

# REACT NATIVE

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

# MI AZ A REACT NATIVE?

- Mobil alkalmazások fejlesztése
- Cross-platform: Android, iOS
- Javascript + React
- Components, props, states, lifecycles
- Tényleges natív kód (~~WebView, Hybrid app~~)
- Nyilt forráskód (3-Clause BSD License)



## REACT NATIVE ELŐNYESI...

- Cross-platform
- SOKKAL olcsóbb
- Egyszerűbb kód
- Sok idő megspórolása

# Kik használják?



Facebook

Facebook Ads  
Manager

Instagram



Tesla



Airbnb



Skype

# REACT

- Virtual dom
- Declarative (JSX)
- Component-Based
- Learn Once, Write Anywhere
- 5,580,072 npm downloads in the last month

# COMPONENTS AND PROPS

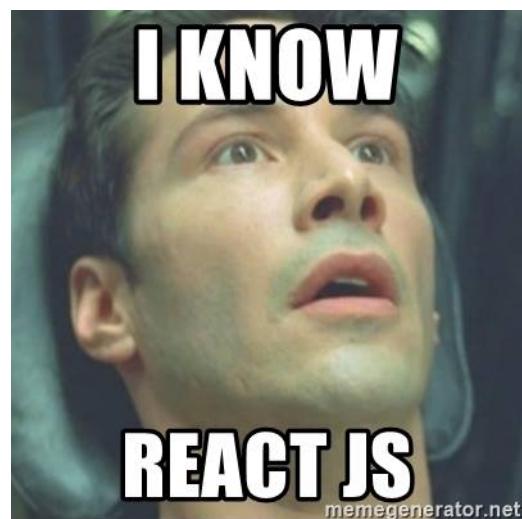
The screenshot shows a development interface for React Native. On the left, there is a "LIVE PREVIEW" window which is currently empty. On the right, there is a "SOURCE CODE" window containing the following JavaScript code:

```
// Everything extends React.Component
/*
 */
//render(<MyComponent />, "root");
```

# STATE AND LIFECYCLE

The screenshot shows a development interface for React Native. On the left, there is a "LIVE PREVIEW" window which is currently blank. On the right, there is a "SOURCE CODE" window containing the following JavaScript code:

```
class Timer extends React.Component {
  constructor(props) {
    super(props);
  }
  tick() {
  }
  componentDidMount() {
  }
  render() {
    return (
      <div>
        Seconds:
      </div>
    );
  }
}
//render(<Timer />, "root");
```



# REACT NATIVE

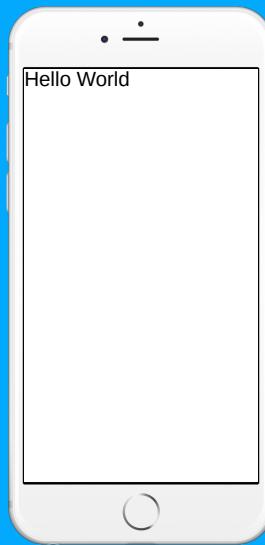
- Community-driven
- Maximum code reuse & cost saving  
"90%"
- Live reload
- Strong performance for mobile environments
- Use native code when you need to
- Modular and intuitive architecture similar to React

## BASE COMPONENTS

```
1 import React, { Component } from 'react'
2 import { AppRegistry, StyleSheet, Text, View } from 'react-native'
3
4 class App extends Component {
5   render() {
6     return (
7       <View>
8         <Text>
9           Hello World
10        </Text>
11      </View>
12    )
13  }
14}
15
16 AppRegistry.registerComponent('App', () => App)
17
```

No Errors

Show Details

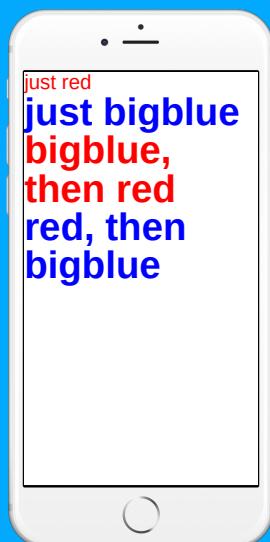


## STYLE

```
1 import React, { Component } from 'react';
2 import { AppRegistry, StyleSheet, Text, View } from 'react-native';
3
4 export default class LotsOfStyles extends Component {
5   render() {
6     return (
7       <View>
8         <Text style={styles.red}>just red</Text>
9         <Text style={styles.bigblue}>just bigblue</Text>
10        <Text style={[styles.bigblue, styles.red]}>bigblue, then red</Text>
11        <Text style={[styles.red, styles.bigblue]}>red, then bigblue</Text>
12     </View>
13   );
14 }
15
16 const styles = StyleSheet.create({
17   bigblue: {
18     color: 'blue',
19   }
}
```

