

## Taller 2 – Python (Qt Designer - Raspberry)

### A. GUI (Qt Designer)

1. Implementar una calculadora de operaciones aritméticas (suma, resta, multiplicación, cociente y residuo de una división) y trigonométricas básicas (seno, coseno, tangente, cotangente, secante y cosecante) utilizando objetos *Push button* (operaciones), dos objetos *Edit Text* para ingresar los valores y un objeto *Static Text* para mostrar el resultado.
2. Implementar un *Push button* para graficar funciones trigonométricas (seno, coseno, tangente, cotangente, secante y cosecante) en un objeto tipo *axes*. Para realizar la selección de las funciones se debe utilizar un objeto tipo *Pop Up Menú*. En dos objetos tipo *Edit Text* debe ingresar el valor mínimo y máximo de la función a graficar.
3. Utilizando un objeto *pop-up menú*, escoger entre robot cartesiano, esférico o cilíndrico. Indicar cuantas articulaciones tiene y de qué tipo son en un *Static Text*, además de visualizar una imagen en un objeto *axes* con el DIAGRAMA CINEMÁTICO para cada robot.
4. A través de tres *slider* modificar la resistencia, la capacitancia y el voltaje para graficar de manera dinámica (tiempo real) la carga y descarga de un condensador en un solo objeto tipo *axes*.
5. Cargar una imagen que sea seleccionada por el usuario a través de la búsqueda de dicha imagen en los documentos del PC a través de un objeto tipo *Push button* y posteriormente identificar los contornos de la imagen seleccionada.

### B. GUI con GPIOs (Qt Designer)

1. **Sliders y servomotores:** Utilizar un objeto tipo *slider* para mover dos servomotores de 0 a 180°. La selección del servomotor se debe realizar utilizando un objeto tipo *Edit Text*. En un objeto tipo *Static Text* debemos visualizar el ángulo actual en grados del servomotor a medida que vamos moviendo el objeto tipo *slider*.
2. **Escritura de puertos:** Utilizar dos *Push button* para encender y apagar dos diodos LED, un *Push button* para cada led y por cada pulsación se debe realizar una acción (encender o apagar), en la interfaz gráfica el color de los *Push Button* debe ser el mismo que el LED encendido. Además, utilizar dos *slider* para aumentar o disminuir el brillo de otros dos diodos LED diferentes a los anteriores.
3. **Comunicación I2C:** Realizar la lectura de cualquier sensor por I2C en un *Static Text* y mostrar dicha lectura durante un tiempo definido por el usuario a través de un objeto tipo *Edit Text* al presionar un objeto tipo *Push button*.

4. **Lectura de puertos digitales:** Realizar la lectura de un pin digital de la Raspberry, donde el color y el texto del *Static Text* cambia en función del estado, de la siguiente manera:

4.1. Estado alto -> color rojo y texto “alto”

4.2. Estado bajo -> color azul y texto “bajo”

5. **Motores paso a paso:** Realizar el giro de un motor paso a paso una cantidad de vueltas (ej: 0.5, 1, 2.5, etc.) determinada por el usuario a través de un objeto tipo *Edit Text* y un objeto *Push button* para accionar el movimiento del actuador.

Cada programa debe tener el nombre de los integrantes y mostrar el logo de la universidad en el archivo .ui. Por ejemplo:

