



Jorge Quintana González

Introducción

Vamos a hacer un trabajo en el módulo Diseño de interfaces (DI) en el cual tendremos que hacer una aplicación.

objetivo

Tendremos que utilizar la herramienta Microsoft Visual Studio para realizar una aplicación donde el usuario introducirá su nivel en varios idiomas. Dependiendo del nivel, cambiará el color el label y se desbloquearán otros botones.

Actividad

mainPage.xaml

En este código está diseñada para que los usuarios ingresen su nivel de dominio en varios idiomas, que pueden ser *B* (bajo), *M* (medio) o *A* (alto). He utilizado un `ScrollView` para asegurarme de que la interfaz sea desplazable, lo que es útil en pantallas pequeñas o cuando el contenido se extiende verticalmente. Luego, dentro de un `VerticalStackLayout`, he añadido un espaciado entre los elementos y un margen superior para una disposición ordenada.

Primero, he añadido dos etiquetas. La primera, con un texto de mayor tamaño, muestra el título "Nivel de idiomas", y la segunda etiqueta explica la convención de niveles con el texto

"B: bajo, M: medio, A: alto" y utiliza un estilo predefinido para que su apariencia esté unificada con el resto de la aplicación.

A continuación, he añadido tres secciones para que el usuario pueda ingresar su nivel en valenciano, inglés y francés. Para cada idioma, he creado un `HorizontalStackLayout` que contiene una `Label` para indicar el idioma y una entrada (`Entry`) para que el usuario ingrese un único carácter (B, M o A) que representa su nivel. He configurado cada entrada para aceptar solo una letra en mayúsculas y cambiar el color de fondo según el valor ingresado: *Verde* para alto, *naranja* para medio y *rojo* para bajo. Para lograr esto, he utilizado `Triggers` que reaccionan según el valor ingresado en la propiedad `Text`. También he añadido un botón "Insertar" junto a cada entrada, que ejecuta un método específico para cada idioma cuando el usuario hace clic, permitiendo manejar el evento de clic en el código detrás.

Después de las secciones de idiomas, he añadido un botón "Comprobar" que ejecutará una función para verificar los niveles de idioma ingresados. Debajo, he incluido un `HorizontalStackLayout` que contiene tres botones (Avanzado 1, Avanzado 2, Avanzado 3). Estos botones están inicialmente deshabilitados y solo se activarán mediante `DataTriggers` que dependen del valor de una propiedad de enlace llamada `comprobar`. Cada botón se habilita si el valor de `comprobar` coincide con el valor especificado en el desencadenador (1, 2 o 3), indicando diferentes niveles de conocimiento avanzado en los idiomas ingresados.

