

Traqueur Solaire

Projet d'Innovation Technologique (IT)

G. Quentin (Modélisation et Soudure)

B. Nathan (Câblage et Assemblage)

R. Antoine (Codage et Diapositive)

Traqueur solaire

Un **traqueur solaire** ou suiveur de Soleil est une installation de **production d'énergie** utilisant le principe de **l'héliostat**.

C'est une structure portante qui permet **d'orienter des panneaux solaires** afin d'en **augmenter la productivité**



Image non contractuel

A qui rend-t-il service ?

Utilisateur

Sur quoi agit-il ?

Le panneau solaire

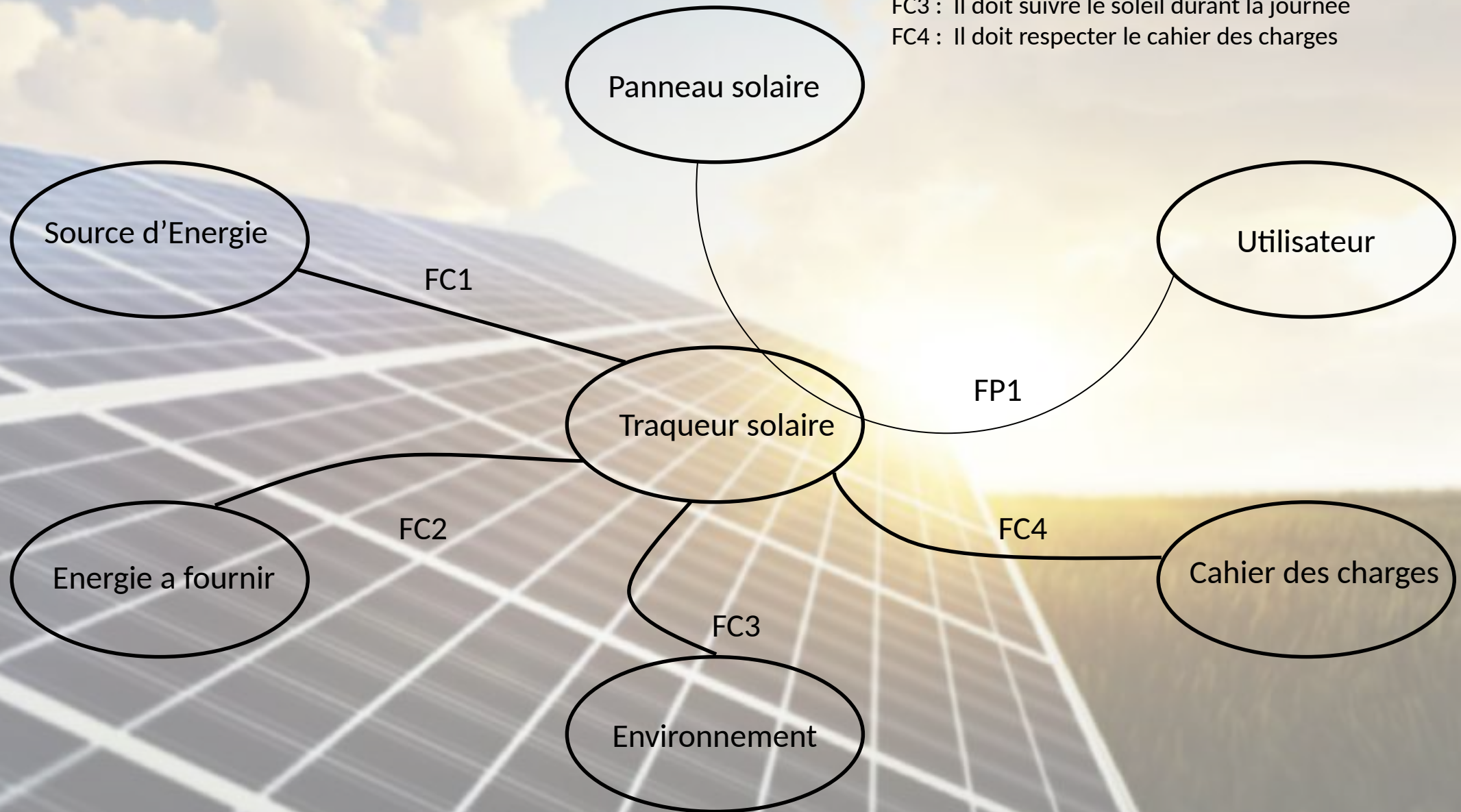
Produit

Traqueur Solaire

Pour quoi faire ?

Orienter le panneau solaire et rentabiliser la production d'énergie.

FP1 : Rentabiliser son utilisation
FC1 : Il nécessite 6V pour fonctionner
FC2 : Il doit fournir assez d'énergie pour alimenter un téléphone
FC3 : Il doit suivre le soleil durant la journée
FC4 : Il doit respecter le cahier des charges



Cahier des Charges

Objectif du projet :

L'objectif du projet est de réaliser un traqueur solaire permettant de suivre le soleil tout au long de la journée afin d'optimiser l'énergie reçue par le panneau photovoltaïque.
Il doit être automatisé et alimenter un téléphone.

Fonctionnement

- Panneau solaire est branché a une batterie pour faire tourner des servomoteurs.

- Tourne dans l'axe x et y grâce aux servomoteurs.

