

2023/2024

NOM : HAVARD-COLSON  
PRENOM : Alexis

- Stage Scientifique et Technique (S7)  
 Stage de fin d'étude (Stage Ingénieur - S10)

## Rapport de stage

**ENTREPRISE :** UNIVERSIDAD DE VALPARAISO

**DATES :** 24/07/2023 au 17/11/2023

**SUJET DE STAGE :** Stage en laboratoire pour l'acquisition de données et de l'ingénierie de prétraitement

**TUTEUR EN ENTREPRISE :** Ana AGUILERA FARACO – Professeur d'université



### CADRE RÉSERVÉ POUR LES STAGES DE FIN D'ÉTUDES (STAGE INGÉNIEUR)

**OPTION**  SE  BIO  EOC  NRJ  CSS  LD  DSMT  NG  TVI  
 DI  CS

**CONFIDENTIALITÉ** mon rapport est confidentiel niveau  1 (moy)  2 (élevé)  
*Un mail a été envoyé par votre tuteur à Mme Baronnet (Dijon), Mme Delevaux (Paris) ou Mme Omont (Angers) au moins un avant votre soutenance, sinon l'ESEO retient le niveau 0 cf. fiche d'encadrement.*

**DOMAINE ENTREPRISE**  Auto.  Aéron.  Banques-fi-assur.  Biomédical-santé  Energie  
 NTIC  Télécoms  Autres (précisez) :

**AUTRES POINTS**  Stage à dominante **management** ou  Stage à dominante **recherche**  
 E5e effectuée sous Contrat Pro  
 Mon tuteur sera présent à ma soutenance  
 Mon tuteur participera au déjeuner le jour de ma soutenance

**ENGAGEMENT DE NON PLAGIAT**

Je soussigné(e), (votre prénom et NOM), étudiant(e)(e) à l'ESEO, atteste avoir pris connaissance du contenu du Règlement intérieur de l'École et de l'engagement de « non-plagiat ». Je déclare m'y conformer dans le cadre de la rédaction de ce document. Je déclare sur l'honneur que le contenu du présent mémoire est original et reflète mon travail personnel. J'atteste que les citations sont correctement signalées par des guillemets et que les sources de tous les emprunts ponctuels à d'autres auteur(e)s, textuels ou non textuels, sont indiquées. Le non-respect de cet engagement m'exposerait à des sanctions dont j'ai bien pris connaissance.



Fait à Viña de Mar le 12/11/2023

## REMERCIEMENTS

Avant de commencer mon rapport, j'aimerais remercier toutes les personnes qui ont contribuées au succès de mon stage effectué au sein de l'université de Valparaiso.

Dans un premier temps, je tiens à remercier tous les professeurs de l'ESEO qui m'ont suivi lors de mon stage à l'étranger M. SCHANG Daniel et M. LATRACH Mohamed.

Ensuite, ma maître de stage, Mme AGUILERA FARACO Ana, professeur au sein de l'université, pour son accueil au sein de l'université, pour le projet qui m'a été confié, et surtout pour son suivi régulier et les multiples aides et conseils apportés pour mener à bien celui-ci.

Je tiens remercier toute l'équipe autour d'Ana qui ont eux-aussi suivi mes avancements et apporté de multiples conseils, Judith CARDINALE et Dominique GARRIDO.

Et pour finir, je tiens à remercier toute l'équipe au sein de l'université, et aussi toute l'équipe pédagogique qui m'ont suivi jusque-là, sans qui ce stage et ce projet n'aurait pas pu avoir lieu.

## Table des matières

FICHE SYNTHESE DU STAGE .....	5
ABSTARCT .....	6
INTRODUCTION.....	7
I. Présentation de l'environnement.....	8
A. Description du lieu .....	8
B. Description de l'université et du pôle.....	9
C. Objectifs du rapport.....	9
II. Objectif du stage et problématique .....	10
A. Mise en contexte.....	10
B. Problématique.....	10
C. Mission .....	10
III. Plannings prévisionnels .....	11
A. Organisation du stage.....	11
B. GANTT prévisionnel.....	11
IV. Tâches effectuées et déroulement du stage.....	13
A. Base du projet.....	13
1. Système électronique .....	13
2. Code en Arduino .....	14
3. La base de données .....	15
4. Code en python .....	17
5. Code web .....	20
B. Amélioration du système.....	23
1. Système électronique .....	23
2. Code en Arduino .....	27
3. Code en Python et base de données .....	29
4. Code web .....	31
C. Complément au projet.....	37
1. Fonctionnement interne .....	37
2. Notice .....	38
3. Intégration d'un écran lcd.....	42
4. Piste de recherche .....	43
V. Impact socio-écologique .....	44
VI. Analyse budgétaire du projet .....	45
VII. Comparaison planning prévisionnel et planning effectif :.....	45
VIII. Rendu finale du projet.....	45
CONCLUSION .....	47
A. Conclusion du projet .....	47
B. Conclusion personnelle.....	47
ANNEXES.....	48
Annexe A : CV .....	48
Annexe B : Planning détaillé de la mission de stage.....	49
Annexe C : Références bibliographiques utilisées, sources internes et externes .....	50
Annexe E : Table des figures et illustrations utilisées dans le rapport.....	51
Annexe D : Programmes .....	52

# FICHE SYNTHÈSE DU STAGE

## Sujet de stage

Imaginer un système constitué de différents capteurs, intégrant le sujet « *Framework para la construcción de soluciones inteligentes en ambientes de Computación Ubicua aplicando modelos de Machine Learning* ». Récupérer ces données, les stocker et les afficher en vue d'utilisation ou d'analyses futures.

## Entreprise

Université de Valparaiso, Gral. Cruz 222, 2362905 Valparaíso, au Chili. Le stage a été effectué au sein du pôle informatique de cette école, en coordination avec Mme AGUILERA FARACO Ana, ma maître de stage.

## Date de stage

Ce stage a été effectué du 17 Juillet 2023 au 20 Novembre 2023, pour une durée totale de 17 semaines.

## Réalisation

Réalisation d'un système complet, constitué d'une partie hardware avec de l'Arduino et de multiples autres composants, que l'on verra dans ce rapport ci-dessous : Une partie software, avec d'une part un code sous python afin de transférer les données acquises via mon système hardware à une base de données en MySql.

Et pour finir, une page web codée en différents types de langages : CSS, HTML, PYTHON, JAVASCRIPT.

## ABSTRACT

Nowadays, the use of sensors is increasingly present in our daily lives. That's why today, I'm presenting this report of my work at the University of Valparaíso in Chile. This work was carried out in full coordination with Mrs AGUILERA FARACO Ana, Mrs Judith CARDINALE and Mrs Dominique GARRIDO.

Today, the problem is to imagine a system made up of different sensors, integrating the subject "Framework for the construction of intelligent solutions in computing environments using Machine Learning models". uploading data, storing, and displaying them for future use or analysis.

That's why, during my 4-month internship with the university's IT department, I created this system from scratch. I imagined the possibility of using this system in a closed environment, such as a classroom or a home.

I started from scratch, knowing that I only had Arduino hardware tools, and I wanted to create a whole ecosystem around them.

That's why I started my task looking into storing this data in a dedicated database.

Then, the aim of my mission was to analyze this data. That's why I decided to create a dashboard to visualize this data graphically.

This internship report will be the detailed presentation of this project, with a first part consisting of the basis of the project, from where I started.

The second part is the final part of the project, with all the improvements I've made to it.

From a personal point of view, I'm quite satisfied with the support I've given within the topic "Framework para la construcción de soluciones inteligentes en ambientes de Computación Ubicua aplicando modelos de Machine Learning", as the result is fully functional.

Additionally, in the last paragraph, you'll see project's prospects, and all the things I wasn't able to do due to lack of time and resources.

I hope I've been concise in this abstract, and I hope you enjoy reading this report.

## INTRODUCTION

L'objectif de ce rapport est de constituer un résumé de toutes les tâches effectuées au sein de mon stage ainsi que les connaissances acquises, tout au long de ces 4 mois effectués au sein de l'université de Valparaiso, Chili.

Dans un premier temps, je vais parler de l'environnement dans lequel j'ai effectué mon stage, en commençant par le pays et son histoire, puis présenter l'université et la ville où j'ai pu effectuer ce stage.

Dans un second temps, j'expliquerai en détail la problématique de mon stage, et mes objectifs avec une quantification des impacts de mon projet.

Dans un troisième temps, j'exposerai mon planning prévisionnel pour mener à bien ma mission.

Je présenterai par la suite, le déroulement de mon stage, avec les tâches effectuées au fur et à mesure de la mission, les difficultés rencontrées, et les solutions apportées à celles-ci.

Et, je finirai par une conclusion, que ce soit du point de vue professionnel ou personnel de mon stage.

En supplément de ce rapport, de multiples annexes seront disponibles afin d'illustrer mes propos et d'apporter de multiples informations supplémentaires sans surcharger ce rapport.

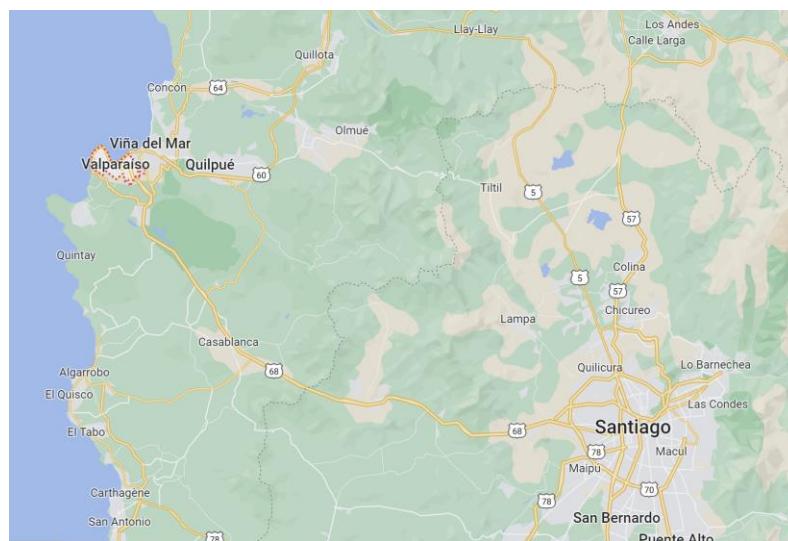
## I. Présentation de l'environnement

### A. Description du lieu



**IMG-1 Le Chili sur la carte du monde**

Mon stage s'est déroulé au Chili qui est un pays situé sur la côte ouest de l'Amérique Latine. Pour son histoire, le Chili était avant la conquête Espagnole, menée par Pedro de Valdivia, un peuple Mapuche et Inca. Le Chili a été dès lors sous la pression espagnole pendant près de trois siècles, pour ensuite connaître son indépendance en 1818 après une guerre d'indépendance. Fut ensuite un coup d'état militaire dans les années 1970, qui renverse le gouvernement démocratique de Salvador Allende pour installer une dictature sous Augusto Pinochet. La démocratie a été rétablie en 1990. Depuis celle-ci, le Chili a connu de nombreuses réformes politiques et sociales, et son économie est en forte croissance.



**IMG-2 Valparaiso**

Mon stage s'est déroulé plus particulièrement dans la ville de Valparaiso qui est une ville côtière située à 1H30 de Santiago, la capitale.

## B. Description de l'université et du pôle



**IMG-3 L'université**

Dans la ville se situe plusieurs pôles universitaires, chacun regroupant différents domaines, moi je me trouvais dans celui, indiqué ci-dessus, qui était à la faculté d'ingénierie.

Je faisais partie du pôle informatique de l'université dirigé par Roberto Muñoz. J'ai été accueilli par Mme AGUILERA FARACO Ana, professeur à l'université, ma maître de stage. Je travaillais dans une salle où était présent professeurs et étudiants en stage, avec qui je pouvais partager et échanger des connaissances.

## C. Objectifs du rapport

L'objectif de ce rapport est donc de constituer un résumé de toutes les tâches effectuées au sein de mon stage ainsi que les connaissances acquises tout au long des 4 mois effectués au sein de l'université de Valparaiso, au Chili.

## II. Objectif du stage et problématique

### A. Mise en contexte

De nos jours, l'utilisation de système autonome et l'analyse de données sont de plus en plus nécessaires dans nos vies quotidiennes. On a pu le voir avec l'apparition de la domotique qui se répand de plus en plus dans notre quotidien et, qui d'une part nous aide de plus en plus dans notre confort personnel, avec des tâches manuelles qui sont facilitées, ou totalement automatisées. De plus, la plupart de nos systèmes actuels sont optimisés pour consommer le moins d'énergie possible, ce qui réduit le coût pour le client, et l'impact sur l'environnement.

C'est dans ce contexte d'utilisation de capteurs dans nos vies quotidiennes et le traitement de ces données que je vais effectuer ma mission.

Mon objectif est d'imaginer ce système, le mettre en place, acquérir les données dans une base de données et créer un Dashboard qui permettrait la visualisation de celle-ci, et l'analyse si possible.

J'ai ainsi pensé à un système qui pourrait s'adapter à des environnements intérieurs : maison, salle de classe, chambre, pour respecter un environnement stable et sans danger pour les personnes présentes.

### B. Problématique

Dans le cadre du projet d'investigation '*Framework para la construcción de soluciones inteligentes en ambientes de Computación Ubicua aplicando modelos de Machine Learning*' dirigé par Mme. Ana Aguilera Faraco et son étudiante Dominique Garrido Araya, ma mission a été d'imaginer un système complet composé de plusieurs capteurs, permettant à l'homme d'intervenir dessus, soit dans un but futur de machine learning pour le traitement de données et son analyse; ou dans un but plus simple d'automatisation et l'utilisation de la domotique en complément de celui-ci.

### C. Mission

Dans un premier temps, ma mission a été d'imaginer un système de fonctionnement sur une base stable permettant l'acquisition de données, ainsi que le stockage de celles-ci en vue de leurs traitements.

Une fois, cette première problématique solutionnée, mon objectif a été d'apporter de nettes améliorations au système. C'est-à-dire d'une part, l'utilisation d'un maximum de capteurs pour ainsi avoir le maximum de données possibles, mais aussi des améliorations externes, comme la création d'un Dashboard ou tout autre complément qui apporterait un ou des atouts supplémentaires à ce système.

### III. Plannings prévisionnels

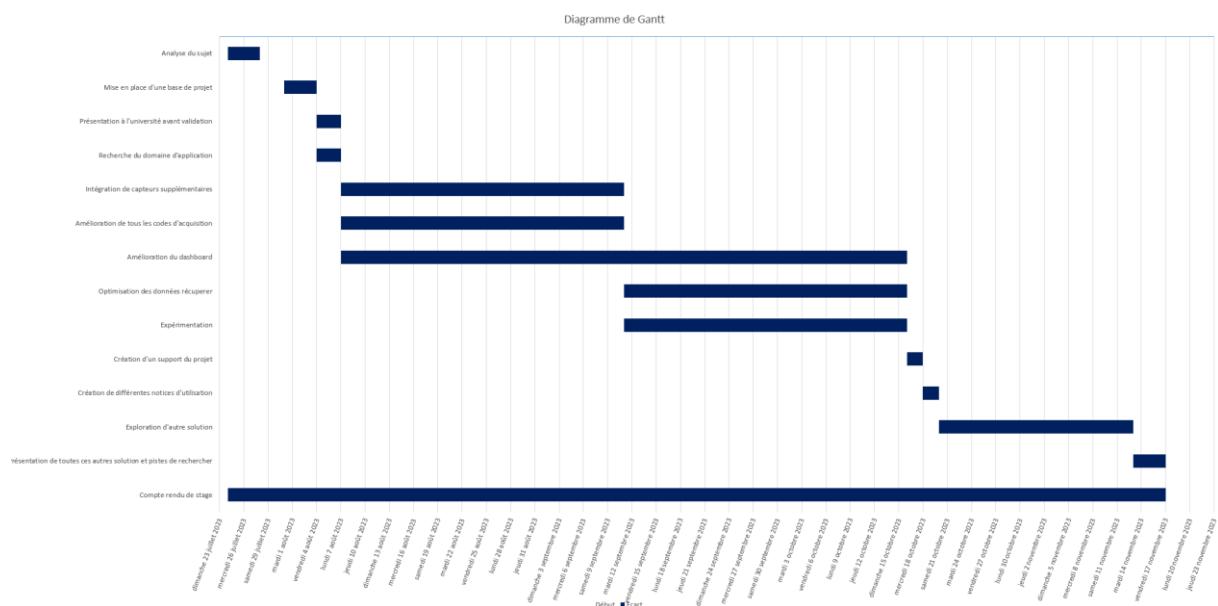
Comme dit précédemment, mon objectif a été vraiment de dimensionner et de créer un système constitué de capteurs, avec tout ce qu'il y a autour. De ce fait, j'avais une grande liberté quant à la création de celui-ci.

