



OBJETIVOS:

- Aprender los conceptos de diseño del entorno, decisiones y timers en Net Maui
- Usar de forma correcta los operadores lógicos algunas funciones y técnicas para trabajar con Controles, decisiones y números aleatorios

MATERIAL:

Visual studio

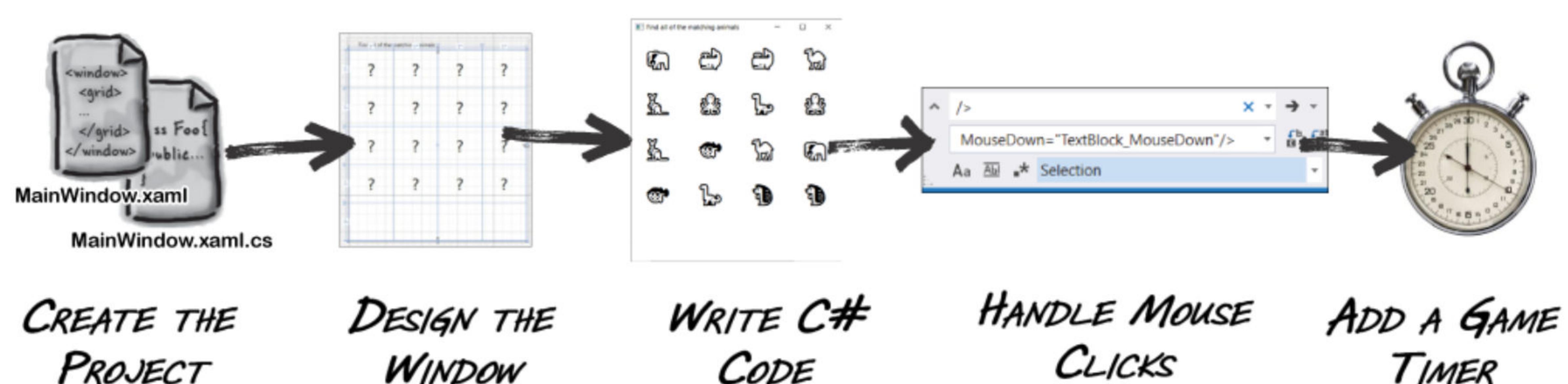
Computadora

TEMA DE LA PRÁCTICA: MatchGameSuCarnet

INTRODUCCIÓN TEÓRICA:

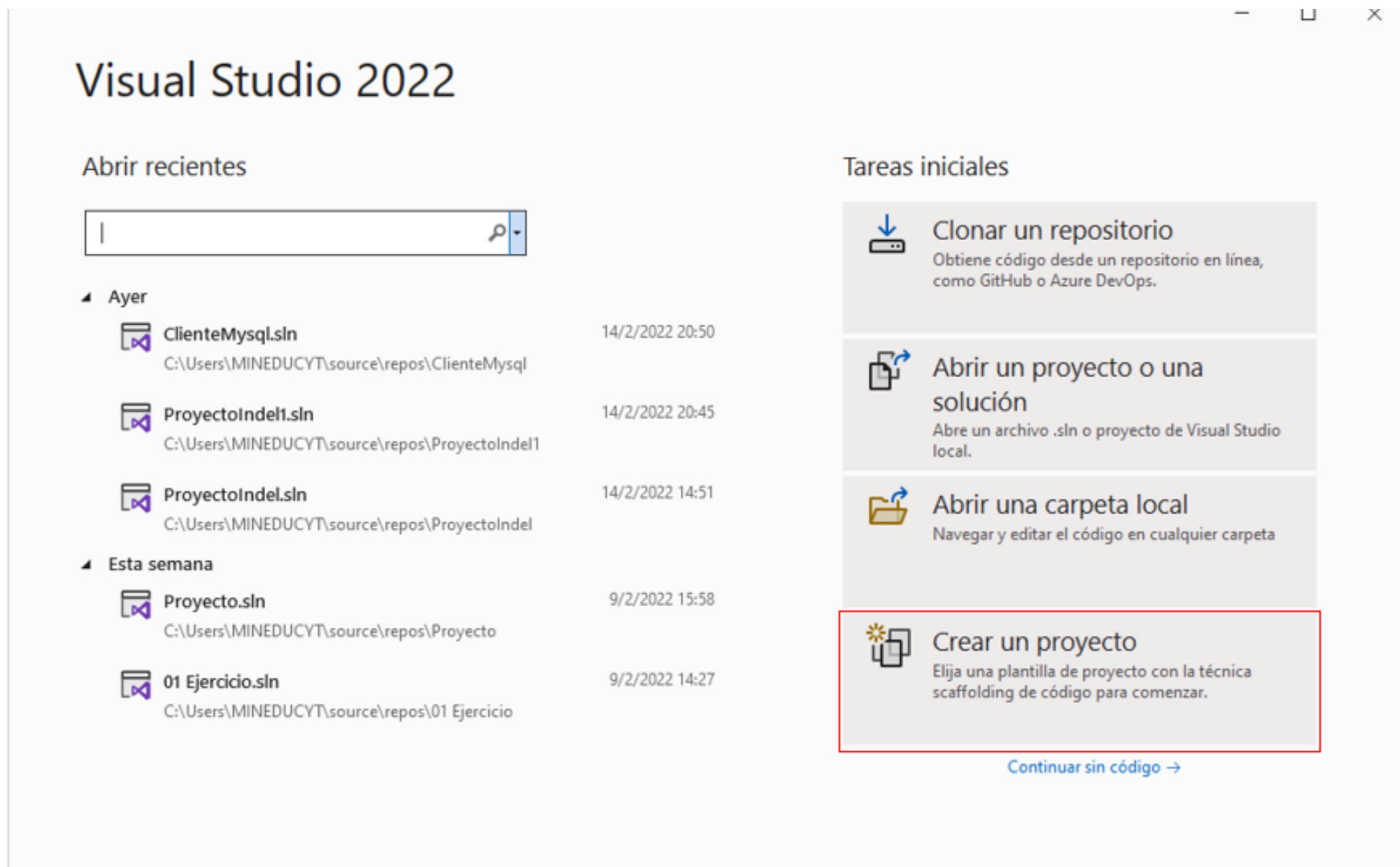
Después de haber estudiado los conceptos de xaml en las prácticas anteriores,nos introduciremos en la realización de un juego simple que nos dará la lógica para seguir aprendiendo este maravilloso lenguaje

Antes de hacer el programa necesitamos repasar qué es lo que vamos a crear, para ello mira la siguiente imagen:





1. Abre visual studio y crea un nuevo proyecto

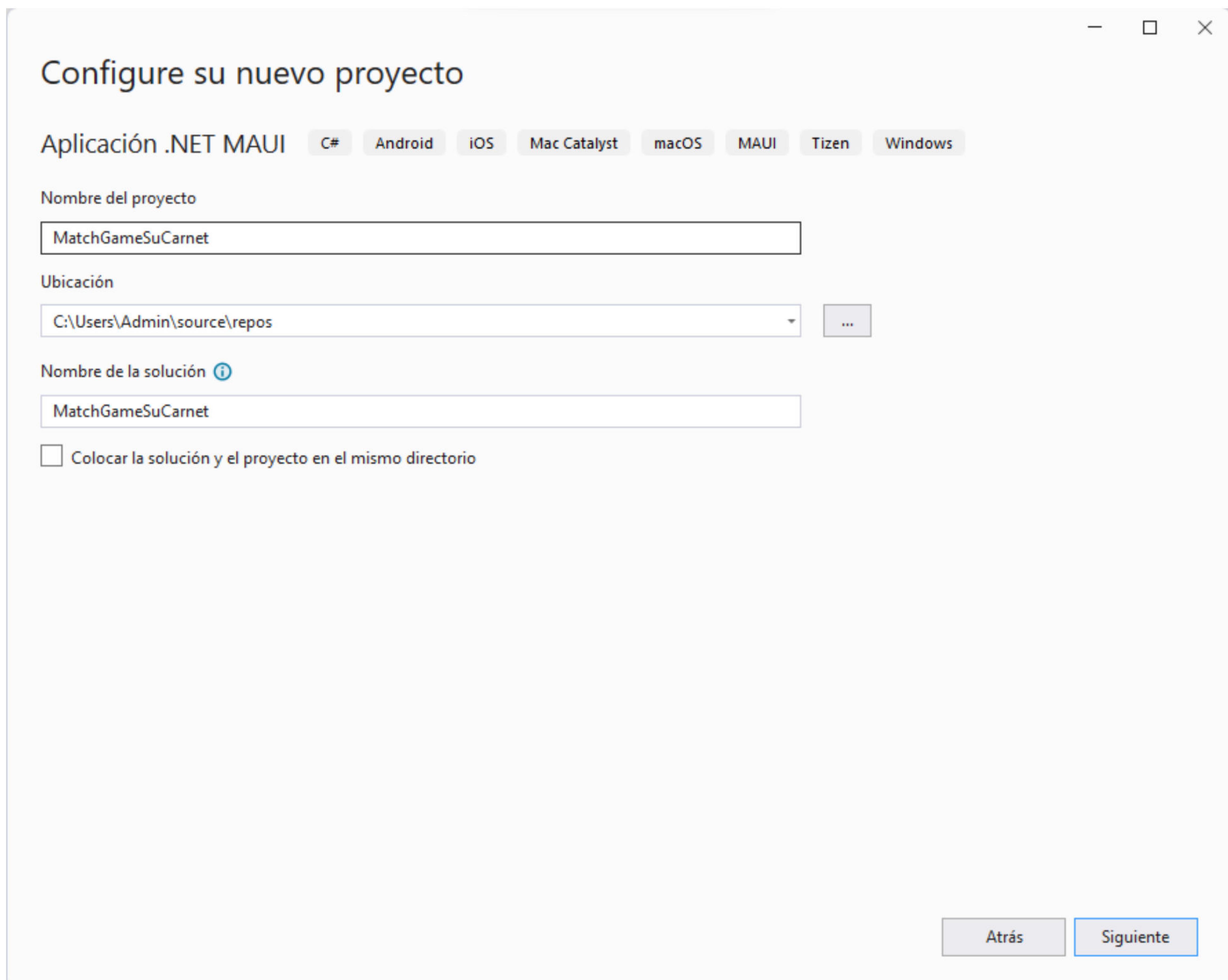




2. Selecciona Aplicación .NET MAUI y luego clic en siguiente.



3. Escribe el nombre del programa, MatchGameSuCarnet



4. Entra al diseño XAML y en el archivo **MainPage.xaml** digita lo siguiente:

```
<?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="MatchGameSuCarnet.MainPage"

    Shell.NavBarIsVisible="False"
```

INSTITUTO NACIONAL CANTON LOURDES
DOCENTE:FRANCISCO CANALES
PRÁCTICAS DE LABORATORIO



Title="Match Game Indel"

>

<ScrollView>

<VerticalStackLayout

Spacing="25"

Padding="30,0"

VerticalOptions="Center" >

<Label Text="Match Game Indel" x:Name="lblPrincipal" FontSize="Large"
HorizontalOptions="Center" />

