

Actiereeksen: If-then-constructies in Freemarker

Auteur: Rogier Visser (Laansloot IT)

Datum: 7 maart 2020

Naam: Laansloot-Actiereeksen-Freemarker-if-then.pdf

Versie: 0.2

Doelgroep: TOPdesk-applicatiebeheerders

Introductie

Actiereeksen binnen TOPdesk worden nog krachtiger wanneer je gebruik maakt van Freemarker. Met Freemarker kun je een extra laag logica inbouwen, waardoor je eindeloze mogelijkheden krijgt bij het maken van actiereeksen. Enige programmeerervaring is overigens wel gewenst.

Deze handleiding sluit aan bij het actiereeks-archief op het TOPdesk-forum van Laansloot IT. Het actiereeks-archief is een naslagwerk van actiereeksen waarmee je terugkerende taken binnen TOPdesk kunt automatiseren.

Je vindt het actiereeks-archief op de volgende locatie:

<https://topdeskforum.laansloot.nl/c/actiereeks-archief>

Disclaimer

Deze handleiding is puur van informatieve aard. Laansloot IT kan niet aansprakelijk gesteld worden voor enigerlei schade die zou kunnen voortkomen uit het volgen van deze handleiding. Laansloot IT geeft verder geen garantie dat actiereeksen doen wat ze moeten doen, en Laansloot IT geeft ook geen support hierop.

Heb je behoefte aan hulp? Stel dan je vraag op het TOPdesk-forum:

<https://topdeskforum.laansloot.nl/>

Copyright en licentie

Copyright op de teksten in deze handleiding ligt bij Laansloot IT. Copyright op de screenshots van TOPdesk ligt bij TOPdesk. Deze handleiding wordt gepubliceerd onder de licentie Creative Commons BY-NC-ND 4.0:

<https://creativecommons.org/licenses/by-nc-nd/4.0/deed.nl>

Waar gaat dit document precies over?

In de documenten "Laansloot-Actiereeksen-GET-call.pdf" en "Laansloot-Actiereeksen-werken-met-stappen.pdf" is uitgelegd hoe je een GET-call kunt doen op TOPdesk, en hoe je de responsdata daarvan kunt gebruiken in vervolgstappen binnen de actiereeks:

<https://www.laansloot.nl/forum/ActArch/Laansloot-Actiereeksen-GET-call.pdf>

<https://www.laansloot.nl/forum/ActArch/Laansloot-Actiereeksen-werken-met-stappen.pdf>

In dit document gaan we een stapje verder, en kijken we hoe we logische regels kunnen opstellen binnen een actiereeks. Dit doen we met Freemarker.

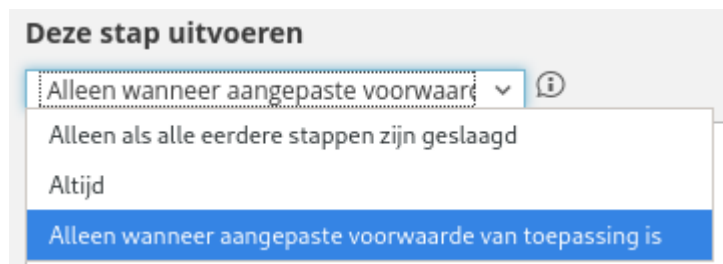
Freemarker maakt het mogelijk om actiereeksen nog specifiek te maken, zodat ze in nog meer situaties taken kunnen automatiseren. Enige programmeerervaring helpt wel bij het werken met Freemarker.

Inhoud

- p.3: If-then-constructie in "Aangepaste voorwaarde"
- p.9: If-then-constructie in de JSON-body
- p.11: Uitgebreide If-then-constructies
- p.13: ?contains-functie

If-then-constructie in "Aangepaste voorwaarde"

Bij het maken van stappen binnen een actiereeks zie je linksonder het veld "Deze stap uitvoeren". In dit zoeklijstje staat de optie "Alleen wanneer aangepaste voorwaarde van toepassing is":



Met deze optie kun je regelen wanneer de stap uitgevoerd of overgeslagen moet worden. Hier kun je bijvoorbeeld instellen dat de volgende actie-update verschijnt, wanneer de categorie gelijk is aan "Huisvesting":



AUTOMATION,

De categorie is Huisvesting

Is de categorie niet gelijk aan "Huisvesting"? Dan moet deze stap in de actiereeks niet worden uitgevoerd. Bovenstaande actie-update verschijnt in dat geval dan niet. Dit is een zogeheten If-then-constructie.

