

# Respec documentatie generatie

Tevens een voorbeeld van een gegenereerd Respec document



VNG Realisatie Handreiking  
In Gebruik versie 10 januari 2024

**Deze versie:**

<https://vng-realisatie.github.io/publicatie/cim/bv/respec-template/0.1.0>

**Laatst gepubliceerde versie:**

<https://vng-realisatie.github.io/publicatie/cim/bv/respec-template>

**Vorige versie:**

<https://vng-realisatie.github.io/publicatie/cim/bv/respec-template/0.0.4>

**Redacteur:**

Robert Melskens (VNG Realisatie)

Dit document is ook beschikbaar in dit niet-normatieve formaat: [pdf](#)



Dit document valt onder de volgende licentie:  
[EUROPEAN UNION PUBLIC LICENCE v. 1.2](#)

## Samenvatting

Dit document bevat een uitleg van de toepassing van Respec documentatie binnen VNG Realisatie.

- In hoofdstuk 1 wordt de werking van het template uitgelegd. Hoe je het kunt gebruiken voor je eigen specificatie, wat er precies gebeurt als je er mee aan de gang gaat en waar je daarbij op moet letten.
- Hoofdstuk 2 beschrijft hoe je binnen Imvector (indien toegepast) een deel van de content van je specificatie kan genereren. Dit hoofdstuk is optioneel aangezien je Respec document niet per definitie betrekking hoeft te hebben op een Informatiemodel.
- Hoofdstuk 3 beschrijft waar je op moet letten als je het GitHub Respec template kopieert (zowel als admin als als gebruiker), hoe je de Respec documentatie, evt. naast de door Imvector gegenereerde content, nog kan voorzien van andere content en hoe je door het aanpassen van configuration properties de stijl van het document kan veranderen daarbij rekening houdend met wat wij binnen VNG Realisatie met elkaar hebben afgesproken. In de flowchart van hoofdstuk 5 wordt dit hoofdstuk met de 'Generatie tak' geïllustreerd.
- Hoe de gegenereerde Respec bestanden gepubliceerd kunnen worden wordt in Hoofdstuk 4 beschreven. In de flowchart van hoofdstuk 5 wordt dit hoofdstuk met de 'Publicatie tak' geïllustreerd.
- In hoofdstuk 5 is een flowchart van het gehele proces voor het vervaardigen van Respec documentatie uitgewerkt. Dit is echter tevens een voorbeeld van het gebruik van Mermaid, een manier om flowcharts te vervaardigen.
- De hoofdstukken volgend op hoofdstuk 5 hebben slechts een illustratieve functie. Op basis daarvan moet je, samen met de uitleg in hoofdstuk 3, in staat zijn de getoonde functionaliteit te gebruiken.

## Status van dit document

Dit document is 'In Gebruik'.

## Inhoudsopgave

### Samenvatting

### Status van dit document

1. **ReSpec template instructies**
  - 1.1 Vereiste voor gebruik
  - 1.2 Gebruikersinstructie
  - 1.3 Rendering, automatische controles en publicatie

1.3.1	Rendering
1.3.2	Checken
<b>2.</b>	<b>Respec documentatie generatie m.b.v. Invertor</b>
<b>3.</b>	<b>Samenstellen Respec documentatie in GitHub (under construction)</b>
3.1	Door administrator uit te voeren acties
3.2	Door repository eigenaar uit te voeren acties
3.2.1	Invertor resultaat plaatsen
3.2.2	De content van het Respec document aanpassen
3.2.3	Secties met 'id' attribuutwaarde 'sotd'
3.2.4	Secties met 'id' attribuutwaarde 'abstract'
3.2.5	Secties met 'id' attribuutwaarde 'conformance'
3.2.6	Secties met 'id' attribuutwaarde 'tof'
3.2.7	Secties met 'id' attribuutwaarde 'index'
3.2.8	Secties met een andere 'id' attribuutwaarde
3.2.9	Secties met <code>data-include-format="html"</code>
3.2.10	Andersoortige secties
3.2.11	Bijlage N Referenties
3.2.12	Images in de documentatie
3.3	Lokale Respec configuratie properties
3.4	Functie Respec configuratie properties
<b>4.</b>	<b>Publiceren Respec documentatie (under construction)</b>
4.1	Inrichten publicatie repository
4.2	Publiceren van documenten
<b>5.</b>	<b>Foutoplossing</b>
<b>6.</b>	<b>Het VNG-R Respec proces</b>
<b>7.</b>	<b>Niet-normatieve deel</b>
<b>8.</b>	<b>Meer inhoud</b>
8.1	Definities
8.2	Afbeeldingen
8.3	Referenties
8.4	Optioneel
<b>9.</b>	<b>Conformiteit</b>
<b>10.</b>	<b>Lijst met figuren</b>
<b>A.</b>	<b>Index</b>
A.1	Begrippen gedefinieerd door deze specificatie
A.2	Begrippen gedefinieerd door verwijzing
<b>B.</b>	<b>Referenties</b>
B.1	Normatieve referenties

## 1. ReSpec template instructies

ReSpec is een tool om html en pdf documenten te genereren op basis van markdown en html content. Test.

Organisatie administrators dienen de knop [Use this template](#) te gebruiken om een kopie van de template repository aan te maken. Deze kan daarna door jouzelf aangepast en uitgebreid worden.

De dynamische pagina van het template document is [hier](#) te zien.

Deze repository bevat ook de GitHub Workflows om een statische HTML-pagina en PDF-document te genereren en enkele controles uit te voeren. Deze workflows worden automatisch gerund zodra er een aanpassing gedaan wordt aan de main branch. Een beschrijving van deze acties vind je onderaan dit hoofdstuk.

## 1.1 Vereiste voor gebruik

- Kennis van git/GitHub
- Kennis van Markdown en/of HTML
- Een webserver om de documentatie te hosten

Voor de laatste gebruiken wij GitHub. Kennis van de vorm van een Javascript object is handig om de in dit template voorkomende scripts aan te kunnen passen maar zonder die kennis kom je m.b.v. deze documentatie ook al heel ver.

## 1.2 Gebruikersinstructie

Om het gebruik van dit template makkelijker te maken raden we het aan om een IDE te gebruiken. Die geeft een voorbeeld van hoe de markdown eruit zal zien, kan laten zien of de config files nog in de correcte vorm zijn en kan helpen in het gebruik van git.

Een gratis voorbeeld van een IDE is: [Visual studio code](#). Een combinatie van GitHub desktop en je eigen favoriete Markdown editor is echter ook mogelijk.

