Plan van aanpak

MongoExplorer

Ciyan Cöcelli 548189

Roel de Man 489962

Alex Post 531671

Niels Wijers 564429

Marijn Martin 547235

Marijn Martin 547235

ICT

Webdevelopment

14 november 2016

Versie 1.1

Hogeschool Arnhem Nijmegen

DWA –Sep

Inhoudsopgave

[1. Inleiding 3](#_Toc466626701)

[2. Context 4](#_Toc466626702)

[2.1. Bedrijfsachtergrond 4](#_Toc466626703)

[2.2. Afdeling 4](#_Toc466626704)

[2.3. Semester 4](#_Toc466626705)

[3. Aanleiding 5](#_Toc466626706)

[4. Doelstelling 6](#_Toc466626707)

[4.1 Resultaten 6](#_Toc466626708)

[4.1.1 Plan van aanpak 6](#_Toc466626709)

[4.1.2 Onderzoeksrapport 6](#_Toc466626710)

[4.1.3 Software guidebook 6](#_Toc466626711)

[4.1.4 De applicatie 6](#_Toc466626712)

[5. Projectgrenzen 7](#_Toc466626713)

[6. Randvoorwaarden 8](#_Toc466626714)

[6.1. Werkruimte 8](#_Toc466626715)

[6.2. Schoolritme 8](#_Toc466626716)

[6.3. Beschikbaarheid product-owner 8](#_Toc466626717)

[6.4. Software 8](#_Toc466626718)

[7. Producten en kwaliteit 9](#_Toc466626719)

[7.1. Plan van aanpak 9](#_Toc466626720)

[7.2 Onderzoeksplan 9](#_Toc466626721)

[7.2 Onderzoeksrapport 9](#_Toc466626722)

[7.3 Software guidebook 9](#_Toc466626723)

[7.4 Client side code 9](#_Toc466626724)

[7.5 Server side code 9](#_Toc466626725)

[7.6 Eindpresentatie 9](#_Toc466626726)

[8. Ontwikkelmethodes   10](#_Toc466626727)

[8.1. De watervalmethode 10](#_Toc466626728)

[8.2. Scrum 10](#_Toc466626729)

[8.3. Conclusie 11](#_Toc466626730)

[9. Technieken 12](#_Toc466626731)

[9.1 Ontwerpen 12](#_Toc466626732)

[9.2 Realisatie 12](#_Toc466626733)

[9.3 Testen 12](#_Toc466626734)

[10. Projectorganisatie en communicatie 13](#_Toc466626735)

[10.1 Taakverdeling en informatie 13](#_Toc466626736)

[11. Planning  14](#_Toc466626737)

[12. Risico’s   15](#_Toc466626738)

[12.1 Risicoschema 15](#_Toc466626739)

[13. Bibliografie 16](#_Toc466626740)

[14. Bijlagen 17](#_Toc466626741)

[14.1 Bijlage A – Globale planning, MongoExplorer 17](#_Toc466626742)

# 1. Inleiding

Dit document is geschreven voor het project ‘MongoExplorer’, dat hoort bij het project van Develop a Web Application (DWA) (07-11-2016 t/m 13-01-2016) bij de Hogeschool Arnhem en Nijmegen (HAN) te Arnhem.

De MongoExplorer is een webapplicatie die ervoor zorgt dat mensen zonder IT-kennis een document-oriented database kunnen inzien en kleine veranderingen kunnen aanbrengen.

De HAN wil graag zien dat mensen zonder bijzondere kennis van een document-oriented database de database kunnen aanpassen in een GUI(Graphical User Interface).

Het doel van dit document is om de lezer inzicht te geven in de opdracht die de komende acht weken uitgevoerd gaat worden. Het plan van aanpak bestaat uit een aantal globale onderdelen, deze worden hieronder beschreven.

Als eerste wordt er algemene informatie gegeven. Deze informatie beschrijft onder andere de probleemstelling en de daarbij behorende opdracht die de studenten van de HAN is toevertrouwd. Vervolgens worden de te gebruiken methoden en technieken toegelicht in combinatie met de op te leveren producten die volgens de normen van de methoden en technieken gemaakt zullen worden.

