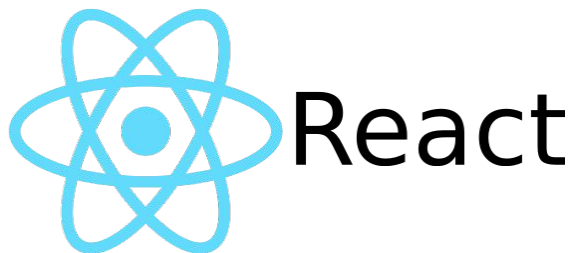


# React Native

Marcell Bán  
marcell.ban@lab.coop

# What is React?

- a framework for the web
- one-way reactive data flow
- declarative
- virtual DOM



# What is React Native?

- a framework to develop iOS and Android apps
- open source
- brings the React concept to mobile development
- learn once, write anywhere
- ES6(ish) JavaScript
- native components - not a WebView wrapper

# Native components

- standard platform components
- only if there is the React counterpart

```
//JavaScript
var { DrawerLayoutAndroid, ProgressBarAndroid } = React;
var App = React.createClass({
  render: function() {
    return (
      <DrawerLayoutAndroid
        renderNavigationView={() => <Text>React Native</Text>}
      >
        <ProgressBarAndroid />
      </DrawerLayoutAndroid>
    );
  },
});
```

# Asynchronous execution



- operations are async between the native platform and the JS code
- UI is always responsive
- you can do processing in the background
- Chrome Developer Tools

# Styling

- flexbox
- common web styles, not complete CSS support

```
//JavaScript
var styles = StyleSheet.create({
  image: { width: 40, height: 40, marginRight: 10 },
  title: { fontSize: 11, fontWeight: 'bold' },
});
```

```
<Image
  source={{uri: 'http://facebook.github.io/react/img/logo_og.png'}}
  style={styles.image}/>

<Text style={styles.title}>
  React Native
</Text>
```

# Polyfills



- npm install libraries? - yes
- already done:
  - flexbox
  - network - XHR, WebSocket, fetch
  - timing - setTimeout, interval, immediate, requestAnimationFrame
  - geolocation

# Extensibility



- <https://react.parts/native>

```
//Java
public class MyCustomModule extends
ReactContextBaseJavaModule {

    @ReactMethod
    public void processString(String input, Callback
callback) {
        callback.invoke(input.replace("Goodbye", "Hello"));
    }
}
```



# Extensibility



**//JavaScript**

```
var { NativeModules, Text } = React;

. . .
componentDidMount() {
  NativeModules.MyCustomModule.processString("GoodBye",
(text) => {
    this.setState({text});
  });
}

. . .
render: function() {
  return (
    <Text>{this.state.text}</Text> // Hello
  );
}
```

# How to get started?

- OS X - lucky
- linux - almost
- windows - nah

```
$ npm install -g react-native-cli
```

```
$ react-native init AwesomeProject
```

# Debugging



- shake it!
- use chrome developer tools
- live reload

# ES6ish



- [babeljs](#)
- **Arrow functions:** `<C onPress={() => this.setState({pressed: true})}`
- **Block scoping:** `let greeting = 'hi';`
- **Call spread:** `Math.max(...array);`
- **Classes:** `class C extends React.Component { render() { return <View />; } }`
- **Constants:** `const answer = 42;`
- **Destructuring:** `var {isActive, style} = this.props;`
- **Modules:** `import React, { Component } from 'react-native';`
- **Computed Properties:** `var key = 'abc'; var obj = {[key]: 10};`
- **Object Consise Method:** `var obj = { method() { return 10; } };`
- **Object Short Notation:** `var name = 'vjeux'; var obj = { name };`
- **Rest Params:** `function(type, ...args) { }`
- **Template Literals:** `var who = 'world'; var str = `Hello ${who}`;`

- **Object Spread:**
  - `var extended = { ...obj, a: 10 };`
- **Function Trailing Comma:**
  - `function f(a, b, c,) { }`
- **Async Functions:**
  - ```
async function doStuffAsync() {  
    const foo = await doOtherStuffAsync();  
};
```

# Who uses it?



- it's a new thing
  - iOS - early 2015
  - Android - 2015 september
- Facebook groups, adverts manager
- Discovery channel
- me
- you?

The background features several overlapping, semi-transparent geometric shapes in various shades of green and yellow. These shapes are layered to create a sense of depth and movement. The word "Questions?" is centered over these shapes in a dark, bold, sans-serif font.

**Questions?**