



ASP.NET Core 6.0

Aus der Praxis

Folien & Code auf
www.dotnetconsulting.eu/blog



Thorsten Kansy (tkansy@dotnetconsulting.eu)

Meine Person- Thorsten Kansy

Freier Consultant, Software Architekt,
Entwickler, Trainer & Fachautor



Azure Cosmos DB

Mein Service- Ihr Benefit

- Individuelle Inhouse Trainings
- (Online on-demand) Projektbegleitung
- Beratung
 - Problemanalyse und Lösungen
 - Technologieentscheidungen





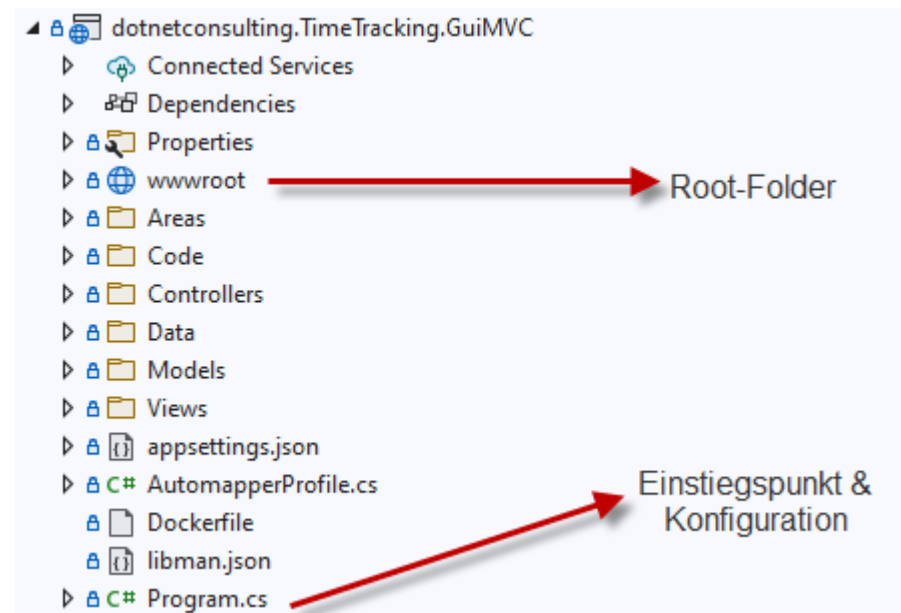
Program.cs & Minimal API



Program.cs

```
var builder = WebApplication.CreateBuilder(args);  
  
// Add services to the container.  
builder.Services.AddControllersWithViews();  
  
var app = builder.Build();  
  
// Configure the HTTP request pipeline.  
if (!app.Environment.IsDevelopment())  
{  
    app.UseExceptionHandler("/Home/Error");  
    // The default HSTS value is 30 days. You may want to change this for production scenarios, see https://aka.ms/aspnetcore-hsts.  
    app.UseHsts();  
}  
  
app.UseHttpsRedirection();  
app.UseStaticFiles();  
app.UseRouting();  
app.UseAuthorization();  
app.MapControllerRoute(  
    name: "default",  
    pattern: "{controller=Home}/{action=Index}/{id?}");  
app.Run();
```


Projektaufbau (Basis)



Implicit Namespace Imports

Global Namespace Importe

```
<PropertyGroup>
  <OutputType>Exe</OutputType>
  <TargetFramework>netcoreapp6.0</TargetFramework>
  <!-- Standard für .NET 6.0 -->
  <LangVersion>preview</LangVersion>
  <ImplicitUsings>true</ImplicitUsings>
</PropertyGroup>

<ItemGroup>
  <Using Include="System.Web" />
  <Using Remove="System.Threading.Tasks" />
</ItemGroup>
```

Alternative im Code

```
10
11 global using System.Text.Json;
```

Console/Library

System
System.Collections.Generic
System.IO
System.Linq
System.Net.Http
System.Threading
System.Threading.Tasks

Web

In addition to the console/library namespaces :

System.Net.Http.Json
Microsoft.AspNetCore.Builder
Microsoft.AspNetCore.Hosting
Microsoft.AspNetCore.Http
Microsoft.AspNetCore.Routing
Microsoft.Extensions.Configuration
Microsoft.Extensions.DependencyInjection
Microsoft.Extensions.Hosting
Microsoft.Extensions.Logging

