



Ontwikkeling van een REST API

Plan van Aanpak

Bachelor in de toegepaste informatica
keuzerichting applicatieontwikkeling

Academiejaar 2024-2025

Nicolas Van Dyck

Campus Geel, Kleinhoefstraat 4, BE-2440 Geel

Inhoudsopgave

INHOUDSOPGAVE	2
1.ACHTERGROND	3
2.HUIDIGE SITUATIE	3
3.STAGEOPDRACHT	4
4.VERWACHT RESULTAAT (SHARED VISION)	5
4.1. Inleiding	5
4.2. Verwachtingen	5
5.BUSINESS CASE	6
5.1. Inleiding	6
5.2. Voordeel	6
5.3. Doelgroep	6
6.PROJECTOMVANG EN GRENZEN	7
6.1. Inleiding	7
6.2. Scope	7
6.3. Out of Scope	7
7.PLANNING	7
8.COMMUNICATIE	8

Inleiding

1. Achtergrond

Pension Architects werd opgericht in 2009 en ondersteunt klanten bij het beheren van pensioenfondsen voor grote ondernemingen en sectoren, met een sterke focus op automatisatie. Dit beheer omvat onder andere de berekening van het aanvullend pensioen per werknemer en de communicatie hierover. Om dit proces zo efficiënt en eenvoudig mogelijk te maken, heeft Pension Architects een eigen beheersplatform ontwikkeld. Dit platform wordt momenteel onderhouden door een team van elf fulltime softwareontwikkelaars en is geoptimaliseerd voor gebruik met de clouddiensten van Amazon Web Services (AWS).

2. Huidige situatie

Het platform is ontwikkeld in Java en maakt gebruik van Apache Tapestry, een component gebaseerd webframework. Hoewel Tapestry enkele jaren geleden een populaire keuze was, heeft het inmiddels moeite om bij te blijven met de nodige updates. Hierdoor ontstaat een belangrijk technisch aandachtspunt: de frontend en backend zijn sterk verweven (tightly coupled), waardoor wijzigingen aan de frontend vaak directe aanpassingen in de backend vereisen en vice versa. Dit beperkt de flexibiliteit van het systeem en maakt onderhoud en uitbreiding complexer.

Om dit probleem aan te pakken, is Pension Architects al begonnen met een API te ontwikkelen die tussen de bestaande service-layer en het nieuwe frontend framework zal leven. Dit zal de frontend en backend van elkaar scheiden (loosely coupled architecture), waardoor het platform toekomstbestendig wordt en een overstap naar een ander framework op termijn eenvoudiger wordt.

3. Stageopdracht

Tijdens mijn stage richt ik mij op het uitbreiden van de API, zodat deze in de toekomst daadwerkelijk in gebruik kan worden genomen. Dit proces omvat analyse, ontwerp, overleg en implementatie en wordt bij elke epic opnieuw doorlopen. Om ervoor te zorgen dat mijn stagementor en ik op dezelfde lijn zitten, stel ik eerst een Plan van Aanpak op. Hierin beschrijf ik de verwachte doelstellingen en de business case. Daarnaast maak ik een voorlopige planning, zodat we een duidelijk beeld krijgen van de haalbaarheid en de timing van de verschillende taken. Tot slot beschrijf ik hoe en wanneer de communicatie met mijn stagementor en stagebegeleider zal verlopen.

Doelstelling

4. Verwacht Resultaat (Shared vision)

4.1. Inleiding

Het doel van deze stage is het ontwikkelen van een REST API boven op de bestaande service-layer, zodat de koppeling tussen de frontend en backend wordt losgemaakt. Deze aanpassing zal bijdragen aan een flexibelere architectuur en maakt het op termijn eenvoudiger om over te schakelen naar een ander framework.

Om dit te realiseren zal ik onderzoek doen naar best practices voor API-design, beperkte beveiliging opzetten in de vorm van JWT, logging en notificaties implementeren, AWS aanpassen en de service layer verbeteren. Vervolgens zal ik deze API testen en documenteren, zodat het beheer en de toekomstige uitbreiding ervan zo efficiënt mogelijk verloopt.

