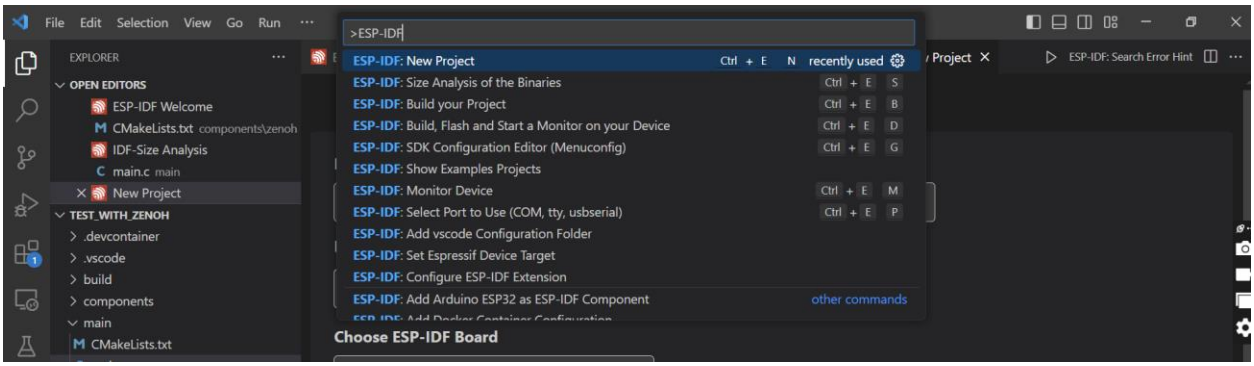
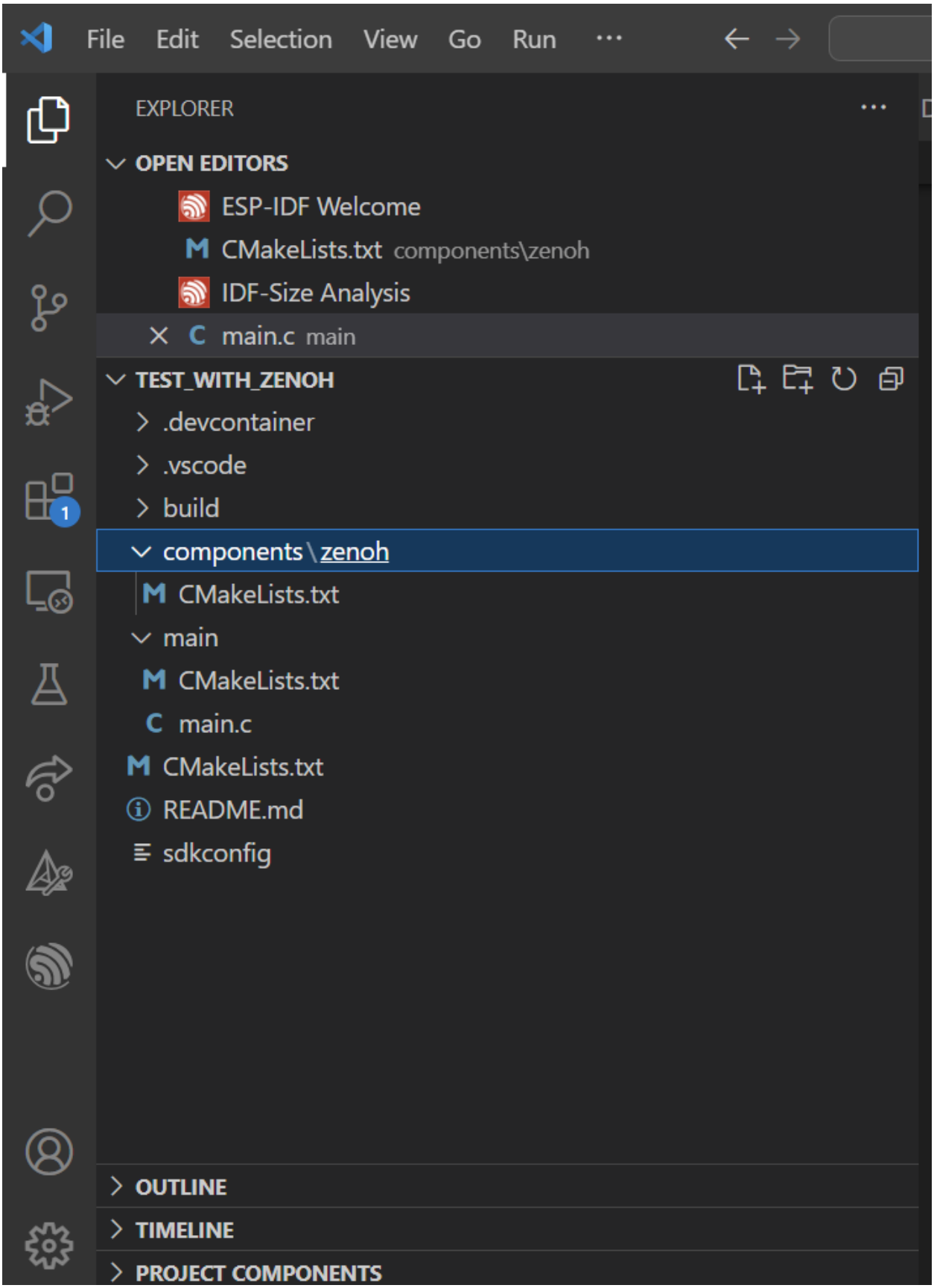
Setup ESP-IDF with Zenoh for ESP32