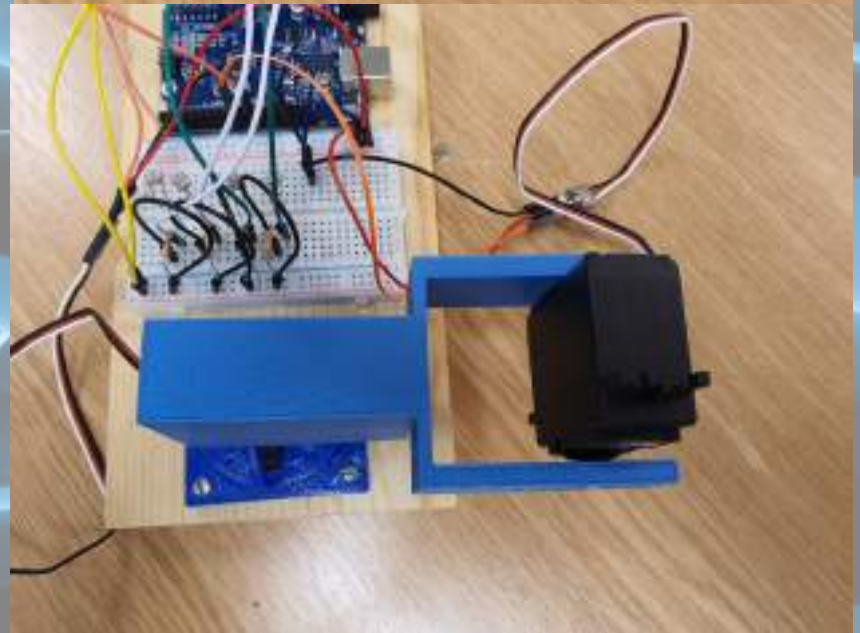
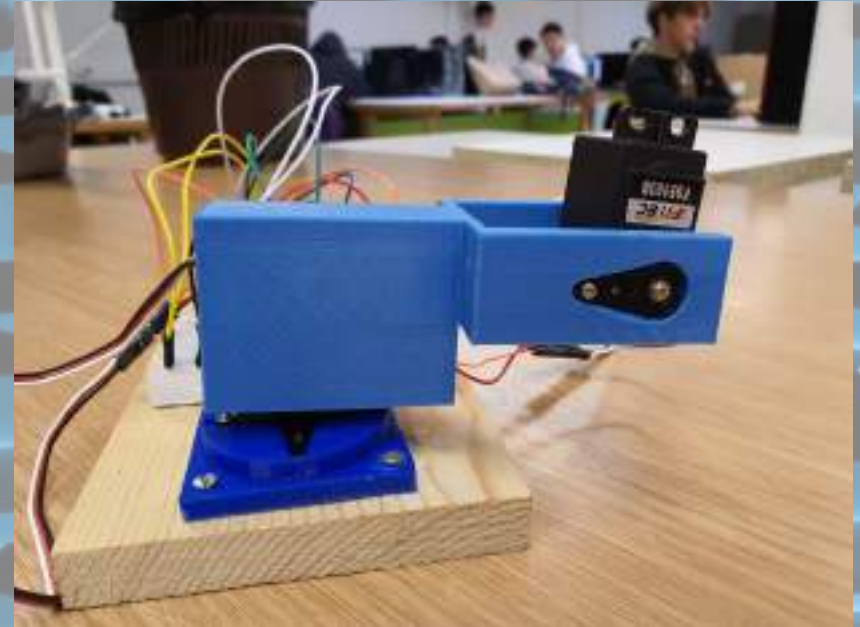
- Tourne grâce a l'aide de photorésistances.



