

IOT & Systèmes embarqués 2025	INFRES TP IOT & Systèmes embarqués 2025 Développement software embarqué Module Actionneur TP Partie 1	TP - IMT 2025 IoT Part. 1.docx
Créé le 05/04/202521/05/2025 00:49:00		Rev. 0
Créé par Lazarewicz Julien		Page 1 sur 26
		 IMT Mines Alès École Mines-Télécom

TP IOT & Systèmes embarqués.

Maison connectées

Partie 1

1.	PRESENTATION DE L'ENVIRONNEMENT DE DEVELOPPEMENT	2
2.	PRESENTATION DU FRAMEWORK ARDUINO	2
2.1.	STRUCTURE D'UN PROGRAMME ARDUINO	3
2.1.1.	Au début du fichier, directives préprocesseur et déclarations globales :	3
2.1.2.	La fonction setup()	3
2.1.3.	La fonction loop()	3
3.	MATERIEL UTILISE	4
3.1.	MODULE CAPTEUR	4
3.2.	CAPTEUR DE TEMPERATURE / HUMIDITE	4
3.3.	CAPTEUR DE LUMINOSITE	5
3.4.	CAPTEUR DE QUALITE DE L'AIR	6
3.5.	MODULE ACTIONNEUR	6
3.6.	SERVEUR DE COLLECTE DES DONNEES	7
3.7.	SCHEMA DE FONCTIONNEMENT	8
4.	INSTALLATION DES LOGICIELS REQUIS	9
4.1.	VISUAL STUDIO CODE	9
4.1.1.	Installation pour Windows	9
4.1.2.	Installation pour MacOS	9
4.1.3.	Installation pour Linux	9
4.2.	GIT	9
4.2.1.	Installation pour Windows	9
4.2.2.	Installation pour MacOS	10
4.2.3.	Installation pour Linux	10
4.3.	INSTALLATION DE CLANG-FORMAT	11
4.3.1.	Installation pour Windows	11
4.3.2.	Installation pour MacOS	11
4.3.3.	Installation pour Linux	12
4.4.	INSTALLATION DE PLATFORMIO	13
5.	MONTAGE DES ELEMENTS SUR LA PLATINE D'ESSAI	15
6.	DEVELOPPEMENT SUR FIRMWARE POUR LE « MODULE ACTIONNEUR	17
6.1.	RESSOURCES A VOTRE DISPOSITION	17
6.1.1.	Algorithme	18
6.1.2.	Récupération du template	21
6.1.3.	Ouvrir le template dans VS Code :	23
6.1.4.	Contenu du projet template :	23
7.	CONNECTION A L'INTERFACE WEB DU SERVEUR THINGSBOARD	24
7.1.	ACCEDER A L'INTERFACE WEB DU SERVEUR THINGSBOARD	24
7.2.	VOTRE COMPTE THINGSBOARD	25

IOT & Systèmes embarqués 2025	INFRES TP IOT & Systèmes embarqués 2025 Développement software embarqué Module Actionneur TP Partie 1	TP - IMT 2025 IoT Part. 1.docx
Créé le 05/04/2025 21/05/2025 00:49:00		Rev. 0
Créé par Lazarewicz Julien		Page 2 sur 26
		 IMT Mines Alès École Mines-Télécom

1. Présentation de l'environnement de développement

Pour ce TP nous allons utiliser un microcontrôleur ESP WROOM. Il existe au moins 3 frameworks pouvant être utilisés pour programmer ce type de microcontrôleur :

- SDK natif Espressif
- Framework Arduino
- Micro-Python

Pour ce TP nous utiliserons le framework Arduino avec PlatformIO et VS Code. Nous aurions pu utiliser l'Arduino IDE mais la solution choisie est plus proche des solutions utilisées en entreprise.

Pourquoi pas Python ?

Python est un langage en vogue et depuis peu, il peut être utilisé pour le développement embarqué. C'est un bon choix pour du prototypage rapide mais comporte quelques inconvénients :

- Peu optimisé
- Très coûteux en ressources matérielles (CPU, mémoire, stockage)
- Pas utilisé dans l'industrie

2. Présentation du framework Arduino

Le framework Arduino est un ensemble d'outils logiciels, de bibliothèques et de conventions permettant de programmer des microcontrôleurs de manière simple et accessible, même sans formation approfondie en électronique ou en développement embarqué.

Il repose sur :

- Un langage simplifié dérivé du C/C++.
- Une API unifiée pour manipuler facilement les entrées/sorties, les ports série, les capteurs, moteurs, etc.
- A l'origine développé pour les microcontrôleurs AVR 8 bits, il supporte maintenant une très grande variété de plateformes matérielles :
 - AVR (ATmega328, ATmega2560, etc.)
 - ARM Cortex-M (STM32, SAMD21, NXP, etc.)
 - ESP32 / ESP8266
 - RISC-V

IOT & Systèmes embarqués 2025	INFRES TP IOT & Systèmes embarqués 2025 Développement software embarqué Module Actionneur TP Partie 1	TP - IMT 2025 IoT Part. 1.docx
Créé le 05/04/2025 21/05/2025 00:49:00		Rev. 0
Créé par Lazarewicz Julien		Page 3 sur 26
		 IMT Mines Alès École Mines-Télécom

2.1. Structure d'un programme Arduino

Une programme Arduino se compose de 3 éléments essentiels :

2.1.1. Au début du fichier, directives préprocesseur et déclarations globales :

- Les directives pour l'inclusion de bibliothèques « #include »
- Les directives pour la déclaration de constantes ou de macro « #define »
- Les déclarations de variables globales et objets visible dans tous le programme.
- Les déclarations ou définitions de fonctions utilisées dans le programme.

Exemples :

```
1. #include <Arduino.h>
2. #define LED_PIN 13
3. WiFiClient espClient;
```

2.1.2. La fonction setup()

La fonction setup() est exécutée une seule fois au démarrage du programme. Elle est utilisée pour :

- Configurer les broches (entrée/sortie)
- Initialiser la communication série, wifi, Ethernet
- Démarrer des capteurs, modules, etc.

Exemple :

```
1. void setup()
2. {
3.     pinMode(13, OUTPUT); // Configure la broche 13 en sortie
4.     Serial.begin(9600); // Démarre la communication série
5. }
```

2.1.3. La fonction loop()

La fonction loop() est exécutée en boucle indéfiniment, tant que la carte est sous tension. C'est ici qu'on place la logique principale du programme :

- Lecture de capteurs
- Commande d'actionneurs
- Calculs, affichage, communication, etc.

Exemple :

```
1. void loop()
2. {
3.     digitalWrite(13, HIGH); // Allume la LED
4.     delay(500);             // Pause 500 ms
5.     digitalWrite(13, LOW);  // Éteint la LED
6.     delay(500);             // Pause 500 ms
7. }
```

IOT & Systèmes embarqués 2025	INFRES TP IOT & Systèmes embarqués 2025 Développement software embarqué Module Actionneur TP Partie 1	TP - IMT 2025 IoT Part. 1.docx
Créé le 05/04/2025 21/05/2025 00:49:00		Rev. 0
Créé par Lazarewicz Julien		Page 4 sur 26
		 IMT Mines Alès École Mines-Télécom

3. Matériel utilisé

Dans ce TP vous serez emmené à utiliser le matériel suivant :

3.1. Module capteur

Le module capteur interface les différents capteurs que l'on peut trouver dans une maison « connectée ». Il communique avec le serveur de collecte des données via Wifi et transmet les valeurs des capteurs connectés via le protocole MQTT.

Il est connecté aux capteurs via un bus I2C.

Il est alimenté via une batterie LiPO qui peut être rechargée en le branchant en USB.

Pour économiser la batterie, il collecte la valeur des capteurs, se connecte au serveur de collecte, transmet les valeurs des capteurs via MQTT puis entre en mode « deep sleep ».

Via une interruption hardware, il se réveille toutes les X secondes pour recommencer l'opération.

Caractéristiques :

- Carte de développement Adafruit ESP32 Feather V2
- MCU ESP32 WROOM
- 8 MB de mémoire flash
- 2 MS de mémoire PSRAM

3.2. Capteur de température / humidité

Ce module est basé sur un capteur ASAIR AHT20. Il mesure la température ambiante et le taux d'humidité présent dans l'air ambiant.

