

Sistemas de Control y Servicios

Proyecto Casa IoT: Implementación del ESP-IDF

Implementación del ESP-IDF en el Proyecto de Casa IoT

Introducción

Este informe detalla la implementación del framework de desarrollo ESP-IDF (Espressif IoT Development Framework) en un proyecto de casa inteligente (IoT). Este proyecto es parte del trabajo final para la materia de Sistemas de Control y Servicios de la Tecnicatura Superior en Telecomunicaciones en el ISPC. La maqueta de la casa incluye diversos dispositivos electrónicos y sensores, todos ellos controlados mediante microcontroladores ESP32-C3. Además, se integrará con Google Home Assistant para control por voz.

¿Qué es el ESP-IDF?

El ESP-IDF (Espressif IoT Development Framework) es un entorno de desarrollo integral proporcionado por Espressif Systems para programar y desarrollar aplicaciones en sus microcontroladores de la serie ESP32. El ESP-IDF proporciona una plataforma robusta y rica en funcionalidades diseñada para facilitar la creación de soluciones IoT (Internet de las Cosas). A continuación, se ofrece una explicación detallada de sus componentes y características.

Componentes del ESP-IDF

- SDK (Software Development Kit):
Incluye una colección de bibliotecas y ejemplos de código que permiten interactuar con el hardware del ESP32 y desarrollar aplicaciones complejas.
- Herramientas de Desarrollo:
Compilador Xtensa GCC: Compilador basado en GCC adaptado para el microprocesador Xtensa utilizado en el ESP32.
idf.py: Herramienta de línea de comandos para gestionar proyectos, compilar código y flashear el firmware al dispositivo.
Monitor: Herramienta para visualizar los mensajes de depuración y logs en tiempo real desde el dispositivo.
- Framework de Aplicación:
FreeRTOS: Sistema operativo en tiempo real integrado que permite gestionar tareas concurrentes.
LWIP: Biblioteca de red ligera que facilita la conectividad TCP/IP.
ESP-MQTT: Biblioteca para el protocolo MQTT, que facilita la comunicación entre dispositivos IoT.
ESP-HTTP: Biblioteca para la comunicación HTTP, útil para servicios web y APIs RESTful.

- Componentes de Conectividad:
Wi-Fi: Soporte para múltiples modos (STA, AP, STA+AP) y funcionalidades avanzadas como mesh networking.
Bluetooth: Soporte para Bluetooth clásico y BLE (Bluetooth Low Energy).

- Periféricos y Sensores:

Bibliotecas para interactuar con GPIO, ADC, DAC, I2C, SPI, UART, PWM, entre otros.

Drivers para sensores comunes como DHT11, BMP180, y dispositivos como servomotores y displays.

Características Clave

- Modularidad:
El ESP-IDF es altamente modular, permitiendo al desarrollador seleccionar y configurar solo los componentes necesarios para su aplicación, lo que optimiza el uso de recursos.
- Configuración Sencilla:
Utiliza menuconfig, una interfaz de configuración basada en menús que facilita la personalización de las opciones del sistema y los componentes.
- Soporte para Actualización OTA (Over-the-Air):
Permite actualizar el firmware del dispositivo de forma remota sin necesidad de conexión física, facilitando la gestión y mantenimiento de dispositivos desplegados en el campo.
- Seguridad:
Incluye funcionalidades de seguridad como cifrado de la comunicación, autenticación, y gestión de claves para asegurar la integridad y confidencialidad de los datos.

Beneficios del Uso de ESP-IDF

- Soporte Extensivo de Hardware:
Acceso a una amplia gama de periféricos y funcionalidades del ESP32.
- Desarrollo Modular y Reutilizable:
Posibilidad de desarrollar componentes de software modulares y reutilizables.
- Herramientas Integradas para Depuración y Gestión:
Herramientas que facilitan la depuración, el monitoreo y la gestión de proyectos.
- Actualizaciones y Mejora Continua:

