

TECNOLÓGICO  
NACIONAL DE MÉXICO



## **INSTITUTO TECNOLÓGICO DE TLAXIACO**

**Nombre del alumno:** Rael Gabriel Bautista

**Tema:** Componentes y Librerías

**Facilitador:** Prof. José Alfredo Román Cruz

**Materia:** Tópicos Avanzados de Programación

**Practica 5:** Diseño de Control Personalizado

**Carrera:** Ing. En sistemas computacionales

**Semestre:** Cuarto semestre

## Objetivo

El objetivo de esta práctica es que el estudiante realice la implementación de un componente personalizado para su implementación como control en Visual Studio, con el fin de que comprenda la importancia que tienen las librerías en la creación de componentes.

## Descripción

1. Visualizar videos en YouTube para realizar la práctica del control personalizado
2. Leer el siguiente artículo para implementar el control personalizado.  
<https://rjcodeadvance.com/toggle-button-custom-controls-winform-c/>
3. Diseñar una biblioteca DLL para la implementación del control personalizado.
4. Declarar campos y propiedades heredadas de un control nativo de Visual Studio
5. Compilar la biblioteca para poder reutilizarla en nuevos proyectos de Visual Studio.
6. Realizar una aplicación de escritorio utilizando la nueva aplicación de escritorio.

## Materiales

- Computadora
- Acceso a internet
- Visual Studio

Lista de Figuras	
Figura 1	Figura 1.6
Figura 1.1	Figura 1.7
Figura 1.2	Figura 1.8
Figura 1.3	Figura 1.9
Figura 1.4	Figura 2
Figura 1.5	Figura 2.1

1.- Se crea un Proyecto en Visual Studio, posteriormente agregamos una carpeta para guardar las clases de los componentes personalizados.

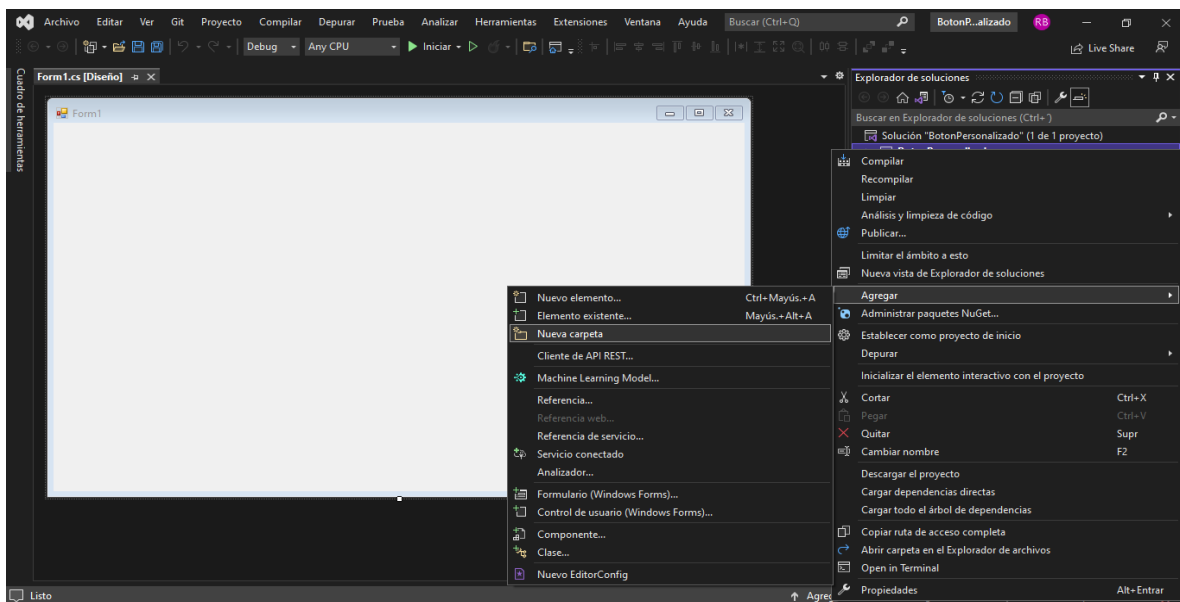


Figura 1

2.- Creamos la clase “Boton1Personalizado” implementando todas las librerías que se requieren para la creación del componente.