Il communique avec le module capteur via un bus I2C.

Caractéristiques :

- Module Adafruit AHT20
- Communication I2C
- Précision humidité +- 2%
- Précision température +- 3°C entre 20% et 80% d'humidité

3.3. Capteur de luminosité

Ce module est basé sur un capteur ROHM BH1750. Il mesure la luminosité ambiante en Lux.

Il communique avec le module capteur via un bus I2C.

Caractéristiques :

- Module Adafruit BH1750
- Communication I2C
- Plage de mesures de 1 à 65K Lux

Exemple :

Lumière directe du soleil	32 000 – 100 000 Lux
Board de mer, plage (été midi)	70 000 – 90 000 Lux
Journée ensoleillée (été midi)	10 000 – 25 000 Lux
Ciel nuageux (midi)	1000 – 2000 Lux
Luminosité bureau (conseillée)	320 – 500 Lux
Ciel très nuageux (midi)	100 - 250 Lux
Salon familial, lumière tamisé	80 Lux
Nuit, environnement urbain, ciel dégagé, pas d'éclairage	3.4 Lux
Pleine lune, ciel dégagé, extra urbain	0.05–0.3 Lux
Nuit sans lune, ciel nuageux	0.0001 Lux

3.4. Capteur de qualité de l'air

Ce module est basé sur un capteur SENSIRION SGP40. Il est capable de mesurer une grande variété de VOCs « Volatile Organic Compounds » ainsi que le H₂. Il retourne un indice de qualité de l'air de 0 à 500 en VOC. Il communique avec le module capteur via un bus I2C.

Exemple :

<u>Index VOC</u>	<u>Qualité de l'air</u>	<u>Conséquence sur la santé (à long terme)</u>
0-50	Excellent	Air pur, bon pour la santé
51-100	Bon	Sans danger
101-150	Légèrement pollué	Convenable, sans danger
151-200	Modérément pollué	Irritation possible des voies respiratoires
201-250	Pollué	Possible maux de tête, irritation des voies respiratoires
251-350	Très pollué	Danger pour la santé
351-500	Extrêmement pollué	Possible effets neurotoxiques

3.5. Module actionneur

Le module actionneur pilote les éléments suivants que l'on peut trouver dans une maison « connectée » :

- VMC (ventilation)
- Chauffage
- Climatisation
- Éclairage

Pour simplifier ce TP, les actionneurs seront remplacés par LEDs indiquant le statu des actionneurs, LED allumée actionneur activé, LED éteinte actionneur désactivé.

IOT & Systèmes embarqués 2025	INFRES TP IOT & Systèmes embarqués 2025 Développement software embarqué Module Actionneur TP Partie 1	TP - IMT 2025 IoT Part. 1.docx
Créé le 05/04/2025 21/05/2025 00:49:00		Rev. 0
Créé par Lazarewicz Julien		Page 7 sur 26
		 IMT Mines Alès École Mines-Télécom

Il communique avec le serveur de collecte des données via Wifi et reçoit des commandes RPC via le protocole MQTT.

Il est alimenté via un câble USB.

Caractéristiques :

- Carte de développement Adafruit ESP32 Feather V2
- MCU ESP32 WROOM
- 8 MB de mémoire flash
- 2 MS de mémoire PSRAM

3.6. Serveur de collecte des données

Le serveur de collecte des données est l'élément central de la solution IoT pour maison « connectée ».

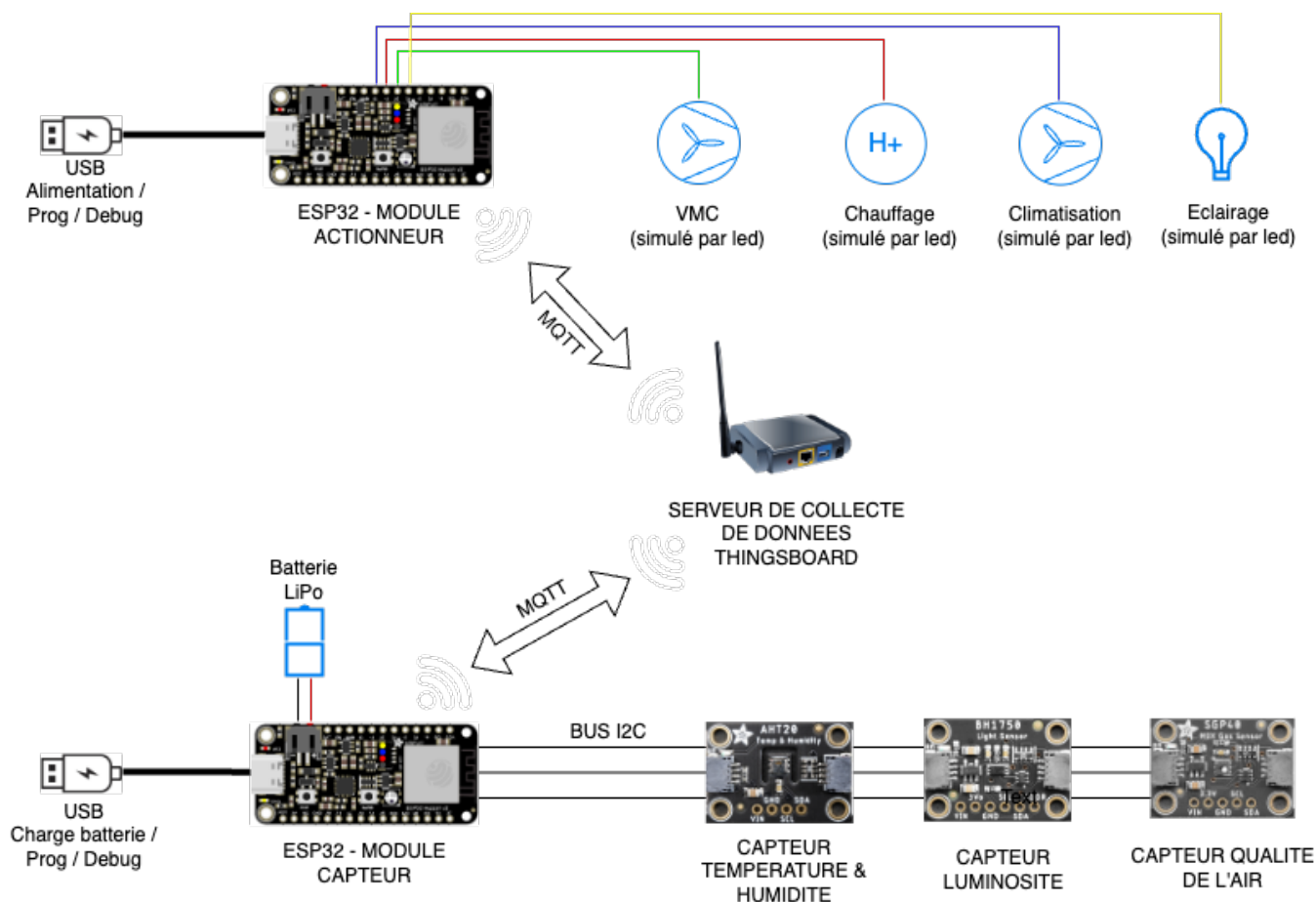
Son rôle est de :

- Recueillir et stocker les informations provenant des capteurs
- Commander à distance les actionneurs
- Fournir une interface graphique permettant de visualiser les valeurs des capteurs
- Fournir une interface graphique pour contrôler et visualiser l'état des actionneurs
- Définir des règles d'automatisation
- Fournir un point d'accès Wifi.