Aanpassingen maken aan het document gaat op 2 manieren:

- De configuratie van het document aanpassing in de config files
- Markdown of html files toevoegen/veranderen

De **configuratie files** bevatten informatie over de organisatie en over de status van het document. Helemaal onderaan hoofdstuk 3 vind je meer informatie over de configuratie opties, daarnaast kun je ook de [Logius ReSpec wiki](#) bezoeken. De files zijn gesplitst in 2 files die weer in 2 verschillende repositories zijn ondergebracht: [organisation-config.js](#) en [config.js](#).

De `organisation_config` (`organisation-config.js`) bevat configuratie properties die betrekking hebben op alle VNG-R Respec documentatie, de properties in deze file zullen zelden veranderen zoals bijv. de naam van de organisatie.

De `document_config` (`config.js`) bevat configuratie properties die alleen relevant is voor het betreffende Respec document en hoort dan ook in elke Respec renderende repository thuis.

Beide configuratie bestanden worden gelinkt in de `index.html` file waardoor ze beide bij het renderen van de Respec documentatie automatisch worden samengevoegd. Daardoor zijn de organisatie specifieke configuraties over alle Respec documentatie van VNG-R gelijk en hoeft deze niet steeds gekopieerd te worden. Op deze wijze zorgen we er voor dat alle VNG-R Respec documenten zo eenduidig mogelijk zijn en blijven.

In [het volgende hoofdstuk](#) staat beschreven hoe je de inhoud van het Respec document naar wens kunt aanpassen.

## 1.3 Rendering, automatische controles en publicatie

Het bestand `'github/workflows/build.yml'` bevat een action script waarmee automatisch een drietal acties worden uitgevoerd nadat een bestand in de repository wordt gewijzigd, toegevoegd of verwijderd:

- het renderen van het Respec document;
- het checken of de gerenderde Respec html wel correct is en voldoet aan de toegankelijkheidseisen;
- het publiceren van de gegenereerde statische html en pdf naar een centrale Respec publicatie repository.

We beschrijven de eerste 2 acties in het kort hieronder. Aangezien we de laatste actie nog niet werkende hebben wordt deze voorlopig nog handmatig uitgevoerd, dat beschrijven we in hoofdstuk 4. De log van deze acties is te vinden in het tabblad Actions in de GitHub repository.

### 1.3.1 Rendering

Deze actie start het renderen van de Respec html. Vervolgens wordt er op basis daarvan een statische html en een pdf bestand gegenereerd. Die gebruiken we uiteindelijk om te publiceren.

De PDF-versie wordt aangemaakt omdat de property `alternateFormats` in de `organisation_config` als volgt geconfigureerd staat:

```
alternateFormats: [  
  {  
    label: "pdf",
```

```

    uri: "snapshot.pdf",
  },
]

```

Er moet nog worden bepaald of we de document eigenaar zelf willen laten bepalen of hij/zij een pdf wil genereren. Indien we dat willen verhuisd deze property naar de `document_config`. De waarde van de property `uri` kan dan naar wens worden aangepast.

### 1.3.2 Checken

Na het renderen van de Respec html en pdf worden er via github actions 2 controles uitgevoerd op de html:

- een WCAG-check (Web Content Accessibility Guidelines), deze guidelines gemaakt door W3C zorgen voor een verbetering van de toegankelijkheid van webapplicaties verbeterd voor zowel verschillende apparaten als voor mensen met een beperking. Ook wordt de validiteit van het HTML bestand gecheckt, bijv.:
  - of er geen `<section>` elementen met 'id' attributen voorkomen die al voorkomen in het bestand;
  - of er geen `<a>` elementen voorkomen met 'href' attributen die verwijzen naar `<section>` elementen die helemaal niet bestaan.

Deze check moet eerst succesvol uitgevoerd zijn voordat wordt begonnen aan de volgende check. In de 'Action' die start met het woord 'Update' (zie het Actions tabblad) kun je in de actie 'Check/WCAG' de step 'Run pa11y snapshot.html' vinden. Daar kun je zien welke fouten geconstateerd zijn.

- een link-check, deze check controleert of alle links die in het document staan ook bestaan. Het gaat dan bijv. om de links die worden vermeldt in:
  - 'Deze versie'
  - 'Laatst gepubliceerde versie'
  - 'Laatste werkversie'
  - 'Vorige versie'

Deze links verwijzen naar specifieke locaties in de GitHub Pages interface van de 'publicatie' GitHub repository (zie de volgende subparagraaf voor meer uitleg). Om deze check goed te kunnen doorstaan moeten deze locaties dus al bestaan in die interface. Indien dat nog niet gedaan is moet daar de folder voor het nieuwe versienummer van de Respec documentatie al worden aangemaakt. Plaats in die folder dan ook een tijdelijk 'index.html' bestand. De inhoud van dat bestand is (nog) niet van belang.

#### LET OP!

Onderstaand tekst is slechts een voorstel. De definitieve url kan indien gewenst nog andere onderdelen bevatten zoals bijv. `publishDate`, `previousPublishDate`, `specStatus` en `previousMaturity`.

Bij het genereren van de links zijn op dit moment de volgende configuration properties van belang:

- `nl_organisationPublishURL`  
De basis url van de GitHub Pages interface van de 'publicatie' GitHub repository, op dit moment: `https://vng-realisatie.github.io/publicatie`. Deze is gedefinieerd in de `organisation_config` aangezien deze altijd gelijk blijft.
- `pubDomain`  
Het publicatie domein. Aangezien we vooralsnog slechts voor Conceptuele Modellen Respec documentatie genereren heeft deze de waarde `cim` en staat deze gedefinieerd in de `organisation_config`. Zo nodig kan deze overruled worden in de `document_config`. Vergeet in dat geval niet om ook de structuur in de 'publicatie' GitHub repository uit te breiden. Wordt gebruikt in 'latestVersion', 'thisVersion' en 'prevVersion'.
- `specStatus`
- `latestVersion`  
Wordt opgebouwd a.d.h.v. een aantal andere configuratie properties uit zowel de `organisation_config` als de `document_config` en enkele vaste karakters. Deze is gedefinieerd in de `organisation_config` aangezien deze altijd gelijk blijft.
- `thisVersion`  
Wordt opgebouwd a.d.h.v. een aantal andere configuratie properties uit zowel de `organisation_config` als de `document_config` en enkele vaste karakters. Deze is gedefinieerd in de `organisation_config` aangezien deze altijd gelijk blijft.