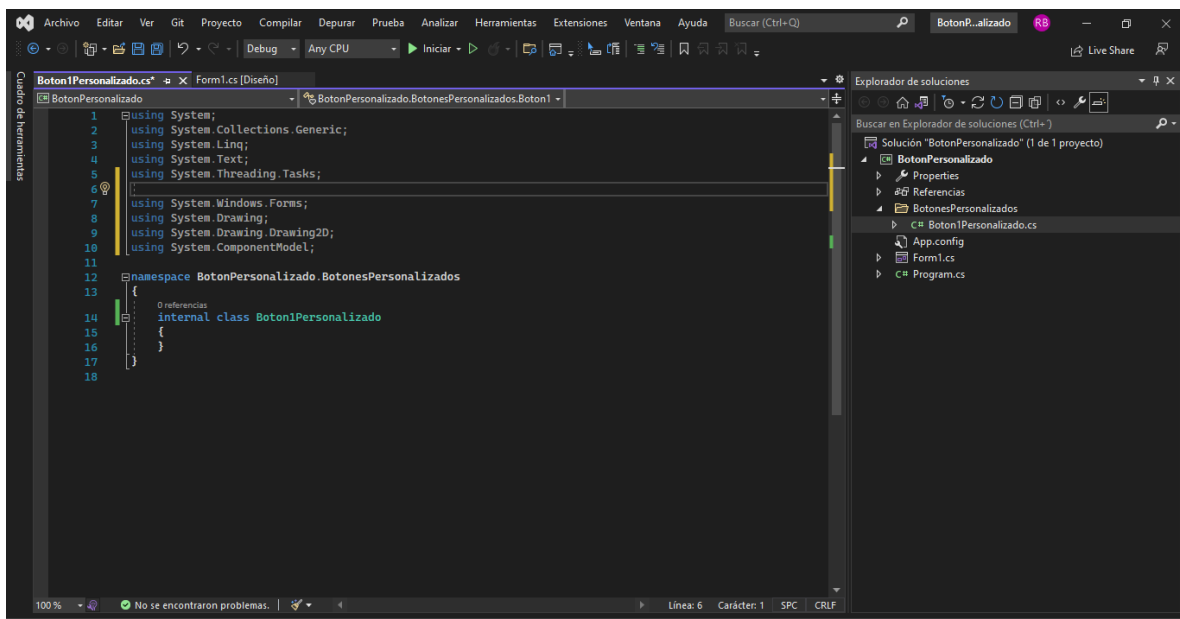


Figura 1.1

3.- Heredamos del control CheckBox para que nuestro botón tenga todas las mismas propiedades.

