Blazor Conf 2023

Blazor United.
Un salto nel futuro

Andrea Dottor
@dottor

Un grazie agli sponsor

E alle community che ci hanno supportato

DOTNETSIDE

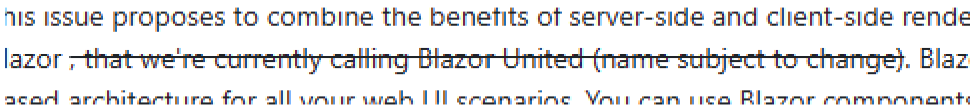
DotNet
Toscana

dotnet
LOMBARDIA

{codemotion}

~~Blazor United~~

- Blazor SSR (Server Side Rendering)
- Full Stack Blazor
- Full stack web UI with Blazor



his issue proposes to combine the benefits of server-side and client-side rende
lazor ~~, that we're currently calling Blazor United (name subject to change)~~. Blaz
ased architecture for all your web UI scenarios. You can use Blazor component

danroth27 changed the title ~~Blazor United~~ Full stack web UI with Blazor 2 weeks ago

.NET Community Standup

Blazor United in .NET 8

Steve Sanderson, Dan Roth, Mackinnon Buck
Javier Calvarro Nelson, Jon Galloway

Full Stack Blazor

The Blazor team shares early thoughts on using Blazor in .NET 8 to create a single unified model for all your web UI scenarios.