Worker

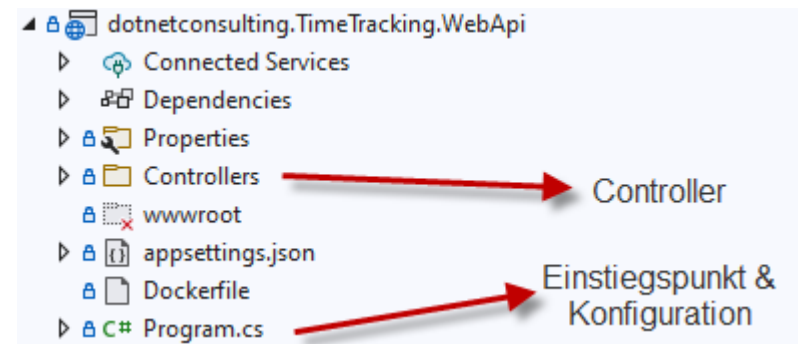
In addition to the console/library namespaces :

Microsoft.Extensions.Configuration
Microsoft.Extensions.DependencyInjection
Microsoft.Extensions.Hosting
Microsoft.Extensions.Logging

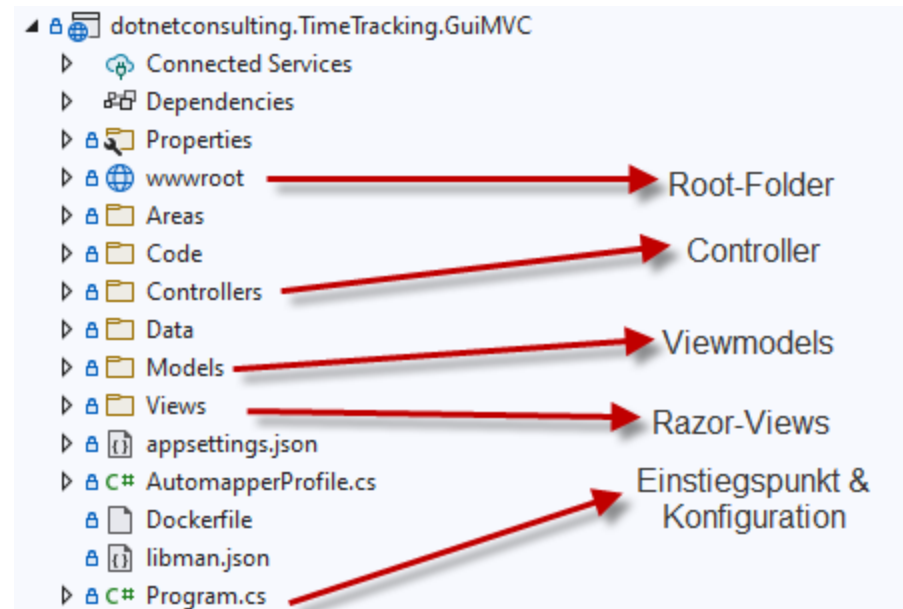
Anwendungs-Startup

- ASP.NET ist ein startbares Program
 - EXE
 - DLL für z.B. IIS
- Kestrel
 - Standard seit VS2022 & .NET 6
- Program.cs
 - Dependency Injection-Konfiguration
 - ASP.NET Core HTTP Request Pipeline-Konfiguration

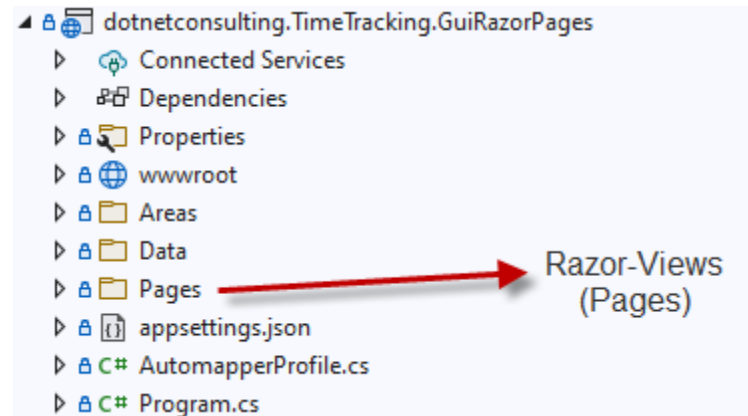
Projektaufbau (Web API)



Projektaufbau (MVC)



Projektaufbau (Razor Pages)



