## Rapport de projet :

### **PROJET KYUH**



### PARTIE PERSONNELLE

**Hugo ARBONA** 

# BTS SNIR Lycée Costebelle



# **SOMMAIRE**

### Tables des matières

lacktriangle	Sommaire	2		
•	Analyse des besoins			
•	Analyse de l'existant			
•	Semainier	5		
•	Fonctionnalité du système			
•	Contraintes	6		
•	Diagramme de cas d'utilisation	7		
•	Diagramme de séquence			
•	Diagramme de blocs			
•	Diagramme d'exigence			
•	Base de données			
•	Présentation du matériel			
•	Présentation des fonctions	14		
	STOCKAGE LOCAL DES DONNÉES	14		
	<ul> <li>TRANSMETTRE LES DONNÉES VERS LE SERVEUR</li> </ul>			
	DISTANT	17		
	<ul> <li>ALIMENTER LES DIFFÉRENTS MODULES DU</li> </ul>			
	SYSTÈME SUR SITE	21		
•	Conclusion	24		

### **ANALYSE DES BESOINS**

Dans le cadre d'élevage d'abeilles à miel menacées, sans cesse, par l'attaque de frelons, j'ai la mission de gérer la création et la configuration du réseau système ainsi que l'alimentation de tous les périphériques et composants servant à la défense de la ruche.

Afin de les protéger, le projet doit permettre de :

- Surveiller les abords immédiats des ruches afin de détecter la
- présence de frelons asiatiques.
- Mettre en place de manière automatique un bouclier anti intrusion temporaire à l'entrée des ruches .
- Rendre compte à l'apiculteur d'une éventuelle présence des frelons afin de pouvoir mettre en place une stratégie d'éloignement du prédateur, par exemple par le garnissage de bac à appâts plus éloigné du rucher.

L'ensemble des données collectées seront disponibles sur internet et pourront être visualisées à distance indifféremment à partir d'un navigateur internet classique ou à l'aide d'un smartphone.

D'autre part, en cas d'alerte critique et de manière journalière sous forme de récapitulatif de mesures, une notification sera envoyée à l'apiculteur sur son smartphone récapitulant l'état de chacune des ruches.

Cet équipement ne doit en aucun cas gêner les abeilles et l'apiculteur dans son travail, doit pouvoir être ajouté à une ruche classique facilement



### **ANALYSE DE L'EXISTANT**

Quelques dispositifs mécaniques anti intrusion existent.

Suivant l'expérience de certains apiculteurs, ces dispositifs ne donnent pas entièrement satisfaction.

Si leur installation est permanente, il semblerait qu'ils perturbent l'envol des abeilles et qu'ils ne soient pas totalement efficaces.

Si leur installation est temporaire, cela oblige l'apiculteur à de nombreuses manipulations .

La mise en place de bac à appât en lisière du rucher semble donner satisfaction mais cela oblige l'apiculteur à renouveler régulièrement les appâts constitués de matière périssable.

### **BOUCLIER ANTI-FRELONS STOP-IT®**



## MON SEMAINIER

Semaine 1 - 05 au 08 janvier	Analyse du dossier. Répartitions des tâches avec le reste du groupe.	
Semaine 2 - 11 au 15 janvier	Début de la réalisation des diagrammes de bloc et de séquence. Listing du matériel pour le groupe.	
Semaine 3 - 18 au 22 janvier	Fin des diagrammes de bloc et de séquence. Fin du cahier des charges. Préparations de la revue 1.	
Semaine 4 - 25 au 29 janvier	Fin de la revue 1. Début du devis.	
Semaine 5 - 01 au 05 février	Fin du devis. Installation et initialisation du raspberry Recherche de solution technique pour la base de données.	
Semaine 6 - 08 au 12 février	Installation et initialisation de la base de données.	
Semaine 7 - 15 au 19 février	Réalisation de la revue 2. Début de création du devis. Réalisation du rapport de conception.	
<b>Semaine 8</b> - 08 au 12 mars	Création d'un réseau privé Raspap	
<b>Semaine 9</b> - 15 au 19 mars	Installation d'un serveur FTP	
<b>Semaine 10 -</b> 22 au 26 mars	Mise en ligne sur le réseau privé de l'IHM.	
Semaine 11 - 29 mars au 02 avril	Installation de PHP sur le rasberry.	
<b>Semaine 12</b> - 06 au 09 avril	Édition des rapports de projet en groupe et individuel.	