No Errors

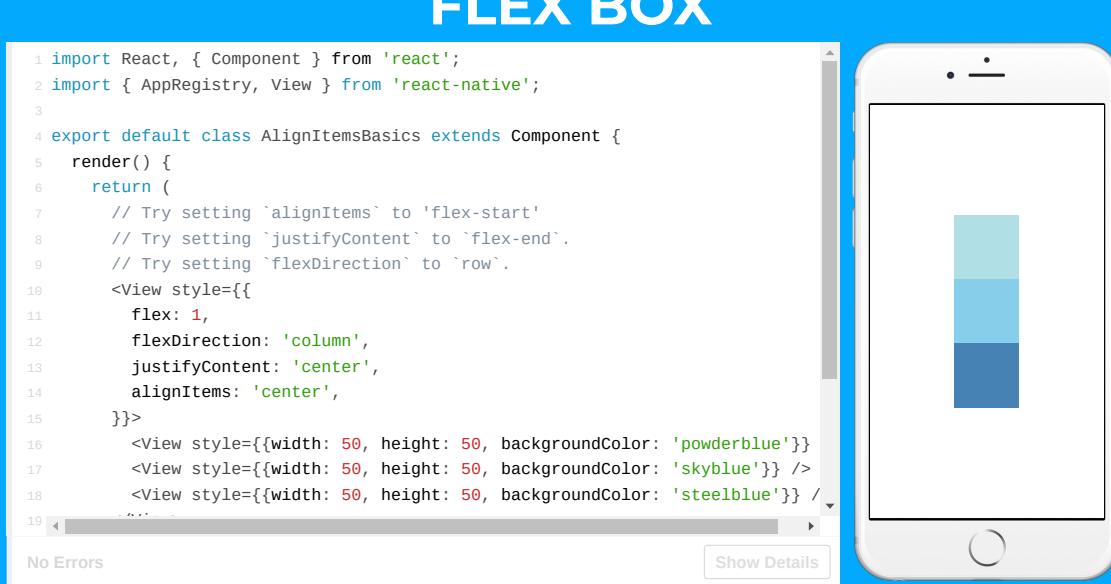
Show Details



## FLEX BOX

```
1 import React, { Component } from 'react';
2 import { AppRegistry, View } from 'react-native';
3
4 export default class AlignItemsBasics extends Component {
5   render() {
6     return (
7       // Try setting `alignItems` to `flex-start`
8       // Try setting `justifyContent` to `flex-end`.
9       // Try setting `flexDirection` to `row`.
10      <View style={{
11        flex: 1,
12        flexDirection: 'column',
13        justifyContent: 'center',
14        alignItems: 'center',
15      }}>
16        <View style={{width: 50, height: 50, backgroundColor: 'powderblue'}}>
17          <View style={{width: 50, height: 50, backgroundColor: 'skyblue'}} />
18          <View style={{width: 50, height: 50, backgroundColor: 'steelblue'}} />
19        </View>
20      </View>
21    );
22  }
23}
```

