Nemesis

### Projectplan

Fontys HBO ICT

*Semester 7*

*Versie 1.0*

*Eindhoven 02-09-2020*

*J.M.G. (Remy) Witlox*

*Studentnummer: 3407462*

*J. (Joe) van Kollenburg*

*Studentnummer: 2857138*

*T.J. (Tim) Markus*

*Studentnummer: 2506963*

*R. (Rob) de Haan*

*Studentnummer: 383317*

Docent M.H.M. (Mark) Mestrom  
Docent W.R. (Wilrik) de Loose

## Wijzigingshistorie

|  |  |  |  |
| --- | --- | --- | --- |
| **Versie** | **Datum** | **Beschrijving** | **Auteur(s)** |
| **1.0** | 02-09-2020 | Initiële opzet van het document | Remy, Joe, Tim, Rob |

## Verspreiding

|  |  |  |  |
| --- | --- | --- | --- |
| **Versie** | **Datum** | **Aan** | **Media** |
| **1.0** | 14-09-2020 | Mark Mestrom en Wilrik de Loose | Canvas, [Azure DevOps](https://remyw.visualstudio.com/Nemesis/_wiki/wikis/Nemesis.wiki/1/Documentatie) |

## Definities, acroniemen en afkortingen

|  |  |
| --- | --- |
| **Term** | **Definitie** |
| **Backend** | Het onderdeel of onderdelen van een software applicatie waar transport, transformatie, manipulatie en persistentie van gegevens wordt verwerkt. |
| **Decomposition** | Een ICT onderzoeksmethode. |
| **Design pattern** | Een abstractie / sjabloon welke generiek wordt opgezet, waarmee veel voorkomende ontwerpproblemen kunnen worden opgelost. |
| **Framework (software)** | Een verzameling van softwarecomponenten (herbruikbare oplossingen en structuren) welke gebruikt kunnen worden bij het bouwen van software applicaties. |
| **Front-end** | Het onderdeel van een software applicatie waar de interface en visuele aspecten worden verzorgt. |
| **Klant** | Onderwijsinstelling die een overeenkomst heeft met Advitrae[7] voor het afnemen van het roosterprogramma ‘Xedule’[6]. |
| **Library**  **(software)** | Vergelijkbaar met een ‘Framework’ maar minimalistischer. Een library biedt een collectie van bestanden, routines en scripts waarnaar gerefereerd kan worden vanuit project code. |
| **POC** | **Proof of Concept.** Een ‘klein’ prototype om aan te tonen dat/hoe een bepaalde software oplossing werkt. |
| **SLA** | **Service Level Agreement.** Dit is een overeenkomst waarin afspraken staan tussen een een afnemer en de aanbieder van een dienst of product. Hierin staan o.a. de afgesproken prestatie indicatoren en kwaliteitseisen. |
| **Ticketsysteem** | Dit is een managementtool die service verzoeken van klanten verwerkt en categoriseert. |
| **Melding** | Een taak, een wens of een bug. |
| **Functioneel Beheerder** | Een persoon die aangewezen is vanuit de klant van Advitrae die taken, wensen en bugs in mag dienen in het ticketsysteem. |
| **Support Medewerker** | Een medewerker van Advitrae die de meldingen in behandeling neemt. |
| **Key-user** | Een persoon die aangewezen is vanuit de klant van Advitrae die bugs in mag dienen in het ticketsysteem. |
| **Gebruiker** | De medewerkers van de onderwijsinstellingen die meldingen mogen indienen. |
| **UX** | **User Experience.** Gebruikerservaring van van een applicatie. |

