



# ReactNative

## по-быстрому

Презентация компании

[www.firstlinesoftware.ru](http://www.firstlinesoftware.ru)

2017

# Примеры приложений



<https://facebook.github.io/react-native/showcase.html>

# Отличия React от React Native

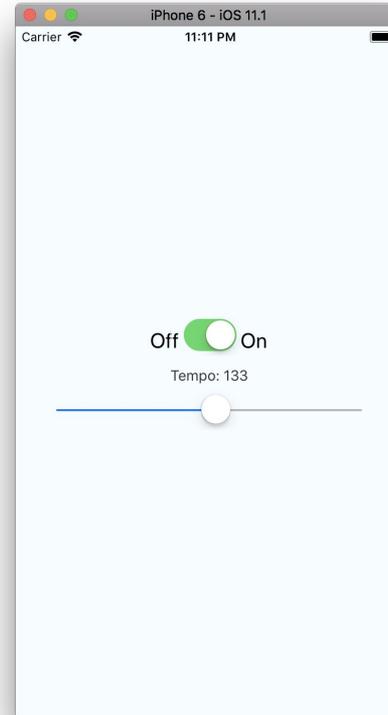
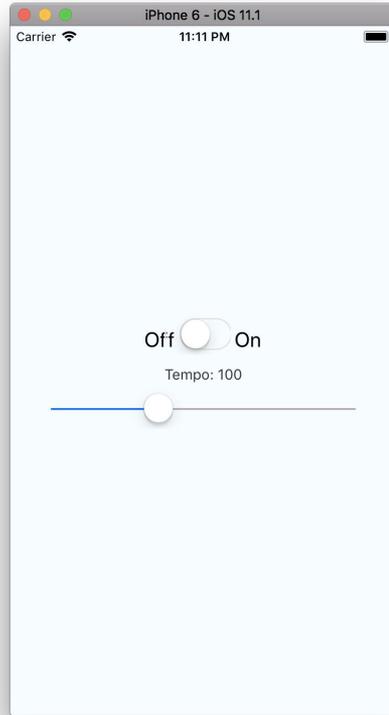
React

- Знает, что такое компонент
- Знает, как заставить работать компоненты вместе

React Native

- Знает, как отрисовать компоненты на экране
- Предоставляет набор базовых компонентов

# Пример простейшего приложения на React Native



# Установка библиотеки и инициализация приложения

- `npm install -g react-native-cli`
- `react-native init MyMobileApp`
- `cd MyMobileApp`
- `react-native run-ios`

## Импорт зависимостей

```
1 import React, { Component } from 'react';
2 import {
3   AppRegistry,
4   StyleSheet,
5   Text,
6   Slider,
7   Switch,
8   View
9 } from 'react-native';
10 import Sound from 'react-native-sound';
11
```

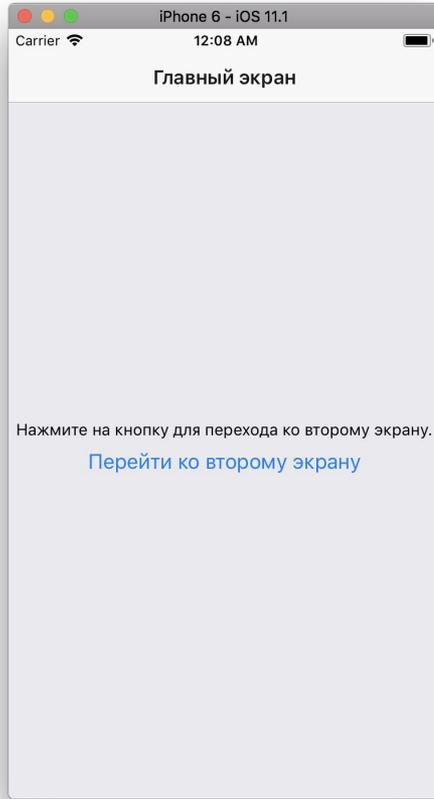
# Стили и пара констант

```
12  const tick = new Sound('metronome.mp3', Sound.MAIN_BUNDLE, (error) => { return; });
13  //...
14
15  const styles = StyleSheet.create({
16    container: {
17      flex: 1,
18      justifyContent: 'center',
19      alignItems: 'center',
20      backgroundColor: '#F5FCFF',
21    },
22    //...
23  });
```

