epsi® l'école d'ingénierie informatique
wis® l'école supérieure des métiers du digital

La Machine de « Rube Goldberg »

Workshop B1 / 2025-26 /

Groupe 6 - Montpellier

Team G6 :

RAHANTANIRINA Jordy

LENOIS-GELAS Nicolas

BENINI Ivann

ABBAOUI Wassim

Objectif du projet :

- Le projet vise à construire une machine inutilement complexe pour réaliser une tâche simple via une réaction en chaîne, développant ainsi créativité, logique et travail manuel. La machine doit permettre des enchaînements fluides et un parcours complet de la bille sur toutes ses faces, avec des entrées et sorties différentes.

Story Time :

- Nous sommes en 2068 et nous avons créé une attraction inutilement complexe et loufoque. L'objectif est de s'amuser et de laisser libre cours à notre imagination. Nous espérons que cela vous plaira.

Par où débuter ?

- Le brainstorming est une phase cruciale de tout projet, car il détermine les étapes suivantes et guide l'équipe dans la bonne direction. À partir de cette base vide, toute une architecture se met en place pour donner vie à nos idées.

Premier pas...

- Le projet est désormais lancé, les premières idées rédigées se transforment en actions concrètes. Des questions commencent à apparaître pour évaluer si c'était réellement une bonne idée, mais l'objectif reste précis : continuer.

Chaque étape d'une machine de Goldberg correspond à une instruction dans un code.

- Chaque phase de la machine fonctionne comme une commande dans un programme : elle doit s'exécuter dans une séquence précise pour garantir la réussite de la tâche. Pour ce faire, le code a été construit sous la forme d'un automate à états de sorte à pouvoir gérer au mieux les divers composants. Cette façon de programmer nous a permis de gérer les risques de défaillance.

Utilisation de Fusion 360 !

- Après avoir eu recours à notre savoir-faire manuel, nous avons décidé de modéliser notre première pièce à l'aide de l'imprimante 3D du MyDil (notre superbe ascenseur).

IoT :

- Grâce au plugin PlatformIO, nous avons pu développer l'IoT en utilisant l'IDE (Integrated Developement Environment) 'Clion'.

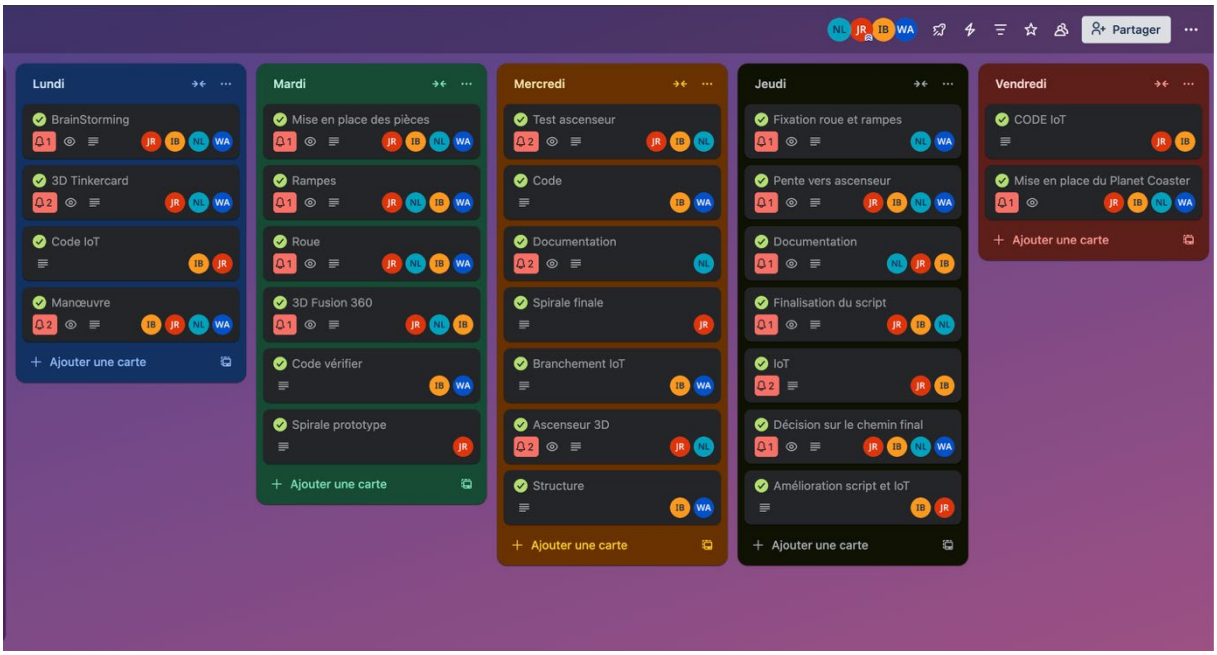
Modules et Composants :

