

Plan van Aanpak

Het CMS voor iedereen

Dante Klijn



Contactgegevens

Student

Naam	Dante Klijn
Studentnummer	4565908
Academisch jaar	2023/2024
E-mail	dante.klijn@student.nhlstenden.com
Telefoonnummer	+31 (0)6 24 76 59 74

Onderwijsinstelling

Naam	NHL Stenden University of Applied Sciences
Course	HBO-ICT
Locatie	Rengerslaan 8-10, 8917 DD, Leeuwarden
Telefoonnummer	+31 (0)88 991 7000

Docentbegeleider

Naam	Stefan Rolink
Email	stefan.rolink@nhlstenden.com
Telefoonnummer	+31 (0)6 42 28 30 77

Afstudeercommissie

Email	afstuderenschoolofict@nhlstenden.com
-------	--------------------------------------

Examencommissie

Email	examencommissiehboict@nhlstenden.com
-------	--------------------------------------

Organisatie

Naam	Snakeware New Media B.V
Locatie	Veemarktplein 1, 8601 DA, Sneek
Telefoonnummer	+31 (0)515 431 895

Bedrijfsbegeleider

Naam	Thom Koenders
Email	thom@snakeware.com
Telefoonnummer	+31 (0)6 13 09 18 51
Rol	Senior software developer

Versiebeheer

Versie	Datum	Veranderingen
1.0	13 oktober	Eerste versie

Woordenlijst

Content management systeem Een contentmanagementsysteem is een softwaretoepassing, meestal een webapplicatie, die het mogelijk maakt dat mensen eenvoudig, zonder veel technische kennis, documenten en gegevens op internet kunnen publiceren (contentmanagement). Als afkorting wordt ook wel CMS gebruikt.

Graphical user interface Een graphical user interface, is een manier van interacteren met een computer waarbij grafische beelden, widgets en tekst gebruikt worden.

Search engine optimization Search Engine Optimisation, zijn alle processen en verbeteringen die als doen hebben een website hoger in Google te laten verschijnen.

Software development life cycle de software development life cycle (SDLC) is een processmatige manier van werken met als doel goede kwaliteit software te produceren met lage kosten in een korte tijd. De SDLC bestaat uit 5 fases: *Requirements analysis*, *Design*, *Implementation*, *Testing*, en *Evolution* (Zuci Systems, g.d.).

User journeys User journeys zijn de verschillende acties die een gebruiker moet uitvoeren (Meestal via een interface) om een eind resultaat te bereiken.

Inhoudsopgave

Woordenlijst	iv
1 Inleiding	1
1.1 Organisatieomschrijving	1
1.2 Context	1
1.3 Aanleiding	2
1.4 Opdrachtoomschrijving	2
2 Onderzoek	4
2.1 Hoofd- en deelvragen	4
2.2 Onderzoeksmethoden	4
2.3 Onderzoeksopzet	5
2.4 Tijdlijn	6
3 Projectresultaat	7
3.1 Doelstelling	7
3.2 Eindproduct	7
4 Projectactiviteiten	9
4.1 Plan van Aanpak	9
4.2 Onderzoek	9
4.3 Product	9
4.4 Eindpresentatie	10
4.5 Ondersteuning	10
5 Projectgrenzen	11
5.1 Afbakening	11
5.2 Definition of done	12
5.3 Randvoorwaarden	12
6 Kwaliteitsbewaking	13
6.1 Proces	13
6.1.1 Code reviews	14
6.1.2 Versiebeheer	14
6.2 Ontwerpen	14
6.3 Testing	15
6.4 Documentatie	15

7	Planning	16
7.1	Op te leveren producten	16
7.2	Strokenplanning	16
A	Strokenplanning	19

Hoofdstuk 1

Inleiding

Dit document beschrijft het plan van aanpak voor het “Het CMS voor iedereen” project. Dit is een onderdeel van de afstudeerstage van NHL Stenden Hogeschool en wordt uitgevoerd bij Snakeware New Media B.V. In dit plan van aanpak worden de verschillende aspecten van de afstudeerstage beschreven.

In dit hoofdstuk wordt een inleiding gegeven op de opdracht en wordt de context geschetst waarin deze opdracht wordt uitgevoerd.

1.1 Organisatieomschrijving

Snakeware New Media B.V. (Snakeware) is een E-business bureau gevestigd in Nederland. Haar aangeboden diensten omvatten het adviseren, bouwen en onderhouden van digitale producties, met een focus op websites, webshops en mobiele apps (Snakeware, 2022b). Op het moment van schrijven telt Snakeware meer dan 60 werknemers, elk met verschillende specialiteiten. Ze leveren services aan welbekende organisaties zoals DPG Media, DekaMarkt en Poiesz supermarkten (Snakeware, 2022a).