# 

# Inhoudsopgave

[**Voorblad**](#_heading=h.fh4ah512uiqw) **1**

[Wijzigingshistorie](#_heading=h.gqa17qokhnit) 1

[Verspreiding](#_heading=h.pm5bacasn6ka) 1

[Definities, acroniemen en afkortingen](#_heading=h.op41yjguxnm4) 1

[**Inhoudsopgave**](#_heading=h.uckt7z57q0bw) **3**

[**Het idee**](#_heading=h.1upk68vjbatd) **5**

[Oorsprong idee](#_heading=h.shwutyk1s31i) 5

[Het bedrijf](#_heading=h.6wozbrnchku5) 5

[Stakeholders](#_heading=h.f4voy830yqkl) 6

[Het Projectteam](#_heading=h.5l1x0td6fal5) 6

[**Requirements**](#_heading=h.9h0h1gl507wg) **7**

[Business Requirements](#_heading=h.ghiquhb7wdw6) 7

[User Requirements](#_heading=h.f2r6z7al1vee) 7

[**De Planning**](#_heading=h.xje02dymtmpd) **8**

[Onderzoek](#_heading=h.wbotusvqmhq9) 8

[Sprints](#_heading=h.e2zqwtc6s3h3) 8

[Leerdoelen](#_heading=h.sjj3xi610idw) 8

[**De Data**](#_heading=h.z1rscjdpnd04) **12**

[Entiteiten](#_heading=h.dwb1imq0ewhn) 12

[Attributen](#_heading=h.80tu6mn95j5l) 12

[**Kwaliteitsborging**](#_heading=h.8hbsyru9ovr) **13**

[Communicatie](#_heading=h.n29hk5nv1goc) 13

[Versiebeheer](#_heading=h.al86d8n5fy6) 13

[Risico’s en Maatregelen](#_heading=h.pp49y7el0pgr) 13

[**Bronnen**](#_heading=h.yczp3pnt59c8) **14**

# Het idee

## Oorsprong idee

Bij de huidige werkgever van Remy Witlox wordt een ticketsysteem (Zendesk[1]) gebruikt dat tot vele frustraties leidt. De frustraties komen vooral uit de traagheid van het systeem en de user experience. Vooral gebruikers die niet dagelijks in het ticketsysteem werken, maar hier wel meldingen moeten aanmaken vinden het lastig een goede melding aan te maken. De support medewerkers ervaren daarom ook dat meldingen onvolledig of incorrect worden ingediend en dit resulteert weer in een langere verwerkingstijd. Er is daarom vraag naar een nieuw ticketsysteem die beter ingericht is naar de wensen van Advitrae[7].

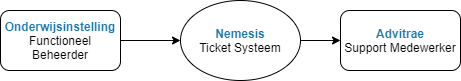
## Het bedrijf

Advitrae[7] is een onafhankelijke Nederlandse onderneming gevestigd in Eindhoven. Het doel van Advitrae[7] is om de logistieke uitdagingen die een onderwijsinstelling kent met werkende oplossingen te ondersteunen. Oplossingen die kunnen variëren van advies, training, software tot en met de ondersteuning bij de implementatie van deze software. Dit alles volledig gericht op de onderwijslogistiek, het plannen en roosteren van het door u gewenste onderwijs.

De software die Advitrae[7] aanbiedt is Xedule[6]. Xedule[6] is een plan- en roostersysteem voor het onderwijs. Wij gebruiken hiervoor de term onderwijslogistiek. Xedule[6] is configureerbaar en daardoor te gebruiken in het Hoger Onderwijs (HBO/WO), het Middelbaar BeroepsOnderwijs (MBO) en het Voortgezet Onderwijs (VO).

Binnen Advitrae[7] is er een support-afdeling, waar meldingen binnenkomen, behandeld worden en eventueel doorgestuurd worden naar andere afdelingen. Een melding kan bestaan uit een opmerking, wens, incident (bug), vraag en een taak. Naast de supportafdeling is er ook een product-owner die de wensen behandeld.

De klanten van Advitrae[7] zijn onderwijsinstellingen. De gebruikers van Xedule[6] zijn vooral roostermakers maar kunnen ook personen zijn met andere rollen binnen een onderwijsinstelling. Er zijn afspraken gemaakt met de onderwijsinstellingen over welke gebruikers meldingen mogen aanmaken. Dit zijn vooral de functioneel beheerders van de onderwijsinstellingen.



Tussen Advitrae[7] en de onderwijsinstellingen zijn bepaalde afspraken gemaakt in een Service Level Agreement (SLA). Deze SLA beschrijft onder andere binnen welk termijn bepaalde meldingen opgelost dienen te zijn. Tickets kunnen om deze reden ook een bepaalde prioriteit krijgen. Om na te gaan of de SLA nagekomen is dienen er analyses te worden gemaakt van de tickets die door een onderwijsinstelling ingediend zijn. Deze resultaten kan een onderwijsinstelling met de Service Manager van Advitrae[7] bespreken.

