



**CENTRO DE DESARROLLO
TECNOLÓGICO**



GABRIEL EDUARDO RENGIFO
Asesor Tecnológico y empresario
Centro de desarrollo tecnológico CreaTIC



Asesor tecnológico



Microsoft Student Partner



ION | DEVELOP STUDIOS
THINK IT SMARTER, DO IT BETTER

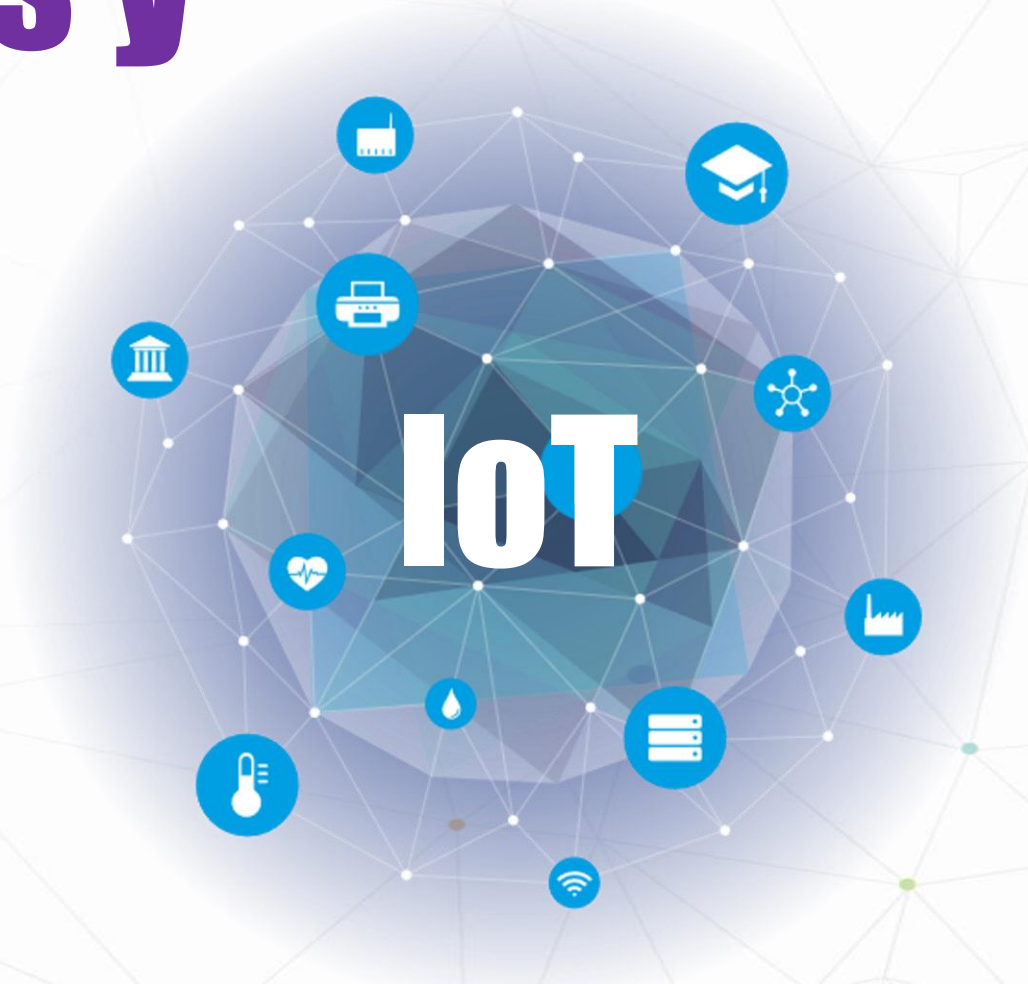
CTO and Co-founder



CPO and Co-founder



Aplicaciones y servicios



Agenda

Conceptos básicos:

- Aplicación de IoT en la industria
- Dispositivos embebidos y de prototipado
- Arquitectura de soluciones IoT
- Taller de ideación y planteamiento proyectos

Agenda

Arduino y Protocolo MQTT:

- Manejo avanzado de Arduino
- Sensores, actuadores y microcontroladores
- Taller de conectividad

Tracking usando GPS y RFID

Bluetooth Low Energy

- Creación de Cliente/Servidor con MQTT

Contralando dispositivos mediante MQTT

Agenda

Raspberry PI como Servidor:

- Configuración del SO y manejo de SSH
- Configuración de puertos de enlace y exposición a LAN
- Puentes de comunicación con Arduinos y otros dispositivos

Agenda

Monitoreo en tiempo real:

- Generación de un Dashboard con Node.js, React, Express y mongo
- Integración MQTT con Node.js
- Proyecto final

Bonus.

