|  |  |
| --- | --- |
| CESI  CUBE 2  COMPTE – RENDU | **Comment et pourquoi le siège s’est décroché ?**  THIL Quentin, KIBOUT Abdel-Akram, RIABOV Oleksii |

Table des matières

[Introduction 2](#_Toc157065628)

[Préinstallation : 2](#_Toc157065629)

[Programmation de la carte ESP : 2](#_Toc157065630)

[Programmation de la base de données : 3](#_Toc157065631)

[Programmation base de l’API : 3](#_Toc157065632)

[Conclusion : 3](#_Toc157065633)

[Généralisation du problème : 3](#_Toc157065634)

# Introduction

## Contexte :

Dans un monde de plus en plus axé sur la collecte et l'analyse des données environnementales, la surveillance des températures et de l'humidité revêt une importance cruciale dans divers domaines tels que l'agriculture, la gestion des entrepôts, et même le contrôle climatique des habitations. Notre projet vise à répondre à ce besoin en développant une solution innovante utilisant une sonde de température et d'humidité connectée à un microcontrôleur ESP (Système Embarqué Programmable). Ce microcontrôleur, grâce à son accès Internet, communiquera avec une API (Interface de Programmation Applicative) hébergée sur un serveur distant. L'API servira de pont entre le microcontrôleur et un site web où les données collectées seront stockées et visualisées de manière conviviale. Cette approche intégrée permettra une surveillance en temps réel, une analyse approfondie des tendances climatiques, et offrira des insights précieux pour une gestion efficace des environnements sensibles à la température et à l'humidité.

## Composants à disposition :

**-Raspberry Pi Zero**

**-ESP8266 : Contrôleur WI-FI programmable**

**-CH 340 : convertisseur USB to Serial Port**

**-GY-21 HTU21 : capteur de température**

**-Ecran type I2C SSD1306**

## Problématique :

Comment produire une solution de collecte de données pouvant remplacer celle d’Atmos pour un coût moindre avec les composants dont nous disposons ?

# Préinstallation

## Installation matérielle :

Avant toute manipulation informatique sur les composants programmables, nous devons d’abord effectuer les branchements appropriés. Ainsi, notre projet se compose de deux ensembles de composants :

* Nous avons d’un côté la partie « Sonde » composée de l’ESP 8266, un contrôleur WI-FI, qui a pour utilité de permettre l’envoi de données tirées du capteur de température vers le Raspberry Pi. On a donc aussi le capteur de température GY-21 qui sera relié au contrôleur WI-FI. La liaison entre les deux est assurée par un adaptateur série CH340 qui fait le lien entre les deux et permet de Flasher l’ESP pour le programmer.

Branchements :

Convertisseur CH340 et GY-21 pin à pin:

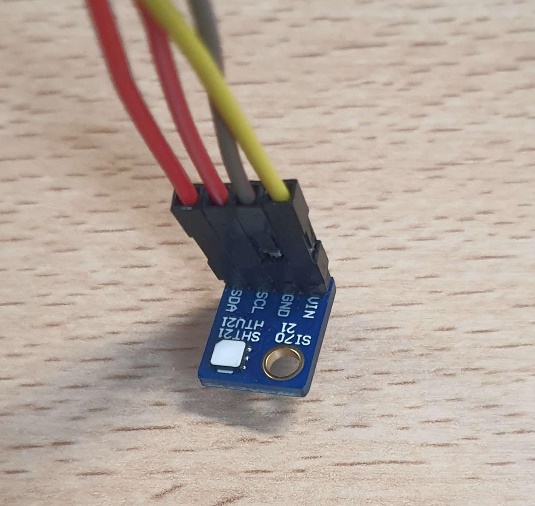
-3v3 -> VIN

-I00 -> SCL

-I02 -> SDA

-GND -> GND

Pour l’ESP et le convertisseur, il faut brancher les 8 pins dans le support jaune prévu à cet effet, dans le sens permettant à l’ESP de se superposer avec le convertisseur sans dépasser. Cela évitera de se tromper de port.