De actiereeks die hiervoor zorgt is als volgt opgebouwd:

1. een GET-call om de data van de melding op te halen
2. een PUT-call om de actie-update te plaatsen
 - Aangepaste voorwaarde: alleen uitvoeren wanneer de categorie van de melding (in de responsedata van stap 1) gelijk is aan "Huisvesting"

De syntax zorgt voor de aangepaste voorwaarde:

```
<#if _responses.getincident.body.category.name == "Huisvesting">true</#if>
```

Er is ook een alternatieve syntax. In de rest van dit document zet ik de twee verschillende manieren direct onder elkaar.

```
${_responses.getincident.body.category.name == "Huisvesting"}
```

Zie de volgende pagina voor een screenshot van de gehele actiereeks.

Actiereeks met If-then-constructie

Stap 1 ↑ ↓ Verwijderen

Naam

HTTP-methode

URL

Headers

+

Deze stap uitvoeren

Stap 2 ↑ ↓ Verwijderen

Naam

HTTP-methode

URL

Headers

+

Body

☒ JSON-escaping toepassen op ingevoegde waarden

```
{  
  "action" : "De categorie is Huisvesting"  
}
```

Deze stap uitvoeren

```
<#if _responses.getincident.body.category.name == "Huisvesting">true</#if>
```

Logische functies

De actiereeks op de vorige pagina controleert of een bepaald element gelijk is aan een bepaalde waarde (element `_responses.getincident.body.category.name`, waarde Huisvesting). Deze "Gelijk aan" functie roep je aan met de syntax `==`.

De belangrijkste functies, met syntax, op een rij:

Gelijk aan: `==`

```
<#if _responses.getincident.body.category.name == "Huisvesting">true</#if>
${_responses.getincident.body.category.name == "Huisvesting"}
```

Ongelijk aan: `!=`

```
<#if _responses.getincident.body.category.name != "Huisvesting">true</#if>
${_responses.getincident.body.category.name != "Huisvesting"}
```

Gelijk aan A of B: `||`

```
<#if _responses.getincident.body.category.name == "Huisvesting" ||
_responses.getincident.body.category.name == "Diensten">true</#if>
${_responses.getincident.body.category.name == "Huisvesting" ||
_responses.getincident.body.category.name == "Diensten"}
```

Gelijk aan A en B: `&&`

```
<#if _responses.getincident.body.category.name == "Huisvesting" &&
_responses.getincident.body.subcategory.name == "Lift installaties">true</#if>
${_responses.getincident.body.category.name == "Huisvesting" &&
_responses.getincident.body.subcategory.name == "Lift installaties"}
```

Als element bestaat: `??`

```
<#if _responses.getincident.body.category.name??>true</#if>
${_responses.getincident.body.category.name??}
```

Als element bestaat en is ingevuld: `?has_content`

```
<#if _responses.getincident.body.category.name?has_content>true</#if>
${_responses.getincident.body.category.name?has_content}
```

EN-operator && en OF-operator || combineren met haakjes

De syntax van Freemarker maakt het ook mogelijk om EN && en OF || te combineren. Stel dat de stap alleen uitgevoerd moet worden wanneer:

- Categorie is gelijk aan "Huisvesting" EN
- Subcategorie is gelijk aan
 - "Lift Installaties" OF
 - "Sanitair"

Dit kun je instellen door de twee OF-voorwaardes te bundelen binnen haakjes:

```
<#if _responses.getincident.body.category.name == "Huisvesting" &&  
    (_responses.getincident.body.subcategory.name == "Lift installaties" ||  
    _responses.getincident.body.subcategory.name == "Sanitair")>true</#if>  
${  
    _responses.getincident.body.category.name == "Huisvesting" &&  
    (_responses.getincident.body.subcategory.name == "Lift installaties" ||  
    _responses.getincident.body.subcategory.name == "Sanitair")}
```

Deze functionaliteit is waardevol, want dit is niet goed mogelijk wanneer je vergelijkbare logica probeert te bouwen in de gebeurtenisverkenner:

Voorwaarden

Toepassen



Gebeurtenis treedt op als



Alle voorwaarden zijn van toepassing (AN



Kenmerk

Waarde

Categorie is gelijk aan



Huisvesting



[Verwijderen](#)

en

Subcategorie is gelijk aan



Huisvesting - Lift installaties



[Verwijderen](#)

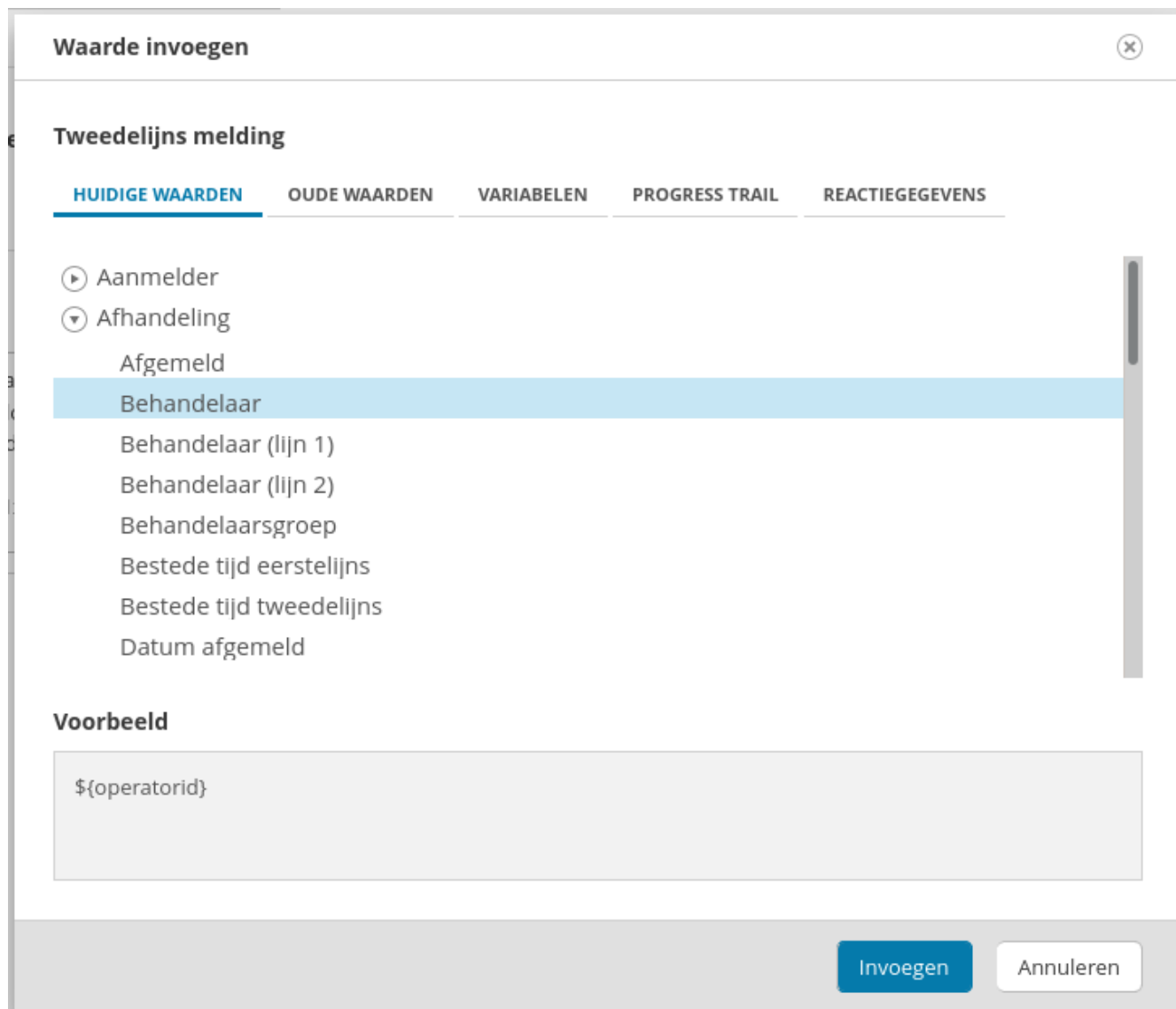
[Toevoegen](#)

Hier loop je vast, omdat je niet Subcategorie A *OF* Subcategorie B kunt kiezen, in combinatie met een bepaalde categorie. De workaround binnen de gebeurtenisverkenner is om hier twee losse gebeurtenissen voor aan te maken, voor beide subcategorieën.

TOPdesk-velden in aangepaste voorwaarde

Het invoerveld voor "Aangepaste voorwaarde" heeft ook de mogelijkheid om direct een waarde uit de TOPdesk-database in te voeren, met het knopje "[] -Waarde invoegen- rechtsboven.

Je kunt bijvoorbeeld de code `${operatorid}` gebruiken om direct de behandelaar van de kaart op te zoeken:



Waarde invoegen

Tweedelijns melding

HUIDIGE WAARDEN OUDE WAARDEN VARIABELEN PROGRESS TRAIL REACTIEGEGEVENS

- ▶ Aanmelder
- ▼ Afhandeling
 - Afgemeld
 - Behandelaar**
 - Behandelaar (lijn 1)
 - Behandelaar (lijn 2)
 - Behandelaarsgroep
 - Bestede tijd eerstelijns
 - Bestede tijd tweedelijns
 - Datum afgemeld

Voorbeeld

`${operatorid}`

Invoegen **Annuleren**

Maar let op! Met deze variabelen roep je de UNID op, en niet de naam van de kaart.

Toch kun je ook met de UNID in een aantal scenarios uit de voeten. Je kunt bijvoorbeeld controleren of er een behandelaar is ingevuld.

Let trouwens op de syntax! Binnen de accolades {} of haken <> gebruik je niet nogmaals de accolades. Dus daarom wordt \${operatorid} dan operatorid in onderstaande voorbeelden.

Als veld is ingevuld: ?has_content

```
<#if operatorid?has_content>true</#if>
${operatorid?has_content}
```

Daarnaast kun je ook logische regels opstellen als je de UNID al weet. In het document "Laansloot-Actiereeksen-tips.pdf" staat hoe je de UNID van een kaart kunt achterhalen:

<https://www.laansloot.nl/forum/ActArch/Laansloot-Actiereeksen-tips.pdf>

Gelijk aan: ==

```
<#if operatorid == "a74e5723-6143-564d-836f-8965c6099c38">true</#if>
${operatorid == "a74e5723-6143-564d-836f-8965c6099c38"}
```

Ongelijk aan: !=

```
<#if operatorid != "a74e5723-6143-564d-836f-8965c6099c38">true</#if>
${operatorid != "a74e5723-6143-564d-836f-8965c6099c38"}
```

Verdere syntax is gelijk aan wat eerder is uitgelegd, met de EN-operator &&, de OF-operator ||, en de haakjes.

Bronnen combineren

Je kunt ook de responsedata uit een eerdere stap combineren met een variabele uit de TOPdesk-database, binnen de aangepaste voorwaarde:

```
<#if _responses.getincident.body.category.name == "Huisvesting" &&
operatorid == "a74e5723-6143-564d-836f-8965c6099c38">true</#if>
${
_responses.getincident.body.category.name == "Huisvesting" &&
operatorid == "a74e5723-6143-564d-836f-8965c6099c38"}
```


If-then-constructie in de JSON-body

Met If-then-constructies in de JSON-body kun je heel fijnmazig instellen welke data in de body terechtkomt. Hieronder is als voorbeeld een actie-update met de categorie en de subcategorie:



AUTOMATION,

De categorie is Huisvesting en de subcategorie is Lift installaties

Maar als de subcategorie niet is ingevuld op de kaart, dan willen we alleen de naam van de categorie:



AUTOMATION,

De categorie is Huisvesting

Dit kan met een If-then-constructie in de JSON-body, met de optie `?has_content`:

Body

☒ JSON-escaping toepassen op ingevoegde waarden ⓘ ⓘ

```
{
  "action" : "De categorie is ${_responses.getincident.body.category.name}<#if
_responses.getincident.body.subcategory?has_content> en de subcategorie is
${_responses.getincident.body.subcategory.name}</#if>"
}
```

De gele gehighlightte code is de If-then-constructie, en de blauw gehighlightte code is de waarde die we invoegen wanneer de expressie Waar is.

```
}
"action" : "De categorie is ${_responses.getincident.body.category.name}<#if
_responses.getincident.body.subcategory?has_content> en de subcategorie is
${_responses.getincident.body.subcategory.name}</#if>"
}
```

