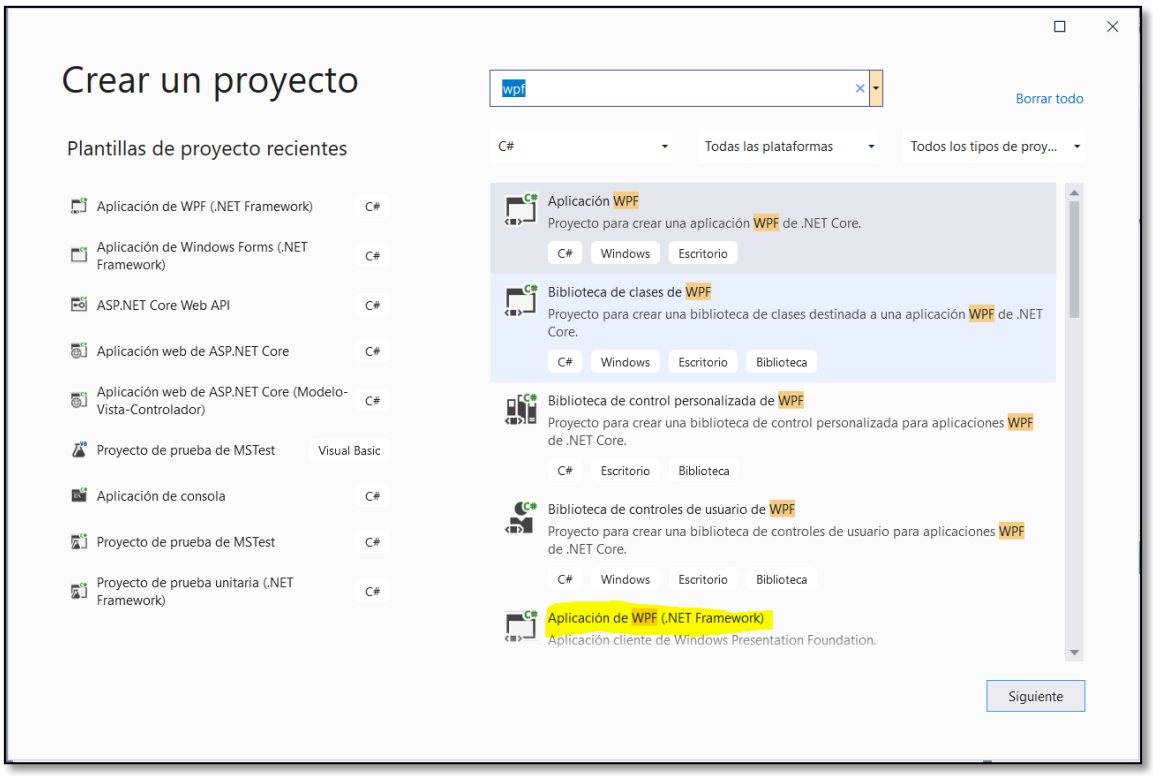
WPF

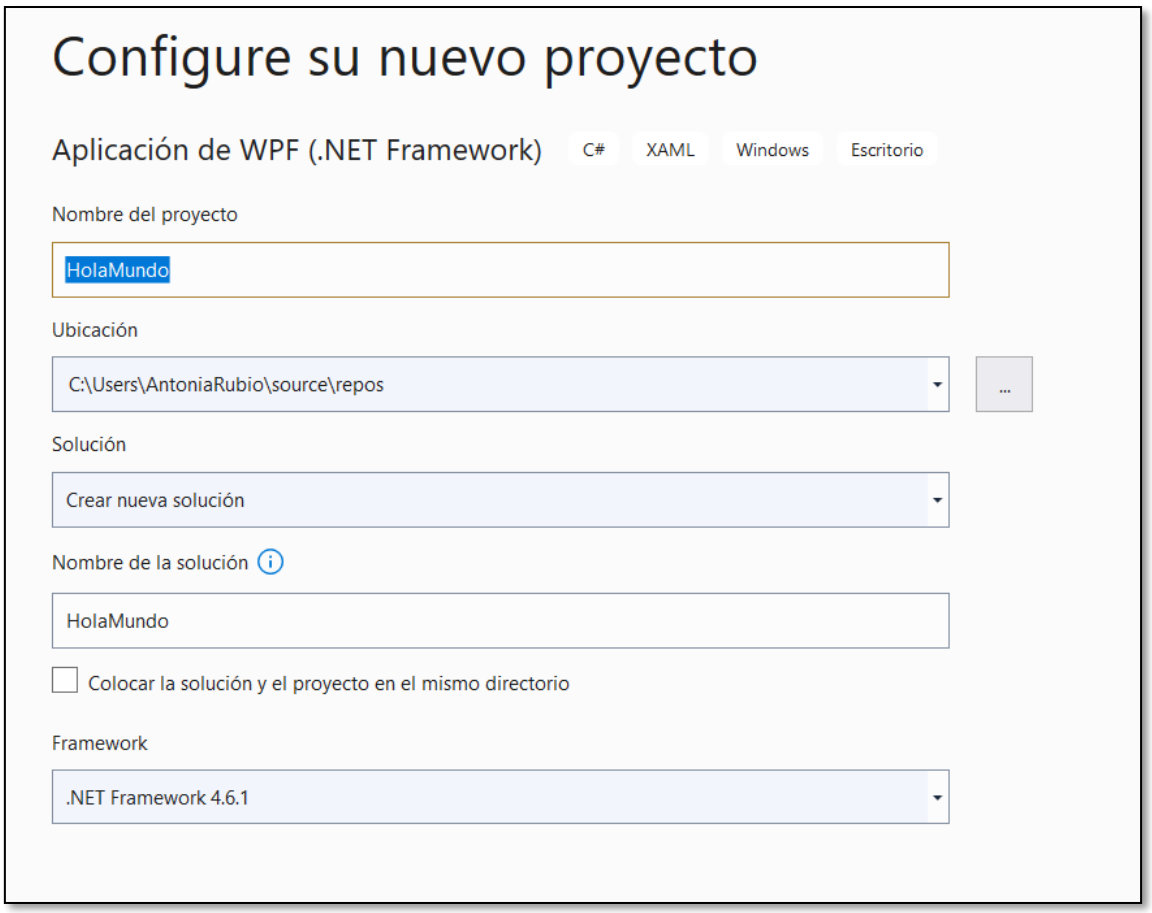
- WPF es una API que pertenece al framework .NET para la creación de interfaces de usuario en Windows.
- Utiliza el lenguaje XAML (similar a HTML).
- Sucesor de Windows Forms con nuevas funcionalidades como 3D, tipografía avanzada y documentos similares a PDF.