### FONCTIONNALITÉ DU SYSTÈME

Nous proposons comme solution la mise en place d'une station d'appât à frelon avec surveillance caméra. Cette caméra permettra une détection des frelons asiatique et alertera l'apiculteur en cas de reconnaissance. Nous couplons à cela une motorisation d'un bouclier, vu dans l'état de l'art, posé sur les ruches qui se ferme en cas d'alerte, pour une durée de 15 minutes.

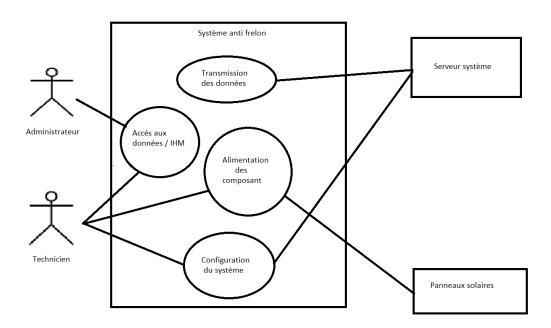
Afin d'alimenter le tout, nous utiliserons l'énergie solaire, grâce à des panneaux solaires.

### **CONTRAINTES**

L'utilisation de l'énergie solaire peut nous poser quelques problèmes. En effet, en cas de mauvais temps—nuageux,pluvieux—ni notre apport énergétique ni notre système de défense ne pourront fonctionner.



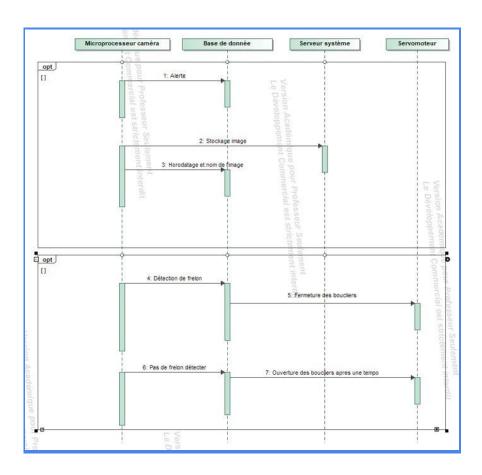
### **DIAGRAMME DE CAS DE D'UTILISATION**



Dans mon diagramme de cas d'utilisation, on voit deux acteurs l'Administrateur et le Technicien. Qui interviennent sur différents cas d'utilisation :

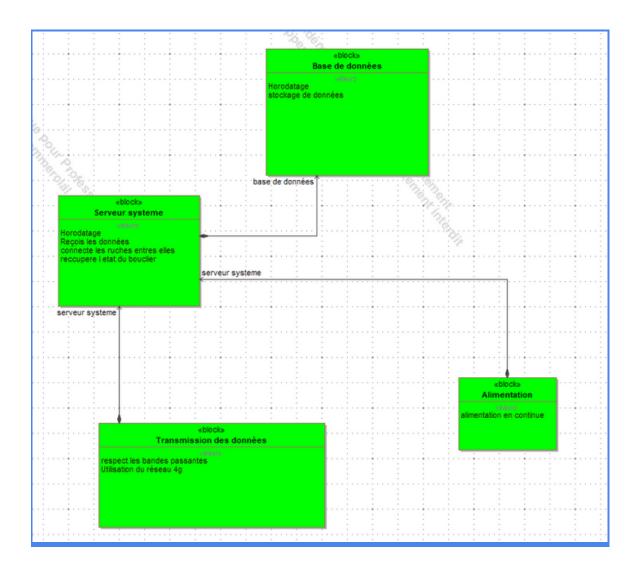
- Le technicien lui configure le système pour permettre un accès facile au futur Administrateur à l'IHM, configure le réseau pour que les transmissions des données puissent être transmises au serveur système et s'occupe de l'alimentation de l'ensemble des appareils.
- L'administrateur lui peut accéder à l'IHM afin de voir les données.