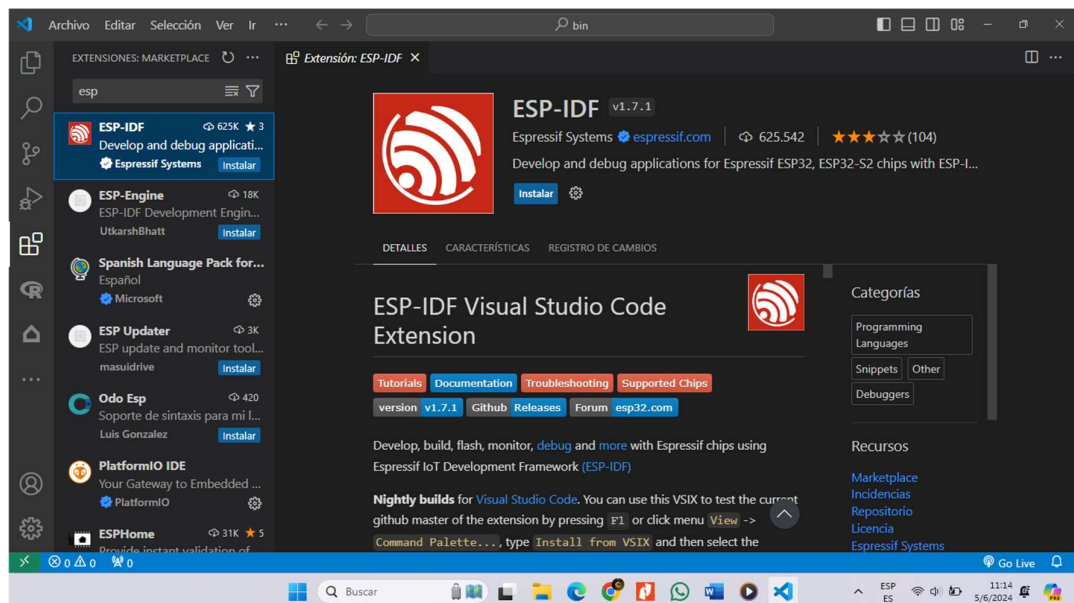
Actualizaciones regulares y mejoras continuas del framework por parte de Espressif.

- Comunidad y Soporte:
Gran comunidad de desarrolladores y soporte activo, facilitando la resolución de problemas y el intercambio de conocimientos.

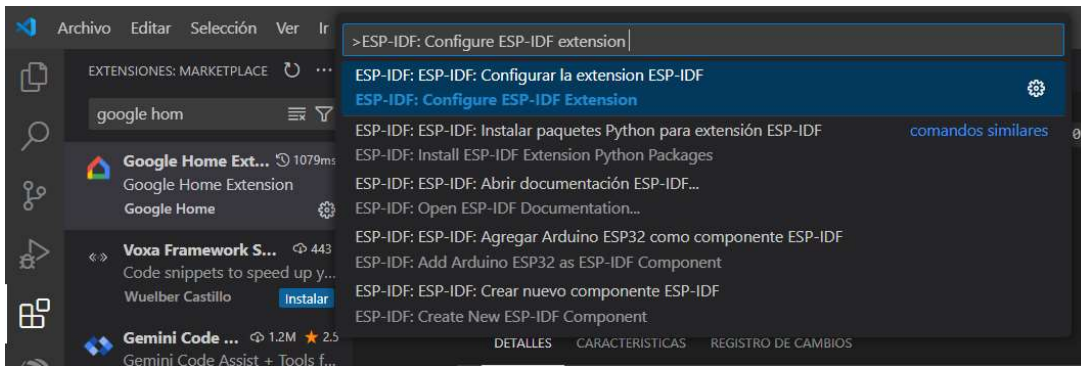
Instalación y Configuración:

Para usar la extensión ESP-IDF (Espressif IoT Development Framework) de Espressif con Visual Studio Code, se deben seguir los siguientes pasos:

1. Tener instalado Visual Studio Code.
2. Instalación de la extensión ESP-IDF
Abrir Visual Studio Code. En el panel izquierdo de la ventana, ir a la sección de extensiones. Buscar "ESP-IDF" y seleccionar la extensión oficial de Espressif. Instalar la extensión.