Pour réaliser ces tâches les logiciels open source suivants sont utilisés :

- OS Linux
- SGBDR PostgreSQL (stockage)
- Thingsboard (gestion des périphériques, collecte, traitement et affichage des données)

3.7. Schéma de fonctionnement



IOT & Systèmes embarqués 2025	INFRES TP IOT & Systèmes embarqués 2025 Développement software embarqué Module Actionneur TP Partie 1	TP - IMT 2025 IoT Part. 1.docx
Créé le 05/04/2025 21/05/2025 00:49:00		Rev. 0
Créé par Lazarewicz Julien		Page 9 sur 26
		 IMT Mines Alès École Mines-Télécom

4. Installation des logiciels requis

4.1. Visual Studio Code

Visual Studio Code est l'environnement de développement intégré qui sera utilisé pour ce TP. Il est gratuit et hautement personnalisable, pour plus d'information voir : <https://code.visualstudio.com/>

4.1.1. Installation pour Windows

- Télécharger l'installateur à cette URL :
<https://code.visualstudio.com/sha/download?build=stable&os=win32-x64-user>
- Exécuter l'installateur en laissant les paramètres par défaut.

4.1.2. Installation pour MacOS

- Télécharger l'installateur à cette URL :
<https://code.visualstudio.com/sha/download?build=stable&os=darwin-universal>
- Extraire le fichier : « Visual Studio Code.app » contenu dans le fichier *.zip téléchargé dans le dossier « Applications »

4.1.3. Installation pour Linux

L'installation de VS Code étant différente suivant la distribution Linux utilisée, je vous conseille de suivre le tutoriel adapté à votre distribution.

Rendez-vous à l'URL suivante et suivre les instructions concernant votre distribution :

https://code.visualstudio.com/docs/setup/linux#_install-vs-code-on-linux

4.2. Git

Git est un système de contrôle de version qui effectue intelligemment le suivi des modifications apportées aux fichiers. Git est particulièrement utile lorsque vous et un groupe de personnes apportez des modifications aux mêmes fichiers en même temps.

Il vous sera nécessaire pour télécharger le squelette du projet pour ce TP.

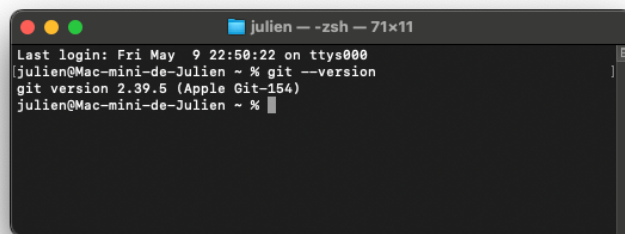
4.2.1. Installation pour Windows

- Télécharger l'installateur à cette URL : <https://github.com/git-for-windows/git/releases/download/v2.49.0.windows.1/Git-2.49.0-64-bit.exe>
- Exécuter l'installateur en laissant les paramètres par défaut.

4.2.2. Installation pour MacOS

Git est présent par défaut sur les version moderne de MacOS si vous avez installé Xcode. Pour vérifier la présence de Git sur votre système :

- Aller dans « Applications » puis dans « Utilitaires » et lancer « Terminal »
- Dans le terminal nouvellement ouvert taper la commande « git --version »
- Si Git est bien installé vous devriez avoir un retour comme sur la capture d'écran suivante.



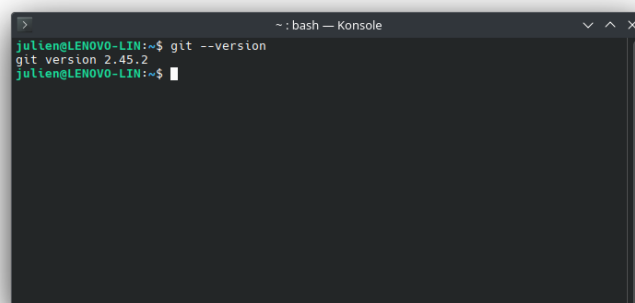
```
julien ~ -zsh - 71x11
Last login: Fri May  9 22:50:22 on ttys000
juliaen@Mac-mini-de-Julien ~ % git --version
git version 2.39.5 (Apple Git-154)
juliaen@Mac-mini-de-Julien ~ %
```

- Sinon dans le terminal taper la commande « xcode-select --install »

4.2.3. Installation pour Linux

Git est présent par défaut sur les distributions récentes Linux. Pour vérifier la présence de Git sur votre système :

- Ouvrir un terminal (dépend de votre distribution)
- Dans le terminal nouvellement ouvert taper la commande « git --version »
- Si Git est bien installé vous devriez avoir un retour comme sur la capture d'écran suivante (exemple Ubuntu / KDE plasma 6).



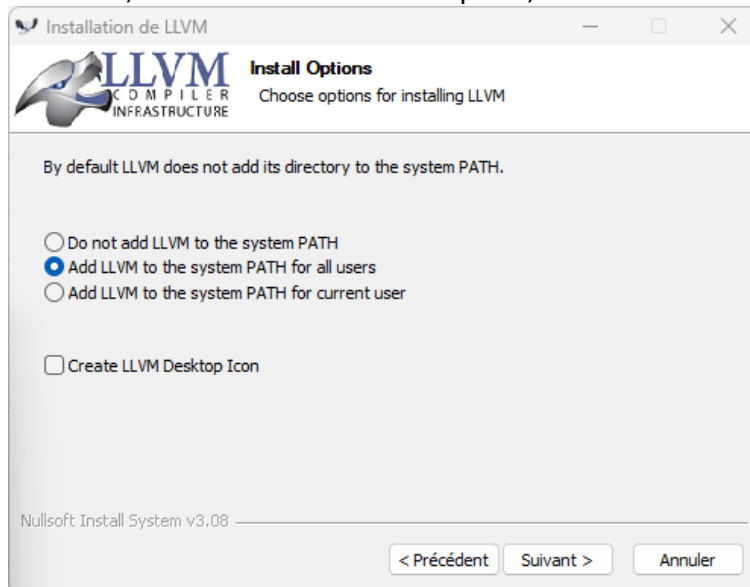
```
~: bash - Konsole
juliaen@LENOVO-LIN:~$ git --version
git version 2.45.2
juliaen@LENOVO-LIN:~$
```

4.3. Installation de clang-format

Clang-format permet de formater automatiquement votre code C/C++.

4.3.1. Installation pour Windows

- Télécharger l'installateur à cette URL : <https://github.com/llvm/llvm-project/releases/download/llvmorg-20.1.5/LLVM-20.1.5-win64.exe>
- Exécuter l'installateur
- Si Windows vous alerte disant que le programme peut être dangereux choisissez « exécuter quand même ». Initialisation.
- Lorsque l'installateur vous demande si vous voulez ajouter le programme à votre « PATH », choisissez la deuxième option, comme sur le Screenshot suivant.



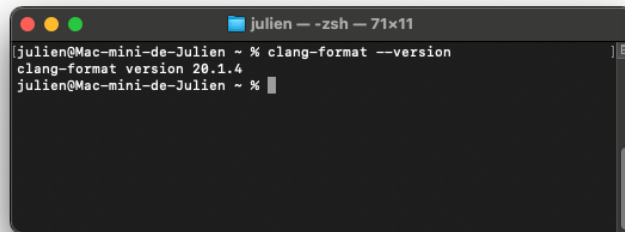
- Pour le reste choisissez les options par défaut.

4.3.2. Installation pour MacOS

Clang-format est présent par défaut sur les version moderne de MacOS si vous avez installé Xcode. Pour vérifier la présence de Git sur votre système :

- Aller dans « Applications » puis dans « Utilitaires » et lancer « Terminal »
- Dans le terminal nouvellement ouvert taper la commande « clang-format --version »
- Si clang-format est bien installé vous devriez avoir un retour comme sur la capture d'écran suivante.