Actividad 20

1. Creamos un nuevo proyecto de “Aplicación de WPF (.NET Framework)”

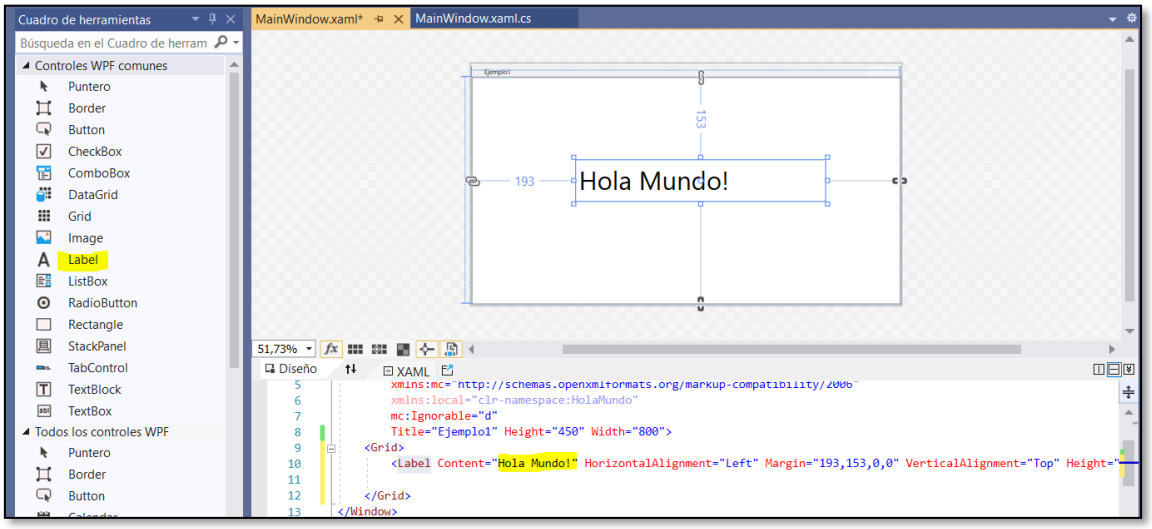


Con el nombre “HolaMundo” y con la versión de FrameWork 4.6.1

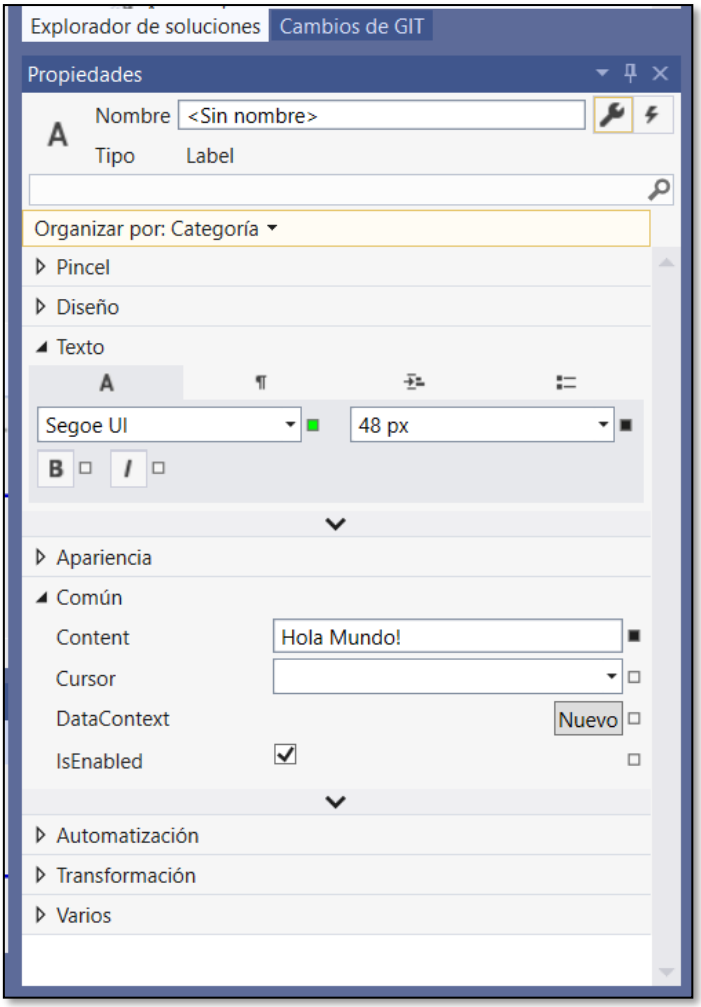


2. Editamos el titulo de la interfaz en la etiqueta “Title” y el tamaño en “Height” y “Width”.

3. Añadimos un elemento “Label” y editamos su contenido como “Hola Mundo!”.



Para editar el tipo de letra y el tamaño utilizamos las propiedades del elemento.