- 1 Module ESP8266.
- 2 Actionneurs :
    - 1 Servo-Moteur SG90,
    - 1 Moteur DC avec driver L298N.
- 2 Capteurs :
    - 1 Capteur infrarouge de proximité,
    - 1 Capteur de vibration.
- Divers
    - Boite 400x400x200cm,
    - Breadboard,
    - Câbles Dupont M/M – M/F – F/F,
    - Alimentation externe 7.5V,
    - Polystyrènes,
    - Cartons,
    - Brochettes,
    - Colle, scotch, etc.

Machines Numériques :

- Imprimante 3D Creality K2 Plus avec sécheur de filament
- Découpeuse/Graveuse laser Creality Falcon A1

Brainstorming et Plan du projet :



Lundi Brainstorming :

- Mise en commun des idées de chacun,
- Jordy : division de la boîte en 3 parties (horizontales ou verticales),
- Nico : idée de l'ascenseur et de la roue,
- Ivann : propulsion de la bille avec une rampe,
- Wassim : proposition d'un looping,

Modélisation 3D :

- Connexion de tout le groupe sur une même page Tinkercad puis Fusion360.
- Première modélisation des idées issues du brainstorming : spirale, rampe, poulie, ascenseur, roue.,

Code IoT :

- Installation des logiciels nécessaires.,
- Création du premier dossier de code pour débuter la programmation.,

Manœuvre (construction physique) :

- Réalisation des premières rampes.,
- Construction de la première roue.

Résumé – Mardi Mise en place des pièces :

- Fixation des deux premières rampes et de la roue.,

Rampes :

- Refonte des modèles initiaux pour obtenir des rampes plus solides et stables.,

Roue :

- Amélioration du premier prototype de roue avec un nouveau modèle plus fonctionnel.,

Modélisation 3D (Fusion 360) :

- Jordy et Nico ont conçu la majorité des pièces nécessaires,
- Conception de l'ascenseur,
- Réalisation d'une poulie pour enrouler le fil de l'ascenseur,
- Création d'adaptateurs pour relier les composants aux autres objets,

Vérification du code :

- Premier test d'une partie du code IoT,

Résumé – Mercredi Test de l'ascenseur :

- Difficultés initiales pour la fixation de l'ascenseur,
- Stabilisation réussie en fin de journée,

Documentation :

- Remplissage du PDF et du PowerPoint,
- Ajout des composants utilisés dans le projet,

Spirale (prototype) :

- Première spirale fonctionnelle même sans support central,
- Observation : en agrandissant le centre, il serait possible d'y intégrer l'ascenseur,

Ascenseur 3D :

- Conception du modèle final, plus grand et plus résistant,
- Objectif : assurer la montée correcte de la bille,

Spirale (finale) :

- Réalisation du modèle final avec :
- Diamètre plus grand.
    - Design amélioré,
    - Meilleure stabilité pour permettre le passage de l'ascenseur,

Structure :

- Ivann a prolongé son temps de travail jusqu'à 18h30,
- Ajout de parois en polystyrène,
- Renforcement de la stabilité de l'ascenseur et de la spirale,

Branchement IoT :

- Réalisation des premiers branchements des composants IoT,

Code :

- Test des premiers scripts programmés.,

Résumé – Jeudi Fixation roue et rampes :

