

Let's turn your website into
a hybrid .NET MAUI app

Mark Allibone

iDIP Solution

@mallibone



Hello my name is Mark

 Mark Allibone

 iDIP Solution

 @mallibone.com

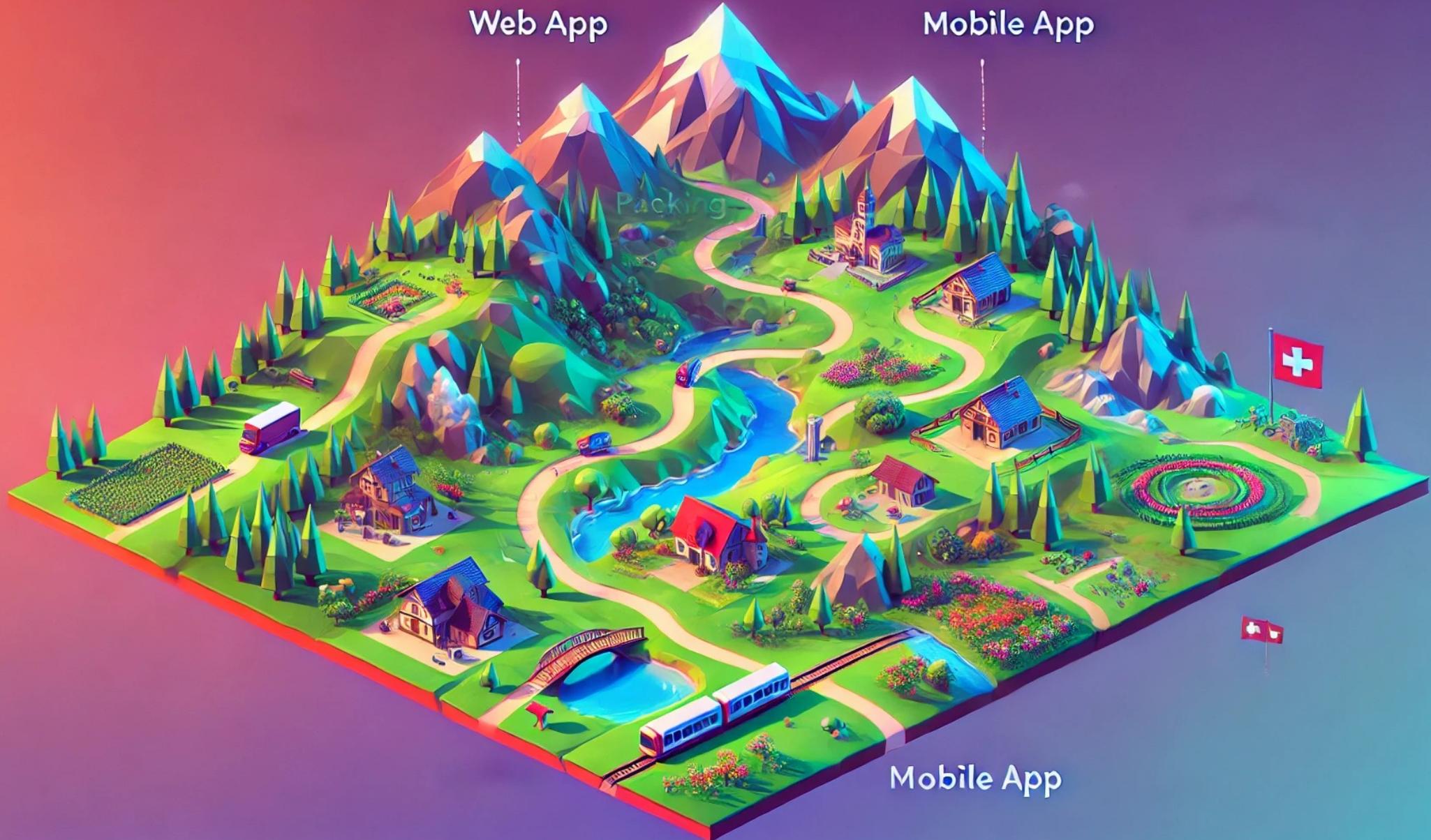
 <https://mallibone.com>

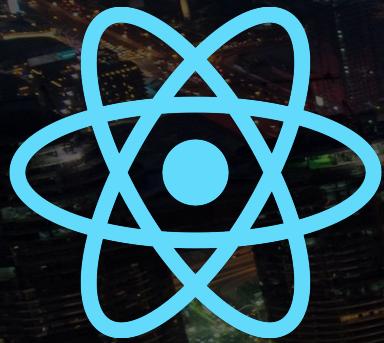
 <https://nullpointers.io>



Web App

Mobile App





JS

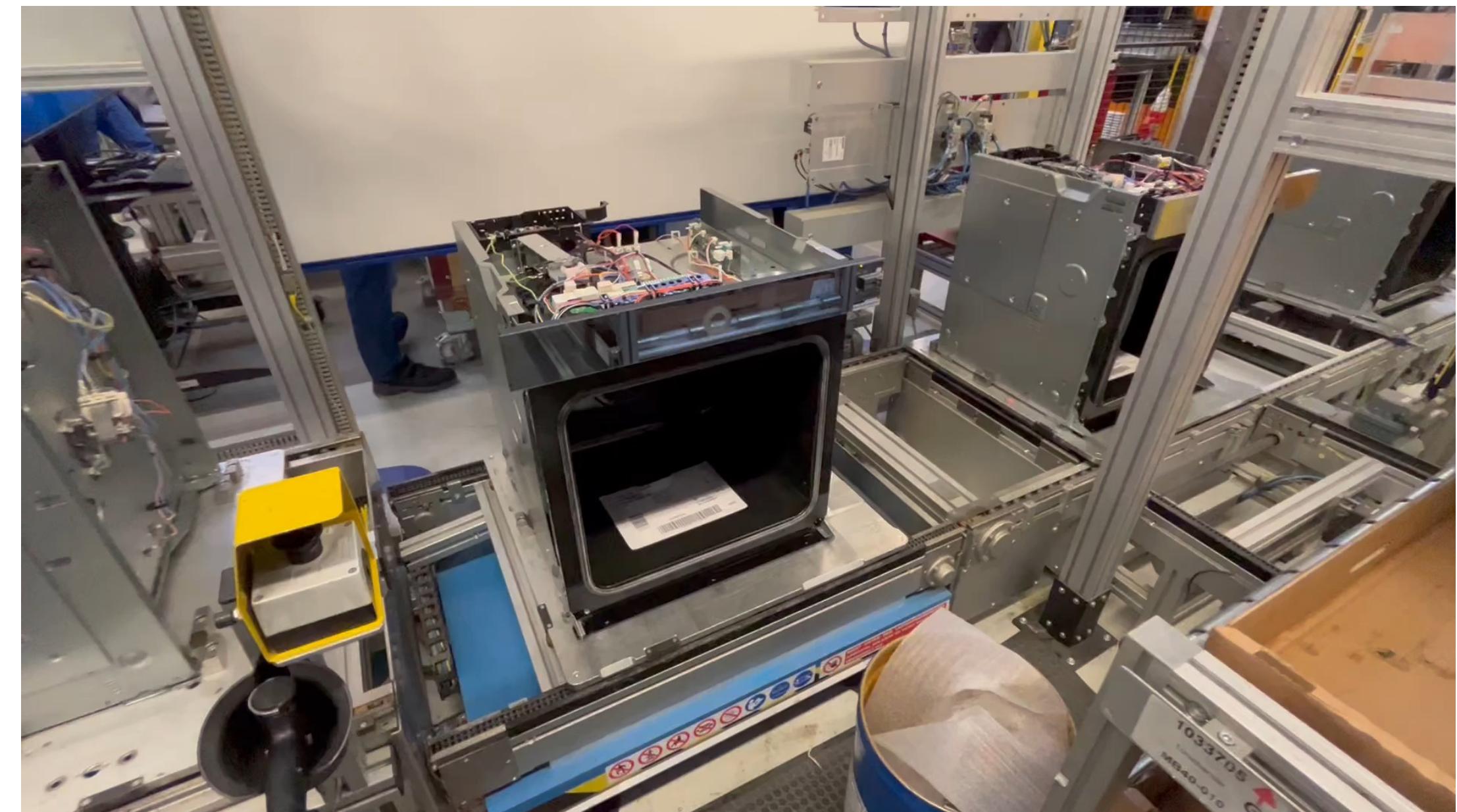
This Photo by Unknown Author is
licensed under [CC BY-SA](#)



[This Photo](#) by Unknown Author is licensed under [CC BY-SA-NC](#)



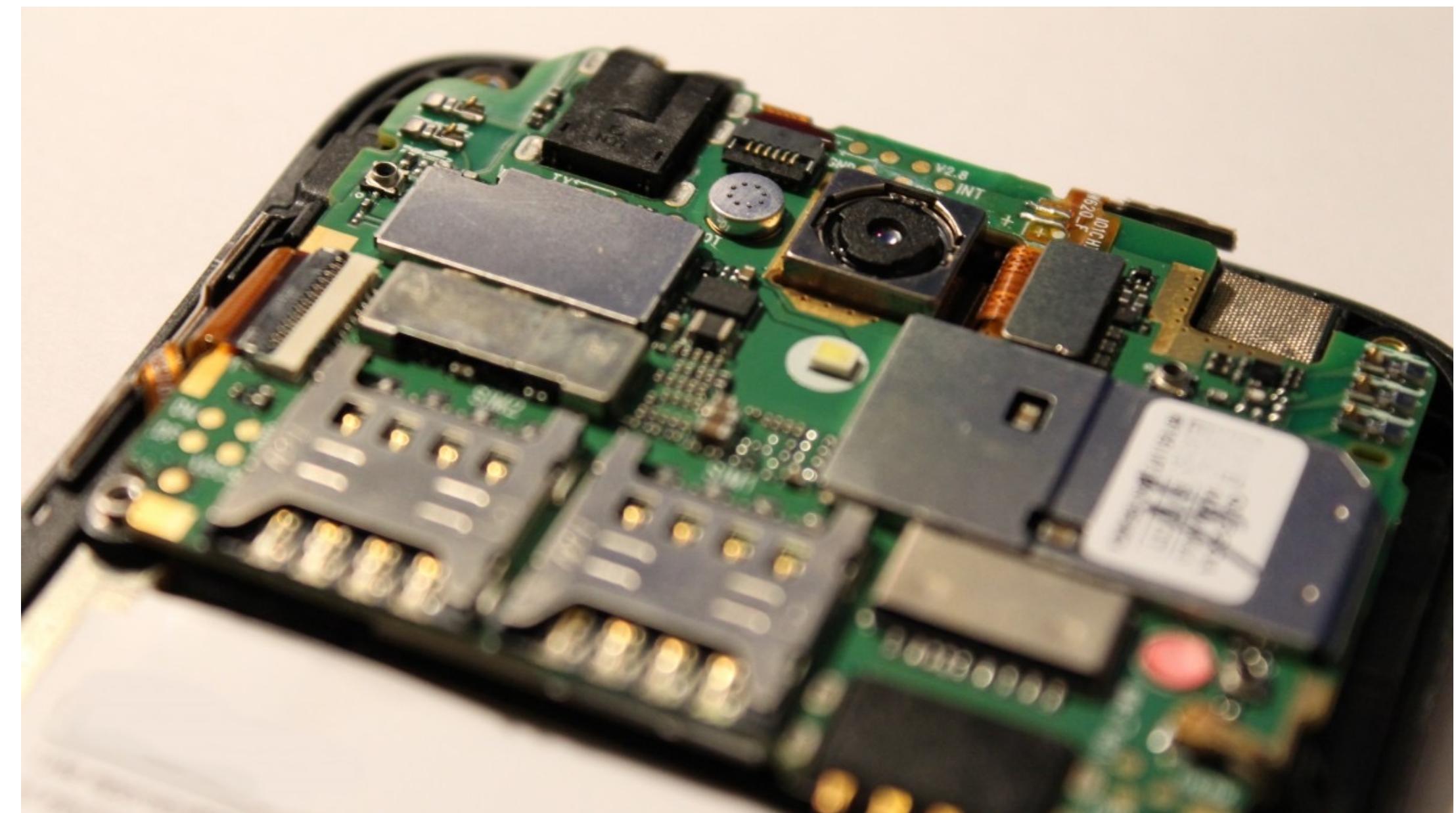
@mallibone

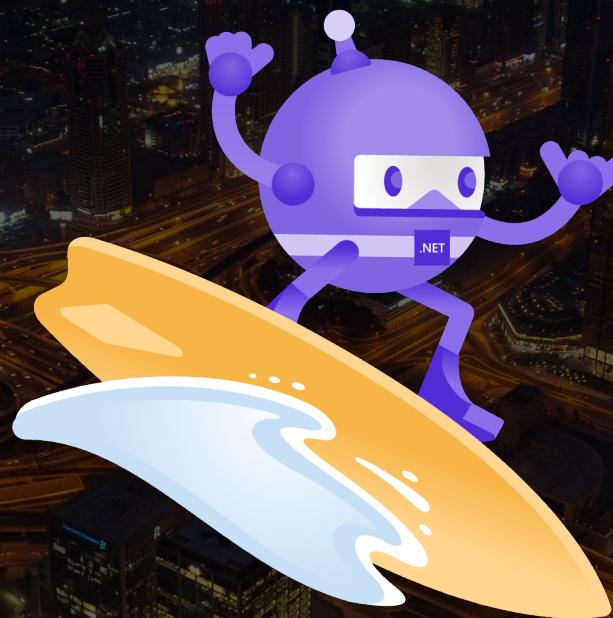




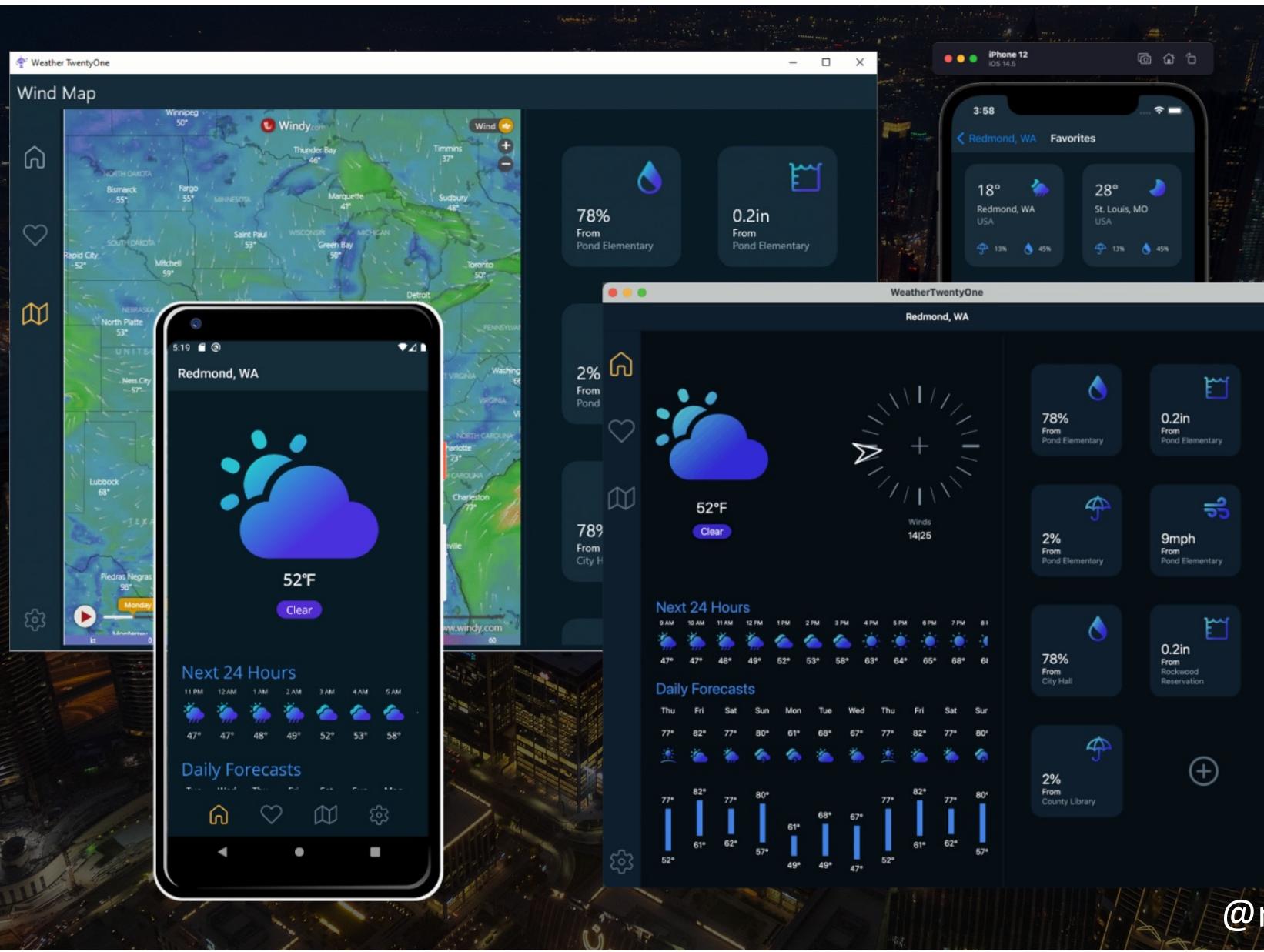
[This Photo](#) by Unknown Author is licensed under [CC BY-NC](#)

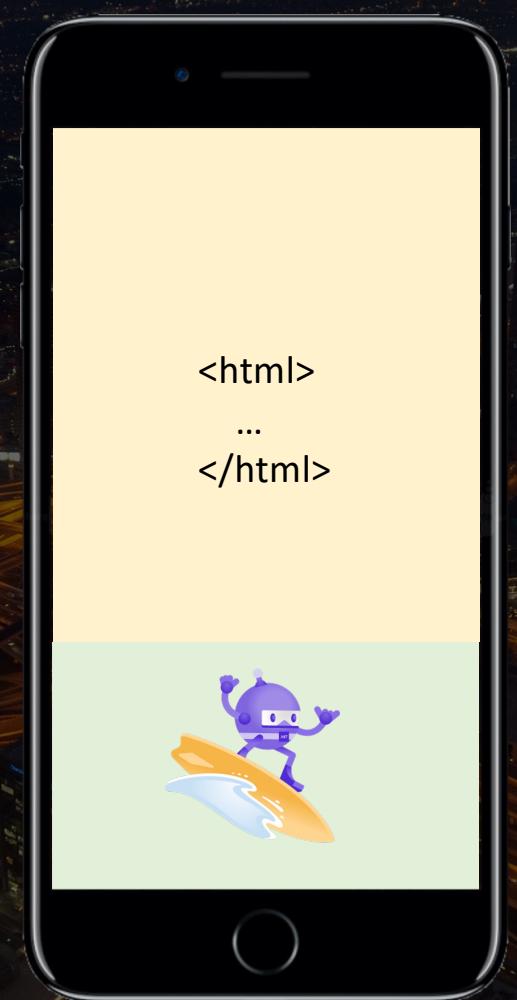
@mallibone





@mallibone

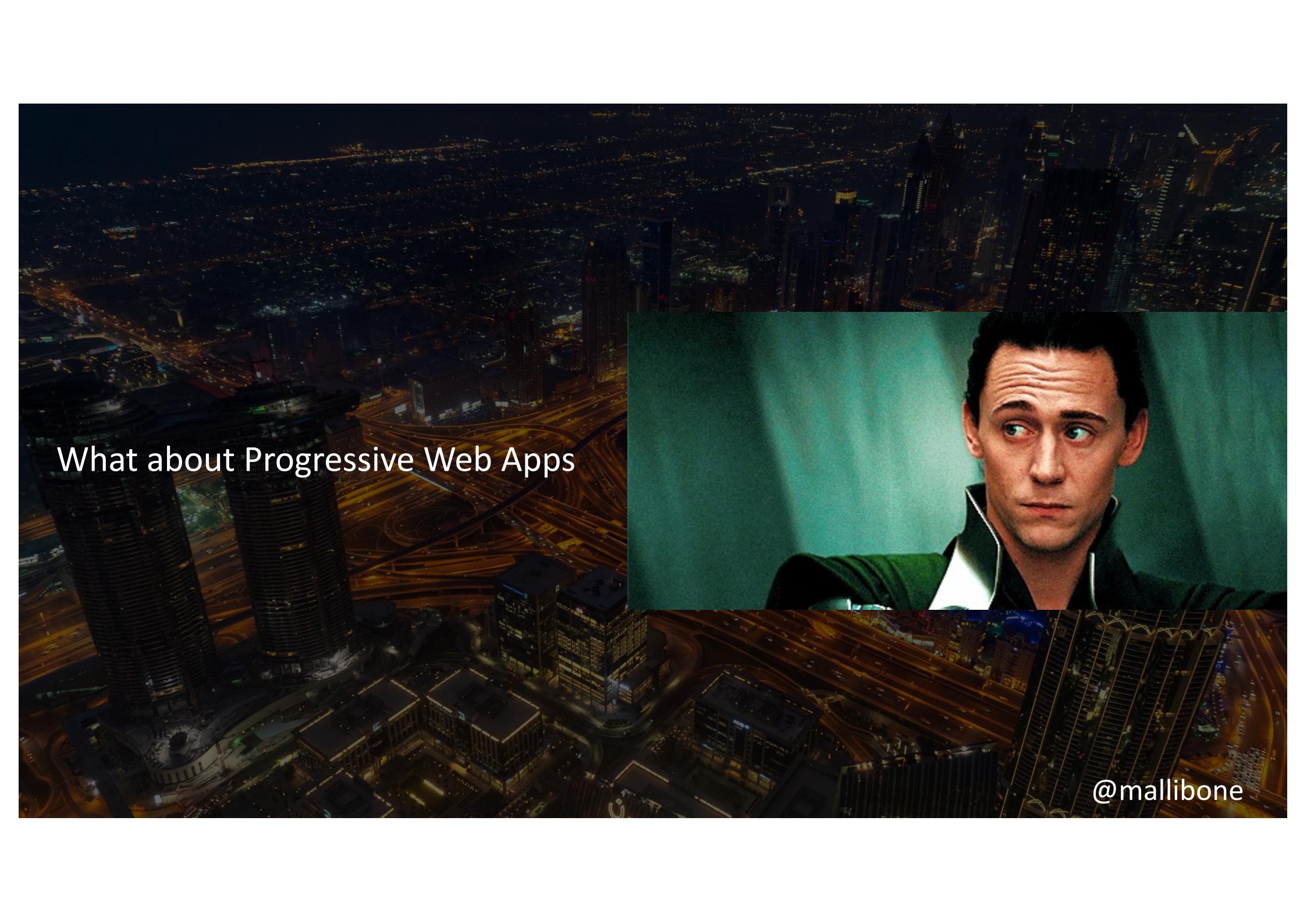




```
<html>
...
</html>
```



@mallibone



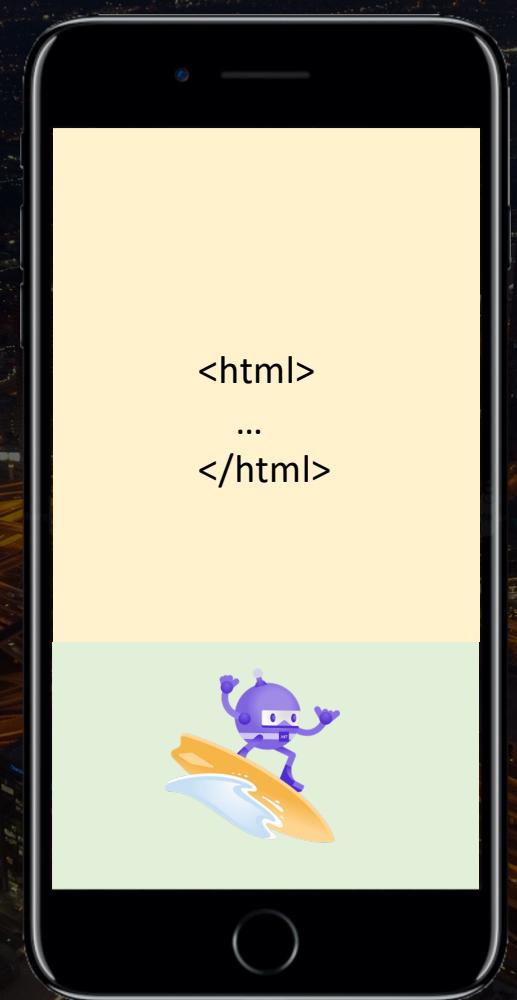
What about Progressive Web Apps



@mallibone







```
<html>
...
</html>
```



@mallibone

**YEAH IF YOU COULD JUST
SIMPLY TURN IT INTO A MOBILE APP**

THAT WOULD BE GREAT

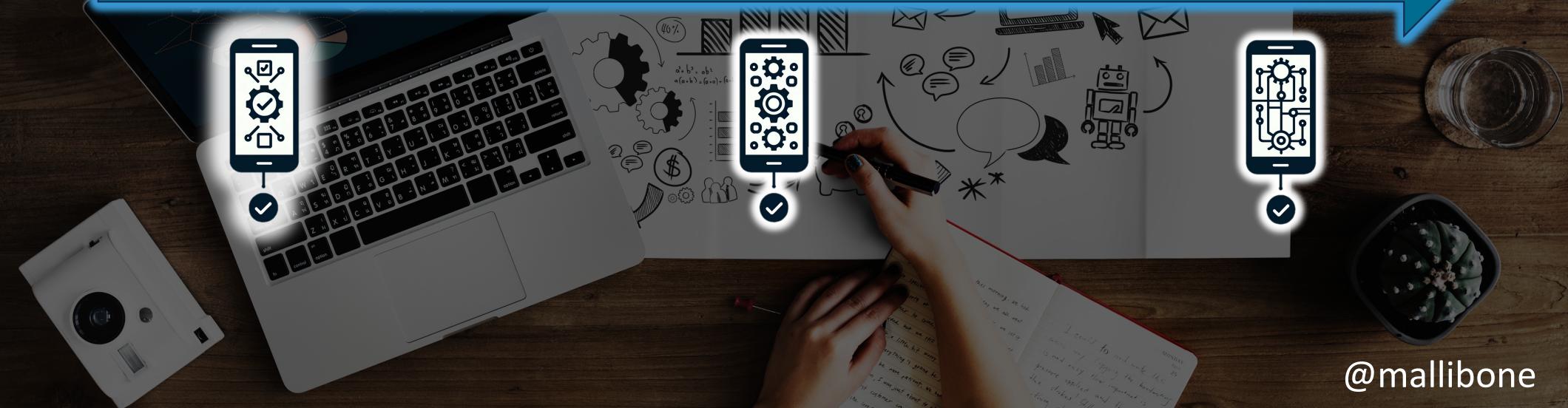
imgflip.com

@mallibone

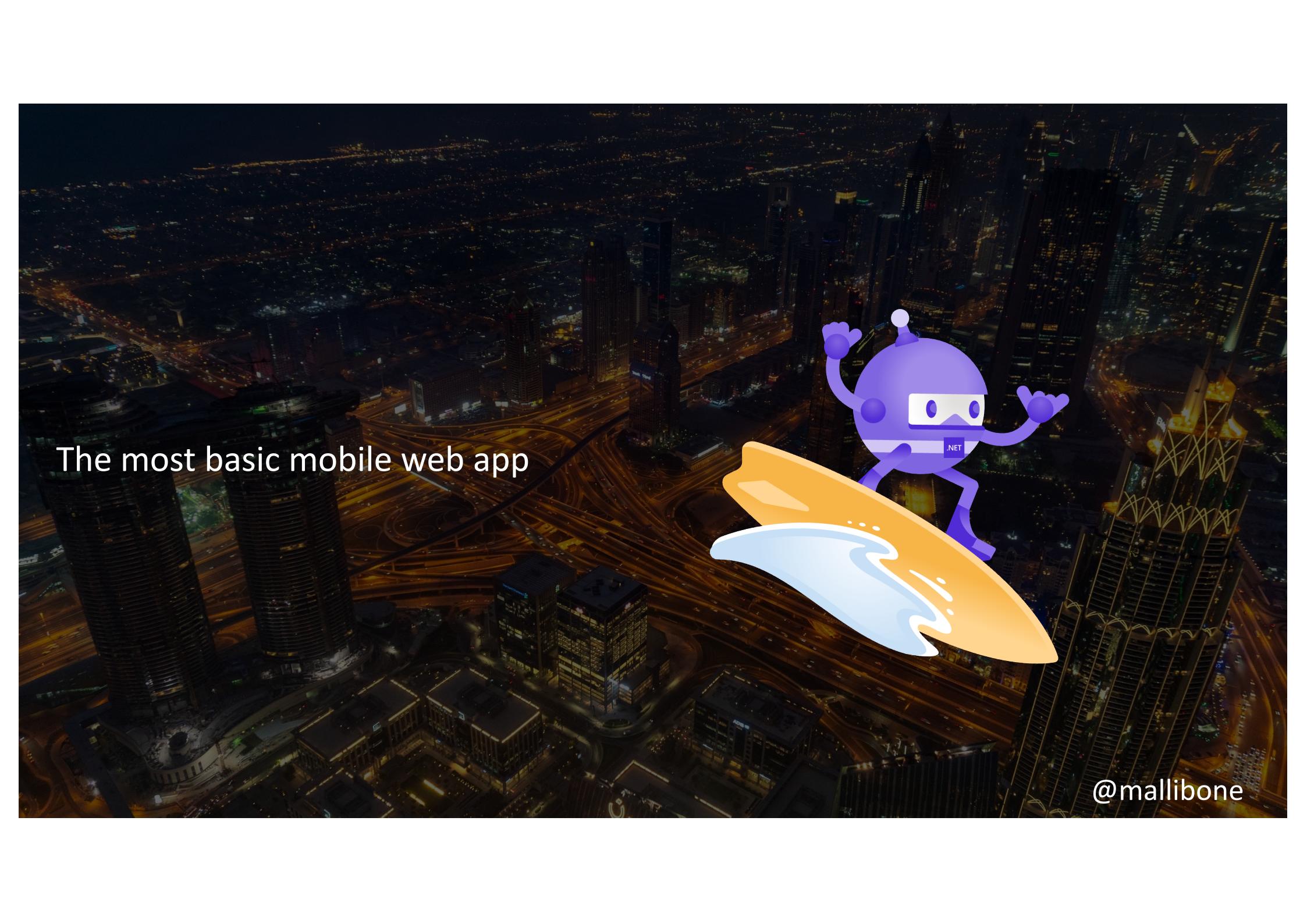
Hybrid continuum



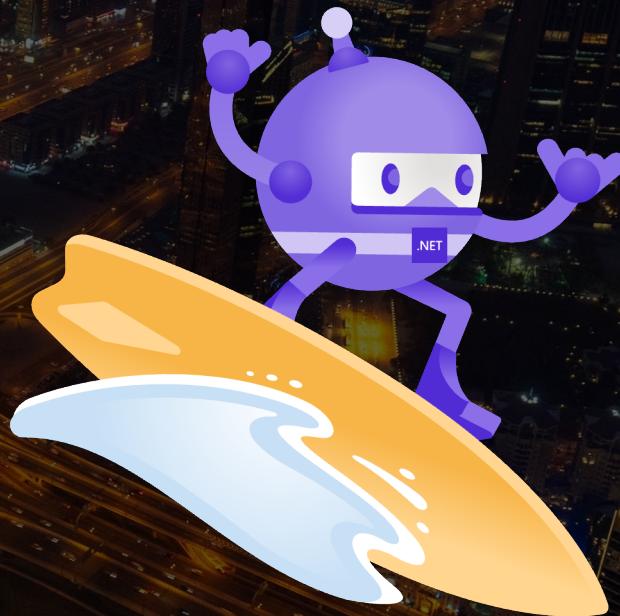
Hybrid continuum



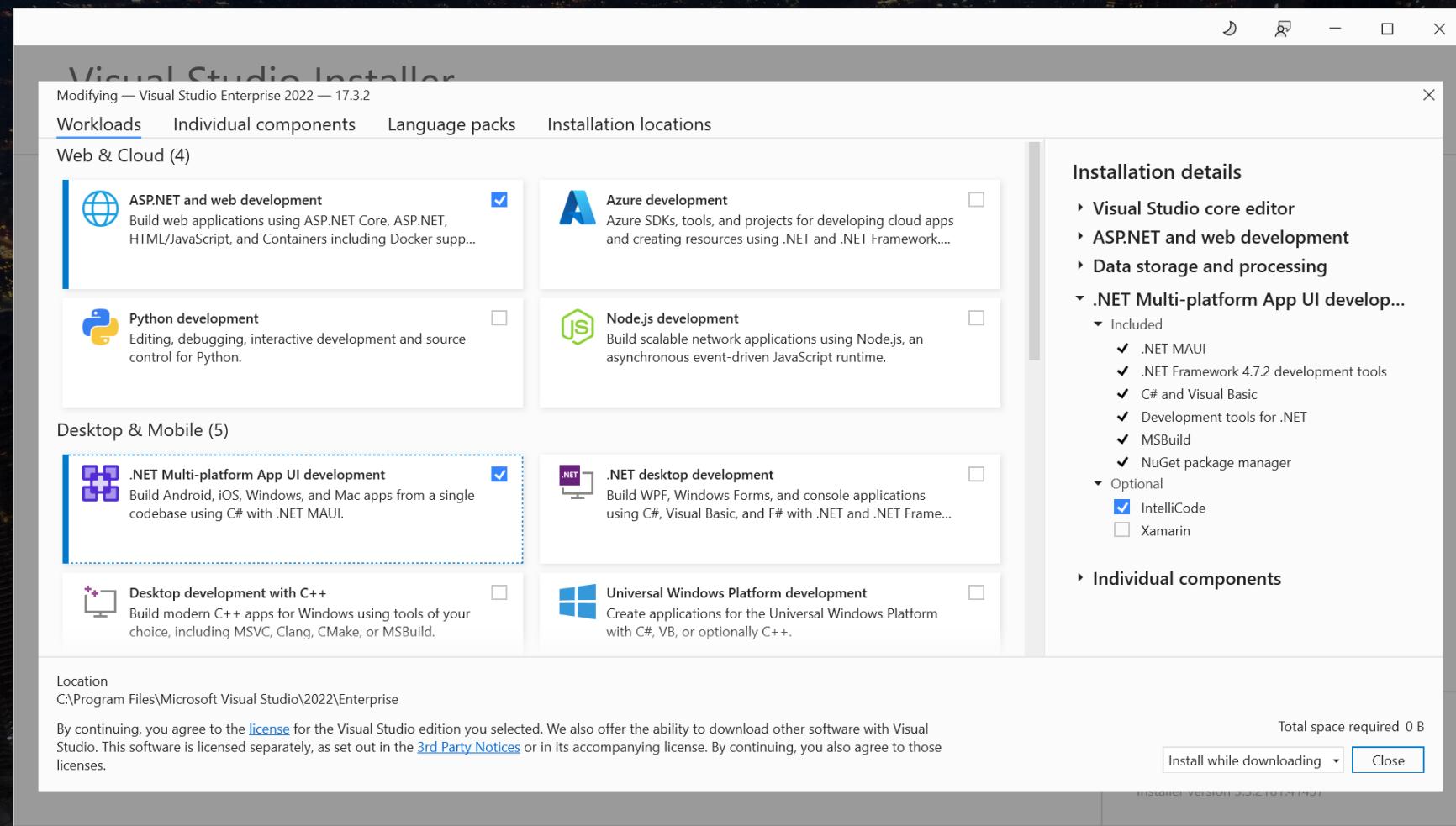
@mallibone



The most basic mobile web app



@mallibone



@mallibone

Visual Studio Installer

Modifying — Visual Studio Enterprise 2022 — 17.3.2

Workloads Individual components Language packs Installation locations

Web & Cloud (4)



ASP.NET and web development

Build web applications using ASP.NET Core, ASP.NET, HTML/JavaScript, and Containers including Docker supp...



Azure development

Azure SDKs, tools, and projects for developing cloud apps and creating resources using .NET and .NET Framework....



Python development

Editing, debugging, interactive development and source control for Python.



Node.js development

Build scalable network applications using Node.js, an asynchronous event-driven JavaScript runtime.



Desktop & Mobile (5)



.NET Multi-platform App UI development

Build Android, iOS, Windows, and Mac apps from a single codebase using C# with .NET MAUI.



.NET desktop development

Build WPF, Windows Forms, and console applications using C#, Visual Basic, and F# with .NET and .NET Frame...



Desktop development with C++

Build modern C++ apps for Windows using tools of your choice, including MSVC, Clang, CMake, or MSBuild.



Universal Windows Platform development

Create applications for the Universal Windows Platform with C#, VB, or optionally C++.



Location

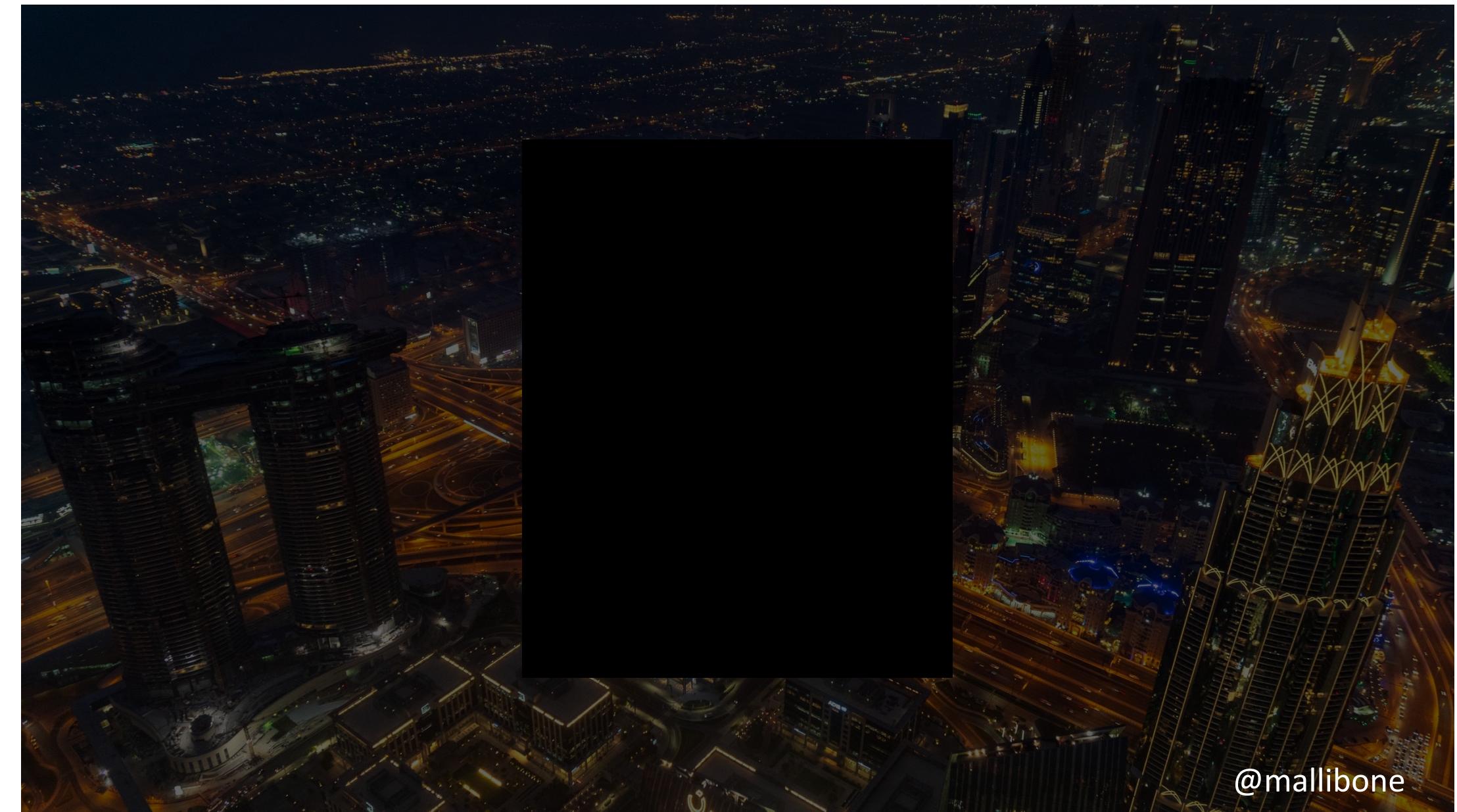
C:\Program Files\Microsoft Visual Studio\2022\Enterprise

By continuing, you agree to the [license](#) for the Visual Studio edition you selected. We also offer the ability to download other software with Visual Studio. This software is licensed separately, as set out in the [3rd Party Notices](#) or in its accompanying license. By continuing, you also agree to those licenses.

Total space require

Install while downloading Click to expand

Installer version 3.5.2 (01.41437)



@mallibone

Create a new project

Recent project templates

.NET MAUI App

C#

All languages

All platforms

All project types

.NET MAUI App

A project for creating a .NET MAUI application for iOS, Android, Mac Catalyst, WinUI and Tizen

C# Android iOS Mac Catalyst macOS MAUI Tizen
Windows

.NET MAUI Blazor App

A project for creating a .NET MAUI application for iOS, Android, Mac Catalyst, WinUI, and Tizen using Blazor

C# Android Blazor iOS Mac Catalyst macOS MAUI
Tizen Windows

.NET MAUI Class Library

A project for creating a .NET MAUI class library

C# Android iOS Mac Catalyst macOS MAUI Tizen
Windows

Console App (.NET Framework)

A project for creating a command-line application

C# Windows Console

ASP.NET Web Application (.NET Framework)



Clear all

New

New

Next

@mallibone

File Edit View Git Project Build Debug XML Test Analyze Tools Extensions Window Help Search **MauiApp1**

Debug Any CPU Android Emulator Live Share

Toolbox SQL Server Object Explorer

MauiApp1.csproj

```
<Project Sdk="Microsoft.NET.Sdk">
  <PropertyGroup>
    <TargetFrameworks>net6.0-android;net6.0-ios;net6.0-maccatalyst</TargetFrameworks>
    <TargetFrameworks Condition="\"$([MSBuild]::IsOSPlatform('windows'))\">$([TargetFrameworks]);net6.0-windows10.0.19041.0</TargetFrameworks>
    <!-- Uncomment to also build the tizen app. You will need to install tizen by following this: https://github.com/Samsung/Tizen.NET -->
    <!-- <TargetFrameworks>$([TargetFrameworks]);net6.0-tizen</TargetFrameworks> -->
    <OutputType>Exe</OutputType>
    <RootNamespace>MauiApp1</RootNamespace>
    <UseMaui>true</UseMaui>
    <SingleProject>true</SingleProject>
    <ImplicitUsings>enable</ImplicitUsings>

    <!-- Display name -->
    <ApplicationTitle>MauiApp1</ApplicationTitle>

    <!-- App Identifier -->
    <ApplicationId>com.companyname.mauiapp1</ApplicationId>
    <ApplicationIdGuid>2128AE37-4BEC-46F8-B35E-B17F617BB30D</ApplicationIdGuid>

    <!-- Versions -->
    <ApplicationDisplayVersion>1.0</ApplicationDisplayVersion>
    <ApplicationVersion>1</ApplicationVersion>

    <SupportedOSPlatformVersion Condition="\"$([MSBuild]::GetTargetPlatformIdentifier('$(TargetFramework)')\" == 'ios')\">14.2</SupportedOSPlatformVersion>
    <SupportedOSPlatformVersion Condition="\"$([MSBuild]::GetTargetPlatformIdentifier('$(TargetFramework)')\" == 'maccatalyst')\">14.0</SupportedOSPlatformVersion>
    <SupportedOSPlatformVersion Condition="\"$([MSBuild]::GetTargetPlatformIdentifier('$(TargetFramework)')\" == 'android')\">21.0</SupportedOSPlatformVersion>
    <SupportedOSPlatformVersion Condition="\"$([MSBuild]::GetTargetPlatformIdentifier('$(TargetFramework)')\" == 'windows')\">10.0.17763.0</SupportedOSPlatformVersion>
    <TargetPlatformMinVersion Condition="\"$([MSBuild]::GetTargetPlatformIdentifier('$(TargetFramework)')\" == 'windows')\">10.0.17763.0</TargetPlatformMinVersion>
    <SupportedOSPlatformVersion Condition="\"$([MSBuild]::GetTargetPlatformIdentifier('$(TargetFramework)')\" == 'tizen')\">6.5</SupportedOSPlatformVersion>
  </PropertyGroup>

  <ItemGroup>
    <!-- App Icon -->
    <MauiIcon Include="Resources\Icon\appicon.svg" ForegroundFile="Resources\Icon\appiconfg.svg" Color="#512BD4" />

    <!-- Splash Screen -->
    <MauiSplashScreen Include="Resources\Splash\splash.svg" Color="#512BD4" BaseSize="128,128" />

    <!-- Images -->
    <MauiImage Include="Resources\Images\*" />
    <MauiImage Update="Resources\Images\dotnet_bot.svg" BaseSize="168,208" />

    <!-- Custom Fonts -->
    <MauiFont Include="Resources\Fonts\*" />

    <!-- Raw Assets (also remove the "Resources\Raw" prefix) -->
    <MauiAsset Include="Resources\Raw\**" LogicalName="%(RecursiveDir)%(Filename)%(Extension)" />
  </ItemGroup>

```

100 % No issues found Ln: 1 Ch: 1 TABS CRLF

Error List Output

Solution Explorer

Search Solution Explorer (Ctrl+U)

Solution 'MauiApp1' (1 of 1 project)

MauiApp1

- Dependencies
- Properties
- Platforms
 - Android
 - iOS
 - MacCatalyst
 - Tizen
 - Windows
- Resources
- App.xaml
- AppShell.xaml
- MainPage.xaml
- MauiProgram.cs

Solution Explorer Git Changes

Properties

XML Document

Misc

Encoding Unicode (UTF-8)

Output

Schemas "C:\Program Files\Microsoft Visual Studio\2022\Community\SDK\NuGetFallbackFolder\Microsoft.Maui\6.0.0\ref\net6.0\Microsoft.Maui.dll"

Stylesheet

Encoding

Character encoding of the document.

Add to Source Control Select Repository



Demo Retro

- Sometimes all you need is a WebView
- And some custom injected JS (and CSS)
- WebView != Browser

@mallibone



WEBVIEW

MANAGEMENT

REALITY

MauiApp1

Live Share

Solution Explorer

Search Solution Explorer (Ctrl+U)

Solution 'MauiApp1' (1 of 1 project)

MauiApp1

- Dependencies
- Properties
- Platforms
 - Android
 - iOS
 - MacCatalyst
 - Tizen
 - Windows
- Resources
- App.xaml
- AppShell.xaml
- MainPage.xaml
- C# MauiProgram.cs

Solution Explorer | Git Changes

Properties

XML Document

Misc

Encoding Unicode (UTF-8)

Output

Schemas "C:\Program Files\Microsoft Visual Studio\2022\Community\SDK\NuGetFallbackFolder\Microsoft.Maui\6.0.0\ref\net6.0\Microsoft.Maui.dll"

Stylesheet

Encoding

Character encoding of the document.

File Edit View Git Project Build Debug XML Test Analyze Tools Extensions Window Help Search

MauiApp1.csproj

```
<Project Sdk="Microsoft.NET.Sdk">
  <PropertyGroup>
    <TargetFrameworks>net6.0-android;net6.0-ios;net6.0-maccatalyst</TargetFrameworks>
    <TargetFrameworks Condition="$(MSBuild)::IsOSPlatform('windows'))">$(<TargetFrameworks>);net6.0-windows10.0.19041.0</TargetFrameworks>
    <!-- Uncomment to also build the tizen app. You will need to install tizen by following this: https://github.com/Samsung/Tizen.NET -->
    <!-- <TargetFrameworks>$(<TargetFrameworks>);net6.0-tizen</TargetFrameworks> -->
    <OutputType>Exe</OutputType>
    <RootNamespace>MauiApp1</RootNamespace>
    <UseMaui>true</UseMaui>
    <SingleProject>true</SingleProject>
    <ImplicitUsings>enable</ImplicitUsings>

    <!-- Display name -->
    <ApplicationTitle>MauiApp1</ApplicationTitle>

    <!-- App Identifier -->
    <ApplicationId>com.companyname.mauiapp1</ApplicationId>
    <ApplicationIdGuid>2128AE37-4BEC-46F8-B35E-B17F617BB30D</ApplicationIdGuid>

    <!-- Versions -->
    <ApplicationDisplayVersion>1.0</ApplicationDisplayVersion>
    <ApplicationVersion>1</ApplicationVersion>

    <SupportedOSPlatformVersion Condition="$(MSBuild)::GetTargetPlatformIdentifier('$(TargetFramework)') == 'ios'">14.2</SupportedOSPlatformVersion>
    <SupportedOSPlatformVersion Condition="$(MSBuild)::GetTargetPlatformIdentifier('$(TargetFramework)') == 'maccatalyst'">14.0</SupportedOSPlatformVersion>
    <SupportedOSPlatformVersion Condition="$(MSBuild)::GetTargetPlatformIdentifier('$(TargetFramework)') == 'android'">21.0</SupportedOSPlatformVersion>
    <SupportedOSPlatformVersion Condition="$(MSBuild)::GetTargetPlatformIdentifier('$(TargetFramework)') == 'windows'">10.0.17763.0</SupportedOSPlatformVersion>
    <TargetPlatformMinVersion Condition="$(MSBuild)::GetTargetPlatformIdentifier('$(TargetFramework)') == 'windows'">10.0.17763.0</TargetPlatformMinVersion>
    <SupportedOSPlatformVersion Condition="$(MSBuild)::GetTargetPlatformIdentifier('$(TargetFramework)') == 'tizen'">6.5</SupportedOSPlatformVersion>
  </PropertyGroup>

  <ItemGroup>
    <!-- App Icon -->
    <MauiIcon Include="Resources\Icon\appicon.svg" ForegroundFile="Resources\Icon\appiconfg.svg" Color="#512BD4" />

    <!-- Splash Screen -->
    <MauiSplashScreen Include="Resources\Splash\splash.svg" Color="#512BD4" BaseSize="128,128" />

    <!-- Images -->
    <MauiImage Include="Resources\Images\*" />
    <MauiImage Update="Resources\Images\dotnet_bot.svg" BaseSize="168,208" />

    <!-- Custom Fonts -->
    <MauiFont Include="Resources\Fonts\*" />

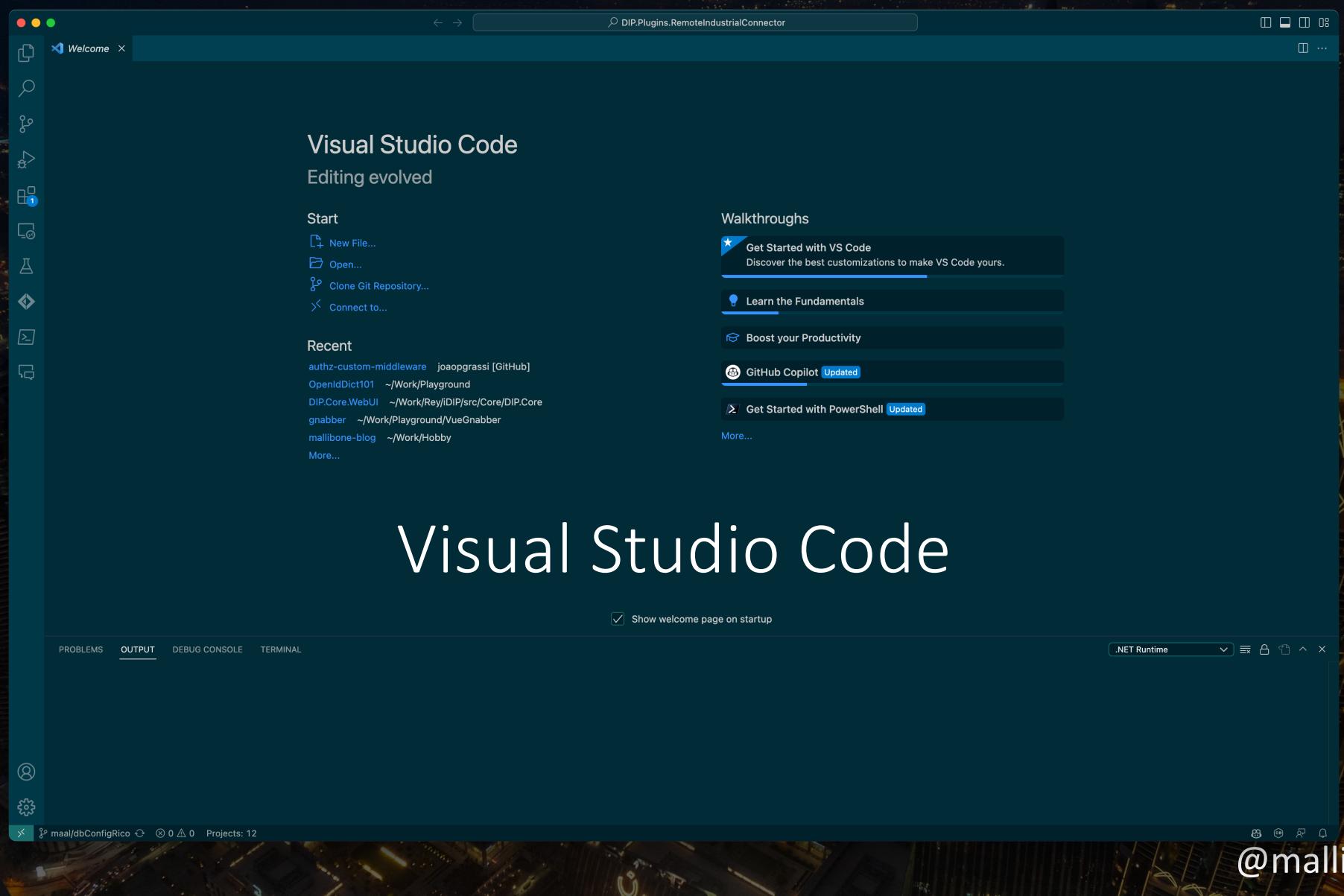
    <!-- Raw Assets (also remove the "Resources\Raw" prefix) -->
    <MauiAsset Include="Resources\Raw\**" LogicalName="%(RecursiveDir)%(Filename)%(Extension)" />
  </ItemGroup>

```

100 % No issues found Ln: 1 Ch: 1 TABS CRLF

Error List Output

Ready Add to Source Control Select Repository



@mallibone

The image shows a screenshot of a Windows desktop environment with a night cityscape background. In the foreground, there are two floating windows from the Microsoft Edge Dev browser, both displaying the same C# code for a 'Podcast.Web' project. The code is located in the 'Player' folder and contains methods for handling volume and muted status changes. A large, semi-transparent circular watermark with the letters 'RD' is overlaid on the bottom left of the screen.

```
private async Task ChangeVolume(ChangeEventArgs e)
{
    if (e.Value != null && int.TryParse(e.Value.ToString(), out var volume))
    {
        Volume = volume;
        await OnVolumeChange.InvokeAsync(volume);
    }
}

private void OnInputCurrentTime(ChangeEventArgs e)
{
    if (e.Value != null && int.TryParse(e.Value.ToString(), out var time))
    {
        seekingTime = time;
    }
}

private async Task OnChangeCurrentTime(ChangeEventArgs e)
{
    if (e.Value != null && int.TryParse(e.Value.ToString(), out var time))
    {
        await SetCurrentTime(time);
        seekingTime = null;
    }
}
```

@mallibone

VS for Mac

The screenshot displays two instances of the Visual Studio for Mac IDE running on a Mac OS X desktop. The desktop background features a night-time cityscape with illuminated skyscrapers.

Left Instance (Podcast.Web):

- Solution Explorer: Shows the project structure for Podcast.Web (main), including subfolders like Podcast.Client, Podcast.Components, Events, Extensions, ListenTogether, Player, Shared, wwwroot, and Pages.
- Code Editor: The FloatingPlayer.razor file is open, showing C# code for a floating player component. The code includes methods like ChangeVolume and OnInputCurrentTime.

Right Instance (Podcast.Server):

- Solution Explorer: Shows the project structure for Podcast.Server (main), including subfolders like Podcast.Client, Podcast.Components, Events, Extensions, ListenTogether, Player, Shared, wwwroot, and Pages.
- Code Editor: The FloatingPlayer.razor file is open, showing C# code for a floating player component. The code includes methods like ChangeVolume and OnInputCurrentTime.

@mallibone



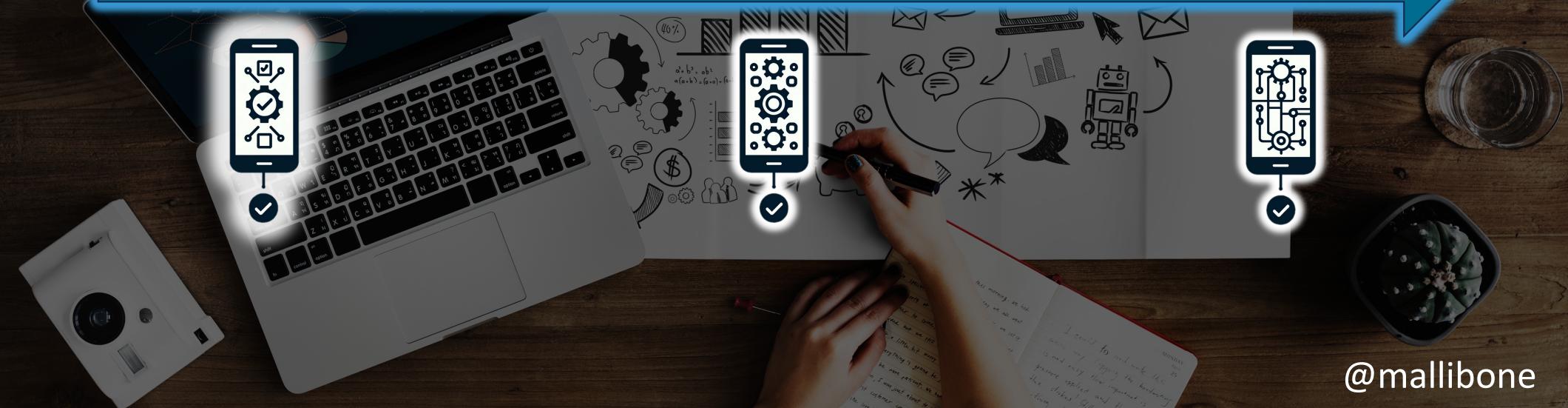
imgflip.com

@mallibone

Hybrid continuum



Hybrid continuum

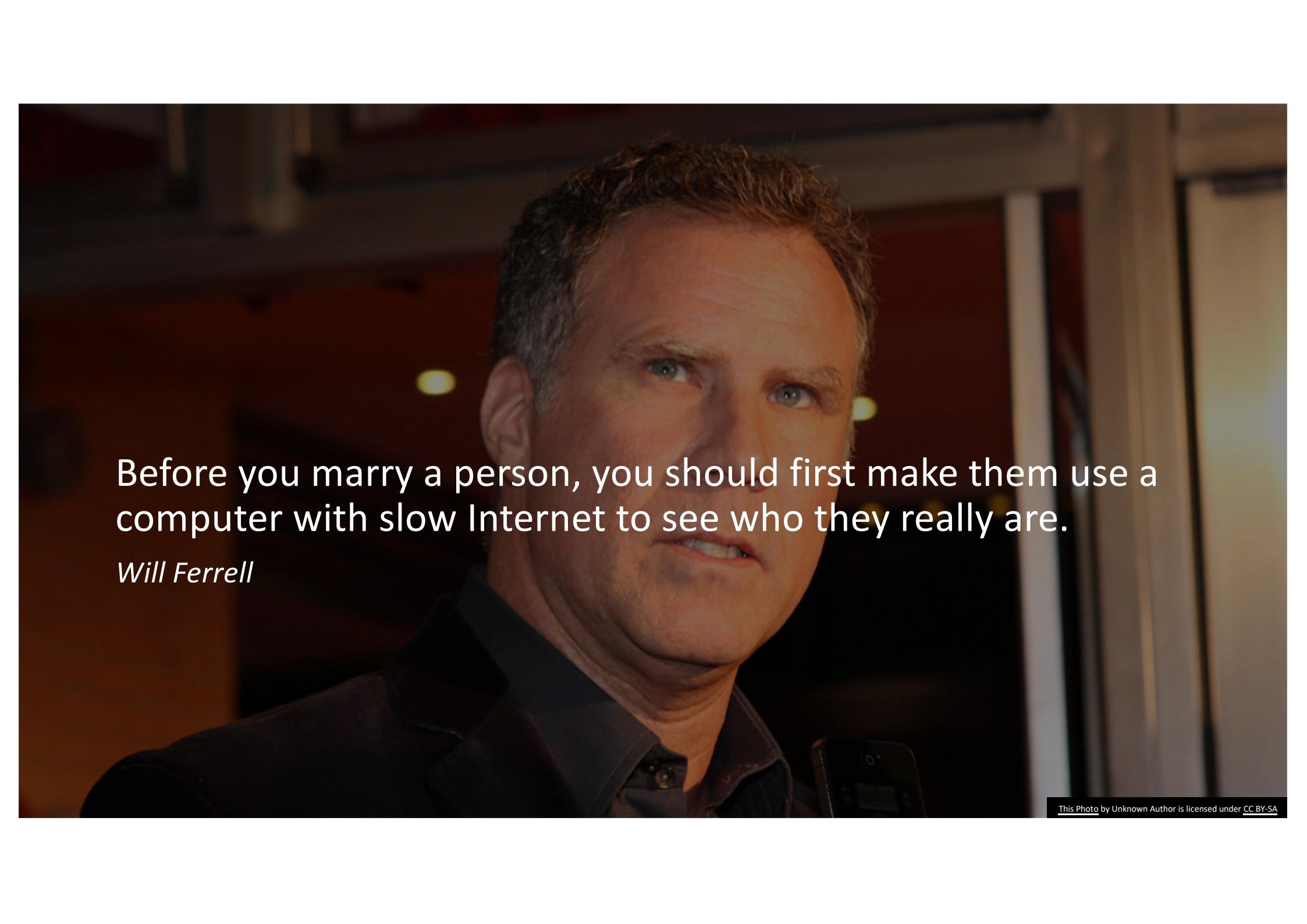


@mallibone



Challenge





Before you marry a person, you should first make them use a computer with slow Internet to see who they really are.

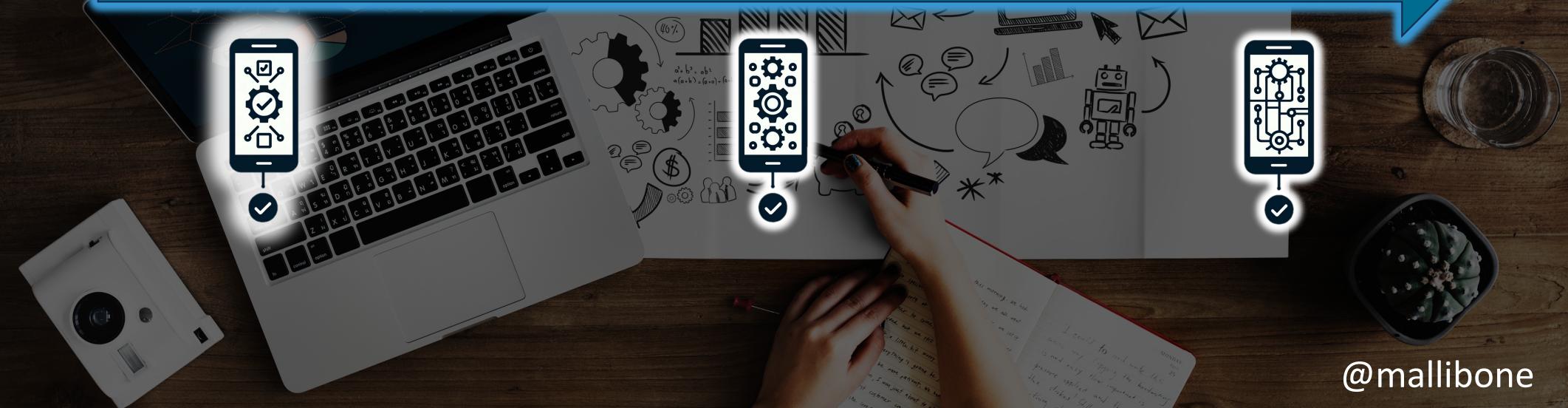
Will Ferrell

OFFLINE

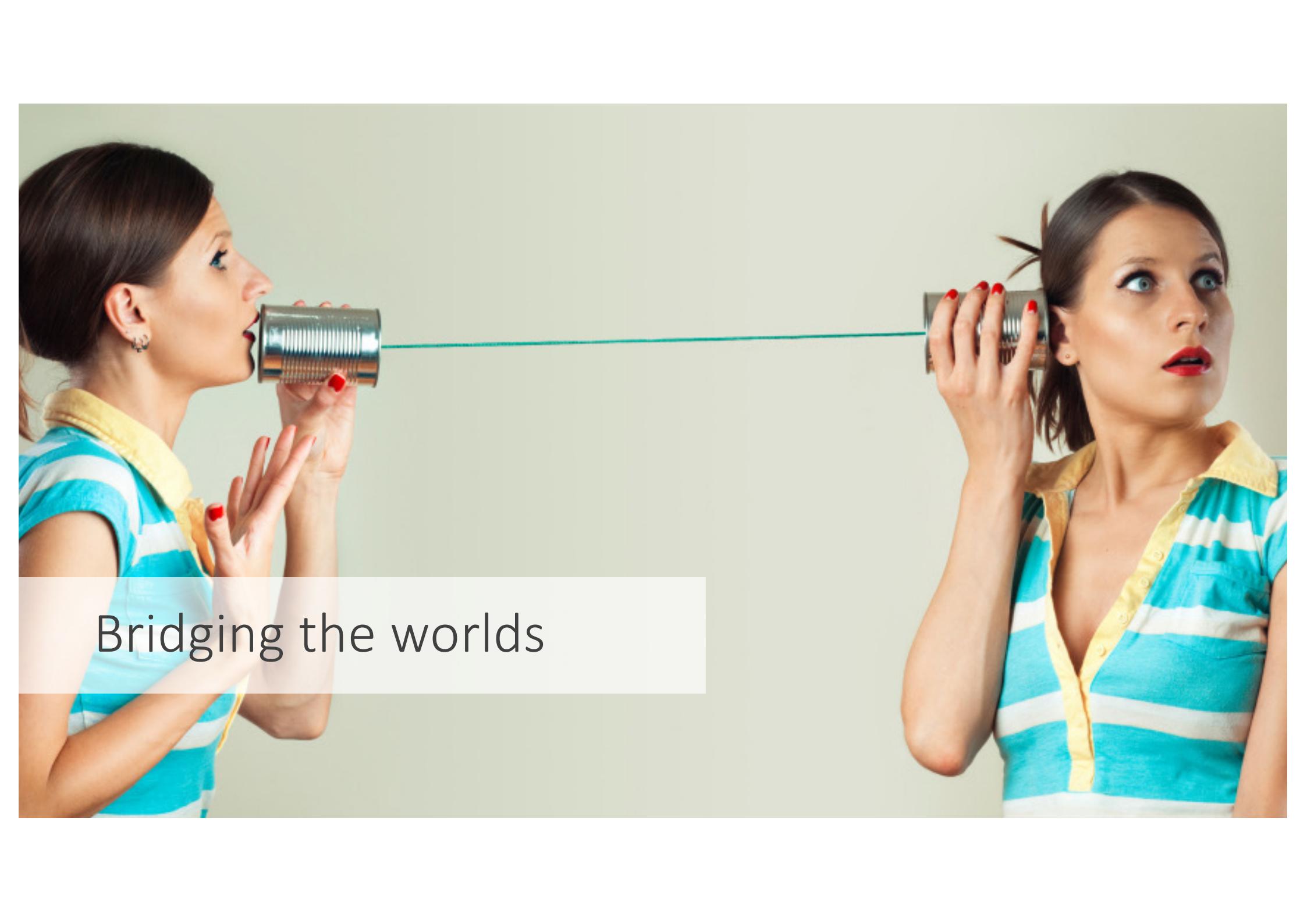
Hybrid continuum



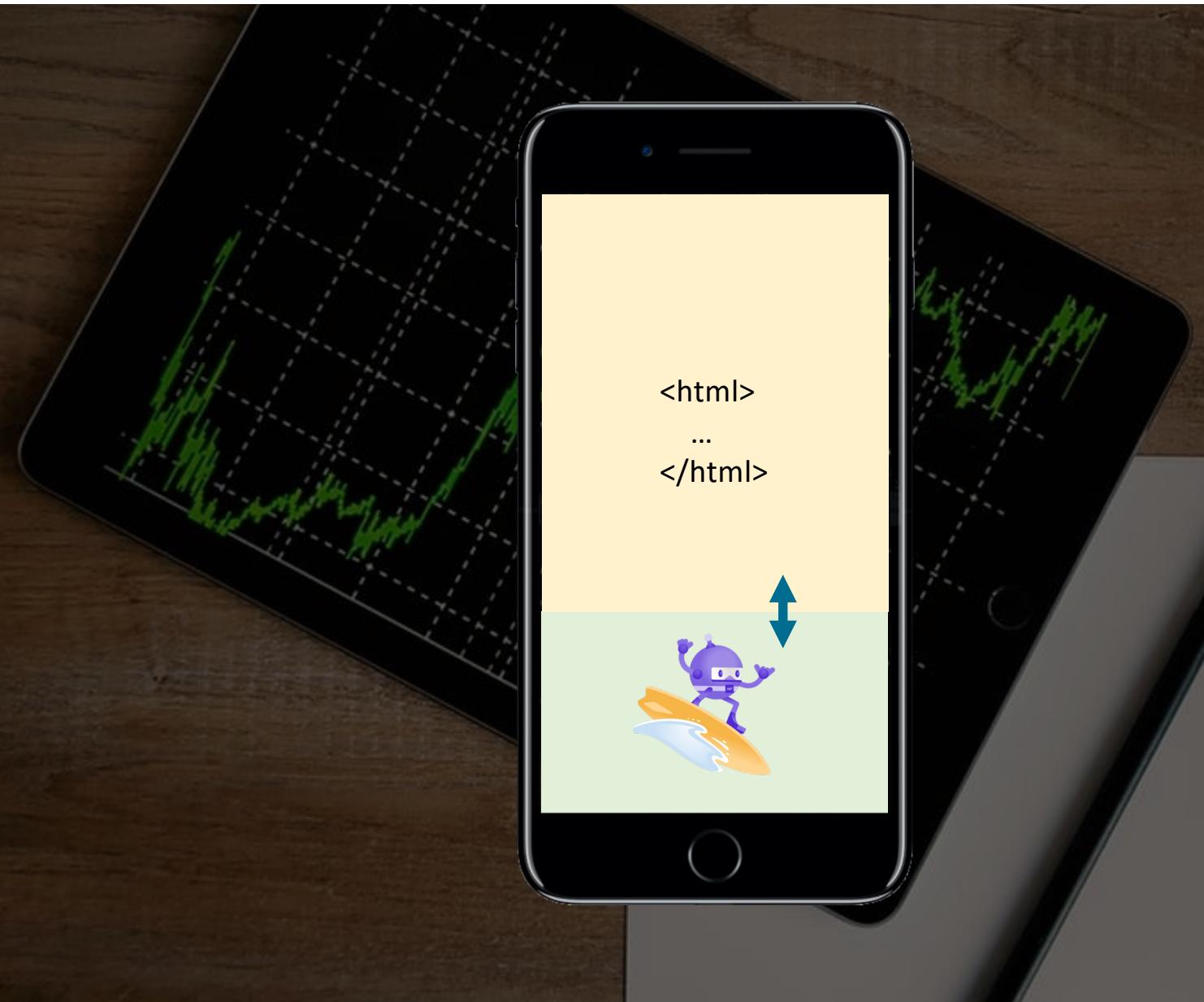
Hybrid continuum



@mallibone

A photograph of a woman in a blue and white striped shirt using a tin can telephone to communicate with another woman across a room. A thin green line connects the two cans. The woman on the left is speaking into her can, while the woman on the right holds her can to her ear, looking towards the source of the sound.

Bridging the worlds



@mallibone

File Edit View Git Project Build Debug XML Test Analyze Tools Extensions Window Help Search **MauiApp1**

Live Share

MauiApp1.csproj

```
<Project Sdk="Microsoft.NET.Sdk">
  <PropertyGroup>
    <TargetFrameworks>net6.0-android;net6.0-ios;net6.0-maccatalyst</TargetFrameworks>
    <TargetFrameworks Condition="\"$([MSBuild]::IsOSPlatform('windows'))\">$([TargetFrameworks]);net6.0-windows10.0.19041.0</TargetFrameworks>
    <!-- Uncomment to also build the tizen app. You will need to install tizen by following this: https://github.com/Samsung/Tizen.NET -->
    <!-- <TargetFrameworks>$([TargetFrameworks]);net6.0-tizen</TargetFrameworks> -->
    <OutputType>Exe</OutputType>
    <RootNamespace>MauiApp1</RootNamespace>
    <UseMaui>true</UseMaui>
    <SingleProject>true</SingleProject>
    <ImplicitUsings>enable</ImplicitUsings>

    <!-- Display name -->
    <ApplicationTitle>MauiApp1</ApplicationTitle>

    <!-- App Identifier -->
    <ApplicationId>com.companyname.mauiapp1</ApplicationId>
    <ApplicationIdGuid>2128AE37-4BEC-46F8-B35E-B17F617BB30D</ApplicationIdGuid>

    <!-- Versions -->
    <ApplicationDisplayVersion>1.0</ApplicationDisplayVersion>
    <ApplicationVersion>1</ApplicationVersion>

    <SupportedOSPlatformVersion Condition="\"$([MSBuild]::GetTargetPlatformIdentifier('$(TargetFramework)')\" == 'ios')\">14.2</SupportedOSPlatformVersion>
    <SupportedOSPlatformVersion Condition="\"$([MSBuild]::GetTargetPlatformIdentifier('$(TargetFramework)')\" == 'maccatalyst')\">14.0</SupportedOSPlatformVersion>
    <SupportedOSPlatformVersion Condition="\"$([MSBuild]::GetTargetPlatformIdentifier('$(TargetFramework)')\" == 'android')\">21.0</SupportedOSPlatformVersion>
    <SupportedOSPlatformVersion Condition="\"$([MSBuild]::GetTargetPlatformIdentifier('$(TargetFramework)')\" == 'windows')\">10.0.17763.0</SupportedOSPlatformVersion>
    <TargetPlatformMinVersion Condition="\"$([MSBuild]::GetTargetPlatformIdentifier('$(TargetFramework)')\" == 'windows')\">10.0.17763.0</TargetPlatformMinVersion>
    <SupportedOSPlatformVersion Condition="\"$([MSBuild]::GetTargetPlatformIdentifier('$(TargetFramework)')\" == 'tizen')\">6.5</SupportedOSPlatformVersion>
  </PropertyGroup>

  <ItemGroup>
    <!-- App Icon -->
    <MauiIcon Include="Resources\Icon\appicon.svg" ForegroundFile="Resources\Icon\appiconfg.svg" Color="#512BD4" />

    <!-- Splash Screen -->
    <MauiSplashScreen Include="Resources\Splash\splash.svg" Color="#512BD4" BaseSize="128,128" />

    <!-- Images -->
    <MauiImage Include="Resources\Images\*" />
    <MauiImage Update="Resources\Images\dotnet_bot.svg" BaseSize="168,208" />

    <!-- Custom Fonts -->
    <MauiFont Include="Resources\Fonts\*" />

    <!-- Raw Assets (also remove the "Resources\Raw" prefix) -->
    <MauiAsset Include="Resources\Raw\**" LogicalName="%(RecursiveDir)%(Filename)%(Extension)" />
  </ItemGroup>

```

100 % No issues found Ln: 1 Ch: 1 TABS CRLF

Error List Output

Solution Explorer

Solution 'MauiApp1' (1 of 1 project)

- MauiApp1
 - Dependencies
 - Properties
 - Platforms
 - Android
 - iOS
 - MacCatalyst
 - Tizen
 - Windows
 - Resources
 - App.xaml
 - AppShell.xaml
 - MainPage.xaml
 - MauiProgram.cs

Properties

XML Document

Misc

Encoding Unicode (UTF-8)

Output

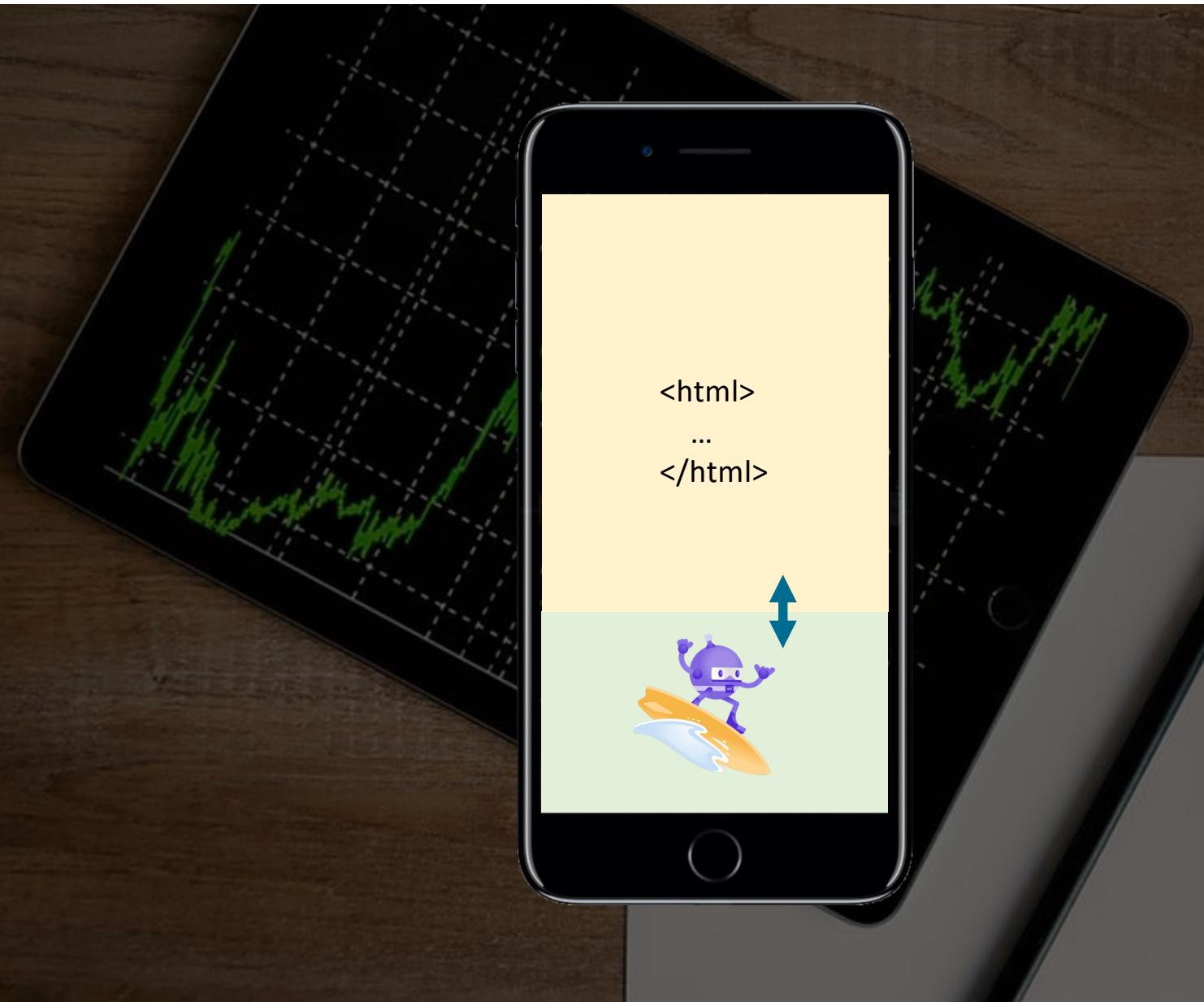
Schemas "C:\Program Files\Microsoft Visual Studio\2022\Community\SDK\NuGetFallbackFolder\Microsoft.Maui\6.0.0\ref\net6.0\Microsoft.Maui.dll"

Stylesheet

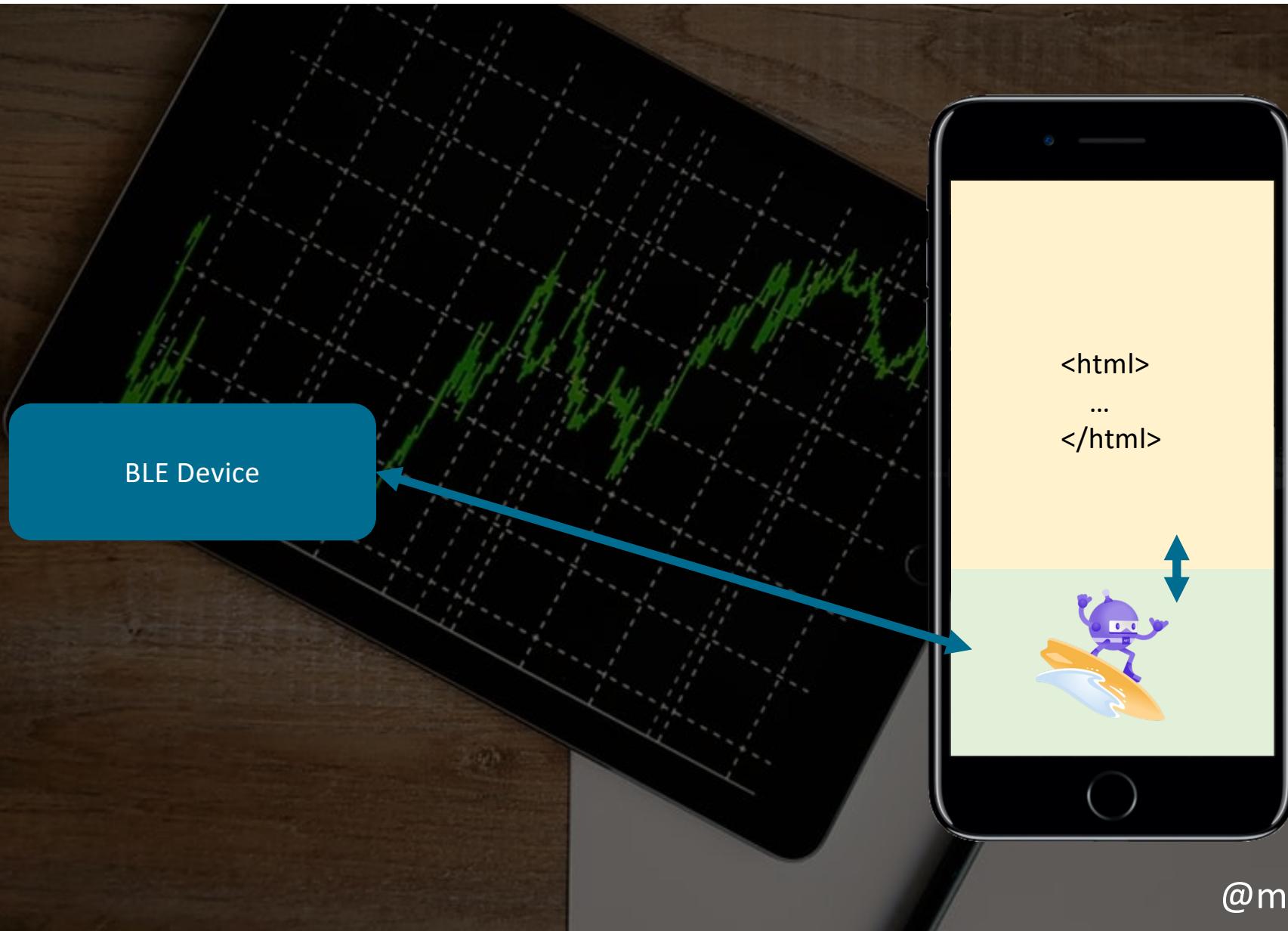
Encoding

Character encoding of the document.

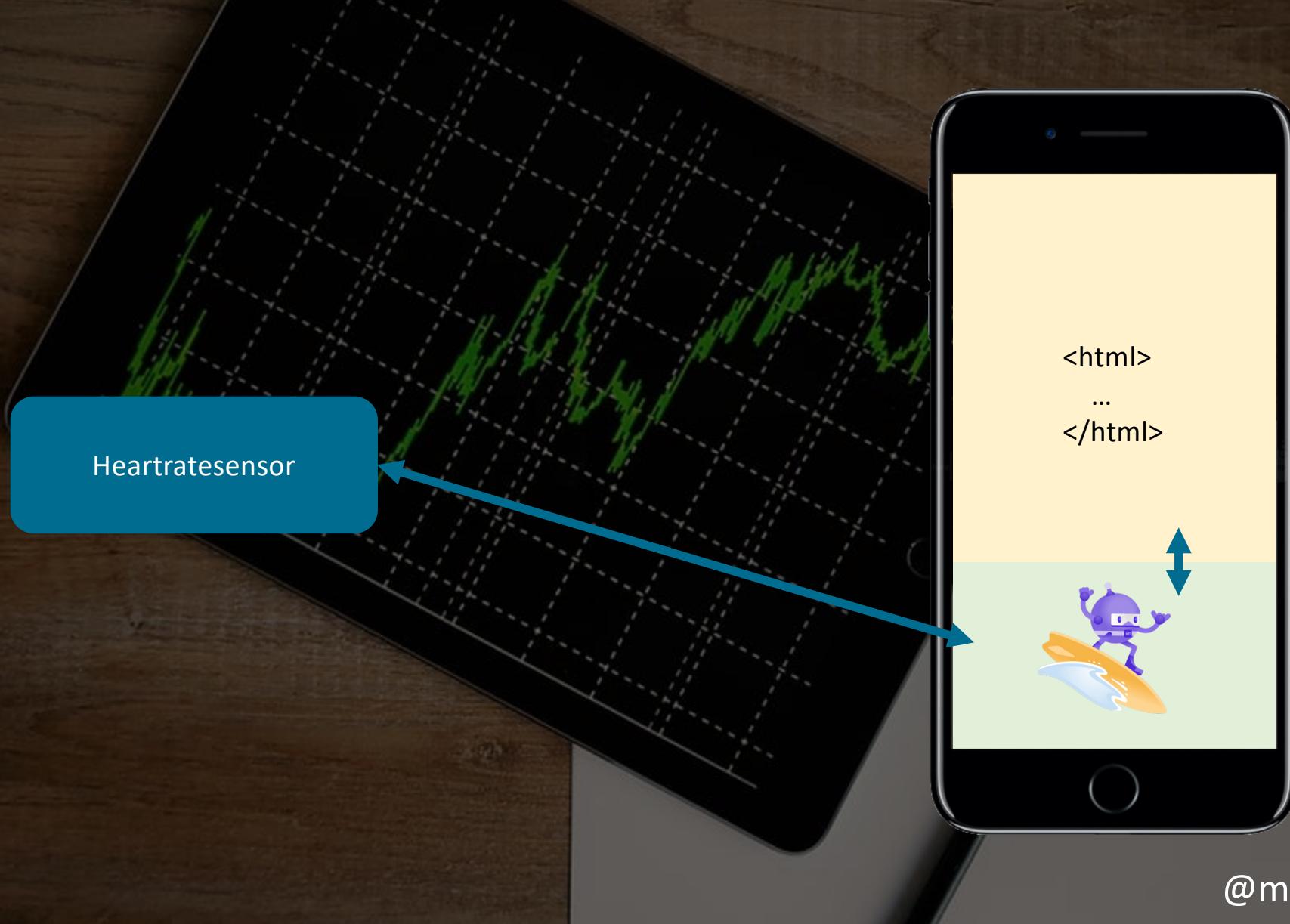
Add to Source Control Select Repository



@mallibone



@mallibone



Heartratesensor

<html>
...
</html>



@mallibone

File Edit View Git Project Build Debug XML Test Analyze Tools Extensions Window Help Search **MauiApp1**

Live Share

MauiApp1.csproj

```
<Project Sdk="Microsoft.NET.Sdk">
  <PropertyGroup>
    <TargetFrameworks>net6.0-android;net6.0-ios;net6.0-maccatalyst</TargetFrameworks>
    <TargetFrameworks Condition="\"$([MSBuild]::IsOSPlatform('windows'))\">$([TargetFrameworks]);net6.0-windows10.0.19041.0</TargetFrameworks>
    <!-- Uncomment to also build the tizen app. You will need to install tizen by following this: https://github.com/Samsung/Tizen.NET -->
    <!-- <TargetFrameworks>$([TargetFrameworks]);net6.0-tizen</TargetFrameworks> -->
    <OutputType>Exe</OutputType>
    <RootNamespace>MauiApp1</RootNamespace>
    <UseMaui>true</UseMaui>
    <SingleProject>true</SingleProject>
    <ImplicitUsings>enable</ImplicitUsings>

    <!-- Display name -->
    <ApplicationTitle>MauiApp1</ApplicationTitle>

    <!-- App Identifier -->
    <ApplicationId>com.companyname.mauiapp1</ApplicationId>
    <ApplicationIdGuid>2128AE37-4BEC-46F8-B35E-B17F617BB30D</ApplicationIdGuid>

    <!-- Versions -->
    <ApplicationDisplayVersion>1.0</ApplicationDisplayVersion>
    <ApplicationVersion>1</ApplicationVersion>

    <SupportedOSPlatformVersion Condition="\"$([MSBuild]::GetTargetPlatformIdentifier('$(TargetFramework)')\" == 'ios')\">14.2</SupportedOSPlatformVersion>
    <SupportedOSPlatformVersion Condition="\"$([MSBuild]::GetTargetPlatformIdentifier('$(TargetFramework)')\" == 'maccatalyst')\">14.0</SupportedOSPlatformVersion>
    <SupportedOSPlatformVersion Condition="\"$([MSBuild]::GetTargetPlatformIdentifier('$(TargetFramework)')\" == 'android')\">21.0</SupportedOSPlatformVersion>
    <SupportedOSPlatformVersion Condition="\"$([MSBuild]::GetTargetPlatformIdentifier('$(TargetFramework)')\" == 'windows')\">10.0.17763.0</SupportedOSPlatformVersion>
    <TargetPlatformMinVersion Condition="\"$([MSBuild]::GetTargetPlatformIdentifier('$(TargetFramework)')\" == 'windows')\">10.0.17763.0</TargetPlatformMinVersion>
    <SupportedOSPlatformVersion Condition="\"$([MSBuild]::GetTargetPlatformIdentifier('$(TargetFramework)')\" == 'tizen')\">6.5</SupportedOSPlatformVersion>
  </PropertyGroup>

  <ItemGroup>
    <!-- App Icon -->
    <MauiIcon Include="Resources\Icon\appicon.svg" ForegroundFile="Resources\Icon\appiconfg.svg" Color="#512BD4" />

    <!-- Splash Screen -->
    <MauiSplashScreen Include="Resources\Splash\splash.svg" Color="#512BD4" BaseSize="128,128" />

    <!-- Images -->
    <MauiImage Include="Resources\Images\*" />
    <MauiImage Update="Resources\Images\dotnet_bot.svg" BaseSize="168,208" />

    <!-- Custom Fonts -->
    <MauiFont Include="Resources\Fonts\*" />

    <!-- Raw Assets (also remove the "Resources\Raw" prefix) -->
    <MauiAsset Include="Resources\Raw\**" LogicalName="%(RecursiveDir)%(Filename)%(Extension)" />
  </ItemGroup>

```

100 % No issues found Ln: 1 Ch: 1 TABS CRLF

Error List Output

Solution Explorer

Solution 'MauiApp1' (1 of 1 project)

- MauiApp1
 - Dependencies
 - Properties
 - Platforms
 - Android
 - iOS
 - MacCatalyst
 - Tizen
 - Windows
 - Resources
 - App.xaml
 - AppShell.xaml
 - MainPage.xaml
 - MauiProgram.cs

Properties

XML Document

Misc

Encoding Unicode (UTF-8)

Output

Schemas "C:\Program Files\Microsoft Visual Studio\2022\Community\SDK\NuGetFallbackFolder\Microsoft.Maui\6.0.0\ref\net6.0\Microsoft.Maui.dll"

Stylesheet

Encoding

Character encoding of the document.

Add to Source Control Select Repository

Demo Retro

- HybridWebView since .NET 9
- Bundle your WebApp with your .NET MAUI App
- Use .NET MAUI and it's NuGet friends
- Use your favourite JS-Frontendframework

@mallibone

No Hotreload 😭



@mallibone

You are here

.NET 7 Nov 2022

.NET 8 Nov 2023 May 2024

.NET 9 Nov 2024

.NET 10 Nov 2025 May 2026

.NET 11 Nov 2026

Latest release

STANDARD TERM SUPPORT
Patches for 18 months

LONG TERM SUPPORT
Patches for 36 months

<https://dotnet.microsoft.com/en-us/platform/support/policy/dotnet-core>

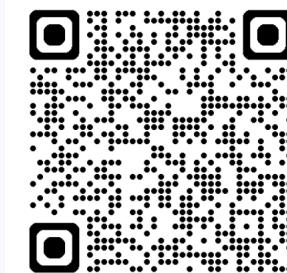
@mallibone

› Blogs > .NET Blog > .NET 10 Preview 5 is now available!

June 10th, 2025

3 reactions

.NET 10 Preview 5 is now available!



.NET

.NET Team

Today, we are excited to announce the **fifth** preview release of .NET 10 with enhancements across the .NET Runtime, SDK, libraries, C#, ASP.NET Core, Blazor, .NET MAUI, and more. Check out the full release notes linked below and get started today.

[Download .NET 10 Preview 5](#)



3



1



7

This release contains the following improvements.

Intercept Web Requests

The `HybridWebView` now allows you to intercept when the browser requests a web resource in order to take action before it executes, such as adding a header to the request. To do this, add a listener to the `WebResourceRequested` event.

```
<HybridWebView WebResourceRequested="HybridWebView_WebResourceRequested" />
```

```
private void HybridWebView_WebResourceRequested(object sender, HybridWebViewWebResourceRequestedEventArgs e)
{
    // NOTES:
    // * This method MUST be synchronous, as it is called from the WebView's thread.
    // * This method MUST return a response (even if it is not yet complete), otherwise the
    //   WebView may freeze or return an error response.
    // * The response must be set using the SetResponse method of the event args.

    // Only handle requests for the specific image URL
    if (!e.Uri.ToString().Contains("sample-image.png"))
        return;

    // Prevent the default behavior of the web view
    e.Handled = true;

    // Return the stream or task of stream that contains the content
    // NOTE: the method is NOT awaited, the WebView will continue to load the content
    e.SetResponse(200, "OK", "image/png", GetStreamAsync());
}
```

.NET for Android

Intercept Web Requests

The `HybridWebView` now allows you to intercept when the browser requests a web resource in order to take action before it executes, such as adding a header to the request. To do this, add a listener to the `WebResourceRequested` event.

```
<HybridWebView WebResourceRequested="HybridWebView_WebResourceRequested" />
```

```
private void HybridWebView_WebResourceRequested(object sender, HybridWebViewWebResourceRequestedEventArgs e)
```

```
{  
    // NOTES:  
    // * This method MUST be synchronous, as it is called from the WebView's thread.  
    // * This method MUST return a response (even if it is not yet complete), otherwise the  
    //   WebView may freeze or return an error response.  
    // * The response must be set using the SetResponse method of the event args.
```

```
    // Only handle requests for the specific image URL  
    if (!e.Uri.ToString().Contains("sample-image.png"))  
        return;  
  
    // Prevent the default behavior of the web view  
    e.Handled = true;  
  
    // Return the stream or task of stream that contains the content  
    // NOTE: the method is NOT awaited, the WebView will continue to load the content  
    e.SetResponse(200, "OK", "image/png", GetStreamAsync());  
}
```

.NET for Android

Intercept Web Requests

The `HybridWebView` now allows you to intercept when the browser requests a web resource in order to take action before it executes, such as adding a header to the request. To do this, add a listener to the `WebResourceRequested` event.

```
<HybridWebView WebResourceRequested="HybridWebView_WebResourceRequested" />
```

```
private void HybridWebView_WebResourceRequested(object sender, HybridWebViewWebResourceRequestedEventArgs e)
```

```
{
```

```
    // NOTES:
```

```
    // * This method MUST be synchronous, as it is called from the WebView's thread.
```

```
    // * This method MUST return a response (even if it is not yet complete), otherwise the  
    //   WebView may freeze or return an error response.
```

```
    // * The response must be set using the SetResponse method of the event args.
```

```
    // Only handle requests for the specific image URL
```

```
    if (!e.Uri.ToString().Contains("sample-image.png"))  
        return;
```

```
    // Prevent the default behavior of the web view
```

```
    e.Handled = true;
```

```
    // Return the stream or task of stream that contains the content
```

```
    // NOTE: the method is NOT awaited, the WebView will continue to load the content
```

```
    e.SetResponse(200, "OK", "image/png", GetStreamAsync());
```

```
}
```

.NET for Android

Intercept Web Requests

The `HybridWebView` now allows you to intercept when the browser requests a web resource in order to take action before it executes, such as adding a header to the request. To do this, add a listener to the `WebResourceRequested` event.

```
<HybridWebView WebResourceRequested="HybridWebView_WebResourceRequested" />

private void HybridWebView_WebResourceRequested(object sender, HybridWebViewWebResourceRequestedEventArgs e)
{
    // NOTES:
    // * This method MUST be synchronous, as it is called from the WebView's thread.
    // * This method MUST return a response (even if it is not yet complete), otherwise the
    //   WebView may freeze or return an error response.
    // * The response must be set using the SetResponse method of the event args.

    // Only handle requests for the specific image URL
    if (!e.Uri.ToString().Contains("sample-image.png"))
        return;

    // Prevent the default behavior of the web view
    e.Handled = true;

    // Return the stream or task of stream that contains the content
    // NOTE: the method is NOT awaited, the WebView will continue to load the content
    e.SetResponse(200, "OK", "image/png", GetStreamAsync());
}
```

.NET for Android

Intercept Web Requests

The `HybridWebView` now allows you to intercept when the browser requests a web resource in order to take action before it executes, such as adding a header to the request. To do this, add a listener to the `WebResourceRequested` event.

```
<HybridWebView WebResourceRequested="HybridWebView_WebResourceRequested" />

private void HybridWebView_WebResourceRequested(object sender, HybridWebViewWebResourceRequestedEventArgs e)
{
    // NOTES:
    // * This method MUST be synchronous, as it is called from the WebView's thread.
    // * This method MUST return a response (even if it is not yet complete), otherwise the
    //   WebView may freeze or return an error response.
    // * The response must be set using the SetResponse method of the event args.

    // Only handle requests for the specific image URL
    if (!e.Uri.ToString().Contains("sample-image.png"))
        return;

    // Prevent the default behavior of the web view
    e.Handled = true;

    // Return the stream or task of stream that contains the content
    // NOTE: the method is NOT awaited, the WebView will continue to load the content
    e.SetResponse(200, "OK", "image/png", GetStreamAsync());
}
```

.NET for Android

dotnet / maui

Type to search

Code Issues 3.8k Pull requests 521 Discussions Actions Projects 2 Wiki Security Insights

is:issue state:open|label:area-controls-hybridwebview

Open 19 Closed 33 Author Labels Projects Milestones Assignees Types Newest

MAUI HybridWebView Zoom area-controls-hybridwebview proposal/open #30126 · OvrBtn opened last week · .NET 11 Planning

[iOS] HybridWebView "Resource Not Found" Issue After Returning from Background area-controls-hybridwebview platform/ios s/needs-attention t/bug #29738 · freedev82 opened on May 30

HybridWebView on Android doesn't allow Select dropdowns area-controls-hybridwebview platform/android s/triaged s/verified t/bug #29454 · lhughey opened on May 12 · Backlog

[iOS] HybridWebView can't display an iFrame area-controls-hybridwebview platform/ios s/needs-attention s/triaged s/verified t/bug #29396 · Saurbaum opened on May 8 · Backlog

Add method for BlazorWebView to execute JavaScript area-controls-hybridwebview proposal/open #29383 · spadapet opened on May 7 · .NET 10 Planning

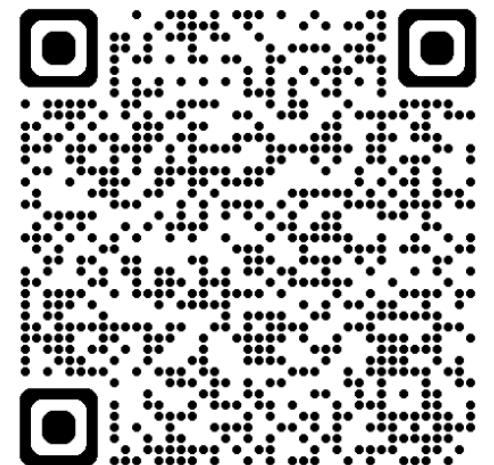
[Android] HybridWebView crash app area-controls-hybridwebview platform/android t/bug #28801 · michalpobuta opened on Apr 4 · Backlog

Provide more customization features for BlazorWebView & HybridWebView 1/6 area-blazor area-controls-hybridwebview proposal/underconsideration Feature #28202 · BethMassi opened on Mar 6 · .NET 10.0-rc1

[Enhancement] HybridWebView.SetInvokeJavaScriptTarget() Multi-target support OR Generator to produce target area-controls-hybridwebview proposal/open t/enhancement #27990 · Sane7222 opened on Feb 23 · Backlog

HybridWebView throwing exceptions in Windows even with simplest configuration (causing crashes) area-controls-hybridwebview platform/windows s/needs-attention s/triaged s/verified t/bug #27954 · jomdev opened on Feb 21 · .NET 9 SR9

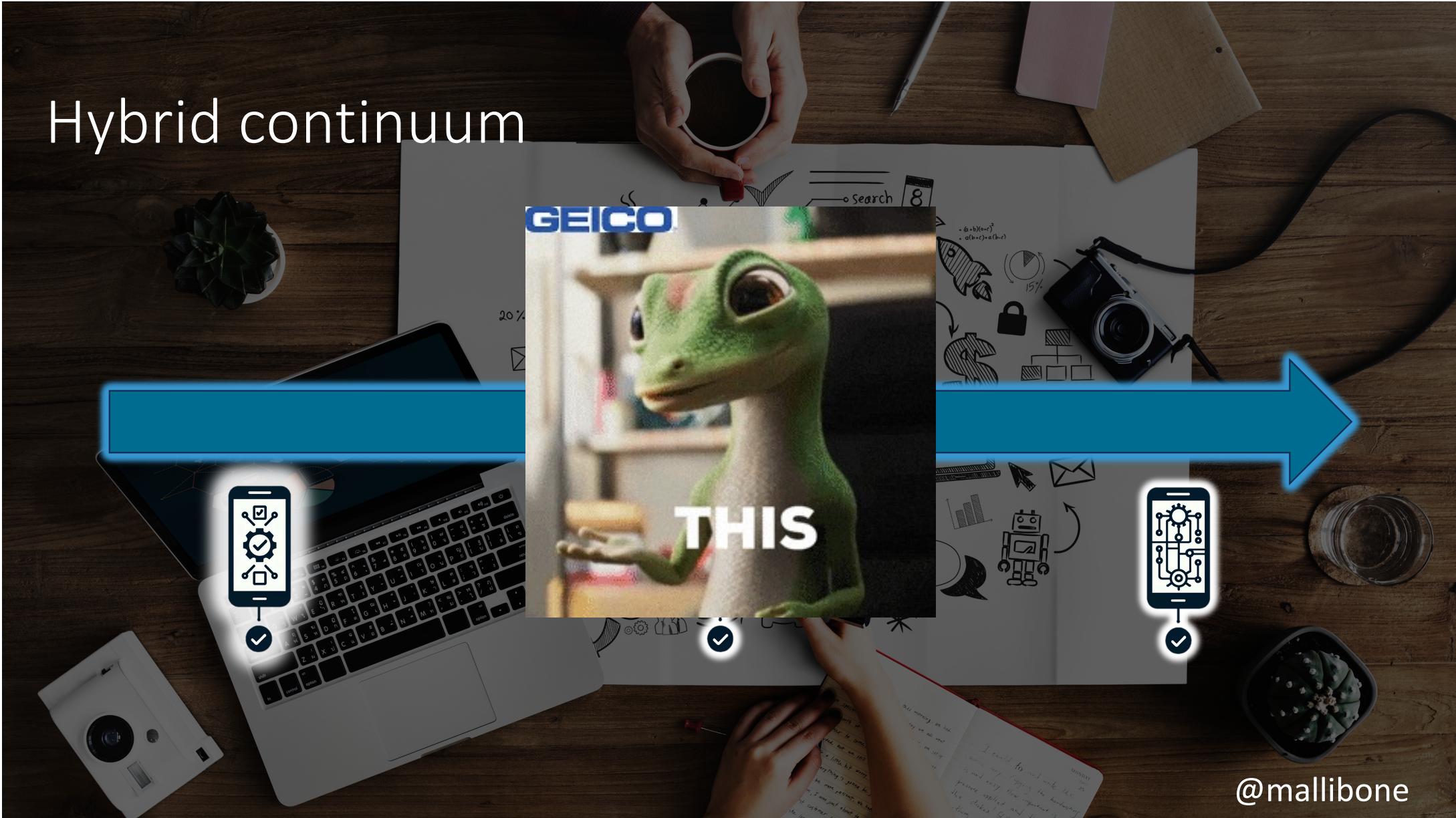
HybridWebView.InvokeJavaScriptAsync is not super user-friendly area-controls-hybridwebview s/triaged Feature #27382 · mattleibow opened on Jan 27 · Backlog



A photograph of a silver service bell with a glass dome resting on a wooden desk. In the background, a lamp and a book are visible.

Reality check

Hybrid continuum



@mallibone

Choose the option that makes business sense and not because it is cool

@mallibone

AND THEN WE JUST DEPLOY THE SAME APP

Dev

Marketing

EVERWHERE

imgflip.com

@mallibone

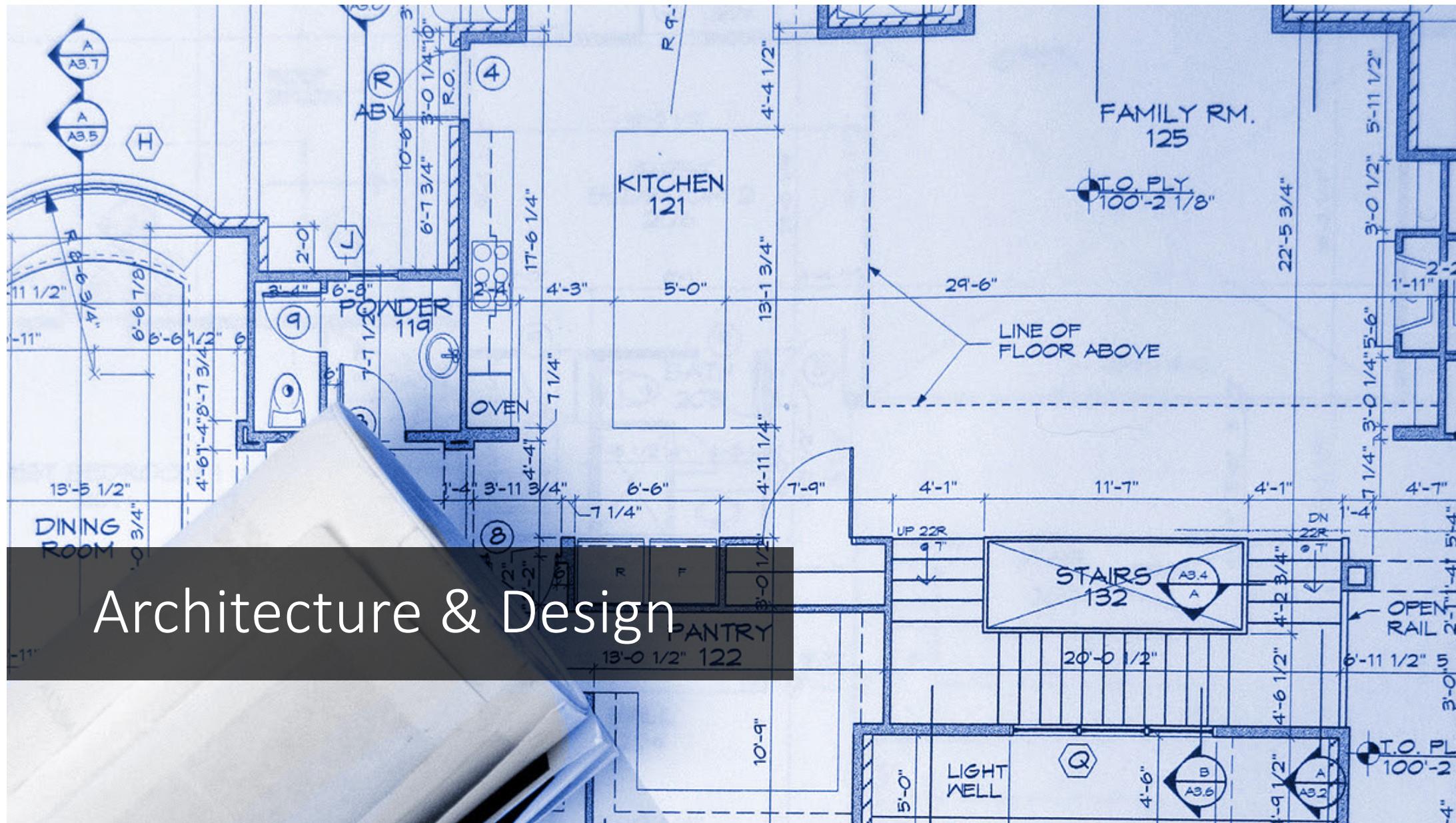


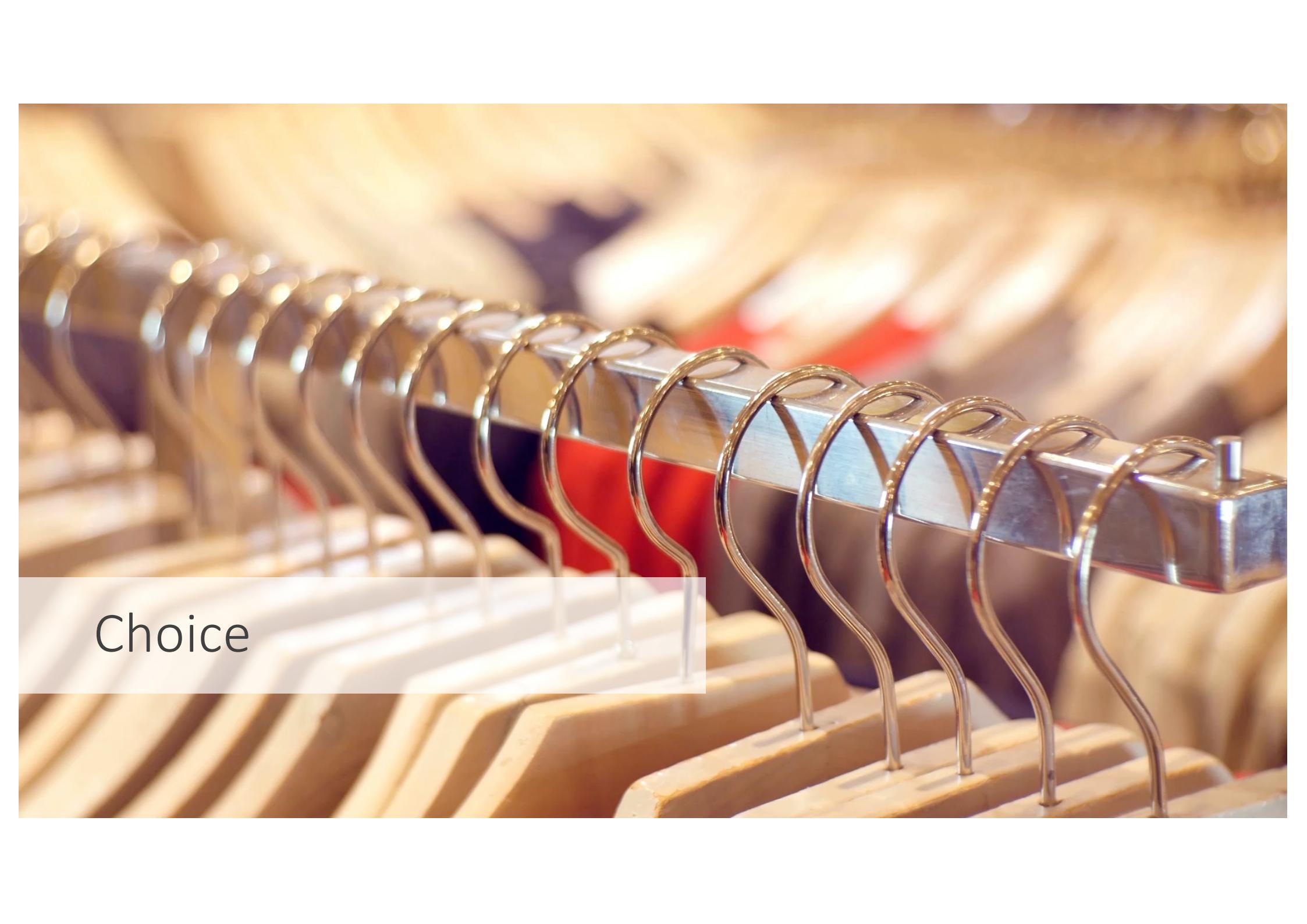
Multiplatform vs
Multiparadigm

[This Photo](#) by Unknown Author is licensed under [CC BY-NC](#)

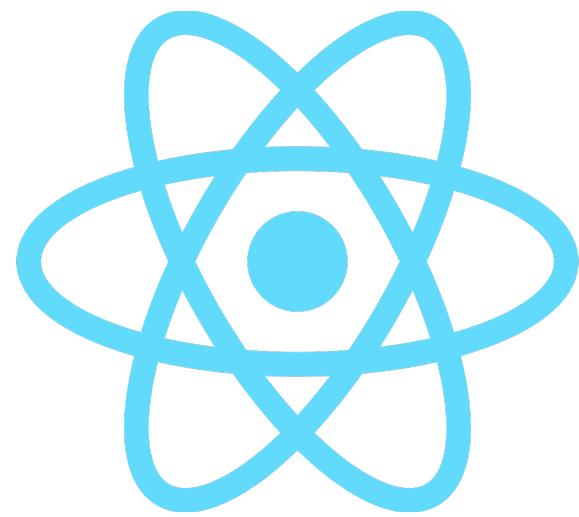
@mallibone

Architecture & Design

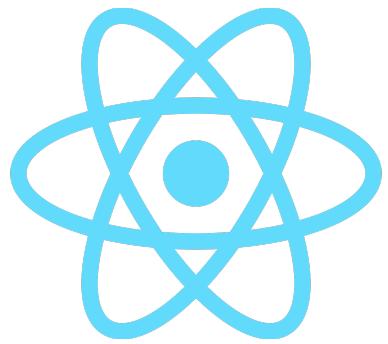




Choice



@mallibone



@mallibone



@mallibone



@mallibone

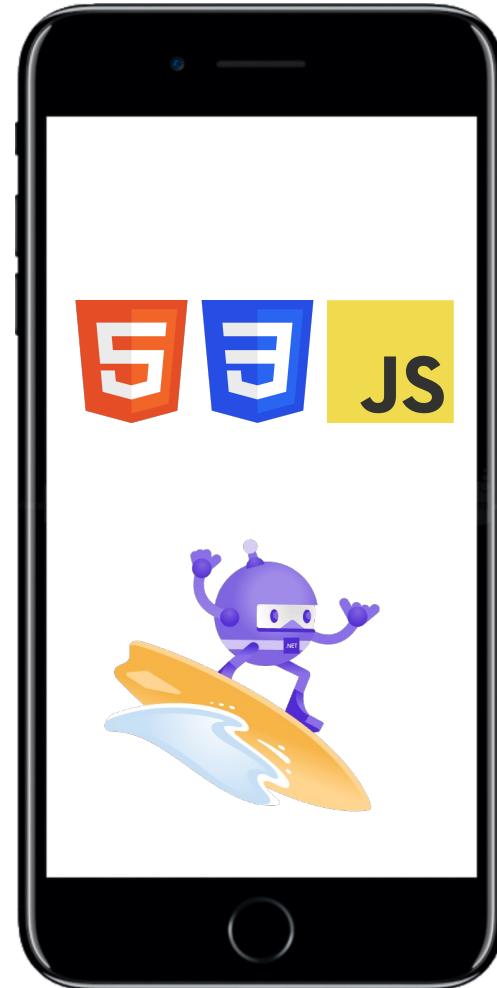
Takeaways



@mallibone

Takeaways

- Bring your web apps to mobile using .NET MAUI
- Use your favourite Single-Page-Application framework
- HybridWebView makes working with JS-Hybrid apps easier



@mallibone



Thank you for your time!



Mark Allibone



@mallibone.com



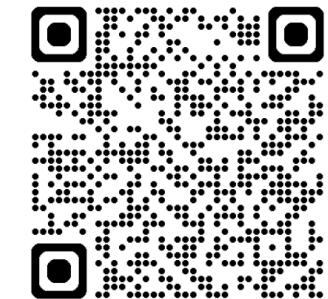
iDIP Solution



<https://mallibone.com>



<https://nullpointers.io>



Scan me