No Errors Show Details

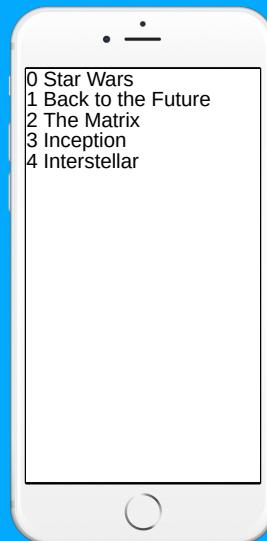


# NETWORKING

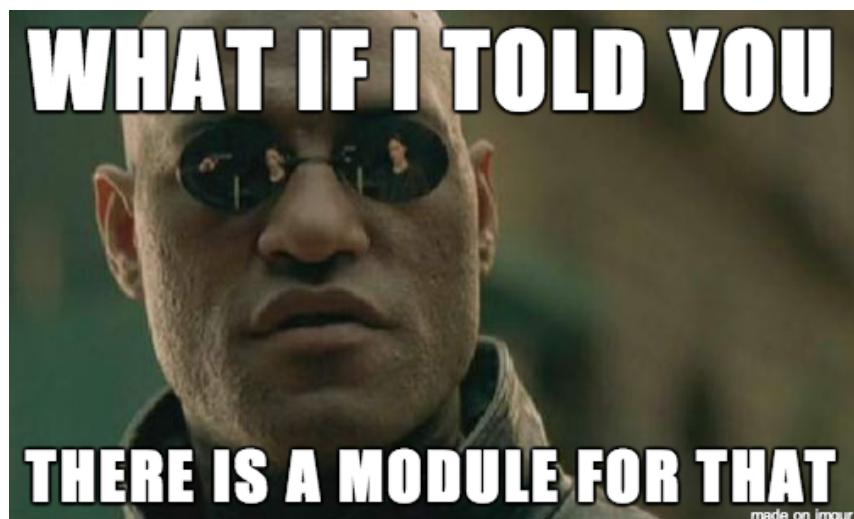
```
1 import React, { Component } from 'react';
2 import { AppRegistry, View, StyleSheet, Text } from 'react-native';
3
4 export default class FetchBasics extends Component {
5   constructor(props){
6     super(props);
7     this.state = {
8       movieData: []
9     }
10  }
11
12  componentDidMount() {
13    return fetch('https://facebook.github.io/react-native/movies.json')
14      .then((response) => response.json())
15      .then((responseJson) => {
16        let movieData = [];
17        responseJson.movies.forEach((item, idx) => {
18          movieData.push(<Text> {idx} {item.title} </Text>);
19        });
20      });
21  }
22}
```

No Errors

Show Details

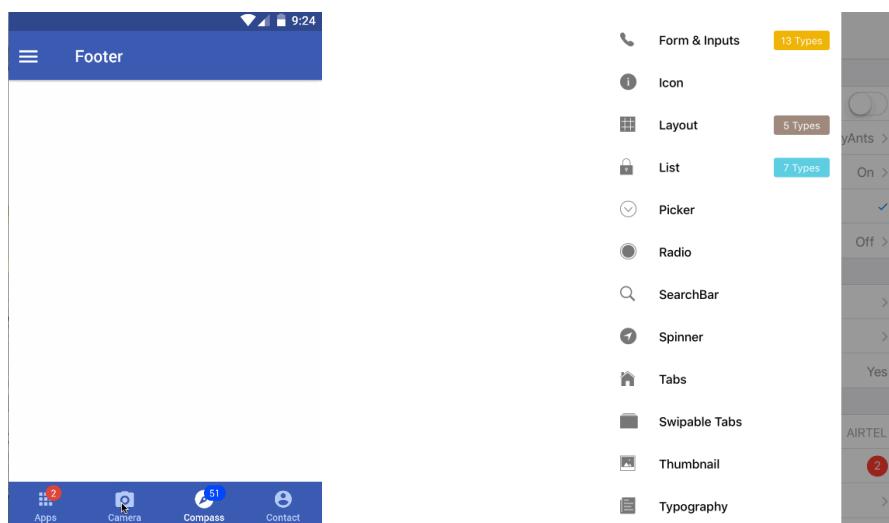


**LEHET EZ MÉG EGYSZERŰBB?  
KÜLSŐ KOMPONENS CSOMAGOK...**



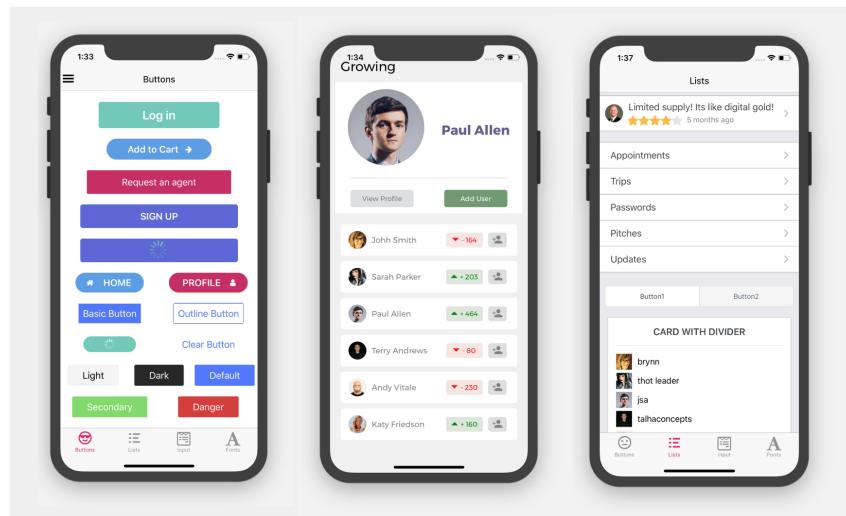
# NativeBase

```
$ npm install native-base --save
```