Meldingen die gemaakt zijn om Xedule[6] te verbeteren dienen te worden opgelost door een ontwikkelaar. De ontwikkelaars gebruiken Azure DevOps[5] om hun taken en bugs bij te houden. De support medewerker maakt zelf bugs en taken aan in DevOps voor de ontwikkelaar.

## Stakeholders

De tickets zullen behandeld worden door de support medewerkers van Advitrae[7]. Daarnaast zal een Service Manager de tickets moeten kunnen inzien en analyseren. Een beheerder van Advitrae[7] dient gebruikers te kunnen aanmaken en instellingen te kunnen aanpassen. De beheerder is de manager van de support afdeling en heeft ook de rol van support medewerker. Daarnaast is er een product owner vanuit Advitrae[7] die de wensen in het ticketsysteem dient te behandelen.

De tickets kunnen worden ingediend door de support medewerkers, adviseurs van Advitrae[7] en de functioneel beheerders van de onderwijsinstellingen (de klanten). Naast functioneel beheerder zijn er ook key users die meldingen kunnen indienen. Key Users kunnen echter geen taken of wensen indienen. Vanuit de onderwijsinstellingen is er ook een Contractmanager per onderwijsinstelling die de analyses van de meldingen moet kunnen inzien.

|  |  |  |
| --- | --- | --- |
| **Stakeholders** | **Primair** Heeft direct invloed op de aanpak of het resultaat | **Secundair**  Heer indirect invloed op de aanpak of het resultaat |
| **Interne stakeholder** Bij het project betrokken vanuit de eigen organisatie | * Support medewerkers * Manager Support * Project Opdrachtgevers * Projectteamleden * Product Owners | * Service Managers * Directie * Adviseurs * Ontwikkelaars |
| **Externe stakeholder** Bij het project betrokken vanuit een externe partij | * Functioneel Beheerders van klanten * Contractmanagers van klanten * Key Users van klanten | * Concurrenten * Overig personeel van klanten |
| **Interface stakeholder** Niet bij het project betrokken partij, die wel een legitiem belang heeft | * Familie * Universiteiten * Lokale overheid | * Maatschappij * Onderwijs |

## Teamleden

Het team is opgedeeld in de volgende deelnemers:

|  |  |  |
| --- | --- | --- |
| **Teamleden** | **Rol** | **Beschikbaarheid** |
| **R. (Rob) de Haan** | UX specialist | Dagelijks vanaf 18:00 uur Zaterdag |
| **T.J. (Tim) Markus** | Front-End developer | Dagelijks vanaf 18:00 uur Zaterdag + Zondag |
| **J. (Joe) van Kollenburg** | Front-End developer | Dagelijks vanaf 18:00 uur Zaterdag + Zondag |
| **J.M.G. (Remy) Witlox** | Backend developer | Dagelijks vanaf 18:00 uur Zaterdag + Zondag |

Tim en Joe zullen zich vooral richten op een onderzoek naar de mogelijkheden op het gebied van Front-End. Remy zal zich bezighouden met een onderzoek naar de mogelijkheden op het gebied van Back-End in de specifieke programmeertaal C#. Rob zal zich gaan richten op de UX (user experience) en zal daar verder onderzoek naar doen.

|  |  |
| --- | --- |
| **Afwezigheid** | **Afwezigheden** |
| **R. (Rob) de Haan** |  |
| **T.J. (Tim) Markus** | ‘Langere’ afwezigheid: 28-31 december |
| **J. (Joe) van Kollenburg** |  |
| **J.M.G. (Remy) Witlox** | 17-09, 25-09 |

# Requirements

## Business Requirements

1. De mogelijkheid voor functioneel beheerders om meldingen aan te maken, zodat aangeven waar het programma verbeterd dient te worden.

