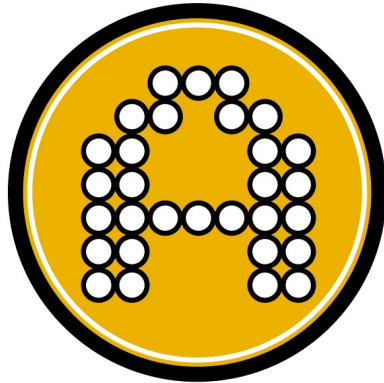


## Project Introductie



**AMAZE**  
**Internet services**

(copyright Amaze)



## Inhoud

Inleiding.....	4
Projecten.....	5
Diagram.....	6
Specificaties.....	7
Planning.....	9
Development.....	11
Statusrapportage.....	13
Opleveringen.....	15
Acceptatie.....	17

## Inleiding

In de afgelopen jaren is binnen Amaze veel interesse ontstaan in project management. Vanuit Amaze is dit vooral door Jeroen Braan gestimuleerd, geholpen door een aantal externe personen waaronder Hjalmar van der Schaaf van NWHProjects. Het doel is altijd geweest onderdelen van bestaande methodieken toe te passen binnen de meer agile gerichte dagelijkse praktijk.

Elementen uit onder andere Feature Driven Development, DSDM en ook Prince2 zijn terug te vinden. Zo is een boekwerkje ontstaan waarin deze kennis is samengevoegd. Het is overigens meer bedoeld als naslagwerk of richtlijn dan als echte projectaanpak.

Als u meer informatie wilt over onze projectaanpak kunt u altijd contact met ons opnemen.

# Projecten

Projecten bestaan uit twee opeenvolgende trajecten: een specificatie traject en een bouw traject. Voor beide trajecten worden contractuele afspraken met klanten gemaakt.

## ***Specificatie traject***

Het specificatie traject bestaat uit het opstellen van een Functioneel-, Grafisch- en Technisch Ontwerp. De specificaties moeten voldoende zijn om een realistische inschatting en planning te maken. Niet alles hoeft in detail beschreven te zijn maar het development team moet in staat zijn een eerste versie van het systeem op te leveren op basis van deze specificaties.

## ***Bouw traject***

Voor het bouw traject wordt een fixed price contract opgesteld. Anders dan bij traditionele fixed price contracten worden inschatting en planning als vast gezien en moet het mogelijk zijn functionele specificaties aan te passen als dat nodig blijkt te zijn. In het kort:

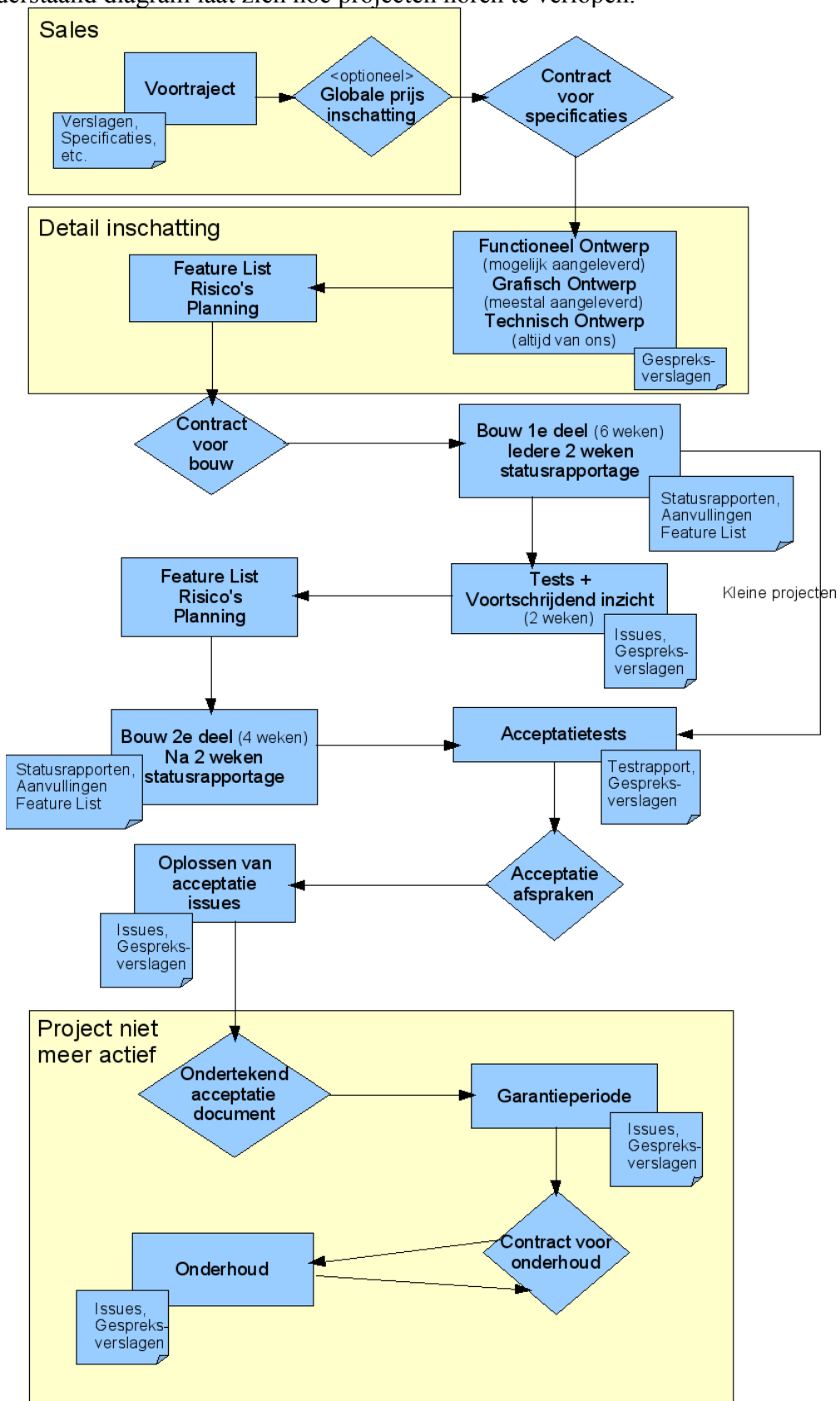
- de basis van het contract wordt gevormd door de inschatting en de planning
- het is tijdens het bouwtraject mogelijk in overleg functionele specificaties aan te passen als dit technisch verstandig blijkt te zijn
- er is na de eerste versie van het systeem ruimte om voortschrijdend inzicht van de klant mee te nemen. Dit is mogelijk zolang de planning en oorspronkelijke inschatting niet aangepast hoeven te worden

Een standaard bouw traject bestaat uit 4 stappen:

1. De bouw van een eerste werkende versie van het systeem, in 6 weken. Iedere 2 weken wordt de voortgang met de klant doorgesproken, na 6 weken wordt een testbaar systeem opgeleverd inclusief testhandleiding.
2. Een overlegfase van 2 weken. Hierin wordt het systeem door de klant getest, wordt voortschrijdend inzicht besproken en wordt een planning voor de laatste 4 weken opgesteld.
3. De bouw van het op te leveren systeem, in 4 weken. Ook hier wordt na 2 weken de voortgang met de klant doorgesproken.
4. Het systeem wordt opgeleverd en de acceptatieperiode start, hierin bepaalt de klant of het systeem voldoet aan de verwachtingen.

# Diagram

Onderstaand diagram laat zien hoe projecten horen te verlopen.



## Specificaties

Projecten beginnen met een Functioneel Ontwerp (FO), Grafisch Ontwerp (GO) en Technisch Ontwerp (TO). We willen zo snel mogelijk aan de slag met de juiste hoeveelheid informatie. Een FO beschrijft wat de klant voor ogen heeft (het 'wat'). Een GO beschrijft in ieder geval de navigatie, de look&feel en de presentatie van content ('hoe ziet het eruit'). Een TO beschrijft de technische richting op basis van onze oplossingen (het 'hoe'). De resultaten van fase 1 worden gebruikt voor deze fase.

Als er geen FO is zullen eerst een of meerdere sessies plaatsvinden bij de klant. Het is belangrijk zo snel mogelijk ook aan het GO te beginnen, een klant wil zien wat er opgeleverd gaat worden. Deze sessies kunnen bestaan uit gesprekken, workshops, meelopen in het bedrijf, etc. Na iedere sessie wordt een verslag naar de klant gestuurd, dit verslag kan ook al grafische onderdelen bevatten. Uiteindelijk worden de verslagen samengevoegd en geordend, dit is Ontwerp versie 0. De feedback van de klant wordt verwerkt tot er een definitief Ontwerp is.

