



Vereniging TestNet p/a Diadeemstraat 91 1336 TT Almere
www.testnet.org secretaris@testnet.org

Van de redactie

Door Meile Posthuma

tnn@testnet.org

Voor onze meeste leden zal de vakantieperiode weer aanbreken. Normaal gesproken een tijd dat er weinig nieuws is: kranten worden dunner, maar dat geldt niet voor TNN. We hebben gelukkig voldoende enthousiaste leden die bijdragen willen leveren aan TNN. De redactie gaat de zomermaanden dan ook gewoon verder met het vergaren van kopij. Ook in deze TNN is weer voldoende voor ieders gading te vinden. Wat staat er in deze TNN? Een oproep van de werkgroep Testdata, Riks Column, open source tools, hoe vergaat het ZZP'ers (ook in volgend nummer), alweer drie boekrecensies en de vaste rubrieken. Maar eerst zal Bob hieronder ingaan op wat geografische metrieken.



Van de Voorzitter

Door Bob van de Burgt

voorzitter@testnet.org

Wie zijn onze leden?

In september 2005 heb ik diverse inzichten gegeven in ons ledenbestand dat toen bestond uit 407 leden. Aangezien we de laatste twee jaar een enorme toestroom van nieuwe leden hebben doorgemaakt is het leuk om eens te kijken hoe ons ledenbestand er nu uitziet ten opzichte van 2005.

IN DIT NUMMER

Van de redactie	1
Van de voorzitter	1
Van de penningmeester	3
Werkgroep Testdata OPROEP	3
Ervaringen van een test-ZZPer	3
Open source test tools	5
Rik's Column	7
The making of...	8
Vijf vragen aan...	9
De dag van...	12
Boekrecensies Test-M	14
Boekrecensies Van Implementatie tot Acceptatie	15
Boekrecensies Implementing Lean Software Development	16
En nu in het echt	18
Petje af voor TestNet Quiz	19
Evenementen	21

TestNet Nieuws verschijnt eenmaal per kwartaal. Kopij aanleveren per e-mail aan de redactie. Het is niet toegestaan om de nieuwsbrief of delen eruit zonder bronvermelding over te nemen.

De volgende overzichten zijn gebaseerd op de ledenadministratie van 29 april 2009.

Waar wonen we?

Ten opzichte van 2005 zijn Amstelveen en Hilversum uit de top 10 gevallen. Den Haag en Arnhem zijn nieuw in de top 10. De woonplaats met de grootste stijging is Amsterdam met een ledengroei van 61 ten opzichte van 2005.

Colofon

Redactie

Paul Beving
Hylke ten Cate
Hans van Loenhoud
Meile Posthuma
Johan Vink
tnn@testnet.org

Bestuur

Bob van de Burgt
Rob Baarda
Meile Posthuma
Huib Schoots
Michiel Vroon
Bart Watertor

Voorzitter

Vice-voorzitter, Marktverkenning & Werkgroepen
Penningmeester
Informatievoorziening & Beheer
Evenementen & Thema-avonden
Secretaris, 2e penningmeester,

	2009		2005	
	# leden	%	# leden	%
Totaal # leden	1553		407	
Amsterdam	87	6%	26	6%
Utrecht	65	4%	13	3%
Rotterdam	61	4%	10	2%
Den Haag	58	4%	<7	
Almere	46	3%	14	3%
Apeldoorn	30	2%	12	3%
Amersfoort	28	2%	11	3%
Eindhoven	25	2%	9	2%
Groningen	31	2%	<7	
Arnhem	26	2%	7	2%

De meeste leden wonen in Zuid-Holland. De top 5 van het aantal leden per provincie ziet er als volgt uit:

Provincie	# leden	%
Zuid-Holland	341	22%
Utrecht	267	17%
Noord-Holland	244	16%
Gelderland	171	11%
Noord-Brabant	156	10%

Evenementen in het NBC lijkt dus qua locatie nog steeds een zeer goede keuze.

Waar werken we?

De top 10 werkgevers van TestNet leden ziet er als volgt uit:

	2009		2005	
	# leden	%	# leden	%
Totaal # leden	1553		405	
Logica	125	8%	44	11%
Ordina	110	7%	13	3%
Atos Origin	62	4%	12	3%
Sogeti	46	3%	26	6%
Cappgemini	40	3%		
Qualityhouse	33	2%	17	4%
Valori	37	2%	11	3%
Polteq IT Services	29	2%		
Test Value BV	26	2%		
Getronics	23	1%	12	3%

Verdwenen uit de Top 10 zijn ING, ABN Amro en Kadaster. Nieuw in de top 10 zijn Cappgemini, Polteq en Test Value. De organisatie met de grootste stijging in het aantal leden is Ordina met een stijging van 97 leden.

Buiten het feit dat het aardig is om te zien waar onze leden wonen en werken is het natuurlijk geweldig dat TestNet meer dan 1500 leden heeft en weer hard onderweg is richting de 1600.

Ik hoop ook dat we op het voorjaarsevenement een nieuw record gaan neerzetten en de 600 aanwezige leden halen. Tot 22 juni!

Bij de berekeningen zijn leden waarvan bepaalde gegevens niet bekend zijn in onze administratie niet meegeteld.

Van de penningmeester

Door Rob Baarda

penningmeester@testnet.org

De ALV heeft ingestemd met mijn aantreden als penningmeester. Dat is een boeiende functie tot nu toe. Mijn voorganger Han Toan Lim heeft de overgang van administratiekantoor In Capo naar A Solution prima geregeld. De overgang kan als af worden beschouwd. De nieuwe administrateur is nu de procedures aan het fine-tunen.

Zoals je hebt kunnen merken is de contributie-inning al gestart. Velen hebben hun contributie overgemaakt. Degenen die dat nog niet hebben gedaan, wordt vriendelijk verzocht dit zo spoedig mogelijk te doen. Er komen een paar dure evenementen aan, de vereniging kan het geld goed gebruiken.

De door de ALV goedgekeurde begroting wordt goed gevolgd. De grotere opkomst op de thema-avonden leidt tot overschrijding van de uitgaven maar dat wordt gecompenseerd door het niet doorgaan van het mini-event over EuroSTAR. De feitelijke uitkomst van de begroting zal bepaald worden door de opkomst op de events en de niet-betaalde contributie.

Uw penningmeester houdt de vinger aan de pols.



Werkgroep Testdata – OPROEP!

Door Edwin van Vliet en Marco de Haan

wergroepen@testnet.org

De werkgroep Testdata heeft tot doel om testdata samen te stellen voor

testend Nederland. Testdata die vrij is van privacygevoelige informatie, maar tegelijk gebruikmaakt van basisgegevens (onder andere postcodetabellen) uit productie voor optimale mogelijkheden om productielijke te kunnen testen.

Hiermee hopen we de 'moeilijkheden van testdata' om te zetten naar een 'kracht van testdata' voor iedere tester.

Heb jij interesse om met testcollegae mee te denken over de behoeftes van testende organisaties met betrekking tot testdata, analyses te maken of om bij bedrijven testdata ten behoeve van de basisgegevens te organiseren, stuur dan alsjeblieft een mailtje naar wergroepen@testnet.org

Meer informatie vind je op de site van Testnet, onderdeel werkgroepen.



