



React Native

GAME-CHANGER FOR MOBILE DEVELOPMENT

React Bangkok 1.0.0
August 27, 2016

PanJ

PANJAMAPONG SERMSAWATSRI



CTO at TakeMeTour

Full-stack developer

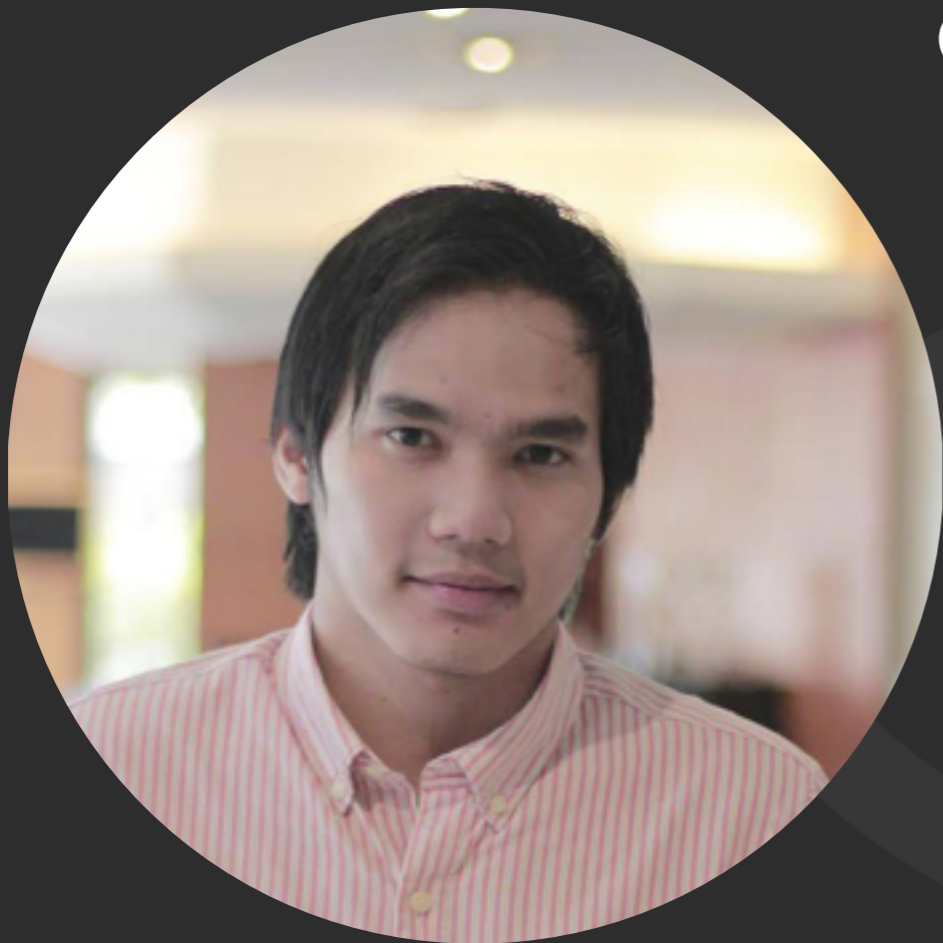
9 years on web development

DevOps beginner

日本のことは大好きだよ!

Chern

RANATCHAI CHERNBAMRUNG



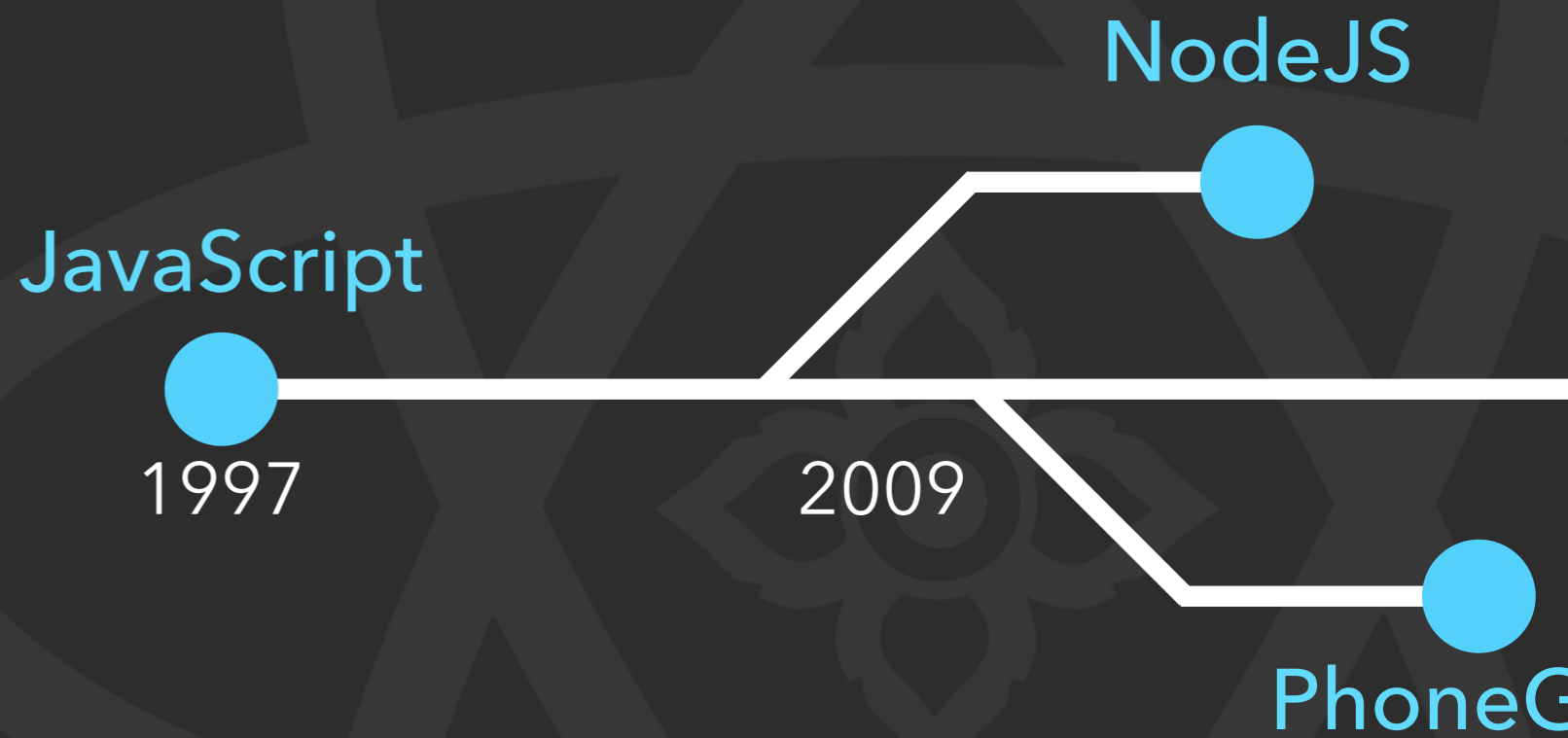
CTO at NextCover

Full-stack developer

Main stream Front-end

DevOps beginner

History of JavaScript



What is great about Web technology

Open standards

Instant distribution

Fast iteration

Unified team

What is great about Web technology

Open standards

Instant distribution

Fast iteration

Unified team

What is great about Web technology

Open standards

Instant distribution

Fast iteration

Unified team

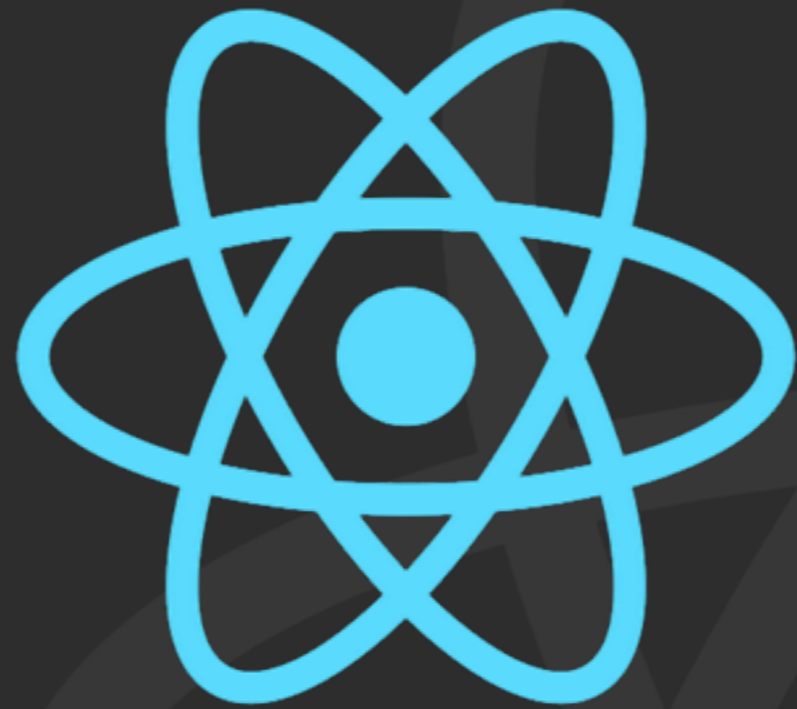
What is great about Web technology

Open standards

Instant distribution

Fast iteration

Unified team



React Native

Native

Proprietary platforms

Installed binaries

Compile cycles

Platform-specific team

React Native

Open standards

Instant distribution

Fast iteration

Unified team

Native

Proprietary platforms

Installed binaries

Compile cycles

Platform-specific team

React Native

Open standards

Instant distribution

Fast iteration

Unified team

Native

Proprietary platforms

Installed binaries

Compile cycles

Platform-specific team

React Native

Open standards

Instant distribution

Fast iteration

Unified team

Native

Proprietary platforms

Installed binaries

Compile cycles

Platform-specific team

React Native

Open standards

Instant distribution

Fast iteration

Unified team



“Using React is a Business Decision,
Not a Technology Choice”

- *Eric Baer*

React

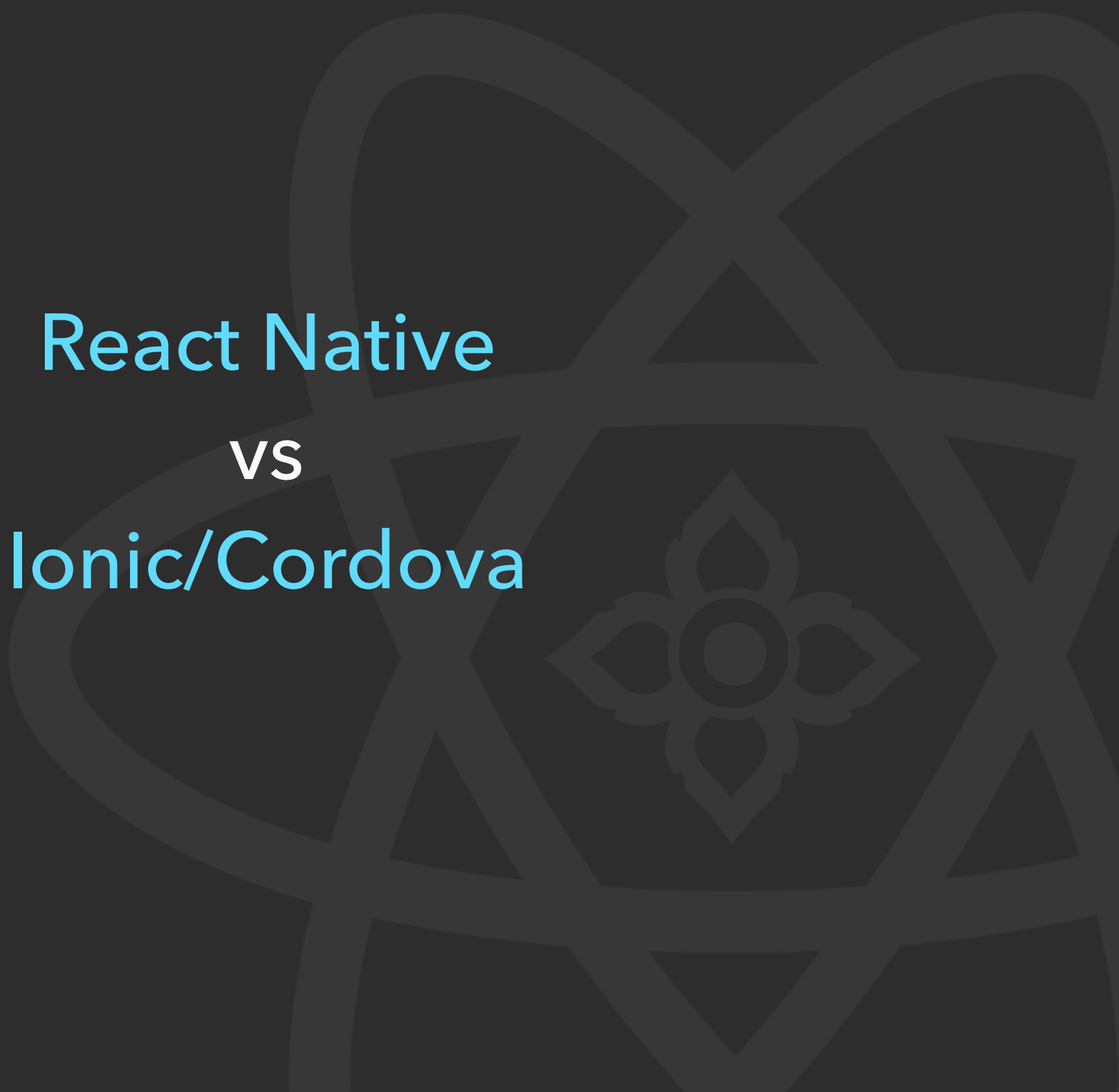
DOM <div>

```
  
</div>
```

React Native

Native Component

```
<Image source={require('logo.png')} />  
</View>
```



React Native
vs
Ionic/Cordova

	React Native	Ionic/Cordova
Hot Reloading	✓	✗
Code Push	✓	✓
View Engine	Native Components	WebView
Memory Usage	Moderate	High



React Native

VS

Native

React Native

Native

Code Sharing

✓

✗

“JavaScript will be the next universal language after English”

- ~~Mark Zuckerberg~~
didn't said



Let's get hands-on!

WORKSHOP TIME!

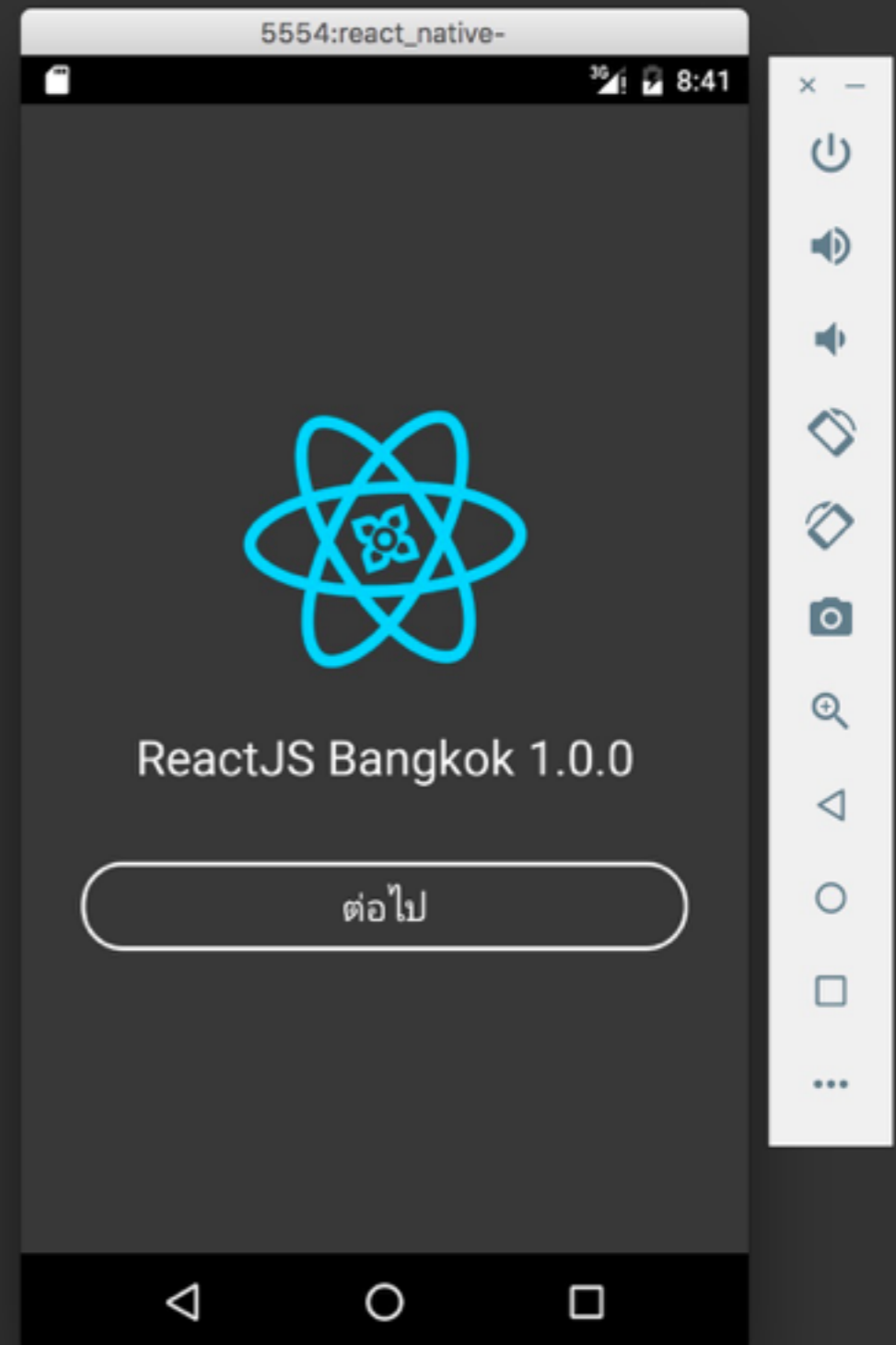
React Native In Touch

<https://github.com/ranatchai/react-native-bkk-example>

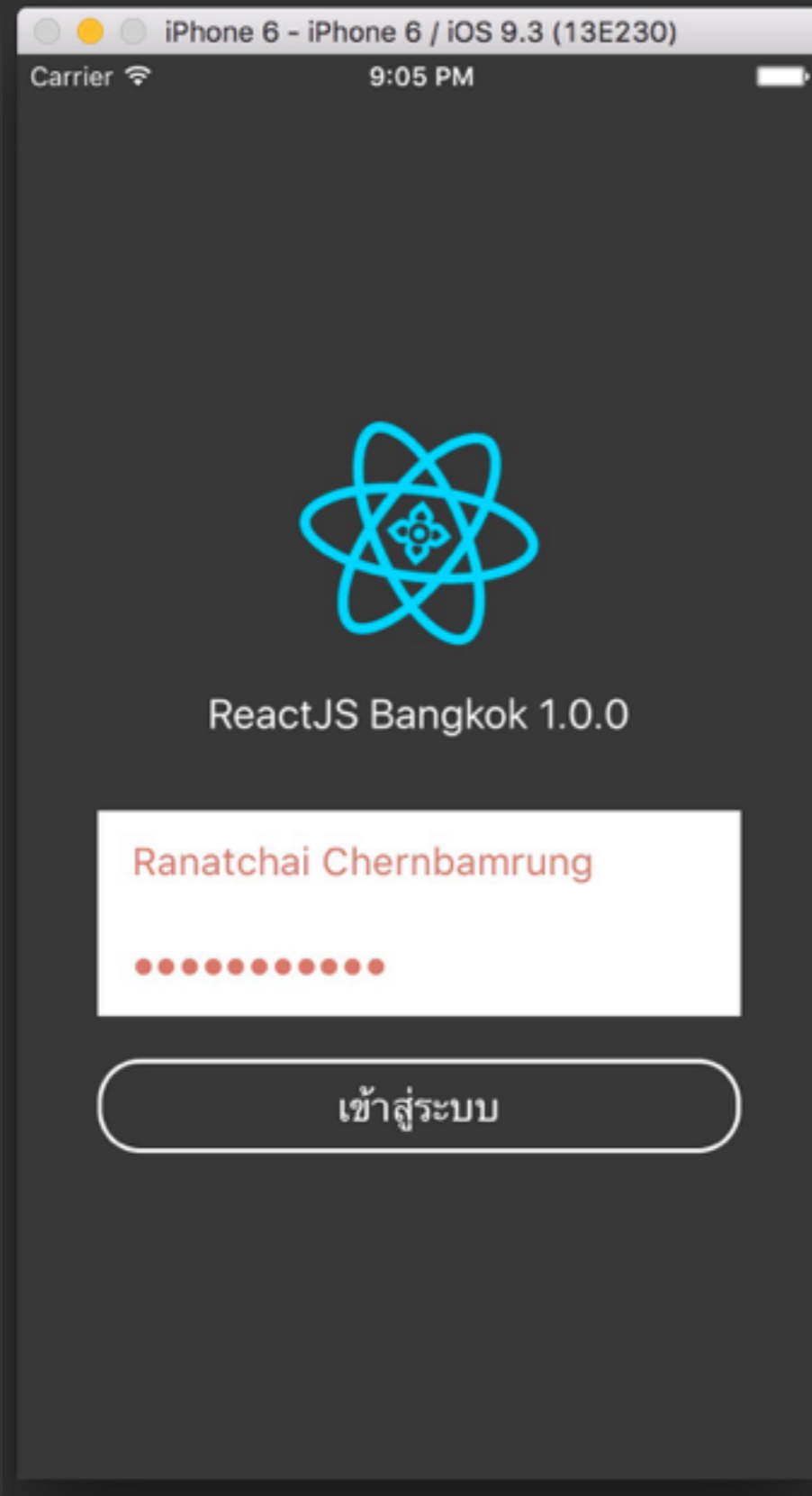
- Different Between iOS and Android
- Code Overview, Styling Component
- Dev Process
- It Great but ... ?



