



Marcel Birkner

 @marcelbirkner

 github.com/marcelbirkner

 marcel.birkner@codecentric.de



Bastian Krol

 @bastiankrol

 github.com/basti1302

 bastian.krol@codecentric.de

About codecentric



Big Data Nerds



Agile Ninjas



Continuous Delivery Gurus



Java Specialists



Performance Geeks



Hipster Developers

> 250 employees

And we are looking for more!



Delivery Pipeline for the Frontend

Why do we need this?

- JavaScript and CSS have come a long way in the last years.
- They are very powerful tools.
- It's not for pixel pushers anymore! (If it ever was)
- There is no app without JS & CSS

Frontend Code is Mission Critical!

- Broken JavaScript → broken app
- Broken CSS → broken layout and/or broken app
- It impacts the perceived performance drastically
- Even more so on mobile

- CI/CD best practices are often neglected for the frontend code

We can do better!

What we will cover

- Asset Optimization
 - Testing
 - The Delivery Pipeline
-
- **This is frontend only**
 - Ideal: backend & frontend in one CD pipeline







Asset Optimization

Common Problems

- Bad Code Quality
- Misconfigured Caching
- Lots of assets (JS, CSS, images) \Rightarrow Lots of HTTP requests \Rightarrow Slow

Tools

-  Grunt — Task Runner
GRUNT
-  ESLint — Static Code Analysis
- grunt-contrib-concat — Concatenation
- grunt-contrib-uglify/UglifyJS — Minification
-  SASS — CSS preprocessor
-  compass — images → base 64 data URIs
- grunt-version-assets — Versioned File Names

of HTTP Requests — Concatenation

JAVASCRIPT

```
concat: {  
  options: {  
    separator: '\n';  
  },  
  app: {  
    src: [  
      'client/js/**/*.js',  
    ],  
    dest: '<%= jsTargetDir %>/app.js'  
  }  
},
```

of HTTP Requests — Minification

JAVASCRIPT

```
uglify: {  
  app: {  
    files: {  
      '<%= jsTargetDir %>/app.min.js': [ '<%= jsTargetDir %>/app.js' ],  
    }  
  }  
},
```


of HTTP Requests — Embed Images in CSS

CSS

```
.img-foo {  
  background-image: inline-image("foo.png");  
}
```

JAVASCRIPT

```
sass: {  
  app: {  
    options: { compass: true, },  
    files: { '<%= cssTargetDir %>/master.css': '<%= cssSrcDir %>/master.scss', }  
  }  
},
```

CSS

```
.img-foo {  
  background-image:  
    url('data:image/png;base64,iVBORw0KGgoAAAANSUhEUgAAIAA...');  
}
```

- The best HTTP requests are those that do not happen in the first place
- Use HTTP caching headers:
 - Cache-Control: max-age=31536000 // ~= one year
 - Expires: Tue, 10 Oct 2015 15:45:00 GMT // optional

of HTTP Requests — Versioned File Names

JAVASCRIPT

```
versioning: {  
  options: {  
    grepFiles: [ 'static/**/*.*html', ]  
  },  
  js: {  
    src: [  
      '<%= jsTargetDir %>/app.min.js',  
      '<%= jsTargetDir %>/vendor.js',  
    ]  
  },  
  css: {  
    src: [  
      '<%= cssTargetDir %>/master.css',  
    ]  
  },  
},
```

Measure it

- Google PageSpeed
- Yslow
- Fiddler



Testing

Front end unit tests

Karma and Mocha (JS Test Framework)

JAVASCRIPT

```
var expect = chai.expect;
beforeEach(module('project-staffing'));
describe('UpperCase Test', function() {
  it('should convert first character to UpperCase', inject(function(uppercaseFilter) {
    expect(uppercaseFilter('a')).to.equal('A');
    expect(uppercaseFilter('hello world')).to.equal('Hello World');
  }));
});
```

Running Unit Tests with Karma

BASH

```
npm install -g karma-cli
karma start karma.conf.js
// or
grunt karma
```

Sinon (Mocking Framework)

JAVASCRIPT

```
var Activity;
var $http;

beforeEach(inject(function(_Activity_, _$http_) {
    Activity = _Activity_;
    $http = _$http_;
    sinon.stub($http, 'post', function(){});
}));

describe('Activity Service', function() {
    it('should have send http POST to backend after saving one activity', inject(function(Activity) {
        Activity.saveActivity('user', 'action', 'object');
        expect($http.post.callCount).to.equal(1);
    }));
});
```


Chai (BDD/TDD framework)

Should

```
chai.should();  
foo.should.be.a('string');  
foo.should.equal('bar');
```

JAVASCRIPT

Expect

```
var expect = chai.expect;  
expect(foo).to.be.a('string');  
expect(foo).to.equal('bar');
```

