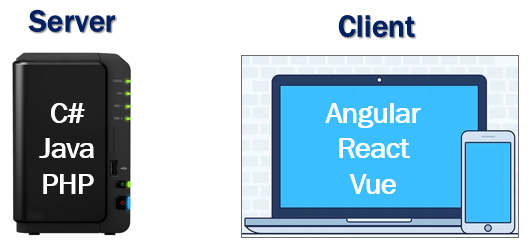
What is Blazor

In this article, we will discuss

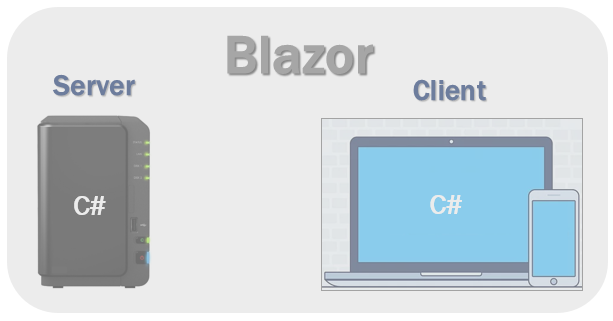
* How we develop web applications today?
* What is Blazor, and why should we use it?
* Setup our local machine for Blazor application development by installing the software and tools required.
* How we develop web applications today?
* what is Blazor?

How we develop web applications today?



* For server-side development, we use programming languages like C#, Java, PHP etc. These are the server-side programming languages.
* For the client-side development we use JavaScript frameworks like Angular, React, Vue etc. There’s no doubt these JavaScript frameworks dominated client-side development up until recently.
* To stay in the business as a developer and remain competitive, it's inevitable we learn both a server-side programming language and a client-side programming language.
* But the question is, why should we learn and use 2 different sets of programming languages and frameworks.

Can we use C# both for server-side and client-side development?

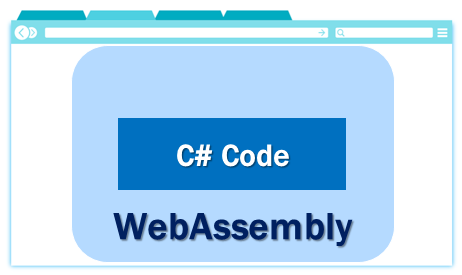


Well, we can and that's exactly why we use Blazor. With Blazor we can now build interactive web UIs using C# instead of JavaScript. C# code can be executed both on the server and in the client browser. This means existing .Net developers can reuse their c# skills rather than learning new JavaScript frameworks and their huge learning curve.

How can a browser execute C# code?

Browsers understand and execute only JavaScript. How can we execute c# code in the client browser?

Well, the answer is WebAssembly.



Blazor can run C# code directly in the browser, using WebAssembly. It runs in the same security sandbox as JavaScript frameworks like Angular, React, Vue etc. Not just C#, in fact, we can run any type of code in the browser using WebAssembly.

WebAssembly is based on open web standards. So it is a native part of all modern browsers including mobile browsers. This means for the blazor application to work, there is no need to install any special plugin like back in the days of silver light and flash.

Blazor hosting models

Blazor offers 2 hosting models. Blazor WebAssembly and Blazor Server.

Blazor WebAssembly

This is also called the client-side hosting model and in this model, the application runs directly in the browser on WebAssembly. So, everything the application needs i.e the compiled application code itself, it's dependencies and the .NET runtime are downloaded to the browser. We use the Blazor WebAssembly App template, to create a Blazor application with the client-side hosting model. We will see this in action in our upcoming videos.

Blazor server

This is also called the server hosting model and in this model, the application is executed on the server from within an ASP.NET Core application. Between the client and the server, a SignalR connection is established. When an event occurs on the client such as a button click, for example, the information about the event is sent to the server over the SignalR connection. The server handles the event and for the generated HTML a diff (difference) is calculated. The entire HTML is not sent back again to the client, it's only the diff that is sent to the client over the established SignalR connection. The browser then updates the UI. Blazor embraces the single page application architecture which rewrites the same page dynamically in response to the user action. Since only the diff is applied to update the UI, the application feels faster and more responsive to the user.

We use the Blazor Server App template, to create a Blazor application with the server hosting model. We will create a blazor application with both the hosting models and discuss the pros and cons in detail in our upcoming videos.

Machine setup for Blazor development

[Install .NET Core SDK 3.1 or later](https://dotnet.microsoft.com/download/)

To verify the list of .NET Core SDKs installed in your machine, run the following command from the command line.

dotnet --list-sdks

Copy

You can build blazor applications using Visual Studio 2019, Visual Studio Code or the .NET Core CLI.

You can download Visual Studio 2019 from the following URL

[Download visual studio 2019](https://visualstudio.microsoft.com/downloads/)

If you are using Visual Studio, make sure asp.net and web development workload is installed. To verify if you have this workload installed, click on Tools and then select Get Tools and Features.

Blazor Prerequisites

* C#
* HTML
* CSS

Blazor is gaining great traction and it is here to stay. If you are a .Net developer, it’s invaluable to have Blazor in your tool belt.