# React Native Elements

```
$ npm install react-native-elements --save
```



# Navigáció a képernyők között



# React Navigation

```
$ npm install --save react-navigation
```

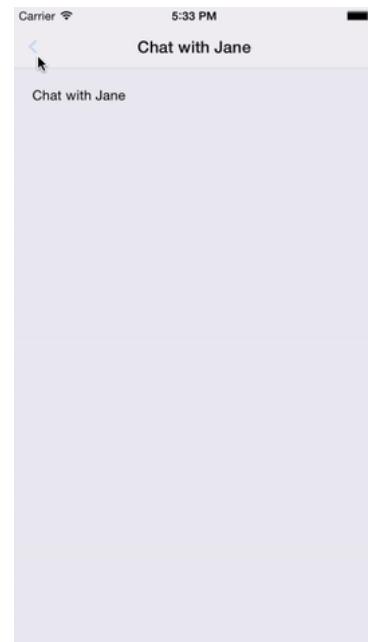
## BasicApp.js

```
import { StackNavigator } from 'react-navigation';

const BasicApp = StackNavigator({
  Main: {screen: MainScreen},
  Profile: {screen: ProfileScreen},
});
```

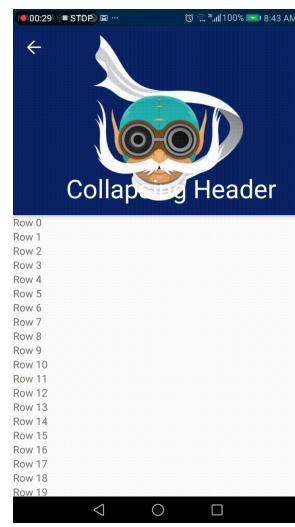
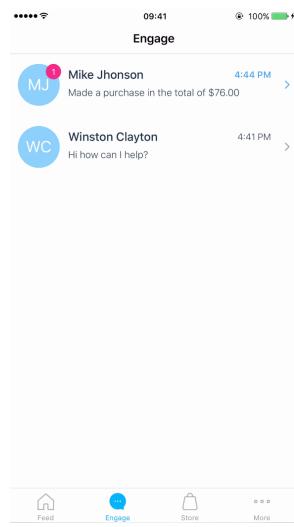
## MainScreen.js

```
class MainScreen extends React.Component {
  static navigationOptions = { title: 'Welcome' };
  render() {
    const { navigate } = this.props.navigation;
    return ( <Button onPress={() => navigate('Profile', {
      name: 'Jane'
    })} /> );
  }
}
```



# React Native Navigation

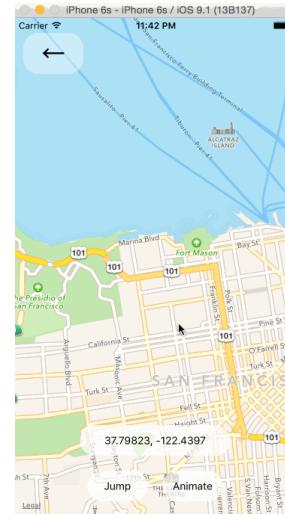
```
$ npm install react-native-navigation@latest
```



# react-native-maps

```
$ npm install react-native-maps --save
```

```
<MapView  
    initialRegion={{  
      latitude: 37.78825,  
      longitude: -122.4324,  
      latitudeDelta: 0.0922,  
      longitudeDelta: 0.0421,  
    }}  
/>
```