Introducción a GIT



crea un repositorio nuevo

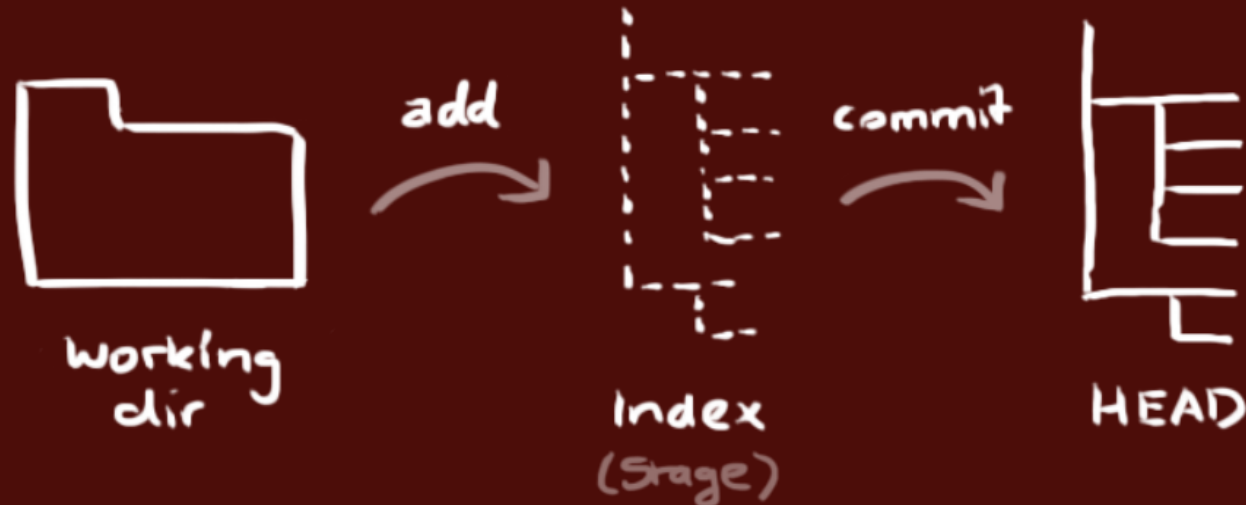
Crea un directorio nuevo, ábrelo y ejecuta

```
git init
```

para crear un nuevo repositorio de git.

flujo de trabajo

Tu repositorio local esta compuesto por tres "árboles" administrados por git. El primero es tu **Directorio de trabajo** que contiene los archivos, el segundo es el **Index** que actua como una zona intermedia, y el último es el **HEAD** que apunta al último commit realizado.



add & commit

Puedes registrar cambios (añadirlos al **Index**) usando

```
git add <filename>
```

```
git add .
```

Este es el primer paso en el flujo de trabajo básico. Para hacer commit a

estos cambios usa

```
git commit -m "Commit message"
```

Ahora el archivo está incluido en el **HEAD**, pero aún no en tu

repositorio remoto.

envío de cambios

Tus cambios están ahora en el **HEAD** de tu copia local. Para enviar estos cambios a tu repositorio remoto ejecuta

```
git push origin master
```

Reemplaza *master* por la rama a la que quieres enviar tus cambios.

Si no has clonado un repositorio ya existente y quieres conectar tu repositorio local a un repositorio remoto, usa

```
git remote add origin <server>
```

Ahora podrás subir tus cambios al repositorio remoto seleccionado.

ramas

Las ramas son utilizadas para desarrollar funcionalidades aisladas unas de otras. La rama *master* es la rama "por defecto" cuando creas un repositorio. Crea nuevas ramas durante el desarrollo y fusi6nalas a la rama principal cuando termines.



actualiza & fusiona

Para actualizar tu repositorio local al commit más nuevo, ejecuta

```
git pull
```

en tu directorio de trabajo para *bajar* y *fusionar* los cambios remotos.