1.2 Context

Snakeware heeft een platform genaamd “Snakeware Cloud” dit platform is een content management systeem (CMS) waarmee ze digitale content kunnen leveren voor haar (grotere) klanten. Snakeware cloud is een applicatie waarmee Snakeware en haar klanten webapplicaties kan inrichten en voorzien van content.

De klant van Snakeware kan zijn of haar website zelf inrichten door middel van het specificeren van de content op de verschillende pagina's. Dit wordt gedaan door middel van artikelen die door het CMS gebruikt kunnen worden. De content van het artikel kan verschillen tussen simpele tekst, vragenlijst, webshop items, etc. Hiernaast zijn er ook search engine optimization (SEO) opties binnen Snakeware cloud om de site goed te kunnen vinden op het internet. Hierbij kun je denken aan titel tags en zoekwoorden kunnen toegevoegen in de head (Mozilla, 2023c).

Hierom heeft Snakeware cloud veel features en configuratie stappen wat het complex en duur maakt om een relatief kleine webapplicatie te maken voor kleinere klanten. Dit zorgt ervoor dat Snakeware zich niet kan vestigen in een markt met veel kleinere klanten, en hierdoor omzet mis loopt.

1.3 Aanleiding

Het huidige platform is 21 jaar oud en er is veel functionaliteit in de loop der jaren aan toegevoegd. Omdat Snakeware Cloud een oud platform is zijn er veel technieken en best practices gebruikt die nu niet meer als optimaal worden beschouwd. Deze technieken waren erg geïntegreerd in Snakeware cloud en er is het verleden gekozen om niet de code herschrijven om het aan de huidige standaarden te voldoen van andere projecten. Een voorbeeld hiervan is tabel naam prefix afkortingen bij elke kolom zetten, of gigantische C# (Microsoft, 2022) files van 10 000 regels met verschillende functies. Deze functies houden zich niet aan de *Single Responsibility Principle* van de SOLID ontwerpmethode (Watts, 2020) wat het moeilijk maakt om het huidige CMS te onderhouden.

Ook zijn er technieken toegepast die nu niet meer relevant zijn. Een voorbeeld hiervan is dat het CMS gebruikmaakt van JavaScript (Mozilla, 2023b) en toen ze er mee begonnen bestonden JavaScript classes (Mozilla, 2023a) nog niet, dus hebben ze die zelf geïmplementeerd. Deze oudere technieken en standaarden zorgen ervoor dat het meer tijd kost om het CMS te onderhouden vanwege de extra code. Dit zorgt ervoor dat het meer tijd en geld kost om het Snakeware cloud uit te breiden.

Een van de voornaamste uitdaging met Snakeware Cloud betreft de verouderde datastructuur van de applicatie. Deze veroudering is het gevolg van een initiële ontwikkeling waarbij onvoldoende rekening werd gehouden met toekomstige functionaliteitsuitbreidingen in het systeem. Als gevolg daarvan is de onderliggende datastructuur niet aangepast, maar zijn er elementen aan toegevoegd. Dit heeft geresulteerd in database queries van duizenden regels en complexe relaties tussen tabellen in de database. Dit huidige scenario bemoeilijkt aanzienlijk het toevoegen van nieuwe functionaliteiten, wat resulteert in aanzienlijke tijds- en kosteninvesteringen.

Hierom wilt Snakeware dat er een nieuwe datastructuur komt met daar bij een CMS-API. Omdat er een nieuwe datastructuur moet komen en de logica van het oude systeem nauw verbonden is met de datastructuur is het niet mogelijk om de oude code opnieuw te gebruiken.

1.4 Opdrachtomschrijving

De opdracht is om een proof of concept CMS-API te ontwikkelen die gebruikt maakt van een datamodel en systeemarchitectuur dat flexibeler, onderhoudbaarder en gebruikt maakt van moderne best practices. Tijdens de afstudeeropdracht wordt er primair op het datamodel en de systeemarchitectuur gefocust. Omdat er nog geen concreet datamodel en systeemarchitectuur is zal dit onderzocht/ontworpen moeten worden.

De opdracht omvat het achterhalen van de requirements, ontwerpen en ontwikkelen van het proof of concept met als focus een nieuw datamodel, met de essentiële functionaliteiten.

Het huidige Snakeware cloud platform bestaat uit 2 verschillende graphical user interface (GUI):

- Snakeware cloud GUI
- klant webapplicatie

Met de Snakeware cloud GUI kan de klant de content van de website aanpassen. Door middel

van de webapplicatie kan de eindgebruiker de content bekijken en interacteren. Er is voor gekozen om niet de Snakeware cloud GUI te realiseren om de afstudeeropdracht in scope te houden. Er is wel voor gekozen om de klant webapplicatie in zijn minimaal uit te werken.