Er zal ook ingegaan worden op de kwaliteitseisen waaraan de op te leveren producten moeten voldoen. Dit plan van aanpak is ook voorzien van een stukje risicomanagement, waarbij mogelijke scenario’s worden uitgewerkt met bijbehorende oplossingen.

# 2. Context

De context beschrijft de bedrijfsachtergrond van het bedrijf. Verder beschrijft dit hoofdstuk op welke locatie het project plaatsvindt. Als laatste wordt het bijbehorende semester toegelicht.

## 2.1. Bedrijfsachtergrond

De Hogeschool Arnhem en Nijmegen (HAN) is opgericht in 1996 en behoort tot de grootste hogescholen van Nederland. De hogeschool bevat vijf faculteiten en deze zijn verdeeld over de campussen die gevestigd zijn in Arnhem en Nijmegen. Met ruim 2.500 medewerkers verzorgt de HAN meer dan 63 bachelor en 19 masteropleidingen voor ongeveer 31.000 studenten. (HAN, 2016)

## 2.2. Afdeling

Het project wordt uitgevoerd op de Hoge Technische School (HTS) in Arnhem op Ruitenberglaan 26. De projectgroep bestaat uit vijf ICT-studenten die studeren aan de Informatica Communicatie Academie (ICA). Om een indicatie te geven waar de studenten zich binnen de HAN bevinden wordt in afbeelding 1 een organogram weergegeven.

Afbeelding 1: Organogram HAN

## 2.3. Semester

Dit half jaar neemt deze projectgroep deel aan het semester DWA. In de eerste tien weken bevindt zich de coursefase. Dat houdt in dat er lessen web development worden gegeven met bijbehorende technieken zoals React/Redux, NodeJS, ExpressJS en MongoDB. De kennis die de studenten tot zich nemen wordt vervolgens in de laatste tien weken toegepast in de projectfase. Het project zorgt ervoor dat de student kan aantonen dat hij/zij competent is voor dit semester om vervolgens de bijbehorende studiepunten te halen.

# 3. Aanleiding

Tijdens de course van het semester DWA wordt er gebruik gemaakt van de applicatie RoboMongo. RoboMongo is ontwikkeld om NoSQL document-oriented databases van MongoDB overzichtelijk te hebben binnen één venster. Hier kunnen de databases, collections en documents door een paar simpele muisklikken in een overzichtelijke weergave weergegeven worden, zonder dat er queries geschreven hoeven te worden om resultaten te krijgen. Tevens is het mogelijk om gebruik te maken van de command-line om complexe queries uit te voeren. (Paralect, 2016)

Bij het gebruiken van een document-oriented database is de database voor de software-engineers goed te onderhouden, omdat zij de kennis bevatten om een database aan te passen door middel van de command line. Voor een persoon die deze kennis niet bevat is het nagenoeg onmogelijk om data aan te passen op deze manier.

RoboMongo is geschreven in C++. De opdrachtgever wil een webapplicatie die geïnspireerd is op RoboMongo, geschreven in JavaScript.

# 4. Doelstelling

De bedoeling is dat personen zonder kennis van databases de database kunnen inzien, en mocht er tijd over zijn binnen de acht weken, aanpassen. Ook beheerders die wel kennis hebben van databases willen wij een aangename gebruikservaring bieden met dit product. Dit doormiddel van een simpele GUI waarbij duidelijk is aangegeven waar welke waarde van een document aangepast kan worden. Het is de bedoeling dat de webapplicatie meerdere connecties kan leggen tussen de gebruiker en meerdere servers waar diegene toegang tot heeft. Per verbinding wordt er duidelijk gemaakt van welk type document-oriented database dit is (bijvoorbeeld MongoDB (MongoDB inc., Z.D.)) met hieronder alle onderliggende databases. Per database worden de onderliggende collecties en documenten weergegeven, deze kunnen geopend en bekeken worden.

In het geopende scherm wordt de inhoud weergegeven waar waardes (eventueel in een later stadium) kunnen worden aangepast en worden toegevoegd. Dit wordt in zijn geheel als webapplicatie ontwikkeld.

## 4.1 Resultaten

In dit kopje zal worden verteld wat er opgeleverd wordt aan het einde van het project.

### 4.1.1 Plan van aanpak

Dit document geeft inzicht over de aanpak van de huidige opdracht.