Modélisation 3D

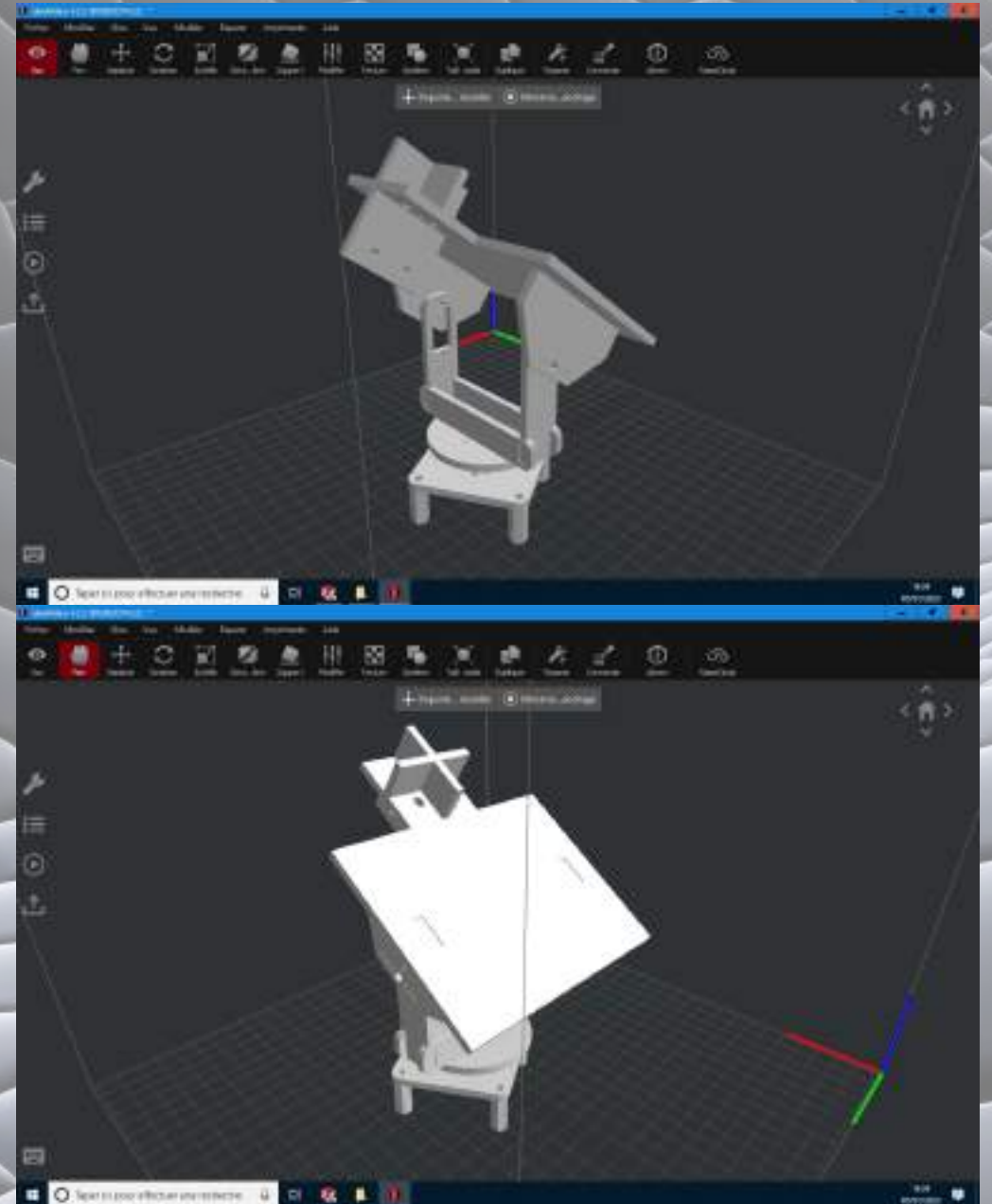
Projet de base

Modèle 3D Test



Modélisation 3D mi-parcours

Projet de base fini → Modèle 3D mi-parcours ▲



Modélisation 3D mi-parcours

