

Tutoriel d'utilisation de VSCode et PlatformIO pour le hackathon 2024

Par Antoine Chassaigne

Pour AREM



Sommaire

Sommaire	1
Installation de VSCode (Windows)	2
Téléchargement du code du robot	3
Installation de l'extension PlatformIO dans VSCode	5
Ouverture du projet Hackathon	7

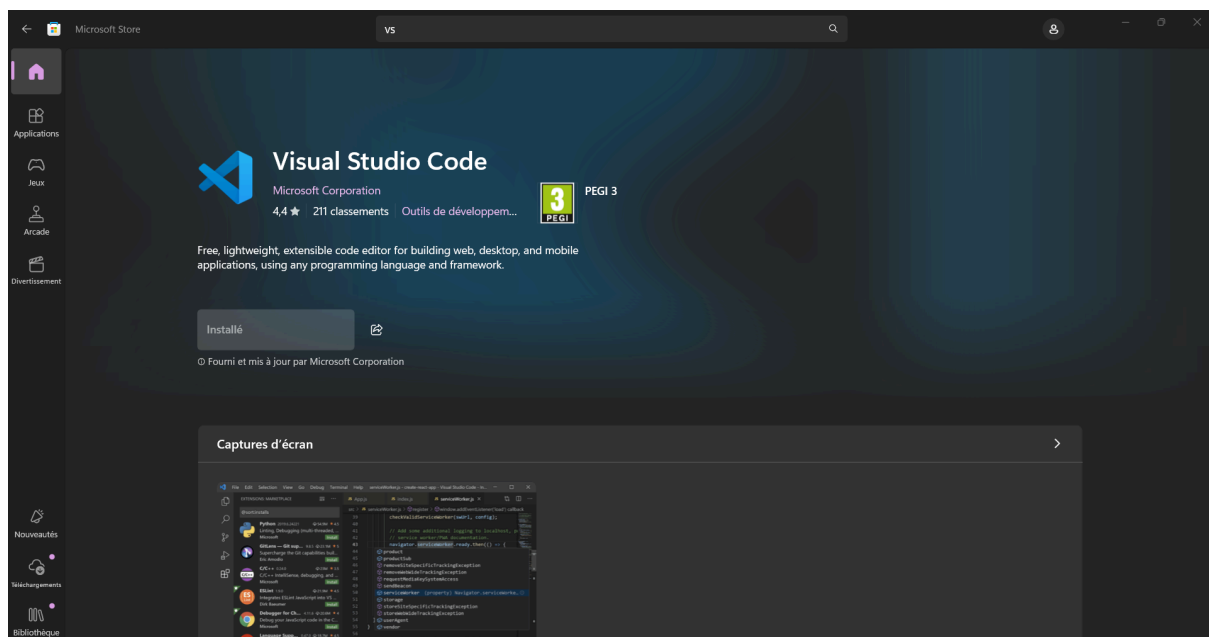
Installation de VSCode (Windows)

Un petit peu d'explications:

VSCode est avant tout et surtout un éditeur de texte. Sa puissance réside dans la possibilité d'y ajouter des extensions pour toutes sortes de tâches: coloration syntaxique et autocomplétion pour n'importe quel langage de programmation, lecteur de pdf, visionneuse de modèles 3D, ... Certaines extensions permettent de compiler du code mais je ne vais pas m'étendre sur le sujet. VSCode possède également une interface simplifiée pour utiliser le gestionnaire de versions git ainsi qu'un terminal intégré (le même que celui de windows (cmd) mais directement dans le logiciel ce qui est très pratique.).

Si vous avez un MAC ou un PC sous linux je vous invite à regarder les tutos en ligne. Normalement ce n'est pas très différent.

Sur windows, rien de plus simple, l'application est directement disponible sur le microsoft store:

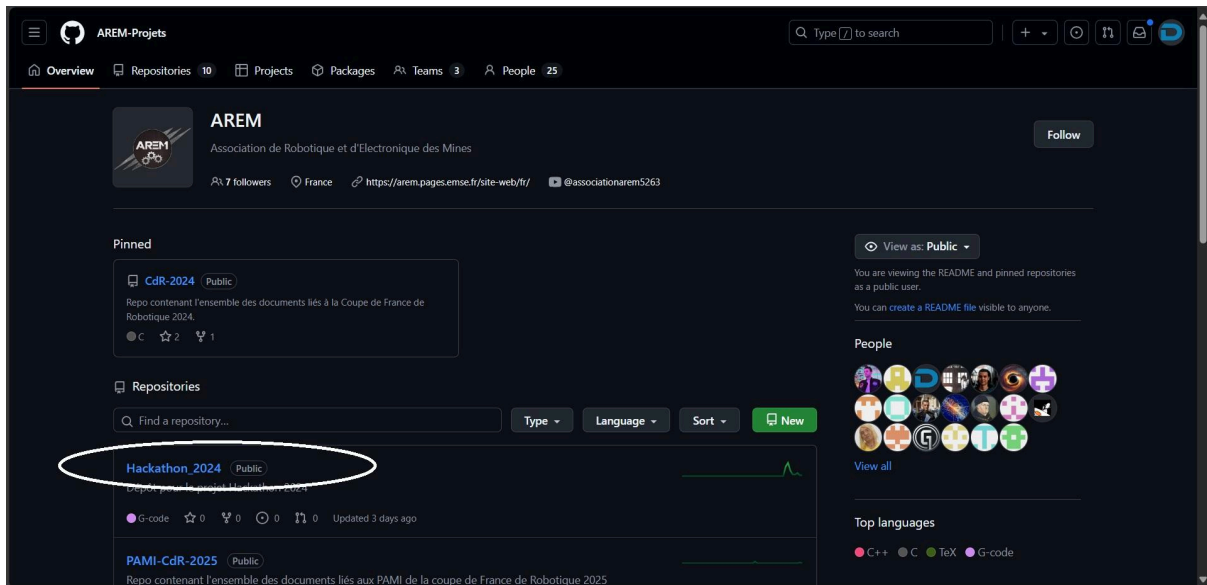


Téléchargement du code du robot

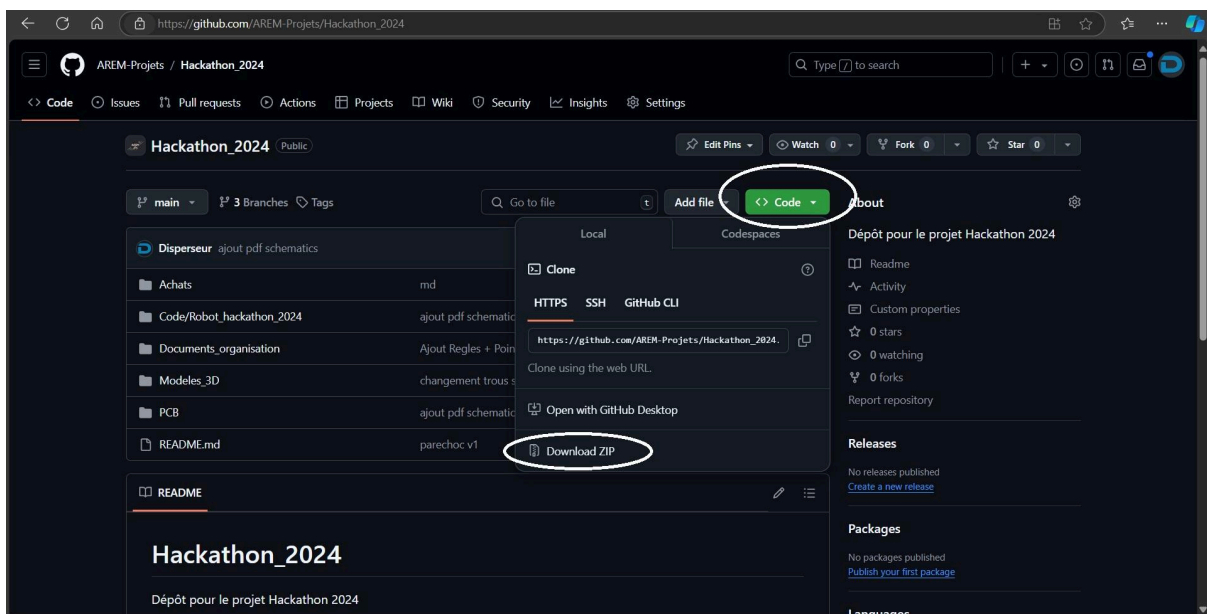
Le code du robot, comme tous les autres fichiers du développement de ce dernier, se trouvent dans le répertoire Hackathon_2024 du github AREM. Vous pouvez le trouver ici:

The screenshot shows a Google search interface with the query 'github arem'. Below the search bar, there are tabs for 'RECHERCHER', 'COPILOT', 'IMAGES', 'VIDÉOS', 'CARTES', 'ACTUALITÉS', and 'SHOPPING'. The search results indicate 'Environ 181 000 résultats'. The first result is a GitHub repository titled 'CdR-2024' with the URL 'https://github.com/AREM-Projets/CdR-2024'. The description states it is a repository containing documents for the Coupe de France de Robotique 2024. It shows 1 Watched, 2 Starred, and 1 Forks. Below the repository name, there is a tag 'cdr' and a language indicator 'Langage principal : C'. Two buttons are visible: 'Afficher le projet' and 'Copier l'URL du clone'. Below this, a second result is partially visible, titled 'AREM - GitHub' with the URL 'https://github.com/AREM-Projets'. This second result is circled in black. The description for the second result mentions '20 oct. 2023 · Repositories. Contient la liste des capteurs présents au local AREM ainsi que de la documentation associée. Repo contenant l'ensemble des documents liés à la Coupe de France ...'.

Une fois dans le github de l'asso, vous pourrez trouver le répertoire "Hackathon_2024" un peu plus bas:



Une fois ouvert, vous avez le contenu du répertoire sous les yeux. Par soucis de simplicité, vous allez télécharger le contenu du répertoire en format .zip comme montré ci-dessous:



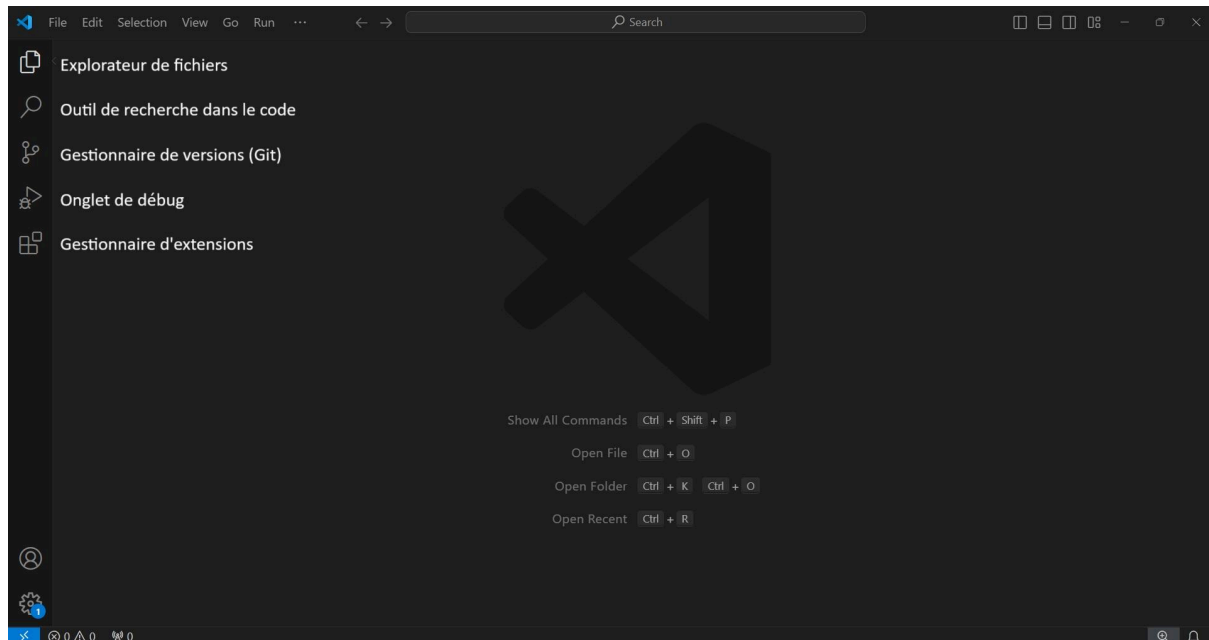
Veillez à enregistrer le .zip à un endroit facile d'accès pour que vous n'ayez pas de difficulté à le retrouver plus tard.

Installation de l'extension PlatformIO dans VSCode

Comme expliqué plus tôt, VSCode est un éditeur de texte extensible via des modules. Ici nous aurons besoin d'une extension très particulière qui s'appelle PlatformIO. C'est une plateforme de développement pour microcontrôleurs intégrée dans VSCode. Elle permet d'écrire, de compiler et de téléverser du code dans une grande variété de microcontrôleurs dont ceux de STMicroelectronics en utilisant par exemple un framework Arduino (en gros la syntaxe et les commandes arduino). Elle permet aussi d'utiliser des bibliothèques de fonctions et comporte plein d'autres fonctionnalités très utiles. En soit c'est un peu une usine à gaz pour reprendre l'expression d'un certain 3A mais cette extension s'avère pratique pour faire des choses qui fonctionnent bien sans trop se préoccuper de comment cela se passe en profondeur. Le driver du robot du hackathon a été programmé sur cette plateforme par souci de simplicité pour la mise en œuvre. C'est cet outil que je vais vous faire installer ci-dessous.