#### A. Organisation du stage

Pour ce qui s'agit de l'organisation du stage, j'avançais et je proposais des solutions tout au long de mon stage. Toutes les semaines, une réunion était organisée avec tous les intervenants liés au projet afin que je leur présente mes avancements. Ceci m'a beaucoup aidé notamment dans la critique; car grâce à cela j'ai pu visualiser ce qui était bien ou non. De plus, ma maître de stage Ana Aguilera Faraco était tout le temps disponible pour le moindre questionnement et m'aider à mener à bien ma mission.

#### B. GANTT prévisionnel

Pour ce qui est du planning prévisionnel, ne sachant pas trop dans quoi je partais exactement, j'ai dû voir dans les grandes lignes, c'est-à-dire que mon gantt constitué non-pas les tâches faites précisément mais le groupe de tâches effectuées.



Celui-ci me permet alors, de pouvoir situer périodiquement si sur mon futur projet je resterai dans les temps qui sont prévu ou non.

### **Objectifs SMART :**

#### **Objectif spécifique**

Etablir un système constitué de capteur, pouvant stocker les données de ceux-ci et les afficher sur un dashboard fonctionnel.

#### **Objectif mesurable**

Les données sortant de mes capteurs doivent être au plus proche de la réalité soit par un étalonnage constructeur, ou un étalonnage manuel.

#### **Objectif atteignable**

Utiliser des capteurs approprier à l'environnement contraint. Création d'un système de base fonctionnel, ne serait-ce qu'avec un seul capteur.

#### **Objectif atteignable**

Acquisition de différentes données de plusieurs capteur, stockage dans une base de données et affichage sur un dashboard crée de toute pièce.

#### **Temporellement défini**

La durée de stage est de 4 mois rédaction du rapport compris. Je me donne alors 3 mois et demi, pour pouvoir finaliser la tâche qui m'a été demandé.

## IV. Tâches effectuées et déroulement du stage

### A. Base du projet

Dans cette première partie des tâches effectuées et du déroulement du stage, nous allons voir ce que j'ai nommé la base du projet. C'est-à-dire tous les éléments sur lesquels mon projet va se reposer. D'une part, nous aurons la base du système électronique, puis tous les différents codes associés. Il faut comprendre que dans cette partie, les explications techniques seront plus détaillées, mais non-constituées de tous les éléments finaux. Je vais essayer d'exprimer au maximum, point par point, mes démarches afin d'aboutir à cette base de système ainsi que les différentes difficultés rencontrées.

#### 1. Système électronique

Pour la partie électronique, le but était de trouver un système pouvant à la fois acquérir les données de capteurs et les transférer à mon ordinateur.

Pour ce faire, l'université avait à disposition des systèmes Arduino, j'ai donc décidé de les intégrer à mon système.

Arduino est une marque de composant électronique. Ces cartes sont directement compatibles avec le logiciel de programmation nommée « Arduino » se rapprochant du langage C, se rédigeant directement sur leur logiciel « Arduino ».

Ainsi, l'utilisation de l'Arduino reste très simpliste, mais efficace.

Le kit proposé était celui-ci produit par SparkFun (sous marque de Arduino), mais ayant en tout point les mêmes caractéristiques :

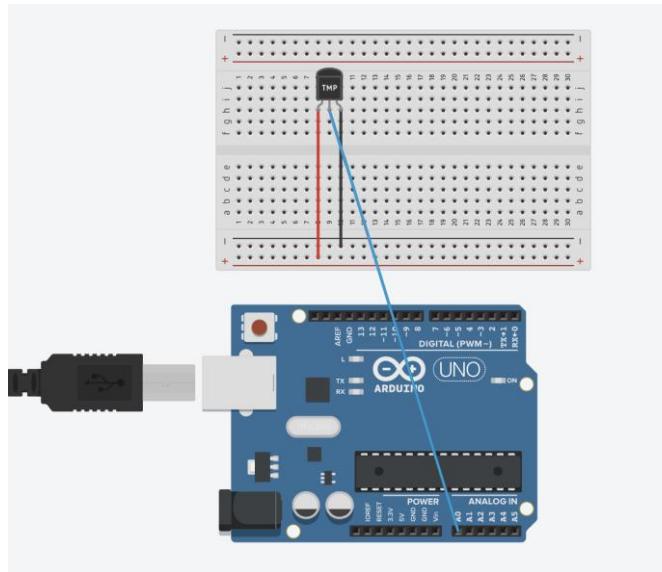


**IMG-4 Ensemble des composants de base**

Celui-ci était composé de différentes composantes, ici pour mon projet, j'en ai retenu seulement quelques-unes qui sont :

- La carte SparkFun (type Arduino uno)
- Les différents capteurs (température, photorésistance)
- Cable alimentation, résistance, file, led, buzzer

Une fois le système électronique dimensionné, je n'avais plus qu'à effectuer les branchements nécessaires :



Dans un premier temps, j'ai commencé par un branchement assez simple pour tester le code qui va suivre. Nous aurons alors pour l'instant simplement le capteur de température branché comme ci-dessus.

## 2. Code en Arduino

Pour la partie hardware, comme cité précédemment, le code qui sera envoyé à la carte s'effectuera sur le logiciel Arduino qui se présente comme ceci :

```
sketch_oct19a | Arduino 1.8.16
Fichier Édition Croquis Outils Aide
sketch_oct19a
void setup() {
  // put your setup code here, to run once:
}

void loop() {
  // put your main code here, to run repeatedly:
}
```

Nous avons une partie « `setup` », où je vais pouvoir initialiser toutes les composantes de mon système, et une partie « `loop` » qui va être mon code qui s'effectuera en boucle.

Grâce à cela, je peux procéder à l'incrémantation du code dans la carte Arduino, ainsi que le débogage du code et la détection d'erreur.

Le fonctionnement pour tous les capteurs qui seront alors ajoutés à mon système, seront à quelque chose près, très similaires.

Voici un premier exemple avec le code pour le capteur de température :

```
sketch_nov11a §
void setup() {
    pinMode(A0, INPUT);
    Serial.begin(9600);
}

void loop() {
    Serial.println(A0);
}
```

On peut voir dans un premier temps que j'initialise mon entrée analogique en « input ». Ensuite, j'initialise mon moniteur de série à 9600 bauds, chose qui sera importante par la suite pour me connecter via mon code en python à mon moniteur de série Arduino. Une fois fait, je peux y ajouter des variables associant une équation d'étalonnage. Ici, le capteur de température constitue une équation constructeur ce qui me permet directement, sans avoir à faire d'étalonnage manuellement, d'avoir la valeur de la température :

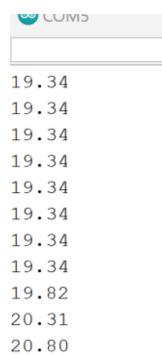
```
void setup() {
    pinMode(A0, INPUT);
    Serial.begin(9600);
}

void loop() {
    float voltage = analogRead(A0) * 0.004882814;
    float temperature= (voltage-0.5)*100.0;

    Serial.println(temperature);

    delay(1000);
}
```

Une fois le code compilé et envoyé dans la carte, tout fonctionnait parfaitement bien comme on peut le voir ci-dessous. Les valeurs de la température s'affichaient bien dans mon moniteur de série :



The screenshot shows a terminal window titled "LUVIVIS" with the following text output:

```
19.34
19.34
19.34
19.34
19.34
19.34
19.34
19.34
19.82
20.31
20.80
```

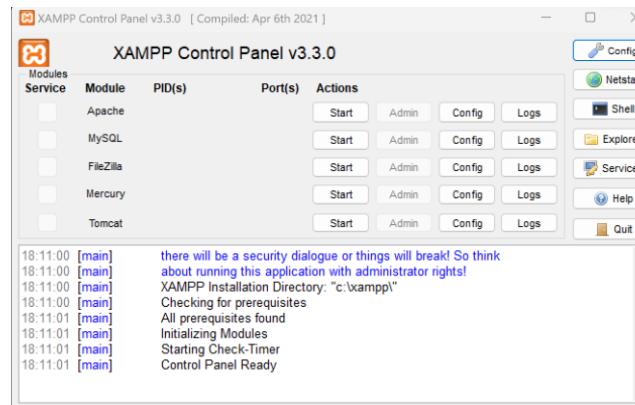
### 3. La base de données

Maintenant que les données sont envoyées dans mon ordinateur, il me fallait une solution pour les stocker. Durant mes études, j'ai pu voir les machines virtuelles sous Apache, ainsi que les bases de données en MySql. J'ai donc voulu réutiliser ces compétences afin de stocker mes données arrivant de mes capteurs.

La contrainte est que je voulais une manière simple de création de la base de données et de la machine virtuelle, afin de faciliter la méthode d'installation et d'utilisation pour tout utilisateur du projet.

J'ai alors commencé des recherches sur un système qui me permettrait facilement de lancer une machine virtuelle et une base de données, afin de simplifier l'utilisation pour les utilisateurs futurs.

Pour ce faire, je suis tombé sur le logiciel XAMPP, développé par Apache Friend. Celui-ci se présente comme ceci :

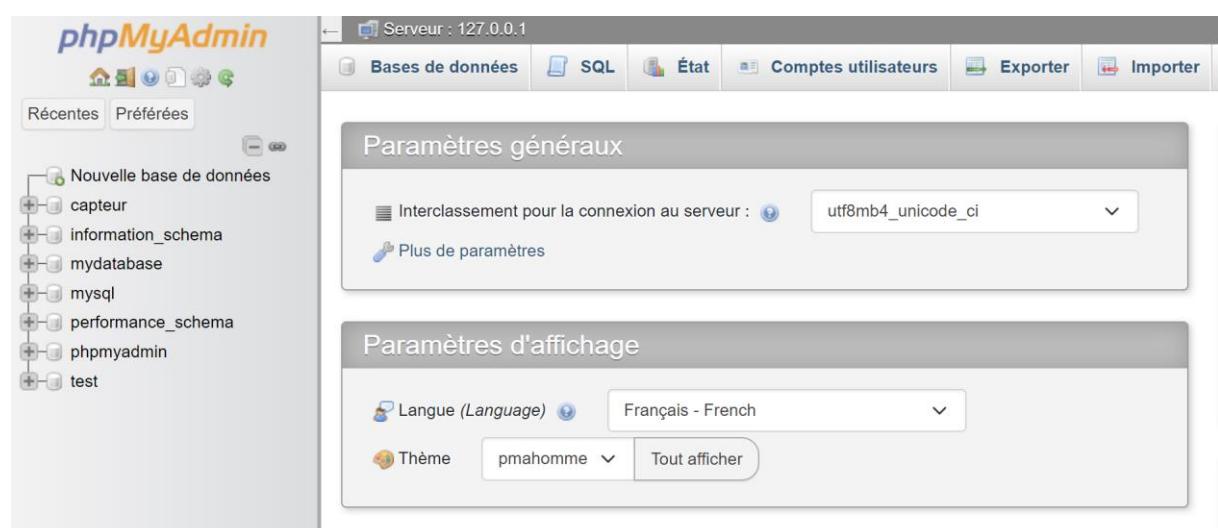


Ici, deux parties vont nous intéresser : D'une part, la machine virtuelle sous Apache et d'autre part la base de données en MySql.

Pour ce qui en est de son utilisation, elle est assez simple. Il suffit de démarrer les deux outils en appuyant sur le bouton 'start', ainsi la machine virtuelle se lance et la base de données aussi.

Pour accéder à cette base de données, deux options s'offrent à nous. La première est seulement d'appuyer sur bouton 'admin' du logiciel. La deuxième est d'ouvrir n'importe quel navigateur web et d'aller sur l'adresse ci-contre : <http://localhost/phpmyadmin/>

La base de données se présente comme ceci :



A présent, je peux créer une base de données que j'ai nommé ici « mydatabase ». Dans celle-ci, j'ai créé une table nommée « mydata » contenant plusieurs variables. Dans un premier temps, une valeur de type intégrale en tant que primary key pour connaître la combien -ième donnée a été acquise. Ensuite, une valeur dateA et hoursA pour avoir l'horaire et le jour d'acquisition de la donnée. Et pour finir, toutes les données de mes capteurs que je souhaite stocker en VARCHAR(20). Ici nous allons commencer par seulement y intégrer la valeur de notre capteur de température.

Voici les commandes de création de tous les éléments précédemment cités :

**➔ Celle de la base de données**

```
CREATE DATABASE mydatabase;
```

**➔ Celle de la table dans cette base de données**

```
CREATE TABLE mydata (
    id INT AUTO_INCREMENT PRIMARY KEY,
    dateA DATE,
    hoursA TIME,
    temperature VARCHAR(20)
);
```

#### 4. Code en python

Ce code-ci sera le code qui va nous servir à connecter tous nos éléments entre eux. J'ai d'une part des données qui arrivent dans mon moniteur de série dans un format ASCII (American Standard Code for Information Interchange), qui est considéré comme du texte; et d'autre part, ma base de données qui attend d'acquérir les données présentes dans mon moniteur de série.

Pour effectuer ce transfert, je vais me servir d'un programme en Python qui est un langage multiplateforme qui facilite la programmation impérative structurée.

Voici comment le code se présente :

```

1 import mysql.connector as msqlc
2 from datetime import date, datetime
3 import serial
4 import time
5
6 try:
7     bd = msqlc.connect(
8         host = "localhost",
9         port = 3306 ,
10        user = "root",
11        passwd = "",
12        database = "mydatabase"
13    )
14 except Exception:
15     print("connexion impossible à la base de données")
16
17 cursor = bd.cursor()
18 query = "INSERT INTO mydata (dateA, hoursA, temperature) VALUES (%s, %s, %s)"
19
20 with serial.Serial('COM5', 9600, timeout=0.05) as comarduino:
21     while True:
22         data = comarduino.readline()
23         print(data.decode('utf-8'))
24         values = data.decode('utf-8').strip()
25         now = datetime.now()
26         dateA = now.strftime("%Y-%m-%d")
27         hoursA = now.strftime("%H:%M:%S")
28
29         # Enregistrement des valeurs dans la base de données
30         cursor.execute(query, (dateA, hoursA, values))
31
32         bd.commit()
33
34         time.sleep(2)

```

Dans un premier temps, j'initialise toutes les bibliothèques dont j'aurai besoin. C'est-à-dire celles nécessaires pour me connecter à ma base de données; ensuite celle pour la connexion avec le moniteur de série à 9600 bauds et pour finir la récupération de la date et l'heure au moment de la récupération des données.

Ensuite, je commence par essayer de me connecter à ma base de données, avec tous les éléments nécessaires à la connexion à celle-ci : le nom du host, le mot de passe, le port et l'utilisateur.

Ici j'ai décidé de garder les paramètres de base du logiciel XAMPP. Si on le souhaite, on peut les modifier simplement grâce au logiciel XAMPP.

Et pour finir, je tente de me connecter à ma base de données myDataBase. Si cette connexion ne se fait pas, je lève une exception.

A la ligne 18, je passe à ma commande qui va être envoyée dans ma base de données en mySql : "INSERT INTO Data ( dateA, hoursA, temperature ) VALUES (%s,%s,%s)"

Celle-ci va me permettre d'insérer dans ma table « mydata », la date, l'heure et la valeur de mon capteur de température. Les valeurs « %s » sont les variables que je vais devoir compléter respectivement pour appeler ma fonction.

Pour finir de la ligne 20 à 34, j'ai ma boucle qui va dans un premier temps, se connecter à mon moniteur de série (9600 bauds) ; puis dans un second temps, lire la ligne arrivant sur celui-ci et la décoder, celle-ci arrivant sous forme binaire (0 et 1), grâce à l'encodage UTF-8 sur 8 bits je vais pouvoir décoder ASCII. On arrive alors à retrouver une chaîne de caractère.

Pour finir, je récupère la date et l'heure d'acquisition en les remettant au bon format, de type année/mois/jours et heures/minutes/seconde.

Une fois toutes mes data acquises, il me suffit juste de pointer mon curseur sur la base de données et de faire appel à ma fonction présentée ci-dessus en y intégrant la date, l'heure, et la donnée de mon capteur de température.