Projet de base fini

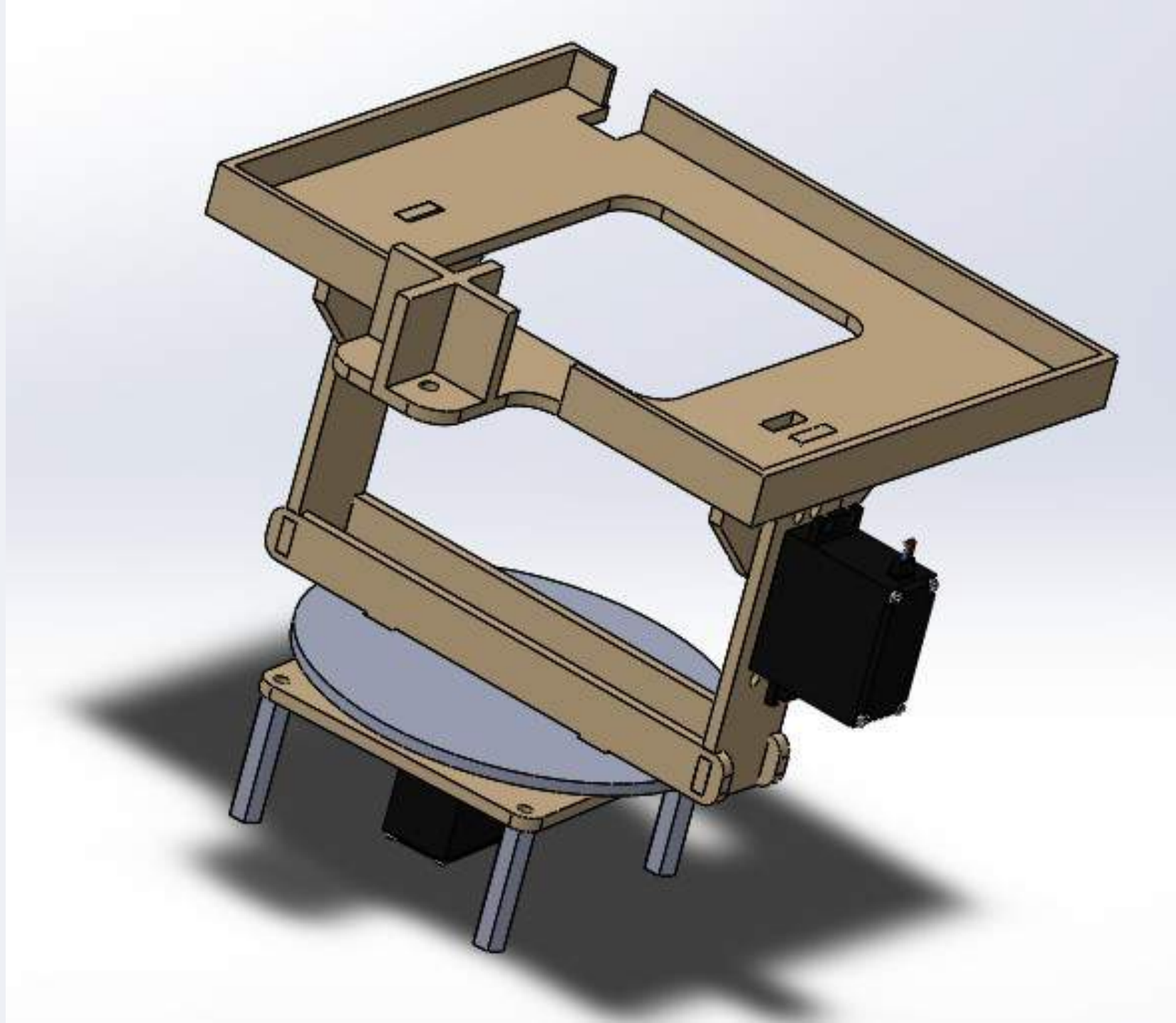


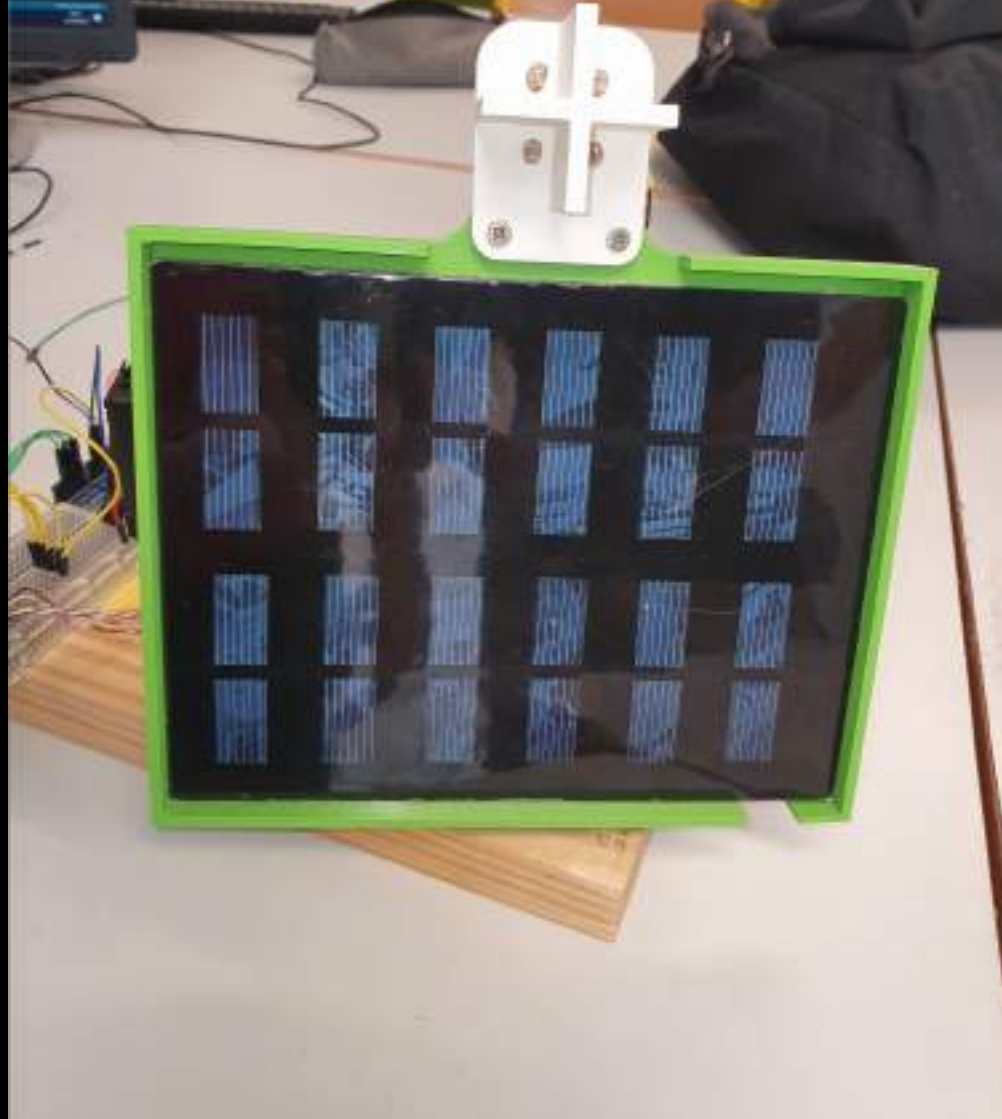
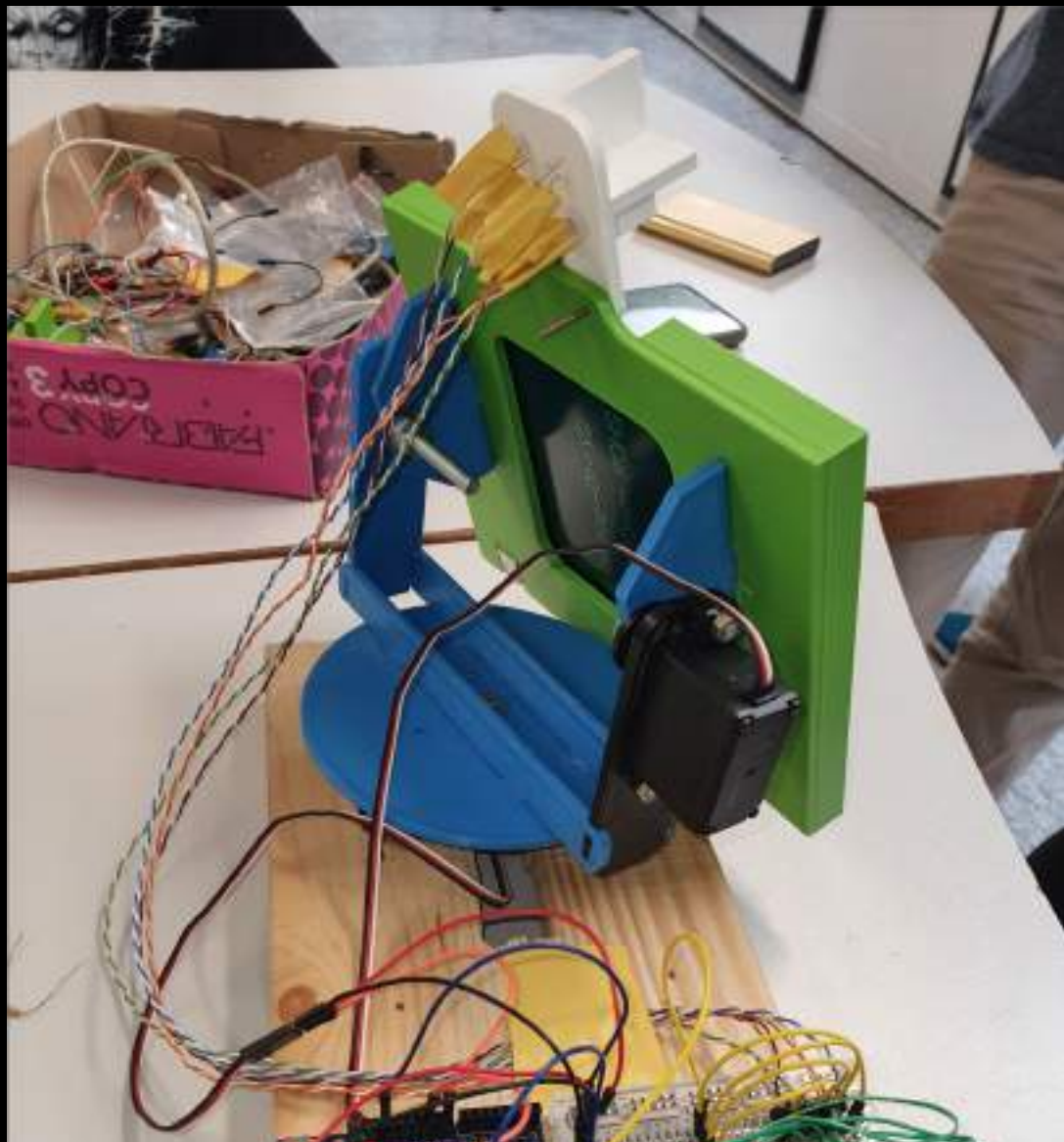
Modèle 3D mi-parcours