Watch now →

Full stack web UI with Blazor #46636

⊙ Open danroth27 opened this issue on Feb 14 · 105 comments

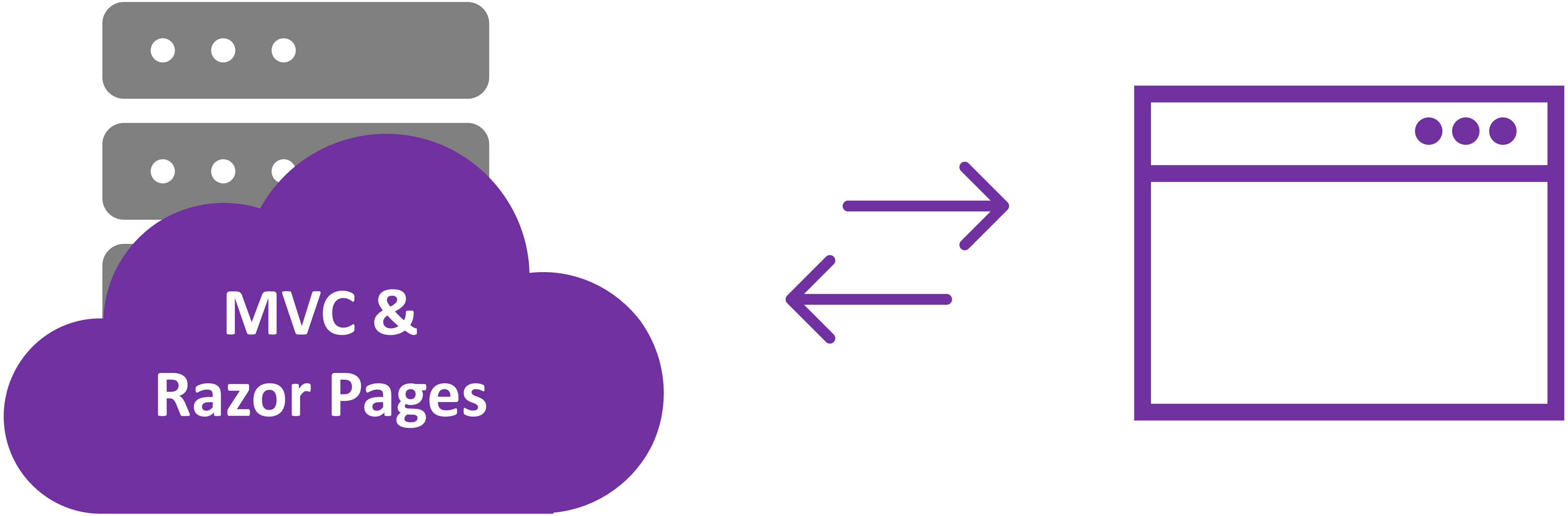
Blazor Conf 2023

~~Blazor United.~~

Full stack web UI with Blazor. Un salto nel futuro

Andrea Dottor
@dottor

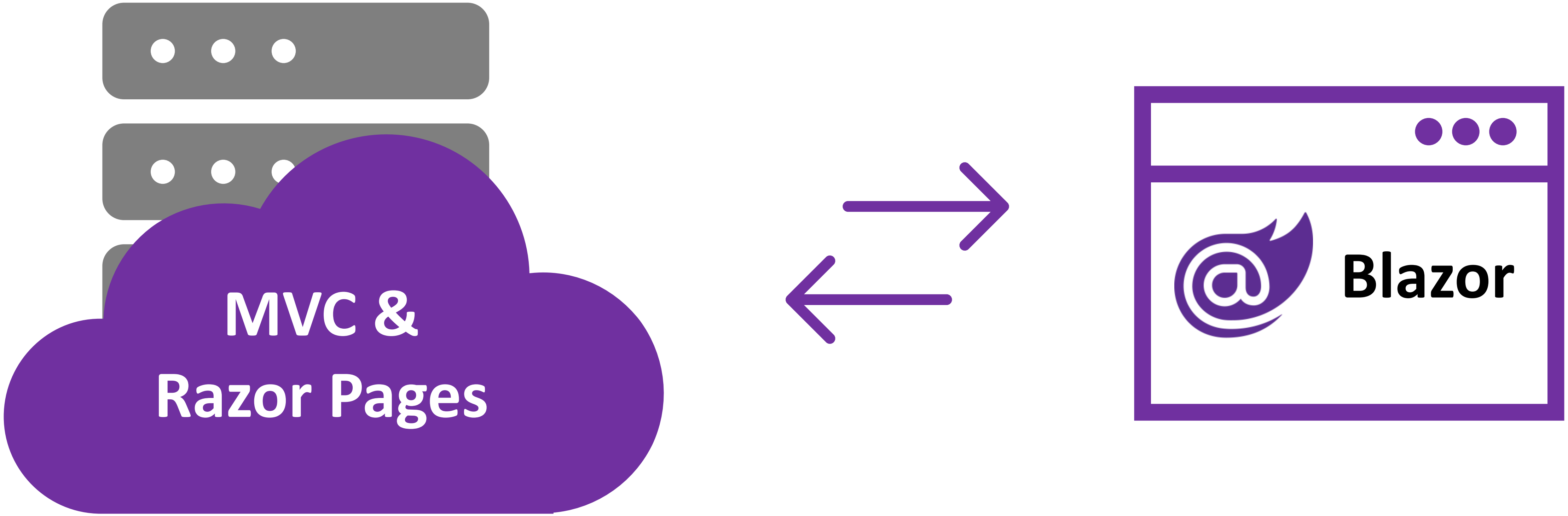
Server vs client



MVC &
Razor Pages

Server-side rendering (SSR) 🚀

Server vs client



MVC & Razor Pages

Blazor

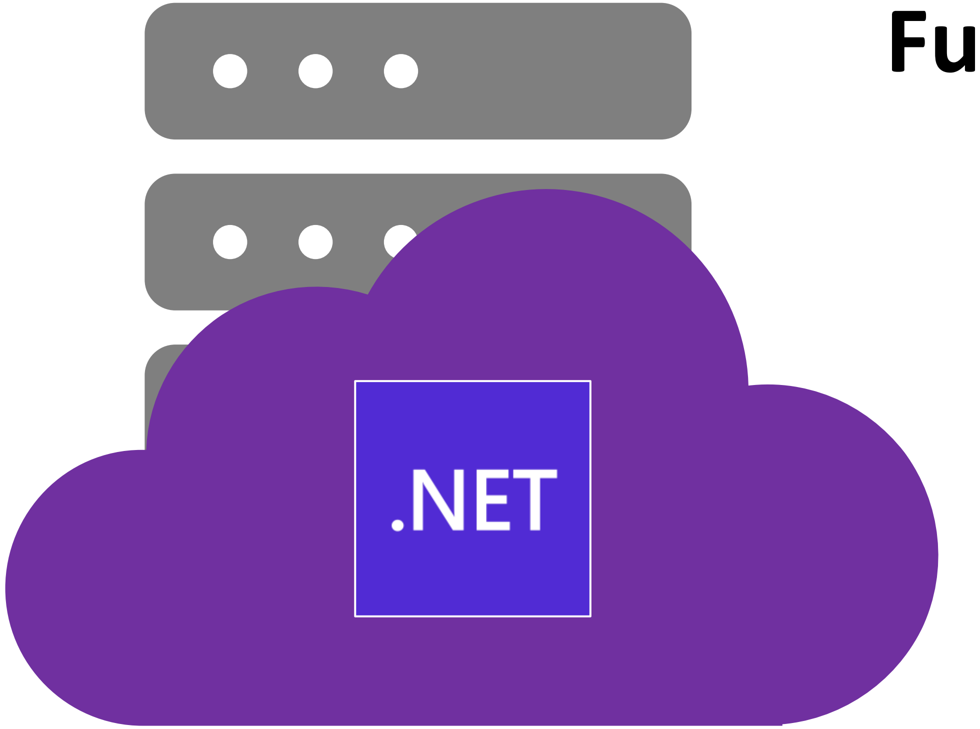
Server-side rendering (SSR) 🚀

Client-side rendering (CSR) 🎮

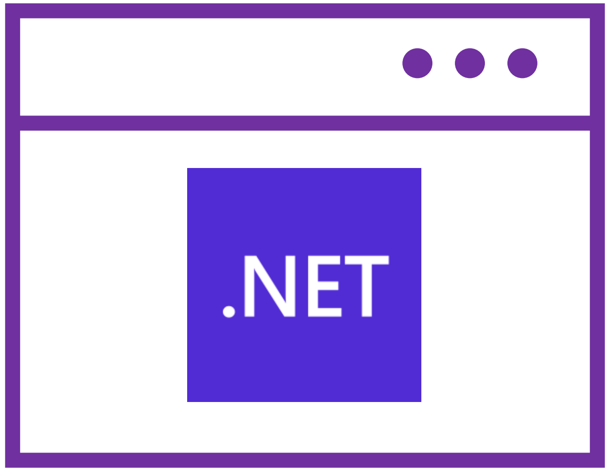
Server vs client

~~Blazor United~~

Full stack web UI with Blazor



Server-side rendering (SSR) 🚀



Client-side rendering (CSR) 🎮

isn't a new framework

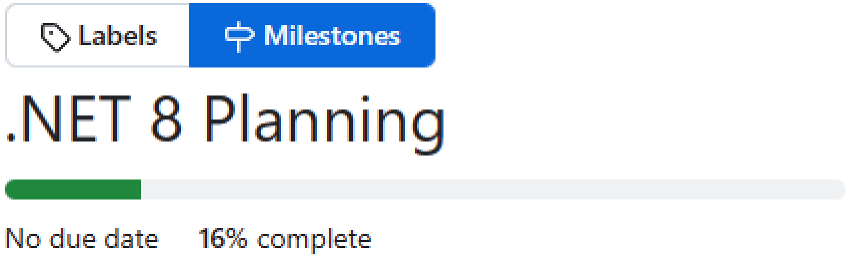
Daniel Roth:

Up till now we've been referring to this effort as "Blazor United". ***But "Blazor United" isn't a new framework or even a new Blazor hosting model. Instead, it's a collection of new Blazor features and enhancements that you can add to any Blazor web app.***

