

What's new in MAUI .Net 9: Focus on the HybridWebView Control

Marco Bortolin

Sponsor



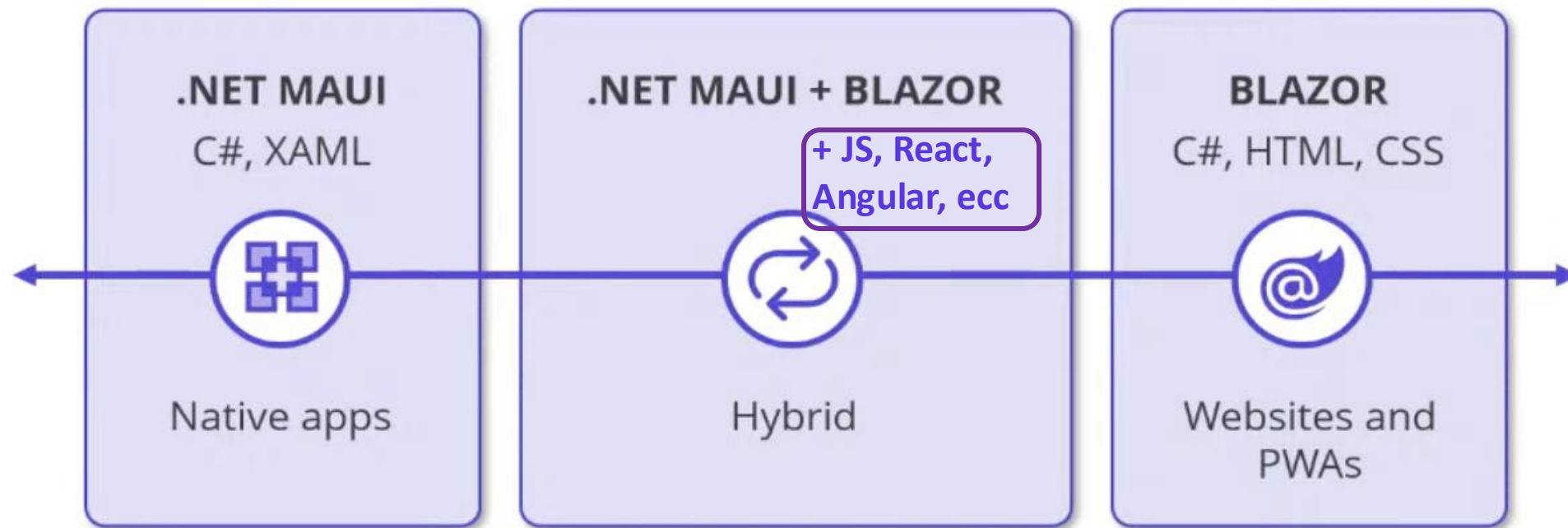
.NET MAUI in .NET 9

- HybridWebView
- TitleBar (Windows)
- Syncfusion ToolKit
- Miglioramenti a controlli esistenti
- Upgrade .NET 8 -> .NET9

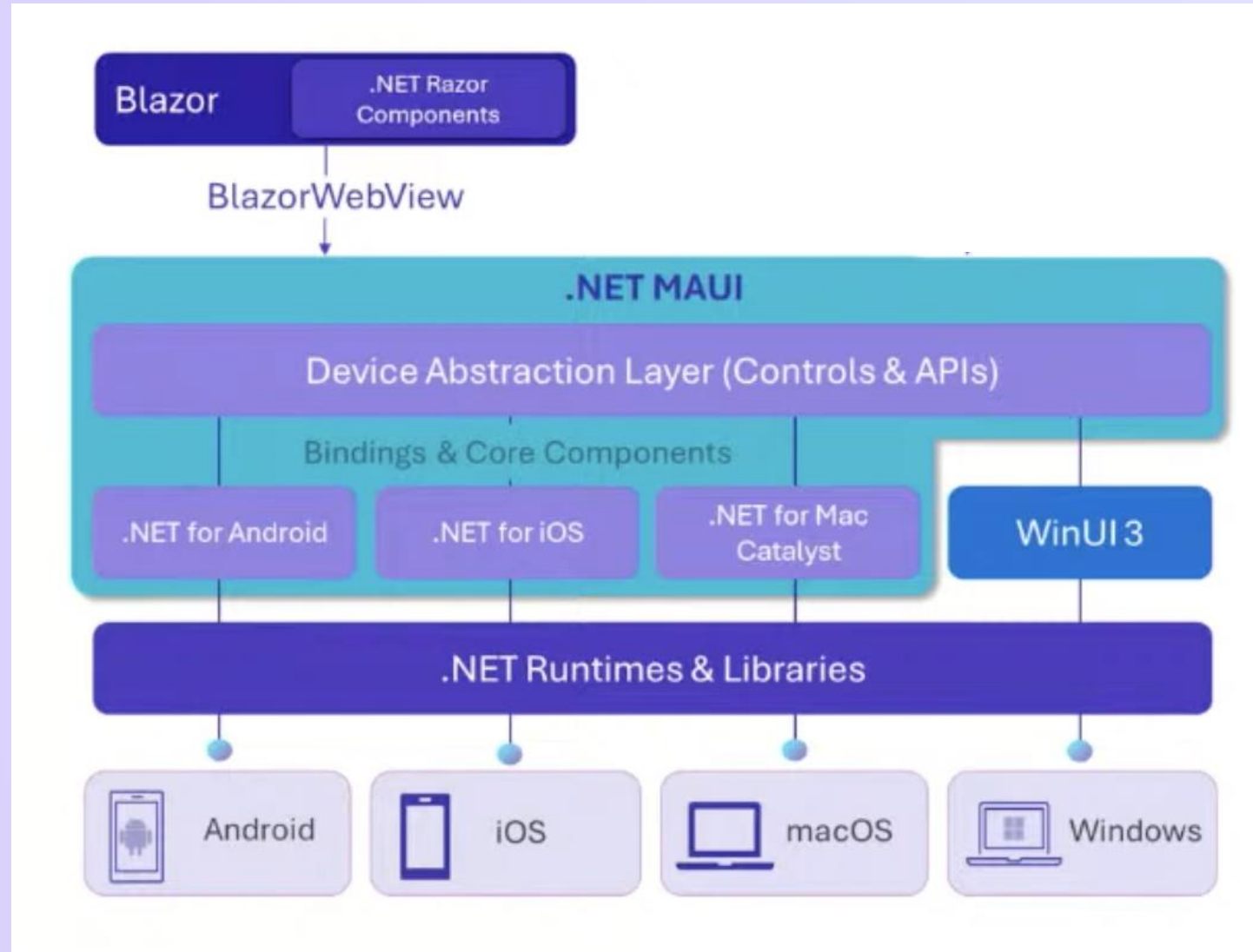


MAUI .NET 9

.NET offers choice for client app developers

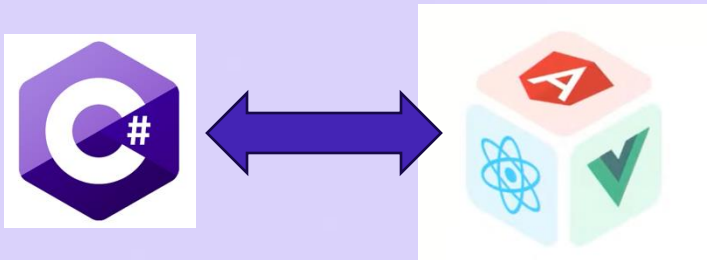


MAUI HYBRID



HybridWebView

- Hosting di contenuto HTML/JS/CSS arbitrario
- Comunicazione tra il codice JavaScript e il codice nativo (C#/.NET)