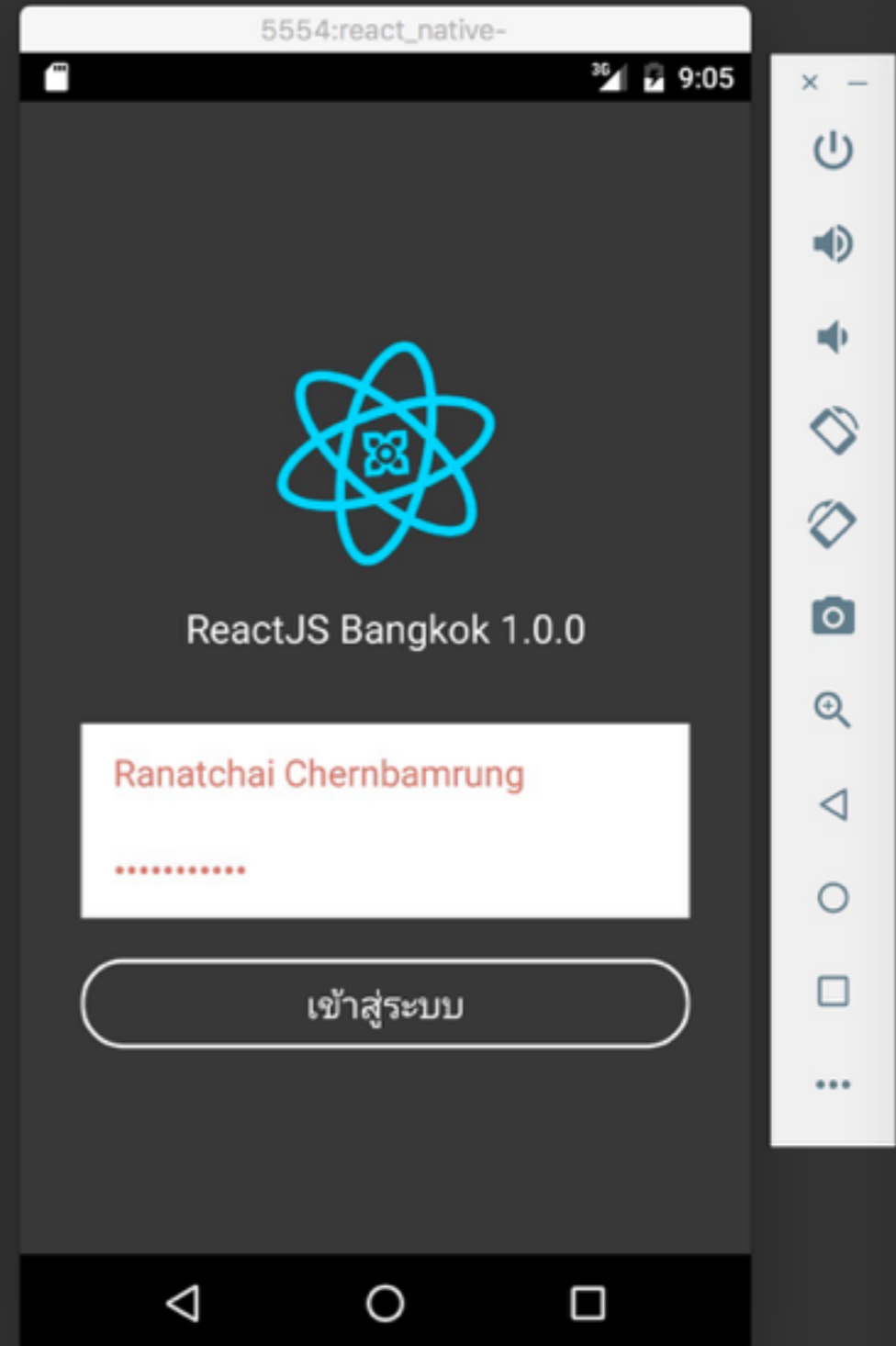
iOS



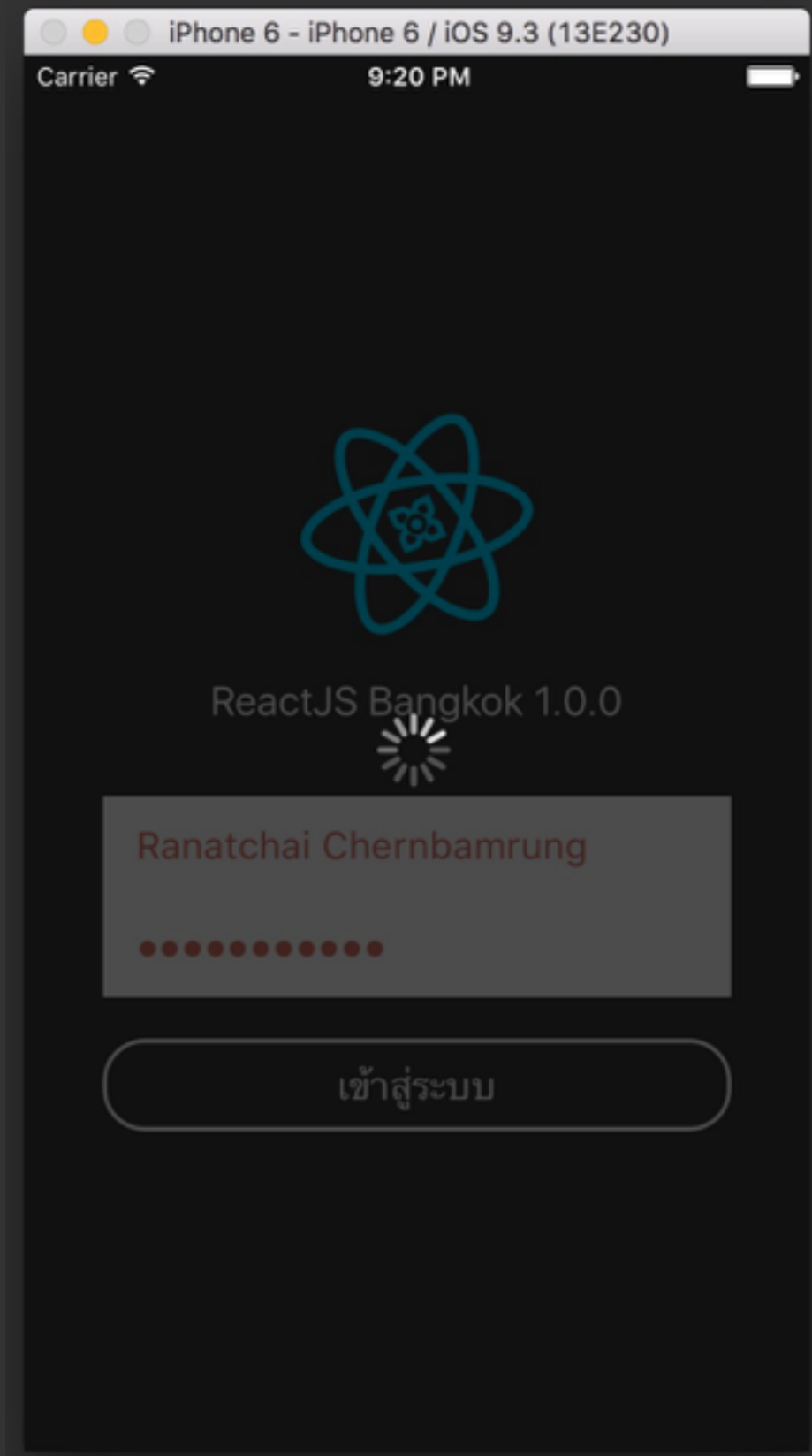
Android



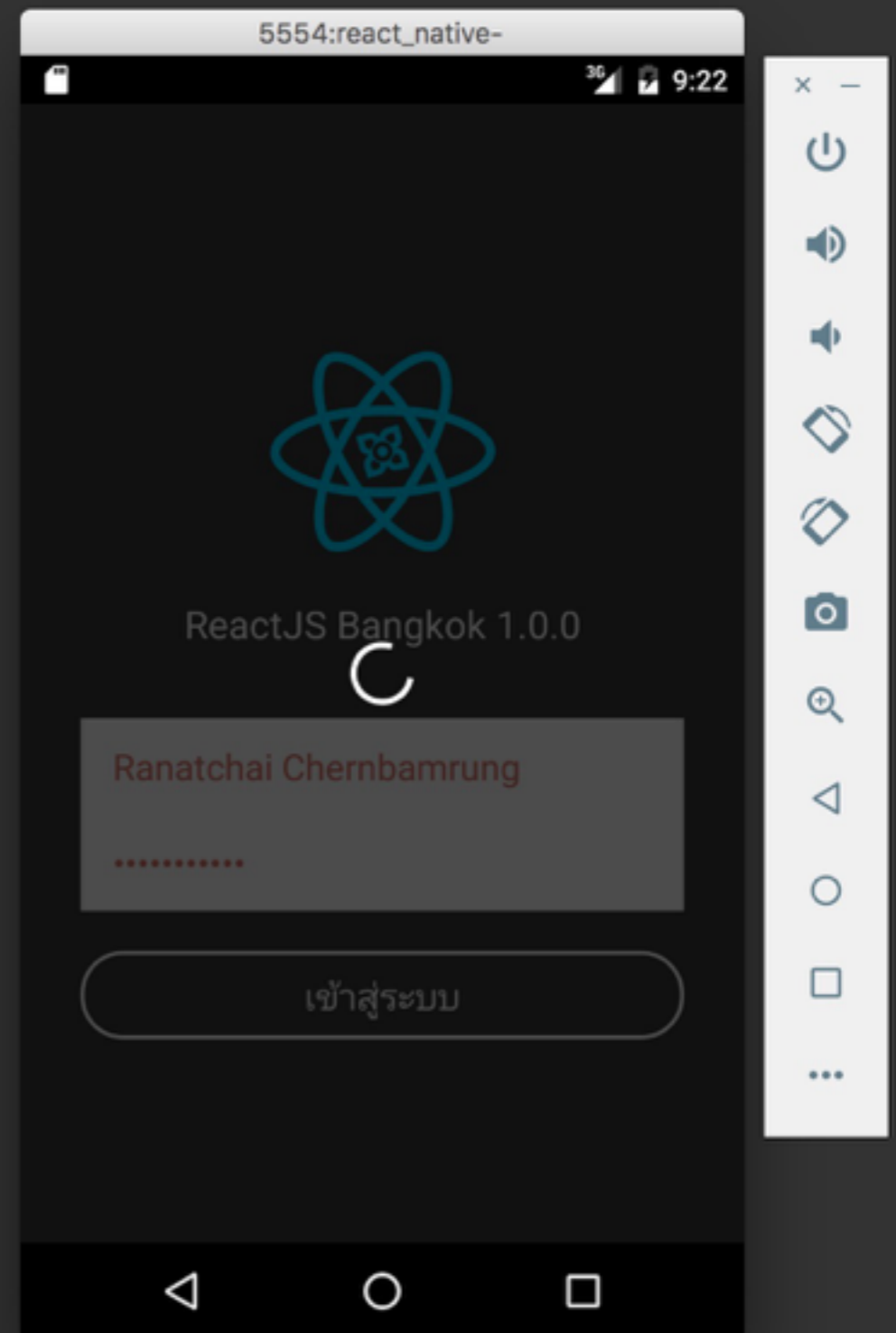
iOS



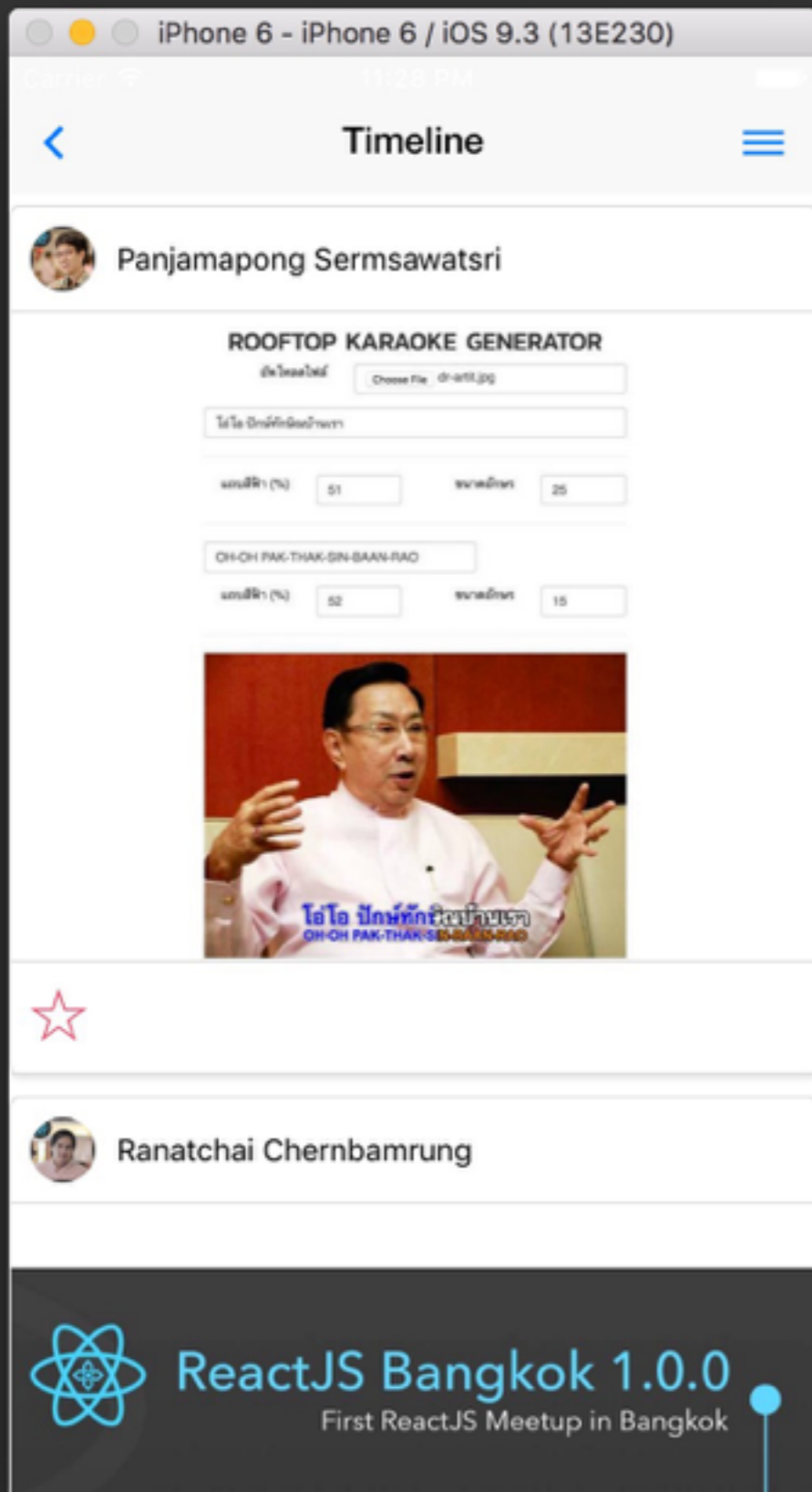
Android



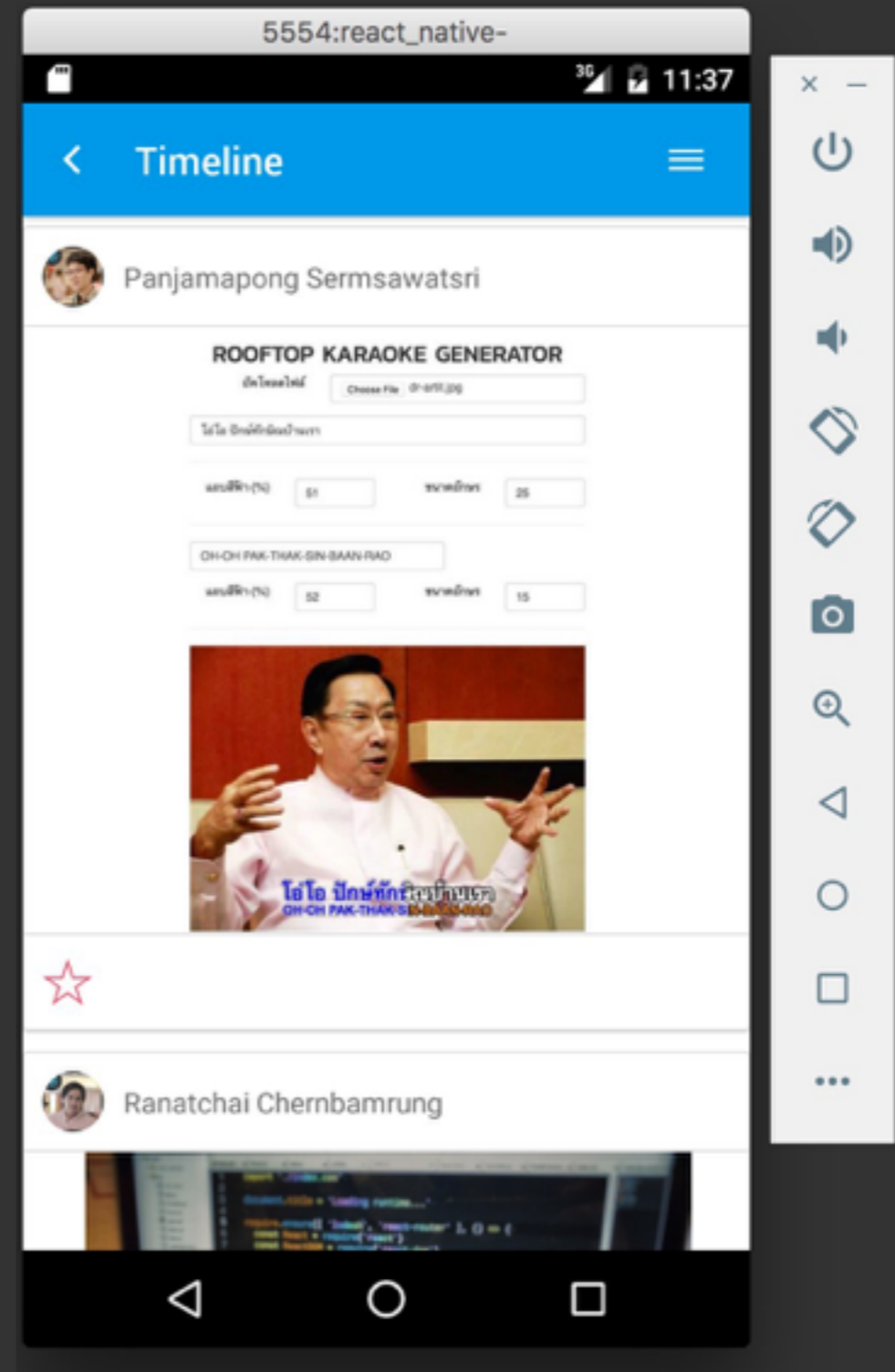
iOS



Android



iOS



Android

Thing We Learn

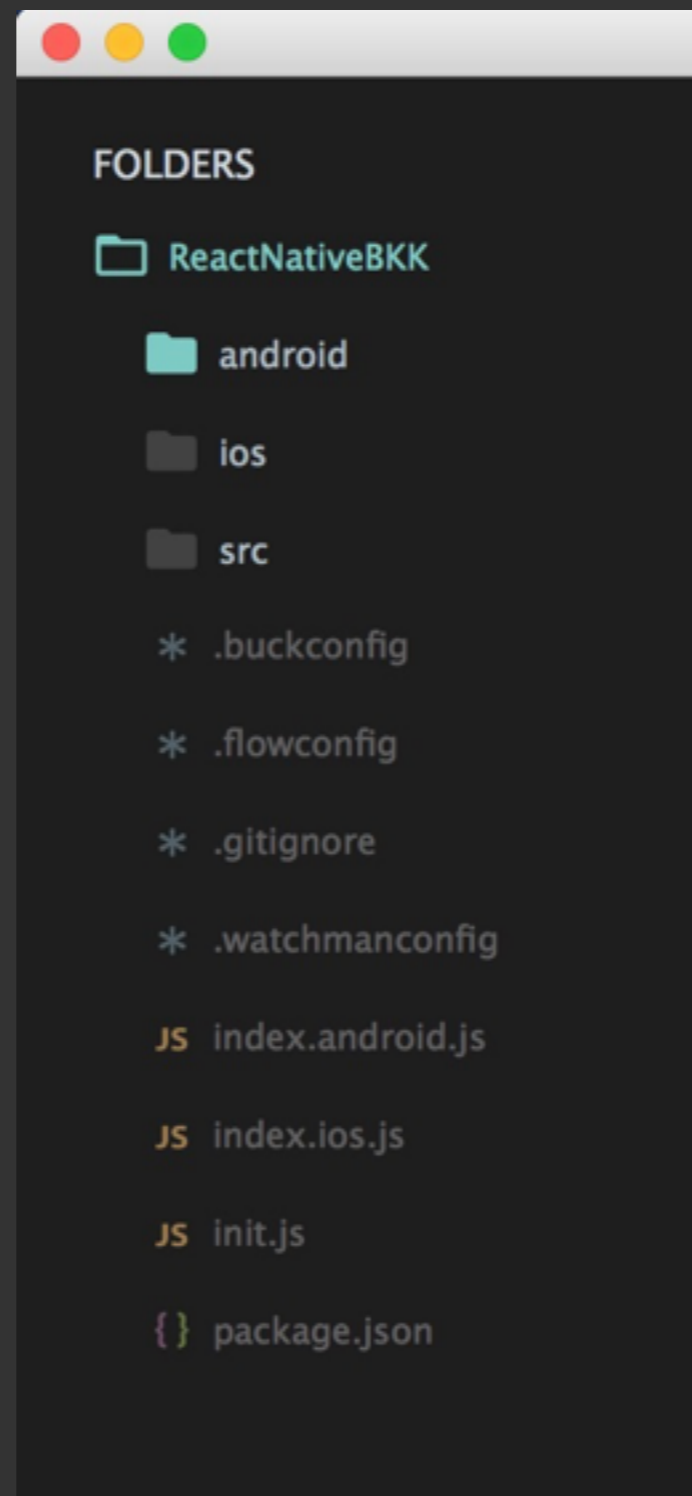
- Result in Android and iOS is Different

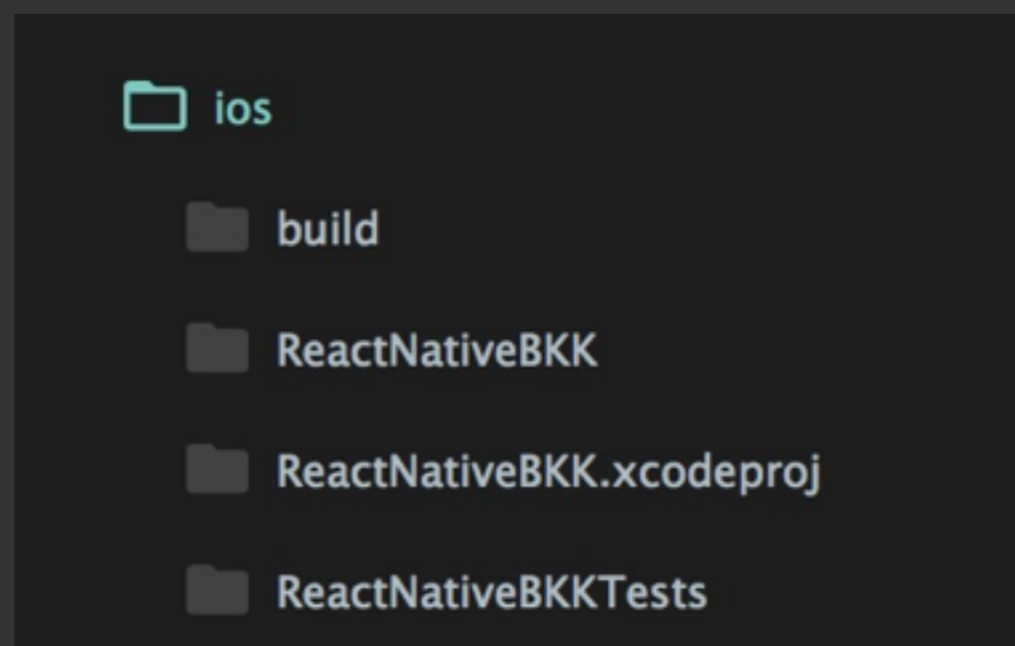
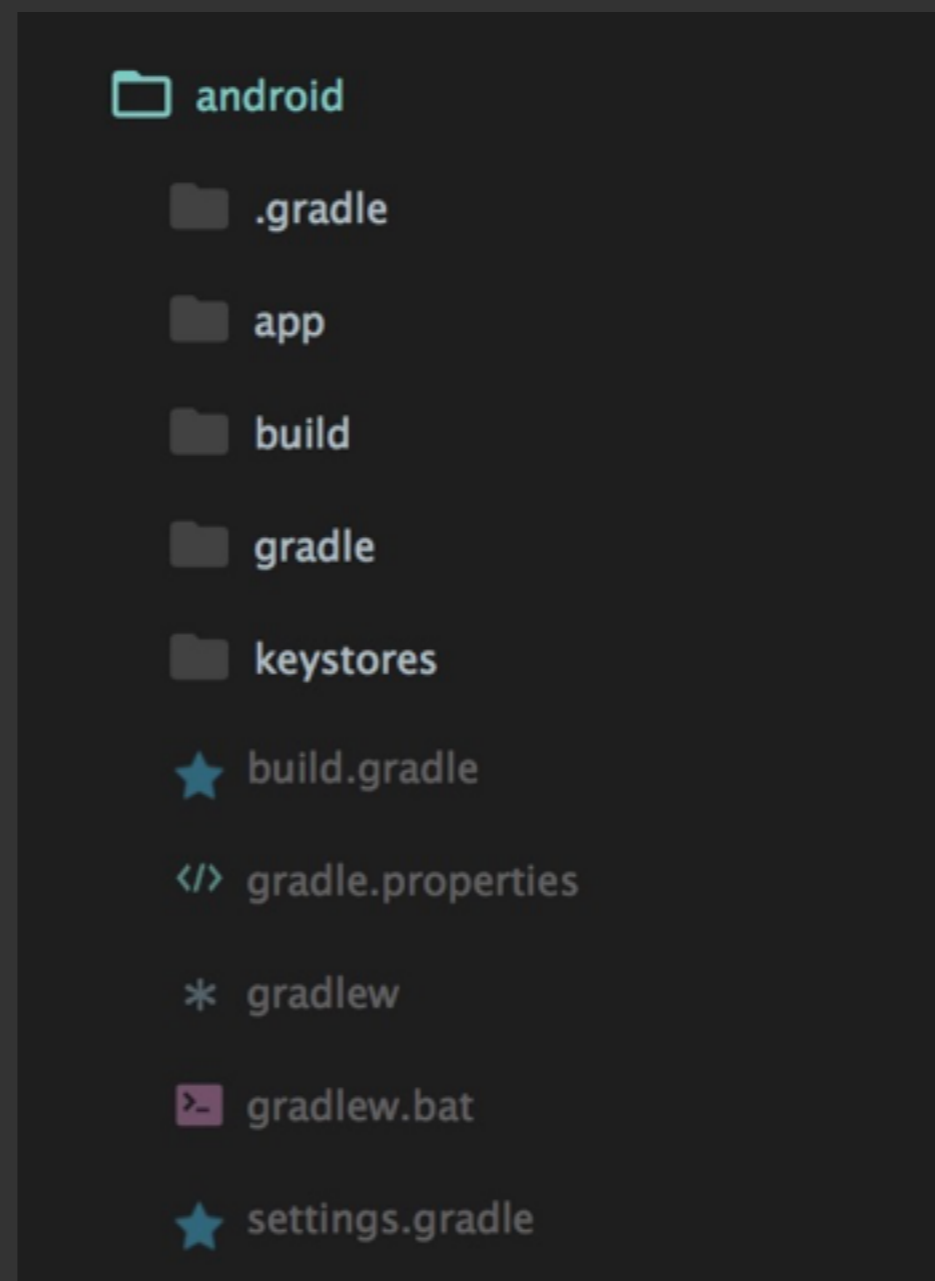
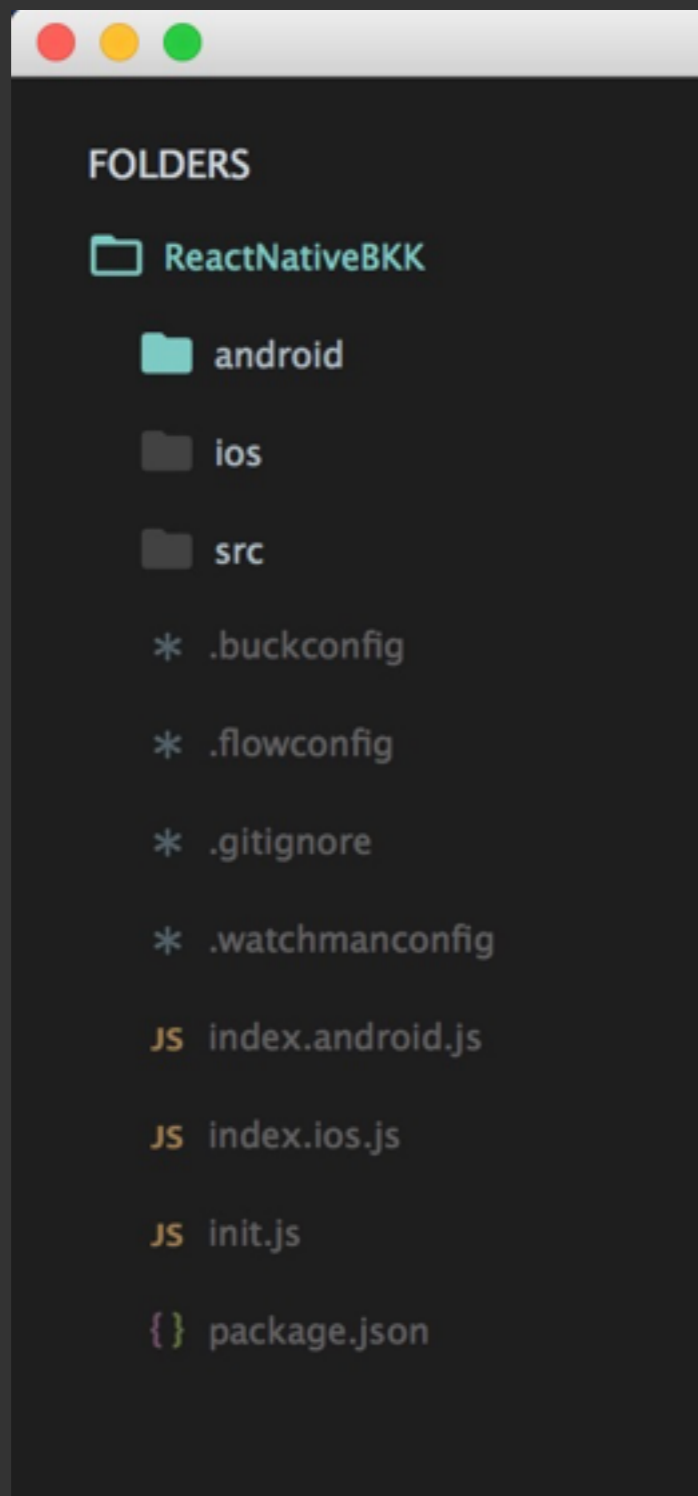
<https://github.com/ranatchai/react-native-bkk-example>

