

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي والبحث العلمي
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

جامعة عين تموشنت بلحاج بوشعيب
Université- Aïn Témouchent -BELHADJ Bouchaib
معهدتكنولوجيا
Institut de Technologie



قسم هندسة الكهربائية
Département Génie Electrique

Mémoire

Présenté en vue de l'obtention du diplôme de Master en Réseaux de Telecommunications.
Spécialité : Réseaux de Telecommunications

Réalisation d'une station météorologique dans plateforme Arduino avec NodeMCU 8266 Wi-Fi

Réalisé par :

Mlle BECHERIF Sabrina Yamna.

Encadrant :

Dr. BEMMOUSSETE Chems eddine

Devant le jury composé de :

Dr BENGANA Abdel Fatih

Dr BENOSMAN Mourad

Présenté le 18/07/2021

REMERCIEMENTS

*Je tiens à remercier dans un premier temps, **ALLAH**, qui nous aide et nous donne la patience et le courage durant ces années d'étude.*

*Je tiens à remercier grandement mon encadrant Dr **BEMMOUSSAT Chems Eddine**, d'avoir accepté de diriger ce travail,*

Je tien à remercier Dr benosman mourad et Dr bengana abdel fatih d'avoir accepté de le jury et d'évaluer ce projet de fin d'études.

Je remercie également tous nos enseignants du Département Génie Électrique du Université d'Ain Témouchent Belhadj Bouchaïb, pour leur soutien et disponibilité.

Dédicace

A ma Chère Mère

A mon Père

Dont le mérite, les sacrifices et les qualités humaines m'ont permis de vivre ce jour.

A mes Frères.

Ce travail est également dédié à tous mes enseignants et à tous mes amis.

Sabrina Yamna.

Table des matières

I.	Introduction générale.....	1
II.	Chapitre I : les stations météorologiques:.....	2
II.1.	La Définition:	2
I.2.1 :	services de la station météo :.....	5
I.2.2 :	Les composants électriques de la station météo :.....	5
II.2.	Plateforme Arduino :.....	5
II.2.1.	Historique :	5
II.2.2.	La carte Arduino :.....	6
III.	Chapitre II : équipement logiciel utilisé	11
III.1 :	Présentation de équipement:.....	11
III.1.2 :	Nodemcu ESP8266:	11
III.1.2 :	Les capteurs :.....	13
III.1.2.1.	DHT_11 :	13
III.1.2.2.	Le capteur d'eau(ST045) :.....	14
III.1.2.3.	capteur de détection de neige/gouttes de pluie MH-RD :.....	15
III.2.	Présentation du logiciel :	16
III.2.1.	IDE Arduino :.....	16
III.2 .1.1.	Les boutons :	17
III .2.2.	Schéma électrique du montage sur Fritzing :	18
III.2.3.	HTML/CSS :	19
III.2.3.1.	HTML :.....	19
III.2.3.2.	Css :.....	21
IV.	Chapitre III : Notre solution.....	5
IV.1.	Shéma globale de notre solution :	5
IV.2.	Notre solution propose 3 service :.....	5
III.	Conclusion générale	35
	Bibliographie	38
IV.	Annexe(s).....	42

Table des matières

Figure I :Station météo manuelle	3
Figure II :Station météo automatique	3
Figure III :Chronologie des produits Arduino (Hughes, 2007)	6
Figure IV Image de la carte Arduino (CENTER, 2020).....	6
Figure V Les cartes Arduino disponible dans le marché Algerien.....	Error! Bookmark not defined.
Figure VII : carte NodeMCU ESP8266 V3.....	11
Figure VIII : ESP-12E (content.instructables, 2019)	12
Figure IX : ESP-12E (instructables, 2019).....	12
Figure X : le module ESP-8266 de Wi-Fi (instructables, ORIG, 2019).....	13
Figure XI : capteur de température et d'Humidité DHT11 (skywodd, 2017).....	14
Figure XII : capteur de niveau d'eau.	14
Figure XIII : Image capteur de pluie MH-RD.....	15
Figure XIV :ecran principal de l'IDE Arduino	17
Figure XV : Présentation des boutons	17
Figure XVI :ecran principal de fritzing	18
Figure XVII :Structure d'un document HTML	20
Figure XVIII :fichier HTML	23
Figure XIX :fichier HTML 2.....	23
Figure XX :fichier HTML 3	24
Figure XXI :une feuille de style CSS	25
Figure XXII : Schéma globale.....	5
Figure XXIII : montage de DHT11 avec l'esp8266.....	6
Figure XXIV :Résultats local_dht11	6
Figure XXV :Résultats sur le site.....	7
Figure XXVI : montage du capteur de niveau d'eau avec l'ESP8266.....	7
Figure XXVII :Résultats du niveau d'eau sur le moniteur serie	8
Figure XXVIII : Résultatsdu niveau d'eau sur le site web.....	8
Figure XXIX : montage du capteur de la pluie avec l'esp8266	9
Figure XXX : Resultats sur le monteur serie.....	9
Figure XXXI :Resultats sur le site web.....	10

[Introduction général]

I. Introduction générale

L'électronique à cette époque est partout, totalement partout et le domaine de l'électronique est toujours étendu à toutes choses que l'homme peut toucher ou avec lesquelles peut interagir. Elle est en développement constant, ce qui facilite la vie quotidienne humaine.

Notre projet intitulé «Réalisation d'une mini station météorologique dans plateforme Arduino avec NodMCU 8266 wifi», a pour but de concevoir une mini station météorologique qui peut être mobile ou bien installée dans des maisons, des bureaux et des administrations et qu'elle peut donner des mesures exactes sur le climat.

Dans notre travail nous avons réalisé une station météorologique qui mesure les phénomènes météorologiques existants du climat et de la météo (Pression (P), Température (ET), Densité de l'air (ρ), Contenu en eau (q)). Cette station sera connectée par Ethernet au même réseau internet des clients (PC, Téléphone mobile) pour consultation des résultats de la station. Dans l'ensemble, la station comprend plusieurs capteurs tels que: capteur de température, capteur d'humidité, capteur de pression, capteur d'eau et capteur anémomètre. Une carte Nodemcu 8266 a pour but de traiter les données mesurées de ces capteurs et transférer vers un site web à travers le module (Ethernet Shield) ESP 8266 intégré sur la même carte Nodemcu.

Notre mémoire est organisée en trois chapitres :

En commençant d'abord par le premier chapitre dans lequel nous présenterons la station météo, quelques généralités et l'historique de l'Arduino. Ensuite dans le deuxième chapitre nous parlerons des équipements et logiciels utilisés pour la réalisation de ce projet. Le dernier chapitre est consacré pour la conception, réalisation, les exécutions et les tests sur notre station météorologique.

Nous finalisons ce rapport de fin d'étude par une conclusion générale qui synthétise notre travail où nous citons nos résultats ainsi que nos perspectives concernant le rapport.

〔 Chapitre 1 : les stations météorologiques 〕

II. Chapitre I : les stations météorologiques:

II.1. La Définition:

Une station météo est un ensemble de capteurs qui enregistrent et fournies les mesures physiques et les paramètres météorologiques liés aux changements climatique, ces capteurs sont placés dans un caisson (résistant au vent et aux intempéries), offrant l'équilibre de la chaleur avec l'air et protège le thermomètre du rayonnement solaire.

Les variables à mesurer sont la température, la pression, la vitesse et la direction du vent, humidité, précipitations, hauteur et type de nuages, type et intensité précipitations et visibilité. La station peut inclure tous les capteurs ou une partie . Ces informations dépendent de leur nature : météorologie agricole, aéroport, météo routière, climat, etc.

Les stations météorologiques sont généralement situées à des coordonnées fixes sur l'île Et sur le continent, mais ils peuvent aussi être utilisés dans les avions, les navires (surtout navire météo).

Ils peuvent également être déplacés à d'autres fins Détails tels que des expériences scientifiques ou des conditions déterminantes Météo lors de catastrophes telles que les incendies de forêt. Les données que nous obtenons peuvent être Dans le cas d'une station automatique, il est envoyé directement sous forme de bulletin météo, Ou faites à partir de l'observation de l'observateur humain. (KARA LYDIA, 2018, p. 02)[1]

Il existe deux types de station météorologique : manuelle et automatique et elles sont définies comme suit :

➤ **Station météo manuelle :**

La mesure de cette station est effectuée par un technicien, les stations météorologiques font des observations météorologiques régulières,



Figure I :Station météo manuelle

➤ **Station météo automatique** : (weather stations, 2018)[2]

Développés pour les endroits difficiles d'accès (zones isolées ou bouées météorologiques marines) .une station automatique est une station dont les capteurs rapportent une série de données météorologiques sans intervention manuelle.