<Grid x:Name="Grid1">

<Grid.ColumnDefinitions>

<ColumnDefinition/>

<ColumnDefinition/>

<ColumnDefinition/>

<ColumnDefinition/>

</Grid.ColumnDefinitions>

<Grid.RowDefinitions>

<RowDefinition/>

<RowDefinition/>

<RowDefinition/>

<RowDefinition/>



```
<RowDefinition/>

</Grid.RowDefinitions>

<Button Text="?" FontSize="36" Clicked="Button_Clicked"/>

<Button Text="?" FontSize="36" Grid.Column="1" Clicked="Button_Clicked"/>

<Button Text="?" FontSize="36" Grid.Column="2" Clicked="Button_Clicked"/>

<Button Text="?" FontSize="36" Grid.Column="3" Clicked="Button_Clicked"/>

<Button Text="?" FontSize="36" Grid.Row="1" Clicked="Button_Clicked"/>

    <Button Text="?" FontSize="36" Grid.Row="1" Grid.Column="1"
Clicked="Button_Clicked"/>

    <Button Text="?" FontSize="36" Grid.Row="1" Grid.Column="2"
Clicked="Button_Clicked"/>

    <Button Text="?" FontSize="36" Grid.Row="1" Grid.Column="3"
Clicked="Button_Clicked"/>

    <Button Text="?" FontSize="36" Grid.Row="2" Clicked="Button_Clicked"/>

    <Button Text="?" FontSize="36" Grid.Row="2" Grid.Column="1"
Clicked="Button_Clicked"/>

    <Button Text="?" FontSize="36" Grid.Row="2" Grid.Column="2"
Clicked="Button_Clicked"/>

    <Button Text="?" FontSize="36" Grid.Row="2" Grid.Column="3"
Clicked="Button_Clicked"/>

    <Button Text="?" FontSize="36" Grid.Row="3" Clicked="Button_Clicked"/>

    <Button Text="?" FontSize="36" Grid.Row="3" Grid.Column="1"
Clicked="Button_Clicked"/>

    <Button Text="?" FontSize="36" Grid.Row="3" Grid.Column="2"
Clicked="Button_Clicked"/>

    <Button Text="?" FontSize="36" Grid.Row="3" Grid.Column="3"
Clicked="Button_Clicked"/>
```



</Grid>

</VerticalStackLayout>

</ScrollView>

</ContentPage>

5. Ahora que tenemos que crear los métodos del juego, para ello digita lo siguiente en el **MainPage.xaml.cs**

namespace MatchGameSuCarnet;

public partial class MainPage : ContentPage

{

public MainPage()

{

InitializeComponent();

SetUpGame();



}

```
private void SetUpGame()
{
    List<string> animalEmoji = new List<string>()

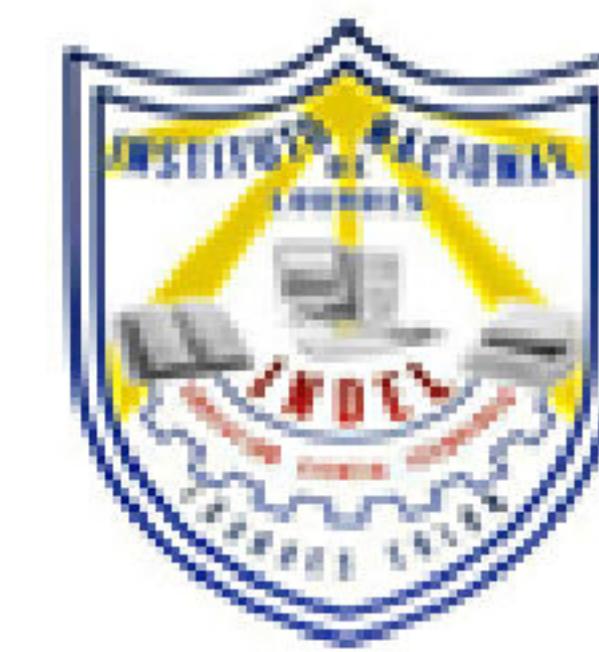
    {
        "🐶", "🐶",
        "🐵", "🐵",
        "🐸", "🐸",
        "🐘", "🐘",
        "🦓", "🦓",
        "🦒", "🦒",
        "🐍", "🐍",
        "🐬", "🐬",
    };

    Random random = new Random();

    foreach (Button view in Grid1.Children)
    {

        int index = random.Next(animalEmoji.Count);

        string nextEmoji = animalEmoji[index];
```



```
view.Text = nextEmoji;

animalEmoji.RemoveAt(index);

}

}

Button ultimoButtonClicked;

bool encontrandoMatch = false;

private void Button_Clicked(object sender, EventArgs e)

{

    Button button = sender as Button;

    if (encontrandoMatch == false)

    {

        button.Visible = false;

        ultimoButtonClicked = button;

        encontrandoMatch = true;

    }

    else if (button.Text == ultimoButtonClicked.Text)

    {

        button.Visible = false;

        encontrandoMatch = false;

    }

}

else
```



```
{  
    ultimoButtonClicked.isVisible = true;  
  
    encontrandoMatch = false;  
  
}  
  
}  
  
}
```