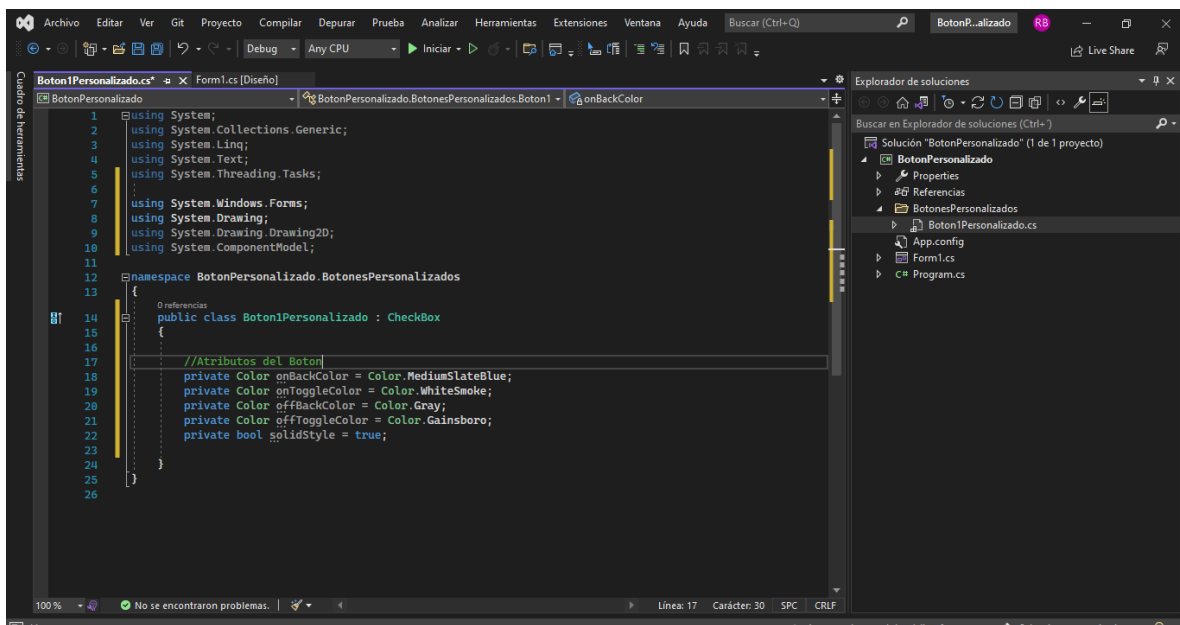


Figura 1.2

4. Declaramos campos para la apariencia del control “Boton1Personalizado” y asignar sus valores predeterminados, por ejemplo, color de fondo y palanca en estado encendido y apagado.