Para fusionar otra rama a tu rama activa (por ejemplo master), utiliza

```
git merge <branch>
```

en ambos casos git intentará fusionar automáticamente los cambios.

Desafortunadamente, no siempre será posible y se podrán producir *conflictos*. Tú eres responsable de fusionar esos *conflictos* manualmente

al editar los archivos mostrados por git. Después de modificarlos,

necesitas marcarlos como fusionados con

```
git add <filename>
```

“

por Roger Dudler (traducido por @lfbarragan y @adrimatellanes)

créditos a @tfnico, @fhd y Namics

<https://rogerdudler.github.io/git-guide>

2.

Arduino y Protocolo MQTT

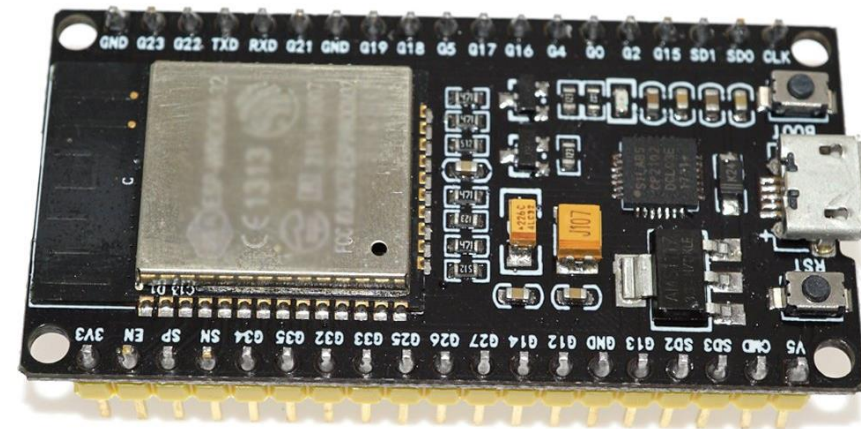


¡Hola Mundo!

2

Requisitos:

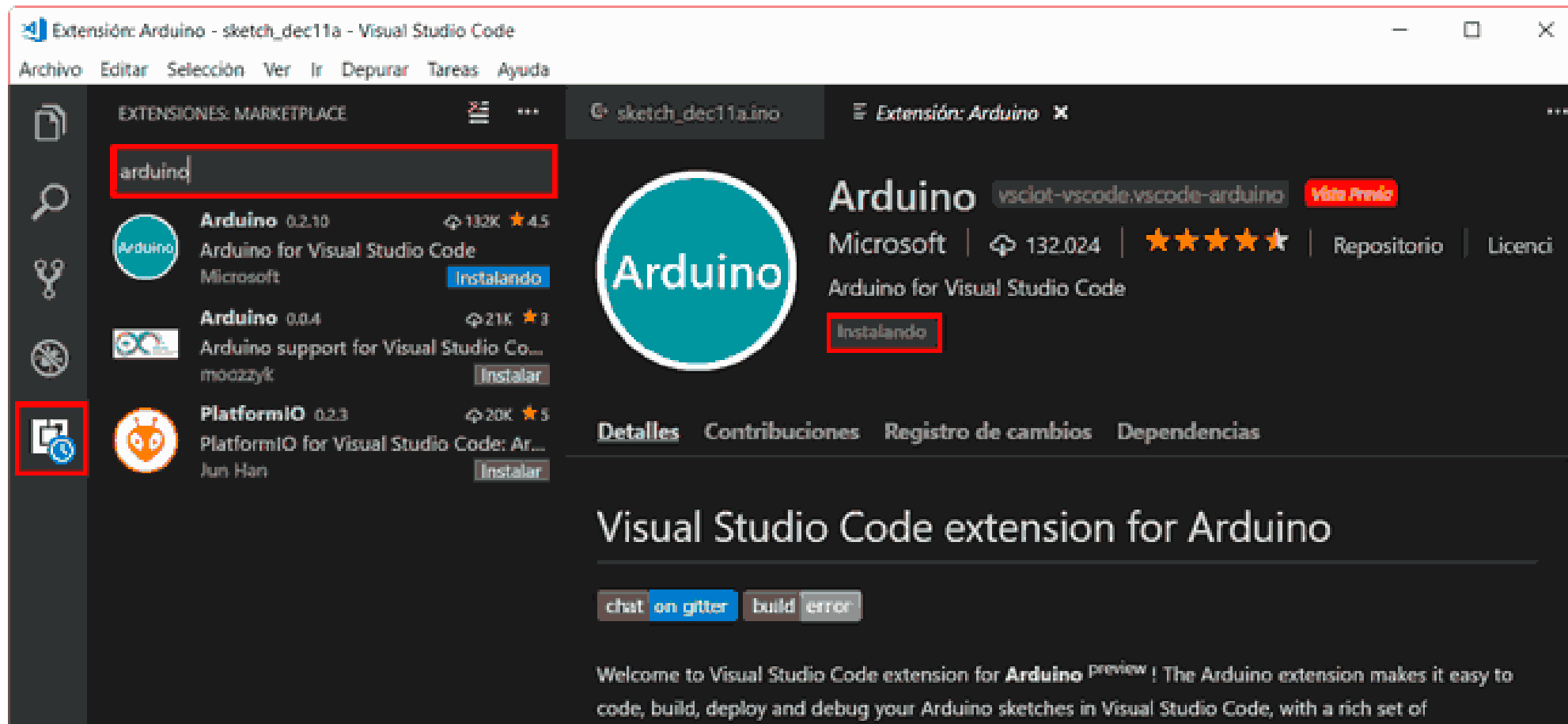
- Arduino IDE
- Controlador de Arduino
- ESP32/ ESP8266
- Cable USB
- VS Code