### DIAGRAMME DE SÉQUENCE



Dans mon diagramme de séquence, il y a quatre lignes de vie qui montrent les actions du microprocesseur avec la base de données, le serveur système et les servomoteurs. Lorsque la caméra détecte un frelon, le microprocesseur envoie une alerte à la base de donnée qui directement suit par la réception de l'image ainsi que l'horodatage de celle-ci. L'état des servomoteurs dépend de la présence d'alerte ou non.

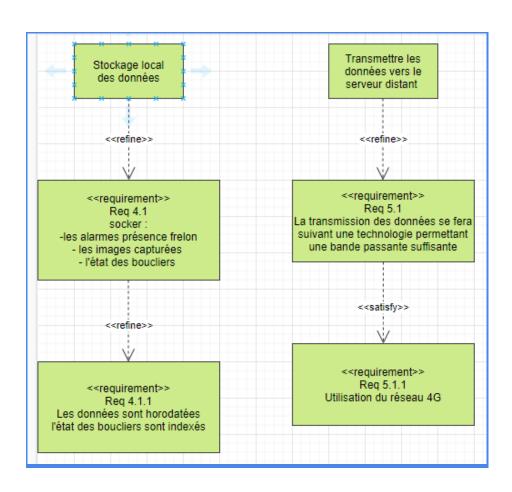
### **DIAGRAMME DE BLOCS**



Dans mon diagramme de blocs, on peut voir quatres blocs : Base de données, Alimentation, Serveur système et Transmission des données.

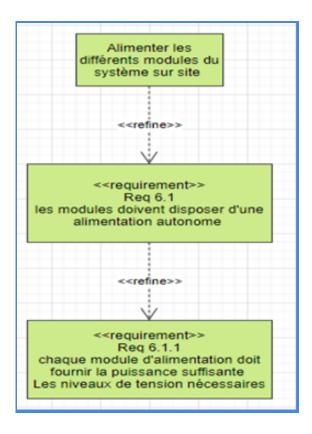
- Le Serveur système est relié à tous les autres blocs, il a comme fonction de recevoir les données reçues de la caméra ainsi que de les horodater pour ensuite les envoyer aux servomoteurs des ruches.
- La Base de données reliés au serveur système, permet le stockage des données reçues par le serveur système.
- La Transmission des données finalement en WIFI, permet l'échange de données entre les différents composants.
- L'Alimentation permet de faire fonctionner tous les composants en continu.

### **DIAGRAMME D'EXIGENCE**



Dans mon diagramme de classe, on peut voir trois classes: Stockage local des données, Transmettre les données vers le serveur distant et Alimenter les différents modules du système sur site.

- Le Stockage local des données doit être capable de stocker l'état de l'alarme, les images capturées ainsi que l'état des boucliers.
- La Transmission des données vers le serveur distant doit être capable de transmettre les données entre les composants grâce à la technologie WIFI.

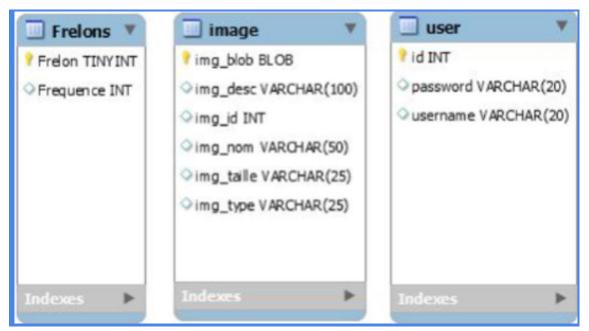


 L'Alimentation des différents modules du système sur site doit être capable d'alimenter tous les modules en continu même dans les pires conditions. KYUH Projet BTS SNIR - Dossier technique

