

JAVASCRIPT TRAINING

Case 1 - Opzet/draaien van omgeving

Omschrijving

Zojuist is verteld hoe de Todo applicatie eruit ziet. Middels deze applicatie zul je alle cases in deze training gaan uitwerken.

Het doel van deze opdracht is te wennen aan de ontwikkelomgeving waarmee de cases ontwikkeld gaan worden. Na deze opdracht weet je hoe je bestanden kunt bewerken in de ontwikkelomgeving en weet je hoe je de webserver start waarmee in latere opdrachten data vandaan is te halen. Tevens weet je hoe je de Todo applicatie benaderd en hoe de REST service er globaal uitziet.

Opdracht 1 - Configureren Virtual Machine

Zorg ervoor dat VirtualBox geïnstalleerd is. Download deze als je deze nog niet hebt van de website: <https://www.virtualbox.org/>

Als je VirtualBox hebt geopend, maak dan een nieuwe virtual machine aan. Gebruik in de wizard onderstaande waardes:

- **Name:** Javascript training
- **Type:** Linux
- **Version:** Ubuntu
- **Memory size:** 1024MB
- **Use an existing virtual hard drive file:** gebruik de file die tijdens de training wordt aangereikt.

Als de machine is aangemaakt, gebruik dan de settings om onderstaande waardes te configureren:

- **Display:** Enable 3D video acceleration
- **Network**
Attached to: Bridged Adapter
Name: de enige die er staat, anders even vragen welke je moet kiezen

Start de virtual machine op. Als het goed is kom je in een Linux desktop omgeving terecht.

Opdracht 2 - Opstarten ontwikkelomgeving

Stap 1

Eerst dient gecontroleerd te worden of je op internet komt.

1. Open de browser en controleer of je op het internet komt (hint: probeer www.google.nl)
2. [Download de source](#) voor deze opdracht

Stap 2

Op de desktop staat een icoon genaamd "WebStorm", dit is de ontwikkelomgeving die we voor de training gebruiken. Start "Webstorm" op.

Stap 3

Links zie je alle project files staan. Hier staat nu een "server" directory en een lege "opdrachten" directory, hieraan voegen we per opdracht de sources toe. Open hiervoor middels het "map" icoontje in de balk onderaan in het scherm de file browser. Navigeer naar Downloads, als het goed is zie je hier de eerder gedownloadte zip file die de sources voor deze case bevat.

Kopieer deze zip file naar de directory "Home/WebstormProjects/javascript_training/opdrachten". Klik hierna met de rechter muisknop op de zip file en kies voor "Extract here". Als je nu teruggaat naar WebStorm zie je in de "opdrachten" folder de zojuist uitgedeelte sources van deze case erbij staan.

Opdracht 3 - Starten van de server

Als REST server gebruiken we NodeJS, deze verzorgt de mogelijkheden om data op te slaan en data op te halen. Tevens fungeert deze als WebSocket server. Tijdens de cursus hoef je aan de serverkant niets te doen, echter dien je wel de server te starten en bekend te zijn met de REST structuur die we hebben opgezet.

Stap 1

Bovenin WebStorm naast de tekst "NodeJS Server" staan een "Run" en een "Debug" knop. Gebruik de "Run" knop om de server op te starten.

Stap 2

De NodeJS server draait op poort 8000. Bekijk de REST service documentatie door in de browser te navigeren naar "<http://localhost:8000/docs>"

Beantwoord onderstaande vragen:

1. Welke entiteiten kunnen opgeslagen worden?
2. Op welk pad is de opdracht benaderbaar?
3. Welke URL roep je vermoedelijk aan als je Todo "1" van gebruiker "1" wilt ophalen? Probeer dit tevens uit, en controleer of het resultaat is wat je verwacht.
4. Welke URL roep je vermoedelijk aan als je een nieuwe Todo aan gebruiker "1" wilt toevoegen? Beargumenteer je keuze.

Stap 3

Probeer aan de hand van de REST documentatie die je net in de browser hebt bekeken de applicatie te benaderen. Hint: de pagina (het path) dat je moet laden is "/opdrachten/case1/template.html". Als je deze stap succesvol uitvoert, verschijnt er een pagina waarop een gebruikersselectie is te maken in je browser.

Stap 4

Open in WebStorm het bestand "opdrachten/case1/template.html". Zoek de "select" met als id "gebruikersdropdown" en voeg hier je eigen naam aan toe, op de manier zoals er nu reeds 3 namen staan. Reload de client die je in stap 3 in je browser hebt geladen. Als het goed is zie je in de dropdown nu de door jouw toegevoegde gebruikersnaam staan.

Je hebt waarschijnlijk gemerkt dat je niet de NodeJS server hebt hoeven restarten, of in WebStorm het bestand hebt moeten opslaan. Voor aanpassingen die je gedurende deze training doet zijn beide zaken niet nodig. NodeJS zorgt ervoor dat altijd de actuele bestanden van het filesystem worden geladen en WebStorm zorgt dat na het editen van files meteen een "auto-save" wordt uitgevoerd.

Opdracht 4 - Omgaan met NPM/Grunt

Het enige wat je voor nu hoeft te weten van NPM (Node Package Manager) en Grunt, is dat deze wat zaken makkelijker maken m.b.t. de ontwikkelomgeving. Middels NPM worden dependencies geregeld. Denk herbij aan libraries voor draaien van unittests (jasmine), code checking (JSHint) en een heuse compiler (Google's closure compiler).

Als je deze libraries eenmaal hebt, kun je ermee taken uitvoeren. Denk bijvoorbeeld aan het draaien van unittests of het daadwerkelijk checken van code. Grunt biedt de mogelijkheid taken te configureren en deze af te trappen.

De twee commando's die nu en in de volgende cases van belang zijn, zijn:

- npm install
- grunt

Probeer deze commando's uit via de command line in de directory "Home/WebstormProjects/javascript_training/opdrachten/case1" en beantwoord onderstaande vragen/voer onderstaande taken uit:

1. Waartoe dient het bestand package.json?
2. Waartoe dient het bestand Gruntfile.js?
3. Welke taken werden er door Grunt uitgevoerd?
4. Welke taken zijn er door Grunt uit te voeren?
5. Probeer te achterhalen wat de andere geconfigureerde taak inhoudt
6. Pas de template.html zo aan, dat er twee elementen zijn met hetzelfde id-attribuut. Voer grunt uit en beschrijf wat er gebeurd.

PUBLISHED ON **June 30th, 2015**