

EOS React

Miroslav Jančařík, Martin Jurča
miroslav.jancarik@firmaseznam.cz
martin.jurca@firmaseznam.cz

Introduction

- Your name and surname
- Your position
- Where do you work
- Your experience
- Your expectations

Rules

- No cell phone
- No notebook
- One person speaks at a time

Schedule

- ES6/Jasmine
- Workshop
- Less/React/Enzyme
- Workshop
- Workshop
- Workshop

Why ES6





ES6

- Assignment Destructuring
- Spread operator and rest parameters
- Arrow functions
- Template literals
- Let and Const
- Promises
- Classes
- Modules

Assignment Destructuring

- `var color = car.color;`
- `var wheels = car.wheels;`
- `var { wheels } = car === var wheels = car.wheels`
- `var { color: paint } = car === var paint = car.color`

Spread operator and rest parameters

- Rest parameters are a better alternative to arguments
- **function foo (bar, ...rest) {}**
- Removes the need for .apply when calling functions, **fn(...[1, 2, 3])** is equivalent to **fn(1, 2, 3)**

Arrow functions

- The `this` keyword is inherited from the parent context
- **param => returnValue**
- `[1, 2].map(x => x * 2)` vs `[1,2].map(function(x){ return x * 2; })`

Template literals

- Template literals provide string interpolation, e.g. ``car is ${color}`` where color is a variable
- You can use any valid JavaScript expressions in the interpolation, such as ``${2 * 3}`` or ``${foo()}``

Let and Const

- **let and const are alternatives to var** for declaring variables
- **let is block-scoped** instead of lexically scoped to a function
- let is hoisted to the top of the block, while var declarations are hoisted to top of the function
- const is also block-scoped, hoisted, and constrained by TDZ semantics

Promises

```
node95.js x
1 var floppy = require('floppy');
2
3 floppy.load('disk1', function (data1) {
4     floppy.prompt('Please insert disk 2', function () {
5         floppy.load('disk2', function (data2) {
6             floppy.prompt('Please insert disk 3', function () {
7                 floppy.load('disk3', function (data3) {
8                     floppy.prompt('Please insert disk 4', function () {
9                         floppy.load('disk4', function (data4) {
10                            floppy.prompt('Please insert disk 5', function () {
11                                floppy.load('disk5', function (data5) {
12                                    // if node.js would have existed in 1995
13                                    });
14                                });
15                            });
16                        });
17                    });
18                });
19            });
20        });
21    });
22}
```

Promises

- The Promise object is used for **asynchronous operations**. A Promise represents an operation that might not have completed yet, but is expected in the future.

Classes

- Not “traditional” classes, **syntax sugar** on top of prototypal inheritance
- Syntax similar to declaring objects, **class Car {}**
- Instance methods
- Static methods
- Constructor method
- Prototypal inheritance with a simple syntax **class Car extends Vehicle**

Modules

- **export default value** exports a default binding
- **export var car = 'car'** exports a named binding
- **export { car, vehicle }** exports all bindings using a single declaration

