

# De essenties van testrapportage

Hoewel sinds de 80-er jaren van de vorige eeuw veel is veranderd in softwareontwikkeling, is de essentie van testrapportage gelijk gebleven: adviseren over kwaliteit en risico's. Dat betekent echter niet dat het opstellen van een goede testrapportage voor iedereen gesneden koek is. Het komt regelmatig voor dat collega's op zoek zijn naar een goed voorbeeld van of sjabloon voor een testrapport in agile, waterval, devops, outsourced, standaard-pakket of nog een andere context. Ervaring leert dat achter zo'n vraag vaak een andere vraag schuilt: wat en hoe moet ik in mijn situatie rapporteren over testen?

In deze whitepaper ga ik in op die vraag, o.a. gebaseerd op drie citaten van Michael Bolton, welke naar mijn mening de essenties van rapporteren over testen heel goed omschrijven. In andere publicaties ga ik in op het toepassen van de basis van rapporteren in hedendaagse context zoals die van Agile en DevOps teams.

## Product story

### Citaat van Michael Bolton



**Level 1:** Tell the product story. The product story is a qualitative report on how the product can work, how it fails, and how it might fail in ways that matter to our clients. “Working”, “failure”, and “what matters” are all qualitative evaluations. Quality is value to some person; in a business setting, quality is value to some person who matters to the business. A qualitative report about a product requires us to relate the nature of the product, the people who matter, and the presence or absence of value, risks, and problems for those people. Qualitative information makes it possible for our clients to make informed decisions about quality.

De kwaliteit van het product dat we testen, moeten we beoordelen op de waarde die het product heeft voor onze klanten. De klant is hier de partij voor wie we testen. De informatie die we met testen vergaren, moeten we dus vertalen naar de mate waarin het product doet wat de klant nodig heeft. We moeten inzicht geven in wat er goed gaat en ook in welke problemen er nog zijn. Met die informatie kan onze klant gewogen beslissingen nemen die gerelateerd zijn aan de kwaliteit van het product, zoals het al of niet in gebruik nemen ervan.

We moeten dus een goed inzicht hebben in wat van waarde is voor de klant. Anders kunnen we de vertaalslag niet maken en weten we niet hoe we met testen relevante informatie kunnen verzamelen. Door middel van testen leren we het gedrag van het product kennen. Onze productobservaties lichten we in de product story van de rapportage zo goed mogelijk toe. De **product story** bevat zowel de plussen als de minnen die we hebben geconstateerd.

Als we de onderwerpen uit de volgende twee blogs daaraan toevoegen, ontstaat een compleet testrapport. Sommige organisaties beschouwen zo'n rapport als een vrijgaveadvies. Ik vind het belangrijk dat de verantwoordelijkheden blijven waar ze horen: de klant neemt de beslissing en wij staan garant voor de kwaliteit van de informatie in het rapport.

Naast inzicht in problemen, wil de klant ook inzicht in de testdekking, plat gezegd: zijn er nog zaken die onvoldoende getest zijn?

## Testing story

### Citaat van Michael Bolton



**Level 2:** To make the product story credible, tell the testing story. The testing story is about how we configured, operated, observed, and evaluated the product; what we actually did and what we actually saw. The testing story gives warrant to the product story; it helps our clients understand why they should believe and trust the product story we're giving. The testing story is centred around the coverage that we obtained and the oracles that we applied. Coverage is the extent to which we've tested the program; it's about where we've looked and how we've looked, and it's also about what's uncovered—where we might not have looked yet, and where we don't intend to look. Oracles are central to evaluation; they're the principles and mechanisms that allow us to recognize a problem. The product story will likely feature problems in the product; the testing story, where necessary, includes an account of how we knew they were problems, for whom they would be

problems, and inferences about how serious the problems it might be. We can make inferences about the significance of problems, but not ultimate conclusions, since the decision of what matters and what constitutes a problem lies with the product owner. The product story and our clients' reactions to it will influence the ongoing testing story, and vice versa.

Waarop en hoe hebben we het product geëvalueerd en op basis waarvan konden we vaststellen of het product goed is?

In de **testing story** geven we antwoord op deze vragen door te vertellen wat we hebben getest, hoe we hebben getest, waar we diep hebben getest, waar we globaal hebben getest, waar we niet hebben getest, waar we wilden testen maar niet hebben gedaan en waar we sowieso niet van plan waren om te testen. Ook geven we aan welke bronnen we geraadpleegd hebben om te bepalen of wat we zagen goed was of niet. Bolton noemt dat 'oracles', wij kennen hiervoor ook het begrip testbasis. Welke bronnen hebben we hiervoor gebruikt? Dit kunnen documenten zijn, maar in de praktijk spelen ook andere bronnen een grote rol, zoals materiedeskundigen en voorbeelden van gelijksoortige producten.

De antwoorden op al deze vragen geven autoriteit aan de product story. De klant kan daarmee beoordelen of het juiste getest is en of er voldoende getest is. Het kan dus gebeuren dat de klant op basis van de 'testing story' besluit dat verder testen nodig is.

Wanneer we met testen tegen beperkingen aanlopen bij het vervullen van de opdracht van de klant om een gewenste testdekking te bereiken, geven we dat ook aan.

## The story about the quality of the testing

### Citaat van Michael Bolton



**Level 3:** To make the testing story credible, tell a story about the quality of the testing. Just as the product story needs warrant, so too does the testing story. To tell a story about the quality of testing requires us to describe why the testing we've done has been good enough, and why the testing we haven't done hasn't been so important so far. The quality-of-testing story includes details on what made testing harder or slower, what made the product more or less testable, what the risks and costs of testing are, and what we might need or recommend in order to provide better, more accurate, more timely information. The quality-of-testing story will shape and be shaped by the other two stories.

Gaat de testing story over de dekking die is bereikt, **the story about the quality of the testing** beschrijft de problemen die zijn ervaren met het bereiken van die dekking of problemen die het bereiken van de gewenste dekking nog in de weg staan. Dit gaat vaak over de testbaarheid van het product, de testomgeving en testdata: hoe goed kunnen we testen en tegen welke kosten? Daarnaast kan het zijn dat we tijd, tools, spullen, testers of expertise missen. We bespreken dat met de klant als de belanghebbende. De klant kan besluiten om een deel van de problemen te laten oplossen. Zo niet, dan accepteert de klant het risico of de kosten die een gevolg zijn van de beperkingen in de te bereiken testdekking.

De story about the quality of the testing vind ik een lastig deel van de testrapportage. Dat komt omdat het vaak gaat over zaken die ik niet zelf in de hand heb. Wat ik wel zelf kan doen is knelpunten altijd meteen onder de aandacht bij de testklant te brengen: om hulp te krijgen of om de verwachtingen voor het maximaal te bereiken testresultaat alvast bij te stellen.

## Projectmanagersgrip

### De behoefte van de projectmanager aan kwantitatieve informatie

Hoeveel testgevallen zijn er al uitgevoerd?

Hoeveel testgevallen zijn er ok en nok?

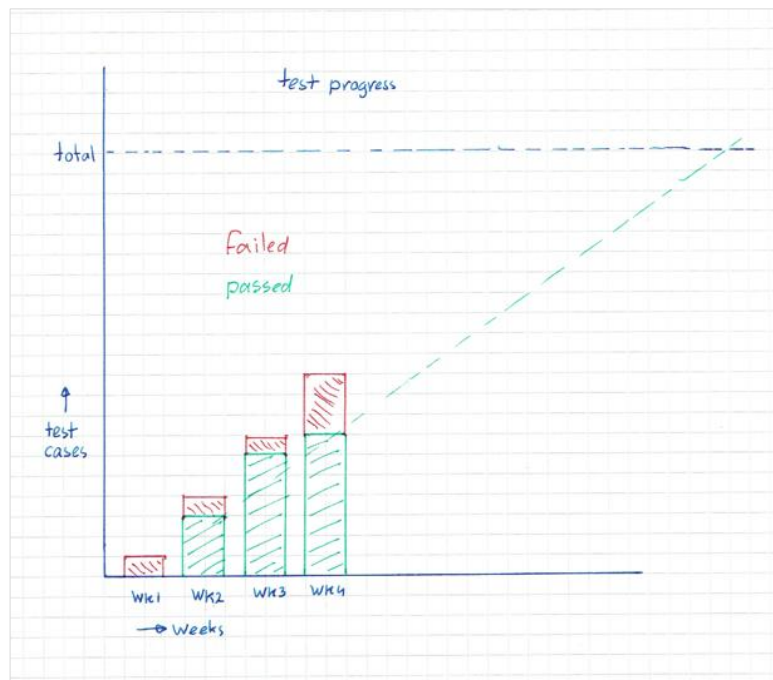
Hoeveel testgevallen zijn er nodig?

Wanneer zijn we klaar?

Zomaar wat vragen die een projectmanager op je af kan vuren. Ik begrijp de behoefte aan antwoorden op die vragen wel. De projectmanager (PM) wil graag een voorspelling van het projecteinde en wil zien hoe hard er aan bepaalde touwtjes getrokken moet worden om de gewenste opleverdatum te halen. Ook als er in sprints wordt gewerkt, heeft de klant vaak een wensdatum voor de oplevering van een *minimum viable product*. De eerste drie blogs over de essenties van testrapportage hebben een kwalitatieve focus. Hoe kunnen we van daaruit de brug slaan naar de traditionele behoefte van de PM aan meer kwantitatieve gegevens die meetbaar zijn en bijdragen aan een hogere voorspelbaarheid van *wanneer het klaar is*?

### Welke kwantitatieve informatie hebben we voor de PM?

Dat brengt me terug naar mijn allereerste testcoördinatieklus. Met drie collega's moest ik *performance monitoring* testen. We hadden goede specificaties (context: telecom, waarbij je aan gedetailleerd uitgewerkte standaarden moet voldoen) en we konden daar een representatieve set met testgevallen uit afleiden. Zodra we de beschikking hadden over de versie van het systeem die performance monitoring ondersteunde, begonnen we met testen. Ik hield per week bij wat de resultaten waren. Dat deed ik op papier, een ruitjesvel op een tafel in het testlab geplakt. Dat moet er ongeveer uitgezien hebben als hieronder. De grafiek geeft aan welke van de beschreven testgevallen inmiddels succesvol waren (groen) en met welke testgevallen we op een probleem stuitten (rood) waarvoor nog een oplossing moest komen.



De grafiek hierboven geeft een projectmanager bepaalde informatie:

- De gefaalde tests laten zien dat er nog problemen zijn: de PM kan meer prioriteit geven aan het oplossen van bevindingen om het aantal resterende problemen te verlagen.
- Als je een lijn trekt door de groene staafjes krijg je een eerste orde voorspelling van het moment dat alles oké is.

Door dat punt week bij week in de gaten te houden kan de PM beoordelen of aanvullende maatregelen nodig zijn. Een goede PM snapt uiteraard dat gegevens uit het verleden geen garanties bieden voor de toekomst. De kwaliteit van het product in gebieden die we nog niet hebben bekeken, heeft bijvoorbeeld een grote invloed op de toekomstige

testvoortgang. Aan PM's die deze bijsluiters niet goed (willen) begrijpen, heb je als testcoördinator soms je handen vol... En er is meer slecht nieuws, want testgevallen zijn helaas vaak onbetrouwbare makers.

## **Waarom tellen van testgevallen vaak geen houvast biedt**

**De specificaties zijn onvoldoende representatief.** In mijn historische voorbeeld waren de specs en de afgeleide testgevallen representatief. Die luxe hebben we niet zo vaak in projecten. De PM kan er dus niet vanuit gaan dat het project klaar is als alle tests op basis van de beperkte specificaties *passed* zijn.

**Verschillen in de omvang van tests heeft invloed op het extrapoleren van de testvoortgang naar een project-einde.** Wat is een test? Eén controle die tien minuten duurt of het doorlopen van een scenario met misschien wel tien checks waar je een halve dag mee zoet bent? Als een set van tests grote verschillen vertoont tussen de tijd die het kost om tests uit te voeren, dan is de onzekerheidsmarge om de verspelde einddatum groot en biedt het de PM minder houvast.

**Het telbaar maken van tests is geen prioriteit voor Agile teams.** Een agile team richt zich op het opleveren van werkende software, waarbij wel veel wordt getest, maar niet per se met achterlating van gedocumenteerde tests. Testen laat zich niet als geïsoleerde activiteit monitoren. Daarnaast is het team niet bezig met het bepalen van concrete tests voor alle volgende sprints. Het voorspellen van het aantal tests dat nodig is tot het *minimum viable product* is opgeleverd, is daarmee een hachelijke zaak.

**Exploratory testen.** We laten ons bij het testen leiden door allerlei informatie, die deels pas beschikbaar komt tijdens het testen zelf. We laten ons dus niet alleen leiden of beperken door vooraf opgestelde testgevallen. Omdat deze manier van testen, gelukkig, in de meeste projecten plaatsvindt, beschikt de PM ook daarom niet over aantallen tests als basis voor het voorspellen van het projecteinde.

## **Acceptatiecriteria bieden houvast**

Projecten die geen acceptatiecriteria hebben, missen een belangrijk kijkvenster op het project. 'Ergens' zou men een lijst moeten hebben met criteria op basis waarvan de klant of businessvertegenwoordiger bepaalt of er een go gegeven kan worden. De scenario's die zijn opgesteld in een *business driven development* proces (BDD), bijvoorbeeld, zijn de acceptatiecriteria. User stories kunnen ook als kapstok dienen voor acceptatiecriteria, of de business processen die ondersteund moeten worden; liefst alle drie met elkaar gekoppeld. Vervulde acceptatiecriteria zijn *eenheden van vertrouwen in het product* en de voortgang daarin geeft een goed beeld van wat er af is en wat er nog niet af is.

Als we nu *test cases* in mijn voorbeeldplaatje vervangen door *BDD-scenario's* of *acceptatiecriteria* dan kunnen we voor de PM best een interessant grafiekje maken. Groen staat voor acceptatiecriterium gehaald en bij rood is er nog een probleem op te lossen. We geven daarmee niet per se antwoorden op de vragen aan het begin van dit artikel, maar we dragen wel bij aan de beantwoording van de vraag achter de vragen van de PM: zijn aanvullende maatregelen nodig om project- en productrisico's te beheersen?

## **Kwaliteit is de essentie**

Dat de blogs over de *essenties van testrapportage* geen aandacht schenken aan kwantitatieve gegevens uit het testproces is geen manco. Projectmanagers begrijpen namelijk steeds beter dat de kwaliteit van een softwareproduct niet is uit te drukken in getallen. Niet voor niets gaat het in de *product story* helemaal niet over testen maar over kwaliteit. De *testing story* en de *story about the quality of the testing* onderbouwen de *product story*. Kwaliteit is de essentie!

**Kees Blokland, testarchitect**