



**Facultad Regional Tucumán**  
**Departamento Electrónica**

**Técnicas Digitales II**

**Actividad de Formación Práctica 1**

**Tema: Entorno de Desarrollo Integrado STM32CubeIDE,  
programación de micro controladores.**

**Profesores:**

Ing. Rubén Darío Mansilla

**ATTP:**

Ing. Lucas Abdala

**Grupo 5**

**Integrante:**

- Angelillo Alessandro
- Plaate Iván
- Torres Juan
- Hualampa Leonel Javier

*Año: 2025*

- 1) Elabore un tutorial en soporte digital, que explique cómo se aborda el uso del **STM32CubeIDE**, que contenga lo que se indica a continuación:

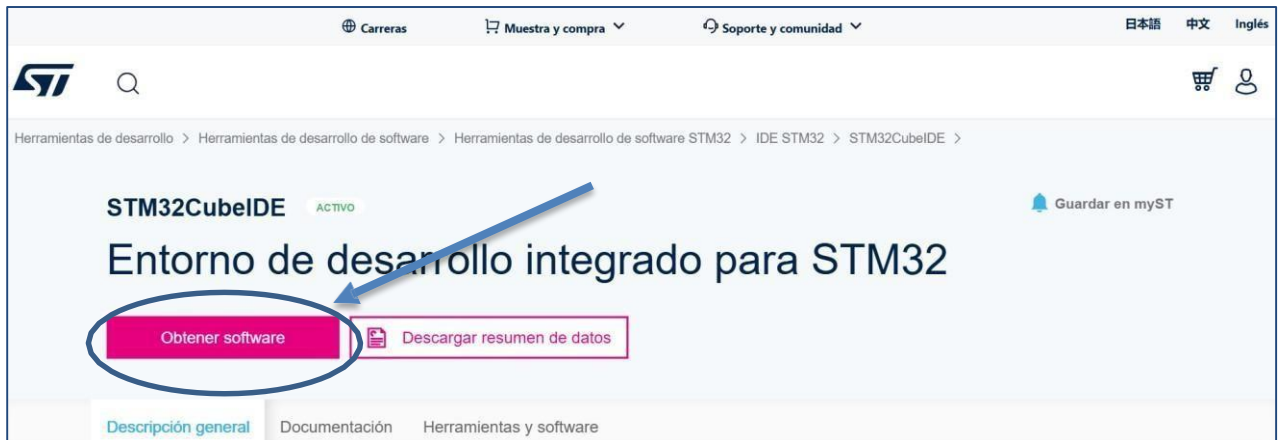
## Instalacion

### • COMO DESCARGAR

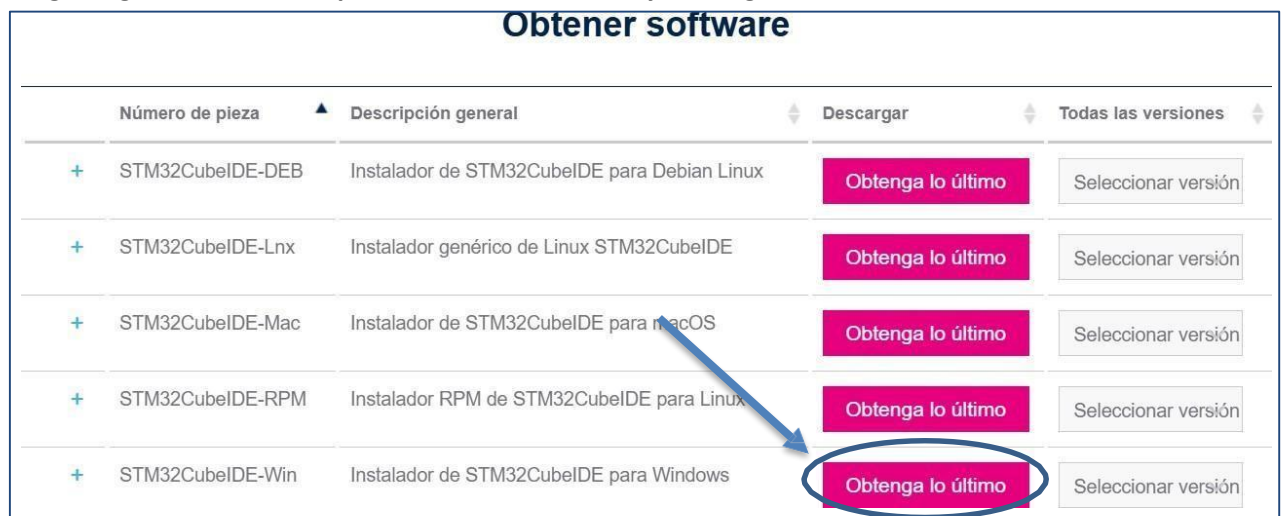
Primer paso descargamos el software **STM32CubeIDE** haciendo click en el siguiente link que le dejamos a continuación:

[Descargar stm32cubeide](#)

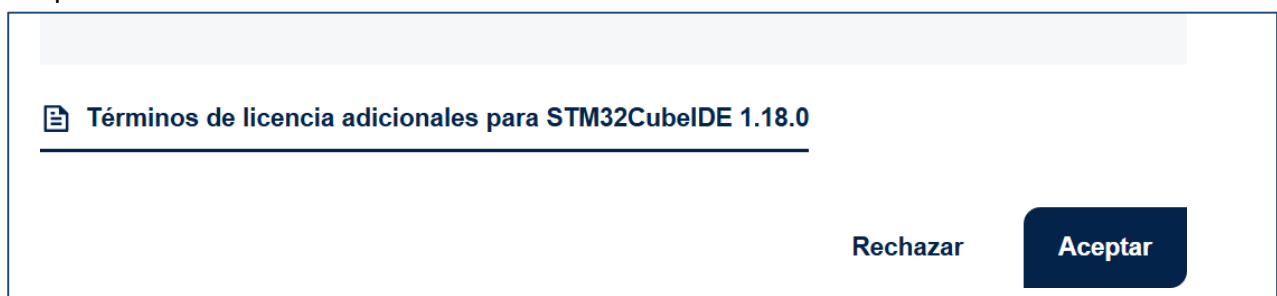
Nos dirigimos a la siguiente casilla y hacemos click



Luego elegimos el sistema operativo estamos usando y descargamos la última versión.

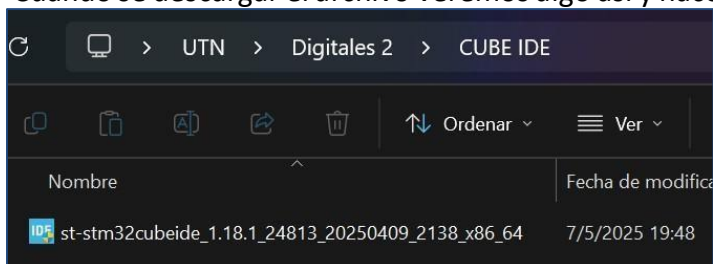


Aceptamos el acuerdo de licencia



Una vez ahí nos pedirá si queremos (**Descargar como invitado**) o (**Crear una cuenta en MyST**)

Cuando se descargar el archivo veremos algo así y hacemos doble click sobre el

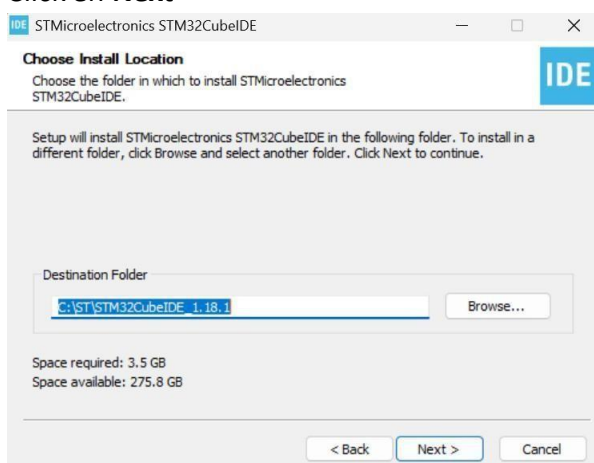


## • COMO INSTALAR

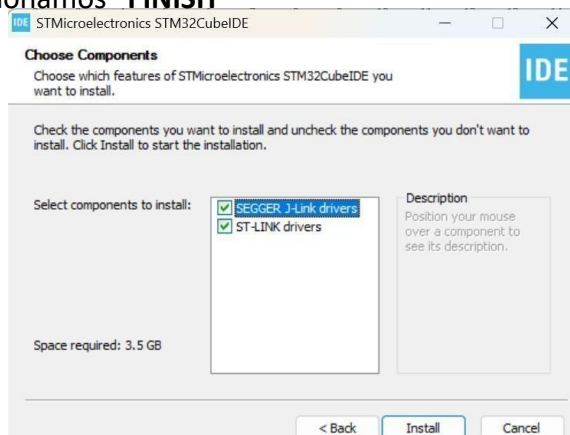
Se nos abre la siguiente ventana y hacemos click en “Next”