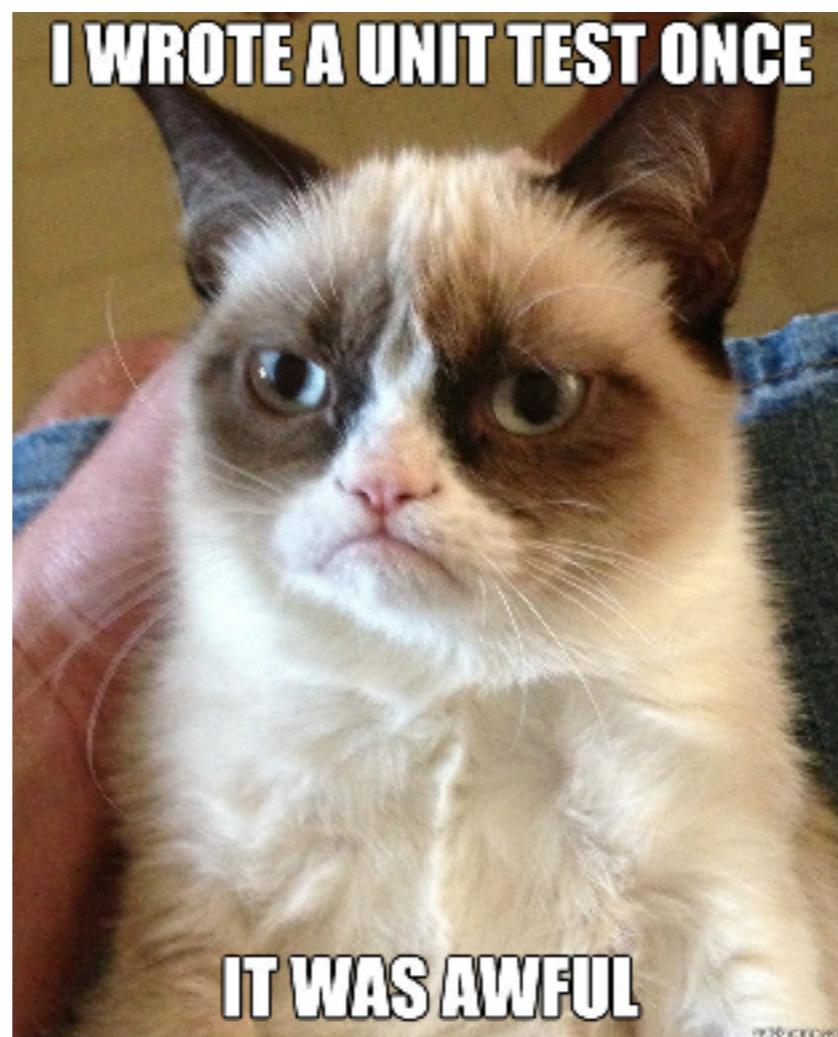
Modules

- **import 'car'** loads the car module into the current module
- **import car from 'car'** assigns the default export of the car module to a local car variable
- **import { car, vehicle } from 'machine'** imports named exports car and vehicle from the machine module
- **import { car as baseCar } from 'machine'** imports named export car but aliased as a baseCar variable
- **import { default } from 'car'** also imports the default export
- **import { default as car } from 'car'** imports the default export aliased as car
- **import machine, { car, vehicle } from 'machine'** mixes default machine with named exports car and vehicle in one declaration
- **import * as machine from 'machine'** imports all exports from the machine module as a machine variable (an object containing all exported bindings)

