

ASP Core 3.1 inkl VueCLI mit mehreren VueApps verwenden

```
using System;
using System.Collections.Generic;
using System.IO;
using System.Linq;
using System.Threading.Tasks;
using Microsoft.AspNetCore.Builder;
using Microsoft.AspNetCore.Hosting;
using Microsoft.AspNetCore.Http;
using Microsoft.AspNetCore.SpaServices;
using Microsoft.Extensions.DependencyInjection;
using Microsoft.Extensions.FileProviders;
using Microsoft.Extensions.Hosting;
using VueCliMiddleware;

namespace ASPMultipleVueSPA
{
    public class Startup
    {
        // This method gets called by the runtime. Use this method to
        // add services to the container.
        // For more information on how to configure your application,
        // visit https://go.microsoft.com/fwlink/?LinkID=398940
        public void ConfigureServices(IServiceCollection services)
        {
        }

        // This method gets called by the runtime. Use this method to
        // configure the HTTP request pipeline.
        public void Configure(IApplicationBuilder app,
            IWebHostEnvironment env)
        {
            if (env.IsDevelopment())
            {
```

```

app.UseDeveloperExceptionPage();
}

app.UseRouting();

app.UseEndpoints(endpoints =>
{
endpoints.MapGet("/", async context =>
{
await context.Response.WriteAsync("Hello World!");
});

/*
* // Wichtig: In vue.config.js publicPath eintragen! z.B.
module.exports = {
publicPath: '/en/'
};
*/

endpoints.MapToVueCliProxy(
pattern: "en/{*path}",
options: new SpaOptions { SourcePath = "ClientApp" },
npmScript: (System.Diagnostics.Debugger.IsAttached) ? "serve"
: null,
port: 8080,
https: false,
runner: ScriptRunnerType.Npm,
regex: "Compiled successfully",
forceKill: true,
wsl: false
);

endpoints.MapToVueCliProxy(
pattern: "fr/{*path}", // Wichtig: In vue.config.js publicPath
eintragen: publicPath: '/fr/'
options: new SpaOptions { SourcePath = "ClientApp2" },
npmScript: (System.Diagnostics.Debugger.IsAttached) ? "serve"
: null,

```

```
port: 8081,  
https: false,  
runner: ScriptRunnerType.Npm,  
regex: "Compiled successfully",  
forceKill: true,  
wsl: false  
);  
  
});  
  
}  
}  
}
```