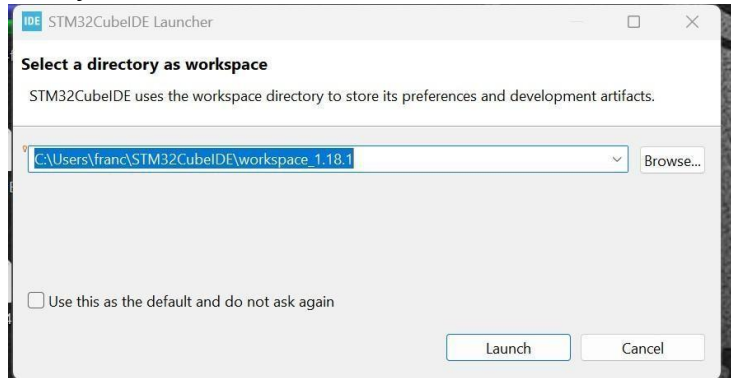
Click en Next



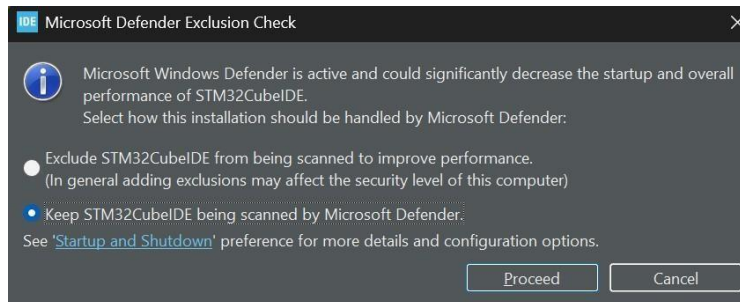
Dejamos marcados los componentes que deseamos que se instalen. Esperamos y presionamos “FINISH”



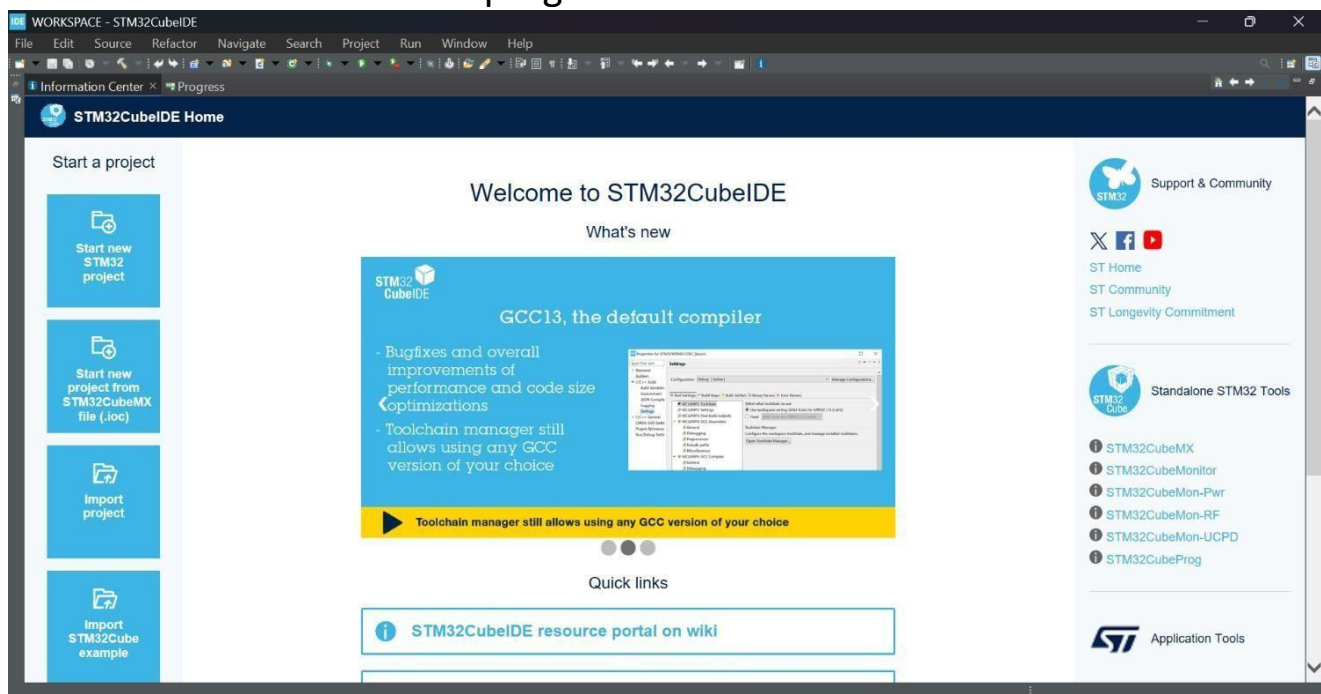
- La Primera vez que ejecutamos el programa nos pedirá una Dirección para el espacio de trabajo Presionamos en “Launch”