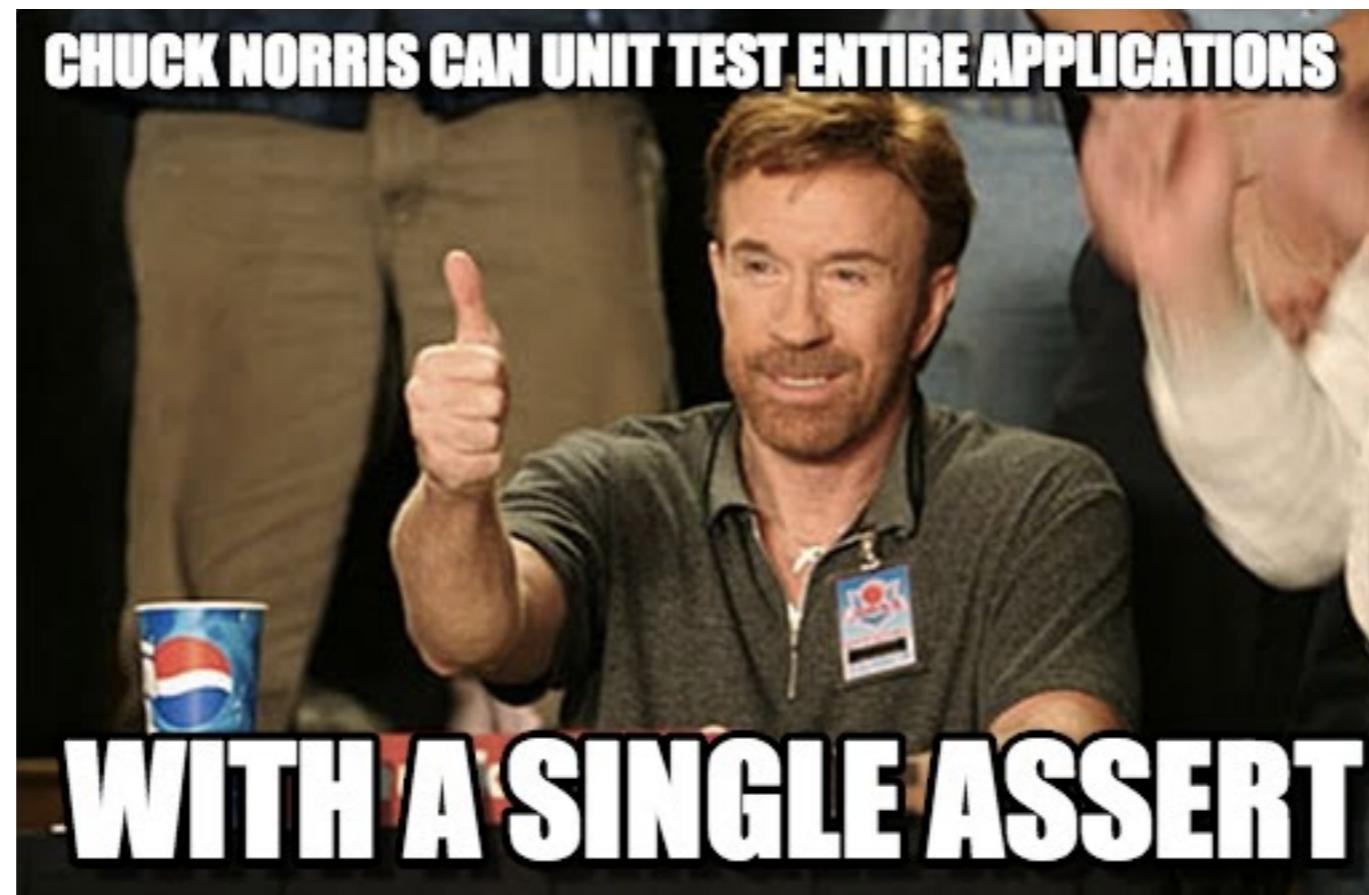
Perfect code

Unit tests

Unit tests



Unit tests



Unit tests

- Easy
- Quick
- Isolated

Unit tests

- Arrange
- Act
- Assert

Frameworks

- Jasmine
- Mocha (chai, sinon)
- Jest

Reporters

- Dot
- Spec
- JUnit

Runners

- Karma
- Node.js

Jasmine syntax

```
describe('Jasmine syntax', function() {  
  it('hello world', function() {  
    expect(typeof 'hello world').toEqual('string');  
  });  
});
```

Jasmine syntax

<http://jasmine.github.io/edge/introduction.html>

Demo

Workshop

[462, -70]

[0,0]

[562, 30]

[1024, 768]

Retrospective

Less

- The dynamic stylesheet language.
- The LESS syntax is similar to CSS.
- Enables **variables**, **mixins**, **nested rules**, etc.

React

- Javascript library for building user interfaces
- **Virtual DOM**
- It's all about components
- Data flow

React

- MPA
- SPA
- IMA

React Top level API

- **createElement**
- **createFactory**
- **createClass**
- **Component** - base class for ES6

ReactDOM Top level API

- **render** - renders a ReactElement into the DOM
- **unmountComponentAtNode**
- **findDOMNode** - returns the corresponding native DOM element

ReactDOMServer Top API

- **renderToString** - renders a ReactElement to its initial HTML
- **renderToStaticMarkup** - similar to renderToString, except this doesn't create extra DOM attributes used by React

React Top level API

[https://facebook.github.io/react/
docs/component-api.html](https://facebook.github.io/react/docs/component-api.html)