Vous pouvez démarrer VScode.

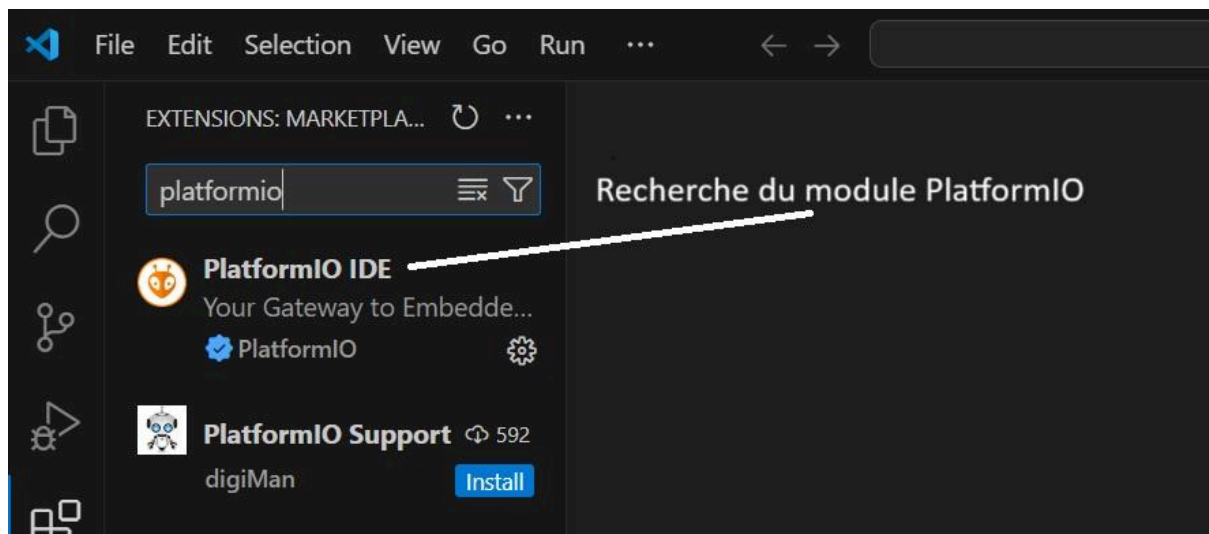
Normalement (modulo les extensions déjà installées ou pas), vous avez au minimum cette interface: (peut être un fichier ouvert automatiquement avec les mises à jour possibles ou quoi. Pas de panique, vous pouvez le fermer.)



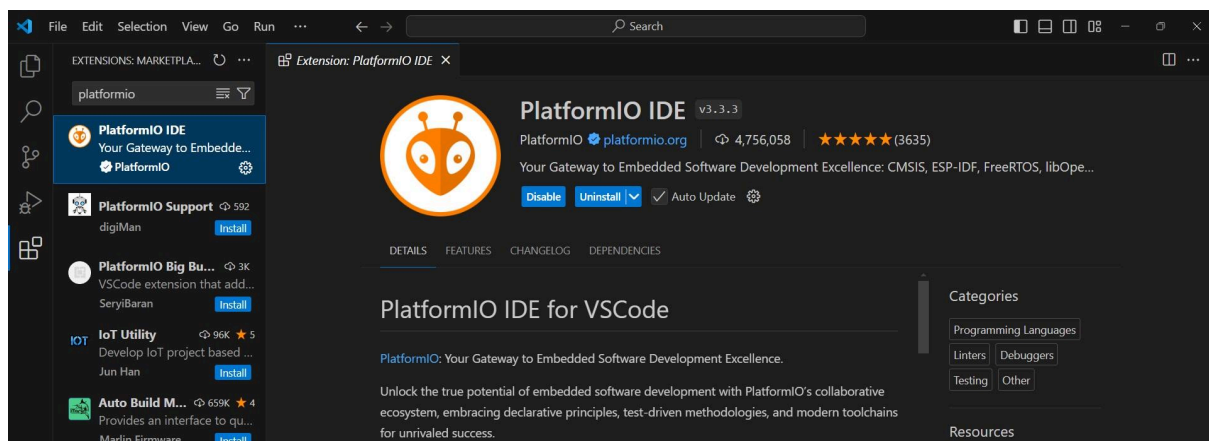
Vous avez donc plusieurs outils disponibles dans l'application sur la barre de gauche.