Nos pregunta si queremos darle permiso Windows defender

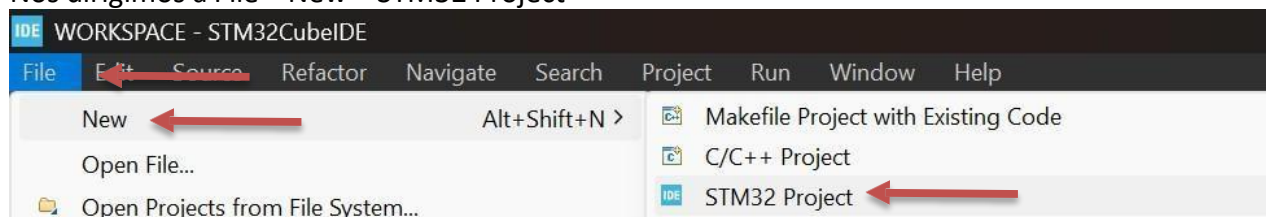


- Primera vista del programa

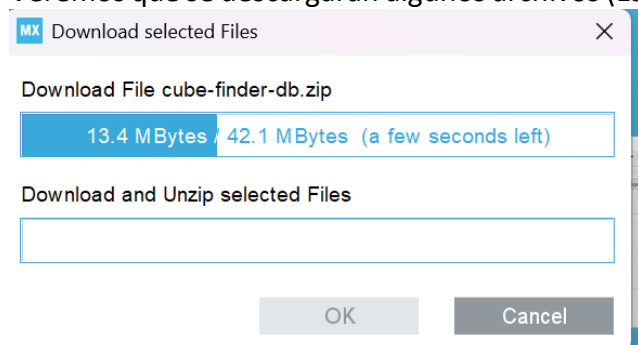


## Crear Un Nuevo Proyecto

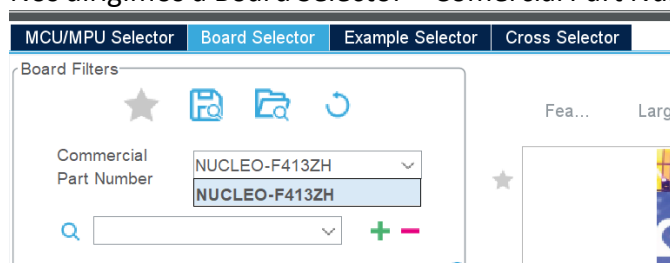
Nos dirigimos a File – New – STM32 Project