1- Install VSCode
2- Install 'ESP-IDF' extension from VSCode:
    a. Follow steps from https://github.com/espressif/vscode-esp-idf-extension/blob/master/docs/tutorial/install.md
3- Download the CP210x USB to UART Bridge drivers. This is necessary to be able to flash code on the board (NOTE: the driver depends on the specific ESP32 board that you are using) https://www.silabs.com/developers/usb-to-uart-bridge-vcp-drivers?tab=downloads
4- Download the zenoh-pico repository (clone it from GitHub into your local machine).
    a. https://github.com/eclipse-zenoh/zenoh-pico/
    b. Keep note of the path to the 'zenoh-pico' folder
5- Create a new ESP-IDF project



    a. Choose 'ESP-WROVER-KIT 3.3V' as the board and click on Choose Template.
    b. Choose the 'sample_project' template (empty project) and generate project.

6- Create a new folder called 'components' at the same level as the 'main' folder
    a. Create a subfolder called 'zenoh' inside of components.
    b. The folder structure should look like in the screenshot below.


File
Edit
Selection
View
Go
Run
EXPLORER
OPEN EDITORS
ESP-IDF Welcome
CMakeLists.txt components\zenoh
IDF-Size Analysis
main.c main
TEST_WITH_ZENOH
.devcontainer
.vscode
build
components \ zenoh
CMakeLists.txt
main
CMakeLists.txt
main.c
CMakeLists.txt
README.md
sdkconfig
OUTLINE
TIMELINE
PROJECT COMPONENTS

7- Inside of the components/zenoh folder, create a new file called ‘CMakeLists.txt’
    a. Add the following inside of this CMakeLists.txt file
    b. Make sure you change your ZENOH_SRC_DIR and ZENOH_INC_DIR for your installation location.

```
set(ZENOH_SRC_DIR "../zenoh-pico/src")
set(ZENOH_INC_DIR "../zenoh-pico/include")

idf_component_register(
    SRC_DIRS "${ZENOH_SRC_DIR}/api"
    SRC_DIRS "${ZENOH_SRC_DIR}/collections"
    SRC_DIRS "${ZENOH_SRC_DIR}/link"
    SRC_DIRS "${ZENOH_SRC_DIR}/link/config"
    SRC_DIRS "${ZENOH_SRC_DIR}/link/multicast"
    SRC_DIRS "${ZENOH_SRC_DIR}/link/unicast"
    SRC_DIRS "${ZENOH_SRC_DIR}/net"
    SRC_DIRS "${ZENOH_SRC_DIR}/protocol"
    SRC_DIRS "${ZENOH_SRC_DIR}/protocol/codec"
    SRC_DIRS "${ZENOH_SRC_DIR}/protocol/definitions"
    SRC_DIRS "${ZENOH_SRC_DIR}/session"
    SRC_DIRS "${ZENOH_SRC_DIR}/transport"
    SRC_DIRS "${ZENOH_SRC_DIR}/transport/common"
    SRC_DIRS "${ZENOH_SRC_DIR}/transport/multicast"
    SRC_DIRS "${ZENOH_SRC_DIR}/transport/raweth"
    SRC_DIRS "${ZENOH_SRC_DIR}/system"
    SRC_DIRS "${ZENOH_SRC_DIR}/system/espidf"
    #SRC_DIRS "${ZENOH_SRC_DIR}/system/freertos_plus_tcp"
    SRC_DIRS "${ZENOH_SRC_DIR}/transport/unicast"
    SRC_DIRS "${ZENOH_SRC_DIR}/utils"
    INCLUDE_DIRS "${ZENOH_INC_DIR}"
    REQUIRES esp_driver_uart
)
```

8- Find the include file called ‘platform_common.h’. The file path starting from the base zenoh-pico folder looks like: “..\zenoh-pico\include\zenoh-pico\system\platform_common.h”.
    a. Add the following code on line 27 and save changes

```
#ifndef ZENOH_PICO_SYSTEM_PLATFORM_COMMON_H
#define ZENOH_PICO_SYSTEM_PLATFORM_COMMON_H

#ifndef SPHINX_DOCS
// For some reason sphinx/clang doesn't handle bool types correctly if stdbool.h is in
#include <stdbool.h>
#endif
#include <stdint.h>

#include "zenoh-pico/api/olv_macros.h"
#include "zenoh-pico/config.h"
#include "zenoh-pico/utils/result.h"
#define ZENOH_ESPIDF // Added this here manually, otherwise doesn't work
#if defined(ZENOH_LINUX) || defined(ZENOH_MACOS) || defined(ZENOH_BSD)
#include "zenoh-pico/system/platform/unix.h"
```

9- To test out that it indeed works, go to: https://github.com/eclipse-zenoh/zenoh-pico/tree/main/examples/espidf

   a. Download one of the .c files and put it into the main component or paste the code into the VSCode main.c file.

10- Build project