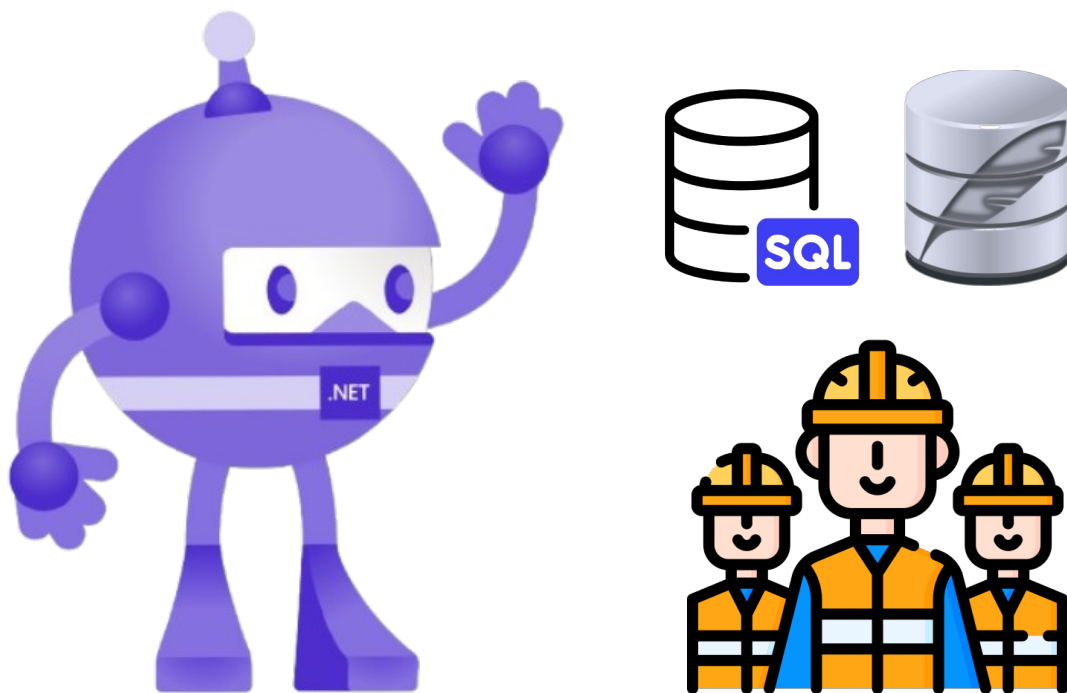


DESARROLLO DE INTERFACES

Interfaz con Bases de Datos Gestión Trabajadores



Marcos Zahonero

Índice

| | |
|---|-----------|
| Introducción..... | 4 |
| Tiempo utilizado..... | 5 |
| Material utilizado..... | 6 |
| Mi PC de casa..... | 6 |
| Objetivos de la práctica..... | 7 |
| Desarrollo de la Actividad..... | 8 |
| Página - MainPage..... | 8 |
| Añadir usuarios..... | 9 |
| Visual..... | 9 |
| Código..... | 9 |
| Error al no introducir nombre..... | 9 |
| Trabajador seleccionado..... | 10 |
| Visual..... | 10 |
| Código..... | 10 |
| Actualizar usuario..... | 11 |
| Visual..... | 11 |
| Código..... | 11 |
| Error al no introducir nombre..... | 11 |
| Error al no seleccionar un trabajador..... | 12 |
| Eliminar usuario..... | 12 |
| Visual..... | 12 |
| Código..... | 12 |
| Error al no seleccionar un trabajador..... | 13 |
| CollectionView (collectionDatabase)..... | 13 |
| Visual..... | 13 |
| Código..... | 14 |
| Resultado Final – Vista Móvil..... | 15 |
| Resultado Final – Vista Ordenador..... | 15 |
| GitHub..... | 16 |
| Conclusión..... | 17 |
| Problemas y/o sugerencias en la actividad..... | 18 |
| 1# - Mi extra a la actividad..... | 18 |
| Webgrafía..... | 19 |

Introducción

En esta actividad veremos como podemos incorporar a una interfaz en MAUI una base de datos, con su tabla, sus campos y sus datos, haremos que se vea el contenido desde la propia interfaz y facilitaremos la forma de modificar/borrar los datos mediante botones que programaremos para que funcione correctamente, avisándonos de avisos en caso de dar problemas en las consultas.