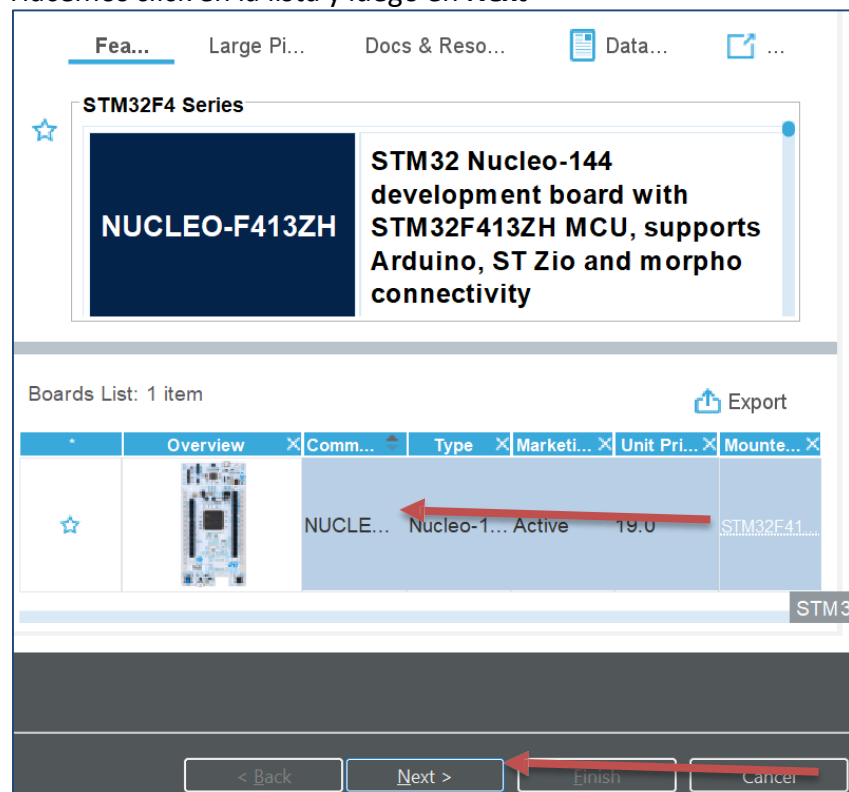
Veremos que se descargaran algunos archivos (Esperamos)



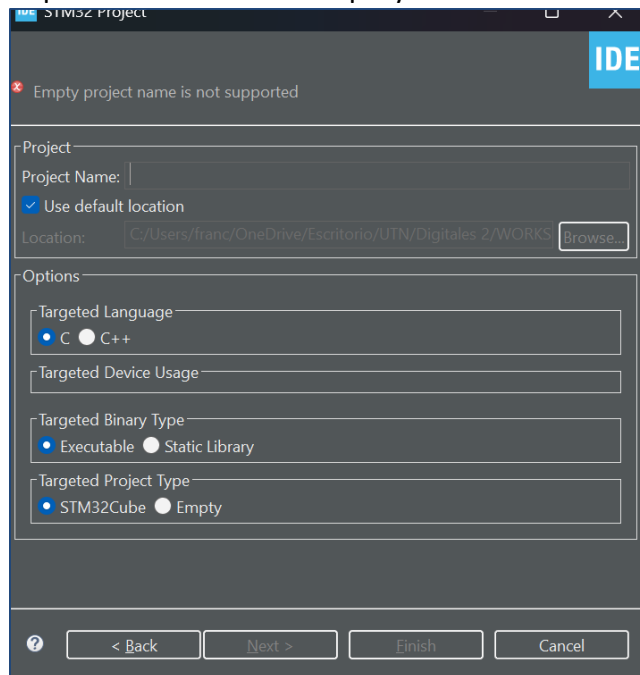
Nos dirigimos a Board Selector – Commercial Part Number – NUCLEO – NUMERO DE LA PLACA



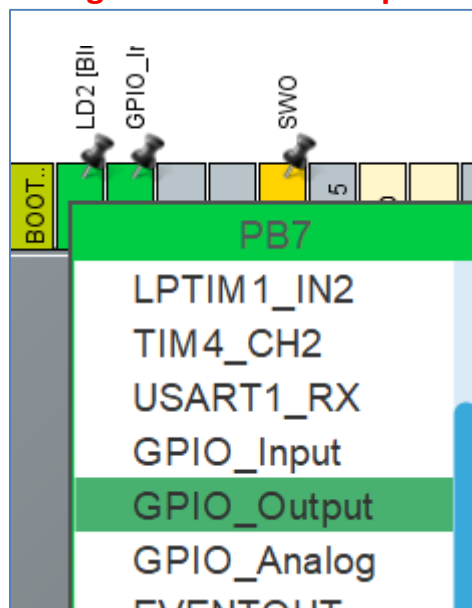
Hacemos click en la lista y luego en **Next**