To avoid confusing people into thinking that we're building a completely new framework, we've decided to stop using the name "Blazor United" *in our messaging and instead focus on the new Blazor features, specifically: server-side rendering with Razor components, streaming rendering, enhanced navigation and form handling, enabling client interactivity per component or page (from the server or via WebAssembly), and the ability to change the component render mode at runtime.*

https://github.com/dotnet/aspnetcore/issues/46636#issuecomment-1520840129

.NET 8 Planning



.NET 8 Planning

No due date 16% complete

- ✓ Streaming rendering #46352
- ✓ [Design] Blazor United: Layering architecture #46393
- ⊙ Render a server component into the page #46353
- ⊙ Render a webassembly component into the page #46354
- ⊙ We can interleave Static/Server/WebAssembly components within an app #46394
- ⊙ We have a project template where we can add support for Server and Webassembly #46395
- ✓ API for registering endpoints that render Razor components. #46350
- ⊙ Auto switching for interactive components #46397
- ⊙ Unified project system #46398
- ⊙ Progressive enhancement of form submission & navigation #46399

https://github.com/dotnet/aspnetcore/issues/46636

Current version

.NET 8 preview 4

16 Maggio

https://devblogs.microsoft.com/dotnet/asp-net-core-updates-in-dotnet-8-preview-4/