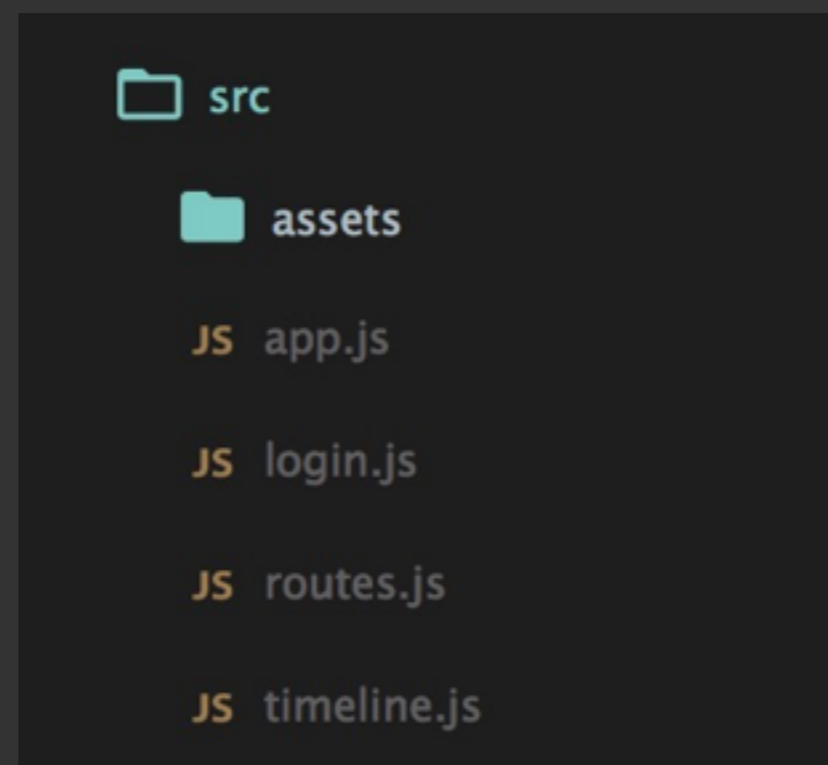
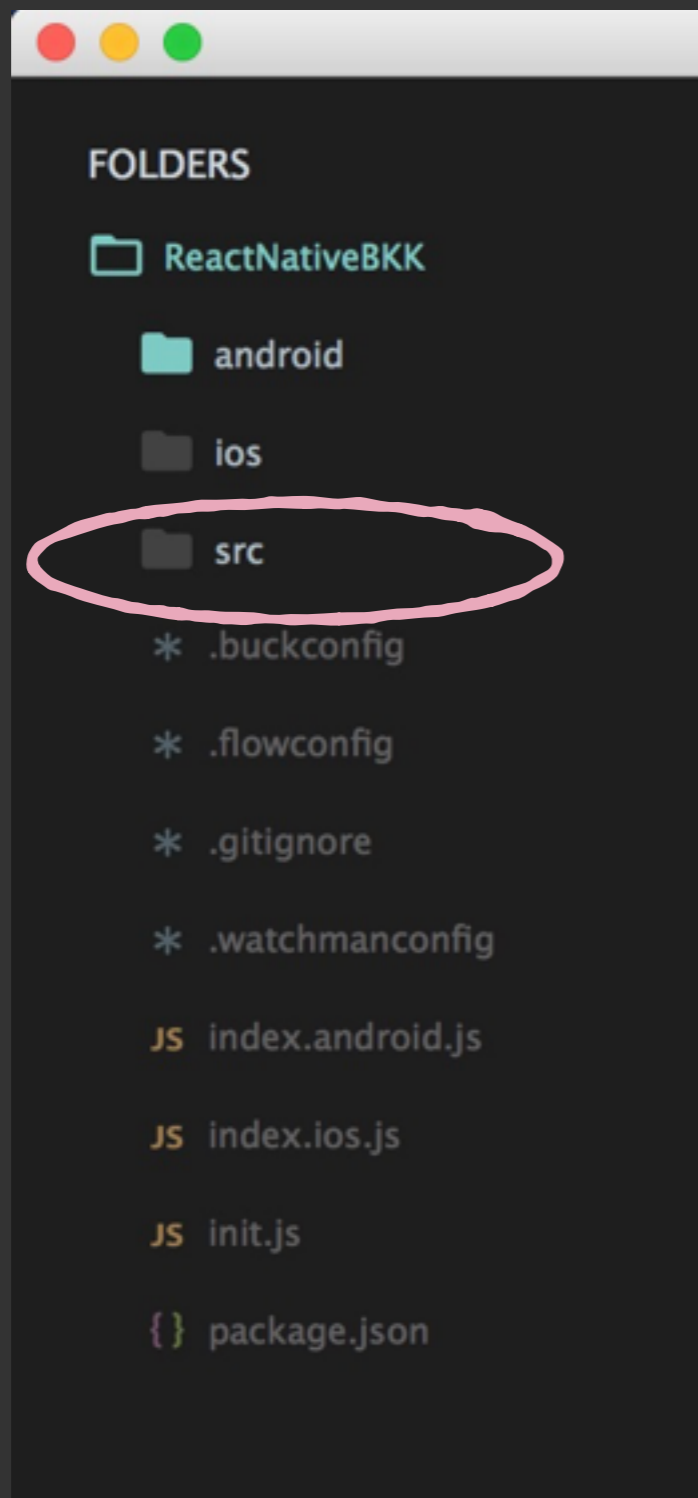
Code Overview

<https://github.com/ranatchai/react-native-bkk-example>

Project Structure



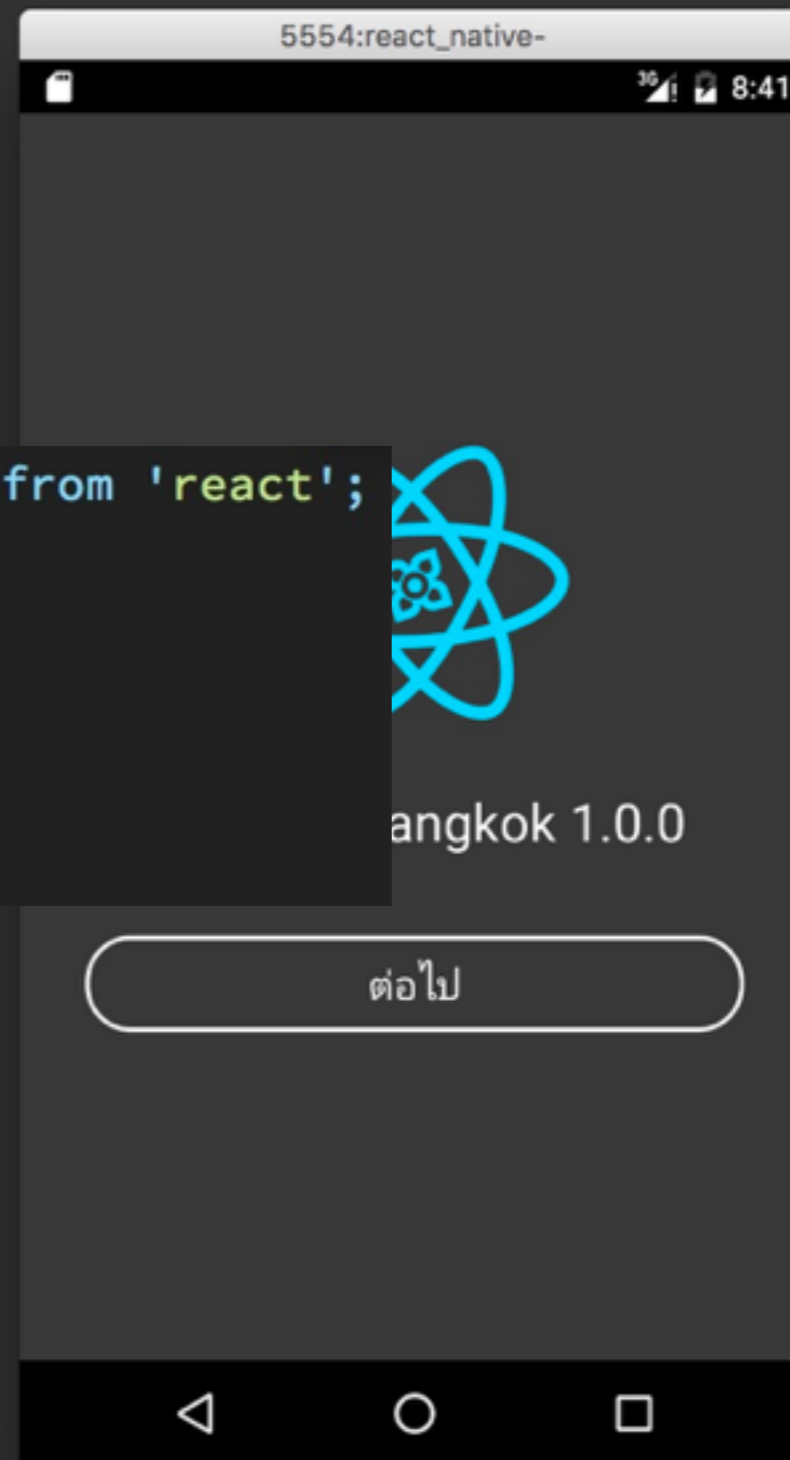




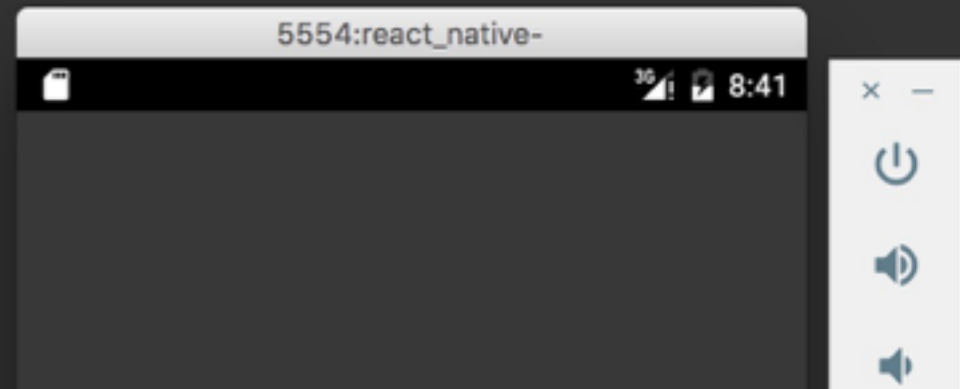
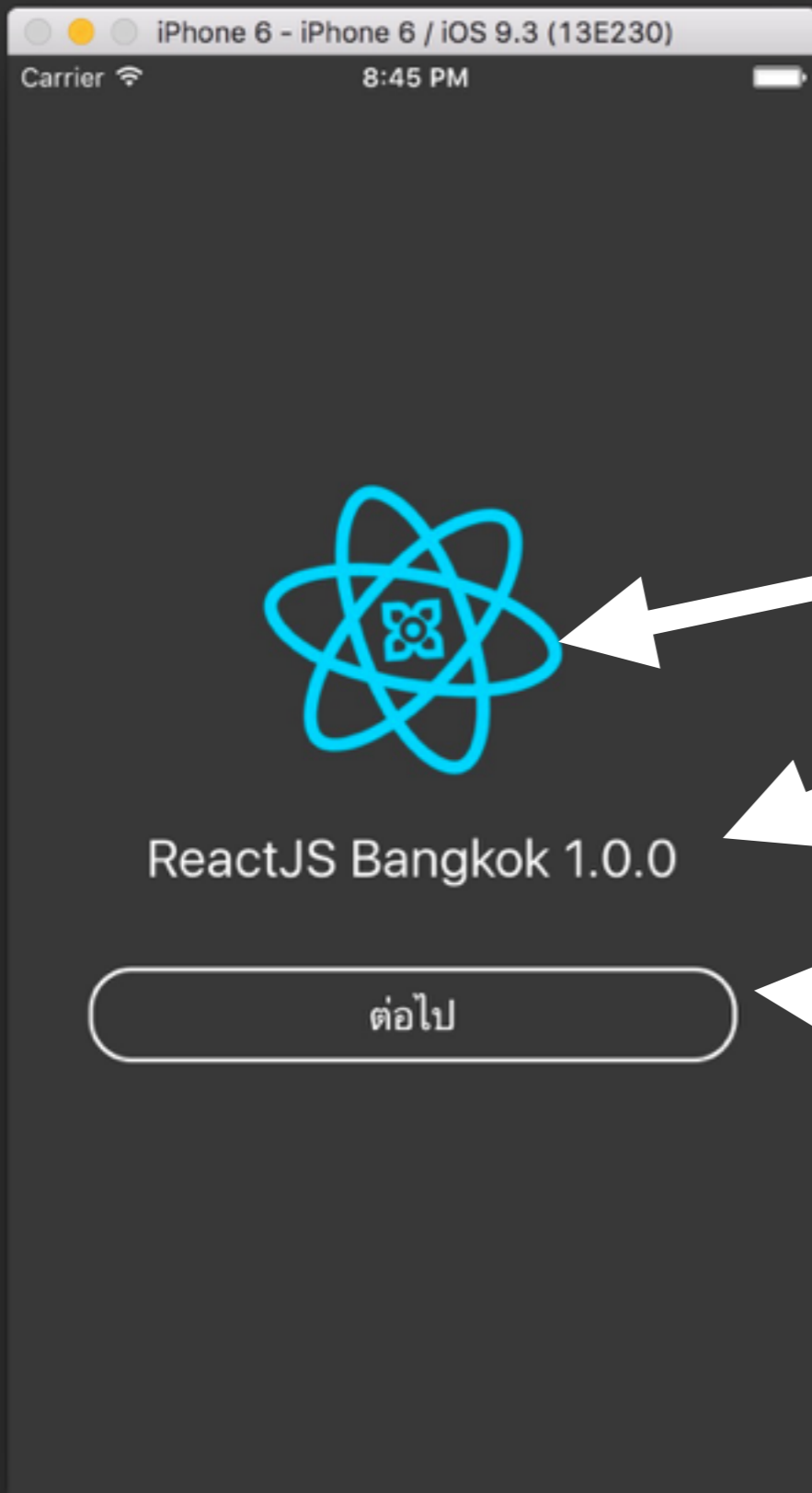


iOS

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



Android



```
return (  
  <View style={styles.container}>  
    <View style={styles.content}>  
      <View style={[styles.logoContainer, showLoginForm && styles.logoContainerSmall]}>  
        <Image  
          source={require('./assets/logo.png')}  
          style={styles.logo}  
        />  
        <Text style={styles.title}>ReactJS Bangkok 1.0.0</Text>  
      </View>  
      <View style={styles.buttonContainer}>  
        <Button style={styles.buttonStyle} text={showLoginForm ? 'ต่อไป' : 'ก่อน'} onPress={this.renderPreLogin()} />  
      </View>  
    </View>  
  </View>  
);
```

```
renderPreLogin() {  
  return (  
    <View style={{ width: 300 }}>  
      <Button  
        style={styles.buttonStyle}  
        textStyle={styles.buttonTextStyle}  
        onPress={() => {  
          this.setState({  
            showLoginForm: true  
          })  
        }}  
      />  
      <Text style={styles.buttonText}>ต่อไป</Text>  
    </Button>  
  </View>  
);  
}
```

iOS

```
iPhone 6 - iPhone 6 / iOS 9.3 (13E230)
Carrier 8:45 PM

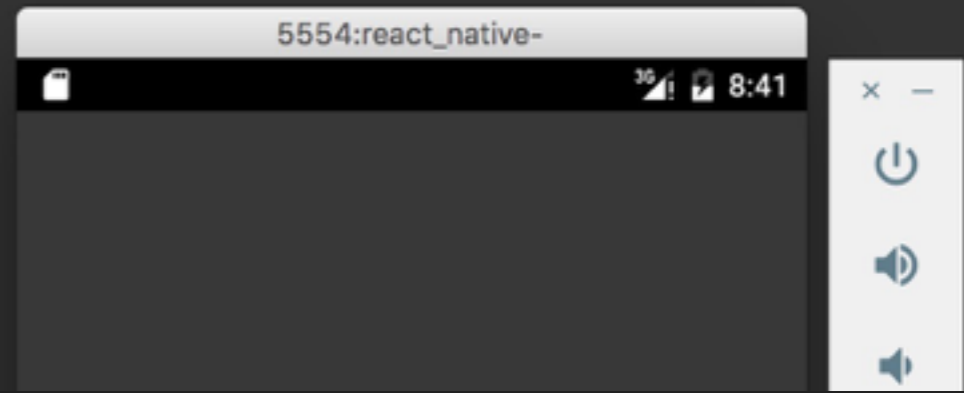
const styles = StyleSheet.create({
  container: {
    flex: 1,
    backgroundColor: '#383838',
    justifyContent: 'center',
  },
  content: {
    alignSelf: 'center'
  },
  logoContainer: {
    transform: [{ scale: 1 }],
    alignSelf: 'center'
  },
  logo: {
    width: 150,
    height: 150,
    alignSelf: 'center'
  },
  title: {
    fontSize: 24,
    color: 'white',
    marginTop: 16,
    marginBottom: 16,
    textAlign: 'center'
  },
},
```

iOS

```
5554:react_native-
3G 8:41

<View style={styles.container}>
  <View style={styles.content}>
    <View style={[styles.logoContainer, showLoginForm && styles.logoContainerSmall]}>
      <Image
        source={require('./assets/logo.png')}
        style={styles.logo}
      />
      <Text style={styles.title}>ReactJS Bangkok 1.0.0</Text>
    </View>
    <View>
      {showLoginForm ? this.renderLoginForm() : this.renderPreLogin()}
    </View>
  </View>
</View>
```

Android



```
<View style={styles.container}>
  <View style={styles.content}>
    <View style={[styles.logoContainer, showLoginForm && styles.logoContainerSmall]}>
      <Image
        source={require('./assets/logo.png')}
        style={styles.logo}
      />
      <Text style={styles.title}>ReactJS Bangkok 1.0.0</Text>
    </View>
    <View>
      {showLoginForm? this.renderLoginForm(): this.renderPreLogin()}
    </View>
  </View>
</View>
```

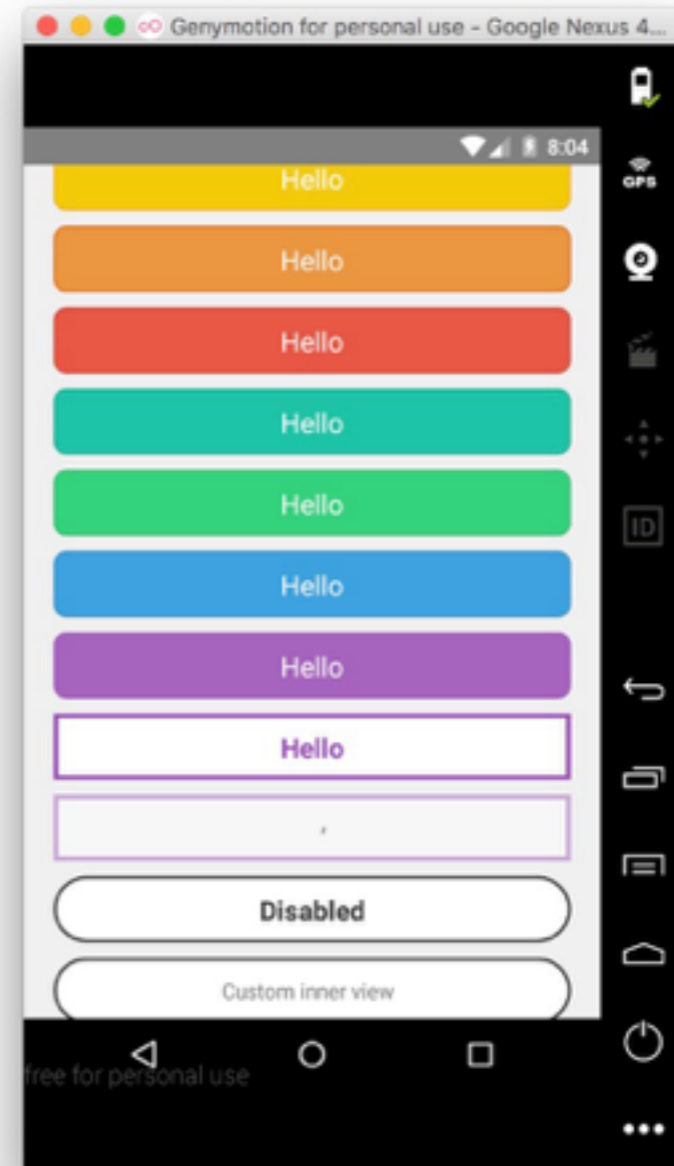
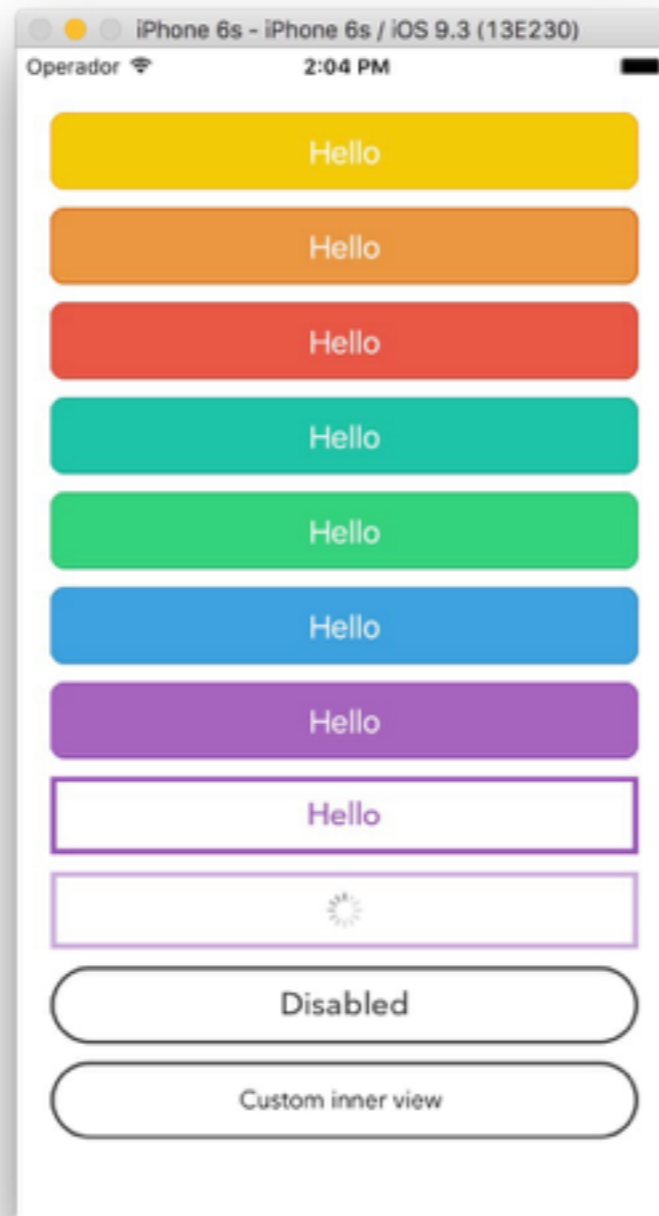
```
renderPreLogin() {
  return (
    <View style={{ width: 300 }}>
      <Button
        style={styles.buttonStyle}
        textStyle={styles.buttonTextStyle}
        onPress={() => {
          this.setState({
            showLoginForm: true
          })
        }}
      />
    </View>
  );
}
```

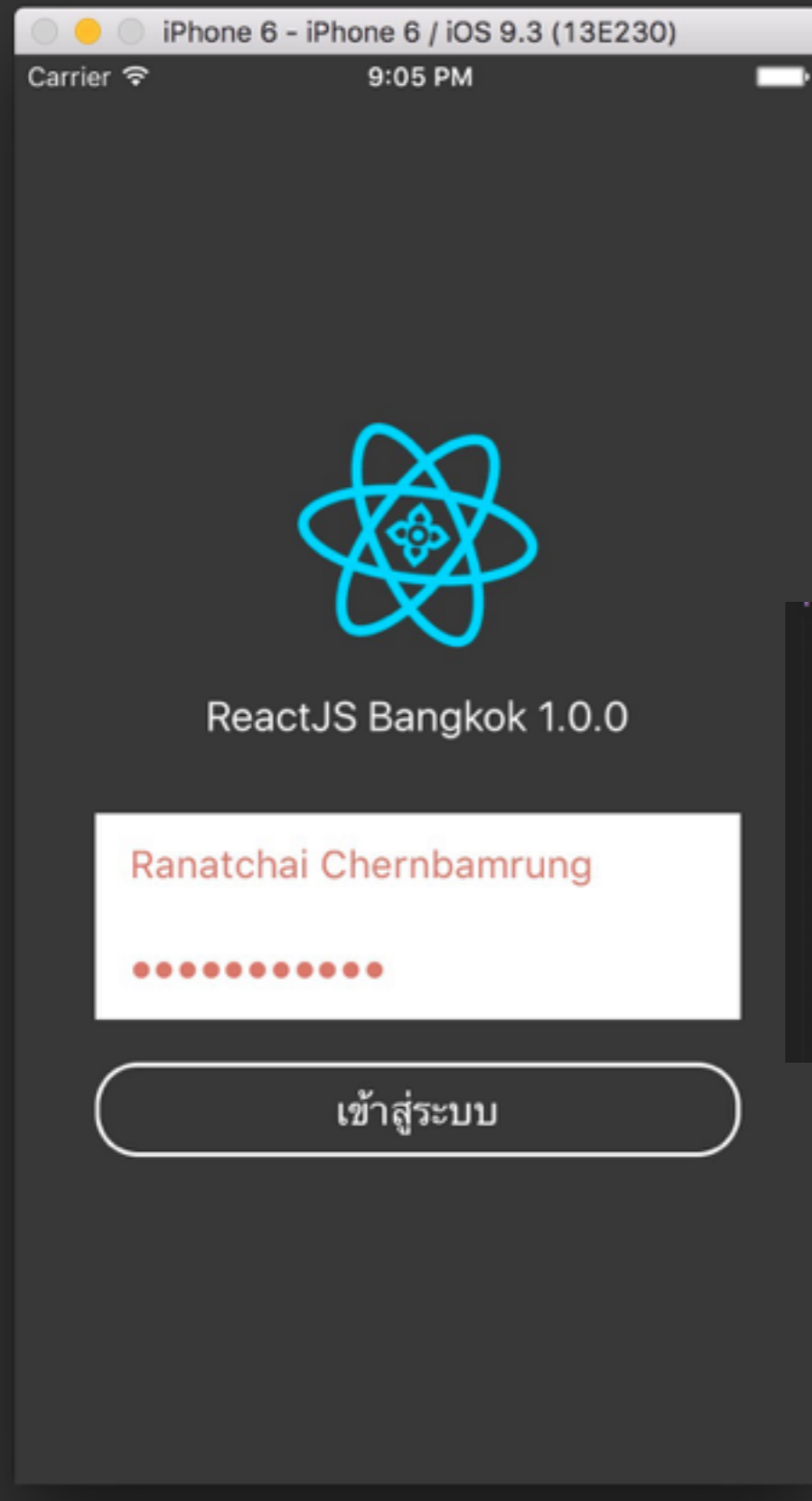
```
import Button from 'apsl-react-native-button'
```

iOS

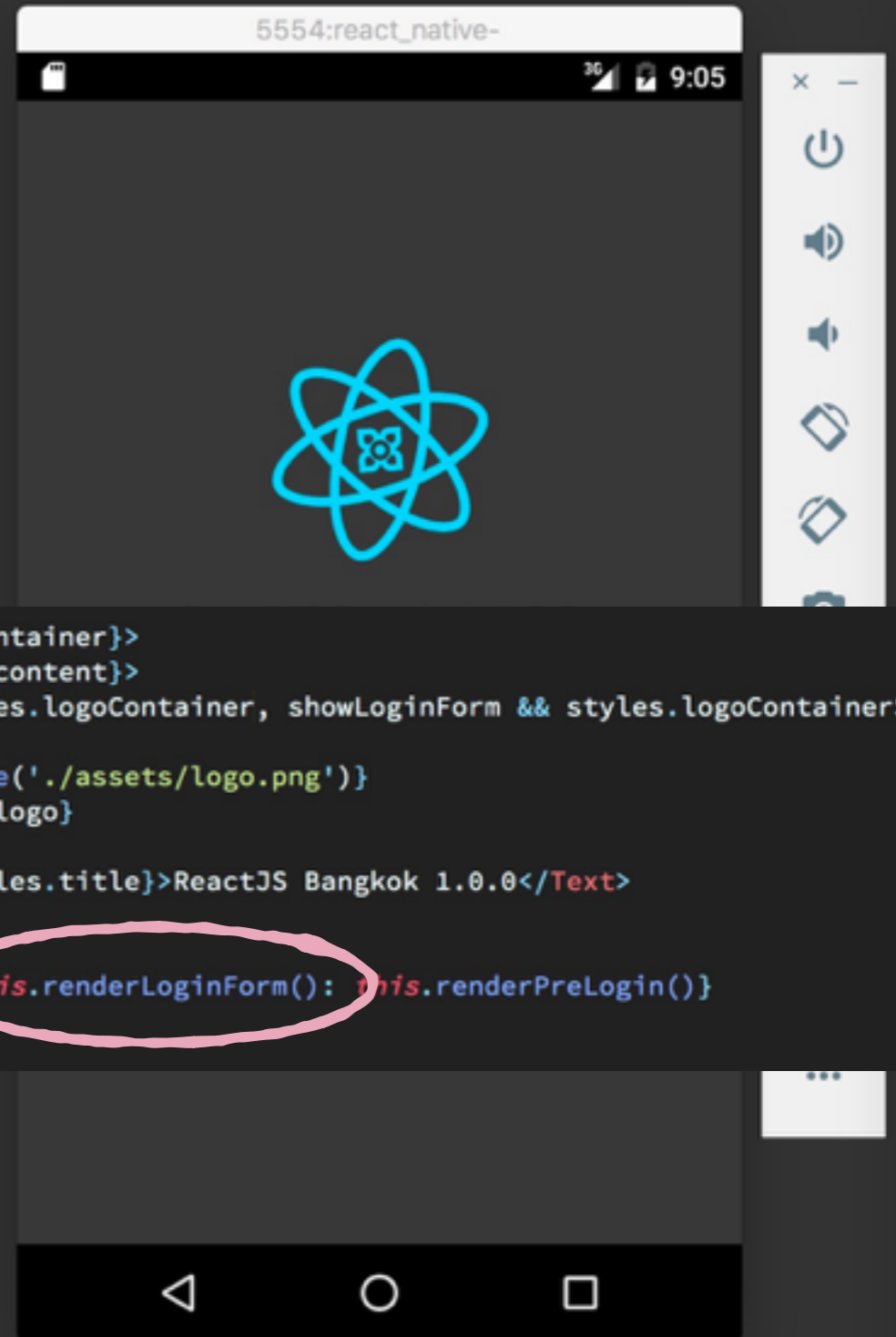
apsl-react-native-button

A React Native button component customizable via `style` props.