Le ponemos un nombre al proyecto

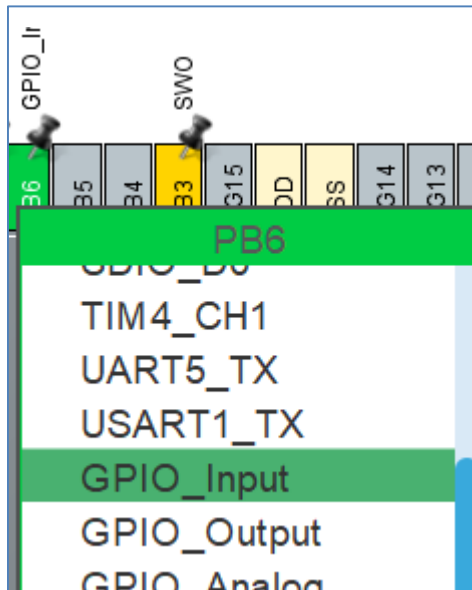


## Configuraciones basicas para el uso del modulo GPIO



Hacemos click sobre el PIN **PB7** y definimos al LED2 como salida **GPIO\_OUTPUT**

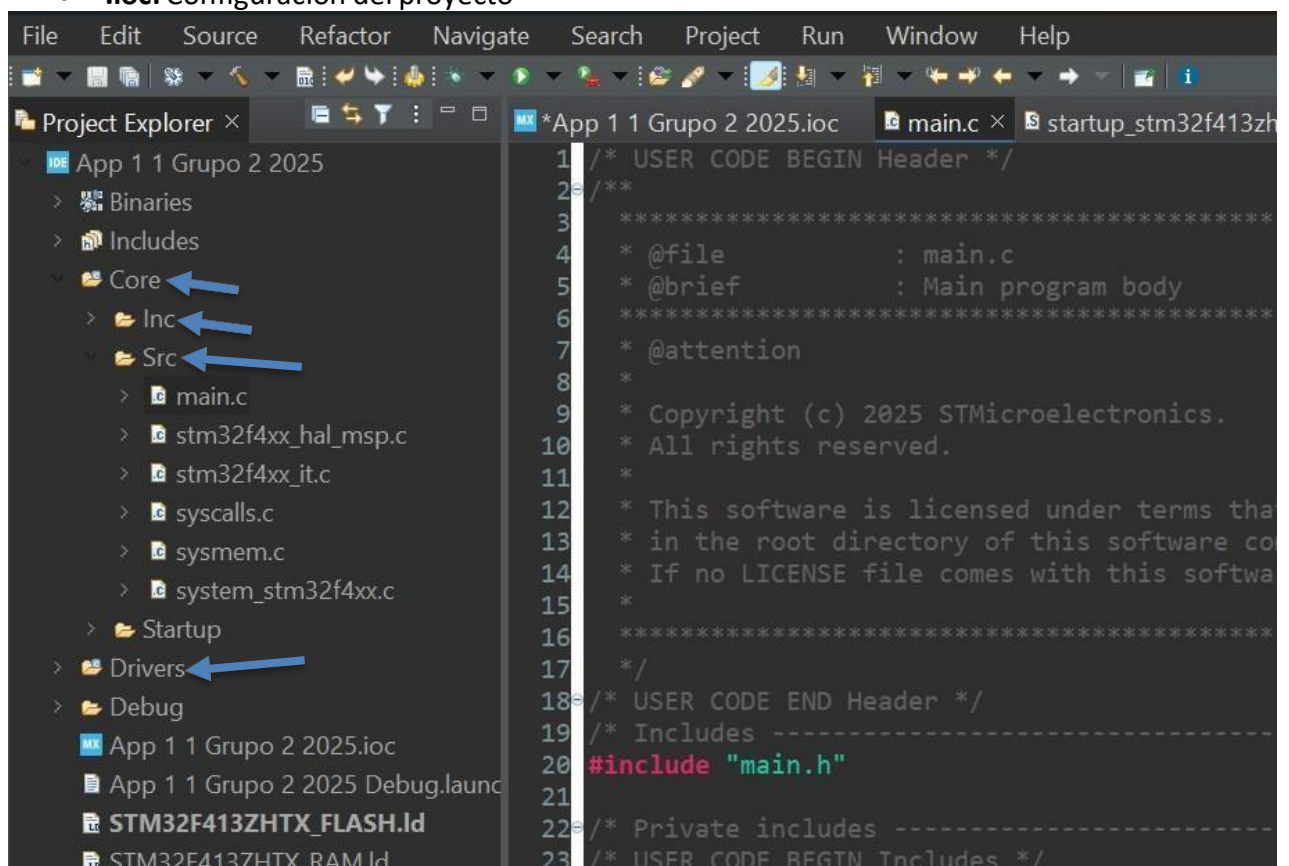
Hacemos click sobre el PIN **PB6** Definimos como pulsador de usuario **GPIO\_Input**



