

Blazor

Doelgroep Cursus Blazor

De cursus Blazor is bedoeld voor developers die interactieve client-side Web User Interfaces willen maken met C#, .NET en het Blazor Framework.

Voorkennis Cursus Blazor

Kennis van .NET en C# en de basis architectuur van Web en Single Page applicaties is gewenst om aan deze cursus te kunnen deelnemen.

Uitvoering Training Blazor

De concepten worden behandeld aan de hand van presentaties. De concepten worden toegelicht met demo's in Visual Studio. Theorie en exercises wisselen elkaar af. De cursustijden zijn van 9.30 tot 16.30.

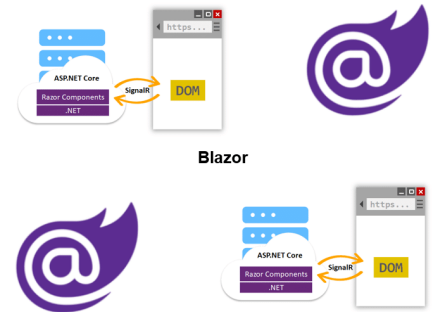
Certificaat Blazor Cursus

De deelnemers krijgen na het goed doorlopen van de cursus een certificaat Blazor.

Duur: 2 dagen

Prijs: € 1399

[Open Rooster](#)



Inhoud Cursus Blazor

In de cursus Blazor leren de deelnemers hoe het Blazor Framework gebruikt kan worden voor de creatie van Single Page Applicaties met C# en .NET in plaats van met JavaScript. Met Blazor kan client-side C# code door middel van WebAssembly direct in de browser worden uitgevoerd. En aangezien echte .NET code op WebAssembly draait kun je server side code en libraries hergebruiken.

Blazor Intro

De cursus Blazor gaat van start met een bespreking van de rol van Blazor binnen ASP.NET dat een uitbreiding is van het .NET Framework met tools en libraries voor Web Applicaties. Ook wordt de Razor Markup geïntroduceerd en wordt ingegaan op de SignalR library voor real time functionaliteit in web applicaties.

Data Binding

Vervolgens komen One and Two way data binding tussen user interface elementen en C# code en lambda functies aan bod. Ook event binding, event propagation en validatie worden dan besproken.

Blazor Architecture

Ook wordt ingegaan op de Blazor Architecture die gebaseerd is op UI componenten en waarbij veranderingen in de browser DOM via een in memory tree worden bijgehouden en de DOM wordt bijgewerkt.

Dependency Injection

Ook wordt besproken hoe in Blazor centraal geregistreerde services zoals de HttpClient en de NavigationManager via dependency injection beschikbaar worden gesteld aan Blazor Components. Het toevoegen van services aan een Blazor Webassembly App en een Blazor Service App komt aan de orde.

Routing

En ook routing in Blazor Apps met de Router component passeert de revue. Aandacht wordt besteed aan hoe route informatie wordt gevonden door het scannen van componenten met een RouteAttribute. En ook wordt besproken hoe RouteView componenten RouteData en parameters binnen krijgen.

JavaScript Interoperability

Tenslotte komt JavaScript Interoperability aan bod. Uitgelegd wordt hoe JavaScript vanuit .NET code kan worden aangeroepen met behulp van een geïnjecteerde service die het IJSRuntime interface implementeert.

Modules Cursus 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