

## Segundo corte Eroneri.

¿Que es un WPF?

WPF (Window Presentation Function) nos permite crear aplicaciones de escritorio con diseño visual asombroso, es un motor de renderizado basado en vectores e independientes con el fin de aprovechar al máximo el hardware de grafico moderno. Se ampara con características de escritorio que incluyen lenguajes de marcado, controles, gráficos diseños, y gráficos 2D y 3D, animación estilo textos, entre otros.

### Version Net

Se trata de código abierto de WPF que está alojado en GitHub y se ejecuta en .NET o Superior para usuarios. Se requiere Visual Studio 2019 o superior aunque .NET es multiplataforma. WPF no es y no se ejecuta en Windows.

### Version .NET Framework 4

Es compatible con Visual Studio 2017 y 2019 es versión de .NET que es solo para Windows y se considera componente de él y se distribuye con .NET Framework.

WPF existe como un subconjunto de C# y WPF, estructura, etc. de .NET que se ejecutan en el espacio de nombres en System.Windows. La experiencia fundamental para hacer instancias de clase, establecer propiedades, etc. e incluso para mejorar propiedades, etc. e incluso para mejorar propiedades y eventos como dependencias y que dan resultados.



## XAML

Es un lenguaje de marcado declarativo para crear interfaces visuales y separar la definición de interfaz de usuario de los en tiempo de ejecución por código escondido.

Representa la creación de instancias en un ensamblado es diferentes a otros lenguajes de marcado como HTML con sistema de tipos que lo respaldan. Permite un flujo de trabajo pudiendo trabajar en interfaz lógica utilizando herramientas diferentes cuando son texto se representan XML que tienen la extensión .xaml de codificación típica es la UTF8.

### Sintaxis en XAML

Define sus propios conceptos dentro del lenguaje XAML declarar una instancia de un tipo, un elemento representa un objeto, se define lo les que nos referencia siempre comienzan con un paréntesis luego el nombre del tipo. el nombre y un contenido y termina con un paréntesis de cierre.

Cada instancia se crea llamando al constructor sin parámetro y cada elemento es una instancia de algún tipo, es decir que es un objeto.

### Sintaxis Propiedades

La sintaxis de atributos no es posible por que el objeto o información necesaria pero proporcionar el valor de la propiedad no se puede expresar dentro de comillas y corchetes de sintaxis de atributos.

## Emmanuel Izaguirre Cones.

### Sintaxis colecciones.

El lenguaje XAML incluye optimizaciones que producen un marcado más legible para los programadores. Es que en particular toman un tipo de colección para tomar parte de la colección y así como se establece.

### Sintaxis de propiedades de contenido

Una clase o tipo puede designar un peso que sea la propiedad de contenido XAML únicamente para la propiedad de contenido. Podemos omitir un tipo de propiedad al establecer esa propiedad en el marcado XAML.