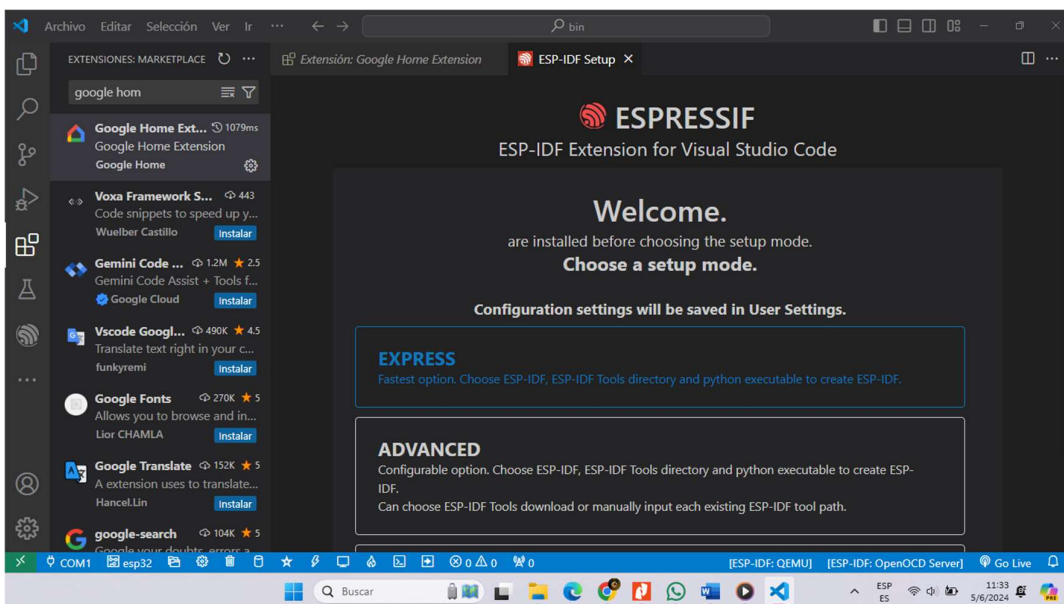


3. Configuración del entorno ESP-IDF:
 - Abrir la paleta de comandos con Ctrl+Shift+P. Escribir ESP-IDF: Configure ESP-IDF extension y seleccionarlo.

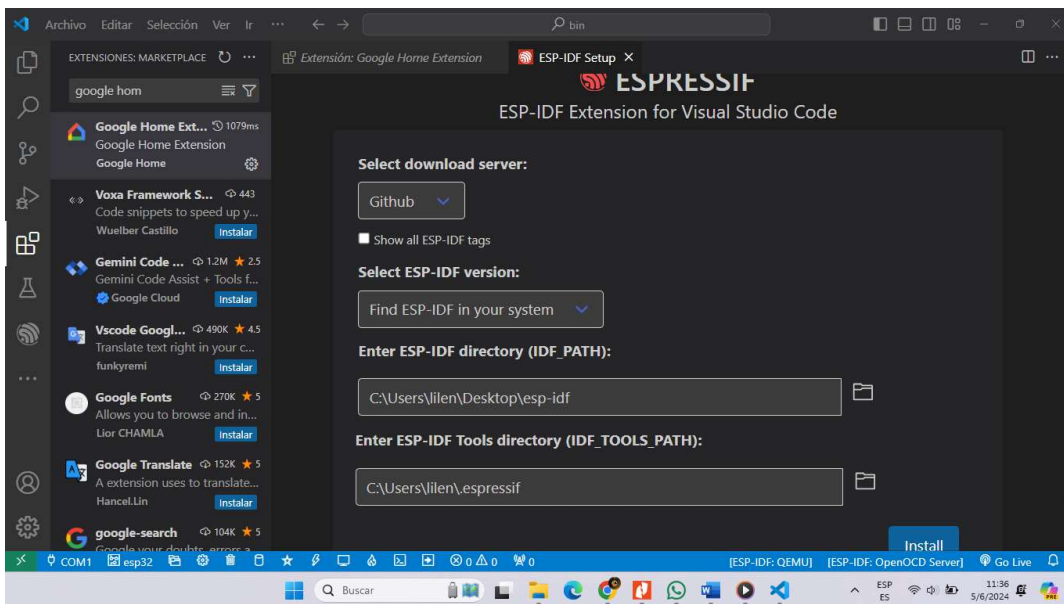


- Seguir las instrucciones para instalar las dependencias necesarias, que incluyen (Python, Git y ESP-IDF)

