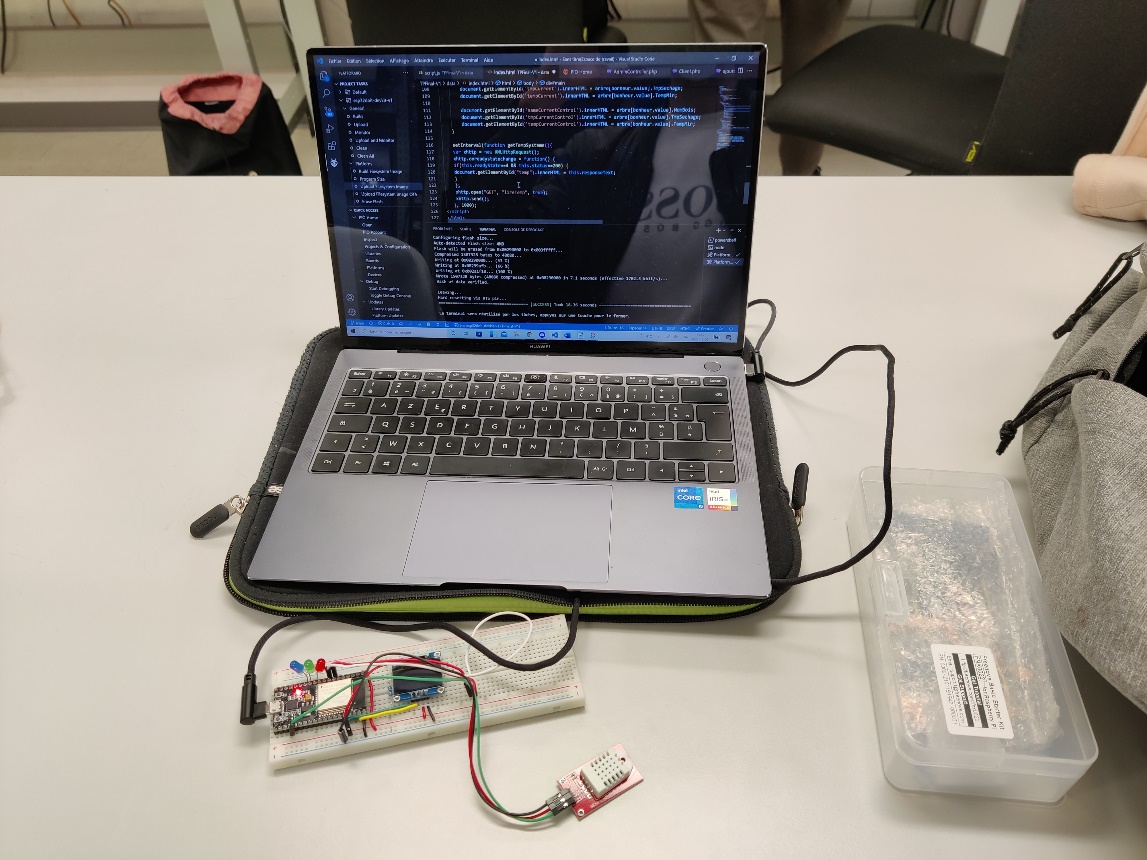
**

OBJETS CONNECTÉS

*Document de mise en service*

LECOMTE Guillaume – GUILLOU Quentin

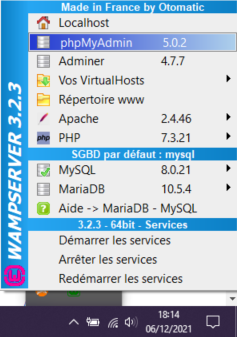
08/12/2021

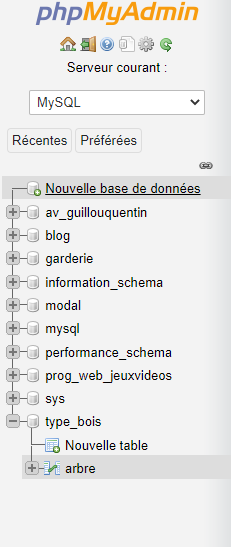
OBJETS CONNECTÉS

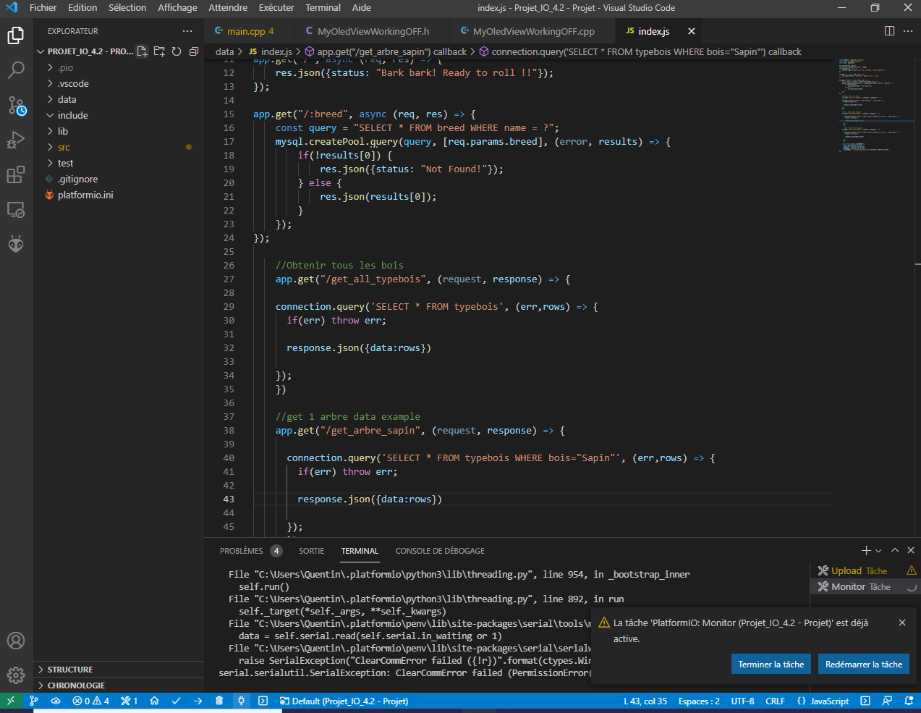
Document de mise en service

Nous allons maintenant vous expliquer comment mettre tout ce qu’il faut en service pour pouvoir accès à tous les services de l’application et de l’ESP.

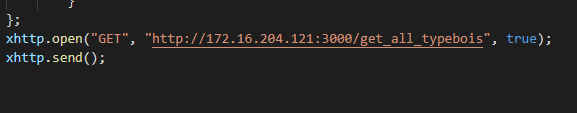
Dans un premier temps, il faut installer WampServer et de le lancer une fois installer. Une fois cela réalisé, vous lancez phpMyAdmin en faisant un clic gauche sur son icone ainsi que tous les fichiers sur notre GitHub.



Il vous suffit de vous connecter ensuite avec les identifiants de base qui sont « root » et « ». Une fois connecté, vous vous retrouver sur l’interface de phpMyAdmin. Dedans, vous devez cliquer sur la création d’une nouvelle base de donner et choisir « importer » dans la barre horizontale. En faisant cela, vous n’aurez qu’à aller chercher le fichier contenant la base de données sur le GitHub et à l’importer.

Dans un second temps, une fois tous les fichiers téléchargés et la base de données opérationnelle, vous devez installer node.js pour pouvoir utiliser l’API et récupérer les données. Quand cela sera fait, ouvrer le dossier contenant le projet avec une application comme visual studio code par exemple.

Il vous faudra changer une partie de code dans le dossier script.js qui se trouve dans le dossier data :



Vous allez remplacer l’IP présent par le vôtre. Pour le connaitre, ouvrez une invite de commande en tapant cmd dans a barre de recherche Windows et écrire « ipconfig ». Prenez l’adresse IP qui correspond à votre réseau.