# **HOGYAN ÁLLJUNK NEKI ?**

# CREATE-REACT-NATIVE-APP

Create a React Native app on any OS with no build config.

```
$ npm install -g create-react-native-app  
$ create-react-native-app my-app  
$ cd my-app/  
$ npm start
```

# Expo

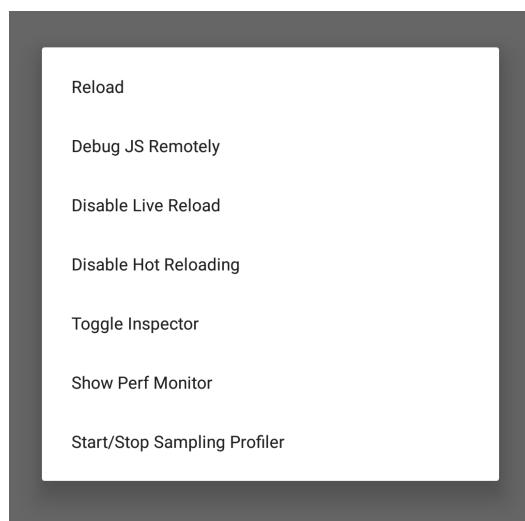
The Expo client app for Android and iOS.



# DEBUGGING

- Live Reload
- Hot Reload
- In-app Errors and Warnings
- React Native Inspector
- Performance Monitor
- Debug Remotely - Chrome Developer Tools

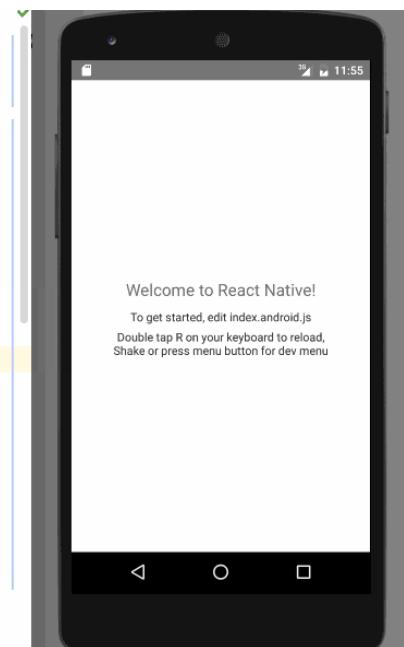
# FEJLESZTŐI MENÜ



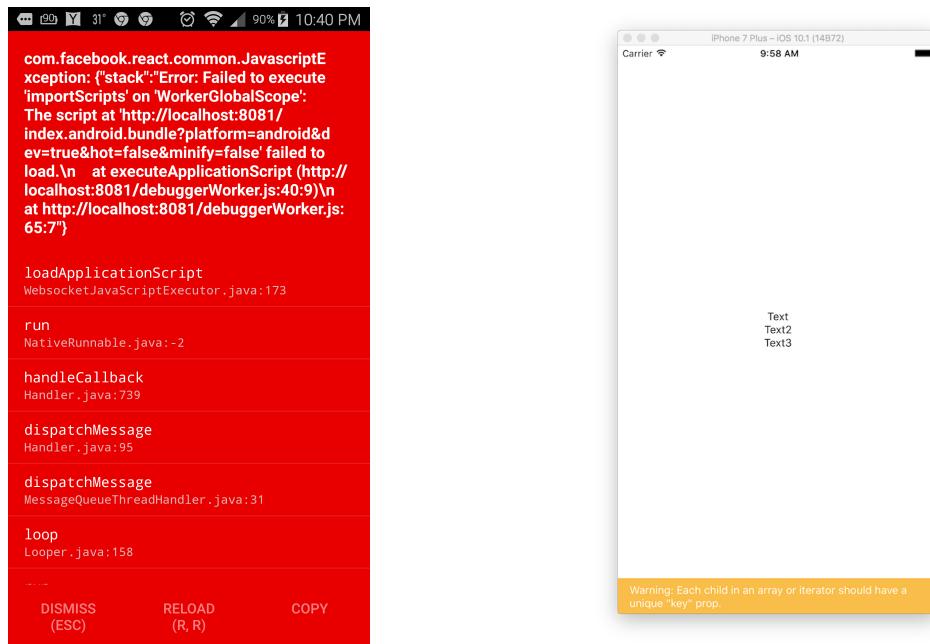
# LIVE RELOAD, HOT RELOAD

```
import React, { Component } from 'react';
import {
  AppRegistry,
  StyleSheet,
  Text,
  View
} from 'react-native';

export default class AwesomeProject extends Component {
  render() {
    return (
      <View style={styles.container}>
        <Text style={styles.welcome}>
          Welcome to React Native!
        </Text>
        <Text style={styles.instructions}>
          To get started, edit index.android.js
        </Text>
        <Text style={styles.instructions}>
          Double tap R on your keyboard to reload,{'\n'}
          Shake or press menu button for dev menu
        </Text>
      </View>
    );
  }
}
```