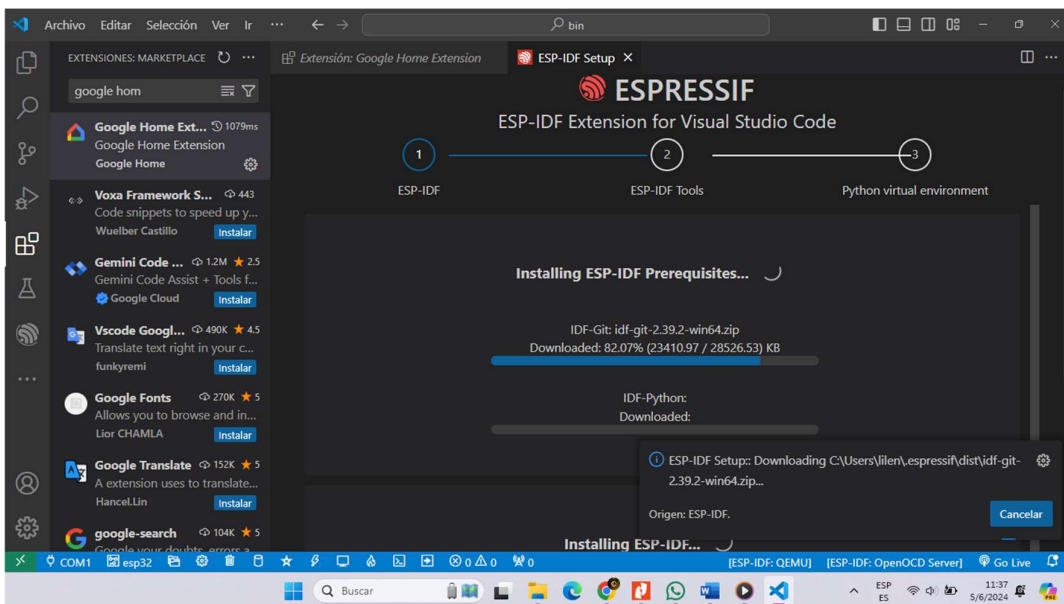
❖ Seleccionar tipo de instalación EXPRESS



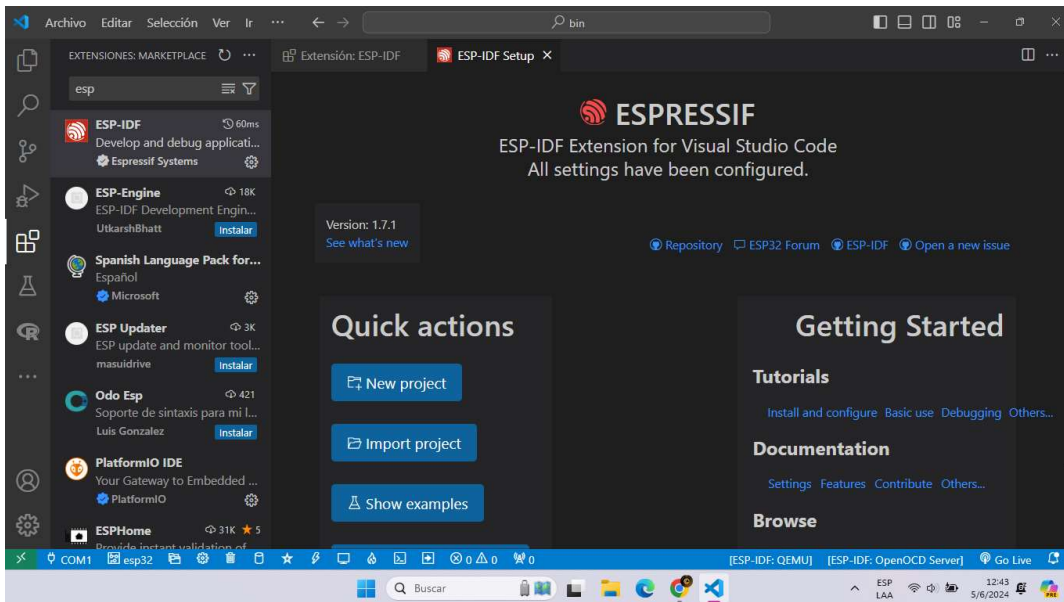
❖ Seleccionar la versión de ESP-IDF que vamos a instalar.



❖ Esperar que termine la instalacion.