### 4.1.2 Onderzoeksrapport

Een document dat alle onderzoeken bevat die de projectgroep nodig heeft om het project te kunnen starten. Deze documenten zullen alle onderzoeksresultaten bevatten.

### 4.1.3 Software guidebook

Het software guidebook is een handleiding voor de applicatie. Hier zal worden uitgelegd hoe de applicatie te gebruiken is en wat de mogelijkheden zijn.

### 4.1.4 De applicatie

Een applicatie die voldoet aan de standaarden en requirements van de opdrachtgever.

# 5. Projectgrenzen

Het project begint op 14 november 2016 en duurt in totaal 8 weken, de precieze einddatum is nog onbekend. Om het project succesvol af te ronden stellen wij een aantal grenzen die aangeven wat er niet in de MongoExplorer komt. Deze grenzen zijn nodig om een realistisch doel te kunnen stellen binnen deze tijd.

* De webapplicatie gaat alleen document-oriented databasemanagementsystemen ondersteunen.
* De webapplicatie zal in eerste instantie read-only zijn. Dit houdt in dat het aanpassen, verwijderen of toevoegen van databases en documenten niet mogelijk is. Als er nog tijd over is in dit project zal dit eventueel toch wel geïmplementeerd kunnen worden.
* De webapplicatie gaat geen database-functies en -gebruikers ondersteunen. De reden hiervoor is dat het gebruik van functies en gebruikers niet bij onze doelgroep past en het dus niet veel zin zal hebben deze te implementeren.

# 6. Randvoorwaarden

In dit hoofdstuk staan een aantal randvoorwaarden. Deze randvoorwaarden zijn voorwaarden waar het team en de opdrachtgever zich aan moeten houden, om zo het project succesvol te kunnen afronden.

## 6.1. Werkruimte

Er moet binnen de school een lokaal beschikbaar zijn waarin de groep kan werken. Dit lokaal moet beschikken over stopcontacten en een draadloze internetverbinding.

## 6.2. Schoolritme

Er is door de school al een bepaald ritme uitgedacht waar het team zich aan zal moeten houden. Dit ritme gaat als volgt:

* Het team zal werken met sprints van 2 weken. Maandag aan het begin van de sprint is er een sprint-meeting. Aan het einde van elke sprint is er een sprint-review. Verder zal er elke woensdag een vergadering zijn met de coach.
* De projectgroep is op maandag tot en met vrijdag aanwezig van 9:00 uur tot 17:00 uur.
* Elk teamlid moet wekelijks een individueel voortgangsverslag inleveren, de groep heeft samen met de coach besloten dit in te leveren voor het wekelijks gesprek met de coach

## 6.3. Beschikbaarheid product-owner

Het zal soms nodig zijn om contact met de product-owner te hebben. De product-owner heeft aangegeven beschikbaar te zijn via de mail en daar snel op te zullen reageren.

## 6.4. Software

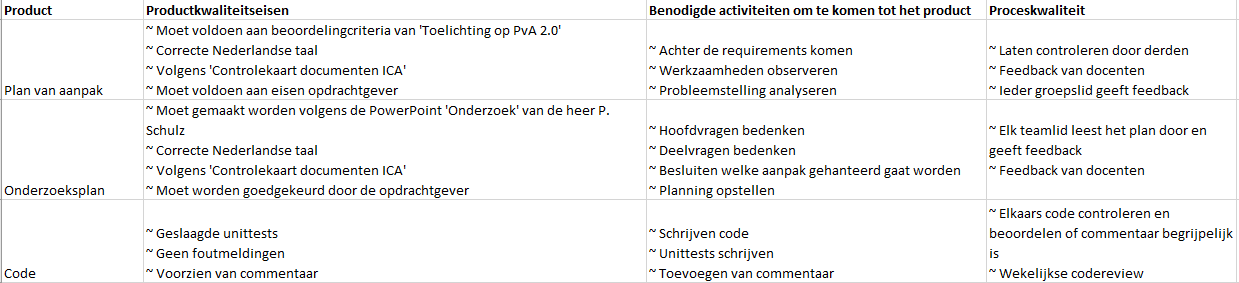
De volgende software is nodig om goed te kunnen werken:

* Javascript IDE
* MongoDB
* Git beheersysteem

Deze software is gratis beschikbaar of te krijgen met een studentenlicentie.