4.2. Verwachtingen

De stage zal bijdragen in een functionerende REST API die de front- en backend scheidt en de onderlinge communicatie optimaliseert. Daarnaast zal de architectuur verbeterd worden, zodat uitbreidbaarheid en onderhoudsgemak gewaarborgd blijven. De documentatie en richtlijnen die tijdens de ontwikkeling worden opgesteld, zorgen ervoor dat de API verder uitgebreid kan worden. Door al deze aanpassingen aan de API zal het platform efficiënter en toekomstbestendiger worden.

Naast deze technische verbeteringen biedt de stage een leerervaring op het gebied van Java webontwikkeling, objectgeoriënteerd programmeren, design patterns en technologieën zoals Spring, Tapestry, Maven en AWS.

5. Business case

5.1. Inleiding

Het beheersplatform van Pension Architects speelt een essentiële rol in het administratieve beheer van pensioenfondsen voor bijna een miljoen werknemers. De huidige architectuur, die gebaseerd is op Apache Tapestry, zorgt echter voor een sterke koppeling tussen de frontend en backend. Deze nauwe verwevenheid maakt het systeem minder flexibel en brengt uitdagingen met zich mee op het vlak van onderhoud en uitbreidbaarheid.

5.2. Voordeel

Om deze beperking aan te pakken, wil Pension Architects een API ontwikkelen boven op de bestaande service-layer. Dit maakt het mogelijk om de front- en backend van elkaar te koppelen, waardoor nieuwe technologieën of frontend frameworks eenvoudiger kunnen worden geïntegreerd zonder ingrijpende wijzigingen aan de backend. Door deze aanpassing wordt het platform niet alleen flexibeler, maar ook is het ook beter voorbereid op toekomstige technologische evoluties.

5.3. Doelgroep

Pension Architects developers: Omdat de implementatie van de API geen extra functionaliteit aan de webapp zal toevoegen is de enige en ook wel de belangrijkste doelgroep van deze implementatie de developers zelf. Door het gebruik van de API zal het team op termijn de frontend eenvoudig met de backend kunnen koppelen door gebruik te maken van de API.

6. Projectomvang en grenzen

6.1. Inleiding

Om vanaf het begin duidelijkheid te scheppen, wordt bepaald welke taken binnen en buiten de scope van de stageopdracht vallen. Dit zorgt ervoor dat beide partijen een goed beeld hebben van de omvang van het project. Deze afbakening is echter niet volledig strikt; als tijdens de stage blijkt dat een taak, die eerder buiten de scope viel, toch noodzakelijk is, kan deze in overleg worden opgenomen. Dit zal doorgaans ten koste gaan van een andere taak. Deze afbakening dient vooral om de haalbaarheid van de opdracht te bewaken, maar binnen een professionele omgeving is enige flexibiliteit noodzakelijk om in te spelen op veranderende behoeften.

6.2. Scope

- Logging en Metrics bijhouden van API REST requests, foutmeldingen.
- Notificaties versturen naar Slack in geval van kritieke foutmeldingen
- IAC (infrastructure as code). Soms zal het nodig zijn om deze aan te passen om de API in AWS te doen werken.
- Aanpassingen in AWS. Omdat de API in AWS draait, zal het soms nodig zijn om hier ook aanpassingen te doen zodat bepaalde publieke REST endpoints aangeroepen kunnen worden
- De service layer bijwerken zodat die kan samenwerken met de API.
- Documenteren van het geleverde werk.
- Onderzoeken en analyses maken voor implementaties die nieuw zijn voor het team van Pension Architects

6.3. Out of Scope

- Security. Het is niet de bedoeling dat ik de gehele beveiliging van de REST API zal doen.
- Alles wat niet met de REST API te maken heeft. Denk aan frontend aanpassingen.

