

ASP.NET Core Blazor

Article • 09/28/2023

Welcome to Blazor!

Blazor is a framework for building interactive client-side web UI with [.NET](#):

- Create rich interactive UIs using [C#](#).
- Share server-side and client-side app logic written in .NET.
- Render the UI as HTML and CSS for wide browser support, including mobile browsers.
- Build hybrid desktop and mobile apps with .NET and Blazor.

Using .NET for client-side web development offers the following advantages:

- Write code in [C#](#), which can improve productivity in app development and maintenance.
- Leverage the existing .NET ecosystem of [.NET libraries](#).
- Benefit from .NET's performance, reliability, and security.
- Stay productive on Windows, Linux, or macOS with a development environment, such as [Visual Studio](#) or [Visual Studio Code](#). Integrate with modern hosting platforms, such as [Docker](#).
- Build on a common set of languages, frameworks, and tools that are stable, feature-rich, and easy to use.

ⓘ Note

For a Blazor quick start tutorial, see [Build your first Blazor app](#).

Components

Blazor apps are based on *components*. A component in Blazor is an element of UI, such as a page, dialog, or data entry form.

Components are .NET C# classes built into [.NET assemblies](#) that:

- Define flexible UI rendering logic.
- Handle user events.
- Can be nested and reused.