# 7. Producten en kwaliteit

In dit hoofdstuk worden alle producten, die in hoofdstuk vier zijn beschreven, onderverdeeld in kleine onderdelen. Deze onderdelen worden tot in detail beschreven. Vervolgens worden de producten beschreven met de bijbehorende kwaliteitseisen en de benodigde activiteiten om het gewenste eindresultaat te behalen.



# 8. Ontwikkelmethodes

Dit hoofdstuk gaat over ontwikkelmethodes. Ontwikkelmethodes worden gebruikt bij het ontwikkelen van software. In dit hoofdstuk stelt de projectgroep vast welke methode gebruikt gaat worden.

De HAN schrijft de projectgroep een methode voor genaamd scrum. Wij, als projectgroep, zijn ook enthousiast om deze methode te gebruiken. Er zijn een aantal redenen waarom scrum ten opzichte van andere ontwikkelmethodes een positieve impact kan hebben op het resultaat.

Er zijn een aantal verschillende ontwikkelmethodes. Degenen waar het projectteam mee in aanraking zijn gekomen zijn de watervalmethode en scrum. In de volgende twee paragrafen zal eerst worden uitgelegd hoe deze methodes in elkaar zitten.

## 8.1. De watervalmethode

De watervalmethode is een methode die stapsgewijs werkt. De methode bestaat uit verschillende fases. De verschillende fases zijn de analysefase, ontwikkelen van basisontwerp, het technisch ontwerp, het bouwen van de code, het testen en als laatste de integratie van het project.

De watervalmethode begint altijd bij de eerste fase, de analyse, en eindigt altijd bij het integreren van het project. De voordelen hiervan zijn dat fouten in het begin van het project sneller kunnen worden herkend en herstelt. Alle fases moeten eerst correct en compleet uitgevoerd worden, voordat het projectteam door kan gaan naar de volgende fase. Het tweede voordeel is dat bij de watervalmethode de nadruk ligt op de documentatie. Nieuwe ontwikkelaars die midden in het project aangenomen worden kunnen zo meteen aan de slag. Het derde voordeel is dat het een duidelijke methode is, elk lid van het team weet waar die mee bezig is en daardoor zit iedereen op één lijn. (Baars, z.d.)

De watervalmethode bevat ook nadelen. Het eerste nadeel is dat veel projecten afhankelijk zijn van externe factoren, bijvoorbeeld de opdrachtgever. De watervalmethode gaat ervan uit dat tijdens het project niks veranderd gaat worden. Als de opdrachtgever midden in het project van gedachten verandert, is het in theorie niet mogelijk om de doelstelling aan te passen. Het tweede punt is dat het lastig is om de duur en de kosten van het project in te schatten. Het derde nadeel is het testen. Testen gebeurt in een van de laatste fases, hierdoor kan er tijdens het testen tegen bugs aangelopen worden die veel tijd in beslag nemen. Het laatste nadeel is dat de watervalmethode niet geschikt is voor kleinere projecten omdat er zo veel nadruk wordt gelegd op de documentatie. (Tong, 2015)

## 8.2. Scrum

Scrum is een methode die flexibel werkt. De uitgangspunten van scrum zijn dat iedereen toewijding, focus, openheid, respect en lef toont. De scrummethode kent verschillende rollen. Deze rollen zijn: de product-owner, het projectteam en de scrummaster. De product-owner is de opdrachtgever of een tussenpersoon daarvan. Het ontwikkelteam bestaat uit de personen die het product gaan realiseren. Zij maken de analyse, het ontwerp, schrijven de code, testen het product en documenteren alles wat erbij komt kijken. De scrummaster begeleidt het projectteam en helpt ze met eventuele randvoorwaarden. Hij zorgt ervoor dat het team geen afleiding krijgt. (Schwaber, 2013)

Bij scrum werkt het team bij voorkeur in één ruimte, hierdoor is overleg binnen het team makkelijk. In de scrum-wereld werkt het projectteam in iteraties die 2 tot 4 weken duren. Deze iteraties worden sprints genoemd. Aan het einde van een sprint levert het team een werkend product op. De eerste maandag van de sprint zal er een sprint-meeting plaatsvinden waarin het team en de product-owner gezamenlijk een sprint-backlog opstellen. Op de laatste vrijdag van de sprint vindt de sprint-review plaats, in deze review presenteert het team het gerealiseerde product aan de product-owner.

