

SOA/MSA beschrijving voor UHasselt

Ismaël Warnants

Aangezien ik vernomen heb dat UHasselt vorig jaar (of 2 jaar geleden) problemen had met hun backend omdat het niet goed onderhouden was, zou ik graag met een SOA voorstellen hoe dit kon vermeden worden. Ik ga er van uit dat hun backend in .NET draaide en dat daar steeds op is aangepast aangezien ik niet heel de backend functionaliteit ken. Volgens wat ik gehoord heb was het niet voldoende onderhouden om nog binnen een redelijke tijd features toe te kunnen voegen. Ik vermoed dat dit vermeden had kunnen worden als er met een SOA gewerkt was. Maar er bestaan ook andere manieren om dat te vermijden.

De IT dienst van UHasselt kon bijvoorbeeld werken met een MSA, hierbij zouden verschillende aspecten van de universiteit opgesplitst worden in services die ieder een eigen taak hebben. Enkele voorbeelden van services die UHasselt had kunnen maken:

- Voor het beheer van gegevens in het studentendossier
- Voor het plannen van lessen- en examenroosters
- Voor het beheer van betalingen, pingping, ...
- Voor het beheer van bibliotheek gegevens
- Voor het verspreiden van informatie over opleidingen
- Voor het beheer van de studiegids
- Voor het platform voor personeel
- Voor het IT-portaal
- Voor de authenticatie van studenten, personeel, onderzoekers, ...
- ...

Al deze services hadden samen kunnen werken om dezelfde functionaliteit te bieden die UHasselt in .NET had gemaakt. Maar hierbij was het flexibeler om sneller componenten te wijzigen aangezien ze onafhankelijk van elkaar kunnen functioneren. De uitbreiding naar nieuwe features is dan ook makkelijker omdat de services gemakkelijk samen kunnen communiceren. Waardoor het aansluiten van nieuwe services gemakkelijk zal zijn.

Om een voorbeeld te geven:

Stel UHasselt wilt een platform toevoegen om bij Umami eten te bestellen, hiervoor is bijvoorbeeld de authenticatie nodig van de student, maar ook hun betalingen met pingping. Hiervoor zou UHasselt dan zich meer kunnen focussen op het platform zelf en de werking ervan, want de authenticatie en pingping services bestaan al en zijn klaar om met het nieuwe platform toe te voegen zonder dat deze bestaande services gewijzigd moeten worden. Het platform moet enkel de api aanspreken van de authenticatie en pingping en dan integreren in het platform.

In het pong project is een gelijkaardig concept toegepast, hierbij is een vrij simpel pong spel uitgebreid met authenticatie dat door een aparte service wordt afgehandeld. Er is een andere service die de data bijhoudt van alle spelers hun scores. Er is een service die het spel zelf draait. En er is een service die het scorebord weergeeft. Het is geen complex programma, maar toont hoe een complexer programma zou kunnen overeenkomen.

Beschrijving van het pong spel:

- Express rendert het pong spel
- C# handelt de authenticatie af en slaat de authenticatie gegevens op
- Laravel rendert het scorebord aan de hand met de data die in python opgeslagen staat
- Python slaat de scores van de spelers op en hangt alles aan elkaar

Stel wij zijn een site die van plan zijn om nu naast het pong spel een nieuw spel aan te willen bieden. Dan kunnen dezelfde gebruikers ook met hun bestaande account aan de nieuwe spellen geraken omdat de authenticatie service kan hergebruikt worden.