

Generating Respec documentation 1.0.0

VNG Realisatie Handreiking

Vastgestelde versie 14 december 2023

**Deze versie:**

<https://melsk-r.github.io/publicatie/cim/respec-template/1.0.0>

Laatst gepubliceerde versie:

<https://melsk-r.github.io/publicatie/cim/respec-template>

Laatste werkversie:

<https://melsk-r.github.io/VNG-R-Respec-Template/>

Redacteur:**Auteur:**

Robert Melskens ([VNG Realisatie](#))

Doe mee:

[GitHub melsk-r/VNG-R-Respec-Template](#)

[Dien een melding in](#)

[Revisiehistorie](#)

[Pull requests](#)

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

Tekst

Status van dit document

Dit is de definitieve versie van dit document. Wijzigingen naar aanleiding van consultaties zijn doorgevoerd.

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
 - 1.4 Publiceren van documenten
- 2. Respec documentatie (under construction)**
 - 2.1 Generatie m.b.v. Imvortor
 - 2.2 Toepassing in GitHub
 - 2.2.1 Door administrator uit te voeren acties
 - 2.2.2 Door repository eigenaar uit te voeren acties
 - 2.2.3 De content van het Respec document aanpassen
 - 2.2.3.1 Secties met 'id' attribuutwaarde 'abstract'
 - 2.2.3.2 Secties met 'id' attribuutwaarde 'sotd'
 - 2.2.3.3 Secties met `data-include-format="html"`
 - 2.2.3.4 Secties met 'id' attribuutwaarde 'conformance'
 - 2.2.3.5 Secties met 'id' attribuutwaarde 'tof'
 - 2.2.3.6 Secties met 'id' attribuutwaarde 'index'

2.2.3.7	Bijlage N Referenties
2.2.3.8	Secties met een 'id' attribuutwaarde die niet overeenkomt met de hierboven geschetste varianten
2.2.4	Lokale Respec configuratie properties
2.2.5	Functie Respec configuratie properties
3.	ReSpec template instructies
3.1	Vereiste voor gebruik
3.2	Gebruikersinstructie
3.3	Rendering, automatische controles en publicatie
3.3.1	Rendering
3.3.2	Checken
3.4	Publiceren van documenten
4.	Niet-normatieve deel
5.	Meer inhoud
5.1	Definities
5.2	Afbeeldingen
5.3	Referenties
5.4	Optioneel
6.	Mermaid diagram
7.	Conformiteit
8.	Lijst met figuren
A.	Index
A.1	Begrippen gedefinieerd door deze specificatie
A.2	Begrippen gedefinieerd door verwijzing
B.	Referenties
B.1	Normatieve referenties
B.2	Informatieve referenties

1. ReSpec template instructies

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

Gebruik de knop [Use this template](#) om aan de slag te gaan. Dit maakt een kopie van de template in uw eigen GitHub repository die dan aangepast en uitgebreid kan 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. <-- Deze heb ik deels nog niet werkende. Eenbeschrijvingvan deze acties vind je onderaan dit hoofdstuk.

1.1 Vereiste voor gebruik

- Kennis van git/github
- Kennis van markdown en/of HTML
- Kennis van de vorm van een Javascript object
- Een webserver om de documentatie te hosten

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](#).

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. Bekijk de [Logius ReSpec wiki](#) voor meer informatie over de configuratie opties. 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 `VNG-R-WOW.md` 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 alle 3 de acties in het kort hieronder. Aangezien we de laatste actie nog niet werkende hebben wordt deze voorlopig nog handmatig uitgevoerd, ook dat beschrijven we. De resultaten van deze acties zijn 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 laatste 2 worden in de laatste actie gebruikt om te publiceren.

De PDF-versie wordt alleen aangemaakt indien `alternateFormats` in de `document_config` (`config.js`) als volgt geconfigureerd staat:

```
alternateFormats: [  
  {  
    label: "pdf",  
    uri: "template.pdf",  
  },  
]
```

De waarde van de property `uri` moet natuurlijk 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 evenals de structuur van de publicatie GitHub repository. De definitieve url kan indien gewenst nog andere onderdelen bevatten zoals bijv. `publishDate`, `previousPublishDate`, `specStatus`, `previousMaturity`.

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

- `nl_organisationPublishURL`
De basis url van de GitHub Pages interface van de 'publicatie' GitHub repository, op dit moment: `https://melsk-r.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 aan te passen.
- `latestVersion`
Wordt opgebouwd a.d.h.v. een aantal andere configuratie properties uit zowel de `organisation_config` als de `document_config` en enkele statische waardes. 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 statische waardes. 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 statische waardes. 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.
- `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.
- `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.

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

1.4 Publiceren van documenten

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 alle Respec documenten gebeurt via de GitHub repository 'publicatie' waarin de volgende consistente structuur vereist is:

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

De statische html en pdf worden vanuit de GitHub repository waarin ze gegenereerd zijn gekopieerd naar de gewenste folders in de 'publicatie' repository. Als het goed is heb je de gewenste folders in een van de voorgaande stappen al aangemaakt.

2. Respec documentatie (under construction)

2.1 Generatie m.b.v. Imvertor

Het is mogelijk om met Imvertor Respec documentatie te genereren van een model. Daarbij spelen de volgende configuratieproperties 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'. Tussen het genereren van respec als msword treden geen verschillen op. Vooralsnog maakt het dus niet uit welke variant je voor deze property kiest.
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'.

Configuratieproperty	Mogelijke waarden	Uitleg
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 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.

2.2 Toepassing in GitHub

De acties die in de voorgaande paragraaf staan beschreven leveren alleen het html bestand voor de Respec documentatie op. Respec documentatie bestaat echter uit meer dan dat html bestand. Een deel van de content van de Respec documentatie wordt door het Respec framework in GitHub gegenereerd a.d.h.v. een aantal variabelen en toe te voegen html en/of md bestanden. Daarnaast verzorgt dat framework ook de vormgeving dat essentieel is voor 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 daarbinnen bestaan nog mogelijkheden om jouw Respec documentatie een eigen tintje te geven.

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

2.2.1 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 is 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;

2.2.2 Door repository eigenaar uit te voeren acties

8. Je beschikt nu over een repository die je kunt gaan vullen en waarin je je persoonlijke configuratie properties van een waarde kunt voorzien. Plaats daartoe als eerste het in de voorgaande paragraaf gegenereerde html bestand in de root van de repository;
9. Van het bestand dat we zojuist geplaatst hebben 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;
10. Open het bestand 'index.html' en plaatst daarin op de gewenste plaats 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 zojuist aangepaste bestand en 'XXXX' door een id dat de sectie

duidelijk en uniek identificeert. Als je nu op de website link onder 'About' klikt dan vind je de eerste versie van je Respec document;

2.2.3 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.

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 'sectie' 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.

Hieronder volgt per sectie een toelichting.

2.2.3.1 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.
- 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.

2.2.3.2 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.

2.2.3.3 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.

2.2.3.4 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.

2.2.3.5 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.

2.2.3.6 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.

2.2.3.7 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.

2.2.3.8 Secties met een 'id' attribuutwaarde die niet overeenkomt met de hierboven geschetste varianten

Indien het 'id' niet overeenkomt met een van de bekende id's dan wordt de sectie genegeerd.

2.2.4 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').

2.2.5 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 verdient 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	Link	Type	Globaal/Lokaal	Vaste globale waarde of default waarde	Gerelateerd property	Beschrijving
addSectionLinks	link	boolean	Globaal	false		Bepaald of er een paragraafteken (§), na de paragraaf waar het teken vóór komt gegenereerd of niet. Kan handig zijn om anderen de gelegenheid te geven om links naar specifieke paragrafen in te kopiëren en elders te gebruiken. Kan lokaal overruled worden.
alternateFormats	link	Array met per formaat de properties 'label' en 'url'.	?			Hiermee kun je aangeven of je de Respec documentatie ook in een ander formaat dan html aan het publiceren moment alleen pdf. Deze configuratie property zorgt er voor dat het bestand wordt gegenereerd en dat er in de documentatie een zin gewijd wordt aan het gebruik van daarin de link naar het pdf bestand.
authors	link	Array met per naam de properties 'name', 'company' en 'companyURL'.	Lokaal			Bevat 1 of meerdere beschrijvingen van de auteurs die hebben bijgedragen aan de tot stand komst van de specificatie. Editors hebben de voorkeur boven auteurs.

Property	Link	Type	Globaal/Lokaal	Vaste globale waarde of default waarde	Gerelateerd property	Beschrijving
content	n.t.b.	Array (zie een beschrijving onder deze tabel).	Lokaal			Te gebruiken voor het toevoegen van content aan een Respec document.
editors	link	Array met per naam de properties 'name', 'company' en 'companyURL'.	Lokaal			1 of meerdere beschrijvingen van personen die hebben bijgedragen aan de tot stand koming van de Respec documentatie. Editors hebben de voorkeur boven authors.
formerEditors	link	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	link	URI of een array van de properties 'repoURL' en 'branch'.	Lokaal			Gebruikt voor het genereren van de link naar de GitHub repository van de Respec documentatie. worden met <ul style="list-style-type: none"> • een url naar een GitHub repository • het deel van de url van een GitHub repository dat komt na 'https://github.com/' • en set van properties bestaande uit <ul style="list-style-type: none"> ◦ repoURL: Een van bovenstaande ◦ branch: de branch waarin de documentatie ook issues staan opgeslagen. <p>Wordt tevens gebruikt voor het genereren van de link naar de GitHub pages van de laatste versie van de Respec documentatie op basis van de hier gedefinieerde link voor de GitHub repository. De GitHub Pages link automatisch bepaald.</p>
labelColor	link	Hexadecimale colorcode.	Globaal			Definieert de bij de in 'LocalizationStrings' gedefinieerde waarden horende kleuren. Mag niet Lokaal overruled worden.
latestVersion	link	Combinatie van strings en	Globaal			Url van de laatst gepubliceerde versie. Wordt opgebouwd m.b.v. andere gedefinieerde configuration properties en '/' tekens.

Property	Link	Type	Gloobaal/Lokaal	Vaste globale waarde of default waarde	Gerelateerd property	Beschrijving
		configuration propertynamen.				Mag niet lokaal overruled worden.
license	link	enumeration	Gloobaal	eupl		<p>Definieert het licentietype dat van toepassing is op de specificatie. VNG-R hanteert de 'EUPL' als standaard. Gewenst kan ook gekozen worden voor 'CC-BY-ND'. Toegestane waarden zijn 'cc-by-nd'. Wordt gebruikt om licentie-bijbehorende link in het document te geven. Licentie types en het bijbehorende logo worden opgenomen in de globale property 'licenses' worden gebruikt.</p> <p>Kan lokaal overruled worden.</p>
licenses	link	Array met per licentiecode de properties 'name', 'short', 'url' en 'image'.	Gloobaal			<p>Definieert middels een array van configuratie gebruiken soorten licenties waarnaar kan worden verwezen in de configuratie.</p> <p>Mag waarschijnlijk lokaal overruled worden.</p>
localBiblio	link	1 of meerdere objecten met set van properties.	?	n.t.b.		<p>Hiermee kan een lijst met referenties in de 'Referenties' worden gegenereerd. Die bevatten meta-informatie (bijv. auteur, jaar en status) en links naar de betreffende referenties. De referenties worden echt opgenomen in dat hoofdstuk als er in het document naar verwezen wordt middels de volgende syntax: <code>[[Referentienaam] [Referentie]]</code> geldt voor zowel html als markdown documenten.</p> <p>Indien een link wordt opgenomen in een documentdeel zal de referentie terecht in de subparagraaf 'Normatieve referenties' opgenomen in een informatief document. Deze in de subparagraaf 'Informatieve referenties' terecht.</p> <p>Gereferencieerd kan worden aan specificaties die zijn in de SpecRef database https://www.specref.org/ (zie ook https://github.com/tobie/specref) deze property gedefinieerde referentie de inhoud van de localBiblio property is opgenomen in https://github.com/tobie/specref/blob/main/reference.json.</p> <p>Deze property kan zowel lokaal als globaal geconfigureerd worden maar het is niet mogelijk om de property zowel lokaal als globaal te definiëren. Het bevat ze andere inhoud. Hier is wel ingediend bij Logius (https://github.com/Logius/standaarden/respec/issues/52).</p>
localizationStrings	link	Array van properties per taalcode	Gloobaal			Bevat voor een aantal doelen en talen van gebruikte codes en de daarbij horende

Property	Link	Type	Globaal/Lokaal	Vaste globale waarde of default waarde	Gerelateerd property	Beschrijving
logos	link	Array per logo van de properties 'src', 'alt', 'id', 'height' en 'url'.	Globaal	VNG Realisatie logo		definieert de src, alternate tekst, url en linksboven in de specificatie te plaatsen. Mag niet Lokaal overruled worden.
maxTocLevel	link	Integer	Lokaal	0		Bepaald het aantal niveau's dat maximaal opgenomen in de inhoudsopgave van het document.
nl_organisationName	link	String	Globaal	VNG Realisatie		Wordt gebruikt om de subtitel en het veld linksboven te genereren. Kan lokaal overruled worden.
nl_organisationPublishURL	link	URL	Globaal	?		Wordt gebruikt voor het genereren van GitHub pages van de huidige, de vorig gepubliceerde versie. Een link die leidt naar document in de GitHub Pages interface 'publicatie' GitHub repository. De laatste versie is overigens wat anders dan de laatste versie van de inhoud. Kan worden gebruikt in de properties 'thisVersion' en 'prevVersion'. Mag niet lokaal overruled worden.
nl_organisationStylesURL	link	URL	Globaal			Definieert de lokatie waar het te gebruiken bestand staat excl. dat bestand zelf. Mag niet Lokaal overruled worden.
noTOC	link	boolean	Lokaal	false		Bepaald of er links van de inhoud een inhoudsopgave gegenereerd moet worden.

Property	Link	Type	Gloaal/Lokaal	Vaste globale waarde of default waarde	Gerelateerd property	Beschrijving
otherLinks	link	Array van properties	Lokaal			Genereert een sectie in de header van het document met als titel de key van deze inhoud een of meerdere links.
postProcess	link		?	?		Bevat een of meer JavaScript functies die achtereenvolgend opgestart worden na de generatie met generatie van het Respec document.
previousMaturity	link	enumeration	Lokaal			Status van de voorgaande versie. Kan evt. worden gebruikt in de property 'previousPublishDate' maar wordt daar nu niet gebruikt.
previousPublishDate	link	Datum in het formaat YYYY-MM-DD	Lokaal			Publicatiedatum van de voorgaande versie. Kan evt. worden gebruikt in de property 'previousPublishVersion' maar wordt daar nu niet gebruikt.
previousPublishVersion	link	SemVer notatie	Lokaal			Versienummer van de voorgaande versie in SemVer notatie (https://semver.org/lang/nl/). Wordt gebruikt in de property 'prevVersion'.
prevVersion	link	Combinatie van strings en propertynamen	?			Url van de voorgaande versie. Wordt opgebouwd m.b.v. andere gedefinieerde configuration properties en '/' tekens. Indien deze configuration property niet gedefinieerd is, wordt de gerelateerde rubriek in de header niet aangemaakt.
pubDomain	link	enumeration	Lokaal			Definieert het publicatie domein van de publicatie. Momenteel heeft op dit moment de waarde 'cim'. Wordt nu gebruikt in de properties 'lastVersion' en 'thisVersion' en 'prevVersion'.

Property	Link	Type	Globaal/Lokaal	Vaste globale waarde of default waarde	Gerelateerd property	Beschrijving
publishDate	link	Datum in het formaat YYYY-MM-DD	Lokaal			<p>Publicatiedatum van de huidige versie.</p> <p>Kan evt. worden gebruikt in de property 'thisVersion'.</p>
publishVersion	link	SemVer notatie	Lokaal			<p>Versienummer van de huidige versie in (https://semver.org/lang/nl/).</p> <p>Wordt gebruikt in de properties 'lastVersion' en 'thisVersion'.</p>
shortName	link	String	Lokaal			<p>Korte naam (mnemonic) van de specificatie.</p> <p>Wordt gebruikt in de properties 'lastVersion' en 'prevVersion'.</p>
sotdText	link	Array van properties per taalcode.	Globaal			<p>Bevat voor een aantal 'specStatus'sen de gebruikte codes en de daarbij horende beschrijvingen.</p> <p>Mag niet lokaal overruled worden.</p>
specStatus	link	enumeration	Lokaal			<p>Definieert de status van de specificatie om de subtitel en het vertikale label line genereren. Bepaald ook de kleur van de line in de lokale configuratie gedefinieerd te worden.</p> <p>De kleuren voor de VNG-R statussen komen gedefinieerd in de globale optie 'labelColors'.</p> <p>Kan evt. ook worden gebruikt in de properties 'latestVersion', 'thisVersion' en 'prevVersion'.</p>
spectype	link	enumeration	Lokaal			<p>Definieert het type van de specificatie om de subtitel en het vertikale label line genereren. In het template heeft dit de waarde 'VNG-R Respec'.</p> <p>Aangezien we bij VNG-R Respec veelal om Informatiemodellen mee te publiceren.</p> <p>Kan evt. ook worden gebruikt in de properties 'latestVersion', 'thisVersion' en 'prevVersion'.</p>
subtitle	link	String	Lokaal	n.v.t.		<p>String die als subtitel van de titel van het document dient. Wordt geplaatst boven de gegevens waarin de organisatienaam, documenttype en versiedatum worden gebruikt.</p>

