Kan de gemaakte app in cache draaien, zodat er offline gewerkt kan worden?

## Doel

De gebruikers van onze app hebben lang niet altijd internetverbinding, zo lopen ze vaak in kelders en parkeergarages, waar het hebben van internet lang niet altijd gegarandeerd is. Ook kan het voorkomen dat gebruikers geen tablet hebben met een data abonnement.

Toch moeten deze gebruikers onze app kunnen benaderen om zo toch de lokale data te kunnen gebruiken. Met dit onderzoek ga ik uitzoeken welke mogelijkheden hiervoor zijn.

## Strategie



Deze deelvraag wordt beantwoord met de hulp van Library(Literature study) waarna ik de vergaarde informatie ga toepassen door middel van een prototype(Workshop).

## Uitwerking

Om meer inzicht te krijgen in welke mogelijkheden er zijn, heb ik eerst zoekwerk gedaan, waarbij ik vele blogs en pagina’s af heb gezocht.

**PWA (service worker)**

Wanneer je naar offline Blazor app zoekt krijg je alleen maar het advies om er een PWA (progressieve Web Application) van te maken. Hiermee download je de applicatie, eigenlijk dus de hele web app, door middel van een knop op de browser. Hiermee wordt er op jouw device een icoontje op je home-scherm aangemaakt, waardoor je de app kan starten. Ook kan de app in dit geval gestart worden zonder dat er internet verbinding is, aangezien alle content, css, libraries en cache op de device zijn gedownload. De gebruiker heeft in principe alleen maar eenmalig een internet verbinding nodig om zijn applicatie te downloaden (en in ons geval om de metingen naar de lokale database te downloaden)



Huidige browser support PWA

Om een applicatie een PWA te maken zijn er een aantal requirements:

* De data wordt verstuurd over een beveiligde HTTPS verbinding.
* Er moet een service worker worden geïnstalleerd die het caching afhandelt.
* Er moet een app manifest worden ingevuld, bestaande uit enkele simpele properties.

Om een Blazor PWA applicatie te maken, dient er alleen een vinkje afgevinkt te worden bij het maken van een project. Helaas heb ik daar vooraf niet aan gedacht.

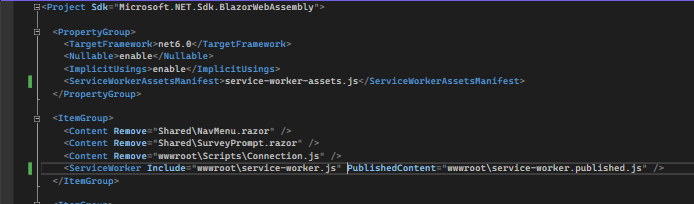
Afbeelding met tekst

Automatisch gegenereerde beschrijving

Aanmaken van PWA bij nieuw project

Toch is er een manier om een Blazor app simpel om te bouwen naar een PWA. Hiervoor dient de eerder genoemde Service Worker toegevoegd te worden aan het bestaande project. De stappen om mijn niet-PWA om te zetten naar een PWA zijn:

* Het downloaden van vier files downloaden van Microsoft GITHub: icon-512.png, manifest.json, service-worker.js, service-worker.published.js
* Het toevoegen van deze files aan de index.html van mijn applicatie
* Het registreren van de serviceworkers in mijn project



Toevoegen van serviceworkers aan csproj

Met deze simpele stappen zorg ik er voor dat ik binnen vijf minuten mijn regulieren WASM applicatie heb omgezet naar een Blazor PWA. Wanneer de app nu wordt geopend zie ik dat ik de app kan installeren.

Afbeelding met tekst

Automatisch gegenereerde beschrijving

Installeren PWA

Bij het testen is meteen te zien dat zelfs wanneer de browser offline is het hoofdscherm geladen wordt.

## Conclusie

Voorafgaand aan dit onderzoek was ik bang dat ik erg veel verschillende manieren voorgeschoteld kreeg. Gelukkig ben ik door de vele adviezen van lotgenoten in de richting van een PWA gedwuwd. Hiermee ben er ik meteen, zonder het uitproberen van andere opties, zeker van dat dit de manier is om de applicatie offline bruikbaar te maken. Hiermee blijven we zowel als website als offline applicatie beschikbaar voor al onze gebruikers, of ze nou een desktop, laptop, telefoon of tablet gebruiken.

## Bronnen

Galloway, J. (2020, 4 november). *Building a Progressive Web App with Blazor*. Visual Studio Blog. https://devblogs.microsoft.com/visualstudio/building-a-progressive-web-app-with-blazor/

*How do I create a Blazor WebAssembly PWA to work offline? | Blazor FAQ*. (z.d.). Syncfusion. https://www.syncfusion.com/faq/blazor/webassembly/how-do-i-create-a-blazor-webassembly-pwa-to-work-offline

*Cache Storage in Blazor WebAssembly .NET 6 - Blazor School*. (z.d.). https://blazorschool.com/tutorial/blazor-wasm/dotnet6/cache-storage-658620