

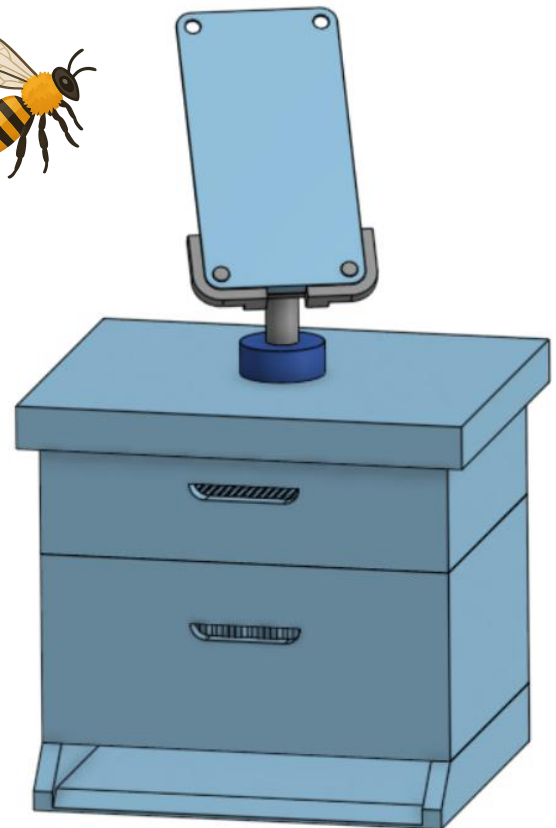
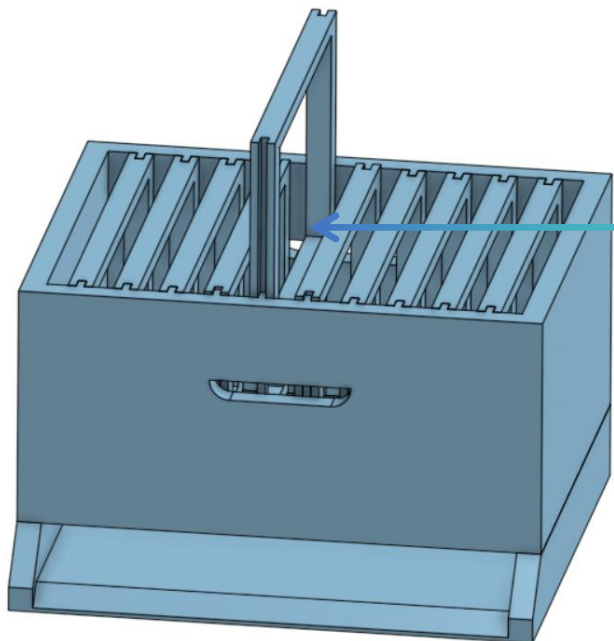
## RAPPORT DE SÉANCE 1 :



Durant la **première séance** nous avons pris connaissance du sujet et nous avons testé si le panneau solaire fonctionnait avec un **voltmètre/ampèremètre**. Puis, j'ai pu réaliser la ruche en 3D en prenant les dimensions standard d'une ruche de type **Langstroth** (518 x 423 x 240 mm pour le corps) avec le logiciel **OnShape**.



Il faudra reprendre les dimensions réelles de la **ruche** et du **panneau solaire** afin de créer plus précisément le **support du panneau solaire** ainsi que le **support de la PCB dans le bac de végétaline**.



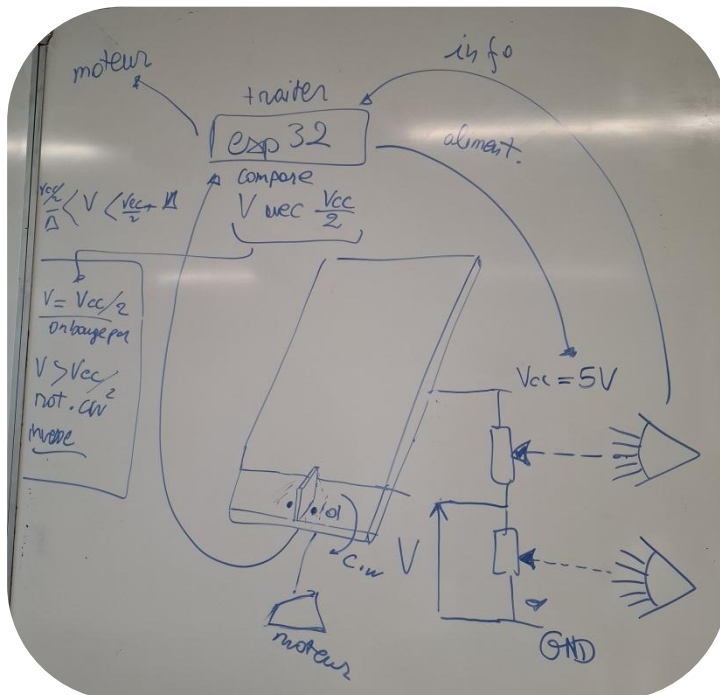
Le **bac de végétaline** se situe ici. C'est l'équivalent d'un **cadre de ruche** mais rempli de **végétaline** (liquide lorsque celle-ci est chaude).

