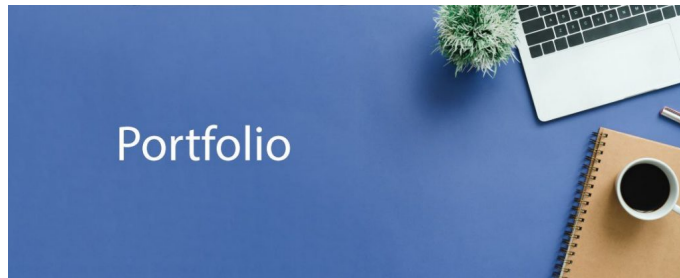


# PDR

Persoonlijke ontwikkeling



*Auth eur: Peeters, R.H.F. Rik*

*Datum: 02-03-2021*

*Versie: 1.1*

*Student aan Fontys University of Applied science*

# Samenvatting

Voor dit semester was het zoals elk semester van belang om leerdoelen aan te tonen en hiermee je bekwaamheid te meten. In dit document leest U hoe ik mij heb georiënteerd in de verschillende leerdoelen van dit semester. Hierbij kunt u ook lezen hoe ik alle leerdoelen afgerond met een proficient en de bewijslast die wordt uitgeschreven. Ook leest u in dit document mijn ervaringen en wat mijn leermomenten waren. Deze leermomenten zijn zowel op het gebied van groepsprojecten als individueel. In elk leerdoel beschrijf ik het proces in chronologische volgorde. Hierbij wordt het begin status omschreven en hoe ik mijzelf heb georiënteerd om vervolgens het leerdoel uit te werken. Aan het einde van elk leerdoel beschrijf ik waarom ik persoonlijk vind waarom ik een bepaald leerdoel heb gehaald met een minimale "Proficient".

# Inhoudsopgave

<b>1</b>	<b>Inleiding</b>	<b>1</b>
<b>2</b>	<b>Leeruitkomsten</b>	<b>2</b>
2.1	Ontwikkelen van bedrijfssoftware als teaminspanning . . . . .	2
2.1.1	Ontwikkel process . . . . .	3
2.1.2	Eind beoordeling / reflectie . . . . .	4
2.2	Context Based Research . . . . .	5
2.2.1	Ontwikkel process . . . . .	5
2.2.2	Eind beoordeling / reflectie . . . . .	7
2.3	Voorbereiding op levenslang leren . . . . .	8
2.3.1	Toegepaste technieken . . . . .	8
2.3.2	Eind beoordeling / reflectie . . . . .	10
2.4	Schaalbare architecturen . . . . .	11
2.5	Ontwikkeling en uitvoering (DevOps) . . . . .	12
2.5.1	Ontwikkel process . . . . .	12
2.5.2	Eind beoordeling / reflectie . . . . .	13
2.6	Clouddiensten . . . . .	14
2.6.1	Ontwikkel process . . . . .	14
2.6.2	Eind beoordeling / reflectie . . . . .	16
2.7	Veiligheid door ontwerp . . . . .	17
2.7.1	Ontwikkel process . . . . .	17
2.8	Gedistribueerde gegevens . . . . .	18
2.8.1	Ontwikkel process . . . . .	18
2.8.2	Eind beoordeling / reflectie . . . . .	18
<b>3</b>	<b>Retrospect</b>	<b>21</b>
<b>4</b>	<b>Conclusie</b>	<b>22</b>
	<b>Appendices</b>	<b>23</b>
<b>A</b>	<b>SWOT</b>	<b>25</b>
<b>B</b>	<b>Context based research</b>	<b>31</b>

# Hoofdstuk 1

## Inleiding

( 500 words)

Dit PDR-verslag documenteert de verschillende processen die hebben plaatsgevonden voor het ontwikkelen van het uiteindelijke product. In dit geval is dat de software 'Flow-control' die het distributieproces voor distributiecentra digitaliseert en automatiseert. Ook het groepsproject 'Smilie' komt aan bod. Verder worden de knelpunten, leermomenten en onderzoeksvragen nauwkeurig behandeld. Tot slot worden de conclusies en aanbevelingen voor het project beschreven.

### Individueel

Voor mijn individueel project ga ik een ERP-pakket (Enterprise Resource Planning) ontwikkelen genaamd Flowcontrol. Flowcontrol gaat er voor zorgen dat verschillende processen die binnen een distributiecentrum worden uitgevoerd, worden gedigitaliseerd maar ook worden geautomatiseerd. Hierdoor hebben medewerkers meer tijd voor andere werkzaamheden.

### Groep

Sinds covid-19 maakt de GGZ gebruik van videoverbindingen om hun cliënten te bellen om contactmomenten te voorkomen. Een nadeel hiervan is dat het voor medewerkers soms moeilijk is om in te schatten hoe cliënten zich voelen. Hiervoor is de oplossing bedacht om beter op te merken hoe cliënten zich voelen, door middel van het meten van de hartslag en galvanische huidreacties. In de toekomst zullen ook gezichtsuitdrukkingen en de toon van de stem worden gemeten. De resultaten van deze metingen zijn gevoelige informatie en moeten veilig worden verstuurd naar de diensten die deze gegevens omzetten in logische informatie.

Onze groep gaat werken aan de veilige overgang van gegevens. Onze groep zal er ook voor zorgen dat de diensten in de cloud zullen gaan draaien in plaats van lokaal, wat nu gebeurt.

## Hoofdstuk 2

# Leeruitkomsten

De leeruitkomsten die dit semester worden behandeld en ook moeten worden aangetoond zijn als volgt.

- Ontwikkelen van bedrijfssoftware als teaminspanning
- Context Based Research
- Voorbereiding op levenslang leren
- Schaalbare architecturen
- Clouddiensten
- Veiligheid door ontwerp
- Gedistribueerde gegevens

### 2.1 Ontwikkelen van bedrijfssoftware als teaminspanning

Dit semester wil ik leren om enterprise software op te leveren in teamverband, aan een echte klant. Hierbij bespreken we samen met de klant zijn wensen en functionaliteiten. Voor deze samenwerking ga ik de Agile Scrum methodiek gebruiken. Daarnaast moet het systeem dat door ons als groep wordt ontwikkeld voldoen aan de GDPR-wetgeving.

Dit leerdoel ga ik bereiken door een strakke planning en goede communicatie met het team en klant. Ook zullen we samen technische aspecten bespreken en doelgericht onderzoek uitvoeren om zo tot oplossingen te komen voor onze problemen.

Hierbij is het van belang een gemotiveerd team te hebben met leergierigen individuen en een klant die pro-actief meedenkt over wat hij wil in het eindproduct. Communicatie is hierbij belangrijk met het team, maar ook met de externe partijen.

Tijdens ons groepsproject is er communicatie met verschillende teams en contact personen. Zo is er een groep bezig aan de front-end en is er een afstudeer student bezig met de technische kant van de front-end.

Ik ga dit leerdoel meetbaar maken door middel van een evaluatie op het einde van elke sprint. Hierbij noteer ik mijn sterke en zwakke punten die worden aangekaart door mijn team zodat ik deze in de toekomst kan verbeteren. Ook is het project en de deelproducten die wij als groep opleveren een goede indicatie van mijn bekwaamheid tot deze leeruitkomst.

## 2.1.1 Ontwikkel process

### Sprint 0

**Evaluatie: 12-03-2021** Dit is de ontdekkingsfase; wat is het project, welke technieken komen erbij kijken en welke technieken kunnen we het beste toepassen voor ons project. Hoe gaan we als groep een succesvol project opleveren voor onze klant; wat zijn functionele en non functionele eisen/wensen. Hierover hebben we samen met de klant gesproken in verschillende online bijeenkomsten. De uitkomsten hiervan zijn vastgelegd in de deel documenten. Deze zijn terug te vinden in het canvas dashboard.

Hieronder is een opsomming van de uitkomsten die de professionaliteit van de sprint evalueren. Hierbij heb ik persoonlijk aan mijn teamleden gevraagd waar voor mij verbetering ligt bij deze sprint.