Configurar el entorno:

¡Hola Mundo!

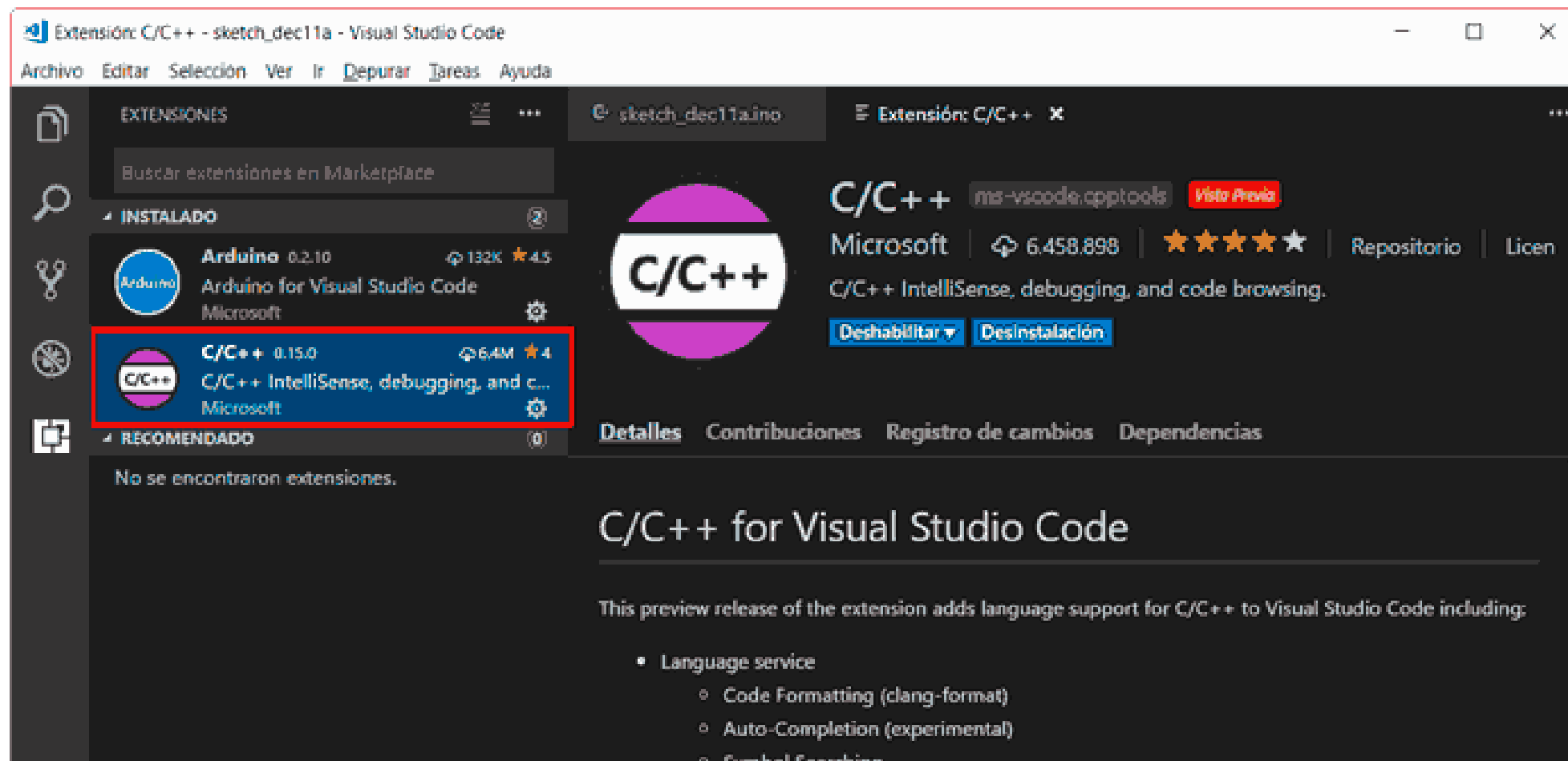
2



Configurar el entorno:

¡Hola Mundo!