Demo

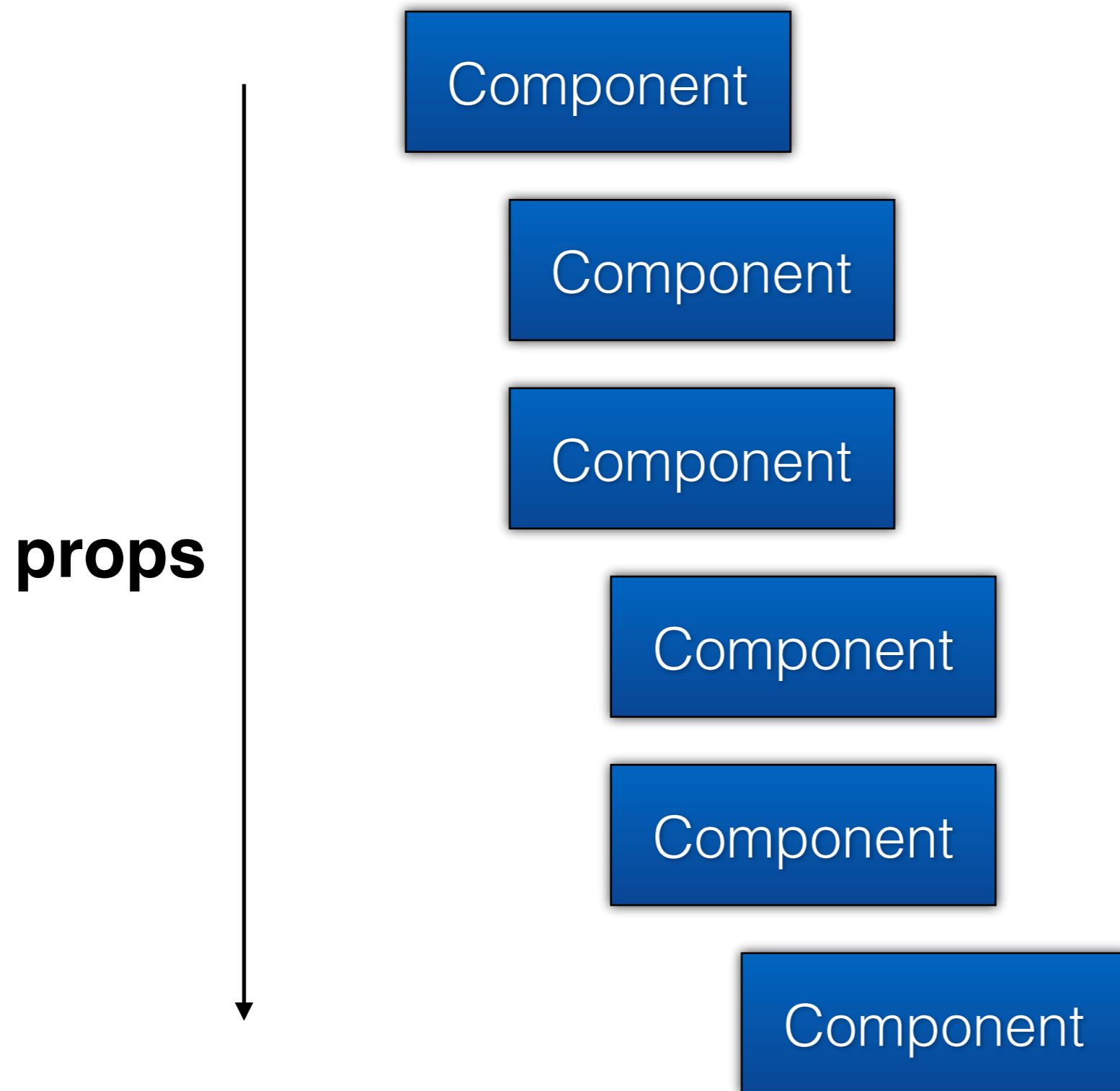
JSX Gotchas

- JSX is not HTML
- attributes that are not in the HTML specification will be ignored
- escapes all the strings (hard to inject custom HTML)
- Components' render() method must return a single React element or null

Component properties

- **props** - data passed in from a parent component
- **state** - component's internal data
- **refs** - references to DOM elements (and other components) rendered by the component

Components



Component API

- **setState** - shallow merge of the provided state object into the current state
- **forceUpdate** - force re-render, skip shouldComponentUpdate method check (usually a bad idea)

Demo

Component Lifecycle

- **componentWillMount** - invoked once at the server and client
- **componentDidMount** - invoked once only at the client
- **render** - (should be) pure function
- **componentWillUnmount** - invoked once at the client, use for cleanup of timeouts, intervals, etc.

Component Lifecycle

- **componentWillReceiveProps**
- **shouldComponentUpdate** - used for performance boost
- **componentWillUpdate** - (should be) a pure function
- **componentDidUpdate**

Component Lifecycle

[https://facebook.github.io/react/
docs/component-specs.html](https://facebook.github.io/react/docs/component-specs.html)

Demo

Reusable components

- **defaultProps**
- **propTypes**

PropTypes

- React.PropTypes.{**type**}
- string, bool, number, object, fun, shape, instanceOf, ...

React Add-ons

- **update**
- **shallowCompare**
- **Perf**
- **TestUtils**

UI unit tests

- Given properties and state, what structure our rendered element tree will have?
- Given an output of render(), is there a possibility to transition from state A to state B?