Demo

Minimal API

APIs ohne Controller

```
app.MapGet(Urls.GETWEATHER,  
    (IWeatherService weatherService) => weatherService.GetWeather());
```

```
app.MapGet(Urls.GETWEATHERLOCATION,  
    (IWeatherService weatherService, string location) =>  
        weatherService.GetWeather(location));
```

```
// app.MapPost(...);  
// app.MapPut(...);  
// app.MapDelete(...);
```

HTTP-Protokoll

Version 1.1

```
GET http://localhost:50154/api/events/45 HTTP/1.1
```

```
POST http://localhost:50393/api/speakerbasic HTTP/1.1
```

```
Content-Type: application/json; charset=utf-8
```

```
{  
  "id": 5,  
  "homepage": "http://www.dotnetconsulting.de",  
  "infos": "Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam!",  
  "name": "Speaker 5",  
  "created": "2017-12-06T16:26:22.1733333",  
  "isDeleted": false,  
  "updated": null,  
  "speakerSessions": []  
}
```

```
DELETE http://localhost:50393/api/speakerbasic/5 HTTP/1.1
```



WebAssembly

PageTitle, HeadContent & HeadOutlet

```
var builder = WebAssemblyHostBuilder.CreateDefault(args);  
  
builder.RootComponents.Add<App>("app");  
builder.RootComponents.Add<HeadOutlet>("head::after");
```

Program.cs

```
...  
<PageTitle>TimeTracking: @DateTime.FromDateTime(DateTime.Today)</PageTitle>  
...  
<HeadContent>  
    <meta name="description" content="Wie machen dies und das" />  
    <meta name="author" content="Blazor Author">  
    <link rel="icon" href="favicon.ico" type="image/x-icon">  
</HeadContent>  
...
```

*.razor

 Demo 

Error Boundary

“Html-Try-Catch-Block”

```
...  
<ErrorBoundary>  
  <ChildContent>  
    <BadComponent /> <!-- Löst eine Exception aus -->  
  </ChildContent>  
  <ErrorContent Context="ex">  
    <p>  
      Exception: @ex.Message  
    </p>  
  </ErrorContent>  
</ErrorBoundary>  
...
```

*.razor

 Demo 

Required parameters

```
[Parameter, EditorRequired]
public string Label { get; set; }
[Parameter]
public DateTime Value { get; set; }
[Parameter]
public EventCallback<DateTime> ValueChanged { get; set; }
[Parameter]
public Expression<Func<DateTime>> ValidationMessage { get; set; }
```

```
<DatePicker @bind-Value="Entry.Day"
            Label="Tag:" ValidationMessage="@(() => Entry.Day)"/>
```

DatePicker.razor: 31 + Edit.razor: 47

Demo

AOT- Ahead-of-time compliation

Ehrlich? Kein großer Unterschied

A photograph of a tropical village scene. In the foreground, a dirt road with some puddles leads towards a row of simple, single-story houses with brown roofs. The houses are surrounded by lush greenery, including several tall palm trees and other tropical plants. The sky is clear and blue. A dark blue horizontal band is overlaid across the middle of the image, containing the word 'Kestrel' in white text.

Kestrel

Kestrel Logging

Logging erweitert

```
{  
  "Logging": {  
    "LogLevel": {  
      "Default": "Information",  
      "Microsoft.AspNetCore": "Warning",  
      "Microsoft.AspNetCore.Kestrel.BadRequests": "Debug"  
    }  
  }  
}
```


Http Logging



Http Logging

- Loggen von Headers, Bodies, etc.

```
builder.Services.AddHttpLogging(logging =>
{
    // Customize HTTP logging here.
    logging.LoggingFields = HttpLoggingFields.All;
    logging.RequestHeaders.Add("My-Request-Header");
    logging.ResponseHeaders.Add("Sensitive");
    logging.ResponseHeaders.Add("Non-Sensitive");
    logging.MediaTypeOptions.AddText("text/html");
    logging.RequestBodyLogLimit = 4096;
    logging.ResponseBodyLogLimit = 4096;
});

app.UseHttpLogging();
```