Para finalizar, he añadido una etiqueta llamada `labelIdiomas` que muestra el número de idiomas ingresados a nivel avanzado, y un último botón "Ver créditos" que ejecuta el método `verCreditos` cuando el usuario hace clic en él, mostrará mi nombre.

```
<?xml version="1.0" encoding="utf-8" ?>
<ContentPage xmlns="http://schemas.microsoft.com/dotnet/2021/maui"
    xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
    x:Class="Unidad2Parte2Jorge.MainPage">

    <ScrollView>
        <VerticalStackLayout
            Padding="30,0"
            Spacing="25">
            <Label
                Text="Nivel de idiomas"

                FontSize="Large"
            />
            <Label
                Text="B: bajo, M:medio, A:alto"
                Style="{StaticResource Headline}"
                SemanticProperties.HeadingLevel="Level1" />
            <HorizontalStackLayout>
                <Label Text="Nivel de Valenciano [B,M,A]"/>
```

```

        <Entry x:Name="entryValenciano" MaxLength="1" TextTransform="Uppercase"
BackgroundColor="White">
        <Entry.Triggers>
            <Trigger TargetType="Entry" Property="Text" Value="A">
                <Setter Property="BackgroundColor" Value="LightGreen" />
            </Trigger>
            <Trigger TargetType="Entry" Property="Text" Value="B">
                <Setter Property="BackgroundColor" Value="Red" />
            </Trigger>
            <Trigger TargetType="Entry" Property="Text" Value="M">
                <Setter Property="BackgroundColor" Value="Orange" />
            </Trigger>
        </Entry.Triggers>
    </Entry>
    <Button Text="Insertar" Clicked="InputValenciano"></Button>
</HorizontalStackLayout>
<HorizontalStackLayout>
    <Label Text="Nivel de inglés [B,M,A]" />
    <Entry x:Name="entryIngles" MaxLength="1" TextTransform="Uppercase"
BackgroundColor="White" >
        <Entry.Triggers>
            <Trigger TargetType="Entry" Property="Text" Value="A">
                <Setter Property="BackgroundColor" Value="LightGreen" />
            </Trigger>
            <Trigger TargetType="Entry" Property="Text" Value="B">
                <Setter Property="BackgroundColor" Value="Red" />
            </Trigger>
            <Trigger TargetType="Entry" Property="Text" Value="M">
                <Setter Property="BackgroundColor" Value="Orange" />
            </Trigger>
        </Entry.Triggers>
    </Entry>
    <Button Text="Insertar" Clicked="InputIngles"></Button>
</HorizontalStackLayout>
<HorizontalStackLayout>
    <Label Text="Nivel de francés [B,M,A]" />
    <Entry x:Name="entryFrances" MaxLength="1" BackgroundColor="White"
Text="">
        <Entry.Triggers>
            <Trigger TargetType="Entry" Property="Text" Value="A">
                <Setter Property="BackgroundColor" Value="LightGreen" />
            </Trigger>
            <Trigger TargetType="Entry" Property="Text" Value="B">
                <Setter Property="BackgroundColor" Value="Red" />
            </Trigger>
            <Trigger TargetType="Entry" Property="Text" Value="M">
                <Setter Property="BackgroundColor" Value="Orange" />
            </Trigger>
        </Entry.Triggers>
    </Entry>
    <Button Text="Insertar" Clicked="InputFrances"></Button>
</HorizontalStackLayout>
</HorizontalStackLayout>

```

```

        </Entry.Triggers>
    </Entry>
    <Button Text="Insertar" Clicked="InputFrances"></Button>
</HorizontalStackLayout>
<Button Text="Comprobar" Clicked="BtnComprobar" ></Button>
<HorizontalStackLayout>
    <Button Text="Avanzado 1" IsEnabled="False" Margin="10">
        <Button.Triggers>
            <DataTrigger TargetType="Button" Binding="{Binding comprobar}" Value="1"
>
                <Setter Property="IsEnabled" Value="True"></Setter>
            </DataTrigger>
        </Button.Triggers>
    </Button>
    <Button Text="Avanzado 2" IsEnabled="False" Margin="10">
        <Button.Triggers>
            <DataTrigger TargetType="Button" Binding="{Binding comprobar}" Value="2"
>
                <Setter Property="IsEnabled" Value="True"></Setter>
            </DataTrigger>
        </Button.Triggers>
    </Button>
    <Button Text="Avanzado 3" IsEnabled="False" Margin="10">
        <Button.Triggers>
            <DataTrigger TargetType="Button" Binding="{Binding comprobar}" Value="3"
>
                <Setter Property="IsEnabled" Value="True"></Setter>
            </DataTrigger>
        </Button.Triggers>
    </Button>
</HorizontalStackLayout>
<Label x:Name="labelIdiomas" Text="Idiomas a nivel avanzado ="></Label>
<Button Text="Ver créditos" Clicked="verCreditos"></Button>
</VerticalStackLayout>
</ScrollView>

</ContentPage>

```

mainPage.xaml.cs

En este archivo de código C#, he definido la lógica de la clase MainPage, que se conecta a la interfaz XAML para manejar la interacción de usuario con una página de contenido en una aplicación MAUI. La clase MainPage extiende ContentPage e implementa varias variables y métodos para registrar el nivel de habilidad del usuario en tres idiomas: valenciano, inglés y francés.

Para empezar, he declarado varias variables enteras: count para un contador general (aunque no se usa en el código), _comprobar para controlar el número de idiomas en nivel

avanzado, y tres grupos de variables (valencianoA, valencianoB, valencianoC, inglesA, inglesB, inglesC, francesA, francesB, francesC) que almacenan las cantidades de niveles de competencia específicos para cada idioma.

Dentro del constructor MainPage(), llamo a InitializeComponent() para inicializar los componentes de la interfaz y establezco el contexto de enlace de datos (BindingContext) a la propia instancia de MainPage. Esto permite que el enlace de datos funcione entre la interfaz XAML y la lógica de C#.

