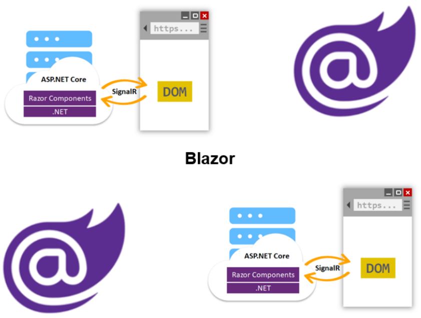
Blazor

Duration: 2 days **Price: € 1299**

Open Schedule



Audience Course Blazor

The course Blazor is intended for developers who want to create interactive client-side Web User Interfaces using C#, .NET and the Blazor Framework.

Prerequisites Course Blazor

Knowledge of .NET and C# and the basic architecture of Web and Single Page applications is required to participate in this course.

Realization Training Blazor

The concepts are covered on the basis of presentations. The concepts are further explained with demos in Visual Studio. Theory and exercises are interchanged. The course times are from 9.30 to 16.30.

Certificate Blazor Course

After successful completion of the course participants receive a Blazor certificate.

Content Course Blazor

In the course Blazor participants learn how the Blazor Framework can be used to create Single Page Applications with C# and .NET instead of JavaScript. With Blazor client-side C# code can be run directly in the browser through WebAssembly. And since real .NET code runs on WebAssembly, you can reuse server side code and libraries.

Blazor Intro

The course Blazor starts with a discussion of Blazor's role within ASP.NET which is an extension of the .NET Framework with tools and libraries for Web Applications. The Razor Markup is introduced as well and the SignalR library for real time functionality in web applications is discussed.

Data Binding

Next one and two way data binding between user interface elements and C# code and lambda functions are treated. And event binding, event propagation and validation are also covered.

Blazor Architecture

The Blazor Architecture is explained. Blazor is based on UI components and changes in the browser DOM are tracked via an in memory tree and the DOM is updated where needed.

Dependency Injection

Attention is paid to how centrally registered services in Blazor such as the HttpClient and the NavigationManager are made available to Blazor Components via dependency injection. Adding services to a Blazor Webassembly App and a Blazor Service App is covered as well.

Routing

Routing in Blazor Apps with the Router component is on the course schedule as well. Explained is how route information is found by scanning components with a RouteAttribute. And it is treated how RouteView components receive RouteData and parameters.

JavaScript Interoperability

Finally JavaScript Interoperability is covered. The course discusses how to invoke JavaScript from .NET code using an injected service that implements the IJSRuntime interface.

Modules Course Blazor

Module 1 : Blazor Intro	Module 2 : Data Binding	Module 3 : Blazor Architecture
What is Blazor? C# Browser Apps .NET Core Blazor Components Razor Markup .NET Libraries Using Visual Studio Blazor Projects Server SignalR Connection Client Side Blazor	One Way Data Binding Attribute Binding Conditional Attributes Event Binding Syntax Event Arguments C# Lambda Functions Two Way Data Binding Preventing Default Actions Stop Event Propagation Reporting Changes Validation	Blazor Components UI Elements C# and HTML UI Composition Razor Templates View and View Model Parent-Child Communication Type Parameters Life Cycle Hooks Cascading Properties Component Libraries
Module 4 : Dependency Injection	**Module 5 : Routing**	**Module 6 : JavaScript Interoperability**
Inversion of Control Container Injection Singleton Dependencies Transient Dependencies Scoped Dependencies Disposing Dependencies Adding Services Register Common Services Service Lifetime Constructor Injection @inject Statements	Blazor Routing Navigation Router Component Route Templates @page Directive RouteAttribute RouteView Component Navigation FocusOnNavigate Route Parameters Route Constraints	Call JavaScript from C# Glue Function JSInvocable Methods Pass Reference to JavaScript ILocalStorage Service DOM Interaction Asynchronous Calls Object Serialization JavaScript Initializers Location of JavaScript JavaScript Modules