

Plan van aanpak

Level Indication Based On Temperature (LIBOT)



Student:	Michiel Buevink
Opleiding:	Hogeschool van Arnhem en Nijmegen Informatica
Studentnummer:	481072
Datum:	4 maart 2015
Docent:	Leon van Houten
Project:	Project LIBOT
Stagebedrijf:	Inteqnion BV
Stagebegeleider:	Mark Smeenk
Opdrachtgever:	Charl van Kesteren
Versie	V3

Versiebeheer

Versie	Datum	Aanpassing
0.1	04-02-2015	Hoofdstukken: inhoud, inleiding, context, projectdefinitie, gekozen competenties gemaakt
0.2	05-02-2015	Versiebeheer, methode en technieken, globale project aanpak, tussenresultaten gemaakt. Overige hoofdstukken bijgewerkt
0.3	06-02-2015	Planning en projectactiviteiten toegevoegd, hoofdstukken kwaliteit en risico's toegevoegd, bronnen aangepast, definition of done aan kwaliteit toegevoegd.
1	06-02-2015	Eindcontrole voor feedback
1.1	11-02-2015	Verwerking feedback Mark Smeenk en Charl van Kesteren <ul style="list-style-type: none"> - Inleiding herschreven - Uitleg RUP aangepast - Uitleg keuze RUP toegevoegd - Uitleg complexiteit toegevoegd - Planning vervangen - Hoofdstukken/alinea's herschreven - Extra risico's toegevoegd Update sjabloon
1.9	13-02-2015	Eindredactie deel 1 <ul style="list-style-type: none"> - Hoofdstukken/alinea's/zinnen herschreven
2	16-02-2015	Eindredactie deel 2
2.1	17-02-2015	Hoofdstuk nummering verbeterd
2.2	03-03-2015	Verwerking feedback Leon <ul style="list-style-type: none"> - Indicatoren toegevoegd - Wijziging volgorde hoofdstukken - Kleine aanpassingen in verschillende hoofdstukken
3	04-03-2015	Eindredactie

Inhoud

Versiebeheer	2
Inleiding.....	5
1 Context.....	6
1.1 Triott Group.....	6
1.2 Inteqnion BV	6
1.3 PLC/SCADA groep	7
2 De projectdefinitie	8
2.1 Achtergrond	8
2.2 Software	9
2.3 Complexiteit	10
2.4 Producten	10
2.5 Doelstelling.....	11
2.6 Randvoorwaarden.....	11
3 Methode en technieken.....	12
3.1 Projectmanagementmethode.....	12
3.1.1 Afweging methode.....	14
3.1.2 Plannen	15
3.2 Technieken	16
3.2.1 Analyse technieken	16
3.2.2 Realisatie/Ontwerp technieken	17
3.2.3 Test technieken	17
4 Globale projectaanpak.....	18
4.1.1 Inceptiefase.....	18
4.1.2 Elaboratiefase	18
4.1.3 Constructiefase	18
4.1.4 Transitiefase.....	19
5 Tussenresultaten.....	20
6 Fasering en planning	22

6.1	Iteraties	22
6.2	Globale activiteiten en planning	22
6.2.1	Iteratie 1.....	23
6.2.2	Iteratie 2.....	23
6.2.3	Iteratie 3.....	24
6.2.4	Iteratie 4.....	24
6.2.5	Iteratie 5.....	25
6.2.6	Iteratie 6.....	25
6.2.7	Iteratie 7.....	26
7	Projectorganisatie	27
7.1	Opdrachtgever	27
7.2	Opdrachtnemer/student.....	27
7.3	Bedrijfsbegeleider	27
7.4	Stage docent.....	27
8	Communicatie	28
8.1	Communicatie met de bedrijfsbegeleider	28
8.2	Communicatie met de opdrachtgever	28
8.3	Communicatie stagedocent	28
9	Kwaliteit	29
9.1	Kwaliteitsrichtlijnen.....	29
9.2	Definition of done	30
10	Risico's.....	32
11	Competenties.....	33
11.1	Indicatoren	33
12	Bibliografie	35

Inleiding

Dit is het plan van aanpak voor de stage van Michiel Buevink en het daarbij horende project. Het project heeft de naam LIBOT wat staat voor Level Indication Based On Temperature. De stage wordt uitgevoerd bij Inteqnion BV in Aalten, een bedrijf dat gespecialiseerd is in besturings- en automatiseringsoplossingen.

Kort samengevat houdt het project in dat er een applicatie gerealiseerd moet worden dat op basis van temperatuurmetingen in grondstofsilo's een indicatie kan geven van hoeveel er in zit. Dit project wordt gerealiseerd tijdens de 20 weken durende stageperiode van 2 februari tot 26 juni.

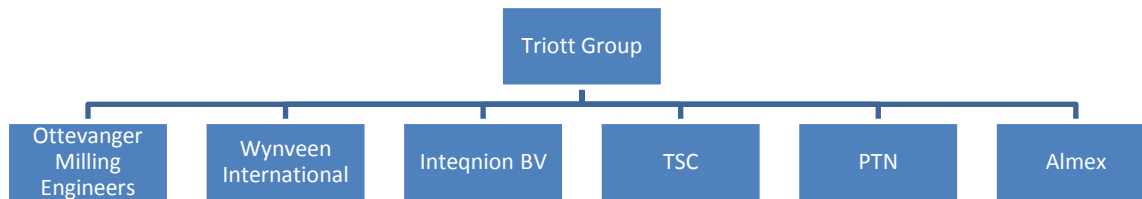
In het plan van aanpak is te lezen hoe de stage opgebouwd gaat worden en wat er gedaan wordt om de stage goed af te ronden. Dit is gedaan door het project af te bakenen en de risico's in kaart te brengen. Daarnaast is in dit document vastgelegd hoe er voor gezorgd wordt dat het project opgeleverd wordt naar de wens van de opdrachtgever en wat de planning is. Wie de opdrachtgever is en een overzicht van de overige rollen is te vinden in hoofdstuk 7.

1 Context

De stage wordt uitgevoerd bij Inteqnion BV. Hieronder is te vinden wat voor bedrijf Inteqnion BV is en wat het doet.

1.1 Triott Group

Inteqnion BV is een onderdeel van de Triott Group. De Triott Group is een holding maatschappij met een zestal werkmaatschappijen: Ottevanger Milling Engineers, Wynveen International, Inteqnion BV, TSC, PTN en Almex.



Afb. 1 Organogram van de Triott Group

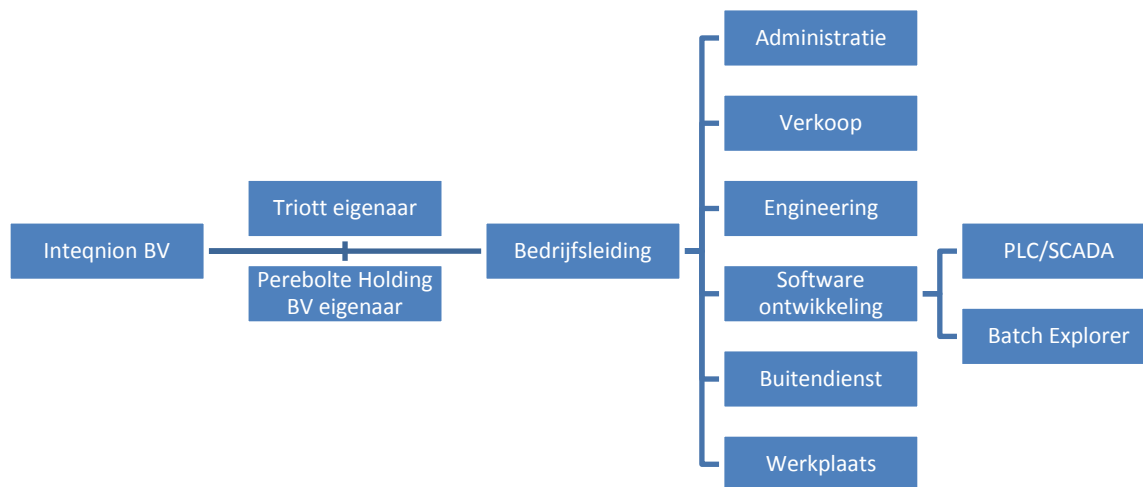
Elke werkmaatschappij is gespecialiseerd in één of meer aspecten die een rol spelen bij de ontwerp en bouw van fabrieken voor de productie van diervoeders.

Hierbij valt te denken aan silo's, machines voor de productie, behandeling en verpakking van diervoeders en de automatisering van de fabriek. Tevens is de Triott Groep actief in de bio-energie sector. (Inteqnion BV)

1.2 Inteqnion BV

Inteqnion BV is het onderdeel van de Triott Group dat gespecialiseerd is in het ontwerp, de bouw en de implementatie van besturings- en automatiseringsoplossingen. Wat inhoudt dat Inteqnion BV voor zijn klanten de besturing van de fabriek realiseert. Hierbij gebruikt Inteqnion BV PLC's voor het Proces Control System en SCADA voor de procesvisualisatie. Daarnaast heeft Inteqnion BV Batch Explorer voor het beheer van de batch processen. Inteqnion BV is gevestigd in Aalten en heeft ongeveer 30 medewerkers.

Op volgende pagina is de organisatiestructuur van Inteqnion BV te zien. De software ontwikkel afdeling bestaat uit twee groepen. Een groep die verantwoordelijk is voor Batch Explorer en een groep die verantwoordelijk is voor de PLC/SCADA software.



Afb. 2 Organogram van Inteqnion BV

1.3 PLC/SCADA groep

De opdracht wordt uitgevoerd bij de PLC/SCADA groep van Inteqnion BV. Hier werken zeven werknemers om de PLC en SCADA software te realiseren, te onderhouden en het geven van support.

De stage wordt op de PLC/SCADA afdeling uitgevoerd en niet op de Batch Explorer afdeling omdat de applicatie die ontwikkeld zal worden een uitbreiding is naast de SCADA software. Daarnaast is informatie nodig die beschikbaar is in de SCADA software.



Afb. 3 De SCADA software, foto van de software in gebruik en Batch Explorer

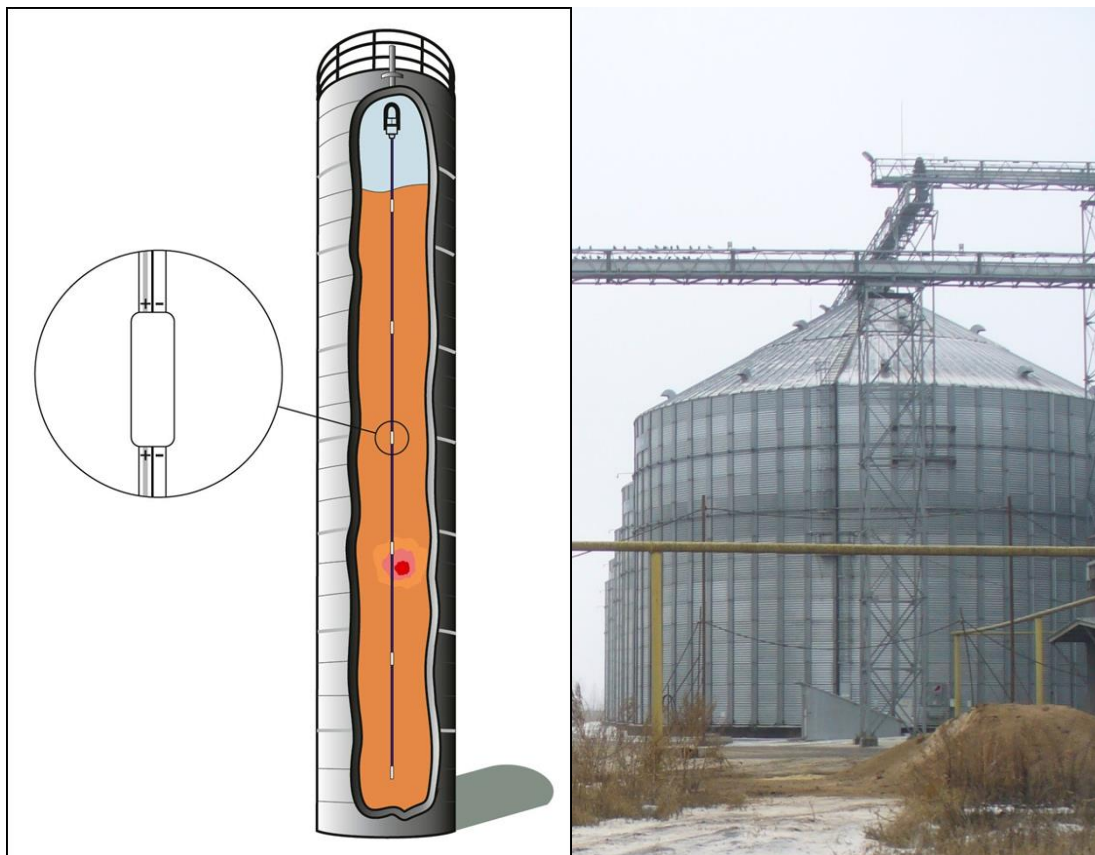
2 De projectdefinitie

Hieronder zijn de aanleiding van het project, de doelstelling van de opdrachtgever en de randvoorwaarden te lezen. Ook wordt er in dit hoofdstuk aandacht besteed aan de punten die het project complex maken.

2.1 Achtergrond

Bij de productie van mengvoeder worden veel grondstoffen in grote silo's opgeslagen. Deze silo's bevatten temperatuursensoren voor de detectie van broei. Die sensoren zitten aan strengen die in de silo's hangen zoals te zien in de afbeelding hieronder. De silo's kunnen meerdere strengen hebben of maar één streng, dit is afhankelijk van de omvang van de silo's. Ook het aantal sensoren en de afstand tussen de sensoren kan variëren.

Op dit moment wordt in die silo's gekeken naar hoeveel grondstof er wordt gestort in de silo en hoeveel eruit wordt gehaald. Echter kan de afname niet exact bepaald worden omdat de afname niet gewogen wordt. Naar deze waarde wordt verwezen als de berekende waarde.



Afb. 4 Schematische weergave van een silo met sensoren (Grain-Watch® Level Indication) en een afbeelding van een silo

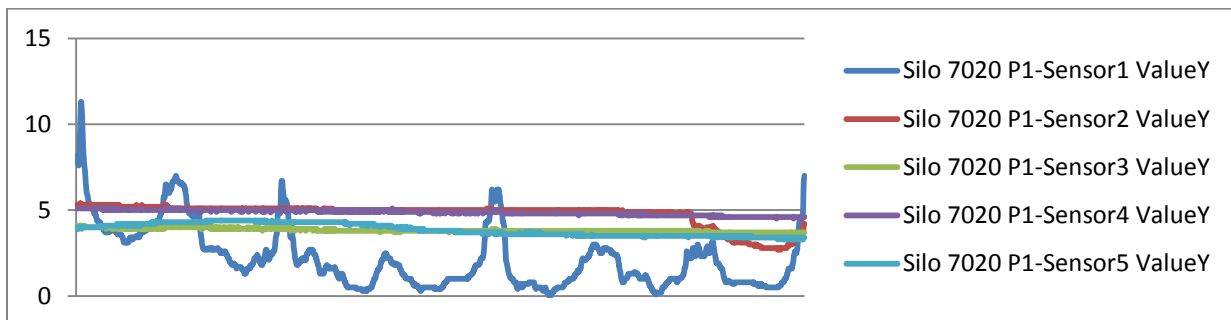
2.2 Software

Een klant van Inteqnion BV is geïnteresseerd in een aanvullend systeem. Met dat systeem moet de inhoud geschat worden op basis van de metingen met de temperatuursensoren. Hiervoor bestaat al wel software die niet door Inteqnion BV zelf is ontwikkeld maar door de leverancier van de sensoren.

Dit systeem zal als losstaande applicatie worden gerealiseerd. De applicatie moet de temperatuurgegevens uit een database halen die er door de SCADA software is ingezet, analyseren, verwerken en vervolgens de resultaten weergeven en in de database plaatsen. Per silo moet weergegeven worden hoeveel dat er volgens de schatting inzit. Daarnaast moet er moet een grafiek getoond kunnen worden met de temperaturen, de bepaalde (op temperatuurbasis) en berekende (op basis van in en uitvoer) inhoud. Het moet in de applicatie mogelijk zijn om silo's toe te voegen en te configureren zodat er met verschillende silo's en sensoropstellingen gewerkt kan worden.

Het is wenselijk dat het bepalen van de inhoud zo nauwkeurig mogelijk gebeurt. De exacte nauwkeurigheid is afhankelijk van verschillende factoren zoals aantal sensoren, aantal strengen en grootte van de temperatuurfluctuaties buiten. Gedurende het project moet op basis van data van een andere fabriek bepaald worden hoe nauwkeurig dat het systeem kan werken. Het systeem van de sensorleverancier heeft de temperatuurgegevens van vier dagen nodig om de inhoud te bepalen, in dit project wordt hier ook naar gestreefd.

Om aan de hand van een temperatuursensor te bepalen of er op die hoogte een grondstof zit kan er gekeken worden naar veranderingen in de temperatuur. De buitentemperatuur zal er namelijk voor zorgen dat de temperatuur in de silo verandert. Grondstof heeft een isolerende werking waardoor sensoren die zich in de grondstof bevinden langzamer veranderen en minder effect van de buitentemperatuur zullen waarnemen. In de grafiek hieronder is dat effect duidelijk te zien.



Afb. 5 Meetresultaten van zeven dagen met de sensoren in een silo die buiten staat

2.3 Complexiteit

De complexiteit van het project wordt voor een deel bepaald door het algoritme wat de inhoud schat. Dit algoritme moet namelijk rekening houden met een groot aantal factoren, zo kunnen silo's variëren in formaat, aantal strengen, aantal sensoren en afstand tussen de sensoren. Daarbij geeft het gebruikmaken van de temperatuurverschillen van dag en nacht nog extra problemen. Zo kunnen de verschillen tussen dag en nacht klein worden of juist heel groot waardoor de fluctuatie in de delen met grondstof ook klein of groot zijn. Per grondstof kan het ook nog verschillen hoe snel het product reageert op temperatuursveranderingen.

Verder kunnen sensorwaarden ook onderling afwijken, wanneer er grondstoffen worden toegevoegd kunnen die namelijk een afwijkende temperatuur hebben. In een deel van de silo kan ook broei ontstaan hierbij zullen de temperaturen ook afwijken van wat normaal verwacht wordt.

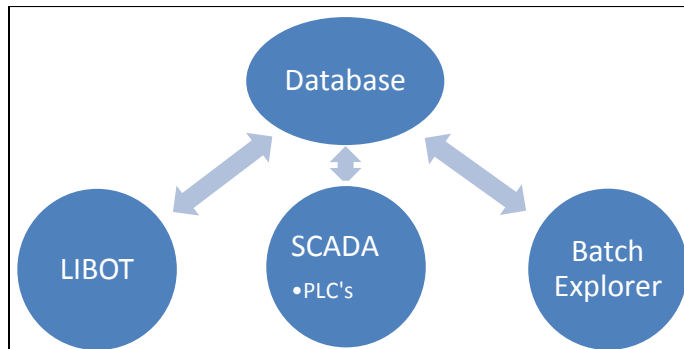
Een andere punt wat de complexiteit verhoogd is het data protocol. Iedere minuut zullen de waardes van de temperatuurmeting in SCADA worden opgeslagen naar de database. Deze data wordt dan iedere minuut opgehaald uit de database en vervolgens in de software verwerkt, hiervoor zijn aparte threads of processen nodig om dit niet te laten storen met de activiteiten in de GUI.

In de GUI zelf zit ook nog een complex punt. Doordat er veel met schematische weergaves gewerkt zal worden, zal ook in de weergave rekening gehouden moeten worden met de verschillende soorten en aantallen silo's. Hiervoor moet de GUI goed schalen.

2.4 Producten

De applicatie is niet het enige product dat opgeleverd zal worden. Zo zal er een rapport opgesteld worden van de gemaakte analyses en een rapport voor de gemaakte ontwerpen. Daarnaast zullen er database tabellen en stored procedures gemaakt worden voor het opslaan en ophalen van gegevens.

In hoofdstuk 5 is verder te lezen wat deze producten inhouden en welke producten er nog bij komen vanuit de HAN.



Afb. 6 Communicatie tussen de verschillende software

2.5 Doelstelling

Het ontwikkelen van een applicatie die de inhoud van een silo kan bepalen op basis van de temperatuurmetingen in de silo's.

2.6 Randvoorwaarden

Vanuit Inteqnion BV is het van belang dat er vanuit dezelfde database wordt gewerkt als waarmee de SCADA software werkt. Op die manier kunnen de temperatuurgegevens makkelijk gelogd worden vanuit de SCADA software. De gebruikte database is een Microsoft SQL database.

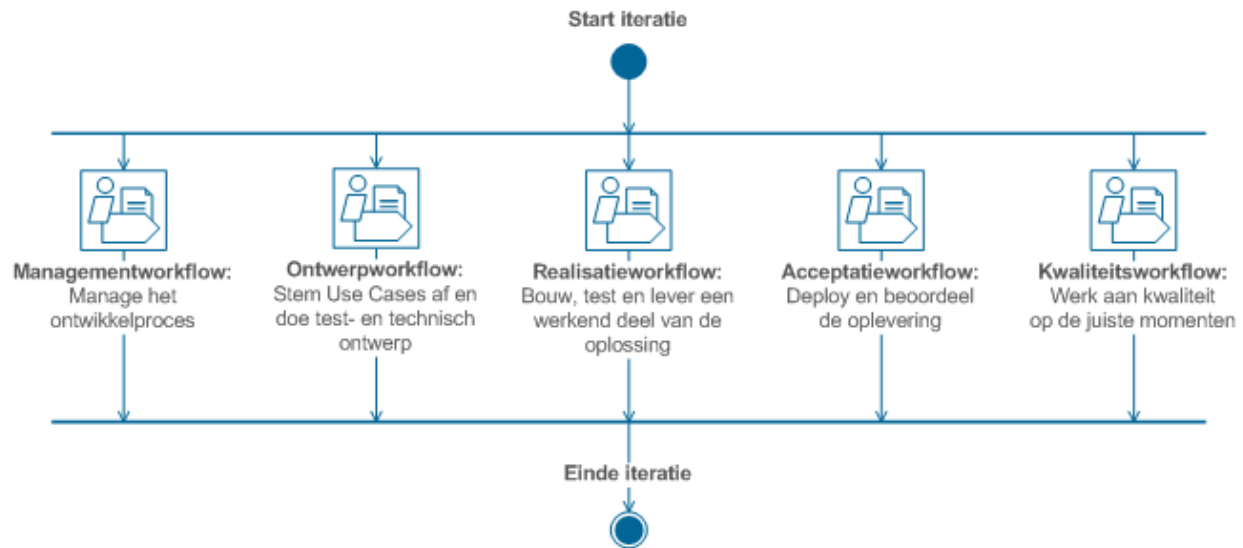
Daarnaast wordt er ontwikkeld in C#. De Batch Explorer groep maakt hier ook gebruik van, hierdoor is er van kennis van C# binnen het bedrijf wat het met mogelijk maakt om de codekwaliteit te controleren

3 Methode en technieken

Om het project in goede banen te leiden wordt gebruik gemaakt van een projectmanagement methode. Deze methode stelt vast hoe in het project gewerkt en gepland gaat worden. Daarnaast worden een aantal technieken die tijdens het project gebruikt worden vastgelegd.

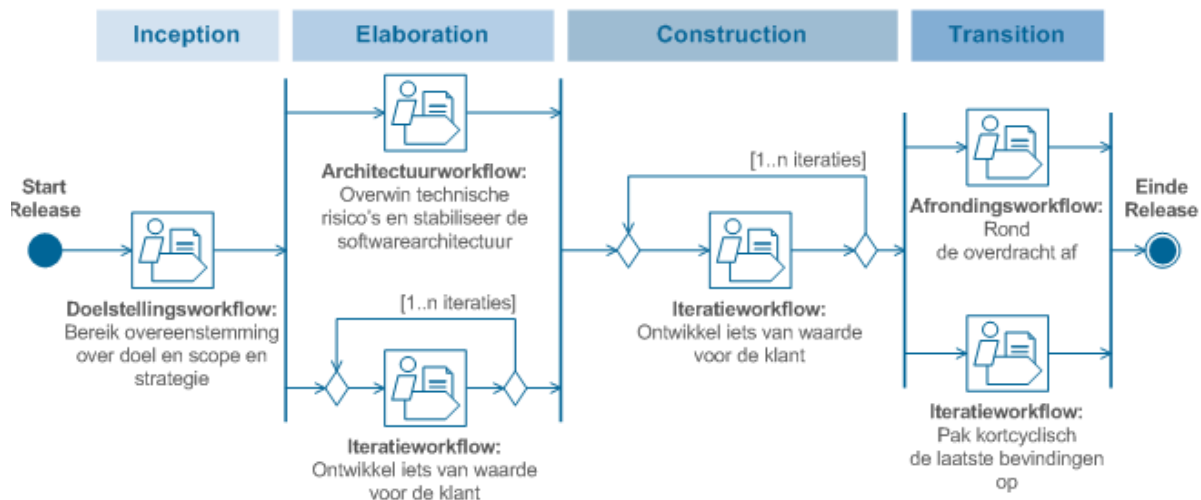
3.1 Projectmanagementmethode

Als projectmanagementmethode is gekozen voor RUP. RUP is een iteratieve methode, wat inhoudt dat gedurende een iteratie de stappen plannen, analyseren, ontwerpen, realiseren, testen en evalueren worden uitgevoerd zoals te zien in de afbeelding hieronder.



Afb. 7 Iteratief ontwikkel model (Collaris & Dekker)

Voor dit project is er gekozen voor iteraties van drie weken. Aan het einde van een iteratie is er dan een werkend prototype met een deel van de functionaliteit van het eindproduct. Het prototype wordt vervolgens getoond aan de opdrachtgever. Het is dan voor de opdrachtgever mogelijk om na iedere iteratie het project bij te sturen.



Afb. 8 De fasen binnen RUP (Collaris & Dekker)

RUP deelt het project op in vier fasen, Inception, Elaboration, Construction en Transition zoals hierboven te zien is. De fasen houden het volgende in:

“Inceptiefase (Aanvang) - De haalbaarheid van het project, de inhoud (scope) en de begrenzungen worden bepaald. Gedurende de inceptiefase wordt het oorspronkelijke idee omgezet in een productvisie (vision). De business drivers worden geëvalueerd om ze helder te krijgen. De globale kostprijs en de verwachte baten van het project worden geschat. De belangrijkste risico's worden geïdentificeerd en ingeschat. Het uiteindelijke doel van deze fase is een haalbaarheidsstudie (business case) voor het project, of anders gesteld een GO/NOGO voor het project.

Elaboratiefase (Detaillering) – Het merendeel van de functionele requirements (use cases) wordt gespecificeerd en de systeemarchitectuur wordt ontworpen. De nadruk ligt op de technische haalbaarheid van het project. Uiteindelijke doel van deze fase is een projectplan, waarin onder meer de gedetailleerde inhoud (scope), timing (schedule) en kostenraming (estimations) voor het project zijn opgenomen.

Constructiefase (Bouw) – Het product wordt ontwikkeld vanaf de architectuur tot een systeem dat compleet genoeg is om te testen.

Transitiefase (Overgang) – Via testen wordt het product gevalideerd door de belanghebbenden. Andere activiteiten in deze fase zijn: voorbereiding en inproductiename (deployment), nazorg, en overdracht van de verantwoordelijkheden. Deze fase wordt afgesloten met een inventarisatie van de opgedane ervaringen (lessons learned) voor volgende projecten.”
(Wikipedia)

3.1.1 Afweging methode

Bij de afweging welk soort methode (waterval, incrementeel of evolutionair) moest rekening gehouden met een aantal factoren:

- Elke week wordt nieuwe data aangeleverd om mee te testen, daardoor kan het iedere iteratie mogelijk zijn dat het algoritme aangepast moet worden. De reden dat er iedere week nieuwe data beschikbaar komt is dat er getest wordt met exports van de temperatuur historie van een bestaande fabriek. Die historie wordt maar een week bewaard dus het is niet mogelijk de data in één keer te krijgen.
- Mogelijk komt er een mogelijkheid om een prototype in de praktijk te testen.
- Er kunnen nog geen definitieve eisen worden vastgelegd omdat de opdrachtgever nog niet alle eisen duidelijk heeft.



Afb. 9 Silo's gebruik voor de testdata

Door deze factoren valt de watervalmethode direct af omdat daarbij alle eisen moeten worden vastgesteld aan het begin van het project en er pas een prototype beschikbaar is aan het einde van het project. Het algoritme moet mogelijk iedere keer wanneer er nieuwe testdata beschikbaar komt worden aangepast, hiervoor is een incrementele methode niet praktisch. Het project is te groot om volledig evolutionair uit te voeren met één persoon. Daarom is een combinatie van een incrementele en evolutionaire methode de beste keuze.

Mogelijke keuzes zijn dan: RUP, extreme programming, Scrum en EVO. Hierbij is EVO vooral gericht op korte iteraties met weinig documentatie. Doordat dit project met maar één persoon wordt gerealiseerd heeft het project baat bij langere iteraties. Ook documentatie is van belang

voor het project omdat documentatie als bewijslast kan gelden voor de stage. Dan kan de software ook gemakkelijk door een derde worden begrepen en gecontroleerd.

Extreme programming en Scrum zijn gefocust op het optimaal presteren in een teamverband doormiddel van onder andere daily stand-ups, pair programming en retrospectives. Hierdoor vallen de voordelen van Scrum weg bij de voordelen van RUP. Hier komt bij dat RUP veel vastgelegd heeft wat betreft documentatie en er per iteratie gepland wordt. RUP heeft wel net als alle andere evolutionaire methoden als nadeel dat het gevoelig is voor uitloop.

3.1.2 Plannen

Elke iteratie wordt er in overleg met de opdrachtgever bepaald welke functies verbeterd en/of toegevoegd worden. De functies worden onderverdeeld in kleine taken die op YouTrack worden geplaatst. YouTrack is een online ticket systeem en planbord waardoor er een overzicht ontstaat van de voortgang van het project.

Er is voor YouTrack gekozen omdat dit systeem door Inteqnion BV wordt gebruikt en uitgebreider is dan de vrij beschikbare systemen.

3.2 Technieken

Tijdens het project zal gebruik worden gemaakt van de onderstaande technieken. Er is onderscheid gemaakt tussen de verschillende soorten technieken om een idee te geven waarbij die technieken zullen worden toegepast.

3.2.1 Analyse technieken

Voor het maken van de analyse zal onder andere gebruik worden gemaakt van de onderstaande technieken.

FURPS+

De eisen worden volgens FURPS+ opgesteld, wat inhoud dat de eisen worden onderscheiden de volgende delen:

- Functionality
- Usability
- Reliability
- Performance
- Supportability
- Ontwerp eisen
- Implementatie eisen
- Interface eisen

Op die manier zijn eisen overzichtelijk gecategoriseerd en kan lezer makkelijk kijken naar de voor hem relevante eisen. (Eeles, 2005)

Use-cases

Een use-case wordt door middel van een brief format en het twee-kolomsformat omschreven. Op die manier wordt op systematische wijze geanalyseerd welke gebruikssituaties in de software gerealiseerd moet worden. Voor de opdrachtgever is het brief format interessant omdat hij dan kan lezen hoe dat de opdrachtnemer het gebruik van de software voor zich ziet. Het twee-kolomsformat is bij het uitwerken van de use-case handig om zo de goede “flows” in het programma te verwerken (Larmann, 2004).

3.2.2 Realisatie/Ontwerp technieken

Doormiddel van de onderstaande ontwerptechnieken wordt gestreefd naar een duidelijk en bruikbaar ontwerp.

Class diagram

Een class diagram is een middel om alle classes die binnen de software gebruikt gaan worden vooraf vast te leggen. Het class diagram is ook van belang bij gebruik van publieke interfaces, op die manier zijn de interfaces vooraf al vastgelegd. Ook wordt in een class diagram vastgelegd hoe de classes zich tot elkaar verhouden en welke (publieke) functies en variabelen en class bevat. (Larmann, 2004)

Design patterns

Voor veel problemen bij het object georiënteerd programmeren zijn er al oplossingen bedacht door onderzoekers. De 23 Gang of Four patterns worden gezien als de basis voor alle design patterns (.NET Design Patterns). Hierom wordt daar waar mogelijk geprobeerd deze patronen te gebruiken. Daarnaast heeft Microsoft zijn eigen visie op verschillende design patterns. Omdat die van Microsoft gericht zijn op C# zullen ook de patronen van Microsoft in overweging worden genomen.

3.2.3 Test technieken

Om te controleren dat de applicatie naar behoren werkt moeten tests worden uitgevoerd. Hieronder staat hoe dat het programma getest gaat worden.

Unit testing

Unit tests zijn tests waarbij gecontroleerd wordt of een functie in de applicatie bij een bepaalde in de juiste waarden terug geeft. Op die manier kan gecontroleerd worden of de applicatie bij bepaalde metingen de correcte uitvoer terug geeft.

Proefdraaien

Er wordt gekeken of er een mogelijkheid gecreëerd worden waarbij de applicatie voor een bepaalde tijd kan draaien in een fabriek. Dan kan aan de hand van de grafieken gecontroleerd worden of de applicatie de correcte uitvoer berekend.

4 Globale projectaanpak

Volgens het idee van RUP wordt het project opgedeeld in vier fasen, hieronder wordt per fase toegelicht hoe die ingericht zal worden. Wanneer de eindproducten van een fase klaar zijn maar gewacht moet worden op feedback zal er alvast begonnen worden aan de volgende fase.

Het project wordt opgebouwd uit iteraties van drie weken, op die manier is er voldoende tijd om toonbare producten te realiseren. De Inceptiefase en de Elaboratiefase worden in de eerste iteratie van het project afgerond. De Transitiefase wordt tijdens de laatste iteratie uitgevoerd. De overige iteraties vallen onder de Constructiefase.

4.1.1 Inceptiefase

In deze fase wordt de context van het project duidelijk gemaakt en wordt er gekeken naar hoe de opdracht er globaal uit komt te zien. Daarnaast worden de projectgrenzen vast gelegd. Tijdens deze fase zal de opdrachtschrijving worden gemaakt en wordt het plan van aanpak opgesteld. De fase is afgerond als de opdrachtschrijving en het plan van aanpak zijn goedgekeurd. In deze iteratie wordt ook al een globale planning voor het project gemaakt. De risico analyse is in het plan van aanpak al opgesteld dus zal niet voor iedere iteratie een risico analyse worden gemaakt zoals dit bij RUP gebruikelijk is.

4.1.2 Elaboratiefase

In deze fase wordt de basis gelegd voor het project. Er wordt geanalyseerd wat de opdracht exact inhoud en de eisen van het project worden vastgesteld. Deze eisen worden aan de opdrachtgever gepresenteerd voor goedkeuring. Verder worden de meeste use-cases zoals die dan bekend zijn uitgewerkt.

Wanneer dat gerealiseerd is, wordt een begin gemaakt met het ontwerpen van de systeem architectuur. Dit wordt de basis van het class diagram en het project. Voor de opdrachtgever zullen ook schermontwerpen worden gemaakt om met de opdrachtgever af te stemmen hoe de software eruit moet komen te zien.

Deze fase moet aan het einde van de eerste drie weken afgerond zijn.

4.1.3 Constructiefase

Met ingang van de tweede iteratie wordt er begonnen aan de constructie fase. Iedere iteratie begint met het opstellen van een planning voor de iteratie. In die planning wordt bepaald welke functionaliteit uitgewerkt zal worden en zal deze opgedeeld worden in taken die verdeeld worden over de beschikbare dagen. Na het plannen wordt er indien nodig gewerkt aan het bijwerken van de analyses. Dit wordt opgevolgd door het uitbreiden van het ontwerp om de geplande functionaliteit uit te werken of te verbeteren. Daarna wordt er begonnen met het

realiseren en testen. Na het realiseren wordt het ontwerp bijgewerkt zodat er geen verschillen meer zijn tussen ontwerp en realisatie.

Dan wordt een acceptatietest uitgevoerd door de resultaten van de iteratie te tonen aan de opdrachtgever. Aan het einde van een iteratie wordt de iteratie geëvalueerd om gemaakte fouten in de toekomst te voorkomen.

Tijdens de constructiefase zal tussen de werkzaamheden door ook gewerkt worden aan het stageverslag zodat die ook voor feedback kan worden opgeleverd.

4.1.4 Transitiefase

De laatste iteratie is de Transitiefase. In deze iteratie zal er geen nieuwe functionaliteit worden gerealiseerd. Er wordt enkel gefocust op het oplever klaar maken van de producten en het voorbereiden van de presentatie.

5 Tussenresultaten

Hieronder staan de mijlpaalproducten die horen bij het project met een korte beschrijving van het doel en de inhoud.

Analyserapport

In dit rapport zijn de resultaten van de uitgevoerde analyses terug te vinden om er voor te zorgen dat de gewenste functionaliteit van de applicatie en uitgevoerde onderzoeken zijn vast gelegd.

De globale inhoudsopgave ziet er zo uit:

- Eisen
- Use-cases
- Analyse temperatuurdata
- Analyse configuraties silo's

Ontwerprapport

Dit rapport bevat de ontwerpen van de software, aan de hand van wat in dit rapport staat worden de database en applicatie gerealiseerd.

De globale inhoudsopgave ziet er zo uit:

- Globale beschrijving werking algoritme
- Schermontwerpen
- Class diagram software
- Database ontwerp
 - Tabel temperatuur invoer
 - Tabel resultaten uitvoer
 - Tabel configuratie

Applicatie

De applicatie die de inhoud van silo's bepaald op basis van de resultaten van de temperatuurmetingen.

De applicatie bevat globaal de volgende onderdelen

- Algoritme
- Overzicht resultaten
- Grafiek van de resultaten van een silo
- Configuratiescherm

Database

De database is de interface tussen de SCADA software van Inteqnion BV en de te ontwikkelen applicatie. Daarnaast wordt de database gebruikt voor het opslaan van de configuraties van de silo's.

De database bevat globaal de volgende onderdelen

- Tabellen en stored procedures voor de configuratie
- Tabel en stored procedures voor resultaten
- Tabel en stored procedures voor temperatuur invoer

Stageverslag

Een lezer met enige deskundigheid op het vakgebied moet een voldoende diepgaand totaal beeld krijgen van de inhoud, aanpak, verloop en resultaat van het project.

De globale inhoudsopgave ziet er zo uit:

- Managementsamenvatting
- Bedrijfsbeschrijving/context
- Opdrachtdefinitie
- Aanpak
- Proces en resultaten
- Conclusie

Evaluatieverslag

In het evaluatieverslag valt te lezen in hoeverre het stage succesvol is verlopen. Daarnaast wordt er overdacht en teruggekeken hoe de persoonlijke ontwikkeling van de stagiair is verlopen.

De globale inhoudsopgave ziet er zo uit:

- Projectevaluatie (professionele reflectie)
 - Domeincompetenties
- Persoonlijke reflectie

Eindpresentatie

De eindpresentatie is er om de uitkomst van het project te presenteren.

De voorbereiding voor de eindpresentatie bevat de volgende onderdelen:

- Spreekschema
- Hulpmiddelen

6 Fasering en planning

In dit hoofdstuk is een overzicht te vinden van alle activiteiten die dit project uitgevoerd dienen te worden en een globale planning.

6.1 Iteraties

Er wordt gebruik gemaakt van iteraties van drie weken, dat geeft de volgende verdeling van de iteraties. Voor de invulling van de iteraties zie paragraaf 6.2.1.

Stage weken	Eerste dag	Fase	Iteratie
1,2,3	2 februari 2015	Inceptiefase en elaboratiefase	1
4,5,6	23 februari 2015	Constructiefase	2
7,8,9	16 maart 2015	Constructiefase	3
10, 11, 12	6 april 2015	Constructiefase	4
13,14, 15	27 april 2015	Constructiefase	5
16,17,18	18 mei 2015	Constructiefase	6
19,20	8 juni 2015	Transitiefase	7

6.2 Globale activiteiten en planning

Omdat bij RUP niet met de watervalmethode wordt gewerkt worden producten niet in één keer afgerond maar worden de producten iedere iteratie uitgebreid en daar waar nodig aangepast (incrementeel/evolutionair). Dit zorgt er voor dat het definitieve oplevermoment dus aan het einde van het project ligt voor de meeste producten. Uitzonderingen hierop zijn de opdrachtschrijving en het plan van aanpak, die moeten namelijk al in een vroeg stadium worden goedgekeurd.

In de opsomming hieronder is te zien dat het project is opgedeeld in vijf hoofdfuncties die verdeeld zijn onder de constructie iteraties. Per functie is er beschreven aan welk product er gewerkt wordt en wat er globaal de activiteiten zijn. Ook staat er aangegeven in welke iteratie er verwacht wordt te gaan werken aan een functie. Tijdens een iteratie wordt er niet alle gewerkt aan de nieuwe functie maar zullen de functies uit de voorgaande iteraties ook weer terug komen om te verbeteren aan de hand van de wensen van de opdrachtgever. Dit is echter afhankelijk van hoeveel feedback de opdrachtgever heeft.

De zesde iteratie is dan ook om uitloop door die feedback op te vangen en zal besteed worden aan het optimaliseren en testen van het algoritme. Iteratie zeven is er om de producten oplever klaar te maken en zal er niet meer gewerkt worden aan uitbreiding van de functionaliteiten van de applicatie.

6.2.1 Iteratie 1

Inception

- Plan van aanpak
 - Opstellen
 - Eindredactie uitvoeren
 - Feedback verwerken
- Opdracht omschrijving
 - Opstellen
 - Eindredactie uitvoeren
 - Feedback verwerken

Elaboration

- Analyserapport
 - Structuur maken
 - Requirements opstellen
 - Opstellen eerste versie use-cases
- Ontwerprapport
 - structuur maken
 - Eerste versie scherm ontwerpen
 - Globale architectuur ontwerpen
- Realisatie
 - Globale architectuur realiseren
- Stageverslag
 - Documentstructuur opzetten

6.2.2 Iteratie 2

Construction Algoritme

- Analyserapport
 - Analyseren Temperatuurdatabank
- Ontwerprapport
 - Tussentabel applicatie en SCADA ontwerpen
 - Class diagram bijwerken
- Realiseren
 - Algoritme realiseren
 - Tussentabel realiseren
 - Stored procedures voor tussen tabel realiseren
- Testdata inladen

Stageverslag

- Bedrijfsbeschrijving opstellen
- Opdrachtdefinitie opstellen

6.2.3 Iteratie 3

Construction overzichtsscherm

- Analyserapport
 - Use-case bijwerken
- Ontwerprapport
 - Schermontwerpen bijwerken
 - Class diagram uitbreiden
- Realiseren
 - Scherm
 - Stored procedure voor ophalen data

Stageverslag

- Aanpak beschrijven

6.2.4 Iteratie 4

Construction output opslaan in database

- Ontwerprapport
 - Output tabel
 - Class diagram uitbreiden
- Realiseren
 - Output tabel
 - Stored procedure voor opslaan
 - Realiseren wegschrijven

Construction grafiek scherm

- Analyserapport
 - Use-case bijwerken
- Ontwerprapport
 - Scherm ontwerp bijwerken
 - Class diagram uitbreiden
- Realiseren
 - Scherm
 - Stored procedure voor ophalen data

6.2.5 Iteratie 5

Construction configuratiescherm

- Analyserapport
 - Use-case bijwerken
 - Analyse configuraties silo's
- Ontwerprapport
 - Class diagram
 - Opslag van configuraties
 - Scherm ontwerp bijwerken
- Realiseren
 - Opslag van configuraties
 - Scherm
 - Aanpassen algoritme

Stageverslag

- Proces en resultaten beschrijven

6.2.6 Iteratie 6

Construction evolutie algoritme

- Testen met alle testdata
- Ontwerprapport
 - Class diagram bijwerken
- Realiseren
 - Algoritme updaten

Stageverslag

- Proces en resultaten uitbreiden
- Conclusie schrijven

Evaluatieverslag

- Project evaluatie opstellen

6.2.7 Iteratie 7

Transitie

- Analyserapport
 - Use-cases controleren
- Ontwerprapport
 - Ontwerpen bijwerken
- Realisatie
 - Code controleren
 - Database controleren
- Stage verslag
 - Eindredactie
- Presentatie Voorbereiden
- Evaluatieverslag
 - Persoonlijke reflectie opstellen
 - Eindredactie

7 Projectorganisatie

In dit hoofdstuk staan alle betrokkenen bij het project, zoals de opdrachtgever, opdrachtnemer en de partijen die vanuit de HAN betrokkenheid hebben bij dit project.

7.1 Opdrachtgever

Vanuit Inteqnion BV functioneert Charl van Kesteren als opdrachtgever. Hij is regelmatig aanwezig en beschikbaar voor vragen. Voor vragen over de opdracht kan ook bedrijfsbegeleider Mark Smeenk worden aangesproken.

Contactgegevens

Charl van Kesteren

c.vanKesteren@inteqnion.com

7.2 Opdrachtnemer/student

De opdrachtnemer en stagiair is Michiel Buevink, student Informatica aan de Hogeschool van Arnhem en Nijmegen. De student is verantwoordelijk voor de uitvoer van de opdracht en de communicatie met de bedrijfsbegeleider en stage docent.

Contactgegevens

Michiel Buevink

m.buevink@inteqnion.com

0625515822

7.3 Bedrijfsbegeleider

De bedrijfsbegeleider functioneert als begeleider voor de student en is regelmatig beschikbaar voor overleg. Daarnaast beoordeelt hij de student doormiddel van het beoordelingsformulier.

Contactpersoon

Mark Smeenk

m.smeenk@inteqnion.com

7.4 Stage docent

De stagedocent begeleidt de student, bewaakt de voortgang en beoordeelt het niveau van de producten. Daarnaast is hij eindbeoordelaar van de stage.

Contactgegevens

Leon van Houten

Leon.Vanhouten@han.nl

8 Communicatie

Tussen de verschillende partijen wordt op de onderstaande manieren gecommuniceerd.

8.1 Communicatie met de bedrijfsbegeleider

De bedrijfsbegeleider is dinsdag t/m vrijdag aanwezig op de afdeling, de communicatie zal dus veelal mondeling gaan. Indien nodig kan de begeleider ook via Skype of telefonisch bereikt worden.

Iedere vrijdag is er een gesprek over de voortgang van het project en het functioneren van de stagiair. Daarnaast zullen beide partijen op iedere eerste vrijdag van de maand het Feedbackformulier (te vinden op <https://online.han.nl/sites/ICA-Trainingen/stagenw/Stage%20algemeen/Feedbackformulier%20bedrijfsbegeleider.docx0>) invullen en bespreken. Op die manier kan er gekeken worden of de student naar behoren functioneert en wat de student zelf van zijn functioneren vindt.

8.2 Communicatie met de opdrachtgever

Vragen over de opdracht kunnen eerst gesteld worden aan de bedrijfsbegeleider. Wanneer hij het antwoord niet weet kan de vraag gesteld worden aan de opdrachtgever. De opdrachtgever is aanwezig bij Inteqnion BV en bij vragen kunnen die dus mondeling gecommuniceerd worden maar indien nodig is het ook via e-mail mogelijk.

Aan het einde van een iteratie zullen de resultaten van de iteratie aan de opdrachtgever worden getoond voor feedback of goedkeuring. Wanneer het product uitgebreid behandeld moeten worden zoals een verslag vooraf via e-mail een met het product gestuurd. Op die manier kan de opdrachtgever de producten eerst doornemen.

8.3 Communicatie stagedocent

De communicatie met de stagedocent gaat via e-mail. De student stuurt iedere twee weken een voortgangsrapportage. Daarnaast komt de stagedocent in de week van 2 maart een keer langs op het bedrijf, indien nodig zal dit in april of mei nog een keer gebeuren. Verder stuurt de student het stageverslag en de eindproducten voor beoordeling naar de stagedocent. Bij vragen over andere producten kan de student ook contact opnemen met de stagedocent.

9 Kwaliteit

In dit hoofdstuk is te lezen welke maatregelen er zijn genomen om de kwaliteit van het project te waarborgen.

9.1 Kwaliteitsrichtlijnen

Om de kwaliteit van het project te waarborgen zijn de onderstaande richtlijnen vastgesteld.

Documenten

Alle documenten worden van versiebeheer voorzien waar de aanpassingen te zien zijn. Voor de opbouw van documenten is dit document de richtlijn. Documenten zullen door de stagedocent, bedrijfsbegeleider en de opdrachtgever worden beoordeeld en voorzien van eventuele feedback om zo te garanderen dat ze van goede kwaliteit zijn.

Feedback van de opdrachtgever

Aan het einde van iedere iteratie worden de resultaten van de iteratie getoond aan de opdrachtgever voor feedback. Daarnaast zullen alle afgeronde documenten naar de opdrachtgever gestuurd worden voor goedkeuring of feedback. Dit zorgt er voor dat er geen documenten worden opgeleverd waar de opdrachtgever niet mee akkoord is gegaan en heeft de opdrachtgever ruimte voor feedback.

Code standaarden

Voor de ontwikkeling van het C# deel zal net als op de Batch Explorer Afdeling gebruik gemaakt worden van Microsofts C# Coding Conventions als styleguide. Deze zijn te vinden op <http://msdn.microsoft.com/en-us/library/vstudio/ff926074.aspx>. Voor de naamgeving standaard is deze door de Batch Explorer afdeling aangevuld met de volgende standaard: <http://www.dofactory.com/reference/csharp-coding-standards.aspx>. Op deze manier zal er een consistente code stijl zijn.

Code reviews

Aan het einde van iedere iteratie zal alle code gecontroleerd worden of deze voldoet aan de code standaard. Verder zal er gecontroleerd worden of de code is voorzien van commentaar en overeen komt met het ontwerp. Op die manier zal de code altijd aan de standaard voldoen wanneer het getoond wordt aan de opdrachtgever.

Daarnaast zal er een aantal keer samen met een ontwikkelaar van Inteqnion BV een review moment zijn. Die kan dan de kwaliteit van de code beoordelen volgens de normen van Inteqnion BV.

Versiebeheer software

Om te voorkomen dat er werk verloren gaat, zal er voor de software gebruik gemaakt worden van Team Foundation Version Control. Team Foundation is een versiebeheer systeem van Microsoft dat geïntegreerd is in Visual Studio en er voor zorgt dat alle wijzigingen worden bijgehouden. Op die manier kan er tijdens de ontwikkeling altijd terug gegaan worden naar een werkende versie of kan er een specifieke wijziging ongedaan worden gemaakt.

De Team Foundation repository bestaat standaard uit een master branche, hier wordt een Development branche van afgetakt. Hierin staat de versie welke aan het einde van de iteratie wordt opgeleverd. Wanneer er functionaliteit ontwikkelt wordt zal er een aftakking gemaakt worden van de Development branche met een duidelijke naam die aangeeft wat het doel van de branche is. In die branche wordt dan gewerkt tot de functie compleet is en voldoet aan de code standaard en het ontwerp. Dan kan de aftakking samengevoegd worden met de Development branche. Na goedkeuring van de opdrachtgever kan de Development branche samengevoegd worden in de master branche.

Testen

Door de opdrachtgever wordt data ter beschikking gesteld om de software gedurende de ontwikkeling te testen. Er zal gekeken worden naar een mogelijkheid om de software in de praktijk te testen door de applicatie te draaien in een bestaande fabriek en dan te controleren of de uitvoer klopt met de verwachting. Daarnaast zal aan het einde van een iteratie met de opdrachtgever gekeken worden of er aan de functionele eisen is voldaan.

9.2 Definition of done

De definition of done geeft aan wanneer iets af is. Wanneer er niet aan alle punten van de definition of done is voldaan wordt is het product niet af.

Per use-case

Een use-case is af als aan de volgende eisen is voldaan:

- Alle code compileert en is geüpload naar de Team Foundation Server
- De code voldoet aan de code standaard
- De code is voorzien van commentaar
- De code komt overeen met het ontwerp
- De use-case kan worden uitgevoerd, inclusief alle extensies

Per iteratie

Een iteratie is afgerond als aan de volgende eisen is voldaan:

- Het analyserapport is up to date met de laatste informatie van de opdrachtgever

- Het ontwerprapport komt overeen met de gerealiseerde code
- Alle taken behorende bij de iteratie zijn afgerond
- De iteratiedoelen zijn gehaald
- De opdrachtgever is akkoord met het resultaat van de iteratie

10 Risico's

In dit hoofdstuk worden mogelijk risico's beschreven die de uitvoer van kunnen beïnvloeden en waar de stagiair geen invloed op kan uitoefenen. Alleen reële risico's worden hier vermeld.

#	Risico beschrijving	Impact	Kans	Preventieve maatregel	Correctieve maatregel
1	De data waarmee getest wordt tijdens de ontwikkeling wijkt te veel af van de uiteindelijke data waardoor het meetalgoritme niet klopt.	Groot	Klein	Met zo veel mogelijk verschillende data testen om er zo voor te zorgen dat er zo veel mogelijk situaties zijn afgevangen.	Het algoritme aanpassen en de verloren tijd opvangen door andere functionaliteit te laten vallen.
2	Doordat het evolutionair ontwikkelen van het project kan de harde deadline aan het einde van de stage niet worden behaald.	Groot	Klein	Iedere iteratie wordt er gekeken hoe de totale planning van het project er uit ziet en wordt de planning bijgestuurd indien nodig.	In overleg zullen functies geschrapt worden of zullen bestaande onderdelen niet verder doorontwikkeld worden.
3	Stagiair en het stagebedrijf hebben weinig ervaring met RUP hierdoor kan planning te ver gaan afwijken van de RUP principes.	Redelijk	Klein	RUP documentatie wordt als leidraad gebruikt	Een docent of andere expert wordt geraadpleegd en de planning wordt bijgesteld.
4	De stageopdracht voldoet niet aan de eisen van de school.	Groot	Klein	Opdracht wordt gekeurd door de school en de student is op de hoogte van de beoordelingscriteria.	De opdracht wordt aangepast aan de wensen van de school om de stage met een voldoende af te kunnen ronden.

11 Competenties

Voor het behalen van de stage moet er een keuze gemaakt worden voor twee van de drie domeincompetenties: analyseren, ontwerpen en realiseren.

De te ontwikkelen applicatie moet helemaal vanaf de grond worden opgebouwd. Dit houdt in dat er geanalyseerd moet worden hoe de applicatie eruit moet komen te zien en wat de gebruikssituaties zijn. Daarnaast moet er een ontwerp van de applicatie en de database worden gemaakt. Deze ontwerpen zullen daarna gerealiseerd moeten worden.

Hierbij komen dus alle drie competenties voor. Ik kies er voor om de domeincompetenties Ontwerpen en Realiseren/Testen uit te werken. Dit zijn de meest interessante competenties in mijn ogen. Daarnaast zullen deze twee competenties ook de meeste inspanning vereisen gedurende het project.

11.1 Indicatoren

Tijdens de uitvoer van de opdracht zal er aan de volgende indicatoren worden gewerkt, deze indicatoren staan nog niet vast:

Analyseren/onderzoeken

1. Student brengt door middel van interviews en op basis relevante documenten de eisen en wensen van de opdrachtgever in kaart. Hierbij gebruikt de student FURPS+ voor de lijst met eisen en use-cases in het twee-kolomsformaat om de interactie met het systeem in kaart te brengen.
2. De student kan op basis van de data van de opdrachtgever een analyse maken van de bruikbaarheid van de data.
3. De student kan onderzoek doen naar bestaande oplossingen voor zijn probleemstelling en weegt die af.
4. De student kan analyseren welke parameters er niet vastliggen en dus configureerbaar gemaakt kunnen worden.

Ontwerpen

1. De student produceert op grond van een gegeven logisch design class diagram waarin operaties en attributen met hun visibility zijn aangegeven, en waarin getoond wordt welke rollen van een associatie navigable zijn, de code voor een specifieke OO programmeertaal die dat design class diagram implementeert. Student kan een datamodel ontwerpen die bij verschillende programma's bruikbaar is.
2. De student is in staat om gemaakte ontwerpbeslissingen toe te lichten en legt dit vast.
3. Student maakt schermontwerpen vast te leggen hoe dat informatie in de software gepresenteerd moet worden.

4. Door middel van design patterns scheid de student het data-model van de weergave in de software.
5. De student past daar waar nodig multithreading toe.

Realiseren/Testen

1. De student realiseert een applicatie aan de hand van de vastgestelde ontwerpen
2. De kandidaat motiveert de keuze voor de middleware. De keuze moet gemotiveerd worden waarbij andere opties ook overwogen zijn. Denk aan aspecten als: relevantie, toepasbaarheid, meerwaarde t.o.v. andere middleware.
3. De student implementeert gekozen design patterns gebruik makend van de kracht van zijn programmeertaal.
4. De student documenteert de broncode middels gestructureerd commentaar.
5. De student stelt code guidelines op en volgt deze.
6. Er wordt door de student gebruik gemaakt van versiebeheer software om wijzigingen in de broncode te volgen.

Communiceren

1. De student geeft een presentatie over zijn producten en houdt daarbij rekening met de (niet-technische) achtergrond van het publiek.
2. De Student schrijft documentatie waarin de analyse en ontwerpkeuzen worden toegelicht.
3. De student stelt een plan van aanpak op en laat deze goedkeuren door de opdrachtgever.

Samenwerken

1. De student kan samen met de opdracht overeenstemming bereiken over de inhoud van het project.

Planmatig werken

1. De student kan een plan van aanpak opstellen.
2. De student kan een planning opstellen en deze gedurende het project bijhouden en bijsturen.
3. De student is in staat om een projectmanagement methode te kiezen en volgens die methode te werken.

12 Bibliografie

Grain-Watch® Level Indication. LIROS.

.NET Design Patterns. (sd). Opgehaald van Dofactory: <http://www.dofactory.com/net/design-patterns>

Collaris, R., & Dekker, E. (sd). *Iteratieworkflow (Elaboration, Construction en Transition)*.

Opgehaald van RUP op Maat <naslagsite>:

http://www.rupopmaat.nl/naslagsite2011/index_downloads.html

Eeles, P. (2005, 11 15). *Capturing Architectural Requirements*. Retrieved from IBM:

<http://www.ibm.com/developerworks/rational/library/4706.html#N100A7>

Inteqnion BV. (sd). *Triott group*. Opgehaald van Inteqnion:

<http://www.inteqnion.com/nl/pagina/home/triott-group.html>

Larmann, C. (2004). *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development*. Upper Saddle River: Addison Wesley Professional.

Wikipedia. (sd). *Rational Unified Process*. Opgehaald van Wikipedia:

http://nl.wikipedia.org/wiki/Rational_Unified_Process