JAVASCRIPT

Assert

```
var assert = chai.assert;  
assert.typeOf(foo, 'string');  
assert.equal(foo, 'bar');
```

JAVASCRIPT

End2End tests

Protractor and Jasmine (BDD framework)

JAVASCRIPT

```
describe('Manage customer', function() {  
  var ptor;  
  
  beforeEach(function() {  
    browser.get('/');  
    ptor = protractor.getInstance();  
    element(by.id('navEmployees')).click();  
    element(by.id('navListEmployees')).click();  
  });  
  
  it('should navigate to list employees page', function() {  
    expect(ptor.getCurrentUrl()).toMatch(/#\//list-employees/);  
  });  
  
});
```

Protractor and Jasmine

```
it('should find employee Maria on list search page', function() {
    createMultipleEmployees(); // Creates employees: Max, Maria, Daniel, John
    ...
    element(by.id('searchText')).sendKeys('Ma');
    expect(element.all(by.id('employee')).count()).toBe(2);
    element(by.id('searchText')).sendKeys('ria');
    expect(element.all(by.id('employee')).count()).toBe(1);
});
```

JAVASCRIPT

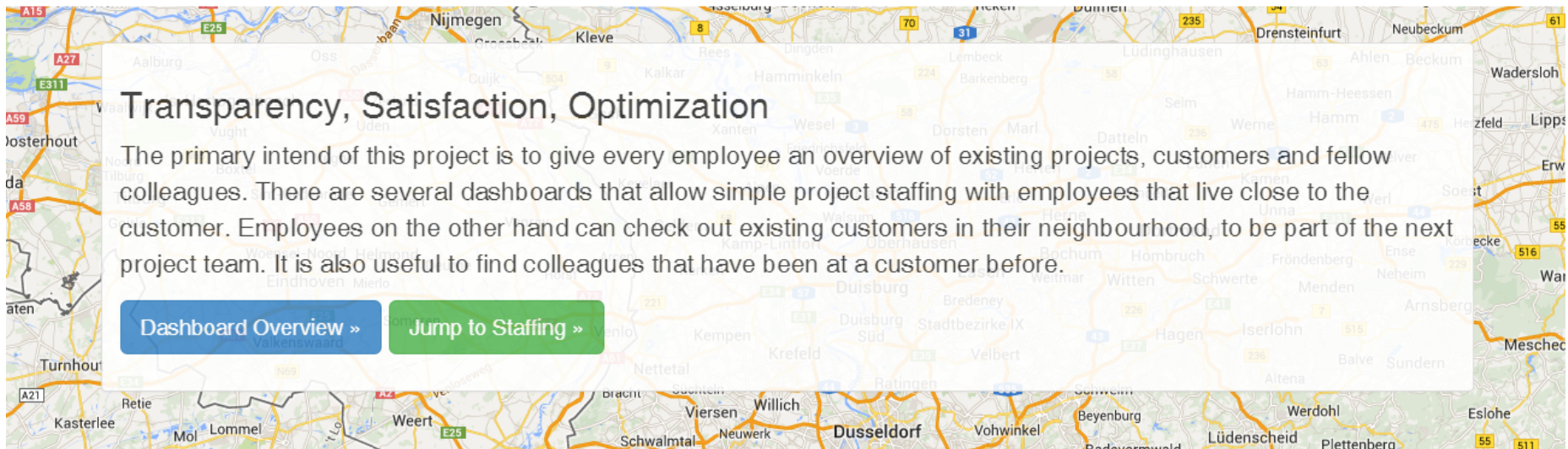
Running End2End Tests with Protractor

```
npm install -g protractor
webdriver-manager start
protractor test/client/e2e/conf.js
```

BASH



Demo



Transparency, Satisfaction, Optimization

The primary intend of this project is to give every employee an overview of existing projects, customers and fellow colleagues. There are several dashboards that allow simple project staffing with employees that live close to the customer. Employees on the other hand can check out existing customers in their neighbourhood, to be part of the next project team. It is also useful to find colleagues that have been at a customer before.