#### Sterke punten:

- Denkt diep na over een topic en brengt een goed technisch beeld over naar teamgenoten
- Levert een goede bijdrage aan het ontwikkelen

#### Verbeter punten:

- Nog n.v.t.

### Sprint 1

**Evaluatie: 01-04-2021** Dit was onze eerste sprint. Tijdens deze sprint hebben we de opdracht vastgesteld en een ontwerp gemaakt van ons project. Bij het maken van het ontwerp was het belangrijk ervoor te zorgen dat de app een modulaire opbouw krijgt, wat er voor zorgt dat het goed overdraagbaar is. Om deze modulariteit te behalen hebben we gekozen voor een microservice architectuur. Het is belangrijk dat het project overdraagbaar is, omdat dit project in de toekomst zal worden ontwikkeld door externe project groepen. Ook is het van belang dat het ontwerp aan de "GDPR" regelgeving voldoet. De GDPR-regelgeving beschermt de gebruiker en zijn gegevens in ons systeem. Ook hebben we tijdens deze sprint verschillende prototypes gemaakt waarvan de klant zeer onder de indruk was. In deze prototypes hebben we nieuwe technische methodes uitwerkt en getest, die we laten willen gaan toepassen in het definitieve project. Deze ontwerpen en prototypes zijn ook besproken met vak experts / docenten. Dit stelt de kwaliteit vast van onze deelproducten en zo kunnen we vroegtijdig de nodige wijzigingen aanbrengen. Ook is deze documentatie belangrijk voor de overdraagbaarheid aan een toekomstig team die aan dit project verder zullen gaan werken.

Hieronder is een opsomming te vinden van de eindbespreking van sprint 1. Hierbij hebben we net zoals voorgaande sprint de tips en tops besproken en waar voor mij verbeteringen liggen in de sprints die nog komen.

#### Sterke punten:

- Initiatief nemend.
- Gemotiveerd en leergierig.
- Brede technische kennis.
- Je zoekt dingen tot het bot uit.
- Algemene samenwerking; je helpt waarnodig mee en helpt anderen goed.

**Verbeter punten:**

- Te lang zichzelf vast kunnen klampen aan een ding. Perfectie is een streven maar het moet niet tegen werken.
- Je kan meer initiatief tonen om aan nieuwe zaken te beginnen wanneer je huidige af zijn.

**Sprint 2**

**Evaluatie: 23-04-2021** Dit was onze eerste echte print oplevering waar we ook echt code gingen opleveren. Hierbij werken we volgens de agile scrum methodiek en hebben we elke 2 weken een sprint oplevering. Deze sprint oplevering presenteren we ons werk voor de verschillende teams, de project eigenaar Wouter en natuurlijk bespreken we naderhand hoe de oplevering is gegaan met onze docenten. Door de positieve feedback en de goede samenwerking en teaminspanning bij het project kan ik mijzelf beoordelen met een "proficient". Natuurlijk zijn we nog niet op het einde van het semester daarom zie ik deze proficient onder voorbehoud. In de aankomende sprint ga ik deze beoordeling definitief maken.

**2.1.2 Eind beoordeling / reflectie**

Wij als groep hebben een mooi degelijk product opgeleverd. In dit project zijn alle wensen verwerkt die door de klant werden opgedragen. Ook hebben we in dit project alle technieken die voor dit semester van toepassing waren verwerkt. Dit project kan in de toekomst verder worden doorontwikkeld door andere project groepen. Binnen ons team is er een ontzettend goede communicatie. Afspraken waren daarom ook duidelijk binnen het gehele team. Ook ondersteunde we elkaar waar nodig, sommigen meer dan anderen maar over het algemeen ben ik zeer tevreden.

Door de behaalde resultaten definieer ik mijn bekwaamheid tot dit leerdoel als:

**"PROFICIENT"**

## 2.2 Context Based Research

Dit semester wil ik leren om doelgericht onderzoek te doen waarvan de uitkomsten kunnen bijdragen aan het ontwikkelen van technische en complexe software oplossingen. Onderzoek is nodig om kennis op te doen bij een bepaald onderwerp en om deze zo beter te kunnen begrijpen. Vanuit daar kun je je kennis stapsgewijs (onderzoek voor onderzoek) opbouwen om zo tot een gepaste oplossing te komen voor een probleem.

Door de onderzoeksmethodieken te gebruiken die in het DOT-framework staan omschreven kan ik doelgericht onderzoek doen. Voor het onderzoek ga ik verschillende methodieken combineren om zo tot een oplossing te komen op mijn probleem.

Dit leerdoel ga ik meetbaar maken door onderzoeksdocumenten op te leveren die door zowel mij als door een ander als nuttig worden beschouwd en daarmee ook een bijdrage kunnen leveren aan een oplossing van een complex probleem. Ik definieer mijn kwaliteit op de feedback die ik heb ontvangen van vak bekwame experts "docenten".

Bij zowel het groepsproject als het individueel project heb ik verschillende casestudies gemaakt die voor een klant kunnen helpen bij het maken van een keuze om zo een probleem op te lossen.

### 2.2.1 Ontwikkel process

#### Case Studies

**Evaluatie: 24-04-2021** Tijdens het groepsproject hebben we zes verschillende case studies moeten uitwerken. Deze case studies gaven een complex probleem weer waar een oplossing bij gevonden moest worden. De problemen deden zich voor op verschillende aspecten binnen het ontwikkelen van software. De case studies zijn uitgewerkt door middel van het toepassen van verschillende methodieken uit het DOT-framework. De casestudies die tijdens het profthaak groepje zijn behandeld, zijn goed afgerond en de feedback van de leraren is verwerkt. Daarom beschouw ik mijn huidige oriëntatie als "georiënteerd". Deze case studies zijn terug te vinden in het canvas bij de opdrachtenoverzicht van project groep 3. Bij de verschillende case studies hebben we taken verdeeld. Mijn bijdrage aan de case studies zijn als volgt:

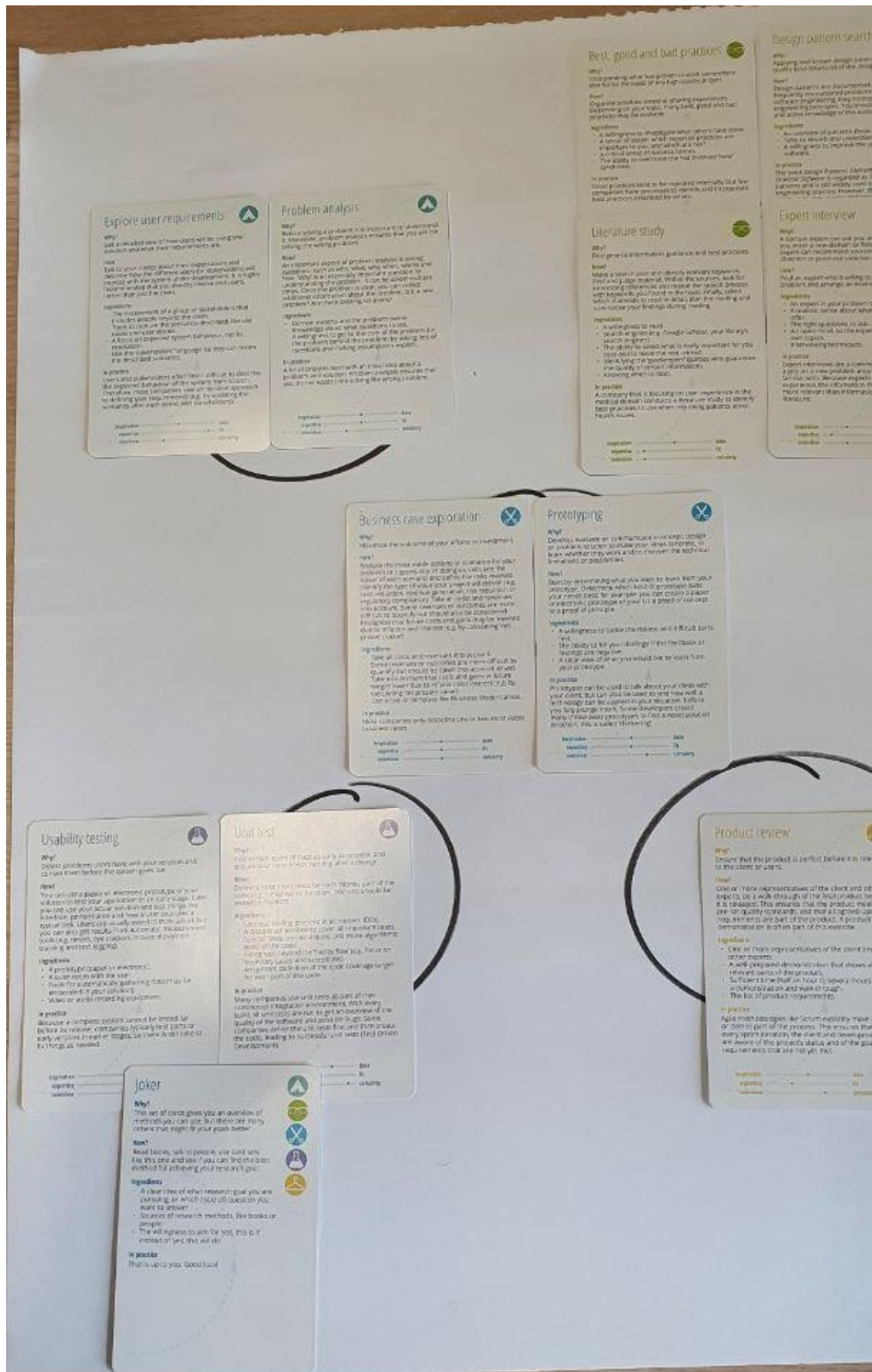
- Gebrainstormd over de case studie; wat houdt deze in en wat vragen ze van ons
- Onderzoeksvragen opgesteld gebaseerd op de brainstorm sessie
- Onderbouwde feedback gegeven op elkaars uitwerking op een vraag en waar nodig ondersteund.
- Onderzoeksvragen en deelvragen uitgewerkt.

Gebaseerd op de feedback van onze docenten ligt onze kwaliteit hoog. Alle feedback die we ontvangen wordt genotuleerd en wordt in de volgende case studie meegenomen.

#### Onderzoek

**Evaluatie: 05-04-2021** Voor het individuele project heb ik ook context based research toegepast voor het "cache van data binnen mijn project". Ook tijdens dit onderzoek is het dot onderzoeks framework toegepast. Uit het dot framework heb ik de volgende methodes uitgekozen:

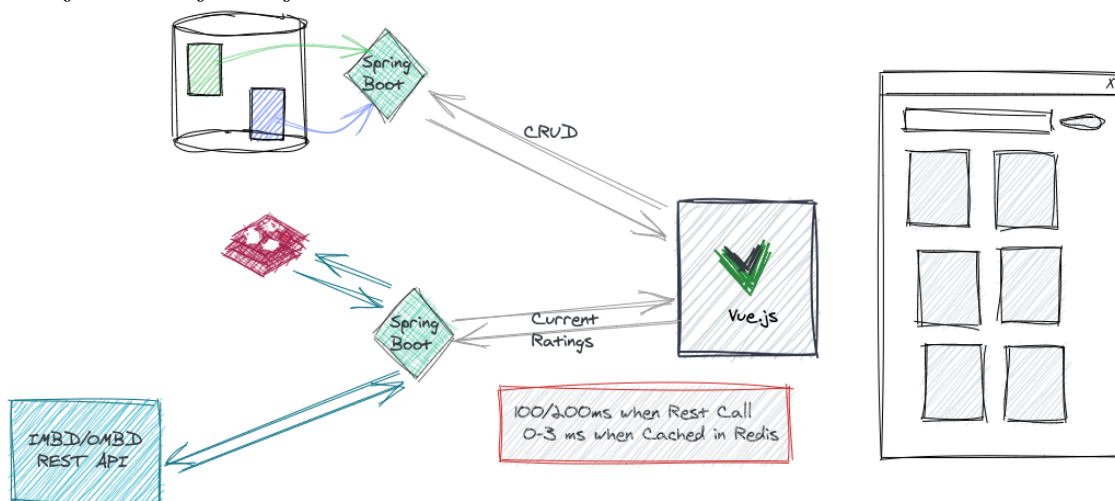




Deze methodes zijn besproken met Tom Langhorst docent gespecialiseerd het contextueel onderzoeken.

De onderzoeksvraag luidt als volgt: "Hoe wordt caching toegepast op enterprise software niveau voor specifieke doeleinde binnen het java framework springboot?"

Tijdens het onderzoek zijn de toepassingen van caching mijzelf zeer duidelijk geworden. Bij het onderzoek hebben we ook enkelen prototypes gemaakt de de snelheid en de toepassing van caching goed laat zien. Na het prototype hebben we aan een vak expert om feedback gevraagd tijdens dit feedback moment hebben we het over verschillende topics gehad bij komen kijken bij het cachen van data.



## 2.2.2 Eind beoordeling / reflectie

Ik ontzette veel geleerd over de contextueel onderzoek doen naar een topic en heb geleerd hoe ik een onderzoeksvraag moet samenstellen. Gebaseerd op de verschillende case studies en groeps onderzoeken en het individueel onderzoek beschouw ik mijn huidige bekwaamheid tot dit leerdoel op:

**"PROFICIENT"**

## 2.3 Voorbereiding op levenslang leren

Voor mijn studie worden veel nieuwe technieken aangedragen door school, maar hoe ga ik er in de toekomst zelf voor zorgen dat ik up-to-date blijf met de laatste trends die zich in de ICT-wereld ontwikkelen. Hiervoor is het daarom belangrijk nu al na te denken over hoe ik dit ga aanpakken. Daarom wil ik dit semester leren om mijzelf voor te bereiden om levenslang te leren.

Om dit te kunnen bereiken moet ik mijzelf blijven motiveren om door te leren na mijn studie. Dit kan zowel door cursussen te volgen als door zelfstudie met doelgericht onderzoek naar nieuwe ontwikkelingen binnen de ICT-wereld. Door up-to-date te blijven van de nieuwste ontwikkelingen blijf ik als software engineer goed op de markt liggen.

Dit leerdoel is meetbaar te maken door het toepassen van deze nieuwe technieken binnen mijn toekomstige projecten.

### 2.3.1 Toegepaste technieken

Hieronder bespreek ik hoe ik verschillende technieken heb onderzocht en toegepast. Deze technieken waren voor mij nog van onbekend terrein. De definitie van life long learning is het analyseren en het toepassen van nieuwe 'trend' technieken. Deze trends laat ik hieronder zien.

Hoe ga ik nieuwe trends ontdekken? Dit ga ik doen door actief online in te zoeken naar methodieken die voor mij een bepaald probleem oplossen of zelf vermakkelijken. Wanneer ik ontdek dat ik bij een techniek mijzelf repeteer en onnodig een complexe oplossing hierbij ga zoeken. Ga ik online zoeken hoe andere mensen tot een oplossing zijn gekomen als zij mij vervolgens informeren dat hier een nieuwe oplossings- trend in is ontwikkeld ga ik hier zelf mee experimenteren. Ook op youtube volg ik veel vak experts die nieuwe technieken analyseren en hier een eigen visie aan geven. Wanneer deze experts goed onderbouwen waarom het toepassen van deze trend een positieve invloed zal hebben op toekomstige projecten zal ik hier zelf natuurlijk ook mee gaan experimenteren. Daarom geeft youtube voor mij een zeer goede bijdrage aan het voordragen van nieuwe 'trends' en technieken. Wanneer ik met een probleem zit ga ik persoonlijk op stackoverflow zoeken naar oplossingen van andere mensen. Hierbij komt het ook voor dat deze personen mij introduceren aan een nieuwe techniek en de toepassing hiervan. Zelf houd ik van een "gedecentraliseerde" informatie voorziening. Hiermee bedoel ik dat een bedrijf mij niet een nieuwe techniek moet aandragen voor hun eigen gewin. Ik moet zelf hier de toegevoegde waarden van inzien en dit doe ik door verschillende bronnen te raadplegen.