Ensuite, nous avons réfléchi sur le **suivi du soleil**. En effet, le panneau solaire est **orientable** à l'aide d'un **moteur pas à pas commandé par un ESP32** à l'**extérieur de la ruche**. Ce système permet ainsi d'**alimenter le dispositif** de manière **autonome** tout en **optimisant l'ensoleillement du panneau**.

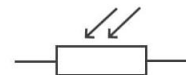
Sous le panneau, nous avons placé **deux photo-résistances** permettant de détecter la **position du soleil**. Ces capteurs forment un **pont diviseur de tension** alimenté en **5V**. La tension aux bornes d'une des **photo-résistances** est **mesurée par l'ESP32**, puis **comparée à la tension théorique  $V_{cc}/2$**  correspondant à un **éclairage équilibré** entre les deux capteurs.

Si la tension mesurée est **inférieure à  $V_{cc}/2$** , cela signifie que le **capteur est moins éclairé** : le **moteur tourne** dans un sens pour **réaligner le panneau**.

Inversement, si la tension est **supérieure à  $V_{cc}/2$** , le **moteur tourne** dans l'autre sens jusqu'à ce que les deux capteurs reçoivent la **même quantité de lumière**.



Représentation symbolique



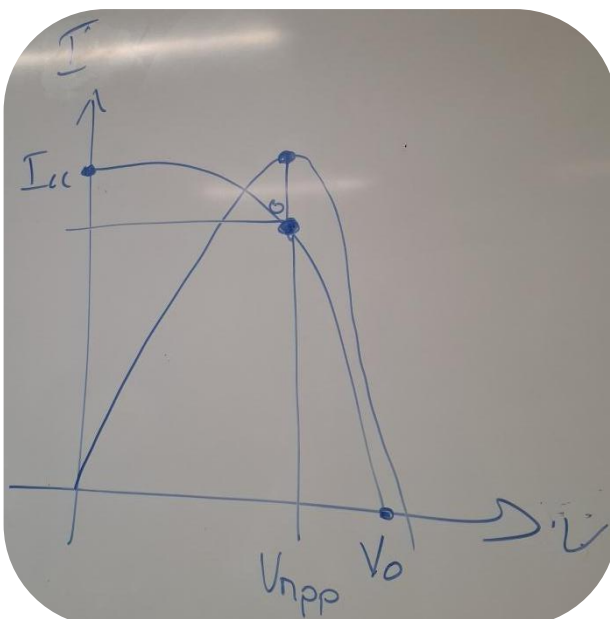
ou



Présentation physique



Exemple d'une photorésistance, modèle GL 5528



Nous observons que le **courant** est **maximal** en **court-circuit** ( $I_{cc}$ ) et **nul** en **circuit ouvert** ( $V_o$ ).

Entre ces deux points se trouve le **point de puissance maximale** (MPP), pour lequel le produit  $U \times I$  est maximal.