- Fixation définitive de la roue et des rampes,
- Ajustements pour que la bille suive correctement le parcours prévu,

Pente vers ascenseur :

- Finalisation de la pente inférieure,
- Construction du chemin menant jusqu'à l'ascenseur,

Documentation :

- Avancée sur le PowerPoint, le PDF et la gestion des tâches sur Trello,

Finalisation du script :

- Script global finalisé pour coordonner l'ensemble des éléments du projet,

IoT :

- Réalisation des branchements finaux,
- Début de l'installation des différents composants IoT,

Décision sur le chemin final :

- Annulation du mouvement de la roue (fonctionnelle sans moteur, évitant ainsi un composant inutile),
- Brainstorming pour enrichir l'expérience (ajout éventuel de LEDs ou de sons buzzer pour un effet "parc d'attraction"),
- Choix final du parcours
- Capteur infrarouge pour détecter la bille à l'entrée.
    - Servo utilisé comme barrière pour libérer la bille vers l'ascenseur,
    - Moteur pour faire monter l'ascenseur,
    - Libération de la bille sur la spirale,
    - Capteur de vibration déclenchant le buzzer,
    - Moteur déroulant l'ascenseur pour le faire redescendre,

Amélioration script et IoT :

- Ivann et Jordy sont restés jusqu'à 19h30,
- Optimisation et mise au point finale du script,
- Installation correcte des premiers composants IoT,

Mise en place des éléments finaux 3D :

- Démontage des éléments prototype et mise en place de la version finale pour l'oral du lendemain matin.

Code du fichier main.cpp :

```
// --- Déclaration des broches ---
const int buzzerpin = D0;       // Buzzer
const int vibpin = D3;          // Capteur de vibration/choc
const int dectectIRpin = D5;    // Capteur infrarouge
const int servopin = D6;        // Servo
const int motorin1 = D1;        // Moteur commande 1 (pont H)
const int motorin2 = D2;        // Moteur commande 2 (pont H)

// --- Constantes de temporisation ---
const int timeUp = 1000;        // Durée d'activation du moteur
const int timeWait = 2500;      // Délai avant lancement du moteur après détection IR

// --- États ---
boolean ascrun = false;         // Séquence moteur à exécuter (après IR)
boolean waitshock = false;      // Attente d'un choc/vibration (avant retour moteur)

// Fonction de Pilotage manuel du servo via impulsions PWM
void servoSetAngle(int _angle) {
    static unsigned long lastTime = 0;
    unsigned long currentTime = millis();

    // Impulsion envoyée toutes les ~20 ms
    if (currentTime - lastTime > 20) {
        digitalWrite(servopin, HIGH);
        delayMicroseconds(map(_angle,0,180,650,2600));  // Largeur proportionnelle à l'angle
        digitalWrite(servopin, LOW);
        lastTime = currentTime;
    }
}
```

```
// Fonction système d'initialisation des broches utilisées
void setup() {
    // Configuration des broches
    pinMode(buzzerpin, OUTPUT);
    pinMode(vibpin, INPUT);
    pinMode(dectectIRpin, INPUT);
    pinMode(servopin, OUTPUT);
    pinMode(motorin1, OUTPUT);
    pinMode(motorin2, OUTPUT);
    pinMode(LED_BUILTIN, OUTPUT);

    // Moteur à l'arrêt au démarrage
    digitalWrite(motorin1,LOW);
    digitalWrite(motorin2,LOW);
}

// Fonction principale exécutée en boucle
void loop() {
    static unsigned long timeRun = 0;
    unsigned long currentTime = millis();

    // --- Étape 1 : Détection IR ---
    // Si un objet est détecté, le servo s'oriente et on prépare la séquence moteur
    if (digitalRead(dectectIRpin) == LOW) {
        digitalWrite(LED_BUILTIN,LOW);   // LED ON
        servoSetAngle(120);              // Servo ouvert
        ascrun = true;                   // Activer la séquence moteur
     } else {
        digitalWrite(LED_BUILTIN,HIGH);  // LED OFF
        servoSetAngle(0);                // Servo refermé
    }

    // --- Étape 2 : Attente puis activation moteur dans un sens ---
    if (ascrun) {
        if (timeRun == 0) { timeRun = currentTime; }  // Début du chrono
        if (currentTime - timeRun > timeWait) {       // Attente écoulée
```