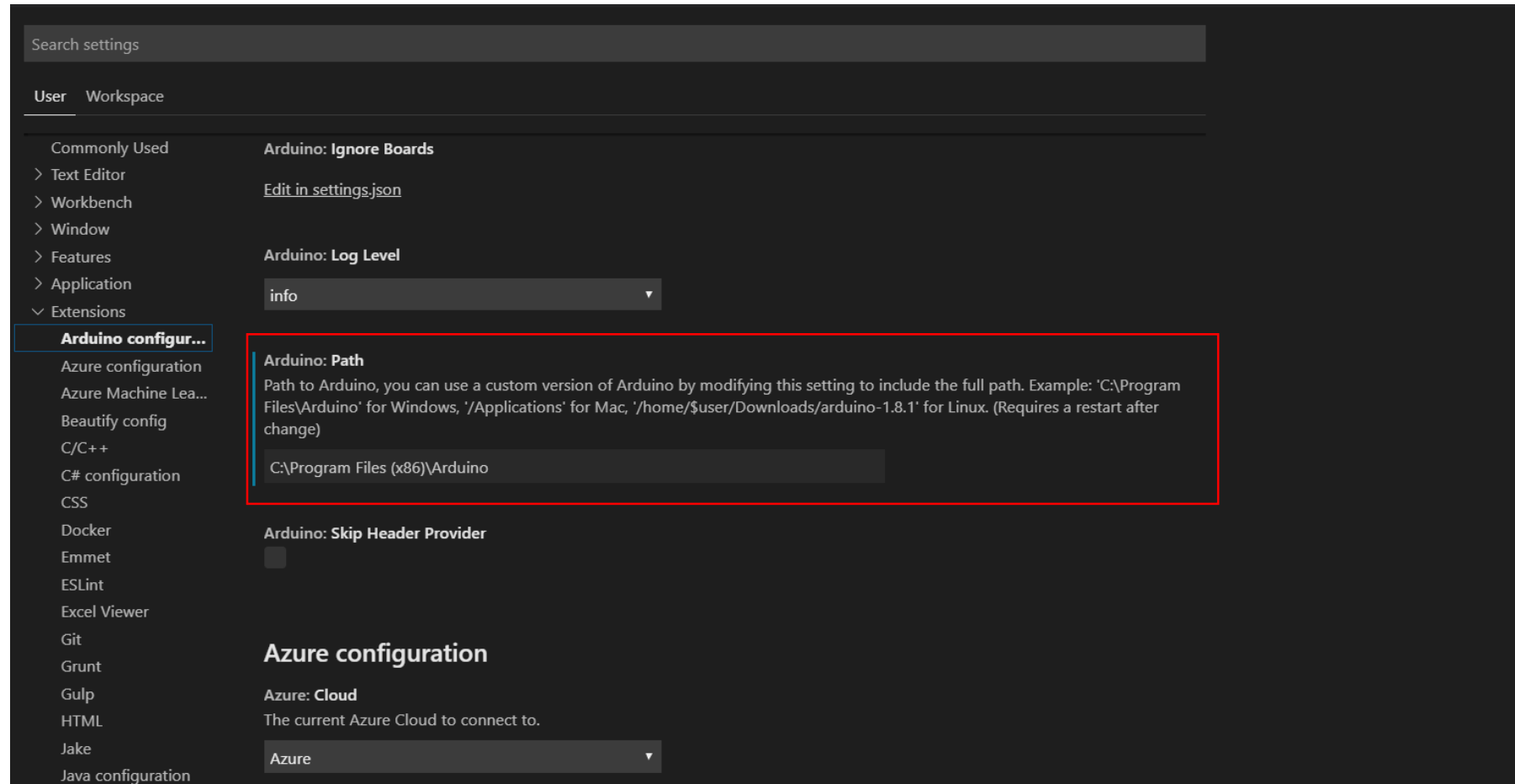
2



Configurar el entorno:

¡Hola Mundo!