# IN-APP ERRORS AND WARNINGS



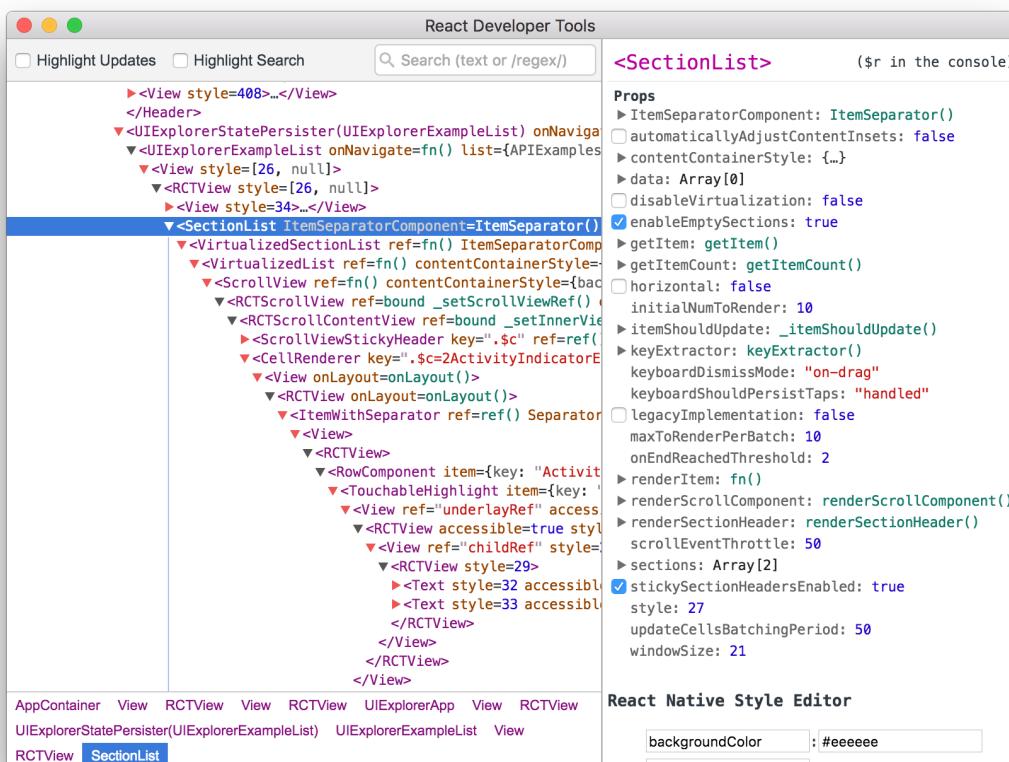
# REACT NATIVE INSPECTOR



# PERFORMANCE MONITOR

RAM	JSC	Views	UI	JS
71.02	0.00	0	24	24
MB	MB	0		

## DEBUG REMOTELY - CHROME DEVELOPER TOOLS



# NATIVE MODULES

- Hardware
- Performance
- Wrap an existing Java/iOS API
- Hybrid Native/React-Native Apps

# EXAMPLE: THE TOAST MODULE

```
// Extend: com.facebook.react.bridge.ReactContextBaseJavaModule
public class ToastModule extends ReactContextBaseJavaModule {

    private static final String DURATION_SHORT_KEY = "SHORT";
    private static final String DURATION_LONG_KEY = "LONG";

    public ToastModule(ReactApplicationContext reactContext) {
        super(reactContext);
    }

    // implement getName method so that we can access it through React.NativeModules.ToastExample in JavaScript.
    @Override
    public String getName() {
        return "ToastExample";
    }

    // Optional
    // very useful to key pre-defined values that need to be communicated from JavaScript to Java in sync.
    @Override
    public Map<String, Object> getConstants() {
        final Map<String, Object> constants = new HashMap<>();
        constants.put(DURATION_SHORT_KEY, Toast.LENGTH_SHORT);
        constants.put(DURATION_LONG_KEY, Toast.LENGTH_LONG);
        return constants;
    }

    @ReactMethod
    public void show(String message, int duration) {
        Toast.makeText(getApplicationContext(), message, duration).show();
    }
}
```

