajout nous aurait permis de la faire communiquer avec notre volet et d'ainsi mieux mettre en valeur l'ajout de la Raspberry.

Nous avons en somme un volet automatisé, contrôlé via une application d'où nous pouvons consulter tous les paramètres des capteurs, régler l'ouverture des volets mais aussi actionner des scénarios. Nous avons aussi mis un place un site de maintenance qui inscrit d'autant plus notre projet dans une simulation complète de développement de produit. Toutes les fonctionnalités énoncées dans l'introductions ont été mises en place, ce qui prouve que nos séances ont été géré correctement et que le travail a été correctement répartie pour optimiser l'avancée du projet. Nous avons bien anticipé le nombre de séances car tous les objectifs, aussi ambitieux soient-ils, ont été terminés durant ou avant l'antépénultième séance. Cette légère avance nous aura permis d'avoir le temps d'ordonner nos idées afin d'expliquer au mieux comment nous avons atteint nos objectifs.

Concernant le semestre prochain, nous projetons de mettre en avant nôtre projet d'APP en développant d'avantage nôtre site internet en ajoutant des images, des vidéos ou encore un retour des valeurs mesurées par les capteurs. Ce site représentera un moyen de communication pour nous car il est important de communiquer efficacement pour mettre en avant nôtre produit.

Un autre élément important pour un produit technique comme celui-ci est la documentation technique. Rédiger les documents de mode d'emploi et de documentation techniques seront aussi une de nos missions pour le semestre 9.

Pour chaque entreprise, un produit créé est un produit destiné à être vendu. Dans cette logique, durant le dernier semestre nous réaliseront une étude de marché pour nôtre produit afin de comprendre les besoins du marché, évaluer la demande, analyser la concurrence et évaluer le potentiel de revenus.

Avec toutes ces étapes, nous arriverons sur un produit qui sera terminé et près à être positionné sur le marché.

7. APPORTS PERSONNELS

- **Hippolyte**

Ce semestre a été très enrichissant pour moi en termes de compétences liées à notre formation. Ce qui m'a le plus marqué, c'est l'acquisition de compétences en gestion de projet. En effet, lors de nos séances, nous avons dû planifier des échéances à moyen terme pour un projet qui s'étale sur trois années de formation. Travaillant en groupe composé de 5 personnes effectuant des tâches différentes, la communication et l'organisation ont également été des compétences clés à développer. En ce qui concerne les compétences techniques, j'ai pu me familiariser avec les composants électroniques grâce à la mise en place du circuit électrique. J'ai également pu développer mes compétences en Arduino grâce aux échanges que j'ai pu avoir avec le membre qui s'est chargé de cette partie du projet. Pour finir, j'ai pu en apprendre plus sur le fonctionnement de Home Assistant, je suis donc satisfait des bases solides en domotique que ce semestre d'APP m'a apporté.

- **Johan**

Durant toute la durée des APP du semestre 8, j'ai pu continuer de développer certaines compétences découvertes dans le passé et en développer de nouvelles. La première est bien entendue le travail en équipe, car c'est l'essence même d'un projet, d'avancer ensemble pour avancer plus vite et mieux. J'avais commencé à développer cela mais avec des équipes plus petites, il a donc fallu que je m'adapte et que je me perfectionne à ce niveau. La seconde a été l'organisation, car les APP durent longtemps, il a donc été important de rédiger des comptes-rendus de séance pour pouvoir garder une trace de notre travail. Durant ce semestre 8, j'ai aussi pu revoir des choses comme le codage de site internet ou encore le câblage et bien sûr découvrir de nouvelles choses comme Home Assistant. Dans une idée générale, les APP m'ont beaucoup plu et j'ai hâte de pouvoir continuer de développer notre maquette au semestre prochain.

- **Arnaud**

Lors de ce semestre, j'ai pu apprendre de nombreuses choses. J'ai pu me familiariser avec un nouveau langage, le YAML, en observant son fonctionnement mais aussi en codant dans celui-ci. J'ai compris comment marchait le protocole de communication MQTT en le mettant en œuvre avec mon groupe. Via ce protocole, mais aussi via nos comptes-rendus de séance, j'ai pu, comme tout au long de ce semestre, voir l'importance de la communication au sein d'un groupe. Enfin, j'ai pu voir aussi comment fonctionnait, mais aussi comment on pouvait utiliser home assistant en codant et me servant de ses différentes fonctions, un logiciel que je ne connaissais pas.

- **Zaïde:**

Grâce aux APP et aux problèmes rencontrés, j'ai appris à identifier les grandes lignes d'un projet de type objets connectés. Les aspects techniques tels que l'électronique, la mécanique, la programmation et le

réseau m'ont permis de mieux comprendre les cours vus tous au long de notre cursus. La découverte de nouveaux logiciels et de nouvelles techniques de programmation a été pour moi l'occasion de développer mon adaptabilité. En tant qu'aspirant ingénieur, il est important d'apprendre à apprendre étant donné la vitesse à laquelle sont déployées les innovations technologiques. De plus, j'ai eu l'opportunité de travailler au sein d'une équipe organisée et productive. La bienveillance et la volonté de chacun m'ont permis de rester motivé et optimiste. De belles choses ont été faites et cela montre que les qualités humaines sont à valoriser pour la réussite d'un projet.

- **Hervé :**

Ce semestre a été l'occasion de nous plonger dans la partie technique du projet en travaillant sur le développement logiciel et matériel. J'ai pour ma part pu réaliser l'intégration des différents capteurs et moteurs et toutes les fonctionnalités nécessaires à leurs bons fonctionnements sur l'esp32. Cela a aussi été l'occasion de travailler en groupe pour faire fonctionner ensemble les différents éléments du projet (Raspberry, Esp32 et site de maintenance).