

Introduktion til Blazor

Prosa
December 2022

Hvad er Blazor

- Web brugerflader i C#
 - Ikke JavaScript (næsten)
- Applikationslogik i C#
- Brug af standarder
 - HTML og CSS
 - WASM
- Browser, mobile og desktop
- Alle nye browsere er supporteret
- Simplificeret SPA udvikling
 - "Alternativ" til Angular, React, Vue mv
- Stort community
 - open source projekter
- Brug/genbrug eksisterende .NET komponenter
- Del logik mellem klient og server
- Effektivt udviklingsmiljø(er)

Forudsætninger

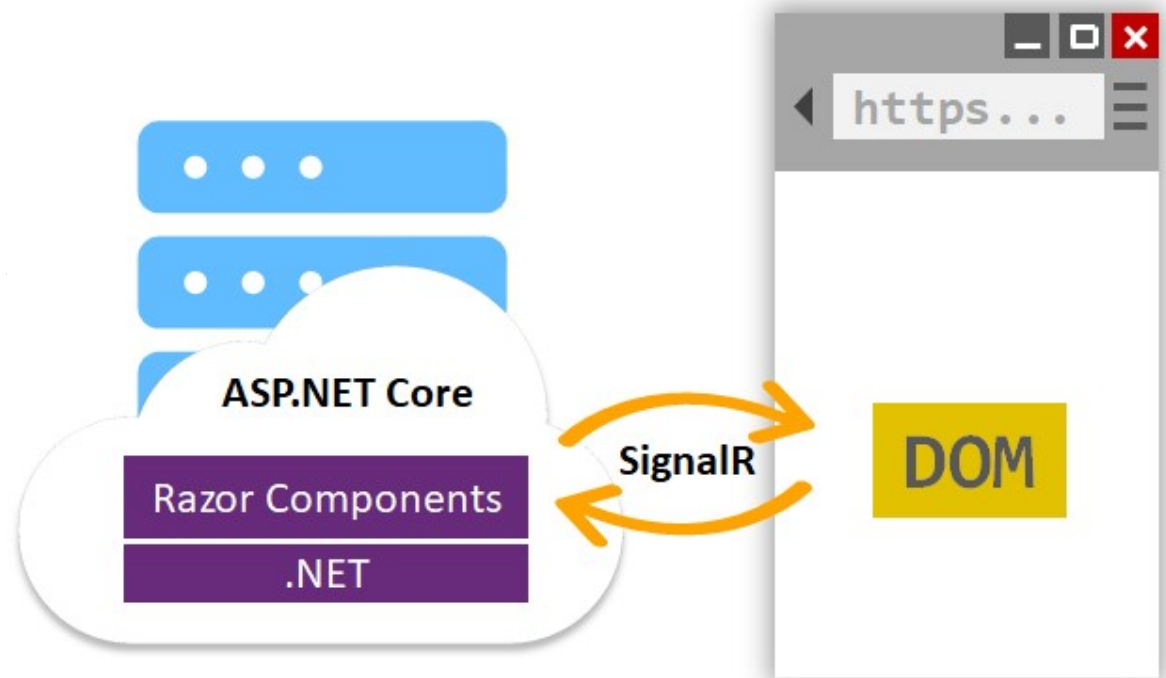
- God erfaring med C# (9+)
- God erfaring med web udvikling
 - HTTP
 - HTML
 - CSS
 - (JavaScript)
- Viden om cloud

Udviklingsværktøjer

- .NET 7 SDK
 - dotnet new list
 - dotnet new blazorserver[-empty] -n MyApp
 - dotnet new blazorwasm[-empty] -n MyApp
 - Visual Studio Code
 - C# Extension
 - BlazorWasmDebuggingExtension
 - Blazor Snippet Pack
 - Prettier
- Visual Studio 2022
 - Opdater til .NET 7
- Rider

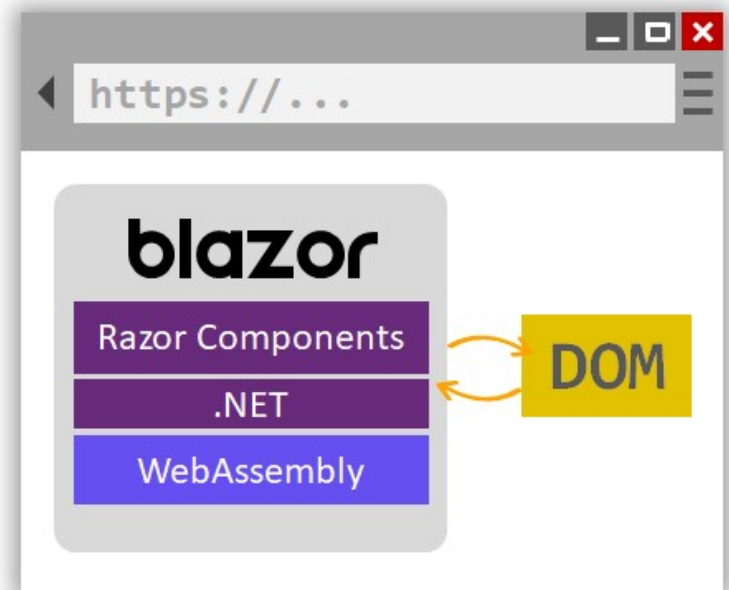
Blazor Server

- Genererer komponenter på klienten
 - JS
- Alle hændelser kommunikerer serveren over SignalR / Websockets
- Ændringer i brugerflade kommunikerer retur