- **prevVersion**  
Wordt opgebouwd a.d.h.v. een aantal andere configuratie properties uit zowel de `organisation_config` als de `document_config` en enkele vaste karakters. Deze is gedefinieerd in de `organisation_config` aangezien deze altijd gelijk blijft.
- **shortName**  
De project-mnemonic, een afkorting van het project. Zo wordt het project 'Open Raadsinformatie' wordt bijv. afgekort als 'ori'. Deze is gedefinieerd in de `document_config` aangezien deze natuurlijk afhankelijk is van het te genereren Respec document. Wordt gebruikt in 'latestVersion', 'thisVersion' en 'prevVersion'.
- **publishVersion**  
De versie van het te publiceren Respec document. Komt overeen met de Tagged Value 'Version' in het Enterprise Architect bestand van het model en heeft een waarde dat voldoet aan het formaat `x.x.x`, bijv. `2.0.0`. Deze is gedefinieerd in de `document_config` aangezien deze natuurlijk afhankelijk is van het te genereren Respec document. Wordt gebruikt in de titel van het Respec document maar ook in 'thisVersion'.
- **previousVersion**  
De voorgaande versie van het te publiceren Respec document. Komt overeen met de Tagged Value 'Version' in het Enterprise Architect bestand van het voorgaande versie van het model en heeft een waarde dat voldoet aan het formaat `x.x.x`, bijv. `2.0.0`. Deze is gedefinieerd in de `document_config` aangezien deze natuurlijk afhankelijk is van het te genereren Respec document. Wordt gebruikt in 'prevVersion'.

Het consistent en nauwgezet invullen van de configuratie properties in de `document_config` is essentieel voor een goede werking van de links.

De bovenstaande properties hebben invloed op de wijze waarop het eerste deel van de Respec documentatie wordt gegenereerd. Hieronder sommen we de regels op. Indien wordt besloten de properties 'latestVersion', 'thisVersion' en 'prevVersion' een andere inhoud te geven dan zullen onderstaande regels herzien moeten worden.

- Als de parameter 'specStatus' de waarde 'WV' heeft dan wordt de waarde van de parameter 'thisVersion' niet gebruikt voor het bepalen van 'Deze versie' maar wordt daar dezelfde waarde neergezet als bij 'Laatste werkversie'.
- Als de parameter 'specStatus' de waarde 'WV' heeft dan wordt de waarde van 'Subtitel 2' niet gebaseerd op de parameter 'publishDate' maar op de datum waarop de Respec documentatie door GitHub wordt gegenereerd.
- Als de parameter 'specStatus' de waarde 'WV' heeft dan wordt het versienummer niet in de titel van het document opgenomen.
- Als de parameter 'previousPublishVersion' niet bestaat dan kan 'prevVersion' niet bepaald worden en wordt 'Vorige versie' niet gegenereerd.
- Als de parameter 'publishVersion' niet bestaat dan kan 'thisVersion' niet bepaald worden en wordt 'Deze versie' niet gegenereerd.

## 2. Respec documentatie generatie m.b.v. Invertor

Het is mogelijk om met Invertor Respec documentatie te genereren van een model. Voorwaarde is wel dat het model MIM compliant is. Bij het genereren spelen de volgende Invertor configuratieproperties (**LET OP!** Dus niet een van de Respec property bestanden) een rol:

Configuratieproperty	Mogelijke waarden	Uitleg
createoffice	html, doc, none	Hiermee geef je aan of je een documentatie bestand wil genereren en zo ja in welk formaat (html of MsWord). De defaultwaarde is 'none', behalve in het geval van een SIM, daar is de default 'html'. De 'doc' optie is nog niet geïmplementeerd.
createofficeanchor	name, id	Geeft aan op welke basis hyperlink anchors moeten worden gegenereerd (op basis van id's of op basis van namen). De default is 'name'. Vooralsnog maakt het niet uit welke variant je voor deze property kiest, beide varianten leiden tot hetzelfde resultaat.
createofficemode	plain, click	Definieert of er in het te genereren bestand hyperlinks moeten worden gegenereert. Bij de waarde 'click' is dat het geval. De defaultwaarde is 'plain'.
createofficevariant	respec, msword	Definieert het type te genereren document. Een Respec html document of een MsWord html variant.
createimagemap	yes, no	Definieert of van de Diagrammen een imagemap moet worden gegenereerd. De default is 'yes'.

Voor het genereren van Respec documentatie is het essentieel om in je lokale Imvector property bestand de property 'createofficevariant' de waarde 'respec' te geven. Normaliter zal je dan ook de property 'createofficemode' de waarde 'click' geven. Dit resulteert er in dat in de folder 'app/cat' 2 Respec bestanden geplaatst, 1 in html en de ander in xhtml.

### 3. Samenstellen Respec documentatie in GitHub (under construction)

De acties die in het voorgaande hoofdstuk staan beschreven leveren een html bestand voor de Respec documentatie op waarin een informatiemodel wordt beschreven. Respec documentatie hoeft echter niet persé over informatiemodellen te gaan, voor de Respec documentatie die je nu leest is dat immers ook niet het geval. Het resultaat van het voorgaande hoofdstuk kan samen met andere html of markdown bestanden worden gebundeld tot de Respec documentatie. Daarnaast wordt een deel van de content van de Respec documentatie door het Respec framework in GitHub gegenereerd a.d.h.v. een aantal variabelen. Dat framework verzorgt daarnaast ook de vormgeving dat essentieel is voor de Respec documentatie.

Binnen VNG-R maken we gebruik van een door Logius vervaardigde extensie op het W3C Respec framework. We volgen daarbij andere organisaties in Nederland die hetzelfde doen zoals Geonovum. Van het door Logius beschikbaar gestelde template is een VNG-R versie beschikbaar binnen de VNG-Realisatie GitHub organisatie. Dat geeft de mogelijkheid om te verwijzen naar een VNG-R Respec configuratie waardoor we specifiek voor VNG-Realisatie geldende configuraties, zoals bijv. het VNG-Realisatie logo, kunnen aanbrengen. Deze vind je in de repository 'Respec-Organization-configurations'. Het template zelf kan echter door eenieder worden gebruikt om de eigen Respec documentatie te vervaardigen en daarin bestaan nog mogelijkheden om jouw Respec documentatie een invulling tintje te geven.

Hieronder wordt de werkwijze beschreven waarbij de 7 in de volgende paragraaf beschreven stappen moeten worden uitgevoerd door een GitHub organisatie administrator. Voorzie hem daarvoor van de gewenste repository naam.

#### 3.1 Door administrator uit te voeren acties

1. Open het [VNG-R Respec template](#) en klik in de README op die pagina op de link 'Use this template';
2. Je komt nu in het menu om een nieuwe repository aan te maken waarbij al een aantal velden zijn ingevuld. De te maken repository mag niet private zijn want dat maakt het gebruik van GitHub Pages onmogelijk. Geef de van de aanvrager verkregen repository naam in en klik op 'Create repository';
3. Voer de acties, zoals beschreven in [de handleiding voor het initieel inrichten van GitHub repositories](#), uit;
4. Verwijder in de root van de repository het 'README.md' bestand en hernoem 'Alt-README.md' naar 'README.md'

Dat bestand moet nog gecreëerd worden in het template;

6. Activeer GitHub Pages voor de nieuwe repository. Selecteer daarvoor het tabblad 'Settings' en kies daar 'Pages';
7. Kies daar waar bij Branch 'None' staat voor 'main' en klik op 'Save';
8. Nadat de build en deployment is uitgevoerd ga je naar het 'Code' tabblad, klikt daar op het tandwiel bij 'About' en klikt op de checkbox naast 'Use your GitHub Pages website'. Klikken op de resulterende link onder 'About' brengt je naar de standaard gegenereerde Respec documentatie die nu kan worden aangepast door de eigenaar van de repository;

#### 3.2 Door repository eigenaar uit te voeren acties

Je beschikt nu over een repository die je kunt gaan vullen en waarin je je persoonlijke configuratie properties van een waarde kunt voorzien. Indien je een met Imvector gegenereerd Respec html bestand wil gebruiken dan volg je de beschrijving van de volgende paragraaf, zo niet dan ga je direct naar de daarop volgende paragraaf.

##### 3.2.1 Imvector resultaat plaatsen

Plaats het met Imvector gegenereerde bestand in de root van de repository. Van dat bestand gebruiken we alleen de 'section' met het id 'cat'. Verwijder alle andere content behalve de processing instruction 'DOCTYPE HTML' aan het begin van dit bestand en commit het bestand. Open vervolgens het bestand 'index.html' en plaats daarin op de gewenste locatie het volgende html fragment:

```
<section id="XXXX" data-include-format="html" data-include="XXXX.html"></section>
```

Waarbij je 'XXXX.html' vervangt door de naam van het juist aangepaste bestand en 'XXXX' door een id dat de sectie duidelijk en uniek identificeert.

### 3.2.2 De content van het Respec document aanpassen

Een Respec document kan op 2 verschillende manier van content worden voorzien:

- m.b.v. de 'content' configuratie property;
- door de 'sectie' elementen aan het 'index.html' bestand toe te voegen.

Beide methodes kunnen elkaar aanvullen en kennen eigen functionaliteiten.

Het Respec document zoals dat van het VNG-R Respec template is overgenomen moet nog aangepast worden. Deels kan dat door in de 'index.html' secties aan te passen danwel te vervangen en deels door de configuration property 'content' aan te passen.

M.b.v. de 'content' configuratie property kunnen alleen secties waarvan de content in markdown bestanden staat worden toegevoegd. In deze property kan per bestand worden aangegeven of die sectie informatief is. Is dat het geval dan wordt automatisch de tekst `Dit onderdeel is niet normatief.` aan het hoofdstuk toegevoegd. Het toevoegen van bestanden aan de 'content' configuratie property doe je door de naam van het bestand (zonder de extensie) en een eventueel relevante CSS class in de property te plaatsen. De volgorde van bestanden binnen content bepaalt de volgorde in het resulterende document.

De code content: `{ "ch01": "informative", "mermaid": "" }`, voegt 2 markdown bestanden toe, te weten:

- `ch01.md` met de CSS class `informative`;
- `mermaid.md` zonder CSS class.

Voor een volledige lijst van CSS classes zie de [ReSpec Documentation](#). Deze classes zijn ook binnen de markdown files te gebruiken op de volgende manier:

```
<div class="example">voorbeeld</div>
```

Het gebruik van de 'content' properties is niet verplicht, er mag voor worden gekozen nieuwe content alleen toe te voegen door het 'index.html' bestand aan te passen. De 'content' property moet dan wel uit het lokale 'js/config.js' bestand worden verwijderd of worden uitcommentariseerd. Ook kan de plaats waar de in 'content' gedefinieerde hoofdstukken moeten worden toegevoegd worden aangepast. Zorg er dan voor dat het 'section' element waarna je die chapters wil toevoegen een 'id' attribuut met een waarde heeft en wijzig in het script in 'index.html' de regel

```
document.getElementById("id-van-sectie").insertAdjacentHTML('afterend', content);
```

zodanig dat de waarde 'id-van-sectie' de waarde van het id heeft.

In tegenstelling tot de methode met de 'content' configuratie property kunnen aan het 'index.html' bestand zowel 'sectie' elementen worden toegevoegd waarvan de content uit markdown bestaat als 'sectie' elementen waarvan de content uit html bestaat. Aangezien het gegenereerde Respec bestand een html bestand is kunnen we het alleen toevoegen aan het Respec document door een 'sectie' element toe te voegen aan het index.html bestand.

Bij de methode met de 'section' elementen maken we nog verschil tussen 'sectie' elementen met specifieke waarden voor het 'id' attribuut en 'sectie' elementen die andere waarden voor dat 'id' attribuut hebben of die zelfs helemaal geen 'id' attribuut hebben.

In de onderstaande paragrafen volgt per sectie een toelichting.

### 3.2.3 Secties met 'id' attribuutwaarde 'sotd'

Toe te voegen m.b.v. `<section id="sotd"></section>`. Leidt ertoe dat het hoofdstuk met de titel 'Status van het document' wordt toegevoegd met als inhoud de, van de waarde van de configuration property 'specStatus' afhankelijke, content van de configuration property 'sotdText'.

Tevens wordt een TOC gegenereerd waarin de titels (incl. evt. hoofdstuk en paragraafnummers) van alle, in het document opgenomen, hoofdstukken en paragrafen worden opgenomen afhankelijk van de configuratie property 'maxTocLevel'. Ook de titels van 'sectie' elementen zonder 'id' attribuut worden daar opgenomen.

Indien de configuration property 'content' bestaat dan worden de daarin gedefinieerde markdown bestanden na de 'sotd' sectie opgenomen. Zo niet dan worden de in de 'content' configuratie property gedefinieerde secties ook niet toegevoegd en wordt er ook geen TOC gegenereerd.

### 3.2.4 Secties met 'id' attribuutwaarde 'abstract'

Indien de sectie wordt toegevoegd met `<sectie id="abstract" data-include-format="markdown" data-include="filenaam.md">` dan krijgt het hoofdstuk de titel Samenvatting zonder hoofdstuknr. als inhoud wordt de inhoud van het bestand 'filenaam.md' toegevoegd.

### 3.2.5 Secties met 'id' attribuutwaarde 'conformance'

Door `<section id='conformance'></section>` wordt een hoofdstuk met als titel 'Conformiteit' toegevoegd.

De inhoud komt waarschijnlijk uit <https://github.com/Logius-standaarden/respec>. Het is nog niet duidelijk hoe dit hoofdstuk zijn inhoud krijgt.

### 3.2.6 Secties met 'id' attribuutwaarde 'tof'

`<section id='tof'></section>` genereert een hoofdstuk met als titel 'Lijst met Figuren' als er in minimaal een van de opgenomen bestanden minimaal een html 'figure' element met een 'figcaption' element is opgenomen of een markdown equivalent daarvan ( '[Tekstueel alternatief voor toegankelijkheid](pad naar illustratie bestand "Onderschrift")' ). In de markdown variant mag het onderschrift ontbreken.

De titel komt waarschijnlijk uit <https://github.com/Logius-standaarden/respec>. Het is nog niet duidelijk hoe die titel wordt toegekend.

### 3.2.7 Secties met 'id' attribuutwaarde 'index'

`<section id="index"></section>` genereert een hoofdstuk met als titel 'Bijlage N Index' als er in minimaal 1 van de in het document opgenomen bestanden (zowel markdown als html) minimaal 1 'dfn' element is opgenomen. Vanuit de tekst kan naar dat element verwezen worden door een 'a' element op te nemen zonder attributen maar met als inhoud de naam van een 'dfn' element.

### 3.2.8 Secties met een andere 'id' attribuutwaarde

- Indien de sectie wordt toegevoegd met `<sectie id="nnnnnn" data-include-format="markdown" data-include="filenaam.md">` dan wordt het hoofdstuk gevuld met de inhoud van 'filenaam.md'. Als 'filenaam.md' met een markdown titel start (ongeacht het level en het aantal blanco regels er voor) dan wordt een hoofdstuknummer voor die titel gegenereerd anders wordt de content zonder titel toegevoegd aan het document. Een evt. titel wordt ook opgenomen in de TOC.
- Indien de sectie wordt toegevoegd met `<sectie data-include-format="markdown" data-include="filenaam.md">` dan wijkt het resultaat niet af van die van hierboven. Alleen wordt bij deze variant het 'id' van de sectie en de gerelateerde 'href' in de TOC gegenereerd op basis van de titel van deze sectie.

In alle gevallen is `data-include-format="markdown"` verplicht.

### 3.2.9 Secties met `data-include-format="html"`

Dit soort secties wordt direct opgenomen op de plaats waar `<section id="nnnn" data-include-format="html" data-include="filenaam.html"></section>` is geplaatst.

Het html fragment in het bestand hoeft niet te bestaan uit 1 root element. Sterker nog als dat wel het geval is en het fragment heeft de root 'div' of 'sectie' dan wordt het fragment niet vertaald naar een separaat hoofdstuk.

Om een separaat hoofdstuk te kunnen starten dient het document wel met een 'hx' element te starten (h1, h2, h3, etc..).

De titel wordt dan ook opgenomen in de TOC.

Dit soort secties mag ook zonder 'id' attribuut worden opgenomen. Die variant geeft geen ander resultaat dan die hiervoor geschetst. Alleen wordt bij deze variant het id van de sectie en de gerelateerde href in de TOC gegenereerd op basis van de titel van deze sectie.

`data-include-format="html"` mag worden weggelaten.



### 3.2.10 Andersoortige secties

Indien een sectie element leeg is en het 'id' komt niet overeen met een van de, in de voorgaande paragrafen beschreven, bekende id's dan wordt de sectie genegeerd.

### 3.2.11 Bijlage N Referenties

Wordt alleen opgenomen als er in een van de andere documenten (zowel markdown als html) een referentie is opgenomen in de vorm '[[Ref]]' en die referentie in config.js of organisation-config.js is gedefinieerd.

### 3.2.12 Images in de documentatie

Plaats eventuele images die je in de Respec documentatie wil opnemen in de 'media' folder. Daarbinnen mag je elke door jou gewenste folderstructuur creëren.

## 3.3 Lokale Respec configuratie properties

Zoals aangegeven maken we in het Respec framework gebruik van een aantal VNG-R properties. Properties die er voor zorgen dat alle Respec documentatie van VNG-R eenzelfde look en feel heeft. Er zijn echter ook een aantal lokale configuratie properties waarmee voor ieder Respec document eigen keuzes kunnen worden gemaakt. Denk daarbij aan de status die het document heeft, de publicatie datum, de editors, etc...

Alle lokale configuratie properties kun je vinden in 'js/config.js' en mag je naar eigen inzicht aanpassen.

Er moet nog bepaald worden welke properties lokaal moeten zijn en welke globaal (dus welke behoren te staan in de repository 'Respec-Organization-configurations').

## 3.4 Functie Respec configuratie properties

Hieronder vind je de totale lijst van Configuratie properties. De vierde kolom geeft aan of het als een globale of lokale property wordt ingeschat, daar moet echter nog een discussie over gevoerd worden. Voor sommige properties is die inschatting heel logisch, Zo zijn 'localizationStrings' en 'logos' logischerwijs globaal, 'github' en 'previousPublishVersion' zijn juist lokaal. Een aantal properties worden globaal gedefinieerd maar kunnen lokaal overruled worden zoals 'useLogo'. In de op een na laatste kolom staan vragen of opmerkingen die we moeten bediscussieren en de laatste kolom geeft aan of de omschrijving van de property nog aandacht verdiend of van voldoende kwaliteit is.

De laatste 2 kolommen hebben de bedoeling het proces om het onderliggende document te vervolmaken te ondersteunen en kunnen, als alle properties bediscussieerd zijn, worden verwijderd. De kolom 'Documentatie status' geeft daarbij aan of betreffende property voldoende gedocumenteerd is om bediscussieerd te kunnen worden.

Property	<a href="#">Link</a>	Type	Gloobaal/Lokaal	Vaste globale waarde of default waarde	Gerelateerd property	Beschrijving
						Bepaald of er een paragraafteken (§), n de paragraaf waar het teken vóór komt gegenereerd of niet.
addSectionLinks	<a href="#">link</a>	boolean	Gloobaal	false		Kan handig zijn om anderen de gelegenheid om links naar specifieke paragrafen in te kopiëren en elders te gebruiken.
						Kan lokaal overruled worden.
alternateFormats	<a href="#">link</a>	Array met per formaat de properties 'label' en 'url'.	Lokaal			Hiermee kun je aangeven of je de Respec ook in een ander formaat dan html aan het moment alleen pdf. Deze configuratie property zorgt er voor dat een bestand wordt gegenereerd en dat er in de documentatie een zin gewijd wordt aan het bestand daarin de link naar het pdf bestand.

Property	<a href="#">Link</a>	Type	Globaal/Lokaal	Vaste globale waarde of default waarde	Gerelateerd property	Beschrijving
authors	<a href="#">link</a>	Array met per naam de properties 'name', 'company' en 'companyURL'.	Lokaal			<p>Bevat 1 of meerdere beschrijvingen van auteurs die hebben bijgedragen aan de tot stand koming van de specificatie.</p> <p>Het heeft de voorkeur editors te gebruiken voor auteurs. Indien deze configuratie property is wordt 'Auteurs' niet getoond.</p>
content	n.t.b.	Array (zie een beschrijving onder deze tabel).	Lokaal			Te gebruiken voor het toevoegen van content aan een Respec document.
editors	<a href="#">link</a>	Array met per naam de properties 'name', 'company' en 'companyURL'.	Lokaal			<p>1 of meerdere beschrijvingen van personen die hebben bijgedragen aan de tot stand koming van de specificatie.</p> <p>Het heeft de voorkeur editors te gebruiken voor auteurs. Indien deze configuratie property is wordt 'Redacteurs' getoond zonder vermelding van de naam.</p>
formerEditors	<a href="#">link</a>	Array met per naam de properties 'name', 'company' en 'companyURL'.	Lokaal			Bevat 1 of meerdere beschrijvingen van personen die in het verleden hebben bijgedragen aan de tot stand koming van de specificatie.
github	<a href="#">link</a>	URI of een array van de properties 'repoURL' en 'branch'.	Lokaal			<p>Gebruikt voor het genereren van de link naar de GitHub repository van de lin tabel bovenin de Respec documentatie. Het wordt gebruikt met</p> <ul style="list-style-type: none"> <li>• een url naar een GitHub repository</li> <li>• het deel van de url van een GitHub repository dat na 'https://github.com/' komt</li> <li>• een set van properties bestaande uit <ul style="list-style-type: none"> <li>◦ repoURL: Een van bovenstaande</li> <li>◦ branch: de branch waarin de documentatie ook issues staan opgeslagen.</li> </ul> </li> </ul> <p>Verwijst naar de GitHub repository waar het Informatiemodel wordt beheerd.</p>



Property	<a href="#">Link</a>	Type	Globaal/Lokaal	Vaste globale waarde of default waarde	Gerelateerd property	Beschrijving
						<p>Deze property kan zowel lokaal als globaal overruled worden.</p> <p>Voor referenties waarvan we verwachten dat deze vaker gebruikt worden dient er een default waarde te worden gedaan deze op te nemen in de organisation-config.js. Beheerders van repositories zijn er vanaf dat moment verantwoordelijk voor dat deze referenties in de config.js worden verwijderd.</p>
localizationStrings	<a href="#">link</a>	Array van properties per taalcode	Globaal			Bevat voor een aantal doelen en talen verschillende teksten en de daarbij horende codes.
logos	<a href="#">link</a>	Array per logo van de properties 'src', 'alt', 'id', 'height' en 'url'.	Globaal	VNG Realisatie logo		<p>definieert de src, alternate tekst, url en linksboven in de specificatie te plaatsen.</p> <p>Mag niet Lokaal overruled worden.</p>
maxTocLevel	<a href="#">link</a>	Integer	Lokaal	0		<p>Bepaald het aantal niveau's dat maximaal opgenomen in de inhoudsopgave van het document.</p> <p>Kan lokaal overruled worden.</p>
nl_organisationName	<a href="#">link</a>	String	Globaal	VNG Realisatie		<p>Wordt gebruikt om de subtitel en het logo linksboven te genereren.</p> <p>Kan lokaal overruled worden.</p>
nl_organisationPublishURL	<a href="#">link</a>	URL	Globaal	?		<p>Wordt gebruikt voor het genereren van GitHub pages van de huidige, de vorige gepubliceerde versie. Een link die leidt naar het document in de GitHub Pages interface van de 'publicatie' GitHub repository. De laatste gepubliceerde versie is overigens wat anders dan de laatste versie van de lokaal gepubliceerde versie.</p> <p>Kan worden gebruikt in de properties 'thisVersion' en 'prevVersion'.</p>

Property	<a href="#">Link</a>	Type	Gloobaal/Lokaal	Vaste globale waarde of default waarde	Gerelateerd property	Beschrijving
						Mag niet lokaal overruled worden.
nl_organisationStylesURL	<a href="#">link</a>	URL	Gloobaal			Definieert de lokatie waar het te gebruiken bestand staat excl. dat bestand zelf.  Mag niet Lokaal overruled worden.
noTOC	<a href="#">link</a>	boolean	Lokaal	false		Bepaald of er links van de inhoud een inhoudsopgave gegenereerd moet worden.
otherLinks	<a href="#">link</a>	Array van properties	Lokaal			Genereert een sectie in de header van het document met als titel de key van deze property. Inhoud een of meerdere links.
postProcess	<a href="#">link</a>		?	?		Bevat een of meer JavaScript functies die achtereenvolgend opgestart worden na de generatie van het Respec document.
previousMaturity	<a href="#">link</a>	enumeration	Lokaal			Status van de voorgaande versie.  Kan evt. worden gebruikt in de property 'previousVersion' maar wordt daar nu niet gebruikt.
previousPublishDate	<a href="#">link</a>	Datum in het formaat YYYY-MM-DD	Lokaal			Publicatiedatum van de voorgaande versie.  Kan evt. worden gebruikt in de property 'previousVersion' maar wordt daar nu niet gebruikt.
previousPublishVersion	<a href="#">link</a>	SemVer notatie	Lokaal			Versienummer van de voorgaande versie in SemVer notatie ( <a href="https://semver.org/lang/nl/">https://semver.org/lang/nl/</a> ).  Wordt gebruikt in de property 'prevVersion'.
prevVersion	<a href="#">link</a>	Combinatie van strings en propertynamen	?			Url van de voorgaande versie. Wordt opgebouwd m.b.v. andere gedefinieerde configuration properties en '/' tekens. De voorkomende hoofdletters worden omgezet in kleine letters.  Indien deze configuration property niet gedefinieerd is dan wordt de gerelateerde rubriek in de configuration niet aangemaakt.
pubDomain	<a href="#">link</a>	enumeration	Lokaal			Definieert het publicatie domein van de publicatie heeft op dit moment de waarde 'cim'.  Wordt nu gebruikt in de properties 'lastVersion', 'thisVersion' en 'prevVersion'.

Property	<a href="#">Link</a>	Type	Globaal/Lokaal	Vaste globale waarde of default waarde	Gerelateerd property	Beschrijving
publishDate	<a href="#">link</a>	Datum in het formaat YYYY-MM-DD	Lokaal			Publicatiedatum van de huidige versie.  Kan evt. worden gebruikt in de properties 'lastVersion' en 'thisVersion'.
publishVersion	<a href="#">link</a>	SemVer notatie	Lokaal			Versienummer van de huidige versie in (https://semver.org/lang/nl/).  Wordt gebruikt in de properties 'lastVersion' en 'thisVersion'.
shortName	<a href="#">link</a>	String	Lokaal			Korte naam (mnemonic) van de specificatie.  Wordt gebruikt in de properties 'lastVersion', 'thisVersion' en 'prevVersion'.
sotdText	<a href="#">link</a>	Array van properties per taalcode.	Globaal			Bevat voor een aantal 'specStatus's en de daarbij horende labels.  Mag niet lokaal overruled worden.
specStatus	<a href="#">link</a>	enumeration	Lokaal			Definieert de status van de specificatie om de subtitel en het verticale label line te genereren. Bepaald ook de kleur van de status in de lokale configuratie gedefinieerd te worden.  De kleuren voor de VNG-R statussen komen voort uit de gedefinieerde opties 'labelColor' en 'labelColor2' in de globale opties.
spectype	<a href="#">link</a>	enumeration	Lokaal			Kan evt. ook worden gebruikt in de properties 'latestVersion', 'thisVersion' en 'prevVersion'.  Definieert het type van de specificatie, om de subtitel en het verticale label line te genereren. In het template heeft dit de waarde 'VNG-R Respecificatie'.  aangezien we bij VNG-R Respecificatie veelal om Informatiemodellen mee te publiceren.

Property	<a href="#">Link</a>	Type	Globaal/Lokaal	Vaste globale waarde of default waarde	Gerelateerd property	Beschrijving
						Kan evt. ook worden gebruikt in de properties 'latestVersion', 'thisVersion' en 'prevVersion'.
subtitle	<a href="#">link</a>	String	Lokaal	n.v.t.		String die als subtitel van de titel van het document dient. Wordt geplaatst boven de gegenereerde inhoud waarin de organisatienaam, documenttitel en versiedatum worden gebruikt.
testSuiteURI	<a href="#">link</a>	URL	Lokaal	n.v.t.		Genereert een sectie in de header van het document met als titel 'Test suite' en als inhoud een lijst met testsuites. Wellicht te gebruiken voor een Testplatform maar alleen als we Respe gebruiken voor de API's.
thisVersion	<a href="#">link</a>	Combinatie van strings en propertynamen	Globaal			<p>Url van de huidige versie.</p> <p>Wordt opgebouwd m.b.v. andere gedefinieerde configuration properties en '/' tekens. De afzonderlijke voorkomende hoofdletters worden omgezet in kleine letters.</p> <p>Het is verplicht deze property van een waarde te voorzien en deze mag niet Lokaal overruled worden.</p>
title	-		Lokaal			De titel van de betreffende specificatie
useLabel	<a href="#">link</a>	boolean	Globaal	true		<p>Bepaald of het verticale label aan de linkzijde van de inhoudsopgave gegenereerd moet worden.</p> <p>Kan lokaal overruled worden.</p>
useLogo	<a href="#">link</a>	boolean	Globaal	true		<p>Bepaald of het VNG-Realisatie logo in de bovenzijde van het document geplaatst moet worden.</p> <p>Kan lokaal overruled worden.</p>
edDraftURI	<a href="#">link</a>	URL	?			Beschrijft de url waar de draft van de specificatie wordt worden bekeken. Er wordt geadviseerd niet te publiceren. Indien deze property niet is gedefinieerd wordt 'Laatste werkversie' niet gebruikt.

## 4. Publiceren Respec documentatie (under construction)

Het is de bedoeling dat het publiceren van de statische html en pdf geautomatiseerd gaat verlopen. Dat hebben we helaas nog niet op orde en om die reden doen we dit voorlopig handmatig.

Het publiceren van Respec documentatie gebeurt niet vanuit de GitHub repository waarin deze wordt samengesteld en gegenereerd. In die repository kan nl. slechts één versie van die Respec documentatie met GitHub Pages worden aangeboden terwijl we de mogelijkheid willen hebben alle voorgaande versies ook aan te bieden. Anders zouden de links in het eerst deel van de Respec documenten ook dood lopen. Om die reden wordt bij het genereren van Respec documentatie het bestanden 'snapshot.html' en een pdf bestand gegenereerd zodat we deze bestanden kunnen kopiëren naar de 'publicatie' repository vanwaaruit we alle Respec documentatie publiceren.

### 4.1 Inrichten publicatie repository

#### LET OP!

De hieronder uitgewerkte structuur van de publicatie GitHub repository is slechts een voorstel.

Voor het publiceren van alle Respec documenten in de GitHub repository 'publicatie' is de hieronder beschreven consistente structuur vereist. Deze consistente structuur moet overeenkomen met de waarde van in het globale configuratiebestand gedefinieerde properties 'latestVersion', 'thisVersion' en 'prevVersion'. Indien wordt besloten de inhoud van die properties aan te passen dan heeft dat ook gevolgen voor de repository structuur.

folder- of bestandsnaam	Opmerking
cim	Het publicatie domein
[project-mnemonic]	De afkorting van het project. 'Open Raadsinformatie' wordt bijv. afgekort als 'ori'.
[x.x.x]	Het versienummer van het Respec document. Deze folder kan zich herhalen met steeds een ander versienummer. Komt minimaal 1x voor met als naam het huidige versienummer.
media	Folder waarin evt. illustraties staan opgenomen
index.html	Gegenereerde statische html (snapshot.html) welke overeenkomt met de versie, hernoemd.
cim-[project-mnemonic]-[x.x.x].pdf	Gegenereerd pdf bestand welke overeenkomt met de versie.
media	Folder waarin evt. illustraties staan opgenomen
index.html	Gegenereerde statische html (snapshot.html), welke overeenkomt met de huidige versie, hernoemd.
cim-[project-mnemonic]-[x.x.x].pdf	Gegenereerd pdf bestand welke overeenkomt met de huidige versie.