IOT & Systèmes embarqués 2025	INFRES TP IOT & Systèmes embarqués 2025 Développement software embarqué Module Actionneur TP Partie 1	TP - IMT 2025 IoT Part. 1.docx
Créé le 05/04/2025 21/05/2025 00:49:00		Rev. 0
Créé par Lazarewicz Julien		Page 12 sur 26
		 IMT Mines Alès École Mines-Télécom



```

julien@Mac-mini-de-Julien ~ % clang-format --version
clang-format version 20.1.4
julien@Mac-mini-de-Julien ~ %

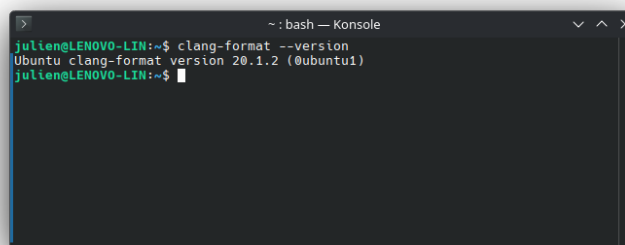
```

- Sinon dans le terminal taper la commande « `xcode-select --install` »

4.3.3. Installation pour Linux

Clang-format est présent par défaut sur les distributions récentes Linux. Pour vérifier la présence de clang-format sur votre système :

- Ouvrir un terminal (dépend de votre distribution)
- Dans le terminal nouvellement ouvert taper la commande « `git --version` »
- Si clang-format est bien installé vous devriez avoir un retour comme sur la capture d'écran suivante (exemple Ubuntu / KDE plasma 6).



```

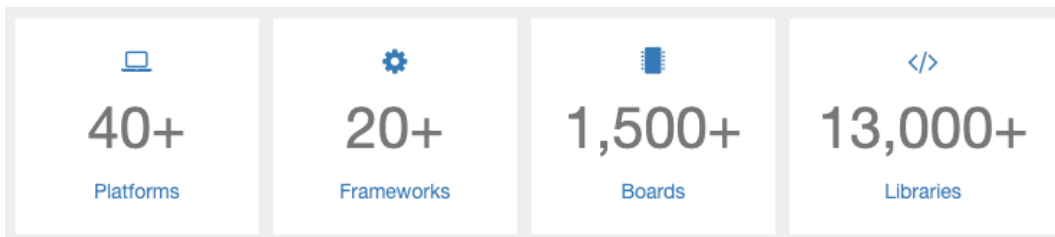
~: bash — Konsole
julieng@LENOVO-LIN:~$ clang-format --version
Ubuntu clang-format version 20.1.2 (0ubuntu1)
julieng@LENOVO-LIN:~$

```

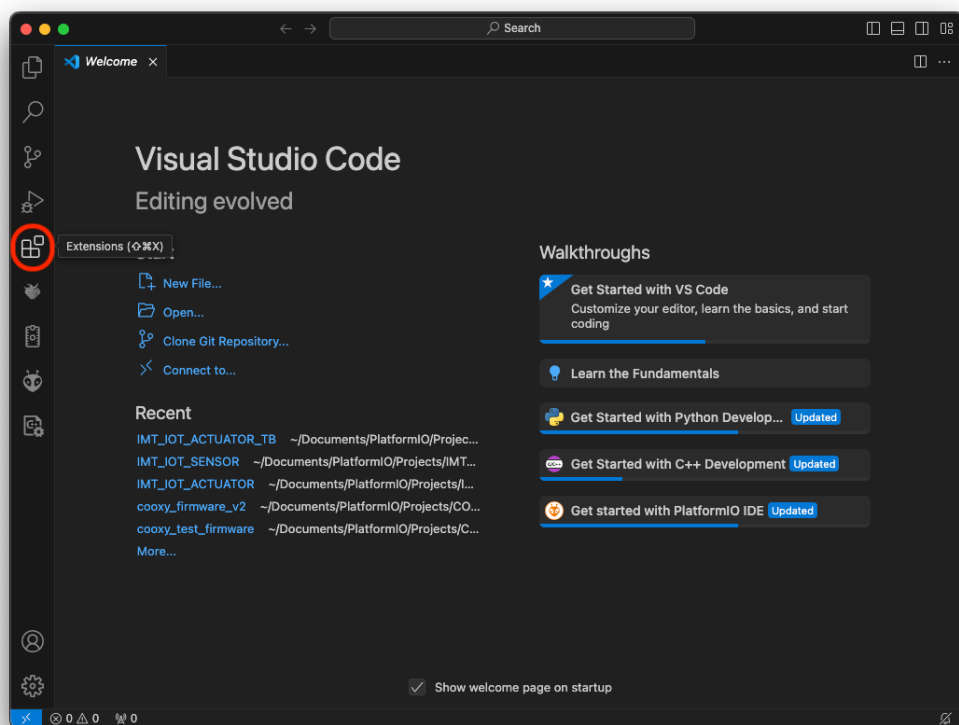
4.4. Installation de PlatformIO

PlatformIO est un écosystème open source dédié au développement embarqué. Il fournit un environnement de développement C/C++ pour les systèmes embarqués supportés. Il est multi-plateformes (Windows, Mac et GNU/Linux). Il facilite l'installation des différents composants nécessaires au développement embarqué (compilateur, linker, debugger, SDK, ...)

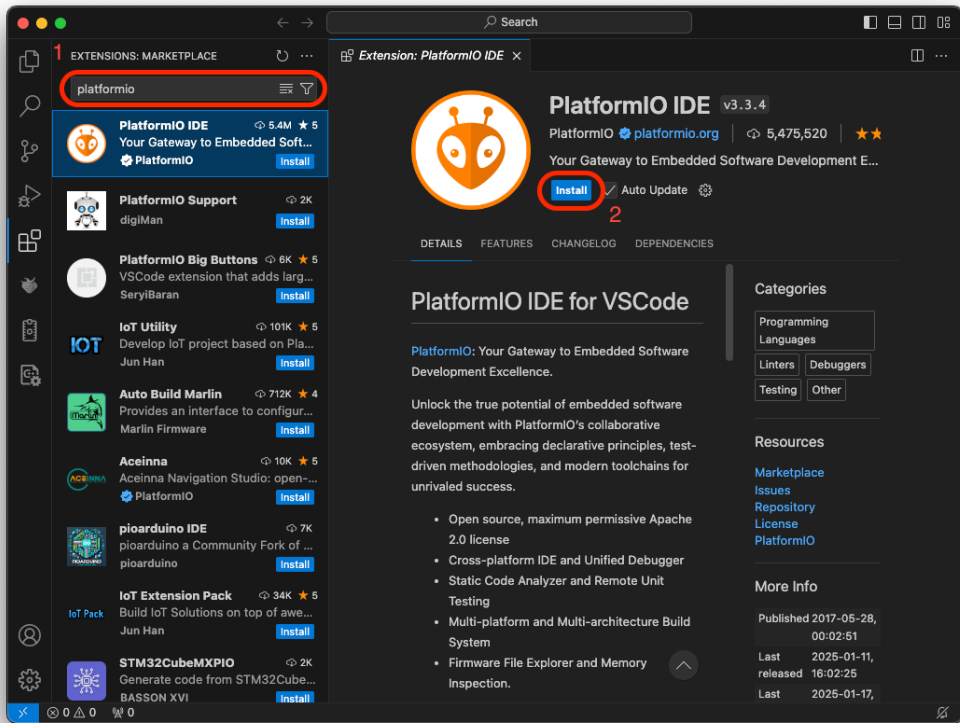
Il supporte plus de 40 plateformes, plus de 20 Framework, supporte plus de 1500 cartes de développement et intègre plus de 15 000 librairies.



- Lancer VS Code
- Cliquez sur l'icône « extensions »

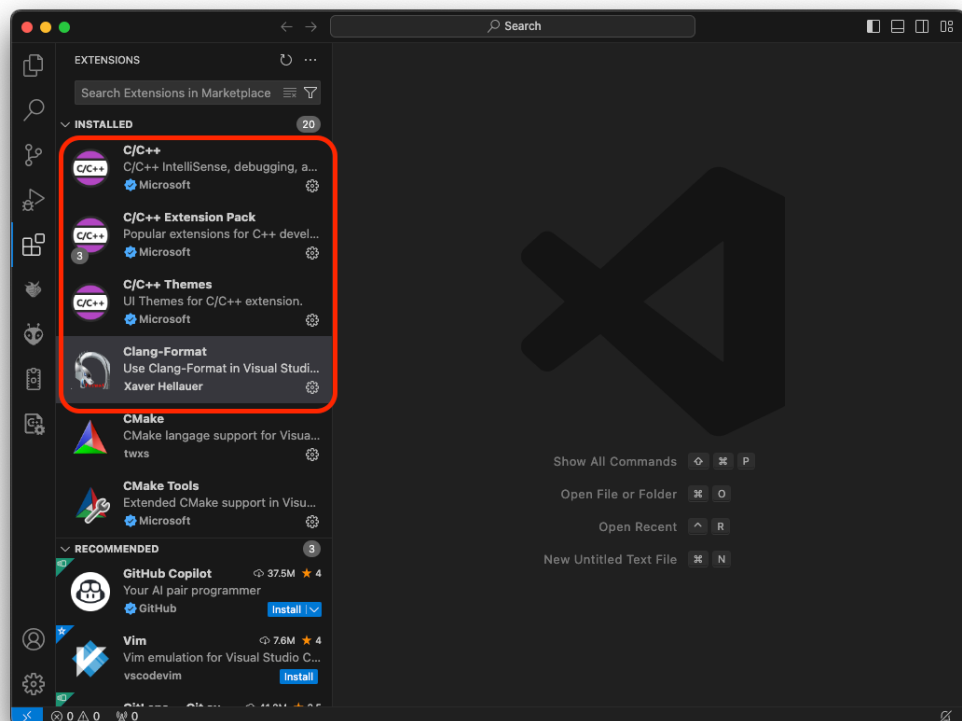


- Dans le champ de recherche tapez « PlatformIO » (1), sélectionnez dans la liste des résultat « PlatformIO IDE » puis cliquez sur le bouton « install » (2).