Property	<u>Link</u>	Type	Globaal/Lokaal	Vaste globale waarde of default waarde	Gerelateerd property	Beschrijving
testSuiteURI	link	URL	Lokaal	n.v.t.		Genereert een sectie in de header van het document met als titel 'Test suite' en verwijst naar een testsuite. Wellicht te gebruiken met Testplatform maar alleen als we Respec gebruiken voor de API's.
thisVersion	link	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.</p> <p>Het is verplicht deze property van een versie te voorzien en deze mag niet Lokaal overruled worden.</p>
title	-		Lokaal			De titel van de betreffende specificatie
useLabel	link	boolean	Globaal	true		<p>Bepaald of het verticale label aan de link van de inhoudsopgave gegenereerd moet worden.</p> <p>Kan lokaal overruled worden.</p>
useLogo	link	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	link	URL				Beschrijft de url waar de draft van de specificatie worden bekeken. Er wordt geadviseerd niet te publiceren.

3. ReSpec template instructies

Dit onderdeel is niet normatief.

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

Gebruik de knop [Use this template](#) om aan de slag te gaan. Dit maakt een kopie van de template in uw eigen GitHub repository die dan aangepast en uitgebreid kan 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. <-- Deze heb ik deels nog niet werkende. Een beschrijving van deze acties vindt je onderaan dit hoofdstuk.

3.1 Vereiste voor gebruik

- Kennis van git/github
- Kennis van markdown en/of HTML
- Kennis van de vorm van een Javascript object
- Een webserver om de documentatie te hosten

3.2 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](#).

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. Bekijk de [Logius ReSpec wiki](#) voor meer informatie over de configuratie opties. 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 `VNG-R-WOW.md` staat beschreven hoe je de inhoud van het Respec document naar wens kunt aanpassen.

3.3 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 alle 3 de acties in het kort hieronder. Aangezien we de laatste actie nog niet werkende hebben wordt deze voorlopig nog handmatig uitgevoerd, ook dat beschrijven we. De resultaten van deze acties zijn te vinden in het tabblad **Actions** in de GitHub repository.

3.3.1 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 laatste 2 worden in de laatste actie gebruikt om te publiceren.

De PDF-versie wordt alleen aangemaakt indien `alternateFormats` in de `document_config` (`config.js`) als volgt geconfigureerd staat:

```
alternateFormats: [  
  {  
    label: "pdf",  
    uri: "template.pdf",  
  },  
]
```

De waarde van de property `uri` moet natuurlijk naar wens worden aangepast.

3.3.2 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 evenals de structuur van de publicatie GitHub repository. De definitieve url kan indien gewenst nog andere onderdelen bevatten zoals bijv. `publishDate`, `previousPublishDate`, `specStatus`, `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://melsk-r.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 aan te passen.
- `latestVersion`
Wordt opgebouwd a.d.h.v. een aantal andere configuratie properties uit zowel de `organisation_config` als de

document_config en enkele statische waardes. 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 statische waardes. 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 statische waardes. 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.
- 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.
- 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.