### Microservices

**Eerste evaluatie - 16-03-2021** Voor mijn individueel project ben ik bezig aan een distributie app. Context over dit project is te lezen in de inleiding Deze distributie app ga ik ontwikkelen met een modulaire architectuur die bestaat uit microservices. Helaas wist ik in het begin nog vrij weinig wat microservices precies inhielden en voor welke toepassingen ze werden gebruikt. Na onderzoek en verschillende prototypes heb ik mijzelf wegwijs gemaakt in de toepassing hiervan.

Voor mijn individueel project ben ik begonnen met een ontwerp en verschillende workflows die deze app zou moeten gaan automatiseren. Vervolgens ben ik begonnen met het ontwikkelen van prototypes specifiek voor het project. In deze prototypes heb ik ook de technieken uitgewerkt die ik van plan was te gaan te gebruiken. Vervolgens ben ik deze prototypes gaan bespreken met Software docente Merel. Hierbij heb ik de verschillende technieken laten zien die ik van plan was te gaan gebruiken. Bij deze technieken kun je bijvoorbeeld denken aan HATEOAS HAL en springboot cloud gateway. Dit zijn allemaal

technieken die zorgen voor modulariteit binnen mijn microservices. De terugkoppeling die ik hierbij heb ontvangen was positief en we hebben tijdens dit terugkoppelingsmoment ook over complexe ideologieën gesproken, zoals de generieke response content types van mijn rest API. Na dit gesprek heeft merel geconstateerd dat mijn bekwaamheid tot verschillende leerdoelen op "beginning" mag worden gezet.

## Deployment en schalen

**Eerste evaluatie - 02-03-2021** De eerste weken heb ik onderzoek gedaan naar hoe Docker werkt en hoe ik een image kan maken van een project. Vervolgens heb ik een compose file gemaakt die het mogelijk maakt verschillende images tegelijkertijd te starten.

Met deze compose files heb ik vervolgens een github actions (CI/CD) workflow aangeemaakt die de verschillende testen automatiseerde. Deze workflow is te zien in de project repository op github.

De volgende stap is het deployen van deze image naar een cloud omgeving om zo de app live te zetten.

## Protocollen

**Eerste evaluatie - 02-03-2021** De eerste weken heb ik onderzoek gedaan naar de verschillende protocollen die worden gebruikt voor het streamen van data. Dit was van belang voor het groepsproject waarbij een online video platform moet worden ontwikkeld. Welke protocollen passen bijvoorbeeld het beste bij het huidige groepsproject. UTP of TCP / RTMP of webrtc of zelf een ander protocol. Hierbij heb ik gekeken hoe bekende streaming diensten tot een oplossing zijn gekomen en waarom dat ze voor een bepaald protocol hebben gekozen. Hierbij ben ik tot de conclusie gekomen dat voor onze toepassing het beste kan worden gewerkt met webrtc. Webrtc is een opvolger van het verouderde, maar nog steeds veel gebruikte RTMP-protocol.

## Gateway

**Eerste evaluatie - 20-04-2021** Omdat mijn applicatie met een microservice architectuur werkt en omdat het van belang is dat de frontend goed kan communiceren met deze back-end, heb ik er voor gekozen om een gateway te implementeren. Een gateway is een soort van service balie van een applicatie die je naar de juiste plek verwijst. Elke microservice binnen mijn app heeft zijn eigen poort wanneer deze vervolgens gaat schalen kunnen dit er nog meer worden. Als oplossing hiervoor heb ik gekozen om Zuul gateway in combinatie met Eureka discovery te implementeren. Deze dependencies zijn beide ontwikkeld door Netflix. Eureka kan automatisch zien wanneer een applicatie zich 'aanmeldt'. Wanneer er dus een request naar de Zuul gateway wordt gestuurd kan Zuul aan Eureka vragen wat het juiste doorverwijs adres is.

Hieronder een afbeelding van de microservices die zich aan hebben gemeld bij Eureka.

Instances currently registered with Eureka			
Application	AMIs	Availability Zones	Status
ARTICLE-SERVICE	n/a (1)	(1)	UP (1) - 192.168.16.120:article-service:7078
GATEWAY-AUTHSERVICE	n/a (1)	(1)	UP (1) - 192.168.16.120:gateway-authservice:7070
ZUUL-SERVICE	n/a (1)	(1)	UP (1) - 192.168.16.120:zuul-service:8762

### 2.3.2 Eind beoordeling / reflectie

Door het uitwerken van verschillende trends binnen zowel het individueel als het groepsproject oriënteer ik mijzelf als:

**"PROFICIENT"**

De bewijslast van het toepassen van deze technieken zijn zowel hierboven omschreven en terug te vinden in de implementatie binnen de projecten.

## 2.4 Schaalbare architecturen

Ik wil leren hoe ik enterprise software maak die schaalbaar en gedistribueerd is. Binnen de enterprise software moeten de microservices onafhankelijk van elkaar opereren. Microservices zijn onafhankelijke programma's die met elkaar kunnen communiceren door middel van protocollen. Dit is mogelijk door een goed technische ontwerp van het op te leveren product. Dit ontwerp wordt gedocumenteerd in verschillende technische documenten zoals het Software Architecture Document (SAD).

Hierbij is het van belang dat ik de nodige kennis op doe over microservices. Daarnaast zal ik onderzoek moeten doen naar hoe deze protocollen onderling werken en hoe ik deze moet toepassen.

Hierbij is het ook van belang om goede tooling toe te passen zoals UML of C4 voor een goed technische ontwerp. Bij het ontwikkelen is het ook belangrijk om een goede en intelligente IDE te gebruiken, zodat ik efficiënt te werk kan gaan.

Ik ga dit leerdoel meetbaar maken door een goed project op te leveren, zowel voor het individueel project als het groeps project. Hierbij is het belangrijk dat ik stevige argumenten heb om mijn keuzes te onderbouwen die ik heb gemaakt tijdens het ontwikkel proces. Het project moet natuurlijk schaalbaar zijn dit ga ik aantonen door een microservice architectuur toe te passen.

Dit leerdoel sluit ook goed aan bij het leerdoel "Voorbereiding op levenslang leren". In dit hoofdstuk omschrijf ik hoe ik microservices heb toegepast voor mijn individueel project. Graag wil ik u, de lezer, dan ook verwijzen om meer te lezen over microservices in dat hoofdstuk.