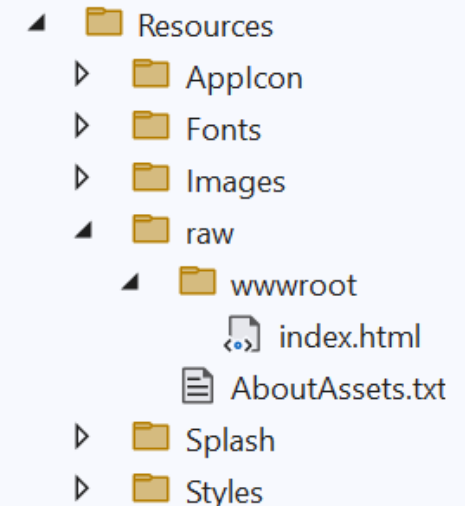


```
<ContentPage xmlns="http://schemas.microsoft.com/dotnet/2021/maui"
              xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
              x:Class="MauiApp3.MainPage">
    <HybridWebView />
</ContentPage>
```

Il contenuto Web dell'app deve essere incluso come parte di un progetto MAUI .NET come asset non elaborato.

Per un valore predefinito HybridWebView, il contenuto Web deve essere inserito nella cartella **Resources\Raw\wwwroot** , con il file principale denominato **index.html**.

<https://learn.microsoft.com/en-us/dotnet/maui/user-interface/controls/hybridwebview?view=net-maui-9.0>



Demo



HybridWebView – Comunicazione tramite messaggi

```
<Grid RowDefinitions="Auto,*"
      ColumnDefinitions="*">
  <Button Text="Send message to JavaScript"
          Clicked="OnSendMessageButtonClicked" />
  <HybridWebView x:Name="hybridWebView"
                 RawMessageReceived="OnHybridWebViewRawMessageReceived"
                 Grid.Row="1" />
</Grid>
```

```
private void OnSendMessageButtonClicked(object sender, EventArgs e)
{
    hybridWebView.SendRawMessage($"Hello from C#!");
}

private async void OnHybridWebViewRawMessageReceived(object sender, HybridWebViewRawMessageReceivedEventArgs e)
{
    await DisplayAlert("Raw Message Received", e.Message, "OK");
}
```


HybridWebView – index.html

```
<!DOCTYPE html>

<html lang="en" xmlns="http://www.w3.org/1999/xhtml">
<head>
  <meta charset="utf-8" />
  <title></title>
  <link rel="icon" href="data:,">
  <script src="scripts/HybridWebView.js"></script>
  <script>
    window.addEventListener(
      "HybridWebViewMessageReceived",
      function (e) {
        var messageFromCSharp = document.getElementById("messageFromCSharp");
        messageFromCSharp.value += '\r\n' + e.detail.message;
      });
  </script>
</head>
<body>
  <h1>HybridWebView app!</h1>
  <div>
    <button onclick="window.HybridWebView.SendRawMessage('Message from JS!')">Send message to C#</button>
  </div>
  <div>
    Messages from C#: <textarea readonly id="messageFromCSharp" style="width: 80%; height: 300px;"></textarea>
  </div>
</body>
</html>
```

Demo



HybridWebView – JavaScript from C#

JS

JavaScript

```
function AddNumbers(a, b) {  
    return a + b;  
}
```



C#

```
double x = 123d;  
double y = 321d;  
  
double result = await hybridWebView.InvokeJavaScriptAsync<double>(   
    "AddNumbers", // JavaScript method name  
    HybridSampleJSContext.Default.Double, // JSON serialization info for return type  
    [x, y], // Parameter values  
    [HybridSampleJSContext.Default.Double, HybridSampleJSContext.Default.Double]); // JSON serialization info
```

```
[JsonSourceGenerationOptions(WriteIndented = true)]  
[JsonSerializable(typeof(double))]  
internal partial class HybridSampleJsContext : JsonSerializerContext  
{  
    // This type's attributes specify JSON serialization info to preserve type structure  
    // for trimmed builds.  
}
```

```
string result = await webView.EvaluateJavaScriptAsync("document.getElementById('searchTxt').value");
```

Demo



HybridWebView - C# from JavaScript

JavaScript

```
await window.HybridWebView.InvokeDotNet('DoSyncWork');  
await window.HybridWebView.InvokeDotNet('DoSyncWorkParams', [123, 'hello']);  
const retValue = await window.HybridWebView.InvokeDotNet('DoSyncWorkReturn');  
const retValue = await window.HybridWebView.InvokeDotNet('DoSyncWorkParamsReturn', [123, 'hello']);
```

Demo



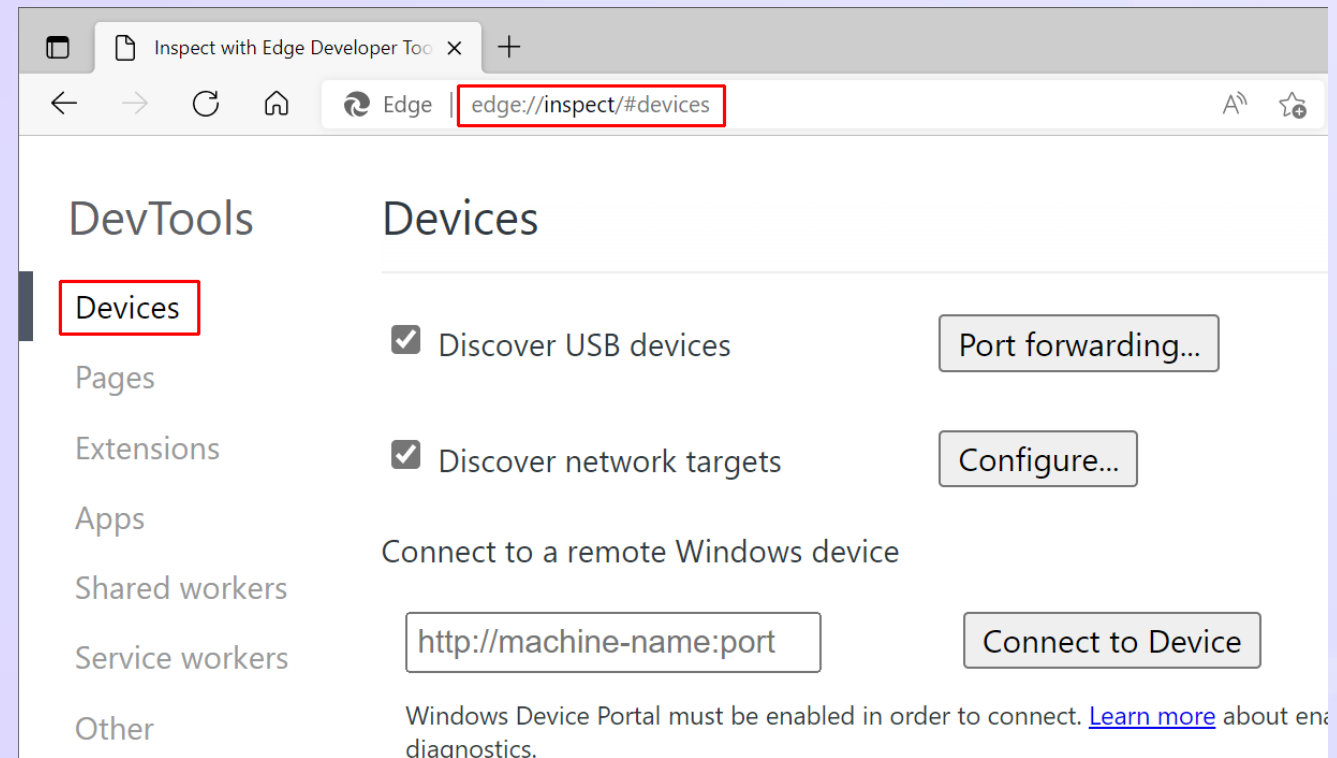
Debug – Dev Tools

```
var builder = MauiApp.CreateBuilder();
builder
    .UseMauiApp<App>()
    .ConfigureFonts(fonts =>
    {
        fonts.AddFont("OpenSans-Regular.ttf", "OpenSans-Regular");
        fonts.AddFont("OpenSans-Semibold.ttf", "OpenSans-Semibold");
    });

#if DEBUG
builder.Services.AddHybridWebViewDeveloperTools();
builder.Logging.AddDebug();
#endif

// Register any app services on the IServiceCollection

return builder.Build();
```



Remote Target #LOCALHOST

sdk_gphone_x86_64 #EMULATOR-5554

WebView in com.companyname.mauiblazor (83.0.4103.106) [trace](#)

MAUIBlazor https://0.0.0.0/
at (0, 66) size 1080 × 2142

[inspect](#) [pause](#)

<https://learn.microsoft.com/en-us/aspnet/core/blazor/hybrid/developer-tools?view=aspnetcore-9.0>

Demo



TitleBar

ⓘ Nota

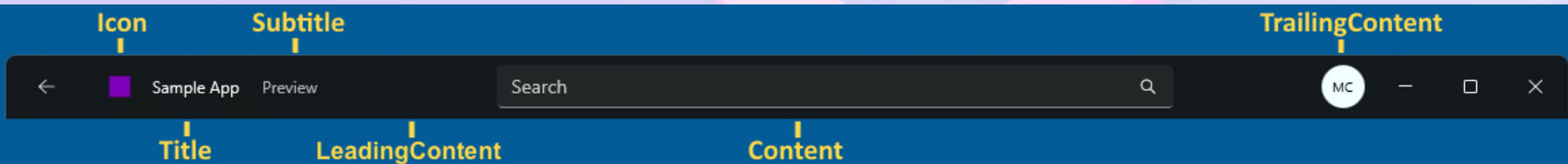
Il supporto di Mac Catalyst per il TitleBar controllo verrà aggiunto in una versione futura.

Il controllo **TitleBar** consente di aggiungere una barra del titolo personalizzata all'app in Windows

E' altamente personalizzabile tramite le relative proprietà:

- **LeadingContent**
- **Content**
- **TrailingContent**

```
<Window.TitleBar>
  <TitleBar x:Name="TeamsTitleBar"
    Title="Hello World"
    Icon="appicon.png"
    HeightRequest="46">
    <TitleBar.Content>
      <SearchBar Placeholder="Search"
        PlaceholderColor="White"
        MaximumWidthRequest="300"
        HorizontalOptions="Fill"
        VerticalOptions="Center" />
    </TitleBar.Content>
  </TitleBar>
</Window.TitleBar>
```



Syncfusion Toolkit



**Syncfusion Toolkit
for .NET MAUI**

14 free, open-source controls available for .NET MAUI!

Open-source on github.com/syncfusion/maui-toolkit

Package Syncfusion.Maui.Toolkit available on NuGet

 Syncfusion

<https://devblogs.microsoft.com/dotnet/dotnet-maui-welcomes-syncfusion-open-source-contributions/>

BackButtonBehavior

La modalità di binding per le proprietà **IsVisible** e **IsEnabled** adesso non è più in modalità **BindingMode.OneTime** ma **BindingMode.OneWay**

```
<ContentPage ...>
  <Shell.BackButtonBehavior>
    <BackButtonBehavior Command="{Binding BackCommand}"
                        IsVisible="{Binding IsBackButtonVisible}"
                        IconOverride="back.png" />
  </Shell.BackButtonBehavior>
  ...
</ContentPage>
```



Si controlla più facilmente il comportamento del pulsante indietro in fase di esecuzione: soprattutto in MVVM

CollectionView and CarouselView

Due nuovi handler **opzionali** per iOS e Mac Catalyst:

- `Microsoft.Maui.Controls.Handlers.Items2.CollectionViewHandler2`
- `Microsoft.Maui.Controls.Handlers.Items2.CarouselViewHandler2`

```
#if IOS || MACCATALYST
builder.ConfigureMauiHandlers(handlers =>
{
    handlers.AddHandler<Microsoft.Maui.Controls.CollectionView, Microsoft.Maui.Controls.Handlers.Items2.CollectionViewHandler2>();
    handlers.AddHandler<Microsoft.Maui.Controls.CarouselView, Microsoft.Maui.Controls.Handlers.Items2.CarouselViewHandler2>();
});
#endif
```

Sfruttano le API native di UICollectionView

Miglioramenti in termini di **prestazioni e stabilità**

Altre novità

- **ContentPage:** La proprietà `HideSoftInputOnTapped` è ora supportata su Mac Catalyst, Android e iOS
- **Supporto tastiera virtuale:** Nuovo supporto per **Password**, **Data** e **Ora** su Editor e Entry

```
<Entry Keyboard="Date" />
```