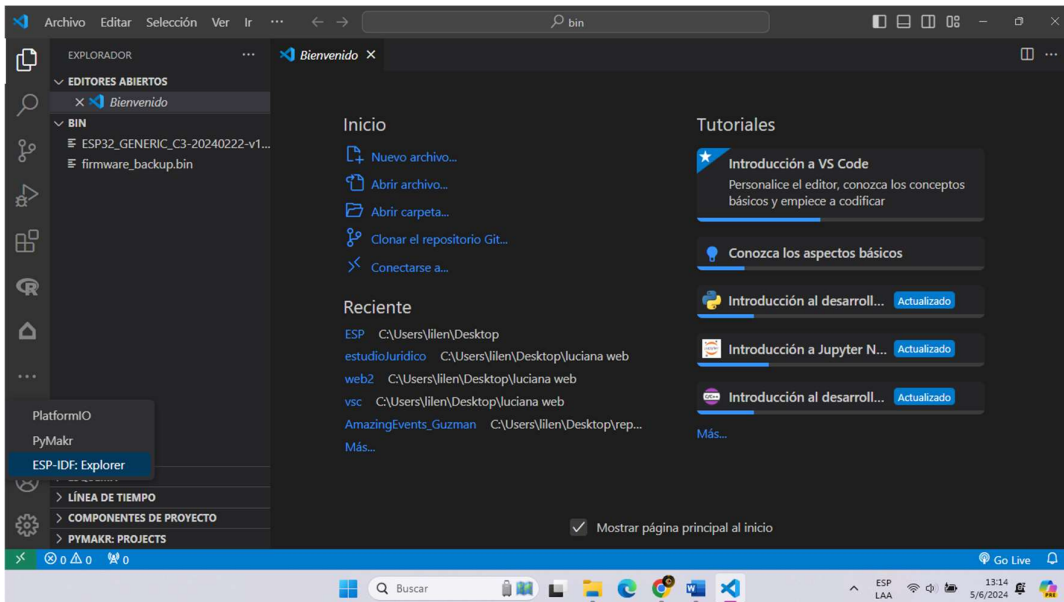


❖ ESP-IDF esta listo para usar

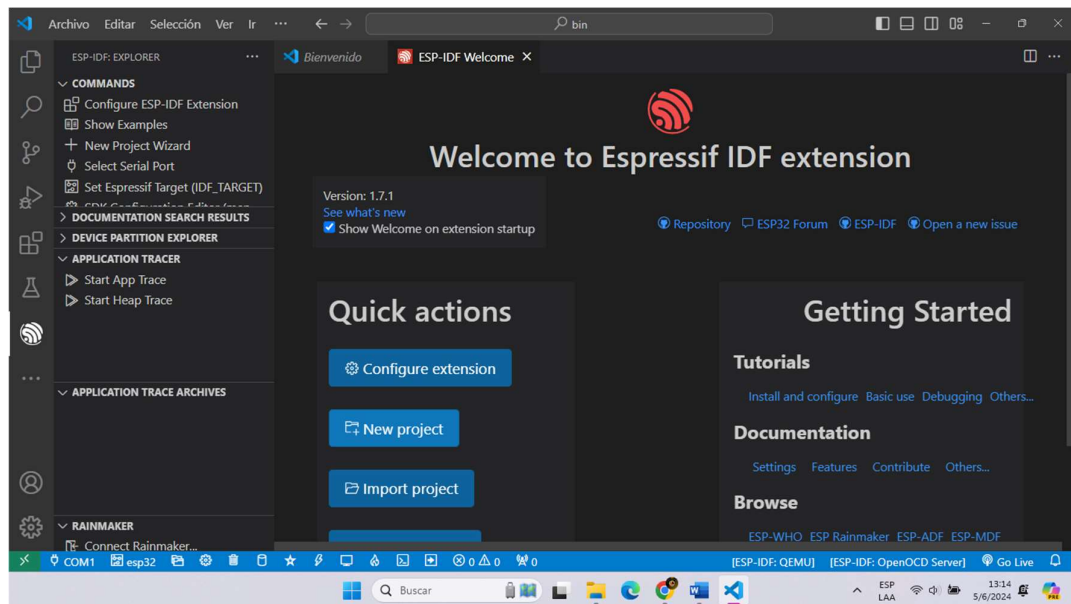