Enzyme

- Shallow rendering
- Full DOM rendering
- Static rendered markup

Demo

Workshop

Retrospective

Pattern lab

ATOMS ▾ MOLECULES ▾ ORGANISMS ▾ TEMPLATES ▾ PAGES ▾ ALL

SIZE 1251PX / 78.19EM S M L FULL RAND ⚙



Workshop



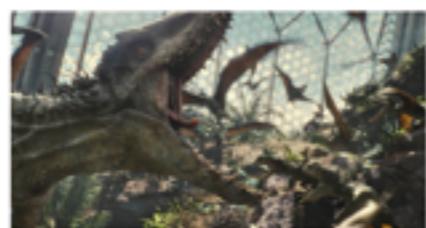
Mart'án

Astronaut Mark Watney (Matt Damon) během mise na Mars málem zahynul v prašné bouři. Zbytek jeho posádky planetu opustil v domnění, že je mrtvý. Ale Watney přežil a ocitne se tak omylem sám na nepřátelské planetě.



Everest

Everest v režii Baltasara Kormákura rekapituluje jednu z nejtragičtějších expedic, při které majestátní masív poslal svým vyzyvatelům mrazivé varování. Píše se polovina devadesátých let a Himaláje se stávají oblíbenou turistickou destinací. Na osmitisícovky už nešplhají jen zkušení horolezci, ale táhnou se tam celá procesí dobrodruhů z celého světa.



Jurský svět

Před dvaadvaceti lety tragicky skončil sen excentrického milionáře Johna Hammonda, který chtěl na odlehém ostrově z dinosaurovi DNA vypěstovat Jurský park s živými exponáty. Od té doby se ale mnohé změnilo, park jede na plné obrátky, ročně jím projdou milióny nadšených návštěvníků, kteří s očima navrch hlavy sledují v akci desítky „vyhynulých“ živočichů.



Hotel Transylvánie 2

Zdá se, že Hotelu Transylvánie se začíná dařit... Dracula konečně ustoupil ze svých striktních zásad ubytovávat v hotelu pouze příšery, a ten se proto otevírá i lidským návštěvníkům. Za zavřenými dveřmi krypty ale Draculu sžírají pochybnosti o tom, zda se jeho napůl lidský a napůl upíří vnuk Dennis vůbec někdy stane upírem.



Osm hrozných

Několik let po občanské válce se zasněženými kopci Wyomingu řítí dostavník. Veze se v něm John Ruth (Kurt Russell), lovec lidí, známý také jako „Kat“, a Daisy Domergueová (Jennifer Jason Leigh). Jeho čeká odměna, vypsaná na její hlavu, ji soud a oprátku. Po cestě potkají dva muže – majora Marquise Warrena (Samuel L. Jackson), bývalého vojáka, ze kterého se také stal obávaný lovec lidí a Chrise Mannixe (Walton Goggins), jižanského odpadlíka, který o sobě tvrdí, že je novým šerifem ve městě, kam mají všichni namířeno. Když jim silná vichřice zabrání pokračo



Creed

Adonis Johnson nikdy nepoznal svého slavného otce, světového šampióna v těžké váze Apolla Creeda, který zemřel ještě před jeho narozením. Přesto však svůj původ nezapře a box má v krvi, takže ho to stále tahá do Philadelphie, města, kde se odehrál legendární souboj Apollo Creed vs. Rocky Balboa. Adonis Rockyho vyhledá a požádá ho, aby se stal jeho trenérem. Rocky zpočátku odmítá, protože ring je pro něho už jen minulost, ale postupně v Adonisovi objevuje sílu a odhodlání jeho otce Apolla, bývalého rivala a blízkého přítele. A tak nakonec souhlasí, že se m...



Vyhledávání...



▶ 0:02 / 3:07



Mart'an



Astronaut Mark Watney (Matt Damon) během mise na Mars málem zahynul v prašné bouři. Zbytek jeho posádky planetu opustil v domnění, že je mrtvý. Ale Watney přežil a ocitne se tak omylem sám na nepřátelské planetě. Má k dispozici jen hubené zásoby a tak musí využít veškerý důvtip, invenci a technické znalosti, aby nejen přežil, ale také nalezl způsob, jak vyslat signál na Zemi, že je naživu. V okamžiku, kdy se mu to podaří, začne miliony kilometrů daleko od něj NASA a mezinárodní tým vědců neúnavně pracovat, aby dostali tohoto „Martana“ domů. Zatímco členové jeho posádky souběžně prokazují svou odvahu na téměř nemožné záchranné misi.

Hrají: Matt Damon, Kate Mara, Jessica Chastain

Retrospective

Busting myths

- jQuery is enough
- Immutable.js is a must
- No-one builds big applications with React

Retrospective

- What did you learn today?
- What will you do differently tomorrow?

Thank you for your attention

Miroslav Jančařík, Martin Jurča
miroslav.jancarik@firma.seznam.cz
martin.jurca@firma.seznam.cz