Modélisation 3D Finale

- Rebord
- Plateau
- Viseur





Programming

Essai

```
digitalWrite(moteurH, HIGH);  
delay(100);  
  
digitalWrite(moteurV, HIGH);  
delay(100);  
digitalWrite(moteurH, LOW);  
delay(100);  
digitalWrite(moteurV, LOW);  
delay(100);  
}
```

```
void setup() {  
  Serial.begin(115200);  
}  
void loop() {  
  Serial.println(analogRead(A0));  
  delay(1000);  
}
```

```
int moteurH = 8;  
int moteurV = 10;  
int val;  
void setup() {  
  pinMode(moteurH,A0);  
  pinMode(moteurV,A3);  
  
}  
void loop() {  
  val = analogRead(0);  
  val = map(val, 0, 1023 , 0, 180);  
  moteurH.write(val=180);  
  delay(15)  
}
```

Programme Avant

```
void setup() {
  ServoH.attach(pinServoH); // la commande du servomoteur est reliée à son connecteur
  ServoV.attach(pinServoV); // la commande du servomoteur est reliée à son connecteur
}
void loop() {
  ServoH(5); // parallele au sol
  delay(2000); // nouvelle attente
  ServoV(5); // Est
  delay(2000); // nouvelle attente
  ServoH(45); // 45° par rapport au sol
  delay(2000); // nouvelle attente
  ServoV(5); // Est
  delay(2000); // nouvelle attente
  ServoH(90); // perpendiculaire au sol
  delay(2000); // nouvelle attente
  ServoV(175); // ouest
  delay(4000); // nouvelle attente
  ServoH(45); // 45° par rapport au sol
  delay(2000); // nouvelle attente
  ServoV(175); // ouest
  delay(2000); // nouvelle attente
  ServoH(5); // 45° par rapport au sol
  delay(2000); // nouvelle attente
}
```

