



PLANT MANAGER

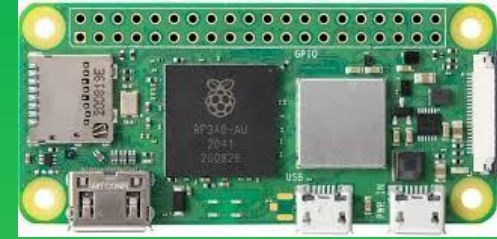
VOTRE ASSISTANT INTELLIGENT POUR UNE GESTION PARFAITE DE
VOS PLANTES

GOMES Thomas
PICHOT Alexandre

SOMMAIRE

- Architecture du projet
- Développement Matériel
 - Organisation
 - Tests capteurs unitaires
 - Carte Pompe
 - Carte plante
 - Boitier plante
 - Boitier pompe
- Développement Logiciel
 - Script python
 - Base de données
 - Serveur Node.js
 - Interface web

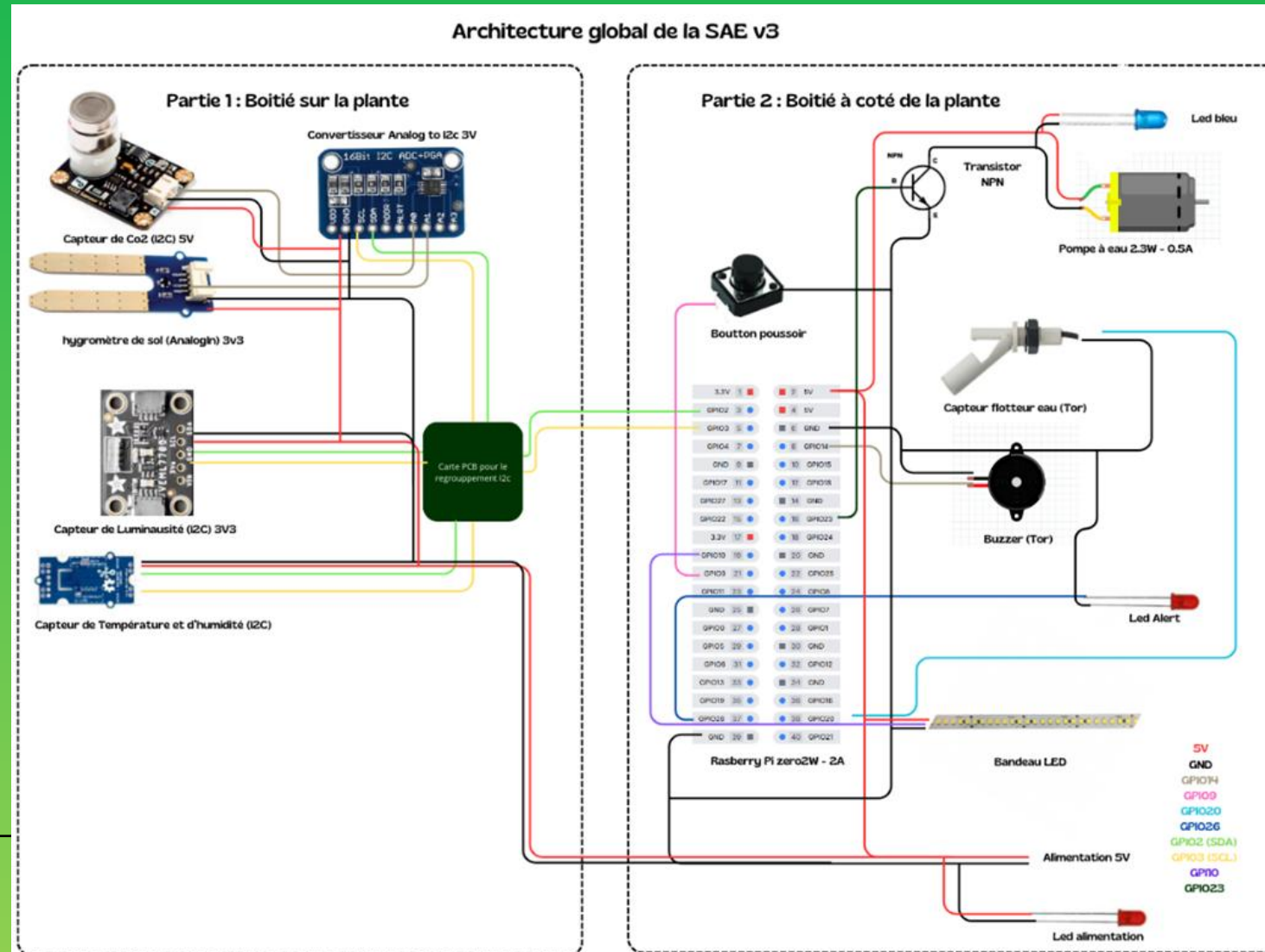
INTRODUCTION



- **Deux boîtiers** : capteurs et gestion de l'arrosage.
- **Capteurs** : Co2, température, lumière...
- **Actionneurs** : Pompe à eau + capteur flotteur pour irrigation.
- **Raspberry Pi Zero 2W** pour le contrôle.
- Parfait pour **pallier les absences et les oublis d'arrosage**.