- Faire la même chose pour les extensions suivantes :

- C/C++
- C/C++ Themes
- C/C++ Extension Pack
- Clang-Format



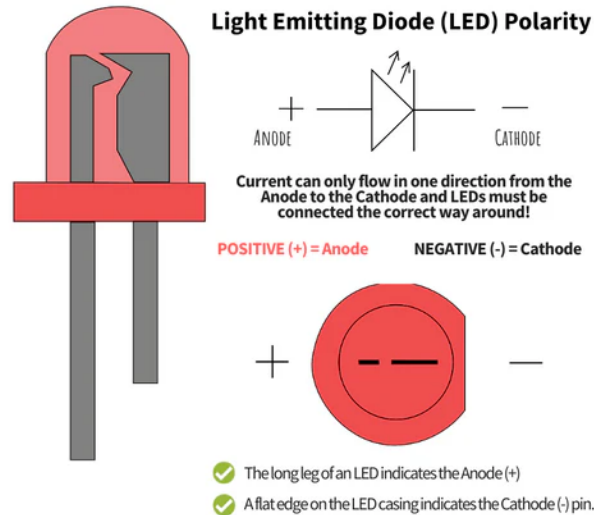
5. Montage des éléments sur la platine d'essai

- Monter les deux cartes de développement ESP32
- Monter les résistances, elles doivent avoir une valeur de 68 ou 120Ω
- Monter les 4 leds, attention les leds sont polarisés ! L'anode, borne positive doit être connectée à la résistance, la cathode, borne négative doit être connectée à la masse (ligne bleue sur la platine d'essai). Pour reconnaître l'anode et la cathode d'une led voir l'illustration ci-dessous.

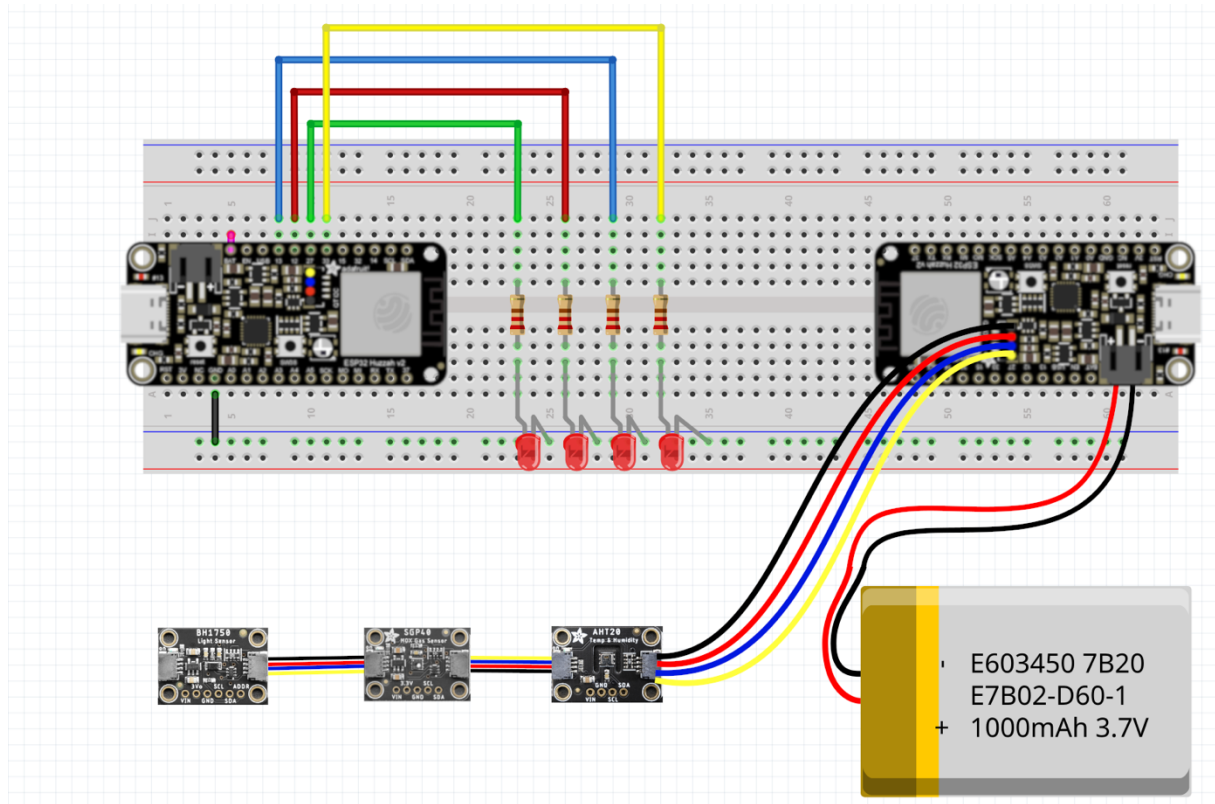


INFRES

TP IOT & Systèmes embarqués 2025 Développement software embarqué Module Actionneur TP Partie 1



- Connecter ensuite les modules de capteur à la carte « MODULE CAPTEUR » avec les câbles i2c.
- Pour finir connecter la batterie à la carte « MODULE CAPTEUR ».
ATTENTION ! BIEN VÉRIFIER LA POLARITÉ, LES CONNECTEURS JST DES BATTERIES NE SONT PAS NORMALISÉS !



IOT & Systèmes embarqués 2025	INFRES TP IOT & Systèmes embarqués 2025 Développement software embarqué Module Actionneur TP Partie 1	TP - IMT 2025 IoT Part. 1.docx
Créé le 05/04/2025 21/05/2025 00:49:00		Rev. 0
Créé par Lazarewicz Julien		Page 17 sur 26
		 IMT Mines Alès École Mines-Télécom

6. Développement sur firmware pour le « MODULE ACTIONNEUR »

Dans une première phase la travail demandé dans ce TP est de développer le firmware pour le « MODULE ACTIONNEUR ».

Le principe de fonctionnement du « MODULE ACTIONNEUR » est le suivant :

- Initialisation des actionneurs (VMC, chauffage, éclairage et climatisation)
- Connection au point d'accès Wifi
- Connection (non sécurisé) au serveur Thingsboard
- Publication de la valeur, via MQTT, des attributs (statu) des actionneurs
- Publication de la valeur, via MQTT, de l'attribut « VERSION » (version du firmware)
- Abonnement aux appels RPC (Remote Procedure Call) du serveur Thingsboard :
 - Demande du statu de l'actionneur VMC
 - Demande du statu de l'actionneur chauffage
 - Demande du statu de l'actionneur éclairage
 - Demande du statu de l'actionneur climatisation
 - Modification du statu de l'actionneur VMC
 - Modification du statu de l'actionneur chauffage
 - Modification du statu de l'actionneur éclairage
 - Modification du statu de l'actionneur climatisation
- Développement des méthodes et fonctions pour répondre aux appels RPC du serveur Thingsboard
- Développement des fonctions pour modifier le statu des actionneurs
- Vérifier le bon fonctionnement de votre firmware depuis votre session Thingsboard.

