Espressif IoT Development Framework (ESP-IDF) merupakan native software development framework untuk ESP32. Yang diperlukan untuk menggunakan ESP-IDF yaitu toolchain untuk build dan text editor untuk project C/C++ seperti Eclipse IDE (optional). ESP-IDF memiliki API (software libraries and source code) untuk ESP32 dan scripts untuk mengoperasikan toolchain. ESP-IDF didapat dari repository github (<https://github.com/espressif/esp-idf>) dan diharuskan untuk menginstall Python 3.7 (untuk ESP-IDF V4.0.1) serta Git. Instalasi dapat dilakukan sesuai dengan panduan (<https://docs.espressif.com/projects/esp-idf/en/stable/get-started/index.html>) ESP-IDF menggunakan bahasa pemrograman C melalui proses flashing dan dibutuhkan pemahaman yang cukup mendalam mengenai C agar dapat mengerti dan memrogram menggunakan ESP-IDF. Walaupun demikian, pemrograman ESP32 menggunakan ESP-IDF dapat membawa penggunaan ESP32 menjadi semaksimum mungkin karena ESP-IDF diperuntukkan khusus bagi ESP32 serta bahasa pemrograman yang ringan. Selain itu, juga terdapat banyak dokumentasi (contohnya <https://leanpub.com/kolban-ESP32>) yang mendeskripsikan ESP32 dengan ESP-IDF secara mendalam dan detil sehingga ESP32 dapat digunakan dengan se-efisien mungkin.

Pemrograman dengan ESP-IDF sebaiknya menggunakan IDE C/C++ seperti Eclipse IDE karena pemrograman yang terjadi memiliki level yang cukup rendah karena ESP32 memiliki API untuk masing-masing fungsionalitasnya dan juga berbagai alasan lainnya

Untuk pemula ataupun penggemar mikrokontroller khususnya ESP32 dan tidak memiliki pemahaman bahasa C secara mendalam, sebaiknya menggunakan Arduino IDE untuk memrogram ESP32. Selain itu, jika program dibutuhkan juga untuk diimplementasikan pada board selain ESP32, sebaiknya menggunaan Arduino IDE karena bahasa yang universal. Arduino IDE juga dapat menggunakan (include) ESP-IDF pada untuk memrogram ESP32.

Berikut screenshot hasil uji coba saya menggunakan ESP-IDF menggunakan program Hello World yang sudah disediakan. Menggunakan Eclipse IDE memang jauh lebih mudah dibandingkan menggunakan text editor biasa. Namun ada beberapa hal yang harus di setting terlebih dahulu seperti PATH, dll. Selain itu penggunaan Eclipse IDE juga dapat mempercepat waktu build karena parallel build yang dimiliki oleh Eclipse IDE. Plugin ESP-IDF yang tersedia di Eclipse IDE juga memiliki fitur Application Size Analysis (provides a way to analyze the static memory footprint of your application) tetapi saya tidak berhasil menjalankannya karena terdapat exception Java NullPointerException.