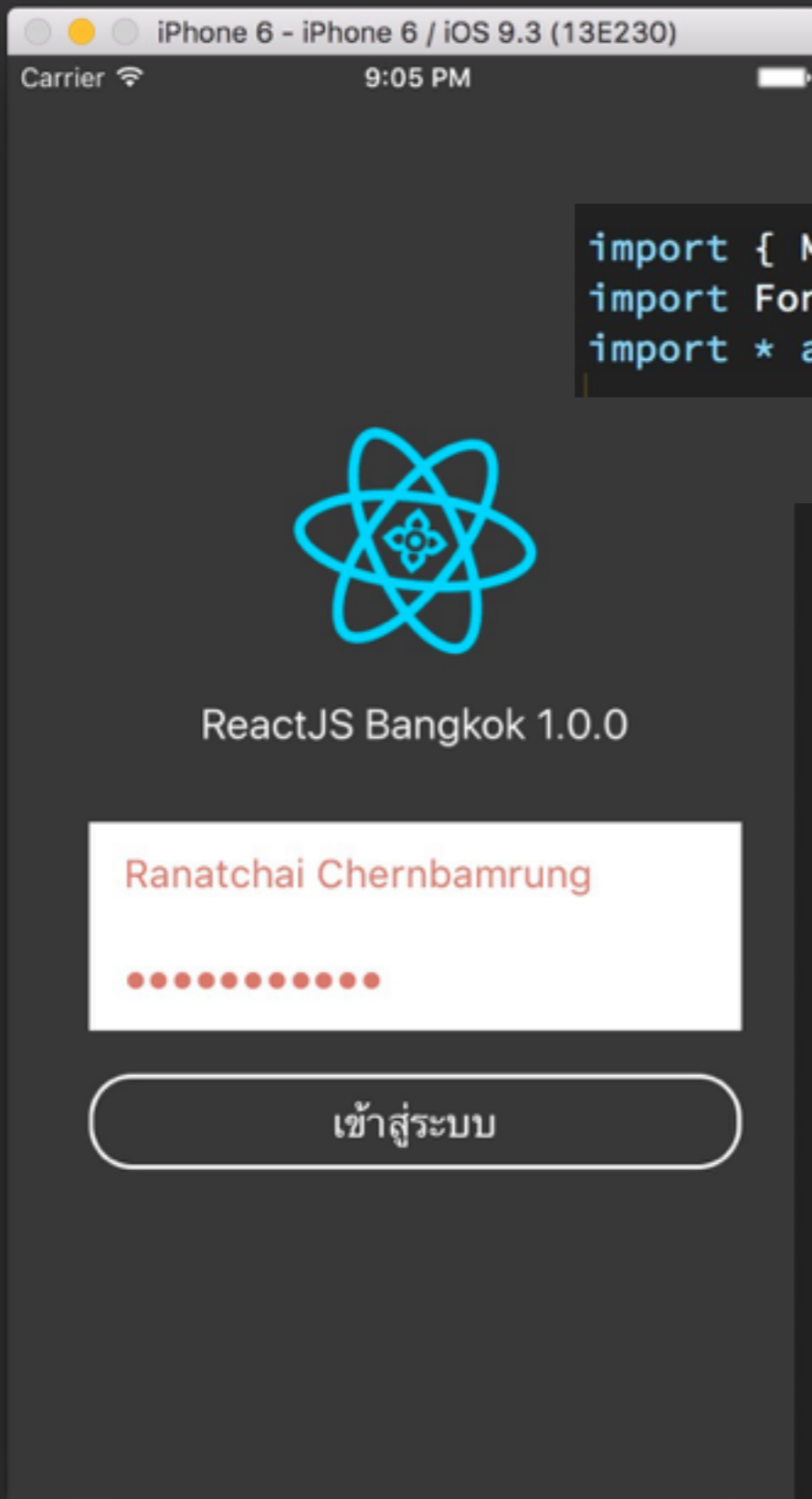




iOS



Android



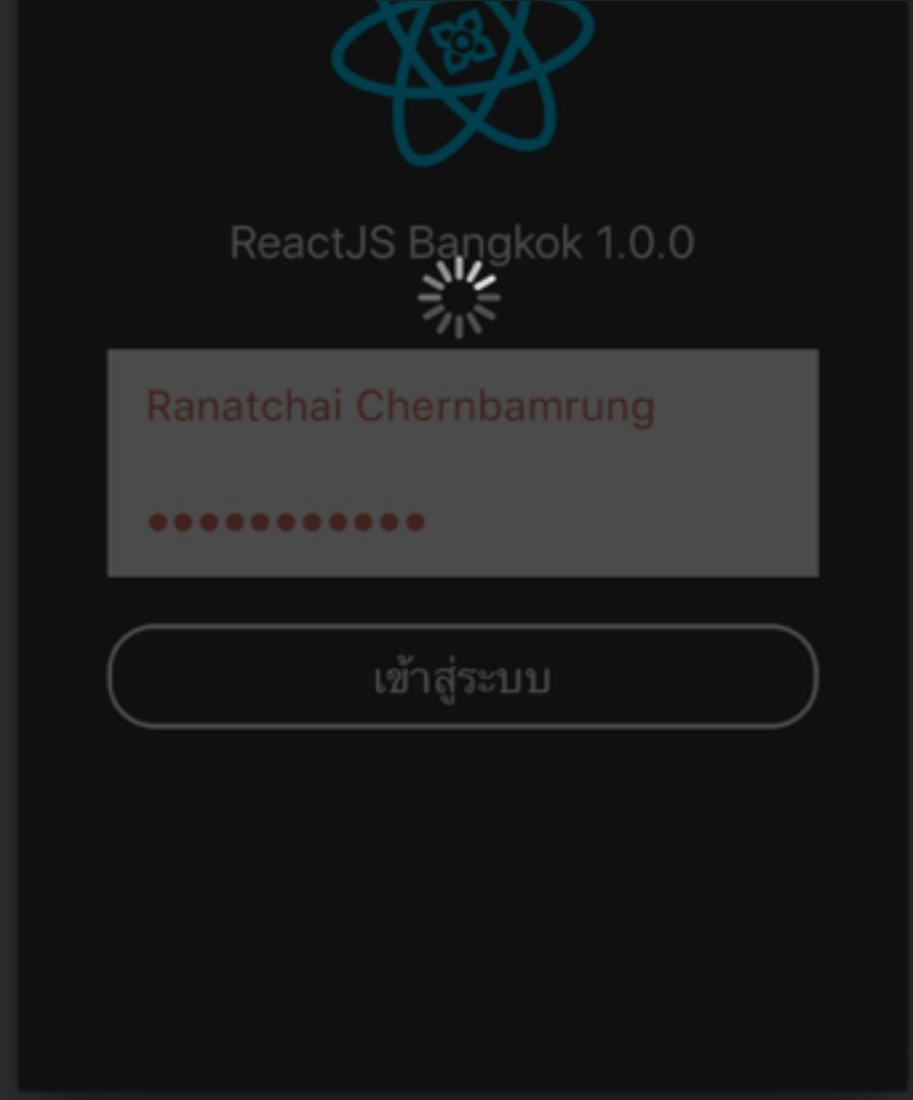
```
import { Makiko as TextInput } from 'react-native-textinput-effects';  
import FontAwesomeIcon from 'react-native-vector-icons/FontAwesome';  
import * as Animatable from 'react-native-animatable';
```

```
renderLoginForm() {  
  return (  
    <Animatable.View animation="slideInUp" duration={300} easing="linear" style={{ width: 300 }}>  
      <TextInput  
        label={'Comment'}  
        iconClass={FontAwesomeIcon}  
        iconName={'comment'}  
        iconColor={'white'}  
        inputStyle={{ color: '#db786d' }}  
        value="Ranatchai Chernbamrung"  
        placeholder={'Full Name'}  
      />  
      <TextInput  
        label={'Comment'}  
        iconClass={FontAwesomeIcon}  
        iconName={'comment'}  
        iconColor={'white'}  
        inputStyle={{ color: '#db786d' }}  
        value="MY=Password"  
        placeholder={'Password'}  
        secureTextEntry  
      />  
      <Button  
        style={styles.buttonStyle}  
        textStyle={styles.buttonTextStyle}  
        key="login"  
        onPress={this.handleLogin}  
      />  
      เข้าสู่ระบบ  
    </Animatable.View>  
  );  
}
```

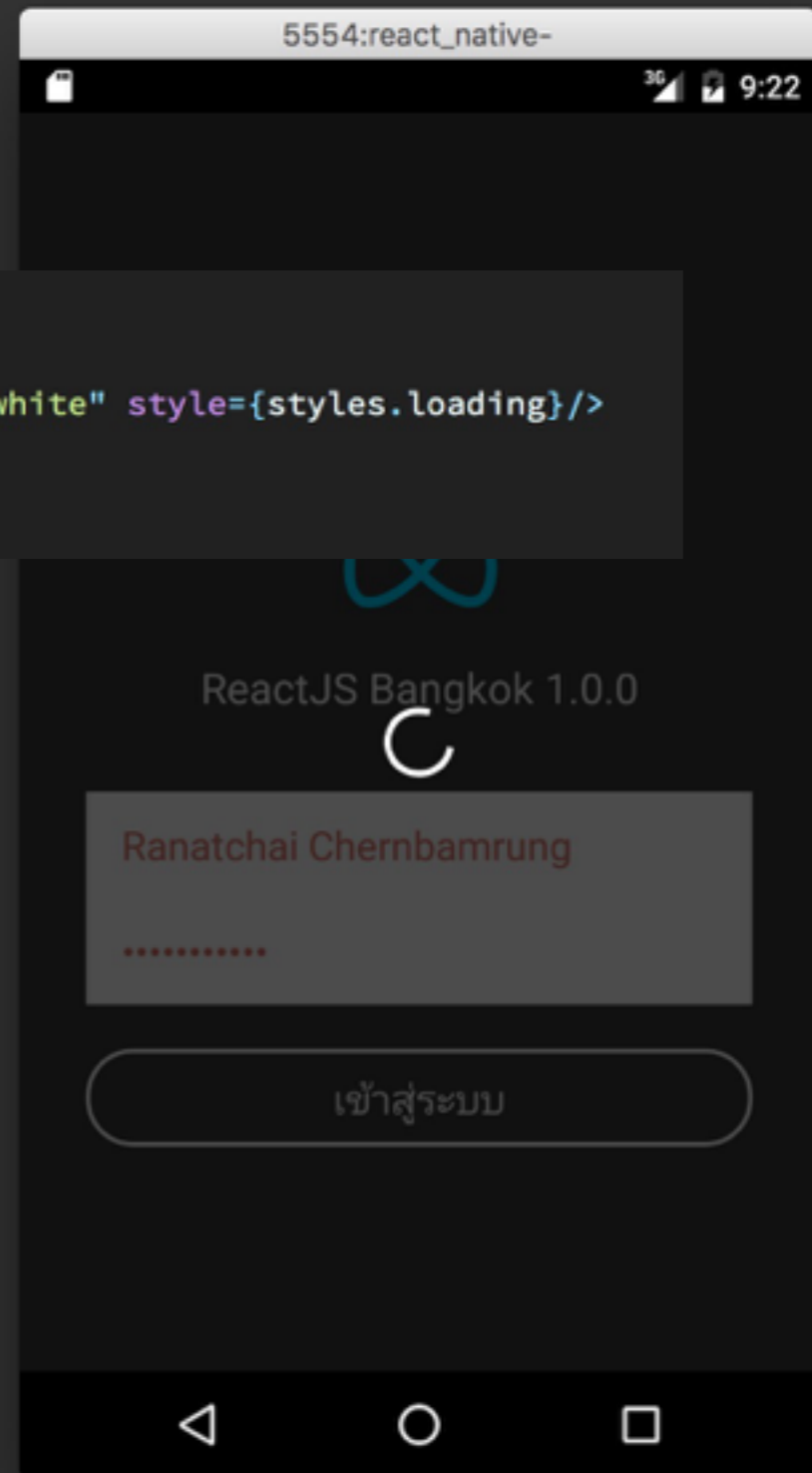
iOS

```
LayoutAnimation,  
ActivityIndicator,  
} from 'react-native';
```

```
</View>  
{showLoading && (  
  <View style={styles.loadingContainer}>  
    <ActivityIndicator size="large" color="white" style={styles.loading}/>  
  </View>  
)}  
</View>
```



iOS



Android

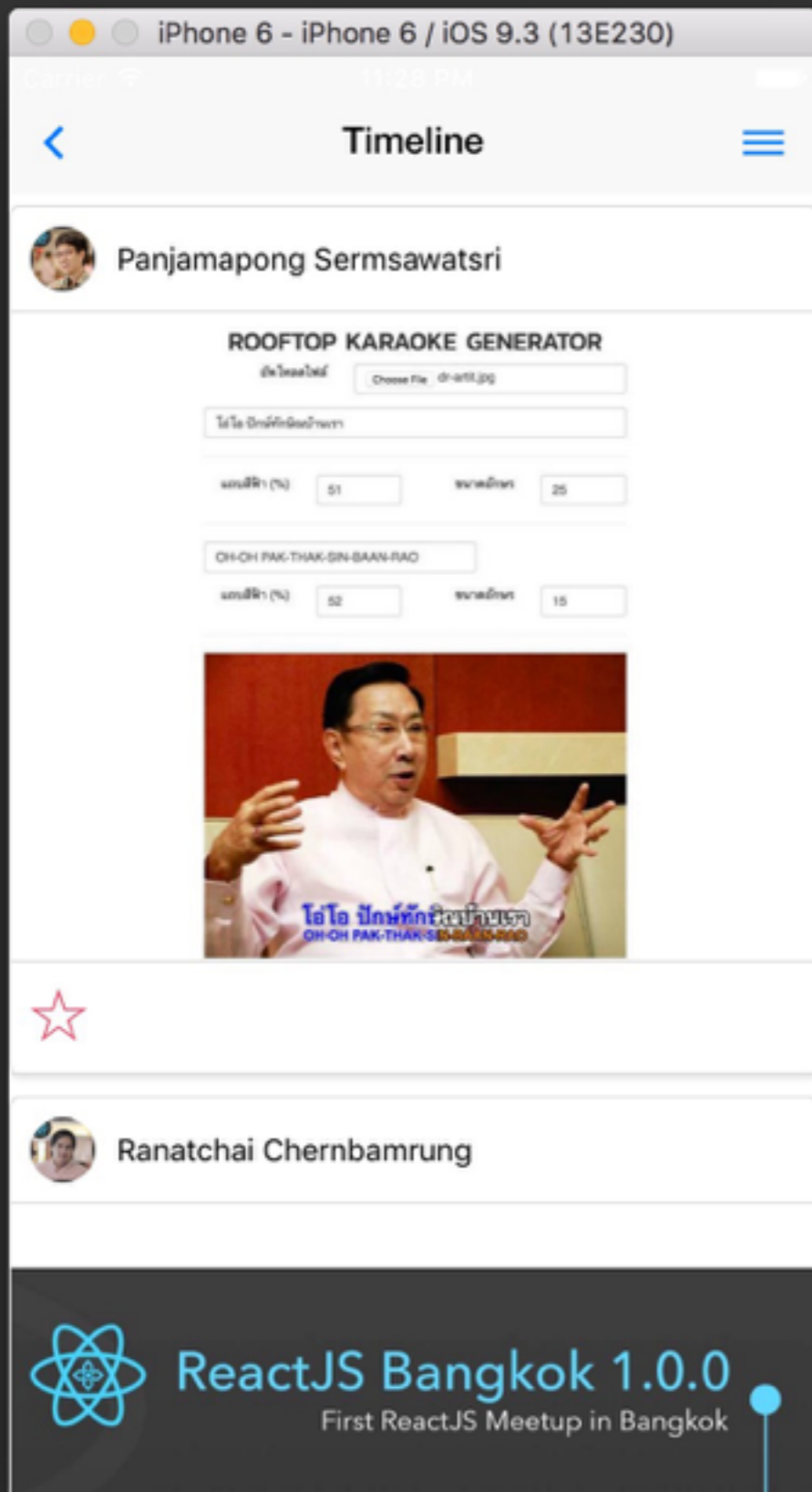
Thing We Learn

- Native bundle per Platform (iOS/Android)
- Write code like React web but can't use DOM
- Easy Stylesheet
- find Component from github

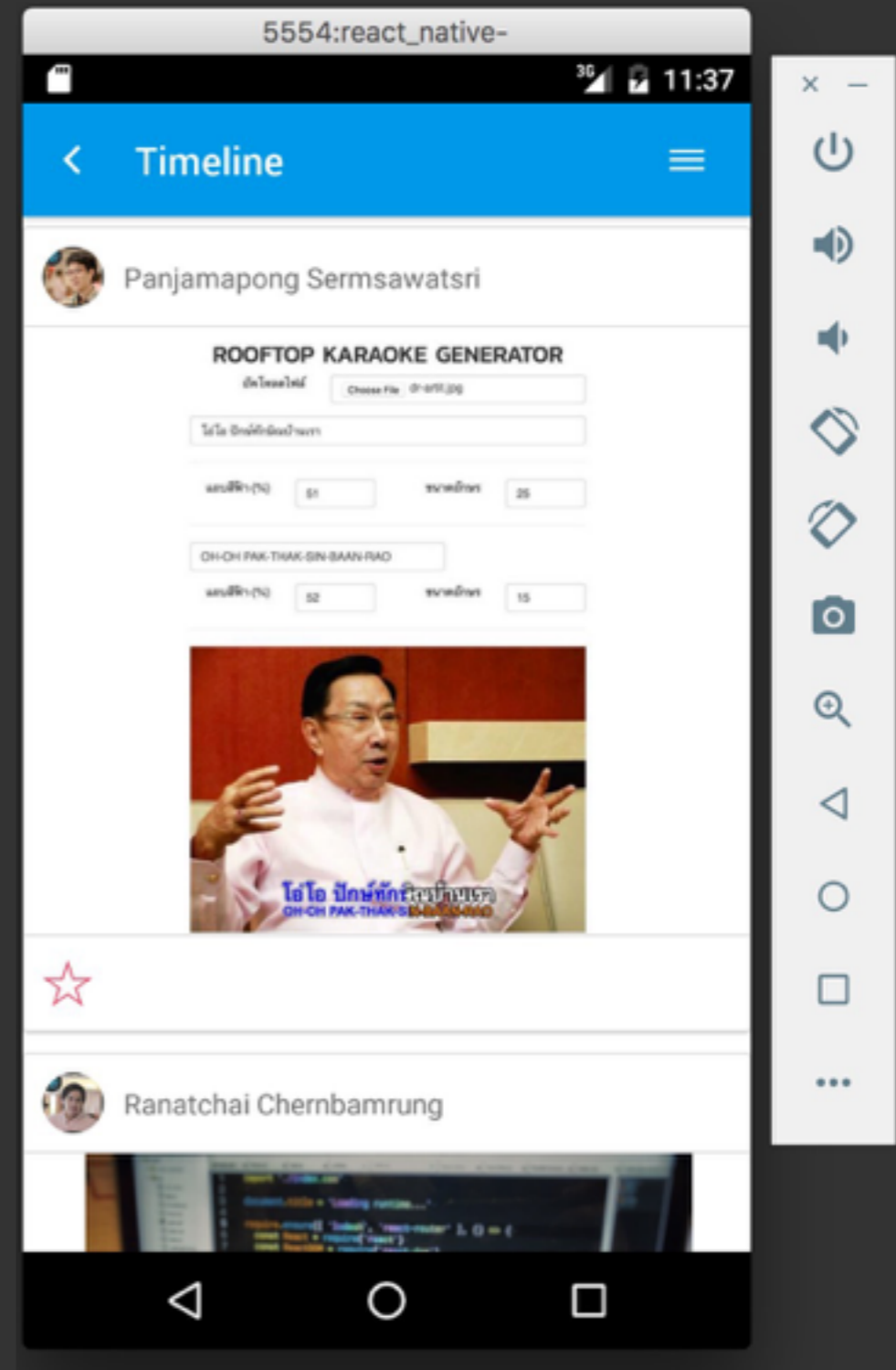
<https://github.com/ranatchai/react-native-bkk-example>

Code Overview (2)

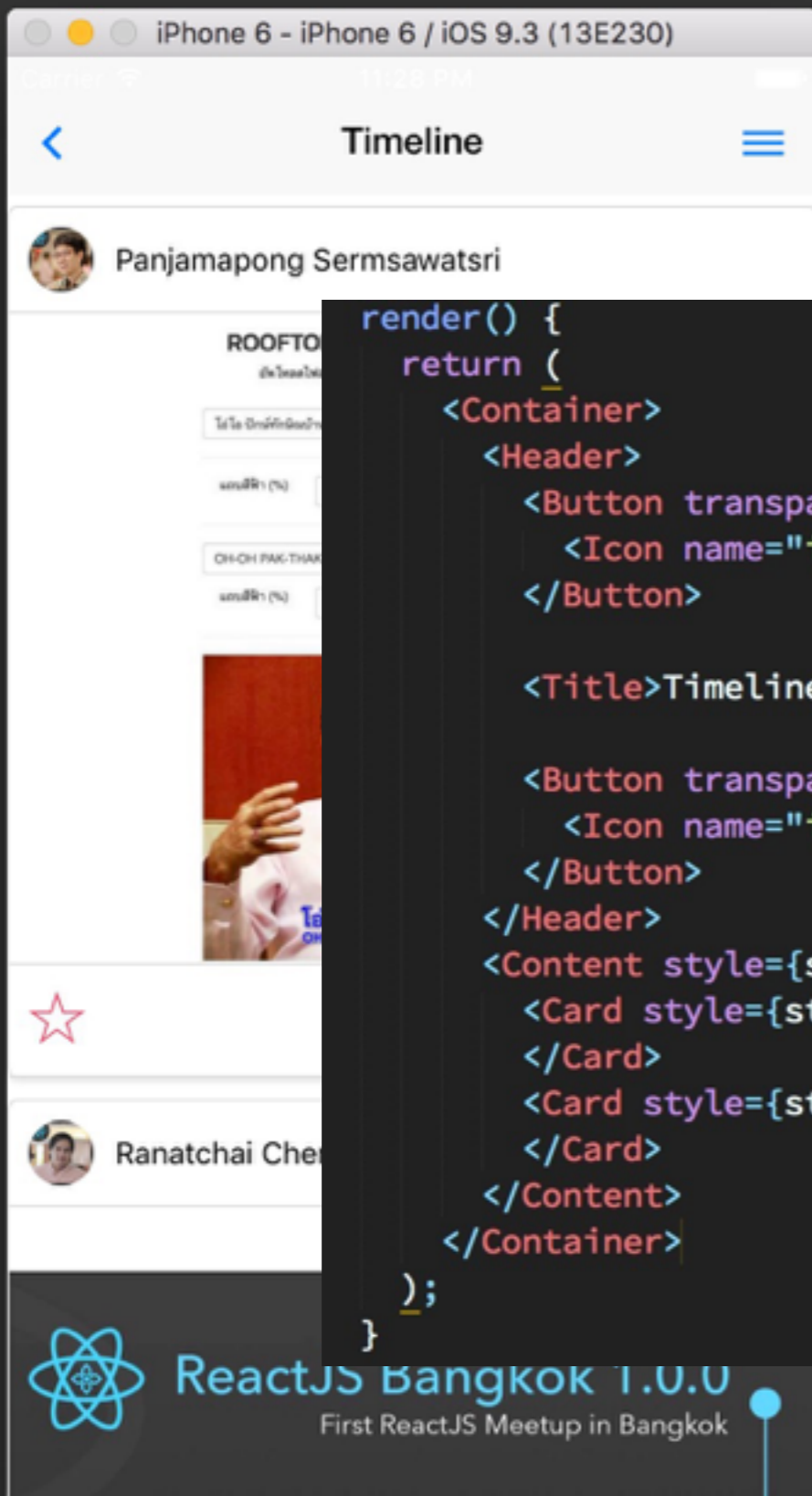
<https://github.com/ranatchai/react-native-bkk-example>