### Estructura del arbol de archivos

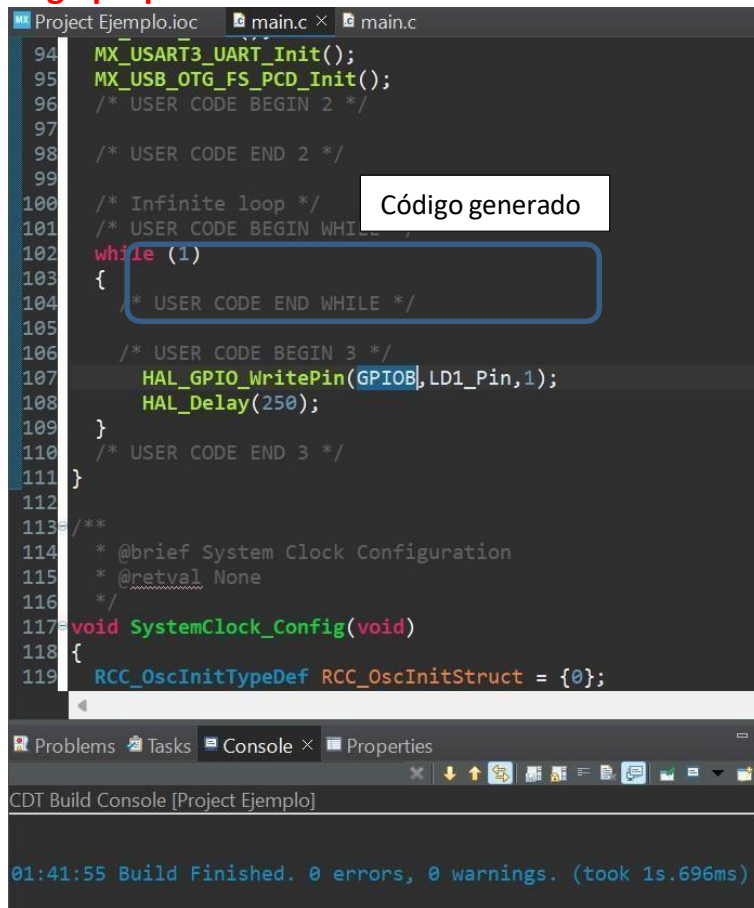
Al crear un proyecto podemos ver:

- **Core:** Código fuente
- **Inc:** Archivo cabecera (contiene **main.h**)
- **Src:** Archivo fuente (contiene **main.c**)
- **Drivers:** Bibliotecas por defecto
- **.ioc:** Configuración del proyecto