## Основной код приложения

```
25 export default class Metronome extends Component {
26   constructor(props) {
27     super(props);
28     //...
29   }
30
31   //...
32
33   render() {
34     return (
35       <View style={styles.container}>
36         <Text style={styles.switchLabel}>
37           Off <Switch
38             value={!this.state.boom}
39             onChange={this.toggleMetronome}
40           /> On
41         </Text>
42         <Text style={styles.tempo}>
43           Tempo: {this.state.tempo}
44         </Text>
45         <Slider
46           style={{ width: 300 }}
47           step={1}
48           minimumValue={minTempo}
49           maximumValue={maxTempo}
50           value={this.state.tempo}
51           onChange={val => this.setState({ tempo: val })}
52         />
53       </View>
54     );
55   }
56 }
57
58 AppRegistry.registerComponent('Metronome', () => Metronome);
```

# Пример многоэкранного приложения



```
1 import React from 'react';
2 import { View, Button, Text } from 'react-native';
3 import { StackNavigator } from 'react-navigation';
4
5 const HomeScreen = ({ navigation }) => (
6   <View style={{ flex: 1, alignItems: 'center', justifyContent: 'center' }}>
7     <Text>Нажмите на кнопку для перехода ко второму экрану.</Text>
8     <Button
9       onPress={() => navigation.navigate('Second')}
10      title="Перейти ко второму экрану"
11    />
12   </View>
13 );
14
15 const SecondScreen = () => (
16   <View style={{ flex: 1, alignItems: 'center', justifyContent: 'center' }}>
17     <Text>Второй экран.</Text>
18     <Text>Вы можете вернуться на главный экран нажав на кнопку "Back".</Text>
19   </View>
20 );
21
22 const RootNavigator = StackNavigator({
23   Home: {
24     screen: HomeScreen,
25     navigationOptions: {
26       headerTitle: 'Главный экран',
27     },
28   },
29   Second: {
30     screen: SecondScreen,
31     navigationOptions: {
32       headerTitle: 'Второй экран',
33     },
34   },
35 });
36
37 export default RootNavigator;
38
```

# Взаимодействие с сетью

## Fetch

```
componentDidMount() {  
  fetch('https://some.url/items.json')  
    .then(response => response.json())  
    .then((data) => {  
      if (data && data.length) {  
        this.setState({  
          items: data,  
        });  
      } else {  
        this.setState({  
          items: [],  
        });  
      }  
    });  
}
```

## Другие

- WebSocket
- XMLHttpRequest
- Frisbee
- Axios

# Тестирование и отладка

```
iPhone 6 - iOS 11.1

Invalid YGJustify 'top'. should be one of: (
  center,
  "flex-end",
  "flex-start",
  "space-around",
  "space-between"
)

RCTConvertEnumValue
RCTConvert.m:250

+[RCTConvert YGJustify:]

__49-[RCTComponentData createPropBlock:i
sShadowView:]_block_invoke.123

__49-[RCTComponentData createPropBlock:i
sShadowView:]_block_invoke.167

__49-[RCTComponentData propBlockForKey:i
sShadowView:]_block_invoke_2

RCTPerformBlockWithLogFunction

RCTPerformBlockWithLogPrefix

__49-[RCTComponentData propBlockForKey:i
sShadowView:]_block_invoke

__49-[RCTComponentData propBlockForKey:i
sShadowView:]_block_invoke

Dismiss (ESC)  Reload JS (⌘R)  Copy (⌘C)
```

- Jest как средство написания unit

## ТЕСТОВ

- Call стек

- Chrome dev tools

# Ограничения

- iOS > 7, Android > 4.1
- Есть зависимые от платформы свойства компонентов
- Есть компоненты доступные только для определённых платформ
- При отсутствии необходимого компонента придётся самому писать Bridge

# ССЫЛКИ

<https://reactjs.org>

<https://facebook.github.io/react-native>

<https://github.com/reactjs/redux>

<https://github.com/react-community/react-navigation>

<https://expo.io>

# Спасибо!

[www.firstlinesoftware.ru](http://www.firstlinesoftware.ru)