ARCHITECTURE

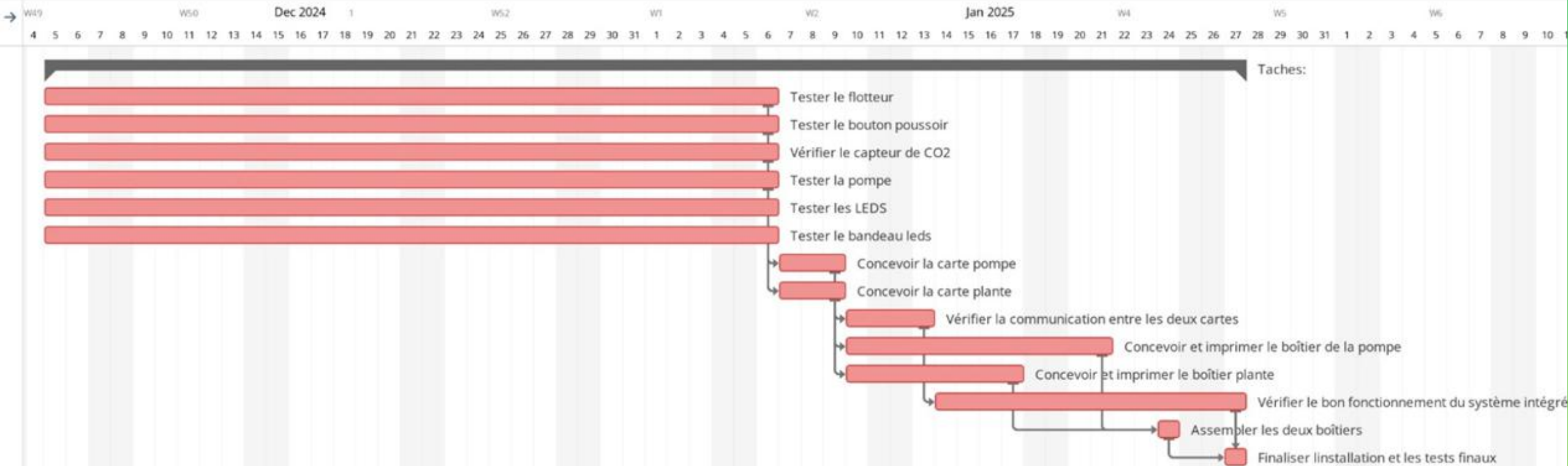


ORGANISATION

Taches Pichot

Read-only view, generated on 24 Feb 2025

Instagantt

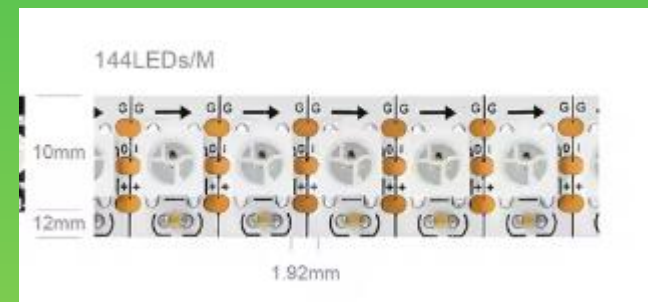
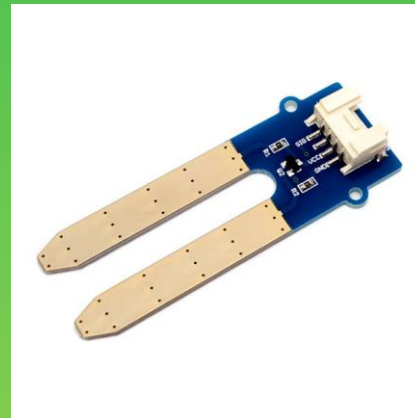
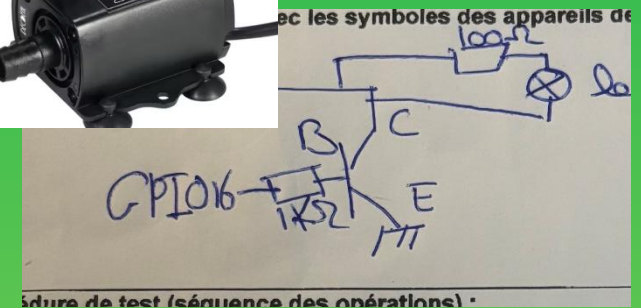