Om de user journeys te testen wordt er gebruikgemaakt van postman workflows (Postman.com, 2023). Het doel van het proof of concept is dat er aangetoond kan worden dat door het gebruiken van een nieuw datamodel en systeemarchitectuur ook services verleend kunnen worden aan kleinere klanten. Dit zou eventueel ook een startpunt zijn om op verder te bouwen.

Hoofdstuk 2

Onderzoek

In de onderzoeksfase wordt er onderzocht wat de requirements zijn die worden gesteld aan het softwaresysteem. Het onderzoek wordt door middel van de methode van Verhoeven (Verhoeven, 2018) uitgevoerd. Er wordt daarnaast gebruikgemaakt van de onderzoeksmethoden die beschreven zijn door HBO-I (HBO-I, 2018).

2.1 Hoofd- en deelvragen

In dit hoofdstuk worden de hoofd- en deelvragen behandeld. Eerst wordt de hoofdvraag behandeld en daarna de deelvragen.

Hoofdvraag: Wat zijn de requirements die worden gesteld aan een contentmanagementsysteem waarmee Snakeware webapplicaties ook aan kleinere klanten kan aanbieden?

Deelvraag 1: Wie zijn de stakeholders van het het systeem?

Deelvraag 2: Wat is de huidige softwarearchitectuur van Snakeware Cloud?

Deelvraag 3: Wat zijn de huidige knelpunten in Snakeware Cloud?

Deelvraag 4: Welke requirements worden aan het systeem gesteld door de stakeholders?

Deelvraag 5: Wat zijn de prioriteiten die worden gesteld aan de requirements?

2.2 Onderzoeksmethoden

In dit hoofdstuk worden de onderzoeksmethoden beschreven die worden gebruikt tijdens het onderzoek.

***Deelvraag 1:** Wie zijn de stakeholders van het het systeem?*

Om te weten wat de eisen en wensen zijn voor het systeem is het belangrijk dat je weet voor wie je het maakt. Daarom worden de stakeholders van het project worden in kaart gebracht door middel van een **stakeholdersanalyse**. Als de stakeholders zijn geïdentificeerd worden ze daar na geprioriteerd op basis van invloed en belang.

***Deelvraag 2:** Wat is de huidige softwarearchitectuur van Snakeware Cloud?*

Om de huidige problemen van het Snakeware cloud platform in beeld te brengen is het belangrijk dat er gekeken naar de huidige architectuur van snakeware cloud. Daarom wordt er door middel van **IT architecture sketching** met samenwerking van het R&D team de

huidige architectuur in beeld gebracht. Hier uit wordt een lijst met problemen verzameld die de huidige architectuur nu heeft, en wordt ter ondersteuning gebruikt van deelvraag 3 (*Wat zijn de huidige knelpunten in Snakeware Cloud?*).

Deelvraag 3: Wat zijn de huidige knelpunten in Snakeware Cloud?

Bij deze deelvraag wordt er onderzocht wat nu de huidige knelpunten zijn van het Snakeware Cloud platform. Door middel van gestructureerde **expertinterviews** wordt er onderzocht waar de huidige klachten zitten, en wat het veroorzaakt. Na het expertinterview wordt er een **task analyse** gedaan om de knelpunten in beeld te krijgen. De interviews worden gedaan met de stakeholders, hierdoor kunnen alle huidige klachten in beeld worden gebracht.

Deelvraag 4: Welke requirements worden aan het systeem gesteld door de stakeholders?

Deze deelvraag wordt onderzocht door de stakeholders te interviewen door middel van gestructureerde **interviews**. De vragen voor dit interview worden van tevoren opgesteld en gevalideerd met de product owner en de bedrijfsbegeleider. Vervolgens worden de wensen en eisen in beeld gebracht door middel van **explore user requirements**.

Deelvraag 5: Wat zijn de prioriteiten die worden gesteld aan de requirements?

Deze deelvraag beantwoordt de prioriteit van de verschillende requirements. Na dat alle requirements zijn verzameld worden er prioriteiten aan toegekend door middel van **requirements prioritization**. Vervolgens worden ze gecategoriseerd door middel van de MoSCoW methode (Monday.com, 2022). Als de requirements zijn geprioriteerd zijn dan wordt er samen met de bedrijfsbegeleider bepaald welke requirements gerealiseerd worden, dit wordt gedaan om de afstudeeropdracht scope haalbaar te maken.