Capteur de température GY-21

Une image contenant texte, Ingénierie électronique, Appareils électroniques, Composant de circuit

Description générée automatiquement

Convertisseur USB et Branchements pour le capteur

Une image contenant fils électriques, câble, Ingénierie électronique, Composant de circuit

Description générée automatiquement

Branchement global de la sonde

Pour ce qui est du Raspberry, les seuls branchements nécessaires sont celui de l’écran ainsi que l’alimentation du Raspberry grâce à un câble USB-3 comme ci-dessous.

Branchements écran -> Raspberry Pi

GND -> pin GND

VCC -> 3.3V

SDA -> SDA

SCK -> SCL

On peut également voir sur la photo qu’une fois terminé, il sera possible d’afficher les relevés du capteur sur l’écran. Il pourra nous servir d’interface permettant de tester les différents ainsi que le bon fonctionnement des différents éléments.

Une image contenant fils électriques, Ingénierie électronique, Composant électronique, Composant de circuit

Description générée automatiquement

Branchements Raspberry Pi/Ecran

## Installation du système d’exploitation :

Une fois les branchements effectués, il faut installer un Système d’exploitation ou OS dans le Raspberry pi. Ici, on peut télécharger tout ça par le biais du logiciel Pi Imager. Ici on choisit le Raspberry Pi OS Lite en sélectionnant la référence du Raspberry pi qu’on possède, Ici un Raspberry Pi Zero WH, à mettre sur une carte SD. Lors de la configuration, veuillez bien à noter les identifiants et mots de passe demandés pour vous connecter ainsi que de rentrer correctement les identifiants de connexion en accord avec ceux de votre connexion internet, pour pouvoir manipuler ensuite le Raspberry pi par connections sh depuis votre ordinateur.

Une image contenant texte, capture d’écran, logiciel, Page web

Description générée automatiquement

Une image contenant texte, capture d’écran, logiciel, Page web

Description générée automatiquement

Une fois que le Raspberry Pi est branché avec l’OS à l’intérieur, ainsi que votre connexion activée, vous pourrez récupérer l’IP du Raspberry et vous connecter en SSH depuis votre ordinateur grâce à la commande « ssh nom\_utilsateur@IP\_adress » entré depuis une invite de commande suivi du mot de passe que vous avez entré lors de la génération de l’OS. Une fois ceci fait, le Raspberry est accessible et vous pouvez créer des dossiers, installer des logiciels et créer des fichiers.

# Étape 1 : Programmation de la carte ESP8266 à l’aide du logiciel Arduino IDE et le convertisseur série USB

## Sélection du port et des librairies

Lors de la programmation de l’ESP grâce à l’IDE Arduino, il faudra, pour téléverser le code dans celui-ci, choisir le bon port pour téléverser. Pour cela, on sélectionne dans l’IDE le bon numéro de port, numéro qu’on peut retrouver dans le gestionnaire de périphérique lorsque la sonde est branchée en USB

Une image contenant texte, capture d’écran, Police, nombre

Description générée automatiquement**Une image contenant texte, Police, capture d’écran, algèbre

Description générée automatiquement**

Si le téléversement est impossible, il s’agit probablement d’un problème de driver, fréquent avec Windows 11. Réinstaller les drivers sur des versions antérieures devrait régler ce problème

La deuxième étape importante c’est de choisir les librairies nécessaires pour faire fonctionner le code.

Une image contenant texte, capture d’écran, logiciel, Logiciel multimédia

Description générée automatiquement

Menu d’installation manuelle des bibliothèques

On peut également installer les bibliothèques par le biais d’URLs :

<http://arduino.esp8266.com/stable/package_esp8266com_index.json,https://arduino.esp8266.com/stable/package_esp8266com_index.json>

Une image contenant texte, capture d’écran, logiciel, Logiciel multimédia

Description générée automatiquement

Une image contenant texte, capture d’écran, logiciel

Description générée automatiquementNous pouvons enfin passer au code : en voici un aperçu. Il est trouvable de manière complète sur le dépôt git fourni avec le projet.