TESTS CAPTEURS & ACTIONNEURS UNITAIRES

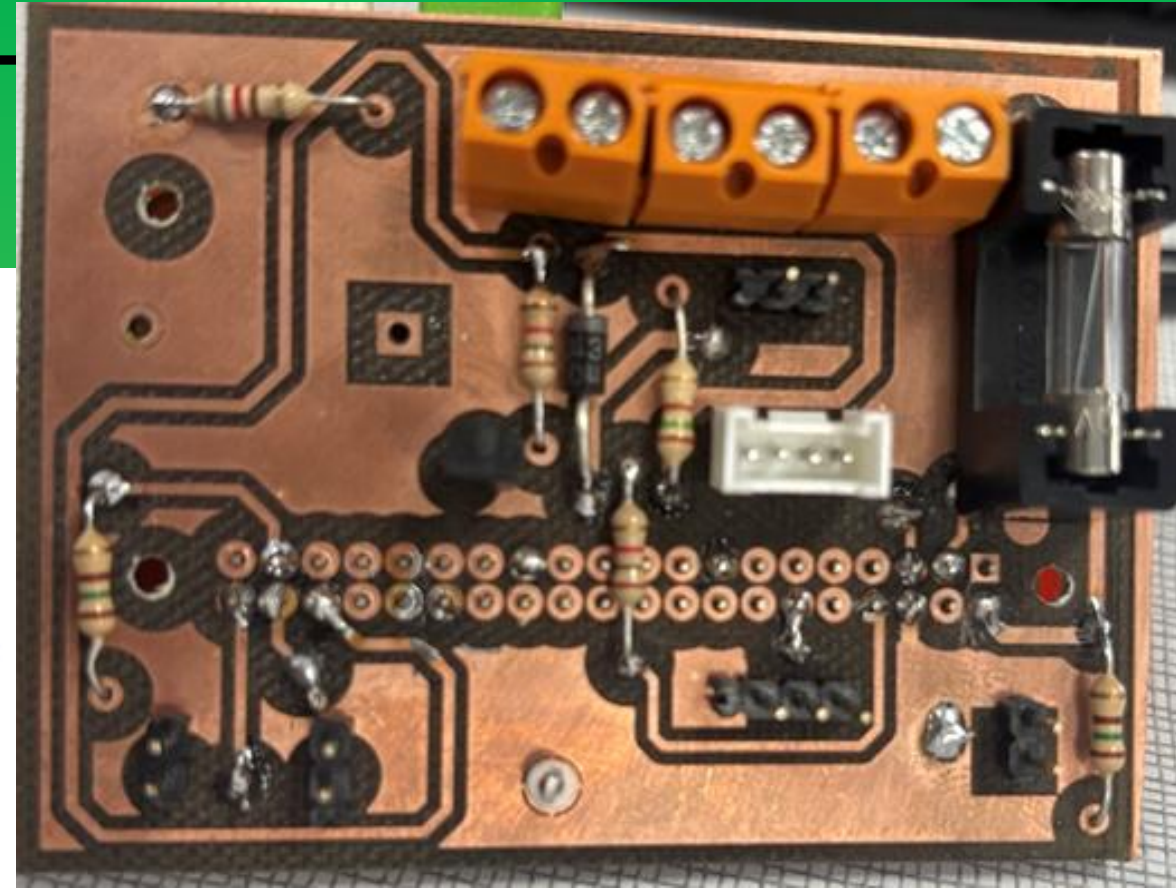
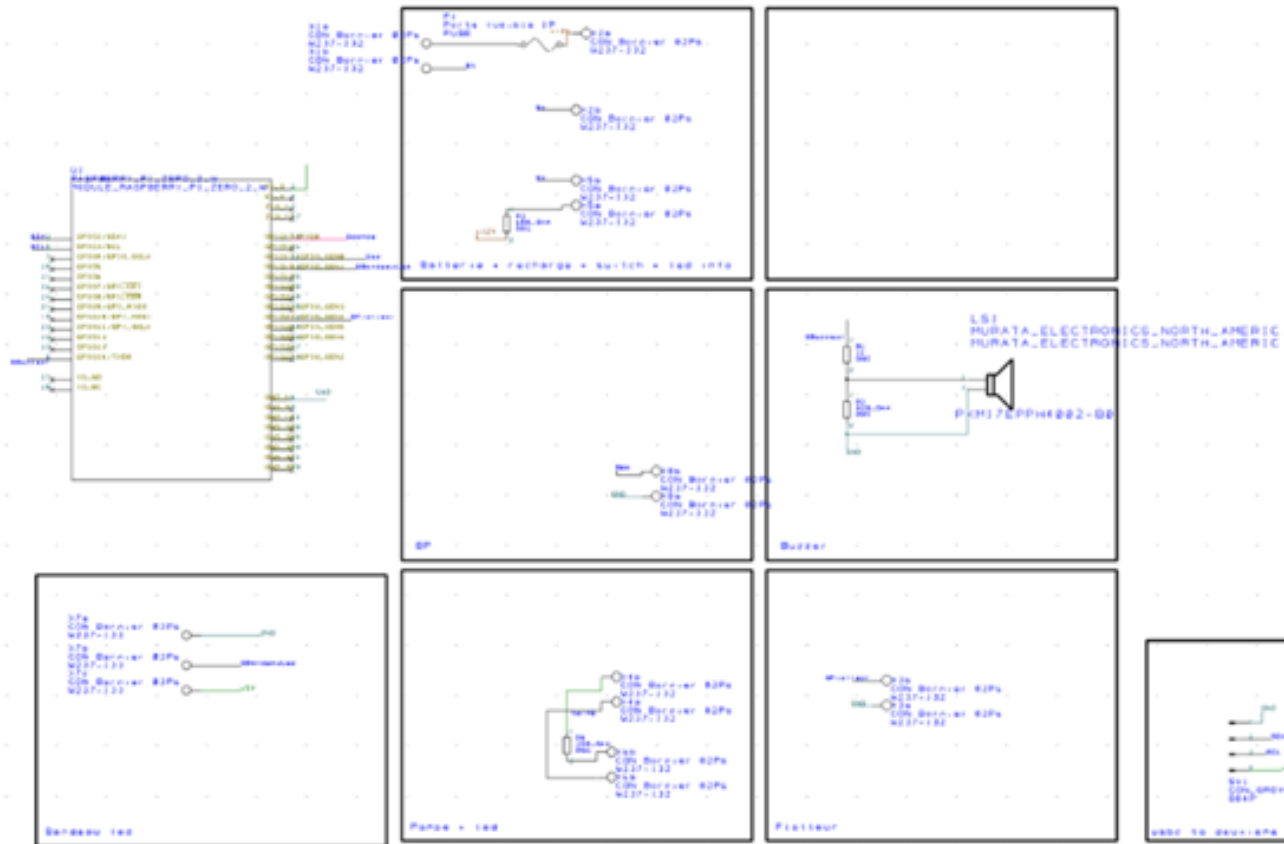
- Capteur de CO2
- Capteur d'humidité et de température
- Capteur de luminosité
- Humidité de sol
- Capteur du niveau de l'eau
- Bouton poussoir



