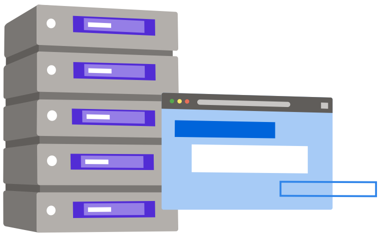
Blazor | Build client web apps with C#

Interactive web UI with C#

Blazor lets you build interactive web UIs using C# instead of JavaScript. Blazor apps are composed of reusable web UI components implemented using C#, HTML, and CSS. Both client and server code is written in C#, allowing you to share code and libraries.

Blazor is a feature of ASP.NET (https://dotnet.microsoft.com/en-us/apps/aspnet), the popular web development framework that extends the .NET developer platform (https://dotnet.microsoft.com/en-us/learn/dotnet/what-is-dotnet) with tools and libraries for building web apps.



Run on WebAssembly or the server

Blazor can run your client-side C# code directly in the browser, using WebAssembly. Because it's real .NET running on WebAssembly, you can re-use code and libraries from server-side parts of your application.

Alternatively, Blazor can run your client logic on the server. Client UI events are sent back to the server using SignalR - a real-time messaging framework. Once execution completes, the required UI changes are sent to the client and merged into the DOM.

Tooltip: The document object model(DOM) is a programming interface that represents all elements on an HTML page as nodes in a tree structure. Using the DOM, elements can be updated, added, and removed from the page.

Build native apps with Blazor Hybrid

Build native client apps using existing Blazor web UI components with Blazor Hybrid. Share the same Blazor components across mobile, desktop, and web while taking advantage of full access to native client capabilities. Use Blazor Hybrid to build cross-platform apps with .NET MAUI, or to modernize existing WPF and Windows Forms apps.

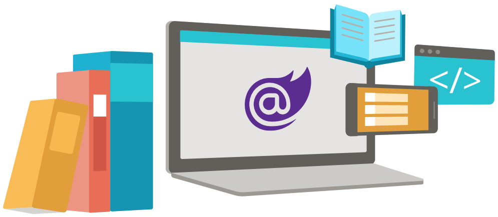
Built on open web standards

Blazor uses open web standards without plug-ins or code transpilation.

Tooltip: Transpilation is the process of converting code written in one programming language to another. Transpilation to JavaScript is a common approach to enabling languages to run in the browser, but often provides limited functionality.

Blazor works in all modern web browsers, including mobile browsers.

Code running in the browser executes in the same security sandbox as JavaScript frameworks. Blazor code executing on the server has the flexibility to do anything you would normally do on the server, such as connecting directly to a database.



Share code and libraries

Blazor apps can use existing .NET libraries, thanks to .NET Standard (https://dotnet.microsoft.com/en-us/platform/dotnet-standard)—a formal specification of .NET APIs that are common across all .NET implementations.

.NET Standard allows the same code and libraries to be used on the server, in the browser, or anywhere you write .NET code.

JavaScript interop

Your C# code can easily call JavaScript APIs and libraries. You can continue to use the large ecosystem of JavaScript libraries that exist for client side UI while writing your logic in C#.

When using server-side code execution, Blazor takes care of seamlessly executing any JavaScript code on the client.

Let's learn Blazor together

Love to watch videos? So do we. We've recorded a full Blazor for Beginners video series to take you through the essentials of building web apps with Blazor.

Free tools for every operating system

Visual Studio and Visual Studio Code provide a great Blazor development experience on Linux, Windows, and macOS.

If you prefer to use a different editor, there are .NET command-line tools that allow you to work with the editor of your choice.

UI component ecosystem

Get productive fast with re-usable UI components from top component vendors like Telerik (https://www.telerik.com/blazor-ui), DevExpress (https://www.devexpress.com/blazor-razor-components/), Syncfusion (https://www.syncfusion.com/aspnet-core-blazor-components), Radzen (https://blazor.radzen.com/), Infragistics (https://www.infragistics.com/products/ignite-ui-blazor), GrapeCity (https://www.grapecity.com/componentone/blazor-ui-controls), jQWidgets (https://www.htmlelements.com/blazor/), and others. Or use one of the many open-source component libraries (https://github.com/AdrienTorris/awesome-blazor?WT.mc_id=dotnet-35129-website#libraries--extensions) from the Blazor community.

Open source and free

Blazor is part of the open-source .NET platform (https://dotnet.microsoft.com/en-us/platform/open-source) that has a strong community of contributors from more than 3,700 companies.

.NET is free (https://dotnet.microsoft.com/en-us/platform/free), and that includes Blazor. There are no fees or licensing costs, including for commercial use.