2. Een helder overzicht van de openstaande meldingen zodat de support medewerker goed ziet wat er nog moet gebeuren.

3. Het automatiseren voor het uit handen nemen van administratieve taken.

## User Requirements

1.1 Als support medewerkers wil ik kunnen zien welke meldingen nog open staan, zodat ik weet waar ik aan moet werken. [M]

1.2 Als klant wil ik meldingen kunnen indienen, zodat ik kan laten weten waar Xedule verbeterd kan worden. [M]

2.1 Als support medewerkers wil ik meldingen verschillende types meegeven zodat het overzichtelijk is wat iedere melding betreft. [S]

2.2 Als support medewerker wil ik meldingen verschillende statussen geven zodat overzichtelijk is wat er nog voor de melding gedaan moet worden. [S]

2.3 Als support medewerker wil ik diverse overzichten van groepjes meldingen. [S]

2.4 Als klant wil ik een overzicht van mijn meldingen zodat ik na kan gaan wat de status is per melding. [C]

2.5 Als medewerker wil ik de per ticket-type de velden van tickets kunnen toevoegen zodat ik naar eigen hand het ticket-formulier kan uitbreiden. [C]

2.6 Als support medewerker wil ik zelf nieuwe overzichten van meldingen kunnen maken. [C]

3.1 Als Service Manager wil ik een analytisch overzicht van tickets per onderwijsinstelling zodat ik deze kan bespreken. [S]

3.2 Als klant wil ik mijn gegeven kunnen aanpassen zodat ik ook zelf kan bijhouden wie meldingen mag indienen. [C]

3.3 Als ontwikkelaar wil ik dat gesloten bugs en taken in DevOps direct een notificatie naar de melding sturen en de melding klaarzetten om te sluiten. [C]

### Opmerkingen:

Voor 2.3 zal er een view komen voor alle openstaande meldingen, een view voor openstaande meldingen gekoppeld aan de ingelogde medewerker en een view voor gesloten meldingen. Er is reeds besloten dat de backend van het systeem in C# dient te worden geschreven.

### 

|  |  |
| --- | --- |
| **Tot het project behoort:** | **Tot het project behoort niet:** |
| C# backend | Mobiele applicatie |
| Frontend voor support medewerkers | Live hosting |
| Frontend voor gebruikers |  |
| Automatische build + test infrastructuur |  |

# Het Onderzoek

Onze gezamenlijke hoofdvraag is: Wat is de meest gebruiksvriendelijke manier om een ticketing systeem te presenteren en welk front-end framework of library is het meest geschikt om om dit te realiseren in combinatie met een in C# opgezette backend?

## Onderzoek

Voor dit project zal er onderzoek noodzakelijk zijn om te zorgen voor een optimaal resultaat. Allereerst dient er onderzocht te worden wat er momenteel al op de markt beschikbaar is. Dit wordt ook wel de ‘Available product analysis’ genoemd. Vanuit Advitrae[7] is er reeds aangegeven dat zij met Zendesk[1] werken en ook al eens eerder hebben gekeken naar de mogelijkheden van TOPdesk[2]. Beide sloten echter niet aan op hun specifieke wensen. Binnen de requirements is ook al reeds aangegeven welke wensen diverse stakeholders hebben. Er zal nog een uitgebreidere ‘Stakeholder analyse’ komen voor een nog duidelijker inzicht in de wensen van diverse stakeholders.

### BackEnd Onderzoeksvraag:

Hoe ontwikkel ik een backend in C# voor een ticketsysteem voor het eind van het semester?

|  |  |
| --- | --- |
| **Strategie** | **Deelvraag** |
|  | **Wat is er nodig om een C# backend te ontwikkelen?**  Library -> Available product analysis om inzicht te krijgen in de hulpmiddelen die ik kan gebruiken voor het schrijven van de code.  Library -> Community research om na te gaan hoe andere ontwikkelaars een C# backend hebben opgezet.  Workshop -> Prototyping om een concept op te zetten en direct de grootste risico’s te tackelen. |
|  | **Welke backend architectuur is geschikt voor een ticketsysteem?**  Library -> Design pattern research om na te gaan welke architectuur de kwaliteit en structuur van de software kan optimaliseren.  Field -> Domain modelling om inzichtelijk te krijgen hoe alle onderdelen aan elkaar gekoppeld kunnen worden.  Workshop -> IT architecture sketching voor het bespreken van de backend architectuur met de teamleden. |
| Thumbnail for version as of 18:05, 9 February 2018 | **Hoe zorg ik ervoor dat de backend schaalbaar is, waarbij schaalbaarheid wordt gedefinieerd als de mogelijkheid om nieuwe functionaliteiten toe te voegen?** Library -> Expert interview om advies in te winnen over de mogelijkheden en verwachtte valkuilen bij het schaalbaar maken van een backend.  Showroom -> Static program analysis om een indruk te krijgen van hoe herbruikbaar de code is geschreven.  Workshop -> Decomposition om het systeem in kleinere delen op te splitsen voor een goede schaalbaarheid. |
| Thumbnail for version as of 18:05, 9 February 2018 | **Hoe zorg ik ervoor dat de backend gestructureerd opgebouwd is?**  Showroom -> Guideline conformity analysis om te zorgen voor een gestructureerde aanpak in het schrijven van de code.  Lab -> Component test om er voor te zorgen dat systemen en componenten ook afzonderlijk naar verwachting functioneren. |