## Argument Types (@ReactMethod)

JavaScript	Native
Bool	Boolean
Number	Integer
Number	Double
Number	Float
String	String
function	Callback
Object	ReadableMap
Array	ReadableArray

## EXAMPLE: REGISTER THE MODULE

```
public class AnExampleReactPackage implements ReactPackage {  
  
    @Override  
    public List<ViewManager> createViewManagers(ReactApplicationContext reactContext) {  
        return Collections.emptyList();  
    }  
  
    @Override  
    public List<NativeModule> createNativeModules( ReactApplicationContext reactContext) {  
        List<NativeModule> modules = new ArrayList<>();  
  
        modules.add(new ToastModule(reactContext));  
  
        return modules;  
    }  
}
```

# EXAMPLE: USING FROM JS

```
/**  
 * This exposes the native ToastExample module as a JS module. This has a  
 * function 'show' which takes the following parameters:  
 *  
 * 1. String message: A string with the text to toast  
 * 2. int duration: The duration of the toast. May be ToastExample.SHORT or  
 *    ToastExample.LONG  
 */  
import {NativeModules} from 'react-native';  
module.exports = NativeModules.ToastExample;
```

```
import ToastExample from './ToastExample';  
  
ToastExample.show('Awesome', ToastExample.SHORT);
```

# PLATFORM FÜGGŐ ELEMÉK

## PLATFORM-SPECIFIC COMPONENTS

```
> CustomComponent.ios.js  
> CustomComponent.android.js
```

```
// Note - No extension specified  
import CustomComponent from './CustomComponent';
```

# PLATFORM-SPECIFIC STYLES

```
import { Platform, StyleSheet } from 'react-native';

const styles = StyleSheet.create({
  paddingTop: (Platform.OS === 'ios') ? 30 : 0,
});
```

# UNIT TESTS: JEST

- Snapshot Testing
- Zero configuration (.test.js, \_\_tests\_\_ )
- Built-in code coverage reports

```
PASS app/components/custom/add_issue_flow/__tests__/back_to_issues_button.spec.js (10.745s)
PASS app/components/custom/v2/navigation_bar/__tests__/home_nav_bar.spec.js (9.719s)
PASS app/components/custom/headers/__tests__/save_icon.spec.js
PASS app/components/authentication/__tests__/enter_login.spec.js (11.603s)

Test Suites: 33 passed, 33 total
Tests: 229 passed, 229 total. Run the following command to upgrade: bun
Schemas: 229 passed, 229 total
Time: 14.782s
Ran all test suites.
```

# END-TO-END TESTS: DETOX

The screenshot shows a Mac OS X desktop environment. In the center is a terminal window titled "test — npm run e2e — npm — node · npm TMPDIR=/var/folders/0k/dm84f0\_j5cs9twb7chv". The terminal output is as follows:

```
detox info 5: Launching org.reactjs.native.example.example...
detox info 5: org.reactjs.native.example.example launched
  Sanity
    ✓ should have welcome screen (367ms)
    ✓ should show hello screen after tap (309ms)
    ✓ should show world screen after tap (310ms)

  Matchers
    ✓ should match elements by (accesibility) label (367ms)
    ✓ should match elements by (accesibility) id (316ms)
    ✓ should match elements by type (native class) (354ms)
    ✓ should match elements by accesibility trait (311ms)
    ✓ should match elements with ancestor (parent)
    ✓ should match elements with descendant (child) (39ms)
    ✓ should match elements by using two matchers together with and
      - should choose from multiple elements matching the same matcher

  Actions
    ✓ should tap on an element (1450ms)
    ✓ should long press on an element (2586ms)
```

To the right of the terminal is a sidebar titled "Choose a test" with the following options:

- Sanity
- Matchers
- Actions
- Assertions
- WaitFor
- Stress
- Switch Root
- Timeouts

# DEMO

# KÉRDÉSEK?