Blazor Web Assembly

- SPA framework
- Åbne standarder
- WASM er byte kode i browseren
- Runtime (Mono) downloades
- IL kode afvikles
- PWA klient



Blazor Hybrid

- Mulighed for at benytte Blazor komponenter i
 - MAUI
 - Desktop
 - Mobile
 - WPF
 - Winform
 - MVC
- Benytter ikke WASM men lokalt interop

```
<?xml version="1.0" encoding="utf-8" ?>
<ContentPage xmlns="http://schemas.microsoft.com/dotnet/2021/maui"
  xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
  xmlns:local="clr-namespace:MauiApp1"
  x:Class="MauiApp1.MainPage"
  BackgroundColor="{DynamicResource PageBackgroundColor}">

  <BlazorWebView HostPage="wwwroot/index.html">
    <BlazorWebView.RootComponents>
      <RootComponent Selector="#app" ComponentType="{x:Type local:Main}" />
    </BlazorWebView.RootComponents>
  </BlazorWebView>

</ContentPage>
```

Alt er baseret på komponenter

- Blazor er baseret på komponenter (.razor)
 - Side
 - Dialog boks
 - Data form
 - Med videre
- En komponent har
 - Fleksibel og effektiv UI rendering
 - Metoder til effektiv håndtering af hændelser
 - Mulighed for blive både embedded og genbrugt
 - Kan deles mellem mange forskellige typer af projekter

Komponenter

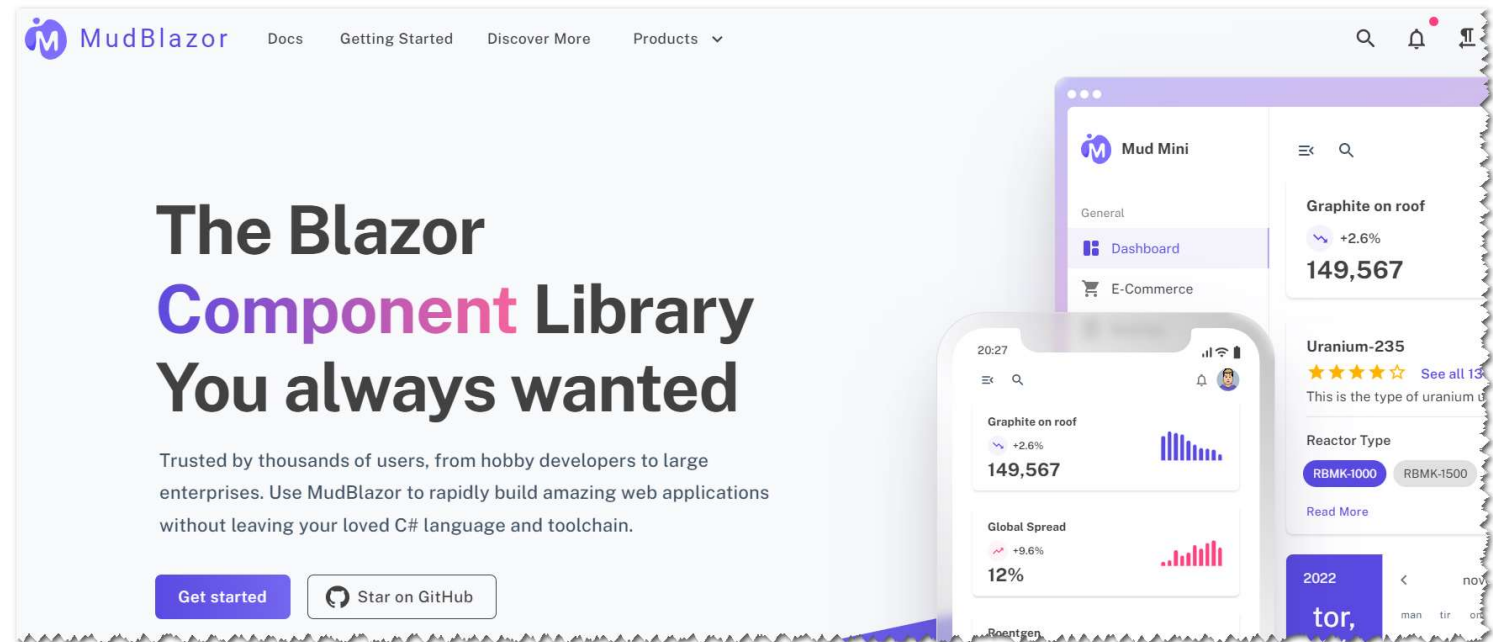
- Kan være en rute eller blot en komponent
 - @page "/hello-world"
 - @page "/test/{text?}"
- Data kan findes gennem en attribut
 - [Parameter]
- Configuration
 - @inject IConfiguration Configuration
 - @Configuration["h1FontSize"]
- State
 - DI
 - Browser storage

Databinding

- Razor syntaks
 - @ + C#
- Simpel databinding med @...
- Brug af @bind syntaks
- Lifecycle
 - OnInitialized
 - OnParametersSet
 - OnAfterRender

Blazor framework

- MudBlazor
- Blazorise
- Se Awesome Blazor



JavaScript InterOp

- Hvis der er behov kan man fra Blazor
 - Kalde en JavaScript funktion
 - Blive kaldt fra en JavaScript funktion

```
@page "/counter"

@inject IJSRuntime jsRuntime;

<PageTitle>Counter</PageTitle>

<h1>Counter</h1>
```