### **BASE DE DONNÉES**

Solution	MySQL	PostgreSQL	SAP HANA	Microsoft SQL Server
Langue Codage	C,C++, Eiffel,Java,Perl,PHP, Python et TCL, API MySQL	C,C++, Java,PHP,Python Et TCL API MySQL	SQLScript, JavaScript, Python,Go,et Java	SQL C,C++,lava
Taille max BDD	illimité	Illimité	illimité	illimité
Prix	Open Source	Open Source	Open Source	8998 licence 2019
Solution	MySQL	PostgreSQL	SAP HANA	Microsoft SQL Server
Langue Codage	ARLLY	Anna	***	44
Taille max BDD	*****	*****	*****	*****
Prix	Arter	****	****	*

Pour le choix de la base de données j'ai choisi la base de données MySQL, car c'est avec celle-ci que j'ai toujours travaillé et elle dispose de plus de langue compatible que les autres base de données.



Modélisation de la base de données

La base de données est composée de 3 tables : Frelons, Image et User

- Dans la table Frelons, il y figure le nombre de frelons rencontrés ainsi que leur fréquence d'apparition.
- Dans la table Image, il y figure les paramètres permettant le stockage des images initialement prévu en FTP.
- Dans la table User, il y figure les noms d'utilisateur et les mots de passe permettant d'accéder à l'IHM.

### PRÉSENTATION DU MATÉRIEL

Solution	Raspberry pi 3 B+	Raspberry Pi 4 B
Prix	20.59 €	37,95€
USB	2x USB3.0 2x USB2.0 USB-C OTG	4x USB2.0
Serial	RX/TX UART	
GPU	VideoCore IV	VideoCore VI
Power sources	microUSB, GPIO	USB-C
Power ratings	1,13 A @5V	1,25 A @5V

Pour le choix du serveur système, nous avons choisi le Raspberry Pi 3 B+ pour son prix nettement intéressant mais aussi pour le fait qu'il dispose du port série RX/TX .

Solution	Panneau solaire SOL10P
Prix	36,50 €
Puissance	10 W
Intensité	610 mA
Dimensions	356 x 253 x 30 mm

Pour l'alimentation nous avons choisi de partir sur de l'énergie solaire, étant une énergie renouvelable et après l'achat des panneaux solaire totalement gratuite.

Nous disposions déjà de 3 mini panneaux solaires permettant l'alimentation des micro processeur ainsi que de la caméra.

Pour le serveur système nous avions besoin d'un panneau solaire plus puissant, nous avons donc choisi le panneau solaire SOL10P pour sa puissance ainsi que son prix.

### PRÉSENTATION DES FONCTIONS

• STOCKAGE LOCAL DES DONNÉES

Initialisation du Raspberry pour convenir à tous les besoins :

1. Apache

**Installation:** 

sudo apt update
sudo apt upgrade
sudo apt update
sudo apt install apache2

**Modification des droits:** 

sudo chown -R pi:www-data /var/www/html/ sudo chmod -R 770 /var/www/html/

Pour vérifier si Apache est bien installé et fonctionne, il suffit d'accéder à <a href="http://l27.0.0.1">http://localhost</a> via le raspberry, si "It works!" apparaît c'est que l'installation s'est bien passée.

2. <u>PHP</u>

**Installation:** 

sudo apt install php php-mbstring

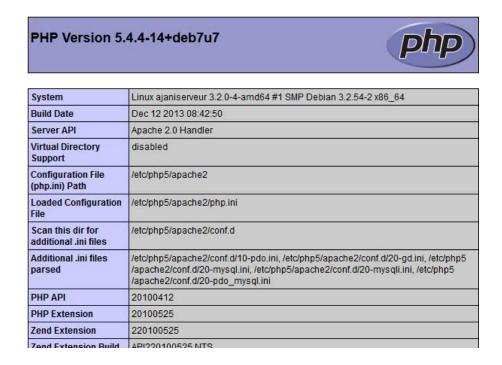
Pour savoir si PHP fonctionne correctement, il faut dans un premier lieu supprimer le fichier « index.html » dans le répertoire « /var/www/html ».

sudo rm /var/www/html/index.html

Puis créez un fichier « index.php » dans ce répertoire, avec cette ligne de commande :

#### echo "<?php phpinfo(); ?>" > /var/www/html/index.php

Ensuite il suffit d'accéder à <a href="http://localhost">http://localhost</a> et l'index est censé contenir cette image :



#### 3. MYSQL

#### **Installation:**

#### sudo apt install mariadb-server php-mysql

Pour vérifier le fonctionnement de MySQL, cette fois **nous utiliserons uniquement la ligne de commande.** Pour cela, nous allons simplement nous connecter via la commande :

#### sudo mysql --user=root

Nous allons maintenant **supprimer l'utilisateur root et créer un nouvel utilisateur root**, car celui par défaut n'est utilisable que par le compte administrateur du système, et n'est donc pas accessible aux script PHP du serveur.