In dit overzicht vertegenwoordigd `x.x.x` het versienummer van het Respec document. De folder met die naam kan zich herhalen maar komt minimaal 1x voor, de huidige versie. De inhoud van die folder wordt dan eveneens geplaatst in de folder `[project-mnemonic].project-mnemonic` is Hieronder zie je een voorbeeld van deze structuur:





*Figuur 1*

Tijdens het samenstellen van de inhoud van een nieuwe versie van een bestaand of geheel nieuw Respec document kun je de structuur in de 'publicatie' repository al aanpassen. Betreft het de eerste versie van een Respec document creëer dan direct in de folder 'cim' een nieuwe folder die als naam de mnemonic van het project krijgt. Daarbinnen creëer je een versiefolder met als naam het versienummer waaronder je de eerste versie publiceert. Bestaat de projectfolder al wel dan creëer in die folder een nieuwe versiefolder met als naam het versienummer waaronder je de nieuwe versie publiceert. In beide folders creëer je voorlopig een leeg 'index.html' bestand.

## 4.2 Publiceren van documenten

Na generatie kunnen de bestanden 'snapshot.html' en 'snapshot.pdf' vanuit de GitHub repository waarin ze gegenereerd zijn worden gekopieerd projectfolder en naar de naar de zojuist aangemaakte versiefolder in de 'publicatie' repository. Het bestand 'index.html' verwijder je vervolgens in beide folders waarna je 'snapshot.html', eveneens in beide folder, hernoemt naar 'index.html'. Indien je in je Resoec documentatie images hebt geplaatst dan heb je die bestanden in de 'media' folder opgeslagen, kopieer in dat geval de 'media' folder eveneens naar beide folders.

Pas vervolgens in de root van de publicatie repository de 'index.md' aan zodat je toegang hebt tot alle Respec documentatie en de http links kunt kopiëren voor gebruik in andere GitHub Pages documentatie.

## 5. Foutoplossing

Nadat een bestand in een op de VNG-R-Respec-Temp[late repository gebaseerde repository wordt gewijzigd, toegevoegd of verwijderd treedt het 'Build' action script in werking. Er kunnen zich daarbij diverse problemen voordoen. Detectie van een probleem start altijd bij het onderzoeken van de Action workflows. Deze kun je benaderen door in GitHub het tabblad 'Actions' aan te klikken waarna. In het daarop volgende scherm tref je vervolgens een tweetal typen workflows aan.

- Het eerste type draagt over het algemeen de naam van een gewijzigd bestand en dit zal over het algemeen het type workflow zijn waarin zich de problemen voordoen. Om beter te kunnen bepalen wat er precies fout ging moet je op de workflow klikken. In het daarop volgende menu kun je de uitgevoerde modules zien en waar het fout is gegaan, ook daar kun je weer op klikken. Tenslotte kun je daar weer de foutgelopen stap zien Ter ondersteuning bij het oplossen van die problemen beschrijven we er hier een aantal en geven we aan wat daar de mogelijke oorzaak van is en hoe je dat kunt oplossen.

Onderstaande flowchart beschrijft het proces zoals we dat binnen VNG Realisatie hanteren om tot Respec documentatie te komen. Daarnaast is het echter ook een voorbeeld van het gebruik van de Mermaid syntax voor het vervaardigen van zo'n

flowchart. Zie de [Flowchart syntax](#) voor een uitleg van de Mermaid syntax.





*Figuur 3 Het VNG-R Respec proces (Mermaid voorbeeld)*

## 7. Niet-normatieve deel

*Dit onderdeel is niet normatief.*

Bijvoorbeeld een introductie.

NOOT: index

Dit hoofdstuk is toegevoegd met `class="informative"` in `config.js`.

## 8. Meer inhoud

### 8.1 Definities

**Definitie:** Een definitie is een beschrijving van een woord. Een ander woord voor definitie is betekenis of beschrijving. De notatiewijze voor het definiëren van een definitie is als volgt:

`<dfn>Definitie</dfn>`

Deze definities worden opgenomen in de bijlage 'Index'.

### 8.2 Afbeeldingen

Afbeeldingen krijgen een nummer en vermelding in de figurenlijst [10. Lijst met figuren](#).



*Figuur 4 Onderschrift*

### 8.3 Referenties

Referentie uit globale [\[SemVer\]](#) of lokale [\[MIM\]](#) localBiblio lijst. Deze lijst staat in de `organisation-config.js` of `config.js`, zie hieronder een voorbeeld.

```
localBiblio: {
  "MIM": {
    "href": "https://docs.geostandaarden.nl/mim/mim/",
    "publisher": "Geonovum",
    "title": "MIM - Metamodel Informatie Modelling",
    "date": "Oktober 2013",
    "rawDate": "2023"
  },
}
```

Alleen referenties die ook echt in de tekst voorkomen worden in de bijlage 'Referenties' getoond. De notatie voor het opnemen van een referentie is `[[xxxx]]`.

We gebruiken de notatie `<a>xxxx</a>` om een referentie naar een in het document aangebrachte definitie te creëren. Het resultaat ziet er dan bijv. zo [definitie](#) uit.

### 8.4 Optioneel

De onderstaande secties (*Conformiteit* e.d.) zijn optioneel, zie `index.html`:

```
<body>
  <section id="abstract" data-include-format="markdown" data-include="abstract.md"></section>
  <section id="sotd"></section><!-- Wordt automatisch gevuld -->
  <section data-include-format="markdown" class="informative" data-include="ch01.md"></section>
  <section data-include-format="markdown" data-include="ch02.md"></section>
  <!-- Hieronder optionele secties. Worden automatisch gevuld -->
  <section id='conformance'></section>
  <section id='tof'></section>
  <section id="index"></section>
</body>
```

## 9. Conformiteit

Naast onderdelen die als niet normatief gemarkeerd zijn, zijn ook alle diagrammen, voorbeelden, en noten in dit document niet normatief. Verder is alles in dit document normatief.

## 10. Lijst met figuren

Figuur 1

Figuur 2

Figuur 3 Het VNG-R Respec proces (Mermaid voorbeeld)

Figuur 4 Onderschrift

## A. Index

### A.1 Begrippen gedefinieerd door deze specificatie

Definitie §8.1

### A.2 Begrippen gedefinieerd door verwijzing

## B. Referenties

### B.1 Normatieve referenties

**[MIM]**

*MIM - Metamodel Informatie Modelling*. Geonovum. Oktober 2013. URL: <https://docs.geostandaarden.nl/mim/mim/>

**[SemVer]**

*Reference not found.*

↑