He implementado un método verCreditos que se activa al hacer clic en el botón "Ver créditos" de la interfaz. Este método muestra una ventana emergente (DisplayActionSheet) con información de los créditos (mi nombre), que permite al usuario cerrarla seleccionando "Cancel".

Luego, he definido tres métodos (InputValenciano, InputIngles, y InputFrances) para manejar la entrada del nivel de idioma cuando el usuario hace clic en los botones de "Insertar" en la interfaz. Cada método muestra una ventana emergente para preguntar el nivel deseado en cada idioma (con opciones "A", "B" o "M"). Si el usuario selecciona "A" (alto), se incrementa la variable correspondiente al nivel avanzado (valencianoA, inglesA, o francesA). Finalmente, el nivel seleccionado por el usuario se muestra en el Entry correspondiente (entryValenciano, entryIngles, o entryFrances).

Para el valor comprobar, he definido una propiedad que permite calcular y mostrar cuántos idiomas están en nivel avanzado. La propiedad comprobar actualiza el texto del labelIdiomas en la interfaz y, al hacerlo, notifica a la interfaz de cualquier cambio mediante OnPropertyChanged.

El método BtnComprobar se activa cuando el usuario hace clic en el botón "Comprobar". Este método abre una ventana emergente con dos opciones: "SI" y "NO". Si el usuario elige "SI", el valor de comprobar se establece sumando los niveles avanzados de cada idioma (valencianoA, inglesA, y francesA), y el número total de idiomas avanzados se muestra en labelIdiomas. Si el usuario elige "NO", los campos de entrada (Entry) de cada idioma se limpian.

En resumen, este código coordina la lógica detrás de la entrada de niveles de idioma, la visualización de créditos y el cálculo del total de idiomas en nivel avanzado, permitiendo una interacción controlada y dinámica en la interfaz de usuario.

```
namespace Unidad2Parte2Jorge
{
    public partial class MainPage : ContentPage
    {
        int count = 0;
        int _comprobar;
        int valencianoA;
        int valencianoB;
```

```

int valencianoC;

int inglesA;
int inglesB;
int inglesC;

int francesA;
int francesB;
int francesC;

public MainPage()
{
    InitializeComponent();
    BindingContext = this;
}

private async void verCreditos(object sender, EventArgs e)
{
    string action = await DisplayActionSheet("Creditos:\n Jorge Quintana González",
"Cancell", null);
}

private async void InputValenciano(object sender, EventArgs e)
{
    string action = await DisplayActionSheet("¿Nivel de valenciano?", "Cancell", null, "A",
"B", "M");
    if (action.Equals("A"))
    {
        valencianoA++;
    }
    entryValenciano.Text = action;
}

public async void InputIngles(object sender, EventArgs e)
{
    string action = await DisplayActionSheet("¿Nivel de inglés?", "Cancell", null, "A", "B",
"M");
    if (action.Equals("A"))
    {
        inglesA++;
    }
    entryIngles.Text = action;
}

public async void InputFrances(object sender, EventArgs e)
{
    string action = await DisplayActionSheet("¿Nivel de francés?", "Cancell", null, "A",
"B", "M");
    if (action.Equals("A"))
    {
        francesA++;
    }
}

```

```

    }
    entryFrances.Text = action;
}
public int comprobar
{
    get { return _comprobar; }
    set
    {
        _comprobar = value;
        labelIdiomas.Text = "Idiomas a nivel avanzado =" + value;
        OnPropertyChanged(nameof(comprobar));
    }
}



private async void BtnComprobar(object sender, EventArgs e)
{
    string action = await DisplayActionSheet("¿Quieres ver el número de idiomas avanzados?", "Cancel", null, "SI", "NO");
    if(action.Equals("SI"))
    {
        comprobar = 0;
        comprobar = this.valencianoA + this.inglesA + this.francesA;

    }
    else if(action.Equals("NO"))
    {
        entryValenciano.Text = "";
        entryIngles.Text = "";
        entryFrances.Text = "";
    }

}
}
}

```

Diagrama de Gantt

Tarea	Fecha de Inicio	Fecha de Fin	Duración (días)	Días
Inicio del Proyecto	Miércoles 13	-	-	-
Diseño XAML	Miércoles 13	Sábado 23	10	
Programación en C#	Sábado 23	Lunes 18	4	

Problemas

Me sigue costando entender la parte lógica.

Conclusión