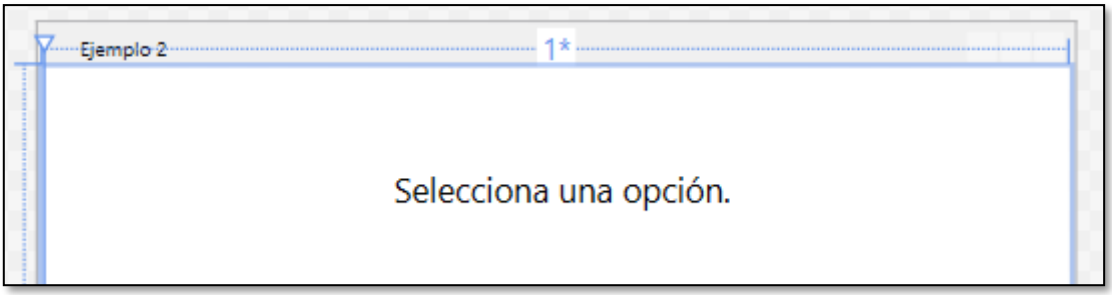


O añadiendo FontSize en la etiqueta.

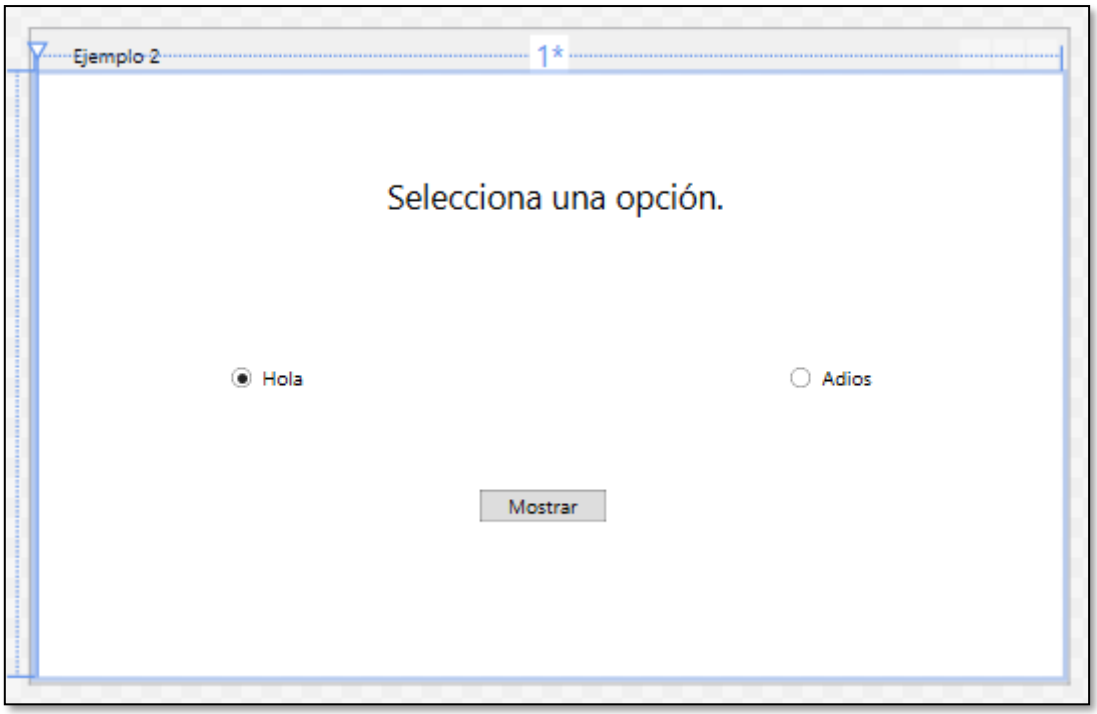
```
" Margin="193,153,0,0" VerticalAlignment="Top" Height="77" Width="462" FontSize="48"/>
```