Ejemplo: Crear proyecto con ESP-IDF en VSC

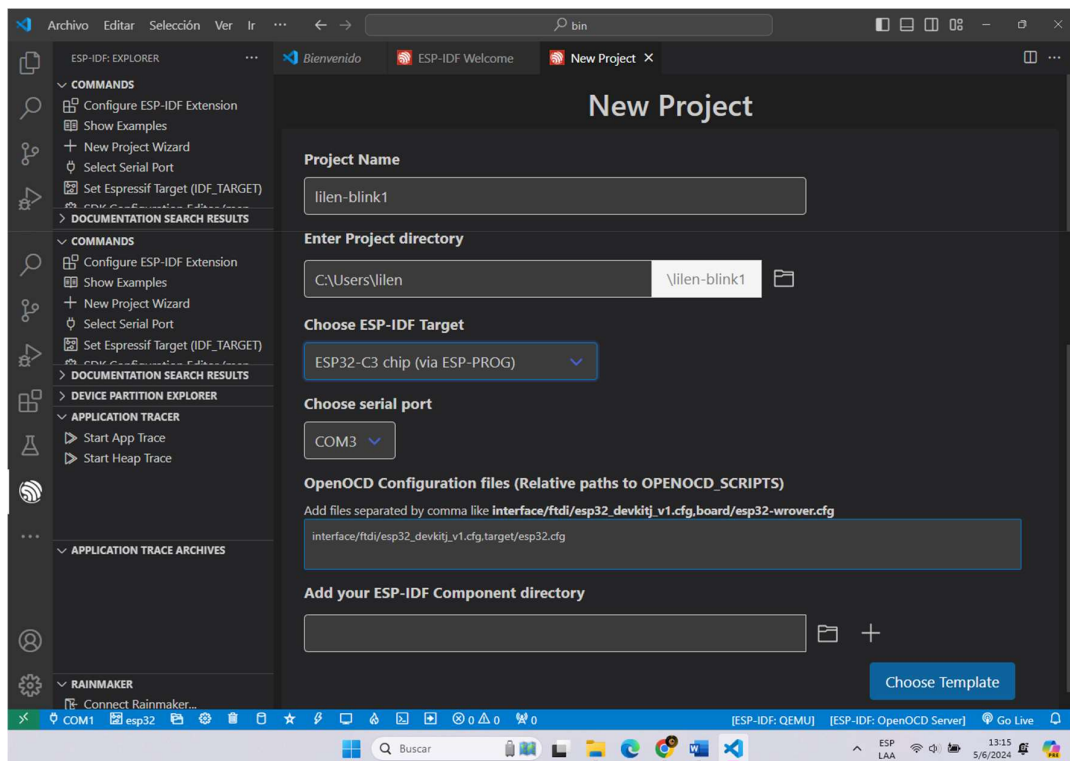
1. En el lado izquierdo de la ventana seleccionar ESP-IDF: Explorer