Ervaringen van een Test-zzp'er

Door Richard Kwint

Richard@kwinttestservices.nl

Sta mij toe mijzelf even voor te stellen: mijn naam is Richard Kwint en twee jaar geleden ben ik als zzp'er begonnen. In onderstaand verhaal lees je een aantal van mijn persoonlijke ervaringen als (beginnend) zelfstandig tester en zal ik een paar tips geven voor degenen die overwegen ook als test-zzp'er te beginnen.

Medio 2007 heb ik de overstap gemaakt van werknemer (tester bij de Dienst ICT Uitvoering van het ministerie van LNV) naar zelfstandig tester zonder personeel. Aanleiding was het voornemen binnen het ministerie om op termijn uitvoerende taken zoveel mogelijk uit te besteden.

Aangezien ik het meeste plezier beleef aan het uitvoerende werk, was mij duidelijk dat mijn toekomst na bijna zeven jaar testen niet meer bij het ministerie lag. Maar waar dan wel? Dat ik uitvoerend wilde testen was voor mij duidelijk. Het liefst in een grote en complexe omgeving. De volgende vraag was voor wie ik dan het liefst zou willen werken. Na rijp beraad ben ik uiteindelijk tot de conclusie gekomen dat ik het liefst voor mijzelf wil werken.

Een keuze met de meeste vrijheid om leuke klussen te zoeken, maar ook met de meeste (financiële) onzekerheid.

Nadat mijn besluit vaststond, heb ik zo snel mogelijk zaken in gang gezet om alle praktische zaken te regelen: ondernemingsplan schrijven, inschrijven KvK, de Belastingdienst, bankzaken, verzekeringen, accountant regelen, ontslag nemen en... een bedrijfsnaam bedenken. Het moet een naam zijn die de lading zo goed mogelijk dekt... Wie ben ik?: Richard Kwint, eenmanszaak. Richard is een leuke voornaam, maar Kwint bekt beter. Akkoord. Wat doe ik? Testen. Kwint Test. Hmm..., de essentie is zo wel vastgelegd. Nog een kleine verfijning en ziedaar: de geboorte van Kwint Test Services!

Nu is het zaak om een eerste klus te vinden. Via een collega test-zzp'er hoorde ik van een leuke klus bij een grote verzekeringsmaatschappij. Na een aantal informele gesprekken met een interne testcoördinator en een gesprek met de detacheerder die de verzekeringsmaatschappij als klant heeft, werd ik uiteindelijk uitgenodigd voor een intakegesprek en... met succes! Helaas duurde het nog een aantal weken voordat ik kon beginnen,

maar gelukkig kon ik nog even interen op mijn spaargeld.

Al met al een vliegende start, maar de financiële crisis is nu ook bij mij aanbeland. Ik heb mijn uurtarief naar beneden bijgesteld en nu ziet het ernaar uit dat ik binnen afzienbare tijd weer beschikbaar ben voor een volgende klus.

Als ik de balans opmaak van mijn ervaringen als zzp'er, is die per saldo positief. Het meest positieve aspect vind ik het grote gevoel van vrijheid. Als meest negatieve punt noem ik het tijdsbeslag van administratieve en bijkomende zaken. Hierdoor zijn veel van mijn sociale contacten op de spaarstand komen te staan.

Ten slotte nog een paar tips voor testers die overwegen om ook zelfstandig te worden:

Ga niet over één nacht ijs. Ga goed bij jezelf te rade of werken als zelfstandige iets voor jou is.

Staar jezelf niet blind op het uurtarief. Omzet is geen inkomen! Geef jezelf een normaal salaris en reserveer voor slechte tijden (en de Belastingdienst).

Neem een goede (AA-)accountant. Dat zal je veel tijd (en geld) besparen. Ga gerust langs voor een oriënterend gesprek. Voor de (hoge) noorderlingen onder jullie: Accountantskantoor "Het Hogeland" is mij tot op heden zeer goed bevallen. (voorheen Riemeijer Accountancy) met vestigingen in Assen, Emmen en Delfzijl.

Als laatste een uitstekende site met veel informatie en praktische tips voor (startende) zzp'ers: www.lancelots.nl.

Open source test tools

Door Ferrie Wolff en Mark Schut

ferrie.wolff@sogeti.nl en mark.schut@sogeti.nl

Naast commerciële tools ter ondersteuning van het testproces komen er steeds meer open source testtools beschikbaar om testen te ondersteunen. De meeste van deze tools zijn gratis te verkrijgen op internet, wat het uiteraard bijzonder interessant maakt. Wij vroegen ons af of de open source testtools ook functioneel en technisch interessante alternatieven zijn.

Net als bij de commerciële testtools is bij open source testtools een driedeling te maken in performance testtools, functionele testtools en testmanagement tools. Hieronder zullen we aan de hand van deze driedeling in het kort de meest opvallende verschillen tussen de commerciële en open source testtools uiteenzetten.

Testmanagement tools

Testmanagement tools worden gebruikt ter ondersteuning van het testproces, waarbij gedacht moet worden aan het opslaan en beheren van requirements, testware en bevindingen. Bij commerciële testmanagement tools is veelal een webinterface beschikbaar waarin testers en testmanagers informatie kunnen opslaan als testcases, bevindingen of documentatie. Daarachter zit meestal een database als SQL Server of Oracle, is een mailfunctionaliteit beschikbaar en een aparte administrator omgeving om de tools te onderhouden. Voor open source testmanagement tools geldt in grote lijnen hetzelfde, met het verschil

dat er uiteraard open technieken worden gebruikt. Bijvoorbeeld een Apache webserver, een MySQL database en een PHP webinterface. Dit zijn open technieken waarbij vaak evenveel mogelijk is als bij de commerciële tools. Echter, bij open source testmanagement tools blijft de functionaliteit bij het merendeel alleen beperkt tot het opslaan en opvragen van de gegevens. De rapportages uit deze tools zijn summier. Verder is het bij de meeste open source testtools niet mogelijk een koppeling te maken tussen bijvoorbeeld requirements, testcases en bevindingen. De meeste commerciële testmanagement tools ondersteunen dit soort koppelingen juist wel, wat hun grote kracht is.

Terwijl bij commerciële tools een compleet installatieprogramma wordt meegeleverd, moet bij open source de omgeving zelf worden opgezet. Dat vraagt dus meer technische kennis. Daarna is het veelal gemakkelijk om met open source tools een project te beginnen en te onderhouden.

We zien dus bij de open source testmanagement tools dat de basisfunctionaliteit, zoals het opslaan van data, aanwezig is, maar de rapportage en analyse te wensen over laat.

Performance tools

Performance testtools worden ingezet wanneer de belasting van een systeem getest moet worden. Performance testtools stellen de tester in staat om een aantal gebruikers te simuleren om te kijken hoe het systeem zich gedraagt onder een reële belasting.

Eén van de succesfactoren voor een goede performance testtool is de

mogelijkheid om verschillende protocollen aan te kunnen. Deze mogelijkheid zien we dan ook terugkomen bij de meeste commerciële performance testtools. Als we kijken naar open source performance testtools en welke protocollen ondersteund worden, zijn het veelal open source protocollen; open source blijft open source. Denk hierbij aan de bekende http of xml protocollen. Zodra een systeem als Oracle of SAP getest moet worden, is er geen enkel open source testtool te vinden die dit protocol ondersteunt.

Ook de aansturing van open source performancetools lijkt lastig te zijn. Functionaliteit voor het aansturen van een aantal pc's tegelijk (bijvoorbeeld de load-generatoren) en het monitoren van verschillende systemen tijdens een performancetest is veelal niet terug te vinden. Er kan eventueel gebruikgemaakt worden van de andere open source monitoring tools, maar deze gegevens kunnen niet direct worden gekoppeld aan de resultaten van een test. Natuurlijk kunnen we aan het einde van een test de verschillende gegevens zelf aan elkaar koppelen, via Excel of iets dergelijks, maar een kant-en-klare rapportage zoals we die bij de commerciële tools vinden, ontbreekt bij de meeste open source tools.

Het parameteriseren van scripts blijkt ook lastig; het is bijna niet mogelijk om dynamische data mee te sturen. Ook correlatie van dynamische data afkomstig van de server side is veelal niet mogelijk. Dit in tegenstelling tot de meeste commerciële testtools, waarbij het parameteriseren en correleren een standaard functionaliteit is.

Belangrijke basisfunctionaliteiten ontbreken bij open source performance testtools. Ze zijn vooral te gebruiken voor snelle en niet al te complexe performance testen.

Testtools voor geautomatiseerd functioneel testen

Functionele testtools maken het voor testers mogelijk om via record- en playback functionaliteit acties op te nemen en uit te voeren waardoor snel en herhaaldelijk (regressie)testen kunnen worden herhaald.

Op het gebied van testtools voor functioneel testen zijn veel soorten open source oplossingen te vinden. Het gaat van kleine modules tot aan volledige applicaties. Een goed voorbeeld hiervan is Selenium, die geïntegreerd is met een web browser waarna het mogelijk is om acties van een gebruiker op te nemen in een script.

Naast applicaties met een eigen GUI en afgebakende functionaliteit zijn er ook vele libraries te vinden die kunnen worden geïmporteerd in een ontwikkelomgeving. Deze libraries bevatten specifieke methoden en functies voor het uitvoeren van tests op een applicatie, zoals het invoeren van velden op een java formulier. Deze libraries zijn met name bestemd voor unit testen, maar het is goed mogelijk om hier als tester zelf mee aan de slag te gaan en bijvoorbeeld een geautomatiseerde testsuite te bouwen voor testers. Dit kost echter veel meer tijd in een open source tool dan in commerciële tools. Het grote verschil bij open source tools zit met name in de object handling. In commerciële

testtools zijn speciale functies beschikbaar (zogenoemde Add in's) die het herkennen (en aansturen) van objecten op het testobject vergemakkelijken. Dit ontbreekt in de meeste open source testtools. Ook functionaliteit zoals controles en rapportages zullen van begin af aan zelf gemaakt moeten worden.

Open source testtools voor functioneel testen zijn vooral bruikbaar voor ontwikkelaars die hun eigen code willen testen. Het bouwen van een geautomatiseerde testsuite voor een testorganisatie zal meer tijd in beslag nemen en de nodige technische kennis vereisen.

Tot slot

Het aanbod van open source testtools is groot. In dit stuk zijn algemene punten besproken die ons opvielen bij het bekijken van verschillende tools. Uiteraard bestaan er tools die iets beter zijn uitgewerkt dan andere. Verder zijn nog talloze open source testtools beschikbaar die we niet hebben bekeken.

In het algemeen valt op dat open source testtools aardig in de buurt komen van commerciële tools, maar meestal nèt niet. Dit komt vooral door de specifieke eigenschappen die een open source testtool vaak heeft; bijvoorbeeld alleen ondersteuning van open source protocollen.

Het hangt uiteindelijk af van de testen applicatie, de doelstellingen van de tester en de organisatie of het zinvol is om een open source testtool in te zetten in plaats van een commerciële testtool.

Riks Column



rik.marselis@sogeti.nl

Begin mei kreeg ik van de TNN redactie het eervolle verzoek om voortaan een 'column' voor de TNN te schrijven. Waarvoor dank!

Even piekeren, wat is een column? Mijn vertaalwoordenboek geeft 'kolom, zuil, pilaar, waarop iets steunt', ik ga dus steun geven. Het Engelse woordenboek zegt: 'regular feature article'. Dat helpt. Van Dale geeft meer houvast: 'regelmatige (min of meer kritische) bijdrage'. Daar kan ik wat mee. Dus voortaan een min of meer kritische beschouwing als steuntje voor testers.

Laat ik als eerste onderwerp de grote hype van dit moment nemen: AGILE.

Onmiddellijk geef ik toe dat Agile testen een eigen plaats binnen de IT verdient. Maar zoals met alle eerdere hypes wordt het nogal overschat. Agile is een van de vele benaderingen, maar niet de oplossing voor alle problemen. Sterker nog, Agile kan het begin van vele nieuwe problemen zijn. Ik heb diverse Agile experts beluisterd of gelezen en er doemde in ieder geval één algemeen beeld op: Agile testen kan alleen gedaan worden door bovengemiddeld goede medewerkers. Laat ik als definitie van bovengemiddeld heel ruimhartig aanhouden dat dit de bovenste 49% van de testers zijn. Dat betekent dus

dat minimaal DE HELFT van alle testers NIET in aanmerking komt om in Agile testprojecten aan de gang te gaan. OK, in paragraaf 30.3.4 van het boek van Anko en Eric lees ik 'het trainen van de testers is dan een noodzakelijke stap', dus misschien kunnen mensen voldoende bijleren. Dat is ook wel nodig, want recent was ik in een zaal met ruim honderd ervaren testers en een vlotte inventarisatie van Leo bracht aan het licht dat ongeveer een derde van hen op enigszins gestructureerde wijze testspecificatietechnieken toepast (tweederde dus niet). Hier blijkt: ook zonder Agile is 'het trainen van de testers een noodzakelijke stap'. We willen wel allemaal graag mooie toolsuites om testgevallen vast te leggen en uit te voeren. Maar eerst even nadenken over welke testgevallen eigenlijk nodig zijn, is lastig. Maar goed, Agile is een prima verbreding van alle mogelijkheden om de kwaliteit van een systeem te meten en de resterende risico's vast te stellen. Iemand die echter beweert dat in zijn organisatie voortaan alleen nog maar Agile getest gaat worden moet zo snel mogelijk weer met beide benen naar de grond getrokken worden.



The making of...

Door Michiel Vroon en Rik Marselis

evenementen@testnet.org

Twee Evenementen; daarvoor zond de evenementencommissie in februari een 'call for papers' rond. Nou, dat hebben we geweten. Een nieuw record, 96 inzendingen!! Alle inzenders natuurlijk hartelijk bedankt! Toen begon voor de commissieleden de zware taak om voor beide evenementen een evenwichtig programma samen te stellen. Elk

commissielid (en dat zijn er dus acht) heeft alle inzendingen gelezen (dat kost je toch wel een substantieel deel van je weekend, maar alles voor de goede zaak hè ;-). Iedereen heeft elke inzending op verschillende aspecten beoordeeld, inhoudelijke criteria als 'vernieuwend' en 'helder verhaal', maar ook 'aansluiting bij het thema' (we bedenken die thema's tenslotte niet voor niets). En natuurlijk 'buikgevoel', want door de jaren heen hebben we geleerd dat het 'Fingerspitzengefühl' een prima indicatie is. Inhoud weegt echter het zwaarst.

Het viel ons op dat de kwaliteit van de inzendingen elk jaar stijgt. Voorheen waren we nog wel eens vlot klaar omdat een deel van de inzendingen ondoorgrondelijk was. De aansluiting met het thema ('testen – de langste dag' op 22 juni en 'testen in verschillende werelden' op 22 september) was niet altijd even duidelijk, maar verder waren er vele boeiende verhalen.

Op 22 april kwamen we bijeen voor het maken van de definitieve keuzes. Onder het genot van de inmiddels traditionele satétjes van het NBC zijn we gaan selecteren. Met de boek-tracks van het voorjaarsevenement waren we vlot klaar. Evenveel inzendingen als ruimte, dus dat was mooi. De workshops van het najaarsevenement waren het tegenovergestelde. Er waren meer dan drie keer zo veel inzendingen dan de beschikbare ruimte. Dus we moeten helaas heel veel workshop inzenders teleurstellen. Maar meer dan de vier geplande workshops van een halve dag, dat gaat gewoon niet passen. We beraden ons trouwens nog even op hoe

er precies voor die workshops ingeschreven moet worden. Want de vier geselecteerde workshops hebben zo boeiende onderwerpen dat we bang zijn dat het de zalen uit gaat barsten. Bereid je dus vast voor op een 'wie het eerst komt, wie het eerst maalt'-principe! en houd je mailbox in de gaten.

De selectie van de presentaties voor beide evenementen hebben we zuiver en zonder politiek gehouden. Er is één simpele regel: dezelfde persoon en hetzelfde bedrijf slechts één keer per evenement. Verschillende commissieleden waren wel wat beteuterd, omdat ook diverse van hun eigen collega's hoog bleken te scoren en dan dus alleen de hoogst scorende persoon van het bedrijf geselecteerd is. Op volgorde beginnend bij de hoogste score hebben we de presentaties in het programma gezet. Nog een nieuwtje: voor beide evenementen hebben we ook een reservepresentatie geselecteerd, want je weet tenslotte nooit wat er nog kan gebeuren. Vervolgens hebben we gepuzzeld om tracks met bij elkaar passende onderwerpen te krijgen zodat TestNet-ers op basis van hun interesse een bepaalde zaal kunnen uitzoeken.

Binnen twee uur konden we trots constateren dat het ons weer gelukt is. Een nieuw record is het totale aantal sprekers op beide evenementen. In totaal meer dan 45 sprekers! Het voorjaarsevenement heeft negen boektracks, tien presentaties, een keynote en een fun sessie. Op het najaarsevenement pakken we nog grootser uit met vier workshops van vier uur, zestien presentaties en drie keynotes! We beginnen EuroSTAR al

aardig naar de kroon te steken, met dien verstande dat het bij ons een stuk goedkoper is, zelfs gewoon gratis voor leden en je hoeft er niet half Europa voor door te reizen!

Dus noteer het vast in je agenda: 22 juni en 22 september!



Vijf vragen aan ...

Door Jaap Boumans

Jaap.boumans@intraffic.nl



Ik vind testen een leuk vak want ...

Testen is 'inzicht geven in kwaliteit'. En dat doen we doorgaans door het zoeken naar fouten. Alle producten met software bevatten fouten. Hoe meer software, des te meer fouten. Hoe complexer de software, hoe groter de fouten. Bijna elk product bevat tegenwoordig software. De werking van apparaten wordt in toenemende mate bepaald door software. De software wordt steeds complexer en de toepassingen steeds kritischer. Je weet dus dat de fouten er zijn, maar nog niet waar ze zitten. Ik vind het een uitdaging is om zo veel mogelijk van die fouten te vinden. En dan vooral die vette fouten, waardoor er van alles mis gaat. Fouten in het ontwerp of zelfs in het concept. Het is een speurtocht waarbij je ook steeds meer van een product en de interactie met de omgeving ervan te weten komt. Het

geeft mij altijd de meeste voldoening als het om systemen gaat die in de maatschappij een belangrijke rol kunnen spelen. Zoals de spoorwegen, OV-chipkaart, medische systemen en televisie. Helaas blijkt het niet mogelijk om alle fouten te vinden en op te lossen voordat een product naar de markt gaat. Elk apparaat bevat fouten. Mijn auto geeft soms tegenstrijdige meldingen. M'n stereo schakelt zichzelf soms spontaan uit. Op de website van m'n zorgverzekering staat in rood aangegeven dat je een bepaald veld niet moet vergeten in te vullen. Als je met de tab-toets door het formulier loopt, wordt dat veld overgeslagen. En als je er postcode en huisnummer invult, komt er een verkeerd adres tevoorschijn. Mijn fietscomputer crasht als ik tijdens het rijden de tripteller op nul zet. Mijn telefoon ... Het testen gaat dus altijd door.

Het grootste misverstand over testen is ...

Dat verschilt nogal per organisatie. Ik kom nog steeds organisaties tegen waar men denkt dat testen zoiets is als 'kijken of het werkt'. Dat begint dus pas als de software is geleverd. In de meeste IT-organisaties hebben we dat gelukkig al achter de rug. Het misverstand dat ik daar nog vaak tegenkom is dat 'requirements based' testen een goede kwaliteit garandeert. Met minimaal één testgeval per requirement en een mooie trace matrix heb je alles gedekt. Niet dus. Meestal beschrijven de requirements zelf slechts een gedeelte van het gedrag van een systeem. En elk testgeval verifieert een requirement in een bepaalde situatie. Maar het systeem moet wel aan alle requirements tegelijk

voldoen. In alle mogelijke combinaties. Met een oneindig variabele historie. Veel fouten kan je met functioneel testen helemaal niet vinden, of alleen bij toeval.

Over vijf jaar zie ik mijzelf in de functie van ...

Vijf jaar is ver weg. In vijf jaar tijd kan er een hoop veranderen. Maar mijn ambitie is om leverancier van testdiensten te zijn voor projecten binnen InTraffic en externe opdrachtgevers. Nu spelen uren en kosten meestal de belangrijkste rol bij een overeenkomst. Over de invulling daarvan worden vaak slechts globale afspraken gemaakt. Ik geef er de voorkeur aan een dienst te leveren. Bijvoorbeeld 'een integratietest verzorgen', 'een teststrategie opstellen' of 'een testprocesverbetering doorvoeren'.

Daarbij moeten de te behalen doelen leidend zijn en de verantwoordelijkheid voor alle aspecten van de te leveren dienst bij het testteam liggen. Het testteam moet dan wel voldoende vrijheid van handelen hebben om naar eigen inzicht invulling te geven aan die verantwoordelijkheid, binnen de afgesproken grenzen van tijd, budget en kwaliteit.

Een tester moet zeker beschikken over de vaardigheid of kennis om ...

Niks moet. Iedereen kan fouten vinden maar om dat gestructureerd en effectief te doen en een gefundeerde uitspraak te kunnen doen over de kwaliteit van een systeem heb je wel kennis nodig van testmethodieken en technieken. Daarnaast helpt het als je

kennis hebt van het domein van het te testen systeem, zodat je in staat bent requirements en systeemgedrag te interpreteren. Verder heb je kennis nodig van de gebruikte ontwerpmethodiek om in staat te zijn blinde vlekken en fouten in het ontwerp te herkennen.

Omdat we het hebben over software is het ook belangrijk dat je kennis hebt van software en liefst ook enige ervaring met het ontwikkelen daarvan. Zodat je code kan reviewen, op de hoogte bent van risico's van de gebruikte ontwikkelmethoden en van getallen als 65536, van afrondingsfouten in berekeningen, strings met '/'n' of ['http://www.camerashop.com/Canon-PowerShot-SX200IS-Stabilized-Black/dp/B001SER45Q/ref=sr_1_1?ie=UTF8&s=photo&price=349.59&qid=1237373638&sr=1-1'](http://www.camerashop.com/Canon-PowerShot-SX200IS-Stabilized-Black/dp/B001SER45Q/ref=sr_1_1?ie=UTF8&s=photo&price=349.59&qid=1237373638&sr=1-1), gebruik van resources, zoals geheugen, de mogelijke verstoringen vanuit het OS.

Om veel fouten te vinden helpt het als je destructief bent ingesteld, analytisch en oog hebt voor details. In elk geval moet je een ontwikkelaar in z'n waarde laten terwijl je wijst op fouten in z'n code. En uiteindelijk moet je kunnen aanvaarden dat je eigen werk ook fouten bevat en dat niet alle fouten die je vindt worden opgelost.

In de toekomst hoop ik dat ??? binnen het testvakgebied is veranderd, omdat ...

Tijdens mijn studie werktuigbouwkunde heb ik geleerd bij het ontwerpen van een systeem te komen tot een wiskundig model van de gekozen oplossing. Door aan dat model te rekenen kan je de oplossing

evalueren en optimaliseren.

Bijvoorbeeld sterkte- en stijfheidberekeningen aan een constructie. Ook in de elektronica worden wiskundige modellen gebruikt om ontwerpen door te rekenen (al was het maar de wet van Ohm) en in de civiele techniek is het al niet anders. Het gebruik van wiskundige modellen is in de meeste takken van industrie al eeuwen een beproefde methode.

Zo niet in de ontwikkeling van software. Softwareontwikkeling is een vreemde industrie. We maken een ontwerp waarvan we weten dat er fouten in zitten. Op basis daarvan schrijven we code waarvan we weten dat er nog meer fouten in zitten. De compiler mag de eerste fouten eruit halen en daarna proberen we gewoon of het werkt. We gaan we er nog eens met de debugger doorheen en dan doorlopen we een aantal testfasen om zoveel mogelijk, maar nooit alle fouten er uit te halen. De uiteindelijke gebruiker vindt in de praktijk ook nog een aantal fouten en als hij geluk heeft krijgt hij een update, waarin die fouten zijn opgelost. De prijs die hij voor die update betaalt, is de introductie van nieuwe fouten.

Ik hoop dat we in de toekomst het zwaartepunt van testen zover in het ontwikkelproces naar voren kunnen brengen dat we kunnen voorkomen dat fouten in software worden aangebracht. Door de toenemende complexiteit van software komt er een moment dat we met de huidige manier van testen niet meer in staat zijn een behoorlijke dekking te behalen binnen een acceptabel tijdsbestek. Het gebruik van formele modellen zou dan wel eens een uitkomst kunnen zijn.

Ik geef de vraag door aan ..., omdat ...

Jeanne Hofmans. Ik heb veel met Jeanne samengewerkt. Ze is een gedreven tester. Ze is de enige tester die ik ken die is afgestudeerd in testen (aan de UvU). Of het daardoor komt, weet ik niet maar Jeanne is buitengewoon goed in staat de theorie van het testen op een succesvolle manier in projecten in de praktijk te brengen.



De Dag van ... Iedereen weer tevreden!

Door Wilcor Toele

wilcor.toele@active-professionals.nl



Om zes uur gaat in huize Toele de wekker. Eerst ga ik eruit en maak dan een ontbijt klaar voor mij en mijn vrouw. Na even te hebben bijgepraat duik ik de badkamer in om er vervolgens herboren uit te komen. Vaak trek ik een pak aan omdat ik verplichtingen buiten de deur heb. Na mijn lieve vrouw gedag te hebben gezegd en natuurlijk een kus te hebben gegeven stap ik rond kwart voor zeven in de auto om te genieten van de dagelijkse beslommingen op de Nederlandse snelwegen.

De reis naar kantoor

Ik woon in het mooie groene hart van Nederland, in Alphen aan den Rijn.

Voor de echte Alphenaren nog steeds een dorp maar voor mij, en vele andere import Alphenaren, begint het toch aardig op een stad te lijken met meer dan 70.000 inwoners. Via de N11 bereik ik bij Bodegraven de oprit van de A12 richting Gouda en Den Haag. Hier begint de eerste irritatie, namelijk vrachtwagens die te langzaam de snelweg oprijden en collega vrachtwagens die dan zo nodig de tweede baan moeten kiezen, anders komen ze niet vooruit. Soms met een beetje stunt- en vliegwerk (ik rijd momenteel in een Renault Laguna GT) lukt het mij nog steeds om snel de derde baan te vinden en dan snelheid te maken. Echter, dan komt knooppunt Gouda om de hoek kijken, waar ik de afslag naar de A20 neem, maar vaak toch nog even het staartje meeneem van de file op de A12.

Gelukkig is de A20 goed begaanbaar op dit tijdstip, zodat ik via de A16 rond kwart over zeven de afslag Capelle aan den IJssel kan nemen en dan na de afslag Rivium / Fascinatio ons kantoor weet te vinden.

Vaak ben ik degene die het eerste op kantoor is. Dus na de gebruikelijke handelingen settel ik mij achter mijn bureau om mijn laptop op te starten en mijn eerste kopje sterke espresso te nuttigen. Thuis heb ik een Nespresso, dus soms is het even slikken. Maar goed, na de eerste drie bakjes begin je het verschil al niet meer te proeven.

Mijn werkzaamheden

Voordat ik verder ga met mijn verhaal wil ik eerst even uitleggen dat ik mij niet dagelijks bezighoud met het zelf uitvoeren van testtrajecten en projecten bij klanten. Ik ben als

Managing Partner bij Active Professionals verantwoordelijk voor de dagelijkse gang van zaken van de test consultancy BV. Dit houdt in dat ik veel contacten heb met potentiële klanten om te kijken of onze consultants passen op testaanvragen. Verder veel communicatie met de medewerkers die op een opdracht zitten: hoe gaat het, waar lopen zij tegenaan, zijn er andere mogelijkheden bij de klant? We hebben een aantal verschillende business units, dus we richten onze pijlen niet alleen op de testmarkt. Zo kunnen we met zijn allen kijken waar de mogelijkheden voor cross selling liggen.

Door de huidige omstandigheden zijn we ook aan het kijken hoe we onze klanten op een andere manier kunnen bedienen, waar de mogelijkheden liggen, hoe we de klanten het beste kunnen helpen of ondersteunen bij hun uitdagingen.

De ochtend

Als ik op kantoor aankom, is een van de eerste dingen die ik doe: mijn mail opstarten om te kijken of er nog aanvragen zijn binnengekomen, of er berichten zijn van mijn collega-managers en natuurlijk van de medewerkers. Daarna is het internet afstruinen aan de beurt: waar worden nog testers gevraagd? Ik moet toegeven dat het vaak voelt als het zoeken van een speld in een hooiberg. We zijn immers niet de enigen die nog kandidaten beschikbaar hebben. Soms moet je gewoon geluk hebben doordat je op het juiste moment op de juiste plaats bent.

Vanmorgen ben ik voornamelijk bezig geweest met het voorbereiden van een klantenavond. Op deze avond gaan wij

onze oplossing voor testautomatisering presenteren aan onze klanten. Eerst heb ik mij bemoeid met de uitnodigingslijst en vervolgens ben ik begonnen met het maken van de presentatie voor die avond. Daarna ben ik in overleg gegaan met twee van onze medewerkers die deze dienst hebben gemaakt. We hebben gekeken wat we nog moeten doen om dan alles onder controle te hebben. Zaken zoals: werkt de applicatie, zijn de scripts goed gedefinieerd, werken de voorbeelden? Verder moeten we voorbereid zijn op vragen van het publiek, dus ook daar hebben we veel aandacht aan besteed. Als laatste ben ik met een van onze medewerkers om de tafel gaan zitten om te kijken wat voor eten we willen serveren tijdens het inloopbuffet. Iedereen heeft er immers een hele dag werken op zitten en dan is een goed buffet een aangename verrassing. Gelukkig kan ik dat aan Valentine overlaten.

Tussen de middag genieten we altijd van een gezamenlijke lunch met iedereen die op dat moment op kantoor is. Gebruikelijk komen dan de verhalen van de avond of het weekend ervoor aan de orde.

De middag

De middag is gespendeerd aan de beoordeling van het eindresultaat van TestBee, het nakijken van de factsheet dat binnengekomen is en het doornemen van de deelnemerslijst voor de presentatie.

Gezamenlijk met de Back office hebben we een lijst van klanten samengesteld die we van een mailing gaan voorzien rondom dit product.

Tevreden ben ik om half zes huiswaarts gekeerd om de volgende dag gewoon weer de draad op te pakken om te kijken of er nog testers in de markt gevraagd worden.



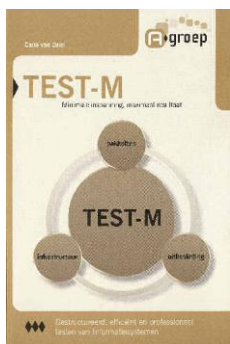
Boekrecensies

Test-M

MINIMALE INSPANNING, MAXIMAAL RESULTAAT

Door Hans van Loenhoud

tnn@testnet.org



Uitgeverij: Awareness Groep, Beneden Leeuwen
ISBN-nummer: 978-90-813110-1-4
Auteur(s): Carlo van Driel

Test-M is een kookboek. Een kookboek van het type 'Men neme een ei en schenke een halve liter koud water in de pan.' De kok is in dit geval de test(manag)er en er staan maar drie recepten in: het testen van pakketten, van infrastructuur en de organisatie van testen bij uitbesteding.

Persoonlijk heb ik wel iets met kookboeken. Ze helpen me om op een handige manier iets lekkers op tafel te zetten, wat me als kok van de koude grond op eigen houtje niet zou lukken. Met een kookboek weet ik welke ingrediënten ik moet hebben in welke hoeveelheden, welke handelingen ik moet

uitvoeren en hoeveel tijd ik ervoor nodig heb, uitgaande van het aantal hongerige magen dat ik moet voeden.

Zodoende is de maaltijd op tijd klaar, sta ik niet onnodig lang in de keuken, heeft iedereen genoeg en hoef ik geen klikjes weg te gooien.

Die analogie gaat aardig op voor Test-M. Met de ondertitel 'Minimale inspanning, maximaal resultaat' pretendeert deze testmethodiek ons te leren om effectiever te testen tegen aanzienlijk lagere kosten en kortere doorlooptijd. En daarmee een van de grootste klachten te voorkómen die buitenstaanders hebben over de uitoefening van ons testvak: 'Het kost te veel'.

Als methodiek ziet Test-M zichzelf als een nieuwe generatie, niet als vervanging maar als aanvulling op bestaande methodieken voor gestructureerd testen. Het nieuwe bestaat uit een grote mate van pragmatisme ('ga niet uit van een optimale maar van een waarschijnlijke uitgangssituatie', 'verspil geen tijd aan het overschrijven van specificaties naar testbaar formaat') gecombineerd met een strakke fasering en de spilfunctie van een professionele testmanager.

De fasering van opstart, voorbereiding, uitvoering en afronding wordt heel concreet en gedetailleerd ingevuld voor het testen van pakketimplementaties, van de consequenties van infrastructuurwijzigingen op de erop draaiende applicaties en voor het testen in geval van uitbesteding van ontwerp en bouw. Die invulling is voor een flink deel uiteraard tamelijk generiek en op bepaalde essentiële punten specifiek voor de betreffende testsituatie. De fases zijn rijkelijk

gelardeerd met tips, tabellen, sjablonen en al wat een doorsneekok nog meer kan gebruiken bij de bereiding van zijn testgerecht.

Al met al een aardig kookboekje, niet voor de beginnende keukenhulp en niet voor de ervaren topkok, maar een beetje daartussenin, bijvoorbeeld voor de testcoördinator die voor het eerst met het testen van een pakketimplementatie te maken krijgt, of de testanalist die even vlug wil nakijken hoe je een risico-analyse-tabel kan opzetten.

Een recensie is uiteraard niet compleet zonder kritiek. Als TNN-redacteur valt me op dat er bij dit Test-M boek wel erg bezuinigd is op de eindredactie. Met als gevolg een aantal storende spel-, taal- en copy/paste-fouten her en der in de tekst. Ik mag hopen dat er een volgende druk komt, want dat biedt de mogelijkheid om die foutjes eruit te halen.



Van Implementatie tot Acceptatie

Door Meile Posthuma
tnn@testnet.org



Uitgeverij: Maak MijnBoek.nl
ISBN-nummer:
Auteur(s): Maurice Siteur

Maurice Siteur zit al 25 jaar in de IT. Deze mijlpaal was voor hem reden om zijn ervaringen buiten het testdomein vast te leggen in dit boek dat in eigen beheer is uitgegeven. Maurice schetst de relaties tussen implementatie-, release- en testmanagement. In drie hoofdstukken behandelt hij zijn ervaringen in de drie verschillende onderdelen om af te sluiten met de overdracht naar de bestaande organisatie. In elk van deze hoofdstukken geeft Maurice in het kort aan waar je tegenaan loopt en wat de aandachtspunten zijn. Tweederde van het boek bestaat uit bijlagen. Hierin staan zo'n beetje alle templates die je tegen zult komen wanneer je de rollen van release-, implementatie- en testmanager vervult. De templates zijn echter wel allemaal ingevuld voor het DWS-systeem (DeurWaarderSysteem) wat van toegevoegde waarde is. Zoals al aangegeven, is dit boek in eigen beheer uitgegeven. Dit is te merken aan het feit dat er geen eindredactie door een uitgeverij is gevoerd. Het

maakt het boek rommelig en minder prettig te lezen. Er zitten regelmatig typfouten in en ook lopen de zinnen niet altijd zoals het zou moeten.

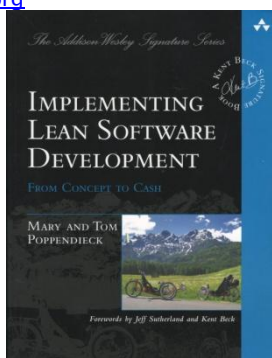


Implementing Lean Software Development

FROM CONCEPT TO CASH

Door Meile Posthuma

tnn@testnet.org



Uitgeverij: Addison Wesley
ISBN-nummer: 978-0-321-43738-9
Auteur(s): Mary and Tom Poppendieck

Het principe van Lean bestaat al een hele tijd en wordt vaak toegepast in combinatie met Six Sigma. De grondlegger van het huidige Lean-principe is Toyota en daarom is het ook bekend geworden als het Toyota Production System. Lean is erop gericht om verspilling, zaken die geen toegevoegde waarde leveren aan de klant, te elimineren. Met als gevolg dat de productiekwaliteit omhoog en de productiekosten omlaag gaan. Dit leidt tot een beter bedrijfsresultaat.

Mary en Tom Poppendieck hebben de lean principes toegepast op softwareontwikkeling. Deze principes zien we vaak al toegepast bij agile ontwikkelmethoden. Toch kunnen vele van de beschreven technieken ook in

de oudere ontwikkelmethoden worden toegepast.

Denk bijvoorbeeld aan Test Driven Development, waarbij eerst component testgevallen worden ontwikkeld voordat er een letter code geschreven is. De code die dan geschreven wordt moet altijd door de testgevallen heen komen. Pas dan wordt deze code vrijgegeven. Maar denk ook aan refactoring (herstructureren) van code en testgevallen. Hiermee wordt code zoveel mogelijk vereenvoudigd zonder de werking te wijzigen met als resultaat dat code beter te lezen is en makkelijker te onderhouden (minimale complexiteit).

Kijk allemaal eens naar het eigen testproces, hoeveel verspilling zit er wel niet in? Denk hierbij aan: wachten op opleveringen, testomgevingen / overproductie van allerlei documenten. Zijn alle processtappen die we volgen wel noodzakelijk? En als ze noodzakelijk zijn, beschrijven we dan niet te veel in detail? Moeten we wel alle testgevallen van te voren specificeren?

Dit boek valt onder de categorie 'Must Read'. Bijna elke pagina voegt waarde toe aan de kennis van de lezer op een zeer leesbare en onderhoudende manier. Pakkend van begin tot eind.



En nu in het echt ...

Door Hans van Loenhoud

loenhoud@inqa.nl

Het 'probleem'

Testen doen we in een testomgeving: een veilige en afgeschermd omgeving, waarmee we kunnen experimenteren en waarin het niet erg is als er wat fout

gaat. Rond dat hele principe hebben we een heel circus opgetuigd van OTAP-straten, overdrachtsprocedures, test-accounts en wat dies meer zij. Dat wordt gelukkig steeds meer gemeengoed in professioneel geleide testtrajecten – en terecht! Want je moet er toch niet aan denken dat testoutput per abuis bij de klant terecht komt. Dan sta je meteen op de voorpagina van NU.NL.

En toch ... je ontkomt er niet aan om als tester zo af en toe iets in een productieomgeving te doen. Hoe pak je dat dan aan? Drie manieren om als tester veilig in een productieomgeving te werken.

1. Niet muteren

Een simpele maar vaak over het hoofd geziene manier is, om geen mutaties aan te brengen in het te testen systeem. Door in het productiesysteem alle raadpleegfuncties langs te lopen, kun je al heel wat over de werking van het systeem te weten komen zonder dat je schade aanricht. Bedenk wel dat veel systemen onder water toch allerlei mutaties aanbrengen tijdens het gebruik van raadpleegfuncties, zoals het loggen van toegangsgegevens en dergelijke. Je zult je er van tevoren van moeten verzekeren dat een raadpleegactie geen kwaad kan. Bijvoorbeeld door formeel toestemming te vragen aan de systeemeigenaar of functioneel beheerder. Zorg bij voorkeur voor een apart testaccount, zodat de eventueel vastgelegde acties te traceren zijn naar een legale testtoegang.

Een variant van deze aanpak is het testen van mutaties, met op het laatste moment de keuze van 'cancel' of iets

dergelijks om de acties ongedaan te maken. Doe dat alleen als je zeker weet dat je daar geen problemen mee veroorzaakt.

Een andere variant is het wel volledig testen van mutaties en dan na afloop van de test de back-up laten terugzetten, waardoor de situatie van vóór de test wordt hersteld. Dat kun je uiteraard alleen doen wanneer je zeker weet dat je als enige op het systeem aan het werk bent, want anders worden mutaties van anderen wellicht ook ongewild ongedaan gemaakt. Deze variant brengt daardoor heel wat coördinatie met zich mee.

2. Meekijken met productie

Een andere simpele manier is om 'echte' productie als een test te beschouwen. Kies een of meer ervaren gebruikers uit en laat hen een aantal representatieve, nog niet behandelde gevallen uit de werkelijkheid verzamelen. Beschouw deze vervolgens als testgevallen en beschrijf, voordat ze in productie worden opgevoerd, de uitgangssituatie, het verwachte procesverloop en de output voorspelling, net zoals bij een gewoon testgeval. De gebruikers voeren deze gevallen vervolgens gewoon in productie in en jij zit er als tester naast om te kijken wat er precies gebeurt. Wat je dan ziet gebeuren kan voor jou dezelfde aanleiding geven voor het vastleggen van bevindingen als zou zijn geschied bij een 'normaal' testgeval. De verdere afhandeling van deze bevindingen volgt de reguliere weg. Lastig aan deze aanpak is het zorgen voor een goede dekkingsgraad. Om dat te bereiken zul je de testdoelen eerst goed in beeld moeten hebben en vervolgens de door de gebruikers

aangedragen testgevallen moeten mappen op deze doelen. Na deze exercitie zul je vaak de gebruikers moeten vragen om nog een paar specifieke extra gevallen op te zoeken om het dekkingsplaatje compleet te maken.

Voordeel van deze aanpak is dat het echte productie door echte gebruikers is met echte gevallen, die sowieso moeten worden uitgevoerd. Beveiliging en autorisatie is dan ook gewoon afgedekt door de reguliere processen.

3. Speciale testobjecten

Een laatste manier die ik hier wil bespreken is het invoegen van speciale testobjecten in productie, bijvoorbeeld dummy klanten of producten met fancy namen als 'Testklant_001' of 'Testproduct_niet_gebruiken'. Dat is nogal tricky, omdat je het risico loopt dat per ongeluk of met opzet dergelijke objecten door andere, reguliere gebruikers worden gebruikt, met allerlei onvoorspelbare consequenties, tot fraude aan toe.

Je zult daar dus nogal wat autorisatie, beveiliging en tracering omheen moeten organiseren. En structureel moeten rapporteren over het gebruik ervan.

Wees er bovendien attent op dat dergelijke dummy objecten ook in allerlei statistische en managementinformatie terecht zouden kunnen komen. Het is jammer als de aan de directie gerapporteerde extra omzet van de afgelopen maand uitsluitend te wijten is aan een overijverige tester, zeker als er ook nog BTW over wordt afgedragen

...

Doen?

Als het even kan, moeten we als tester uit de productieomgeving wegblijven. Want we kunnen geen verantwoordelijkheid nemen voor wat daar fout kan gaan. Bijna altijd kunnen we ons werk goed doen in een afgeschermd testomgeving, waarin onze activiteiten geen gevolgen hebben voor de bedrijfswerkelijkheid.

Maar soms kunnen we er niet omheen toch iets in de productieomgeving te doen. Bijvoorbeeld wanneer een nieuwe productieomgeving voor het eerst wordt opgeleverd.

Algemene aanbevelingen daarbij zijn:

- Doe het veilig
- Doe het traceerbaar
- Doe niets onomkeerbaars
- Zorg voor formele toestemming
- Rapporteer over daadwerkelijk gebruik

Stoot toegangsmogelijkheden / autorisaties weer af zodra je ze niet meer nodig hebt

En realiseer je daarbij: je werkt in productie met productiedata, dus met echte gegevens van echte mensen, echte producten, echte bedragen. In jouw rol hoor je daar eigenlijk niet mee in aanraking te komen. Zorg daarom dat je geen printjes laat slingeren bij de printen en weersta de verleiding om snel even te kijken wat de recente ziektekostendeclaratie van je buurvrouw was.



Petje af voor TestNet

Quiz

Door Rik Marselis, jouw quizmaster

rik@marselis.eu

Toelichting Tussenronde

Tijdens het voorjaarsevenement op 22 juni presenteerde ik met veel genoegen de petje-af-quiz. De quiz werd gewonnen door Marc van 't Veer die 12 tweekeuze vragen op rij goed had en zich nu "TestNet-er van het jaar 2009" mag noemen. Zijn hoofdprijs was een gratis toegang tot de EuroSTAR conferentie in Stockholm van 30 november t/m 3 december dit jaar.



Winnaar Mark van 't Veer (l) krijgt van quizmaster Rik Marselis de oorkonde "TestNet-er van het jaar 2009" overhandigd.

De tussenronde, voor iedereen in de zaal, bestond uit een vraag over testtechnieken. Iedereen kon zijn antwoord op een formulier invullen. Na het voortreffelijke diner trok keynote speaker Isabel Evens de winnaar uit de (slechts 9!) goede antwoorden. Zij overhandigde een fles champagne aan Dick Bronk.



Winnaar tussenronde Dick Bronk (l) krijgt van Isabel Evens een fles champagne

Ik heb beloofd om de uitwerking in de TestNet Nieuws te plaatsen, belofte maakt schuld dus zie hier.

De opgave:

Op een tweebaansweg zijn wegwerkzaamheden waardoor de ene rijbaan gestremd is. Het verkeer moet dus over de overgebleven rijbaan. Van beide kanten komt een auto, we noemen ze auto 1 en auto 2. Aan beide kanten staat een verkeerslicht, verkeerslicht 1 en verkeerslicht 2. We gaan functioneel testen of het verkeer goed geregeld wordt, slechts één van beide auto's mag rijden.

Uit de unit tests is gebleken dat elk verkeerslicht altijd één kleur licht tegelijk geeft en dat alle lampen werken. Een verkeerslicht heeft 3 kleuren licht, alleen ROOD mag op beide verkeerslichten tegelijk verschijnen. Bij Groen of Geel licht mag een auto rijden.

Vragen:

1. Hoeveel condities zijn minimaal nodig om te kunnen testen of het systeem functioneel goed werkt?

(opmerking: alleen J/N (of 0/1 of T/F) condities)

- Hoeveel kolommen in de beslissingstabel krijgen de actie "auto 1 mag rijden"?

Antwoorden:

De juiste antwoorden waren 4 en 2. Dit hadden slechts 9 deelnemers goed (waarbij ik enkele namen zag van mensen die bij mij training testtechnieken hebben gevolgd dus dat deed me deugd).

Bij de foute antwoorden heb ik de meest wilde schattingen en gokjes gezien. Heel vaak zag ik 5 en 2. Helaas. (ook hier trouwens mensen die bij mij training hebben gehad, oeps)

Waarom zijn de juiste antwoorden 4 en 2? Allereerst hoeveel kolommen krijgen de actie "auto 1 mag rijden"? Dat is nogal eenvoudig, in een goede beslissingstabel komt één keer de situatie voor dat het verkeerslicht 1 op groen staat (en verkeerslicht 2 op rood) en één situatie verkeerslicht 1 op geel (en 2 op rood). Dus in bij elkaar zijn dat 2 kolommen waar auto 1 mag rijden, verder niet.

Hoeveel condities zijn nodig? 2 condities per verkeerslicht. Namelijk de

conditie "verkeerslicht is rood" en de conditie "verkeerslicht is geel". Want als beide condities niet waar zijn dan is het verkeerslicht dus groen, aangezien de unit test al heeft aangetoond dat er altijd één licht brandt. Eventueel zou je ook geel/groen of rood/groen als combinatie kunnen hebben, dat maakt voor het aantal condities of het aantal kolommen met de juiste actie verder niet uit. Met 2 verkeerslichten in totaal dus 4 condities.

De ingevulde beslissingstabel met 16 kolommen ziet er dan als volgt uit:

Heb je nog aanvullingen, opmerkingen enz. dan lijkt testforum mij de aangewezen plek om hiermee verder te gaan!



Stoplicht 1 Rood	J	J	J	J	J	J	J	J	J	N	N	N	N	N	N	N	N
Stoplicht 1 Geel	J	J	J	J	N	N	N	N	J	J	J	J	N	N	N	N	N
Stoplicht 2 Rood	J	J	N	N	J	J	N	N	J	J	N	N	J	J	N	N	N
Stoplicht 2 Geel	J	N	J	N	J	N	J	N	J	N	J	N	J	N	J	N	N
Onmogelijk	x	x	x	x	x				x				x				
Niet toegestaan											x	x				x	x
Stoplicht 1 Rood							x	x	x								
Stoplicht 1 Geel											x						
Stoplicht 1 Groen																x	
Stoplicht 2 Rood							x				x					x	
Stoplicht 2 Geel								x									
Stoplicht 2 Groen									x								
Auto 1 mag rijden							-	-	-		x					x	
Auto 2 mag rijden							-	x	x		-					-	

Al onze thema-avonden in 2009 worden gehouden in het NBC:

Plaats:

Nieuwegein
Blokhoeve 1, 3438 LC
NBC

Informatie:

Aanmelden kan tot uiterlijk drie werkdagen van te voren via onze website www.testnet.org onder "Evenementen" > 'Aanmelden evenement'.

Najaarsevenement TestNet

dinsdag
22 september
13:00 – 22:00 uur

Thema-avond TestNet

dinsdag
27 oktober
18:00 – 22:00 uur

Thema-avond TestNet

dinsdag
24 november
18:00 – 22:00 uur

Thema-avond TestNet

Testen van pakketten

maandag
21 december
18:00 – 22:00 uur

Evenementen & Thema-avonden

Cees Dulfer
Erik Hendrixx
Jan-Kees Glijnis
Ine Lutterman-Baars
Rik Marselis
Michiel Vroon

E-mail: evenementen@testnet.org