

Serviceprogramma	
Titel: Acceptatietest	
Studentversie	
Afdeling	ICT
Opleiding	Applicatieontwikkelaar en Mediadeveloper
Crebonummer	(resp.) 95311 en 95313
Niveau + BOL/BBL	4, BOL
Startdatum in curriculum	1-08-2015
Leerjaar	1
Periode	3
Week	5
Niveau serviceprogramma	Verkenning

## ALGEMENE INFORMATIE

Soort serviceprogramma	Vaktechnische competentie	
Competenties	J, K, L, M en O (2.5) en D, J, K en N (3.2)	
Werkprocessen	2.5 en 3.2	
Tijd	180 minuten	
Veronderstelde voorkennis	Leerlingen hebben weleens een checklist voor de oplevering gemaakt.	
Opbouw serviceprogramma	<b>30 minuten</b>	Opstart (welkom, onderwerp bekend maken, doelen doornemen, VOORDOEN)
	<b>140 minuten</b>	Oefenen, Oefenen, Oefenen (bijsturen waar nodig, rondlopen)
	<b>10 minuten</b>	Afsluiting (evalueren doelen, aantekeningen laten maken en in portfoliomap laten stoppen)
Didactische werkvormen	Instructie Oefening Oefening	
Rol trainer	<ul style="list-style-type: none"> <li>Voordoen (expert)</li> </ul>	

	<ul style="list-style-type: none"> <li>• Vragensteller</li> <li>• Feedbackgever</li> </ul>
Doelen	<p>Aan het einde van het serviceprogramma beheerst / weet de student het volgende:</p> <p>VAARDIGHEIDSDOELEN, de student kan:</p> <p>1.</p> <p>KENNISDOELEN, de student weet:</p> <p>1.</p>
Benodigde ruimte	
Leermiddelen en materialen	
Bijlagen	<p>bijlagen</p> <p>1. Persoonlijke leerdoelen</p> <p>2.</p>
Literatuur en bronnen	
Opmerkingen	

## UITVOERING DOOR DOCENT

Opstart (30 minuten)	<ul style="list-style-type: none"> <li>- Studenten laten plaatsnemen aan U-opstelling</li> <li>- Welkom</li> <li>- Onderwerp bekend maken: waarom dit nodig voor je beroep</li> <li>- Doelen laten voorlezen door studenten / doelen op bord noteren: DIT GAAN WE VANDAAG TRAINEN</li> <li>- VOORDOEN</li> </ul>
Oefenen (140 minuten)	<p><b>OPSTELLING GROEPJES / SUBGROEPJES:</b></p> <p>1. Elke leerling maakt beide opdrachten via bijlage 1 en 2</p> <p>Zie bijlage 1</p> <p><b>SPECIFIEKE OPDRACHT / THEMA / OEFENING:</b></p> <p>1. Oefening</p> <p>2. Oefening</p>
Afsluiting (10 min)	<ul style="list-style-type: none"> <li>- Studenten laten plaatsnemen aan U-opstelling</li> <li>- Serviceprogramma evalueren</li> <li>- Studenten aantekeningen laten maken</li> <li>- Student aantekeningen laten stoppen in portfolio map</li> </ul>

# Acceptatie van de applicatie

Deze SP heeft twee bijlagen;

Bijlage 1 gaat over testen in het algemeen; welke testen zijn er allemaal?

Bijlage 2 gaat specifiek over de acceptatietest.

## **Taak 1**

Werk bijlage 1 door.

## **Taak 2**

Lees van bijlage 2 punt 1 t/m 6 goed door en beantwoord dan de volgende vraag:



## Bijlage 1 Testen

---

# Inleiding testplan

Een testplan is een plan om een testobject zoals een machine, een procedure, software of een combinatie daarvan te testen.

Een testplan wordt normaal gesproken gemaakt door de testmanager, die het plan daarna ook gaat uitvoeren. Doel van het testen en van het testplan is om te controleren of het testobject voldoet aan de vooraf opgestelde specificaties. Daarnaast moet het testen een inschatting geven van de kwaliteit van het product en de risico's die men loopt als het testobject gebruikt gaat worden. Een (master of hoofd) testplan is een uitwerking van een teststrategie.

## Vragen

---

1. Heb jij bij jouw project specificaties opgesteld? Als apart document? Dus niet als onderdeel van het Plan van aanpak.
  2. Wat is de naam van dit document?
  3. Is dit document door de opdrachtgever ondertekend?
  4. Als dit document niet door de opdrachtgever is ondertekend, heeft het dan waarde?
  5. In wiens belang is dit document?
-

## Onderdelen

Afhankelijk van het te testen product, de testbasis en de verantwoordelijkheid van de organisatie die het testplan laat opstellen, zal een testplan de volgende onderdelen bevatten:

---

- **Review van de ontwerpdocumenten**

Deze wordt tijdens de ontwikkeling uitgevoerd. Bekeken wordt of de documentatie aan de (vorm)eisen voldoet.

- **Productietest** - Hierbij wordt gekeken of het geheel acceptabel is voor de beheerders. Of het voldoet aan de eisen die zij stellen, of er voldoende documentatie aanwezig is, etc. Soms wordt ook een stresstest uitgevoerd om te zien of het product ook de aantallen aankan die tijdens productie kunnen voorkomen.
- **Unittest** - Wordt uitgevoerd door de programmeurs zelf waarbij zij hun specifieke onderdeel testen, los van het geheel.
- **Integratietest** - Wordt aan het einde van de ontwikkeling uitgevoerd, vaak door speciale testers waarbij het geheel geïntegreerd getest wordt.
- **Acceptatietest** - Wordt uitgevoerd door (vertegenwoordigers van) de gebruikers als het product wordt geïmplementeerd, geïnstalleerd of afgeleverd. Wordt ook wel UAT of User Acceptance Test (gebruikersacceptatietest) genoemd.

## Vraag

6. Welke onderdelen zullen bij jou opgenomen dienen te worden?

Hieronder volgt een algemeen stuk over testen. Voor jullie projecten “De gokkers” (AO) of „In Business” (MED) zijn sommige zaken niet van toepassing. Je moet deze er zelf uitfilteren. Voor volgende projecten zijn bepaalde zaken weer wel van toepassing.