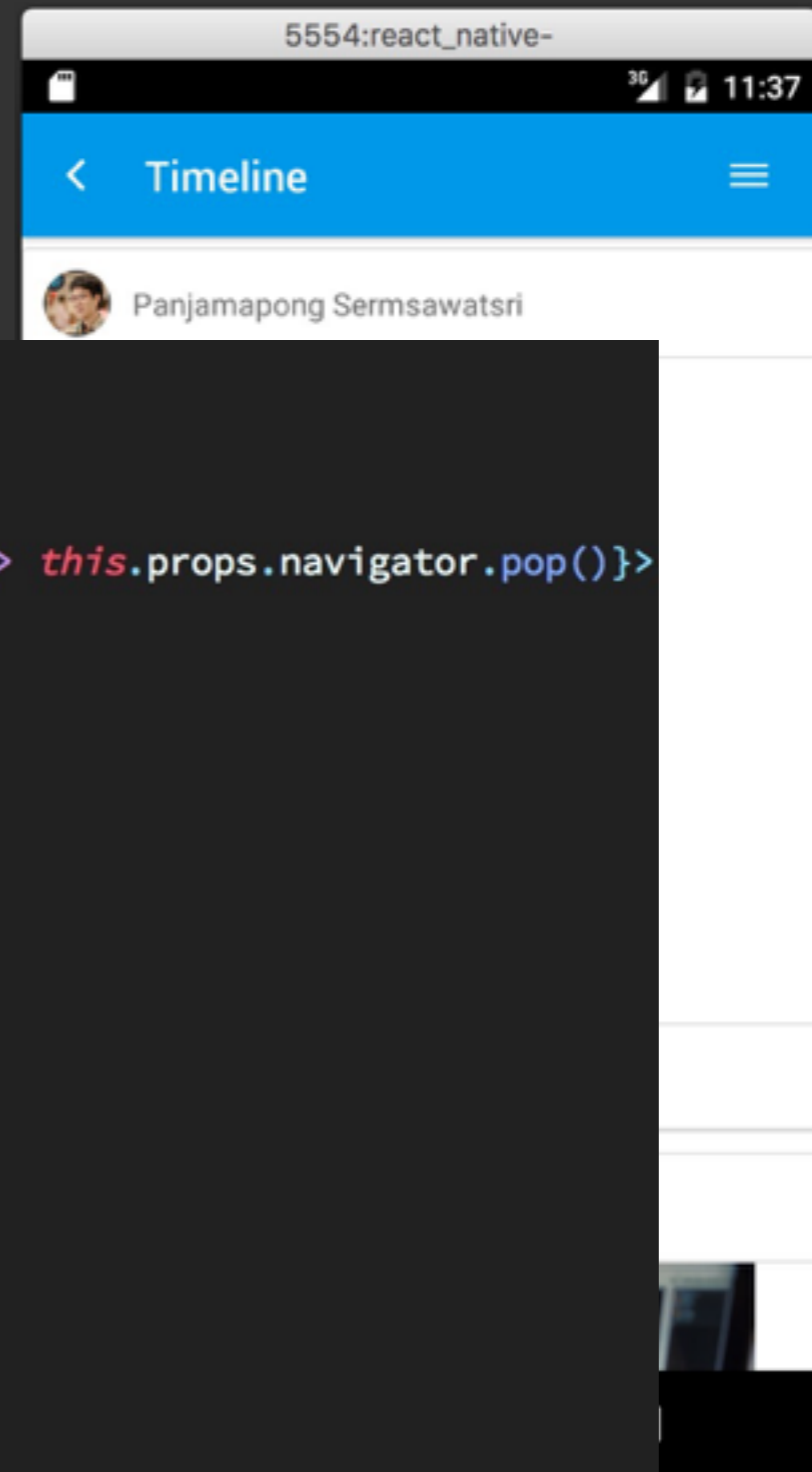
iOS



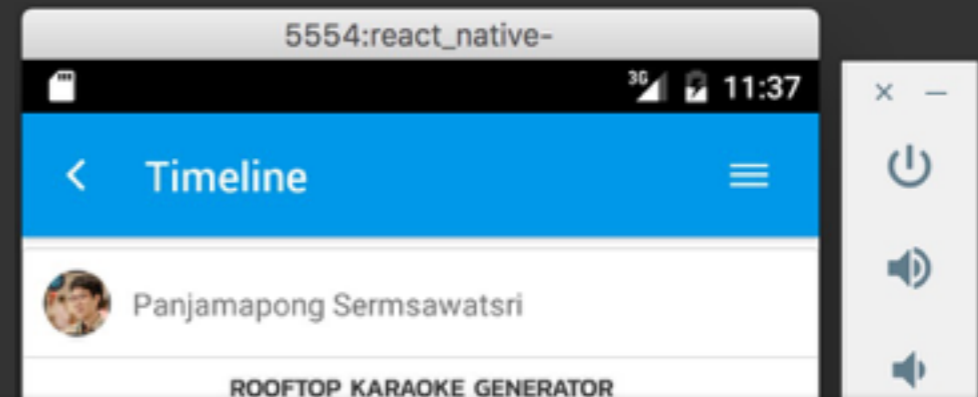
Android



iOS



Android



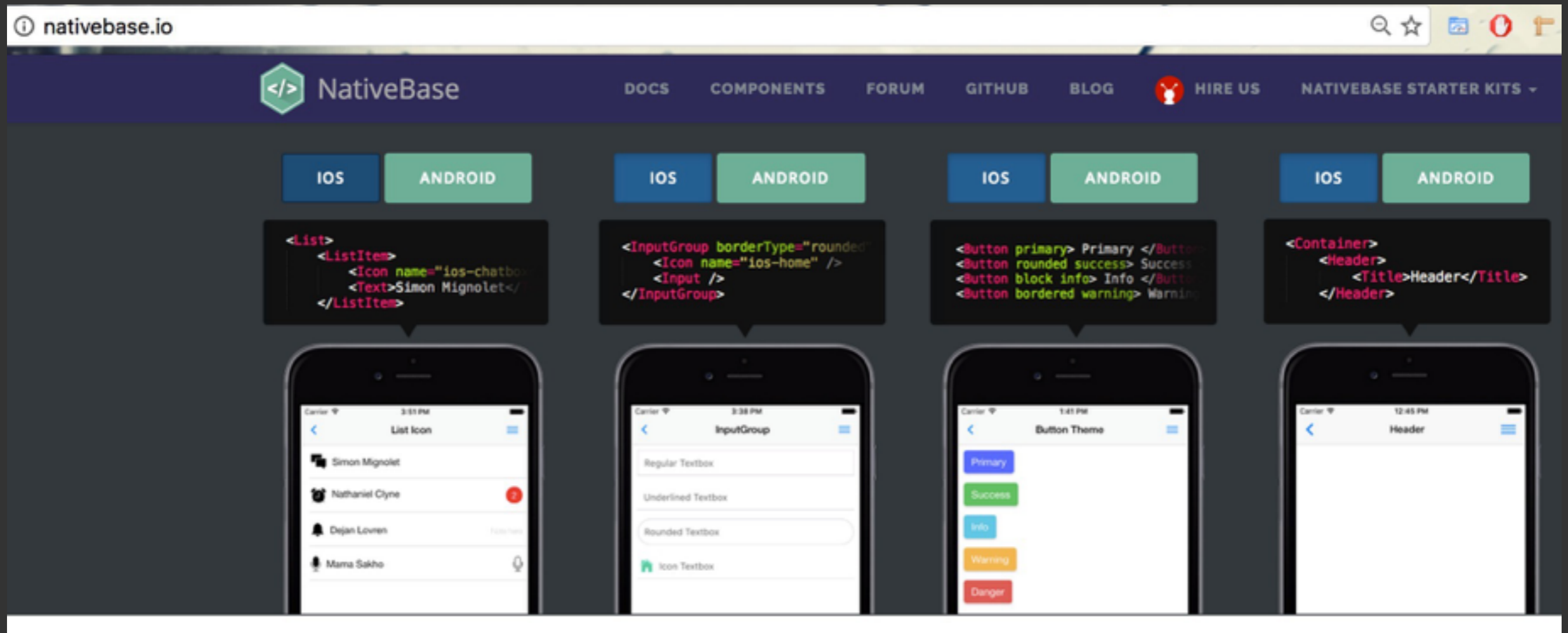
```
render() {  
  return (  
    <Container>  
      <Header>  
        <Button transparent onPress={() =>  
          <Icon name="ios-arrow-back" />  
        } />  
        <Title>Timeline</Title>  
        <Button transparent  
          <Icon name="ios-menu" />  
        />  
      </Header>  
      <Content style={styles.container}>  
        <Card style={styles.card}>  
        </Card>  
        <Card style={styles.card}>  
        </Card>  
      </Content>  
    </Container>  
  );  
}
```

```
import {  
  Container,  
  Header,  
  Title,  
  Content,  
  Button,  
  Icon,  
  Card,  
  CardItem,  
  Thumbnail,  
} from 'native-base';
```

ANDROID

iOS

nativebase.io

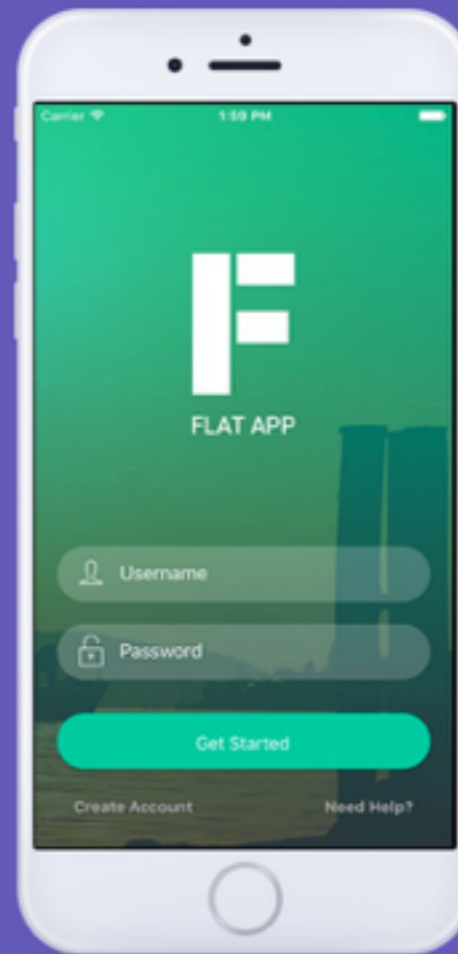


Bootstrap in React-Native

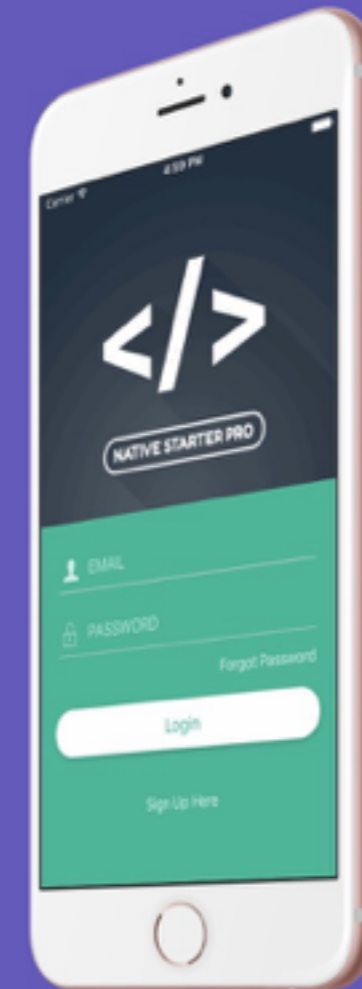
nativebase.io



Native Starter (Free)



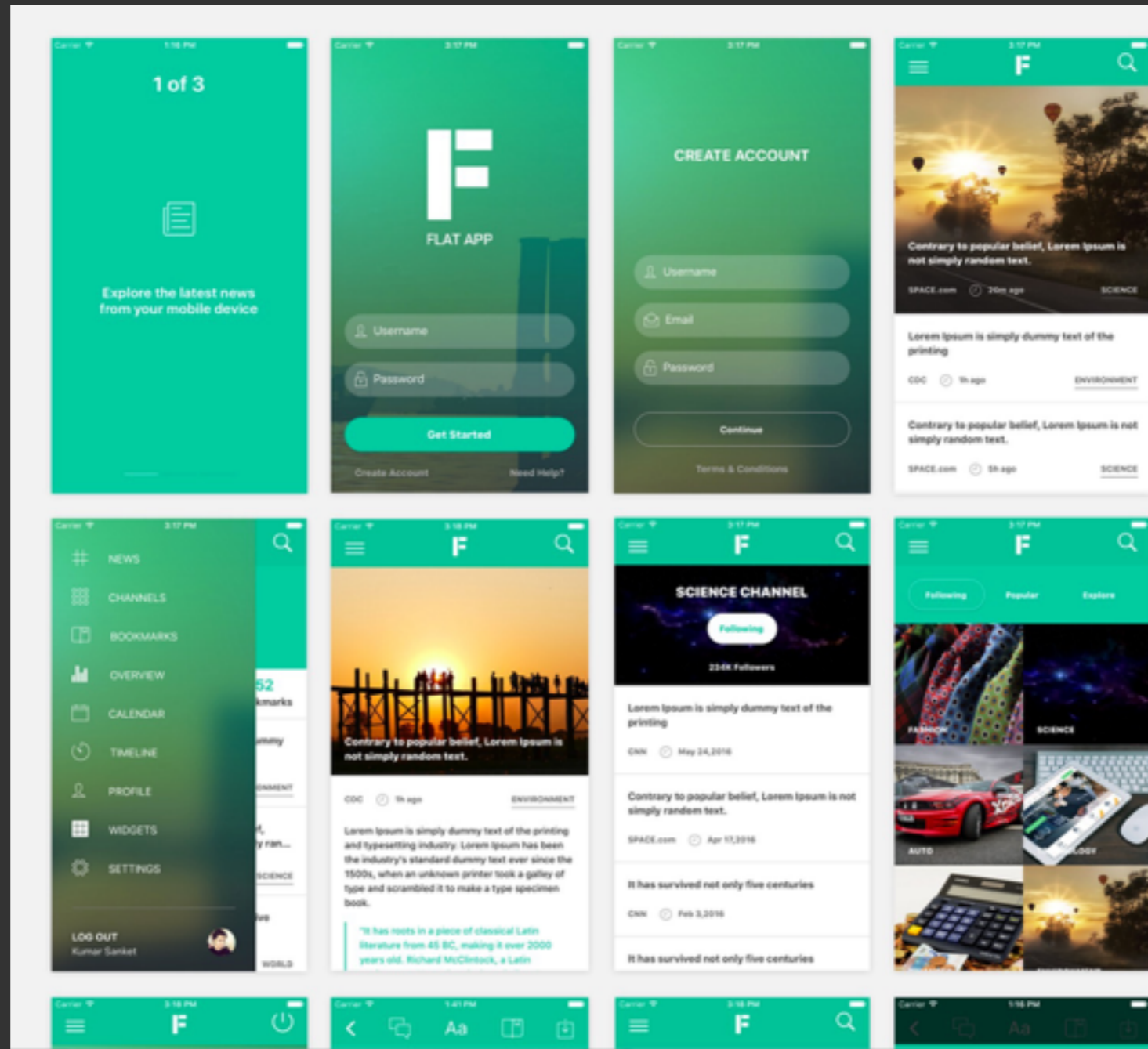
React Native Flat App Theme



Native Starter Pro

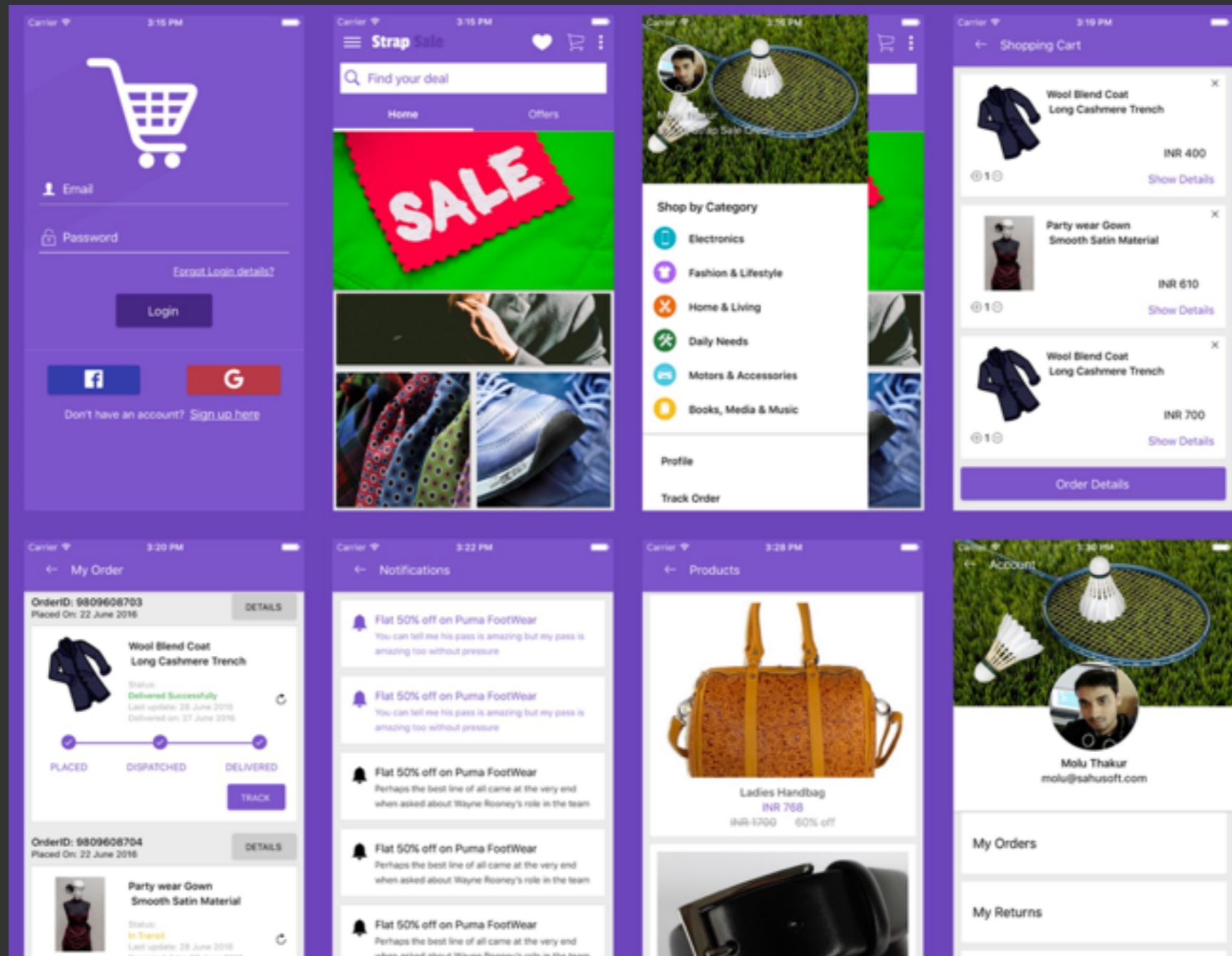
NativeBase Theme

nativebase.io

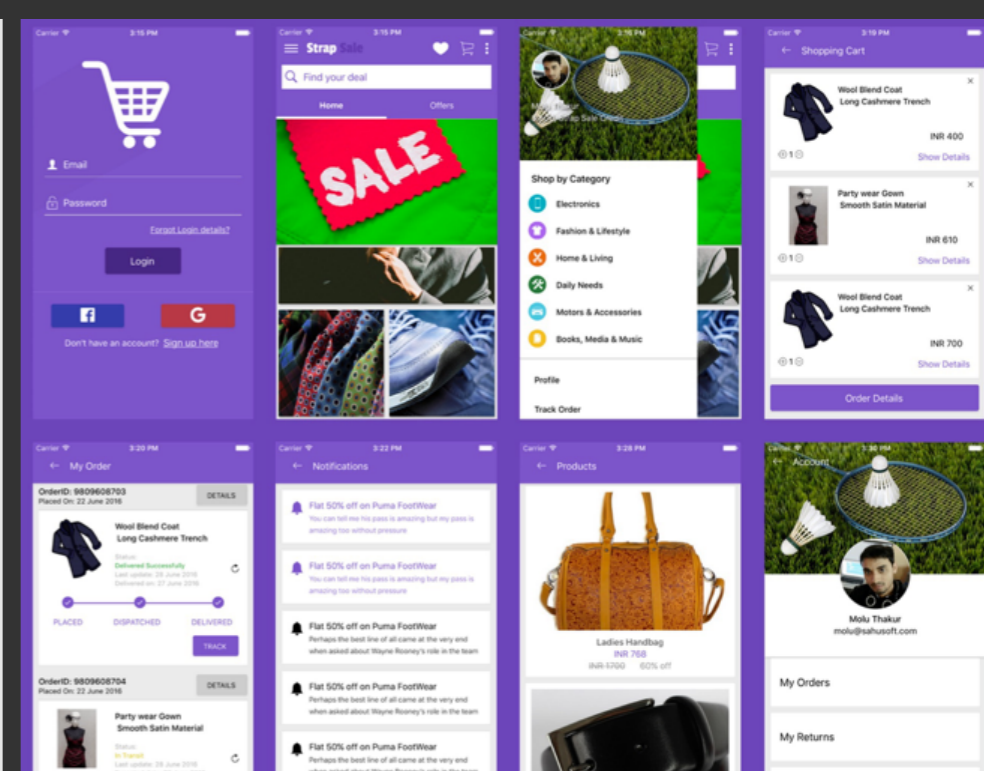
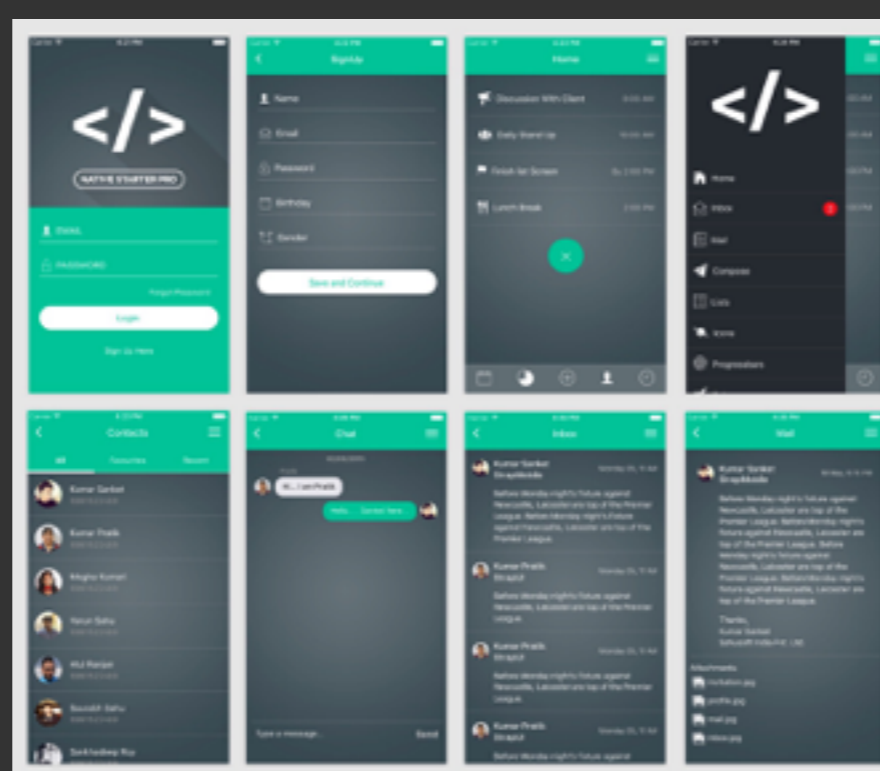
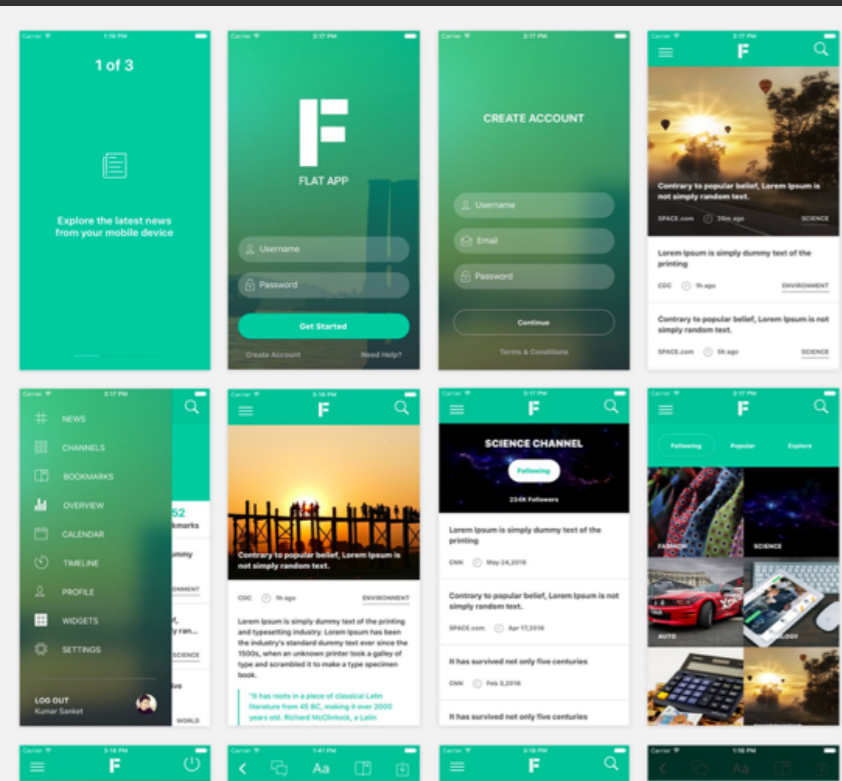


