

JAVASCRIPT TRAINING

Case 8 - AMD

Source

[Download de source](#) voor deze opdracht om te beginnen met de case. Pak de zip file uit zoals je in case 1 hebt geleerd. Als het goed is zie je nu in WebStorm onder de folder "opdrachten" de sources van case 8 staan.

Omschrijving

In case 3 heb je geleerd hoe je functionaliteiten groepeerd in losse javascript files. In case 6 heb je geleerd hoe je met het module pattern scoping issues kunt voorkomen. Hoewel deze methodes een stuk in de juiste richting zijn voor de opzet van je applicatie zijn er toch nog enkele problemen:

- Alle losse JS files moeten in de HTML worden opgenomen;
- Alle losse JS files worden in de browser ingeladen, ook als de code hierin nog niet wordt gebruikt

Zojuist is verteld hoe je deze problemen kunt oplossen met het Asynchronous Module Pattern (AMD). Tevens is je net uitgelegd hoe je met AMD minification voordelen kunt behalen t.o.v. een "handmatig" gestructureerde applicatie. Het doel van deze opdracht het AMD pattern op verschillende manieren toepassen.

In deze opdracht krijg je de hele source van de Todo app tot je beschikking. Deze opdracht omslaat dus meer code dan de eerdere cases, waarmee de voordelen van AMD beter tot zijn recht komen.

Opdracht 1 - Toepassen AMD

Open in je browser de URL "<http://localhost:8000/opdrachten/case8/template.html>" om de achtste case te laden. Je krijgt initieel het gebruikersselectiescherm voor je neus waar je een gebruiker kiest om door de applicatie heen te lopen. Vervolgens krijg je de todo pagina voor je neus waar je todo's kunt assignen aan andere gebruikers en todos kunt aanmaken/bewerken.

Bekijk in WebStorm de javascript bestanden die bij deze opdracht horen. Doorloop onderstaande stappen om de opdracht te voltooien.

Asynchronous Module Definition

1. Blader door de verschillende javascript sources en de js/app folders. Probeer te begrijpen wat de functionaliteiten van de verschillende files zijn. Wat valt je op over de opzet van de applicatie? Welk pattern wordt er toegepast?
2. Open vervolgens het bestand "template.html". Wat valt je op/welk nadeel kun je herkennen aan de huidige opzet?
3. Uncomment in de bestanden "template.html" en "init.js" de regels code uit waar stap 3 bij genoemd staat. In template.html zul je tevens alle script tags die onderaan de pagina staan moeten verwijderen.
4. Pas overal waar mogelijk AMD toe waar nu het module pattern wordt toegepast.

Tips:

- Binnen AMD hoeven modules niet aan een variable geassigned zoals met het module pattern wel het geval is. Het AMD framework (in dit geval RequireJS) handelt dit voor jou af.
 - Met een define statement maak je een module die binnen andere modules gebruikt kunnen worden. Modules die met een define gedefinieerd worden wordt pas uitgevoerd zodra ze nodig zijn.
 - Met require (zelfde syntax als define) kunnen modules gedefinieerd worden die direct worden uitgevoerd. Probeer zelf te bepalen wanneer je define en wanneer je require nodig hebt (hint: het overgrote deel zal bestaan uit modules die met define gemaakt zijn).
5. Refresh je browser en controleer of alles nog werkt als voorheen. Bekijk tevens de FireBug console en controleer dat er geen errors meer optreden. Bekijk tevens het "Net" tabblad in FireBug en controleer dat alle javascript bestanden worden ingeladen. Wat valt je op?

Opdracht 2 - Lazy loading met AMD

Als je opdracht 1 succesvol hebt uitgevoerd, zie je in het "Net" tabblad van FireBug dat alle javascript bestanden in de juiste volgorde worden ingeladen. De GebruikersView hoeft echter pas geladen te worden als je in het menu bovenin op "Gebruikers" klikt. De GebruikersView is een dependency van de GebruikersController. Zorg ervoor dat het inladen van deze javascript file pas plaats vindt op het moment dat je ook op "Gebruikers" klikt in het menu.

Tips:

- Het enige wat je hoeft aan te passen is Navigatie.js

Opdracht 3 - Minification met AMD

Bekijk de Gruntfile.js configuratie.

6. Welke taak denk je dat de minification met AMD uitvoert? Naar welke file denk je dat het resultaat zal worden weggeschreven?
7. Voer de taak die de minification uitvoert uit en bekijk het resultaat. Wat valt je op aan de gegenereerde outputfile?
8. Zonder de outputfile worden modules met AMD gevonden in een file die als naam gelijk is aan de modulenaam. Hoe denk je dat RequireJS de modules nu vind?

Pas het bestand template.html zo aan, dat i.p.v. het init script de initialisatie van de applicatie plaats vind via de gegenereerde output file. Controleer in de browser wat er gebeurt. Als het goed zal alleen de gegenereerde javascript file worden ingeladen met AMD, en niet meer alle losse modules.

PUBLISHED ON **June 30th, 2015**