► Dit document dien je dus goed te bewaren!!

Er zijn verschillende soorten tests die we hier op volgorde van uitvoering zullen behandelen.

---



# Unit test

Unit testen is het testen van stukjes code in plaats van de gehele code. Dit zorgt er voor dat elk deel van de code afzonderlijk goed werkt voordat het in het hoofdproject wordt opgenomen. Unit testing vormt een groot deel van test driven development wat inhoudt dat een programmeur door middel van white box testing eerst de tests schrijft, die dus falen, en daarna de code schrijft om een test te halen. Dus een programmeur probeert steeds verder door de test te komen met steeds weer nieuwe toevoegingen aan de code. Uiteindelijk wordt de hele test gehaald en herschrijft de programmeur de code om aan de standaarden van het team te voldoen.

---

## Vraag

7. Wat dien je te verstaan onder whitebox testing?

# Technische test

Een technische test kijkt vooral naar de “achterkant” van een programma, dus de code. Een functionele test (zie hierna) kijkt vooral naar de “voorkant” van een programma, dus de interface.

Technische tests worden heel vaak geautomatiseerd uitgevoerd. Er zijn programma's die dit voor een ontwikkelaar kunnen doen.

Daarnaast test een ontwikkelaar zelf een gedeelte van het programma. Vooral als dit gaat om het testen van de code zelf. Als het gaat om het testen van het programma zelf, dan zit hier vaak een professionele tester op. Deze tester probeert dan op basis van de geschreven test en zijn eigen kennis bugs naar boven te krijgen die worden gerapporteerd aan het ontwikkelteam.

De technische test heeft onder meer tot doel:

- te meten hoe de performance is;
- na te gaan welke effecten de software heeft op de werking van andere software;
- na te gaan welke problemen zich kunnen voordoen en hoe deze verholpen kunnen worden.

## Vragen

8. Wat zouden bij jouw programma (AO) of websiteprojecten (MED, project In Business) bugs kunnen zijn?
9. Noem wat onderdelen waaruit een technische test hoort te bestaan. Noem er minimaal vijf.
10. Zijn er voor jouw studierichting (AO of MED) programma's of methoden die automatisch een technische test uit kunnen voeren?  
Zo ja, benoem deze en geef zo nauwkeurig mogelijk aan wat ze kunnen.

## Functionele test

Als een applicatie wordt opgeleverd dient te worden getest of de applicatie ook daadwerkelijk doet wat het volgens het ontwerp zou moeten doen. Dit noemt men de functionele test of de functionele systeemtest. Dit is een **black box** test. Er wordt alleen gekeken naar de functionele werking: doet het programma wat het volgens het ontwerp zou moeten doen? Wat het programma zou moeten doen is uiteraard vastgelegd in een document (functioneel model = hernieuwde opdracht) met daaronder de handtekening van de opdrachtgever.

In deze test worden op voorhand testgevallen geschreven op basis van het functioneel ontwerp. Het begint dus met een kritische bestudering van het functioneel ontwerp. Bij fouten wordt gekeken of dit een ontwerpfout of een bouwfout betreft en wordt deze fout vervolgens gemeld bij betreffende afdeling. Vaak gebeurt dit in een aantal ronden waarbij telkens een nieuwe versie wordt opgeleverd, welke vervolgens getest dient te worden. Deze test wordt veelal uitgevoerd door professionele testers. Er zal worden geprobeerd om met zo weinig mogelijk testgevallen zoveel mogelijk voorkomende situaties te raken.

Het doel van de functionele test is:

- evalueren van de functionaliteit ('Zit alles er in wat ik nodig heb?')
- evalueren van de gebruiksvriendelijkheid ('Kan ik er gemakkelijk mee omgaan?')
- evalueren van de efficiency ('Levert het me tijd op?')

Hieraan kun je uiteraard zelf opties toevoegen.

## Vragen

11. Heb je van jouw project een functioneel ontwerp? Zou je dat nu er zo bij kunnen pakken?
12. Welke van de drie hierboven genoemde doelen is/zijn bij jouw project belangrijk?
13. Hierboven staat vermeld dat er gekeken wordt of het programma werkt zoals in het functioneel ontwerp aangegeven is. Is dit het enige waarnaar gekeken wordt of wordt er naar meer gekeken? Zo ja, wat dan nog meer?

## Acceptatietest

Over het algemeen worden aan het eind van de functionele test de gebruikers betrokken bij het testen. Dit noemt men de 'functionele acceptatietest'. In deze fase wordt ook gekeken of een en ander in de praktijk werkbaar is voor de eindgebruikers.

Als de acceptatietest afgerond is, wordt er een rapportage opgeleverd. Hierin wordt aan de opdrachtgever een advies gegeven over de kwaliteit van het opgeleverde systeem. Daarnaast zullen de opgeloste bevindingen worden beschreven en de eventueel nog openstaande bevindingen.

De acceptatietest dient om te controleren of alle afgesproken zaken door de leverancier zijn uitgevoerd.

Er wordt een testplan en een testscript opgesteld en er wordt getest aan de hand van acceptatiecriteria die zijn gebaseerd op de vastgelegde afspraken. De acceptatiecriteria worden ook wel 'opleveringscriteria' genoemd. De software is nooit direct in één keer goed. Uit de test komt een gebreken- of foutenlijst voort, vervolgens worden de gebreken verholpen, opnieuw getest tot uiteindelijk het programma zo goed als foutloos is. Daarna vindt de officiële oplevering plaats en kan de software 'uitgerold' worden in de organisatie. Tijdens de acceptatietest kunnen uiteraard weer nieuwe wensen boven komen drijven. Afhankelijk van de afspraken in het contract kunnen die direct worden gerealiseerd of zal er eerst een aanvullende offerte worden opgesteld. Het tijdstip van de oplevering kan daardoor keer op keer worden opgeschoven. De ervaring leert dat de aanvankelijk geplande opleveringsdatum nooit wordt gehaald. Dat is niet erg, want je kunt er beter iets langer over doen dan met een onaf product te blijven zitten.