Pour l'instant, nous allons avoir besoin de l'onglet gestionnaire d'extensions pour installer PlatformIO. Cliquez sur l'icône correspondante. Vous avez une barre de recherche en haut.

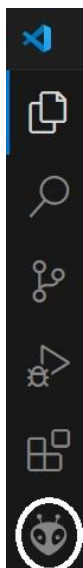
Tapez “platformio” dedans et faites “Entrée”. Normalement (si vous êtes connectés à internet), vous devez avoir des propositions d’extensions correspondantes. Cliquez sur la première:



Un beau bouton bleu “Install” ou “Installer” vous attends, vous pouvez cliquer dessus pour installer l’extension:



(Ce que vous devez avoir une fois l’installation terminée)



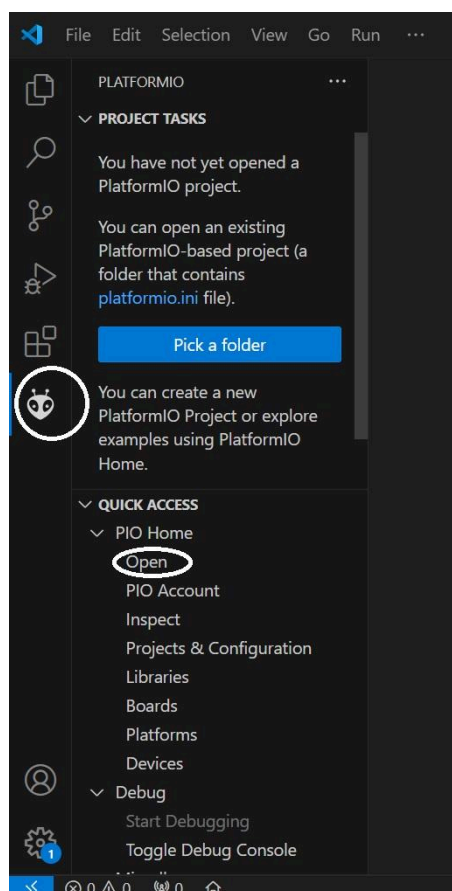
Normalement une fois l’installation terminée vous devrez redémarrer le logiciel et vous aurez une nouvelle icône dans la barre de gauche. Toutes les icônes sous celle du gestionnaire d’extensions sont celles d’extensions. Celle de PlatformIO ressemble à un petit alien. Si elle n’apparaît pas, faites clic-droit dans la barre de gauche sous l’icône du gestionnaire d’extensions puis cochez PlatformIO. En effet on peut choisir quelles extensions parmi celles installées sont affichées dans la barre de gauche.

Ouverture du projet Hackathon

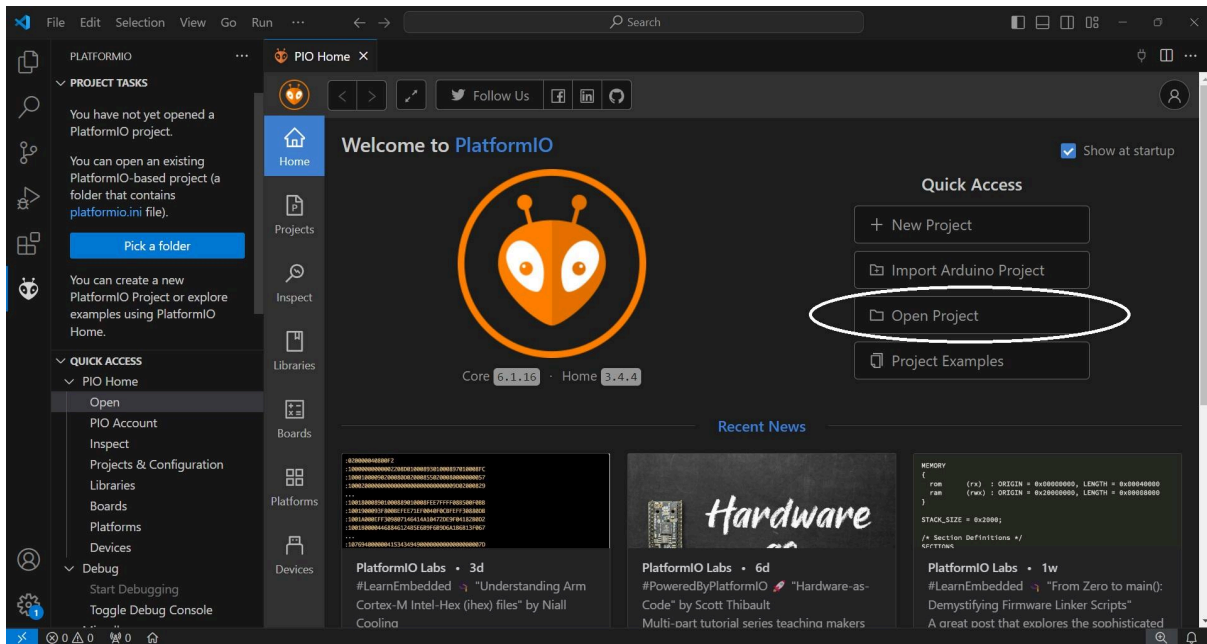
Une fois que votre environnement de travail (VSCode et PlatformIO) est bien paramétré et que vous avez téléchargé le contenu du répertoire Hackathon_2024, il ne reste plus qu'à ouvrir le code pour le modifier et entrer dans le vif du sujet du Hackathon !