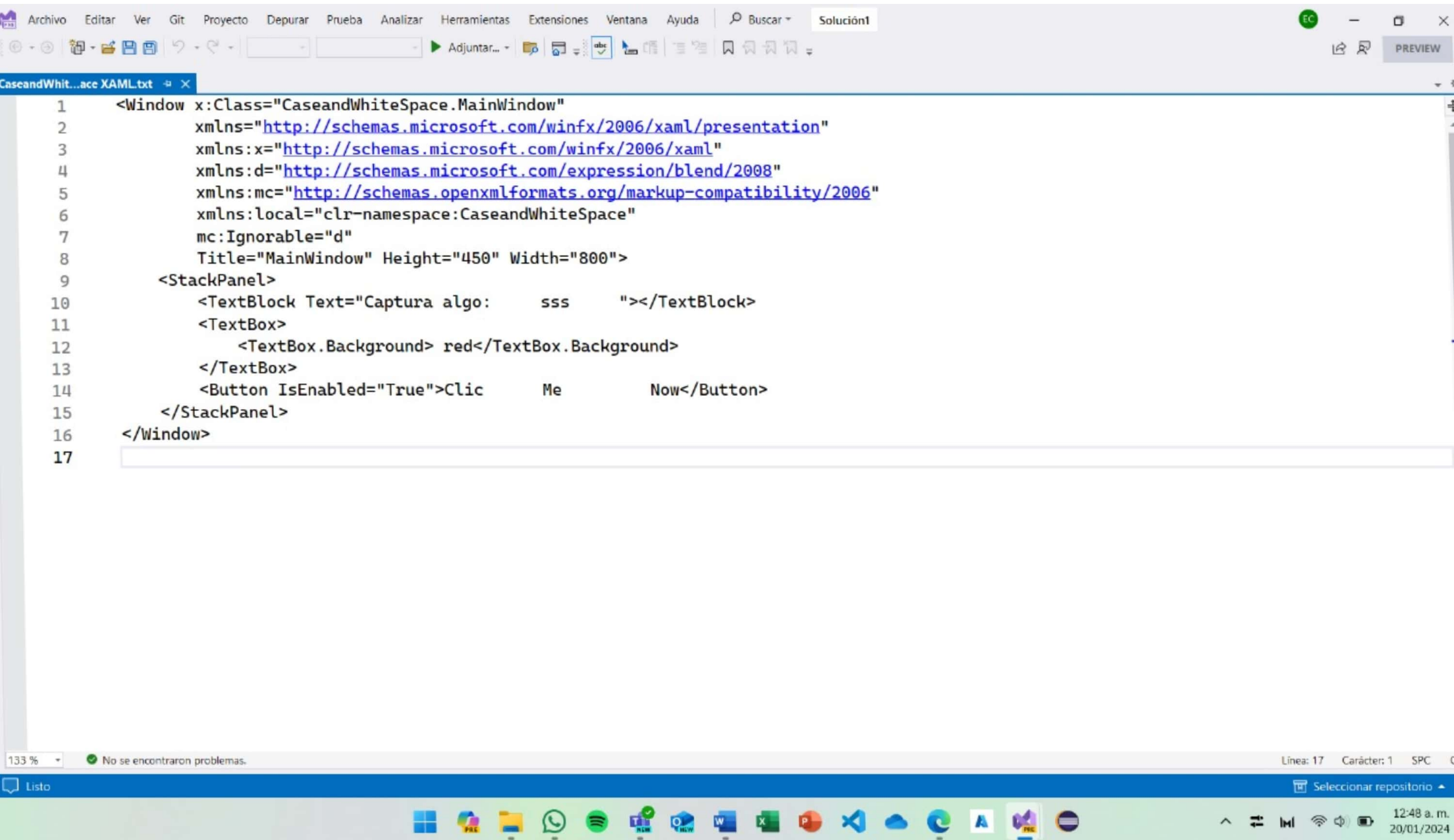
### Sintaxis de texto como contenido

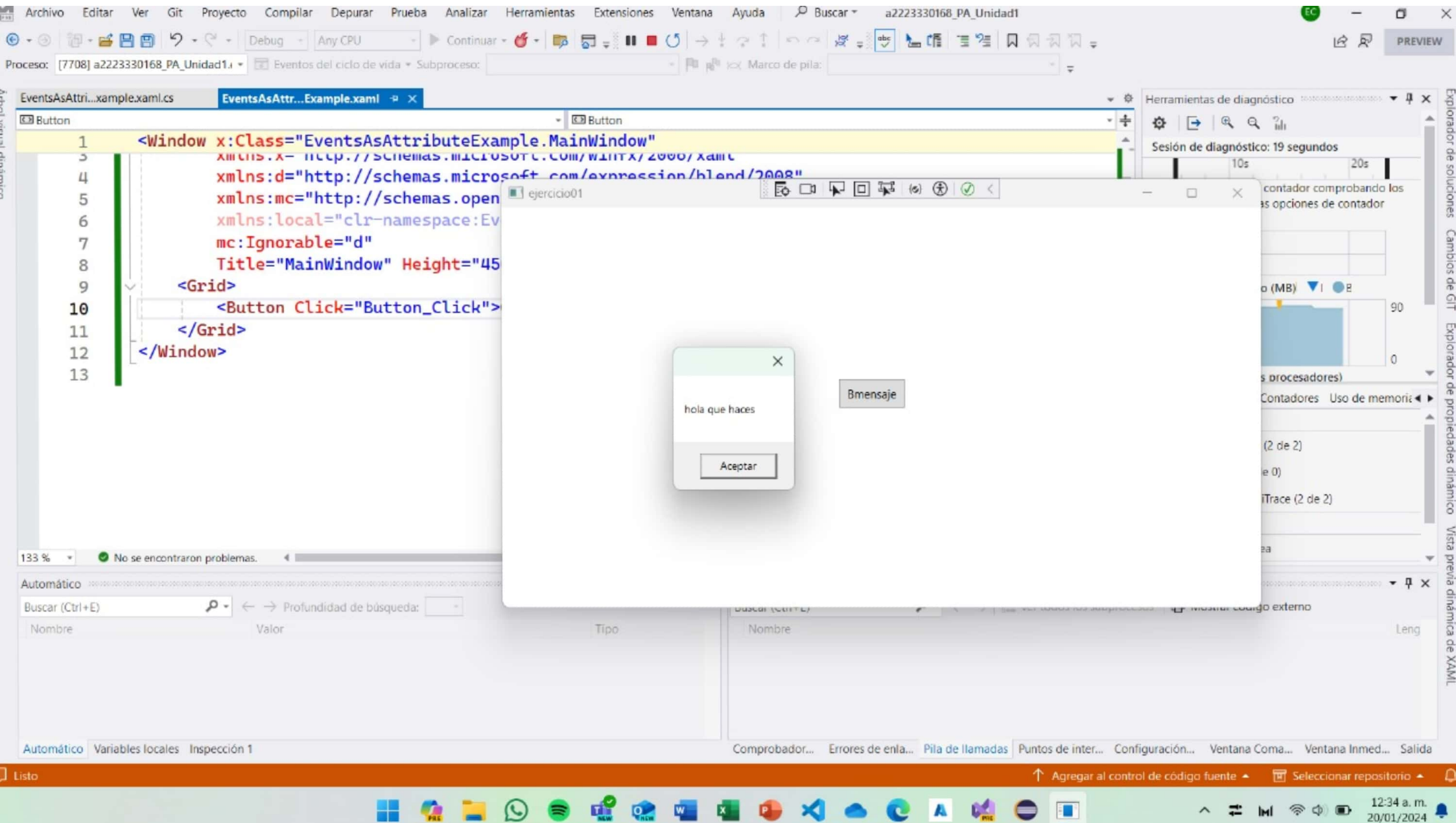
Una pequeña cantidad de elemento en XAML puede proceder texto como su contenido para que esto suceda de uno debe ser verdadero la clase debe declarar una propiedad de contenido debe ser enumerable o una cadena o un string.

### Sintaxis eventos.

Se pueden usar para miembros que son eventos en lugar de propiedades en este caso el nombre del atributo es el nombre del evento. Para implementación de eventos de .NET para XAML el valor del atributo es el nombre de un controlador es decir eventos en un código que responderá al evento.







Archivo Editar Ver Git Proyecto Compilar Depurar Prueba Analizar Herramientas Extensiones Ventana Ayuda Buscar a2223330168\_PA\_Unidad1

Salida EventsAsAttr...ample.xaml.cs EventsAsAttri...teExample.xaml

a2223330168\_PA\_Unidad1 EventsAsAttributeExample.MainWindow MainWindow()

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6 using System.Windows;
7 using System.Windows.Controls;
8 using System.Windows.Data;
9 using System.Windows.Documents;
10 using System.Windows.Input;
11 using System.Windows.Media;
12 using System.Windows.Media.Imaging;
13 using System.Windows.Navigation;
14 using System.Windows.Shapes;
15
16 namespace EventsAsAttributeExample
17 {
18     /// <summary>
19     /// Lógica de interacción para MainWindow.xaml
20     /// </summary>
21     2 referencias
22     public partial class MainWindow : Window
23     {
24         0 referencias
25         public MainWindow()
26         {
27             InitializeComponent();
28         }
29     }
30 }
```

Explorador de soluciones

Buscar en Explorador de soluciones (C)

Solución "a2223330168\_PA\_Unidad1"

- a2223330168\_PA\_Unidad1
  - Dependencias
  - App.xaml
  - AssemblyInfo.cs
  - ejercicio01.xaml
  - ejercicio02.xaml
  - EventsAsAttributeExample.xaml
  - EventsAsAttributeExample.xa

133 % No se encontraron problemas. Línea: 33 Carácter: 2 SPC CRLF

Listo Agregar al control de código fuente Seleccionar repositorio

12:47 a. m. 20/01/2024