[Dashboard Overview »](#) [Jump to Staffing »](#)

Transparency

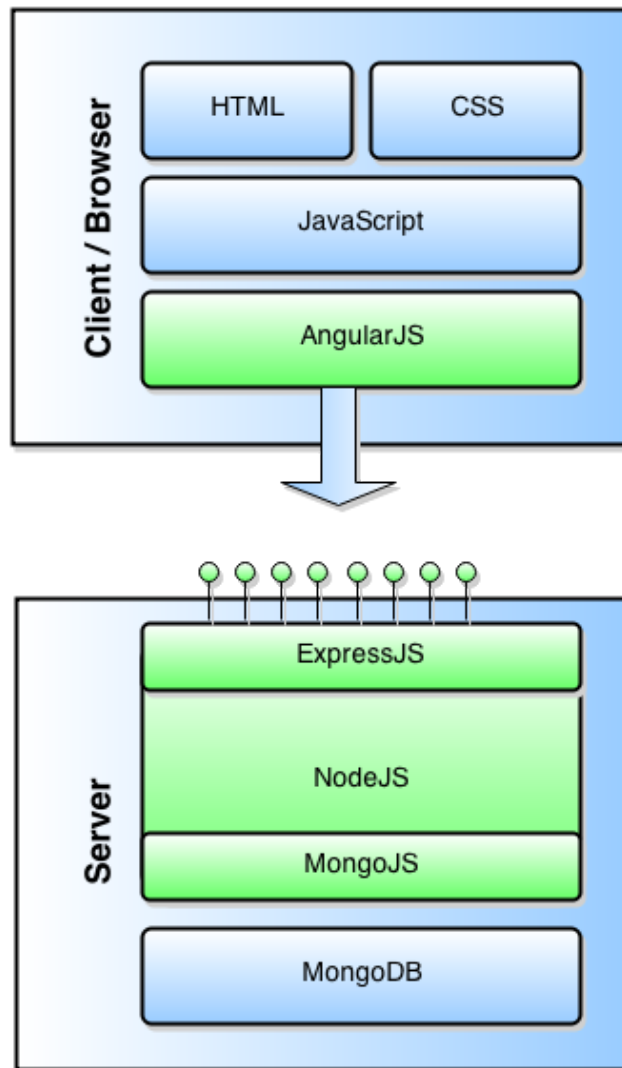
Everybody can see current projects, customers and the location of fellow colleagues. Therefore it is easier to get in contact with the right person when new projects come up.

Satisfaction

Every time managers need to staff new projects they can check who will be available at the project start and decide which possible candidate lives closest to the customer. The dashboards allow employees to proactively check out projects in their area or technologies used in projects that they would like to use as well.

Optimization

By choosing employees for projects that live close to the customers, travel expenses will be reduced and travel times of consultants will be minimized.