Je kunt ook hele elementen binnen- of buitensluiten met een If-then-constructie. In onderstaande JSON-body wordt een behandelaar ingevuld, maar alleen wanneer de subcategorie gelijk is aan "Sanitair":

Body

☒ JSON-escaping toepassen op ingevoegde waarden ⓘ ⓘ

```
{
  "action" : "De categorie is ${_responses.getincident.body.category.name}"<#if
_responses.getincident.body.subcategory.name == "Sanitair">,
  "operator" : { "id" : "a74e5723-6143-564d-836f-8965c6099c38" }</#if>
}
```

```
{
  "action" : "De categorie is ${_responses.getincident.body.category.name}"<#if
_responses.getincident.body.subcategory.name == "Sanitair">,
  "operator" : { "id" : "a74e5723-6143-564d-836f-8965c6099c38" }</#if>
}
```

Is de subcategorie gelijk aan "Sanitair", dan wordt de volgende JSON gegenereerd:

```
Request body: {
  "action" : "De categorie is Huisvesting",
  "operator" : { "id" : "a74e5723-6143-564d-836f-8965c6099c38" }
}
```

Is de subcategorie ongelijk aan "Sanitair", dan is de JSON korter:

```
Request body: {
  "action" : "De categorie is Huisvesting"
}
```

Let op de komma!

De komma die als eerste blauw gehighlight is in de screenshot, is heel belangrijk. Staat die komma er niet, dan is de JSON niet geldig en geeft TOPdesk een error:

```
Request body: {
  "action" : "De categorie is Huisvesting"
  "operator" : { "id" : "a74e5723-6143-564d-836f-8965c6099c38" }
}
```

[2020-02-15T21:08:20.712918Z] Response status: 400 Bad Request

Response headers: {Date=[Sat, 15 Feb 2020 21:08:20 GMT], X-Content-Type-Options=[nosniff], X-XSS-Protection=[1; mode=block], Cache-Control=[no-store], Content-Type=[application/json; charset=utf-8], Content-Length=[100]}

Response body: [{"message":"Request cannot be parsed."}]

Uitgebreide If-then-constructies

Onderstaande JSON-body maakt gebruik van drie verschillende If-then-constructies die door elkaar heen lopen. Dit zijn geneste If-then-constructies:

JSON-body

```
{
  "action": "De categorie is ${_responses.getincident.body.category.name}"
  <#if _responses.getincident.body.category.name == "Huisvesting">
  <#if _responses.getincident.body.subcategory.name == "Lift installaties">,
  "subcategory": { "name": "Sanitair" } </#if>
  <#if _responses.getincident.body.subcategory.name == "Sanitair">,
  "subcategory": { "name": "Lift installaties" } </#if> </#if>
}
```

Hier staat eigenlijk:

1. Plaats een actie-update
2. Is de categorie gelijk aan "Huisvesting"?
 - Is de subcategorie gelijk aan "Lift installaties"?
 - Verander dan de subcategorie naar "Sanitair"
 - Is de subcategorie gelijk aan "Sanitair"?
 - Verander dan de subcategorie naar "Lift installaties"
3. Is de categorie ongelijk aan "Huisvesting"? Verander dan de subcategorie *niet*.

Met geneste If-then-constructies kun je dus heel precies regelen wat je aanpast in TOPdesk en onder welke voorwaarden.

If-elseif

In bovenstaande code gebruiken we drie If-then-constructies om de subcategorie aan te wijzen. Hier kun je ook twee If-then-constructies van maken, door gebruik te maken van de optie <#if><#elseif></#if> :

JSON-body

```
{
  "action": "De categorie is ${_responses.getincident.body.category.name}"
  <#if _responses.getincident.body.category.name == "Huisvesting">
  <#if _responses.getincident.body.subcategory.name == "Lift installaties">,
  "subcategory": { "name": "Sanitair" }
  <#elseif _responses.getincident.body.subcategory.name == "Sanitair">,
  "subcategory": { "name": "Lift installaties" } </#if> </#if>
}
```

Dit is ook een geneste If-then-constructie, en de uitwerking is precies gelijk als in de eerdere variant. Maar als je goed kijkt zie je dat er maar twee <#if>-tags zijn.