6.1. Ressources à votre disposition

Afin de vous aider, vous aurais à votre disposition :

- Un template de projet PlatformIO
- Un compte préconfiguré sur le serveur Thingsboard
- Un algorithme partiel

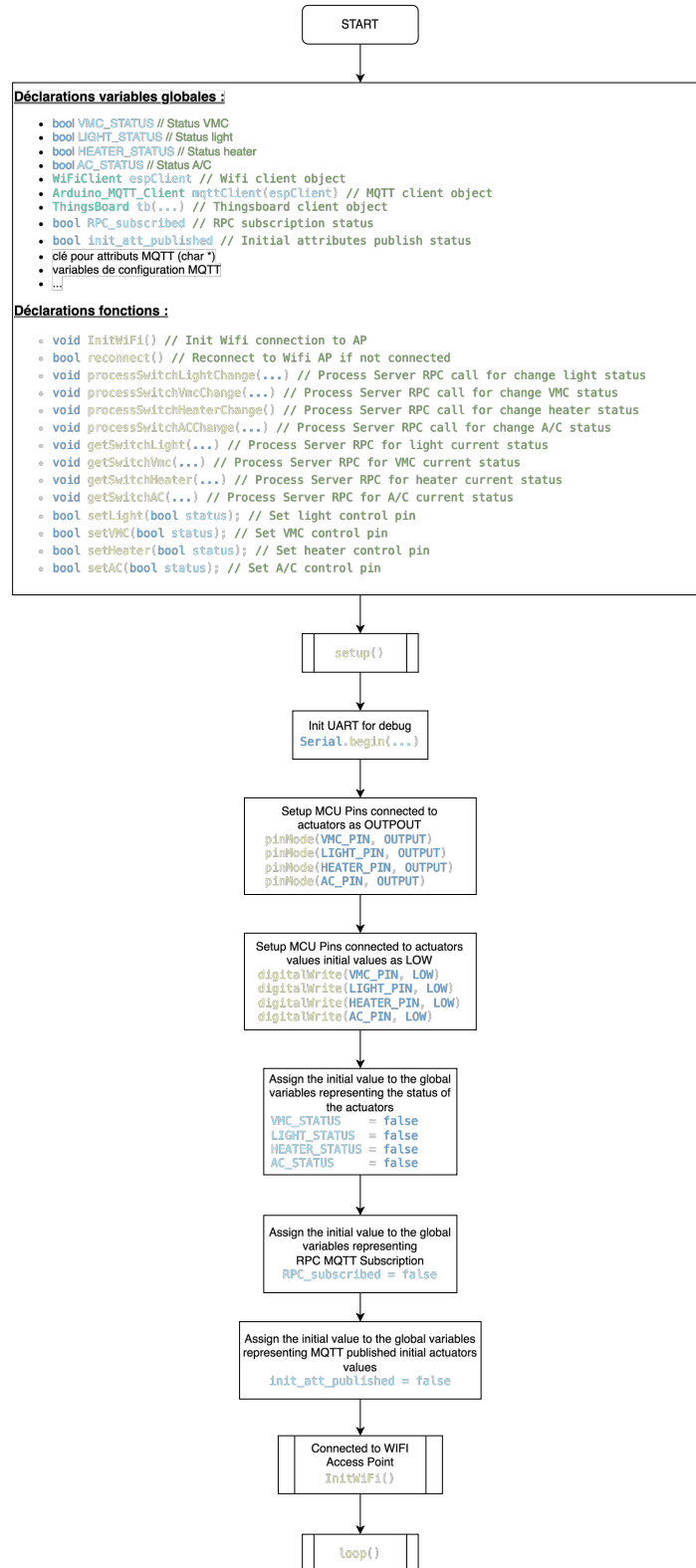
IOT & Systèmes embarqués 2025	INFRES TP IOT & Systèmes embarqués 2025 Développement software embarqué Module Actionneur TP Partie 1	TP - IMT 2025 IoT Part. 1.docx
Créé le 05/04/2025 21/05/2025 00:49:00		Rev. 0
Créé par Lazarewicz Julien		Page 18 sur 26
		 IMT Mines Alès École Mines-Télécom

6.1.1. Algorigramme



INFRES

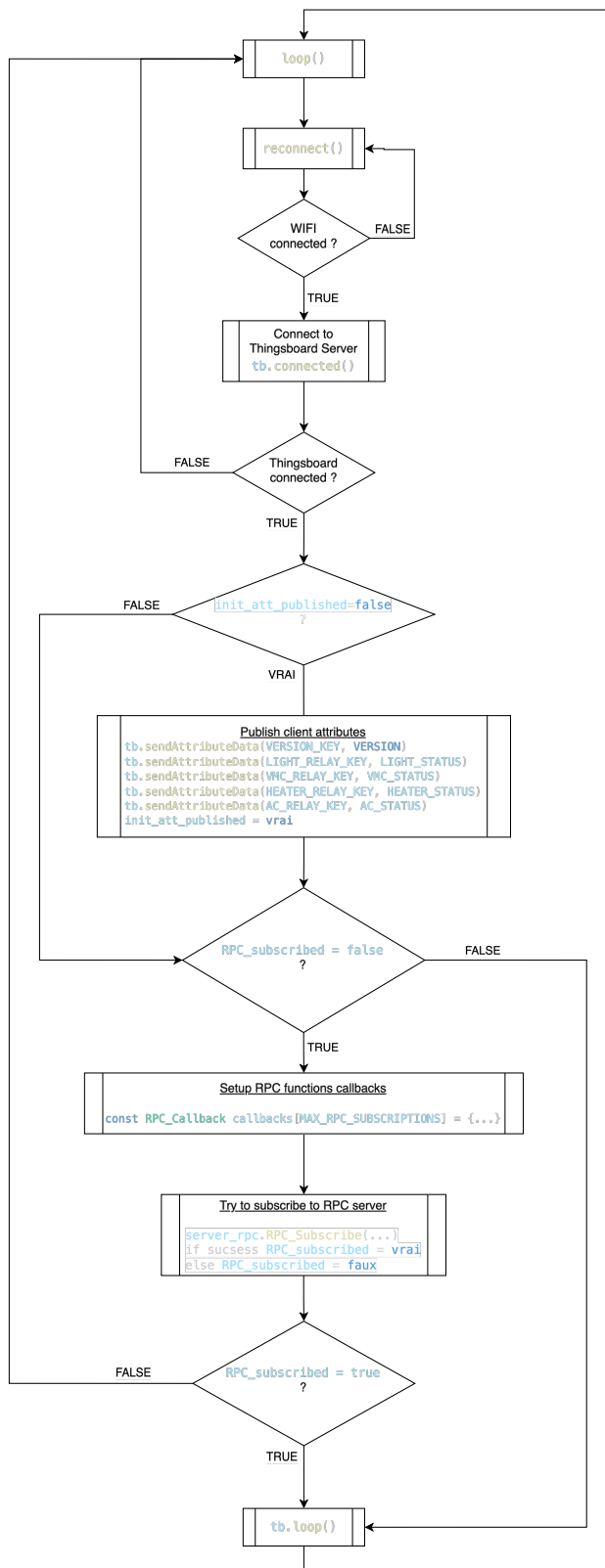
TP IOT & Systèmes embarqués 2025 Développement software embarqué Module Actionneur TP Partie 1