Voor het TO wordt een basismodel gemaakt: verzamel de belangrijke zelfstandig naamwoorden uit het FO en verbind ze, controleer of dit klopt met het GO. Belangrijk voor de klant zijn scenario's over het gebruik van het systeem, denk bijvoorbeeld aan 'Het plaatsen van een nieuwsbericht op de site'. Scenario's zijn eventueel onder te brengen in grote blokken (Onderdelen). Wij beschrijven de scenario's met Features: voor een eindgebruiker begrijpelijke stappen om een scenario te doorlopen, scenario's heten ook wel 'Feature sets'. Features zijn afzonderlijk te testen en zijn binnen 2 weken te implementeren. Als een Feature specifieke eisen stelt aan contact met andere systemen, opslag van data of de User Interface dan wordt dit vermeld.

## Aandachtspunten

Welke afdelingen en gebruikers zullen betrokken zijn bij het systeem? Zijn rollen en rechten van gebruikers bekend? Welke gebruikers hebben toegang tot welke informatie?

Hoeveel beheerders en gebruikers zijn er, hoe verspreid zijn ze, hoe vaak voeren ze bewerkingen uit?

Welke objecten en content types zijn er, is hiervan al aan te geven welke complexiteit te verwachten is en zijn er bijzondere relaties tussen content aan te geven?

Zijn scenario's voor het gebruik van het systeem beschreven in Features? Is er een model?

Is duidelijk welke functionaliteit er verder nodig is en hoe complex deze is?

Hoeveel bezoekers zullen er zijn (de 'load')? Moeten statistieken beschikbaar zijn?

Is de Technische Architectuur duidelijk? Hoe is de hosting geregeld? Is er nieuwe hardware nodig?

Is het systeem afhankelijk van andere systemen en hebben we daar ervaring mee? Is syndicatie nodig?

Welke onderdelen zijn belangrijk voor de klant (bijv. waar ze de core business raken)?

Is Plone nodig? Hoeveel is op te lossen met bestaande componenten? Hoeveel is maatwerk?

Is de workflow uitgewerkt?

Hoe worden errors afgehandeld?

Is er over performance nagedacht? Is personalisatie nodig en heeft dit invloed op performance?

Zijn er duidelijke afspraken over het design gemaakt?

## **Ontwerp (Functioneel en Grafisch)**

*Doel: voldoende informatie krijgen over functionele en grafische wensen en eisen om een globaal model te kunnen maken en een Features List op te kunnen stellen*

- Bestudeer aanwezige documenten
- Optioneel: sessies met klant
- Optioneel: schrijven van sessie verslagen
- Werk Functionele specificaties uit
- Werk navigatie, look and feel en presentation logica uit
- Schrijven van Ontwerp (Functioneel en Grafisch)
- Controleren van Ontwerp met klant

## **Maak een Model**

*Doel: inzicht krijgen in de samenhang tussen objecten en duidelijkheid krijgen over problemen die daarbij naar voren komen*

- Bestudeer documenten
- Maak een globaal model, o.a. op basis van zelfstandige naamwoorden
- Controleer of het model klopt met het FO en GO
- Verfijn het model en neem beslissingen over in te zetten componenten
- Controleren van model en componenten met klant

## **Technisch Ontwerp**

*Doel: basis opzetten voor indeling naar Features en componenten, nadenken over hoe het systeem opgezet gaat worden*

- Bepaal globale Onderdelen van het systeem (bijv. 'User Management')
- Bepaal korte beschrijvingen (titels) van scenario's
- Breng scenario beschrijvingen onder bij globale Onderdelen
- Werk scenario's uit, inclusief opmerkingen over externe systemen, databases en User Interface
- Beschrijf in te zetten componenten in meer detail
- Beschrijf Technische Architectuur
- Maak Feature List: componenten die voor een klant te onderscheiden zijn en voor ons technisch te overzien zijn
- Optioneel: zet eventueel een Proof Of Concept op
- Schrijven van TO
- Controleren van TO met klant

## Planning

Betrokkenen bij het project maken inschattingen op basis van het FO, GO en TO. Scenario's en Features zijn vooral gericht op functionaliteit, er is natuurlijk grafisch design voor Features nodig maar dit wordt toch grotendeels in een apart traject uitgewerkt. Ver uiteenlopende inschattingen worden besproken en mogelijk verdeeld in een optimistische en pessimistische inschatting. Te grote Features worden, indien nodig, gesplitst in kleinere delen en bij onduidelijkheden vindt een klein technisch Proof Of Concept plaats zodat een betere inschatting kan worden gemaakt.

Aandacht moet worden besteed aan het 'critical path': een opeenvolging van activiteiten die afhankelijk zijn van elkaar en waarbij vertraging het hele project zal vertragen. Het is verstandig 'high value, high complexity' onderdelen zo vroeg mogelijk in het project te plaatsen.

Planningen worden opgezet volgens de standaard aanpak. Een omgeving moet in maximaal 12 weken op te leveren zijn. Op basis van de planning worden Features toegewezen aan personen die nogmaals bekijken of de planning haalbaar is. De verantwoordelijke zal grotendeels zelf werken aan de Features maar zal delen ook bij anderen leggen. Ook het grafisch design en de systeemarchitectuur worden in de planning opgenomen.

## ***Aandachtspunten***

Heeft de projectleiding echt inzicht in de materie? Is de ervaring van het projectteam voldoende?

Zijn eindgebruikers betrokken bij het project? Zijn alle relevante personen aanwezig in werkgroepen?

Hebben de eindverantwoordelijken voldoende besluitvaardigheid?

Zijn andere (deel-)projecten afhankelijk van dit project of andersom (marketing, evenement, etc.)?

Is de complete project-organisatie bekend en zijn de relevante gegevens bekend?

Zijn alle taken verdeeld? Heeft iedereen rekening gehouden met taken voor andere projecten?

Is er een centraal e-mail adres opgezet en zijn hier afspraken over gemaakt met de klant?

Zijn de ontwikkel-, test- en productie omgevingen goed opgezet?

Weet iedereen welke deadlines er zijn?

Kan het project op tijd beginnen?

## **Planning**

*Doel: inschattingen maken, werk verdelen en planning maken*

- Bepaal inschattingen op basis van FO, GO en TO – vooral op basis van Features List
- Bepaal grafische aanpak
- Plan opzetten van architectuur in
- Maak planning
- Verdeel Features
- Controleer planning aan de hand van Features, grafisch design en architectuur

## Development

Een project hoort binnen de afgesproken tijd, voor de vooraf berekende kosten met de juiste mensen op te leveren wat is afgesproken. Deze afspraken moeten dan ook bij alle betrokkenen bekend en duidelijk zijn.

Ontwikkeling wordt door de technische mensen uitgevoerd waarbij de contactpersonen op de hoogte blijven over voortgang, zodat communicatie met de klant makkelijker wordt. Onduidelijkheden over Features worden zo snel mogelijk intern of met de klant besproken, de developer neemt hiervoor het initiatief.

Gedurende het hele project moet worden voorkomen dat de scope danwel de feature lijst "onofficieel" wijzigt. Dit komt niet alleen extern voor, ook intern kan bijvoorbeeld onduidelijkheid tot gevolg hebben dat iets anders wordt gemaakt dan is gevraagd. Er wordt in twee delen opgeleverd: het eerste deel wordt in 6 weken gebouwd, het tweede deel in 4 weken. Alleen tussen deze twee delen is ruimte voor de klant om wijzigingen door te voeren.

Om voortgang te kunnen volgen wordt development voor (complexe) features als volgt bijgehouden:

Begrijpen 'wat' gemaakt moet worden	Bepalen 'hoe' je het gaat maken	Het 'wat' en 'hoe' met anderen bespreken	Code schrijven	Code door anderen laten doorkijken	Opnemen in build
1%	40%	3%	45%	10%	1%

Oftewel: een Feature waarvoor code wordt geschreven is 44% klaar (1%+40%+3%), een Feature waarvoor de code is geschreven is 89% klaar (1%+40%+3%+45%). Door de percentages van Features op te tellen kan ook op hoger niveau worden aangegeven hoever het project is. Niet alle stappen zijn nodig maar er wordt impliciet van uit gegaan dat als je bezig bent met een bepaalde stap de percentages van de voorgaande stappen opgeteld kunnen worden. Een simpele Feature zal dus vrij snel op 44% staan omdat het 'wat' en 'hoe' niet uitvoerig hoeft te worden uitgezocht.

## Aandachtspunten

Zijn de prioriteiten van het project duidelijk vastgesteld en krijgt het project voldoende steun?

Wordt er gebruik gemaakt van voorgeschreven methoden en technieken voor systeemontwikkeling?

Is het mogelijk dat het project wordt beïnvloed door een groot personeelsverloop?

Wordt er afgeweken van afspraken en zijn er afspraken over de procedure bij wijzigingen?

Zijn de taken goed verdeeld en is er voldoende overlap om gevolgen van uitval van mensen te minimaliseren?

Worden de ontwikkel, test en productie omgevingen goed gebruikt?

Wordt het centrale mail-account goed gebruikt?

Wordt er wekelijks overlegd?

Is er een aantal keer per week kort overleg?

Worden onduidelijkheden besproken met de klant?

## **Ontwerp complexe Features in meer detail**

*Doel: voorbereiding op schrijven van code*

- Bestudeer model en specificaties
- Optioneel: pas Feature List aan als dit nodig is
- Bepaal welke componenten ingezet gaan worden en of er maatwerk nodig is
- Bepaal welke componenten gebouwd moeten worden
- Optioneel: bespreek gevolgen voor grafisch design, technische architectuur en interactie met andere systemen
- Benader mensen die je nodig hebt voor de Features

## **Ontwikkel Features**

*Doel: bouwen van het systeem*

- Optioneel: Controleer of architectuur op orde is
- Implementeer Features, waar nodig met hulp van anderen
- Optioneel: Laat code controleren en testen
- Release code, eventueel met basis design en dummy code voor interactie met andere systemen
- Zorg dat grafisch design en interactie met andere systemen op orde is
- Release volledige Feature

## **Ontwikkel Technische Architectuur**

*Doel: zorgen dat de architectuur op orde is*

- Bouw Technische Architectuur
- Test Technische Architectuur

## **Ontwikkel grafisch design**

*Doel: zorgen dat grafisch design wordt gemaakt*

- Maak grafisch design, ook voor Features in deze fase
- Test grafisch design

## **Statusrapportage**

Voor oplevering vindt iedere twee weken overleg plaats met de klant. Indien mogelijk wordt hier getoond hoe ver de bouw van het systeem ondertussen is. Het is niet de bedoeling nieuwe wensen van de klant mee te nemen in de planning, dat is alleen mogelijk tussen de eerste en tweede oplevering. Het systeem zelf wordt ook niet getest door de klant.

Wel kunnen technische problemen besproken worden zodat vanuit die invalshoek functionele wensen kunnen worden aangepast.

Er moet in de gaten worden gehouden of de klant nog tevreden is en of de verwachtingen bij de klant overeenkomen met wat is afgesproken; deze aandachtspunten kunnen worden gebruikt voor het inschatten van escalaties.

## ***Aandachtspunten***

Wordt er gebruik gemaakt van geaccepteerde projectplanningsmethoden en voortgangsrapportering?

Hoe is de motivatie en samenwerking binnen het projectteam?

Klopt de planning nog, ook wat betreft assets?

Wordt afgesproken functionaliteit opgeleverd en wordt hierbij het ontwerp gevolgd?

Zijn er schijnbaar vastgelopen processen en kunnen deze worden opgelost?

Zijn er conflicten die moeten worden opgelost?

Is de klant tevreden?

## **Statusrapportage**

*Doel: regelmatig overleg met de klant om inzicht te geven in de voortgang en om eventuele knelpunten te bespreken*

- Verzamel de voortgang van alle Features
- Ga na of er een werkend systeem kan worden getoond
- Bepaal of er technische knelpunten zijn die gevolgen hebben voor functionaliteit
- Bespreek voortgang met klant
- Voeg eventuele functionele wijzigingen toe aan Feature List

## Opleveringen

Het systeem wordt in twee delen opgeleverd, ieder deel wordt eerst intern getest en voorzien van testdocumentatie voor de klant. Bij iedere oplevering worden ook security, load, performance en stress tests uitgevoerd. Eventuele cursussen zijn niet in de projectaanpak zelf meegenomen.

Na oplevering zal de klant het systeem testen. Klanten horen uitsluitend te testen of het systeem voldoet aan de specificaties en of het opgeleverde voldoende is om te accepteren. Fouten in het systeem moet een klant eenvoudig in een issue tracker kunnen plaatsen.

Na de eerste oplevering is er twee weken ruimte om nieuwe wensen te bespreken en de laatste vier weken in te delen. Nieuwe afspraken worden vastgelegd en aan iedereen beschikbaar gesteld die ook de oorspronkelijke afspraken heeft. Bij wijzigingen kan gebruik worden gemaakt van prioriteiten, er worden vier prioriteitsniveaus onderscheiden waarbij de functionaliteit met de hoogste prioriteit het eerst wordt gebouwd. Planning vindt plaats op basis van de eerste drie prioriteitsniveaus en de opdrachtgever geeft de prioriteit aan.

Voor het stellen van prioriteiten wordt onderscheid gemaakt in de volgende 4 niveaus:

- Absoluut noodzakelijk voor het systeem
- Noodzakelijk, maar het is mogelijk een eenvoudiger versie te kiezen
- Gewenst zolang er tijd voor is, afwezigheid beperkt bruikbaarheid van het systeem niet
- Gewenst, maar niet binnen de huidige fase

## ***Aandachtspunten***

Klopt de planning nog?

Wordt afgesproken functionaliteit opgeleverd?

Wordt er afgeweken van afspraken en zijn er afspraken over de procedure bij wijzigingen?

Wordt er gebruik gemaakt van bestaande test- en acceptatieprocedures? Intern en extern?

Wordt er goed intern en extern getest en is hiervoor een issue tracker beschikbaar?

Wordt testdocumentatie opgeleverd?

## **Lever (deel)systeem op**

*Doel: opleveren van getest systeem aan klant*

- Test intern
- Schrijf testdocumentatie en een korte opsomming van functionaliteit
- Gebruik testdocumentatie om door het systeem te lopen, wijzig documentatie indien nodig, meld eventuele issues
- Maak systeem, testdocumentatie en opsomming van functionaliteit beschikbaar voor klant
- De klant moet het systeem testen, vooral op het gebied van functionaliteit en acceptatie
- Verhelp problemen als deze zorgen dat niet goed getest kan worden
- Bespreek test feedback intern en met de klant
- Bespreek gevolgen voor planning en uiteindelijke oplevering
- Wijzig status van Features

## Acceptatie

De acceptatie fase begint direct na de tweede oplevering. Gedurende twee weken test de klant of:

- alle afgesproken functionaliteit is opgeleverd.
- er geen 'showstopper' bugs in het systeem zitten: fouten waardoor het systeem niet in gebruik kan worden genomen.

In die periode worden alleen bugs verholpen die het testen in de weg staan. Aan het einde van deze twee weken levert de klant een testrapport op, gericht op afwezige functionaliteit en showstopper bugs. Andere gemelde problemen zullen niet in het uiteindelijke testrapport terecht moeten komen. Op basis van het testrapport moet een realistische inschatting worden gemaakt voor het oplossen van de fouten.

Acceptatie vindt plaats:

- als het systeem al in gebruik is, tenzij anders afgesproken.
- als het systeem (in belangrijke mate) aangepast is door de klant, tenzij anders afgesproken.
- op het moment van oplevering als geen acceptatie periode is afgesproken.
- een dag na de acceptatie periode als niet voor het eind hiervan een testrapport is opgeleverd en hier geen afspraken over zijn gemaakt.
- op het moment dat de genoemde fouten uit het testrapport zijn opgelost.

Indien wordt afgesproken dat er een 'acceptatie onder voorwaarden' wordt gedaan geldt het systeem als geaccepteerd en kan dus gebruikt worden, er wordt echter afgesproken dat resterende issues nog binnen een bepaalde termijn worden opgelost.

## Aandachtspunten

Zijn er duidelijke afspraken over deze fase gemaakt?

Is er een issue tracker beschikbaar?

Is de garantieperiode duidelijk en is duidelijk wat daarna gebeurt met nog openstaande bugs/issues?

Zijn er voldoende mensen beschikbaar voor de acceptatie fase en de garantieperiode?

Wordt er voldoende onderscheid tussen bugs en wishes gemaakt?

Zijn er afspraken over cursussen?

Is er na acceptatie een 'post mortem'?

## **Oplevering van systeem**

- Test het hele systeem intern op basis van eerdere testdocumenten
- Los eventuele ernstige problemen op
- Plaats kleinere issues in de issue tracker
- Lever het systeem op aan de klant
- De klant test 2 weken het systeem
- In die periode lossen we alleen problemen op die het testen onmogelijk maken
- De klant levert een test rapport op
- Bespreek het rapport intern en met de klant
- Maak een inschatting van de benodigde tijd om resterende issues op te lossen
- Stel een acceptatie document op

## **Issues oplossen**

*Dit zal worden herhaald tot het systeem is geaccepteerd*

- Loop door issues
- Bespreek gewenste volgorde van oplossen intern en met klant
- Los issues op
- Test issues
- Laat opgeloste issues testen door klant

## **Acceptatie**

- Bespreek acceptatie document met klant
- Klant ondertekent het acceptatie document
- Bespreek acceptatie periode met klant
- Bespreek eventuele extra onderhouds- en garantiemogelijkheden met klant