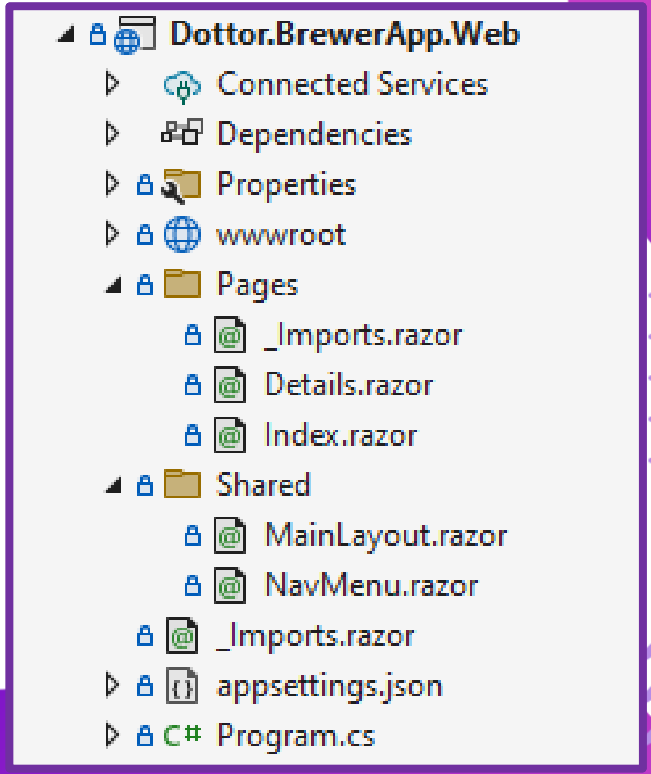
Server Side Rendering

Rendering server-side utilizzando i Razor Component (file *.razor)

I path vengono recuperati dalla direttiva @page dei componenti

Nessun JavaScript necessario

* gli eventi "Blazor" non vengono gestiti es: @onclick



- Dottor.BrewerApp.Web
  - Connected Services
  - Dependencies
  - Properties
  - wwwroot
  - Pages
    - _Imports.razor
    - Details.razor
    - Index.razor
  - Shared
    - MainLayout.razor
    - NavMenu.razor
  - _Imports.razor
  - appsettings.json
  - Program.cs

Server Side Rendering

Da controller e da Minimal API sarà possibile indicare di fare il rendering html di un specifico Razor Component

```
public class HomeController : Controller
{
    public IResult MyComponent()
    {
        return new RazorComponentResult<MyComponent>();
    }
}
```

```
app.MapGet("/my-component", () => new RazorComponentResult<MyComponent>());
```