## 2.5 Ontwikkeling en uitvoering (DevOps)

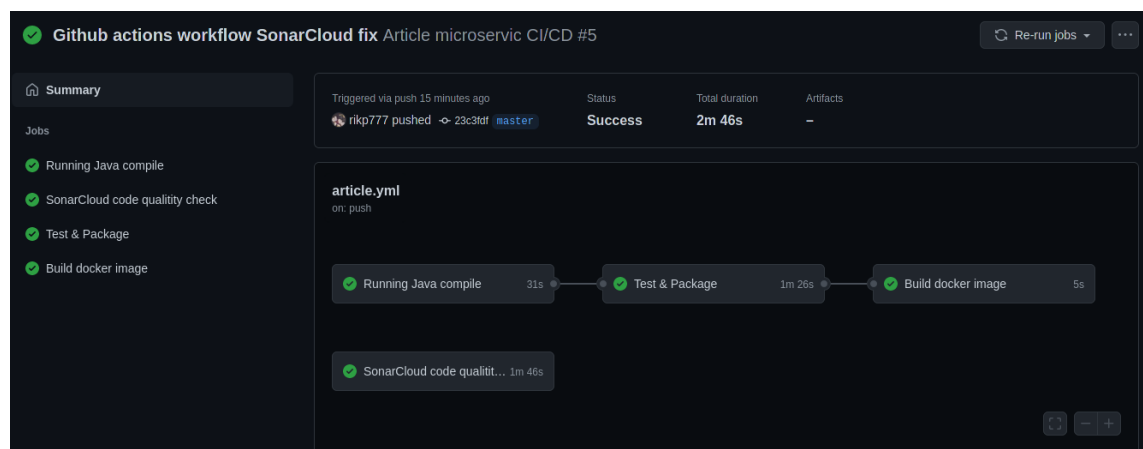
Voor dit leerdoel wil ik leren hoe ik een automatische test pipeline (CI/CD) opzet door middel van Github actions. Hierbij wordt een automatische test flow opgezet die valideert of een build is geslaagd. Wanneer deze build en de daarbij horende testen succesvol zijn uitgevoerd, wordt de image naar een production environment gestuurd waarna de software vervolgens live wordt gezet en gebruikt kan worden door de klant. Dit gebeurt door middel van een Kubernetes cluster. Ook ga ik tooling gebruiken voor het bijhouden van de te ontwikkelen functionaliteiten. Hierbij wordt in het groepsproject Devops gebruikt en voor mijn individueel project ga ik hier het Github issue-board voor gebruiken.

Dit leerdoel is meetbaar te maken door de leraar proactief te informeren over hoe het ontwikkelproces zich vordert. Ook ga ik mijn Github actions CI/CD pipeline laten zien waar de code wordt gecontroleerd op geslaagde testen en kwaliteit. De kwaliteit van de code word getest met SonarCloud.

### 2.5.1 Ontwikkel proces

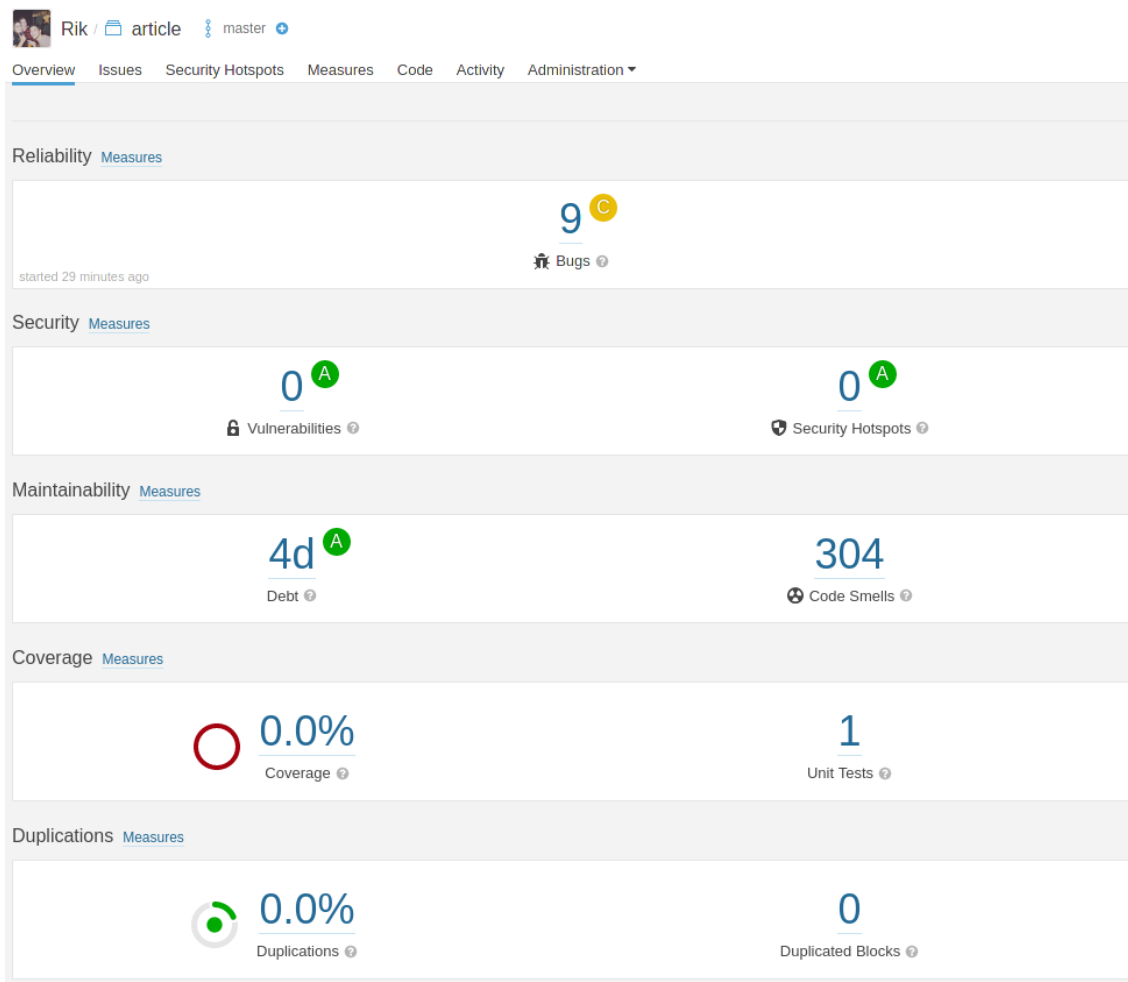
#### Github actions

**Evaluatie: 05-04-2021** Voor het toepassen van de CI/CD pipeline verwacht ik niet enorm veel werk. Dit is mede omdat ik in semester drie maar ook in vier hier al mee heb gewerkt en dit al eerder heb geïmplementeerd. Daarom zie ik tijdens dit semester meer mogelijkheden voor verbetering van dit proces. Voor mijn CI/CD module heb ik gekozen voor het toepassen van Github actions. Mijn pipeline controleert op zowel geslaagde testen als op de code kwaliteit. Elke microservice binnen mijn applicatie heeft zijn eigen pipeline met checks. Hieronder is de uitwerking van een van deze pipelines voor de microservice "Article". Hierbij is te zien dat deze pipeline verschillende checks heeft waar hij doorheen loopt.



Via SonarCloud wordt de code gecontroleerd op kwaliteit en wordt aangekaart welke verbeteringen er kunnen plaatsvinden.





Omdat ik een gehele CI/CD pipeline heb geïmplementeerd voor mijn individueel project en mede door de uitwerking hierboven oriënteer ik mijzelf als:

**"PROFICIENT".**

Ik zou de beoordeling kunnen verbeteren naar een 'outstanding' door middel van het implementeren van een kubernetes cluster.

### 2.5.2 Eind beoordeling / reflectie



## 2.6 Clouddiensten

Voor het hosten in de cloud is enige kennis nodig van het toepassen en welke vormen er zijn van cloud hosting. Ook is het belangrijk om kost bewust te zijn. Wanneer zijn bepaalde toepassingen voordelig zoals het toepassen van een serverless applicatie.

Om dit leerdoel te kunnen halen ga ik kijken naar de verschillende toepassingen van cloud hosting. Hiervoor zal ik aanwezig zijn tijdens de verschillende lectures over dit topic en zal ik meedoen met de workshops. Tijdens deze workshop ga ik een mini applicatie hosten ook ga ik een serverless functie hosten.

### 2.6.1 Ontwikkel process

Ik heb gekeken naar de kosten die komen kijken bij het hosten van een cloud service.