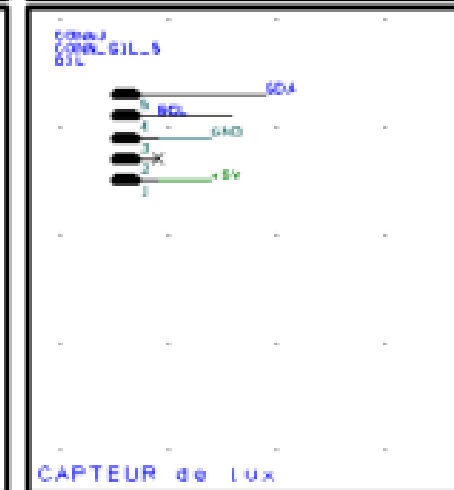
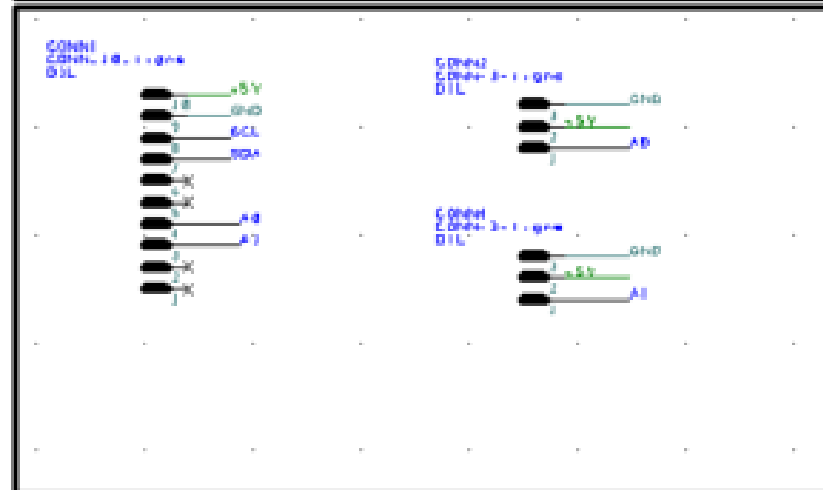
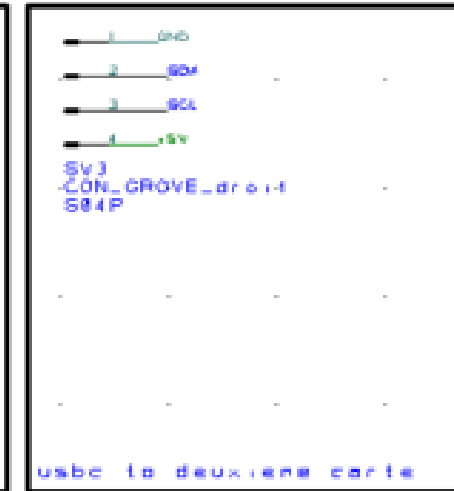
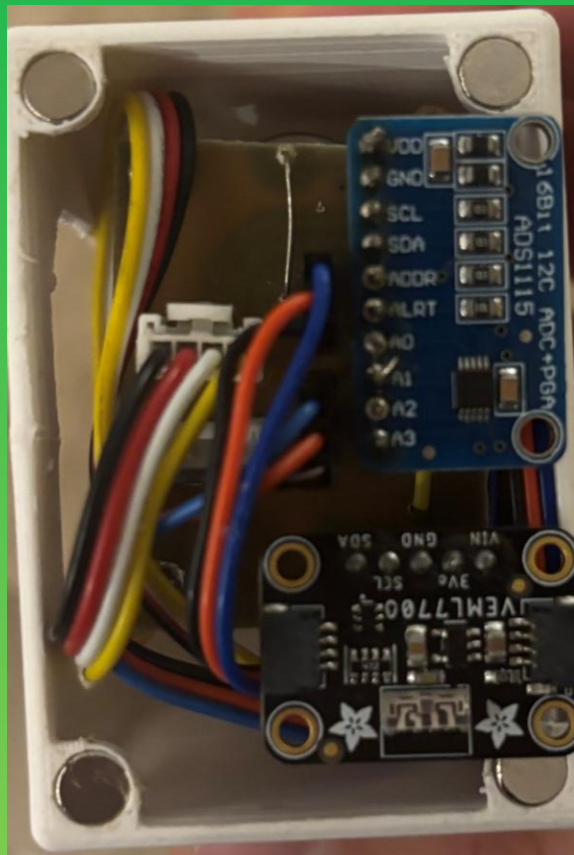
- Pompe
- Leds
- Buzzer
- Ruban led (144)



