

Project Plan

Foodies

Info Support

Datum	:	15/2
Versie	:	2
Status	:	Afgerond
Auteur	:	Zie versie historie

Versie historie

Versie	Datum	Auteur(s)	Veranderingen	Status
v1	15-02-2022	Justin van de Laar, Julian Janssen, Thisoban Mahalingam, Francois Haan, Bas Onrust, David Hellinga	Project plan opgezet en uitgewerkt.	afgerond
v2	21-02-2022	Zie bovenstaand	PBS toegevoegd, c4 model gemaakt, formatting bijgewerkt, risico's toegevoegd, basis pipelines opgezet	afgerond
v3	07-03-2022	Zie bovenstaand	PBS aangepast,	
v4	29-03-2022		Redenen voor technische keuzes	

Distributie

Versie	Datum	Ontvangers

Contents

Project opdracht	4
Context	4
Doel van het project	4
Scope and precondities	4
Strategie	4
Onderzoeksvragen en methodologie	5
Eind producten	6
Architectuur	7
Technische keuzes	8
Restaurant App	8
Admin App	8
Project organisatie	9
Stakeholders en teamgenoten	9
Communicatie	9
Activiteiten en planning	10
Fases van het project	10
Planning en doelstellingen	10
Strategisch testen en configurerend management	11
Strategisch Testen	11
Test omgeving en vereiste bronnen	11
Configuratie management	11
Financien en risico's	12
Risico en Mitigatie	12

1. Project opdracht

1.1 Context

Info Support is een consultancy bedrijf dat ook software pakketten oplevert aan andere bedrijven. Info Support heeft ons gevraagd om een applicatie te maken voor restaurantketens. Het moet een applicatie worden voor een restaurant via een bestelsysteem. Een klant moet kunnen bestellen via een tablet. De kok krijgt via een scherm door wat een klant van welke tafel aan eten heeft besteld. Het barpersoneel krijgt de bestellingen voor de drankjes door via de bestelapplicatie. Zij kunnen aangeven op een scherm waarmee het personeel bezig mee is. Tot slot is er een scherm voor de beheerder om aanpassingen op te doen bijvoorbeeld het menu van wat klanten kunnen bestellen.

1.2 Doel van het project

Het doel van dit project is om een bestelling systeem te maken voor Info Support. De reden dat ze dit nodig hebben is omdat ze in de nabije toekomst een restaurant willen beginnen, waarvoor ze dit bestelling systeem willen toepassen. De gewenste situatie is dat we op het eind van het semester een volledig werkend bestelling systeem hebben.

De voordelen van dit project zijn dat het systeem voor beide de medewerkers en de klanten van het restaurant veel duidelijkheid brengt. Het is namelijk een overzichtelijke manier om te bestellen voor klanten. Medewerkers weten wat zij naar de klanten moeten brengen qua gerechten en dranken.

De mogelijkheden die dit project oplevert is een websocket systeem dat aan de keuken en de bar aangeeft wat de klanten bestellen zonder dat ze hiervoor hoeven te herladen. Aangezien de applicatie op real-time werkt zal het voor de betrokkenen altijd duidelijk zijn wat zij kunnen verwachten of moeten uitvoeren.

1.3 Scope and preconditions

Inside scope:	Outside scope:
Digitaal Menu en Bestelsysteem	Betalingssysteem implementeren
Work Tracking Systeem	Hosting
Admin Dashboard	
Centrale backend	
Data storage	
Bestaande betalingssystemen koppelen	

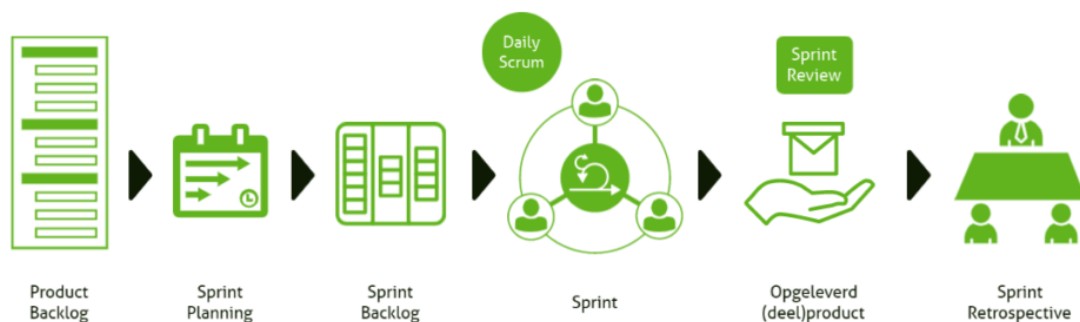
1.4 Strategie

Wij hebben besloten om de SCRUM methodiek te gebruiken. Alle methoden behalve Scrum hebben issues waardoor ze niet voor dit project geschikt zijn. Hieronder zal worden toegelicht waarom andere agile methodes niet geschikt zijn voor ons project.

Kanban en Lean werken niet in sprints maar met continue verbetering waarbij op elk moment het belangrijkste iets wordt opgepakt - vanuit Fontys wordt verwacht dat wij in sprints van drie weken werken, dus is dit voor ons geen optie. XP vereist dat de klant deel is van het ontwikkelteam en op elk moment aanspreekbaar is - dit is bij zulke opdrachten voor Fontys niet mogelijk of gewenst. Crystal is gericht op grootschalige teams en niet buitengewoon geschikt voor een enkel klein team van zes man.”

Bron: ‘Agile Onderzoek’ hoofdstuk ‘Welke Agile methode is geschikt voor het project?’ op pagina 20.

Wat is Scrum?



leansixsigmagroep.nl



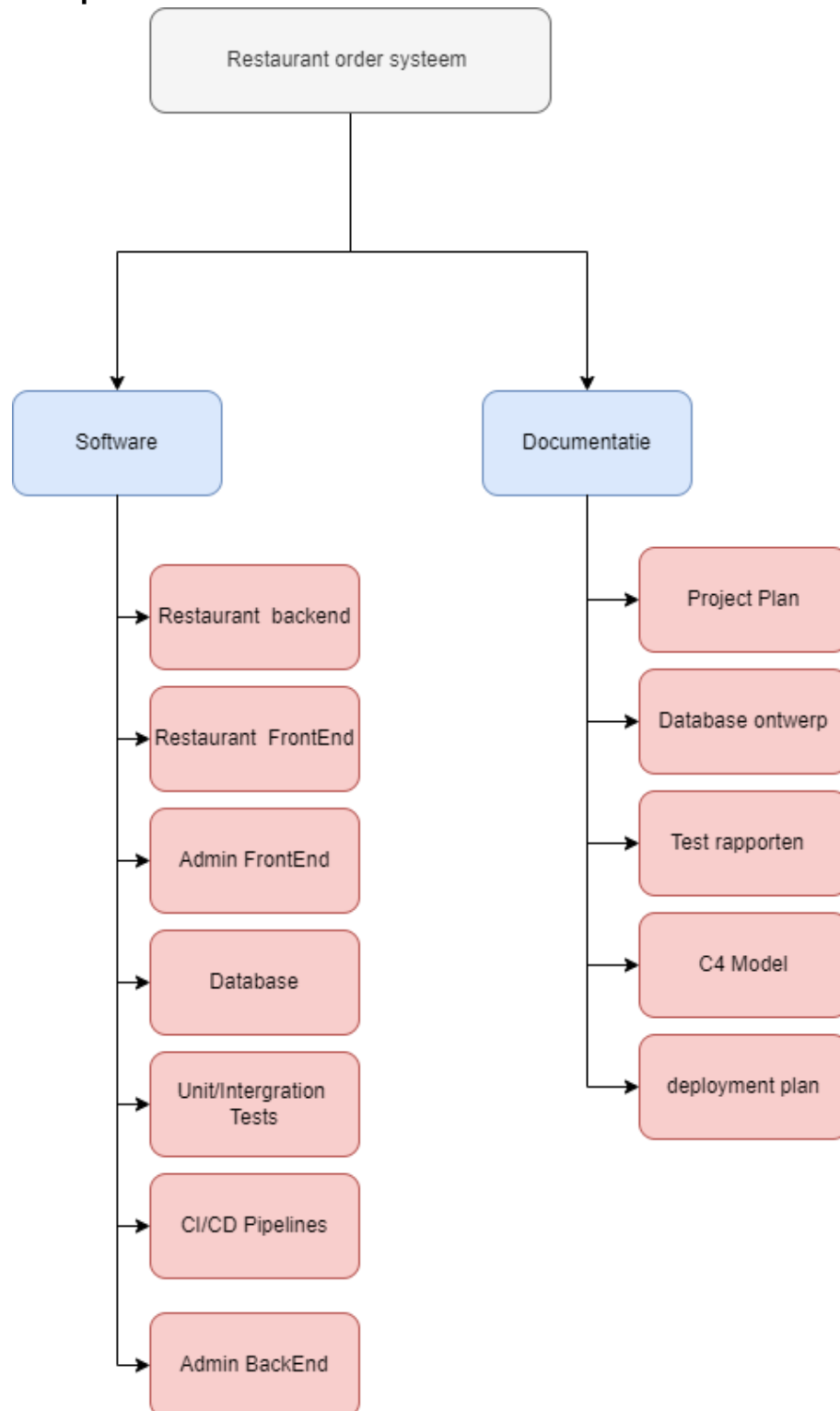
1

1.5 Onderzoeksvragen en methodologie

Not decided yet.

¹ "Wat is Scrum? Hoe werkt het? - Lean Six Sigma Groep."
<https://leansixsigmagroep.nl/lean-agile-en-six-sigma/scrum/>. Geopend op 15 feb.. 2022.

1.6 Eind producten



Hierboven ziet u het Product Breakdown Structure. Het PBS werkt als volgt:

Je begint bovenaan hier heb je het eindproduct. In ons geval is het ordersysteem voor het restaurant. Bij het eindresultaat komen een aantal producten kijken. Dit hebben we onderverdeeld na feedback van onze docent in de volgende twee categorieën. Deze categorieën zijn gekleurd in het blauw in de bovenstaande afbeelding.

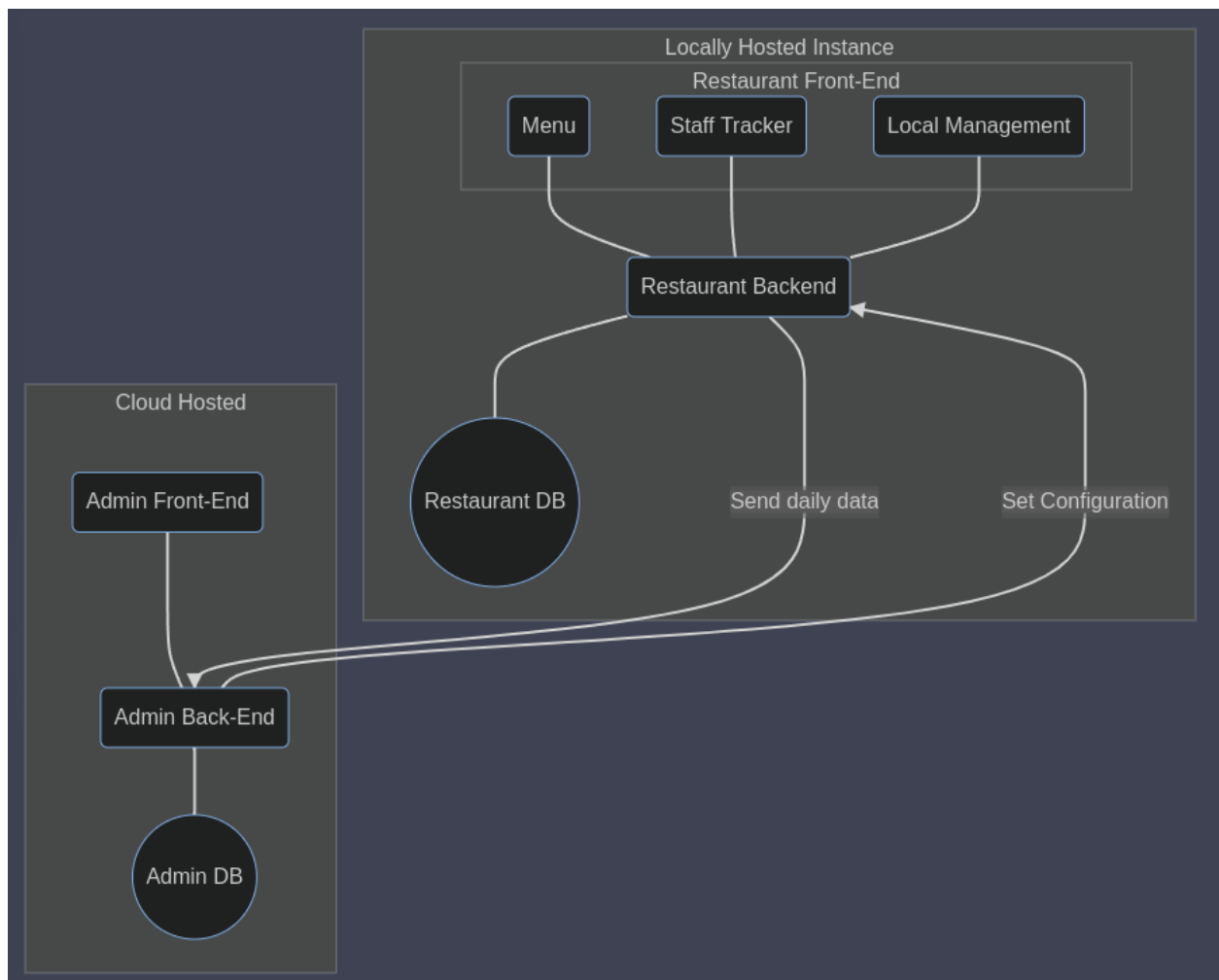
Onder deze subcategorieën komen een aantal producten die we aan het eind opleveren. De verschillende soorten subproducten staan in de rode vakken. Voor documentatie kan je denken aan de volgende soort producten:

1. Het project plan
2. Het database ontwerp
3. Het C4 model

Voor het kopje software is hetzelfde van toepassing als bij documentatie. Aan de volgende producten zou je kunnen denken:

1. FrontEnd voor de Admins
2. FrontEnd voor het Restaurant
3. BackEnd voor de Admins
4. BackEnd voor het Restaurant

1.7 Architectuur



1.8 Technische keuzes

Voor software projecten moeten aan het begin - en in de loop van het project - keuzes worden gemaakt over welke technologieën en talen gebruikt worden. In dit hoofdstuk zullen we de tot nu toe gemaakte keuzes uitleggen.

Restaurant App

De restaurant app is het eerste deel van het project waar de meeste “core” functionaliteit in zal zitten. Hierdoor hebben we gekozen voor technologieën waar we al een basis kennis van hebben zodat we een goede basis kunnen leggen.

1. .NET Core 6

C# / .NET Core was de backend taal waar de meeste van ons bekend mee waren. Daarnaast wilden we gebruik maken van oa. EF Core en Identity.

2. Vue.js + Typescript

Vue was het frontend framework waar de meeste van ons bekend mee waren. Typescript is gekozen om strong typing te hebben - iets dat basis javascript niet ondersteunt. Dit maakt de code leesbaarder en minder foutgevoelig.

3. SQLite voor ontwikkeling

SQLite is een lightweight database die geen eigen server nodig heeft. Het draait op een bestand dat in het project zelf kan worden aangemaakt. Voor productie is het niet geschikt - de functionaliteit en performance en veiligheid zijn niet optimaal daarvoor - maar voor de vroege stadia van ontwikkelen biedt het alle nodige functies. In de loop van het project zal het dus ook vervangen moeten worden met een andere technologie.

4. PostgreSQL als vervanger van SQLite

Bij het kiezen van een database oplossing moeten een paar vragen beantwoord worden. De eerste is of een SQL, NoSQL of ander type database het meest geschikt is. Hierbij hebben we gekozen voor SQL omdat het de eerdere keuzes het beste ondersteunt. We willen in onze .NET applicatie gebruik maken van EF Core - het ORM van .NET Core - om met de database te werken (zie hierboven waarom). Dit vereist het gebruik van een SQL database waarmee onze keuze daarover vaststaat. De volgende vraag is welke SQL database we gebruiken. Gezien dit een studieproject is zijn alleen oplossingen met een redelijke market share relevant: Oracle, MySQL, SQL Server en PostgreSQL. Oracle valt af omdat het erg gericht is op grootschalige enterprise projecten. SQL Server is ook erg gelimiteerd voor gratis gebruik en vereist bij opschalen al snel een prijzige licentie. Dit geeft dus twee opties: MySQL of PostgreSQL. Beide bieden de nodige functionaliteit, zijn voor onze doeleinden volledig gratis bruikbaar en zouden dus een acceptabele keuze zijn. We hebben hierbij gekozen om voor PostgreSQL te gaan omdat die open source is.

Admin App

De admin app wordt een losstaande en redelijk minimale applicatie met focus op functionaliteit. Dit geeft ons de kans om technologieën te kiezen waar we minder bekend mee zijn zodat we daar ervaring mee op kunnen doen.

Java + Spring

Java is naast C# een van de meest wijd verspreide backend talen voor web API's en Spring is een van de meest gebruikte Java frameworks. Daarom hebben we gekozen om voor de admin app deze combinatie te proberen.

Angular

Naast Vue.js zijn er twee andere grote javascript frameworks: React en Angular. We hebben voor Angular gekozen omdat die het meeste lijkt op backend talen zoals C# en Java en we willen testen hoe dat bevalt als frontend framework.

2. Project organisatie

2.1 Stakeholders en teamgenoten

Naam	Afkorting	Rol en functies	Beschikbaarheid
Thisoban Mahalingam	<i>Thiso</i>	<i>Developer</i>	<i>Maandag en dinsdag 09:00 tot 16:00</i>
Justin van de Laar	<i>Justin</i>	<i>Developer</i>	<i>Maandag en dinsdag 09:00 tot 16:00</i>
Julian Janssen	<i>Julian</i>	<i>Developer</i>	<i>Maandag en dinsdag 09:00 tot 16:00</i>
Francois Haan	<i>Francois</i>	<i>Developer</i>	<i>Maandag en dinsdag 09:00 tot 16:00</i>
Bas Onrust	<i>Bas</i>	<i>Developer</i>	<i>Maandag en dinsdag 09:00 tot 16:00</i>
David Hellinga	<i>David</i>	<i>Contactpersoon</i>	<i>Maandag en dinsdag 09:00 tot 16:00</i>
Samuil Angelov	<i>Samuil</i>	<i>Project begeleider</i>	<i>Maandag 09:00 - 12:00 en 13:00 - 16:00</i> <i>Woensdag 09:00 - 12:00</i>
Leon Bokhorst	<i>Leon</i>	<i>Individueel begeleider</i>	<i>Woensdag 09:00 - 12:00</i> <i>Donderdag 09:00 - 12:00 en 13:00 - 16:00</i>
Jeroen Uittenbosch	<i>Jeroen</i>	<i>Product owner</i>	
Sander Klijsen	<i>Sander</i>	<i>Product owner</i>	

2.2 Communicatie

Communicatie gaat via een persoon in de groep. Dit is om de communicatie soepel te laten verlopen via een lijn. Hiermee willen we voorkomen dat iedereen de product owner mailt. Dit voorkomt ook dat er verwarring ontstaat over het project omdat, persoon A iets anders heeft gekregen dan persoon C aan informatie van de product owner.

David Hellinga is de contactpersoon van de groep. Tijdens de kennismaking met Info Support hebben we besloten contact te houden via de mail. We hebben afgesproken met Info Support dat we de mogelijkheid krijgen om feedback te vragen via de mail over bepaalde producten.

Sprints worden gedaan om de 3 weken. Na iedere 3 weken worden een sprint opgeleverd aan de product owner. De oplevering vindt plaats op school of als het is afgesproken bij de product owner op kantoor.

Als we over het team spreken dan spreken we over de groep Foodies

Verder communiceert het team via microsoft teams. Mocht het voorkomen dat op de dag dat we naar school gaan dat een teamlid te laat komt is er ook nog een whatsapp groep aangemaakt. Hier kan het teamlid dan melden waarom de persoon te laat. Als team hebben we de afspraak dat er minimaal een keer per 24u op teams worden gekeken. Dit met uitzondering van het weekend. Hiermee willen voorkomen dat er berichten ongezien blijven bij mensen van het team.

3. Activiteiten en planning

3.1 Fases van het project

De hoofd fases binnen het project zijn de planning en implementatie fases waar we ervoor zorgen dat we in de volgende sprint kunnen beginnen met user stories te verwerken. Dan hebben we de development phase waar alle user stories verwerkt worden en we meer en meer van de applicatie gaan opleveren. Tenslotte hebben we dan nog een afronding/ test fase waar we de laatste dingen toevoegen er ervoor zorgen dat alles goed werkt.

3.2 Planning en doelstellingen

Fases	Start datum	Eind datum
1 Planning en implementatie fase (spring 0)	14/2/2022	7/3/2022
2 Sprint 1	7/3/2022	28/3/2022
3 Sprint 2	28/3/2022	19/4/2022
4 Sprint 3	19/4/2022	16/5/2022
5 Sprint 4	16/5/2022	7/6/2022
6 Afronding en test fase	7/6/2022	20/6/2022

4. Strategisch testen en configurerend management

4.1 Strategisch Testen

Voor het testen van de back-end gaan we unit tests gebruiken. Met de unit tests kunnen we losse stukken code onafhankelijk controleren op eventuele bugs. Wanneer deze tests succesvol zijn uitgevoerd wordt via SonarCloud nogmaals over de code heen gekeken. Dit externe software hulpmiddel zorgt ervoor dat duplicated code, coding standards, unit tests, code coverage en security recommendations worden gedetecteerd of weergegeven. Zowel de unit tests als SonarCloud werken automatisch wanneer zij eenmaal zijn opgezet.²

Voor het controleren van endpoints en responses tussen de back-end en de front-end maken we gebruik van het programma Postman. Ook kan het worden gebruikt voor het debuggen van de API's en het bouwen van onze applicatie op een relatief eenvoudige manier. Kortom het is een handige tool voor het creëren, delen, testen en documenteren van RESTful API's.

De Front-end word getest met selenium test om te checken of de buttons werken en of er een element veranderd. Om te Controleren of het werkende elementen en componenten zijn. Die voldoen aan de eisen.

4.2 Test omgeving en vereiste bronnen

Voor de continuous integration, afgekort CI, willen we Azure DevOps gebruiken. Tijdens het pushen van de code via Github kan via deze CI pipelines worden bijgehouden welke testen slagen en welke niet. Het idee erachter is dat deze testen automatisch worden gedraaid wanneer Github een verandering in de code ontvangt. Voor de continues deployment, afgekort CD, gedeelte willen we het via Docker gaan gebruiken. We plaatsen de backend, frontend en de database ieder als aparte image op Docker.

4.3 Configuratie management

Voor de configuratie management gebruikt ons team Github (<https://github.com/s3gp-foodies>) We hebben drie repos voor de drie softwarecomponenten van het project. Elke repo heeft een main branch waar de "productie" versie van de applicatie staat, een dev branch waar de meest recente stabiele versie op staat en een set feature branches waarop aan user stories wordt gewerkt. Directe commits naar de main en dev branches zijn niet mogelijk. Merge requests naar de dev worden na het halen van de CI tests, code review en acceptatie tests uitgevoerd. Wanneer een nieuwe "productie" versie klaar is wordt een merge request van dev naar main gemaakt. Deze moet ook aan dezelfde tests voldoen voordat het uit wordt gevoerd.

² "Wat is Postman? — SYSQA — Quality Assurance."

<https://sysqa.nl/veelgestelde-vragen-over-quality-assurance-in-it/veelgestelde-vragen-over-testautomatisering-en-test-tools/wat-is-postman/>. Geopend op 15 feb.. 2022.

5. Financien en risico's

5.1 Risico en Mitigatie

Risico	Preventie activiteiten	Mitigatie activiteiten
1 Groepslid is ziek	Eigen verantwoordelijkheid	Corona check doen voordat naar locatie komen.
2 Docent is ziek	Is aan de docent	Vragen voorbereiden voor als de docent weer beter is.
3 Treinen rijden niet	Op de NS site site kijken	Vanuit huis werken.
4 Treinen hebben vertraging	Op de NS site site kijken	Melden in WhatsApp groep
5 Groepslid doet minder of zijn werk niet	Stand up check en in contact blijven	Aanspreken erop en bij herhalend gedrag consequenties opleggen.