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

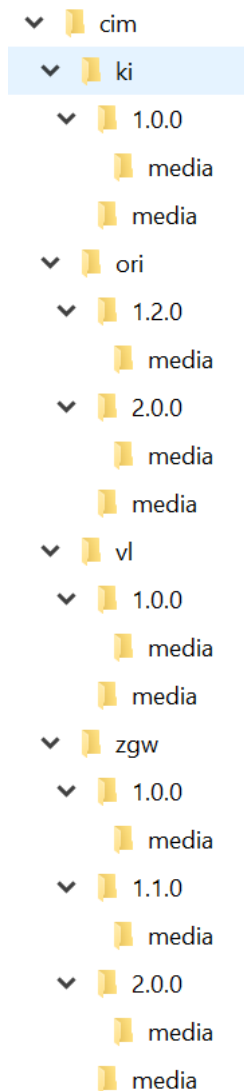
3.4 Publiceren van documenten

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 alle Respec documenten gebeurt via de GitHub repository 'publicatie' waarin de volgende consistente structuur vereist is:

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 2

De statische html en pdf worden vanuit de GitHub repository waarin ze gegenereerd zijn gekopieerd naar de gewenste folders in de 'publicatie' repository. Als het goed is heb je de gewenste folders in een van de voorgaande stappen al aangemaakt.

4. Niet-normatieve deel

Dit onderdeel is niet normatief.

Bijvoorbeeld een introductie.

NOOT: index

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

5. Meer inhoud

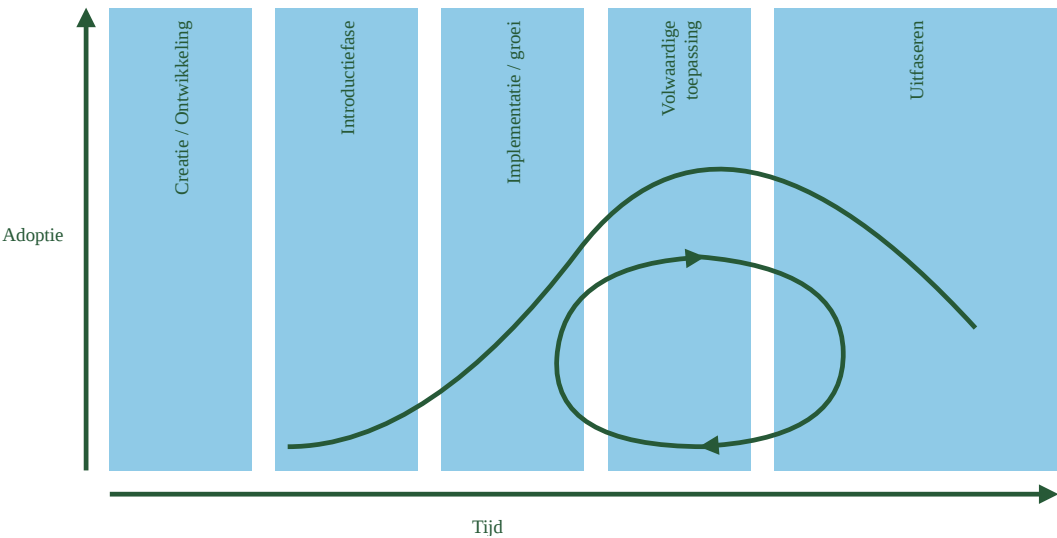
Dit onderdeel is niet normatief.

5.1 Definities

Definitie: Een definitie is een beschrijving van een woord. Een ander woord voor definitie is betekenis of beschrijving.

5.2 Afbeeldingen

Afbeeldingen krijgen een nummer en vermelding in de figurenlijst [8. Lijst met figuren](#).



Figuur 3 Onderschrift

5.3 Referenties

Referentie uit lokale lijst [\[SemVer\]](#). Lijst staat in `organisation-config.js`. Alleen referenties die in de tekst voorkomen worden getoond.

We gebruiken een [definitie](#) om een woord te omschrijven.

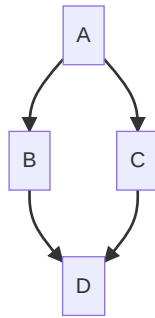
5.4 Optioneel

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>
```

6. Mermaid diagram

Dit onderdeel is niet normatief.



Figuur 4 Mermaid voorbeeld

7. 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.

8. Lijst met figuren

Figuur 1

Figuur 2

Figuur 3 Onderschrift

Figuur 4 Mermaid voorbeeld

A. Index

A.1 Begrippen gedefinieerd door deze specificatie

Definitie §5.1

A.2 Begrippen gedefinieerd door verwijzing

B. Referenties

B.1 Normatieve referenties

[Ref]

Reference not found.

B.2 Informatieve referenties

[SemVer]

Semantic Versioning 2.0.0. T. Preston-Werner. June 2013. URL: <https://semver.org>