Investigación complementaria:

- 1.Agregue a la aplicación la funcionalidad de que las figuras sean aleatorias
- 2.Agregue a la aplicación la capacidad de tener un timer para que la aplicación cuente el tiempo que tarda en resolver el juego
- 3.Agrega a la aplicación la capacidad de reiniciar una vez haya finalizado la partida
- 4.Comente el programa con lo que hace cada linea de codigo, luego suba su aplicación a Github

Análisis de resultados(EN EQUIPO COLABORADORES GITHUB):

A continuación se le presentan 6 aplicaciones, cada uno de los participantes deberá programar una aplicación, luego subir a github y posteriormente enlazar los 6 aplicaciones por medio de un menú y colaboración en github.

- 1.Cree una aplicación en NET MAUI que muestre los números pares del 0 al 100
- 2.Cree una aplicación en NET MAUI que muestre una cuenta regresiva de 10 a 1
- 3.Cree una aplicación en NET MAUI que muestre las tablas de multiplicar del 1 al 10



4.Cree una aplicación en NET MAUI que determine el sueldo semanal de un trabajador con base a las horas que trabaja y el pago por hora que recibe.

5.“La langosta ahumada” es una empresa dedicada a ofrecer banquetes; sus tarifas son las siguientes: el costo de platillo por persona es de \$95.00, pero si el número de personas es mayor a 200 pero menor o igual a 300, el costo es de \$85.00. Para más de 300 personas el costo por platillo es de \$75.00. Se requiere una aplicación en MAUI que ayude a determinar el presupuesto que se debe presentar a los clientes que deseen realizar un evento.

6.Una empresa importadora desea determinar cuántos dólares puede adquirir con equis cantidad de dinero mexicano. Realice una aplicación en NET MAUI para tal fin.

SCRUM:

Llenar la plantilla scrum acorde a sus aplicaciones y roles.

NOMBRE DEL PROYECTO: UNA ESTRELLA

1.0 VISION DEL PROYECTO

2.0 SCRUM MASTER Y STAKEHOLDER

3.0 EQUIPO SCRUM

NOMBRE	ROL

4.0 DESARROLLO DE EPICAS

Nº	HISTORIA DE USUARIO
H1	
H2	
H3	
H4	

5.0 BACKLOG DE TAREAS

Nº	HISTORIA	ENCARGADO	TIEMPO ESTIMADO



6.0 IMPLEMENTACIÓN EN AZURE

- PROYECTO
- BACKLOG

7.0 ORDEN DE EXPLICACIÓN DEL PROYECTO:

EQUIPO	# DE PARTICIPACION

Bibliografía:

Retomado del libro Head first C# 2022

[Head First C#, 4th Edition \[Book\] \(oreilly.com\)](#)

INSTITUTO NACIONAL CANTON LOURDES
DOCENTE:FRANCISCO CANALES
PRÁCTICAS DE LABORATORIO

