

# STRV

## REACT & REACT NATIVE

Daniel Kijkov, Vladimír Vanek, Josef Zavisek

## Overview of the 1<sup>st</sup> developer meetup

- introducing React & React Native
- preparation of environment
- simple chat app from scratch

# What is REACT NATIVE?

- library for building native mobile apps on Android and IOS

***“Write once, run anywhere”***



***“Learn once, write anywhere”***

# What is REACT?

- library for building user web user interfaces
- released 2013
- goes against some practices that we were used to

# Components in apps

CHAT APP

**React messages**

4

NodeJS chat

9

Koa

5



**John Doe**

Donec vitae sapien ut libero venenatis faucibus. Nullam quis ante. Etiam sit amet orci eget eros faucibus tincidunt.



**Brian Black**

Far far away, behind the word mountains, far from the countries Vokalia and Consonantia, there live the blind texts.



**Jack Smith**

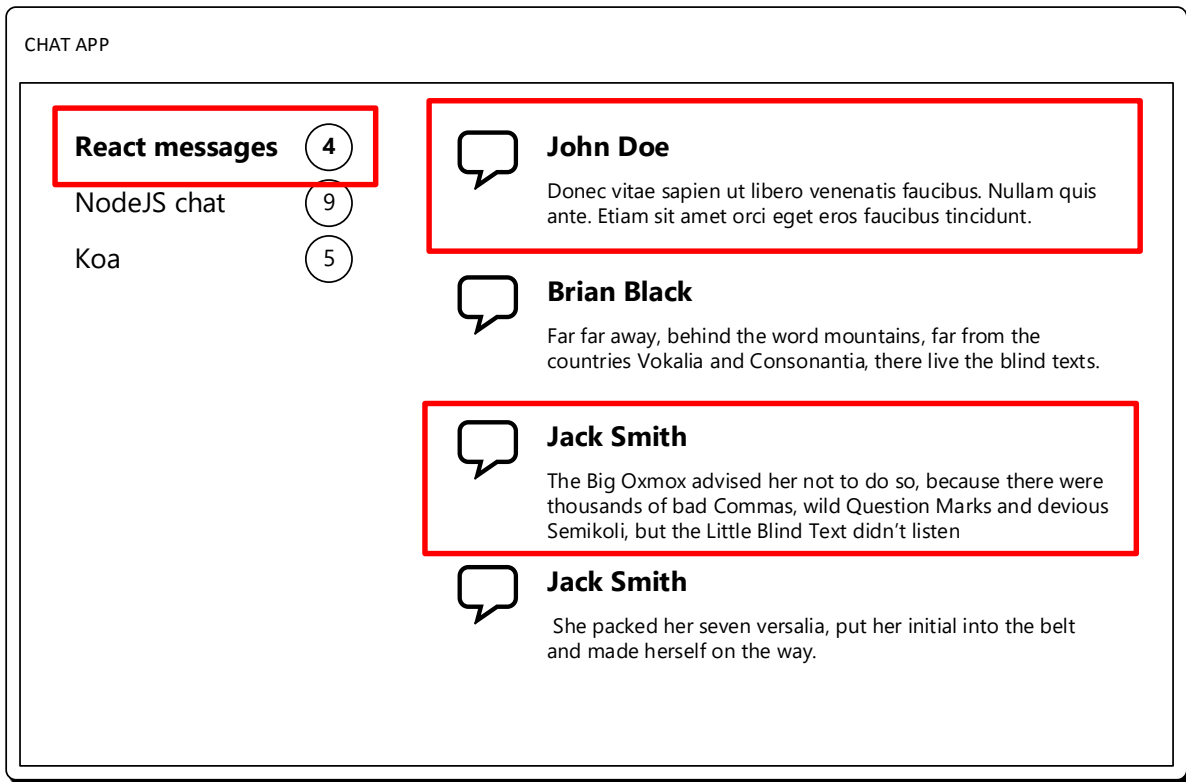
The Big Oxmox advised her not to do so, because there were thousands of bad Commas, wild Question Marks and devious Semikoli, but the Little Blind Text didn't listen