**Estimate the cost of using AWS Lambda functions:**

Number of executions (month)  
10000000

Memory allocation  
128 MB

Estimated average duration (ms)  
300

Include free tier?  
☒ Yes ☐ No

**Results**

<b>Request costs:</b>	<b>\$1.80/month</b>
<b>Execution costs:</b>	<b>\$0/month</b>
<b>Total AWS Lambda costs:</b>	<b>\$1.80/month</b>

Hierbij heb ik gekeken naar het verschillen tussen serverless (FAAS). Hierbij wordt het verschil duidelijk ook heb ik gekeken naar de voordelen en nadelen. Ook ben ik mij bewust van laadtijden en ken ik het onderscheid tussen een cold en warm start. Ook ben ik mij bewust over hoe de werking van een cloud dienst met containers en het health checken. Zelf heb ik ook een serverless function gemaakt die dient als een prototype. Iedereen kens het liedje "99 Bottles of Beer" wel hiervoor heb ik een scripje geschreven in javascript die ik vervolgens heb gedeployed hieronder het resultaat.

Cloud Functions

Function details

EDITDELETECOPY

test-function

Viewed at: 2/24/2024 10:53 AM, 10:16 AM, ...

Metrics

DETAILSDetails

STATUSStatus

FUNCTIONFunction

TRIGGERTrigger

PERMISSIONSPermissions

LOGSLogs

TESTINGTesting

Log (Showing 10 messages)

Event

Filter by type

Log

1

2024-02-27T01:16:19.430Z

test-function

logFunction

Test new event, pass in array.

10

0

2

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

3

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

4

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

5

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

6

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

7

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

8

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

9

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

10

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

11

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

12

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

13

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

14

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

15

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

16

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

17

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

18

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

19

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

20

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

21

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

22

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

23

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

24

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

25

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

26

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

27

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

28

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

29

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

30

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

31

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

32

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

33

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

34

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

35

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

36

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

37

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

38

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

39

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

40

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

41

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

42

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

43

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

44

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

45

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

46

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

47

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

48

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

49

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

50

2024-02-27T01:16:19.430Z

test-function

logFunction

10

0

Ook heb ik voor mijn eigen project een continuous deploy pipeline gemaakt naar het google cloud platform in een Kubernetes cluster. Deze pipeline heb ik gemaakt in github actions. De github actions is relatief nieuw maar een mooie oplossing voor een CI/CD module. Een github action workflow is makkelijk aan te maken en te configureren in een eigen yml file. Hieronder is mijn uitwerking:

```

#git clone https://github.com:root@root:~/k8s-terraform
name: Setup, Build, Publish, and Deploy
run-on: ubuntu-latest
environment: production

# Steps
- name: Checkout
  uses: actions/checkout@v2

# Setup gcloud CLI
- name: google-github-actions/setup-gcloud@v2.0
  id: gcloud
  with:
    service_account_key: ${ secrets.GKE_SA_KEY }
    project_id: ${ secrets.GKE_PROJECT }

# Configure Docker to use the gcloud command-line tool as a credential
# helper for authentication
- name: Docker login
  run: |
    gcloud --quiet auth configure-docker

# Set the GKE credentials so we can deploy to the cluster
- name: google-github-actions/get-gke-credentials@v2.1
  id: gke
  with:
    cluster_name: ${ env.GKE_CLUSTER }
    location: ${ env.GKE_ZONE }
    credentials: ${ secrets.GKE_SA_KEY }

# Build the Docker image
- name: Build
  working-directory: ./article
  run: |
    DOCKER_BUILDKIT=1 docker build --tag "${HOSTNAME}/${PROJECT_ID}/${IMAGE}:${ITHUB_SHA}" .

# Push the Docker image to Google Container Registry
- name: Publish
  run: |
    DOCKER_BUILDKIT=1 docker push "${HOSTNAME}/${PROJECT_ID}/${IMAGE}:${ITHUB_SHA}"

# Set up kustomize
- name: Set up kustomize
  working-directory: ./article
  run: |
    curl -sfs kustomize https://github.com/kubernetes-sigs/kustomize/releases/download/v3.1.0/kustomize_3.1.0_linux_amd64.tar.gz | sudo tar xz -C /usr/bin

# Deploy the Docker image to the GKE cluster
- name: Deploy
  working-directory: ./article
  run: |
    kubectl apply -f ./article.yaml
    kubectl wait --for=condition=Ready pod --all
    kubectl rollout status deployment/DEPLOYMENT_NAME

```

In deze afbeelding kunnen we zien dat we verschillende stages hebben eerst gaan we de app builden wanneer dit vervolgens is gelukt gaan we pushen naar een register. Er zijn verschillende register zoals docker hub in dit geval heb ik gebruik gemaakt van de google cloud register. Wanneer het pushen vervolgens is gelukt gaan we de image deployen. Veel student hebben een account aangemaakt met hun eigen credit card gegevens waarbij je vervolgens 300 credit gratis kunt ontvangen helaas beschik ik niet over zon credit card. Daarom heb via school een google cloud account aangevraagd hierbij heb ik 30 credits gekregen om te besteden en een uitwerking te maken. Hieronder is een afbeelding te zien van de workflow die naar de Kuberneters cluster pushed.

```
Setup, Build, Publish, and Deploy
Run google-github-actions/setup-gcloud@v2.2.0
Run gcloud --quiet auth configure-docker
Run google-github-actions/get-github-token@v2.1.1
Build
Publish
1 Run DOCKER_BUILDKIT=1 docker push "ghcr.io/${PROJECT_ID}/${NAME}-ghcr"
14 The push refers to repository (eu.gcr.io/${PROJECT_ID}/${NAME}-ghcr)
15 7a24302b0e0a: Preparing
16 7a24302b0e0a: Preparing
17 6a302b0e0a: Preparing
18 6a302b0e0a: Preparing
19 ca302b0e0a: Preparing
20 a0302b0e0a: Preparing
21 50a4c2b0f5a: Preparing
22 a0302b0e0a: Pushing
23 50a4c2b0f5a: Pushing
24 ca302b0e0a: Layer already exists
25 a0302b0e0a: Layer already exists
26 50a4c2b0f5a: Layer already exists
27 7a24302b0e0a: Pushed
28 a0302b0e0a: Pushed
29 7a24302b0e0a: Pushed
30 6a302b0e0a: Pushed
31 1a4a0b0e0a: Pushed
Set up Kubernetes
Deploy
1 Run ./kubernetes-deploy.sh
2 ./kubernetes-deploy.sh
3 ./kubernetes-deploy.sh
4 ./kubernetes-deploy.sh
5 ./kubernetes-deploy.sh
6 ./kubernetes-deploy.sh
7 ./kubernetes-deploy.sh
8 ./kubernetes-deploy.sh
9 ./kubernetes-deploy.sh
10 ./kubernetes-deploy.sh
11 ./kubernetes-deploy.sh
12 ./kubernetes-deploy.sh
13 ./kubernetes-deploy.sh
14 ./kubernetes-deploy.sh
15 ./kubernetes-deploy.sh
16 ./kubernetes-deploy.sh
17 ./kubernetes-deploy.sh
18 ./kubernetes-deploy.sh
19 ./kubernetes-deploy.sh
20 ./kubernetes-deploy.sh
21 ./kubernetes-deploy.sh
22 ./kubernetes-deploy.sh
23 ./kubernetes-deploy.sh
24 ./kubernetes-deploy.sh
25 ./kubernetes-deploy.sh
26 ./kubernetes-deploy.sh
27 ./kubernetes-deploy.sh
28 ./kubernetes-deploy.sh
29 ./kubernetes-deploy.sh
30 ./kubernetes-deploy.sh
31 ./kubernetes-deploy.sh
Error: Process completed with exit code 1.
```

Helaas kunnen we ook zien dat de deployment is mislukt. Wanneer we vervolgens naar de Kubernetes engine gaan kijken wat er mis is gegaan kunnen we zien dat de deployment is gefaald omdat hij niet de juiste Database parameters heeft en hierdoor niet volledig kan deployen.

