Práctica 4

## Objetivo:

Implementar un módulo de software para trabajar con la UART.

## Punto 1

Implementar un módulo de software en un archivos fuente **API\_uart.c** con su correspondiente archivo de cabecera **API\_uart.h** y ubicarlos en el proyecto dentro de las carpetas /drivers/API/src y /drivers/API/inc, respectivamente.

En API\_uart.h se deben ubicar los prototipos de las funciones públicas.

bool\_t uartInit();

void uartSendString(uint8\_t \*pstring);

En API\_uart.c se deben ubicar los prototipos de las funciones privadas y la implementación de todas las funciones de módulo, privadas y públicas.

## Punto 2

Consideraciones para la implementación:

1. uartInit() debe realizar toda la inicialización de la UART. Adicionalmente, debe imprimir por la terminal serie los parámetros de configuración.  
     
   La función devuelve:
   * true: si la inicialización es exitosa.
   * false: si la inicialización no es exitosa.
2. uartSendString(uint8\_t \*pstring) recibe un puntero a un string que se desea enviar por la UART y debe utilizar la función de la HAL HAL\_UART\_Transmit(...) para transmitir el sting

## Punto 3

Sobre el ejercicio de la práctica 3 que hacía parpadear forma periódica los tres leds de la placa NUCLEO-F429ZI, implementar un programa que utilice el módulo de uart para informar los períodos de parpadeo que cambian cuando se presiona la tecla USER.

Inicialmente:

* LED1: 100 ms.
* LED2: 500 ms.
* LED3: 1000 ms.

Cada vez que se presione el pulsador los tiempos de los LEDs deben alternar entre la configuración inicial y la alternativa:

* LED1: 500 ms.
* LED2: 500 ms.
* LED3: 500 ms.