Zie verder bijlage 2 voor meer informatie over deze belangrijke test.

# Regressie testen

Testen stopt niet nadat een applicatie in productie is gegaan. Nadat een applicatie is gebouwd en getest wordt het in productie genomen en wordt het in beheer genomen. In deze fase zullen er regelmatig wijzigingen in het bestaande systeem worden aangebracht. Als gevolg van deze aanpassingen kan het gebeuren dat er ongewild fouten ontstaan in ongewijzigde delen. Dit heet regressie; de kwaliteit van het systeem gaat achteruit. Om dit te voorkomen dient er een regressietest te worden uitgevoerd. Deze test richt zich met name op onderlinge samenhang tussen functies. Dit omdat hier de kans op regressie het grootst is.

Regressietesten is dus het testen van niet-gewijzigde programmatuur. Regressietesten hoort bij de acceptatiecyclus vóór in productie name. Het maakt inzichtelijk of de ongewijzigde software nog steeds werkt zoals voorafgaand aan het doorvoeren van een wijziging. Dit is een proces wat zich vaak herhaalt en veel tijd in beslag neemt. Bijvoorbeeld bij websites, waarbij frequente updates worden doorgevoerd met niet altijd grote wijzigingen. Toch wil men dan weten of de website nog steeds naar behoren functioneert.

## Inspanning

Het regressietesten op zich vergt een minder grote inspanning dan het gedetailleerd testen van een wijziging. Vaak kan worden volstaan met enkele testgevallen. Echter doordat een wijziging vaak maar enkele functies betreft en regressietesten alle functies dient te raken, is de totale inspanning vaak toch groter.

## Herhaalbaarheid

Doordat een regressietest bij elke nieuwe release opnieuw dient te worden uitgevoerd in (bijna) ongewijzigde vorm, is het van belang om te zorgen voor een goede en gestructureerde documentatie. Dit maakt de herbruikbaarheid van de test groter. Omdat een regressietest vaak wordt herhaald leent het zich bij uitstek voor geautomatiseerd testen.

## Beheer

Om te zorgen voor het bruikbaar blijven van de regressietest zal deze moeten worden bijgewerkt bij wijzigingen in het systeem. Dit kunnen zowel functionele wijzigingen zijn als wijzigingen als gevolg van opgeloste bevindingen. Op deze manier blijft de test altijd actueel.

## Vraag (Alleen voor MED)

14. Is een regressietest voor jouw project(en) raadzaam? Zo ja, hoe zou die er grofweg uit moeten zien? Geef aan op welk deelproject het betrekking heeft.

Bij de invoering van nieuwe software komt heel wat werk kijken. Het loont daarom de moeite om vóór over te gaan tot de aanschaf van een softwarepakket, een proof of concept (zie hierna) te doen.

# Proof of concept (PoC)

De PoC omvat zowel een technische test als een functionele test. De technische test wordt uitgevoerd aan de hand van acceptatiecriteria die door ICT worden opgesteld.

Het doen van een Proof of Concept is om drie redenen noodzakelijk:

- Een Proof of concept heeft als doel te achterhalen of het geselecteerde softwarepakket voldoet aan de eisen en verwachtingen die de opdrachtgever heeft gesteld. Die eisen en verwachtingen heeft hij in een eerder stadium vastgelegd in het functioneel model, een beschrijving van hetgeen het programma moet doen en waarvoor het wordt gebruikt en in het Programma van Eisen. In het Programma van Eisen (PvE) zijn alle eisen en wensen opgenomen met een weging van het belang dat hij er aan hecht.
- Een PoC heeft daarnaast tot doel te bepalen hoe men het softwarepakket ingericht wil hebben en welke zaken extra aandacht nodig hebben bij een implementatie. Als onderdeel van de PoC kan men ook een proefconversie laten doen van de data uit het huidige systeem naar het nieuwe systeem.
- Een belangrijk doel van de test is te beproeven of de gebruikers met het systeem willen en kunnen omgaan. Een opleiding in het pakket dient deel uit te maken van de PoC, zodat de testers in ieder geval kennis hebben gemaakt met de werkwijze.

Voor de test wordt op basis van het bovenstaande een testplan opgesteld .

Het testplan bestaat uit drie delen:

1. Een beschrijving van de wijze waarop de test zal worden uitgevoerd, met wie en wanneer en een beschrijving van het doel, de middelen, de voorwaarden en de acceptatiecriteria. De voorwaarden zijn bijvoorbeeld dat iedere tester een opleiding moet hebben gedaan, kan beschikken over de software met een handleiding en dat de gebruikersrechten moeten zijn geregeld.
2. Het testscript waarin de opdrachten zijn beschreven die door de testgroep worden uitgevoerd. Het testscript bestaat uit twee onderdelen:
  - toelichting op alle handelingen per functie (waarom en hoe)
  - opdrachten die moeten worden uitgevoerd door de testers
3. Een evaluatieformulier waarop de testers hun bevindingen kunnen noteren. Dat kan aan de hand van vragen of een score.

Het testen zelf kan individueel of groepsgewijs plaatsvinden, maar het moet wel gestructureerd gebeuren. Het beste is de testers voor een bepaalde dag in te roosteren. De test is niet vrijblijvend. Er moet nauwkeurig bijgehouden worden hoe men getest heeft, onder welke omstandigheden, hoe lang men er over gedaan heeft, tegen welke problemen men is aangelopen enz.

De testperiode mag ook niet te lang duren en er moeten afspraken worden gemaakt over de tijd die iedere tester tenminste aan de nieuwe software besteedt.

## **Kosten**

Een Proof of Concept vergt niet alleen veel tijd van de instelling die het softwarepakket overweegt aan te schaffen, maar ook van de leverancier. De software moet worden geïnstalleerd en ingericht en er moet eventueel een proefconversie worden gedaan. Er moeten opleidingen georganiseerd worden en de helpdesk moet beschikbaar zijn. In veel gevallen zal de leverancier ook een of meer consultants de PoC laten begeleiden.