Après exécution de mon code en Python, mes données arrivent bien dans ma base de données :

<input type="checkbox"/> Tout afficher	Nombre de lignes :	25	Filtrer les lignes:	Chercher dans cette table	Trier
Options supplémentaires					
		id	dateA	hoursA	temperature
<input type="checkbox"/>	Éditer	Copier	Supprimer	73 2023-11-11 05:35:36	19.34
<input type="checkbox"/>	Éditer	Copier	Supprimer	74 2023-11-11 05:35:38	18.85
<input type="checkbox"/>	Éditer	Copier	Supprimer	75 2023-11-11 05:35:40	18.85
<input type="checkbox"/>	Éditer	Copier	Supprimer	76 2023-11-11 05:35:42	18.85
<input type="checkbox"/>	Éditer	Copier	Supprimer	77 2023-11-11 05:35:44	18.85
<input type="checkbox"/>	Éditer	Copier	Supprimer	78 2023-11-11 05:35:46	19.34
<input type="checkbox"/>	Éditer	Copier	Supprimer	79 2023-11-11 05:35:48	18.85
<input type="checkbox"/>	Éditer	Copier	Supprimer	80 2023-11-11 05:35:50	19.34
<input type="checkbox"/>	Éditer	Copier	Supprimer	81 2023-11-11 05:35:52	18.85
<input type="checkbox"/>	Éditer	Copier	Supprimer	82 2023-11-11 05:35:54	18.85

Tout cocher      Avec la sélection :      Éditer      Copier      Supprimer      Exporter

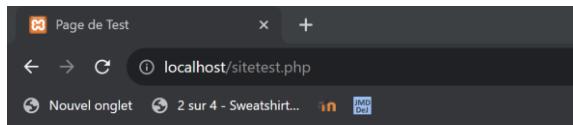
Une fois le code fini, j'ai rencontré quelques soucis, notamment des conflits entre le programme Arduino et le code en Python. Ceux-ci ne pouvaient pas fonctionner simultanément car le port n'acceptait qu'un logiciel de lecture.

Il faut donc bien prêter intention à ce que le logiciel Arduino soit complètement fermé avant de lancer toute tentative de lancement du code en Python.

## 5. Code web

Une fois, mes données arrivant au fur et à mesure dans ma base de données, et cela, de façon fonctionnelle, j'avais pour ambition de passer à la création de mon propre DashBoard

Dans un premier temps, j'ai effectué un test pour vérifier que ma machine virtuelle fonctionnait correctement et qu'une page web s'affichait. J'utilise le logiciel Visual Studio Code pour programmer une page simple en html. Une fois la page créer, il suffisait de glisser le fichier du dossier XAMMP > htdocs. Voici ci-dessous ce que cela donne :



```
C: > xampp > htdocs > sitetest.php > ...
1  <!DOCTYPE html>
2  <html>
3  <head>
4  |   <title>Page de Test</title>
5  </head>
6  <body>
7  |   <h1>Bonjour</h1>
8  </body>
9  </html>
```

Une fois la vérification faite, je suis passé au code lié à mon projet. Pour le moment, je n'avais que la date, l'heure d'acquisition et la température dans ma base donnée. Mon premier objectif a été de récupérer ces valeurs pour les associer à des variables dans code web.

Voici le code en question :

```

1  <?php
2  $host = 'localhost';
3  $user = 'root';
4  $password = '';
5  $database = 'mydatabase';
6
7
8  $conn = new mysqli($host, $user, $password, $database);
9
10 if ($conn->connect_error) {
11     die('Connection failed: ' . $conn->connect_error);
12 }
13
14 $sql = 'SELECT dateA, hoursA, temperature FROM myData';
15 $result = $conn->query($sql);
16
17 $data = array(
18     'date' => array(),
19     'temperature' => array(),
20 );
21 );
22
23 if ($result->num_rows > 0) {
24     while ($row = $result->fetch_assoc()) {
25         $datetime = $row['dateA'] . ' ' . $row['hoursA'];
26         array_push($data['date'], $datetime);
27         array_push($data['temperature'], $row['temperature']);
28     }
29 }
30 }
31
32 $conn->close();
33 header('Content-Type: application/json');
34 echo json_encode($data);
35 ?>

```

Dans un premier temps, on peut voir de la ligne 1 à 8 que j'essaye, comme avec le code en Python de me connecter à ma base de données. De la ligne 10 à 12, je lève une erreur si la connexion n'est pas possible.

Ensuite, je passe à la création de mes fonctions qui vont me permettre la récupération de mes données en MySql, de la ligne 14 à 15.

J'initialise ensuite un tableau data de la ligne 17 à 21, avec comme variable deux clés qui sont la date et la température. Ce tableau me permettra de stocker les données qui arrivent.

De la ligne 23 à 30, je passe à la création d'une boucle qui se déclenche que lorsque des nouvelles données arrivent dans ma base de données d'où la condition \$result->num\_rows. Si celle-ci est validée, alors mes données sont envoyées à chaque itération de ma boucle dans la variable \$row. Et j'associe les données reçues à mes données dans \$data.

Pour finir, je ferme ma connexion à ma base de données, et j'encode celle-ci sous json qui permet une récupération plus rapide avec mon programme web.

Mes données maintenant présentes dans une variable \$data, il ne me fallait plus qu'à les utiliser pour créer un graphique simple.

Voici le premier code que j'ai utilisé :

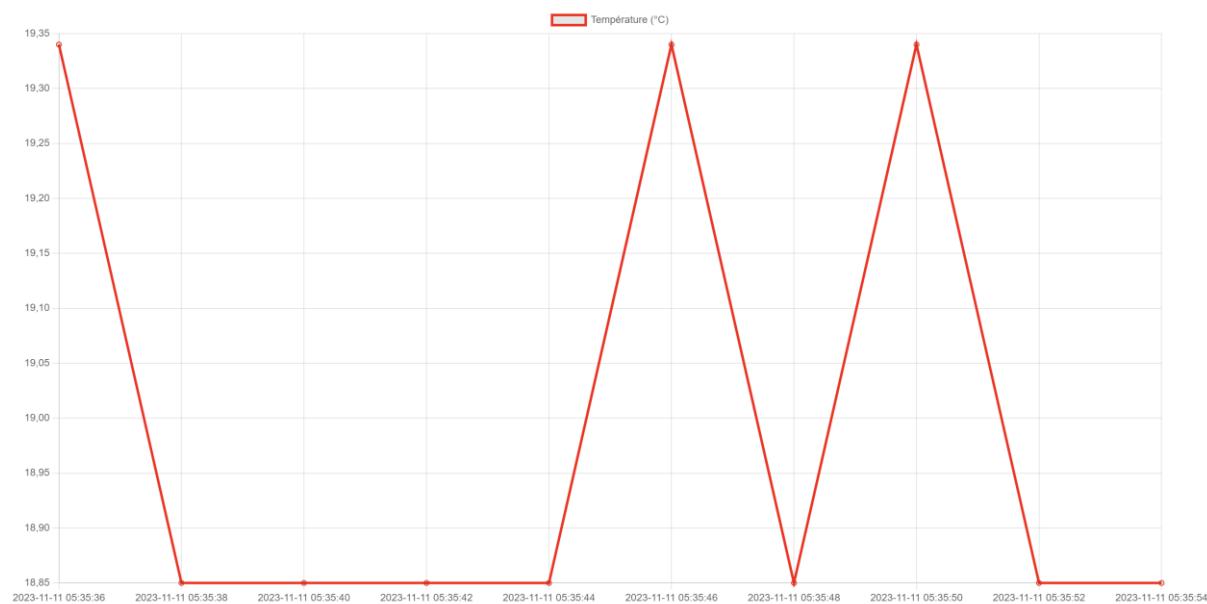
```

1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>Mon graph</title>
5      <script src="https://cdn.jsdelivr.net/npm/chart.js"></script>
6
7  </head>
8  <body>
9      <div class="container">
10         <canvas id="temperatureChart"></canvas>
11     </div>
12     <script>
13         fetch('data.php')
14             .then(response => response.json())
15             .then(data => {
16                 const dates = data.date;
17                 const temperatures = data.temperature;
18
19
20
21                 var temperatureChart = new Chart(document.getElementById('temperatureChart').getContext('2d'), {
22                     type: 'line',
23                     data: {
24                         labels: dates,
25                         datasets: [
26                             {
27                                 label: 'Température (°C)',
28                                 data: temperatures,
29                                 borderColor: 'red',
30                                 fill: false
31                             }
32                         ]
33                     }
34                 });
35             });
36         </script>
37     </body>
38 </html>

```

Pour ce faire, j'utilise la librairie <https://cdn.jsdelivr.net/npm/chart.js>, qui va me permettre d'afficher des graphiques simplement grâce à la fonction Chart. Je vais commencer dans un premier temps par récupérer les datas que j'ai pu acquérir dans la liste températures ainsi que la date et l'heure d'acquisition. Ensuite j'initialise mon graphique avec en ordonnée la température et en abscisse la date et l'heure d'acquisition.

Voici le résultat :



Le résultat est satisfaisant. Nous avons bien la température acquise au cours du temps.

## B. Amélioration du système

Maintenant que j'ai une bonne base de projet, je vais montrer dans cette partie toutes les améliorations effectuées à mon projet. Je ne vais pas reprendre le fonctionnement de chaque outil, car celui-ci a déjà été vu précédemment, mais je vais voir tous les ajouts effectués dans le détail avec chaque composantes et leurs spécifications associées.

### 1. Système électronique

Le projet étant d'acquérir un maximum de données, la partie hardware du projet connaît une amélioration conséquente.

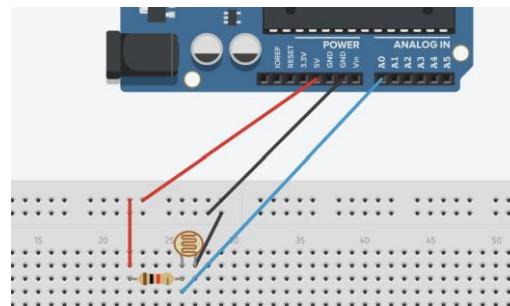
Pour répondre à cette problématique d'utiliser un maximum de capteurs, j'ai effectué mes recherches sur d'une part les capteurs compatibles avec arduino; et d'autre part les capteurs qui pourraient acquérir des données, qui seraient alors utiles à mon projet de : bien être dans un environnement fermé.

Pour ce faire, après de nombreuses recherches j'ai retenu ces capteurs :

- Une photorésistance :

Une photorésistance est une résistance qui varie en fonction de la luminosité extérieure, celle-ci position en tant que deuxième résistance d'un pont diviseur de

tension nous permet d'en ressortir une valeur en « volt » variant en fonction de la luminosité, voici comment elle est câblée :



On a donc d'une part une résistance R1 fixe, et d'une autre part une photorésistance R2 qui varie en fonction de la luminosité externe. En entrée 5V et en sortie la masse 0V.

$$\text{Donc } dataLum = 5 \frac{R2}{R1+R2}$$

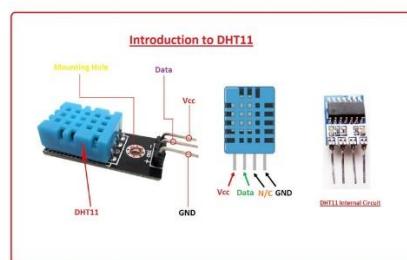
Grâce à cette résistance R2 qui varie en fonction de la luminosité, on a donc dataLum qui varie en fonction de la luminosité. De ce fait, en la branchant à une entrée analogique, on peut en déduire via une équation d'étalonnage, la valeur précise de la luminosité externe.

Malheureusement, faute de matériel, je n'avais pas à disposition de luxmètre afin de pouvoir étalonner la photorésistance. Nous allons donc avoir, simplement l'analyse de la variation de la luminosité au cours du temps.

- Un capteur de température et humidité :

Ici, j'ai utilisé le capteur DHT11, qui est un capteur contenant une librairie directement sur Arduino qui nous permet d'acquérir nos valeurs de température et d'humidité sans besoin d'étalonnage.

Celui-ci se présente comme ceci :



IMG-2 Capteur DHT11

Le branchement reste assez simple, nous avons simplement une entrée 'un' +5V, notre masse, la mise à zéro ; et pour finir notre valeur data à brancher sur n'importe quelle broche de notre carte Arduino.

- Un capteur d'UV ML8511 :

Ici, nous avons un capteur UV (ultraviolet), l'objectif de celui-ci est d'en sortir un indice UV.

Voici les risques des rayons UV sur l'homme :

Indice UV	Description	Mesures de protection contre le soleil
0 - 2	Bas	<ul style="list-style-type: none"> <li>• Protection solaire minimale requise pour les activités normales</li> <li>• Portez des lunettes de soleil les journées ensoleillées. Si à l'extérieur pendant plus d'une heure, couvrez-vous et utilisez un écran solaire</li> <li>• La réflexion par la neige peut presque doubler la force des rayons UV.</li> </ul>
3 - 5	Moderé	<ul style="list-style-type: none"> <li>• Prenez des précautions – couvrez-vous, portez un chapeau et des lunettes de soleil, et appliquez un écran solaire – surtout si vous êtes à l'extérieur pendant 30 minutes ou plus</li> <li>• Cherchez l'ombre en mi-journée quand le soleil est à son plus fort</li> </ul>
6 - 7	Elevé	<ul style="list-style-type: none"> <li>• Protection nécessaire – l'UV endommage la peau et peut causer des coups de soleil</li> <li>• Réduisez l'exposition au soleil entre 11 h et 16 h et prenez toutes les précautions : recherchez l'ombre, portez un chapeau et des lunettes de soleil, et appliquez un écran solaire</li> </ul>
8 - 10	Très élevé	<ul style="list-style-type: none"> <li>• Précautions supplémentaires nécessaires : la peau non protégée sera endommagée et peut brûler rapidement</li> <li>• Évitez le soleil entre 11 h et 16 h et prenez toutes les précautions : recherchez l'ombre, couvrez-vous, portez un chapeau et des lunettes de soleil, et appliquez un écran solaire</li> </ul>
11 et plus	Extrême	<ul style="list-style-type: none"> <li>• Les valeurs de 11 ou plus sont très rares au Canada. Cependant, l'indice UV peut atteindre 14 ou plus dans les tropiques ou le sud des États-Unis.</li> <li>• Prenez toutes les précautions. La peau non protégée sera endommagée et peut brûler en quelques minutes. Évitez le soleil entre 11 h et 16 h, couvrez-vous, portez un chapeau et des lunettes de soleil, et appliquez un écran solaire.</li> <li>• Le sable blanc et les autres surfaces brillantes réfléchissent les UV et augmentent l'exposition aux UV.</li> </ul>

### IMG-6 Risque de l'UV

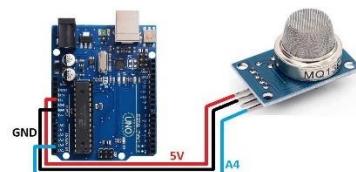
Celui-ci est donc nécessaire pour éviter tout problème et donc s'ajoute parfaitement dans la problématique du bien-être.