NativeBase Theme

nativebase.io



E-Commerce Theme



Third party modules used:

react: 15.1.0
 redux: ^3.5.2
 react-native-modalbox: 1.3.4
 react-native-button: ^1.6.0

react-native: ^0.27.2
 react-redux: ^4.4.5
 react-native-vector-icons: ^2.0.3
 react-native-gifted-spinner: 0.0.5

native-base: 0.4.9
 react-native-code-push: ^1.12.1-beta
 react-native-easy-grid: ^0.1.6
 moment: ^2.13.0

Themeforest for ReactNative

Development Process

<https://github.com/ranatchai/react-native-bkk-example>

Run Packager

```
$ react-native start
```

Run Packager

```
$ react-native start
```

```
Running packager on port 8081.

Keep this packager running while developing on any JS projects. Feel
free to close this tab and run your own packager instance if you
prefer.

https://github.com/facebook/react-native

Looking for JS files in
/Users/Ranatchai/Project/ReactNativeBKK

[8:19:32 PM] <START> Building Dependency Graph
[8:19:33 PM] <START> Crawling File System
[Hot Module Replacement] Server listening on /hot

React packager ready.

Recrawled this watch 14 times, most recently because:
/Users/Ranatchai/Project/ReactNativeBKK: kFSEventStreamEventFlagKernelDropped
To resolve, please review the information on
https://facebook.github.io/watchman/docs/troubleshooting.html#recrawl
To clear this warning, run:
`watchman watch-del /Users/Ranatchai/Project/ReactNativeBKK ; watchman watch-project /Users/Ranatchai/Project/ReactNativeBKK`

watchman warning: Recrawled this watch 14 times, most recently because:
/Users/Ranatchai/Project/ReactNativeBKK: kFSEventStreamEventFlagKernelDropped
To resolve, please review the information on
https://facebook.github.io/watchman/docs/troubleshooting.html#recrawl
To clear this warning, run:
`watchman watch-del /Users/Ranatchai/Project/ReactNativeBKK ; watchman watch-project /Users/Ranatchai/Project/ReactNativeBKK`

[8:19:33 PM] <END> Crawling File System (806ms)
[8:19:33 PM] <START> Building in-memory fs for JavaScript
[8:19:34 PM] <END> Building in-memory fs for JavaScript (297ms)
[8:19:34 PM] <START> Building in-memory fs for Assets
[8:19:34 PM] <END> Building in-memory fs for Assets (130ms)
[8:19:34 PM] <START> Building Haste Map
[8:19:34 PM] <START> Building (deprecated) Asset Map
[8:19:34 PM] <END> Building (deprecated) Asset Map (82ms)
[8:19:34 PM] <END> Building Haste Map (199ms)
[8:19:34 PM] <END> Building Dependency Graph (1462ms)
[8:19:57 PM] <START> request:/index.ios.bundle?platform=ios&dev=true&minify=false
[8:19:57 PM] <START> find dependencies
transformed 755/755 (100%)
[8:19:59 PM] <END> find dependencies (1357ms)
[8:19:59 PM] <END> request:/index.ios.bundle?platform=ios&dev=true&minify=false (1493ms)
```

Run Packager

```
$ react-native start
```

```
Running packager on port 8081.  
  
Keep this packager running while developing on any JS projects. Feel  
free to close this tab and run your own packager instance if you  
prefer.  
  
https://github.com/facebook/react-native
```

```
Looking for JS files in  
/Users/Ranatchai/Project/ReactNativeBKK  
[8:19:32 PM] <START> Building Dependency Graph  
[8:19:33 PM] <START> Crawling File System  
[Hot Module Replacement] Server listening on /hot
```

```
React packager ready.
```

```
Recrawled this watch 14 times, most recently  
/Users/Ranatchai/Project/ReactNativeBKK: kFS  
To resolve, please review the information on  
https://facebook.github.io/watchman/docs/troubleshooting.html#recrawl  
To clear this warning, run:  
'watchman watch-del /Users/Ranatchai/Project
```

```
watchman warning: Recrawled this watch 14 times, most recently because:  
/Users/Ranatchai/Project/ReactNativeBKK: kFSEventStreamEventFlagKernelDropped  
To resolve, please review the information on  
https://facebook.github.io/watchman/docs/troubleshooting.html#recrawl  
To clear this warning, run:  
'watchman watch-del /Users/Ranatchai/Project/ReactNativeBKK ; watchman watch-project /Users/Ranatchai/Project/ReactNativeBKK'
```

```
[8:19:33 PM] <END> Crawling File System (806ms)  
[8:19:33 PM] <START> Building in-memory fs for JavaScript  
[8:19:34 PM] <END> Building in-memory fs for JavaScript (297ms)  
[8:19:34 PM] <START> Building in-memory fs for Assets  
[8:19:34 PM] <END> Building in-memory fs for Assets (130ms)  
[8:19:34 PM] <START> Building Haste Map  
[8:19:34 PM] <START> Building (deprecated) Asset Map  
[8:19:34 PM] <END> Building (deprecated) Asset Map (82ms)  
[8:19:34 PM] <END> Building Haste Map (199ms)  
[8:19:34 PM] <END> Building Dependency Graph (1462ms)  
[8:19:57 PM] <START> request:/index.ios.bundle?platform=ios&dev=true&minify=false  
[8:19:57 PM] <START> find dependencies  
transformed 755/755 (100%)  
[8:19:59 PM] <END> find dependencies (1357ms)  
[8:19:59 PM] <END> request:/index.ios.bundle?platform=ios&dev=true&minify=false (1493ms)
```

```
$ react-native run-ios
```

Run iOS

Run Packager

\$ react-native start

```
Running packager on port 8081.

Keep this packager running while developing on any JS projects. Feel
free to close this tab and run your own packager instance if you
prefer.

https://github.com/facebook/react-native

Looking for JS files in
/Users/Ranatchai/Project/ReactNativeBKK

[8:19:32 PM] <START> Building Dependency Graph
[8:19:33 PM] <START> Crawling File System
[Hot Module Replacement] Server listening on /hot

React packager ready.

Recrawled this watch 14 times, most recently because:
/Users/Ranatchai/Project/ReactNativeBKK: kFSEventStreamEventFlagKernelDropped
To resolve, please review the information on
https://facebook.github.io/watchman/docs/troubleshooting.html#recrawl
To clear this warning, run:
`watchman watch-del /Users/Ranatchai/Project/ReactNativeBKK ; watchman watch-project /Users/Ranatchai/Project/ReactNativeBKK`

watchman warning: Recrawled this watch 14 times, most recently because:
/Users/Ranatchai/Project/ReactNativeBKK: kFSEventStreamEventFlagKernelDropped
To resolve, please review the information on
https://facebook.github.io/watchman/docs/troubleshooting.html#recrawl
To clear this warning, run:
`watchman watch-del /Users/Ranatchai/Project/ReactNativeBKK ; watchman watch-project /Users/Ranatchai/Project/ReactNativeBKK`

[8:19:33 PM] <END> Crawling File System (806ms)
[8:19:33 PM] <START> Building in-memory fs for JavaScript
[8:19:34 PM] <END> Building in-memory fs for JavaScript (297ms)
[8:19:34 PM] <START> Building in-memory fs for Assets
[8:19:34 PM] <END> Building in-memory fs for Assets (130ms)
[8:19:34 PM] <START> Building Haste Map
[8:19:34 PM] <START> Building (deprecated) Asset Map
[8:19:34 PM] <END> Building (deprecated) Asset Map (82ms)
[8:19:34 PM] <END> Building Haste Map (199ms)
[8:19:34 PM] <END> Building Dependency Graph (1462ms)
[8:19:57 PM] <START> request:/index.ios.bundle?platform=ios&dev=true&minify=false
[8:19:57 PM] <START> find dependencies
transformed 755/755 (100%)
[8:19:59 PM] <END> find dependencies (1357ms)
[8:19:59 PM] <END> request:/index.ios.bundle?platform=ios&dev=true&minify=false (1493ms)
```



\$ react-native run-ios

Run iOS

Run Packager

```
$ react-native start
```

```
Running packager on port 8081.

Keep this packager running while developing on any JS projects. Feel
free to close this tab and run your own packager instance if you
prefer.

https://github.com/facebook/react-native

Looking for JS files in
/Users/Ranatchai/Project/ReactNativeBKK

[8:19:32 PM] <START> Building Dependency Graph
[8:19:33 PM] <START> Crawling File System
[Hot Module Replacement] Server listening on /hot

React packager ready.

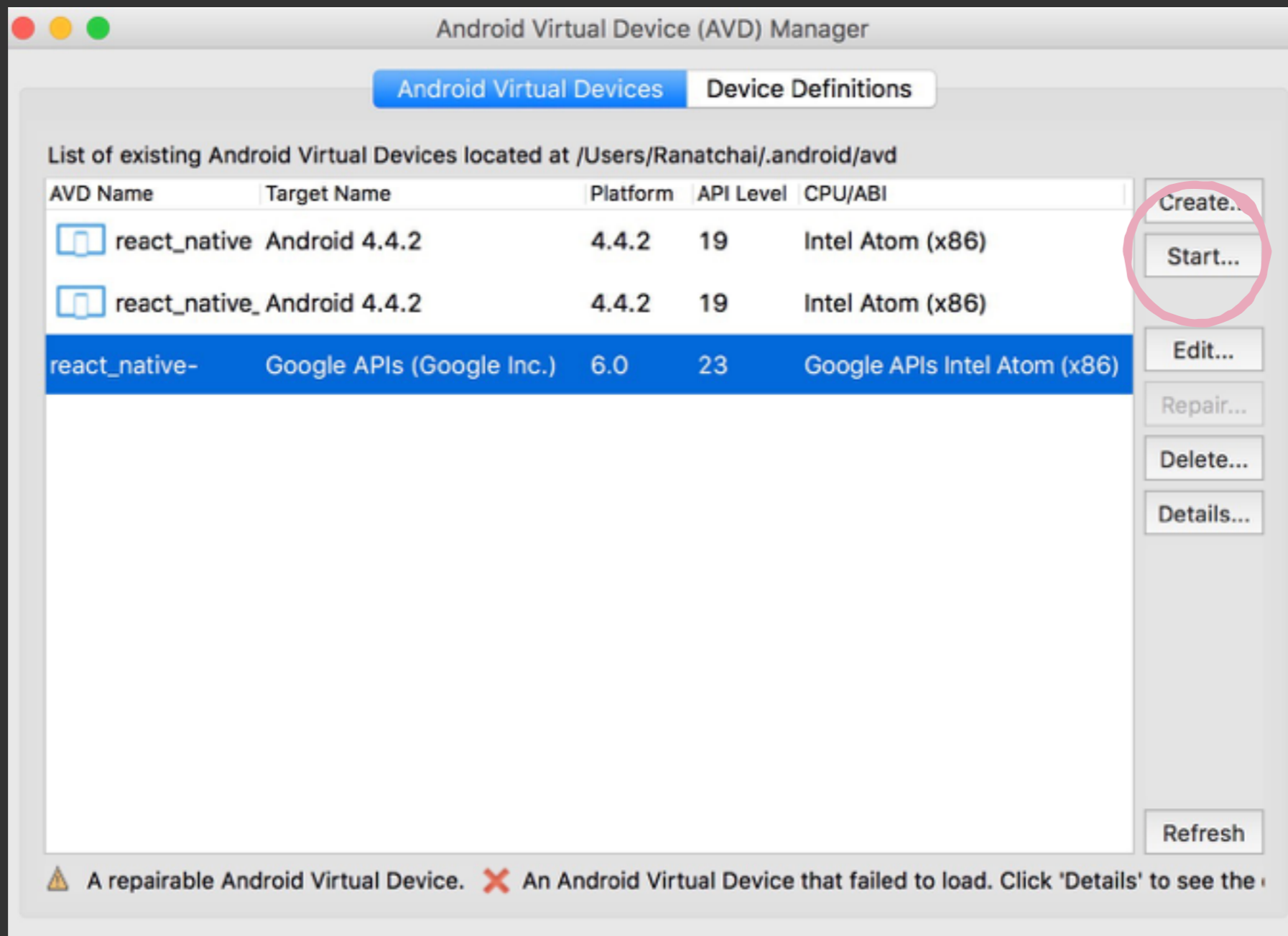
Recrawled this watch 14 times, most recently
/Users/Ranatchai/Project/ReactNativeBKK: kFS
To resolve, please review the information on
https://facebook.github.io/watchman/docs/troubleshooting.html#recrawl
To clear this warning, run:
`watchman watch-del /Users/Ranatchai/Project/ReactNativeBKK`

watchman warning: Recrawled this watch 14 times, most recently because:
/Users/Ranatchai/Project/ReactNativeBKK: kFSEventStreamEventFlagKernelDropped
To resolve, please review the information on
https://facebook.github.io/watchman/docs/troubleshooting.html#recrawl
To clear this warning, run:
`watchman watch-del /Users/Ranatchai/Project/ReactNativeBKK ; watchman watch-project /Users/Ranatchai/Project/ReactNativeBKK`

[8:19:33 PM] <END> Crawling File System (806ms)
[8:19:33 PM] <START> Building in-memory fs for JavaScript
[8:19:34 PM] <END> Building in-memory fs for JavaScript (297ms)
[8:19:34 PM] <START> Building in-memory fs for Assets
[8:19:34 PM] <END> Building in-memory fs for Assets (130ms)
[8:19:34 PM] <START> Building Haste Map
[8:19:34 PM] <START> Building (deprecated) Asset Map
[8:19:34 PM] <END> Building (deprecated) Asset Map (82ms)
[8:19:34 PM] <END> Building Haste Map (199ms)
[8:19:34 PM] <END> Building Dependency Graph (1462ms)
[8:19:57 PM] <START> request:/index.ios.bundle?platform=ios&dev=true&minify=false
[8:19:57 PM] <START> find dependencies
transformed 755/755 (100%)
[8:19:59 PM] <END> find dependencies (1357ms)
[8:19:59 PM] <END> request:/index.ios.bundle?platform=ios&dev=true&minify=false (1493ms)
```

```
$ android avd
```

Run Android Emulator



```
$ react-native avd
```

Run Android

\$ react-native start

```
Running packager on port 8081.

Keep this packager running while developing on any JS projects. Feel
free to close this tab and run your own packager instance if you
prefer.

https://github.com/facebook/react-native

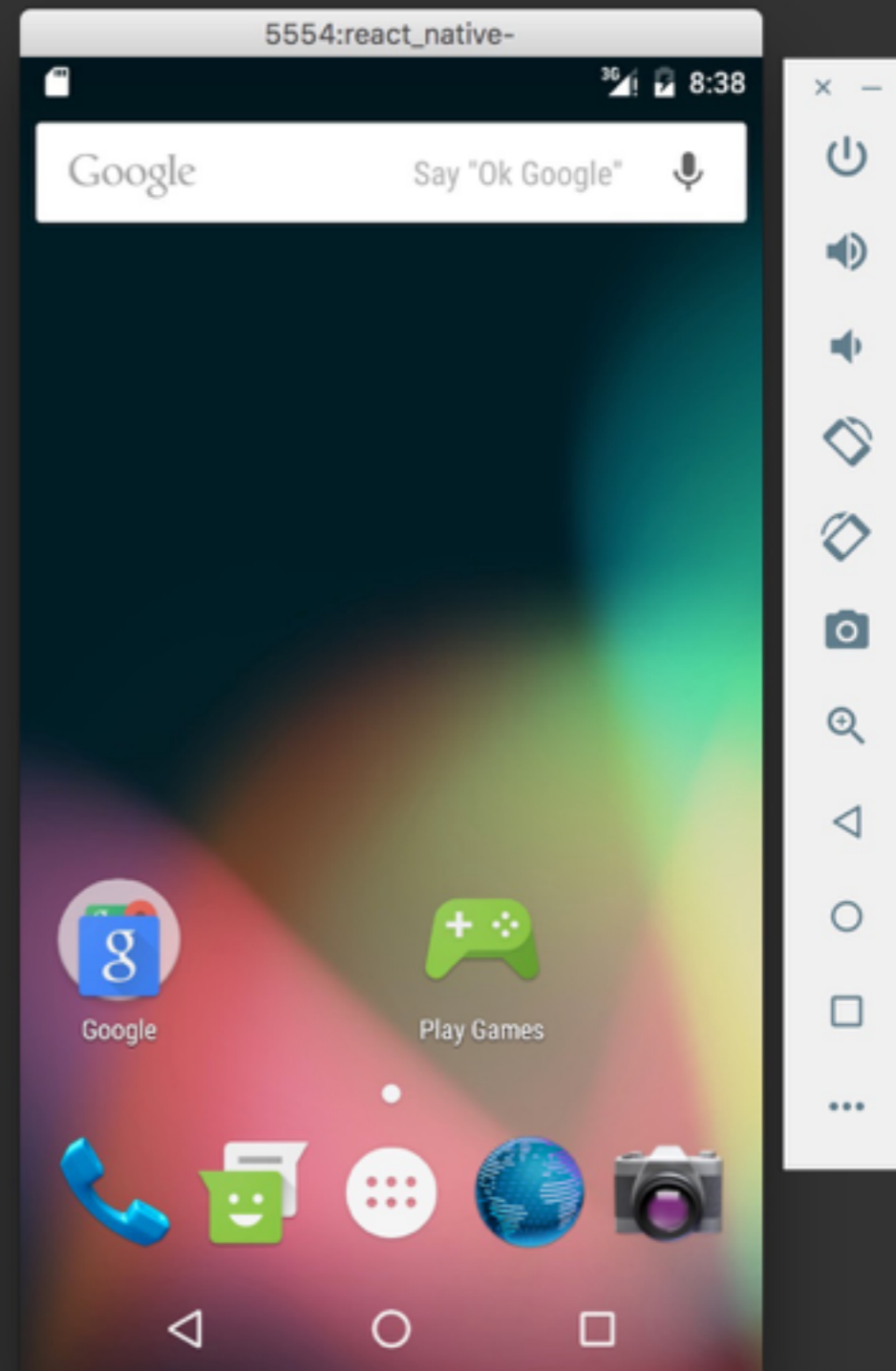
Looking for JS files in
/Users/Ranatchai/Project/ReactNativeBKK
[8:19:32 PM] <START> Building Dependency Graph
[8:19:33 PM] <START> Crawling File System
[Hot Module Replacement] Server listening on /hot

React packager ready.

Recrawled this watch 14 times, most recently because:
/Users/Ranatchai/Project/ReactNativeBKK: kFSEventStreamEventFlagKernelDropped
To resolve, please review the information on
https://facebook.github.io/watchman/docs/troubleshooting.html#recrawl
To clear this warning, run:
`watchman watch-del /Users/Ranatchai/Project/ReactNativeBKK ; watchman watch-project /Users/Ranatchai/Project/ReactNativeBKK`

watchman warning: Recrawled this watch 14 times, most recently because:
/Users/Ranatchai/Project/ReactNativeBKK: kFSEventStreamEventFlagKernelDropped
To resolve, please review the information on
https://facebook.github.io/watchman/docs/troubleshooting.html#recrawl
To clear this warning, run:
`watchman watch-del /Users/Ranatchai/Project/ReactNativeBKK ; watchman watch-project /Users/Ranatchai/Project/ReactNativeBKK`

[8:19:33 PM] <END> Crawling File System (806ms)
[8:19:33 PM] <START> Building in-memory fs for JavaScript
[8:19:34 PM] <END> Building in-memory fs for JavaScript (297ms)
[8:19:34 PM] <START> Building in-memory fs for Assets
[8:19:34 PM] <END> Building in-memory fs for Assets (130ms)
[8:19:34 PM] <START> Building Haste Map
[8:19:34 PM] <START> Building (deprecated) Asset Map
[8:19:34 PM] <END> Building (deprecated) Asset Map (82ms)
[8:19:34 PM] <END> Building Haste Map (199ms)
[8:19:34 PM] <END> Building Dependency Graph (1462ms)
[8:19:57 PM] <START> request:/index.ios.bundle?platform=ios&dev=true&minify=false
[8:19:57 PM] <START> find dependencies
transformed 755/755 (100%)
[8:19:59 PM] <END> find dependencies (1357ms)
[8:19:59 PM] <END> request:/index.ios.bundle?platform=ios&dev=true&minify=false (1493ms)
```



\$ react-native start

```
Running packager on port 8081.

Keep this packager running while developing on any JS projects. Feel
free to close this tab and run your own packager instance if you
prefer.

https://github.com/facebook/react-native

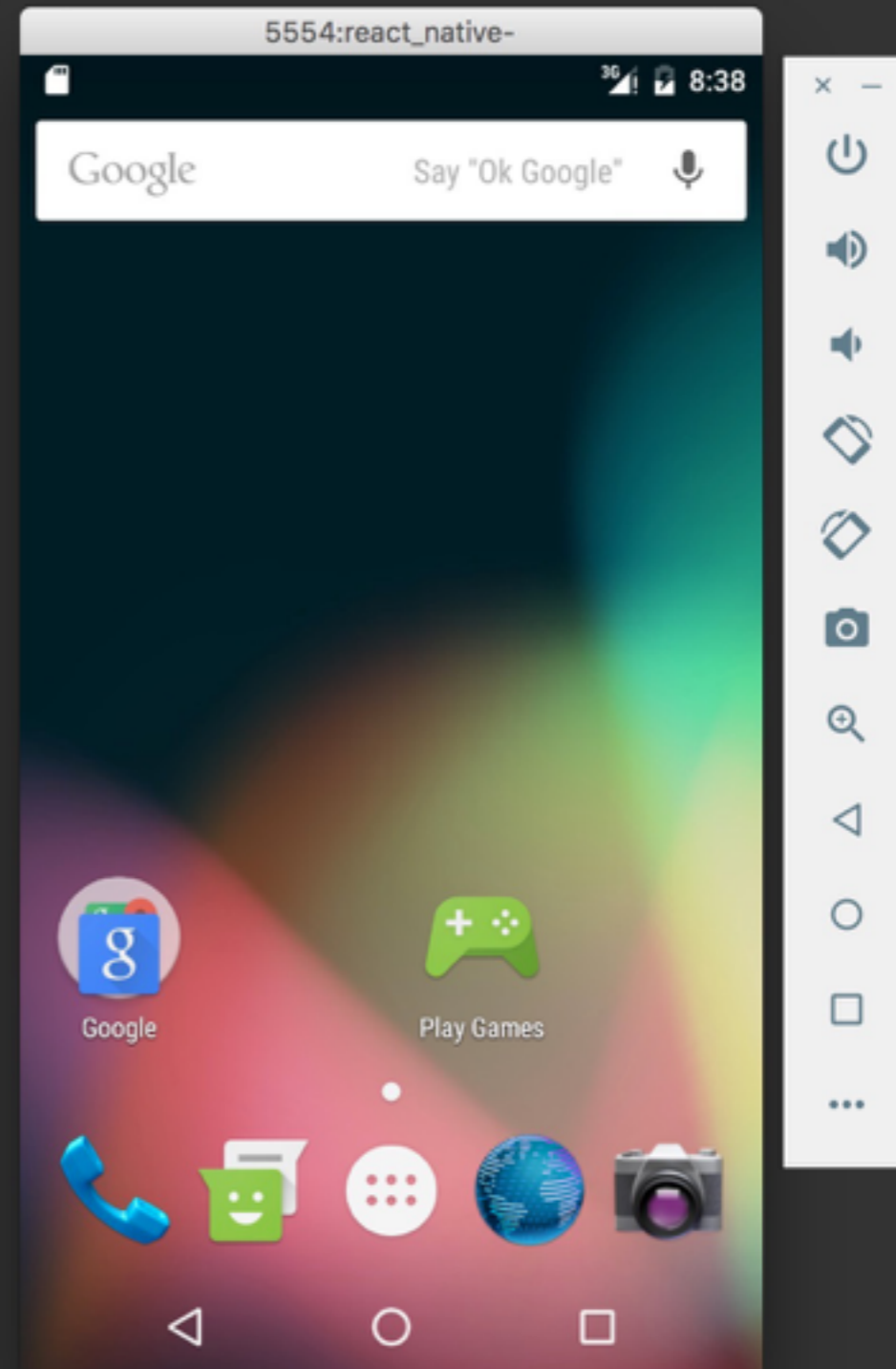
Looking for JS files in
/Users/Ranatchai/Project/ReactNativeBKK
[8:19:32 PM] <START> Building Dependency Graph
[8:19:33 PM] <START> Crawling File System
[Hot Module Replacement] Server listening on /hot

React packager ready.

Recrawled this watch 14 times, most recently because:
/Users/Ranatchai/Project/ReactNativeBKK: kFSEventStreamEventFlagKernelDropped
To resolve, please review the information on
https://facebook.github.io/watchman/docs/troubleshooting.html#recrawl
To clear this warning, run:
`watchman watch-del /Users/Ranatchai/Project/ReactNativeBKK ; watchman watch-project /Users/Ranatchai/Project/ReactNativeBKK`

watchman warning: Recrawled this watch 14 times, most recently because:
/Users/Ranatchai/Project/ReactNativeBKK: kFSEventStreamEventFlagKernelDropped
To resolve, please review the information on
https://facebook.github.io/watchman/docs/troubleshooting.html#recrawl
To clear this warning, run:
`watchman watch-del /Users/Ranatchai/Project/ReactNativeBKK ; watchman watch-project /Users/Ranatchai/Project/ReactNativeBKK`

[8:19:33 PM] <END> Crawling File System (806ms)
[8:19:33 PM] <START> Building in-memory fs for JavaScript
[8:19:34 PM] <END> Building in-memory fs for JavaScript (297ms)
[8:19:34 PM] <START> Building in-memory fs for Assets
[8:19:34 PM] <END> Building in-memory fs for Assets (130ms)
[8:19:34 PM] <START> Building Haste Map
[8:19:34 PM] <START> Building (deprecated) Asset Map
[8:19:34 PM] <END> Building (deprecated) Asset Map (82ms)
[8:19:34 PM] <END> Building Haste Map (199ms)
[8:19:34 PM] <END> Building Dependency Graph (1462ms)
[8:19:57 PM] <START> request:/index.ios.bundle?platform=ios&dev=true&minify=false
[8:19:57 PM] <START> find dependencies
transformed 755/755 (100%)
[8:19:59 PM] <END> find dependencies (1357ms)
[8:19:59 PM] <END> request:/index.ios.bundle?platform=ios&dev=true&minify=false (1493ms)
```