Kortom, het is veel werk en een PoC moet dan ook goed voorbereid worden. Het opstellen van een projectplan waarin de rollen van beide partijen zijn beschreven is aan te bevelen.

Onontbeerlijk is een projectcontract of overeenkomst waarin beide partijen overeenkomen een Proof of Concept te doen en wat de consequenties zijn bij de go/no go beslissing aan het einde van het traject. Zo voorkomt men onduidelijkheid en valse verwachtingen. Het spreekt vanzelf dat een degelijke Proof of Concept niet kosteloos kan worden uitgevoerd. De kosten kunnen, afhankelijk van de afspraken die u maakt, de proefconversie(s), de opleidingen enzovoorts oplopen van 3.000 tot 15.000 euro. Dat lijkt veel geld, maar de kosten worden dikwijls verrekend bij de aanschaf. Mocht er niet tot aanschaf worden over gegaan dan heeft men in ieder geval ervaring opgedaan en een miskoop vermeden. En daarmee het risico ontlopen om tienduizenden tot honderdduizenden euro's uit te geven voor een product dat eigenlijk niet bij de opdrachtgever past.



## Bijlage 2 Acceptatietest

---



## Inhoud

1 Wat is een acceptatietest? .....	18
2 Waarom is een acceptatietest voor de opdrachtgever belangrijk? .....	20
3 Wat moet je testen? .....	21
4 Wanneer kun je met de acceptatietest beginnen? .....	22
5 Hoe voer je een acceptatietest uit? .....	23
6 Wat doe je met de resultaten van de test? .....	27
7 Wanneer eindigt de acceptatietest? .....	28
8 Wat gebeurt er na afloop van de acceptatietest? .....	29
9 Wanneer kan de opdrachtgever de geleverde applicatie in gebruik nemen? .....	30
10 De opdrachtgever wordt niet vrolijk van het volgende: .....	30
11 Acceptatietest uitbesteden .....	31
12 Kosten .....	33
13 Referenties .....	33
14 URL .....	33

## 1 Wat is een acceptatietest?

Wanneer je maatwerk-software hebt ontwikkeld, dan vindt meestal direct na oplevering een acceptatietest plaats. Hiermee worden de activiteiten van de opdrachtgever bedoeld die erop zijn gericht om in een overeengekomen testperiode het geleverde product systematisch te testen en beoordelen.

Het doel van de acceptatietest is vast te stellen dat de software voldoet aan de eisen en wensen van de opdrachtgever en dat het geschikt is voor ingebruikname.

## 2 Waarom doe je een acceptatietest?

Als ontwikkelaar doe je een acceptatietest omdat je wilt weten of je op de goede weg bent met wat je aan het maken bent. De klant is degene die betaalt en dit is degene die tevreden gehouden moet worden.

## 3 Wat willen we er mee bereiken?

Je wilt er mee bereiken dat jouw idee als ontwikkelaar, of jullie idee als team, aansluit op datgene wat je klant wil. Je wilt dus voorkomen dat je iets maakt wat later weggegooid kan worden omdat de klant het er niet mee eens is. Hier heb je dan wel tijd aan besteed die je dan kwijt bent.

Een acceptatietest is GEEN technische test. Het test namelijk niet of onderdelen werken, maar of ze werken zoals de klant dit wil. Zorg er dus voor dat het product van tevoren de technische test heeft gehad zodat er (bijna) geen bugs meer inzitten.

## 4 Hoe ziet zo'n test er dan uit?

Stel dat je voor een klant een inlogschermbt moet maken. Waarschijnlijk heeft je klant je dan verteld wat de kleursamenstelling is en het lettertype. Om te bepalen of alles ook echt goed uitkomt en werkt zoals de klant het wil zou er een acceptatietest geschreven kunnen worden zoals in de tabel hieronder (helemaal op het eind).

Bedenk je hierbij dat je altijd moet vertellen wat er gaat gebeuren en welke actie je verwacht met welk resultaat tot gevolg (Hieronder, aan het eind van dit document, kolom 2). Dit is omdat anders de tester wordt afgeleid met vragen en opmerkingen die niks te maken hebben met wat je nu eigenlijk wilt weten.

Let er op dat je er voor zorgt dat er genoeg ruimte is om notities te maken. (Hieronder, aan het eind van dit document, kolom 5.)

Een andere manier om een acceptatietest uit te voeren is door de tester een lijst met opdrachten te geven en het uitvoeren ervan op te nemen met een geluidsrecorder of videocamera. Het is belangrijk dat de tester wordt verteld dat deze hardop moet zeggen wat er wordt gedacht tijdens het testen.

## Vragen

15. Als jij nu een acceptatietest zou moeten maken, welke aandachtspunten heb je nu al gezien die je erin op wilt nemen?
16. Wat is het doel van de acceptatietest?
17. Wat is de juiste volgorde van het uitvoeren van deze tests (alfabetische volgorde)? Acceptatietest, Functionele test, Technische test?
18. Waarin verschilt de technische test van de acceptatietest?

## 5 Waarom is een acceptatietest voor de opdrachtgever belangrijk?

In de algemene voorwaarden van software- en hardwareleveranciers staat in de paragraaf over garantie vaak een bepaling met de volgende strekking:

*De leverancier kan de kosten van herstel in rekening brengen indien de fouten bij het uitvoeren van de overeengekomen acceptatietest hadden kunnen worden vastgesteld.*

Een overeenkomst met deze bepaling legt een deel van de verantwoordelijkheid voor het testen van de applicatie bij de opdrachtgever. Indien de acceptatietest achterwege wordt gelaten of onvoldoende grondig wordt uitgevoerd, dan wordt de opdrachtgever gestraft met extra kosten voor het herstel van fouten die later worden ontdekt. Door een grondige acceptatietest uit te voeren bespaart hij zich deze onzekere kosten.

Het komt ook voor dat de betreffende bepaling als volgt is geformuleerd:

*Na acceptatie is leverancier op grond van deze overeenkomst niet gehouden tot het herstel van gebreken in de software instellingen.*

Wanneer bovendien een garantieregeling of onderhoudscontract ontbreekt dan is de acceptatietest je enige instrument om de goede kwaliteit van de software te garanderen.

