BOEKTITEL

(Hoofdtitel in hoofdletters)

Zorg ervoor dat het een goed te onthouden titel is en trek de aandacht van lezers. Uit de titel moet blijken waar het boek over gaat en het moet eenvoudig uit te spreken zijn. Vermijd titels die moeilijk klinken of uit te spreken of vreemd klinken wanneer ze hardop worden gezegd.

Als u al een specifieke titel in gedachten hebt, is dat geweldig. Maar ook als u dat niet hebt moet u kijken naar de doelen van uw boek. Als u bijvoorbeeld uw merk wilt opbouwen, noem dan uw merk in de titel. Als u zichzelf wilt vestigen als expert, zorg er dan voor dat uw titel gezaghebbend klinkt. Als u gewoon aandacht wilt trekken en exemplaren wilt verkopen, zorg er dan voor dat de titel provocatief, spannend of of controversieel is.

De kwaliteit van de titel bepaalt de kwaliteit van de inhoud. Stel uzelf daarom specifieke vragen, zoals: Welke waarde wil ik toevoegen aan het leven van de lezer? Wat wilt u dat ze leren? Wat zijn de doelstellingen van de doelgroep en hoe kunnen ze dat met dit boek bereiken? Probeer uzelf te verplaatsen in uw publiek.

Probeer bij het maken van een lijst met mogelijke titels te oefenen met relevante trefwoorden die populair zijn in Google- en Amazon-zoekopdrachten. Denk aan uw genre/doelgroep en maak vervolgens een titel die een belofte doet of een oplossing voor een probleem biedt. Bijvoorbeeld 'Geweldige kinderen opvoeden in harde wereld' of 'De zes geheimen om een miljoenendeal te sluiten.' Vermijd titels die al populair zijn, omdat hiermee uw boek er niet zal uitspringen. Het eerste voorbeeld (Geweldige kinderen opvoeden in een harde wereld) gebruiken we om te laten zien hoe u ideeën creëert voor een boek.

Ondertitel van boek

(Voor Engelstalige boeken: gebruik een hoofdletter voor elke eerste letter van elk woord in de ondertitel)

Een goede ondertitel moet meer inzicht geven in het onderwerp, zodat lezers precies begrijpen waar het boek over gaat. Een titel als 'Geweldige kinderen opvoeden in een harde wereld’ kan bijvoorbeeld goed samengaan met een titel als ‘Maak uw kinderen geestelijk weerbaar en bereid ze voor op het leven'.

**Door**

(Naam auteur invoegen):

© Copyright (jaar van uitgave invoegen) door (naam auteur of uitgever invoegen) - Alle rechten voorbehouden.

Het is niet toegestaan om enig deel van dit document te reproduceren, te dupliceren of door te geven, noch in elektronische vorm, noch in gedrukte vorm. Het opnemen van deze publicatie is streng verboden.

Dit boek is opgedragen aan:

Hier inhoud invoegen...

Dit deel is optioneel en vaak van persoonlijke aard. Vermeld en bedank een persoon of een groep personen die dicht bij u staan of u hebben ondersteund in uw leven. Dat kunnen familieleden, goede vrienden, inspirerende personen, enzovoort zijn.

Inhoud

[Inhoud iv](#_Toc43282155)

[Inleiding 1](#_Toc43282156)

[DEEL I: De Anatomie van de validatieregels-repository 2](#_Toc43282157)

[Hoofdstuk één: Hoofdstuktitel invoegen (Stijl: Koptekst 2) 3](#_Toc43282158)

[Hoofdstuk twee: Hoofdstuktitel invoegen 5](#_Toc43282159)

[DEEL II: De anatomie van een validatieregel 6](#_Toc43282160)

[Hoofdstuk drie: Hoofdstuktitel invoegen 7](#_Toc43282161)

[Hoofdstuk vier: Hoofdstuktitel invoegen 8](#_Toc43282162)

[Hoofdstuk vijf: Hoofdstuktitel invoegen 9](#_Toc43282163)

[Epiloog/conclusie 10](#_Toc43282164)

[Bibliografie 11](#_Toc43282165)

[Ontvangstbevestigingen 12](#_Toc43282166)

[Over de auteur 13](#_Toc43282167)

# Inleiding

Dit artikel hoort bij een github-repository die xml\_schematron heet. Deze directory bevat validatie-regels ten behoeve van IMOW en IMOP data-sets.

# DEEL I: De Anatomie van de validatieregels-repository

Beschrijving van de volledige gityhub-repository plus enkele opmerkinen over te gebruiken software.

## De directory en bestanden van de validatie-regels repository.

(Voor Engelstalige boeken: gebruik een hoofdletter voor elke eerste letter van elke hoofdstuktitel)

De repository bestaat uit een aantal directories en bestanden die op de PC terecht komen als men een “git clone” doet. Sommige directories worden in andere delen van dit document uitgelegd. Dit zijn de directories die per validatie-regel bestaan, en de test-set directory.

### De versie-directories.

Als je “git clone” hebt gedaan krijg je 3 versie-directories en een paar zip-bestanden. De ZIP-bestanden bevatten dezelfde bestanden als de directories die hetzelfde heten, maar de volledige sets, terwijl de directories alleen de validatie-set bevatten die bij die versie hoort.

De huidige versie is in de directory met het hoogste versienummer. In die directory:  
documentatie: daarin bevindt zich dit bestand, en een Excel-sheet die de validatie-matrix bevat.  
src: de omgeving waarin de tpod-validatie directories zijn. Per validatie-regel een subdirectory.  
Validaties: hierin de bestanden die samen een volledige test-set vormen, die worden uitgeleverd via de geonuvum website.

### De src-directory

In de src-directory zitten enkele subdirectories die niet met Test… beginnen.

Dit zijn:  
- saxon9.9.1.5: hierin zitten de saxon-libraries + de hulp-bestanden die van schematron xslt maken. Dit setje bestanden wordt gebruikt vanuit de directory Tests. Ik kom daar later op terug.  
- schemas: hierin zitten de xsd-schema’s die worden gebruikt tijdens het ontwikkelen van de validatie-regels. In de test-bestanden zit een verwijzing naar deze schema’s.  
- Een hele rits bestanden die met Test\_..... beginnen. Per directory wordt een validatie-regel uitgewerkt. In DEEL II ga ik uitleggen hoe dit werkt.  
- Tests: In deze directory zitten wat hulpscriptjes om tests vanuit de bash-command-prompt uit te voeren. Deze directory gebruikt de saxon-directory die ik eerder noemde. Er zitten twee bestanden in: test\_nummer.sh, moet aangeroepen worden met een validatie-soort/nummer. Bijvoorbeeld vanuit de tests-directory: ./test\_nummer.sh TPOD0230 Ook is het mogelijk om alle tests uit te voeren met ./test\_all.sh

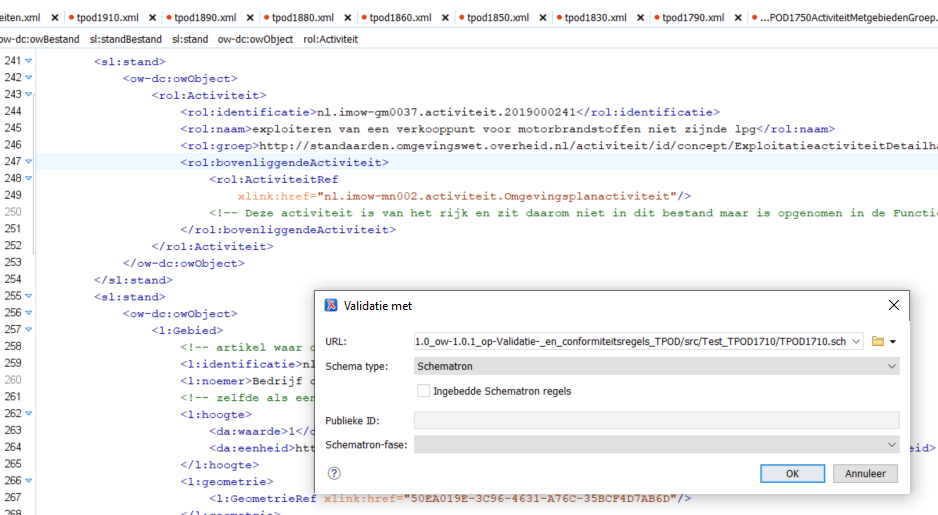
## Benodigdheden en handigheden bij de ontwikkelomgeving

Bepaalde software producten zijn beslist nodig en sommigen zijn wel erg handig bij het ontwikkelen van validatie-regels. Ook zijn enkele producten nodigof handig bij het uitvoeren van de validatie-regels. Sommige tools zijn discutabel, en sommige tools hebben alleen mensen die software ontwikkelen op hun PC. Gelukkig zijn er ook altijd omwegen, maar niet altijd even efficient.

### Oxygen

De Koningin onder de XML-editors. Oxygen (of <oXygen/> zoals ze het zelf noemen) verwerkt alle soorten XML en aanverwante subtalen en scriptalen. Oxygen kan moeiteloos omgaan met XML, XSLT, Schematron, en vele andere XML-gerelateerde toepassingen, daarnaast doet het ook JSON, Yaml, etc.

Dus in Oxygen kun je goed je validaties schrijven, het biedt vele hulpmiddelen, en je kan binnen deze omgeving een schematron uitvoeren met een xml-bestand als target.



In feite kun je in Oxygen alles doen wat noodzakelijk is, maar dan mis je wel mogelijkheden tot automatisering van de ontwikkelomgeving, en het opslaan in versies.

Dus je kan ermee aan de slag, maar het is niet genoeg om efficient te werken.

### Ant

Ant (Engels voor mier) is een veel gebruikte Java-ontwikkeltool. In DEEL III is te lezen hoe een validatie-set eruit ziet. Dat is enigszins complex, maar niet onnodig complex. Lees daar verder over het waarom. Ant helpt om met de validatie-set te kunnen werken. Het kan ingewikkelde opdrachten uitvoeren. Ant is beschikbaar voor Windows en voor alle Unix-smaken.

Om met Ant te kunnen werken is een Java JDK noodzakelijk. Ant is een tool voor ontwikkelaars, en ontwikkelaars hebben altijd een JDK geïnstalleerd. (sommigen zonder het te weten). JDK is de Java Development Kit. Deze bestaat uit de JRE (Java Runtime Environment) plus ontwikkeltools, compilers, etc. Een van de ontwikkeltoosl die Ant nodig heeft zit in tools.jar, en als er geen JDK is geïnstalleerd, dan gaat Ant daarover mopperen: “tools.jar not found”

De validatie-set kan ook zonder Ant worden gebruikt, maar dan moet je een aantasl dingen zelf doen die nu Ant voor je doet. Dit staat uitgelegd in DEEL V.

### Git

Git is noodzakelijk om de gehele ontwikkelset veilig op te slaan en terug te halen. Het onthoudt ook oudere versies en biedt vele features voor versie-beheer.

### Git-bash

Git-bash is een MINGW64 implementatie. Het is een Unix/Posix-client voor Windows. Werkt men vanuit Linux dan is dit niet nodig, Linux zelf is Posix compliant. Posix is een internationaal erkende standaard, ISO‎/IEC 9945, het betekent: Portable Operating-System Interface.

Vanuit Windows is het ook niet perse nodig. Maar het geeft wel gemak omdat Bash (DOS voor Unix) een goede scripttaal heeft waarvan ik gebruik maak.

Men kan echter ook zonder deze. Sommige handige scriptjes die ik heb geschreven moeten dan omgezet worden naar batch bestanden. Ik zou dit niet altijd kunnen, maar de echte Windows-fan, neem ik aan, zal er geen moeite mee hebben om dit te doen.

Zonder deze scriptjes (of de omzetting naar batch) heeft men soortgelijke beperkingen dan als men zonder Ant werkt.

# DEEL II: De anatomie van een validatieregel

Dit deel beschrijft hoe een ontwikkel-omgeving van een validatie-regel eruit ziet.

Het begint met een directory. Zoals elders ook beschreven is het nuttig om deze directory met “Test\_” te beginnen. Wat er achter “Test\_” komt kan worden gebruikt om het op te roepen vanuit het test\_nummer.sh in de Tests directory.

## De naamgeving

Een validatie-regel bevindt zich in een directory. Deze heet bijvoorbeeld Test\_TPOD1860. Dit is de validatie-regel die regel 1860 uitvoert Het schematron bestand heet dan TPOD1860.sch Dit heeft voordeel om dit in stand te houden.

Er bestaat naast de directories met de individuele test-ontwikkelsituaties ook een directory Tests. In DEEL IV is beschreven waarvoor deze nuttig is.

## De test-bestanden

### Beschrijving test-bestanden

De test-bestanden zijn databestanden zoals die voor kunnen komen in een dataset voor de omgevingswet. Het is geen complete dataset, maar het bevat alleen de delen die nodig zijn om de specifieke validatie-regel van die directory te testen.

Technisch gezien zijn de test-bestanden op orde, ook die delen die niet worden gebruikt zijn zoveel mogelijk technisch op orde en behorend bij de validatie-regel en IMOP/IMOW versies.

### Doel test-bestanden

De test-bestanden helpen bij het ontwikkelen en van een validatie-regel. Ze zijn zo ingericht dat een validatie-regel een fout er in vindt, en ook dingen goed keurt, zodat zowel goed als fout wordt getest.

De validatie-regel wordt gedraaid tegen een test-bestand. Soms worden er data getest die verspreid over meerdere bestanden worden opgeroepen. Dan zorgt de test-routine ervoor dat die bestanden worden gevonden. Meestal is dat aan de hand van xml-tags.

### Meerdere test-bestanden

Een validatie-regel-directory kan meerdere test-bestanden hebben. Vaak zit er een besluit-bestand bij. Soms heeft dit alleen de functie om het TPOD-type op te zoeken. Dit geldt voor validatie-regels die maar voor enkele TPOD’s geldig zijn. Soms dienen ze omdat het verband tussen twee bestanden wordt onderzocht.

### Hoe te testen

Tijdens het ontwikkelen is er eigenlijk een manier om te testen die het meest gangbaar is. Men pakt een test xml of gml bestand en opent dit in Oxygen, en via de menu-structuur gaat men naar Document -> Valideer -> Valideer met, en dan kiest men voor het schematron bestand in de directory.

Een andere manier is via het scriptje uit de Tests-directory. Men gaat via een Posix-prompt naar de directory Tests en daar roept men het scripje test\_nummer.sh aan. Bijvoorbeeld: ./test\_nummer.sh TPOD0230

## Het schematron-bestand

Het schematron bestand kan op verschillende wijzen worden opgebouwd, ik heb een stramien opgebouwd die steeds terug komt. Hopelijk wordt in onderstaand verhaal duidelijk waarom dat is, en wat de voordelen zijn.

### Hoe werkt schematron?

Richard (Rick) Alan Jelliffe bedacht schematron in 2001 om tests te kunnen uitvoeren op xml die totdan toe onmogelijk waren. Om dit te doen wordt de sch-namespace in een schematron bestand vertaald naar xslt, en die xslt wordt uitgevoerd om een xml bestand te testen. Voor dit doel worden xslt-bestanden gebruikt die de transformatie bewerkstelligen.

Er zijn dus twee zaken nodig om schematron te gebruiken, dat zijn de genoemde xslt’s en libraries om xslt uit te voeren. In de validatie-set zijn twee directories. Een directory heeft xsl, hierin zijn de door Jeliffe bedachte xslt bestanden, en de directory lib, hierin zijn de saxon-xml bestanden, in Home-Edition-versie (HE), die is gratis en volstaat voor dit doel.

Belangrijk op dit moment om te onthouden is dat de schematron namespace wordt vertaald naar xslt. Dit is een proces met beperkte mogelijkheden. Vandaar dat er xslt-code wordt toegevoegd aan de schematron validatie.

### De onderdelen van het schematron bestand.

Ik heb als voorbeeld Test\_TPOD1730 genomen omdat dit alle onderdelen op redelijk eenvoudige wijze gebruikt.

**De namespaces XSLT**

Bovenin het sch bestand zitten de namespaces. Dit zal iedere XML-expert bekend voorkomen. Eerst worden de namespaces benoemd die door eht xslt-gedeelte van het bestand worden gebruikt. Omdat dit in ieder sch-bestand terugkomt heb ik alle mogelijke namespaces neergezet. In ieder sch-bestand zitten dus namespaces die niet worden gebruikt.

**De namespaces SCH**

Dan worden deze namespaces herhaald in de sch-namespace en in een andere syntax. Deze worden door de Jeliffe-transformatie gebruikt. Die transformatie laat de reguliere namespaces links liggen. Om ook hierin maar een keer te hoeven na te denken zijn alle mogelijke namespaces opgenomen.

Het xslt-gedeelte gebruikt dus namespaces en het sch gedeelte construeert namespaces.

Hier inhoud invoegen...

Gebruik voor subkop 2 een ondertitel als 'Emotionele intelligentie ontwikkelen.' Hier worden praktische stappen behandelt over hoe u kunt kunt helpen met het herkennen van sociale signalen, negatieve emoties, enzovoort.

### Hoofdstuksamenvatting/belangrijkste punten

Hier inhoud invoegen...

Herinner de lezer aan de belangrijkste punten uit het hoofdstuk in een korte alinea. U kunt ook een opsomming gebruiken als hieronder wordt weergegeven:

* Het bijbrengen van sociale waarden en ethiek in een kind zal eraan bijdragen dat he kind goed integreert in de maatschappij.
* Punt 2 van uw tekst...
* Punt 3 van uw tekst...
* enzovoort.

In de volgende hoofdstukkoppen leert u...

Hoofdstukken op logische en goedlopende wijze te laten opvolgen, informeer de lezer over wat er gaat komen. In de laatste alinea van uw hoofdstuk koppelt u de informatie van het volgende hoofdstuk met hetgene dat al is behandeld.

## Hoofdstuk vijf: Hoofdstuktitel invoegen

Begin hier een nieuw hoofdstuk..

Ervan uitgaande dat dit het laatste hoofdstuk van het boek is, maakt u een ietwat toekomstgerichte titel, bijvoorbeeld 'Kinderen van vandaag, ouders van morgen.' Beschrijf hier hoe een overzicht van de strategieën uit het boek, evenals nieuwe wetenschappelijke inzichten, de toekomst zullen vormen. Bespreek ook hoe de wereld eruit zal zien als kinderen niet geleerd wordt hoe ze hun mentaliteit kunnen versterken.

### Subkop 1 invoegen

Hier inhoud invoegen...

Deel uw hoofdstuk op in delen met relevant subkoppen. Subkoppen helpen de lezer bij het lezen van het hoofdstuk en helpen om te laten zien hoe u het onderwerp ziet. Gebruik altijd meer dan één subkop per hoofdstuk, en zorg ervoor dat ze gerelateerd zijn aan het hoofdstukonderwerp.

Als u inhoud voor een specifiek hoofdstuk onderzoekt, kan elk hoogtepunt dat u tegenkomt fungeren als subkop. Subkop 1 voor dit hoofdstuk 1 is bijvoorbeeld 'Aanpassen aan verandering' waar u vervolgens het belang bespreekt van het ontwikkelen van aanpassingsvermogen en flexibiliteit, enzovoort.

### Subkop 2 invoegen

Hier inhoud invoegen...

Gebruik voor subkop 2 een ondertitel als 'Een nalatenschap achterlaten.' Sluit hier alles af door te bespreken hoe alle verstrekte strategieën kunnen zorgen voor een betere toekomst voor iedereen.

### Hoofdstuksamenvatting/belangrijkste punten

Hier inhoud invoegen...