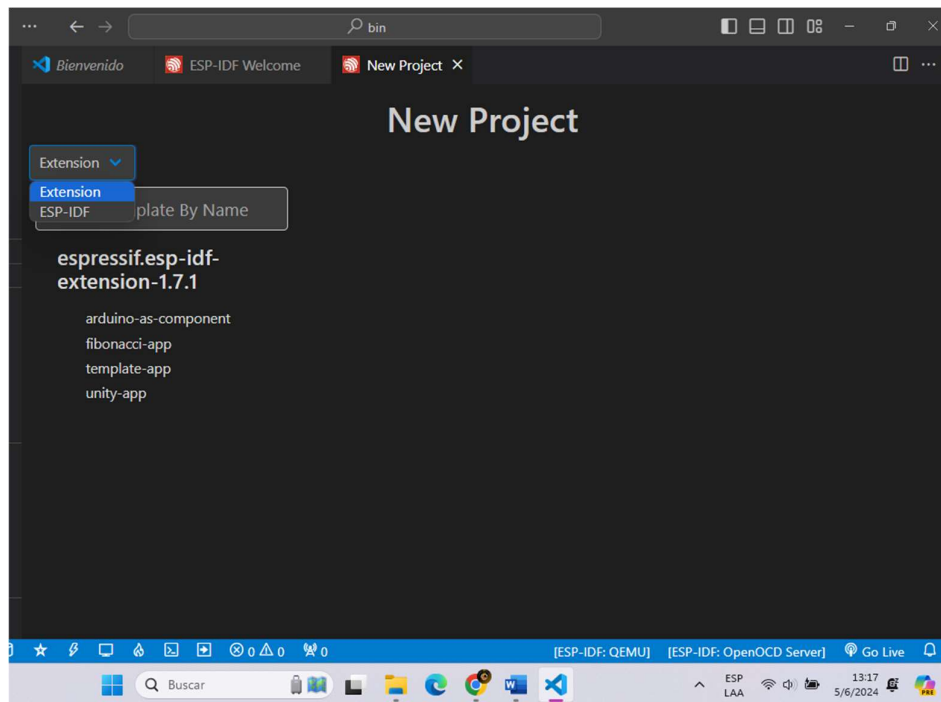
2. Seleccionar New Project



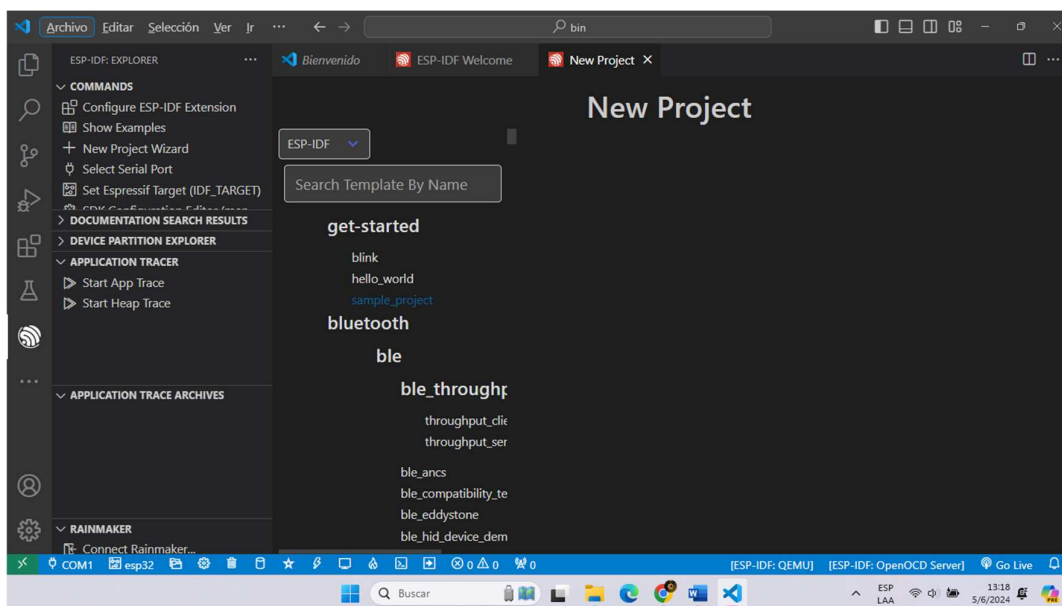
3. Escribir el nombre del nuevo proyecto.
Seleccionar en que carpeta queremos que se guarde.
Seleccionamos el modulo de Esp que estamos usando. En nuestro caso es el Esp-32 C3.
Seleccionamos el puerto usb donde tenemos conectado el Esp-32.
Finalmente seleccionamos Choose Template.