De bepalingen over het herstel van fouten in de testperiode zijn voor de leverancier veelal dwingender dan de overeenkomstige bepalingen voor de garantieperiode of de periode daarna. Het komt ook voor dat de betaling van een deel van de ontwikkelkosten afhankelijk wordt gesteld van een succesvolle afronding van de acceptatietest. De testperiode biedt je daarom een goede gelegenheid om de kwaliteit en betrouwbaarheid van de applicatie te verbeteren.

### Vraag

19. Wat is het mogelijke gevolg van het onvoldoende testen bij de acceptatietest?

## 6 Wat moet je testen?

In de algemene voorwaarden van software- en hardwareleveranciers is vaak een bepaling opgenomen met de volgende strekking:

*Onder onvolkomenheden worden verstaan fouten en gebreken of het op andere wijze niet functioneren overeenkomstig de overeengekomen specificaties.*

Eenzijds is een dergelijke definitie zeer ruim, wegens het ontbreken van een omschrijving of afbakening van de begrippen "fout" en "gebrek". Anderzijds zijn je enige concrete aanknopingspunten de overeengekomen specificaties (functioneel model = hernieuwde opdracht). In de praktijk betreft dit vaak de door de leverancier opgestelde specificaties. Die leverancier heeft er - althans bij een overeenkomst met een vaste prijs - alle belang bij om in de specificaties de functionaliteit zoveel mogelijk in te perken, maar niet om de functionaliteit zodanig concreet en specifiek te omschrijven dat de opdrachtgever eenvoudig kan aantonen dat de applicatie gebreken vertoont.

Je kunt bij je acceptatietest de volgende zaken testen:

- De functionaliteit, dat wil zeggen alle aspecten van gedrag en resultaten die volgens de overeengekomen specificaties aan bepaalde voorwaarden moeten voldoen.
- Koppelingen met andere informatiesystemen indien van toepassing.
- De performance.
- De gebruiksvriendelijkheid.
- De mate waarin de software voldoet aan kwaliteitsgaranties in de overeenkomst met de leverancier.
- De documentatie voor beheerders en gebruikers.
- De procedures voor handmatige werkzaamheden.

### Vraag

20. Welke van bovenstaande zaken zou bij jouw project aan de orde zijn?

Bij de acceptatietest vergelijk je het eindresultaat van het ontwikkelproces met de huidige behoeften van de eindgebruikers. Het is dus zeer wel mogelijk dat je tot nieuwe inzichten komt met betrekking tot de oorspronkelijk geformuleerde functionele eisen. In dat geval moet je duidelijk onderscheid maken tussen onvolkomenheden en nieuwe of gewijzigde functionele eisen. Er zal van je worden verlangd dat

de onvolkomenheden worden hersteld, maar wat betreft de nieuwe of gewijzigde functionele eisen zul je de opdrachtgever een voorstel kunnen doen. Doorgaans zul je deze werkzaamheden als meerwerk opleveren.

## **7 Wanneer kun je met de acceptatietest beginnen?**

Je kunt met de acceptatietest beginnen nadat:

- De ontwikkeling van de applicatie volledig is afgerond.
- Je een technische en een functionele test succesvol hebt uitgevoerd en je de opdrachtgever over de resultaten hebt geïnformeerd.
- De applicatie volledig is opgeleverd en geïnstalleerd.
- Alle benodigde systeemkoppelingen tot stand zijn gebracht, indien van toepassing.
- Alle documentatie beschikbaar is, zoals de handleiding voor eindgebruikers en de handleiding voor de systeembeheerder.
- De software en hardware van voldoende kwaliteit is voor normaal operationeel gebruik onder productieomstandigheden. Er mogen geen componenten meer zijn die niet functioneren.
- Het komt voor dat je de applicatie oplevert voor de acceptatietest, terwijl je nog bezig bent wat af te ronden. Je levert de applicatie dan met een lijst bekende fouten, of de opdrachtgever ontdekt snel fouten die normaal operationeel gebruik van de gehele omgeving belemmeren. In dat geval is er in feite sprake van een bèta-test, niet van een acceptatietest. Je moet dan de leverancier laten weten dat het een bèta -test betreft.

## 8 Hoe voer je een acceptatietest uit?

Stel een globaal testplan op. Hierbij dien je onder andere rekening te houden met oplevering van de software in onderdelen of fasen. In de algemene voorwaarden van softwareleveranciers staat vaak een bepaling met de volgende strekking:

*Indien de software in onderdelen of fasen wordt opgeleverd en getest, laat de niet-acceptatie van een bepaalde fase of onderdeel een eventuele acceptatie van een eerdere fase of onderdeel onverlet.*

Je kunt in dat geval het testen van een onderdeel of fase niet uitstellen tot het testen van het geheel of van het eindresultaat. Daarmee zou je het risico lopen dat fasen of onderdelen stilzwijgend worden geaccepteerd, zodat het herstel van fouten die je later aantreft voor rekening van de opdrachtgever is. Dit geeft ook aan dat het voor de opdrachtgever van belang is om te zorgen dat de fasen en onderdelen van jou testbaar zijn.

Een auto kun je testen, maar kun je een stuur testen?

Aangezien je de acceptatietest doorgaans in een betrekkelijk korte periode moet uitvoeren is het van belang om de test grondig voor te bereiden:

- Richt tijdig de testomgeving in (hardware, software, documentatie, licenties, werkruimte).
- Zorg dat de opdrachtgever de medewerkers aanwijst die de acceptatietest gaan uitvoeren. Het is aan de opdrachtgever om er voor te zorgen dat deze zich vóór aanvang van de test voldoende basisvaardigheden eigen maken met betrekking tot het platform (operating system en user interface). Zorg dat de medewerkers tijdig de gebruikershandleiding krijgen en de functionele specificaties van de te testen software bij hen bekend zijn.
- Schrijf vooraf concrete test-scenario's (gebruikershandelingen) en test sets (data). Hiermee kun je al beginnen zodra de functionele specificaties beschikbaar zijn.
- Zorg dat de testresultaten genoteerd kunnen worden.

Bij het uitvoeren van de acceptatietest zul je waarschijnlijk fouten en onvolkomenheden aantreffen. Het is daarbij van belang dat je nagaat of de fouten de voortgang van de acceptatietest belemmeren. In de algemene voorwaarden van software/hardware leveranciers is vaak een bepaling opgenomen die de opdrachtgever verplicht om een dergelijke situatie bij jou als programmeur te melden. De testperiode kan worden onderbroken totdat de applicatie zodanig is aangepast dat de acceptatietest ongehinderd voortgang kan vinden of opnieuw kan worden gestart. De opdrachtgever zal voortgang belemmerende fouten bij jou melden. Zou hij dit niet doen, dan loopt hij het risico dat bij het einde van de testperiode niet alle functionaliteit voldoende is getest en dat het herstel van fouten die hij daarna ontdekt voor zijn rekening is.

Fouten en onvolkomenheden dien je zorgvuldig en gedetailleerd te administreren. Relevante gegevens zijn:

- Welk test-scenario is uitgevoerd?
- Wat was de begintoestand van de omgeving en/of de applicaties?
- Welke testgegevens zijn ingevoerd?
- Wat is het waargenomen gedrag van de software?
- Wat is het verwachte gedrag?
- Waarin wijkt het waargenomen gedrag precies af van het verwachte gedrag?
- Hoe kan de fout gereproduceerd worden?
- In welke versie van de software is de fout aangetroffen?
- Wat heb je gedaan om mogelijke oorzaken van de fout te achterhalen?

Met name het aspect van de reproduceerbaarheid is van groot belang. In de algemene voorwaarden van software/hardware leveranciers is vaak een bepaling opgenomen met de volgende strekking:

*Een onvolkomenheid wordt onderzocht op voorwaarde dat deze reproduceerbaar is.*

Vanuit het perspectief van jou als programmeur is deze voorwaarde goed te begrijpen. Het onderzoeken van een niet-reproduceerbare fout kan zeer tijdrovend en kostbaar zijn. Het is zeker niet onredelijk dat je reproduceerbare testresultaten oplevert. Vanuit het perspectief van de opdrachtgever is het echter moeilijk te aanvaarden dat niet-reproduceerbare fouten niet onderzocht en hersteld worden.

Dit type fouten is dan ook een bron van conflicten tussen de programmeur en opdrachtgever.

Je doet er goed aan om je acceptatietest zodanig uit te voeren dat tenminste de testresultaten reproduceerbaar zijn. In de praktijk zullen dan de meeste fouten eveneens reproduceerbaar blijken te zijn. Indien je reproduceerbare tests niet-reproduceerbare fouten aan het licht brengen dan zal de opdrachtgever hoogstwaarschijnlijk de acceptatie weigeren op grond van het argument dat het onbetrouwbare gedrag van de software een operationele ingebruikname redelijkerwijs in de weg staat.

Naast concrete fouten kunnen ook kleine schoonheidsfouten en andere ongemakken geregistreerd worden. Jij als programmeur bent wellicht bereid om deze te verbeteren, al kun je uiteraard de kosten voor meerwerk in rekening brengen.



## Vraag

21. Bij ons op school kon een leerling de printers niet meer benaderen. Hij was ingelogd met D<ov-nummer>@edu.rocwb.nl. Hij kon op geen enkele manier de printer installeren (Normaal gesproken via Printer toevoegen, Netwerkprinter, Printer staat niet in de lijst, \\r0161\svc.local\radius). De volgende dag deed hij weer een poging, maar logde in via rocwblocal\D<ov-nummer>@edu.rocwb.nl. Nu lukte het wel. Bestanden of drivers waren aangepast. Welke fout heeft hij gemaakt als hij 100% zeker zou willen weten of het komt dat hij de eerste keer het stukje *rocwblocal* vergeten was? Gebruik voor beantwoording van deze vraag het stukje tekst boven deze vraag.

## 9 Wie voert het uit?

Een acceptatietest wordt **altijd** uitgevoerd door de klant (liever nog de toekomstige gebruiker) of door een vertegenwoordiger hiervan. Dit is degene die het product moet goedkeuren. Het is verstandig om altijd te zorgen dat een acceptatietest wordt voorzien van de naam (en functie) van degene die de test uitvoert met daarbij een handtekening, datum en tijdspanne. Dit om te zorgen dat er bewijslast is mocht een klant toch zeggen dat er iets niet naar wens is.

## Vragen

22. Wat zou de reden zijn dat de leverancier niet zelf mag testen of een demo mag geven?  
23. Wat wordt er met de genoemde tijdspanne bedoeld? Waarom is dit?

Hij moet aan kunnen geven:

- welke fouten en onvolkomenheden hij ontdekt heeft;
- welke fouten en onvolkomenheden naar zijn inzicht een acceptatie in de weg staan;
- welke functionaliteit hij nog niet heeft kunnen testen als gevolg van de aangetroffen fouten;
- schoonheidsfouten en ongemakken die voor verbetering in aanmerking komen.

In het testrapport vermeldt hij bovendien:

- zijn conclusie: accepteert hij de geleverde applicatie of niet?
- wie de acceptatietest heeft opgesteld;
- door wie de acceptatietest is uitgevoerd.

Hij zorgt er verder voor dat alles wordt afgesloten met een handtekening;

## 11 Wat doe je met de resultaten van de test?

Uiterlijk op de laatste dag van de overeengekomen testperiode krijg je een schriftelijk en gedetailleerd testrapport van de opdrachtgever. Daarin beschrijft hij:

- welke fouten en onvolkomenheden hij in de applicatie heeft aangetroffen;
- welke fouten en onvolkomenheden naar zijn inzicht een acceptatie in de weg staan;
- welke functionaliteit hij nog niet heeft kunnen testen als gevolg van de aangetroffen fouten.
- ernstige fouten die de voortgang van de acceptatietest hebben belemmerd;
- fouten en onvolkomenheden die in het kader van een garantieregeling hersteld of verbeterd moeten worden nadat de applicatie is geaccepteerd;
- schoonheidsfouten en ongemakken die voor verbetering in aanmerking komen.