Project Staffing App - TEST Environment

<http://54.220.185.224:9000/>

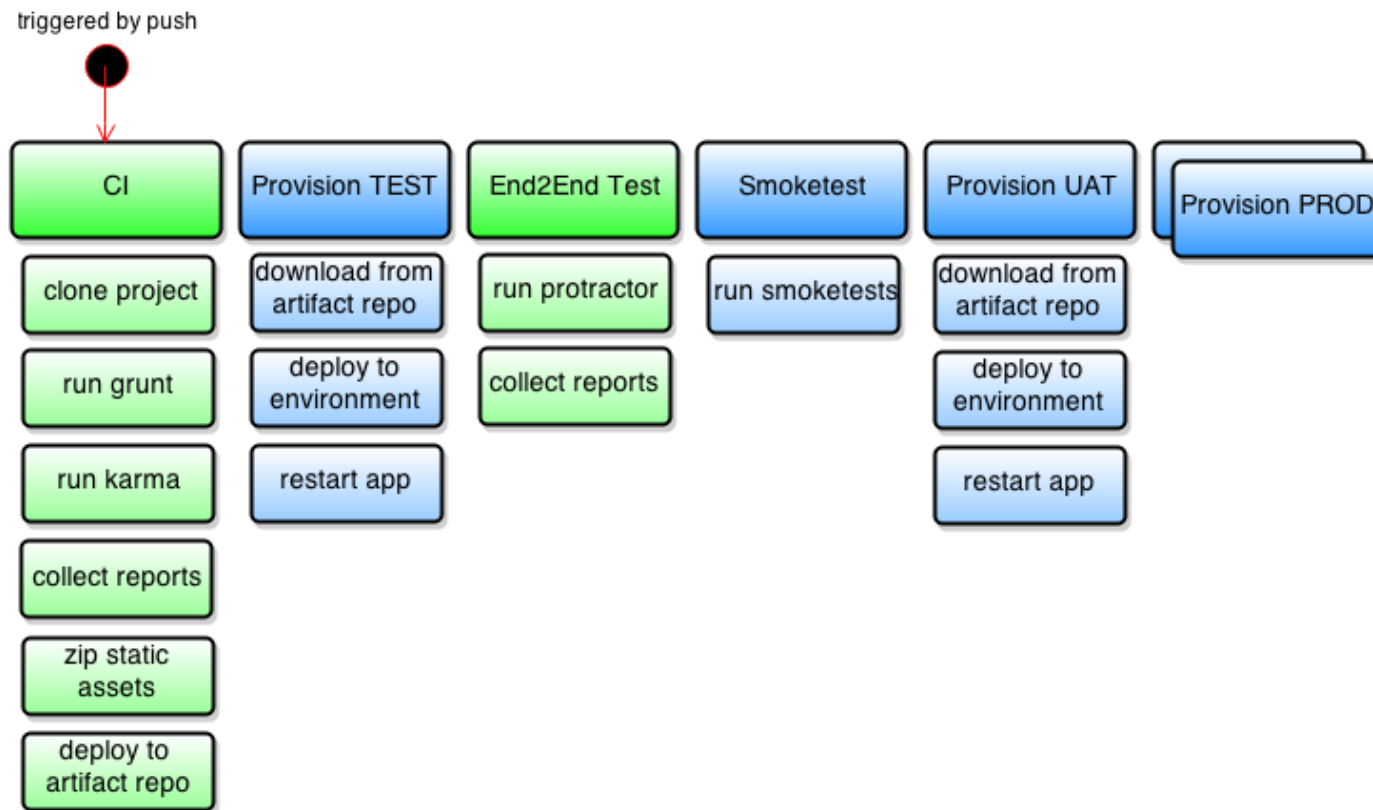
Delivery Pipeline

<http://54.74.89.170/jenkins/view/CL-Pipeline/>

Info: Server will be shutdown after talk

- #1 Early feedback
- #2 Fail fast
- #3 Production readiness

Delivery Pipeline Steps



Collect the reports

- Mocha reporter \Rightarrow unit test
- Jasmine reporter \Rightarrow end2end tests
- JSLint \Rightarrow static code analysis

Mocha Report (unit tests)

Test Result : Filter Test UpperCase Test

0 failures (± 0)

2 tests (± 0)
Took 22 ms.

All Tests

Test name	Duration	Status
<u>should convert first charactor of each whitespace separated string to UpperCase</u>	1 ms	Passed
<u>should convert first charactor of string to UpperCase</u>	21 ms	Passed

Protractor Report (end2end tests)

Jenkins > CL-Jobs > PS - End2End Test > #52 > Test Results > (root)

[ENABLE AUTO REFRESH](#)

[Back to Project](#)

[Status](#)

[Changes](#)

[Console Output](#)

[View as plain text](#)

[Edit Build Information](#)

[History](#)

[Parameters](#)

[Environment Variables](#)

[Git Build Data](#)

[No Tags](#)

[Test Result](#)

[Previous Build](#)

[Next Build](#)

Test Result : (root)

4 failures (-16)



20 tests (±0)

Took 2 min 54 sec.

[add description](#)

All Failed Tests

Test Name	Duration	Age
+ Delete all customer and add new customer.all customers should have been deleted	10 sec	3
+ Delete all customer and add new customer.should navigate to manage customers	10 sec	3
+ List customer on map tests:.navigate to cologne should find customer on map	10 sec	3
+ List customer on map tests:.should navigate to manage customers	16 sec	3

All Tests

Class	Duration	Fail	(diff)	Skip	(diff)	Pass	(diff)	Total	(diff)
Delete all customer and add new customer	38 sec	2	-1	0		1	+1	3	
List customer on map tests:	27 sec	2		0		0		2	
Listing Employees Page	4.5 sec	0	-2	0		2	+2	2	
Manage Employees Page	1 min 26 sec	0	-5	0		5	+5	5	
Project Staffing Start Page	18 sec	0	-8	0		8	+8	8	

JSLint Report (static code analysis)

CheckStyle Result

Warnings Trend

All Warnings	New Warnings	Fixed Warnings
138	138	0

Summary

Total	High Priority	Normal Priority	Low Priority
138	138	0	0

Details

Folders **Files** Warnings Details New

File	Total	Distribution
address_controller.js	2	<div></div>
app.js	2	<div></div>
customer_controller.js	30	<div></div>
dashboard_controller.js	18	<div></div>
employee-directives.js	1	<div></div>
employee_controller.js	13	<div></div>
filter.js	1	<div></div>
navigation_controller.js	1	<div></div>
ngAutocomplete.js	30	<div></div>
project_controller.js	4	<div></div>
routes.js	11	<div></div>
skill_controller.js	3	<div></div>
staffing_controller.js	20	<div></div>
timeline_controller.js	2	<div></div>
Total	138	

Links

Updated Slides

- <http://marcelbirkner.github.io/project-staffing/slides/introduction/>

Project Documentation

- <http://marcelbirkner.github.io/project-staffing/>

Source Code @ Github

- <https://github.com/marcelbirkner/project-staffing>

Thank You!

Do you have comments or questions?

Marcel Birkner - Software Consultant
Bastian Krol - Software Consultant



codecentric



