

Building a PWA with ASP.NET Core 6 Blazor

Introduction to Developing PWAs with Blazor



Matthew Soucoup

Principal – Code Mill Technologies

@codemillmatt | www.codemillmatt.com

What Are PWAs?

Progressive Web Apps

Combine web and native features

Installable on devices

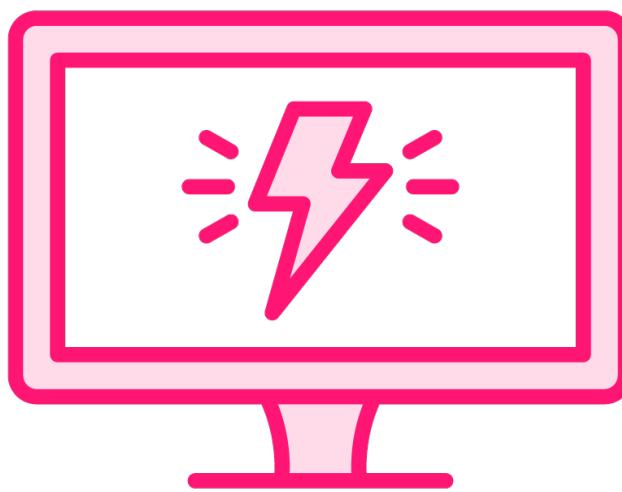
Responsive and fast



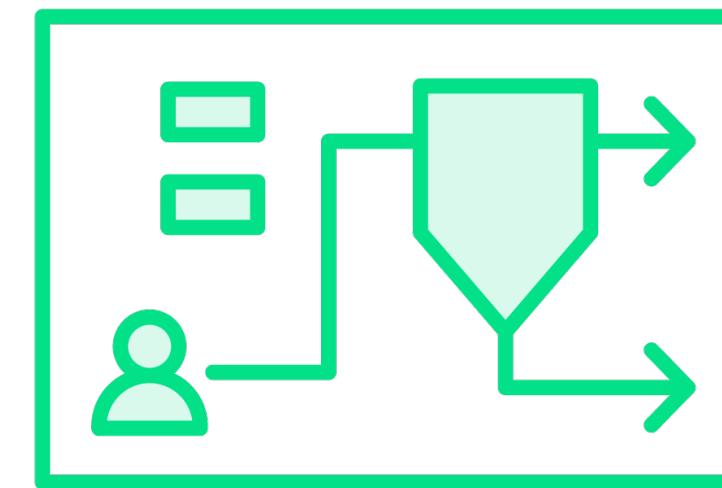
Importance of PWAs



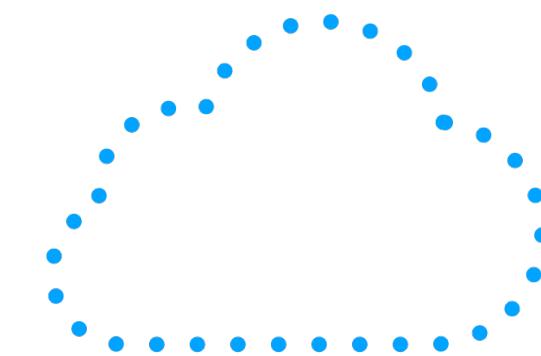
Reach a wider audience



Cross-platform compatibility



Improved user experience



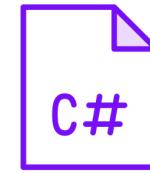
Offline functionality



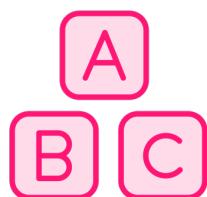
Introduction to Blazor



Framework for building web applications



Uses C# and Razor syntax



Components-based architecture



Runs on web-assembly or server-side



Benefits of Using Blazor to Build PWAs

Leverage existing
.NET skills

Seamless full-stack
development
experience

Interoperability with
JavaScript





Create a Blazor PWA

- Blazor project template
- Visual Studio 2022
- Project structure
- Create the PWA





Modifying the manifest file

- Control PWA appearance
- Set app name, icons, and theme color
- Deploy and install PWA





Adding Blazor functionality

- Components and routing
- Data binding
- Event handling



Summary



Basics of PWAs and Blazor

Why use Blazor to build a PWA

Saw it in action



Up Next:

Service Workers and Offline Support