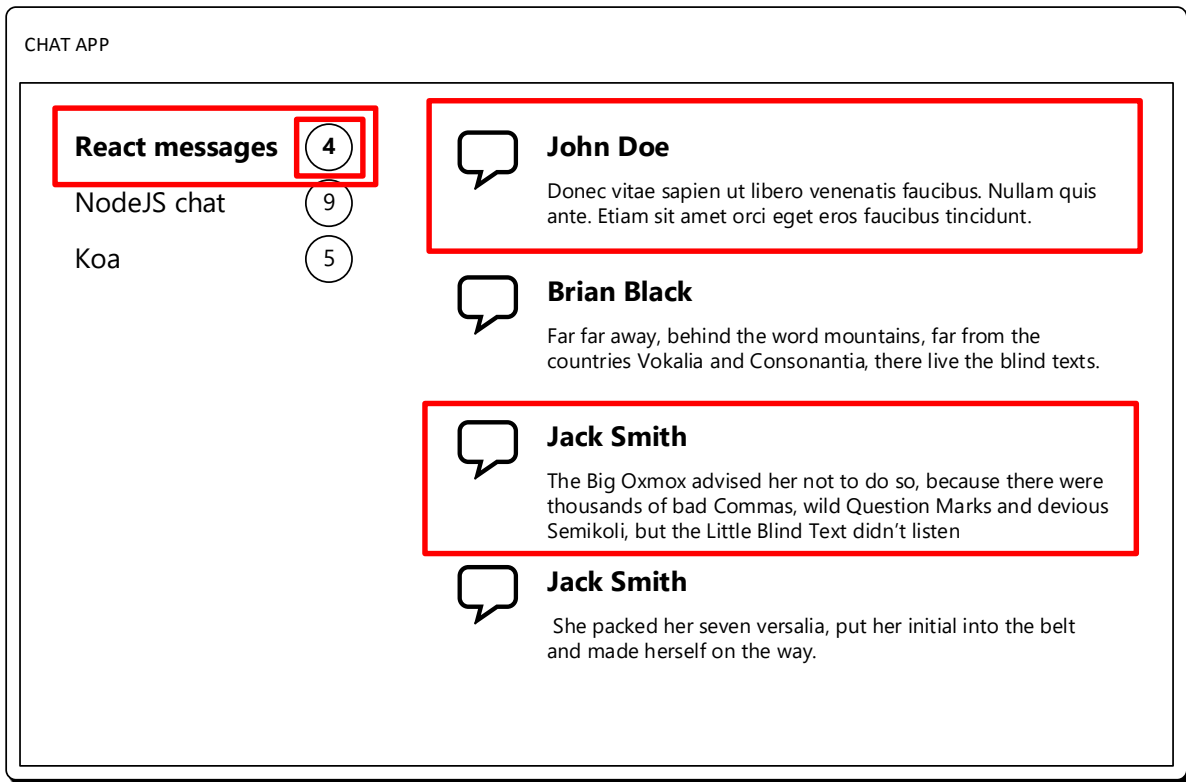
**Jack Smith**

She packed her seven versalia, put her initial into the belt and made herself on the way.

# Components in apps



# Components in apps



# Common way in MVC frameworks, separation of concerns

## HTML template

```
2 <form>
3   <p>First name: <input data-bind="value: firstName" /></p>
4   <p>Last name: <input data-bind="value: lastName" /></p>
5   <p>Full name: <strong data-bind="text: fullName"></strong></p>
6   <button class="btn btn--default" data-bind="click: capitalizeLastName">Go caps</button>
7 </form>
```

## Javascript

```
2 function AppViewModel() {
3   this.firstName = ko.observable("Bert");
4   this.lastName = ko.observable("Bertington");
5
6   this.capitalizeLastName = function() {
7     var currentVal = this.lastName();
8     this.lastName(currentVal.toUpperCase());
9   };
10 }
11
12 ko.applyBindings(new AppViewModel());
```

## CSS

```
2 .btn {
3   display: inline-block;
4   outline: none;
5   cursor: pointer;
6 }
7
8 .btn--default {
9   border: 1px solid #ccc;
10  background-color: #4c9797;
11  border-radius: 3px;
12 }
```



# When trying to change unpredictable code



# Issues with MVC libraries

CHAT APP

**React messages**

4

NodeJS chat

9

Koa

5



**John Doe**

Donec vitae sapien ut libero venenatis faucibus. Nullam quis ante. Etiam sit amet orci eget eros faucibus tincidunt.



**Brian Black**

Far far away, behind the word mountains, far from the countries Vokalia and Consonantia, there live the blind texts.



**Jack Smith**

The Big Oxmox advised her not to do so, because there were thousands of bad Commas, wild Question Marks and devious Semikoli, but the Little Blind Text didn't listen



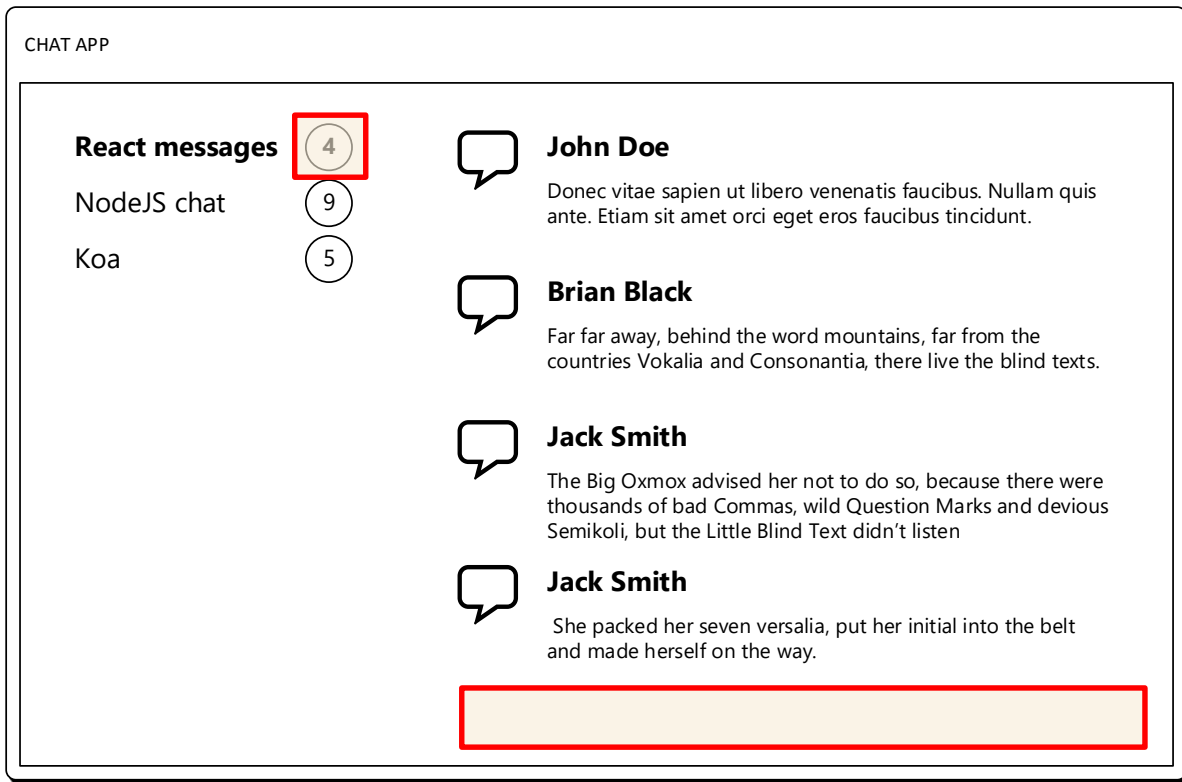
**Jack Smith**

She packed her seven versalia, put her initial into the belt and made herself on the way.