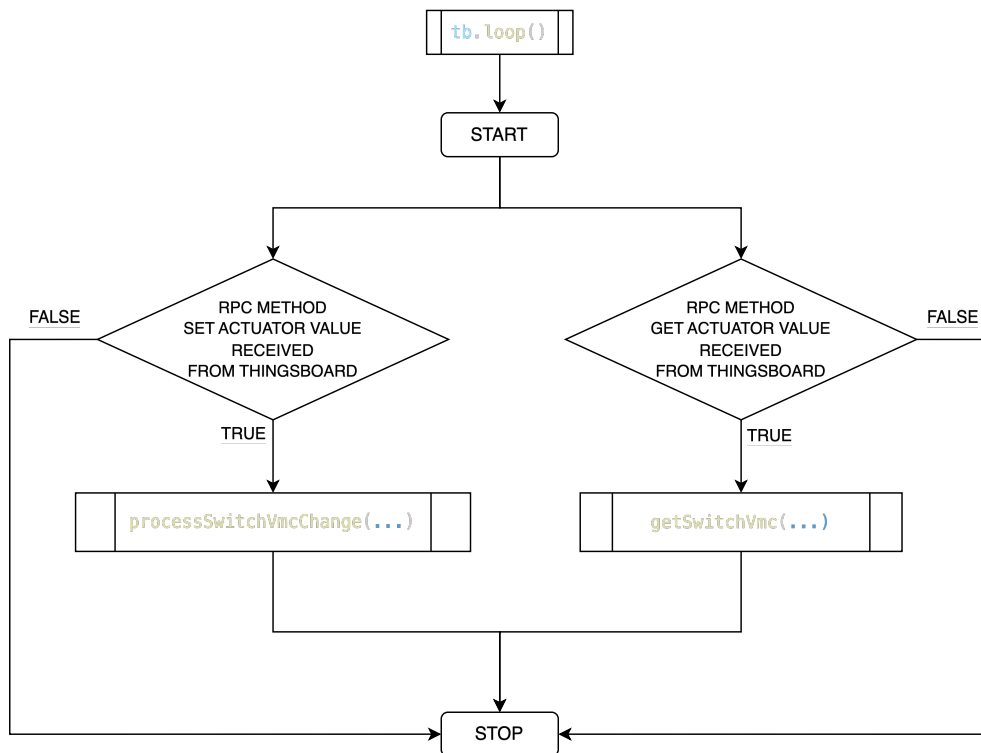




INFRES

TP IOT & Systèmes embarqués 2025 Développement software embarqué Module Actionneur TP Partie 1





6.1.2. Récupération du template

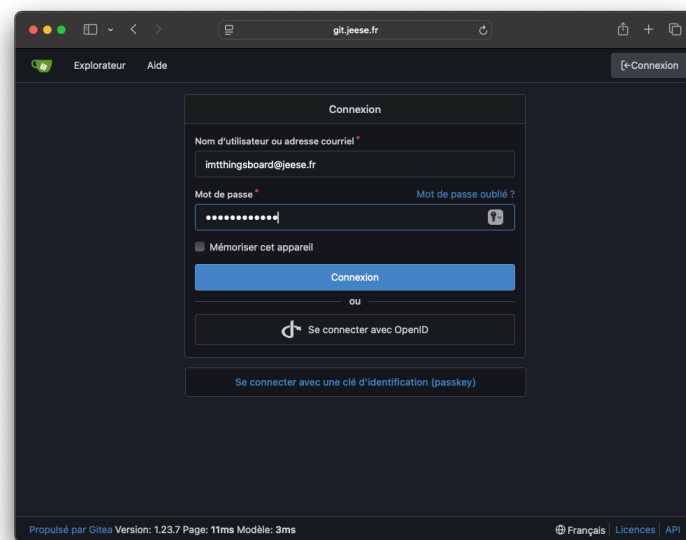
Pour télécharger le template du projet deux options s'offrent à vous :

- Utiliser Git en ligne de commande :
- Ouvrir un terminal
- Naviguer vers l'emplacement où vous souhaitez cloner le projet
- Cloner le projet avec la commande « git clone https://git.jeese.fr/laz/IMT_IOT_ACTUATOR_TEMPLATE.git »

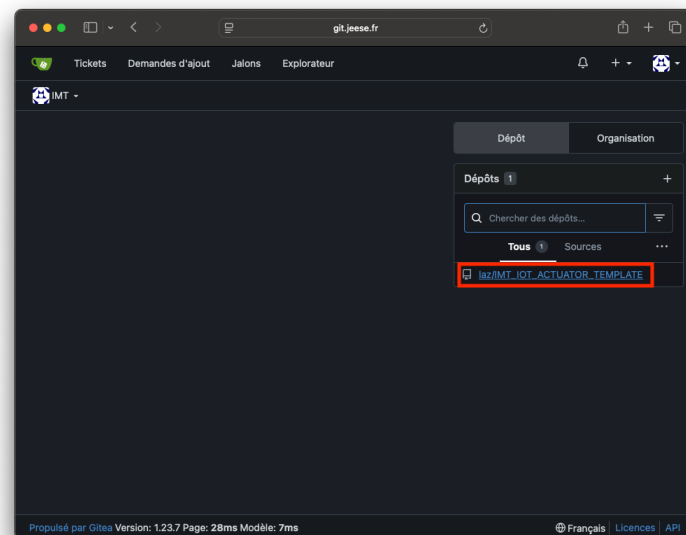
```

julia@Mac-mini-de-Julien ~ % mkdir TP_IOT
julia@Mac-mini-de-Julien ~ % cd TP_IOT
julia@Mac-mini-de-Julien TP_IOT % git clone https://git.jeese.fr/laz/IMT_IOT_ACTUATOR_TEMPLATE.git
Cloning into 'IMT_IOT_ACTUATOR_TEMPLATE'...
remote: Enumerating objects: 15, done.
remote: Counting objects: 100% (15/15), done.
remote: Compressing objects: 100% (10/10), done.
remote: Total 15 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (15/15), 7.23 KiB | 3.62 MiB/s, done.
julia@Mac-mini-de-Julien TP_IOT %
  
```

- Télécharger le projet depuis un navigateur web :
 Dans votre navigateur accédez à l'URL <https://git.jeeese.fr/>
 Identifiez-vous avec le login : imthingsboard@jeeese.fr et le mot de passe : RmK7SaLC9FAD.



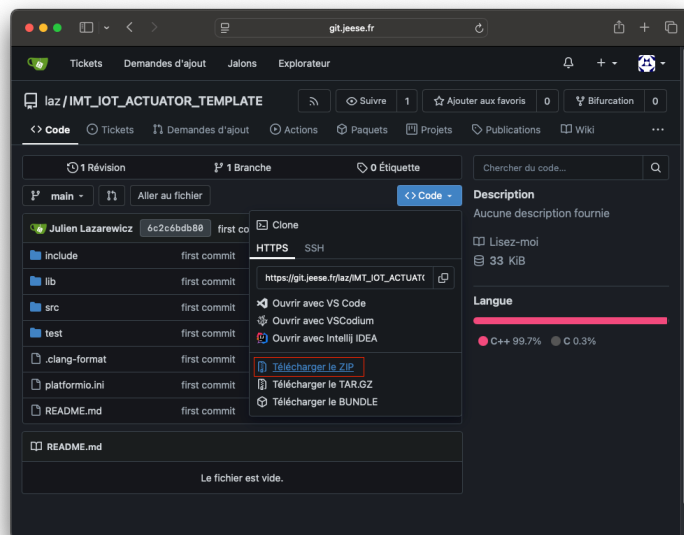
Sélectionnez le dépôt : « laz/IMT_IOT_ACTUATOR_TEMPLATE »



Cliquez sur « <>Code » puis choisissez « Télécharger le ZIP »

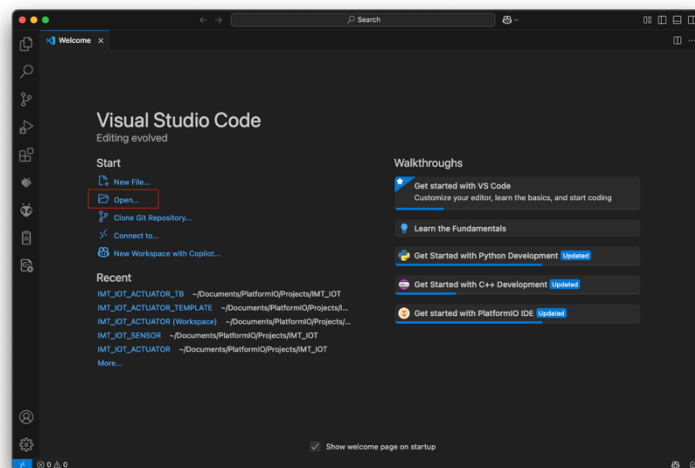
IOT & Systèmes embarqués 2025	INFRES TP IOT & Systèmes embarqués 2025 Développement software embarqué Module Actionneur TP Partie 1	TP - IMT 2025 IoT Part. 1.docx
Créé le 05/04/2025 21:05/2025 00:49:00		Rev. 0
Créé par Lazarewicz Julien		Page 23 sur 26
		 IMT Mines Alès École Mines-Télécom

Décompressez le fichier zip téléchargé.