Name	Status	Type	Pods	Namespace	Cluster
article-service-6dcbcd7778-s5wrk	CreateContainerConfigError	Pod	0/1	default	flowcontrol

**configmap "springdatasourcedriverclassname" not found: CreateContainerConfigError**

**Reason** configmap "springdatasourcedriverclassname" not found: CreateContainerConfigError

**Source** [article-service-6dcbcd7778-s5wrk](#)

Om een database aan te maken in de google cloud omgeving kost credits om dat ik helaas gelimiteerd ben in het gebruik van deze credits heb ik de keuze gemaakt geen database aan te maken in de cloud.

## 2.6.2 Eind beoordeling / reflectie

Ondanks het niet volledig online zetten van de applicatie op een Kubernetes cluster heb ik aangetoond dat ik dit in de praktijk wel kan doormiddel van de juiste parameters mee te geven, helaas waren hier te weinig credits en tijd voor. Door het uitwerken van verschillende trends binnen zowel het individueel als het groepsproject oriënteer ik mijzelf als:

**"PROFICIENT"**

## 2.7 Veiligheid door ontwerp

Voor het ontwikkelen van software is het van groot belang dat je software ook veilig is en voldoet aan de norm die wordt gesteld.

Ik ga de OWASP top 10 beveiligingsrisico's goed door te nemen en de zaken die ik in het vorige semester heb gedocumenteerd en onderzoek toepassen in zowel het groeps als het individueel project.

Door middel van het implementeren van HATEOAS Hal, OAuth2 en Spring Security ga ik aantonen dat ik over deze technieken beschik en dat ik mijn software op deze manier veilig kan maken. Ook ga ik proberen mijn software te kraken met de kennis die ik heb opgedaan uit de twee security specialisatie semesters.

### 2.7.1 Ontwikkel process

#### JWT Tokens

Een JSON Web Token (JWT) is een op zichzelf staande manier definieert voor het veilig verzenden van informatie tussen partijen als een JSON-object. Deze informatie kan worden geverifieerd en vertrouwd omdat ze digitaal ondertekend is. JWT's kunnen worden ondertekend met een geheim of een openbaar/particulier sleutelpaar.

**Evaluatie: 20-04-2021** Voor mijn authenticatie en authenticatie gebruik ik JWT tokens in de vorm van een bearer. Bearer authenticatie (ook wel token authenticatie genoemd) is een HTTP-authenticatieschema dat gebruik maakt van beveiligingstokens die bearer tokens worden genoemd. De naam "Bearer authenticatie" kan worden opgevat als "geef toegang aan de drager van dit token". Het bearer token is een cryptische string, meestal gegenereerd door de server in antwoord op een login verzoek. De client moet dit token meesturen in de Authorization header wanneer hij verzoeken doet aan beschermde bronnen: In mijn applicatie gebruik ik een token voor het authorizeren van interacties met mijn rest api. Elke microservice binnen mijn applicatie heeft een Token nodig wanneer er een request wordt gestuurd. Voor mijn eerste implementatie heb ik ervoor gezorgd dat het artikel service kan worden geautoriseerd.

#### HATEOAS HAL

HATEOAS (Hypermedia as the Engine of Application State) is een beperking van de REST architectuur die de RESTful-stijlarchitectuur uniek houdt ten opzichte van de meeste andere toepassingsarchitecturen. De term "hypermedia" verwijst naar alle inhoud die links bevat naar andere vormen van media, zoals afbeeldingen, films en tekst. Dit houdt dus in dat een api interactief kan worden gemaakt en dat het makkelijk is om naar relational data te verwijzen.

**Evaluatie: 20-04-2021** Omdat ik gebruik maak van microservices is het handig als een frontend makkelijk naar relationele data kan verwijzen. Dit doe ik door middel van HATEOAS met een gestandaardiseerde vorm in HAL formaat

## 2.8 Gedistribueerde gegevens

Het is belangrijk om met de klant te bespreken wat hij graag wil en hierbij onderscheid te maken tussen functionele eisen en non functionele eisen. Een functionele eis van een klant zijn meestal wat hij in de software wil hebben en wat het functioneel moet kunnen met features. Een non functionele eis is bijvoorbeeld hoe snel de app moet zijn en of dat de app in de cloud moet worden gehost of niet.

Ik ben mijzelf bewust van data-eisen en ik ontwikkel bedrijfssystemen die gebruik maken van gedistribueerde data tooling en best practices. Ik heb een kritische houding ten opzichte van mogelijke privacy en ethische kwesties die zich voor kunnen doen tijdens het ontwikkelen.

Voor het toepassen van deze regels is een algemene wet ontwikkeld die gebruikers online bepaalde rechten geeft om zijn of haar gegevens op te vragen en te verwijderen. Deze wetgeving / regelgeving heet de "GDPR" General Data Protection Regulation.

Dit leerdoel ik en het toepassen ervan wil ik gaan aantonen in het groepsproject waarbij dit zeer belangrijk is. Dit is belangrijk, omdat het over een persoon zijn gesteldheid gaat en dit kwetsbare data is die gevoelig voor hem of haar kan zijn. Daarom is het belangrijk dat de gebruikers kan bepalen wat hij met zijn gegevens wil doen.

### 2.8.1 Ontwikkel process

Binnen mijn applicatie wil ik er ook voor zorgen dat een gebruiker in deze kwestie personeel geen gevoelige data van zichzelf kan opslaan binnen mijn systeem. Mijn applicatie zal in deze geen gevoelige data van gebruikers opslaan om dat dit binnen mijn applicatie niet van belang is. Wel zorg ik ervoor dat gevoelige bedrijfs- informatie veilig wordt verwerkt. Dit ga ik doen door verschillende technieken toe te passen die standaard zijn binnen de OWASP top 10. Daarmee zorg ik dat mijn applicatie tot zekere hoogte niet kan worden gehackt en daarmee geen gevoelige bedrijfs- gegevens kan lekken. De framework die worden toegepast nemen een groot deel van de OWASP top 10 weg omdat ze ervoor zorgen dat hun systeem standaard aan deze lijst voldoet. Het is altijd mogelijk dat een developer per ongeluk een onveilige functionaliteit in bouwt, hierbij zal het framework dit dan ook aangeven in zijn benamingen van functies die je probeert toe te passen.

Voorafgaand bij het ontwikkelen van mijn applicatie heb ik rekening gehouden met de stroom die data gaat lopen hierbij heb ik voor een synchronous data stroom gekozen. Door dat een synchronous stroom afhankelijkheid creëert onder verschillende services wou ik caching gaan toepassen. Hierbij contextueel onderzoek gedaan naar caching wat ook goed aansluit bij het onderwerp gedistribueerde gegevens. Caching is het tijdelijk opslaan van gegevens op een service die deze nodig heeft wat voor grote voordelen kan zorgen. Deze voordelen kunnen worden gelezen in het onderzoeks- document. Tijdens het onderzoek heb ik verschillende prototypes gemaakt die caching uitwerkt op een microservice architectuur.

Verder begrijp ik de theorie achter het hosten in de cloud en gedeeltelijk het verleggen van de GDPR-verantwoordelijkheden. Ook begrijp ik de theorie over het omgaan met gevoelige gegevens van gebruikers. Voor mijn stage heb ik de wetgeving "FDA 21 CFR Part 11 Subpart B, Conformiteit" toegepast voor het betreffende bedrijf. Deze wetgeving is de Amerikaanse versie, maar staat "gedeeltelijk/soortgelijk aan de GDPR-wetgeving. Ook zijn deze topics behandeld bij de specialisatie die ik heb gevolgd in Cyber Security en de ethiek achter gevoelige data.

### 2.8.2 Eind beoordeling / reflectie

Met het onderzoeken en het toepassen van de hierboven beschreven topics, oriënteer ik mijzelf als:

**"PROFICIENT".**

Ik geef mijzelf deze beoordeling omdat ik persoonlijk denk dat ik een gedare kennis heb over dit topic en daarmee mij verder kan ontwikkelen in de toekomst.

woordenlijst GDPR - De Algemene verordening gegevensbescherming is een Europese verordening die de regels voor de verwerking van persoonsgegevens door particuliere bedrijven en overheidsinstanties in de hele Europese Unie standaardiseert. [https://nl.wikipedia.org/wiki/Algemene verordening gegevensbescherming](https://nl.wikipedia.org/wiki/Algemene_verordening_gegevensbescherming)

Casestudy - Een casestudy is een gedetailleerde studie van een enkel onderzoeksobject.