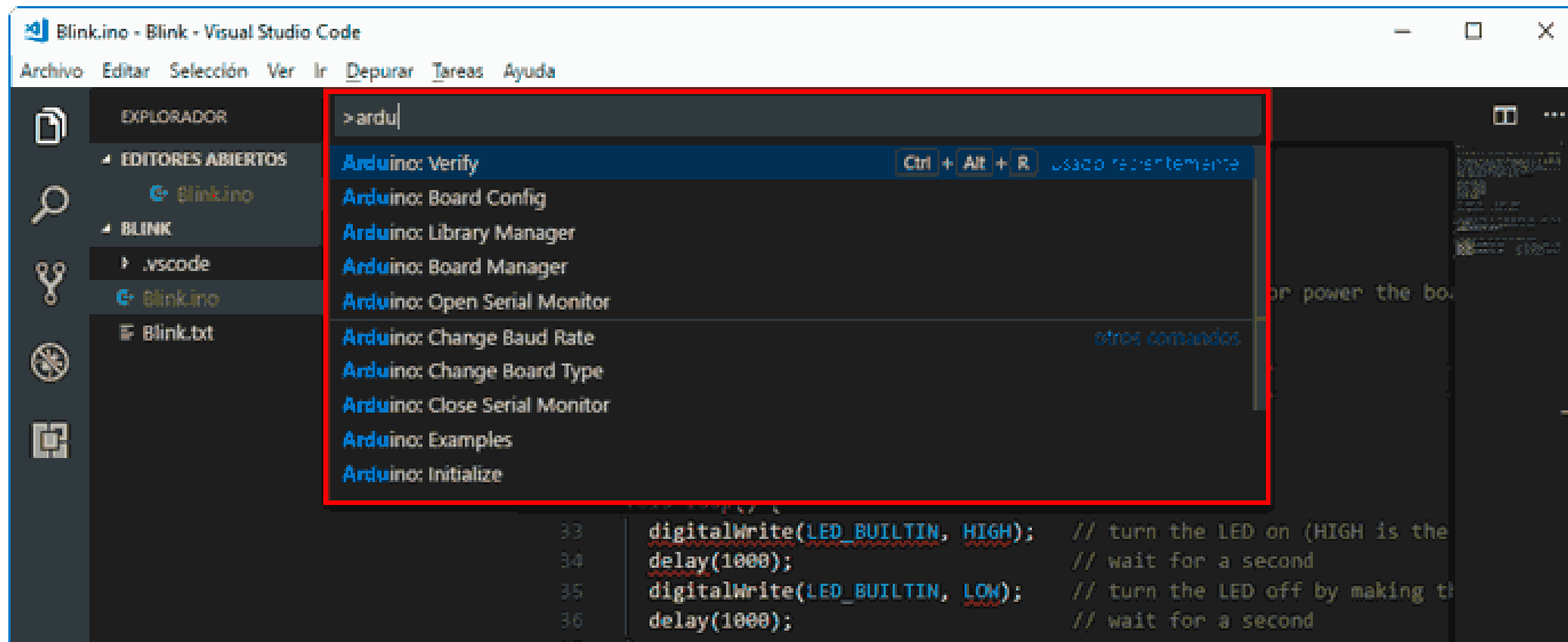
2



Configurar el entorno:

¡Hola Mundo!

