```
/*--------------------------------------------------*\
 * Author    : Salvi Cyril
 * Date      : 8th juny 2017
 * Diploma   : RaspiHome
 * Classroom : T.IS-E2B
 *
 * Description:
 *      RaspiHomeTabletWindows is a program
 *   compatible with the Windows tablet. It's a
 *   program that can be use as tactil graphic
 *   interface to order the component linked with
 *   the other Raspberry Pi.
\*--------------------------------------------------*/

using System;
using Windows.UI;
using Windows.UI.Xaml;
using Windows.UI.Xaml.Controls;
using Windows.UI.Xaml.Media;
using Windows.UI.Xaml.Media.Imaging;

// The User Control item template is documented at http://go.microsoft.com/fwlink/?LinkId=234236

namespace RaspiHomeTabletWindows.Menu.MenuToolbar
{
    public sealed partial class ToolbarButtonView : UserControl
    {
        #region Fields
        #region Constants
        #endregion

        #region Variables
        private ToolbarButtonModel _model = null;

        public event EventHandler _click;

        private bool _isSelected = false;
        private bool _isPressed = false;

        private string _whoseButtonClicked = "";
        #endregion
        #endregion

        #region Properties
        public ToolbarButtonModel Model
        {
            get
            {
                return _model;
            }

            set
            {
                _model = value;
            }
        }

        public bool IsSelected
        {
            get
            {
                return _isSelected;
            }

            set
            {
                _isSelected = value;

                btnToolbar.IsTabStop = _isSelected;

                if (value)
                {
                    btnToolbar.BorderThickness = new Thickness(6.5);
                    btnToolbar.BorderBrush = new SolidColorBrush(Color.FromArgb(255, 73, 130, 5));
                }
                else
                {
                    btnToolbar.BorderThickness = new Thickness(5);
                    btnToolbar.BorderBrush = new SolidColorBrush(Color.FromArgb(255, 76, 74, 75));
                }
            }
        }

        public bool IsPressed
        {
            get
            {
                return _isPressed;
            }

            set
            {
                _isPressed = value;
            }
        }

        public string WhoseButtonClicked
        {
            get
            {
                return _whoseButtonClicked;
            }

            set
            {
                _whoseButtonClicked = value;
            }
        }
        #endregion

        #region Constructor
        /// <summary>
        /// Constructor: Initializer
        /// </summary>
        public ToolbarButtonView(string frameChoose, string description, string iconLink)
        {
            InitializeComponent();

            this.Model = new ToolbarButtonModel(this);

            this.WhoseButtonClicked = frameChoose;

            SetInformation(description, iconLink);
        }
        #endregion

        #region Events
        /// <summary>
        /// Event on the button
        /// </summary>
        /// <param name="sender"></param>
        /// <param name="e"></param>
        private void btnToolbar_Click(object sender, RoutedEventArgs e)
        {
            if (this._click != null)
                this._click(this, null);
            this.IsPressed = true;
        }
        #endregion

        #region Methods
        /// <summary>
        /// Set information of the button on the toolbar
        /// </summary>
        /// <param name="description"></param>
        /// <param name="iconLink"></param>
        private void SetInformation(string description, string iconLink)
        {
            this.Model.SetInformation(description, iconLink);
        }

        /// <summary>
        /// Change the Icon of the button on the toolbar
        /// </summary>
        /// <param name="iconPath"></param>
        public void ChangeIcon(string iconPath)
        {
            if (iconPath != "")
                this.imgMenuToolbarButton.Source = new BitmapImage(new Uri(iconPath));
        }
        #endregion
    }
}
```