Tenslotte dient hij de aangetroffen fouten en onvolkomenheden ten opzichte van de overeengekomen specificaties te wikken en wegen. Bepalend voor acceptatie is of de aangetroffen fouten een productieve en operationele ingebruikname redelijkerwijs in de weg staan. Kleine fouten en onvolkomenheden kunnen ook na acceptatie in het kader van een garantieregeling worden hersteld.

In het testrapport of in de begeleidende brief vermeldt de opdrachtgever uiteraard zijn conclusie: accepteert hij de geleverde applicatie of verlangt hij herstel van fouten en onvolkomenheden?

De resultaten verwerk je in je product. Dit kan zijn meteen als er nog tijd voor is of tijd voor is gepland, dus nog voor de definitieve oplevering. Of na de definitieve oplevering als dit is afgesproken met de klant en is vastgelegd in de afspraken. Dit gebeurt dan in de vorm van een patch.

### Vraag

24. Hoe zou de garantieregeling geformuleerd kunnen zijn? Stel deze dus op.

## 12 Wanneer eindigt de acceptatietest?

In de algemene voorwaarden van software- en hardwareleveranciers is vaak een bepaling opgenomen die de testperiode beperkt tot bijvoorbeeld 2 weken na aflevering. Dat is erg kort, vooral wanneer de test nog voorbereid moet worden. De opdrachtgever zal proberen om een langere testperiode te bedingen, bijvoorbeeld 25% van de daadwerkelijke doorlooptijd van analyse, ontwerp en ontwikkeling met een minimum van 1 maand.

In de algemene voorwaarden van software/hardware leveranciers is vaak een bepaling opgenomen met de volgende strekking:

*Indien de leverancier geen testrapport ontvangt zal de applicatie gelden als geaccepteerd op de eerste dag na de testperiode.*

Wanneer de leverancier de overeengekomen testperiode overschrijdt zonder een testrapport met de ontdekte tekortkomingen naar jou te sturen, dan loopt hij het risico dat herstel van alle tekortkomingen voor zijn rekening is.

### Vraag

25. Is het voor jou als leverancier gunstig of ongunstig als de opdrachtgever een testrapport binnen de testperiode levert? Verklaar je antwoord.

## 13 Wat gebeurt er na afloop van de acceptatietest?

Als de opdrachtgever de applicatie heeft geaccepteerd dan kan hij deze in gebruik nemen.

Als hij de applicatie niet heeft geaccepteerd dan zul jij je moeten inspannen om de gerapporteerde fouten en onvolkomenheden naar beste vermogen te herstellen. In de algemene voorwaarden van software- en hardwareleveranciers is vaak een bepaling opgenomen met de volgende strekking:

*De applicatie zal gelden als geaccepteerd op het moment dat de in het testrapport genoemde fouten zijn hersteld.*

Er volgt in dat geval geen tweede algemene acceptatietest van de verbeterde software. De eerste en enige acceptatietest dient zoveel mogelijk fouten en onvolkomenheden aan het licht te brengen.

Het komt ook voor dat de overeenkomst voorziet in een tweede acceptatietest. Daarbij kan de bepaling opgenomen zijn dat de opdrachtgever het recht heeft de overeenkomst te ontbinden indien de software opnieuw niet geaccepteerd wordt.

Je moet er rekening mee houden dat niet alle aangetoonde fouten jouw verantwoordelijkheid zijn. Het is denkbaar dat bepaalde geconstateerde fouten veroorzaakt worden door fouten in hardware of software van derden, bijvoorbeeld meegeleverde componenten, of runtime-voorzieningen van het Operating System, incompatibele drivers. Jij hebt de verantwoordelijkheid voor het onderzoek en herstel van dergelijke fouten wellicht uitgesloten in de algemene voorwaarden. In dat geval moet je op basis van je licentie voor het betreffende component of voorziening de desbetreffende derde aanspreken. Je kunt daarbij te maken krijgen met verschillende complicaties:

- Je hebt een bewijslast met betrekking tot een fout die misschien diep verstopt zit in ontoegankelijke software.
- Voor wiens rekening is het onderzoek naar de fout?
- De door jou aangesproken derde ontkent wellicht de fout - al dan niet met een voor jou onhanteerbare technische explicatie - en verwijst je terug naar de leverancier.
- De door jou aangesproken derde verlangt dat je de laatste versie van zijn software gebruikt, maar die is niet compatibel met de software van de leverancier.
- De door jou aangesproken derde is slechts gehouden aan zijn overeenkomsten met jou, niet aan de bepalingen in je overeenkomst met jouw opdrachtgever. Zo is de derde partij bijvoorbeeld niet gebonden aan de geplande datum van ingebruikname.
- De gebruiksrechtsovereenkomst met de door jou aangesproken derde voorziet wellicht niet in het onderzoek of herstel van fouten.

## Vraag

26. Hoe zou jij bovenstaande gemarkeerde zin in 'normaal' nederlands formuleren?

27. Als er bij de acceptatietest fouten en tekortkomingen aan het licht komen, wordt er dan een tweede acceptatietest gedaan? Of worden alleen deze zaken getest?

## 14 Wanneer kan de opdrachtgever de geleverde applicatie in gebruik nemen?

Hij kan de applicatie in gebruik nemen nadat hij de geleverde applicatie heeft geaccepteerd.

In de algemene voorwaarden van software/hardware leveranciers zijn vaak bepalingen opgenomen met de volgende strekking:

*Gedurende de testperiode is het niet toegestaan om de applicatie voor productieve of operationele doeleinden te gebruiken.*

De applicatie zal reeds gelden als volledig geaccepteerd indien daarvan vóór het moment van acceptatie enig gebruik voor productieve of operationele doeleinden wordt gemaakt.

Wanneer de opdrachtgever de applicatie vóór acceptatie in gebruik neemt, kun je het herstel van alle resterende fouten voor zijn rekening beschouwen.

## Vraag

28. Wat is het risico voor de opdrachtgever indien hij de applicatie al in gebruik neemt voordat deze officieel geaccepteerd is?