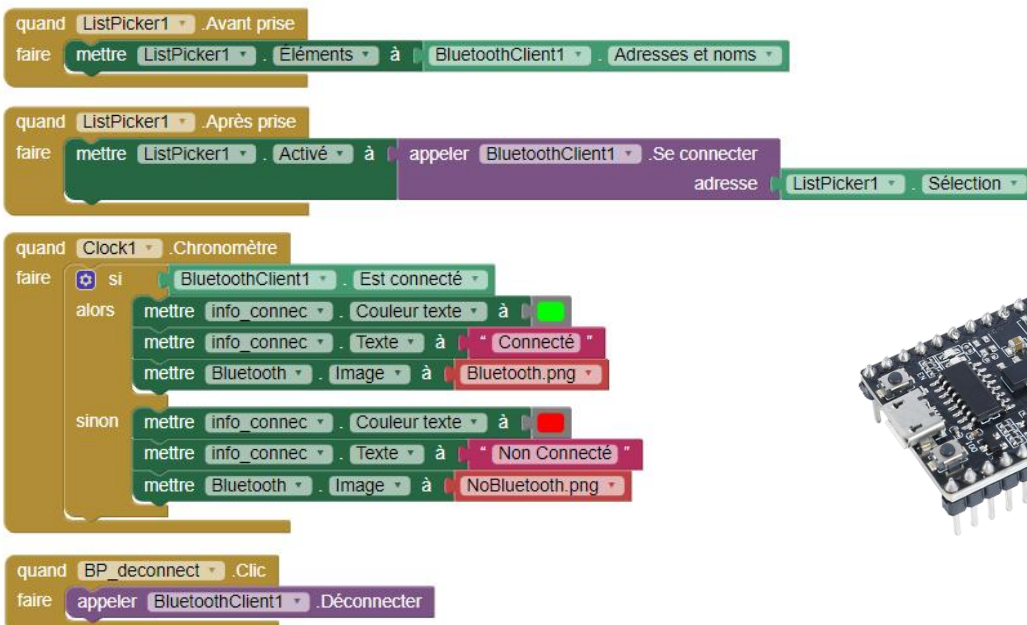
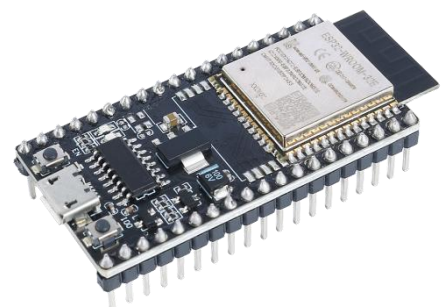
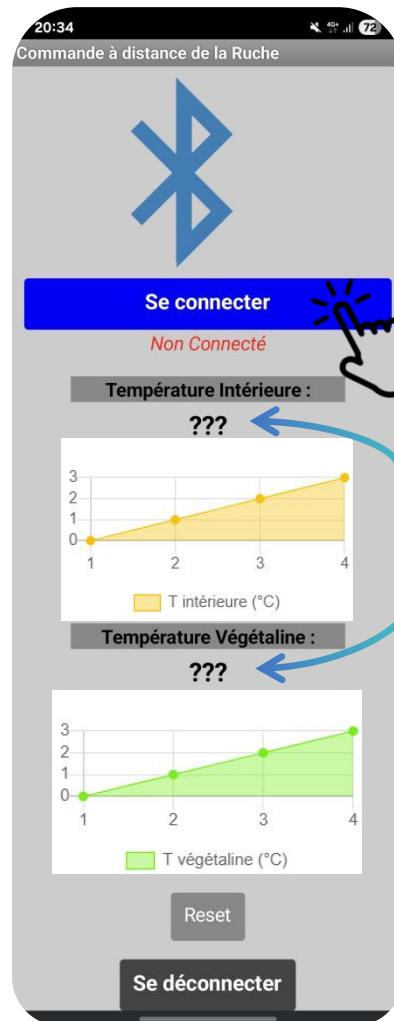
C'est à ce point que le panneau délivre sa **meilleure efficacité énergétique**.



Enfin, nous avons commencé le développement de l'application qui servira à **contrôler la température** à distance par liaison **Bluetooth**.

Nous avons mis **2 graphiques** afin d'afficher l'évolution de la température dans la **ruche** et dans la **végétaline**.

Il y a aussi la **partie « Blocs Scratch »** qui sert à la **connexion Bluetooth** de l'ESP32 dans la ruche avec le smartphone.



Ici, lorsque le **bouton Déconnecter** est cliqué, l'application se déconnecte de l'ESP32.



On peut utiliser ce bloc pour **recupérer le message envoyé par l'ESP32** qui contient les **données de température moyenne de la ruche** et de la **végétaline** sous forme d'une chaîne de caractères (string).



Lorsque l'on a reçu le **message**, celui-ci est **sous la forme « DATA;T\_ruche°C;T\_végétaline°C »**.  
 Donc on récupère seulement les messages commençant par « **DATA** » (pour montrer que la réception des données peut débuter).  
 Puis, on **place les éléments du message** dans une **liste de string** (ici, il y a donc **3 éléments séparés** par « **;** »).  
 Enfin, on peut s'occuper de **l'affichage de la température sur l'application** à la place des « **???** » ( \* ) et aussi sur les **graphiques** mais la suite au prochain épisode sur le prochain rapport ;) ...