Programme Final

```
#include <Servo.h>

int moteurV=10;
int moteurH=8;
Servo ServoV;
Servo ServoH;
int angleV=90;
int angleH=90;

void setup() {
    ServoV.attach(moteurV);
    ServoH.attach(moteurH);
    Serial.begin(115200);
    ServoV.write(angleV);
    ServoH.write(angleH);
}
```

```
void loop() {
    // Mouvement vertical
    int v1 = analogRead(1);
    int v2 = analogRead(0);
    if(v1 > v2) { // ou v1 < v2
        angleV=angleV+5;
        if (angleV>180) { angleV=180; }
        ServoV.write(angleV);
    }
    if(v1 < v2) {
        angleV=angleV-5;
        if (angleV<0) { angleV=0; }
        ServoV.write(angleV);
    }

    // Mouvement Horizontal
    int v3 = analogRead(3);
    int v4 = analogRead(2);
    if(v3 > v4) { // ou v3 < v4
        angleH=angleH+5;
        if (angleH>180) { angleH=180; }
        ServoH.write(angleH);
    }
    if(v3 < v4) {
        angleH=angleH-5;
        if (angleH<0) { angleH=0; }
        ServoH.write(angleH);
    }

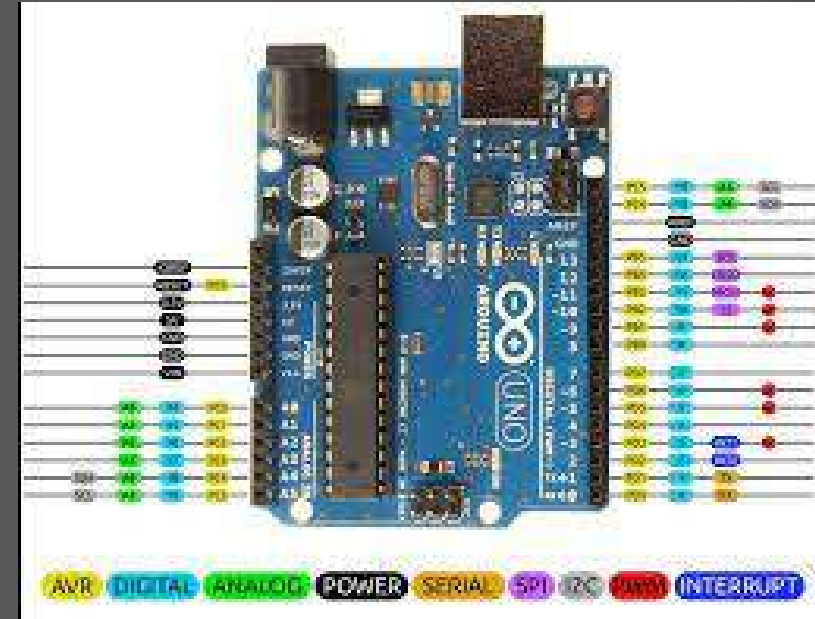
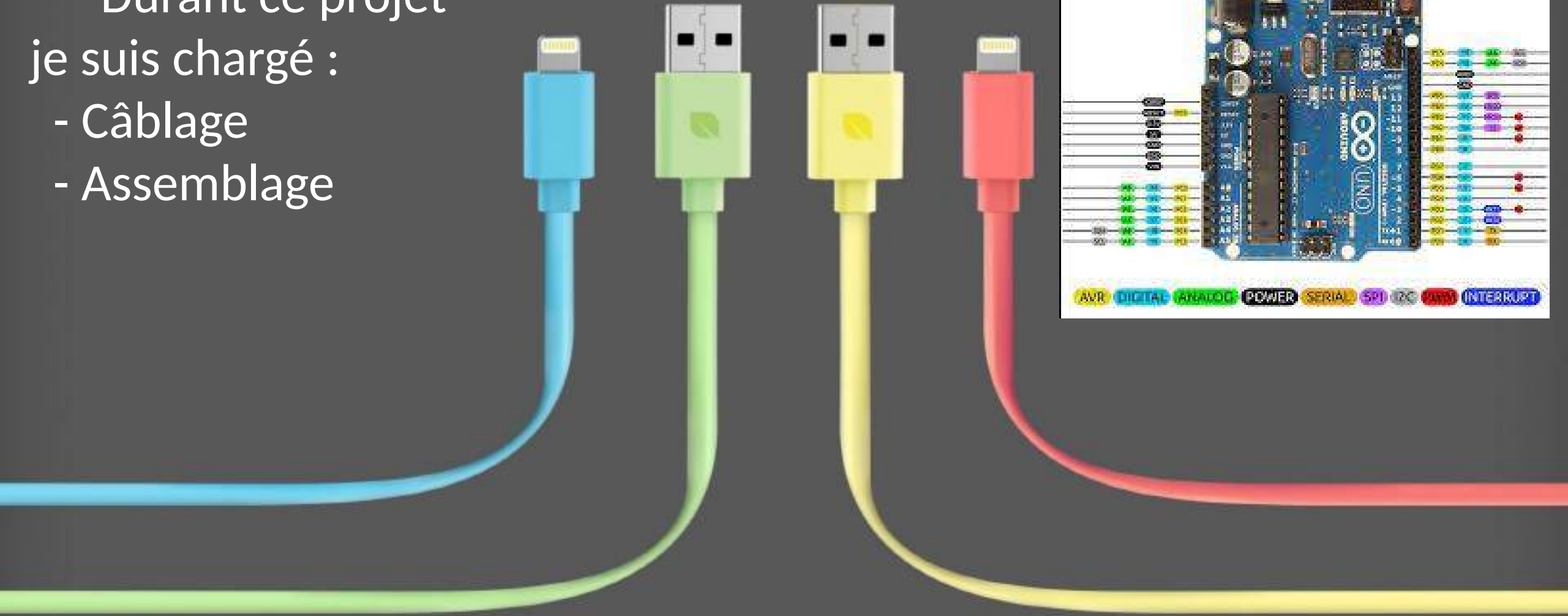
    delay(500);
}
```