CARTE POMPE

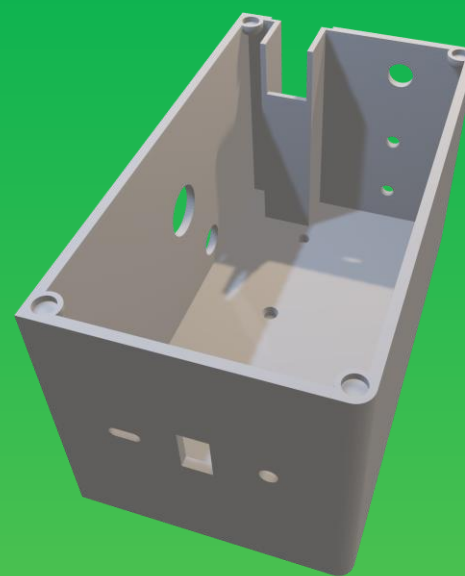


CARTE POMPE

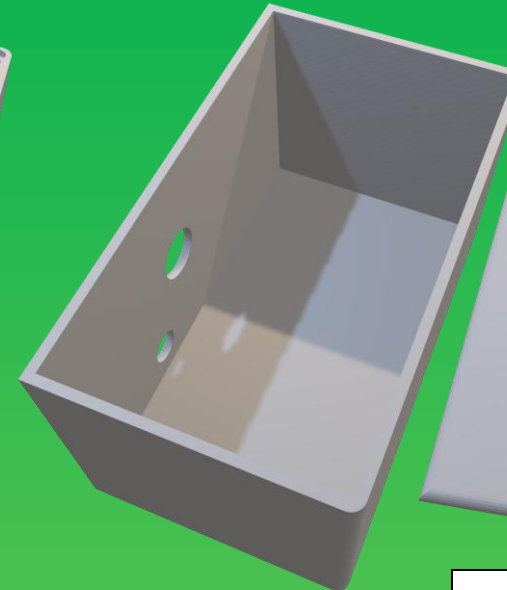


BOITIER PLANTE

- Type de matériaux
- Insert en laiton M2
- Modélisation des équipements
- Aimants
- Minimum de support