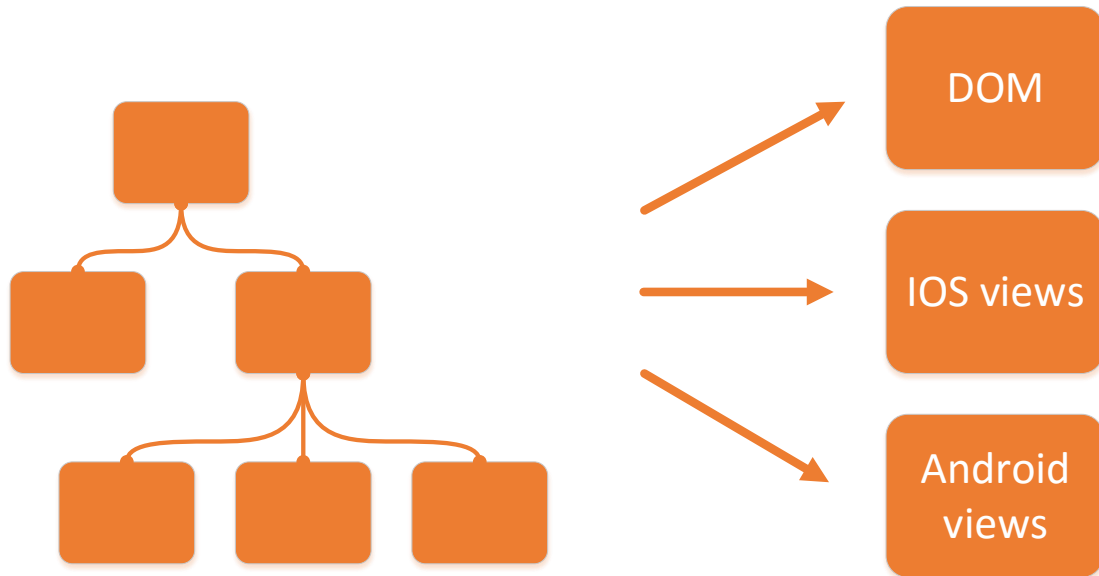
# Issues with MVC libraries



# Issues with MVC libraries



# React components



# How does the component look in REACT?

```
1
2 var Toggle = React.createClass({
3
4   getInitialState: function() {
5     return { on: false };
6   },
7
8   handlePress: function() {
9     this.setState({ on: !this.state.on });
10  },
11
12  render: function() {
13    var text = this.state.on ? "Toggle ON" : "Toggle OFF";
14    var cssClasses = classNames({
15      "toggle": true,
16      "toggle--on": this.state.on
17    });
18
19    return (
20      <div className={cssClasses}>
21        <span>{text}</span>
22      </div>
23    );
24  }
25
26  });
27
```

# Component parts and composition

```
2
3  var Comment = React.createClass({
4    render: function() {
5      return (
6        <div className="comment" onClick={this.props.onClick}>
7          <h2 className="commentAuthor">
8            {this.props.author}
9          </h2>
10         {this.props.children}
11       </div>
12     );
13   }
14 });
15
16 var CommentList = React.createClass({
17   handleClick: function() {
18     var listElement = this.refs.container.getDOMNode();
19     // ...
20   },
21   render: function() {
22     return (
23       <div className="commentList" ref="container">
24         <Comment author="Pete Hunt" onClick={this.handleClick}>This is one comment</Comment>
25         <Comment author="Jordan Walke" onClick={this.handleClick}>This is *another* comment</Comment>
26       </div>
27     );
28   }
29 });
30
```

# JSX transformation to JavaScript

```
1
2 var Hello = React.createClass({
3   handleClick: function() {
4     // do something
5   },
6   render: function() {
7     return (
8       <div className="main" onClick={this.handleClick.bind(this)}>Hello {this.props.name}</div>
9     );
10  }
11 });
12
```



```
1
2 var Hello = React.createClass({
3   handleClick: function() {
4     // do something
5   },
6   render: function() {
7     return React.createElement(
8       "div",
9       { className: "main", onClick: this.handleClick.bind(this) },
10      "Hello ",
11      this.props.name
12    );
13  }
14 });
15
```



# REACT NATIVE

# How does the component look in REACT NATIVE?

## Older syntax

```
1
2 var Toggle = React.createClass({
3
4   getInitialState: function() {
5     return { on: false };
6   },
7
8   handlePress: function() {
9     this.setState({ on: !this.state.on });
10  },
11
12  render: function() {
13    var text = this.state.on ? "Toggle ON" : "Toggle OFF";
14    return (
15      <TouchableHighlightWithoutFeedback
16        onPress={this.handlePress}
17        <View>
18          <Text>
19            {text}
20          </Text>
21        </View>
22      </TouchableHighlightWithoutFeedback>
23    );
24  }
25 });
26
27
```