\$ react-native run-android

Run Android

\$ react-native start

```
Running packager on port 8081.

Keep this packager running while developing on any JS projects. Feel
free to close this tab and run your own packager instance if you
prefer.

https://github.com/facebook/react-native

Looking for JS files in
/Users/Ranatchai/Project/ReactNativeBKK

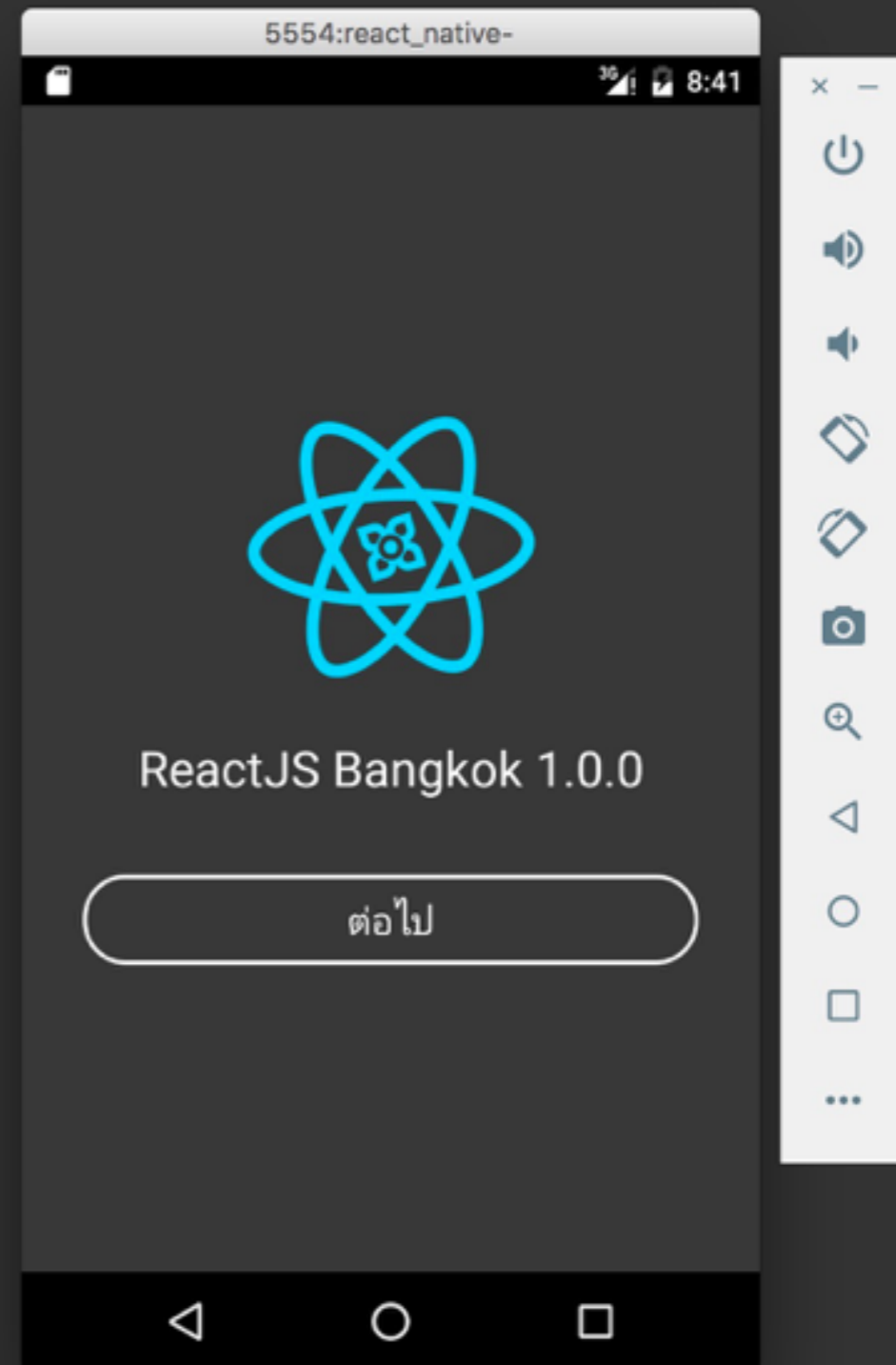
[8:19:32 PM] <START> Building Dependency Graph
[8:19:33 PM] <START> Crawling File System
[Hot Module Replacement] Server listening on /hot

React packager ready.

Recrawled this watch 14 times, most recently because:
/Users/Ranatchai/Project/ReactNativeBKK: kFSEventStreamEventFlagKernelDropped
To resolve, please review the information on
https://facebook.github.io/watchman/docs/troubleshooting.html#recrawl
To clear this warning, run:
`watchman watch-del /Users/Ranatchai/Project/ReactNativeBKK ; watchman watch-project /Users/Ranatchai/Project/ReactNativeBKK`

watchman warning: Recrawled this watch 14 times, most recently because:
/Users/Ranatchai/Project/ReactNativeBKK: kFSEventStreamEventFlagKernelDropped
To resolve, please review the information on
https://facebook.github.io/watchman/docs/troubleshooting.html#recrawl
To clear this warning, run:
`watchman watch-del /Users/Ranatchai/Project/ReactNativeBKK ; watchman watch-project /Users/Ranatchai/Project/ReactNativeBKK`

[8:19:33 PM] <END> Crawling File System (806ms)
[8:19:33 PM] <START> Building in-memory fs for JavaScript
[8:19:34 PM] <END> Building in-memory fs for JavaScript (297ms)
[8:19:34 PM] <START> Building in-memory fs for Assets
[8:19:34 PM] <END> Building in-memory fs for Assets (130ms)
[8:19:34 PM] <START> Building Haste Map
[8:19:34 PM] <START> Building (deprecated) Asset Map
[8:19:34 PM] <END> Building (deprecated) Asset Map (82ms)
[8:19:34 PM] <END> Building Haste Map (199ms)
[8:19:34 PM] <END> Building Dependency Graph (1462ms)
[8:19:57 PM] <START> request:/index.ios.bundle?platform=ios&dev=true&minify=false
[8:19:57 PM] <START> find dependencies
transformed 755/755 (100%)
[8:19:59 PM] <END> find dependencies (1357ms)
[8:19:59 PM] <END> request:/index.ios.bundle?platform=ios&dev=true&minify=false (1493ms)
```

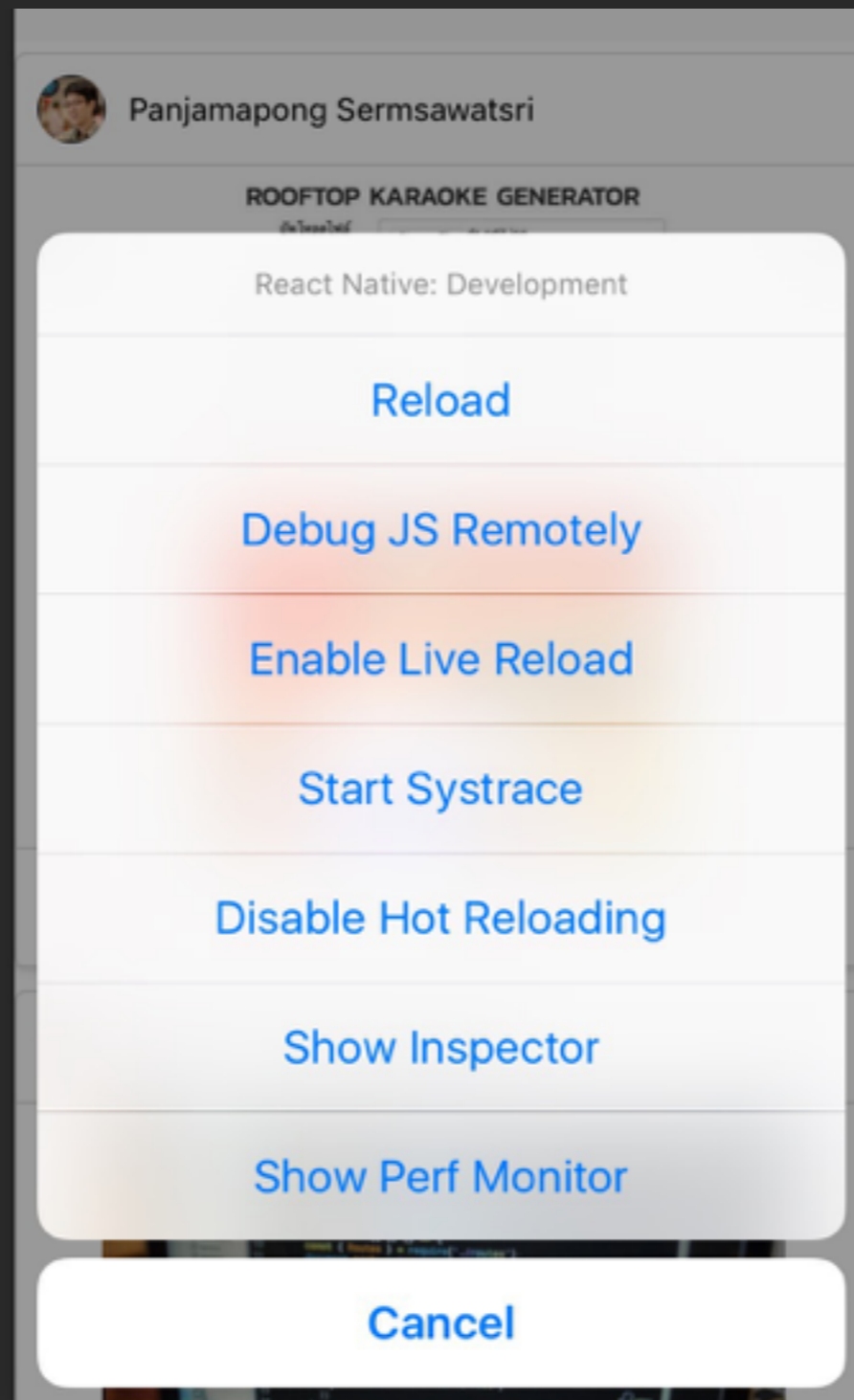


\$ react-native run-android

Run Android

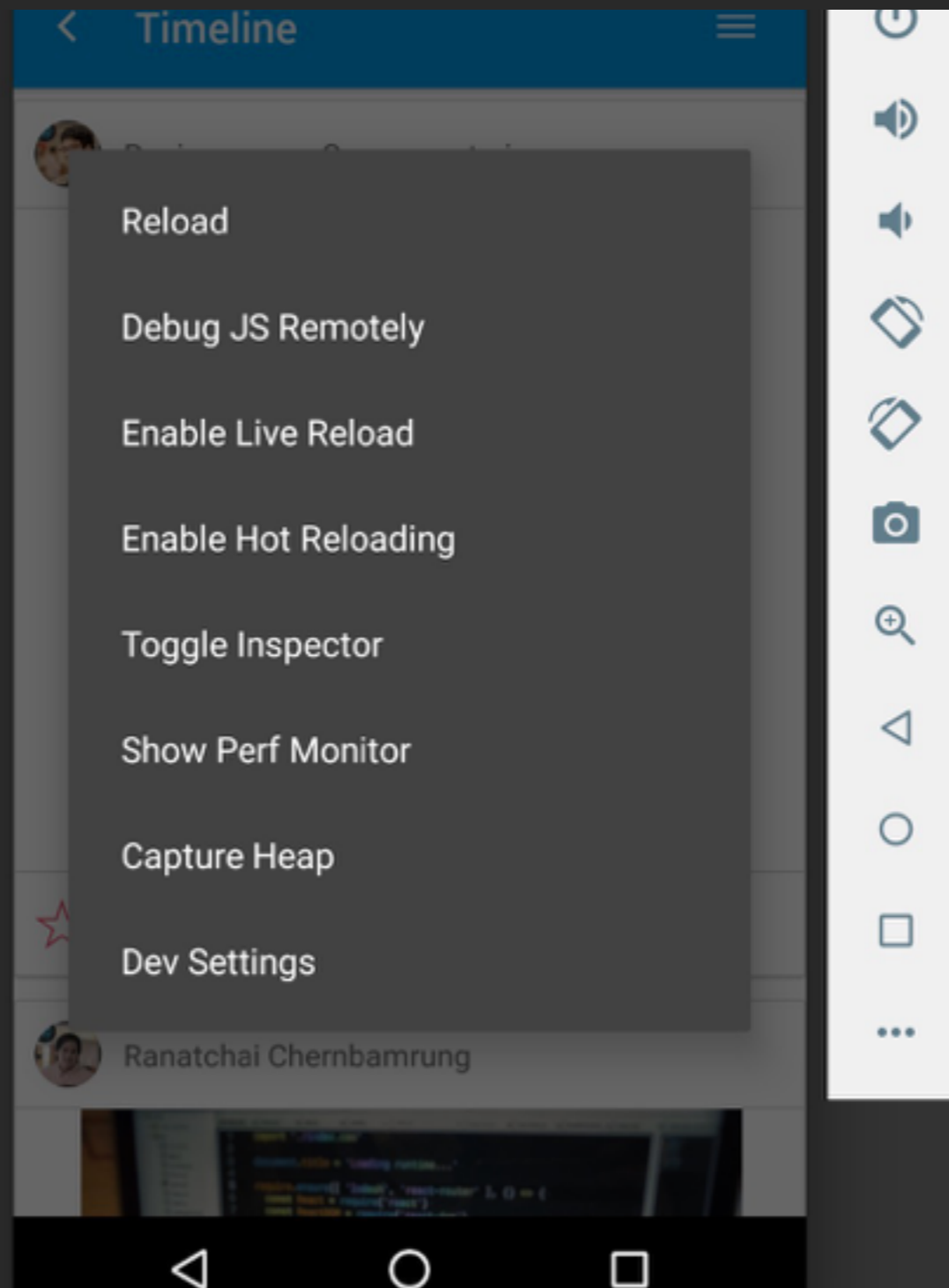
Open Dev tools iOS

CMD + D



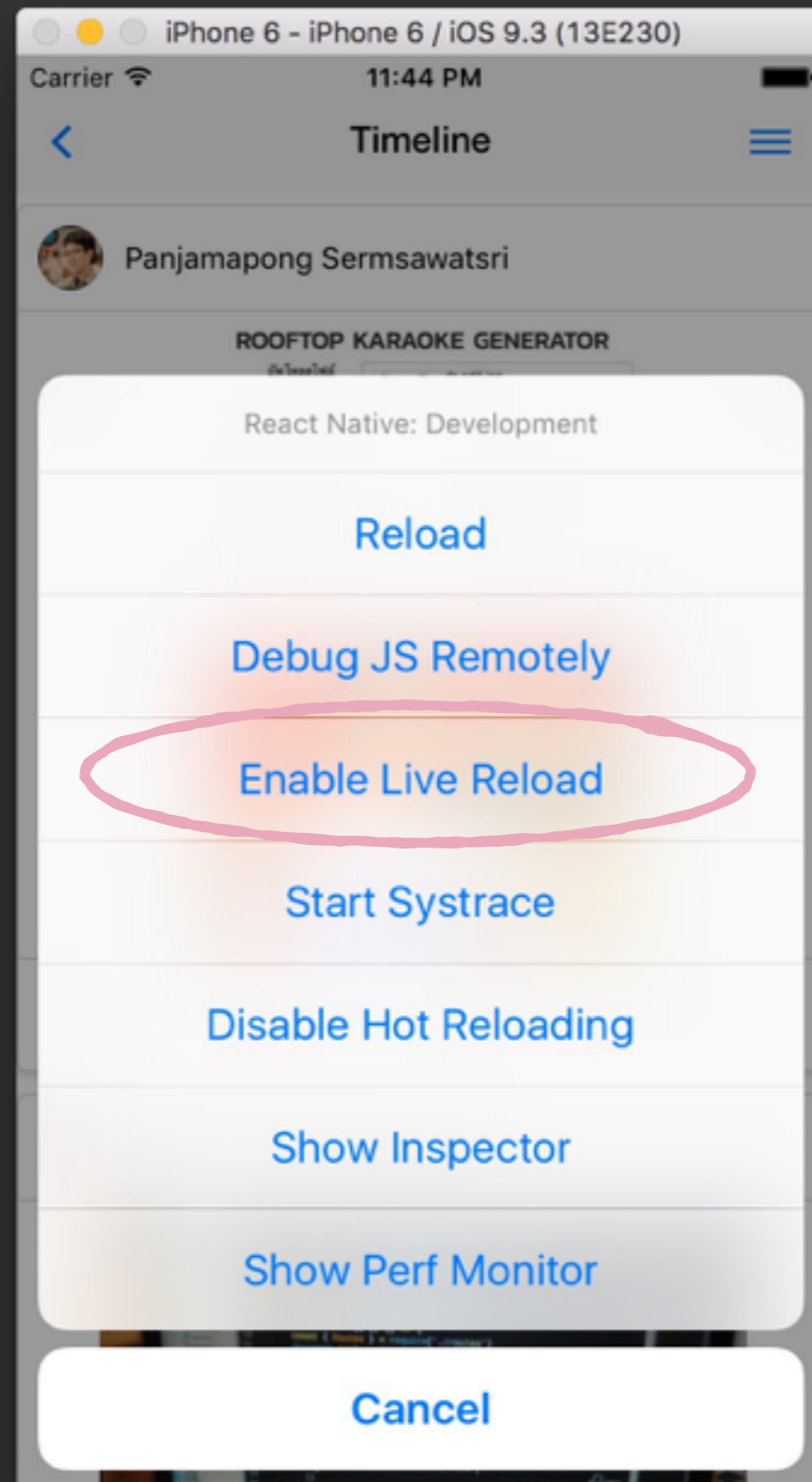
Open Dev tools Android

CMD + M



Hot reload and Live reload

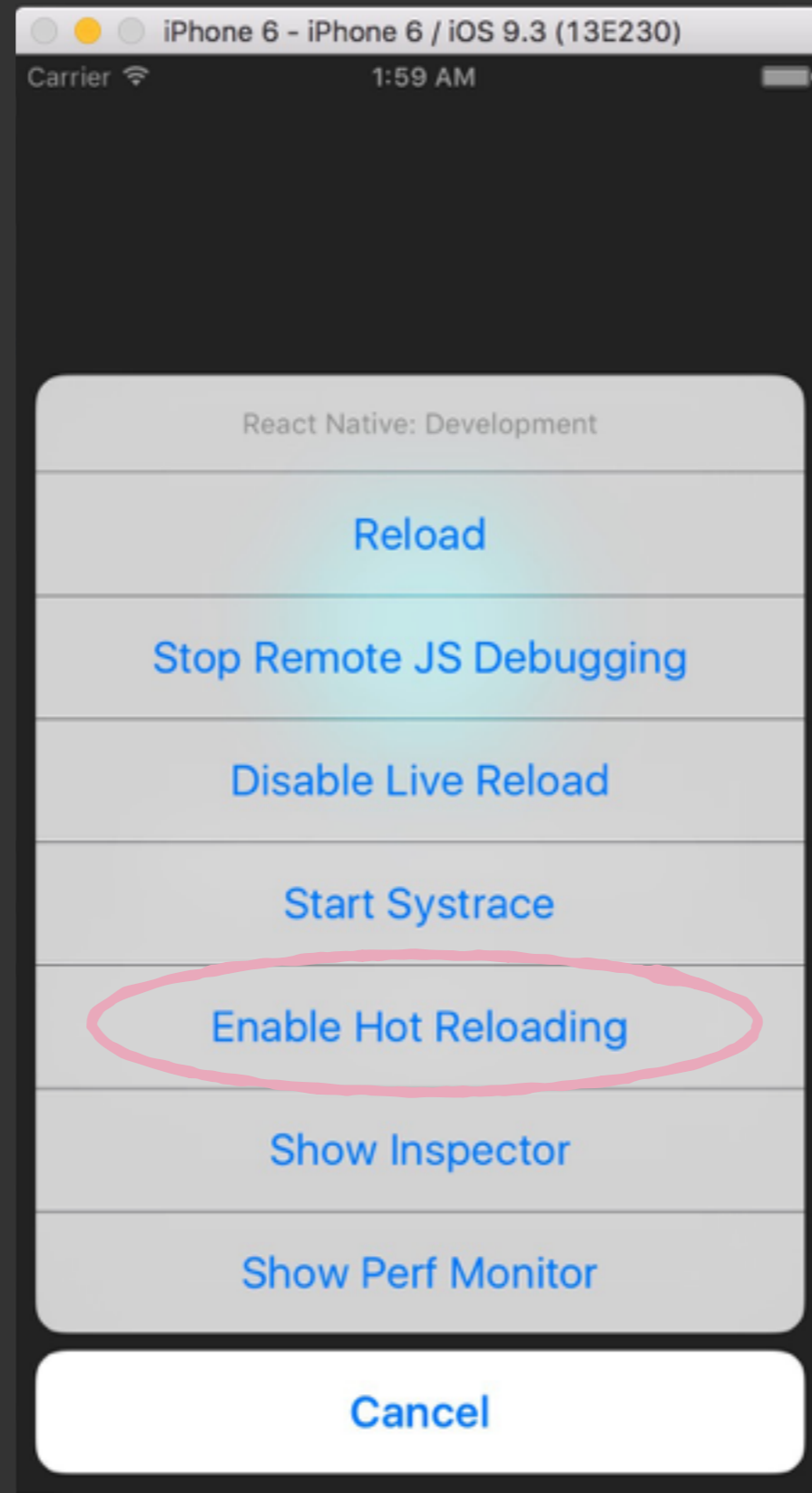
Enable Live Reload



Live Reload Demo

<https://youtu.be/urbcszfA7yU>

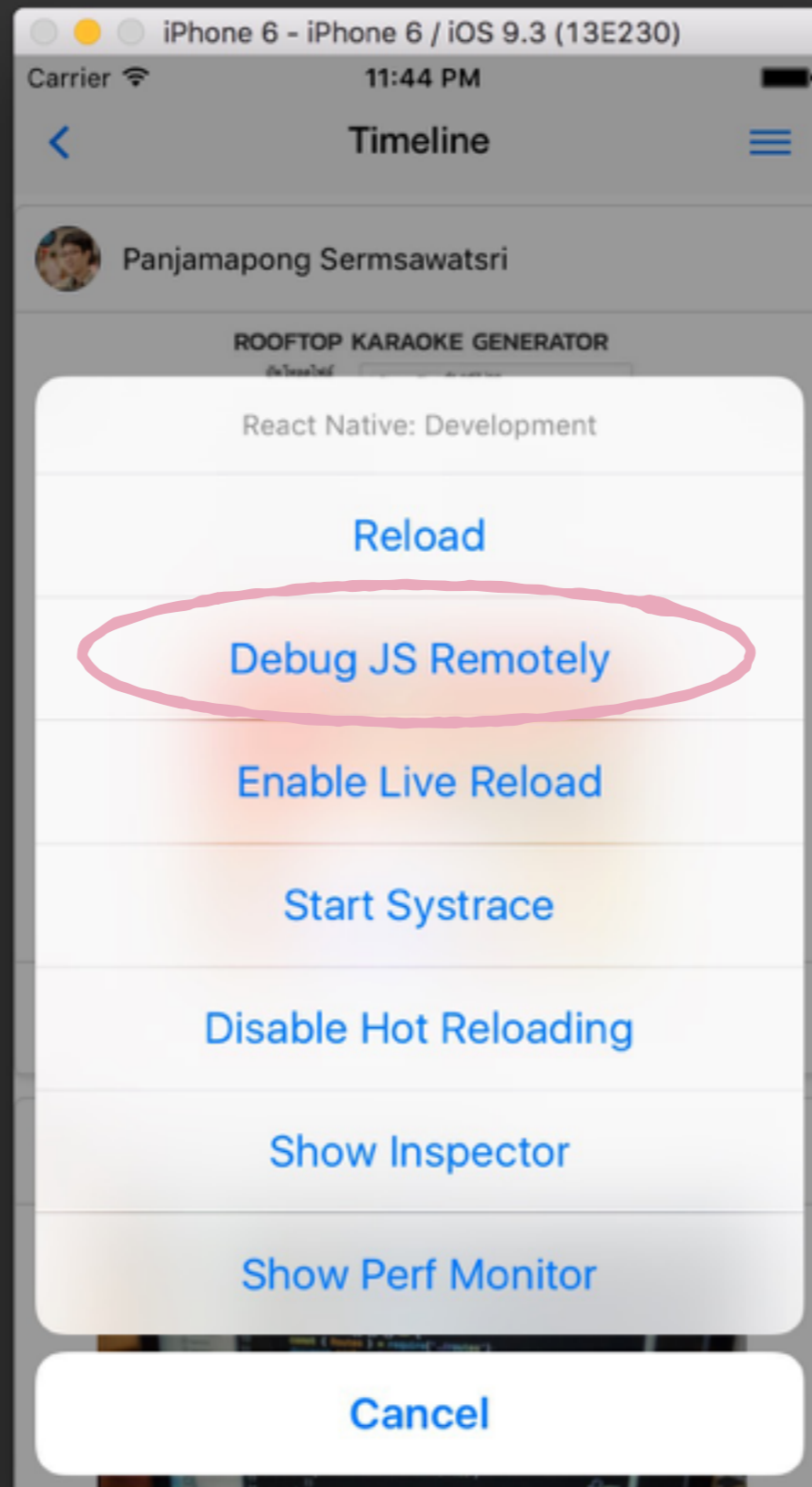
Enable Hot Reload



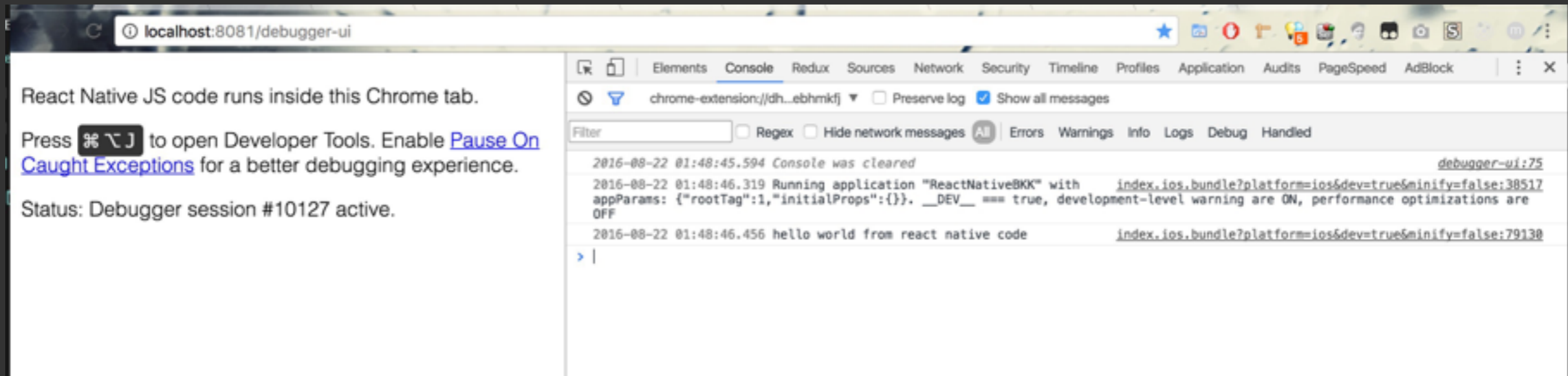
Hot Reload Demo

<https://youtu.be/bfKyoBcu8b0>

Enable Remote Debugging



Debugger: Chrome Dev Tools



```
104     );  
105   }  
106  
107   render() {  
108     console.log('hello world from react native code');  
109   }
```

React Native Debugger

The screenshot displays the React Native Debugger interface. The title bar reads "React Native Debugger - Debugger session #10639 active." The interface is divided into several panels:

- Inspector:** Shows a component tree with a selected `<View style=135>` component. The tree structure includes `<AppContainer>`, `<View style=135>`, `<RCTView style=135>`, `<View key="1" ref=`, and `<YellowBox>`.
- Diff:** Shows "(states are equal)".
- Console:** Displays the following messages:
 - Console was cleared (index.js:121)
 - Running application "ReactNativeBKK" with appParams: {"rootTag":1,"initialProps":{}}. __DEV__ === true, development-level warnings are ON, performance optimizations are OFF (infoLog.js:17)
 - hello world from react native code (login.js:108)
- React Native Style Editor:** Shows a "flex" property with a value of "1".
- Terminal:** Contains the command `$ brew update && brew cask install react-native-debugger`.