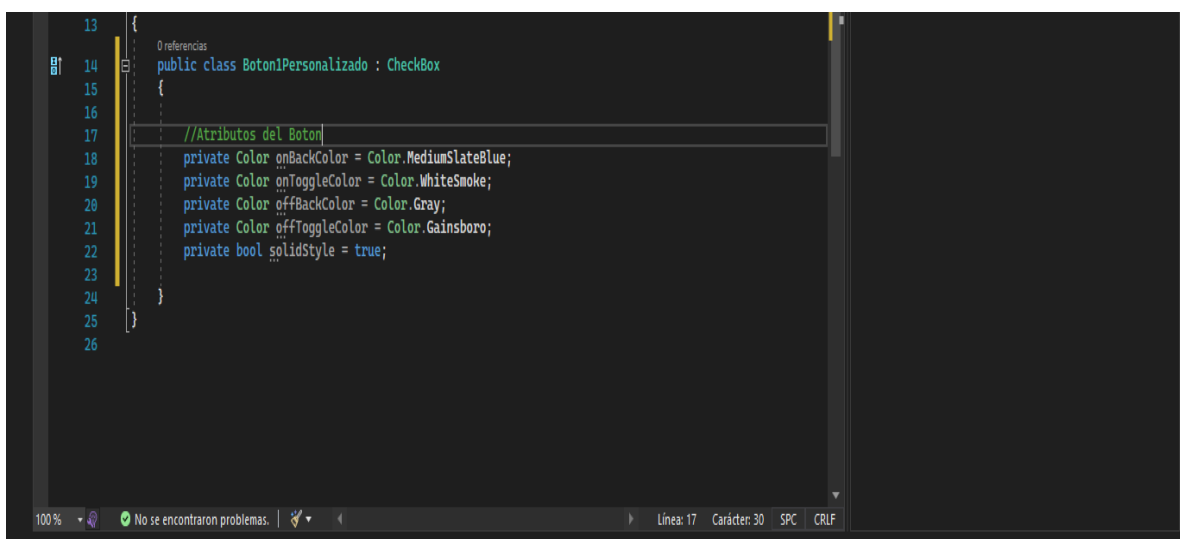


Figura 1.3

5. Generamos propiedades para exponer los campos anteriores declarados.

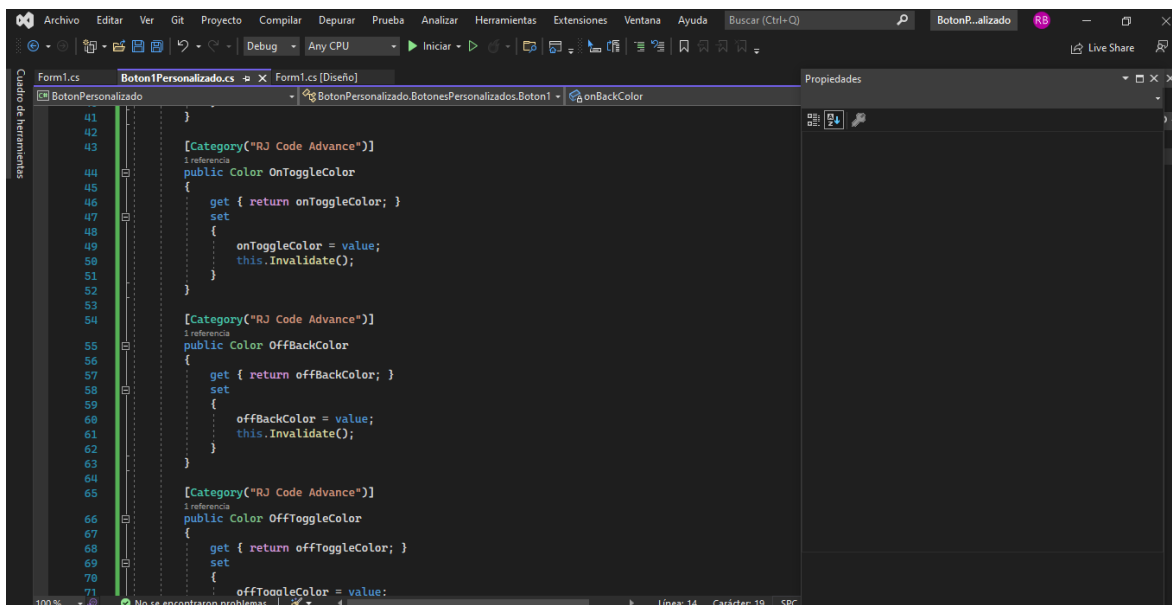


Figura 1.4

7.- En el constructor puedes inicializar las propiedades que deseas, por ejemplo, la propiedad `MinimumSize`, propio del control `CheckBox`.