Boitier Pompe



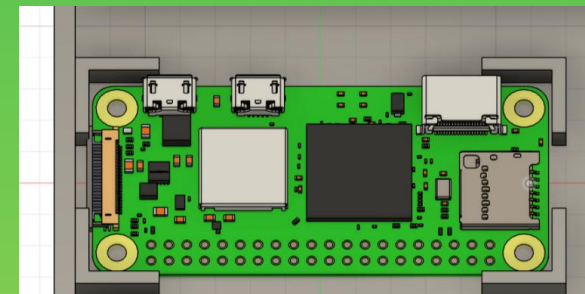
Boitier Eau



Couvercle pour
le boitier pompe



Cache
bandeau leds



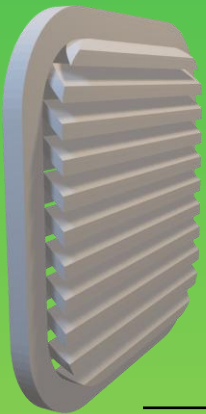
Raspberry pi
Zero 2W

BOITIER PLANTE – RENDU FINALE



DÉVELOPPEMENT MATÉRIEL - BOITIER POMPE

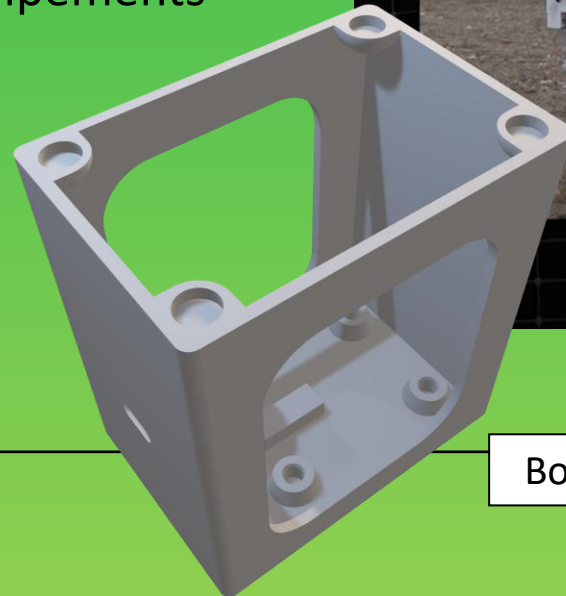
- Invention de Thomas Stevenson
- Type de matériaux et couleur
- Insert en laiton M2
- Modélisation des équipements



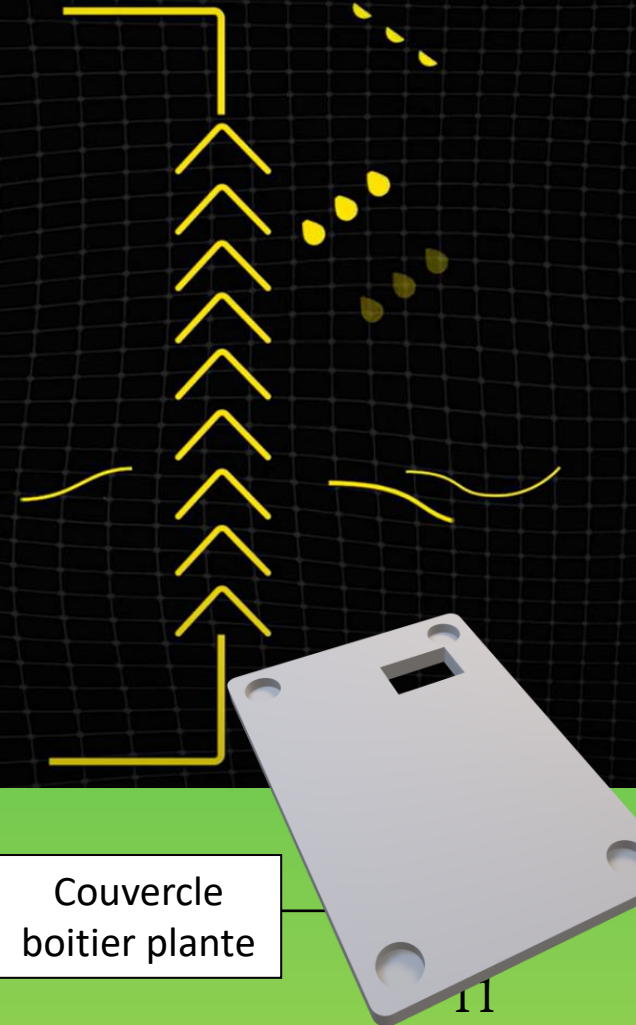
Fenêtre

Aimants

Entretoises



Boitier plante



Couvercle
boitier plante

BOITIER POMPE – RENDU FINALE



DÉVELOPPEMENT LOGICIEL- RASPBERRY

- RASPBERRY PI ZERO 2W
- GPIO, I2C
- Interconnexions
- Gestion le serveur web
- Environnement virtuel python



DÉVELOPPEMENT LOGICIEL- SCRIPT PYTHON

Chaque capteur possède sa propre fonction de lecture de données :

- `lire_luminosite()` : Renvoie la luminosité en lux.
- `lire_co2()` : Convertit la tension mesurée en concentration de CO₂ (ppm).
- `lire_humidite_sol()` : Convertit la tension mesurée en humidité du sol (%).
- `lire_flotteur()` : Vérifie si le réservoir est vide ou plein.
- `lire_bouton_poussoir()` : Vérifie si l'utilisateur appuie sur le bouton.