Une image contenant texte, capture d’écran, menu, logiciel

Description générée automatiquement

Une image contenant texte, capture d’écran, logiciel

Description générée automatiquement

Attention :

Les variables définies au début dépendent des identifiants de connexion sur lequel le Raspberry se connecte.  
 Wire.begin(2, 0); si le branchement du capteur de température est I02 -> SDA et I00 -> SCL

Si le branchement est I00 -> SDA et I02 -> SCL le code doit être Wire.begin(0, 2)

Il ne reste plus qu’à téléverser le code dans l’ESP.

# Étape 2 : Le script d’affichage de l’humidité et de la température sur l’écran

Cette partie-là vise à créer un script permettant d’afficher différentes informations telles que la température, l’humidité relevées par le capteur de température.

On va donc commencer par installer plusieurs choses sur le Raspberry Pi, qui exécutera le script directement, tel que la bibliothèque luma.oled pour utiliser l’écran, L’application Python3, pour coder et exécuter du code en python ainsi que Mosquitto MQTT (Message Queuing Telemetry Transport) qui est un protocole de messagerie léger utilisé en IoT pour faire communiquer des appareils connectés entre eux. On écrit donc le code suivant :

Une image contenant texte, capture d’écran, logiciel

Description générée automatiquement

Encore une fois le script est disponible sur le dépôt Git. Une fois ceci fait, on peut exécuter le code avec la ligne « python NOM\_DE\_FICHIER.py » On obtient donc la valeur sur l’IDE Arduino et L’écran. Attention, dans le script ci-dessus, la variable mqtt\_server varie selon l’IP du Raspberry.

Une image contenant Ingénierie électronique, Composant de circuit, Composant électronique, Appareils électroniques

Description générée automatiquement

# Étape 3 : La base de données

On peut ensuite commencer à installer un SGBDR dans le but de créer deux bases de données pouvant stocker respectivement les données de température et d’humidité ainsi que le Wordpress.

Tout d’abord il faut installer Mariadb :

sudo apt-get install mysql-server

Puis se connecter à Mariadb :

Sudo mysql -u root -p

Ensuite créer la base de données :

USE temperature\_db;

CREATE TABLE temperature\_data (

id INT AUTO\_INCREMENT PRIMARY KEY,

temperature FLOAT,

humidity FLOAT,

recorded\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP

);

Une image contenant texte, capture d’écran, Police, logiciel

Description générée automatiquement

sudo mysql -u root -p

Saisir SELECT \* FROM temperature\_data; pour vérifier que les données sont présentes dans la base.

Output :

Une image contenant capture d’écran, Police, texte, ligne

Description générée automatiquement

Une image contenant texte, capture d’écran, menu, livre

Description générée automatiquement

Une fois ceci fait, il faut en créer une deuxième BDD dans le but de stocker le site internet on télécharge également des paquets supplémentaires et gérer les permissions pour éviter tout conflit :

**CREATE USER 'wordpress'@'localhost' IDENTIFIED BY 'rootwordpress';**

**GRANT ALL ON wordpress.\* TO 'wordpress'@'localhost';**

On peut en profiter pour installer les paquets PHP pour plus tard ainsi que le logiciel Wordpress avant de continuer.

**1.Paquets supplémentaire :**

**sudo apt install php-mysql**

**C.Mise en place de Wordpress :**

**2.Fichier wordpress**

**sudo wget https://wordpress.org/latest.zip -O /var/www/html/wordpress.zip**

**sudo unzip wordpress.zip**

**3.Configuration de permissions:**

**sudo chmod 755 wordpress -R**

**sudo chown www-data wordpress -R**

# Étape 4 : La base de données :

Une API (application programming interface ou « interface de programmation d'application ») est une interface logicielle qui permet de « connecter » un logiciel ou un service à un autre logiciel ou service afin d'échanger des données et des fonctionnalités. Ici l’API va avoir pour but de permettre l’échange de données entre la sonde et le Raspberry. Dans notre, nous allons principalement utiliser Flask, un framework Python utilisé pour créer des applications Web. Plus précisément, nous allons installer :