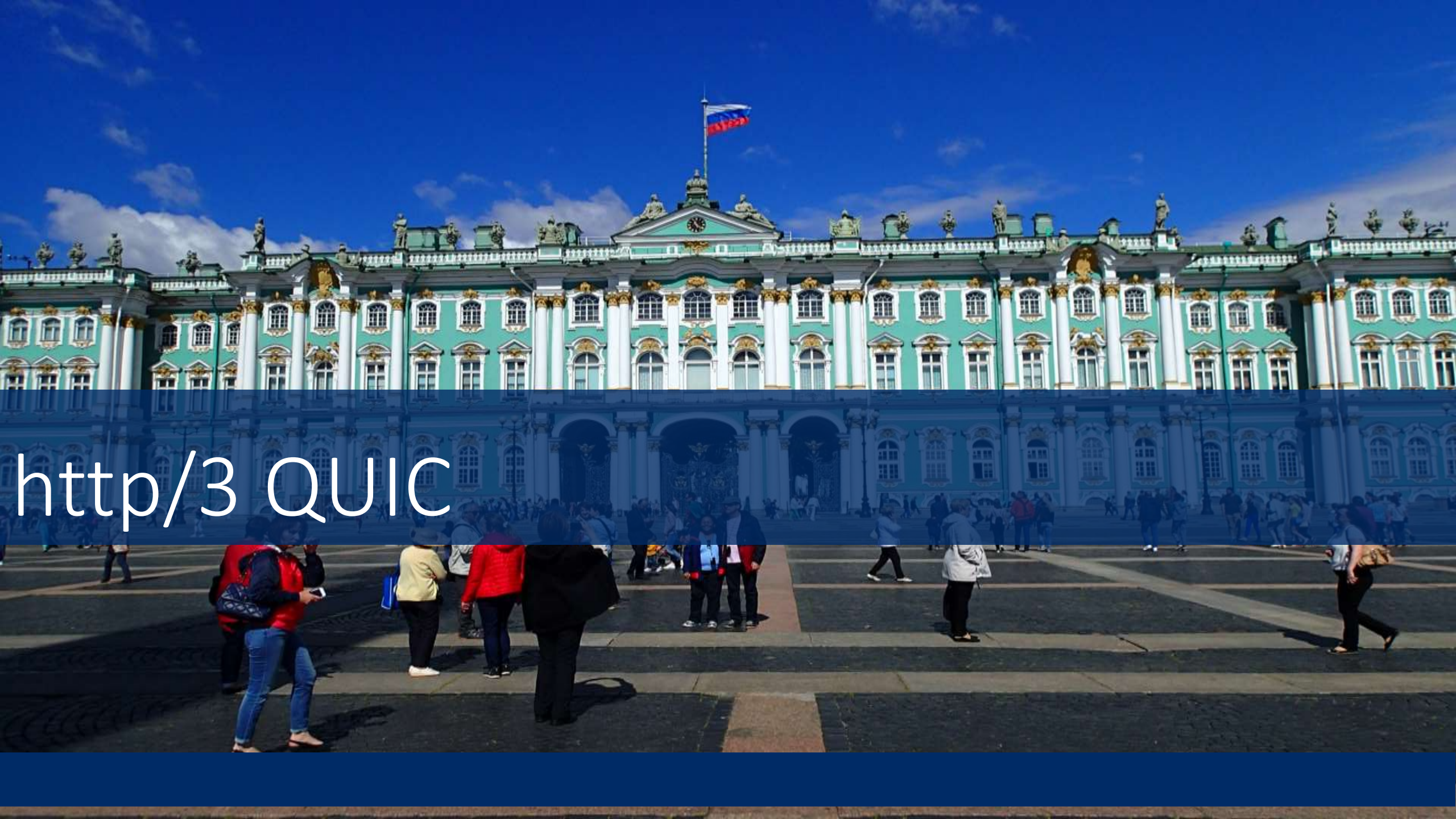
appsettings.json

Einstellungen unter `Microsoft.AspNetCore.Kestrel.*`

```
{
  "Logging": {
    "LogLevel": {
      "Default": "Information",
      "Microsoft.AspNetCore": "Warning",

      "Microsoft.AspNetCore.Kestrel.BadRequests": "Debug",
      "Microsoft.AspNetCore.Kestrel.Connections": "Debug",
      "Microsoft.AspNetCore.Kestrel.Http2": "Debug",
      "Microsoft.AspNetCore.Kestrel.Http3": "Debug"
    }
  },
  "AllowedHosts": "*"
}
```

Demo



http/3 QUIC



Coming soon

HTTP/3 (QUIC)

Http3 Demo Server: <https://pgjones.dev/>

The screenshot shows a web browser displaying the PG Jones website. The website has a dark header with the name 'P G Jones' and a main section titled 'Open Source' with the text 'These are the open source projects that I am involved with, either as the author or as a maintainer.' Below this is a logo for 'Quart' and the text 'Quart is a Python ASGI web'. The Chrome DevTools Network tab is open, showing a list of requests. The requests are all using the h3 protocol. A red arrow points to the 'error.svelte' request.