Tiempo utilizado

He utilizado 3 horas, las de clase del Martes para realizar la actividad y para hacer el documento y vídeos aproximadamente 1 hora más, en total serían 4 horas, esta actividad fue más accesible pero me gasto algo más de tiempo por dedicarle tiempo a la estructura de la interfaz y hacerla atractiva de ver.

Material utilizado

- El portátil de clase, con mi ordenador personal de casa.
- Visual Studio
- Páginas web buscando información

Mi PC de casa

| Componente | Nombre |
|-------------------|-----------------------------|
| Sistema Operativo | Windows 11 Pro |
| Procesador | AMD Ryzen 5 2600X |
| Tarjeta gráfica | NVIDIA GeForce GTX 1650 4GB |
| Memoria RAM | 16 GB |
| Placa base | B450M-A PRO MAX |
| Disco duro 1# | KIOXIA EXCERIA G2 SSD 1 TB |
| Disco duro 2# | ST1000DM010-2EP102 HDD 1 TB |

Objetivos de la práctica

- Realizar una interfaz con SQLite en MAUI
- Entender el funcionamiento de como conectar con la base de datos
- Entender el funcionamiento de los Binding y utilizarlos
- Diseñar una interfaz interesante de ver

Desarrollo de la Actividad

Esta aplicación consta de una única página, con los siguientes elementos:

- 8 Labels
- 2 Entrys
- 3 Buttons
- 1 CollectionView
- 2 ScrollViews

También hacemos uso de un **Grid** con 8 filas y 3 columnas para organizar la interfaz de la aplicación, he utilizado una forma de toda la gestión en la izquierda de la pantalla y a la derecha la base de datos, para así poder interactuar desde una misma vista y no tener que bajar ni nada para probar o usar esta interfaz.

Página – MainPage

Al iniciar la aplicación aparece así, con los Entrys vacíos y una vista actualizada de la base de datos actual:

Home

Añadir empleados

Nombre: Apellido:

Registrar usuario

Empleado seleccionado

Nombre:

Apellido:

Actualizar usuario

Eliminar usuario

Base de datos

Paco asd

Patat sd

Patata Hola

Patata Hola

AAAasd dd

asdsd d

asdas asd

Añadir usuarios

Para añadir usuarios a la base de datos tenemos 2 Entrys, donde sacamos el nombre y el apellido, al escribir y presionar el botón de “Registrar usuario” añadimos un nuevo registro a la base de datos, aunque hay una condición para que al menos este el nombre puesto ya que sino esta no hará nada y saldrá un aviso.

Visual

Añadir empleados

Nombre: Apellido:

Registrar usuario

Código

```
private void btnRegistrarUser_Clicked(object sender, EventArgs e)
{
    String query = "";
    String nombre = eNombre.Text;
    String apellido = eApellido.Text;

    if (eNombre.Text != "" && nombre != null)
    {
        query = "insert into Trabajador (nombre, apellidos) values ('" + nombre + "', '" + apellido + "')";
        ejecutarQuery(query);

        reloadDatabase();
    }
    else
    {
        lblErrores.Text = "¡Pon un nombre!";
    }
}
```

Error al no introducir nombre

Al ser la variable `nombre != null` saldrá el `lblErrores` mostrando texto como se ve:

¡Pon un nombre!

Base de datos

Patat sd

AAAasd dd

Trabajador seleccionado

Visual

Empleado seleccionado

Nombre:

Apellido:

Código

Al seleccionar una elemento de la lista se guardará automáticamente en la variable `selected`, que es un objeto de tipo `Trabajador`, en el cual podrás acceder a su nombre, id y apellido.

```
private Trabajador _selected;
4 referencias
public Trabajador selected
{
    get { return _selected; }
    set
    {
        _selected = value;
        OnPropertyChanged();
    }
}
```

```
<Label FontSize="Medium"
  x:Name="lblNombre"
  Text="Nombre: "
  HeightRequest="58"
  Grid.Column="0"
  Grid.Row="4"/>

<Label FontSize="Medium"
  x:Name="lblSeccNombre"
  Margin="18,0,0,0"
  Text="{Binding selected.Nombre}"
  HeightRequest="58"
  Grid.Column="0"/>
</HorizontalStackLayout>

<HorizontalStackLayout
  Grid.Row="5">

  <Label FontSize="Medium"
    x:Name="lblApellido"
    Text="Apellido: "

    Grid.Column="0"/>

  <Label FontSize="Medium"
    x:Name="lblSeccApellido"
    Margin="18,0,0,0"
    Text="{Binding selected.Apellidos}"
    Grid.Column="0"/>
</HorizontalStackLayout>
```

Actualizar usuario

Visual



Código

Al seleccionar un usuario como he dicho antes saco el Trabajador y gracias a eso puedo aquí saber su ID en la base de datos, hago el UPDATE con los valores de mis Entries y saco la id del selected para modificar el valor.

```
private void btnUpdateUser_Clicked(object sender, EventArgs e)
{
    if (eNombre.Text != "" && selected != null)
    {
        String query = "UPDATE Trabajador" + " SET nombre = \"" + eNombre.Text +
            "\", apellidos = \"" + eApellido.Text + "\" WHERE id = " + selected.Id;

        ejecutarQuery(query);

        lblErrores.Text = "";

        reloadDatabase();
    } else
    {
        if (selected == null)
        {
            lblErrores.Text = "¡Selecciona un usuario!";
        }
        else if (eNombre.Text == "")
        {
            lblErrores.Text = "¡Pon un nombre!";
        }
    }
}
```

Error al no introducir nombre

Al ser la variable `eNombre.Text == ""` saldrá el `lblErrores` mostrando texto como se ve:

¡Pon un nombre!

Base de datos

Patat sd

AAAasd dd

Error al no seleccionar un trabajador

Al ser la variable `selected == null` saldrá el `lblErrores` así:

¡Selecciona un usuario!

Base de datos

Manuel Garcia

Eliminar usuario

Visual

Eliminar usuario

Código

Seleccionamos de nuevo a un usuario y lo borramos sabiendo su ID como también dije antes.

```
private void btnDeleteUser_Clicked(object sender, EventArgs e)
{
    if (selected != null)
    {
        int idTrabajador = selected.Id;

        String query = "DELETE FROM Trabajador WHERE id = " + idTrabajador;
        ejecutarQuery(query);

        reloadDatabase();
        lblErrores.Text = "";
    } else
    {
        lblErrores.Text = "¡Selecciona un usuario!";
    }
}
```

Error al no seleccionar un trabajador

Al ser la variable `selected == null` saldrá el lblErrores así:

¡Selecciona un usuario!

Base de datos

Manuel Garcia

CollectionView (`collectionDatabase`)

Visual

Base de datos

Manuel Garcia

Manuel Garcia

Marcos Zasdadsa

Manuel Garcia

Oscar Garcia

Oscar Garcia

Código

Tiene un ScrollView para que puedas hacer bajar y subir en cuanto hayan muchos más trabajadores de lo esperado y así que no estropee el resto de interfaz.

Tiene un Binding que hace que se vuelve la información de la lista `OcTrabajadores` y cuando seleccionamos a un trabajador ocurre lo que decía antes, coge ese trabajador y lo establece en una variable del código gracias a:

```
SelectedItem="{Binding selected}"
```

El resto es un DataTemplate con StackLayout con 2 Labels para el nombre y apellido y ya está.

```
<ScrollView>
  <CollectionView ItemsSource="{Binding OcTrabajadores}"
    SelectionMode="Single"
    x:Name="collectionDatabase"
    HeightRequest="350"
    SelectedItem="{Binding selected}">
    <CollectionView.ItemTemplate>
      <DataTemplate>
        <StackLayout Orientation="Horizontal"
          Padding="10">
          <!-- Muestra nombre -->
          <Label Text="{Binding Nombre}"
            FontSize="Medium"
            VerticalOptions="Center"/>

          <!-- Muestra apellidos -->
          <Label Text="{Binding Apellidos}"
            FontSize="Medium"
            VerticalOptions="Center"
            Margin="10,0,0,0" />

        </StackLayout>
      </DataTemplate>
    </CollectionView.ItemTemplate>
  </CollectionView>
</ScrollView>
```