Versión 2 Hola mundo

1. Crear un nuevo proyecto como hemos realizado anteriormente.
2. Añadimos un TextBlock y le cambiamos el atributo Text.



3. Añadimos dos RadioButton a los que les editaremos el texto y le añadiremos los controles “HelloButton” y “GoodbyeButton”.




```
</Grid.ColumnDefinitions>
<TextBlock HorizontalAlignment="Left" Margin="210,60,0,0" TextWrapping="Wrap" Text="Selecciona una opción." VerticalAlignment="Top" FontSize="20" Grid.Colum
<RadioButton x:Name="HelloButton" Content="Hola" HorizontalAlignment="Left" Margin="116,175,0,0" VerticalAlignment="Top" IsChecked="True" Grid.ColumnSpan="2
<RadioButton x:Name="GoodbyeButton" Content="Adios" HorizontalAlignment="Left" Margin="451,175,0,0" VerticalAlignment="Top" Grid.ColumnSpan="2"/>
<Button Content="Mostrar" HorizontalAlignment="Left" Margin="266,249,0,0" VerticalAlignment="Top" Width="75" Click="Button_Click" Grid.ColumnSpan="2"/>
```

Añadimos la etiqueta IsChecked=”True” al primer RadioButton.

4. Añadimos un Botón y cambiamos su etiqueta Content.


```
<RadioButton x:Name="GoodbyeButton" Content="Adios
<Button Content="Mostrar" HorizontalAlignment="Lef
```

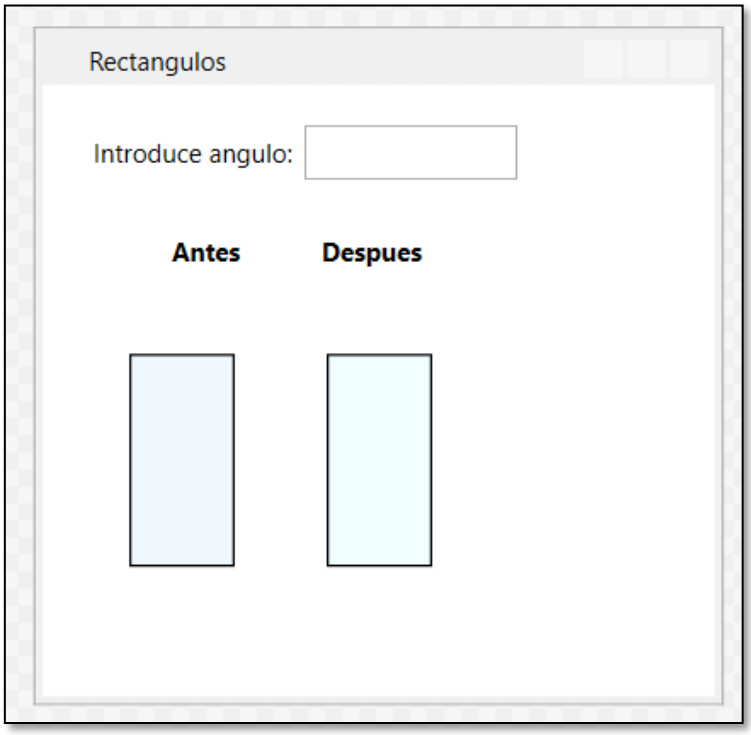
5. Damos funcionalidad al botón “Mostrar”.

```
dow.xaml*   MainWindow.xaml.cs
lo2                                   Ejemplo2.MainWindow
{
    /// <summary>
    /// Lógica de interacción para MainWindow.xaml
    /// </summary>
    2 referencias
    public partial class MainWindow : Window
    {
        0 referencias
        public MainWindow()
        {
            InitializeComponent();
        }

        1 referencia
        private void Button_Click(object sender, RoutedEventArgs e)
        {
            if (HelloButton.IsChecked == true)
            {
                MessageBox.Show("Hola Mundo.");
            }
            else if (GoodbyeButton.IsChecked == true)
            {
                MessageBox.Show("Adios Mundo.");
            }
        }
    }
}
```

Ejercicio:

Realizar una interfaz similar a la siguiente.



El segundo rectángulo debe de girar en función al ángulo que se introduzca por parámetro.