2.3 Onderzoeksopzet

Het onderzoek wordt opgezet door gebruik te maken van de methode van Nel Verhoeven (Verhoeven, 2018). De methode van Verhoeven bestaat uit de volgende 4 fases:

1. Ontwerpen.
2. Gegevens verzamelen.
3. Analyseren.
4. Evalueren en adviseren.

Door middel van de fases wordt het onderzoek opgesteld, dit wordt gerepresenteerd door de verschillende hoofdstukken in het onderzoek:

- De inleiding en Onderzoeksopzet zijn onderdeel van de eerste fase van de methode van Verhoeven (ontwerpen). Hier wordt de context van het onderzoek in beeld gebracht en hoe het onderzoek uitgevoerd gaat worden.
- De gegevens worden verzameld en geanalyseerd (fase 2 en 3 van Verhoeven (Verhoeven, 2018)) in het hoofdstuk resultaten. Dit wordt gedaan door middel van deelvragen en onderzoeksmethodes die op zijn gezet in de eerste fase.
- In de laatste fase (Evalueren en adviseren) wordt gedaan in het hoofdstuk conclusie, hier wordt er een conclusie getrokken uit de resultaten die in de vorige hoofdstukken gekomen zijn.

2.4 Tijdlijn

De onderzoeksfase loopt van het begin van week 42 (16 oktober 2023) tot in het midden van de werkweek 47 (22 november 2023). In week 42 tm/ week 45 worden de onderzoeksmethoden uitgevoerd en resultaten verzameld. In week 46 wordt er een conclusie uit het onderzoek getrokken en een reflectie gemaakt. Tijdens de verslaglegging van het onderzoek zal er een keer per week contact gemaakt worden met de docentbegeleider en bedrijfsbegeleider om het proces naar een goed einde te leiden. Als het onderzoek is afgerond kan er op basis van de requirements een ontwerp gemaakt worden van het systeem.

Hoofdstuk 3

Projectresultaat

In dit hoofdstuk staat het projectresultaat beschreven met daarbij de doelstelling en eindproduct.

3.1 Doelstelling

Het doel is om binnen de afstudeer periode (18 september 2023 tot 12 april 2024) een proof of concept systeem neer te zetten binnen Snakeware. Waarbij er voldaan is aan de belangrijkste (de must have's van de MoSCoW analyse) requirements van de stakeholders.

3.2 Eindproduct

Om de gestelde doelen te bereiken, zullen er vier eindproducten worden ontwikkeld. Deze eindresultaten omvatten het projectresultaat, dat aan het einde van de afstudeerperiode wordt gepresenteerd en gedemonstreerd. De vier producten in kwestie zijn: het Plan van Aanpak, het Onderzoeksverslag, het Technisch Verslag en het Eindproduct.

Plan van Aanpak: Dit document beschrijft in detail de uitvoering van de opdracht.

Onderzoeksverslag: In het onderzoeksverslag staat het uitgewerkte onderzoek. Het onderzoek zal een lijst van geprioriteerde requirements opleveren die gebruikt worden tijdens de ontwerpfase en realisatiefase.

Technisch verslag: In het Technisch Verslag worden de beslissingen die zijn genomen tijdens het uitvoeringsproces duidelijk uiteengezet, en wordt het ontwerp gepresenteerd met onderbouwing van de gemaakte keuzes. Voor het opstellen van zowel het ontwerp als de implementatie is het van cruciaal belang om rekening te houden met de eisen en wensen die zijn voortgekomen uit de requirementanalyse.

Tijdens het ontwerpen wordt er gebruikgemaakt van het 4 + 1 model, waarin de architectuur van het systeem duidelijk zichtbaar wordt. Verder dient het technisch verslag gedetailleerd te beschrijven hoe het systeem functioneert, inclusief de keuzes die zijn gemaakt op basis van het ontwerp. Om het systeem door loops te testen wordt er gebruikgemaakt van het V-Model (Oppermann, 2023). Aan het einde van het technisch verslag wordt er gereflecteerd op de ontwerp- en het realisatieproces, waarbij er gebruikgemaakt wordt van de STARR-methode (Scribbr, 2023).

Product: Het product is een proof of concept CMS-API die de data kan tonen op een simpele frontend. Bekijk hoofdstuk 1.4 voor meer informatie.

Hoofdstuk 4

Projectactiviteiten

In dit hoofdstuk worden de projectactiviteiten van de afstudeerperiode. Deze activiteiten omvatten het opstellen van het plan van aanpak, het uitvoeren van het onderzoek, het ontwikkelen van het product (met inclusie van het technisch verslag), de eindpresentatie en de demonstratie. Daarnaast zullen ook de doorlopende activiteiten worden besproken.

4.1 Plan van Aanpak

Voor dit project wordt een plan van aanpak opgesteld om het project helder en concreet te definiëren. Dit plan van aanpak dient als een leidraad voor zowel de docentbegeleider als Snakeware, waarin wordt beschreven hoe het project wordt opgezet. Tevens wordt de aanpak van het onderzoek in detail uiteengezet binnen dit plan van aanpak, waardoor er direct aan het onderzoek kan worden gewerkt.

4.2 Onderzoek

Het onderzoek wordt uitgevoerd over een periode van 5 weken, hierbij worden de verschillende deelvragen toegewezen aan hun eigen activiteit. Deze activiteiten tot betrekking van de deelvragen zijn:

- Stakeholders analyse doen binnen Snakeware.
- Onderzoeken huidige architectuur van Snakeware Cloud.
- Knelpunten onderzoeken van Snakeware Cloud.
- Explore user requirements uitvoeren bij de stakeholders.
- Requirement prioritization uitvoeren.