Blazor Conf 2023

Demo

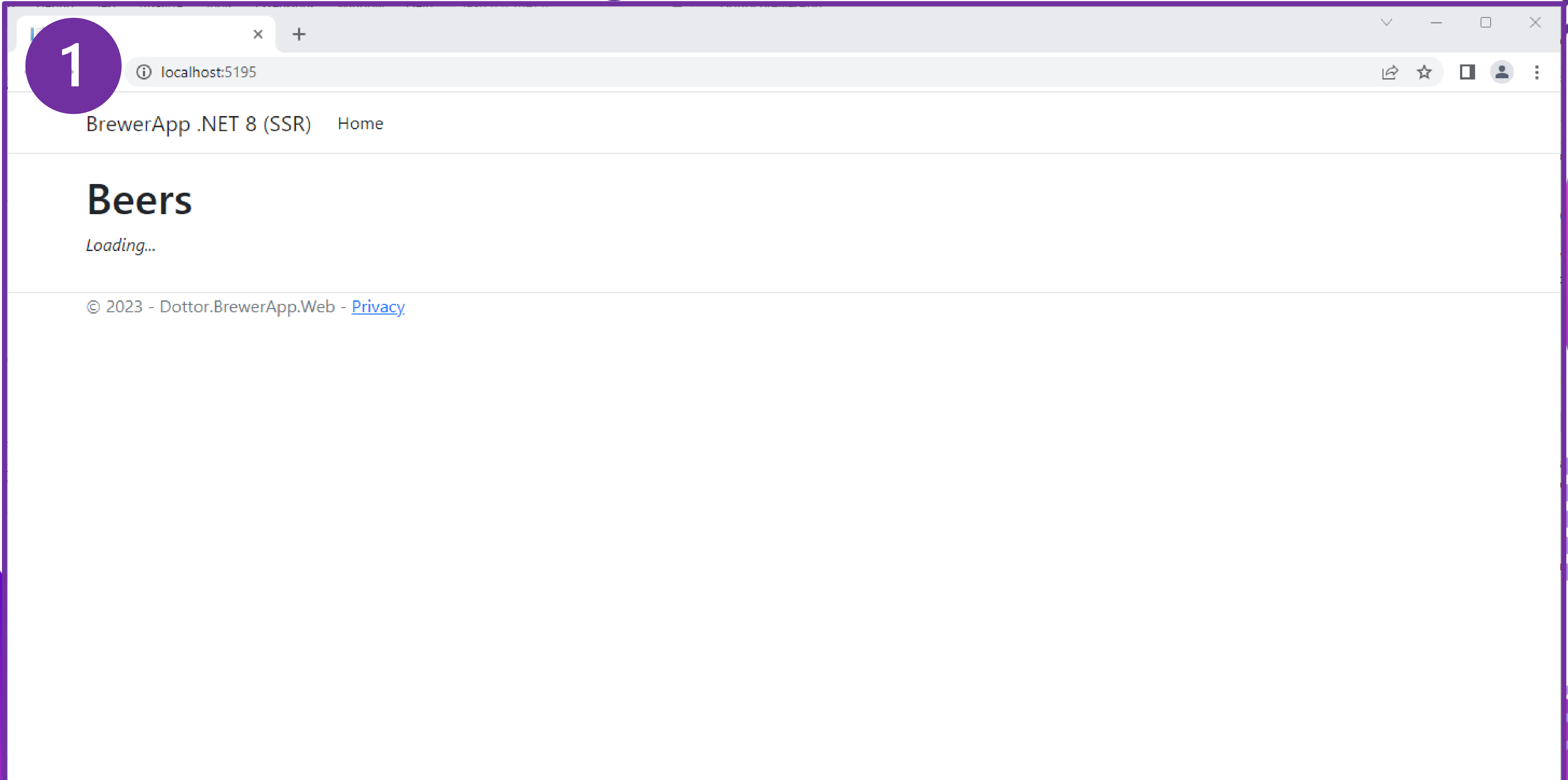
Server Side Rendering

StreamRendering

```
@code {
    private IEnumerable<Beer>? _beers;

    protected override async Task OnInitializedAsync()
    {
        _beers = await brewerService.GetBeersAsync();
    }
}
```

StreamRendering

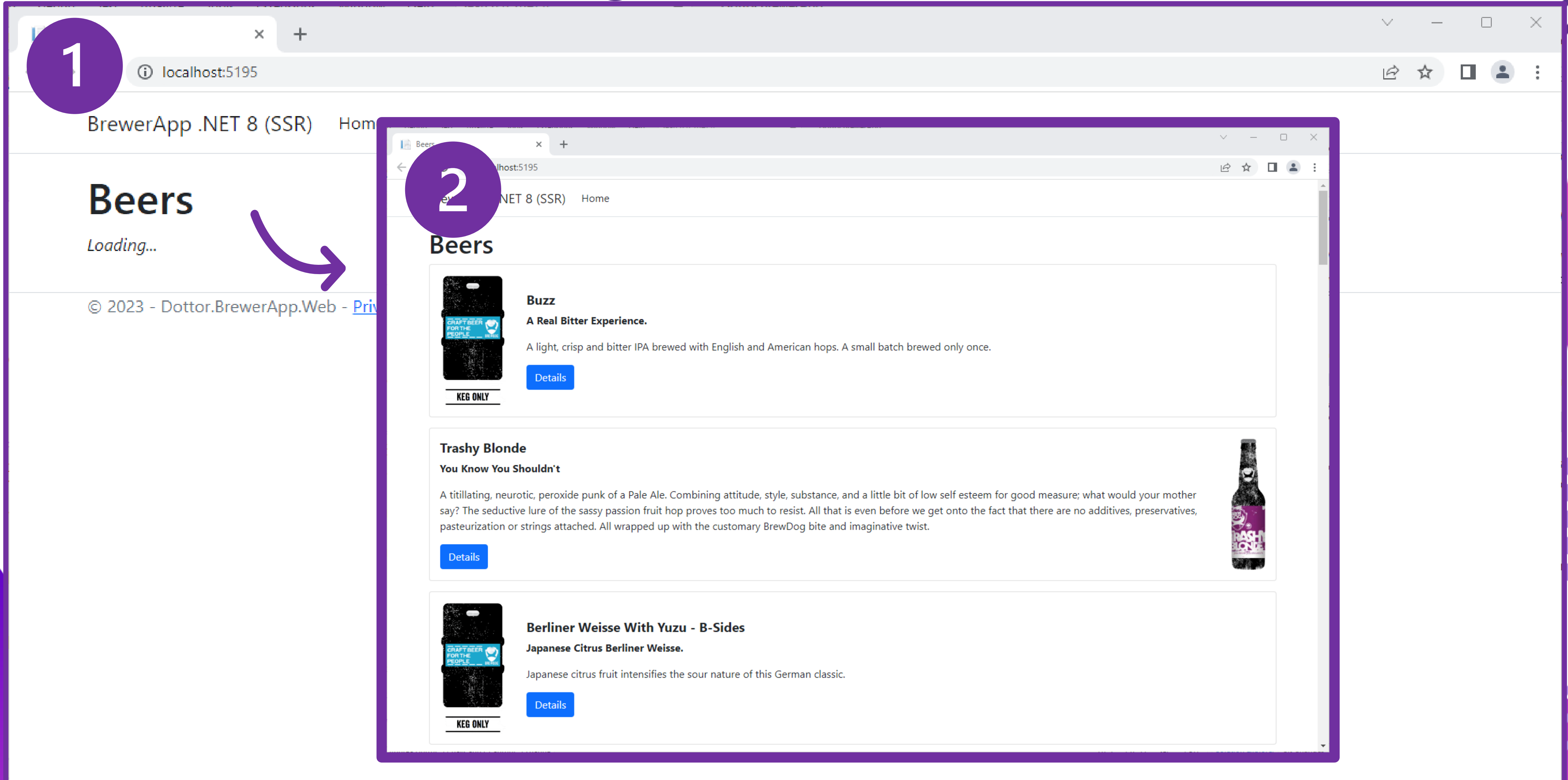


Beers

Loading...

