

INSTITUTO TECNOLÓGICO DE TIJUANA

SUBDIRECCIÓN ACADÉMICA

DEPARTAMENTO DE SISTEMAS Y COMPUTACIÓN

SEMESTRE ENERO JUNIO - 2024

INGENIERÍA EN SISTEMAS COMPUTACIONALES

LENGUAJES DE INTERFAZ

PRÁCTICA 1 - DOCUMENTACIÓN

JULIO ALEJANDRO HERNÁNDEZ LEÓN - 21211963
SANTY FRANCISCO MARTINEZ CASTELLANOS - 21211989
LUIS ROBERTO LEAL LUA - 21211970

RENÉ SOLIS REYES

,				
ı	-	d	_	^
			-	_
		u	•	

Código Utilizado y Descripción.....3

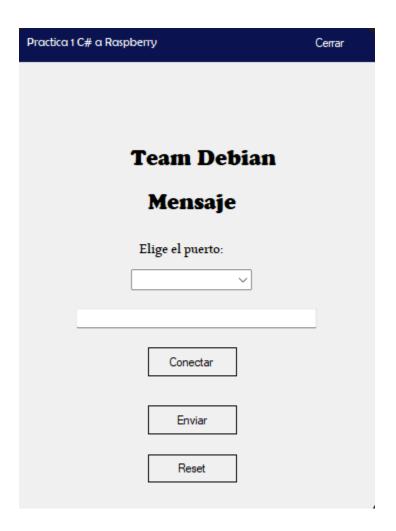
Código Utilizado y Descripción

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Drawing.Drawing2D;
using System.IO.Ports;
using System.Ling;
using System.Text;
using System. Threading. Tasks;
using System.Windows.Forms;
namespace Práctica_1_Team_Debian
{
  public partial class Form1 : Form
  {
          SerialPort puertoar; // Declaración de un objeto SerialPort para la
comunicación serial
    public Form1()
    {
       InitializeComponent(); // Inicializa los componentes del formulario
       // Agrega los nombres de los puertos COM al comboBox1
       for(int i = 0; i < 16; i++)
       {
         comboBox1.Items.Add("COM" + i);
       }
    }
```

```
private void Form1 Load(object sender, EventArgs e)
       GraphicsPath path = new GraphicsPath(); // Crea un objeto GraphicsPath
para definir la ruta de gráficos
       int radius = 20; // Radio para las esquinas redondeadas
       // Definición de la ruta de gráficos con esquinas redondeadas
       path.StartFigure();
       path.AddArc(0, 0, radius * 2, radius * 2, 180, 90);
       path.AddLine(radius, 0, Width - radius, 0);
       path.AddArc(Width - radius * 2, 0, radius * 2, radius * 2, 270, 90);
       path.AddLine(Width, radius, Width, Height - radius);
         path.AddArc(Width - radius * 2, Height - radius * 2, radius * 2, radius * 2,
0, 90);
       path.AddLine(Width - radius, Height, radius, Height);
       path.AddArc(0, Height - radius * 2, radius * 2, radius * 2, 90, 90);
       path.AddLine(0, Height - radius, 0, radius);
       path.CloseFigure();
        this.Region = new Region(path); // Establece la región del formulario con
la forma definida por path
     }
     private void button1 Click(object sender, EventArgs e)
     {
       puertoar = new SerialPort(); // Inicializa el objeto SerialPort
       try
       {
           puertoar.PortName = comboBox1.SelectedItem.ToString(); // Asigna el
nombre del puerto seleccionado
```

```
puertoar.BaudRate = 115200; // Configura la tasa de baudios
         puertoar.DtrEnable = true; // Habilita el control de terminal de datos
         try
         {
            puertoar.Open(); // Intenta abrir el puerto
              MessageBox.Show("Sí se conectó:)"); // Muestra un mensaje si la
conexión es exitosa
         catch (Exception ex)
         {
             MessageBox.Show("No está conectado"); // Muestra un mensaje si
la conexión falla
         }
       }
       catch(Exception a)
       {
          MessageBox.Show("Por favor elige un puerto"); // Muestra un mensaje
si no se selecciona un puerto
       }
    }
    private void button2 Click(object sender, EventArgs e)
    {
       try
       {
               puertoar.WriteLine(textBox1.ToString()); // Envía el contenido de
textBox1 a través del puerto serial
           MessageBox.Show("Se envio correctamente"); // Muestra un mensaje
si el envío es exitoso
       }
```

```
catch (Exception ex)
       {
            MessageBox.Show("No se envió" + ex); // Muestra un mensaje si el
envío falla
       }
     }
     private void button3 Click(object sender, EventArgs e)
     {
       textBox1.Text = ""; // Limpia el contenido de textBox1
       puertoar.Close(); // Cierra el puerto serial
          MessageBox.Show("El puerto ha sido desconectado"); // Muestra un
mensaje indicando que el puerto ha sido desconectado
       comboBox1.ResetText(); // Reinicia el texto del comboBox1
     }
     private void button4_Click(object sender, EventArgs e)
     {
       this.Close(); // Cierra el formulario
     }
     private void panel1 Paint(object sender, PaintEventArgs e)
     {
          // Método vacío para manejar el evento Paint del panel, no se realiza
ninguna acción personalizada
     }
  }
}
```



Se elige el puerto y se escribe el mensaje, no sin antes haber seleccionado la opción de conectar y después enviamos el mensaje.