Daarnaast wordt er ook een conclusie en discussie geschreven.

4.3 Product

In de realisatieperiode van het product zijn er 2 weken gepland om de basis van systeem te ontwerpen. Nadat de basis van het ontwerp opgezet is worden er 9 sprints geplant, waarbij elke sprint 2 weken duurt. Tijdens deze sprints wordt voornamelijk de code geschreven en getest. Door de sprints heen wordt het technisch verslag opgezet en het daarbij behorende 4 + 1 view architectuur (Kruchten, 1995) UML diagrammen. Hierdoor bestaat het realiseren

van het product uit de volgende activiteiten:

- Opstellen van technische documentatie
- Ontwerp opzetten
- Testen
- 4 + 1 architectuurmodel opzetten
- Sprints (realiseren)

4.4 Eindpresentatie

Aan het einde van de afstudeerperiode wordt een afsluitende presentatie gehouden voor de Bedrijfsbegeleider en andere collega's bij Snakeware. Daarnaast wordt eveneens een eindpresentatie gegeven voor de docentbegeleider en de examinatoren. Dit wordt vertaald naar de volgende activiteiten:

- Voorbereiding presentatie
- Presentaties
- Demonstratie product

4.5 Ondersteuning

Tijdens de afstudeerperiode zal er ondersteuning gevraagd worden om de afstudeeropdracht in een goede baan te leiden. Er zal voortdurend contact zijn met de Bedrijfsbegeleider, waarbij er een wekelijkse feedbackmomenten plaatsvindt om de voortgang van het project te bespreken en toekomstige stappen te bepalen. Ook zal er per week een keer afgesproken worden met de docentbegeleider om de stand van zaken te bespreken. Daarnaast wordt er een keer per maand een technisch overleg georganiseerd waarbij kennis wordt gedeeld met betrekking tot technische uitdagingen binnen Snakeware. Dit wordt vertaald naar de volgende activiteiten:

- Contact met de bedrijfsbegeleider
- Contact met de docentbegeleider
- Technisch overleg

Hoofdstuk 5

Projectgrenzen

In dit hoofdstuk worden de projectgrenzen toegelicht. De afbakening van het project licht toe wat er wel en niet gemaakt binnen de context van de afstudeeropdracht. Vervolgens wordt de definition of done beschreven om aan te tonen wanneer de opdracht voldaan is. Daarna worden de randvoorwaarden van de afstudeeropdracht besproken.

5.1 Afbakening

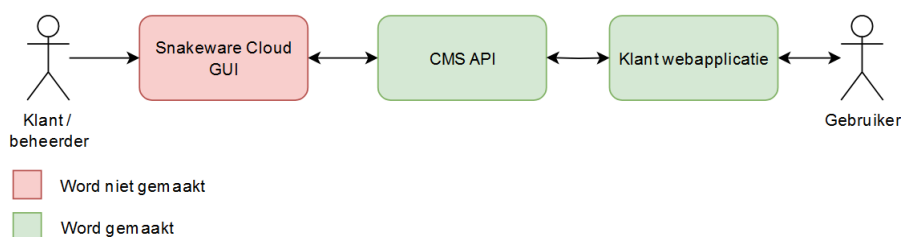
Het huidige Snakware cloud plaform heeft in de loop der jaren veel functionaliteiten gekregen. Hierom is het van belang dat tijdens de afstudeerperiode de opdracht niet te groot wordt gemaakt om het succesvol te kunnen afronden. Daarom wordt er primair gefocust op het realiseren en ontwerpen van de CMS-API.

Snakware cloud bestaat momenteel uit 3 verschillende applicaties, deze applicaties zijn:

- Snakware cloud GUI
- CMS-API
- klant webapplicatie

Er is besloten om niet de Snakware cloud GUI te maken om de scope van de afstudeeropdracht haalbaar te maken. Daarnaast wordt de klant webapplicatie minimaal uitgewerkt om de data van de CMS-API te kunnen tonen. Om de user journeys te kunnen valideren worden ze getest door middel van postman workflows. Voor verheldering is in figuur 5.1 een versimpelde communicate diagram van de producten die gemaakt en niet gemaakt worden tijdens de afstudeer opdracht.

Figuur 5.1: Gesimplificeerde communicatie diagram van systemen



5.2 Definition of done

De definition of done is wanneer er een proof of concept CMS-API en de daarbij behorende klant webapplicatie is opgeleverd. Die voldoen aan de belangrijkste (de must have's van de MoSCoW analyse) eisen en wensen die uit het onderzoek zijn gekomen van de stakeholders. Het systeem moet flexibel opgesteld worden, zodat in de toekomst extra functionaliteiten toegevoegd kunnen worden. Daarnaast is het van belang dat de school documentatie met een voldoende wordt afgerond. De verschillende documenten zijn: plan van aanpak, onderzoeksverslag en het technisch verslag. Daarnaast moet het product en de presentatie met een voldoende afgerond worden.

