Environment variabelen bijwerken

# Environment variabele toevoegen aan helm

Om een helm variabele toe te kunnen voegen aan de online omgeving moet deze toegevoegd worden aan de helm file. Dit is een klein taakje, maar kent wel een aantal valkuilen.

## Stap 1: voeg de waarde toe aan values.yaml

Om de waarde toe te voegen moeten we deze definiëren in het bestand helm/values.yamlAfbeelding met tekst

Automatisch gegenereerde beschrijving

Onder config kunnen we hier een variabele (value) definiëren met wel of geen standaardwaarde. In dit geval voegen we de waarde toe zonder standaardwaarde:

Afbeelding met tekst

Automatisch gegenereerde beschrijving

## Stap 2: voeg de waarde toe aan env.js

Om een waarde toe te voegen moet deze aan de versie van env.js worden toegevoegd die wordt gedeployed vanuit helm. Deze file wordt beschreven in het bestand helm/kiss-frontend/templates/configmap.yaml Afbeelding met tekst

Automatisch gegenereerde beschrijving

Vervolgens kan hier de value worden aangeroepen die we in de vorige stap hebben toegevoegd. Dit doen we door middel van de volgende syntax:

{{ .Values.*yaml.path.to*.*value* }}

Afbeelding met tekst

Automatisch gegenereerde beschrijving

## Stap 3: change publiceren

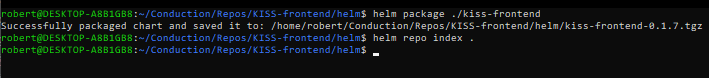
Om deze change vervolgens te kunnen gebruiken moeten er een paar stappen worden gezet. Ten eerste in chart.yaml, waar de waarden version en appVersion een hogere minor versie moeten krijgen (dit versienummer heb je verderop weer nodig).

Daarna wordt de change gepackaged met het volgende commando in de map helm:

helm package ./kiss-frontend

waarna een nieuwe package wordt toegevoegd met de naam kiss-frontend-*version*, hierna moet de repository index worden bijgewerkt met het volgende commando in dezelfde map:

helm repo index .



Daarna kan de change worden gepusht naar de repository. Voor aanpassingen op <https://kiss-dev.commonground.nu/> moet je de aanpassing mergen naar de development branch en deze deployen.

# Environment variabele instellen op omgeving

Om de helm variabele vervolgens in te stellen op de omgeving is het volgende nodig:

## Stap 1 (eenmalig): repository toevoegen aan helm

In de command line, voer het volgende commando uit:

helm repo add kiss-frontend-dev <https://raw.githubusercontent.com/ConductionNL/KISS-frontend/development/helm>

(Voor productie helm-files: vervang development door main, en haal -dev weg uit kiss-frontend-dev)

## Stap 2: Upgrade de installatie

Let wel: hiervoor is nodig dat de kubeconfig file van het cluster op de computer aanwezig is. Deze kan opgevraagd worden bij de kubernetes beheerder.

Om de bestaande installatie van de frontend te updaten gebruiken we het volgende commando. De nieuwe value kunnen we dan setten met de parameter --set *yaml.path.to.value*=*value* Zoals in dit voorbeeld --set config.gatewayBaseUrl= <https://kissdevelopment-backend.commonground.nu/api>

Vervang bij XXX door het versienummer dat je in de chart.yaml gebruikt hebt.

Wanneer er een numerieke value wordt ingevuld die als string moet worden geparsed is het van belang om de waarde tussen double quotes te zetten.

Voor het aanmaken/updaten van een arry gebruik je deze syntax :   
--set config.organizationIds[0]="999990639"

helm upgrade kiss-frontend --namespace frontend kiss-frontend-dev/kiss-frontend --reuse-values --set config.gatewayBaseUrl=https://kissdevelopment-backend.commonground.nu/api --description=<https://raw.githubusercontent.com/ConductionNL/KISS-frontend/development/helm> --version XXX

Voor gebruik van Windows zonder Linux-shell zoals bash, of wanneer de kubeconfig-file niet in de environment variabele KUBECONFIG van de shell zit kan deze worden meegegeven met de flag --kubeconfig=

Bijvoorbeeld:

helm upgrade kiss-frontend --namespace frontend kiss-frontend-dev/kiss-frontend --reuse-values --set config.gatewayBaseUri=https://kissdevelopment-backend.commonground.nu/api --description=https://raw.githubusercontent.com/ConductionNL/KISS-frontend/development/helm --kubeconfig=C:\...\kubeconfig.yaml --version XXX

Hierna kan de container worden herstart met het commando

kubectl rollout restart deployment/kiss-frontend --kubeconfig="c:\...\kubeconfig.yaml" --namespace=frontend