Celui-ci est composé de 5 ports. Ici seulement trois vont nous intéresser :

- VIN en 5V
- OUT, qui va nous permettre de récupérer notre valeur de l'UV
- GND la masse

- Capteur de CO2 MQ-135 :



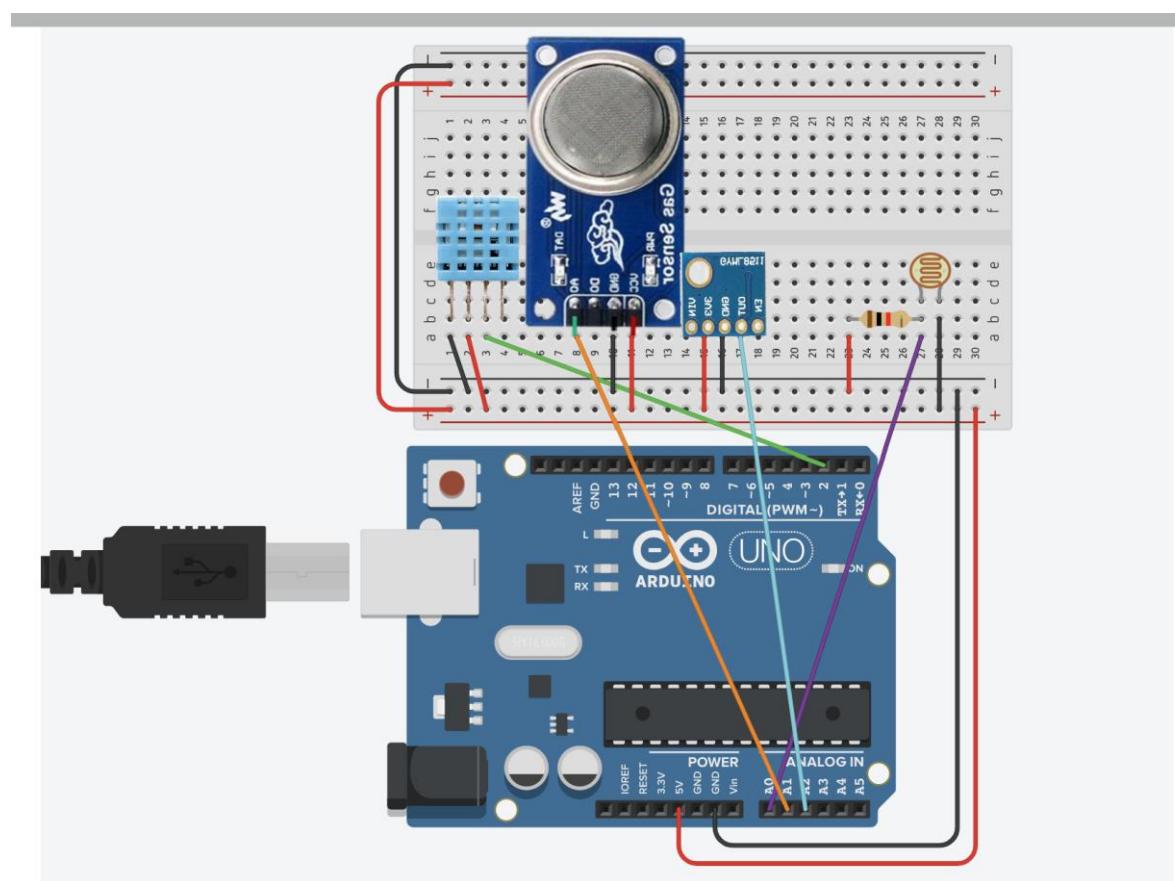
Ce capteur présenté reste assez similaire. Quant au branchement, on a le 5V, la masse et la sortie VOUT, que l'on branche sur un port analogique.

Celui-ci permet la détection de gaz polluant, le capteur MQ-135 est principalement utilisé pour détecter les gaz suivants :

Dioxyde de carbone (CO<sub>2</sub>), Monoxyde de carbone (CO), Ammoniac (NH<sub>3</sub>), Méthane (CH<sub>4</sub>), Propane (C<sub>3</sub>H<sub>8</sub>), Isobutane (C<sub>4</sub>H<sub>10</sub>), Hydrogène (H<sub>2</sub>), Dioxyde de soufre (SO<sub>2</sub>), Oxydes d'azote (NO<sub>x</sub>).

Le résultat des valeurs perçues par ce capteur est exprimé en ppm, partie par millions. Ce capteur va nous permettre de nous assurer que la pièce n'est pas polluée ou encore, pour une maison, qu'il n'y ait pas de fuite de gaz.

Maintenant, sachant que nous avons tous nos capteurs sélectionnés et analysés, voici dans la globalité, le schéma de tout le système hardware :



## 2. Code en Arduino

Pour la programmation de ma carte Arduino, je commence par y apporter toutes librairies nécessaires au bon fonctionnement de mon système :

```
#include "DHT.h"
#include <MQ135.h>
```

La librairie DHT.h s'associera à mon capteur de température et humidité.

La librairie MQ135.h sera quant à elle pour mon capteur de gaz polluant.

Je continue par initialiser toutes mes entrées et mes sorties, ainsi que tous les paramètres associés à mes librairies :

```
#define DHTPIN 2
#define DHTTYPE DHT11

const int luminositePin = A0;
const int airQualityPin = A1;
const int uvSensorPin = A2;

MQ135 gasSensor = MQ135(airQualityPin);
DHT dht(DHTPIN, DHTTYPE);

float voltage;
float degreesC;
float lux;
```

Dans mon setup, j'initialise tous mes ports analogiques en mode INPUT, ainsi que ma connexion à mon moniteur de série en 9600 bauds :

```
void setup() {

    pinMode(uvSensorPin, INPUT);
    pinMode(luminositePin, INPUT);
    dht.begin();
    Serial.begin(9600);
}
```

Dans ma boucle, j'initialise toutes mes équations ainsi que les commandes présentes dans mes librairies vues précédemment afin d'en ressortir des valeurs cohérentes pour toutes mes données comme ceci :

```

void loop() {
    //***** pour la température et humidité
    float temperature = dht.readTemperature();
    float humidity = dht.readHumidity();

    //*****pour le gas
    float airQualityValue = gasSensor.getPPM();

    //***pour la luminosité
    float luminosite = analogRead(luminositePin);
}

```

Pour la température, la luminosité et la qualité de l'air, cela reste assez simpliste car des commandes prédéfinies me permettent d'avoir directement leurs valeurs correspondantes coordonnées avec l'étalonnage constructeur.

Pour la luminosité, sachant que je n'avais pas de luxmètre à ma disposition afin de procéder à l'étalonnage de ma photorésistance, j'ai tout simplement affiché la valeur arrivant sur mon port analogique allant de 0 (pour 0V) à 1023 (pour 5V).

Pour finir, il ne me manque plus que le taux d'UV voici le code associé :

```

//** Pour les UV
lecture = analogRead(uvSensorPin);
int taux = map(lecture,0,669,0,1023);
if (taux<50){
    indice = 0;
}
else if (taux<227){
    indice = 1;
}
else if (taux<318){
    indice = 2;
}
else if (taux<409){
    indice = 3;
}
else if (taux<503){
    indice = 4;
}
else if (taux<606){
    indice = 5;
}
else if (taux<795){
    indice = 6;
}
else if (taux<881){
    indice = 7;
}
else if (taux<976){
    indice = 8;
}
else if (taux<1074){
    indice = 9;
}

```

Color	Índice	Rango	Tipo de riesgo
	0	<50	Ninguno
	1	<227	Bajo
	2	<318	
	3	<408	
	4	<503	Moderado
	5	<606	
	6	<696	Alto
	7	<795	
	8	<881	
	9	<976	Muy alto

IMG 4- Etalonnage UV

Pour rappel, ce capteur UV est branché en 3,3V, la sortie varie donc de 0 à 669. Malheureusement, j'ai à ma disposition simplement un étalonnage, présent sur l'image de droite, correspondant à un capteur dont la valeur analogique varie de 0 à 1023. Pour pallier ce problème, j'ai utilisé dans un premier temps le mapping pour passer toutes valeurs qui sont de 0 à 669 à 0 à 1023 et cela proportionnellement.

Une fois mes valeurs allant de 0 à 1023, il me suffit d'établir plusieurs conditions « if », afin d'avoir la valeur de l'indice UV correspondante.

Pour finir avec le code Arduino, j'envoie toutes mes données acquises dans mon moniteur de série, valeurs que je vais séparer par une virgule ce qui simplifiera plus tard le code en Python pour envoyer mes valeurs dans ma base de données.

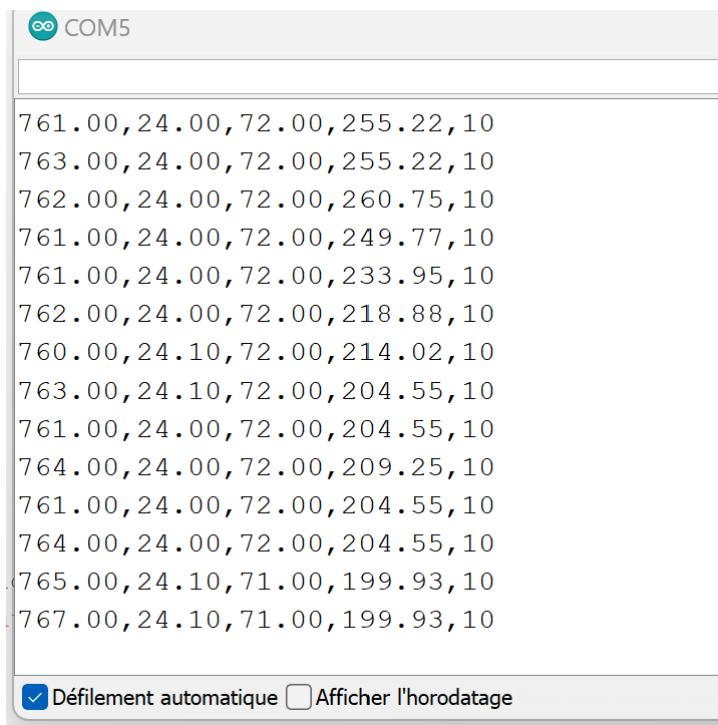
```

Serial.print(luminosite);
Serial.print(",");
Serial.print(temperature);
Serial.print(",");
Serial.print(humidity);
Serial.print(",");
Serial.print(airQualityValue);
Serial.print(",");
Serial.println(uvIntensity);

delay(1000);
}

```

Après vérification, toutes mes données arrivaient bien dans mon moniteur de série séparées par une virgule :



The screenshot shows the Arduino Serial Monitor window titled "COM5". The text area displays a series of sensor readings, each consisting of five values separated by commas. The readings are as follows:

```

761.00,24.00,72.00,255.22,10
763.00,24.00,72.00,255.22,10
762.00,24.00,72.00,260.75,10
761.00,24.00,72.00,249.77,10
761.00,24.00,72.00,233.95,10
762.00,24.00,72.00,218.88,10
760.00,24.10,72.00,214.02,10
763.00,24.10,72.00,204.55,10
761.00,24.00,72.00,204.55,10
764.00,24.00,72.00,209.25,10
761.00,24.00,72.00,204.55,10
764.00,24.00,72.00,204.55,10
765.00,24.10,71.00,199.93,10
767.00,24.10,71.00,199.93,10

```

At the bottom of the window, there are two checkboxes: "Défilement automatique" (checked) and "Afficher l'horodatage" (unchecked).

### 3. Code en Python et base de données

Le code en Python reste similaire au précédent. La différence ici, c'est que je n'ai pas qu'une valeur qui arrive dans mon moniteur de série, mais j'en ai 5. Voici le code en Python :

```

19 cursor = bd.cursor()
20 query = "INSERT INTO donnee (Date, hours, Temperature,Luminosity,Humidity,AirQuality,UVvalue) VALUES (%s, %s, %s, %s, %s,%s)"
21 #boucle qui enregistre les données
22
23
24 with serial.Serial('COM5', 9600, timeout=0.05) as comarduino:
25     while True:
26         data = comarduino.readline()
27         print(data.decode('utf-8'))
28
29         # les deux valeurs doivent être séparées d'une virgule dans le moniteur de série,
30         # ce qui nous permet ensuite de différencier la première de la deuxième
31         values = data.decode('utf-8').strip().split(',')
32         if len(values) == 6:
33             luminosite = values[0]
34             temperature = values[1]
35             humidite = values[2]
36             airQuality = values[3]
37             uvValue = values[4]
38
39             now = datetime.now()
40             date_acquisition = now.strftime("%Y-%m-%d")
41             heure_acquisition = now.strftime("%H:%M:%S")
42
43             # Enregistrement des valeurs dans la base de données
44             cursor.execute(query, (date_acquisition, heure_acquisition, temperature,luminosite, humidite,airQuality,uvValue))
45
46             bd.commit()
47
48
49         time.sleep(200)

```

Pour ce qui est des changements, on peut voir sur la ligne 17, que je commence dans un premier temps par compléter ma fonction qui enverra une commande en mySQL dans ma base de données, cette fois-ci intégrant toutes mes données différentes.

Ensuite, la difficulté était de différencier chaque valeur arrivant dans mon moniteur de série afin de les intégrer correctement dans ma base de données. C'est pour pallier ce problème que dans mon code Arduino j'ai demandé que toutes valeurs soient séparées par une virgule. Grâce à cela, je peux demander de dire dans ma variable 'virgule', chaque valeur qui soit séparée par une virgule soit associée à une colonne différente dans ma liste.

De ce fait, je peux donc associer ma première valeur à la luminosité, la deuxième à la température etc.

Pour finir, je viens appeler ma fonction située ligne 20, avec toutes mes valeurs.

Ma base donnée doit alors pouvoir accueillir plus de variables. Pour ce faire, il me suffit juste d'ajouter à ma table mes listes manquantes :

```

ALTER TABLE myData
ADD COLUMN Luminosity VARCHAR(20),
ADD COLUMN Humidity VARCHAR(20),
ADD COLUMN AirQuality VARCHAR(20),
ADD COLUMN UVvalue VARCHAR(20);

```

Voici ma base données finales :

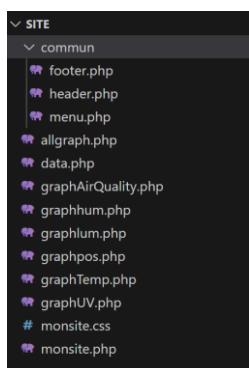
	<input type="button" value="← T →"/>			id	dateA	hoursA	temperature	Luminosity	Humidity	AirQuality	UVvalue		
<input type="checkbox"/>		Éditer		Copier		Supprimer	97	2023-11-11 07:13:55	24.20	770.00	70.00	101.62	10
<input type="checkbox"/>		Éditer		Copier		Supprimer	98	2023-11-11 07:13:57	24.20	773.00	70.00	101.62	10
<input type="checkbox"/>		Éditer		Copier		Supprimer	99	2023-11-11 07:13:59	24.10	771.00	70.00	104.45	10
<input type="checkbox"/>		Éditer		Copier		Supprimer	100	2023-11-11 07:14:01	24.10	773.00	70.00	104.45	10
<input type="checkbox"/>		Éditer		Copier		Supprimer	101	2023-11-11 07:14:03	24.10	770.00	70.00	104.45	10
<input type="checkbox"/>		Éditer		Copier		Supprimer	102	2023-11-11 07:14:05	24.10	773.00	70.00	104.45	10
<input type="checkbox"/>		Éditer		Copier		Supprimer	103	2023-11-11 07:14:07	24.20	767.00	70.00	98.85	10
<input type="checkbox"/>		Éditer		Copier		Supprimer	104	2023-11-11 07:14:09	24.20	773.00	70.00	101.62	10
<input type="checkbox"/>		Éditer		Copier		Supprimer	105	2023-11-11 07:14:11	24.10	771.00	70.00	104.45	10
<input type="checkbox"/>		Éditer		Copier		Supprimer	106	2023-11-11 07:14:13	24.10	769.00	70.00	101.62	10
<input type="checkbox"/>		Éditer		Copier		Supprimer	107	2023-11-11 07:14:15	24.20	765.00	70.00	101.62	10
<input type="checkbox"/>		Éditer		Copier		Supprimer	108	2023-11-11 07:14:17	24.20	769.00	70.00	104.45	10
<input type="checkbox"/>		Éditer		Copier		Supprimer	109	2023-11-11 07:14:19	24.10	766.00	69.00	101.62	10
<input type="checkbox"/>		Éditer		Copier		Supprimer	110	2023-11-11 07:14:21	24.10	769.00	69.00	104.45	10
<input type="checkbox"/>		Éditer		Copier		Supprimer	111	2023-11-11 07:14:23	24.20	761.00	70.00	101.62	10
<input checked="" type="checkbox"/>	Console de requêtes SQL		Supprimer	112	2023-11-11 07:14:25	24.20		765.00	70.00	101.62	10		

#### 4. Code web

Le code web n'affichait précédemment, seulement un graphique. Ainsi, je mettais fixer plusieurs conditions pour avoir un dashboard complet :

- Une page de présentation du projet avec un lien qui permettait directement une redirection vers le GitHub du projet
- Une page constituée de tous mes graphiques
- Et pour finir un onglet avec tous mes graphiques afin d'avoir la possibilité de les avoir en 'grand'

Dans un premier temps, j'ai effectué un tri dans mes fichiers en créant plusieurs fichiers ; chacun associé à un graphique, ainsi que plusieurs fichiers qui sont communs à chacune de mes pages afin de ne pas me répéter dans mes codes :



Je vais revoir étape par étape ce qui se trouve dans chacun de mes fichiers :

- Le footer

Dans le footer se situe seulement la fin de chacune de mes pages en fermant la commande html et le corps du code :

```
commun > footer.php
1   </body>
2   </html>
```

- Le header

Dans celui-ci se trouve donc la partie supérieure de chacune de mes pages, avec notamment le titre de ma page web, l'ouverture de la commande html et pour finir le header, avec l'intégration du fichier en CSS et la librairie chart.js .

```
commun > header.php > ...
1   <!DOCTYPE html>
2   <html>
3   <head>
4       <title>Data Graphics</title>
5       <script src="https://cdn.jsdelivr.net/npm/chart.js"></script>
6
7       <link rel="stylesheet" type="text/css" media="screen" href="monsite.css"/>
8
9   </head>
```

- Menu

Celui-ci met en place ma barre de navigation de ma page web, me permettant de passer d'un fichier à un autre en toute simplicité comme ceci :

```
commun > menu.php > nav
1   <nav>
2       <ul>
3           <li><a href="monsite.php">Home</a></li>
4           <li><a href="allgraph.php">All graphics</a></li>
5           <li><a href="graphTemp.php">Temperature</a></li>
6           <li><a href="graphHum.php">Humidity</a></li>
7           <li><a href="graphLum.php">Brightness</a></li>
8           <li><a href="graphUV.php">uvValue</a></li>
9           <li><a href="graphAirQuality.php">Air quality</a></li>
10
11      </ul>
12  </nav>
```

- Allgraph

Dans cette page en php, j'ai cherché à avoir tous mes graphiques réunis sur une seule et même page.