Chaque capteur possède sa propre fonction de lecture de données :

- `lire_luminosite()` : Renvoie la luminosité en lux.
- `lire_co2()` : Convertit la tension mesurée en concentration de CO₂ (ppm).
- `lire_humidite_sol()` : Convertit la tension mesurée en humidité du sol (%).
- `lire_flotteur()` : Vérifie si le réservoir est vide ou plein.
- `lire_bouton_poussoir()` : Vérifie si l'utilisateur appuie sur le bouton.

DÉVELOPPEMENT LOGICIEL- BASE DE DONNÉES

- Base de données : Gestion des données & configurations environnement

Nom	Type
▼ Tables (5)	
▼ capteurs_configuration	
id	INTEGER
id_configuration	INTEGER
type	TEXT
valeur_ideale	TEXT
unite	TEXT
▼ configurations	
id	INTEGER
nom_configuration	TEXT
nom_plante	TEXT

▼ data_capteur	
id	INTEGER
type	TEXT
valeur	REAL
unite	TEXT
horaire	DATETIME
▼ niveau_eau	
id	INTEGER
valeur	INTEGER
horaire	DATETIME

DÉVELOPPEMENT LOGICIEL- SERVEUR NODE.JS

- express : Gestion des routes et API.
 - express-session : Gestion des sessions utilisateur.
 - sqlite3 : Gestion de la base de données
 - fs : Gestion des fichiers de configuration.
 - nodemailer : Gestion des envois d'alertes par e-mail.
 - onoff : Gestion des GPIO pour la pompe.
 - ws : Gestion des WebSockets pour mise à jour en temps réel.
-
- Méthode GET & POST

DÉVELOPPEMENT LOGICIEL- SERVEUR NODE.JS

Méthode	Endpoint	Objectif
POST	/api/inscription	Inscription d'un nouvel utilisateur. Vérifie si l'email existe déjà, sinon ajoute l'utilisateur à `utilisateurs.json`.
POST	/api/connexion	Connexion d'un utilisateur. Vérifie les identifiants dans `utilisateurs.json` et crée une session.
GET	/api/verifier-session	Vérifie si l'utilisateur est connecté et retourne ses informations (nom, email, mot de passe).
POST	/api/deconnexion	Déconnecte l'utilisateur en détruisant la session.
POST	/api/modifier-utilisateur	Modifie les informations du compte utilisateur connecté (nom, email, mot de passe).

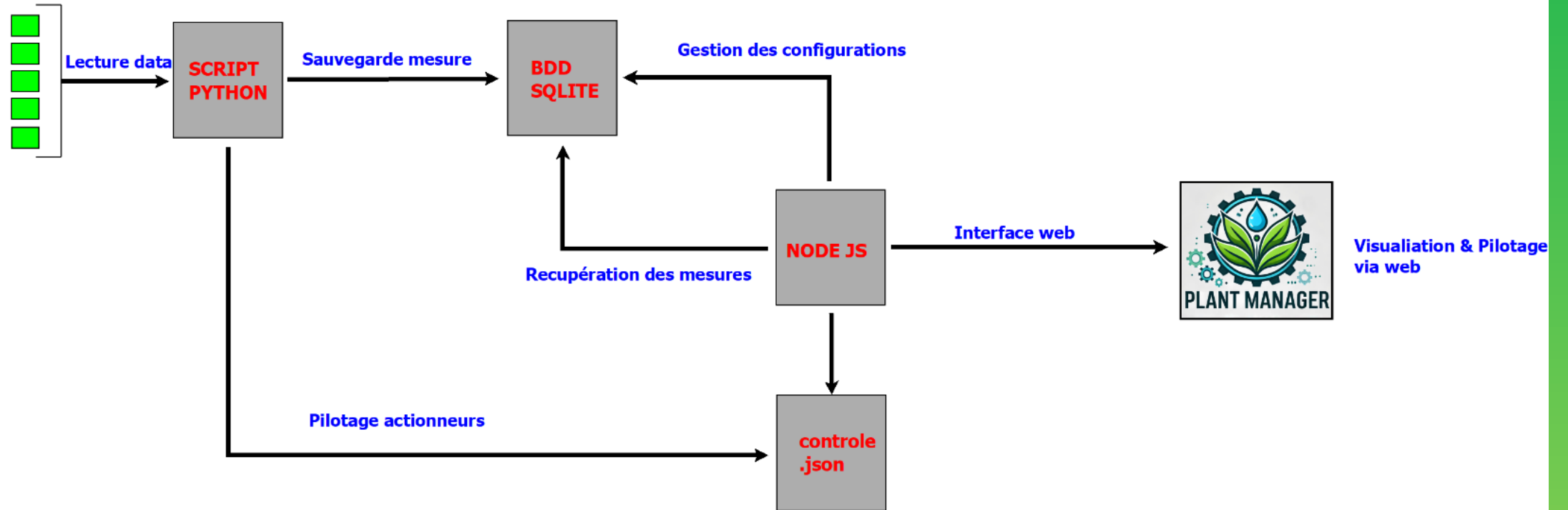
GET	/api/statut-pompe	Récupère l'état actuel de la pompe (on/off) depuis `control.json`.
POST	/api/definir-pompe	Active ou désactive la pompe (écrit dans `control.json`).
GET	/toggle-script?state=on/off	Démarre ou arrête le script `simu_data_capteur.py` qui lit les valeurs des capteurs.