### Front-end Onderzoeksvraag:

Welke front-end framework is het meest geschikt voor het ontwikkelen van een efficiënt ticketsysteem?

|  |  |
| --- | --- |
| **Strategie** | **Deelvraag** |
|  | **Wat maakt een framework geschikt voor een ticketsysteem?**  Library -> Best good and bad practices.  Field -> Explore user requirements |
|  | **Wat maakt een framework efficiënt?**  Library -> Literature study  Library -> Competitive analysis |
|  | **Welke frameworks en libraries zijn er op dit moment?**  Library -> Available product analysis  Field -> Document analysis |
|  | **Welke relevante voordelen hebben bepaalde frameworks?**  Field -> Document analysis  Library -> Literature study  Lab -> Component test  Workshop -> Prototyping |
|  | **Welke frameworks worden gebruikt in reeds bestaande CRM/ticketing systemen?**  Library -> Available product analysis |

### 

### UX Onderzoeksvraag:

Welke UX designs zorgen voor een positieve ervaring voor gebruikers van een ticketsysteem?

|  |  |
| --- | --- |
| **Strategie** | **Deelvraag** |
|  | **Met welke methoden is te meten hoe positief een gebruikerservaring is?**  Library -> Available product analysis  Library -> Literature study  Workshop -> Prototyping |
|  | **Welke factoren worden als positief ervaren met betrekking tot gebruikerservaring?**  Library -> Literature study  Library -> Best good and bad practices  Field -> Interview  Lab -> A/B testing |
|  | **Hoe optimaliseer je de gebruikerservaring voor die factoren?**  Field -> Explore user requirements  Field -> Observation  Lab -> A/B testing  Lab -> Usability testing |

## Sprints

Voor dit project zal er gewerkt worden in Azure DevOps[5]. Hier worden user stories beschreven, taken uitgewerkt en toegekend. Via de sprints wordt er een planning bijgehouden. Per week heeft iedere deelnemer 20 uur de tijd om aan het project te werken. Ieder sprint bestaat uit 3 weken. Gezien het project een agile aanpak zal krijgen heeft dit project nog geen overal tijdsplan. Verder zal de tijdsplanning van het project via Azure DevOps worden ingevuld.

## Planning

Hieronder is grofweg beschreven wat er binnen de sprints opgepakt zal worden.

|  |  |
| --- | --- |
| **Sprint** | **Geplande onderdelen** |
| **Sprint 1** | Kennismaking, opzet projectplan |
| **Sprint 2** | Onderzoek, opzet integrale deel |
| **Sprint 3** | Onderzoeksresultaten bespreken, Implementatie |
| **Sprint 4** | Implementatie, Integratie, verder onderzoek |
| **Sprint 5** | Uitbreiding, Testen |
| **Sprint 6** | Afronding, laatste details uitwerken |

## Leerdoelen

Hieronder zal er per teamlid van het project worden aangegeven hoe de leerdoelen zullen worden aangetoond.