Pour ce faire, j'ai commencé dans un premier temps par initialiser tous mes communs comme ceci :

```
<body>
    <?php
        include("commun/menu.php");
        include("commun/header.php");
    ?>

    <?php
    include("commun/footer.php");
    ?>
```

Cette manipulation sera répétée pour chacune de mes pages.

Je suis tombé sur un premier problème qui était de pouvoir aligner tous mes graphiques les uns à côté des autres. La solution la plus simple que j'ai trouvé aura

été de créer deux colonnes en faisant trois graphiques d'un côté et deux de l'autre comme ceci :

```
<div class="container">
  <div class="column">
    <canvas id="temperatureChart"></canvas>
    <canvas id="humiditeChart"></canvas>
    <canvas id="luminositeChart"></canvas>
  </div>
  <div class="column">
    <canvas id="positionChart"></canvas>
    <canvas id="airQualityChart"></canvas>
    <canvas id="uvValueChart"></canvas>
  </div>
</div>
```

Je passe ensuite comme dans la première partie de ce rapport par la récupération de mes données situées dans le fichier data.php que l'on verra par la suite :

```
<script>
  fetch('data.php')
    .then(response => response.json())
    .then(data => {
      const dates = data.dates;
      const temperatures = data.temperatures;
      const humidites = data.humiditys;
      const luminosites = data.luminositys;
      const airQualitys = data.airQualitys;
      const uvValues = data.uvValue;
```

Une fois, toutes mes variables présentes dans mon fichier au lieu de créer 5 fois le même graphique, j'ai décidé de créer une fonction avec comme paramètre l'ID du graphique, le type de données associées et son unité, et pour finir quel est le tableau de data que je dois sélectionner parmi celles affichées ci-dessus :

```
function createChart(id, label, data) {
  var ctx = document.getElementById(id).getContext('2d');
  return new Chart(ctx, {
    type: 'line',
    data: {
      labels: dates,
      datasets: [
        {
          label: label,
          data: data,
          borderColor: getRandomColor(),
          fill: false
        }
      ]
    }
  });
}
```

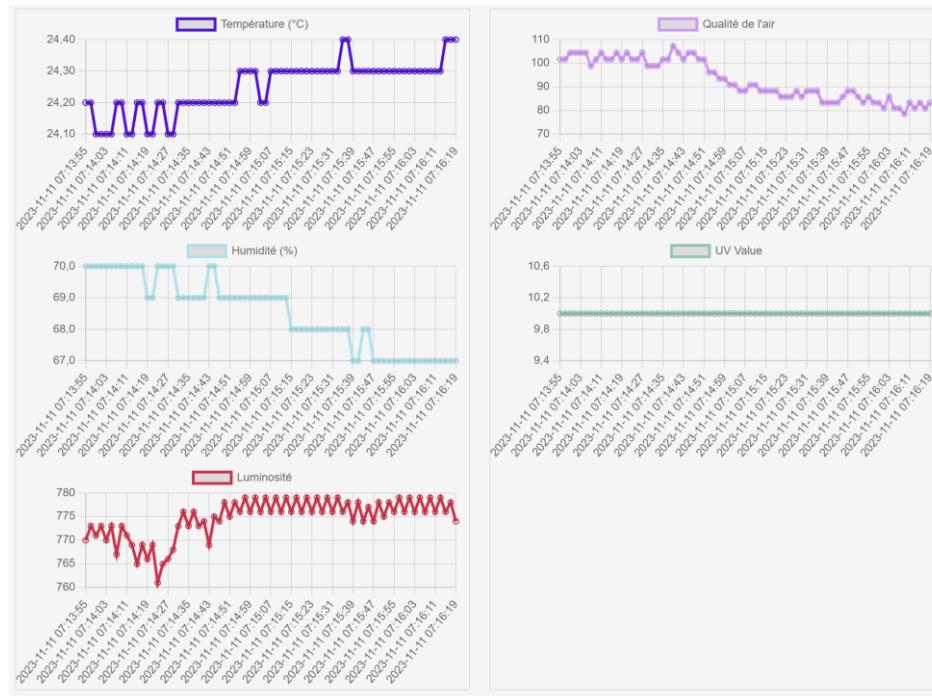
Maintenant que j'ai ma fonction, j'ai tout simplement besoin de l'appeler avec les bons paramètres associés à celles-ci pour chacun de mes graphiques. Pour les différencier, j'ai décidé d'y ajouter une fonction qui choisit une couleur aléatoire à chaque fois que l'un d'entre eux est créé afin de ne pas avoir tous mes graphiques de la même couleur à chaque fois.

```

var temperatureChart = createChart('temperatureChart', 'Température (°C)', temperatures);
var humiditeChart = createChart('humiditeChart', 'Humidité (%)', humidites);
var luminositeChart = createChart('luminositeChart', 'Luminosité', luminosites);
var airQualityChart = createChart('airQualityChart', 'Qualité de l\'air', airQualitys);
var uvValueChart = createChart('uvValueChart', 'UV Value', uvValues);

function getRandomColor() {
    var letters = '0123456789ABCDEF';
    var color = '#';
    for (var i = 0; i < 6; i++) {
        color += letters[Math.floor(Math.random() * 16)];
    }
    return color;
}
});
```

Voici le résultat de cette page affichant tous mes graphiques :



Ici nous avons un résultat satisfaisant avec tous mes graphiques présents sur une seule et même page avec une différentiation visible grâce aux couleurs de ceux-ci.

- Data

```
<?php
$host = 'localhost';
$user = 'root';
$password = '';
$database = 'mydatabase';

$conn = new mysqli($host, $user, $password, $database);

if ($conn->connect_error) {
    die('Connection failed: ' . $conn->connect_error);
}

$sql = 'SELECT dateA, hoursA, temperature, Luminosity, Humidity, AirQuality, UVvalue FROM mydata';
$result = $conn->query($sql);

$data = array(
    'dates' => array(),
    'temperatures' => array(),
    'luminosities' => array(),
    'humiditys' => array(),
    'airQualities' => array(),
    'uvValue' => array()
);

if ($result->num_rows > 0) {
    while ($row = $result->fetch_assoc()) {
        $datetime = $row['dateA'] . ' ' . $row['hoursA'];
        array_push($data['dates'], $datetime);
        array_push($data['temperatures'], $row['temperature']);
        array_push($data['humiditys'], $row['Humidity']);
        array_push($data['luminosities'], $row['Luminosity']);
        array_push($data['airQualities'], $row['AirQuality']);
        array_push($data['uvValue'], $row[('UVvalue')]);
    }
}

$conn->close();
```

Ici, on reprend le même concept que vu précédemment, mais au lieu de récupérer que la température, ici on va récupérer toutes les données souhaitées dans ma base de données. Sur le même principe, on va associer toutes ces données récupérées à des variables dans mon code web, afin de pouvoir les réutiliser simplement.

- Les graphiques individuels

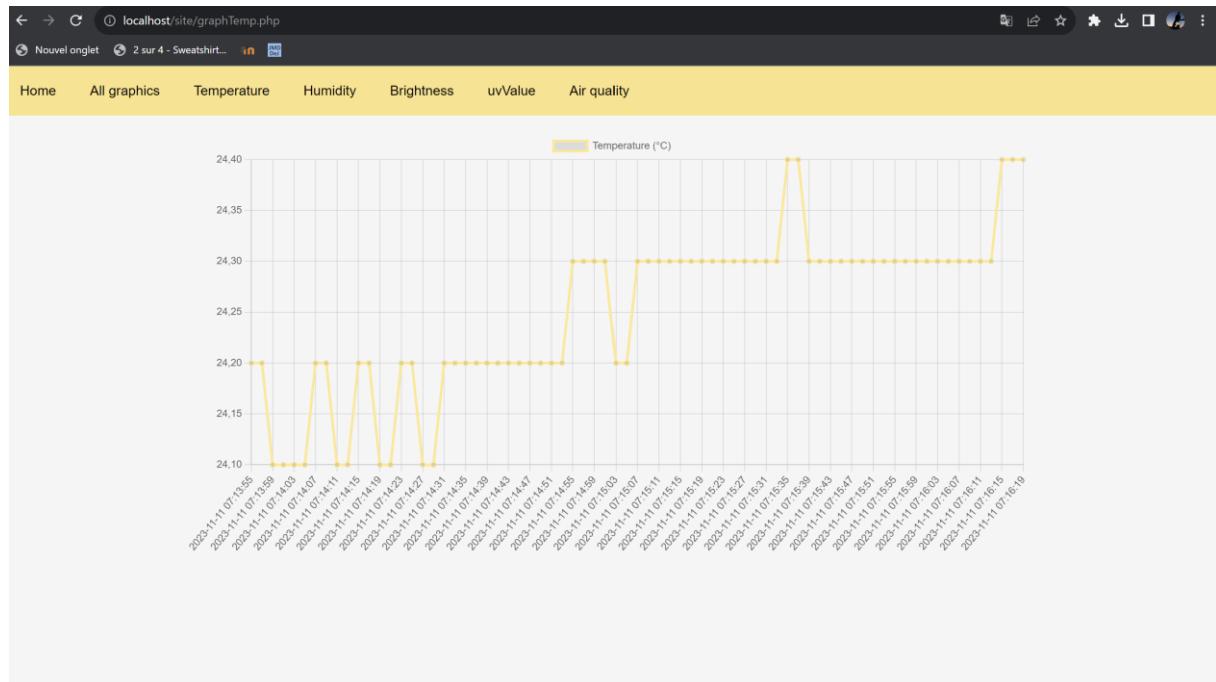
Ici vu que l'on cherche seulement à avoir un graphique par page, je vais reprendre exactement le même code que vu précédemment en y ajoutant mes fichiers communs et pour chaque page en y intégrer les données correspondantes au graphique.

Voici un exemple du code pour le graphique de la Qualité de l'Air :

```
<body>
    <div class="container">
        <canvas id="airChart"></canvas>
    </div>
    <script>
        fetch('data.php')
            .then(response => response.json())
            .then(data => {
                const dates = data.dates;
                const airQuality = data.airQuality;

                var airChart = new Chart(document.getElementById('airChart').getContext('2d'), {
                    type: 'line',
                    data: {
                        labels: dates,
                        datasets: [{
                            label: 'AirQuality',
                            data: airQuality,
                            borderColor: '#FFE798',
                            fill: false
                        }]
                    }
                });
            });
    </script>
```

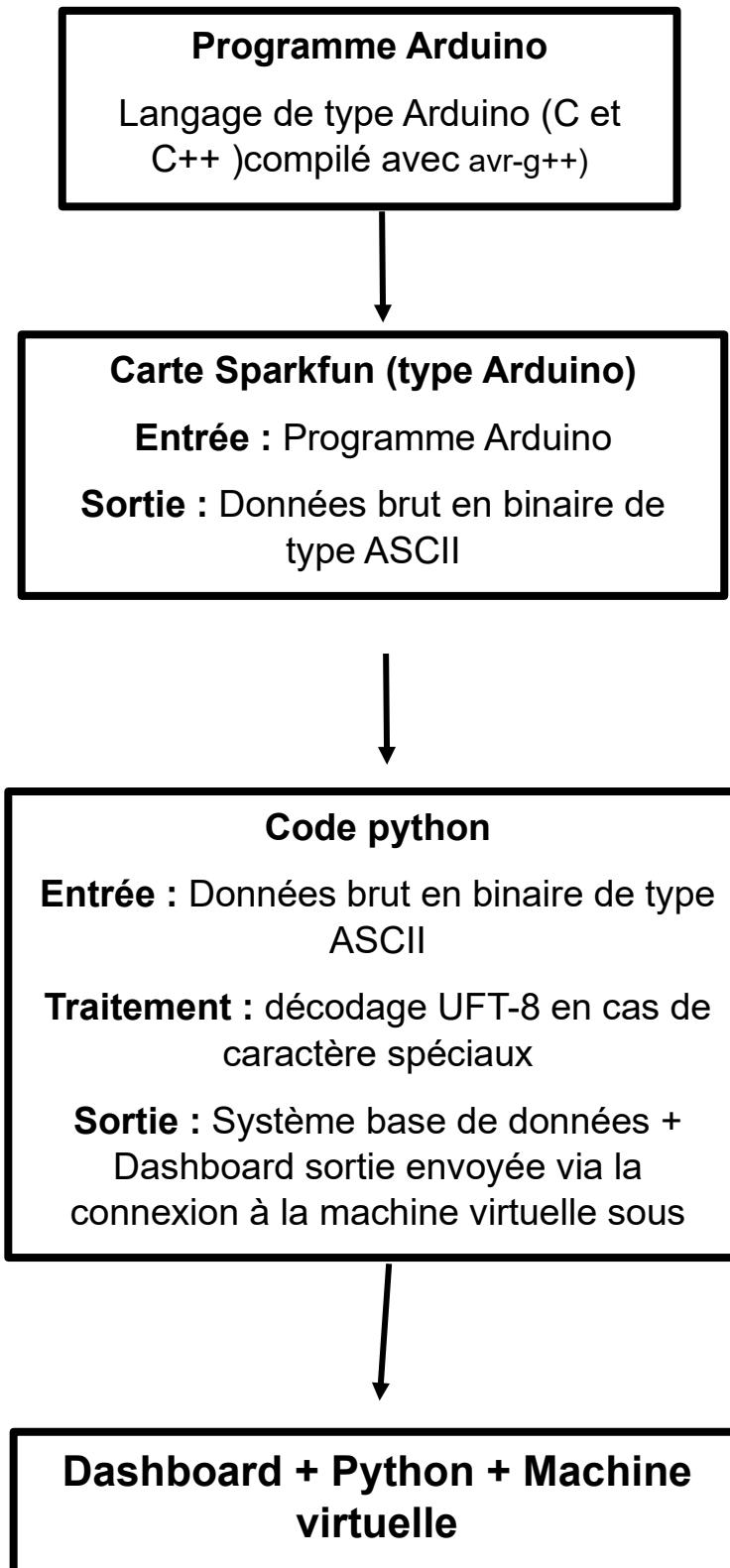
Voici le résultat de ce Dashboard :



Nous avons alors un Dashboard fonctionnel, avec la page constituant tous les graphiques et des pages séparées constituant chaque graphique individuel. Ainsi qu'une page d'accueil pouvant servir à fournir toute informations complémentaire concernant le projet.

## C. Complément au projet

### 1. Fonctionnement interne



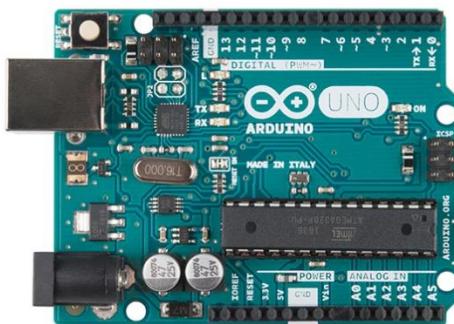
## 2. Notice

Voici la notice d'utilisation de mon projet, afin de faciliter l'installation à tout utilisateur de celui-ci :

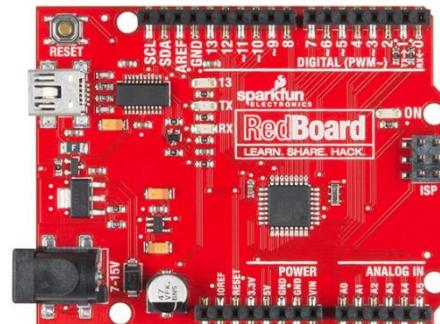
## Arduino

Arduino est un langage de programmation se rapprochant du langage C, il permet de configurer facilement des cartes électroniques Arduino ou du même type. Pour ce faire, nous allons utiliser le logiciel constructeur ici Arduino.

La carte que nous avons est une carte qui se rapproche de l'arduino uno.



Arduino UNO



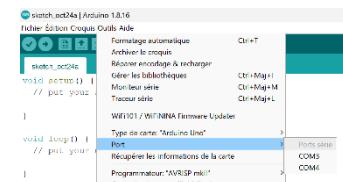
Sparkfun RedBoard

Leurs fonctionnements y sont totalement similaires.

Pour l'utilisation de la carte, vous devez dans un premier temps télécharger le logiciel Arduino disponible gratuitement sur ce lien.

<https://www.arduino.cc/en/software>

Une fois cela fait, il suffit pour le projet d'ouvrir le code présent dans le dossier « project\_sensor\avec\_expl\arduino\_finale\S17\_ecranlcd » le code y est expliqué directement dessus. Pour implanter le code à la carte, il vous faut dans un premier temps brancher la carte Arduino à votre ordinateur et vérifier que le bon port y est associé. Pour ce faire, il suffit d'aller dans l'onglet Outils → Port → Sélectionner le port associé :



L'étape d'après est d'implanter le code dans la carte, il suffit d'appuyer sur ce bouton le code va ainsi s'auto-compiler et s'implanter dans la carte :



```
#include <LiquidCrystal.h>
#include "DHT.h"
#include <MQ135.h>
```

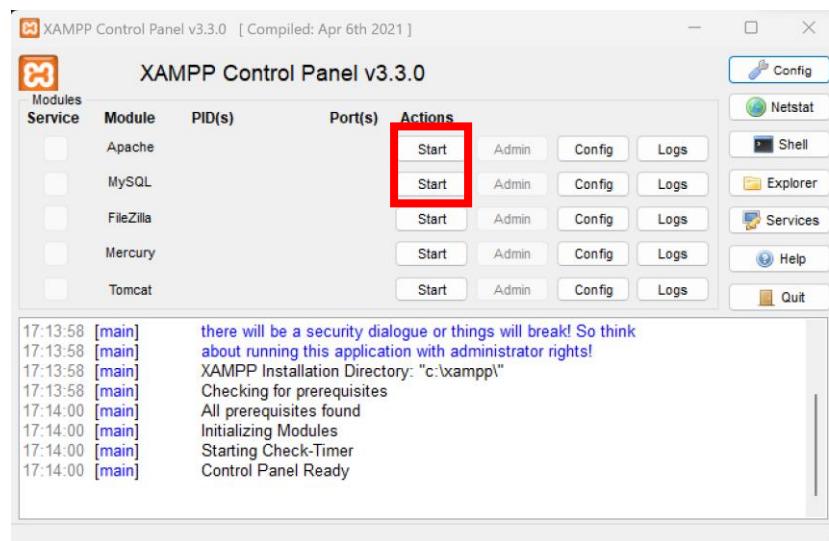
## XAMPP

Xampp est un logiciel, regroupement à la fois une machine virtuelle sur Apache2 et une base de données en MySql, il permet donc de lancer notre base de données facilement et sans besoin de configuration complexe.

Pour ce faire, la première étape à suivre est l'installation du logiciel XAMPP.

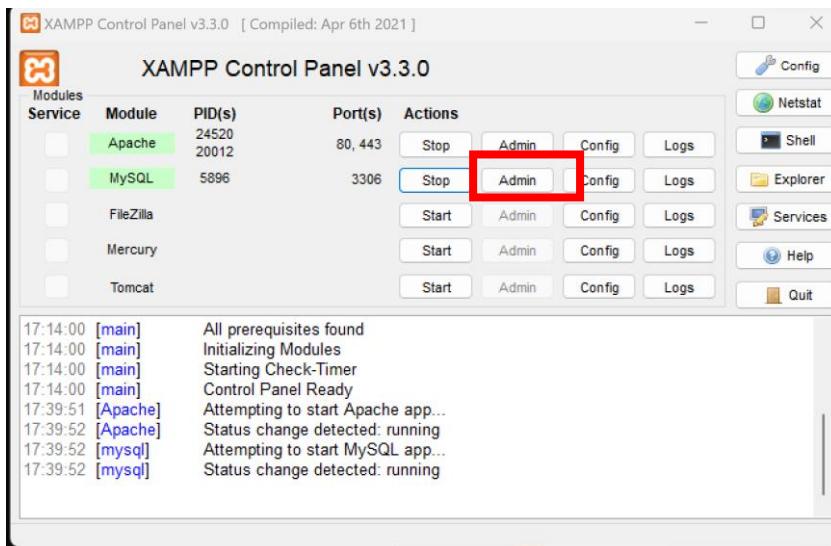
<https://www.apachefriends.org/fr/download.html>

Une fois l'installation terminée, il suffit d'ouvrir le logiciel qui se présente comme ceci :



Dans un premier temps il faut démarrer la machine virtuelle sous Apache ainsi que la base de données sous Mysql. Puis, il suffit de cliquer sur les deux boutons 'start' entourés ci-dessus.

Une fois cela-fait pour accéder à la base de données, il suffit de cliquer le bouton 'Admin' comme ci-dessous.



Si rien ne se produit, ouvrez un navigateur web quelconque et ouvrez <http://localhost/phpmyadmin/>

Une fois dans le myAdmin, il vous faudra créer la base de données qui va accueillir toutes nos données. La première étape est d'aller sur le lien ci-dessus, ensuite d'aller dans mySql et d'entrer cette commande qui va créer notre base de données :

`CREATE DATABASE mydatabase;`

Ensuite, il faut passer dans la DataBase en cliquant dessus, retourner sur bouton mySql et entrer cette commande :

```

CREATE TABLE mydata (
    ID INT AUTO_INCREMENT PRIMARY KEY,
    Date DATE,
    Time TIME,
    Temperature VARCHAR(20),
    Luminosity VARCHAR(20),
    Humidity VARCHAR(20),
    AirQuality VARCHAR(20),
    UVvalue VARCHAR(20)
);

```

## PYTHON

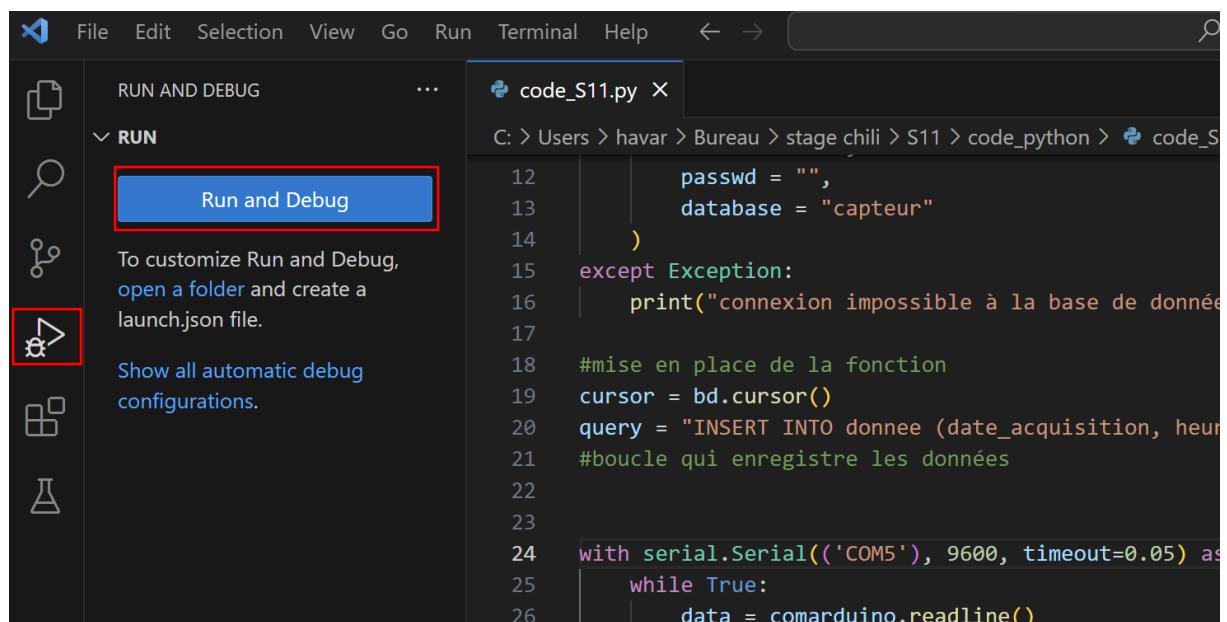
Pour le transfert de données entre nos données qui arrivent grâce au code Arduino sur notre moniteur de série et notre base de données, nous avons besoin d'un code en Python.

Un code en Python peut s'exécuter avec de multiples logiciels. Ici nous avons utilisé Visual Studio code. Pour l'installer, il suffit de suivre ce lien :

<https://code.visualstudio.com/Download>

Une fois cela fait, il vous suffit d'ouvrir le code présent dans le dossier : project\_sensor\avec\_expl\code python

Le code y est expliqué directement dessus. Pour le lancer il suffit d'appuyer sur ce bouton et de sélectionner Python dans le type de code à exécuter.



The screenshot shows the Visual Studio Code interface. On the left, there's a sidebar with icons for file operations, a search bar, and a 'RUN' section. The 'Run and Debug' button is highlighted with a red box. Below it, there's a note about customizing run and debug settings and a link to a launch.json file. The main area shows a Python script named 'code\_S11.py' with some code. The code includes imports for serial and MySQLdb, defines variables for password and database name, handles exceptions, and performs an INSERT INTO operation. It also includes a loop that reads from a serial port (COM5) at 9600 baud rate with a timeout of 0.05 seconds.

```

12     passwd = "",
13     database = "capteur"
14 )
15 except Exception:
16     print("connexion impossible à la base de données")
17
18 #mise en place de la fonction
19 cursor = bd.cursor()
20 query = "INSERT INTO donnee (date_acquisition, heure, valeur) VALUES (%s, %s, %s)"
21 #boucle qui enregistre les données
22
23
24 with serial.Serial('COM5', 9600, timeout=0.05) as comarduino:
25     while True:
26         data = comarduino.readline()

```

Attention à ne pas lancer le code Arduino en même temps, les deux ne pouvant pas fonctionner en simultané car ceux-ci cherchent à lire sur le même moniteur de série. Le but ici est d'envoyer dans un premier temps le code arduino comme expliqué ci-dessus. Une fois sur la carte, le logiciel doit être fermé (il n'a plus d'utilité). Seul le code Python doit être en cours d'exécution.

## Le DashBoard

Vous devrez ouvrir l'explorateur de fichier et aller dans C:\xampp\htdocs. Une fois dans le dossier htdocs , vous devez déplacer le dossier site qui se situe dans project\_sensor

\avec\\_expl\site dans le dossier htdocs. Ensuite, il suffit d'ouvrir un navigateur web quelconque et d'aller sur ce lien. <http://localhost/site/monsite.php>.

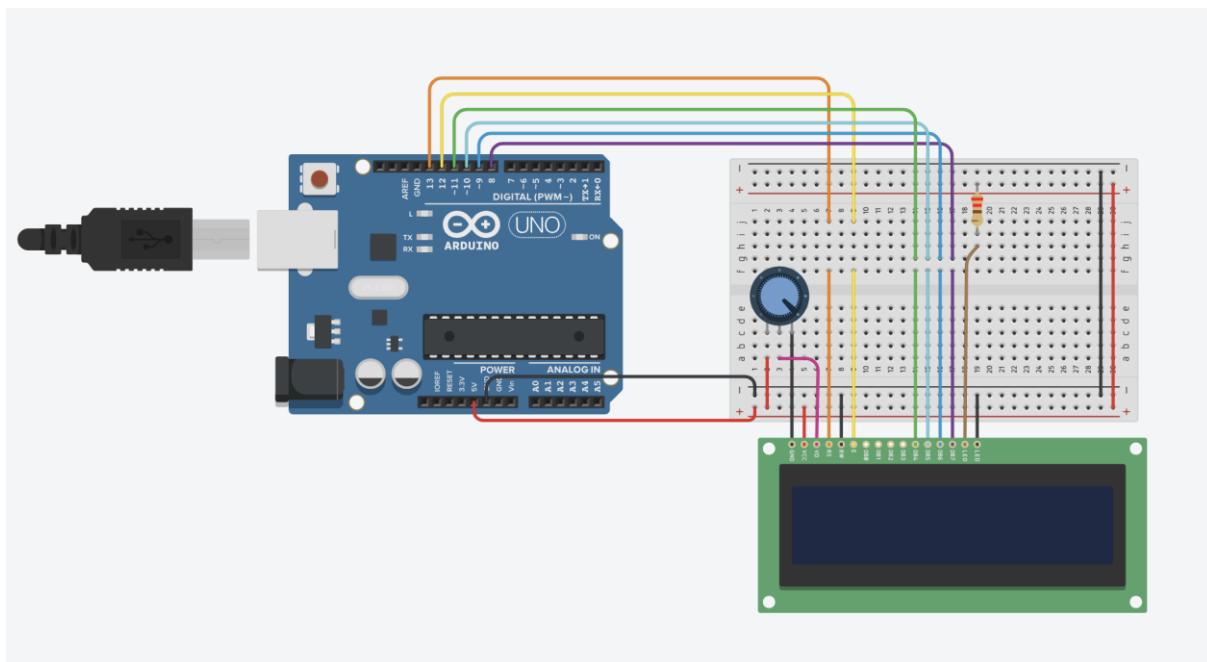
# Information

Pour toutes modifications du code ou du projet, veuillez-vous référer au rapport pour les explications du projet, et au code disposant pour chaque ligne une explication correspondante. Pensez au préalable à sauvegarder la première version du projet en cas d'erreur.

### 3. Intégration d'un écran lcd

Afin de compléter le système, j'ai décidé d'y ajouter un écran LCD, afin de pouvoir lire toutes les valeurs de mes capteurs sans avoir besoin d'allumer le dashboard.

Voici les branchements de cette écran lcd :



Pour afficher toutes les données, il a fallu modifier le code Arduino en y ajoutant la librairie liée à l'écran Lcd ainsi que l'envoi des données dans celui-ci en format intégral pour que celles-ci tiennent sur l'écran.

Voici le code correspondant :

```
#include "DHT.h"
#include <MQ135.h>
#include <LiquidCrystal.h>

#define DHTPIN 2
#define DHTTYPE DHT11

const int luminositePin = A0;
const int airQualityPin = A1;
const int uvSensorPin = A2;

MQ135 gassSensor = MQ135(airQualityPin);
DHT dht(DHTPIN, DHTTYPE);

float voltage;
float degreesC;
float lux;
LiquidCrystal lcd(13,12,11,10,9,8);
void setup() {

pinMode(uvSensorPin, INPUT);
pinMode(luminositePin, INPUT);
pinMode(airQualityPin, INPUT);
dht.begin();
lcd.begin(16,2);
Serial.begin(9600);
}

int luminositeInt = int(luminosite);
int temperatureInt = int(temperature);
int humidityInt = int(humidity);
int airQualityValueInt = int(airQualityValue);
int uvIntensityInt = int(uvIntensity);

lcd.setCursor(0, 0);
lcd.print("L:");
lcd.print(luminositeInt);
lcd.print(" T:");
lcd.print(temperatureInt);
lcd.print(" H:");
lcd.print(humidityInt);

lcd.setCursor(0, 1);
lcd.print(" AQ:");
lcd.print(airQualityValueInt);
lcd.print(" UV:");
lcd.print(uvIntensityInt);
delay(1000);
}
```

#### 4. Piste de recherche

Pour faire suite au projet, plusieurs améliorations pourraient être effectuées :

- Une amélioration nette du Dashboard, avec notamment la possibilité d'y intégrer des curseurs, de zoomer sur ceux-ci ou encore de marquer différents points.
- Création d'un système domotique autour du projet. Ce projet étant composé de plusieurs capteurs, on pourrait utiliser ces variables pour y intégrer des systèmes automatisés, comme un chauffage automatisé relié aux données de mon capteur de température, ou des volets automatisés, eux reliés aux valeurs UV et aux valeurs de la photorésistance etc
- Effectuer un étalonnage précis des capteurs présents dans mon projet, afin de recueillir des données précises.
- Etablir un serveur web, afin que ces données soient accessibles depuis n'importe quel ordinateur
- Création d'un module permettant une connexion sans fil, afin de pouvoir recevoir les données provenant de ma carte Arduino sans avoir à se brancher à un ordinateur.
- La création d'un support sur un logiciel de modélisation tel que Solidworks, et la mise en place d'un « cable management », afin de faciliter l'accessibilité aux capteurs

## V. Impact socio-écologique

### Impact social

Si ce projet est dans un futur fonctionnel au sein d'un environnement comme une salle de classe. Celui-ci pourra apporter de l'aide au quotidien de ces utilisateurs. Ainsi que leur assurer un environnement plus saint grâce au données précise acquise par mes capteurs

D'autre part ce projet peut-être amener à être continuer dans un futur proche, c'est alors qu'intervient le partage de connaissance et l'amélioration de celui-ci. L'utilisation des multiples sources : WEB / mySQL / PYTHON / ARDUINO et la compréhension de celle-ci grâce au projet peut avoir un impact social positif.

Si ce projet ce valide par une entreprise alors intervient la formation à ce système et à la création d'emplois liée à celui-ci.

### Impact Ecologique

Mon projet est constitué d'un système hardware conséquent, celui-ci demandant une certaine ressource en électricité. L'arduino est alimenté au maximum à 1A, 12V, donc consomme 5W, ce qui reste moindre par rapport à d'autre système numérique. De plus si celui-ci est optimisé, il consommera encore moins d'énergie.

Ce système constitue des éléments ayant un cycle de vie défini, comme l'ordinateur que j'utilise actuellement pour la programmation de mon projet, ou encore le système Arduino. Si ceux-ci ne sont pas recyclable ils auront un grand impact sur notre environnement.

## VI. Analyse budgétaire du projet

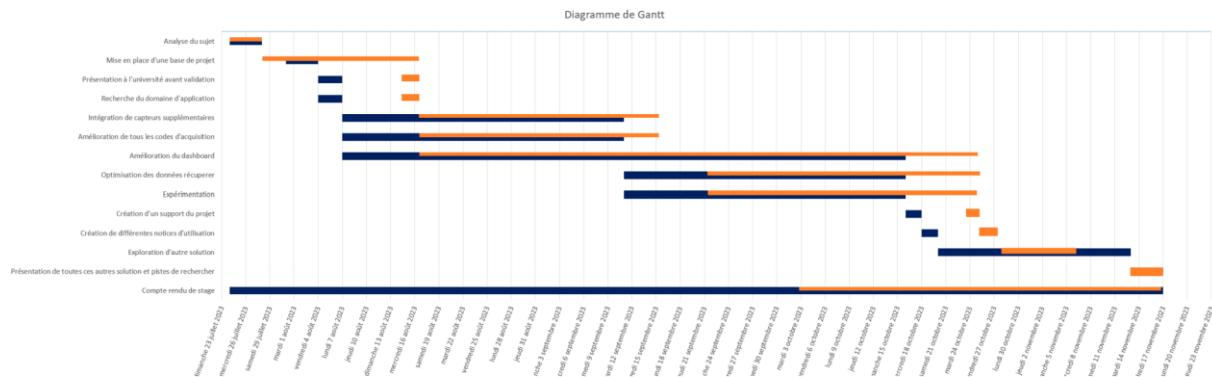
A	B	C
1 Composant	Cout	Liens pour achat
2 Arduino Uno		17,99 <a href="https://articulo.mercadolibre.cl/MLC-435072401-arduino-uno-r3-cable-usb-10-cable-conexion-5mm-y-5hh-max--JM#position=4&amp;search">https://articulo.mercadolibre.cl/MLC-435072401-arduino-uno-r3-cable-usb-10-cable-conexion-5mm-y-5hh-max--JM#position=4&amp;search</a>
3 Capteur DHT11		3 <a href="https://articulo.mercadolibre.cl/MLC-434124622-sensor-de-temperatura-y-humedad-dht11-arduino-pic-max--JM#position=1&amp;search">https://articulo.mercadolibre.cl/MLC-434124622-sensor-de-temperatura-y-humedad-dht11-arduino-pic-max--JM#position=1&amp;search</a>
4 Photorésistance		2 <a href="https://articulo.mercadolibre.cl/MLC-514031032-sensor-ldr-foto-resistencia-digital-salida-ajustable-max--JM">https://articulo.mercadolibre.cl/MLC-514031032-sensor-ldr-foto-resistencia-digital-salida-ajustable-max--JM</a>
5 Resistance		5,99 <a href="https://articulo.mercadolibre.cl/MLC-452121840-pack-600-resistencia-ohm-arduino--JM#position=12&amp;search_layout=stack&amp;type=item&amp;">https://articulo.mercadolibre.cl/MLC-452121840-pack-600-resistencia-ohm-arduino--JM#position=12&amp;search_layout=stack&amp;type=item&amp;</a>
6 Capteur UV ML8511		8,39 <a href="https://articulo.mercadolibre.cl/MLC-446708597-modulo-sensor-ultravioleta-breakout-ml8511--JM#position=15&amp;search_layout=stack&amp;">https://articulo.mercadolibre.cl/MLC-446708597-modulo-sensor-ultravioleta-breakout-ml8511--JM#position=15&amp;search_layout=stack&amp;</a>
7 Ecran lcd		3,99 <a href="https://articulo.mercadolibre.cl/MLC-441134332-pantalla-display-lcd-16x2-con-modulo-interfaz-i2c-arduino--JM#position=3&amp;search_layout=stack&amp;">https://articulo.mercadolibre.cl/MLC-441134332-pantalla-display-lcd-16x2-con-modulo-interfaz-i2c-arduino--JM#position=3&amp;search_layout=stack&amp;</a>
8 Cable		2,6 <a href="https://articulo.mercadolibre.cl/MLC-542729977-cable-jumper-dupont-macho-hembra-40-unid--JM#is_advertising=true&amp;position=2&amp;search_layout=stack&amp;">https://articulo.mercadolibre.cl/MLC-542729977-cable-jumper-dupont-macho-hembra-40-unid--JM#is_advertising=true&amp;position=2&amp;search_layout=stack&amp;</a>
9 Capteur Gaz		3,19 <a href="https://articulo.mercadolibre.cl/MLC-472793881-sensor-de-calidad-de-aire-para-arduino-mq-135--JM#position=1&amp;search_layout=stack&amp;">https://articulo.mercadolibre.cl/MLC-472793881-sensor-de-calidad-de-aire-para-arduino-mq-135--JM#position=1&amp;search_layout=stack&amp;</a>
10		
11 Totale	47,15	
12		
13		

En reprenant chaque composant, je suis aller chercher leurs couts sur le site Mercado Libre, qui est un revendeur de composant informatique au Chili, auquel j'ai eu recours pour ce projet.

Le cout de totale de celui-ci est de 47 150 pesos Chiliens soit environs 48,2 euros.

Ce cout reste très correct pour un projet comme celui-ci en notant que tous les logiciels tiers. Que ce soit pour la programmation, ou pour le stockage de données sont totalement gratuit d'installation et d'utilisation.

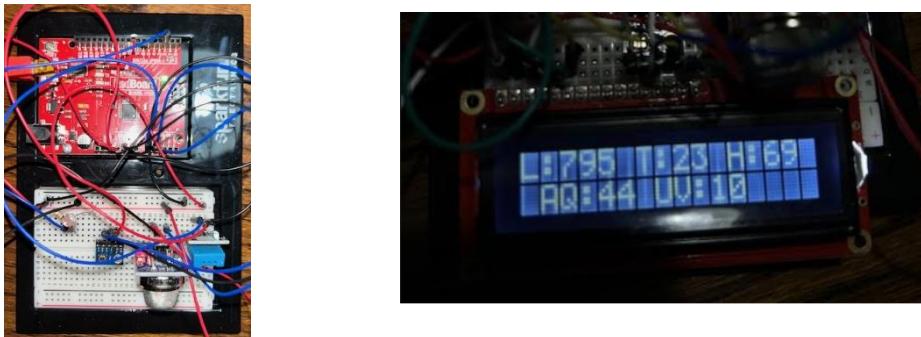
## VII. Comparaison planning prévisionnel et planning effectif :



On peut voir sur ce deuxième diagramme de gantt, que le planning prévisionnel en bleu correspond à peu près au planning effectif en orange. J'ai pris un peu de retard sur la base du projet, mais j'ai toute de même réussi à me rattraper sur la fin du stage.

## VIII. Rendu finale du projet

Pour ce qui est du rendu final du projet, nous avons d'un aspect extérieur tous notre système Arduino comme ceci :



Et d'autre part notre dashboard avec toutes les données de nos capteurs :



En conclusion, nous avons alors un système hardware fonctionnel. Dont, les valeurs de mes capteurs varient en fonction de leurs environnement externe.

Pour ce qui est du dashboard, celui-ci répond à ma demande en affichant graphiquement toutes valeurs de mes capteurs en fonction du temps.

Entre les deux est présent diverse programme, se référer à l'annexe 5 – Programme, pour le détail de ce ceux-ci.

Ce projet s'intègre donc parfaitement dans la problématique '*Framework para la construcción de soluciones inteligentes en ambientes de Computación Ubicua aplicando modelos de Machine Learning*'.

De plus, sera créer un github, permettant de poursuivre toutes recherches concernant ce projet. Tous les différents codes principaux sont visibles dans l'annexe D : Programme.

## CONCLUSION

### A. Conclusion du projet

Le projet que j'ai mené est donc le résultat d'un système d'acquisition, de stockage et d'affichage de données, en partant du système Arduino, avec tout le système hardware et software associé, de la base de données en MySQL sous Apache, le code en Python permettant d'associer mes données sortant de ma carte à la base de données; et pour finir le dashboard codé en CSS,HTML, JAVA.

### B. Conclusion personnelle

D'un point de vue personnel, je reste satisfait de ce projet, celui-ci étant fonctionnel et simple d'utilisation. J'aurais tout de même aimé pouvoir continuer celui-ci et ajouté tous les éléments vus dans les perspectives d'avenir au projet.

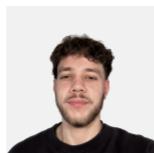
Cette expérience à l'étranger m'a permis de comprendre et connaître le monde du travail à l'étranger. De plus, j'ai pu grâce à ce stage acquérir de nombreuses compétences supplémentaires que ce soit sur la partie hardware avec le système Arduino ou la partie software avec les différents langages de programmation que constitue mon projet. Ce stage confirme mon souhait de poursuivre mes études dans l'ingénierie et m'incite à continuer dans un parcours au sein de l'ESEO.

## ANNEXES

### Annexe A : CV

#### HAVARD-COLSON Alexis

STUDENT AT ESEO ENGINEERING SCHOOL



31 allée des champs fleuris  
45560,Saint Denis en Val,  
France

[alexis.havardcolson@reseau.eseo.fr](mailto:alexis.havardcolson@reseau.eseo.fr)

+33 6 13 67 22 35

#### ABOUT ME

I am currently in the first year of the engineering cycle at ESEO engineering school (3 years studies). I am a curious person who likes to discover new things, in order to acquire new skills. I like creating, optimizing electronic systems. During my studies, I did several projects in informatics sciences and electronics which allow me to have good skills in these both areas.

#### SOFT SKILLS

- Serious
- Resilient
- Punctual
- Curious
- Patient

#### INTERESTS

- Surf/ Ski
- Travel
- Workout

#### LANGUAGES



#### FURTHER INFORMATION

- Sports shooting with qualification five times in a row for the French championship from 2010 to 2015 (Regional championship : first place in 2014)

#### EDUCATION

2022-2023

- ESEO engineering school in Angers (France) : First year of the engineering cycle

2020-2022

- BTS (equivalent to two-year university degree in US) Physical technics for industry and laboratory (engineering option), at Ste Croix St Euverte school (Orléans, France)

- Core subjects :
  - Electronics
  - Mécanics
  - Optics

- 3 months of internship in the R&D department at Maquet SAS (manufacturer of surgical lights at Orleans, France)

- Project study : Development of a CO2 sensor

2020

- Scientific bachelor degree at Ste Croix St Euverte high school (Orléans, France)

- Core subjects :
  - engineering
  - Physics
  - Mathématiques

- Project study : Development of a mock-up of an automatic garage.

#### SOFTWARE AND ELECTRONICS SKILLS

- Pack office: Excel, Word, Powerpoint
- Basic level in Solidworks and Altium
- Basic level in Python and HTML
- Intermediate level in Java, Arduino, VHDL and C
- Expert in electronic tools : Oscilloscope, Multimeter, GBF, etc
- Expert of software related to electronics : Arduino, Pspice, LabView, LTSpice
- Good knowledge of many electronic components

## Annexe B : Planning détaillé de la mission de stage

Tâche	Etat	Date de commencement	Date de fin	Semaine
Présentation du campus, de l'équipe administratif et enseignante.	Fini	24/07/2023	24/07/2023 S1	
Reunion avec les associés aux projets pour redéfinir les objectifs	Fini	24/07/2023	24/07/2023 S1	
Première recherche sur les documents déjà associés au projet	Fini	24/07/2023	28/07/2023 S1	
Mise en place de capteur Arduino, acquisition des données sur un moniteur de série.	Fini	25/07/2023	28/07/2023 S1	
Mise en place de la base de données	Fini	25/07/2023	28/07/2023 S1	
Mise en place du code en python pour la rellia à la base de données	Fini	25/07/2023	28/07/2023 S1	
Amélioration du code arduino pour y intégrer deux capteur	Fini	31/07/2023	05/08/2023 S2	
Amélioration du code en python afin d'y intégrer la connection à deux tableau différent dans la BDD	Fini	31/07/2023	05/08/2023 S2	
Amélioration de la BDD afin d'intégrer la date/la température et l'humidité	Fini	31/07/2023	05/08/2023 S2	
Amélioration du code en python afin d'y ajouter la date et l'heure d'acquisition de la donnée	Fini	31/07/2023	05/08/2023 S2	
Analyse des documents non-vu	Fini	31/07/2023	05/08/2023 S2	
Imaginer un domaine d'application du système	Fini	31/07/2023	05/08/2023 S2	
Recherche des nouveaux capteurs à utiliser pour le système	Fini	07/08/2023	08/08/2023 S3	
Intégration de ceux-ci au système actuel	Fini	08/08/2023	10/08/2023 S3	
Amélioration code arduino pour l'humidité	Fini	08/08/2023	10/08/2023 S3	
Amélioration code python pour l'humidité	Fini	08/08/2023	10/08/2023 S3	
Amélioration la BDD pour l'humidité	Fini	08/08/2023	10/08/2023 S3	
Recherche de plusieurs domaine d'application du système et pourquoi ?	En cours	11/08/2023	semaine pro	S3
Recherche de plusieurs domaine d'application du système et pourquoi ?	Fini	16/08/2023	18/08/2023 S4	
code html pour le site web	Fini	16/08/2023	18/08/2023 S4	
code php	Fini	17/08/2023	18/08/2023 S4	
recherche bitquery	En cours	18/08/2023		S4
lecture document réunion	Fini	21/08/2023	21/08/2023 S5	
analyse capteur différent possible	Fini	21/08/2023	22/08/2023 S5	
communication sans câble	Fini	22/08/2023	22/08/2023 S5	
livraison des composants analyse, modification du code arduino	Fini	23/08/23	25/08/2023 S5	
amélioration de la page web et rafraîchissement automatique de la page	Fini	28/08/2023	31/08/2023 S6	
amélioration du code en python	Fini	28/08/2023	31/08/2023 S6	
changeement de la base de données	Fini	28/08/2023	31/08/2023 S6	
mise en place du module bluetooth	En cours	31/08/2023		S6
mise en place du module bluetooth	En cours	04/09/2023		
création de l'interface de la page web	Fini	04/09/2023	05/09/2023 S7	
étalementage des capteurs	Fini	05/09/2023	07/09/2023 S7	
Ajout des tous les autres capteurs	Fini	11/09/2023	15/09/2023 S8	
Etalonnage de ceux-ci (arduino)	Fini	11/09/2023	15/09/2023 S8	
Modification BDD	Fini	11/09/2023	15/09/2023 S8	
Modification Python	Fini	11/09/2023	15/09/2023 S8	
Modification Page web	En cours	11/09/2023	15/09/2023 S8	
Modification page web	Fini	18/09/2023	20/09/2023 S9	
test de l'application du système actuelle et cohésion des valeurs acquises	Fini	20/09/2023	22/09/2023 S9	
Amélioration du dashboard (zoom sur les graphes)	En cours	25/09/2023		S10
Début de la rédaction du rapport de stage	Fini	25/09/2023	28/09/2023 S10	
Etudier les moments propice à l'acquisition de chaque capteur, et trouver une solution	Fini	25/09/2023	28/09/2023 S10	
Amélioration du dashboard	En cours	2/10/2023	06/10/2023 S11	
Mise en place du bluetooth	Fini	02/10/2023	06/10/2023 S11	
Création de plusieurs mode d'acquisition	A faire	2/10/2023	06/10/2023 S11	
Mise en place de l'écran LCD au système	Fini	02/10/2023	06/10/2023 S11	
Expérience dans la salle de l'université	A faire	2/10/2023	06/10/2023 S11	
Acquisition des données lorsque cela y est utile	En cours	09/10/2023	14/10/2023 S12	
Mise en place de quelques modules en plus led, buzzer etc	Fini	09/10/2023	14/10/2023 S12	
Amélioration du dashboard	Fini	09/10/2023	14/10/2023 S12	
Amélioration du dashboard zoom	Fini	09/10/2023	14/10/2023 S12	
Selection de la période d'acquisition sur les graphiques	En cours	09/10/2023	14/10/2023 S12	
Création d'une notice d'utilisation	Fini	16/10/2023	20/10/2023 S13	
Rédaction du rapport	En cours	16/10/2023	20/10/2023 S13	
Rédaction du rapport	En cours	23/10/2023	26/10/2023 S14	
Zoom sur les graphiques	En cours	23/10/2023	26/10/2023 S14	
Création d'un support potentiel sur solidworks	Fini	23/10/2023	26/10/2023 S14	
Rédaction du rapport	Fini	30/10/2023	03/11/2023 S15	
Optimisation des différents code et explication, (notice sur le code)	Fini	30/10/2023	03/11/2023 S15	
Relecture rapport, compilation de tous les dossiers envoi à l'université	Fini	06/11/2023	13/11/2023 S16	
Rendu du projet et du github	A faire	13/11/2023	17/11/2023 S17	

## Annexe C : Références bibliographiques utilisées, sources internes et externes

Hardware	Forum Arduino DataSheet composants Utilisation du capteur UV Autre	<a href="https://forum.arduino.cc/">https://forum.arduino.cc/</a> <a href="https://www.alldatasheet.com/">https://www.alldatasheet.com/</a> <a href="https://www.youtube.com/watch?v=zbHh3onXqBc">https://www.youtube.com/watch?v=zbHh3onXqBc</a> <a href="https://github.com/">https://github.com/</a>
SoftWare	Python  Code Web  Machine virtuelle  Aide à la correction de code	<a href="https://www.youtube.com/watch?v=vPLGALaVWMY">https://www.youtube.com/watch?v=vPLGALaVWMY</a> <a href="https://www.youtube.com/watch?v=RvAiaLKi43Y">https://www.youtube.com/watch?v=RvAiaLKi43Y</a> <a href="https://python-forum.io/">https://python-forum.io/</a> <a href="https://github.com/">https://github.com/</a>  <a href="https://blogs.perficient.com/2023/08/03/how-to-connect-nodejs-with-mysql-database/">https://blogs.perficient.com/2023/08/03/how-to-connect-nodejs-with-mysql-database/</a> <a href="https://www.w3schools.com/">https://www.w3schools.com/</a> <a href="https://chat.openai.com/">https://chat.openai.com/</a> <a href="https://github.com/">https://github.com/</a>  <a href="https://apachefriends.org/blog.html">https://apachefriends.org/blog.html</a>  <a href="https://chat.openai.com/">https://chat.openai.com/</a>
Autre utilisation	Histoire du Chili  Réécriture d'un rapport  Université de Valparaíso	<a href="https://fr.wikipedia.org/wiki/Histoire_du_Chili">https://fr.wikipedia.org/wiki/Histoire_du_Chili</a>  <a href="https://www.cidj.com/etudes-formations-alternance/avant-le-bac/comment-rediger-un-bon-rapport-de-stage">https://www.cidj.com/etudes-formations-alternance/avant-le-bac/comment-rediger-un-bon-rapport-de-stage</a> <a href="https://ch-fr.indeed.com/conseils-carriere/evolution-professionnelle/comment-rediger-un-rapport-de-stage">https://ch-fr.indeed.com/conseils-carriere/evolution-professionnelle/comment-rediger-un-rapport-de-stage</a> <a href="https://www.letudiant.fr/jobsstages/conseils-stages/article/exemple-de-rapport-de-stage-inspirez-vous-de-notre-modele.html">https://www.letudiant.fr/jobsstages/conseils-stages/article/exemple-de-rapport-de-stage-inspirez-vous-de-notre-modele.html</a>  <a href="https://uv.cl/">https://uv.cl/</a>

## Annexe E : Table des figures et illustrations utilisées dans le rapport

### IMG-1 Le Chili sur la carte du monde :

[https://t3.ftcdn.net/jpg/01/82/53/88/360\\_F\\_182538826\\_3IP4EwsXFH5MjgtfYiQMfApXf9xtVjyO.jpg](https://t3.ftcdn.net/jpg/01/82/53/88/360_F_182538826_3IP4EwsXFH5MjgtfYiQMfApXf9xtVjyO.jpg)

### IMG-2 Valparaiso :

« <https://www.google.com/maps/place/Valpara%C3%ADso/@-33.2205675,-71.4559604,9.64z/data=!4m6!3m5!1s0x9689dde3de20cec7:0xeb0a3a8cbfe19b76!8m2!3d-33.047238!4d-71.6126885!16zL20vMGo4bG4?hl=fr&entry=ttu> »

### IMG-3 L'université :

[https://upload.wikimedia.org/wikipedia/commons/thumb/2/2c/Universidad\\_de\\_Valpara%C3%ADso\\_-\\_Facultad\\_de\\_Ingenier%C3%A1\\_C%C3%A1lculo\\_y\\_Física\\_en\\_Chile.jpg/1024px-Universidad\\_de\\_Valpara%C3%ADso\\_-\\_Facultad\\_de\\_Ingenier%C3%A1\\_C%C3%A1lculo\\_y\\_Física\\_en\\_Chile.jpg](https://upload.wikimedia.org/wikipedia/commons/thumb/2/2c/Universidad_de_Valpara%C3%ADso_-_Facultad_de_Ingenier%C3%A1_C%C3%A1lculo_y_Física_en_Chile.jpg/1024px-Universidad_de_Valpara%C3%ADso_-_Facultad_de_Ingenier%C3%A1_C%C3%A1lculo_y_Física_en_Chile.jpg)

### IMG 4 – Ensemble des composants de base :

<https://www.dhm-online.com/4848534/sparkfun-inventors-kit-for-arduino-uno-v41.jpg>

### IMG-5 Capteur DHT11 :

<https://www.google.com/url?sa=i&url=https%3A%2F%2Fmakerbazar.in%2Fproducts%2Fhumidity-sensor-module-dht11&psig=AOvVaw3wYUd4BpeFrS207fThOvl&ust=1699544364348000&source=images&cd=vfe&opi=89978449&ved=0CBEQjRxqFwoTCNiLvKPetIIDFQAAAAAdAAAAABAE>

### IMG-6 Risque UV :

[https://www.inspq.qc.ca/sites/default/files/images/bise/indice\\_uv\\_figure1.gif](https://www.inspq.qc.ca/sites/default/files/images/bise/indice_uv_figure1.gif)

### IMG-7 Etalonnage UV :

<https://www.youtube.com/watch?v=zbHh3onXqBc>

## Annexe D : Programmes

Dans cette annexe, vous retrouverez tous les différents codes que j'ai pu créer pour mener à bien ce projet. Pour ce qui est du code web, celui-ci aura qu'un exemple pour les pages des graphiques, celle ou tous les graphiques sont affiché car pour les graphiques simples je reprends exactement la même fonction.

### Programme Arduino :

```
S17_ecranlcd

#include "DHT.h" //declaration library humidity/temperature sensor
#include <MQ135.h> //declaration library gas sensor
#include <LiquidCrystal.h> ///declaration library lcd

//humidity and temperature sensor parameters
#define DHTPIN 2
#define DHTTYPE DHT11

//pin initialization
const int luminositePin = A0;
const int airQualityPin = A1;
const int uvSensorPin = A2;

//initialization of library parameters
MQ135 gasSensor = MQ135(airQualityPin);
DHT dht(DHTPIN, DHTTYPE);
LiquidCrystal lcd(13,12,11,10,9,8);
void setup() {
//I/O parameter initialization
pinMode(uvSensorPin, INPUT);
pinMode(luminositePin, INPUT);
pinMode(airQualityPin, INPUT);
dht.begin();
lcd.begin(16,2);
Serial.begin(9600);
}

void loop() {
***** for temperature and humidity
float temperature = dht.readTemperature();
float humidity = dht.readHumidity();
```

```

//*****for gas

float airQualityValue = gasSensor.getPPM();
/**pour la luminosité
float luminosite = analogRead(luminositePin);

/** For UV sensor
int lecture = analogRead(uvSensorPin);
int taux = map(lecture,0,669,0,1023);
int uvIntensity;
if (taux<50){
    uvIntensity =0;
}
else if (taux<227){
    uvIntensity =1;
}
else if (taux<318){
    uvIntensity =2;
}
else if (taux<408){
    uvIntensity =3;
}
else if (taux<503){
    uvIntensity =4;
}
else if (taux<606){
    uvIntensity =5;
}
else if (taux<795){
    uvIntensity =7;
}
else if (taux<881){

}

//displaying variables in the series monitor
Serial.print(luminosite);
Serial.print(",");
Serial.print(temperature);
Serial.print(",");
Serial.print(humidity);
Serial.print(",");
Serial.print(airQualityValue);
Serial.print(",");
Serial.println(uvIntensity);

//converting int variables to float
int luminositeInt = int(luminosite);
int temperatureInt = int(temperature);
int humidityInt = int(humidity);
int airQualityValueInt = int(airQualityValue);
int uvIntensityInt = int(uvIntensity);

//data display on lcd screen
lcd.setCursor(0, 0);
lcd.print("L:");
lcd.print(luminositeInt);
lcd.print(" T:");
lcd.print(temperatureInt);
lcd.print(" H:");
lcd.print(humidityInt);

lcd.setCursor(0, 1);
lcd.print(" AQ:");
lcd.print(airQualityValueInt);
lcd.print(" UV:");

```

## Programme Python :

```

1  #library initialization
2  import mysql.connector as msqlc
3  from datetime import date, datetime
4  import serial
5  import time
6
7  #connexion to database
8  try:
9      bd = msqlc.connect(
10         host = "localhost",
11         port = 3306 ,
12         user = "root",
13         passwd = "",
14         database = "mydatabase"
15     )
16 except Exception:
17     print("connexion impossible à la base de données")
18
19 #mySql function
20 cursor = bd.cursor()
21 query = "INSERT INTO mydata (dateA, hoursA, temperature,Luminosity,Humidity,AirQuality,UVvalue) VALUES (%s, %s, %s, %s, %s,%s,%s)"
22
23
24 #loop for data retrieval and recording
25 with serial.Serial('COM5', 9600, timeout=0.05) as comarduino: #connection to serial monitor 9600 bauds
26     while True:
27         data = comarduino.readline()
28         print(data.decode('utf-8'))
29
30         values = data.decode('Utf-8').strip().split(',')
31     if len(values) == 5:
32         luminosity = values[0]
33         temperature = values[1]
34         humidity = values[2]
35         airQuality = values[3]
36         uvValue = values[4]
37
38         now = datetime.now()
39         date_acquisition = now.strftime("%Y-%m-%d")

```

```

2   #call to the mySql function just above
3   cursor.execute(query, (date_acquisition, heure_acquisition, temperature,luminosity, humidity,airQuality,uvValue))
4
5   bd.commit()
6
7
8   time.sleep(2)

```

## Code CSS :

```

1 body {
2     font-family: Arial, sans-serif;
3     margin: 0;
4     padding: 0;
5     background-color: #f5f5f5;
6 }
7
8 .container {
9     max-width: 1000px;
10    margin: auto;
11    padding: 20px;
12    display: flex;
13    flex-wrap: wrap;
14    justify-content: center;
15 }
16
17 .column {
18     flex: 1;
19     display: flex;
20     flex-direction: column;
21     margin: 10px;
22     border: 1px solid #ddd;
23     cursor: pointer;
24     transition: flex-grow 0.5s;
25 }
26
27 .fullscreen {
28     position: fixed;
29     top: 0;
30     left: 0;
31     width: 100vw;
32     height: 100vh;
33     z-index: 9999;
34 }
35
36 .hidden {
37     display: none;
38 }
39
40 #chartSelection {
41     margin-bottom: 10px;
42 }
43 ul {
44     list-style-type: none;
45     margin: 0;
46     padding: 0;
47     background-color: #fbe288;
48     overflow:hidden;
49 }
50 li a {
51     color : black;
52     padding: 20px;
53     text-decoration: none;
54     display:block;
55 }
56
57 li {
58     float: left;
59 }
60 li a:hover{
61     background-color: #EAC643;
62 }
63

```

## Page CSS avec tous les graphiques :

```

1 <?php
2     include("commun/menu.php");
3     include("commun/header.php");
4
5 ?>
6
7     <div class="container">
8         <div class="column">
9             <canvas id="temperatureChart"></canvas>
10            <canvas id="humiditeChart"></canvas>
11            <canvas id="luminositeChart"></canvas>
12        </div>
13        <div class="column">
14            <canvas id="airQualityChart"></canvas>
15            <canvas id="uvValueChart"></canvas>
16        </div>
17    </div>
18    <script>
19        fetch('data.php')
20            .then(response => response.json())
21            .then(data => {
22                const dates = data.dates;
23                const temperatures = data.temperatures;
24                const humidites = data.humiditys;
25                const luminosites = data.luminositys;
26                const airQualities = data.airQualities;
27                const uvValues = data.uvValue;
28
29                function createChart(id, label, data) {
30                    var ctx = document.getElementById(id).getContext('2d');
31                    return new Chart(ctx, {
32                        type: 'line',
33                        data: {
34                            labels: dates,
35                            datasets: [{
36                                label: label,
37                                data: data,
38                                borderColor: getRandomColor(),
39                                fill: false
40                            }]
41                        })
42                    });
43                }
44
45                var temperatureChart = createChart('temperatureChart', 'Température (°C)', temperatures);
46                var humiditeChart = createChart('humiditeChart', 'Humidité (%)', humidites);
47                var luminositeChart = createChart('luminositeChart', 'Luminosité', luminosites);
48                var airQualityChart = createChart('airQualityChart', 'Qualité de l\'air', airQualities);
49                var uvValueChart = createChart('uvValueChart', 'UV Value', uvValues);
50
51
52                function getRandomColor() {
53                    var letters = '0123456789ABCDEF';
54                    var color = '#';
55                    for (var i = 0; i < 6; i++) {
56                        color += letters[Math.floor(Math.random() * 16)];
57                    }
58                    return color;
59                }
60            });
61        </script>
62
63    <?php
64        include("commun/footer.php");
65
66 ?>
```

## Page d'acquisition de data :

```

2   <?php
3   //connection to database "mydatabase"
4   $host = 'localhost';
5   $user = 'root';
6   $password = '';
7   $database = 'mydatabase';
8
9   //Establish connection to MySQLi
10  $conn = new mysqli($host, $user, $password, $database);
11
12  //Check for the connection
13  if ($conn->connect_error) {
14      die('Connection failed: ' . $conn->connect_error);
15  }
16  //Function to retrieve data from my database
17  $sql = 'SELECT dateA, hoursA, temperature, Luminosity, Humidity, AirQuality, UVvalue FROM mydata';
18  $result = $conn->query($sql);
19
20  //storing data from my sensors in variable data
21  $data = array(
22      'dates' => array(),
23      'temperatures' => array(),
24      'luminositys' => array(),
25      'humiditys' => array(),
26      'airQualities' => array(),
27      'uvValue' => array()
28  );
29
30  //association between data arriving from my database and variables in my web code
31  if ($result->num_rows > 0) {
32      while ($row = $result->fetch_assoc()) {
33          $datetime = $row['dateA'] . ' ' . $row['hoursA'];
34          array_push($data['dates'], $datetime);
35          array_push($data['temperatures'], $row['temperature']);
36          array_push($data['humiditys'], $row['Humidity']);
37          array_push($data['luminositys'], $row['Luminosity']);
38          array_push($data['airQualities'], $row['AirQuality']);
39          array_push($data['uvValue'], $row['UVvalue']);
40      }
41  }
42
43  $conn->close();
44
45  header('Content-Type: application/json');
46  echo json_encode($data);
47  ?>

```