© 2023 - Dottor.BrewerApp.Web - Privacy

StreamRendering



localhost:5195

BrewerApp .NET 8 (SSR) Hom

Beers

Loading...

© 2023 - Dottor.BrewerApp.Web - Priv

Beers

lhost:5195

NET 8 (SSR) Home

Beers

Buzz

A Real Bitter Experience.

A light, crisp and bitter IPA brewed with English and American hops. A small batch brewed only once.

Details

Trashy Blonde

You Know You Shouldn't

A titillating, neurotic, peroxide punk of a Pale Ale. Combining attitude, style, substance, and a little bit of low self esteem for good measure; what would your mother say? The seductive lure of the sassy passion fruit hop proves too much to resist. All that is even before we get onto the fact that there are no additives, preservatives, pasteurization or strings attached. All wrapped up with the customary BrewDog bite and imaginative twist.

Details

Berliner Weisse With Yuzu - B-Sides

Japanese Citrus Berliner Weisse.

Japanese citrus fruit intensifies the sour nature of this German classic.

Details

StreamRendering - html

```
    <!--bl:6-->
    <title>Beers</title><!--/bl:6-->
</head>
<body>
    <div class="container">
        <main role="main" class="pb-3"><!--bl:5-->
            <h1>Beers</h1>
            <p><em>Loading...</em></p><!--/bl:5-->
        </main>
    </div>
[…]
```

StreamRendering - template

```
</html><blazor-ssr><template blazor-component-id="5">
        <h1>Beers</h1>
        <div class="beer-card card mb-3 even">
          ...
        </div>
        ...
    </template><template blazor-component-id="6">
        <title>Beers</title>
    </template></blazor-ssr>
```

StreamRendering

Permette di restituire immediatamente l'html della pagina con dei placeholder, che verranno aggiornati una volta che le operazioni asincrone saranno terminate.

Sfrutta un'unica connessione/richiesta http

In coda alla pagina sarà presente il contenuto dei placeholder

Del codice JavaScript si occuperà di aggiornare opportunamente la pagina

Tempo di risposta molto veloce

Blazor Conf 2023

Demo

StreamRendering

Form - .NET 8 preview 4

Model-Binding tramite il componente `CascadingModelBinder`

```
<CascadingModelBinder>
    @Body
</CascadingModelBinder>
```

Recupero dei valori tramite `FormDataProvider`

```
@inject FormDataProvider FormData

protected override void OnInitialized()
{
    // Manually model bind the form data using the FormDataProvider service
    if (FormData.Entries.TryGetValue("Text", out var text))
    {
        Review.Text = text.First();
    }
}
```

Form - .NET 8 (in un futuro prossimo)

```
//[SupplyParameterFromForm]
//public Review Review { get; set; }

protected override async Task OnInitializedAsync()
{
    if (EditContext.IsModified() && EditContext.Validate())
    {
      ...
    }
}
```

EditForm enhance - .NET 8 (in un futuro prossimo)

Permette di eseguire il post della form sfruttando JavaScript, ed eseguendo una chiamata fetch in POST, così da non dover ricaricare tutta la pagina

```
<EditForm method="post" EditContext="EditContext" enhance>
    <StarRating @bind-Value="EditContext.Model.Rating" />
    <InputTextArea rows="5" name="text" @bind-Value="EditContext.Model.Text" />

    <button type="submit" class="btn btn-primary">Submit</button>
</EditForm>
```