L'instrument est généralement le meme dans les deux cas, mais certaines observations sont plus fiables que l'observateur humain.

Par exemple, il est plus facile de remarquer la quantité et le type de précipitations que les humains à l'aide d'un instrument électronique.

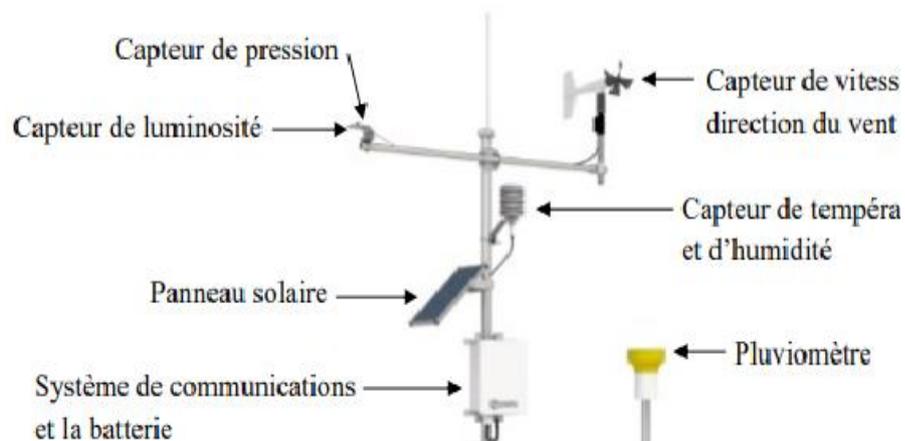


Figure II :Station météo automatique

I.2.1 : services de la station météo :

Il existe de nombreux services mondiaux, comme les sites Web ou les applications pour appareils intelligents, qui permettent de surveiller la météo en fonction de plusieurs informations, et chaque service varie en fonction des informations qu'ils fournissent, de l'expérience d'utilisation et de plusieurs autres données. (MAHFOUD KHA, 2019)[3]

- Vétéran Yahoo Météo ;
- Service AccuWeather ;
- un service conditions météorologiques chaîne ;
- Météo de l'Arabie .

I.2.2 : Les composants électriques de la station météo :

La station est composée des éléments suivants :

- Capteur de pression ;
- Un détecteur de pluie ;
- Un capteur de température et d'humidité ;
- Un capteur de niveau d'eau ;
- Capteur d'Ensoleillement ;
- Girouette ;
- Capteur de Vitesse du Vent .

II.2. Plateforme Arduino :

II.2.1. Historique :

Lancé à l'origine comme un projet de recherche par Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino et David Mellis à l'Institut de design d'interaction d'Ivrea au début des années 2000, il s'appuie sur le projet Processing, un langage pour apprendre à coder dans le contexte des arts visuels développé par Casey Reas et Ben Fry ainsi que sur un projet de thèse de Hernando Barragan sur la carte Wiring.

La première carte Arduino a été introduite en 2005 pour aider les étudiants en design - qui n'avaient aucune expérience préalable en électronique ou en programmation de microcontrôleurs - à créer des prototypes fonctionnels reliant le monde physique au monde numérique. Depuis, elle est devenue l'outil de prototypage électronique le plus populaire, utilisé par les ingénieurs et même les grandes entreprises (arduino CENTER, 2020)[4]

Board name	Year	Microcontroller	Board name	Year	Microcontroller
Diecimila	2007	ATmega168V	Mega 2560	2010	ATmega2560
LilyPad	2007	ATmega168V/ATmega328V	Uno	2010	ATmega328P
Nano	2008	ATmega328/ATmega168	Ethernet	2011	ATmega328
Mini	2008	ATmega168	Mega ADK	2011	ATmega2560
Mini Pro	2008	ATmega328	Leonardo	2012	ATmega32U4
Duemilanove	2008	ATmega168/ATmega328	Esplora	2012	ATmega32U4
Mega	2009	ATmega1280	Micro	2012	ATmega32U4
Fio	2010	ATmega328P	Yún	2013	ATmega32U4 + Linino

Figure III :Chronologie des produits Arduino (Hughes, 2007)

II.2.2. La carte Arduino :

L'Arduino Uno est un microcontrôleur programmable qui permet, comme son nom l'indique, de contrôler des éléments mécaniques : systèmes, lumières, moteurs, etc. Cette carte électronique permet donc à son utilisateur de programmer facilement des choses et de créer des mécanismes automatisés, sans avoir de connaissances particulières en programmation. Il est un outil pensé et destiné aux inventeurs, artistes ou amateurs qui souhaitent créer leur propre système automatique en le codant de toute pièce.

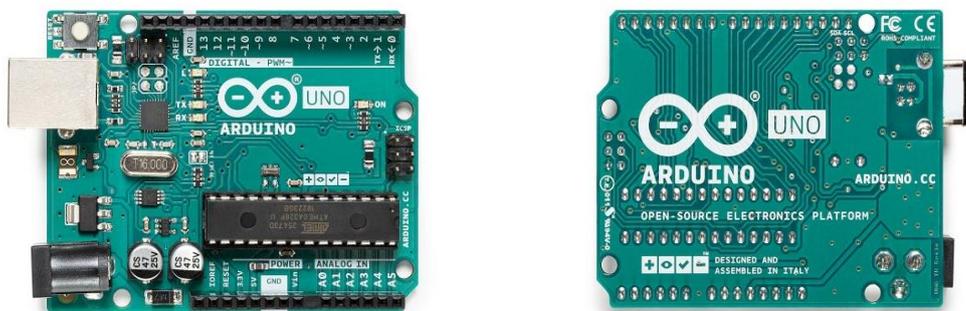


Figure IV Image de la carte Arduino (CENTER, 2020)

**Chapitre 2 : l'équipements et les
logiciels utilisé**

III. Chapitre II : équipement logiciel utilisé

III.1 : Présentation de équipement:

III.1.2 : Nodemcu ESP8266:

NodeMCU ESP8266 est un microcontrôleur avec un module ESP8266 Wi-Fi intégré dans la même carte de développement ,cette carte est open-source¹ (Guidelines, 2007) [5], il a été développé par le fabricant Espressif Systems. Ce microcontrôleur similaire au Arduino microcontrôleur vous pouvez le reprogrammer, mais avec les fonctionnalités complémentaires.

NodeMCU est spécialement conçu pour les applications basées sur l'IoT².

- NodeMCU :

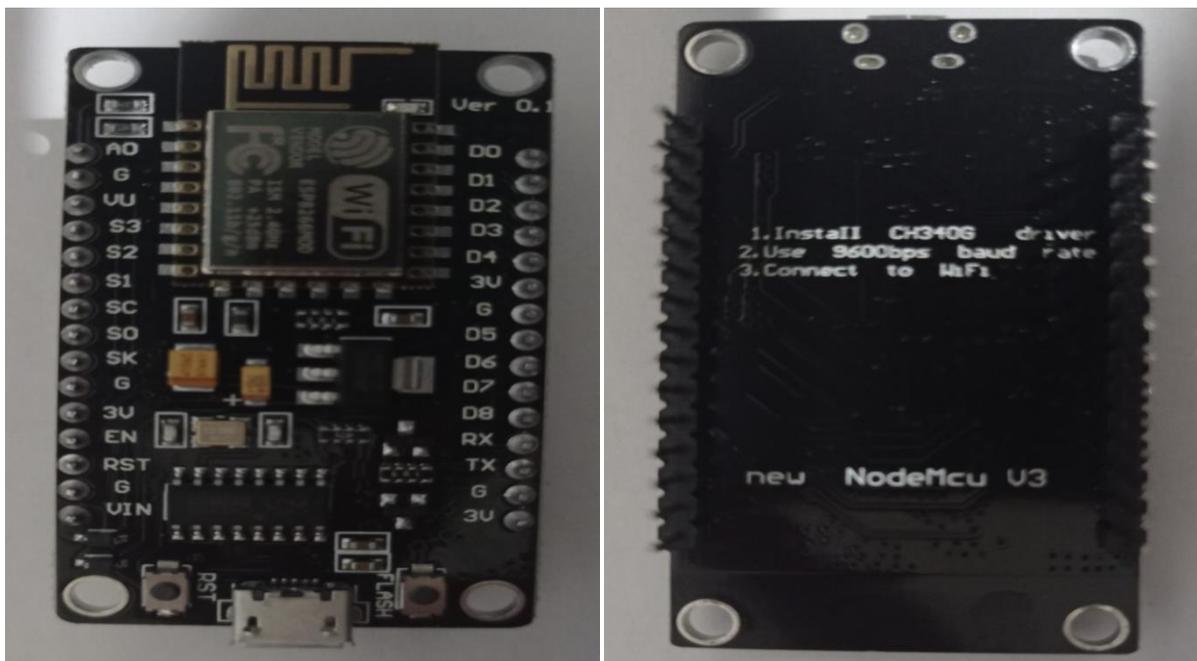


Figure V : carte NodeMCU ESP8266 V3

¹**Open-Source** : ou code source ouvert signifie l'accès au code source. Les conditions de distribution des logiciels open source doivent respecter les critères suivants : Redistribution gratuite, code source..etc (<https://opensource.org/osd>) .