Pour continuer, vous devrez garder ouvert votre invite de commande ou aller dans le terminal de votre application.

Dans ce terminal, vous devez écrire : npm start. Cela à pour action de lancer votre serveur et donc votre API par la même occasion. Ouvrez internet et taper dans la barre de recherche : <http://localhost:3000/>.

Si tout fonctionne, vous devriez voir ce message :

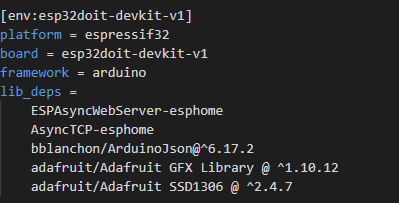
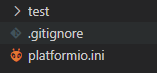


Ensuite, pour savoir si tout fonctionne et si l’appel à la base de données et la table des bois en question se fait, taper l’adresse suivante : localhost:3000/get\_all\_arbre.

Voir devez voir en ce moment ces éléments :

{"Sapin" : {"NomBois" : "Sapin","TypeBois" : "Dur","OrigineBois" : "Nord", "TmpSechage" : "6","TempMin": 15},"Erable" : {"NomBois" : "Erable", "TypeBois": "Dur", ,"OrigineBois" : "Nord-Equateur","TmpSechage" : "6", "TempMin": 18},"Chêne" : {"NomBois" : "Chêne", "TypeBois": "Dur","OrigineBois" : "Europe", "TmpSechage" : "10", "TempMin": 20}

Maintenant, nous savons que notre API fonctionne bien car elle nous retourne des données de la table des bois souhaitée. Nous allons maintenant nous occuper de la partie C++ du projet. Pour que tout fonctionne, il vous faut installer certaines librairies dans voter fichier plateformio.ini



Il vous suffit juste d’écrire les différentes dépendances comme ci-dessus.

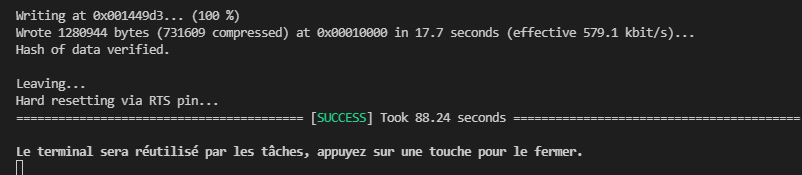
De plus, quand cela sera fait, vérifier que toutes les classes qui se trouvent dans le dossier en question sur notre GitHub sont toutes présentes dans le dossier « lib ». Si cela n’est pas le cas, téléchargez celles qui sont manquantes.

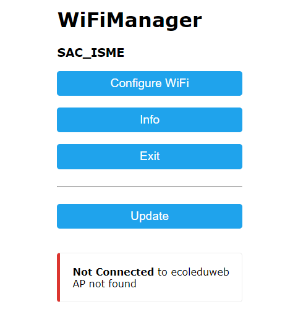
Pour terminer, suivez ces étapes :

1. En bas de votre écran lorsque vous vous trouvez dans l’application, vous verrez ceci :



Cliquer sur la flèche pour lancer le l’upload.

1. Cliquer sur le bouton si cela est nécessaire.
2. Une fois l’upload terminée, cliquez sur la prise nommée « Serial Monitor ».
3. Si un message comme celui-ci s’affiche, connectez vous sur le réseau wifi nommé SCA\_ISME et une fenêtre s’ouvrira dans votre navigateur.



1. Cliquez sur « Configure WIFI »
2. Connectez vous au réseau wifi sur lequel vous vous trouviez en rentrant les identifiants pour s’y connecter et validez.
3. Il vous suffit d’appuyer une nouvelle fois sur le bouton « Serial Monitor » et tout devrait être foinctionnel.