* Lo stesso comportamento potrebbe essere applicata alla navigazione

https://github.com/dotnet/aspnetcore/issues/46399

Blazor Conf 2023

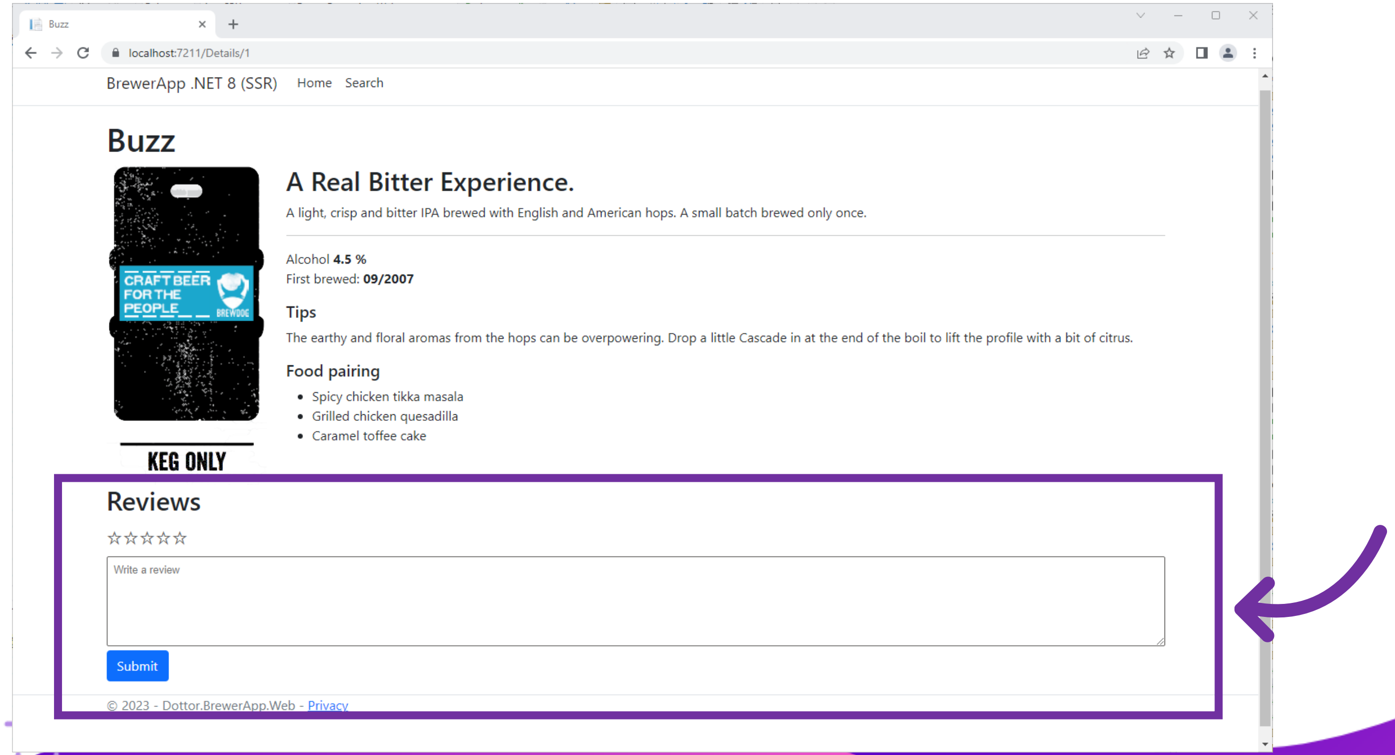
Demo

Form

Component > rendermode

Abilita a livello di singolo componente il rendering "Blazor Server" o "Blazor WebAssembly"

Possibilità di attivare anche il pre-rendering



Buzz

A Real Bitter Experience.

A light, crisp and bitter IPA brewed with English and American hops. A small batch brewed only once.

Alcohol **4.5 %**
First brewed: **09/2007**

Tips

The earthy and floral aromas from the hops can be overpowering. Drop a little Cascade in at the end of the boil to lift the profile with a bit of citrus.