**Flask** : Un framework web en Python utilisé pour créer des applications web.

**Flask-Restx** : Une extension Flask pour la création d'API RESTful.

**Flask-SQLAlchemy** : Une extension Flask qui fournit un support SQLAlchemy pour l'application.

**SQLAlchemy** : Une bibliothèque Python qui facilite la communication avec les bases de données relationnelles.

**Requests** : Une bibliothèque Python qui permet d'envoyer des requêtes HTTP.

**2.**

Tout d’abord nous avons créé un sous-répertoire pour pouvoir y accéder plus facilement. Ensuite nous avons créé le code à partir des informations données dans les consignes et à des cours vus à Cesi et sur internet. Ce code est représenté par ce code en dessous :

Une image contenant texte, capture d’écran

Description générée automatiquement

Une image contenant texte, capture d’écran

Description générée automatiquement

# Étape 5 : le site :

Pour enfin faire le site on va avoir besoin de différents outils :

* Apache2 : Serveur http pour permettre de faire des requêtes

**Une image contenant texte, capture d’écran, Police, logiciel

Description générée automatiquement**

-PHP : langage de programmation web destiné à la création de site web

- Wordpress : Système de gestion de contenu obtenable grâce aux commandes suivantes :

**sudo wget https://wordpress.org/latest.zip -O /var/www/html/wordpress.zip**

**sudo unzip wordpress.zip**

On peut ensuite paramétrer Wordpress :

**Une image contenant texte, capture d’écran, Police, lettre

Description générée automatiquement**

**Une image contenant texte, capture d’écran, nombre, Police

Description générée automatiquement**

On crée un plugin Wordpress dans un nouveau dossier pour faciliter l’utilisation de celui-ci puis deux fichiers : un JSON et un PHP puis on rentre les codes suivants :

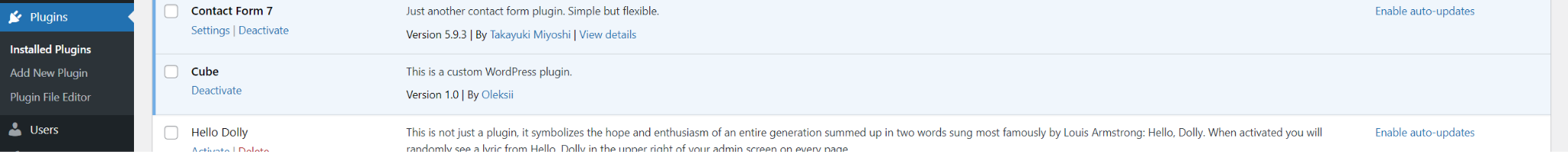
Une image contenant texte, capture d’écran

Description générée automatiquement

Une image contenant texte, capture d’écran

Description générée automatiquement

Une fois cela fait, il ne nous reste plus qu’à activer le plug-in depuis le Dashboard de Wordpress :



Puis de mettre ce code HTML sur Wordpress :

Une image contenant texte, capture d’écran

Description générée automatiquement

Le site devrait maintenant être opérationnel et accessible avec l’URL du site.

# Conclusion :

Dans cette étude, l'intégration d'une API, d'un ESP (Système Embarqué Programmable) et de sondes de température représente une avancée significative dans la gestion et le suivi des données environnementales. L'utilisation d'une API permet une communication fluide entre différents systèmes, facilitant ainsi l'échange et l'exploitation des informations recueillies. L'intégration d'un ESP offre une solution flexible et évolutive pour la collecte en temps réel des données, tout en permettant un contrôle à distance et une automatisation des processus. En combinant ces technologies avec des sondes de température, il devient possible de surveiller efficacement les variations thermiques dans divers environnements, offrant ainsi des insights précieux pour la prise de décision et la gestion des ressources. Cette approche innovante ouvre la voie à de nombreuses applications pratiques, allant de la surveillance environnementale à la gestion intelligente des bâtiments, offrant ainsi des opportunités prometteuses pour l'optimisation des processus et la préservation des ressources.