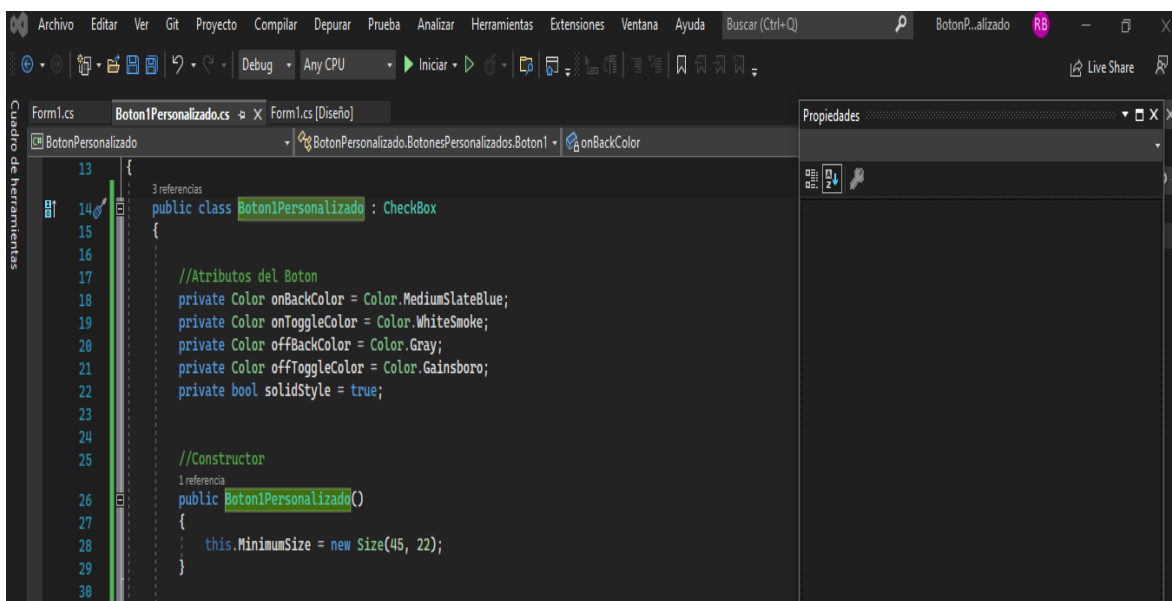


Figura 1.5

## 7.- Declarar método crear ruta de figura

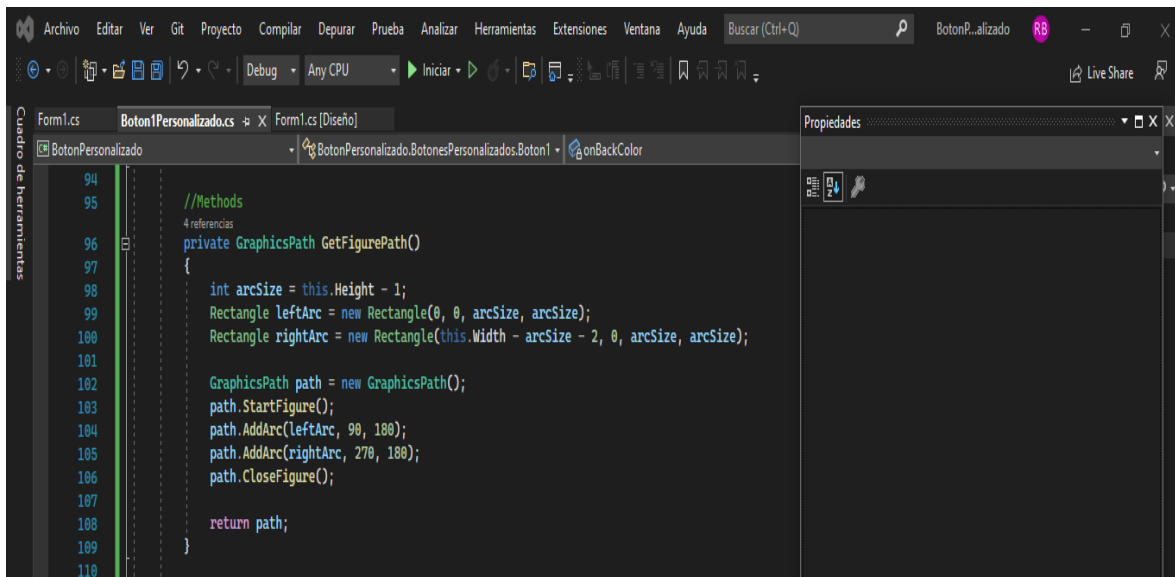


Figura 1.6

8.- Anular el evento Paint del control para volver a dibujar la apariencia del control a nuestra manera. En este caso, básicamente se dibuja el control en estado encendido o apagado.