Name	Status	Protocol	Type	Initiator
start-88cd1609.css	200	h3	stylesheet	[index]
vendor-82544678.css	200	h3	stylesheet	[index]
start-df62298d.js	200	h3	script	[index]
vendor-2649d3b0.js	200	h3	script	[index]
layout.svelte-115dbf93.js	200	h3	script	start-df62298d.js
error.svelte-c5bb32d1.js	200	h3	script	start-df62298d.js
index.svelte-494823f1.js	200	h3	script	start-df62298d.js
Me5udbury.webp	200	h3	text/plain	vendor-2649d3b0.js
Quart.webp	200	h3	text/plain	vendor-2649d3b0.js
Hypercom.webp	200	h3	text/plain	vendor-2649d3b0.js
data:image/svg+xml=...	200	data	svg+xml	theme.css
favicon.ico	200	h3	text/html	Other

15 requests | 2.1 kB transferred | 475 kB resources | Finish: 130 ms | DOMContentLoaded: 68 ms | Load: 70 ms



<https://kinsta.com/blog/http3/>



Coming soon

HTTP/3 & Kestrel

Nur Windows 11/ Windows Server 2022 und neuer

```
var builder = WebApplication.CreateBuilder(args);

builder.WebHost.ConfigureKestrel((context, options) =>
{
    options.Listen(IPAddress.Any, 5001, listenOptions =>
    {
        // Use HTTP/3
        listenOptions.Protocols = HttpProtocols.Http1AndHttp2AndHttp3;
        listenOptions.UseHttps();
    });
});
```

 Demo 



Bonus

HashedId

Vorhersagbare IDs vermeiden

```
[HttpGet]
[Route("{id:hashids}")]
public ActionResult<CustomerDto>
    Get([ModelBinder(typeof(HashidsModelBinder))] int id)
{
    return Ok(customers.SingleOrDefault(c => c.Id == id));
}
```

```
Install-Package.AspNetCore.Hashids
```

 Demo 

K6 Load test

```
C:\Windows\System32\cmd.exe

scenarios: (100.00%) 1 scenario, 5 max VUs, 1m0s max duration (incl. graceful stop):
    * default: 5 looping VUs for 30s (gracefulStop: 30s)

running (0m30.0s), 0/5 VUs, 18825 complete and 0 interrupted iterations
default [=====] 5 VUs 30s

  allUsers

checks.....: 100.00% [ 18825 ] 0
data_received.....: 386 MB 13 MB/s
data_sent.....: 2.5 MB 82 kB/s
http_req_blocked.....: avg=3.62µs min=0s med=0s max=9.08ms p(90)=0s p(95)=0s
http_req_connecting.....: avg=1.87µs min=0s med=0s max=9.08ms p(90)=0s p(95)=0s
http_req_duration.....: avg=7.81ms min=2.01ms med=7.46ms max=212.74ms p(90)=9.32ms p(95)=10.11ms
    { expected_response:true }...: avg=7.81ms min=2.01ms med=7.46ms max=212.74ms p(90)=9.32ms p(95)=10.11ms
http_req_failed.....: 0.00% [ 0 ] 18825
http_req_receiving.....: avg=364.63µs min=0s med=0s max=9.02ms p(90)=972µs p(95)=1.6ms
http_req_sending.....: avg=9.52µs min=0s med=0s max=12.84ms p(90)=0s p(95)=0s
http_req_tls_handshaking.....: avg=0s min=0s med=0s max=0s p(90)=0s p(95)=0s
http_req_waiting.....: avg=7.44ms min=1.05ms med=7.13ms max=212.18ms p(90)=9.01ms p(95)=9.82ms
http_reqs.....: 18825 627.395714/s
iteration_duration.....: avg=7.96ms min=2.02ms med=7.58ms max=212.74ms p(90)=9.44ms p(95)=10.21ms
iterations.....: 18825 627.395714/s
vus.....: 5 min=5 max=5
vus_max.....: 5 min=5 max=5

C:\Workshops\asp.net-core\6.0\dotnetconsulting.UserManagement\dotnetconsulting.UserManagement.LoadTest>
```



<https://k6.io>

 Demo 

Fragen? Jetzt oder später!



www.dotnetconsulting.eu



Links



@Tkansy



tkansy@dotnetconsulting.eu



www.dotnetconsulting.eu