Dit is gelijk ook het grootste voordeel van scrum. De product-owner kan op basis van het geleverde product bepalen hoe het team verder gaat en of er veranderingen nodig zijn.

Voordat het team kan beginnen met een sprint moeten ze eerst een sprint-backlog hebben. Een sprint-backlog bestaat uit een lijst met resultaten die een product-owner gerealiseerd wil hebben. Deze resultaten worden user-stories genoemd. Als een sprint begint moet het team samen met de product-owner bepalen welke user-stories ze gaan uitwerken. (Philippus, z.d.)

## 8.3. Conclusie

Het aankomende project zit nog in zijn beginfase, het team en de product-owner hebben samen nog geen requirements uitgewerkt. Hierdoor vindt de projectgroep het fijn dat de HAN ons de scrummethode toewijst. De scrummethode zorgt ervoor dat het team samen met de product-owner een mooi plan kan bedenken en dit kan uitvoeren. De product-owner ziet na de eerste sprint al een tastbaar resultaat en kan hierdoor beter bepalen, samen met het projectteam, wat hij zelf wil en hoe het project verder moet gaan.

Door bovengenoemde voordelen is scrum voor de projectgroep de perfecte methode.

Scrum past ook bij de projectgroep, omdat het project maar acht weken duurt. Acht weken is een relatief korte tijd en als de projectgroep eerst vier weken kwijt zou zijn aan het analyseren en ontwerpen dan weet de product-owner na vier weken nog steeds niet wat hij kan verwachten. De flexibiliteit van scrum overtreft het lineaire traject van de watervalmethode. De projectgroep is enthousiast om ermee aan de slag te gaan.

Voor dit project wordt officieel scrum gebruikt, maar in realiteit bestaat het project voor de eerste twee weken uit een onderzoeksfase en voor de laatste zes weken uit de softwareontwikkelingsfase. De eerste twee weken zijn dus eigenlijk waterval en de laatste zes weken scrum, waarbij wordt teruggekeken op de resultaten van de onderzoeksfase.

# 9. Technieken

In het volgende hoofdstuk worden de technieken beschreven die wij gebruiken in drie fases: ontwerpen, realiseren en testen.

## 9.1 Ontwerpen

Voordat het projectteam begint met het bouwen van de webapplicatie zullen een aantal ontwerpen gemaakt worden die voorgelegd worden aan de opdrachtgever.

* Low fidelity – schetsen over hoe de schermen eruit gaan zien.
* High fidelity – low fidelity-ontwerpen uitgewerkt in een tekenprogramma.

## 9.2 Realisatie

De webapplicatie wordt opgedeeld in een client-side gedeelte en een server-side gedeelte. Het is de bedoeling dat deze kanten naadloos met elkaar communiceren. Voor de client-side wordt gebruik gemaakt van de volgende technieken:

* React
* Redux
* Superagent
* Webpack
* HTML5
* CSS3 en Bootstrap

Server-side:

* ExpressJS
* NodeJS
* MongoDB

## 9.3 Testen

De software zal getest worden om te garanderen dat de klant deze kan gebruiken zonder problemen.

# 10. Projectorganisatie en communicatie

De organisatie binnen de school met betrekking tot dit project gaat als volgt; de projectgroep bestaat uit een vijftal studenten die dit project gaan realiseren. Zij krijgen hulp van een coach, deze coach zal de studenten helpen bij de voortgang van het project. De coach en het team hebben één keer per week een meeting, deze zal plaatsvinden op woensdagochtend. Naast de coach heeft het team een product-owner, deze heeft een bepaalde visie op het eindresultaat. De product-owner zal samen met de projectgroep het totaalplaatje van het eindproduct in kaart brengen. Het team en de product-owner hebben elke eerste maandag van een sprint een sprint-meeting en de laatste vrijdag van de sprint een sprint-review.

