

Projectplan

Updateproces virtuele machines automatiseren

Dennis Catharina Johannes Kuijs

20 Maart 2025

Inhoudsopgave

1. Projectopdracht	3
1.1. Context	3
1.1.1. Bedrijf	3
1.1.2. Probleemstelling	3
1.1.3. Oplossing	4
1.2. Doel van het project	4
1.3. De opdracht	4
1.4. Scope	4
1.5. Randvoorwaarden	5
1.6. Onderzoeksvragen	5
1.6.1. Hoofdvraag	5
1.6.2. Deelvragen	5
1.7. Eindproducten	6
1.7.1. Projectplan	6
1.7.2. Analysedocument	6
1.7.3. Onderzoeksdocumenten	6
1.7.4. Architectuurdocument	7
1.7.5. Realisatiedocument	7
1.7.6. Adviesdocument	7
1.7.7. Handleidingen en instructies	7
1.7.8. Overdrachtsdocument	7
1.7.9. Technische producten	7
2. Aanpak en Planning	8
2.1. Aanpak	8
2.1.1. Testaanpak	9
2.2. Onderzoeksmethoden	9
2.2.1. Literature Study	9
2.2.2. Best good and bad practices	9
2.2.3. Non-functional tests	9
2.2.4. System tests	9
2.2.5. Prototyping	10
2.2.6. Code review	10
2.2.7. Decomposition	10
2.2.8. Available product analysis	10
2.2.9. Gap Analysis	10

2.3. Leeruitkomsten	11
2.3.1. Analysis	11
2.3.2. Advise	11
2.3.3. Design	11
2.3.4. Realise	11
2.3.5. Manage and Control	11
2.3.6. Personal Leadership	11
2.3.7. Professional Standard	12
2.4. Opdeling van het project	12
2.5. Sprint doelen	13
2.6. Backlog	14
2.7. Tijdplan	14
3. Projectorganisatie	16
3.1. Teamleden	16
3.2. Communicatie	16
3.3. Testomgeving en benodigdheden	17
4. Financiën en Risico's	18
4.1. Kostenbudget	18
4.2. Risico's en uitwijkactiviteiten	18
5. Overig	20

1. Projectopdracht

1.1. Context

1.1.1. Bedrijf

OpenRemote, opgericht in 2015, is een organisatie die een Open-Source IoT platform ontwikkelt. Het doel van het platform is om de integratie van verschillende communicatieprotocollen en databronnen te vereenvoudigen in één gebruiksvriendelijk systeem.

Het platform maakt integratie mogelijk met verschillende soorten sensoren, databronnen, IoT apparaten, en zorgt ervoor dat deze centraal beheerd kunnen worden via een gebruiksvriendelijk managementportaal. De data van deze bronnen kan vervolgens gevisualiseerd worden en worden ingezet in op maat gemaakte applicaties en (mobiele) apps. De software is volledig Open-Source, waardoor iedereen deze gratis kan gebruiken of eraan bijdragen. Momenteel wordt OpenRemote toegepast in diverse branches, zoals het beheer en monitoring van energiesystemen, crowd management en wagenparken.

1.1.2. Probleemstelling

Naast het Open-Source product kunnen bedrijven en (overheids)organisaties ervoor kiezen de software (tegen betaling) te laten beheren en onderhouden door OpenRemote. Deze zogenaamde ‘managed’ service wordt steeds populairder en dat brengt nieuwe uitdagingen met zich mee. Een van de belangrijkste uitdagingen is het onderhouden van de virtuele machines waarop de software van OpenRemote draait.

Het komt regelmatig voor dat deze machines geüpdatet moeten worden, variërend van updates aan het besturingssysteem tot situaties waarbij meerdere (software)pakketten op de virtuele machine moeten worden bijgewerkt. In de meeste gevallen kunnen deze updates zonder problemen en met minimale downtime worden uitgevoerd, waarbij de virtuele machine intact blijft. Echter, wanneer er te veel wijzigingen in het AWS `CloudFormation` bestand zijn aangebracht, zal Amazon de virtuele machine opnieuw opbouwen, waardoor alle data verloren gaat.

Omdat het updaten een risicovolle actie is en het vooraf niet te voorspellen is wanneer Amazon voor deze ingrijpende maatregel kiest, wordt het proces voor iedere klant en virtuele machine momenteel handmatig uitgevoerd via het managementportaal van Amazon, in plaats van via de `CLI` of de `CI/CD` workflow.

Voorafgaand aan het updateproces worden verschillende stappen uitgevoerd, waaronder het maken van een `snapshot` (back-up) van de virtuele machine. Deze extra back-up, naast de automatische

dagelijkse back-ups, zorgt ervoor dat de klantdata beschermd blijft en niet verloren gaat. Na het updateproces wordt de klantdata vervolgens handmatig op de juiste manier hersteld, zodat de klant de software zonder onderbrekingen kan blijven gebruiken.

Door het handmatig uitvoeren van deze taken is het proces tijdrovend, kwetsbaar en foutgevoelig. Naarmate het aantal klanten voor de ‘managed’ service toeneemt, zal het bijwerken steeds meer tijd vergen van het teamlid dat verantwoordelijk is voor deze taak.

1.1.3. Oplossing

OpenRemote is op zoek naar een manier om het updateproces voor de virtuele machines op een schaalbare, veilige en vooral betrouwbare manier te automatiseren. De oplossing moet in staat zijn om op meerdere virtuele machines tegelijk te worden toegepast zonder verdere aanpassingen, waarbij de veiligheid van de virtuele machines gewaarborgd blijft en de klantdata te allen tijde behouden blijft en niet wordt gewijzigd.

Daarnaast moet de oplossing naadloos integreren met bestaande tools en platformen, zoals het Cloudplatform van `Amazon (AWS)`, waarop de virtuele machines draaien, en de `CI/CD` workflow op `GitHub` Actions, die wordt gebruikt voor het opzetten van klantomgevingen op dit Cloudplatform.

1.2. Doel van het project

Het doel van het project is om te onderzoeken hoe het updateproces van de virtuele machines geautomatiseerd kan worden op schaalbare, veilig en vooral betrouwbare manier.

1.3. De opdracht

Onderzoek welke mogelijkheden er zijn om het updateproces van de virtuele machines te automatiseren op een schaalbare, veilig en vooral betrouwbare manier en ontwikkel een `POC` (Proof Of Concept) waarin de mogelijke oplossing(en) worden getest op een ontwikkelomgeving.

1.4. Scope

Tot het project behoort:

Het onderzoeken van de mogelijkheden om het updateproces op een schaalbare, veilige en betrouwbare manier te automatiseren.

Het ontwikkelen van een **POC** (Proof Of Concept) waarin de mogelijke oplossing(en) worden getest in een ontwikkelomgeving.

Tot het project behoort niet:

Het ontwikkelen van functionaliteiten die buiten het updateproces vallen.

Het doorvoeren van wijzigingen aan de bestaande infrastructuur van klanten.

Het implementeren van de **POC** (Proof Of Concept) in een productieomgeving.

1.5. Randvoorwaarden

• De oplossing moet schaalbaar zijn, wat betekent dat deze op meerdere virtuele machines kan worden toegepast zonder verdere aanpassingen. • De oplossing moet veilig zijn, waarbij de veiligheid van de virtuele machines gewaarborgd blijft en niet negatief wordt beïnvloed. • De oplossing moet betrouwbaar zijn, wat inhoudt dat gegarandeerd moet kunnen worden dat de klantdata niet verloren gaat of wordt gewijzigd. • De oplossing moet werken met bestaande tools en platformen, wat betekent dat deze naadloos geïntegreerd moet zijn met het Cloudplatform van Amazon **(AWS)** en de **CI/CD** workflow op **GitHub** Actions.

1.6. Onderzoeksvragen

1.6.1. Hoofdvraag

Hoe kunnen we het updateproces van de virtuele machines op een schaalbare, veilige en betrouwbare manier automatiseren?

1.6.2. Deelvragen

1. Hoe zorgen we ervoor dat de klantdata niet verloren gaat of wordt gewijzigd? Kunnen we de data loskoppelen van de virtuele machines? Bijvoorbeeld door het tijdelijk op te slaan op een aparte volume? Of zijn er nog andere mogelijkheden?
2. Welke controles kunnen we inbouwen om er zeker van te zijn dat de klantdata voorafgaand aan het updateproces is veiliggesteld?

3. Hoe kunnen we de virtuele machines automatisch bijwerken? En hoe zorgen we ervoor dat we enkel de onderdelen bijwerken die noodzakelijk zijn?
4. Hoe kunnen we klanten het beste informeren over de voortgang van het updateproces?
5. Hoe zorgen we ervoor dat klanten de software kunnen blijven gebruiken tijdens het updateproces?
6. Hoe kunnen we het updateproces monitoren? Hoe kunnen we bijvoorbeeld zien welke virtuele machines worden bijgewerkt, en wat de status hiervan is? En wat doen we op moment dat een (fatale) fout optreedt tijdens het bijwerken?
7. Welke mogelijkheden zijn er binnen het Cloudplatform van Amazon om het proces te automatiseren? Welke kosten zijn hieraan verbonden? En kunnen deze mogelijkheden geïntegreerd worden met de huidige `CI/CD` workflow.
8. Hoe kunnen we de oplossing integreren in onze huidige `CI/CD` workflow? Welke aanpassingen moeten we hiervoor maken en wat is de verwachte impact?

In paragraaf 2.2. wordt toegelicht welke onderzoeksmethoden worden gebruikt voor het beantwoorden van de bovenstaande deelvragen.

1.7. Eindproducten

1.7.1. Projectplan

In dit document wordt het initiële plan van de opdracht uiteengezet. Het bepaalt welke onderdelen gerealiseerd en onderzocht worden, evenals de voorwaarden waaraan dit moet voldoen.

1.7.2. Analysedocument

Dit document bevat de probleemanalyse, opgesteld met de SPA-methode. Op basis van de probleemstelling zijn requirements en user stories geformuleerd volgens de `MoSCoW` methode.

1.7.3. Onderzoeksdocumenten

De resultaten van de verschillende onderzoeksvragen, zoals vastgesteld in het projectplan, zijn te vinden in de bijbehorende onderzoeksdocumenten. Om het overzicht te behouden, wordt elk onderzoek (deelvraag) in een apart document uitgewerkt.

1.7.4. Architectuurdokument

De architectuur van de beoogde oplossing, die wordt gerealiseerd op het Cloudplatform van `Amazon (AWS)` en in de `CI/CD` workflow op `GitHub` Actions is in dit document opgenomen. De werking wordt aan de hand van verschillende diagrammen toegelicht.

1.7.5. Realisatiedokument

De werking van de `poc` (Proof Of Concept) wordt in dit document toegelicht met behulp van relevante code-snippets en diagrammen.

1.7.6. Adviesdocument

De (mogelijke) vervolgstappen voor het project zijn in dit document opgenomen. Dit omvat vragen zoals ‘Wat moet er nog gebeuren?’ en ‘Waar moet nog over nagedacht worden?’ enzovoort.

1.7.7. Handleidingen en instructies

In dit document worden de instructies voor het gebruik en de configuratie van de oplossing uitgewerkt.

1.7.8. Overdrachtdokument

Dit document bevat belangrijke informatie voor een succesvolle overdracht van het project aan de opdrachtgever en stakeholders. Het omvat onder andere een samenvatting van de technische werking en configuratie van de oplossing.

1.7.9. Technische producten

Alle (technische) producten die tijdens dit project wordt ontwikkeld, zullen worden opgeleverd in een `GitHub` repository.

2. Aanpak en Planning

2.1. Aanpak

Voor dit project werk ik in `2-weekse` sprints volgens de `Agile SCRUM-methode`. Als onderdeel van het SCRUM-team neem ik deel aan stand-ups, (sprint)opleveringen en retrospectives. De opleveringen worden individueel ingepland in overleg met stakeholders en collega's en maken geen direct deel uit van het SCRUM-team.

Het bijhouden van taken gebeurt via `GitHub` Projects, gebaseerd op de `GitHub` Flow-methode. Dit houdt in dat voor elke bug, feature, epic, enzovoort, een `issue` wordt aangemaakt in de `GitHub` Repository. Zodra een `issue` is afgerond, wordt er een PR (`pull-request`) ingediend om de wijzigingen samen te voegen (mergen) met de `master` branch. Een collega zal altijd de PR (`pull-request`) reviewen (`code-review`) voordat deze wordt samengevoegd.

In eerste instantie houd ik zelf een `GitHub` Repository bij voor mijn issues met betrekking tot onderzoeken en `POC's` (Proof Of Concepts). Zodra ik de oplossing(en) ga testen op een ontwikkelomgeving van OpenRemote, zal deze werkwijze worden toegepast op de `GitHub` Repository van OpenRemote.

Om een goed overzicht te behouden van de hoeveelheid werk en de tijd, zal ik aan elke issue een deadline koppelen. Vooraf maak ik een inschatting van de benodigde tijd om de issue te voltooien en houd ik dit nauwlettend bij. Elke dag begin ik met het bekijken van het bord om te controleren of ik nog op schema lig.

Sommige `issues` worden aangemaakt als een zogenoemde `spike`, bijvoorbeeld bij onderzoeken, waarvan de oplossing nog onzeker is en er nog geen inzicht is in de mogelijke risico's. Verder worden alle issues geprioriteerd op basis van de MoSCoW-methode en kunnen gedurende het project worden aangepast op basis van nieuwe inzichten of wijzigingen.

Iedere werkdag begint om 09:00 uur op het kantoor in Eindhoven. Om 10:00 uur is er de dagelijkse stand-up, waarin wij als stagairs de gelegenheid krijgen om te delen waar we momenteel mee bezig zijn, wat we gisteren hebben gedaan en of we nog tegen problemen aanlopen of ergens hulp bij nodig hebben.

Om de twee weken (op maandag) wordt er een gesprek gepland met de stagebegeleider, Pierre Kil, om de voortgang van de stage te bespreken. Gesprekken over de inhoudelijke opdracht en techniek kunnen naar behoefte worden ingepland met collega's (Wouter Born en Richard Turner) en stakeholders (Don Willems en Pierre Kil)

2.1.1. Testaanpak

Voor het testen van de oplossing worden non-functionele en systeem tests gebruikt. Het project richt zich op de Cloud van Amazon (`AWS`) en de `CI/CD` workflow op `GitHub` Actions. In dit geval zijn integratietests het meest geschikt om de oplossing te valideren voordat deze wordt uitgerold naar de productieomgevingen van klanten.

2.2. Onderzoeksmethoden

2.2.1. Literature Study

Er wordt informatie opgezocht over een onderwerp en met behulp van een prototype kan worden gevalideerd of de opgezochte informatie bruikbaar is voor het project.

Deze onderzoeksmethode zal bij deelvragen 1-2-3-5-6-7 en 8 worden toegepast

2.2.2. Best good and bad practices

In de IT zijn vaak verschillende oplossingen die hetzelfde doel kunnen bereiken. Toch zijn niet alle oplossingen slim om te gebruiken, ook al lossen ze het probleem op. Met name op Cloud platformen wordt er veel aandacht besteed aan de ‘best practices’ die aangeven welke zaken belangrijk zijn bij het oplossen van een probleem.

Deze onderzoeksmethode zal bij deelvragen 1-2-5-6-7-8 worden toegepast.

2.2.3. Non-functional tests

Om de schaalbaarheid en beveiliging te testen, worden non-functionele tests uitgevoerd.

Deze onderzoeksmethode zal bij deelvragen 1-2-3-5-6-7-8 worden toegepast.

2.2.4. System tests

Voordat de oplossing in een productieomgeving kan worden ingezet, worden verschillende tests uitgevoerd om te controleren of het systeem op de juiste manier functioneert.

Deze onderzoeksmethode zal bij deelvragen 1-2-3-5-6-7-8 worden toegepast.

2.2.5. Prototyping

Om (mogelijke) oplossingen te valideren, wordt er soms een prototype ontwikkeld waarmee het onderzochte onderdeel in de praktijk kan worden getest. Op deze manier kan worden vastgesteld of de gekozen oplossing het gewenste resultaat heeft.

Deze onderzoeksmethode zal bij deelvragen 1-2-3-5-6-7-8 worden toegepast.

2.2.6. Code review

Gedurende het project zullen er, indien nodig, ‘code-reviews’ worden uitgevoerd met de stagebegeleiders. Dit zal voornamelijk gebeuren voordat de code wordt geïntegreerd in de bestaande codebase.

Deze onderzoeksmethode zal bij deelvragen 1-2-3-5-6-7-8 worden toegepast.

2.2.7. Decomposition

Om een IT-systeem of probleem beter te begrijpen, kan het nuttig zijn om het op te delen in kleinere onderdelen. Hiervoor worden vaak diagrammen en modellen gebruikt die het systeem en de onderlinge relaties visueel weergeven

Deze onderzoeksmethode zal bij deelvragen 1 en 8 worden toegepast

2.2.8. Available product analysis

Met deze methode wordt onderzocht of er bestaande oplossingen zijn die het probleem (of een gedeelte daarvan) kunnen oplossen. Er wordt onderzocht of het slim is om de oplossing zelf te (her)schrijven of dat de bestaande oplossing geïmplementeerd kan worden.

Deze onderzoeksmethode zal bij deelvragen 6 en 7 worden toegepast

2.2.9. Gap Analysis

Deze methode wordt gebruikt om het verschil tussen de huidige en de gewenste situatie te identificeren. De huidige situatie wordt in kaart gebracht, en er wordt onderzocht wat de nieuwe situatie zou kunnen zijn. Vervolgens worden de stappen bepaald die nodig zijn om het gat (gap) te overbruggen en de gewenste oplossing te bereiken.

Deze onderzoeksmethode zal bij deelvragen 1-4-8 worden toegepast.

2.3. Leeruitkomsten

De leeruitkomsten worden tijdens het project aangetoond met de onderstaande beroepsproducten.

2.3.1. Analysis

- Analysedocument
- Onderzoeken

2.3.2. Advise

- Adviesdocument
- Onderzoeken

2.3.3. Design

- Architectuurdocument
- Diagrammen en modellen

2.3.4. Realise

- Realisatiedocument
- Technische producten

2.3.5. Manage and Control

- Projectplan
- Sprint rapportages
- Handleidingen en instructies
- Overdrachtdocument

2.3.6. Personal Leadership

- Communicatie met collega's, stakeholders en andere belanghebbenden
- Feedback
- (Sprint)opleveringen

- Presentaties
- Retrospectives
- Logboek

2.3.7. Professional Standard

- Documentatie
- (Integratie)tests
- (SCRUM)bord

2.4. Opdeling van het project

Periode	Activiteit
10 - 15 februari	Kennismaking met bedrijf + Start probleemanalyse/projectplan
17 - 21 februari	Afronding probleemanalyse/projectplan
24 februari – 10 maart	Sprint 1
21 maart	Inleveren portfolio en projectplan
10 maart – 24 maart	Sprint 2
24 maart	1e bedrijfsbezoek
24 maart – 7 april	Sprint 3
7 april – 21 april	Sprint 4
22 april	Tussentijdse beoordeling
22 april – 5 mei	Sprint 5
5 mei – 19 mei	Sprint 6
19 mei – 2 juni	Sprint 7
3 juni	Inleveren conceptversie portfolio
2 juni – 16 juni	Sprint 8
17 juni	Inleveren definitieve versie portfolio
23 – 27 juni	2e bedrijfsbezoek
16 juni – 30 juni	Sprint 9
2 – 9 juli	Eindpresentatie

Periode	Activiteit
30 juni – 11 juli	Overdracht/voorbereiding presentatie (Sprint 10)
11 juli	Laatste stagedag
14 – 18 juli	Diploma-uitreiking

2.5. Sprint doelen

Sprint	Inhoud
Sprint 1	Projectplan + Probleemanalyse + Start 1e onderzoeksvraag
Sprint 2	Afronden 1e onderzoeksvraag (Prototype)
Sprint 3	Implementatie 1e onderzoeksvraag + Start 2e onderzoeksvraag
Sprint 4	Afronden/implementatie 2e onderzoeksvraag (Prototype) + Start 3e onderzoeksvraag
Sprint 5	Afronden/implementatie 3e onderzoeksvraag + Start 4e onderzoeksvraag
Sprint 6	Afronden/implementatie 4e onderzoeksvraag + Start 5e onderzoeksvraag
Sprint 7	Afronden/implementatie 5e onderzoeksvraag + Start 6e onderzoeksvraag
Sprint 8	Afronden/implementatie 6e onderzoeksvraag + Start 7e onderzoeksvraag
Sprint 9	Afronden/implementatie 7e onderzoeksvraag + Start 8e onderzoeksvraag
Sprint 10	Afronden/implementatie 8e onderzoeksvraag + Overdracht en voorbereiding presentatie

Belangrijk om te weten:

Er wordt bij OpenRemote gewerkt in sprints van 2 weken met behulp van een **Agile** werkwijze. Veel onderzoeksvragen hebben overlapping met elkaar en zullen wellicht al (deels) beantwoord worden in een andere onderzoeksvraag (of tijdens implementatie) Nadat het afronden van een onderzoek wordt deze met het team besproken en eventuele feedback verwerkt. Bij een positief signaal kan de onderzochte oplossing geïntegreerd worden in de bestaande codebase. Integratie vindt meestal direct na het onderzoek plaats.

De onderzoeksresultaten kunnen ook tot nieuwe inzichten leiden waardoor nieuwe onderzoeksvragen ontstaan, die mogelijk huidige onderzoeksvragen vervangen. Daardoor zijn bovenstaande onderzoeksvragen slechts een leidraad.

2.6. Backlog

Omdat we bij OpenRemote werken met behulp van een **Agile** werkwijze is de backlog uitgewerkt tot en met 20 maart 2025.

Backlog items ontstaan namelijk op basis van onderzoeksresultaten en zijn vooraf nog niet (altijd) bekend. In de eerste weken van de stage heb ik gewerkt aan de 1e onderzoeksvraag, deze is inmiddels afgerond waardoor er taken (issues) voor de implementatie zijn ontstaan die de komende weken worden uitgevoerd.

Dit proces zal voor elke deelvraag worden herhaald (iteraties)

- Onderzoek de mogelijkheden voor het opslaan van de (IoT-) data op een externe **EBS** (Elastic Block Service) volume
 - Voeg functionaliteit toe aan de bestaande **CI/CD** workflow voor het automatisch aanmaken/mounten van de nieuwe **EBS** volume.
 - Ontwikkel een script voor het loskoppelen van de nieuwe **EBS** volume, dit script moet opgeslagen worden bij Amazon **SSM** (Amazon Systems Manager)
 - Voeg functionaliteit toe aan de bestaande **CI/CD** workflow waarmee metrics en alarms van de nieuwe **EBS** volume zichtbaar worden bij de virtuele machine.
 - Voeg functionaliteit toe aan de bestaande **CI/CD** workflow waarmee een **policy** wordt aangemaakt voor het geautomatiseerd maken van snapshots (backups) van de nieuwe **EBS** volume via **DLM** (Amazon Data Lifecycle Manager).
- Onderzoek welke controles er ingebouwd kunnen worden om er zeker van te zijn dat de IoT data is veiliggesteld voorafgaand aan een update.
- Onderzoek de manieren om de virtuele machines automatisch te kunnen bijwerken waarbij alleen de onderdelen die een update nodig hebben worden bijgewerkt.
- Onderzoek de manieren om klanten te informeren wanneer een virtuele machine wordt bijgewerkt.
- Onderzoek de manieren hoe we ervoor kunnen zorgen dat klanten tijdens een update de software kunnen blijven gebruiken (**blue/green deployments**)
- Onderzoek de manieren hoe we het updateproces van de virtuele machines kunnen monitoren.
- Onderzoek de manieren hoe we het updateproces kunnen integreren in de bestaande **CI/CD** workflow.

2.7. Tijdplan

Fasering	Start	Gereed	Beschikbare dagen
Probleemanalyse en Projectplan	10-02-2025	21-02-2025	10 dagen
Start sprint 1	24-02-2025	10-03-2025	10 dagen
Sprint 1 oplevering	10-03-2025	10-03-2025	n.v.t.
Start sprint 2	10-03-2025	24-03-2025	10 dagen
Inleveren Portfolio en Projectplan	21-03-2025	21-03-2025	n.v.t.
Sprint 2 oplevering	24-03-2025	24-03-2025	n.v.t.
Start sprint 3	24-03-2025	07-04-2025	10 dagen
Sprint 3 oplevering	07-04-2025	07-04-2025	n.v.t.
Start sprint 4	07-04-2025	21-04-2025	9 dagen
Sprint 4 oplevering	22-04-2024	22-04-2025	n.v.t.
Tussentijdse beoordeling	22-04-2025	22-04-2025	n.v.t.
Start sprint 5	22-04-2025	05-05-2025	9 dagen
Sprint 5 oplevering	05-05-2025	05-05-2025	n.v.t.
Start sprint 6	05-05-2025	19-05-2025	10 dagen
Sprint 6 oplevering	19-05-2025	19-05-2025	n.v.t.
Start sprint 7	19-05-2025	02-06-2025	9 dagen
Sprint 7 oplevering	02-06-2025	02-06-2025	n.v.t.
Start sprint 8	02-06-2025	16-06-2025	9 dagen
Inleveren conceptversie portfolio	03-06-2025	03-06-2025	n.v.t.
Sprint 8 oplevering	16-06-2025	16-06-2025	n.v.t.
Start sprint 9	16-06-2025	30-06-2025	10 dagen
Inleveren definitieve versie portfolio	17-06-2025	17-06-2025	n.v.t.
Sprint 9 oplevering	30-06-2025	30-06-2025	n.v.t.
Start sprint 10 (Afronding)	30-06-2025	11-07-2025	10 dagen
Sprint 10 oplevering	11-07-2025	11-07-2025	n.v.t.

3. Projectorganisatie

3.1. Teamleden

Naam	Afk. Rol	Taken	Beschikbaarheid
Dennis Kuijs	DK Stagiair	- Onderzoekt mogelijke oplossingen voor de probleemstelling - Ontwikkelt een POC op een ontwikkelomgeving	Op kantoor maandag t/m vrijdag tussen 09:00 en 18:00
Pierre Kil	PK Stagebegeleider	- Begeleidt stagiair wekelijks - Beoordeelt stagiair - Monitort voortgang - Geeft feedback op documenten	Op kantoor maandag t/m vrijdag tussen 09:00 en 18:00
Wouter Born	WB Technische begeleider	Helpt bij technische vragen - Controleert werk - Adviseert over technische keuzes	Op kantoor maandag t/m vrijdag tussen 09:00 en 18:00
Don Willems	DW Stakeholder	Helpt bij vragen over opdracht - Adviseert en informeert over opdracht - Geeft feedback op documenten	Op kantoor op maandag, woensdag en vrijdag tussen 09:00 en 18:00. Overige dagen via e-mail en Slack
Richard Turner	RT Technische begeleider	Helpt bij technische vragen - Controleert werk - Adviseert over technische keuzes en opdracht	Bereikbaar via e-mail en Slack op maandag t/m woensdag tussen 09:00 en 18:00
Gertjan Schouten	GJS 1e assessor	- Begeleidt stagiair wekelijks - Beoordeelt stagiair - Monitort voortgang - Geeft feedback op documenten	Wekelijks contact via e-mail en logboek in Portflow. Overige contactmomenten op afspraak
John Lara Rojas	JLR 2e assessor	- Beoordeelt stagiair	Geen direct contact noodzakelijk

3.2. Communicatie

De communicatie met de stagebegeleider, stakeholder(s) en andere belanghebbenden zal plaatsvinden tijdens kantoor tijden (maandag tot en met vrijdag van 09:00 tot 18:00 uur). Er wordt vijf dagen per week gewerkt vanuit het kantoor (De ApparatenFabriek) in Eindhoven, gelegen op Strijp-S. De meeste

collega's zijn dagelijks op kantoor aanwezig. Als dit niet het geval is, kan er gecommuniceerd worden via Slack of de e-mail.

3.3. Testomgeving en benodigdheden

Voor de testomgeving wordt de bestaande `CI/CD` workflow op `GitHub` Actions gebruikt. Zodra een `pull-request` is goedgekeurd door één of meerdere collega's, wordt de workflow automatisch gestart, inclusief de bijbehorende tests.

4. Financiën en Risico's

4.1. Kostenbudget

Voor het project is geen specifiek kostenbudget vastgesteld. Alle kosten die redelijkerwijs noodzakelijk zijn voor de uitvoering van het project kunnen ter goedkeuring worden voorgelegd aan de stagebegeleider.

4.2. Risico's en uitwijkactiviteiten

Risico	Activiteiten ter voorkoming opgenomen in plan	Uitwijkactiviteiten
Stagebegeleider valt weg door omstandigheden/ziekte	Op zoek naar een vervangende stagebegeleider voor het geval dit nodig zou zijn.	Een andere medewerker binnen de organisatie zal de werkzaamheden overnemen.
Computerapparatuur stopt met functioneren	Er is een reservecomputer beschikbaar die kan worden ingezet bij storingen of onderbrekingen.	De reservecomputer wordt gebruikt bij storingen aan het hoofdapparaat.
Bestanden raken verloren of onherstelbaar beschadigd	Elke dag wordt een back-up gemaakt van alle bestanden die tijdens het project worden aangemaakt of aangepast, deze back-up wordt vervolgens opgeslagen op een externe locatie.	Wanneer een onherstelbare fout optreedt met één van de bestanden, wordt de laatste back-up teruggezet op de doelcomputer.
Er zijn per ongeluk uitzonderlijk hoge kosten gemaakt op het Cloudplatform van Amazon.	Er worden budgetalerts ingesteld om de kosten te monitoren en de gebruiker te waarschuwen wanneer het ingestelde limiet wordt overschreden. Daarnaast wordt voor elke gebruikte service vooraf een kostenberekening gemaakt om inzicht te geven in de verwachte (en mogelijke) kosten.	Als er onverwachts hoge kosten ontstaan, wordt geprobeerd deze te laten kwijtschelden met behulp van Amazon Support.

Risico	Activiteiten ter voorkoming opgenomen in plan	Uitwijkactiviteiten
Er is door een derde partij ongeoorloofd toegang verkregen tot het Amazon Cloudaccount.	Er wordt een apart Cloudaccount aangemaakt met specifieke toegangsrechten om ernstig misbruik te voorkomen. Daarnaast worden wachtwoorden veilig opgeslagen in een beveiligde wachtwoordenkluis en wordt, waar mogelijk, gebruik gemaakt van <code>SSH</code> sleutels voor extra veiligheid.	Als er sprake is van ongeoorloofde toegang, wordt het Cloudaccount onmiddellijk gedeactiveerd en opnieuw aangemaakt met gewijzigde inloggegevens.

5. Overig

Niet van toepassing, geen bijzonderheden.