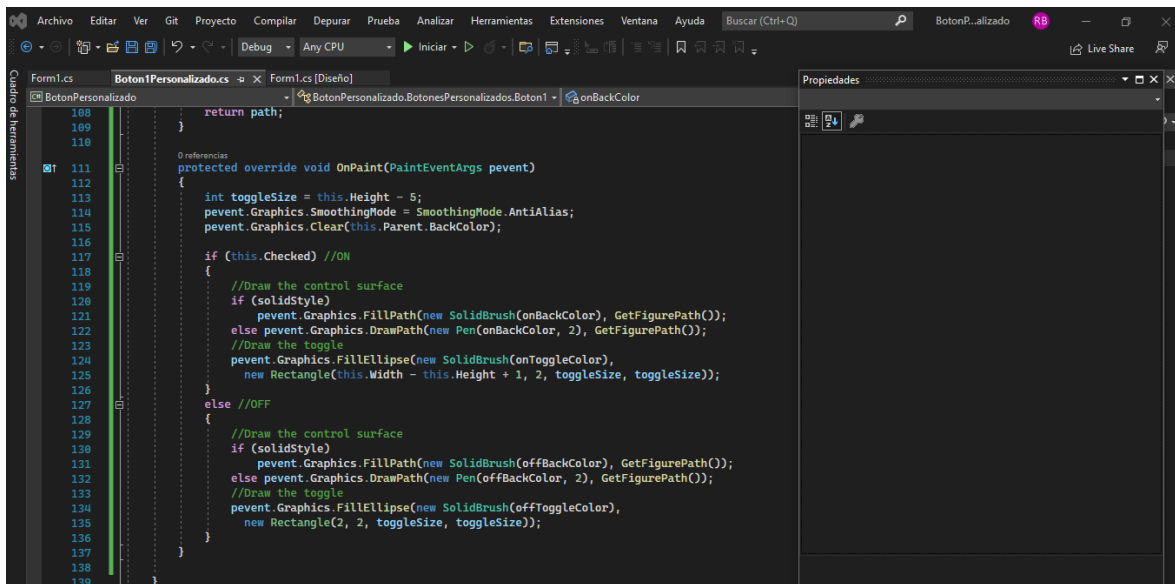


Figura 1.7

## 9.- El botón personalizado aparece en nuestra caja de herramientas

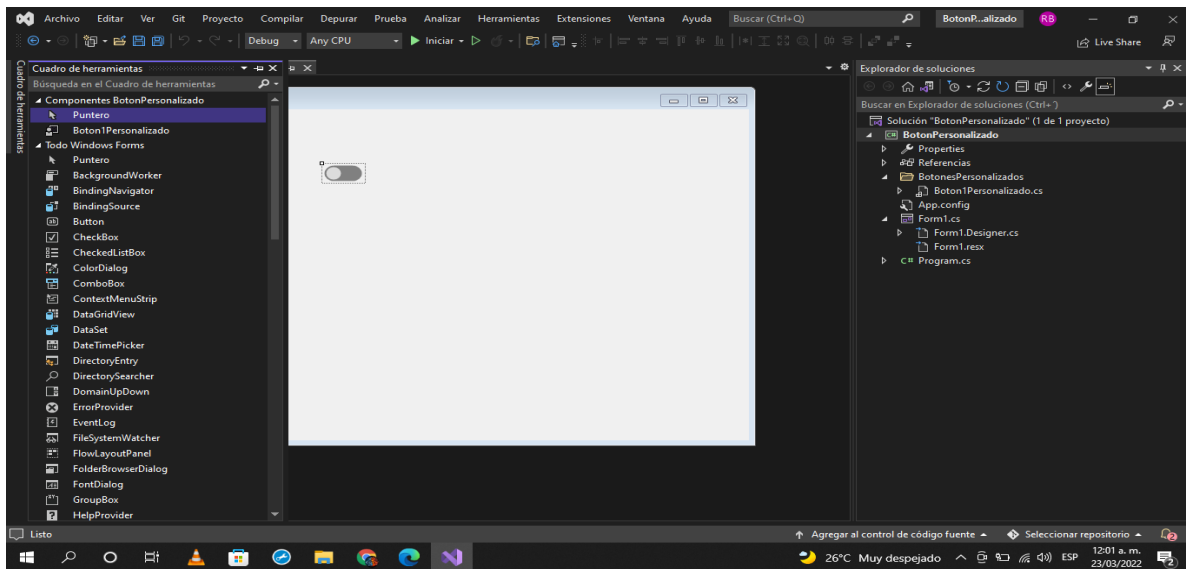


Figura 1.8

## 10.- Hacemos uso del botón personalizado en nuestro proyecto

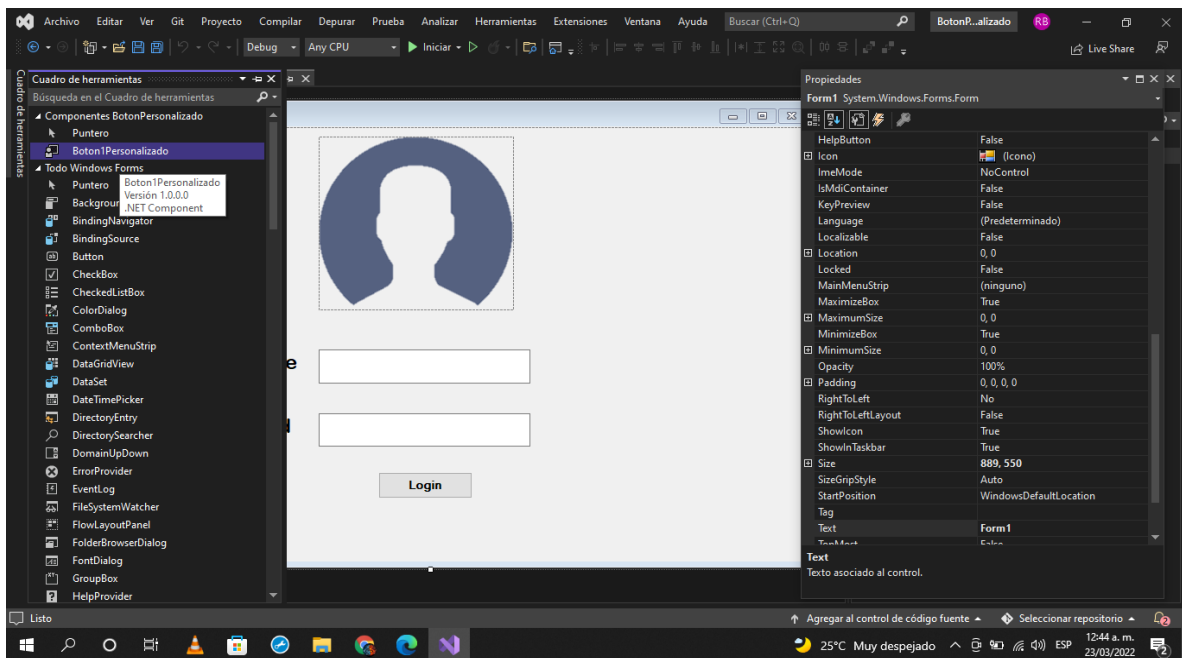


Figura 1.9

## 11.- Agregamos evento al Botón personalizado

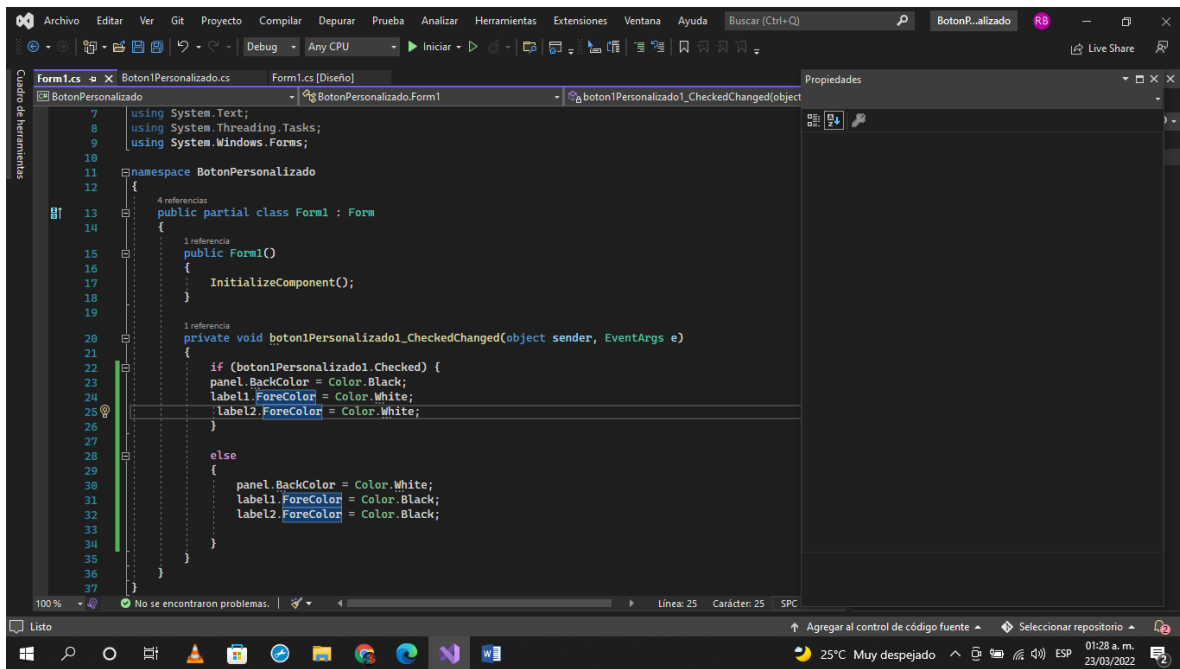
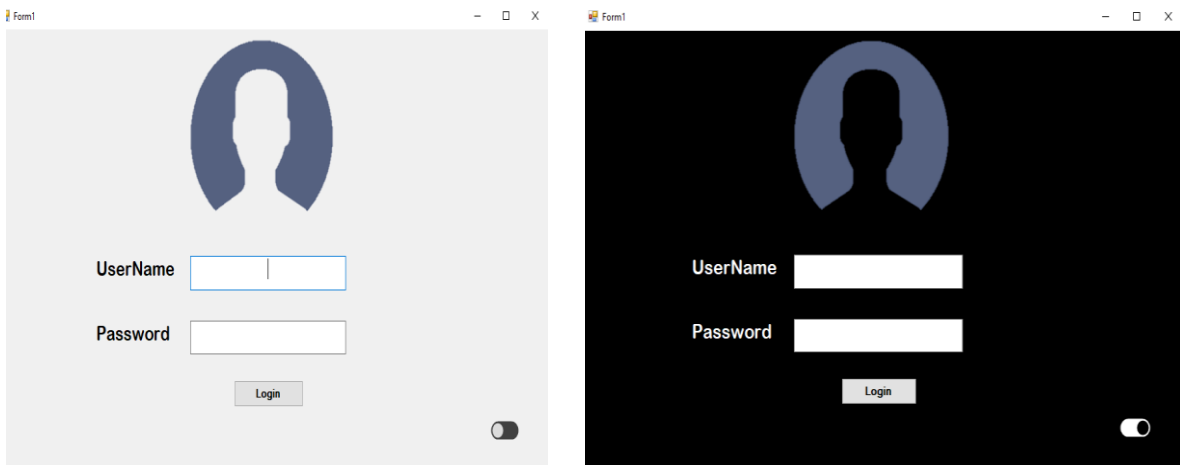


Figura 2

## Resultados





## **Conclusión**

En esta Práctica se realizó la implementación de controles personalizados de Windows Forms con el lenguaje C#, añadiendo este componente a nuestro proyecto creado. Crear Botones personalizados permite al programador crear interfaces más dinámicas e interesantes para el usuario por lo que es de gran importancia aprender a diseñar y crear componentes.

El control que se implementó es un Toggle Button o Switch Button. Este control es muy parecido al control Radio Button o Check Box, ya que solamente presentan 2 estados. Falso o Verdadero, Checked o Unchecked, Encendido o Apagado, por lo que resulta muy útil implementarlo en un proyecto.

Las librerías proporcionan al programador diversas funcionalidades, evitando la duplicidad de código. Además, reducen el tiempo de desarrollo y aumentan calidad del mismo.