Herinner de lezer aan de belangrijkste punten uit het hoofdstuk in een korte alinea. U kunt ook een opsomming gebruiken als hieronder wordt weergegeven:

* In de toekomst zal overleven afhangen van aanpassingsvermogen aan een snel veranderende omgeving.
* Punt 2 van uw tekst...
* Punt 3 van uw tekst...
* enzovoort.

# DEEL III: De anatomie van de test-set

# DEEL IV: Tips en instructies om validatie-regels te ontwikkelen.

# DEEL V: Tips en instructies voor het testen van datasets

# Epiloog/conclusie

Hier inhoud invoegen...

Houd het kort maar krachtig. Benoem de belangrijkste punten uit het boek en de acties die leiden tot de oplossing van de problemen. Herinner de lezer eraan hoe het nemen van de genoemde actiestappen hem of haar ten goede zal komen.

# Bibliografie

Hieronder vindt u een voorbeeld van een lijst van werken in APA-stijl. Rangschik uw lijst met verwijzingen op alfabetische volgorde.

Als u boeken citeert, gebruikt u de notatie die wordt gegeven in de voorbeelden, en pas bibliografiestijl toe op de indeling:

Achternaam auteur, initiaal of initialen. (Publicatienaam). *Boektitel.* Aanvullende informatie. Stad en land van publicatie: Uitgeverij.

King, S. (2000). *Over schrijven: Memoires van een ambacht.* Amsterdam, Nederland: De Bezige Bij.

Wanneer u online-informatiebronnen noemt, gebruik dan de notatie als weergegeven in de volgende voorbeelden:

Voor documenten van het internet

Achternaam auteur, initiaal of initialen. (Datum van publicatie). De titel van het artikel. *Titel van het work*. Opgevraagd van volledige URL

Amir, N. (17 oktober 2018). 4 tips om op schema te blijven bij het schrijven. *Schrijf nu non-fictie!* Opgevraagd van http://writenonfictionnow.com/tips-staying-track-writing/

Voor online tijdschriften

Achternaam auteur, initiaal of initialen. (Datum van publicatie). De titel van het artikel. *Titel van tijdschrift*, nummer en paginanummers. Opgevraagd van volledige URL

Van Eeuwijk, R.L. (4 oktober 2018). Betere titels maken: 7 tips voor boeken, artikelen en vergaderingen. *Writer’s Digest.* Opgevraagd van http://www.writersdigest.com/whats-new/how-to-write-better-titles

Al u tijdschriften noemt, gebruik dan de notatie als weergegeven in de volgende voorbeelden:

Voor tijdschriften

Achternaam auteur, initiaal of initialen. (Publicatienaam). Titel van artikel. *Titel van tijdschrift*, *nummer (indien beschikbaar*), inclusief pagina's.

De Jong, J. (29 april 2013). Conceptnr. 4 *New Yorker*, *89*, 20-25.

Raadpleeg de handleiding APA uitgavehandleiding voor meer informatie en richtlijnen.

# Ontvangstbevestigingen

Hier inhoud invoegen...

Bedankt de belangrijkste mensen die u hebben geïnspireerd in het schrijfproces en hebben geholpen bij het uitgeven van uw werk. Dit is ongeveer hetzelfde als het dankwoord, met het verschil dat u hier meer kunt uitweiden en meer personen kunt noemen.

# Over de auteur

Hier inhoud invoegen...

Schrijf deze pagina in de derde persoon. (Gebruik bijvoorbeeld 'auteur' of uw naam, en niet 'ik'.) De informatie hierin laat de geloofwaardigheid van u zien bij lezers. Vermijd dat u teveel uitweidt. Geef uw achtergrond en kennis aan over het onderwerp van uw boek, evenals andere informatie die bijdraagt aan uw geloofwaardigheid. Bijvoorbeeld:

* Professionele en persoonlijk prestaties die betrekking hebben op het onderwerp
* Lijst met gepubliceerd werk en een koppeling naar uw website
* Opleidingsachtergrond
* Benoem andere vooraanstaande deskundigen in het veld waarmee u hebt gewerkt
* Uw woongebied, burgerlijke staat, hobby's, enzovoort.