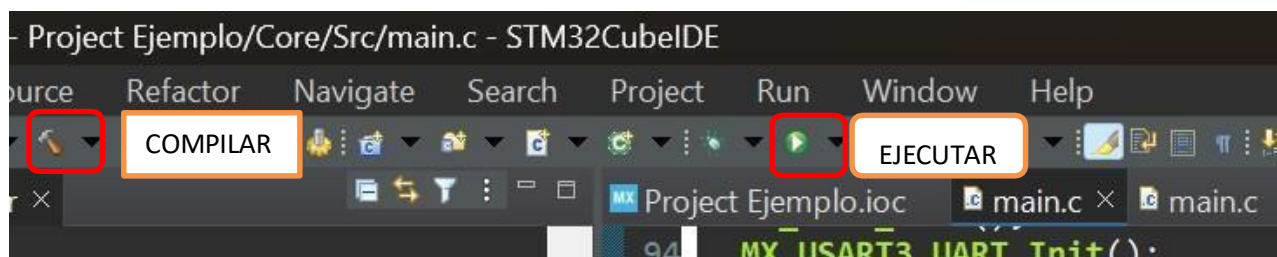


**Desarrollo de un proyecto que utilice uno de los leds integrados en la placa y lo haga parpadear cada 250 ms.**



```
94  MX_USART3_UART_Init();
95  MX_USB_OTG_FS_PCD_Init();
96  /* USER CODE BEGIN 2 */
97
98  /* USER CODE END 2 */
99
100 /* Infinite loop */
101 /* USER CODE BEGIN WHILE */
102 while (1)
103 {
104     /* USER CODE END WHILE */
105
106     /* USER CODE BEGIN 3 */
107     HAL_GPIO_WritePin(GPIOD, LD1_Pin,1);
108     HAL_Delay(250);
109 }
110 /* USER CODE END 3 */
111 }
112
113 /**
114  * @brief System Clock Configuration
115  * @retval None
116  */
117 void SystemClock_Config(void)
118 {
119     RCC_OscInitTypeDef RCC_OscInitStruct = {0};
```

Para compilar



Para cargar el programa en la placa debemos

- Conectar la placa por **USB**
- Presionar **Ejecutar** o **Debug**
- Veremos que se carga el programa (En la placa veremos una luz que parpadea)
- Luego se encenderá el **led** cada 250ms.



## Desarrollo de Aplicaciones: (Hacer click en el hipervínculo para ver el código)

**App 1\_1:** Desarrolle una aplicación que encienda y apague de manera secuencial los tres leds onboard de la placa de desarrollo. La secuencia debe encender 200 ms y apagar 200 ms cada led comenzando por el LED1 (Green), continuando con el LED2 (Blue) y luego el LED3 (Red) para volver a iniciar con el LED1. La aplicación debe ser de carácter general, de manera que pueda extenderse a una cantidad mayor de leds con mínimas modificaciones, por este motivo se sugiere que use un vector para el manejo de los Leds

### [App 1\\_1](#)

**App 1\_2:** Desarrolle una aplicación que utilice el pulsador onboard de la placa de desarrollo para alternar entre dos secuencias diferentes. La aplicación inicia con la secuencia de la App 1.1 y, cuando se presione el pulsador, dicha secuencia debe invertirse y continuar, de manera que, cada vez que se presione el pulsador la secuencia actual se invierta. La aplicación debe ser de carácter general, por lo que aplica la misma recomendación para el punto anterior.

### [APP1\\_2](#)

**App 1\_3:** Desarrolle una aplicación que utilice el pulsador onboard de la placa de desarrollo para alternar entre cuatro secuencias diferentes. La app iniciará con la secuencia 1 hasta que se presione el pulsador y pase a la secuencia 2, luego de presionar de nuevo el pulsador pasará a la secuencia 3 y así, sucesivamente, hasta la secuencia 4 para volver a comenzar con la secuencia 1. Descripción de las secuencias:

- Secuencia 1: Idem App 1.1 con una alternancia de 150 ms.
- Secuencia 2: hace parpadear los tres leds simultáneamente con una alternancia de 300 ms.
- Secuencia 3: hace parpadear el LED1 con una alternancia de 100 ms, el LED2 con una alternancia de 300 ms y el led3 con una alternancia de 600 ms.
- Secuencia 4: Hace parpadear simultáneamente LED1 y LED3, mientras que LED2 lo hará de manera inversa, con una alternancia de 150 ms. Mientras LED1 y LED3 están encendidos, LED2 estará apagado y luego a la inversa.

### [App1\\_3](#)

**App 1\_4:** Desarrolle una aplicación que haga parpadear simultáneamente los 3 leds onboard de la placa y que use el pulsador onboard para cambiar la frecuencia de parpadeo de manera secuencial entre 4 frecuencias predefinidas por los siguientes tiempos de alternancia:

- Tiempo 1: 100 ms.
- Tiempo 2: 250 ms.
- Tiempo 3: 500 ms.
- Tiempo 4: 1000 ms.

La aplicación inicia con la alternancia determinada por el tiempo 1, al presionar el pulsador pasa al tiempo 2 y así sucesivamente hasta llegar al tiempo 4, para reiniciar con el tiempo 1.

### [App1\\_4](#)