Food pairing

- Spicy chicken tikka masala
- Grilled chicken quesadilla
- Caramel toffee cake

Reviews

☆☆☆☆☆

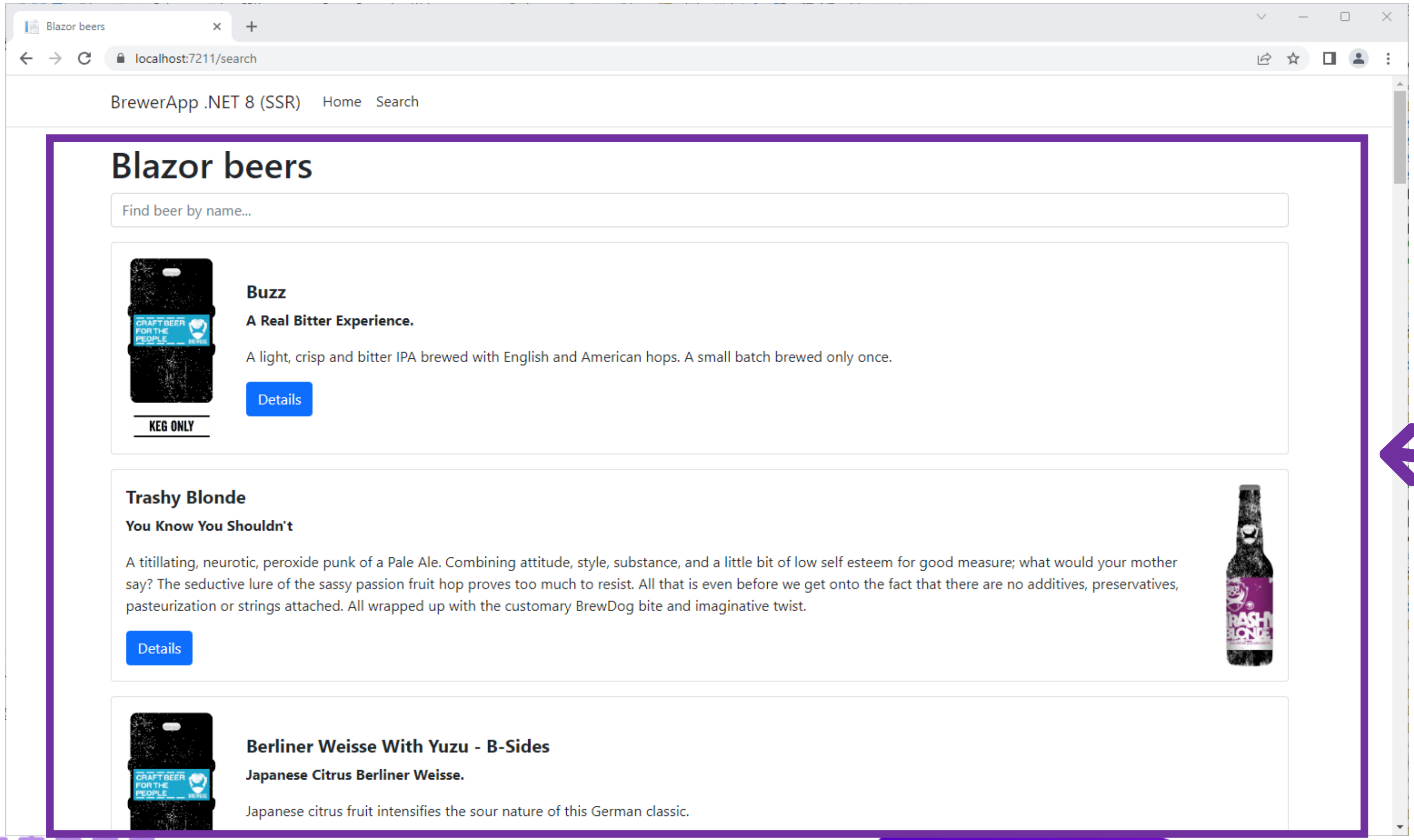
Write a review

Submit

© 2023 - Dottor.BrewerApp.Web - Privacy

Page > RenderMode

Potremo specificare il RenderMode a livello di pagina.



Blazor beers

Find beer by name...

Buzz

A Real Bitter Experience.

A light, crisp and bitter IPA brewed with English and American hops. A small batch brewed only once.

Details

Trashy Blonde

You Know You Shouldn't

A titillating, neurotic, peroxide punk of a Pale Ale. Combining attitude, style, substance, and a little bit of low self esteem for good measure; what would your mother say? The seductive lure of the sassy passion fruit hop proves too much to resist. All that is even before we get onto the fact that there are no additives, preservatives, pasteurization or strings attached. All wrapped up with the customary BrewDog bite and imaginative twist.

Details

Berliner Weisse With Yuzu - B-Sides

Japanese Citrus Berliner Weisse.

Japanese citrus fruit intensifies the sour nature of this German classic.

Demo

RenderMode

RenderMode WebComponentRenderMode.**Auto**

```
@attribute [ComponentRenderMode(WebComponentRenderMode.Auto)]
```

Blazor Conf 2023

Demo

RenderMode Auto

works on my machine

Links

- Blazor United prototype (24 gennaio) https://youtu.be/48G_CEGXZZM
- ASP.NET Community Standup - Blazor United in .NET 8 (14 febbraio) https://www.youtube.com/live/kIH_Py8ytlY
- Milestone .NET 8 Planning https://github.com/dotnet/aspnetcore/milestone/221
- Full stack web UI with Blazor #46636 https://github.com/dotnet/aspnetcore/issues/46636
- SLIDE e DEMO (BrewerApp) https://github.com/andreadottor/BrewerApp

Contatti

Andrea Dottor

Microsoft MVP Developer Technologies

www.dottor.net

andrea@dottor.net

@dottor

Microsoft® Most Valuable Professional

.NET IN PILLOLE il podcast