## 15 De opdrachtgever wordt niet vrolijk van het volgende:

De IT is een onvolwassen bedrijfstak. Zelfs bij gerenommeerde leveranciers van een applicatie komt het regelmatig voor dat de geleverde opstelling in ernstige mate tekortschiet. Enkele voorbeelden:

- De geleverde software laat zich niet goed installeren, laat staan testen.
- De geleverde applicatie blijkt andere systeemeisen te kennen dan met jou overeengekomen, waardoor de testomgeving niet bruikbaar is.
- De geleverde applicatie vertoont allerlei gebreken en tekortkomingen, maar de functionele specificaties zijn onvoldoende specifiek om van ondubbelzinnige fouten te kunnen spreken.

- De gebruikershandleiding ontbreekt, of is onduidelijk.
- Er doen zich regelmatig niet-reproduceerbare fouten voor.
- Problemen die in de testomgeving reproduceerbaar zijn doen zich in jouw testomgeving niet voor.
- De performance is zeer slecht, maar jij vindt de performance normaal en je stelt voor dat de opdrachtgever krachtiger computers aanschafft.
- Jij schrijft fouten toe aan gebreken en tekortkomingen in software van derden.
- Jij hebt een bepaalde eis of wens die in de ontwerpfase is besproken niet uitdrukkelijk vastgelegd in de functionele specificaties en bij oplevering blijkt de software op dit punt niet te voldoen.
- Jij beweert dat bepaalde beperkingen of tekortkomingen niet kunnen worden verbeterd in verband met de beperkingen van een bepaalde technologie.
- Jij stuurt gedurende de testperiode in hoog tempo verbeteringen ("patches"), waardoor het testproces moeilijk te beheersen is.

Het is van belang dat de tester alle fouten, gebreken en onvolkomenheden die de voortgang van de acceptatietest belemmeren onmiddellijk meldt. Hij dient de fout tot in detail aan te geven en op welke wijze en in welke mate de voortgang van de acceptatietest wordt belemmerd.

Controleer zonodig de systeemeisen of compatibiliteitseisen.

Het is belangrijk dat je er in slaagt om op redelijke termijn de geconstateerde tekortkomingen te herstellen. Lukt dat niet, dan wordt het voor de leverancier tijd voor juridisch advies. Hij zal nagaan of hij je in gebreke kan stellen, de overeenkomst ontbinden of zelfs schadevergoeding eisen.

## 16 Acceptatietest uitbesteden

Door het uitvoeren van een deugdelijke acceptatietest kun je de kwaliteit en betrouwbaarheid van de software verbeteren. Het uitvoeren van een acceptatietest is echter verre van eenvoudig. Je krijgt te maken met de volgende complicaties:

- De functionele specificaties moeten tot in detail worden vertaald naar concrete en reproduceerbare test-scenario's.
- De applicatie moet worden getoetst op in de informatica algemeen aanvaarde uitgangspunten van kwaliteit en deugdelijkheid.
- Er moeten reproduceerbare performance-metingen worden uitgevoerd.
- Bij complexe en grootschalige acceptatietests kan het wenselijk zijn om het testproces geheel of gedeeltelijk te automatiseren.

Een acceptatietest is primair een zaak van de gebruikers en materiedeskundigen. Deskundigheid op het terrein van de Software Engineering is echter vereist. Deze deskundigheid zal niet van jou betrokken worden.

Verschillende bedrijven kunnen bij het uitvoeren van de acceptatietest van dienst zijn met testleiding en advies. Bovendien kunnen deze bedrijven je testplan schrijven, in samenwerking met gebruikers en materiedeskundigen test-scenario's en test-sets ontwikkelen, testresultaten analyseren en het testrapport opstellen.

Vraag

29. Wat is een testset?



## 17 Kosten

Als vuistregel kun je hanteren dat de kosten van een deugdelijke acceptatietest 20 tot 40 % van de kosten van de software bedragen. Je mag van de opdrachtgever verwachten dat hij een redelijke inspanning levert, aangezien jij je anders in de garantieperiode kunt beroepen op de bepaling dat het herstel van fouten die tijdens de acceptatietest gevonden hadden kunnen worden voor zijn rekening is.

Verder zal hij de onzekere kosten van herstel van fouten na de testperiode laten meewegen. Een extra investering in de acceptatietest kan lonend zijn, met name wanneer hij voor het herstel van fouten afhankelijk is van jou als enige leverancier.

## 18 Referenties

Software testing in the real world - improving the process, Edward Kit, Addison-Wesley, 1995.  
System Development Methodology, W.S. Turner e.a., Cap Gemini Publishing, 1990.

## 19 URL

<http://softwaretest.pagina.nl>

<http://www.veritest.com>

<http://www.sqatester.com/testingtips/index.htm>

<http://www.testnet.org>

Voorbeeld:

Beginsituatie	Verwachte resultaat	Actie en resultaat	Vraag	Antwoord of opmerking
Het programma is opgestart en het inlogscherf wordt getoond.			Wat vindt u van het kleurenschema?	
			Is het lettertype goed?	
			Staat het logo aan de goede kant?	
			Wordt het logo groot genoeg getoond?	
Het programma is opgestart en het inlogscherf wordt getoond.	-	<ul style="list-style-type: none"> <li>- Klik met de muis in het invoerveld naast "Login".</li> <li>- Voer het volgende in: Test123.</li> <li>- Druk op &lt;Enter&gt;</li> </ul>	Wat vindt u van de positie van de melding?	
	-	Er wordt nu een melding getoond dat het wachtwoord niet is ingevuld.	Wat vindt u van de tekst van de melding?	
	-		Klik op "OK". Wat vindt u van de manier van teruggaan naar het inlogscherf?	

Noot: Dit lijkt goed, maar er is wat over te zeggen. Wat is 'goed'? Wat is 'groot genoeg'?

Beter dus formuleren als “zoals afgesproken in .....”.

Het is een goed idee om een klant altijd zelf zonder begeleiding te laten testen.

Vraag:

Waarom is dit een goed idee?