Pour ce faire, cliquez sur l'icône de PlatformIO (le petit alien) dans la barre de gauche. PlatformIO va alors démarrer (cela peut prendre un peu de temps, notamment lors de la première exécution. Regardez la barre en bas à droite de l'écran, c'est là que s'affiche le statut de démarrage).

Vous devriez arriver sur un écran comme celui-ci:



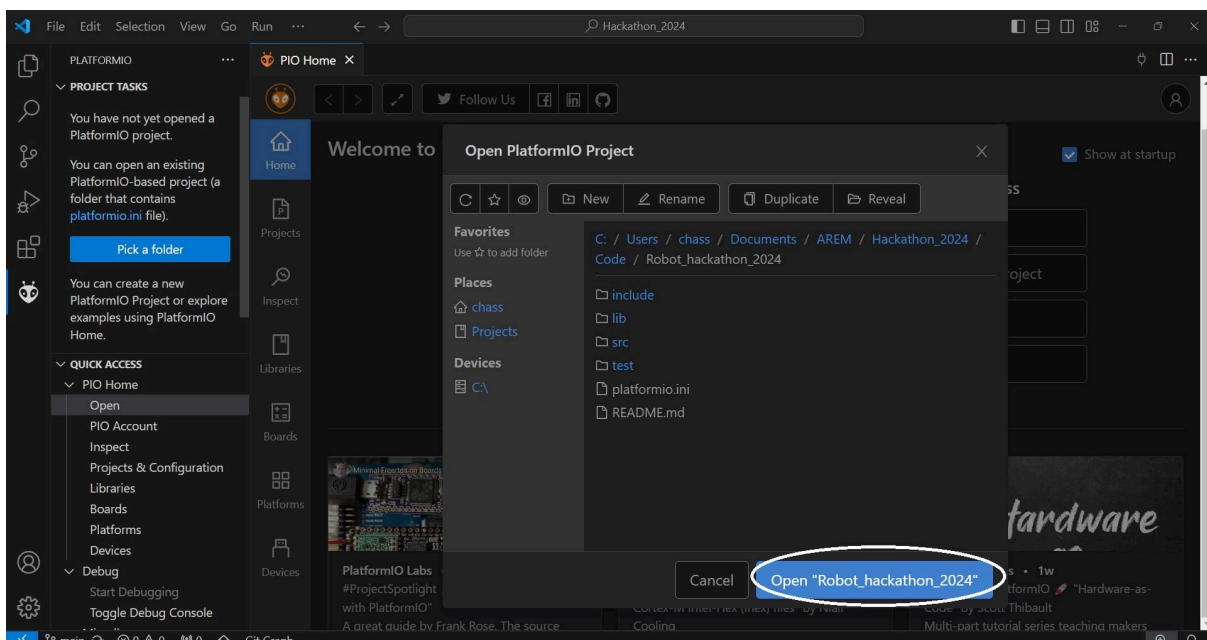
Vous pouvez faire "Open", puis open project:



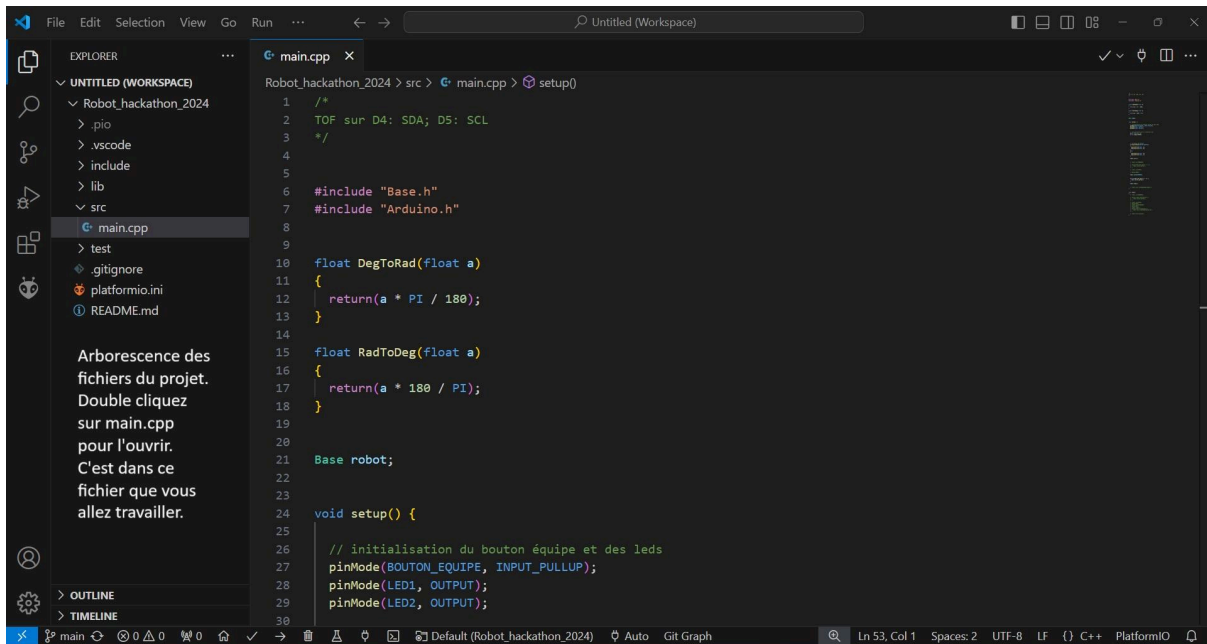
Ensuite parcourez votre arborescence pour retrouver le dossier Hackathon_2024. Le dossier du projet PlatformIO se trouve à cet endroit dans le répertoire Hackathon_2024:

Hackathon_2024 > Code > Robot_hackathon_2024

Une fois ce sous-répertoire ouvert, faites "Open" (bouton bleu en bas, parfois non visible). Il suffit de scroller avec la fenêtre pour le voir):

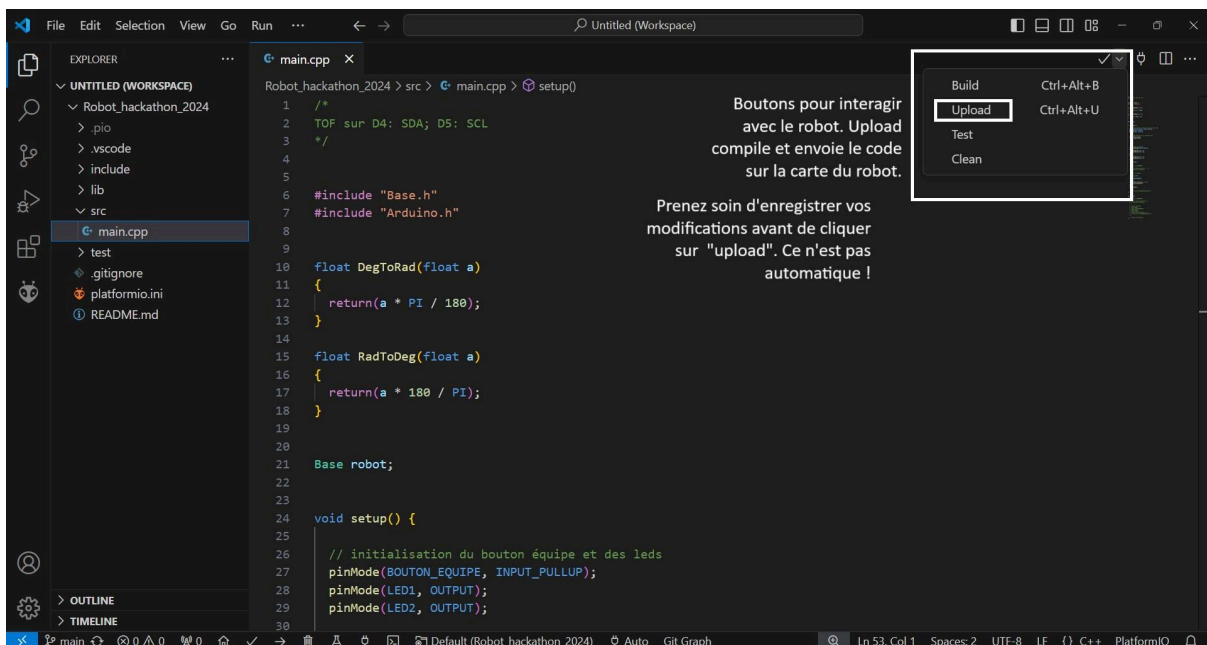


Vous allez ensuite vous trouver dans un environnement de travail prêt pour le hackathon:

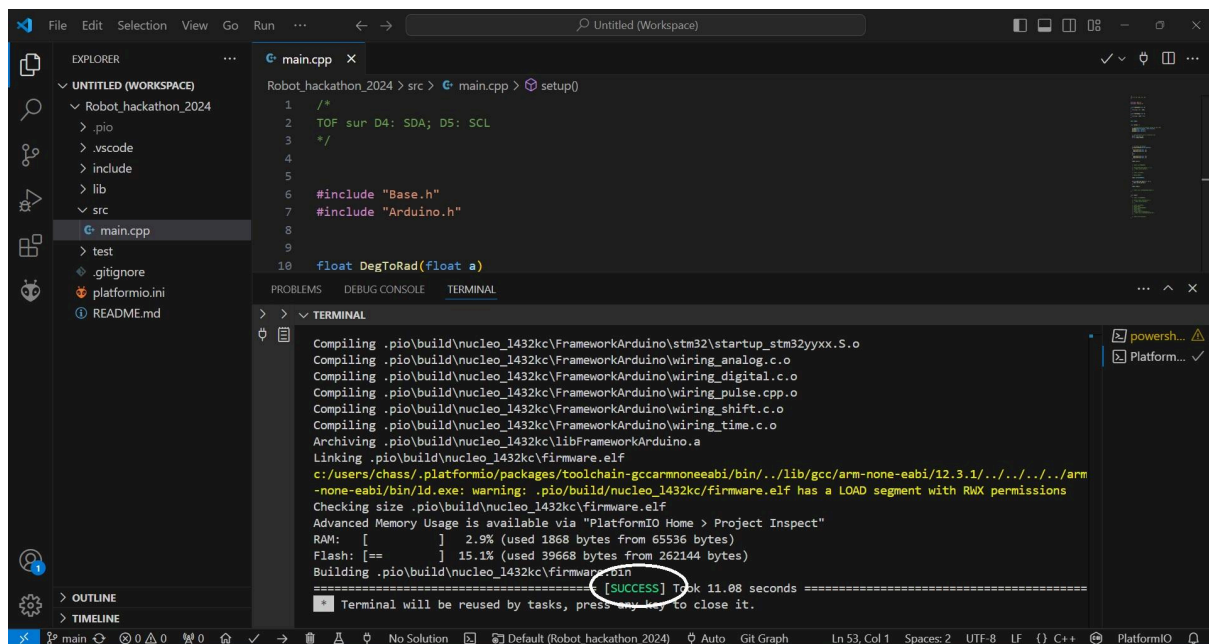


A gauche se trouve l'arborescence du projet. Le seul fichier qui va vous être utile est le **main.cpp** et le pdf contenant le descriptif des commandes (**README.md**).

En haut à droite vous trouverez les boutons pour compiler ou compiler et téléverser le code dans le robot:



Lorsque vous lancez le téléversement ou une compilation, si elle réussit vous devriez voir un "SUCCESS" vert dans le terminal:



The screenshot shows the Visual Studio Code interface. The Explorer panel on the left shows a project named 'Robot_hackathon_2024' with a file tree containing 'pio', '.vscode', 'include', 'lib', 'src' (with 'main.cpp' selected), 'test', '.gitignore', 'platformio.ini', and 'README.md'. The main editor displays 'main.cpp' with the following code:

```
1  /*
2  TOF sur D4: SDA; D5: SCL
3  */
4
5
6  #include "Base.h"
7  #include "Arduino.h"
8
9
10 float DegToRad(float a)
```

The TERMINAL panel at the bottom shows the output of a PlatformIO compilation. The output includes various compilation steps and a final message: `[SUCCESS] Took 11.08 seconds`. The word 'SUCCESS' is circled in green. The status bar at the bottom indicates the active file is 'main.cpp' and the platform is 'PlatformIO'.

Par la suite vous aurez besoin de lire le README.md du projet PlatformIO qui contient les explications des commandes pour contrôler le robot.

Bon hackathon et bonne programmation à tous !

Si vous avez des questions, n'hésitez pas à les poser sur le discord
AREM !