## 10.1 Taakverdeling en informatie

Naam: Sander Leer  
Rol: Coach  
Email: [sander.leer@han.nl](mailto:sander.leer@han.nl)

Naam: Peter Schultz  
Rol: product-owner  
Email: [Peter.Schulz@han.nl](mailto:Peter.Schulz@han.nl)

Naam: Roel de Man  
Rol: Productieteamlid  
Email: [rjmdeman@gmail.com](mailto:rjmdeman@gmail.com)

Naam: Alex Post  
Rol: Productieteamlid  
Email: [alexpost@upcmail.nl](mailto:alexpost@upcmail.nl)

Naam: Ciyan Cöcelli  
Rol: Productieteamlid  
Email: [ciyan97@live.nl](mailto:ciyan97@live.nl)

Naam: Marijn Martin  
Rol: Productieteamlid  
Email: [marijn.martin@hotmail.com](mailto:marijn.martin@hotmail.com)

Naam: Niels Wijers  
Rol: Productieteamlid  
Email: [nielsjlwijers@gmail.com](mailto:nielsjlwijers@gmail.com)

# 11. Planning

In dit hoofdstuk wordt de globale planning uitgelegd. Dit geeft een indicatie van welke taken op welk moment gerealiseerd worden. De planning is onderverdeeld in vier sprints en een presentatie week. Daarbinnen zijn taken gedefinieerd met de verwachtte start-dag en duur in dagen. Elke sprint duurt tien werkdagen/twee weken. Deze planning zal bijgehouden worden gedurende het gehele project.

Per sprint wordt er een meer specifieke planning gemaakt voor de aankomende twee weken, waarin staat wie welke taak op zich neemt en daarbij de benodigde tijd die deze persoon hiervoor denkt nodig te hebben.

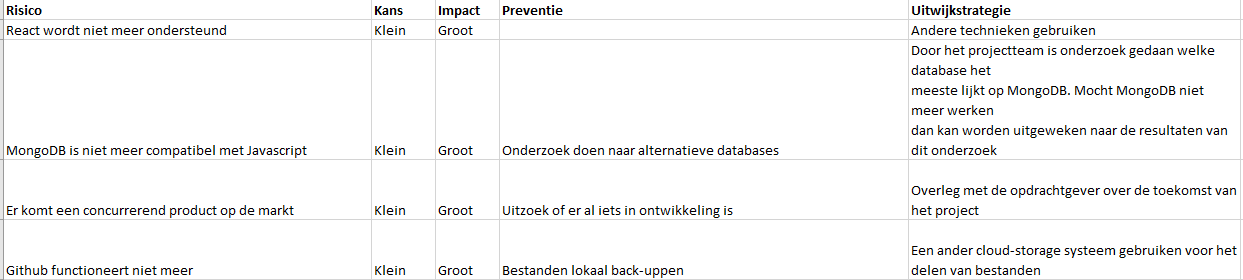
De planning is te vinden als bijlage A (hoofdstuk 14.1).

# 12. Risico’s

In dit hoofdstuk worden meerdere probleemscenario’s beschreven met daarbij de maatregelen die getroffen moeten worden wanneer deze scenario’s realiteit worden.

Daarbij wordt bij ieder probleemscenario beschreven hoe groot de kans ervan is en wat de impact is op het project. In de laatste kolom staat de ‘uitwijkstrategie’, hier wordt beschreven welke maatregelen er getroffen worden wanneer de tegenmaatregel niet van toepassing is of wanneer deze niet uitgevoerd kan worden.

## 12.1 Risicoschema



# 13. Bibliografie

* Baars, W. (z.d.). *Waterval versus Agile(cyclisch) projectmanagement*. Opgehaald van https://www.projectmanagement-training.nl/boek/waterval-versus-agile/
* HAN, H. v. (2016, november 10). *Over de HAN*. Opgehaald van HAN: https://www.han.nl/start/corporate/
* MongoDB inc. (Z.D.). *mongoDB*. Opgehaald van mongoDB: http://www.mongodb.com
* Paralect. (2016, november 10). Opgehaald van Robomongo: https://robomongo.org
* Philippus, E. (z.d.). *Wat is scrum?* Opgehaald van http://www.improvement-services.nl/wat-is-scrum.html
* Schwaber, K. S. (2013, juli). *The scrum guide*. Opgehaald van http://www.scrumguides.org/docs/scrumguide/v1/Scrum-Guide-US.pdf#zoom=100
* Tong, J. (2015, juni 8). *Waterfall sowtware development model*. Opgehaald van http://www.oxagile.com/company/blog/the-waterfall-model/

# 14. Bijlagen

## 14.1 Bijlage A – Globale planning, MongoExplorer

