

# Práctica 2 Kinect

## Interacción Gestual

- Miguel Sánchez Maldonado
- Cristina Zuheros Montes

# Descripción del problema.

- Juego de preguntas: el usuario tendrá que interaccionar mediante kinect para marcar la solución correcta.
- Se pide nivel de dificultad y tema.
- Se hacen rondas de 5 preguntas, cada una con 3 respuestas posibles.
- Se contabilizan las respuestas correctas.
- Nos basamos en ControlBasics-WPF.



HISTORIA



ARTE



DEPORTES



LITERATURA



GEOGRAFIA



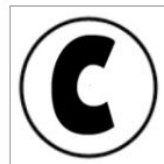
CIENCIA

# ¡SELECCIONE DIFICULTAD!

- A) FÁCIL
- B) INTERMEDIO
- C) DIFÍCIL



0/5



# Seleccionando dificultad

```
private void KinectTileButtonClick2(object sender, RoutedEventArgs e)
{
    var button = (KinectTileButton)e.OriginalSource;
    if (tema == null && dificultad == null && opcion == null)
    {
        dificultad = (button.Label as string);
        this.Imagen_pregunta.Source = new BitmapImage(new Uri(System.IO.Path.GetFullPath("S_Tema.png")));
    }
    else if (tema != null && opcion == null && dificultad != null && !primera)
    {
        opcion = (button.Label as string);
        comprobarPregunta();

        this.Imagen_pregunta.Source = new BitmapImage(new Uri(System.IO.Path.GetFullPath("S_Tema.png")));
        if (n_pregunta >= 5 && tema != null && dificultad != null && opcion != null)
        {
            this.Imagen_pregunta.Source = new BitmapImage(new Uri(System.IO.Path.GetFullPath("S_Tema.png")));
            this.fondo.Visibility = Visibility.Visible;
            tema = null;
            dificultad = null;
            n_correctas = 0;
            this.ContadorResultados.Text = n_correctas.ToString();
            n_pregunta = 1;
            this.correcto.Visibility = Visibility.Visible;
            opcion = null;
            primera = true;
        }
    }
    e.Handled = true;
}
```

Controles Basics



**¡SELECCIONE  
TEMA!**



0/5

# Seleccionando tema

```
private void KinectTileButtonClick(object sender, RoutedEventArgs e)
{
    var button = (KinectTileButton)e.OriginalSource;
    if (tema == null && dificultad != null)
    {
        tema = (button.Label as string);
        this.Imagen_pregunta.Source = new BitmapImage(new Uri(System.IO.Path.GetFullPath("S_PREGUNTA.png")));
        this.fondo.Visibility = Visibility.Visible;
        this.fondo.Source = new BitmapImage(new Uri(System.IO.Path.GetFullPath(tema + ".png")));
    }
    e.Handled = true;
}
```

Controls Basics



**¡PULSE AQUÍ PARA LA  
SIGUIENTE PREGUNTA!**

**A**

**B**

**C**

0/5

# Seleccionando opción.

```
private void KinectTileButtonClick2(object sender, RoutedEventArgs e)
{
    var button = (KinectTileButton)e.OriginalSource;
    if (tema == null && dificultad == null && opcion == null)
    {
        dificultad = (button.Label as string);
        this.Imagen_pregunta.Source = new BitmapImage(new Uri(System.IO.Path.GetFullPath("S_Tema.png")));
    }
    else if (tema != null && opcion == null && dificultad != null && !primera)
    {
        opcion = (button.Label as string);
        comprobarPregunta();

        this.Imagen_pregunta.Source = new BitmapImage(new Uri(System.IO.Path.GetFullPath("S_Pregunta.png")));
        if (n_pregunta >= 5 && tema != null && dificultad != null)
        {
            this.Imagen_pregunta.Source = new BitmapImage(new Uri(System.IO.Path.GetFullPath("DIFICULTAD.png")));
            this.fondo.Visibility = Visibility.Hidden;
            tema = null;
            dificultad = null;
            n_correctas = 0;
            this.ContadorResultados.Text = n_correctas + "/5";
            n_pregunta = 1;
            this.correcto.Visibility = Visibility.Hidden;
            opcion = null;
            primera = true;
        }
    }
    e.Handled = true;
}
```



## Creamos los botones para temas y opciones.

```
private KinectTileButton crearBoton(String nombre)
{
    BitmapImage bi = new BitmapImage();
    bi.BeginInit();
    bi.UriSource = new Uri(nombre + ".png", UriKind.Relative);
    bi.EndInit();
    var button = new KinectTileButton
    {
        Label = nombre.ToString(CultureInfo.CurrentCulture),
        FontSize = 35,
        Background = new ImageBrush(bi),
        LabelBackground = new ImageBrush(),
        Height = 150,
        Width = 200,
        VerticalLabelAlignment = VerticalAlignment.Center,
        HorizontalLabelAlignment = HorizontalAlignment.Center,
    };
    return button;
}
```

```
var button = crearBoton("HISTORIA");
this.wrapPanel.Children.Add(button);
```

```
var button2 = crearBoton2("A");
this.wrapPanel2.Children.Add(button2);
```

# Formulando pregunta

```
private void FormulaPregunta(object sender, RoutedEventArgs e)
{
    var button = (KinectTileButton)e.OriginalSource;

    if (tema != null && primera && dificultad != null)
    {
        this.Imagen_pregunta.Visibility = Visibility.Visible;

        this.Imagen_pregunta.Source = new BitmapImage(new Uri(System.IO.Path.GetFullPath(tema + n_pregunta + dificultad + ".png")));
        primera = false;
    }
    else if (n_pregunta < 5 && tema != null && opcion != null && dificultad != null)
    {
        n_pregunta++;
        this.Imagen_pregunta.Source = new BitmapImage(new Uri(System.IO.Path.GetFullPath(tema + n_pregunta + dificultad + ".png")));
        this.correcto.Visibility = Visibility.Hidden;
        opcion = null;
    }

    e.Handled = true;
}
```



## ASPECTOS DESTACADOS:

- **Fusionar botón de correcto e incorrecto.**

Único botón, en comprobarPregunta() manda la imagen adecuada al botón.

- **Especificar la respuesta correcta.**

Si nos muestra la respuesta correcta, sabríamos todas las respuestas y no tendría sentido seguir jugando.