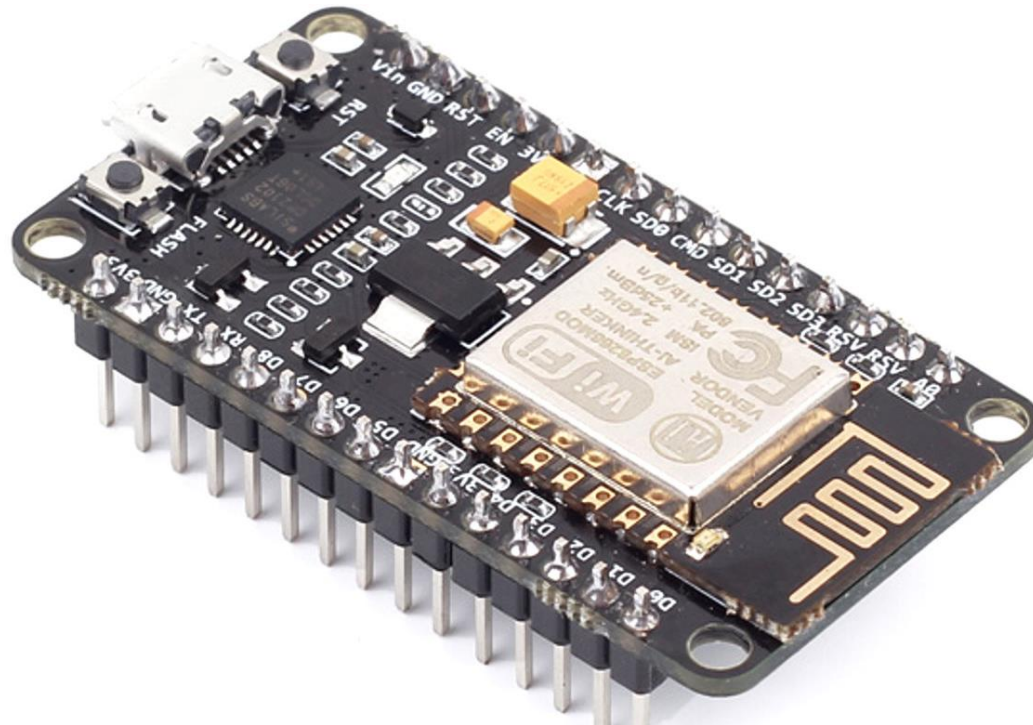
2



2

ESP8266 - NodeMCU

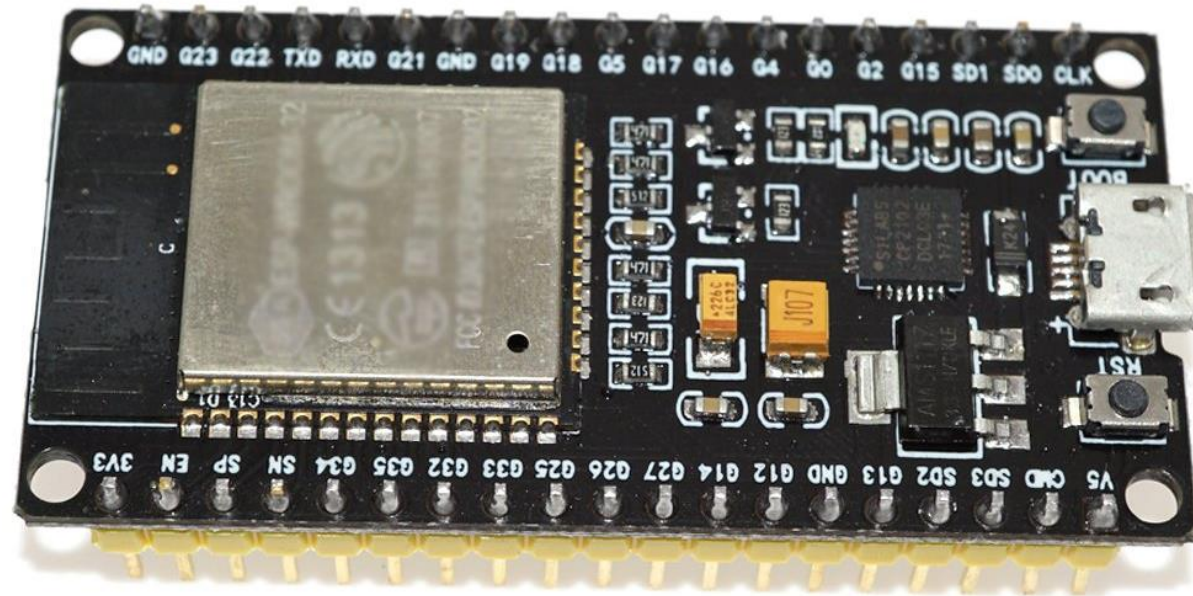
- <https://github.com/esp8266/Arduino>



2

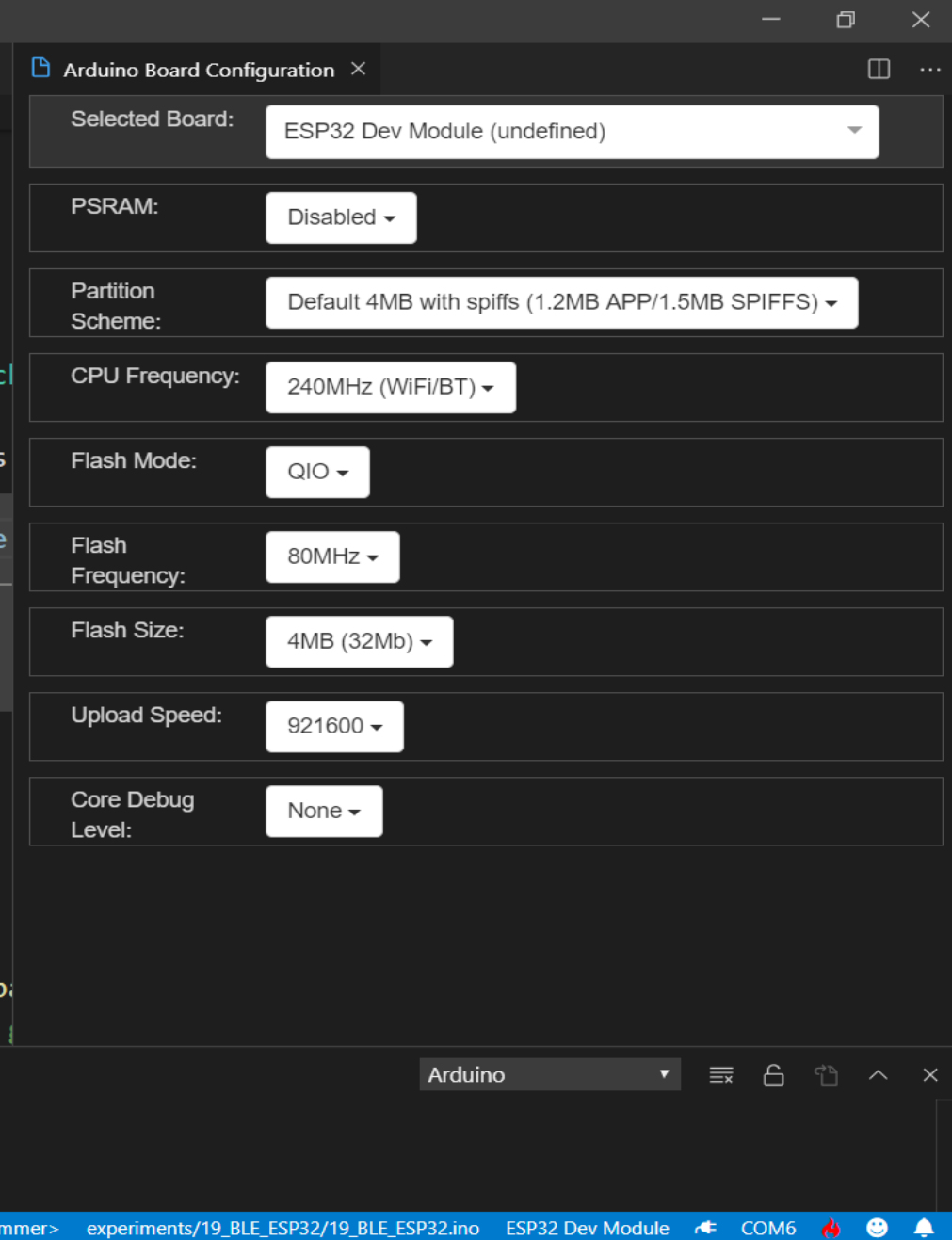
ESP32 - NodeMCU

- <https://github.com/espressif/arduino-esp32>





https://github.com/Gxbx/iot_apps_course



¡Hola Mundo!

2

Conectar Arduino, abrir el IDE y seleccionar la placa y el puerto: