

Hybrid Web and Native apps with .NET MAUI and Blazor

Rachel Kang

Product Manager @ Microsoft

rachelkang@microsoft.com



Hi, I'm Rachel!

- Product Manager @ Microsoft
 - Started as a software engineer
 - ~ 5 years on .NET MAUI team
 - .NET MAUI, VS + VS Code tools, accessibility, and AI
- New York → Boston → D.C.
- Lately I've been... eating, sleeping, being a tourist, learning about the universe

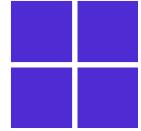


Agenda

- Overview
- .NET MAUI
- Demo
- Blazor
- Blazor Hybrid
- Break
- Demo - Lab
- Close

Microsoft's investments in client development

Windows Native



WinUI3

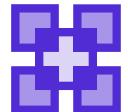


WinForms

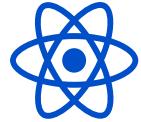


WPF

Cross Platform Native



.NET MAUI



React Native

Hybrid



Blazor Hybrid



WebView2

Web



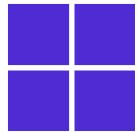
Progressive
Web Apps



Blazor Web

Benefits of these types of client apps

Windows Native



Deepest native experience on Windows

Cross Platform Native



Native experiences you write once and run on multiple devices

Hybrid



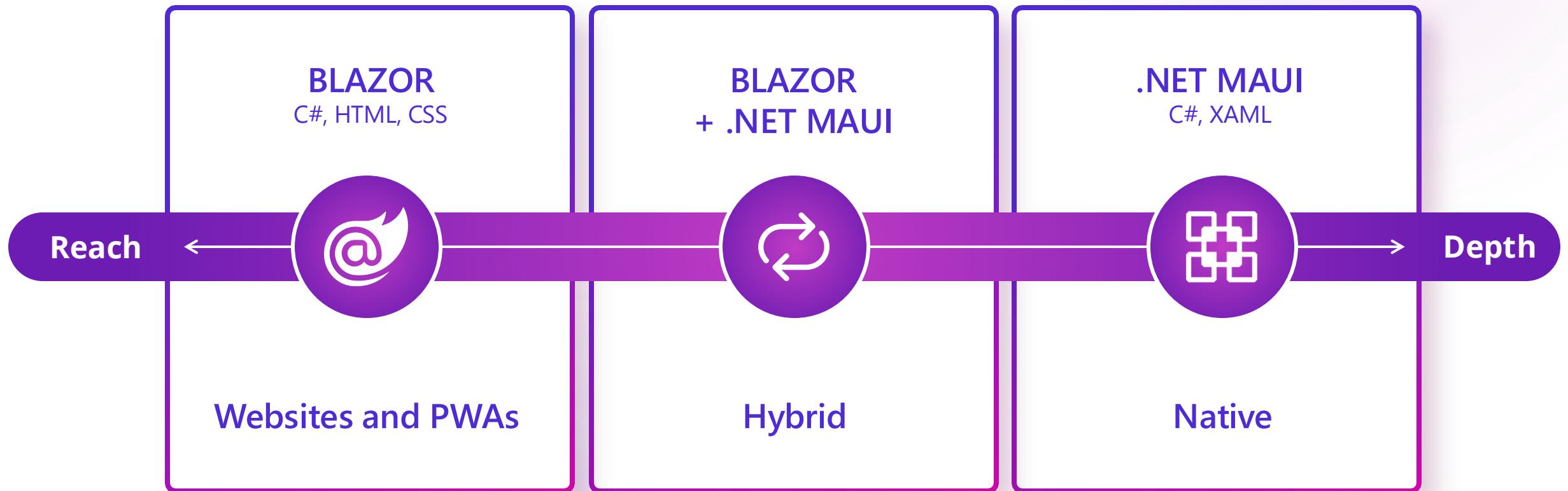
Native experiences with web UI and code reuse across devices and web browsers

Web



Maximum reach across any web browser

.NET offers choice for client app developers



.NET MAUI

.NET Multi-platform App UI (MAUI)

Single project system, single codebase

WinUI



Mac Catalyst



iOS



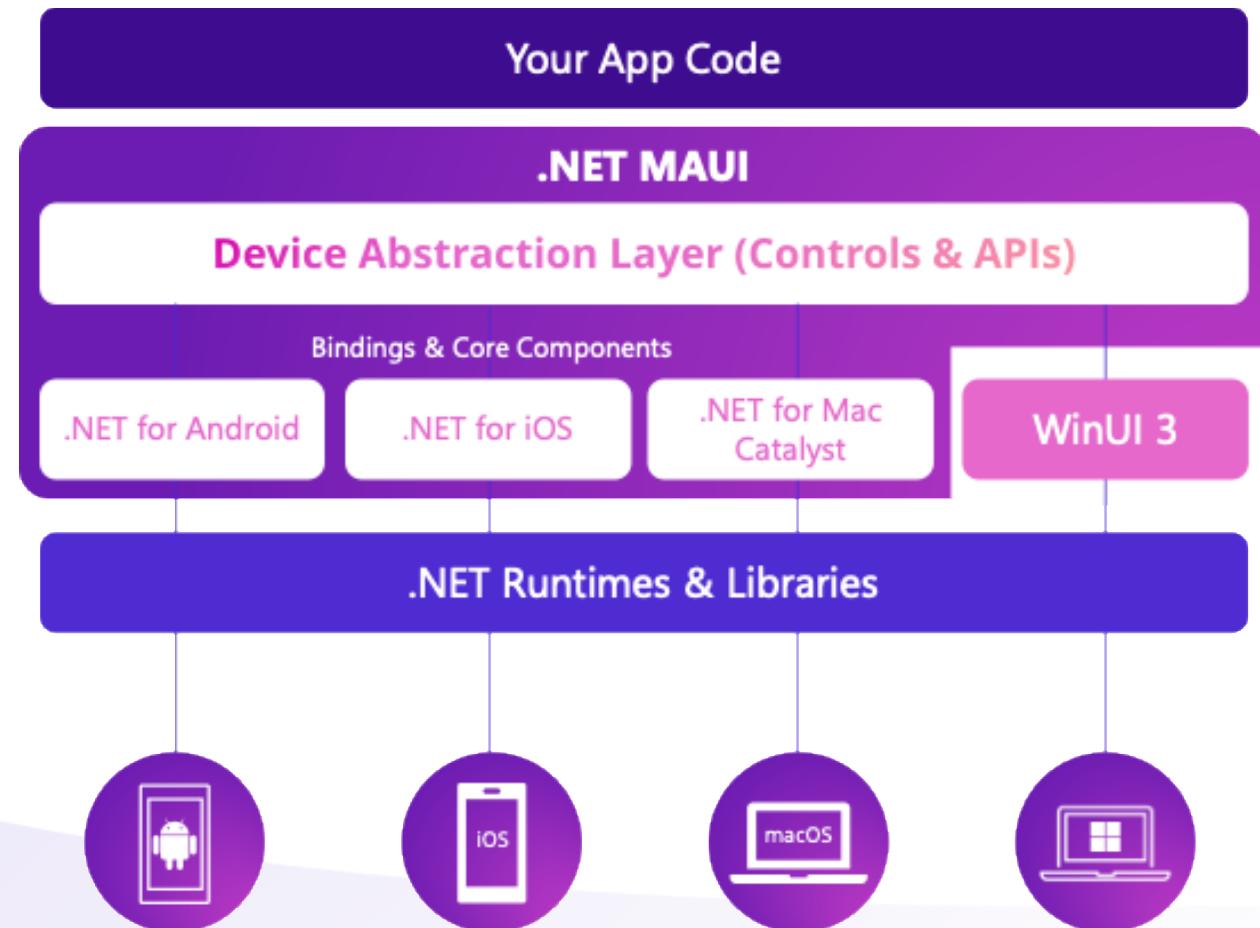
Android



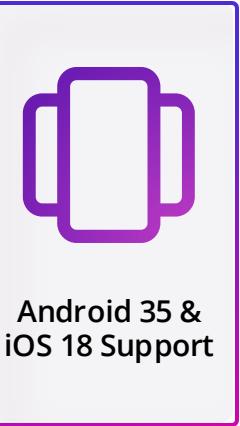
.NET Multi-platform App UI (MAUI)

Build native, multi-platform apps all in one framework

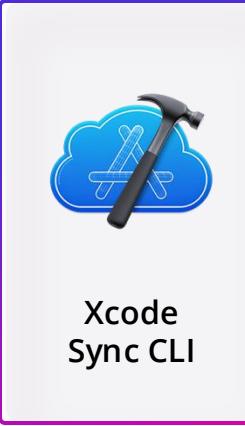
- Cross-platform native mobile and desktop apps for Android, iOS, Mac, Windows
- Modern native client stack for .NET
- Single project, single codebase
- Native controls and styling
- Access over 60 common device APIs
- Access platform specific APIs if needed
- Built on top of .NET, with access to Base Class Libraries
- App store distribution and reach
- Productivity tools for Visual Studio and VS Code
- Platform is open source on GitHub
- Healthy ecosystem of controls and components



.NET MAUI in .NET 9



Android 35 &
iOS 18 Support



Xcode
Sync CLI



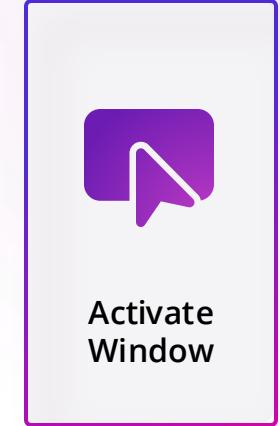
iOS
Multi-target
bindings



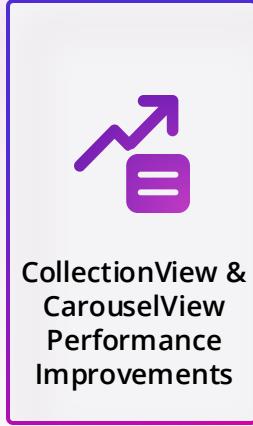
Native Library
Interop



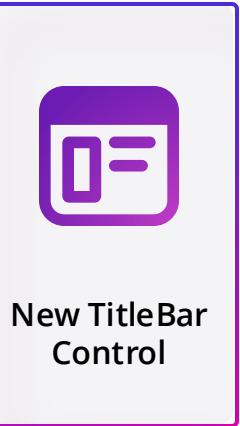
New lifecycle
events



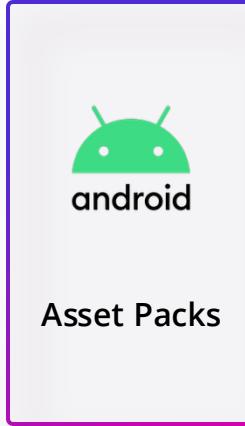
Activate
Window



CollectionView &
CarouselView
Performance
Improvements



New TitleBar
Control



Asset Packs



Native
Embedding
Enhancements



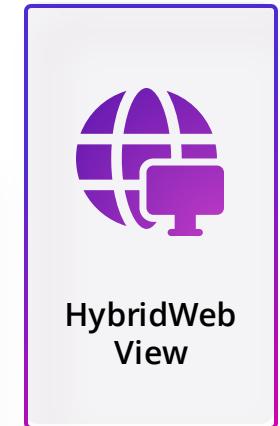
New Input
Keyboards



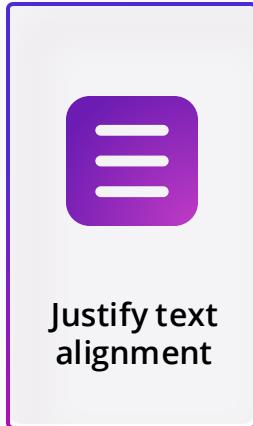
Native AOT
improvements



Trimming
Enhancements



HybridWeb
View



Justify text
alignment



.NET MAUI extension for vs Code

XAML intellisense, tooltips, auto-insert, color swatches, & more

Hot reload for XAML and C#

Xcode sync support with new dotnet tool



Visual Studio 2022

17.12, 17.13 Stable

New .NET MAUI sample template

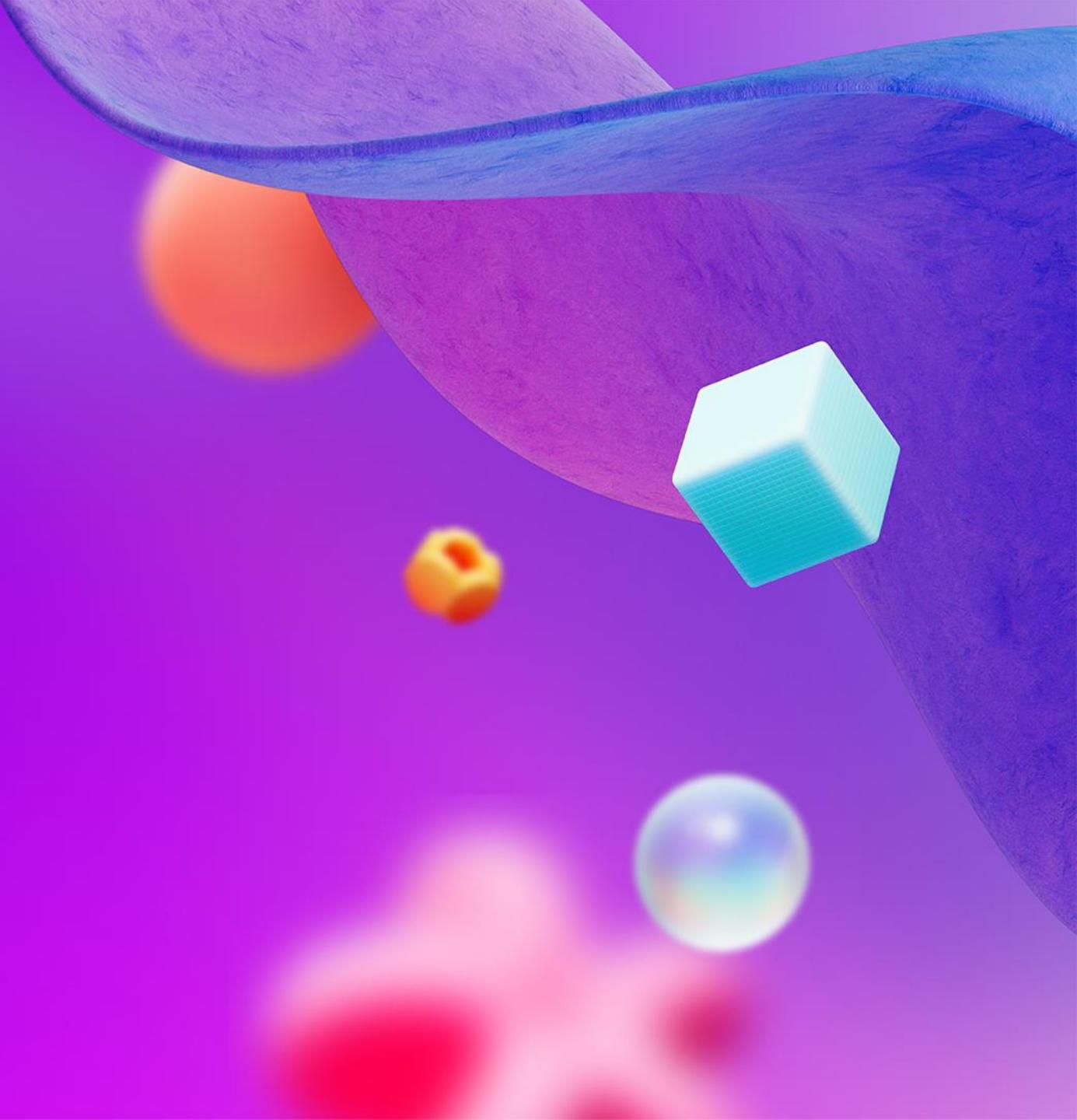
Full Page Hot Reload

GitHub Copilot improvements for .NET MAUI

17.14 Previews

(Preview) Live Preview at design time

DEMO
.NET MAUI



Blazor

Blazor

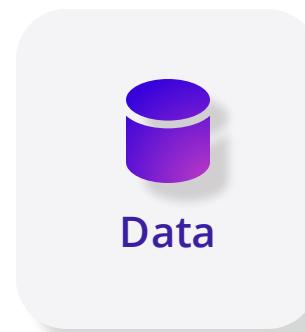
Your full stack web platform with .NET



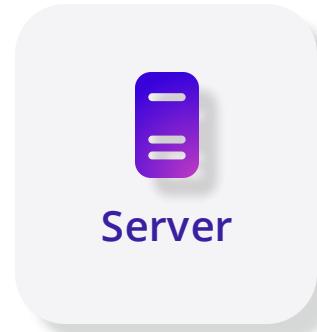
Web UI



APIs



Data



Server



Security



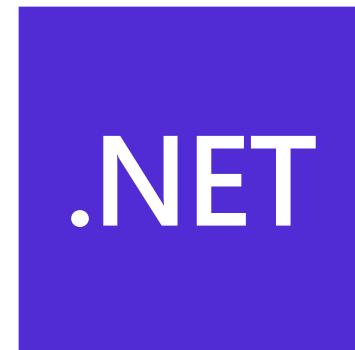
Tooling



https://...

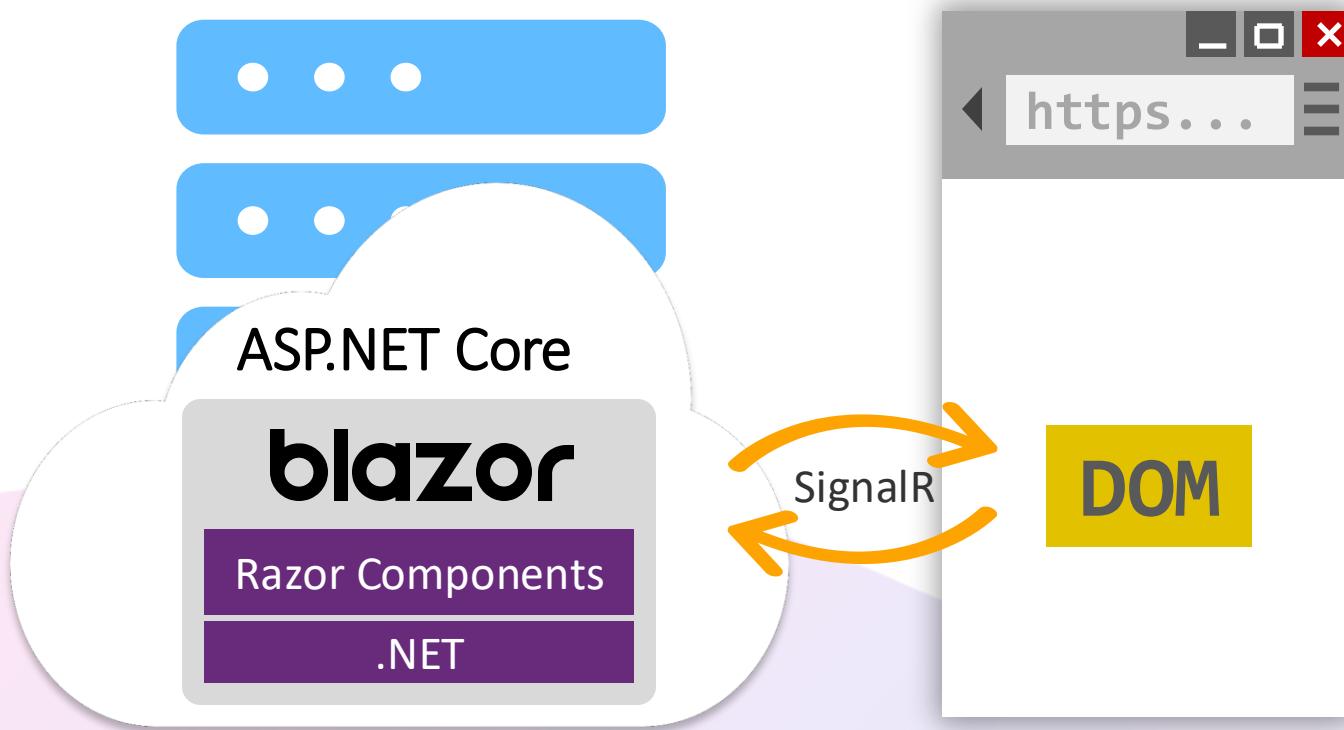


Blazor

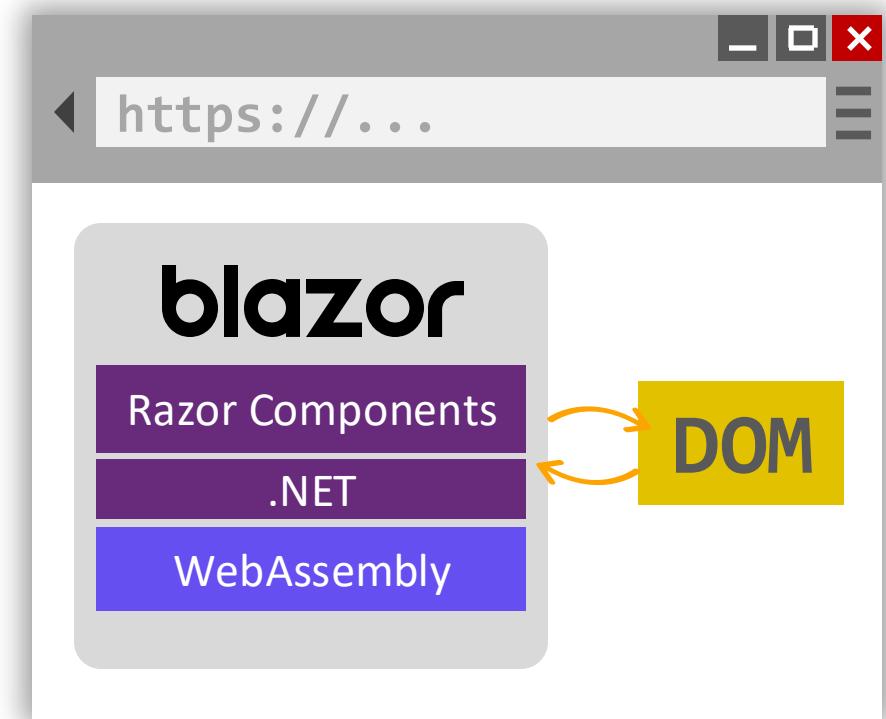


How it works

Blazor Server



Blazor WebAssembly



Choosing your Blazor hosting model

Blazor Server

- ✓ Full access to server capabilities
- ✓ Fast to startup
- ✓ Code never leaves the server
- ✓ **Supports older browsers & thin clients**
- ✗ Requires persistent connection
- ✗ Higher UI latency

Blazor WebAssembly

- ✓ Runs fully client-side
- ✓ No required server component
- ✓ Host as a static site
- ✓ Can execute offline
- ✗ Larger download size
- ✗ Slower runtime performance

Modern web UI with .NET & Blazor



Server



WebAssembly



Hybrid

Blazor Hybrid

What are hybrid apps and why build them?

01 Blend of native and web technology:

UI is written using web technologies such as HTML, CSS, and JavaScript wrapped in a native app container.

02 Access to native platform features and device hardware:

leverage native platform features and device hardware that a web application cannot access easily or even at all.

03 Can mean mobile and/or desktop devices:

run on multiple device platforms without the need for completely separate development.

04 Reach via app store distribution:

hybrid apps are native apps that can be downloaded and installed through the app stores.

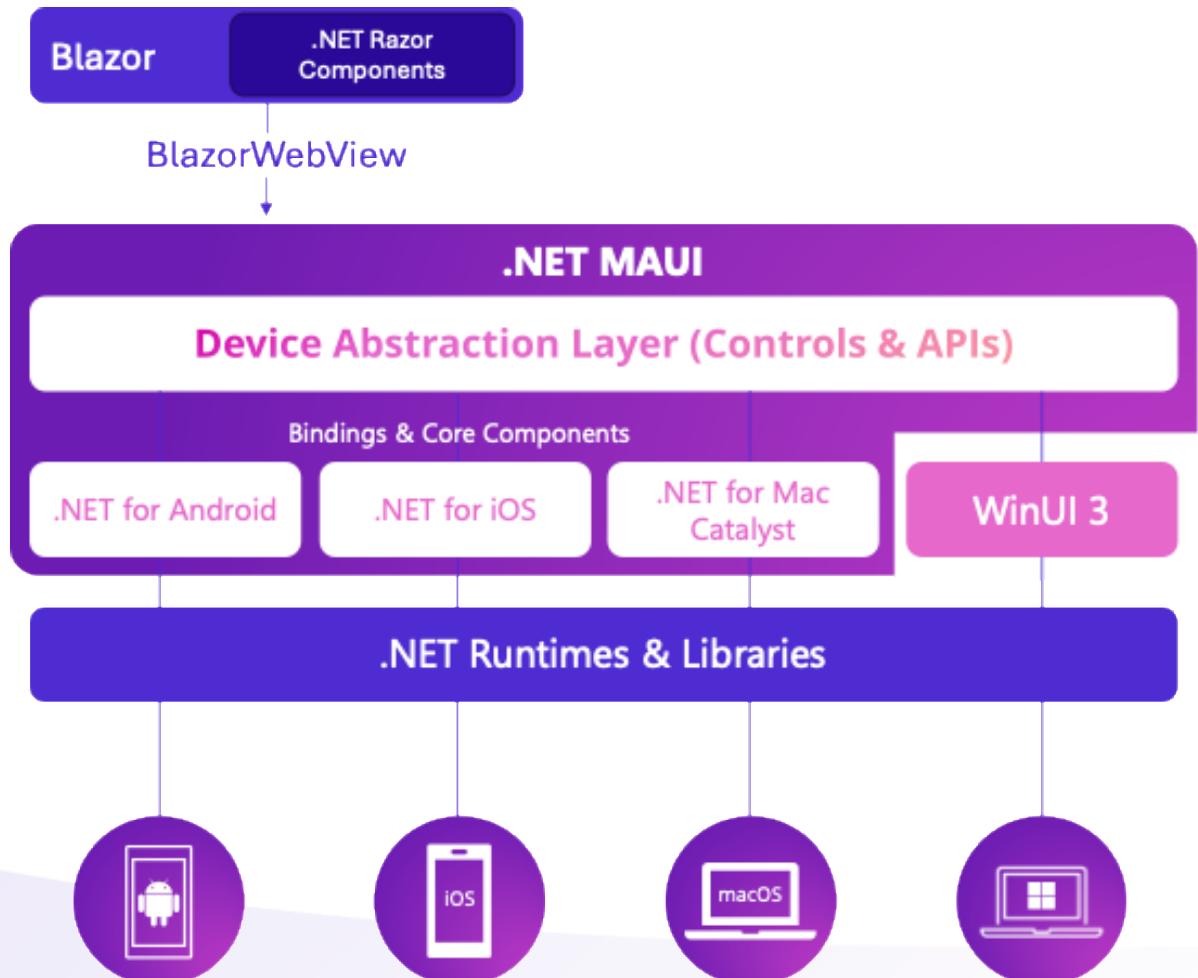
05 Code reuse across device platforms and web browsers:

achieve greater developer productivity via code reuse across devices and web browsers.

.NET MAUI Blazor Hybrid

Build native, multi-platform apps with Blazor web UI.

- Native app container & embedded controls
- Fully native web UI through platform WebView
- No internet required; web assets packaged with app
- Built-in Blazor input controls, validation, forms, auth
- Multi-platform with MAUI abstractions
- App store distribution and reach
- Easily reuse web UI across MAUI native and Blazor web apps with new Solution template
- ... + all the benefits of .NET MAUI & Blazor productivity tooling and unified build with Visual Studio & VS Code



Web UI and Native UI

Look and feel

Web UI typically looks the same across devices (controlled by CSS), while Native UI typically provides device-specific look & feel by default

UX

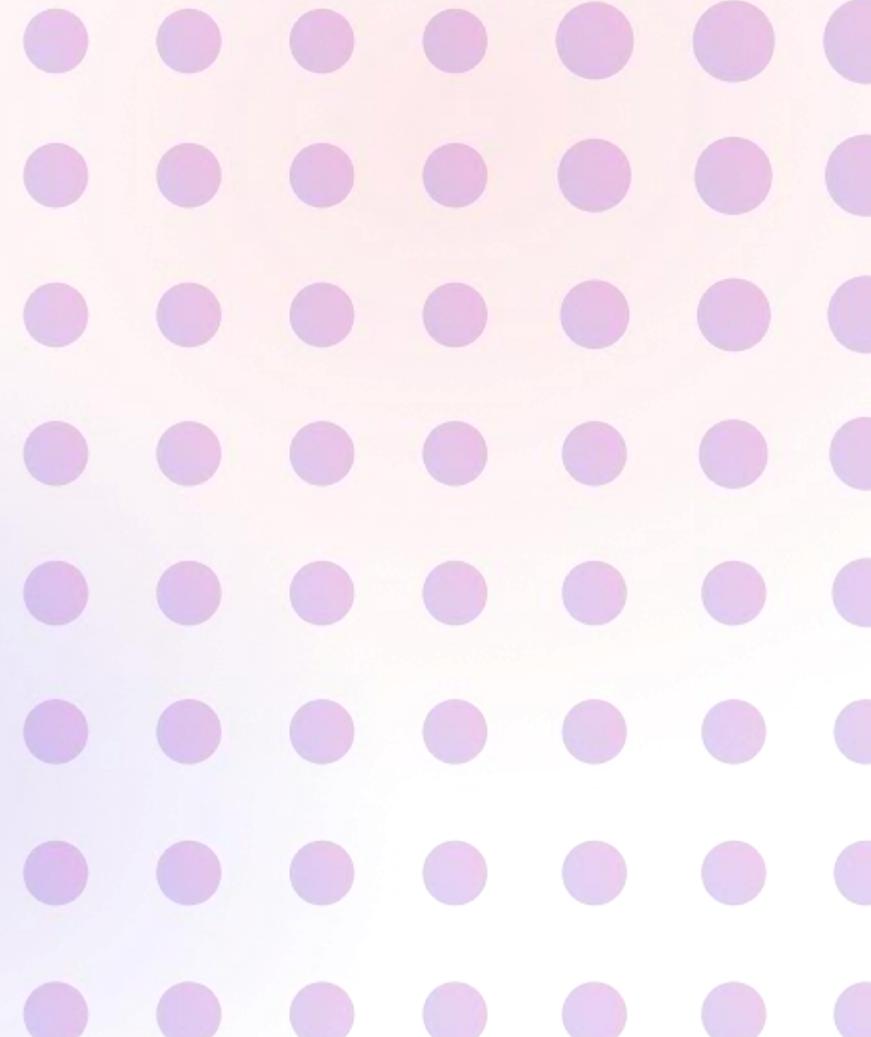
Native UI can provide more seamless user experiences tailored to the specific devices

Code reuse

Web UI can easily be reused in a web app

Skillset

Web UI requires knowledge of HTML, JavaScript, CSS, while Native UI (on the Microsoft stack) requires knowledge of XAML



My Ride K-12

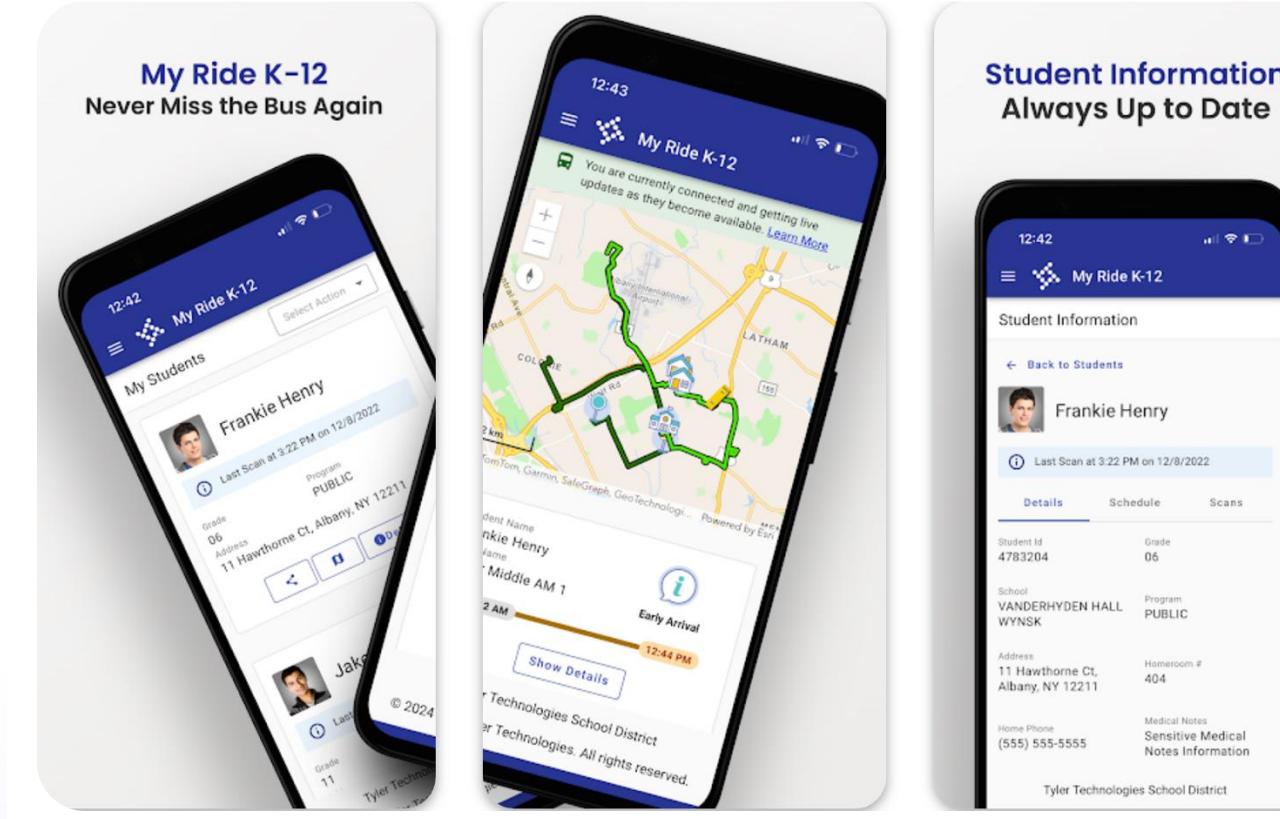


.NET MAUI Blazor Hybrid & Web



On day one, we supported web browsers, tablets, and phones. Every update goes to all platforms. For our QA people, it's just one UI to test. They verify that it works correctly on each platform, but it's the same logic and 99% the same UI across the board. The project has been an overwhelming success for us.

Chris Miller, Software Architect, Tyler Technologies, Inc.



2500+

Schools

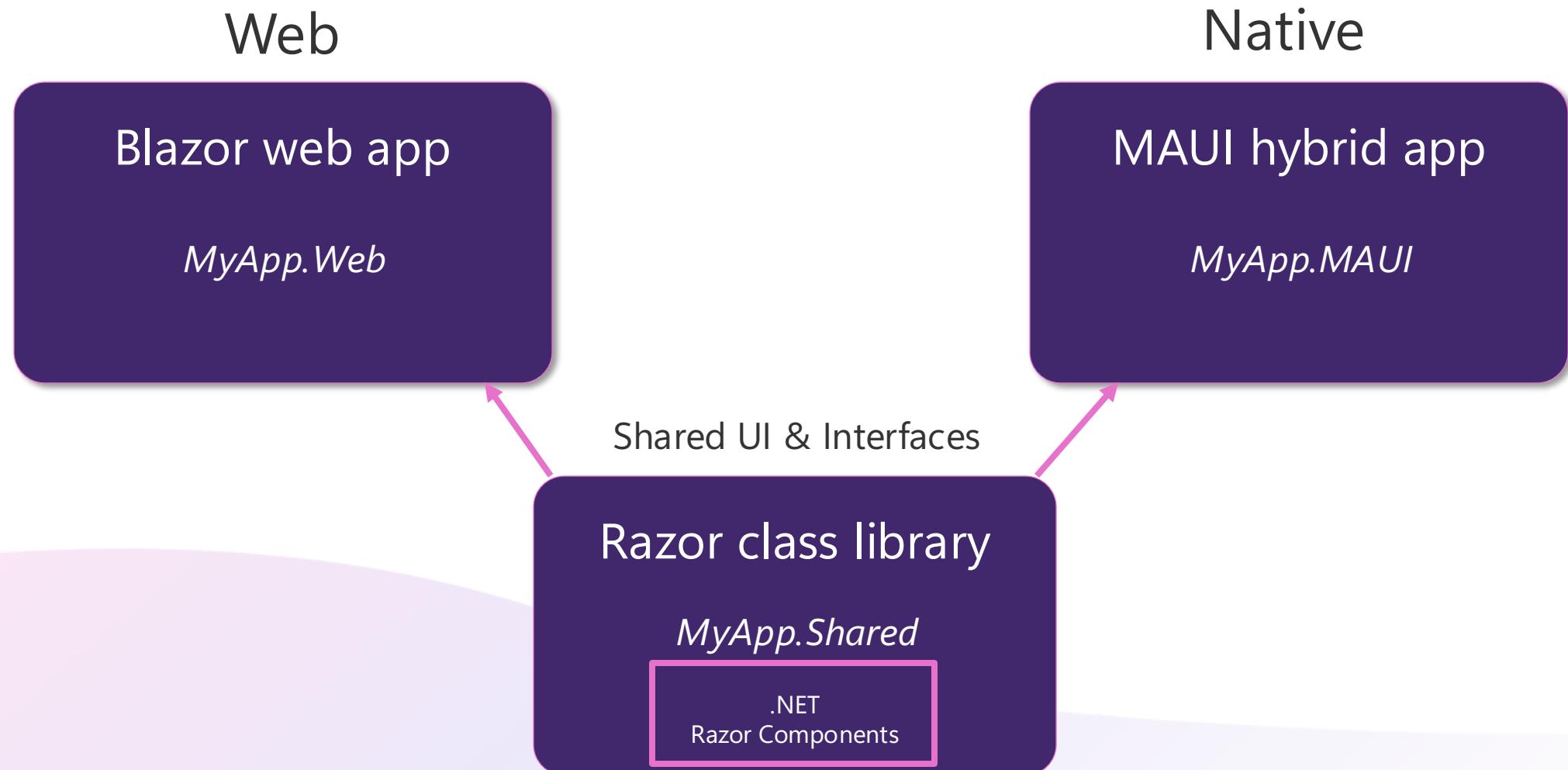
37%

Nation's students
routed by app

750K

Downloads

Reusing UI across native and web



MAUI hybrid apps in .NET 9

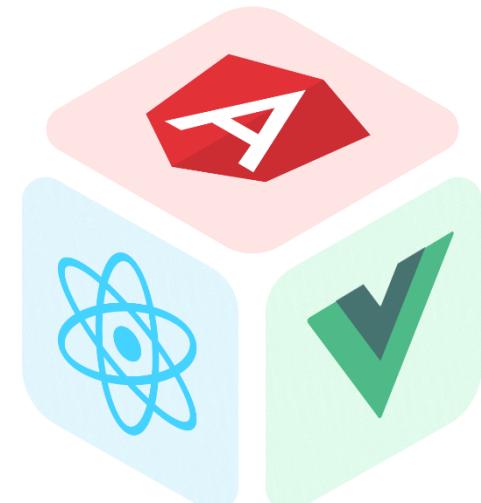
Hybrid apps with Blazor are better than ever

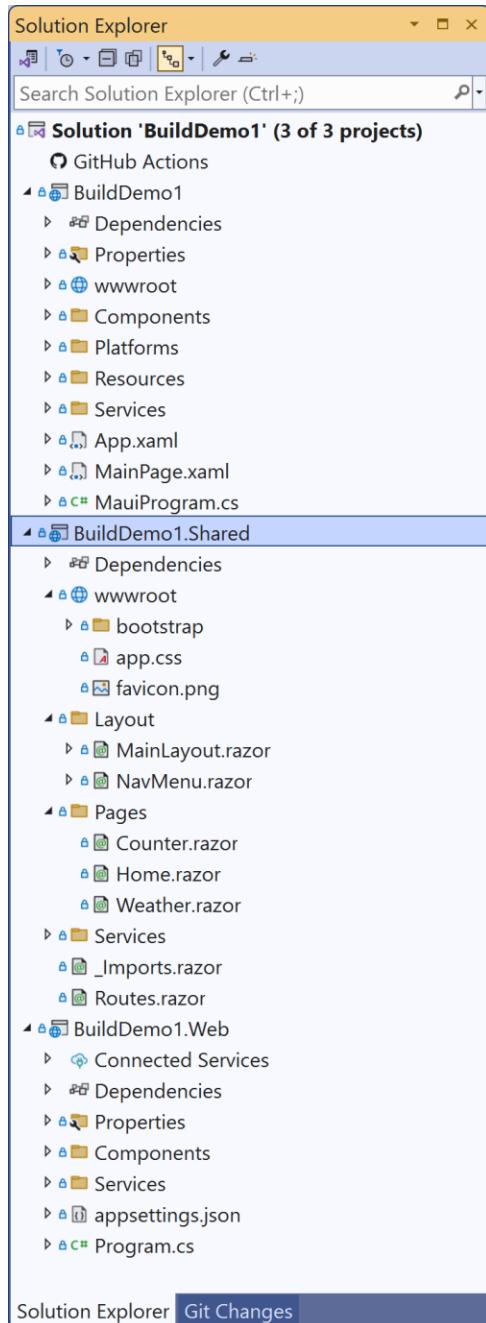
- Solution template for Blazor Hybrid+Web
- Hot reload / inner loop improvements
- .NET MAUI extension for VS Code
- More documentation & bug fixes



Enable hybrid apps with JavaScript frontends

- Reuse Angular, React, etc. investments in MAUI native desktop and mobile apps
- New MAUI control called HybridWebView





.NET MAUI Blazor Hybrid and Web App

- Reuse your UI across MAUI native and Blazor web apps
- Use interface injection to support different hosting models
- Use multi-targeting to create UI for different platforms
- aka.ms/maui-blazor-web

Solution Template available
TODAY with .NET 9

```
PS C:\> dotnet new maui-blazor-web
```

DEMO

.NET MAUI Blazor Hybrid & Web

<https://github.com/dotnet-presentations/build-2025-lab305>



- Hybrid apps are great for web devs that want native platform capabilities & app store distribution
- Reuse web UI across your web apps and native apps with component libraries and interfaces
- Use MAUI for multi-platform abstractions and multi-targeting

Docs: aka.ms/maui-blazor-web

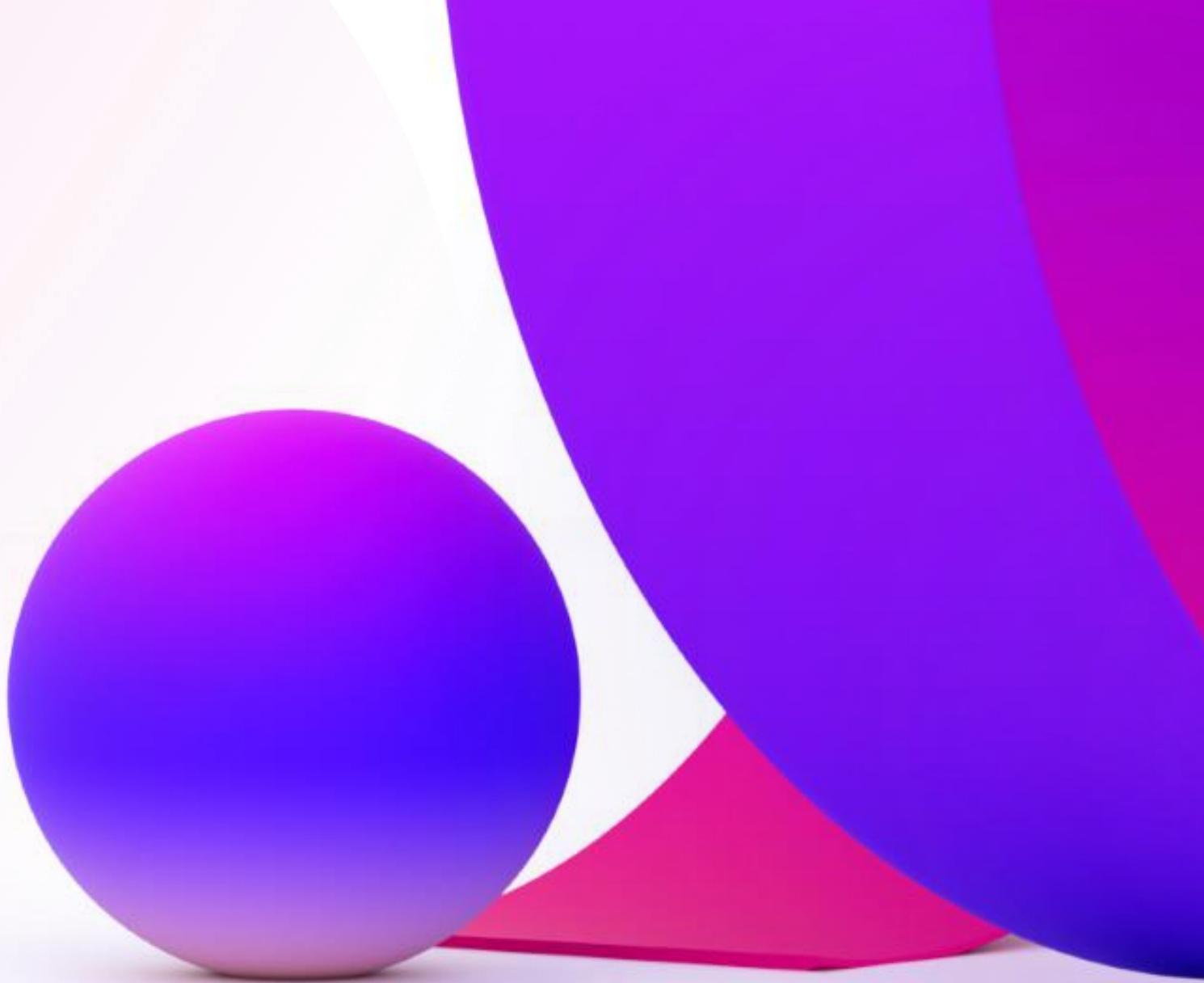
Lab: <https://github.com/dotnet-presentations/build-2025-lab305>

Repo: github.com/dotnet/maui/labels/area-blazor

Microsoft Build

May 19-22, 2025

<https://build.microsoft.com>



Stay in touch!

GitHub

Open issues and join
discussions at
github.com/dotnet/maui

Discord

Have deeper
conversations at
aka.ms/dotnet-discord

YouTube

Catch the monthly
Community Standup at
youtube.com/@dotnet

Thank you!

rachelkang@microsoft.com

@rachelkang on GitHub

@therachelkang on Twitter/X, bsky

[linkedin.com/in/rachel-j-kang](https://www.linkedin.com/in/rachel-j-kang)