5.3 Randvoorwaarden

Om het project goed af te kunnen ronden zijn er randvoorwaarden aan het project verbonden:

- Er is een werkplek nodig om alle werkzaamheden tijdens de afstudeerperiode te kunnen afronden.
- Er wordt een keer per week een afspraak gemaakt met de bedrijfsbegeleider en de docentbegeleider om de progressie te bespreken.
- De afstudeeropdracht moet afgerond worden binnen de duur van de afstudeerperiode. Dit is ongeveer 125 werkdagen en loopt van 18 november 2023 tot 5 april 2024.

Hoofdstuk 6

Kwaliteitsbewaking

In dit hoofdstuk wordt beschreven hoe de kwaliteit van het product gewaarborgd wordt. Dit hoofdstuk is onderverdeeld in de volgende onderdelen: proces, ontwerpen, testing en documentatie.

6.1 Proces

De ontwerp en realisatie fases van het project worden ondersteund door middel van de scrum methodiek (Vennix, g.d.). Tijdens de realisatieperiode worden alle fases van de software development life cycle (SDLC) (figuur 6.1) doorlopen.

Figuur 6.1: De Software Development Life Cycle, afkomstig uit de afstudeerhandleiding (Bosgra e.a., 2022)



Requirement analysis: Door middel van de requirement analyse (het onderzoek dat beschreven staat in hoofdstuk 2) worden de requirements opgesteld. Tijdens de sprint planning worden de requirements verwerkt tot behapbare user stories, zodat ze tijdens de sprints gemaakt kunnen worden.

Design: De eisen en wensen die vanuit de requirement analyse worden gehaald worden gebruikt om een ontwerp te maken. Hoe er ontworpen wordt is toegelicht in hoofdstuk 6.2.

Implementation: De eisen en wensen worden geïmplementeerd volgens het ontwerp.

Testing: Door de loop van het realisatieproces worden de verschillende testen van het V-Model uitgevoerd, zie hoofdstuk 6.3 voor meer informatie.

Evolution: Er wordt met regelmaat feedback momenten gehouden waar het product gedemonstreerd wordt aan de bedrijfsbegeleider. Dit wordt gedaan om de projectvoortgang te bewaken op basis van de gemaakte eisen en wensen.

6.1.1 Code reviews

Tijdens de realisatiefase worden er meerdere code reviews gehouden om de codekwaliteit te waarborgen. Deze code reviews worden gehouden met de backend en frontend developers van Snakeware, de code zal beoordeeld worden op coderingsstandaarden en structuur.

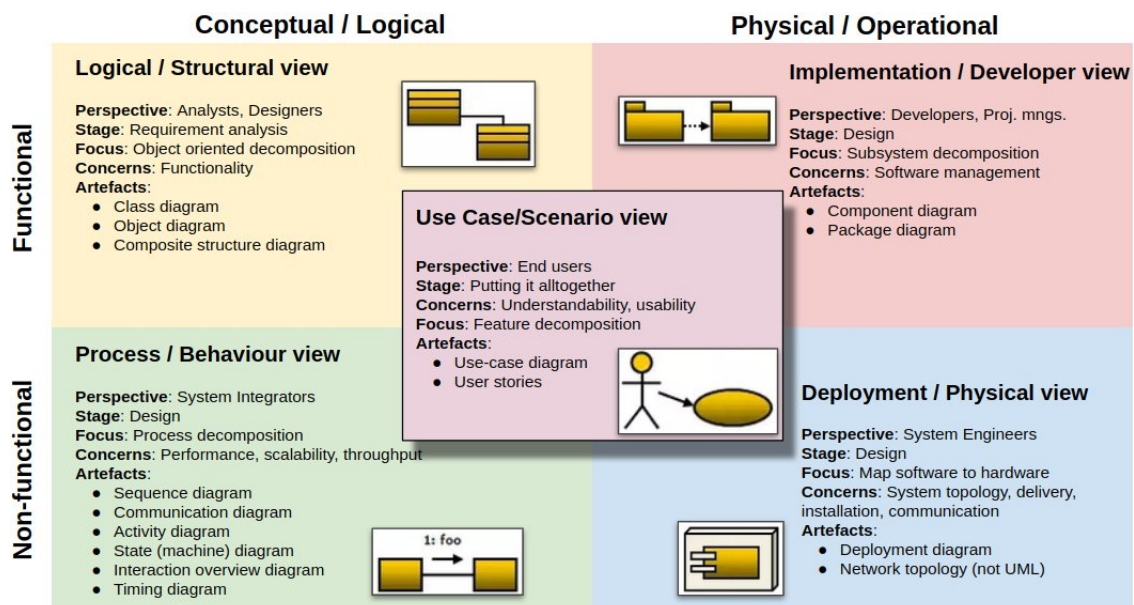
6.1.2 Versiebeheer

Binnen Snakeware wordt er gebruikgemaakt van Bitbucket (Bitbucket, g.d.) en git (Git-scm, g.d.) voor het versiebeheer van de code, dit zal ook gebruikt worden voor de afstudeeropdracht. Verder zal er ook gebruikgemaakt worden van een CI-pipeline die wordt opgezet om de code automatisch te testen.

6.2 Ontwerpen

Tijdens het ontwerpen wordt er gebruikgemaakt van UML. Er zullen verschillende UML diagrammen gemaakt worden op basis van het 4 + 1 view model (Kruchten, 1995) . Dit wordt gerealiseerd na het onderzoek wanneer de requirements bekend zijn. In figuur 6.2 wordt beschreven welke diagrammen in theorie gebruikt kunnen worden voor de verschillende views van het 4 + 1 view model.

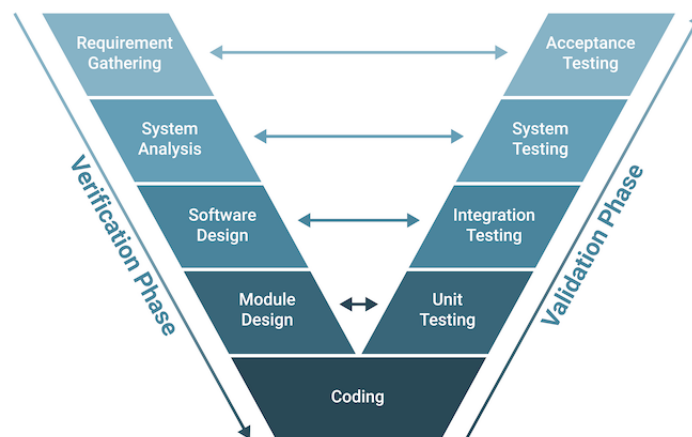
Figuur 6.2: Verschillende UML diagrammen die mogelijk gebruikt kunnen worden bij de verschillende views (hgraca, 2019)



6.3 Testing

Tijdens de realisatieperiode wordt er gebruikgemaakt van de V-Model. Het V-Model 6.3 toont dat er voor de verschillende design lagen van het systeem verschillende testen gebruikt worden. De verschillende units/modules van code worden getest door middel van unit testing. Vervolgens worden de samen hangende units/modules getest door integratie testen om het softwaredesign te kunnen valideren. De laag boven softwaredesign is systeemanalyse, dit wordt getest door middel van systeem testing. Daarna worden de requirements gevalideerd door acceptatie testen uit te voeren met de stakeholders.

Figuur 6.3: V-Model (Oppermann, 2023)



6.4 Documentatie

De kwaliteit van de documentatie wordt gevalideerd door de bedrijfsbegeleider, docentbegeleider en medestudenten. Er wordt met de docent en bedrijfsbegeleider een keer per week afgesproken om de kwaliteit en de voortgang van de verschillende zaken van de afstudeerperiode te bespreken, waaronder de verslaglegging. Verder zal er per hoofdstuk feedback worden gevraagd om de kwaliteit van de documentatie hoog te houden. In de verslagen zullen onderbouwd door bronnen en deze zullen gedocumenteerd worden door APA 7 (Scribbr, g.d.) richtlijnen. Voor het indienen van een document is het vereist dat minimaal een conceptversie is beoordeeld door zowel de bedrijfsbegeleider als de docentbegeleider.

Hoofdstuk 7

Planning

In dit hoofdstuk wordt de initiële planning beschreven, en daarbij de verschillende deadlines voor de op te leveren producten. Verder wordt er een strokenplanning beschreven in hoofdstuk 7.2.

7.1 Op te leveren producten

De deadlines en concept versies van de verschillende eindproducten worden beschreven in Tabel 7.1.