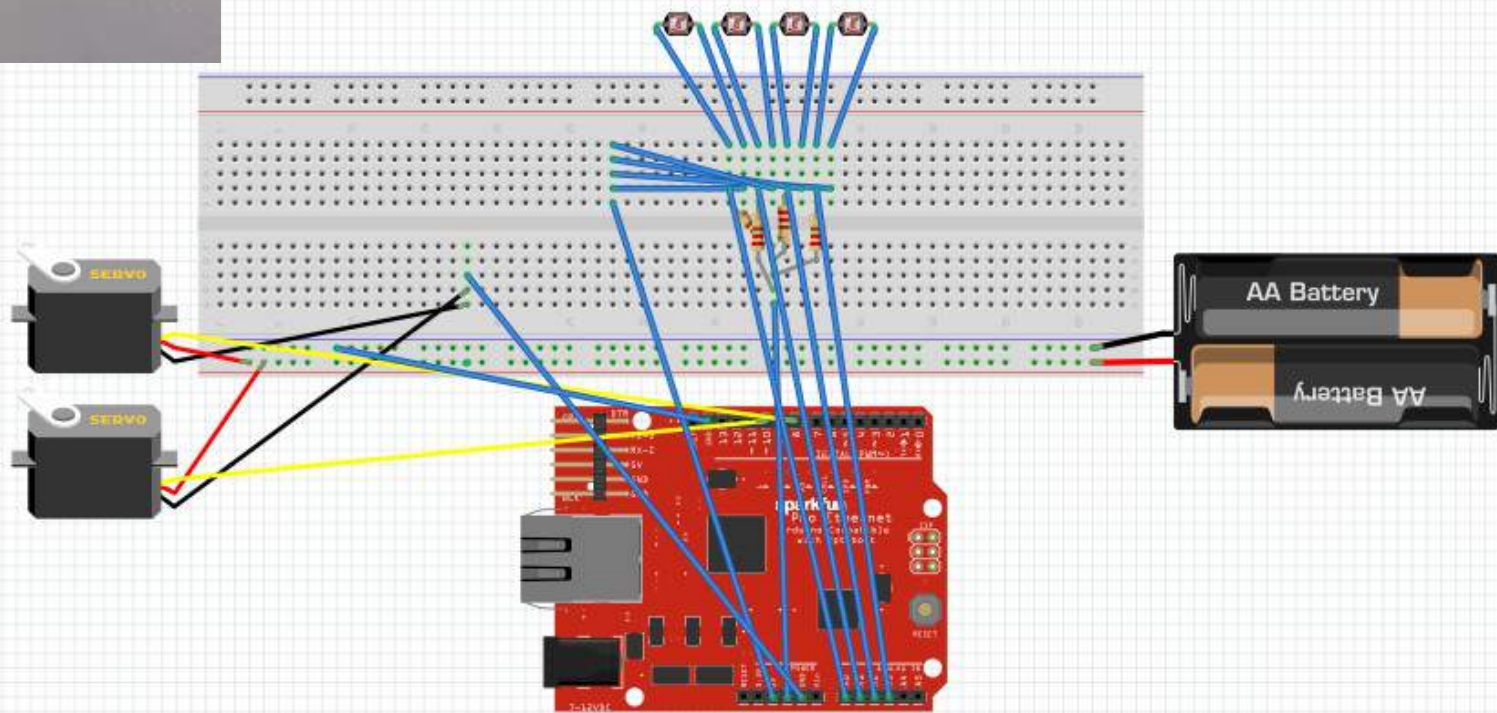
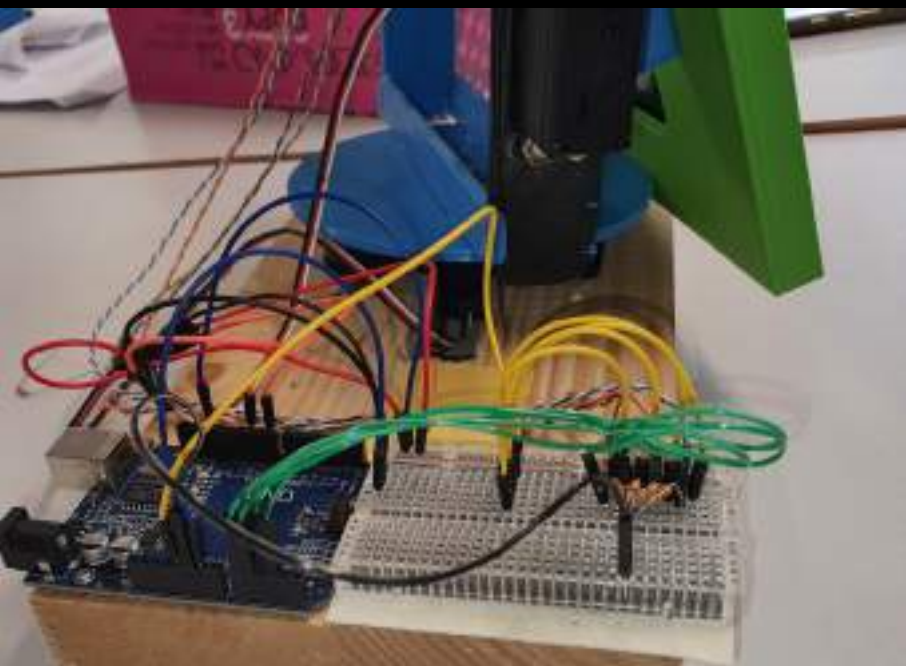
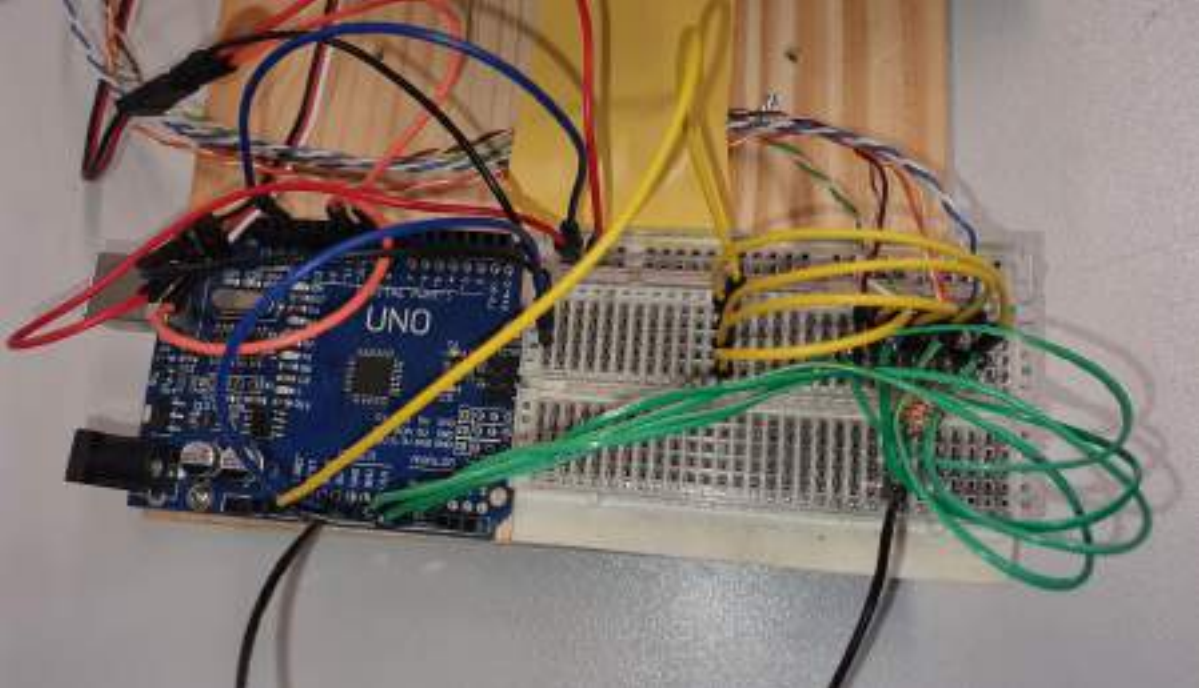