### Remy Witlox

### 

|  |  |
| --- | --- |
| **Leerdoel** | **Hoe?** |
| **Analyse** Ik kan aantonen dat ik een goede analyse kan maken door het uitvoeren van een requirements analyse voor een systeem met meerdere stakeholders. Ik zal acceptatiecriteria kunnen definiëren op basis van kwaliteitseigenschappen en een uitgevoerde risicoanalyse uitvoeren met onder meer aandacht voor veiligheidsaspecten. | Er zijn momenteel user requirements opgesteld op basis van de behoeftes en vereisten van de stakeholders. Er zal nog verder onderzoek uitgevoerd worden voor de backend analyse door een onderzoek naar *goede, slechte en beste praktijken*. Er zal ook een *designpatroon onderzoek* worden gedaan, zodat het duidelijk is welke architectuur het beste past bij dit project. De hoofdonderzoeksvraag zal kunnen worden beantwoord door de sub-onderzoeksvragen te beantwoorden. |
| **Advies** Ik kan aantonen dat ik een goed advies kan geven over de keuze van software architectuur of bestaande software frameworks waarbij kostenaspecten en kwaliteitseigenschappen als beschikbaarheid, performance, veiligheid en schaalbaarheid een rol spelen. Ik kan adviseren over de te volgen aanpak bij het verwerken en raadplegen van grote hoeveelheden data met aandacht voor privacy. Bovendien kan ik advies geven over de organisatie van een softwareontwikkelingsproces, inclusief het testproces. | Na het uitvoeren van het onderzoek zal er gedocumenteerd worden welke keuzes er worden gemaakt en waarom. Het advies zal bestaan uit advies voor een goede infrastructuur en de beste technieken om te gebruiken voor de backend van het systeem. Daarnaast zal er ook een advies wordt gedocumenteerd voor de opbouw van de data en beveiliging van de data. |
| **Design** Ik kan een software-architectuur samenstellen voor een software systeem, en ik kan rekening houden met de kwaliteitseigenschappen van verschillende stakeholders, waaronder beveiliging en schaalbaarheid. ik kan een teststrategie samenstellen voor systeemtesten. | Na het onderzoek en de analyse zal er een design worden gekozen voor de architectuur van de software. Er zal gedocumenteerd worden welke architectuur gekozen is en waarom deze het meest geschikt it. Ook zal er een teststrategie worden opgesteld. |
| **Implementatie** Ik kan en ​​schaalbaar software-systeem bouwen dat correleert met bestaande systemen, volgens de ontworpen architectuur met bestaande frameworks. Ik kan testautomatisering toepassen. | Op basis van het onderzoek zullen er keuzes worden gemaakt die daarna ook geïmplementeerd worden. Daarnaast zal er ook een teststrategie worden geïmplementeerd. Deze code zal later worden aangeleverd. |
| **Management** Ik kan configuratie-, wijzigings- en releasemanagement uitvoeren in combinatie met infrastructuurbeheer. Ik kan een ontwikkelomgeving organiseren met geautomatiseerde build- en testinfrastructuur. | Door scrum te werken zal er na iedere sprint opnieuw worden gekeken welke user stories worden opgepakt. Het project zal een geautomatiseerde build- en testinfrastructuur bevatten. Deze zal na overleg binnen het team worden opgezet. |

### Tim Markus

|  |  |
| --- | --- |
| **Leerdoel** | **Aantoonbaar resultaat** |
| **Analyseren** | Door overwogen keuzes te maken naar aanleiding van de onderzoeksvragen uit het onderzoek en het bouwen van de POC’s. Kan er bepaald worden welke front-end oplossing het meest geschikt is. Hierbij wordt rekening gehouden met de wensen van de stakeholders en de ‘bestaande’ systemen waar de front-end mee zal gaan integreren. **De resultaten van deze keuzes zullen worden vastgelegd in het onderzoek.** |
| **Adviseren** | Door de verschillende front-end frameworks te onderzoeken en tegen elkaar af te wegen middels een aantal POC’s kan er een weloverwogen advies worden gevormd naar de stakeholders. **Aan het einde van het front-end onderzoeksgedeelte van het onderzoeksdocument zal er een advies gegeven worden m.b.t. de beste keus voor een front-end framework.** |
| **Ontwerpen** | **In de vorm van POC’s worden er een aantal mogelijke architecturen opgezet om de front-end mee te kunnen bouwen**. Deze zullen aansluiten en communiceren met de backend welke in C# gebouwd zal zijn en zal moeten aansluiten op de gestelde wensen van de stakeholders. |
| **Implementeren** | De front-end POC’s zullen uiteindelijk gaan communiceren met het C# backend systeem. Deze zullen ieder hun eigen plekken krijgen binnen onze Github[4] omgeving. **Deze omgeving zal inzichtelijk worden gemaakt voor de docenten.** (Of d.m.v. screenshots worden toegevoegd aan documentatie) *(Nog vast te stellen: Plan automatisering testen.)* |
| **Managen** | Door gebruik van Github[4] is het mogelijk om eerdere iteraties van de POC projecten terug te halen en m.b.v. branches verschillende features en bugs losstaand te verwerken. *(Nog vast te stellen: Omgeving met automatische bouw en test infrastructuur).* |

### Joe van Kollenburg

### 

|  |  |
| --- | --- |
| **Leerdoel** | **Aantoonbaar resultaat** |
| **Analyseren** | Door gebruik te maken van het onderzoeksraamwerk, kan er onderzoek gedaan worden om deelvragen van de onderzoeksvragen te beantwoorden. Door te trianguleren kan een conclusie vast worden gesteld om op voort te bouwen. Hieruit zouden dan **POC’s** uitkomen samen met een **onderzoeksverslag**. |
| **Adviseren** | Na het analyseren van meerdere front-end frameworks kan een advies worden opgesteld voor het meest geschikte framework met onderbouwing. Hieruit komt een **adviesversverslag** waarom de gekozen framework het beste is. |
| **Ontwerpen** | Net zoals dat de back-end meerdere architecturen kan hebben, wil ik gaan onderzoeken of er meerdere front-end architecturen zijn en welke, waarvoor het beste geschikt is. Hieruit komt een **onderzoeksverslag** en zal de gekozen architectuur uitgewerkt worden in de implementatiefase |
| **Implementeren** | Voor de implementatiefase wordt de gekozen architectuur gebruikt in alle gekozen frameworks. Hieruit komen **POC’s** van de gekozen frameworks |
| **Management** | Door gebruik te maken van Github, kan een jenkins server opgezet worden die de frontend zelf bouwt en test. Afhankelijk van de relevantie kan hier een **POC** voor worden gemaakt |

### Rob de Haan

|  |  |
| --- | --- |
| **Leerdoel** | **Aantoonbaar resultaat** |
| **Analyseren** | Als eerste stap in mijn onderzoek ga ik analyseren welke tools en methoden veel toegepast worden door **UX** designers. Ook ga ik uitzoeken welke factoren een **UX** design “goed” maken. Deze bevindingen leg ik vast in een verslag. |
| **Ontwerpen** | Op basis van het onderzoek naar wat een **UX** design goed maakt en met behulp van de tools die ik gevonden heb in de analysefase ga ik meerdere (drie) **UX** designs maken. Deze werk ik uit met behulp van wireframes.  Ook houd ik er rekening mee dat de designs uiteindelijk tegen elkaar getest moeten worden en stel ik een strategie op om deze designs met elkaar te kunnen vergelijken. |
| **Implementeren** | Voor elk van de front-end frameworks ga ik samen met de rest van het front-end team een verschillend **UX** design implementeren. Ook wil ik tools gaan implementeren waardoor het mogelijk is om deze designs objectief te gaan testen tegenover elkaar. Daarvoor moet het mogelijk zijn om bijv. clicks van de gebruikers te registreren of de tijd te meten die gebruikers erover doen om bepaalde acties te volbrengen. |
| **Adviseren** | Op basis van mijn bevindingen in de analyse-, implementatie- en testfase zal ik advies uitbrengen over welk **UX** design het beste werkt in dit specifieke geval. Ook zal ik advies uitbrengen over welke technieken misschien handig kunnen zijn voor volgende projecten. |
| **Management** | Het opzetten van automatisch loggen van clicks en tijden tussen bepaalde button clicks om zo in kaart te brengen hoe lang ene gebruiker over een bepaald traject doet. |

# De Data

Voor dit project zal er een MsSQL[3] database worden opgezet.

## Entiteiten

Deze database dient de volgende entiteiten te bevatten: (eventueel kan laten nog besloten worden ‘Klant’ en ‘Medewerker’ toch samen te voegen tot een entiteit ‘gebruiker’).

## Attributen

Deze bovenstaande entiteiten zullen de volgende attributen bevatten:

### 

### Klant

klaId

klaNaam

klaMail

klaTelefoon

klaOpmerking

klaFunctie

klaAccount

### Organisatie

orgId

orgNaam

orgDomein

orgType

orgAfkorting

orgMedId

### Medewerker

medId

medNaam

medAfdeling

medTelefoon

medMail

medAccount

### Ticket

ticId

ticTitel

ticKlaId

ticMedId

ticOrgId

ticType

ticAfwachting -> medAfdeling

ticOpmerking

ticOverleg

ticPrioriteit

ticDatumAangemaakt

### Opmerking

opmId

opmTicId

opmMedId

opmDatum

opmOpenbaar

opmInhoud

# Kwaliteitsborging

## Communicatie

Documenten zullen worden gedeeld via Google Drive[8]. Via WhatsApp[9] is er een groep aangemaakt voor vragen en opmerkingen. Daarnaast zal er op Maandag en Donderdags vergaderd worden via Microsoft Teams[10]. Uitgevoerd werk zal opgeleverd worden via Github[4]. De planning zal bijgehouden via Microsoft Azure DevOps[5].

## Versiebeheer

Door het gebruik van Google Drive[8] voor de documentatie is het mogelijk om per document de geschiedenis van dat document na te gaan, waardoor je eventueel een eerdere versie kunt terugplaatsen. Ook kun je hier makkelijk zien wie bepaalde aanpassingen heeft uitgevoerd.

Voor de code hebben wij een project aangemaakt op Github[4], waar alle code zal worden opgeslagen. Hier kan iedere deelnemer bij en eventueel nieuwe branches aanmaken. Door diverse branches aan te maken zal er een mogelijkheid zijn een voorgaande versie te bekijken/bewerken. Ook geeft Github[4] een goede representatie van gemaakte aanpassingen en wie deze aanpassingen heeft gepushed.

Omdat er meerdere projecten zich zullen bevinden in de repository, is het handig om regels op te stellen met betrekking tot de naamgeving van de branches. Voor algemene branches die door meerdere projectleden gebruikt gaan worden, wordt de branchenaam in groepsoverleg bepaald. Voor het aanmaken van een gebruiker-specifieke branche wordt de volgende naamgeving gehanteerd:

**[Projectlid voornaam]-[Branch project]-[Optionele naam]**

## Risico’s en Maatregelen

|  |  |  |
| --- | --- | --- |
| **Risico** | **Hoe te voorkomen** | **Hoe op te lossen** |
| **Defecte laptop** | Goed onderhoud | Reserve laptop is beschikbaar bij Remy Witlox en zal worden ingeschakeld tot er een nieuwe laptop is geregeld. |
| **Teamlid ziek** | Niet te voorkomen | Leraren inschakelen voor advies + taken in de sprint verdelen onder andere deelnemers. |
| **Teamlid valt weg** | Niet te voorkomen | Leraren inschakelen voor advies + project eventueel verkleinen. |
| **Servers werkstation crasht** | Goed onderhoud | Troubleshooten of eventueel vervangen. |

# Bronnen

Onderstaande bronnen zijn gebruikt voor dit projectplan. In dit projectplan zijn cijfers gebruikt om te verwijzen naar onderstaande bronnen:

|  |  |  |
| --- | --- | --- |
| **Nummer** [x] | **Referentie** | **Website** |
| **[1]** | Zendesk | https://www.zendesk.nl/ |
| **[2]** | TOPdesk | https://www.topdesk.com/nl/ |
| **[3]** | MsSQL | https://www.microsoft.com/nl-nl/sql-server/sql-server-downloads |
| **[4]** | GitHub | https://www.github.com/ |
| **[5]** | Azure DevOps | https://www.azure.microsoft.com/nl-nl/services/devops/ |
| **[6]** | Xedule | https://www.advitrae.nl/xedule/ |
| **[7]** | Advitrae | https://www.advitrae.nl/ |
| **[8]** | Google Drive | http://www.google.nl/drive/apps.html |
| **[9]** | WhatsApp | https://www.whatsapp.com/?lang=nl |
| **[10]** | Microsoft Teams | https://www.microsoft.com/nl-nl/microsoft-365/microsoft-teams/group-chat-software |