Resultado Final – Vista Móvil

Home

Añadir empleados

Nombre: Apellido:

Registrar usuario

Empleado

Nombre:

Apellido:

Actualizar usuario

Eliminar usuario

Base de datos

| | |
|--------|------|
| Paco | asd |
| Patat | sd |
| Patata | Hola |
| Patata | Hola |
| AAAasd | dd |
| asdsd | d |
| asdas | asd |

Resultado Final – Vista Ordenador

Home

Añadir empleados

Nombre: Apellido:

Registrar usuario

Empleado seleccionado

Nombre: Patata

Apellido: Hola

Actualizar usuario

Eliminar usuario

Base de datos

| | |
|--------|------|
| Paco | asd |
| Patat | sd |
| Patata | Hola |
| Patata | Hola |
| AAAasd | dd |
| asdsd | d |
| asdas | asd |

GitHub

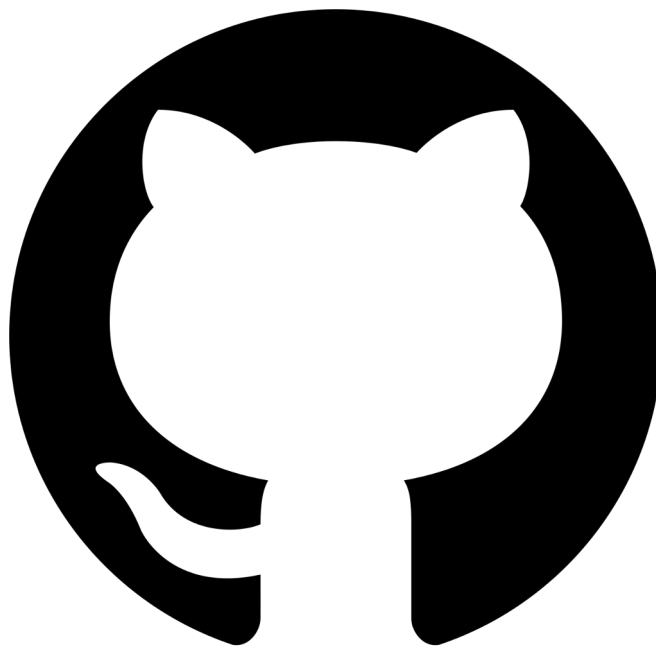


He cambiado el nombre del repositorio así que si has estado tenido problemas con mis actividades al corregir será por eso, en esta actividad está corregido para el resto del curso.



En esta actividad he añadido GitHub para añadir el proyecto completo, junto a una descripción en el [README.md](#) para añadir una breve explicación y preview de cada actividad.

Enlace: [Maek0s/2DAM_DesarrolloInterfaces](#)



Conclusión

En conclusión, no se si es porque tal vez ya me he acostumbrado al entorno y a los errores que suele dar pero he tenido mayor agilidad para entender esto aunque hayan sido nuevos conceptos como son las bases de datos en nuestro caso, fue algo interesante y accesible de entender.

Problemas y/o sugerencias en la actividad

1# – Mi extra a la actividad

Aquí estamos de nuevo, 2 vídeos, uno de su funcionamiento, añadiendo, actualizando, borrando y viendo los posibles errores que da y como reacciona la interfaz para avisarte y poco más, el otro vídeo de explicativo se basará sobre todo en el código principal donde hablaré de como he gestionado para seleccionar el usuario y así poder hacer las funciones que solicita el ejercicio.



Vídeo funcionamiento: <https://youtu.be/ITq8Z3srrXs>

Video explicativo de código: <https://youtu.be/8uUpV4PZrbg>

Webgrafía

- <https://visualstudio.microsoft.com/es/vs/community/>
- <https://dotnet.microsoft.com/es-es/apps/maui>

- **CollectionsView:**

<https://learn.microsoft.com/es-es/dotnet/maui/user-interface/controls/collectionview/>