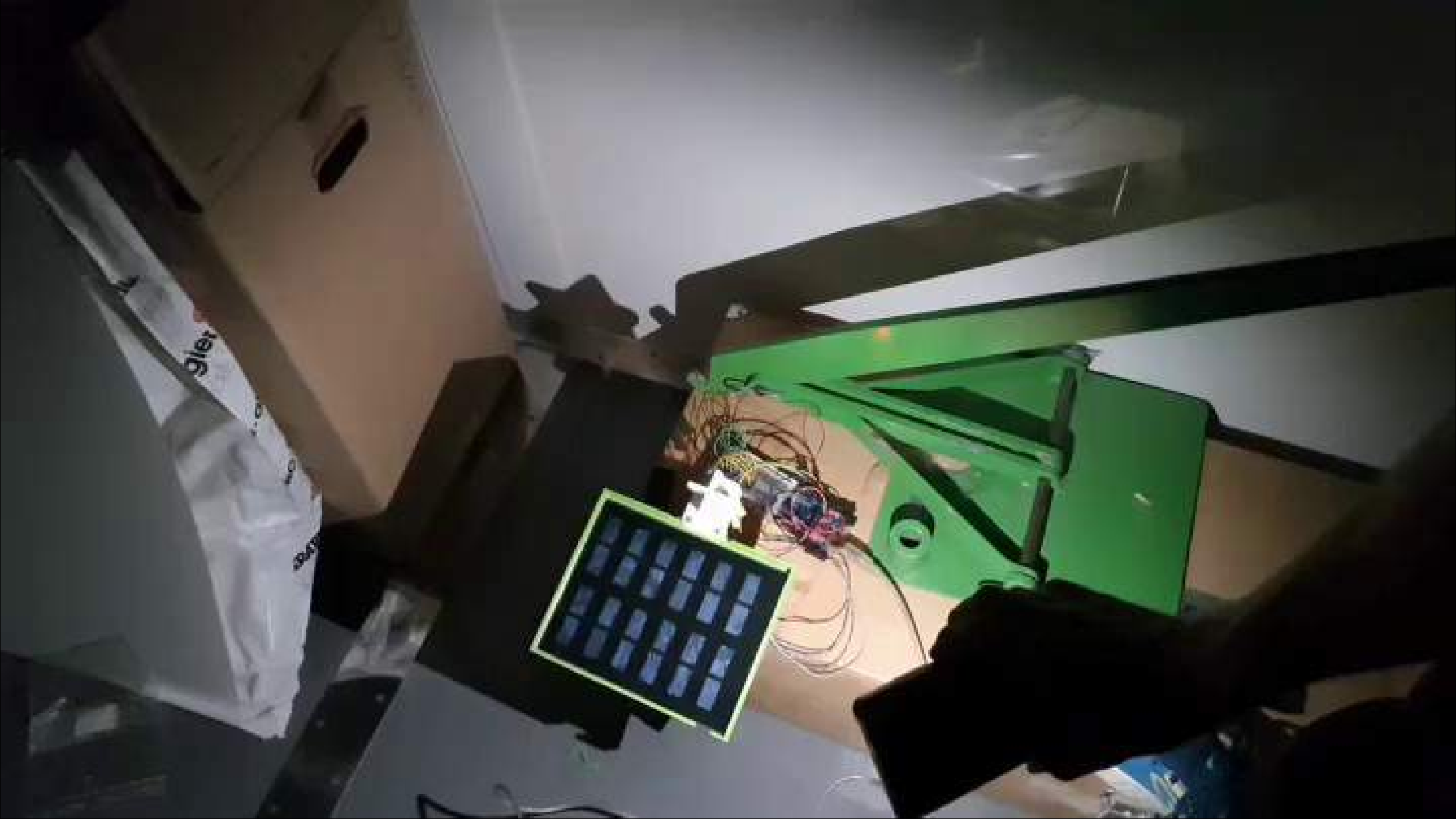
Câblage

Durant ce projet
je suis chargé :

- Câblage
- Assemblage







Problèmes rencontrés

Code

- Test des photorésistances et des servomoteurs
- Perte du fichier

Modélisation

- Trouver un modèle 3D adapté
- Problème d'impression 3D

Câblage

- Court-circuit
- Problème de photorésistances et de servomoteurs

Points à améliorer

- Rentabilité énergétique
- Batterie
- Esthétisme