Pour cela, une fois connecté à MySQL, lancez les commandes suivantes (remplacez password par le mot de passe de votre choix) :

DROP USER 'root'@'localhost';

CREATE USER 'root'@'localhost' IDENTIFIED BY 'password';

GRANT ALL PRIVILEGES ON \*.\* TO 'root'@'localhost' WITH GRANT OPTION

Pour les prochaines connexions, il suffira de faire :

mysql --user=root --password=votremotdepasse

- 4. PHPMyAdmin
- Installation:

sudo apt-get install phpmyadmin

PHPMyAdmin nous pose plusieurs questions concernant ses réglages.

Comme la base de données est déjà configurée, il faut choisir "no" à la question concernant l'utilisation de dbconfig-common.

Puis sélectionner d'utiliser PHPMyAdmin pour un serveur Apache.

Pour le mot de passe root, il s'agit de celui que nous avons utilisé pour MySQL.

Il faut aussi activer l'extension mysqli si cela n'est pas encore fait. Pour cela, utilisez les commandes ci-dessous.

sudo phpenmod mysqli sudo /etc/init.d/apache2 restart

Pour vérifier le bon fonctionnement de PHPMyAdmin, il faut tenter d'y accéder, en utilisant l'adresse de votre Raspberry suivi de /phpmyadmin. Par exemple, en local ce sera « http://127.0.0.1/phpmyadmin ».

- Le rendu visible est :



Le raspberry est maintenant prêt à accueillir un site web et une base de données.

Exécuter

- TRANSMETTRE LES DONNÉES VERS LE SERVEUR DISTANT
- Installation d'un hotspot wifi avec RaspAp:

Pour cela comme précédemment il faut mettre à jour le raspberry Pi

sudo apt-get update sudo apt-get upgrade

Par mesure de sécurité nous allons réaliser une copie de sauvegarde du fichier de configuration original du WiFi wpa\_supplicant.conf, puis le supprimer :

sudo cp /etc/wpa\_supplicant/wpa\_supplicant.conf /etc/wpa\_supplicant/wpa\_supplicant.conf.org sudo rm /etc/wpa\_supplicant/wpa\_supplicant.conf - Installation de RaspAp :

#### wget -q https://git.io/voEUQ -O /tmp/raspap && bash /tmp/raspap

Le Raspberry Pi va afficher RaspAP et nous proposes des dossiers pour l'installation des programmes **RaspAP** et **Lighttpd**. A moins d'avoir de bonnes raisons de le contredire, il faut répondre "Yes".

Après avoir redémarrer le Raspberry Pi, RaspAP a configuré hostapd. L'adresse IP WiFi de notre AP (Access Point = Point d'accès) est 10.3.141.0.

C'est à cette adresse que nous accèderons à l'interface wlan0 du Raspberry Pi configuré en AP.

RaspAP a également configuré le DHCP pour qu'il distribue des adresses entre 10.3.141.50 et 10.3.141.255.

Lorsqu'un appareil (PC, Raspberry Pi, smartphone, tablette etc.) se connectera sur cet AP, le DHCP lui distribuera une adresse IP comprise entre ces deux valeurs.

Pour paramétrer RaspAP, on accède à l'interface d'administration via une page web, en se connectant sur l'adresse de l'AP, c'est-à-dire 10.3.141.1. On peut le faire soit en local sur le Raspberry Pi AP lui-même, soit depuis une autre machine connectée à l'AP.

