

Frontend-tooling

Tools om een webapplicatie mee te bouwen

Agenda

- 16:00 Presentatie tooling
- 16:15 Presentatie Yeoman
- 16:30 Workshop Yeoman Todo Applicatie
- 17:45 Pauze
- 18:30 Presentatie Sass
- 18:45 Presentatie Test
- 19:00 Workshop Sass en Protractor

Welke taken zijn er?

- Automatische genereren van een applicatie
- Javascript frameworks beheren
- CSS en Javascript preprocessors gebruiken
- Wijzigingen 'live' herladen tijdens bewerken
- Testen in verschillende browsers en devices
- Code kwaliteit validatie
- Css bestanden samenvoegen in één css file.
- Javascript bestanden comprimeren in één file.
- Releases automatiseren

Tooling

Javascript Platform en Frameworks

Scaffolding

Package installers

Buildtools

Code validatie

Templating

Test tools

CSS tooling

IDE Support

Platform & Frameworks



Scaffolding

Project genereren

Yeoman



JHipster

Java stack, Angular & Bootstrap

Package installers

NPM - Nodejs package manager

Bower - frameworks/assets/utils manager



Buildtools

Grunt - taak georiënteerd

Gulp - stream georiënteerd

Brunch - Building assets

Brocoli - Asset pipeline



JSLint - Codekwaliteit

Here are some common mistakes that JavaScript Lint looks for:

- Missing semicolons at the end of a line.
- Curly braces without an *if*, *for*, *while*, etc.
- Code that is never run because of a *return*, *throw*, *continue*, or *break*.
- Case statements in a switch that do not have a *break* statement.
- Leading and trailing decimal points on a number.
- A leading zero that turns a number into octal (base 8).
- Comments within comments.
- Ambiguity whether two adjacent lines are part of the same statement.
- Statements that don't do anything.

JavaScript Lint also looks for the following less common mistakes:

- Regular expressions that are not preceded by a left parenthesis, assignment, colon, or comma.

- Statements that are separated by commas instead of semicolons.
- Use of increment (++) and decrement (--) except for simple statements such as "i++; " or "--i;".
- Use of the *void* type.
- Successive plus (e.g. x+++y) or minus (e.g. x---y) signs.
- Use of labeled *for* and *while* loops.
- *if*, *for*, *while*, etc. without curly braces. (This check is disabled by default.)

Javascript - Templating



Test

Jasmine - Behavior Driven Javascript

Mocha - Testframework

QUnit - Javascript Unittesting framework

Karma - Testrunner (test in browser/devices)

Protractor - End-to-end testing in Javascript

Selenium server



CSS & Javascript

Responsive
HTML5/CSS3.0
Modernizer
Bootstrap
Compass - Sass
Less
Coffescript
Typescript

Compass - CSS Preprocessor

Sass - Extensie van CSS3

Preprocess sass files en output CSS

Variabelen

Nesting

Partials - CSS snippets

Imports

Mixins - Groepeer CSS declaraties

Inheritance - CSS declaraties delen tussen selectors

Operators - Rekenen in CSS

LESS - CSS Preprocessor

Variabelen
Mixins
Geneste regels
Operators
Functies
Scope
Namespaces



IDE Support

Intellij - Nodejs plugin of Webstorm

Netbeans 8

Eclipse