

Fhir On Pi - 1



Soms moet je iets doen wat eigenlijk **te moeilijk** voor je is. Niet per ongeluk, maar “met voorbedachten rade”.

Op het verste punt van een wandeling nét nog even dat onbekende pad inslaan, de keukenmixer openschroeven om te kijken of je hem kunt repareren, een gevallen porceleinen schaal lijmen hoewel je eigenlijk weet dat het niet gaat houden. Je weet dat

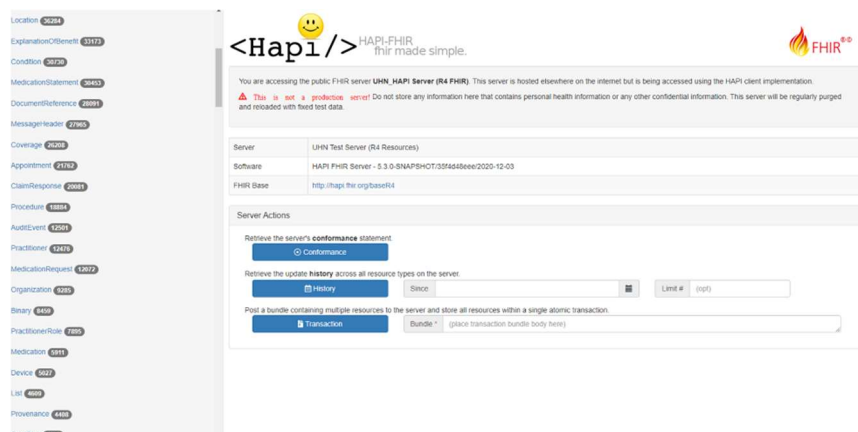
het eindresultaat niet geweldig zal zijn. Je komt te laat thuis, hebt nog dagen spierpijn, je krijgt de mixer niet meer in elkaar omdat er plastic lipjes zijn afgebroken, de schaal valt na een dag spontaan weer uit elkaar. Je doet het gewoon voor de lol en -een beetje- om ervan te leren.

In dezelfde sfeer verliep “**Fhir On Pi**”. Ik rommel al een paar jaar met Arduino en Raspberry Pi computertjes en ik weet vanuit mijn werk iets (maar niet heel veel) van FHIR en, in plaats van een extra lange wandeling of onmogelijke reparatie, wilde ik de Raspberry Pi als FHIR-server inrichten.

Om bij het einde te beginnen: het is me uiteindelijk gelukt.

Over “de weg ernaar toe”: het duurde véél langer dan ik had verwacht, ik weet nu nóg beter dat ik niet veel van de Raspberry Pi weet en ik weet nu héél goed hoe complex en uitgebreid een FHIR-server is. Maar het allerbelangrijkste is: het was héél leuk om te doen. Verder blijf je er droog en warm bij en je houdt schone handen.

Alle stappen, ook de mislukkingen, heb ik bijgehouden en omdat er hopelijk meer mensen zijn die zo iets wel eens willen proberen heb ik besloten om in een aantal artikelen uit te leggen wát ik heb gedaan, waar het misging en wat er wél tot een succes heeft geleid.



Ik zal niet altijd uitleggen waarom iets misging, maar als je mijn stappen opvolgt heb je straks ook een draaiende FHIR-server op een computertje zo groot als een half pakje boter. En in

die mini-computer zitten meer dan 800.000 observations, 130.000 patienten en 12.000 medicatievoorschriften!

Wat je mag verwachten.

De komende dagen zal ik, iedere dag, een aflevering van **Fhir On Pi** op LinkedIn zetten. Iedere aflevering zal een enigszins afgebakend deel van mijn “project” omvatten, een afgebakend stukje werk waarmee ook een (soms beperkt) meetbaar resultaat is bereikt.

In de afleveringen zal ik ook over mijn mislukkingen vertellen. Ik doe dat op zo’n manier dat duidelijk is dat je die weg niet op moet gaan (tenzij je een extra uitdaging zoekt natuurlijk), je hoeft dus niet bang te zijn dat ik je alle fouten laat maken die ik ook gemaakt heb. In de volgende aflevering zal ik kort ingaan op de Raspberry Pi en de eerste voorbereidingen daarmee.

Voor wie alvast wat wil doen

Voor dit project gebruik ik een Raspberry Pi 4 model B met 8Gb intern geheugen. Ik heb hem via de post gekocht bij onderstaande link¹, maar hij is ook op allerlei andere plekken te koop. De Pi draait onder het Raspbian operating system, een Unix-variant. Het helpt als je in ieder geval een klein beetje kennis van Unix hebt, met de voornaamste commando’s.

De Pi staat flink te stampen, vooral bij het “bouwen” van de Java-applicaties dus ik raad niet aan om met een kleinere Pi (Raspberry Pi 3B) te werken, of minder intern geheugen te nemen. Ik gebruikte eerst de volledige Raspbian versie (Raspbian “Buster” Full) op een 8Gb SD-kaartje, maar de SD-kaart liep vol. Uitbreiden met een USB-disk werkte ook niet lekker omdat er (heel) veel in de root-directories wordt geschreven. Wat wél werkte is Raspbian “Buster” Lite op een 16Gb SD kaart. De configuratie die ik dus aanraad is:

- Raspberry Pi 4 Model B, minimaal 8 Gb intern
- Raspbian “Buster” Lite op een 16Gb SD kaartje

Verder heb je (uiteraard) een netwerkverbinding (vast of Wifi) nodig, een voeding voor de Pi (niet te krap) en (om de Pi te configureren) een HDMI beeldscherm, USB toetsenbord en eventueel een USB muis.

Tot de volgende!

¹ <https://www.sossolutions.nl/raspberry-pi-4-model-b-4gb>