A à la fin de l'installation, après le redémarrage , le réseau sans fil sera configuré comme point d'accès avec les paramètres par défaut suivants :

• Adresse IP: 10.3.141.1/24 et 169.254.218.244/16

• Nom d'utilisateur gestion : admin

• Mot de passe gestion : secret

• Plage DHCP: 10.3.141.50 à 10.3.141.255

• SSID : raspi-webgui

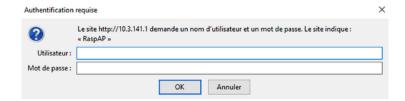
• Mot de passe du SSID : ChangeMe

Il est vivement conseillé de changer le mot de passe du SSID.

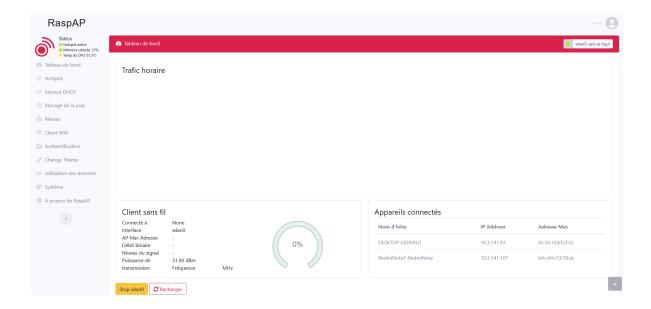
Le réseau est maintenant créé et disponible via les paramètres vue au dessus.

Mais on peut modifier tous ses paramètres en allant sur : <a href="http://10.3.141.1">http://10.3.141.1</a>.

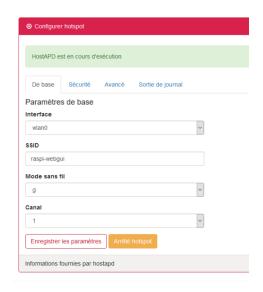
#### Une connexion est nécessaire, RaspAP demande les identifiants Utilisateur : admin / Mot de passe : secret.



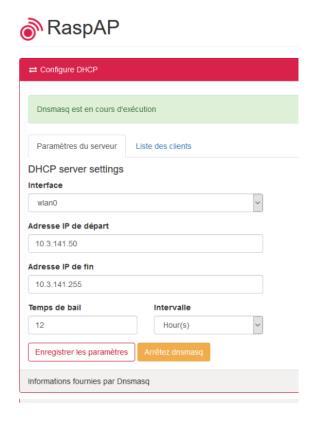
#### On accède au tableau de bord de l'interface de gestion de raspAP:



#### Dans "Hotspot" vous pourrez modifier les paramètres de l'AP (SSID, mode, canal).



Dans l'écran "Serveur DHCP" vous pourrez modifier la plage d'adresse du DHCP ainsi que la durée des baux.



Et voilà le réseau hotspot est créé et configuré.

Maintenant pour montrer que la communication dans ce réseau privé est possible, voici un ping entre mon ordinateur personnelle et le raspberry sur ce même réseau :

```
Envoi d'une requête 'Ping' 10.3.141.107 avec 32 octets de données :
Réponse de 10.3.141.107 : octets=32 temps=362 ms TTL=64
Réponse de 10.3.141.107 : octets=32 temps=132 ms TTL=64
Réponse de 10.3.141.107 : octets=32 temps=259 ms TTL=64
Réponse de 10.3.141.107 : octets=32 temps=36 ms TTL=64

Statistiques Ping pour 10.3.141.107:
Paquets : envoyés = 4, reçus = 4, perdus = 0 (perte 0%),
Durée approximative des boucles en millisecondes :
Minimum = 36ms, Maximum = 362ms, Moyenne = 197ms
```

#### • ALIMENTER LES DIFFÉRENTS MODULES DU SYSTÈME SUR SITE

#### 1. Station d'appât

La station d'appât est composée de la caméra, deux panneaux solaires et d'une batterie.