7. Planning

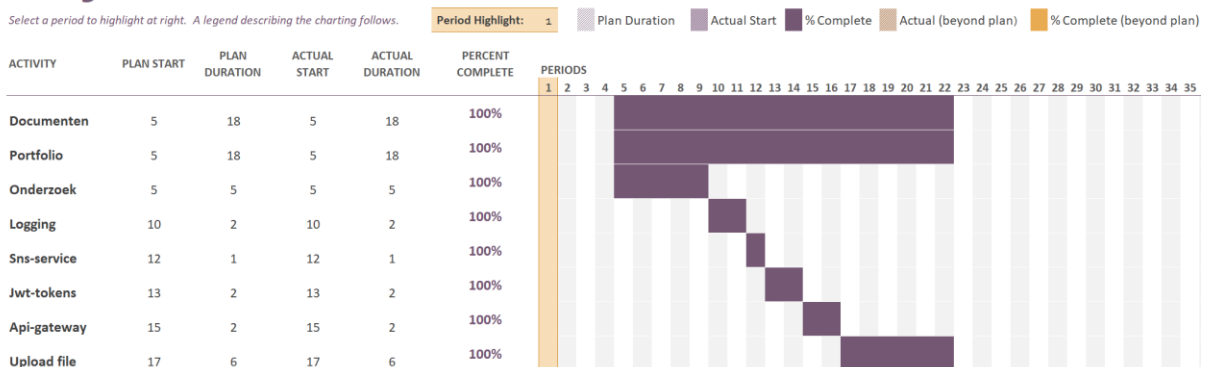
Om een gestructureerd verloop van de stage te garanderen, is een voorlopige planning opgesteld. Deze planning dient als richtlijn en kan indien nodig worden bijgesteld. Een vast schema biedt houvast, maar te rigide afspraken kunnen problematisch zijn wanneer bepaalde onderdelen meer tijd vergen dan voorzien.

In de eerste weken ligt de focus op het opstellen van het Plan van Aanpak en het onderzoeken van de vereisten voor het ontwikkelen van een productieklare API. Dit omvat zowel een analyse van de bestaande implementaties als een evaluatie van mogelijke verbeteringen.

Na deze onderzoeksfase zal ik mijn bevindingen bespreken met de stagementor, waarna we afspraken maken over de exacte scope van de implementatie. Dit zorgt ervoor dat de stageopdracht duidelijk wordt afgebakend en voorkomt dat werkzaamheden buiten de oorspronkelijke doelstellingen vallen. Op basis van deze afbakening wordt een verdere planning opgesteld, met een duidelijke focus op de implementatie van de API.

Project Planner

Select a period to highlight at right. A legend describing the charting follows.



8. Communicatie

De communicatie met mijn stagementor verloopt voornamelijk op vrijdag op kantoor en buiten de werkuren via Slack. Dit directe contact maakt het mogelijk om snel vragen te stellen en problemen op te lossen.

Daarnaast vindt op vrijdag een korte stand-up plaats, waarin de stagementor samen met het team de voortgang van de huidige sprint en openstaande issues bespreekt. Dit biedt een gelegenheid om knelpunten te identificeren en gezamenlijk oplossingen te formuleren.

Voor communicatie met mijn stagebegeleider is afgesproken dat ik hem via Microsoft Teams kan contacteren voor zaken die buiten de stageopdracht vallen. Daarnaast zijn er vaste contactmomenten om de voortgang te bespreken. Het eerste formele overleg tussen mij, mijn stagementor en mijn stagebegeleider vindt plaats in de derde week van de stage, op locatie bij de stageplaats. Dit moment dient als kennismaking en biedt

de gelegenheid om een voorlopige versie van het Plan van Aanpak te presenteren en feedback te ontvangen.

Gedurende de stage zijn er terugkommomenten op de campus. Het eerste moment vindt plaats rond week negen en is bedoeld om het verbeterde Plan van Aanpak voor te stellen aan de stagebegeleider en medestudenten. Tijdens deze sessie wordt verdere feedback gegeven.

Tussentijds vult de stagementor een evaluatie in via het stageportaal, zodat ik mijn aanpak waar nodig kan bijsturen.

Het volgende terugkommoment vindt plaats rond week veertien. Tijdens deze bijeenkomst presenteer ik mijn realisatiedocument en ontvang ik feedback op zowel het Plan van Aanpak als het realisatiedocument. Daarnaast wordt op dit moment ook de tussentijdse evaluatie besproken.