If-else

Iets vergelijkbaars is aan de hand in de volgende JSON. Hier is echter de `<#elseif>`-tag vervangen door een `<#else>`-tag:

JSON-body

```
{
  "action": "De categorie is ${_responses.getincident.body.category.name}"
  <#if _responses.getincident.body.category.name == "Huisvesting">
  <#if _responses.getincident.body.subcategory.name == "Lift installaties">,
  "subcategory": { "name": "Sanitair" }
  <#else>,
  "subcategory": { "name": "Lift installaties" } </#if> </#if>
}
```

Deze actiereeks verandert de subcategorie **altijd** in "Lift installaties". In de vorige twee voorbeelden deed TOPdesk dat alleen in het geval van subcategorie "Sanitair".

If-elseif-else

Als je te maken hebt met drie of meerdere variabelen op basis waarvan je vervolgstappen wilt uitzetten, dan gebruik je `<#if>`, `<#elseif>` en `<#else>`. In onderstaand voorbeeld is sprake van drie subcategorieën.

JSON-body

```
{
  "action": "De categorie is ${_responses.getincident.body.category.name}"
  <#if _responses.getincident.body.category.name == "Huisvesting">
  <#if _responses.getincident.body.subcategory.name == "Lift installaties">,
  "subcategory": { "name": "Sanitair" }
  <#elseif _responses.getincident.body.subcategory.name == "Sanitair">,
  "subcategory": { "name": "Lift installaties" }
  <#else>,
  "subcategory": { "name": "Verlichting" } </#if> </#if>
}
```

De logica in deze actiereeks is een combinatie van de vorige voorbeelden:

1. Plaats een actie-update
2. Is de categorie gelijk aan "Huisvesting"?
 - Is de subcategorie gelijk aan "Lift installaties"?
 - Verander dan de subcategorie naar "Sanitair"
 - Is de subcategorie gelijk aan "Sanitair"?
 - Verander dan de subcategorie naar "Lift installaties"
 - Is de subcategorie niet "Lift installaties" en ook niet "Sanitair"?
 - Verander dan de subcategorie naar "Verlichting"
3. Is de categorie ongelijk aan "Huisvesting"? Verander dan de subcategorie *niet*.

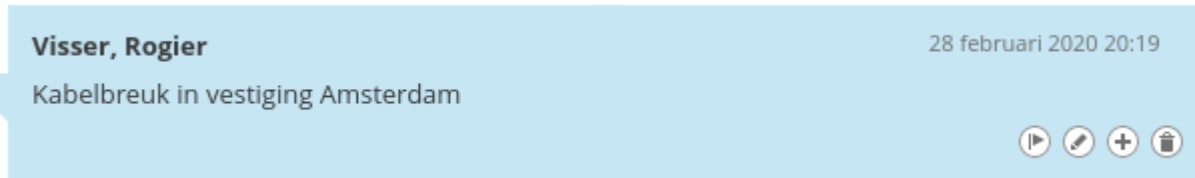
Wil je meer dan 3 variabelen opgeven? Dan kun je meerdere `<#elseif>`-expressies inbouwen, op dezelfde manier als hierboven.

?contains-functie

Het laatste kopje in dit hoofdstuk gaat over de functie ?contains. Met deze functie kun je sleutelwoorden definiëren en daarop functionaliteit bouwen.

- Is de zoekterm gevonden? Dan voert TOPdesk de stap in de actiereeks uit
- Is de zoekterm niet gevonden? Dan voert TOPdesk de stap in de actiereeks *niet* uit

Onderstaande melding gaat over een kabelbreuk:



"Kabelbreuk" is de zoekterm die we opgeven in de "Aangepaste voorwaarde":

Aangepaste voorwaarde

```
<#if _responses.getincident.body.request?contains("Kabelbreuk")>true</#if>
```

Als de zoekterm is gevonden in het verzoekveld, dan wordt de volgende JSON-body verstuurd:

JSON-body

```
{  
  "action": "Het sleutelwoord is gevonden!"  
}
```

Resultaat:



In dit voorbeeld wordt een actie-update geplaatst, maar op dezelfde manier kun je ook een melding routeren naar een bepaalde behandelaarsgroep, of een prioriteit invullen, op basis van de zoekterm. De mogelijkheden zijn eindeloos.

NB1: de ?contains-functie zoekt altijd hoofdlettergevoelig.