#### Caméra (OpenCV H7 Plus):

- Vin: 3,6 Vcc à 5 Vcc
- Toutes les broches d'E/S fournissent une sortie 3,3 V et une tolérance de 5 V
- Pas de tolérance 5V sur P6 en mode ADC ou DAC
- 25 mA max. sur les sorties
- Jusqu'à 120 mA max. (toutes sorties cumulées)
- Consommation (en veille): 140 mA @ 3,3V (avec ou sans μSD)
- Consommation (active): 230 mA @ 3,3V (sans μSD) et 240 mA @ 3,3V (avec μSD)
- Dimensions: 45 x 36 x 29 mm
- Poids: 17g

#### Panneau solaire:

- Tension: 5.5 V Dc/540 mA
- Connecteur: 2 broches type JST
- Tension maxi en circuit ouvert: 8,2 V
- Silicium mono-cristallin.
- Dimensions: 160 x 138 x 3 mm

#### **Batterie**:

- Tension: 3,7 Vcc
- Intensité: 4000 mAh

Ce module se régule tout seul grâce à l'énergie solaire qui charge pendant la journée puis utilise la batterie pendant la nuit.

#### 2. Serveur système:

Le serveur système composé du Raspberry Pi est alimenté aussi grâce à un panneau solaire et une batterie, beaucoup plus puissantes nécessitant un module chargeur solaire.

#### Raspberry PI 3 B+:

- Alimenté en 5V / 2.5A
- Consommation estimée en moyenne à ~500mA.

On a donc pour le raspberry: 2.5W x (24h x 7j)  $\Longrightarrow$  [420 Wh / 84 Ah] pour 168 h.

#### Panneau solaire:

• Tension: 17,6 Vcc

• Tension à vide: 21,8 Vcc

Puissance: 10 WIntensité: 610 mA

• Nombre de cellules: 36 (4 x 9)

Température de service: -40 à 85 °C
Dimensions: 356 x 253 x 30 mm

• Poids: 1,16 kg

#### **Module chargeur solaire:**

• Tension panneau solaire: 15 à 25 Vcc

• Puissance panneau solaire: 100 W

• Tension batterie: 12 Vcc

• Consommation au repos: < 4 mA

• Sorties:

- 12 Vcc/8 A sur borniers à vis

- 5 Vcc/ 5 A sur bornier à vis

- 2 x 5 Vcc/2,5 A sur fiche USB A femelle

• Courant de charge maxi: 4 A (via panneau solaire)

• Dimensions: 68 x 68 mm

#### **Batterie:**

Tension: 12 VccCapacité: 12,0 Ah

• Dimensions: 151 x 98 x 94 mm

• Poids: 3,95 kg

Comme le module précédent le système se régule tout seul grâce à l'énergie solaire et va stocker le surplus dans la batterie pour la nuit ou un temps couvert.

#### 3. Ruche:

La ruche est équipée de deux moteurs aussi alimenter via des panneaux solaires et d'accumulateur permettant le stockage du surplus.

#### **Servomoteur FS5109M**:

- Alimentation: 4.8 à 6Vcc
- Course: 2 x 90°
- Couple:
- 9 kg.cm à 4,8 Vcc
- 10,2 kg.cm à 6 Vcc
- Vitesse:
- - 0,18 sec / 60° à 4,8 Vcc
- - 0,16 sec / 60° à 6 Vcc
- Longueur du cordon JR: 300 mm
- Dimensions: 41 x 20 x 38 mm

#### Panneau solaire:

- Tension: 5.5 V Dc/540 mA
- Connecteur: 2 broches type JST
- Tension maxi en circuit ouvert: 8,2 V
- Silicium mono-cristallin.
- Dimensions: 160 x 138 x 3 mm

•

#### **Accumulateur:**

• Pas les données

### **CONCLUSION**

Ce projet m'a appris comment travailler et s'entraider dans le cadre d'une équipe avec un objectif commun. Cela m'a également appris le fonctionnement d'un réseau privé ainsi que sa création complète ce qui pourra certainement m'être utile dans les années à venir dans mon projet professionnel, notamment au vu du développement des projets mobilisant un réseau privé. Ce fut l'occasion d'être confronté à différentes difficultés. J'ai dû notamment faire face à la technologie WIFI inconnu pour moi jusqu'à présent. Mais cette difficulté m'a permis de comprendre les enjeux et le poids d'avoir une partie importante dans une équipe et très dépendante pour les coéquipiers.