## EcmaScript 2015 syntax

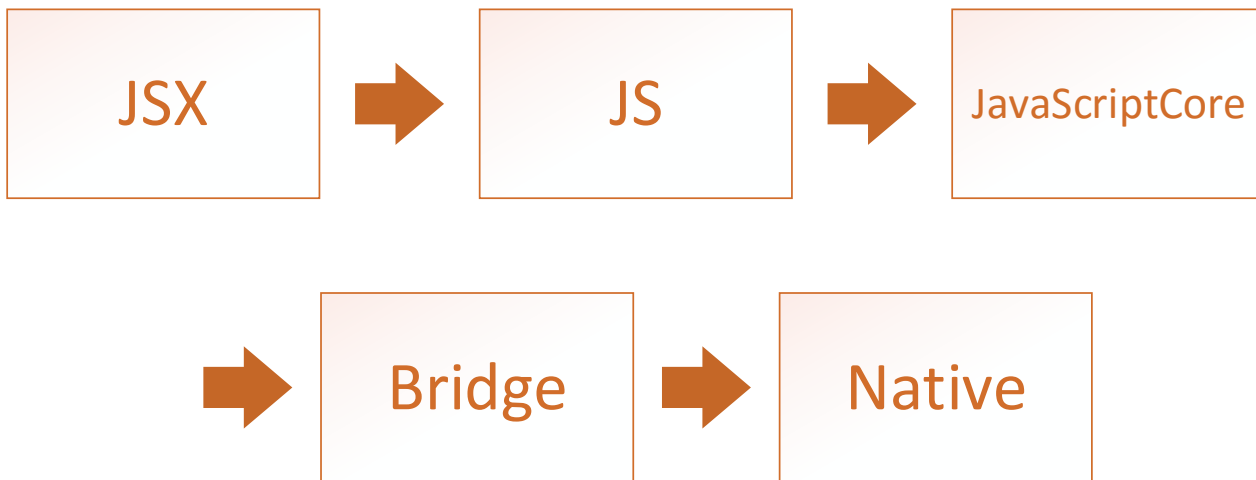
```
1
2 class Toggle extends React.Component {
3
4   constructor(props) {
5     super(props);
6     this.state = { on: false };
7   }
8
9   handlePress() {
10    this.setState({ on: !this.state.on });
11  }
12
13  render() {
14    const text = this.state.on ? "Toggle ON" : "Toggle OFF";
15    return (
16      <TouchableHighlightWithoutFeedback
17        onPress={this.handlePress.bind(this)}
18        <View>
19          <Text>
20            {text}
21          </Text>
22        </View>
23      </TouchableHighlightWithoutFeedback>
24    );
25  }
26
27 }
28
```

# Native components

```
2
3 // Android
4 var React = require('react-native');
5 var { DrawerLayoutAndroid, ProgressBarAndroid } = React;
6
7 var App = React.createClass({
8   render: function() {
9     return (
10       <DrawerLayoutAndroid
11         renderNavigationView={() => <Text>React Native</Text>}>
12         <ProgressBarAndroid />
13       </DrawerLayoutAndroid>
14     );
15   },
16 });
17
18
19
20
21
22 // iOS
23 var React = require('react-native');
24 var { TabBarIOS, NavigatorIOS } = React;
25
26 var App = React.createClass({
27   render: function() {
28     return (
29       <TabBarIOS>
30         <TabBarIOS.Item title="React Native" selected={true}>
31           <NavigatorIOS initialRoute={{ title: 'React Native' }} />
32         </TabBarIOS.Item>
33       </TabBarIOS>
34     );
35   },
36 });
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

## How it works internally

- small JavaScript code that controls native components



## What do you get?

- you don't have to wait for compile times
- declarative way of defining components

*Imperative – focuses on how the program operates*

*Declarative – focuses on what the program should accomplish*

- code sharing between platforms
- predictable – UI reflects the state
- simplicity of JavaScript with high performant UI

# What do you get?

- easy to write tests
- reuse components and their composition
- reuse developers 😊
- can be combined with other frameworks
- easy debugging (Chrome/Safari dev tools, Chrome extension)
- created by Facebook, strong community around
- server-side friendly rendering (virtual DOM, SEO)
- app with native feel

# STRV

Let's create an app

# Environment setup

## **REACT REQUIREMENTS:**

- <https://facebook.github.io/react-native/docs/getting-started.html#content>

## **EDITOR:**

- Sublime - <http://www.sublimetext.com/>
- Atom - <https://atom.io/>

## **FILES:**

- <https://github.com/strvcom/React-Meetup.git>