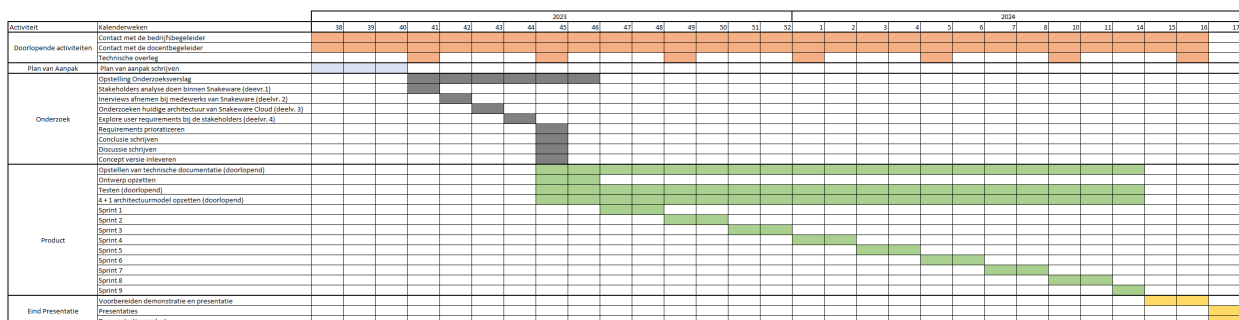
Op te leveren producten	Conceptversie	Deadline
Plan van aanpak	5-10-2023	13-10-2023
Onderzoeksverslag	8-11-2023	22-11-2023
Onderzoeksverslag (2de kans)	26-3-2023	12-4-2024
Technische verslag	26-3-2023	12-4-2024
Product	niet van toepassing	12-4-2024
Presentatie	29-4-2024	6/7/8 mei 2024

Tabel 7.1: Deadlines en conceptversie inlever momenten

7.2 Strokenplanning

In Figuur 7.1 wordt de strokenplanning weergegeven, die een verwachting geeft van de verwachte voortgang van het project. Deze planning omvat alle projectactiviteiten een vergrote versie is te vinden in bijlage A.

Figuur 7.1: Voorlopige Strokenplanning



Bibliografie

- Bitbucket. (g.d.). Verkregen 12 oktober 2023, van <https://bitbucket.org/>
- Bosgra, M., Derksen, L., van der Ploeg, W., & Sariedine, F. (2022, juli). *Afstudeerhandleiding 2022-2023* (PDF) (Verkregen 10 november 2022). NHL Stenden Hogeschool.
- Git-scm. (g.d.). *About*. Verkregen 12 oktober 2023, van <https://git-scm.com/about>
- HBO-I. (2018). *Methods*. Verkregen 2 oktober 2023, van <https://ictresearchmethods.nl/Methods>
- hgraca. (2019). Verkregen 5 oktober 2023, van <https://herbertograca.com/2019/08/12/documenting-software-architecture/>
- Kruchten, P. (1995, november). *Architectural Blueprints—The “4+1” View Model of Software Architecture* (report). Rational Software Corp.
- Microsoft. (2022). *Een rondleiding door de C#-taal*. Verkregen 10 oktober 2023, van <https://learn.microsoft.com/nl-nl/dotnet/csharp/tour-of-csharp/>
- Monday.com. (2022). *The MoSCoW prioritization method explained*. Verkregen 2 oktober 2023, van <https://monday.com/blog/project-management/moscow-prioritization-method/>
- Mozilla. (2023a). *Classes*. Verkregen 11 oktober 2023, van <https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Classes>
- Mozilla. (2023b). *JavaScript*. Verkregen 10 oktober 2023, van 2023-10-10
- Mozilla. (2023c). *What’s in the head? Metadata in HTML*. Verkregen 11 oktober 2023, van https://developer.mozilla.org/en-US/docs/Learn/HTML/Introduction_to_HTML/The_head_metadata_in_HTML
- Oppermann, A. (2023). Verkregen 5 oktober 2023, van <https://builtin.com/software-engineering-perspectives/v-model>
- Postman.com. (2023). *Customize request order in a collection run*. Verkregen 6 oktober 2023, van <https://learning.postman.com/docs/collections/running-collections/building-workflows/>
- Scribbr. (g.d.). *APA-stijl (7de editie) — Verwijzingen in de tekst & bronvermelding*. Verkregen 12 oktober 2023, van <https://www.scribbr.nl/category/apa-stijl/>

- Scribbr. (2023). *Reflecteren met de STARR-methode — Stappenplan + voorbeelden*. Verkregen 3 oktober 2023, van <https://www.scribbr.nl/stage/starr-methode/>
- Snakeware. (2022a). *Cases*. Verkregen 10 oktober 2023, van <https://www.snakeware.nl/cases>
- Snakeware. (2022b). *What we do*. Verkregen 10 oktober 2023, van <https://www.snakeware.com/what-we-do>
- Vennix, M. (g.d.). *Wat is Scrum?* Verkregen 12 oktober 2023, van <https://leanagilegroep.nl/wat-is-scrum/>
- Verhoeven, N. (2018). *Wat is onderzoek* (2de ed.). Boom.
- Watts, S. (2020). *The importance of SOLID Design Principles*. Verkregen 11 oktober 2023, van <https://www.bmc.com/blogs/solid-design-principles/>
- Zuci Systems. (g.d.). *Levenscyclus van softwareontwikkeling — Wat is SDLC*. Verkregen 10 oktober 2023, van <https://www.zucisystems.com/nl/diensten/levenscyclus-van-softwareontwikkeling-wat-is-sdlc/>