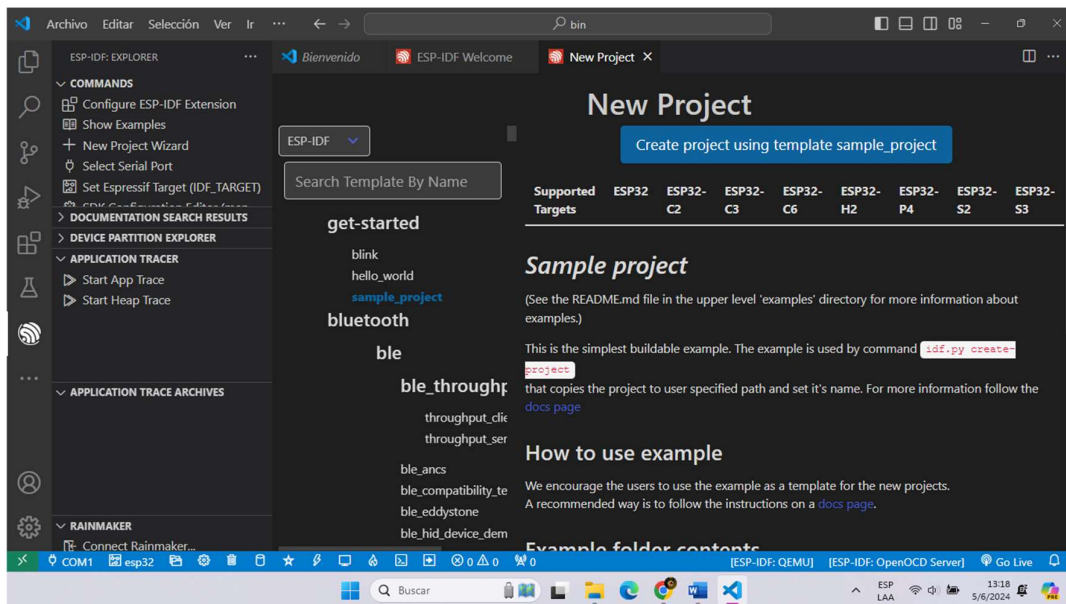


4. Elegimos ESP-IDF

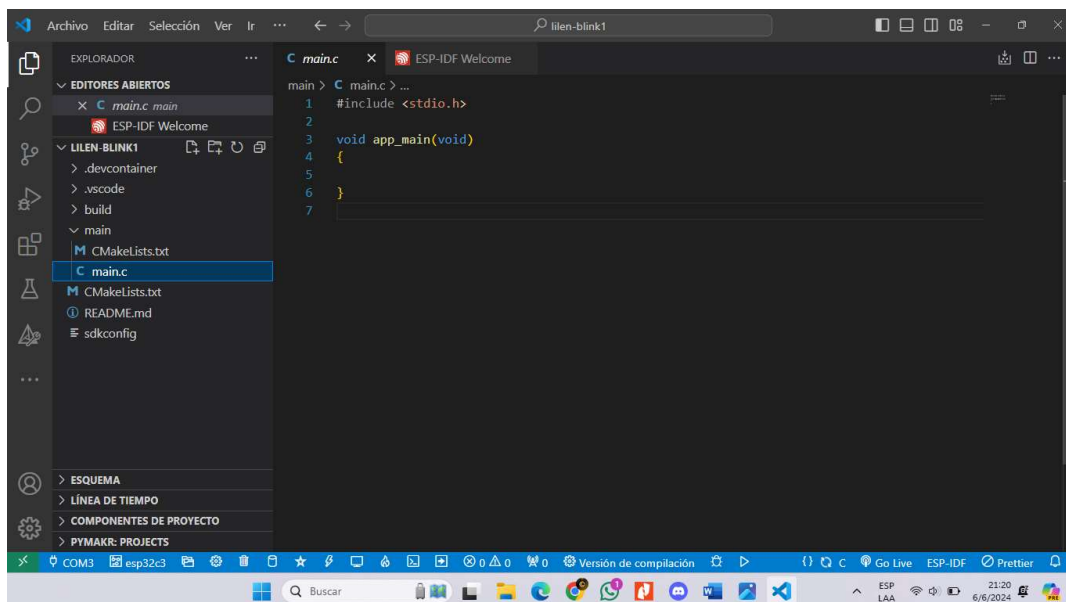


5. Seleccionamos “sample_project” y luego en la nueva ventana seleccionamos “create project using template sample_project” para crear nuestro proyecto desde cero, solo con la estructura del proyecto.





6. Se crea el proyecto vacío con las configuraciones básicas.



Conclusión

El ESP-IDF proporciona una plataforma poderosa y flexible para el desarrollo de aplicaciones IoT con los microcontroladores ESP32. Su arquitectura modular, combinada con un amplio soporte de hardware y funcionalidades avanzadas, permite a los desarrolladores crear soluciones innovadoras y eficientes para una amplia gama de aplicaciones. La integración con herramientas de desarrollo populares y la comunidad activa hacen del ESP-IDF una opción ideal para proyectos de IoT, como la casa domótica que se está desarrollando en este proyecto.