²**IoT** : (en anglais Internet of Things), L'Internet des objets est l'interconnexion entre l'Internet et des objets comme une maison intelligente.

- **ESP-12E :**

C'est un module Wi-Fi miniature présent sur le marché et est utilisé pour établir une connexion réseau sans fil pour un microcontrôleur ou un processeur. Le cœur de l'ESP-12E est ESP8266EX, qui est un SoC (System on Chip) sans fil à haute intégration. Il permet d'intégrer des capacités Wi-Fi aux systèmes ou de fonctionner comme une application autonome. (components101, 2018)

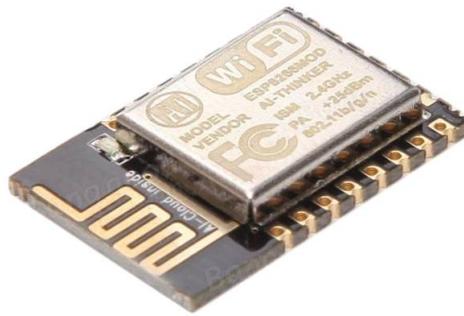


Figure VI : ESP-12E (content.instructables, 2019)

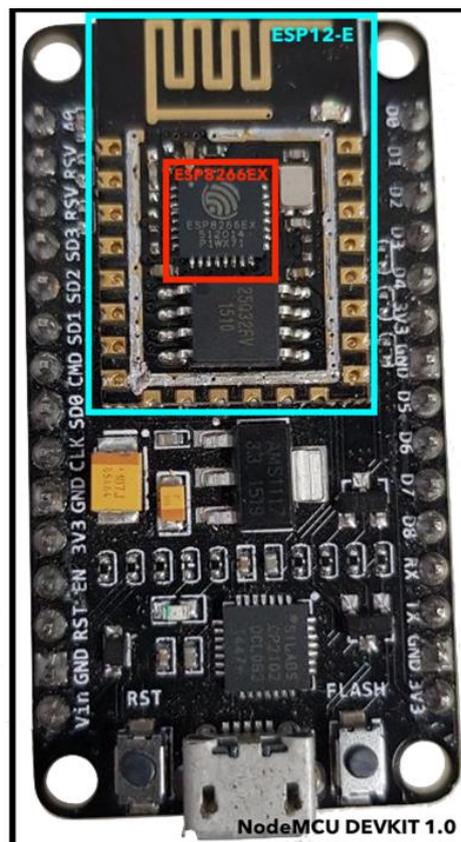


Figure VII : ESP-12E (instructables, 2019)

- **ESP-8266 :**



Figure VIII : le module ESP-8266 de Wi-Fi (instructables, ORIG, 2019)

Applications de NodeMCU:

- Station météo,
- Applications IoT,
- Appareils électroménagers,
- Applications jouets et jeux,
- Systèmes de contrôle sans fil,
- Automatisation de la maison,
- Balises d'identification de sécurité.

III.1.2 : Les capteurs :

Pour construire notre station météo, nous devons d'abord sélectionner des composants sensibles aux phénomènes physiques que nous voulons mesurer. Nous définissons la sélection sur les capteurs de température, d'humidité, de niveau d'eau et de pluie.

III.1.2.1.DHT_11 :

Le dht11 est un capteur de température et d'humiditi de sortie numérique pré-calibré .

Ce module envoie des donnée sous forme de train d'impulsion d'une période de temps spécifique . (skywodd, 2017)[6]

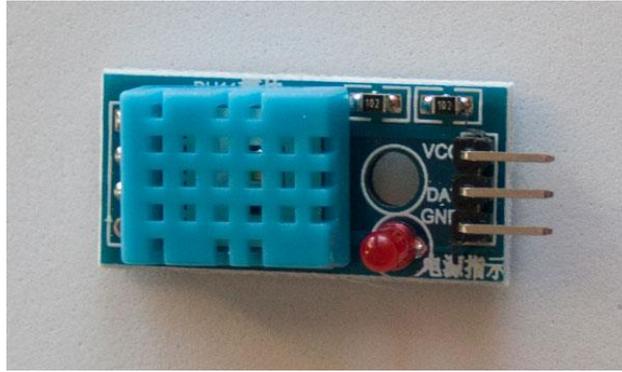


Figure IX : capteur de température et d'Humidité DHT11 (skywodd, 2017)

- **Les caractéristiques :** DHT11 ou RHT01
- Humidité (relative %) : 20 ~ 80% ;
- Précision (humidité) : +/- 5% ;
- Température : 0 ~ +50°C ;
- Précision (température) : +/- 2°C ;
- Fréquence mesure max : 1Hz (1 mesure par seconde) ;
- Tension d'alimentation : 3 ~ 5 volts ;
- Stabilité à long terme : +/- 1% par an.

III.1.2.2. Le capteur d'eau (ST045) :

Capteur de niveau d'eau Le capteur de niveau d'eau est un capteur d'identification de haut niveau / gouttelettes facile à utiliser et économique. Il est réalisé en ayant une série de fils exposés parallèles qui suivent le volume mesuré de gouttelettes / d'eau pour déterminer le niveau d'eau . (gotronic store center, 2018)[7]



Figure X : capteur de niveau d'eau.

- **Les caractéristiques :**
- Poids : 3.5g ;
- Taille : 62*20*8mm ;
- Zone de détection : 40mm*16mm ;
- Tension de travail : DC3-5V ;
- Courant de travail : <20mA ;
- Type de capteur : Analogique/Simulation ;
- Température de fonctionnement : 10 ~ 30celsius .

III.1.2.3. capteur de détection de neige/gouttes de pluie MH-RD :

Ce capteur est très adapté pour détecter les précipitations dans les projets domotiques, ou par exemple dans le cadre de projets agricoles. Le capteur est livré avec un module d'interface (utilisé pour le réglage de la sensibilité et la sélection entre la sortie numérique et la sortie analogique) et son câble ou cavalier pour la connexion.

Lorsque des gouttelettes d'eau court-circuitent les pistes du circuit imprimé, le capteur de pluie détecte les gouttelettes d'eau. Le capteur est comme une résistance variable, il change d'état : 100k ohms lorsqu'il est mouillé, 2M ohms lorsqu'il est sec. Il possède 2 sorties, une sortie numérique 1 ou 0 et une sortie analogique 0 à 1023 (capteurs détecteurs). [8]



Figure XI : Image capteur de pluie MH-RD

- **Les caractéristiques :**
- Tension : 3,3V-5V ;
- Dimension du capteur : ~ 40 x 54 mm ;
- Pistes sur capteur : Double face. Les deux faces du capteur sont active ;.
- Potentiomètre de sensibilité et 2 LED de contrôle intégrées au module ;
- Module livré avec deux câbles Femelle / Femelle.

III.2. Présentation du logiciel :

III.2.1. IDE Arduino :

Integrated **D**evelopment **E**nvironment (en français « environnement de développement ») est un type de logiciel qui regroupe des outils qui permettent le développement d'autres logiciels, tels que des plays mobiles, des logiciels d'ordinateur ou de locker de jeux, des positions Web, etc.; et de créer des bibliothèques ou des frameworks.

L'interface d'Arduino IDE est très simple. Il a un éditeur de code avec coloration syntaxique (1) et barre d'outils rapide (2). ceux-ci sont les deux parties les plus importantes de l'interface sont celles que vous utilisez le plus souvent.

Il existe également une barre de menu plus classique (3) pour accéder aux fonctionnalités avancées de l'IDE. Enfin, il existe une console qui affiche les résultats de la compilation (4) code source, opérations de la carte, etc.

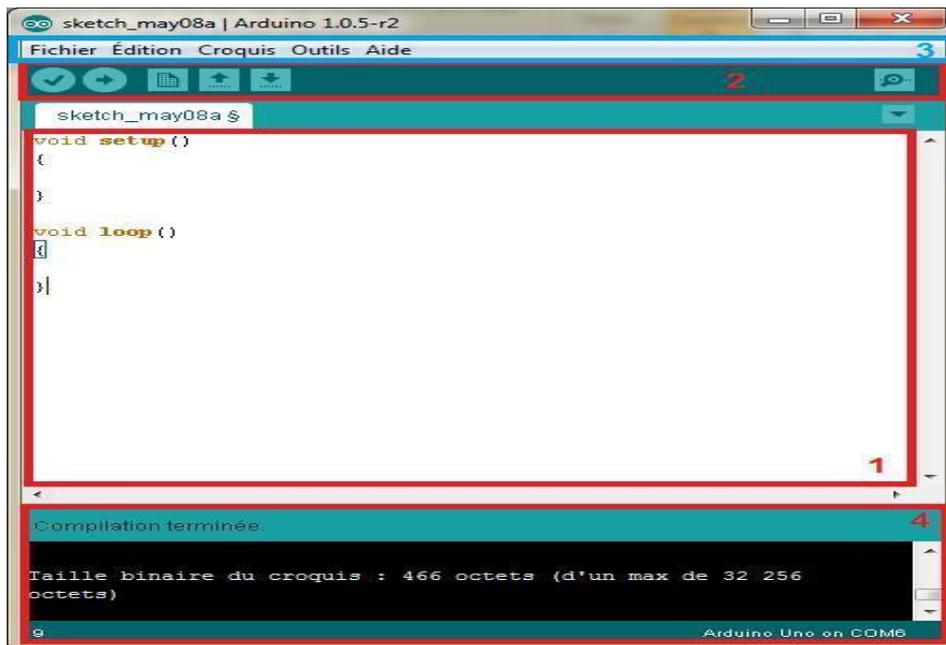


Figure XII :ecran principal de l'IDE Arduino

III.2 .1.1. Les boutons :

Voyons à présent à quoi servent les boutons numérotés en rouge.



Figure XIII : Présentation des boutons

- Bouton 1 (verify): Ce bouton permet de vérifier le programme, il actionne un module qui cherche les erreurs dans votre programme.
- Bouton 2 (upload): Compiler et envoyer le programme vers la carte.
- Bouton 3 (new): Créer un nouveau fichier.
- Bouton 4 (open): Charger un programme existant.
- Bouton 5 (save): Sauvegarder le programme en cours

- Bouton 6 (serial monitor): Permet d'accéder au port série (en RX/TX).

III .2.2. logiciel de Fritzing :

Le logiciel Fritzing est un outil de création des circuits électroniques, il permet aussi l'édition de circuit imprimé, il est disponible gratuitement sur internet.

Il a notamment pour vocation de favoriser l'échange de circuits électroniques libres et d'accompagner l'apprentissage de la stéréotypie de circuits.

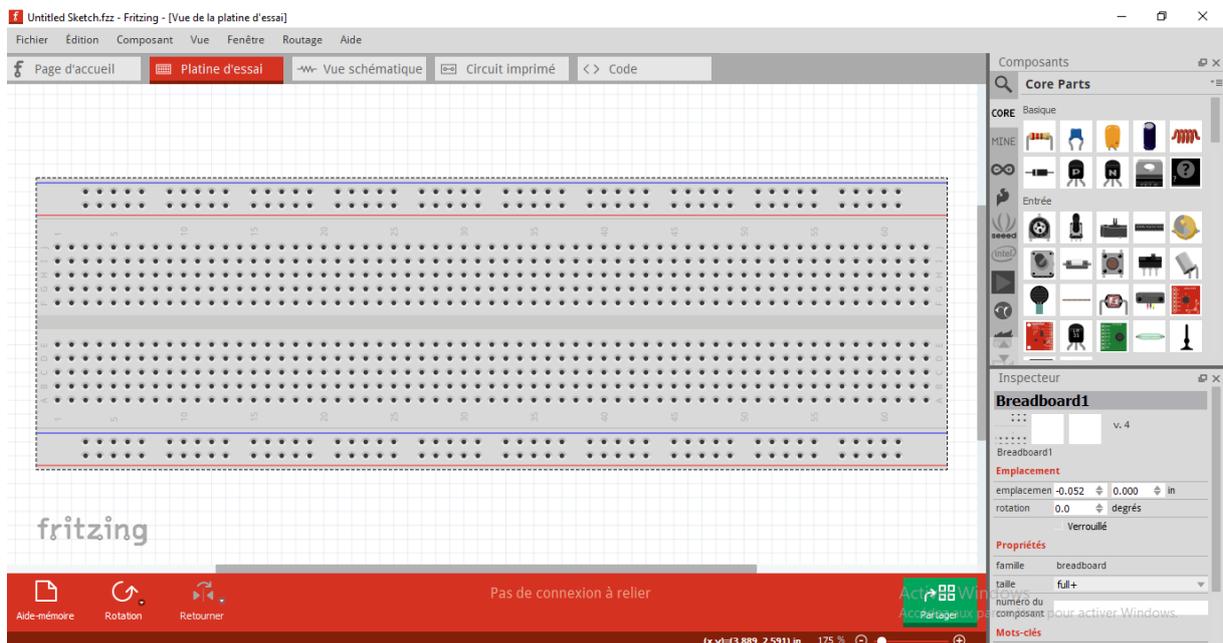


Figure XIV :ecran principal de fritzing

Le logiciel comporte trois vues principales:

La « Platine d'essai », où l'on voit les composants tels qu'ils sont dans la réalité— et où l'on construit le montage.

La « Vue schématique », représentant le schéma fonctionnel du circuit.

La « Circuit imprimé », représentant la vue du circuit imprimé tel qu'il sera sorti— en PDF pour être imprimé (ABDELKADER, 2018).[9]

III.2.3. HTML/CSS :

III.2.3.1.HTML :

- **A l'origine : le langage HTML**

C'est un langage de description permet de structurer et d'afficher différents objets sur un écran (qu'on appellera abusivement "page"). Ces objets peuvent être du texte, des tableaux, des images voir de la vidéo et des sons. Ils peuvent être aussi les éléments devenus classiques des environnements graphiques, à savoir les boutons, pop-listes, listes à ascenseurs, boîtes de dialogue... Sur la plupart de ces objets, il est possible de placer des liens qui vont permettre de se connecter à d'autres pages. Ce langage est donc à l'origine essentiellement statique. Sur le serveur, le fichier HTML décrit une "scène" qui sera envoyée à la demande vers un client (le browser, logiciel de navigation présent sur le poste de consultation). Ce client va décoder les instructions HTML pour afficher la scène qui restera ensuite la même ; seule une nouvelle requête vers le serveur sera à même d'en modifier l'apparence. (Philippe Chaléat, 2002, p. 1)[10]

- **HTML (HyperText Markup Language) :**

il a fait son apparition dès 1991 lors du lancement du Web. Son rôle est de gérer et organiser le contenu. C'est donc en HTML que vous écrirez ce qui doit être affiché sur la page : du texte, des liens, des images... Vous direz par exemple : « Ceci est mon titre, ceci est mon menu, voici le texte principal de la page, voici une image à afficher, etc. ». (Nebra, 2013, p. 11)[11]

Le HyperText Markup Language, généralement abrégé HTML ou dans sa dernière version HTML5, est le langage de balisage conçu pour représenter les pages web.

Ce langage permet :

- d'écrire de l'hypertexte, d'où son nom,
- de structurer sémantiquement la page,
- de mettre en forme le contenu,
- de créer des formulaires de saisie,
- d'inclure des ressources multimédias dont des images, des vidéos, et des programmes informatiques,
- de créer des documents interopérables avec des équipements très variés de manière conforme aux exigences de l'accessibilité du web (Wikipédia, 2021)[12]



Figure XV :Structure d'un document HTML

- **Les balises**

Les pages HTML sont remplies de ce qu'on appelle des balises. Celles-ci sont invisibles à l'écran pour vos visiteurs. Les balises se repèrent facilement. Elles sont entourées de chevrons , c'est-à-dire des symboles < et >, comme ceci : À quoi est-ce qu'elles servent ? Elles indiquent la nature du texte qu'elles encadrent. Elles veulent dire par exemple : 'Ceci est le titre de la page ', ' Ceci est une image', ' Ceci est un paragraphe de texte ', etc. On distingue deux types de balises : les balises en paires et les balises orphelines (NEBRA, 2013, pp. 21-22).[13]

- **Les balises en paires :**

Elles s'ouvrent, contiennent du texte, et se ferment plus loin. Voici à quoi elles ressemblent :

1 < titre > Ceci est un titre< /titre >

On distingue une balise ouvrante (< titre >) et une balise fermante (< /titre >) qui indique que le titre se termine.

- **Les balises orphelines**

Ce sont des balises qui servent le plus souvent à insérer un élément à un endroit précis (par exemple une image). Il n'est pas nécessaire de délimiter le début et la fin de l'image, on veut juste dire à l'ordinateur « Insère une image ici ». Une balise orpheline s'écrit comme ceci :

```
1 <image />
```

- **Les attributs**

Les attributs sont un peu les options des balises. Ils viennent les compléter pour donner des informations supplémentaires. L'attribut se place après le nom de la balise ouvrante et a le plus souvent une valeur, comme ceci :

```
1 <balise attribut =" valeur ">
```

À quoi cela sert-il ? Prenons la balise que nous venons de voir. Seule, elle ne sert pas à grand chose. On pourrait rajouter un attribut qui indique le nom de l'image à charger :

```
1 <imagenom =" photo .jpg" />
```

III.2.3.2.Css :

Les feuilles de styles (en anglais "*Cascading Style Sheets*", abrégé CSS) sont un langage qui permet de gérer la présentation d'une page Web. Le langage CSS est une recommandation du World Wide Web Consortium (W3C), au même titre que HTML ou XML.

Les styles permettent de définir des règles appliquées à un ou plusieurs documents HTML. Ces règles portent sur le positionnement des éléments, l'alignement, les polices de caractères, les couleurs, les marges et espacements, les bordures, les images de fond, etc.

Le but de CSS est séparer la structure d'un document HTML et sa présentation. En effet, avec HTML, on peut définir à la fois la structure (le contenu et la hiérarchie entre les différentes parties d'un document) et la présentation. Mais cela pose quelques problèmes. Avec le couple HTML/CSS, on peut créer des pages web où la structure du document se trouve dans le fichier HTML tandis que la présentation se situe dans un fichier CSS.

Avec CSS on peut par exemple définir un ensemble de règles stylistiques communes à toutes les pages d'un site internet. Cela facilite ainsi la modification de la présentation d'un site entier. CSS permet

aussi de définir des règles différentes pour chaque support d'affichage (un navigateur classique, une télévision, un support mobile, un lecteur braille...). CSS permet aussi d'améliorer l'accessibilité des documents web.

De plus, CSS ajoute des fonctionnalités nouvelles par rapport à HTML au point de vue du style. En effet, HTML permet une gestion assez sommaire du style des documents. (futura, 2020)[14]

- **À quoi sert le CSS ? :**

Le CSS permet donc de définir l'esthétique d'un site web. Sans CSS, une page web ne serait qu'une succession d'éléments noirs sur fond blanc les uns à la suite des autres.

C'est avec du code CSS que l'on définit par exemple :

- la police d'un texte et son aspect (couleur, taille, etc.);
- les marges et le rembourrage (padding) entourant un élément;
- l'apparence d'un menu;
- la création d'éléments géométriques;
- la mise en place de grilles d'images;
- l'ajout de bordure;
- l'application de différents effets ou animations simples;
- le changement de présentation d'un site selon l'appareil utilisé par l'utilisateur (téléphone portable, ordinateur, tablette);
- la manière dont on navigue dans un site internet. (futura, 2020)[15]
- **Où écrit-on le CSS ?** (Nebra, 2013, p. 68) :
Vous avez le choix car on peut écrire du code en langage CSS à trois endroits différents :
 - dans un fichier .css (méthode la plus recommandée) ;

```

1 | <!DOCTYPE html>
2 | <html>
3 |   <head>
4 |     <meta charset="utf-8" />
5 |     <link rel="stylesheet" href="style.css" />
6 |     <title>Premiers tests du CSS</title>
7 |   </head>
8 |
9 |   <body>
10 |     <h1>Mon super site</h1>
11 |
12 |     <p>Bonjour et bienvenue sur mon site !</p>
13 |     <p>Pour le moment, mon site est un peu <em>vide</em>.
14 |       Patientez encore un peu !</p>
15 |   </body>
16 | </html>

```

Figure XVI :fichier HTML

Le contenu de la ligne 5: c'est elle qui indique que ce fichier HTML est associé à un fichier appelé style.css et chargé de la mise en forme. Enregistrez ce fichier sous le nom que vous voulez (par exemple page.html). Pour le moment, rien d'extraordinaire à part la nouvelle balise que nous avons ajoutée.

- dans l'en-tête <head> du fichier HTML ;

Il existe une autre méthode pour utiliser du CSS : cela consiste à insérer le code CSS directement dans une balise .

Voici comment on peut obtenir exactement le même résultat avec un seul fichier .html qui contient le code CSS (lignes 5 à 10) :

```

1 | <!DOCTYPE html>
2 | <html>
3 |   <head>
4 |     <meta charset="utf-8" />
5 |     <style>
6 |       p
7 |       {
8 |         color: blue;
9 |       }
10 |    </style>
11 |    <title>Premiers tests du CSS</title>
12 |  </head>
13 |
14 |  <body>
15 |    <h1>Mon super site</h1>
16 |
17 |    <p>Bonjour et bienvenue sur mon site !</p>
18 |    <p>Pour le moment, mon site est un peu <em>vide</em>.
19 |      Patientez encore un peu !</p>
20 |  </body>
21 | </html>

```

Figure XVII :fichier HTML 2

- directement dans les balises du fichier HTML via un attribut style (méthode la moins recommandée).

Dernière méthode, à manipuler avec précaution : vous pouvez ajouter un attribut style à n'importe quelle balise. Vous insérez votre code CSS directement dans cet attribut :

```
1 | <!DOCTYPE html >
2 | <html >
3 |   <head >
4 |     <meta charset="utf-8" />
5 |     <title>Premiers tests du CSS</title>
6 |   </head >
7 |
8 |   <body >
9 |     <h1>Mon super site</h1>
10 |
11 |     <p style="color: blue;" >Bonjour et bienvenue sur mon
12 |       site !</p>
13 |     <p>Pour le moment, mon site est un peu <em>vide</em>
14 |       Patientez encore un peu !</p>
15 |   </body >
16 | </html >
```

Figure XVIII : fichier HTML 3

- **Selecteurs css (sélectionner une balise):**

Le CSS va nous permettre de mettre en forme nos contenus HTML en appliquant des styles aux différents éléments. Cependant, pour appliquer un style particulier à un ou plusieurs éléments HTML en CSS, il va avant tout falloir les cibler, c'est-à-dire indiquer avec précision à quels éléments doivent s'appliquer les styles créés en CSS.

Il existe de nombreux types de sélecteurs CSS et autant de moyens de cibler des contenus HTML en CSS :

- On peut cibler un type d'élément (comme l'élément p par exemple) en précisant le nom générique du type d'élément auquel on souhaite appliquer certains styles en CSS (auquel cas les styles s'appliqueront à tous les éléments de ce type) ;
- On peut donner un nom précis à un élément en HTML à l'aide d'un attribut class ou id par exemple et ensuite cibler l'élément ou les éléments en CSS grâce à leurs attributs ; (GIRAUD, 2019)
- On peut utiliser ce qu'on appelle des sélecteur CSS combinateurs, c'est-à-dire combiner des sélecteurs CSS simples pour en créer de complexes nous permettant de cibler des contenus HTML de manière très précise ;
- On peut encore cibler les éléments HTML grâce au fait qu'ils possèdent un attribut en particulier ou qu'un attribut possède une valeur ;
- On peut cibler les éléments HTML en CSS en utilisant des pseudo classes et ainsi n'appliquer certains styles à ces éléments que lors d'un état particulier ;
- On peut cibler les éléments HTML en utilisant les pseudo éléments en CSS et donc n'appliquer des styles qu'à une certaine partie de l'élément sélectionné.

- **Appliquer un style :**

Dans un code CSS comme celui-ci, on trouve trois éléments différents :

Des noms de balises : on écrit les noms des balises dont on veut modifier l'apparence. Par exemple, si je veux modifier l'apparence de tous les paragraphes , je dois écrire p.

Des propriétés CSS : les ‘ effets de style ‘ de la page sont rangés dans des propriétés. Il y a par exemple la propriété color qui permet d'indiquer la couleur du texte, font-size qui permet d'indiquer la taille du texte, etc. Il y a beaucoup de propriétés CSS et, comme je vous l'ai dit, je ne vous obligerai pas à les connaître toutes par coeur.

Les valeurs : pour chaque propriété CSS, on doit indiquer une valeur. Par exemple, pour la propriété color, il faut indiquer le nom de la couleur. Pour font-size, il faut indiquer quelle taille on veut, etc (NEBRA, 2013, p. 74)[16]

```
1 | balise1
2 | {
3 |     propriete1: valeur1;
4 |     propriete2: valeur2;
```

Figure XIX :une feuille de style CSS

Chapitre 3 : notre solution

IV.Chapitre III : Notre solution

IV.1.Shéma globale de notre solution :

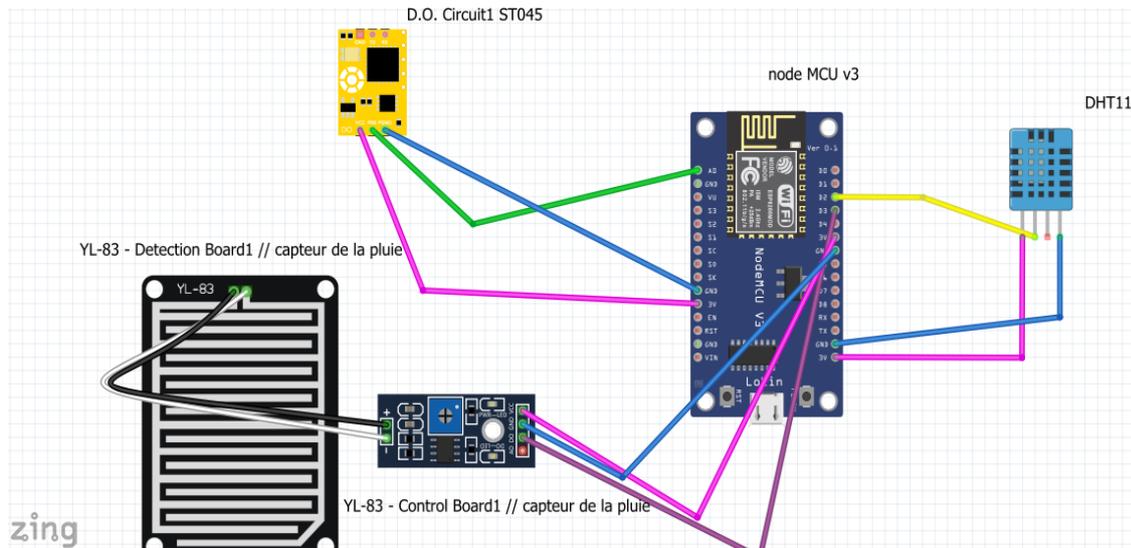


Figure XX : Schéma globale

Notre schéma englobe tous capteurs utilisées durant notre projet ; nous avons fait le cablage entre nos capteurs et le nodeMCU .

IV.2.Notre solution propose 3 service :

Après avoir connecté les différents capteurs (DHT11, ST045 et capteur de pluie) avec le nodeMCU nous avons injecté le programme dans le microcontrôleur qui sera responsable du traitement des informations collectées par les capteurs ensuite les résultats seront affichés sur moniteur série et notre site web .

- ✓ Capteur de TEMPERATURE ET HUMIDITE :
Pour la mesure de la température et l'humidité nous avons utilisé le capteur DHT11 , nous avons mis en point le montage qui consiste le branchement du DHT11 et l'ESP8266 .



Figure XXI : montage de DHT11 avec l'esp8266

Durant nos tests , les resultats de notre capteur seront affiches dans le moniteur serie en premier lieu .

```

COM6
21:00:29.909 -> Temperature: 32.20°C
21:00:29.909 -> Humidity: 40.00%
21:00:30.887 -> Temperature: 32.20°C
21:00:30.887 -> Humidity: 40.00%
21:00:31.910 -> Temperature: 32.20°C
21:00:31.910 -> Humidity: 40.00%
21:00:32.936 -> Temperature: 32.20°C
21:00:32.936 -> Humidity: 40.00%
21:00:33.952 -> Temperature: 32.20°C
21:00:33.952 -> Humidity: 40.00%
21:00:34.973 -> Temperature: 32.20°C
21:00:34.973 -> Humidity: 40.00%
21:00:35.989 -> Temperature: 32.20°C
21:00:35.989 -> Humidity: 40.00%
21:00:36.994 -> Temperature: 32.20°C
21:00:36.994 -> Humidity: 40.00%
21:00:38.005 -> Temperature: 32.20°C
21:00:38.005 -> Humidity: 41.00%
21:00:39.001 -> Temperature: 32.20°C
21:00:39.001 -> Humidity: 41.00%
21:00:40.014 -> Temperature: 32.30°C
21:00:40.014 -> Humidity: 89.00%
21:00:41.028 -> Temperature: 32.30°C
21:00:41.028 -> Humidity: 89.00%
21:00:42.043 -> Temperature: 32.60°C
21:00:42.043 -> Humidity: 95.00%

```

Figure XXII :Résultats local_dht11

Ensuite nous avons ajouté quelques commandes à notre programme pour envoyer les résultats sur notre site web. Les résultats seront affichés comme sur la figure ci-dessous :

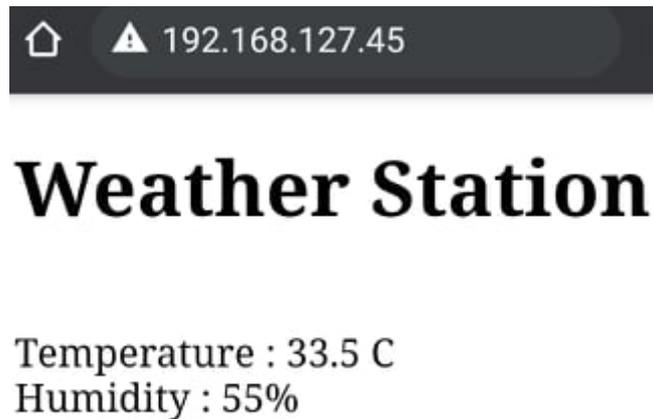


Figure XXIII : Résultats sur le site

- ✓ Capteur de niveau d'eau :
Ce capteur est nécessaire en cas de fortes pluies ou bien des averses pour prévenir les inondations. Le montage de ce dernier avec l'esp8266 est représenté dans la figure suivante :

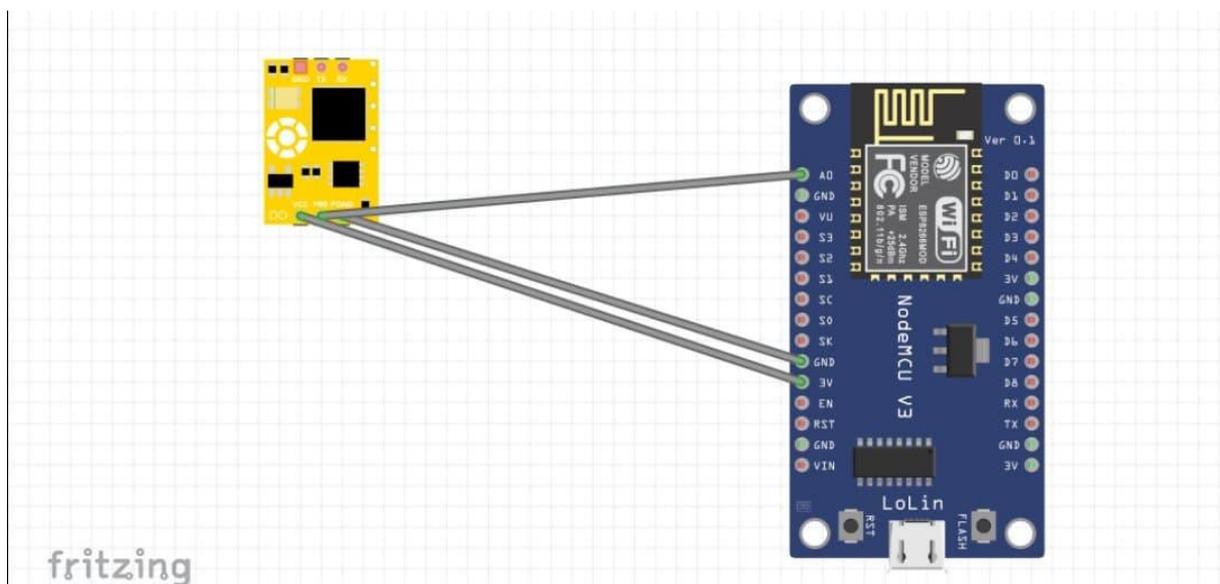
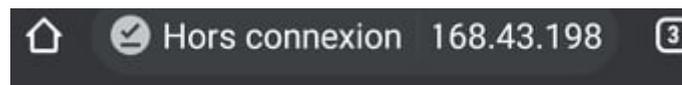


Figure XXIV : montage du capteur de niveau d'eau avec l'ESP8266

```
COM6
21:13:26.249 -> 12
21:13:27.228 -> 12
21:13:28.250 -> 12
21:13:29.270 -> 10
21:13:30.248 -> 57
21:13:31.232 -> 46
21:13:32.263 -> 78
21:13:33.246 -> 117
21:13:34.272 -> 13
21:13:35.256 -> 75
21:13:36.242 -> 12
21:13:37.230 -> 46
21:13:38.258 -> 62
21:13:39.241 -> 39
21:13:40.270 -> 12
21:13:41.259 -> 12
21:13:42.229 -> 12
21:13:43.254 -> 12
21:13:44.234 -> 87
21:13:45.265 -> 12
21:13:46.254 -> 12
21:13:47.240 -> 12
21:13:48.253 -> 31
```

Figure XXV :Résultats du niveau d'eau sur le moniteur serie



Station Météorologique

Niveau d'eau :3

Figure XXVI : Résultatsdu niveau d'eau sur le site web

Comme nous avons vu les résultats sont affichés sur notre site web et le moniteur serie .

✓ Capteur de la pluie :

La figure ci-dessous represnete le montage du capteur de pluie avec l'esp8266 :

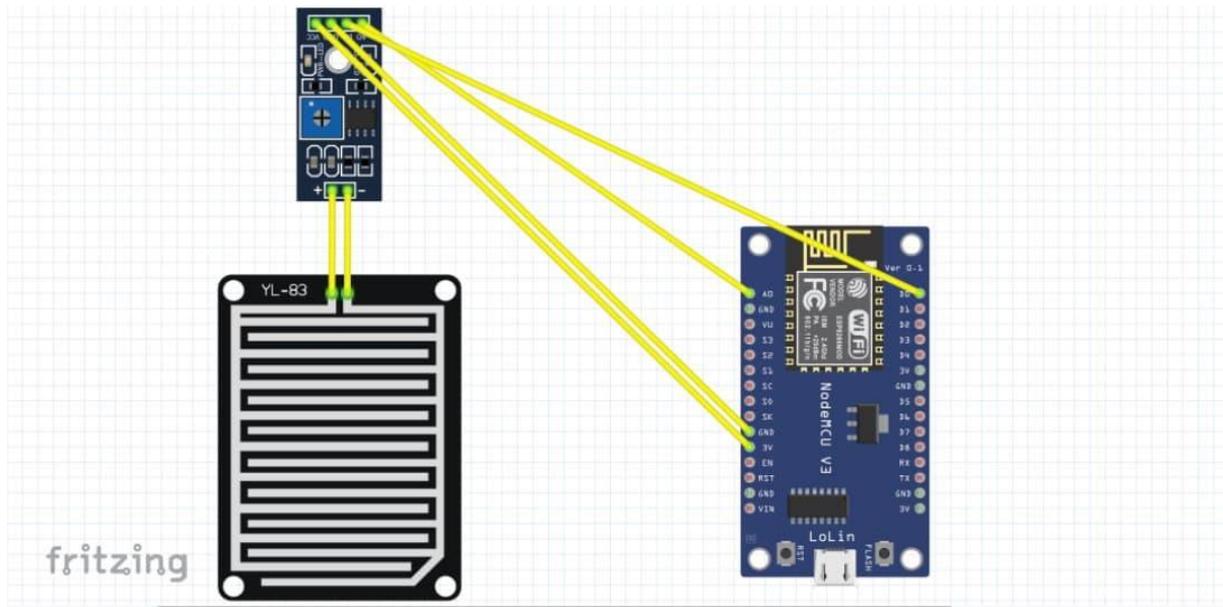


Figure XXVII : montage du capteur de la pluie avec l'esp8266

```

COM6
|
21:07:41.467 -> 1024
21:07:42.492 -> 1024
21:07:43.464 -> 1024
21:07:44.493 -> 519
21:07:45.507 -> 493
21:07:46.476 -> 552
21:07:47.474 -> 501
21:07:48.480 -> 439
21:07:49.492 -> 462
21:07:50.470 -> 465
21:07:51.500 -> 461
21:07:52.486 -> 456
21:07:53.509 -> 364
21:07:54.503 -> 373
21:07:55.508 -> 317
21:07:56.476 -> 315
21:07:57.477 -> 343
21:07:58.479 -> 357
21:07:59.501 -> 313
21:08:00.480 -> 306

```

Figure XXVIII : Resultats sur le monteur serie



Station Météorologique

Il pleut !

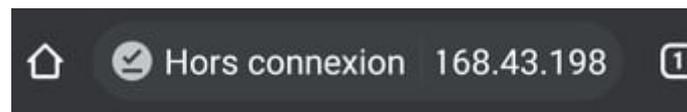


Station Météorologique

Il ne pleut pas !

Figure XXIX :Resultats sur le site web

Après les tests, nous nous sommes assurés que tous les capteurs fonctionnent correctement, avec de bons résultats, l'USB est instable. Le deuxième recours consiste à envoyer avec enthousiasme ces données à un site Web.



Station Météorologique

Témpérature : 32.60 °C
Humidite : 31.00%
Il ne pleut pas !
Niveau d'eau :3

Conclusion générale

III. Conclusion générale

Dans ce projet, nous nous sommes concentrés sur la conception et la réalisation d'une Station météo pour acquérir des données à l'aide de trois capteurs : capteur de température et humidité, capteur de la pluie et capteur niveau d'eau à l'aide d'un protocole de communication sans fil (wifi) avec un ordinateur ou un smartphone.

Nous avons créé un système de mesure en temps réel pour tous les phénomènes sont basées sur une carte Arduino UNO comme unité de contrôle, le rôle de la carte Arduino UNO traitera les données fournies par les capteurs utilisés. Au début nous avons Essayez de connecter le système de mesure au présentoir (PC) via un câble USB Assurez-vous que le capteur fonctionne correctement. Les programmes écrits sur l'IDE Arduino permettent Les résultats sont affichés sur le moniteur série.

La deuxième partie de ce travail consiste à établir une connexion sans fil (via wifi) pour Avec le module ESP8266, nous en avons créé un autre qui peut afficher Les résultats sont affichés en temps réel sur le site Internet sous forme de valeurs.

Ces réalisations ne sont pas sans difficultés. Il est important de noter que nous sommes trop face à de nombreux problèmes cependant, on peut dire que malgré ces difficultés, les résultats obtenus grâce à cette pratique ou théorie de recherche ouvrent la porte à de nouvelles recherches. Nous espérons que ce document est une référence pour ceux qui souhaitent développer et mettre en œuvre des projets et un système basé sur une carte Arduino.

Bibliographie

Bibliographie

ABDELKADER, M. (2018). *Réalisation d'une station météorologique*. ANNABA : UNIVERSITE BADJI MOKHTAR .

arduino CENTER. (2020). *AboutUs*. Récupéré sur arduino: <https://www.arduino.cc/en/Main/AboutUs>

capteurs detecteurs. (s.d.). Récupéré sur euro-makers: <https://euro-makers.com/fr/capteurs-detecteurs%20arduino/1591-capteur-de-pluie-arduino-3701172905342.html>

CENTER, A. (2020). *Arduino uno rev3*. Récupéré sur store.Arduino: <https://store.Arduino.cc/Arduino-uno-rev3>

components101. (2018, 10 17). *esp12e-pinout-datasheet*. Récupéré sur components101: <https://components101.com/wireless/esp12e-pinout-datasheet>

content.instructables. (2019). *ORIG*. Récupéré sur content.instructables: <https://content.instructables.com/ORIG/F2A/M2U1/JH8JBTCZ/F2AM2U1JH8JBTCZ.png>

futura. (2020, 02 09). *internet css*. Récupéré sur futura sciences: <https://www.futura-sciences.com/tech/definitions/internet-css-4050/>

GIRAUD, P. (2019, 07 13). *pierre-giraud*. Récupéré sur pierre-giraud: <https://www.pierre-giraud.com/liste-selecteurs-css-utilisation/>

gotronic store center. (2018). *Module capteur de niveau d'eau ST045*. Récupéré sur gotronic: <https://www.gotronic.fr/art-module-capteur-de-niveau-d-eau-st045-26116.htm>

Guidelines, D. (2007, 03 22). *osd*. Récupéré sur opensource: <https://opensource.org/osd>

Hughes, J. M. (2007). *Arduino-a-technical*. Récupéré sur oreilly: <https://www.oreilly.com/library/view/Arduino-a-technical/9781491934319/ch01.html>

instructables, c. (2019). *ORIG*. Récupéré sur content.instructables: <https://content.instructables.com/ORIG/FH4/FSNU/JH8JBT9I/FH4FSNUJH8JBT9I.png>

instructables, c. (2019). *ORIG*. Récupéré sur content.instructables: <https://content.instructables.com/ORIG/F4Z/OL5G/JH8JBTGA/F4ZOL5GJH8JBTGA.png>

KARA LYDIA, G. S. (2018). *Conception et réalisation d'un système de mesure et transmission de paramètres météorologiques*. TIZI-OUZOU : UNIVERSITE MOULOUD MAMMERI .

MAHFOUD KHA. (2019, 08 07). *meteo dz*. Récupéré sur android-dz: <https://www.android-dz.com/ar/meteo-dz/>

Meteorology. (2018). Récupéré sur for9a: <https://www.for9a.com/specialities/الجوية-الأرصاد-علم-Meteorology>

Nebra, M. (2013). *Apprenez à créer votre site web avec HTML5 et CSS3*. Paris: openclassrooms.

NEBRA, M. (2013). *realisez votre site web avec html5 et css3*. PARIS .

Philippe Chaléat, D. C. (2002). *HTML, JavaScript*. Paris: ÉDITIONS EYROLLES.

skywodd. (2017, 2 21). *Utiliser un capteur de température et d'humidité DHT11 / DHT22 avec une carte Arduino / Genuino*. Récupéré sur carnet du maker:

<https://www.carnetdumaker.net/articles/utiliser-un-capteur-de-temperature-et-dhumidite-dht11-dht22-avec-une-carte-arduino-genuino/>

weather stations. (2018, 06 13). Récupéré sur weathershack:

<http://www.weathershack.com/category/weather-stations.html>

Wikipédia. (2021, 5 16). *Hypertext Markup Language*. Récupéré sur wikipedia:

https://fr.wikipedia.org/wiki/Hypertext_Markup_Language

Annexe

IV. Annexe(s)

Le fichier qui calcule l'exécution du code final :

```
#include <ESP8266WiFi.h>
#include "DHT.h"

#include "index.h" //webpage HTML index.h

//1_Pour : DHT11
#define DHTPIN 0 // Capteur de DHT11 connecter a port(pin) D3 dans NODEMCU (port digitale : pour les
donnes digitale)
#define DHTTYPE DHT11 // type de DHT est : DHT 11
DHT dht(DHTPIN, DHTTYPE);

//3_Pour : Module de pluie
#define RAINPIN 4 // Module de pluie connecter a port(pin) D2 dans NODEMCU (port digitale : pour les
donnes digitale)

//4_Pour : Capteur de niveau d'eau
#define NIVEAUDEAUPIN A0 //Capteur Niveau d'eau connecter a port(pin) A0 dans NODEMCU (port
analogique : pour les donnes analogique)

//SSID(nom) et le mot de passe de 'point d'accès' ou 'WiFi' ATTENTION changer le ssid et password avant la
compilation et l'exution!
const char* ssid = "PFE";
const char* password = "01234567";
WiFiServer server(80);

void setup(){
  Serial.begin(115200);
  Serial.println();

  Serial.printf("[CONNEXION A] :: %s ", ssid);
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED){
    delay(500);
    Serial.print(".");
  }
  Serial.println("[CONNECTE]");

  server.begin();
  Serial.printf("[WEB_SERVER DEMAREE] :: ouvrir @IP %s dans votre navigateur Web\n",
WiFi.localIP().toString().c_str());

  dht.begin();
  pinMode(RAINPIN, INPUT);
  pinMode(NIVEAUDEAUPIN, INPUT);
}

String preparerHtmlPage(){
  String htmlPage = "";
  htmlPage.reserve(1024); // prevent ram fragmentation
  String s1 = HEAD_page; //Read HTML contents

  htmlPage.concat(s1);
```

```

htmlPage.concat("\n    <br> Température :\t");
htmlPage.concat(dht.readTemperature());
htmlPage.concat("\n    &deg;C<br> Humidite :\t");
htmlPage.concat(dht.readHumidity());

int valeur_digital_rain = digitalRead(RAINPIN);
if (valeur_digital_rain == 0)
    htmlPage.concat("%\n    <br> Il pleut !");
else
    htmlPage.concat("%\n    <br> Il ne pleut pas !");

htmlPage.concat("\n    <br>Niveau d'eau :");
htmlPage.concat(analogRead(NIVEAUDEAUPIN));

String s2 = FOOT_page; //Read HTML contents
htmlPage.concat(s2);

return htmlPage;
}

void loop(){
    WiFiClient client = server.available();
    // wait for a client (web browser) to connect
    if (client){
        Serial.println("\n[CLIENT CONNECTE]");
        while (client.connected()){
            // read line by line what the client (web browser) is requesting
            if (client.available()){
                String line = client.readStringUntil('\r');
                Serial.print(line); //Les information des client connecter telque (OS de machine et ca version, Navigateur et
ca version)
                if (line.length() == 1 && line[0] == '\n'){
                    client.println(preparerHtmlPage());
                    break;
                }
            }
        }
    }

    while (client.available()){
        client.read();
    }

    // close the connection:
    client.stop();
    Serial.println("\n[CLIENT DECONNECTE]");
}
}

```

le fichier qui compte exécution html :

```
const char HEAD_page[] PROGMEM = R"=====(
<!DOCTYPE html>
<html>
<!--HTTP/1.1 200 OK-->
<!--Connection: close-->
<head>
<meta charset="UTF-8">
<meta http-equiv="Content-Language" content="fr">
<meta http-equiv="Content-Type" content="text/html">
<meta http-equiv="refresh" content="5" />

<meta name="description" content="Pojet fin d'etude Master">
<meta name="author" content="BECHERIF">
<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Projet MASTER 2021-M2</title>
</head>
<body>
<center><H2>Station Météorologique</H2></center>
)=====";

const char FOOT_page[] PROGMEM = R"=====(
</body>
</html>
)=====";
```

Résumé

Résumé :

L'objectif de notre projet est la réalisation d'un prototype de station météorologique sous la plate-forme NodeMCUE8266 qu'elle est capable de mesurer le facteur météorologique (atmosphérique) dans son environnement. Ce travail s'est déroulé comme suit premièrement nous commençons par le choix du capteurs disponible dans la marche et chaque'un par leur datasheet, puis en fait le câblage, dans la plaque d'essai dans un deuxième temps, nous programmons la carte NodeMCU 8266 pour qu'elle soit capable de traiter les informations des capteurs puis d'afficher ces informations au(x) utilisateur(s) dans leurs navigateurs web.

Mots-Clés :

Arduino, ArduinoIDE, NodeMCU, station météorologique, DHT11, ST045, MH-RD, HTML.

ملخص :

الهدف من مشروعنا هو تحقيق نموذج أولي لمحطة الأرصاد الجوية تحت منصة NodeMCUE8266 بحيث تكون قادرة على قياس عامل الأرصاد الجوية (الغلاف الجوي) في بيئتها. استمر هذا العمل على النحو التالي: أولاً نبدأ باختيار المستشعرات المتوفرة في السوق وكل واحدة من خلال ورقة البيانات الخاصة بها، ثم في الواقع الأسلاك، في لوحة الاختبار ثانياً، نقوم ببرمجة بطاقة NodeMCU 8266 بحيث تكون قادرة على معالجة معلومات أداة الاستشعار ثم عرض هذه المعلومات على المستخدم (المستخدمين) في متصفحات الويب الخاصة بهم

الكلمات المفتاحية:

اردوينو، اردوينو (اي دي او)، محطة جوية، مستشعر الحرارة و الرطوبة، مستشعر مستوى الماء، مستشعر كشف المطر، لغة ترميز النصوص التشعبية.

Abstract :

The objective of our project is the realization of a meteorological station prototype under the NodeMCUE8266 platform that it is able to measure the meteorological factor (atmospheric) in its environment. This work proceeded as follows: first we start with the choice of the sensors available in the market and each one by their datasheet, then in fact the wiring, in the test plate secondly, we program the NodeMCU card 8266 so that it is able to process the information from the sensors and then display this information to the user (s) in their web browsers.

Keywords :

Arduino, ArduinoIDE, NodeMCU, weather station, DHT11, ST045, MH-RD, HTML.