## Hoofdstuk 3

# Retrospect

Net zoals heel veel andere studenten had ik soms moeite de huidige online studievorm vanuit thuis. Dit was ook merkbaar in de vele vergaderingen. Door deze tijdelijk online studievorm was er vanuit mijn kant soms sprake van een gebrek aan motivatie. De docenten hebben ondanks deze situatie geprobeerd mij maximaal te motiveren om het beste in mij naar boven te halen en dit is dankzij hun hulp dan ook gelukt. Daar ben ik dan ook zeer blij mee. Zodra we weer naar 'normaal' gaan hoop ik dat alle lesstof weer op locatie gegeven wordt. Dit is naar mijn mening voor zowel studenten (ikzelf) als voor docenten veel motiverender en daarnaast ook efficiënter. Ook was de leerstof zeer breed en soms ook wel veel/moeilijk voor iemand die nog niets op deze gebieden van de desbetreffende leerstof had gedaan. Dat deze stof online werd behandeld maakte het dan ook niet echt makkelijker. Het kostte me daarom ook enige tijd me te oriënteren in deze verschillende topics.

Omdat ik voor mijzelf niet een goed beeld had van dit semester/waar ik ergens stond, had ik graag meer feedback momenten gehad/willen inplannen. Het groepsproject nam in het begin veel tijd in beslag. De tijd had ik eigenlijk ook aan het persoonlijke project moeten besteden, Hierdoor had ik een achterstand van mijn persoonlijke project die ik in de vakantie heb moeten inhalen.



## Hoofdstuk 4

# Conclusie

Tijdens dit semester heb ik ontzettend veel nieuwe dingen geleerd. Dit op het gebied van software architectures, maar ook van deployment van software. Ik ben zeer tevreden hoe ik mij heb ontwikkeld en ik ben heel blij met de prestaties en de leeruitkomsten die ik heb aangetoond. Ik heb een mooi project opgezet die mijn technische kennis goed laat zien. Dit project kan ik in de toekomst dan ook gebruiken bij het zoeken naar een stage als verwijzing.

Ook had ik een zeer leuk en gemotiveerd project groepje. Ook hier ben ik zeer tevreden met de dingen die we hebben bereikt en ook de klant was zeer tevreden met de resultaten.

# Bijlagen

Add only appendixes that you actually reference in the document. There is no need to add pieces of work to the appendix if they can be found in Canvas.

## Bijlage A

# SWOT

# SWOT-analyse

Persoonlijke leerdoelen



*Autheur: Peeters, R.H.F. Rik*

*Datum: 01-03-2021*

*Versie: 1.1*

*Student aan Fontys University of Applied science*

# Inhoudsopgave

<b>1</b>	<b>Inleiding</b>	<b>1</b>
<b>2</b>	<b>SWOT-analyse</b>	<b>2</b>
2.1	Sterktes . . . . .	2
2.2	Zwaktes . . . . .	2
2.3	Kansen . . . . .	3
2.4	Bedreigingen . . . . .	3

# Hoofdstuk 1

## Inleiding

Dit document bevat de SWOT-analyse die ik heb verricht aan de hand van mijn individueel- en proftaak project. Hierin vertel ik over mijn zwakke maar ook sterke kanten en bespreken we de kansen/bedreigen die deze kanten kunnen vormen omtrent de projecten.

SWOT			
	kansen	bedreigingen	sterktes
			zwaktes

## Hoofdstuk 2

# SWOT-analyse

Een SWOT-analyse is een visueel hulpmiddel dat kan worden gebruikt om specifieke sterktes en zwaktes te identificeren in zakelijke en persoonlijke situaties. Het helpt met besluitvorming en vooruitplannen. Een SWOT-diagram wordt gevormd door een raster van twee bij twee. Elk kwadrant bevat een beschrijving van de sterktes, zwaktes, kansen en bedreigingen van het subject (SWOT staat voor strengths, weaknesses, opportunities en threats). In dit geval ben ik het subject.

### 2.1 Sterktes

In dit hoofdstuk ga ik vertellen wat mijn sterke kanten zijn.

Op het gebied van persoonlijke ontwikkeling:

Ik ben zeer leergierig aangelegd en wil bij complexe problemen altijd de beste oplossing zoeken.

Ik ben een doorzetter, ik wil hoe dan ook dat een project slaagt.

Op gebied van het project:

Flowcontrol is een uitdagend en groot, modulair project met enorm veel ruimte voor uitbreiding.

Het project is zeer modulair en heeft veel verschillende modules. Hierdoor kan ik mij verder blijven ontwikkelen.

Het project biedt de mogelijkheid elk leerdoel aan te raken.

Op het gebied van teamverband:

Ik kan goed werken in teamverband. Binnen deze samenwerking streef ik altijd naar een duidelijke communicatie. Ook trek ik aan de bel als zaken onduidelijk zijn zodat we deze kunnen ophelderen.

Het ontwikkelen van software in teamverband vind ik erg leuk. Ook vind ik het belangrijk te leren van elkaars fouten en positieve punten.

### 2.2 Zwaktes

In dit hoofdstuk ga ik vertellen wat mijn zwakkere kanten zijn.

Op persoonlijk gebied:



Ik kan mij soms te veel verdiepen in een onderwerp. Dit neemt vaak veel tijd in beslag en die heb ik eigenlijk niet.

Uit eerdere ervaringen is gebleken dat ik door het thuiswerken zeer ongestructureerd kan werken.

Op het gebied van het project:

Wanneer een project of een bepaalde zaak niet duidelijk is omschreven en ik hier vervolgens verder onderzoek naar doe/na vraag over doe en dit niets opheldert, irriteert mij dit ontzettend.

Op het gebied van teamverband:

Ook kan ik heel erg vasthouden aan standpunten die ik heb. Ik ben moeilijk te overtuigen als de tegenpartij geen goede argumenten geeft .

Ik kan me ontzettend storen aan profzaak groepjes met 'ongeïnteresseerde' individuen.

## 2.3 Kansen

In dit hoofdstuk ga ik uitleggen waar kansen voor mij liggen binnen mijn project.

Het toepassen van documentatie op een professionele manier die nuttig wordt geacht.

Het team motiveren en aansporen om hun taken tot een goed einde brengen en dat ik mij hierbij ga proberen open te stellen voor andermans bevindingen.

Omdat ik vorig semester ongestructureerd te werk ben gegaan wil ik het dit semester anders gaan aanpakken, door middel van een strak schema.

Ik zie het onderzoek als een kans om mijzelf in de toekomst voor te bereiden voor een levenslang leren.

## 2.4 Bedreigingen

In dit hoofdstuk ga ik uitleggen wat eventuele bedreigingen kunnen zijn tijdens het ontwikkelen van mijn individueel project. In het verleden ben ik tegen een aantal dingen aangelopen die ik nu als een bedreiging kan zien voor het ontwikkelen van het project deze zijn als volgt:

Kennis te laat opdoen die nodig is om mijn leerdoelen aan te tonen.

Wanneer het profzaak-project meer tijd vraagt dan ik verwacht had en ik mij daarop moet verleggen.

Het (online) werken in teamverband en de communicatie is door corona soms nog lastig.

Wanneer ik te gefocust ben op een bepaald probleem verlies ik het zicht over het hele plaatje.

Door mijn dyslexie vind ik het documenteren en daarmee het schrijven van onderzoeken ontzettend lastig en tijdrovend.

Ik ga proberen de dingen die hierboven zijn omschreven te verbeteren in dit semester zodat ik het beste eruit kan halen.

## Bijlage B

# Context based research

nty/global//global/global      nobblfile  
?? **PythonTeX** ??