

BDD End-to-end Browser-Testing mit Node.js

Entwicklertag Frankfurt 2016 · 10.3.2016

Markus Tacker

Channeller of Technological Obscurities (CTO)

[@coderbyheart](#) · [cto.hiv](#)



Turning organizations
into entrepreneurial networks



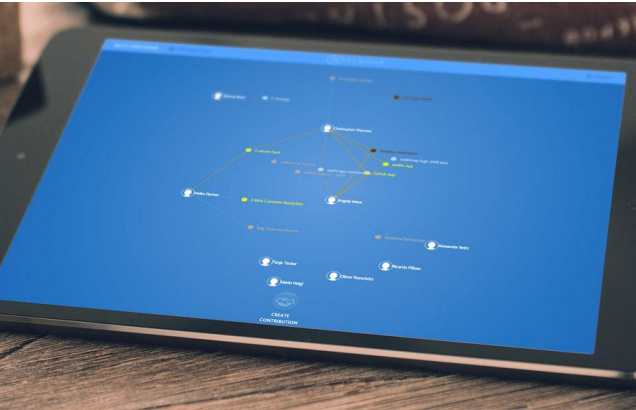
“The way to creating
Leaders at every level is
The Way of Resourceful Humans”

Captain David L. Marquet
US Navy (retired) Captain
Nuclear Submarine USS Santa Fe

Resourceful Technology

- JavaScript (ES6)
 - Backend: Node.js, Redis, CQRS/ES
 - Frontend: Angular.js (1.5+), Bootstrap v4, make
- Android, iOS
- Commodity: in the cloud
 - AWS
 - Elastic Beanstalk
 - Lambda
 - Travis CI
 - Greenkeeper.io
 - Google Apps for Work
 - Slack

Resourceful Tools



netwoRHk

enables, promotes and visualizes truly customer-focused teamwork



staRHs

beautifully visualized 360-degree feedback



caRHds

enables, demands and visualises everybody to contribute to valuable meetings

BDD: Short intro

Let user stories drive your (product) development

Title (one line describing the story)

(Narrative)

As a [role]

I want [feature]

So that [benefit]

Scenario 1...n: Title

Given [context]

And [some more context]...

When [event]

Then [outcome]

And [another outcome]...

Read more at: <http://dannorth.net/whats-in-a-story/>

BDD: Benefits

- plain english
- easy to scan
- easy to discuss about with non-developers
- not tied to code (in contrast to unit tests)
- user-story format widely used in agile teams

Browser-based testing: PITA

applying BDD to browser based testing ...

- makes writing tests more fun
- forces you to write expressive markup
- takes time to implement underlying execution logic

Yadda: BDD for Node.js

<https://github.com/acuminous/yadda>

- General purpose BDD library, can be run with your test runner
- we use mocha
- and Selenium (we want real browser tests)
- headless is also supported via PhantomJS

Yadda: BDD for Node.js

How it works:

- testrunner reads .feature files
- initiates global context with a selenium instance
- executes steps
- for every condition (regexp match) a method in the context files is called
- which calls your favorite assertion library

Context step definition example

Step:

When I enter "187.000 €" into the input amount

Step definition:

```
.when(/^I enter "[^"]+" into the input (.+)$/, function (value, input, next) {  
  this.driver.findElement(By.name(input)).sendKeys(r(value)).then(next);  
})
```

Live demo 

Questions?

Thank you!

Slides

<https://goo.gl/03zXZY>

Code on GitHub

<https://goo.gl/o9OvQP>

Please rate this talk!
or send feedback to

<http://lanyrd.com/sdwtpy>

[@coderbyheart](https://twitter.com/coderbyheart)

markus@resourceful-humans.com