Utilisation d'API pour javascript : fetch

DELETE	/api/reinitialiser-donnees	Vide les tables `data_capteur` et `niveau_eau` dans la base SQLite.
GET	/api/donnees-capteurs	Récupère les dernières valeurs des capteurs stockées dans `data_capteur`.
GET	/api/statut-systeme	Récupère la dernière valeur du niveau d'eau dans `niveau_eau`.
GET	/api/historique-capteurs?limite=X	Récupère l'historique des valeurs des capteurs (limite paramétrable).
GET	/api/historique-eau?limite=X	Récupère l'historique du niveau d'eau (limite paramétrable).
POST	/api/enregistrer-configuration	Enregistre ou met à jour une configuration de plante avec ses capteurs idéaux.
GET	/api/recuperer-configurations	Récupère la liste de toutes les configurations enregistrées.
GET	/api/recuperer-config/:nom_configuration	Récupère une configuration spécifique avec ses capteurs.
DELETE	/api/supprimer-configuration/:nom_configuration	Supprime une configuration de plante et ses capteurs associés.

DÉVELOPPEMENT LOGICIEL- INTERFACE WEB

Capteur/Actionneur



ARBORESCENCES

PlantManager/
node_modules/
public/
 css/
 html/
 javascript/
 media/
controle.json
script_plantmanager.py
package.json
package-lock.json
plant_manager.db
serveur.js
simu-data_capteur.py
utilisateurs.json

- > Dépendances Node.js installées
- > Contient les fichiers vue par le client
- > Styles CSS
 - > Fichiers HTML pour l'interface
 - > Scripts JS pour gérer les interaction
 - > Images
- > Fichier JSON de configuration de contrôle
- > Script pour capteurs physiques
- > Serveur Node.js principal
- > Script de simulation de capteurs
 - > Liste des utilisateurs en JSON

PAGES WEB :

- index
- connexion
- dashboard
- configuration
- data-evolution

RDV SUR LE SITE DE PLANT MANAGER



PLANT MANAGER

PLANT MANAGER - CONCLUSION

- Cahier des charges
 - ✓ Automatisation de l'arrosage
 - ✓ Capteurs intégrés
 - ✓ Gestion du réservoir d'eau
 - ✓ Boitier plante petit et boitier Pompe étanche
 - ✓ Rendu propre avec des ports USB-C
- Points d'améliorations
 - ✓ Développement serveur
 - ✓ Développement script python
 - ✓ Développement base de données
 - ✓ Développement de l'interface web
 - ✓ Prix
 - ✓ Livrables (Video et rapport)

éléments	quantité	prix unitaire	prix tot
USB type c	1	3.88 €	3.88 €
ADS 1115 ADC	1	3.89 €	3.89 €
Ruban Led	1	7.25 €	7.25 €
PLA blanche	0.3	17.99 €	5.40 €
TPU 95 noir	0.2	15.99 €	3.20 €
PLA noir	0.4	17.99 €	7.20 €
pompe	1	5.79 €	5.79 €
cole epoxi bi composants	1	13.00 €	13.00 €
flotteur	1	19.86 €	19.86 €
capteur de Co2	1	58.90 €	58.90 €
capteur de luminosité	1	4.00 €	4.00 €
Entretoise M2	0.2	3.00 €	0.60 €
Raspberry Pi zero 2W	1	20.00 €	20.00 €
Cable USB-C to USB-C	1	10.00 €	10.00 €
		TOTAL :	162.96 €