6.1.3. Ouvrir le template dans VS Code :

- Lancer VS Code et cliquez sur « Open ... »



- Choisissez le dossier « IMT_IOT_ACTUATOR_TEMPLATE »

6.1.4. Contenu du projet template :

La description des différents fichiers et leur usage sera fourni à l'oral.

- Fichier « platformio.ini » : fichier de configuration du projet
- Fichier « include/version.h » header C contenant la version du firmware
- Fichier « include/config.h » header C contenant les paramètres comme le SSID du point d'accès Wifi, l'adresse IP du serveur Thingsboard, ...
- Fichier « main.cpp » source C++ contenant le code source du firmware

7. Connection à l'interface Web du serveur Thingsboard

7.1. Accéder à l'interface Web du serveur Thingsboard

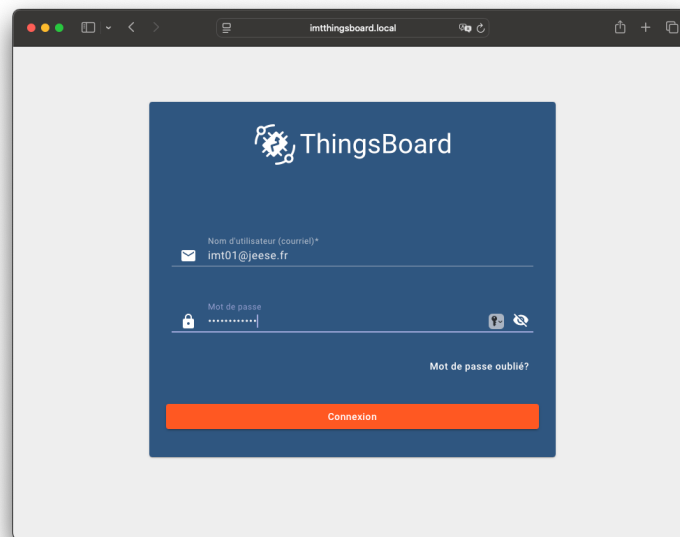
Pour vous connecter à l'interface Web, vous devez en premier lieu connecté votre ordinateur au point d'accès Wifi du serveur. Cela dépend type de système que vous utilisez.

Dans les paramètres Wifi rechercher le réseau Wifi nommée « thingsboard » et connectez-vous avec le mot de passe « thingsboard ».

Une fois le réseau Wifi rejoint, dans votre navigateur Web rendez-vous à l'URL :

- <https://imtthingsboard.local:8080/login> ou
- <https://10.42.0.1:8080/login>

Vous devriez voir apparaître la fenêtre de login suivant :

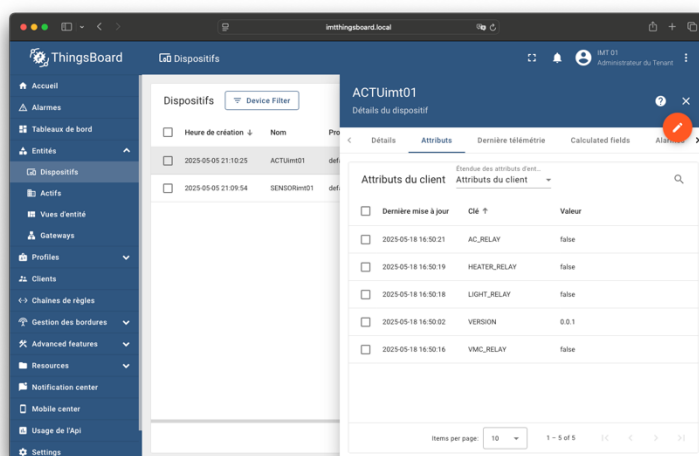
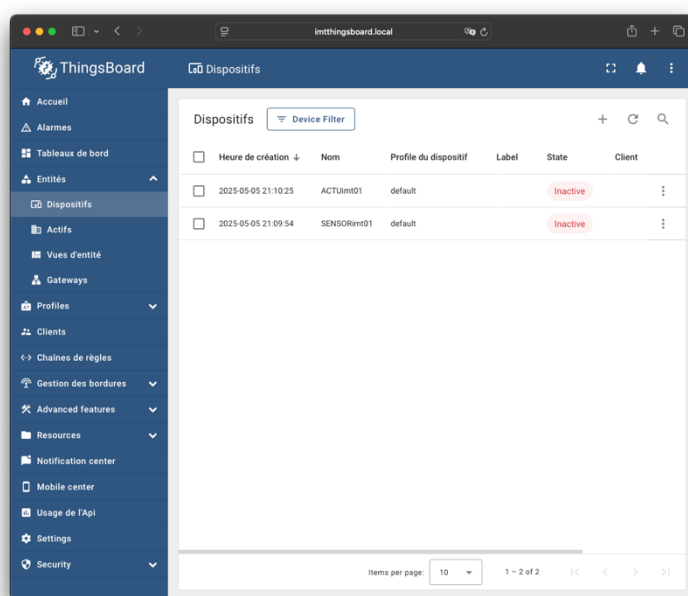


Un couple identifiant, mot de passe vous sera donné par binome le jour du TP.

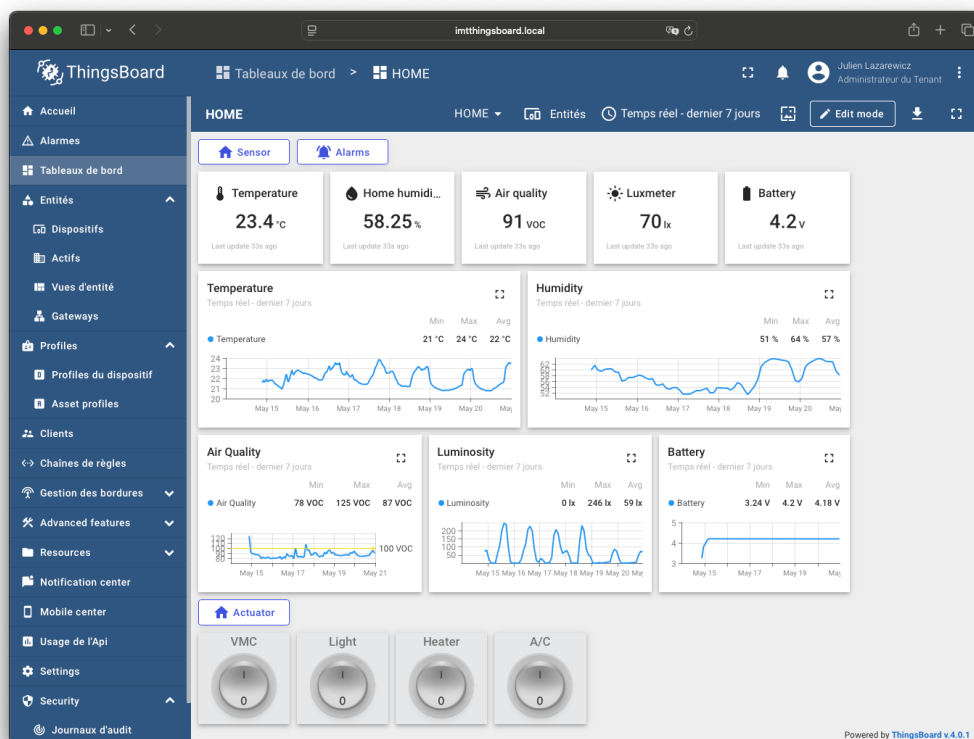
7.2. Votre compte Thingsboard

Votre compte Thingsboard sera préconfiguré. Nous ne pourrions pas aborder toutes les fonctionnalités de Thingsboard dans ce TP. Les éléments qui nous intéressent sont :

- Les entités « Dispositifs », pour ce TP le « MODULE CAPTEUR » et le « MODULE ACTIONNEUR » qui seront préconfigurés.



- Les « Tableaux de bord », un tableau de bord sera préconfiguré



- Si le temps nous le permet, les « Chaînes de règles »