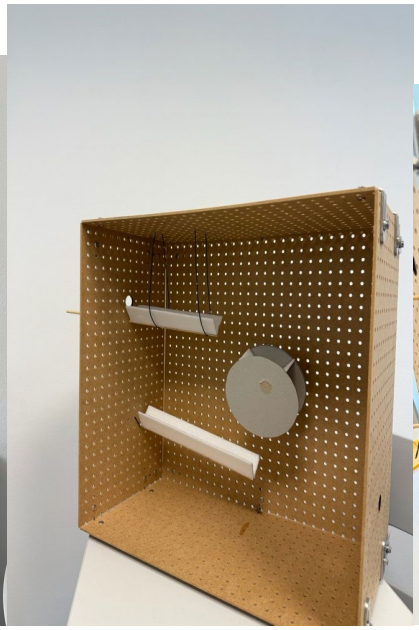
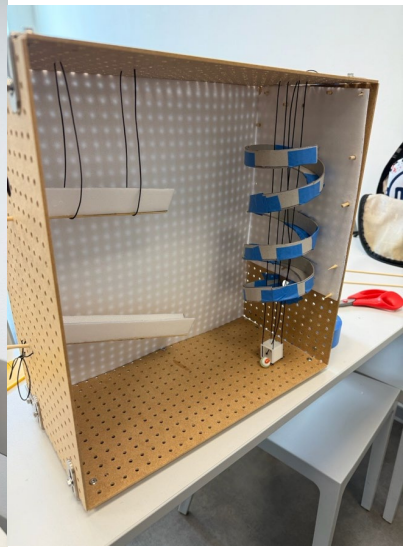
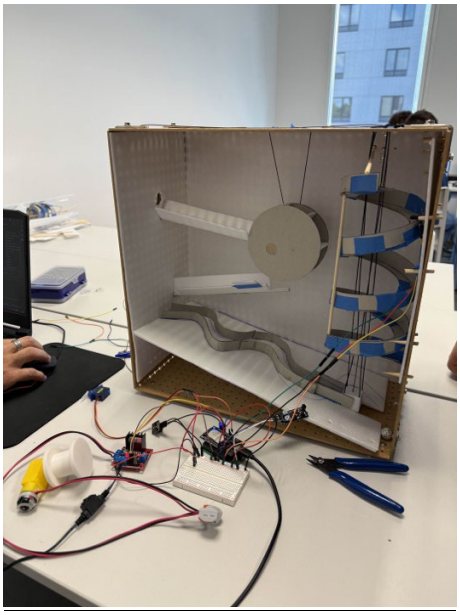
```
            digitalWrite(motorin1,HIGH);              // Moteur avant
            digitalWrite(motorin2,LOW);
            delay(timeUp);                            // Durée d'activation
            digitalWrite(motorin1,LOW);               // Stop moteur
            digitalWrite(motorin2,LOW);
            ascrun = false;
            timeRun = 0;
            waitshock = true;                         // Passer en attente de choc
        }
     }

    // --- Étape 3 : Détection choc → alarme + retour moteur ---
    if (waitshock) {
        if (digitalRead(vibpin) == LOW) {             // Si choc détecté
            delay(1000);                              // Anti-rebond
            digitalWrite(buzzerpin,HIGH);             // Buzzer ON
            delay(2000);                              // Durée buzzer
            digitalWrite(buzzerpin,LOW);              // Buzzer OFF

            waitshock = false;

            digitalWrite(motorin1,LOW);               // Moteur arrière
            digitalWrite(motorin2,HIGH);
            delay(timeUp);
            digitalWrite(motorin1,LOW);               // Stop moteur
            digitalWrite(motorin2,LOW);
        }
    }
}
```

Quelques photos en vracs :

Rendez-vous le mercredi 1er octobre pour assister à l'inauguration du Parc « Planet Coaster » et découvrir sa version finale…