- **Allineamento testo:** Aggiunto il membro **Justify** per allineare il testo nei controlli

```
<Label Text="Lorem ipsum dolor sit amet, consectetur adipiscing elit. In facilisis nulla eu felis fringilla vulputate."  
HorizontalTextAlignment="Justify"/>
```

- **Multi-window:** possibilità di portare una finestra specifica in primo piano su Mac Catalyst e Windows

```
Application.Current?.ActivateWindow(windowToActivate);
```

Upgrade from .NET 8 to .NET 9

Modificare il project file (.csproj) aggiornando tutti i TFM (*Target Framework Monikers*)

.NET 8

```
<TargetFrameworks>net8.0-android;net8.0-ios;net8.0-maccatalyst;net8.0-tizen</TargetFrameworks>  
<TargetFrameworks Condition="$([MSBuild]::IsOSPlatform('windows'))">$(TargetFrameworks);net8.0-windows10.0.19041.0</TargetFrameworks>
```

.NET 9

```
<TargetFrameworks>net9.0-android;net9.0-ios;net9.0-maccatalyst;net9.0-tizen</TargetFrameworks>  
<TargetFrameworks Condition="$([MSBuild]::IsOSPlatform('windows'))">$(TargetFrameworks);net9.0-windows10.0.19041.0</TargetFrameworks>
```

Aggiornare direttamente alla versione .NET 9 i pacchetti NuGet del progetto e/o la variabile di build \$(MauiVersion).

```
<Project Sdk="Microsoft.NET.Sdk">  
  <PropertyGroup>  
    <TargetFrameworks>net8.0-android;r  
    <UseMaui>True</UseMaui>  
    <MauiVersion>8.0.3</MauiVersion>  
  ...
```

```
<ItemGroup>  
  <PackageReference Include="Microsoft.Maui.Controls" Version="$(MauiVersion)" />  
  <PackageReference Include="Microsoft.Maui.Controls.Compatibility" Version="$(MauiVersion)" />  
</ItemGroup>
```

Obsoleta

Deployment targets

	.NET MAUI 8	.NET MAUI 9
Requisiti minimi di distribuzione	Android 5.0 (API 21) o versione successiva iOS 11 o versione successiva macOS 11 o versione successiva usando Mac Catalyst Windows 10 versione 1809 o successiva usando WinUI 3	Android 5.0 (API 21) o versione successiva iOS 12.2 o versione successiva macOS 12.0 o versione successiva usando Mac Catalyst 15 Windows 10 versione 1809 o successiva usando WinUI 3
Compatibilità con Xcode	Xcode 13 o versione successiva	Xcode 16 o versione successiva
Supporto per Blazor	Android 7.0 (API 24) o versione successiva iOS 14 o versione successiva macOS 11 o versione successiva usando Mac Catalyst	Android 7.0 (API 24) o versione successiva iOS 14 o versione successiva macOS 11 o versione successiva usando Mac Catalyst

Upgrade ?

Release schedule, support & patching



- **.NET MAUI ships on the same cadence as .NET**
- **.NET MAUI is an optional .NET workload with external dependencies (such as Xcode & Android SDK tools)**
- **A major version of .NET MAUI receives patches for 6 months after a successor (major version) ships**
- **Xamarin SDKs support ended May 2024**

Resources

- <https://github.com/bortolin/NetSat2025Pn.MauiExample>
- <https://github.com/dotnet/maui-samples/tree/main/9.0/UserInterface/Views/HybridWebViewDemo>
- <https://learn.microsoft.com/en-us/dotnet/maui/user-interface/controls/hybridwebview?view=net-maui-9.0>
- <https://learn.microsoft.com/en-us/dotnet/maui/whats-new/dotnet-9?view=net-maui-9.0>

Q&A



Marco Bortolin

email: m.bortolin@hunext.com

twitter: [@marcobortolin](https://twitter.com/marcobortolin)

<https://github.com/bortolin>

<https://www.linkedin.com/in/marcobortolin>