React Native Debugger

The screenshot displays the React Native Debugger interface. The top window shows the console with the following messages:

- Console was cleared (index.js:121)
- Running application "ReactNativeBKK" with appParams: {"rootTag":1,"initialProps":{}}. __DEV__ === true, development-level warnings are ON, performance optimizations are OFF (infoLog.js:17)
- hello world from react native code (login.js:108, circled in red)

The code editor shows the following code snippet:

```
104 );  
105 }  
106  
107 render() {  
108   console.log('hello world from react native code');  
109 }
```

The component inspector shows the following structure:

```
<View> ($r in the console)  
  <AppContainer>  
    <View style=135>  
      <RCTView style=135>  
        <View key="1" ref=...>  
          <YellowBox>_</Yel...>  
        </RCTView>  
      </View>  
    </AppContainer>
```

The React Native Style Editor shows the following style properties:

```
flex :  
1
```

The terminal at the bottom shows the command:

```
$ brew update && brew cask install react-native-debugger
```

React Native Debugger

The screenshot displays the React Native Debugger interface with the following components:

- Inspector:** Shows a component tree with a filter. The selected component is a `<View style=135>` containing an `<RCTView style=135>` which contains a `<View key="1" ref="main" colla>` and a `<YellowBox>...</YellowBox>`.
- Diff:** Shows "(states are equal)".
- React Native Style Editor:** Shows the props for the selected `<View>` component, including `children: Array[3]` and `style: 135`. The `flex` property is currently set to `1`.
- Sources:** Shows the `login.js` file with the following code snippet:

```
95     onPress={() => {
96       this.setState({
97         showLoginForm: true
98       })
99     }}
100     >
101     หน่อย
102   </Button>
103 </View>
104
105 );
106
107 render() {
108   console.log('hello world from react native code');
109
110   const { showLoginForm, showLoading } = this.state;
111   // const showLoginForm = this.state.showLoginForm;
112   // const showLoading = this.state.showLoading;
113   return (
114     <View style={styles.container}>
115       <View style={styles.content}>
116         <View style={[styles.logoContainer, showLoginForm && styles:
117           <Image
118             source={require('./assets/logo.png')}
119           </Image>
120         </View>
121       </View>
122     </View>
123   );
```
- Debugger:** Shows the current execution state with "Not Paused" and "Scope Watch" options.

At the bottom of the interface, there is a terminal window with the following command:

```
$ brew update && brew cask install react-native-debugger
```


Thing We Learn

- Dev Process very fast with hot reload
- Easy Debugging Code with Chrome Dev Tools

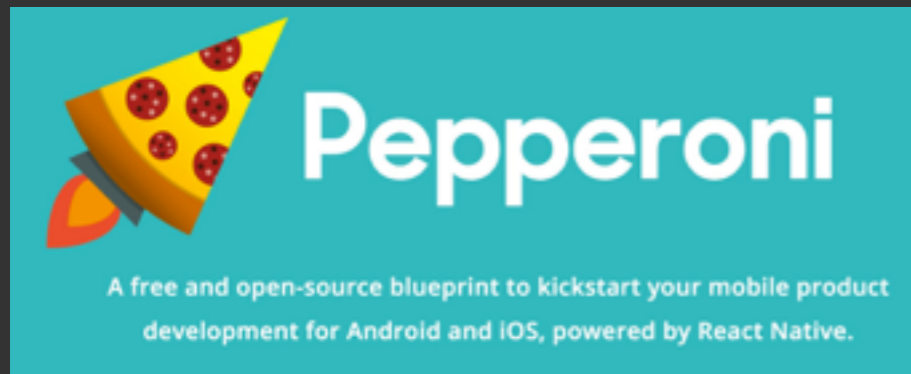
<https://github.com/ranatchai/react-native-bkk-example>

Everything is Great but...

- Can we use React Native with Redux ?
- Testing framework ?
- Offline Storage ?
- Instant App Update ? (Code Push)
- CI & CD ?

Study from Full Stack Boilerplate

<https://github.com/futurice/pepperoni-app-kit>



Application Blueprint

- Always up-to-date [React Native](#) scaffolding
- Modular and well-documented structure for application code
- [Redux](#) and [ImmutableJS](#) for safe and Reasonaboutable™ state management
- [Redux Loop](#) for Elm-style controlled side effects
- Redux-managed Navigators for Stack-based and Tabbed navigation with [NavigationExperimental](#)
- Disk-persisted application state caching for offline support and snappy startup performance
- Clean and testable service layer for interacting with RESTful APIs
- ⚠️ Sample app to show how to wire it all together
- ⭐ JSON Web Token authentication
- ⭐ Multi-environment configuration (dev, staging, production) for iOS and Android
- ⭐ Built-in error handling and customizable error screens

Testing Setup

- [Mocha](#) for unit testing application code
- [Enzyme](#) and fully mocked React Native for unit testing UI components
- [Istanbul](#) code coverage
- Utilities for end-to-end integration testing Redux state, including side effects and asynchronous actions

Development & Deployment Infrastructure

- [Auth0](#) for ready-to-use login and signup screens, user authentication and identity management
- [Bitrise.io](#) configurations for Continuous Integration and beta app distribution
- ⚠️ [Google Tag Manager](#) analytics
- ⭐ [Microsoft Code Push](#) for Continuous Deployment and instant app updates

Who are using React Native?



Facebook



Facebook Ads
Manager



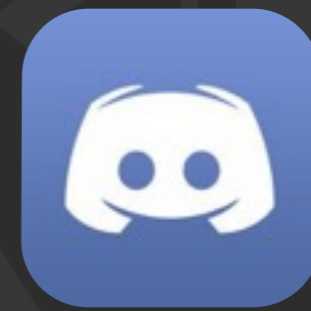
Facebook
Groups



Gyroscope



QQ



Discord

Ready for Production?

- New versions are coming very fast, some have breaking changes
- Open-sourced components are sufficient, but there are some open issues
- There are some limitations on styling components in tricky cases, native bridge is needed
- TakeMeTour was able to launch its first version in both platforms within 2 weeks of development
- An intern took the code written for iOS and made it run **NATIVELY** on Android within 2 days

Tip & Trick

<https://github.com/ranatchai/react-native-bkk-example>

How to Find Component

<https://js.coach>

The screenshot shows the JS.COACH website interface. On the left, there is a navigation sidebar with categories like 'React Native', 'Webpack', and 'Babel'. The main content area displays search results for 'react-native-slider'. The top result is 'react-native-slider' by jeanregisser, which is highlighted. Below the search results, there is a detailed view of the 'react-native-slider' component. This view includes the component's name, version (v0.9.1), download statistics (3k/month), license (MIT), and platform support (ios, android). A description states it is a pure JavaScript <Slider> component for react-native. Below the text, there is a visual demonstration of the slider component with various styles and colors, each accompanied by a code snippet and a version number (0.2).

JS.COACH

SEARCH

MENU

react-native-118n updated today by AlexanderZaytsev
React Native and 118n.js integration.
244 stars 4.5K installs/mo iOS Android

react-native-wordpress-editor updated today by wix
React Native Wrapper for WordPress Rich Text Editor. The WordPress-Editor is the text editor used in the official WordPress mobile apps to create and edit pages & posts.
34 stars 2K installs/mo iOS Android

react-native-ble-manager updated today by innoveit
A BLE module for react native.
46 stars 279 installs/mo iOS Android

react-native-carousel-control updated today by machadogj
React Native Carousel control with support for iOS and Android.
54 stars 119 installs/mo

react-native-modalbox updated today by maxs15
A <Modal/> component for react-native.
476 stars 4.8K installs/mo iOS Android

react-native-slider updated today by jeanregisser
A pure JavaScript <Slider /> component for react-native.
207 stars 2.8K installs/mo iOS Android

react-native-share-extension updated today by alinz
Share extension using react-native for both ios and android.
46 stars 59 installs/mo iOS Android

react-native-epinkit updated today by maxs15

react-native-slider updated today by jeanregisser
View on GitHub

react-native-slider updated today by jeanregisser

react-native-slider

npm v0.9.1 downloads 3k/month license MIT platform ios | android

A pure JavaScript <Slider> component for react-native. This is still very much a work in progress, ideas and contributions are very welcome.

Carrier 11:50 AM

<React.Slider/> 0.2

<Slider/> with default style 0.2

<Slider/> with min, max and custom tints 0.2

<Slider/> with custom style 0.2

<Slider/> with custom style #2 0.2

<Slider/> with custom style #3 0.2

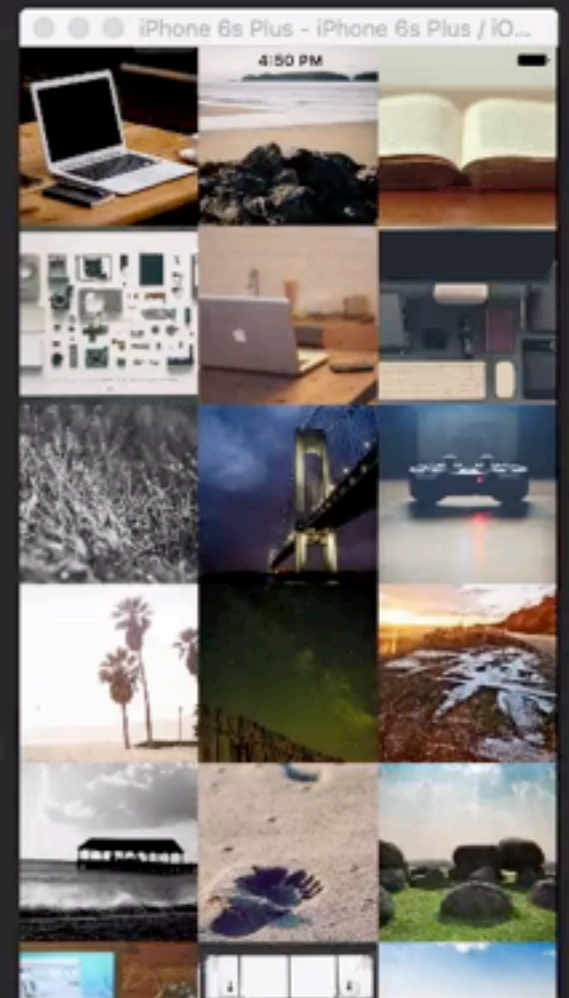
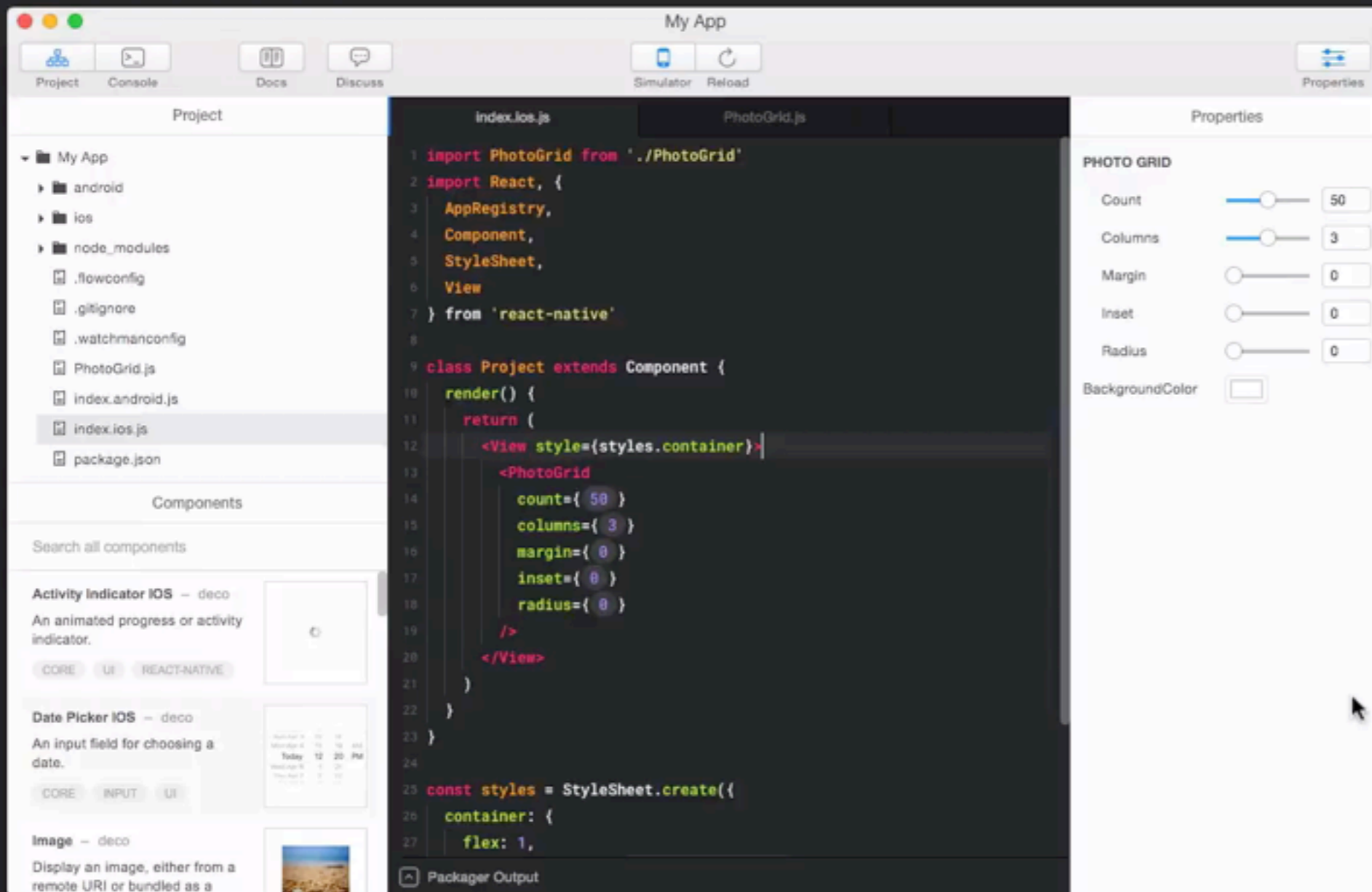
<Slider/> with custom style #4 0.2

<Slider/> with custom style #5 0.2

<Slider/> with custom style #6 0.2

Next Generation IDE for building React Native

<https://decosoftware.com>



React Native for Web & Desktop App

Web

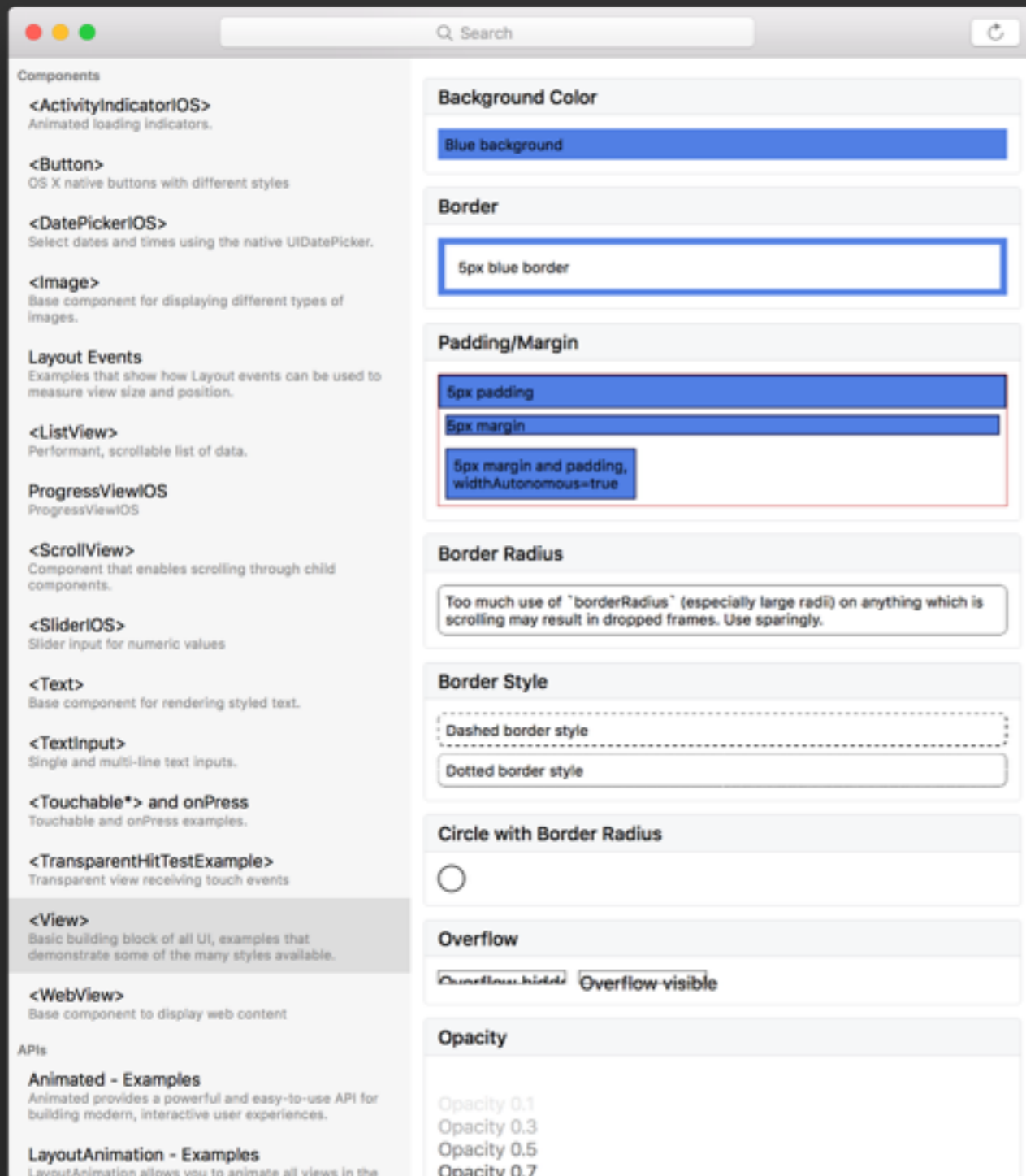
[https://github.com/necolas/
react-native-web](https://github.com/necolas/react-native-web)

Mac

[https://github.com/ptmt/
react-native-macos](https://github.com/ptmt/react-native-macos)

Ubuntu

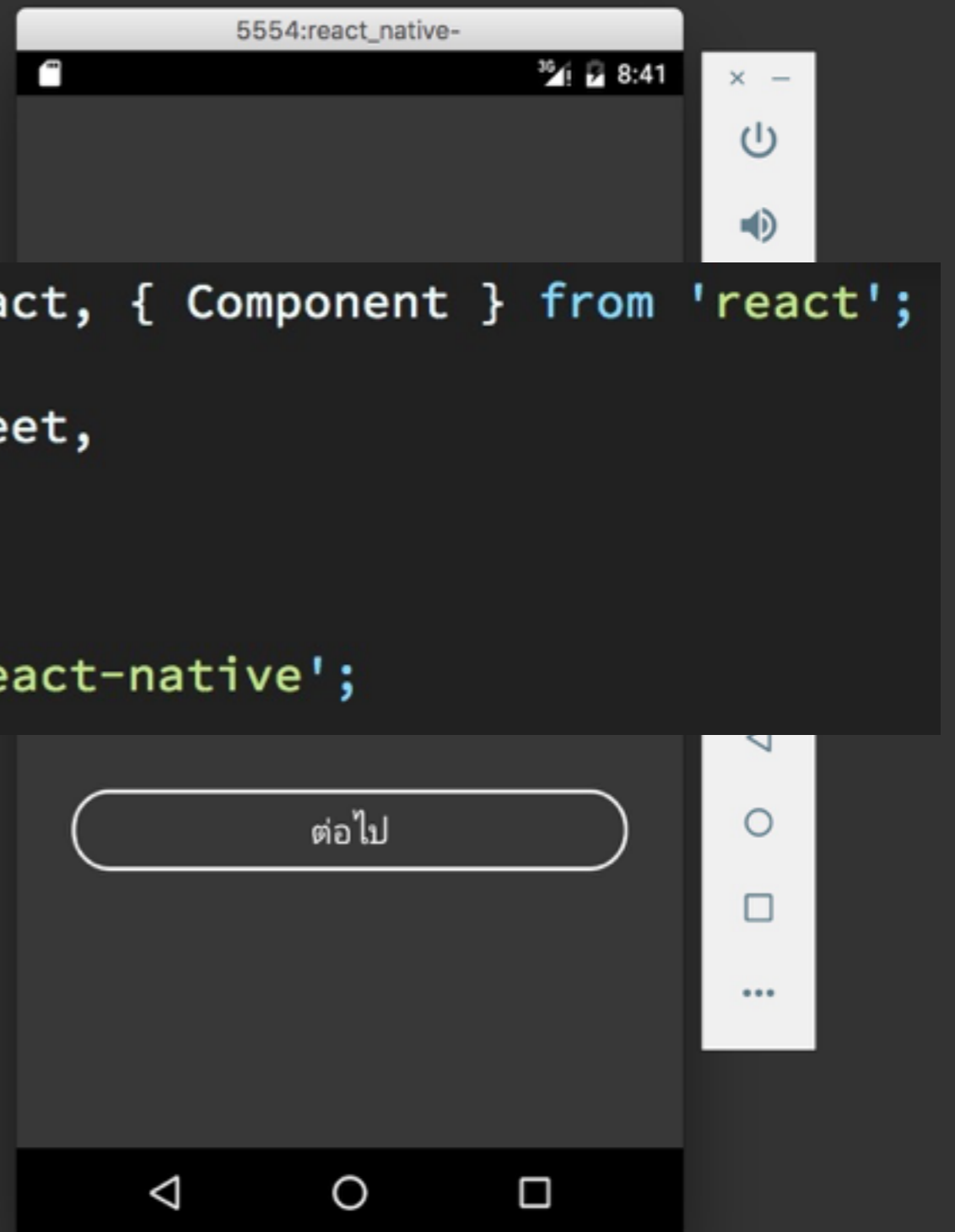
[https://github.com/
CanonicalLtd/react-native](https://github.com/CanonicalLtd/react-native)



Old



iOS



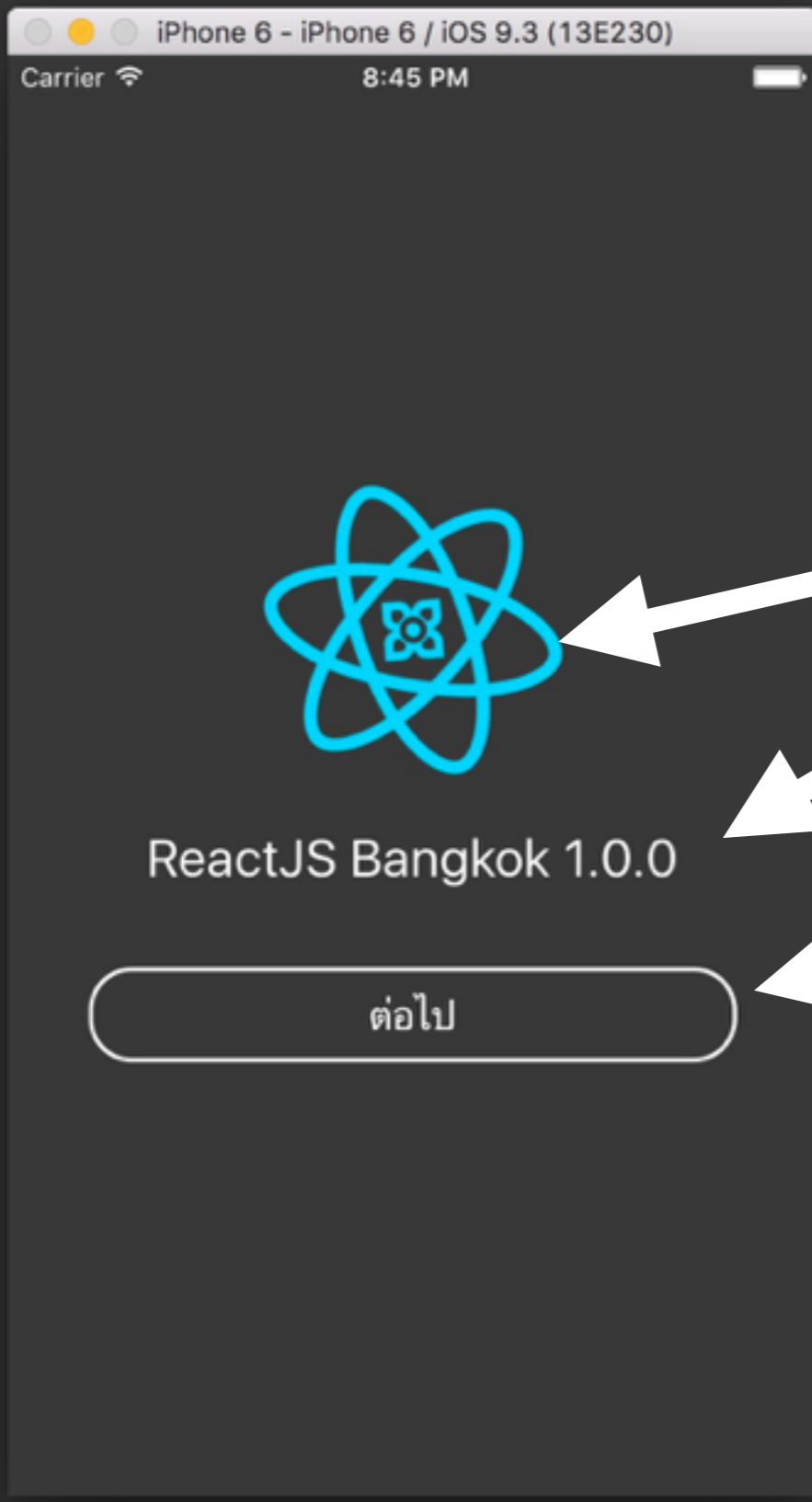
Android



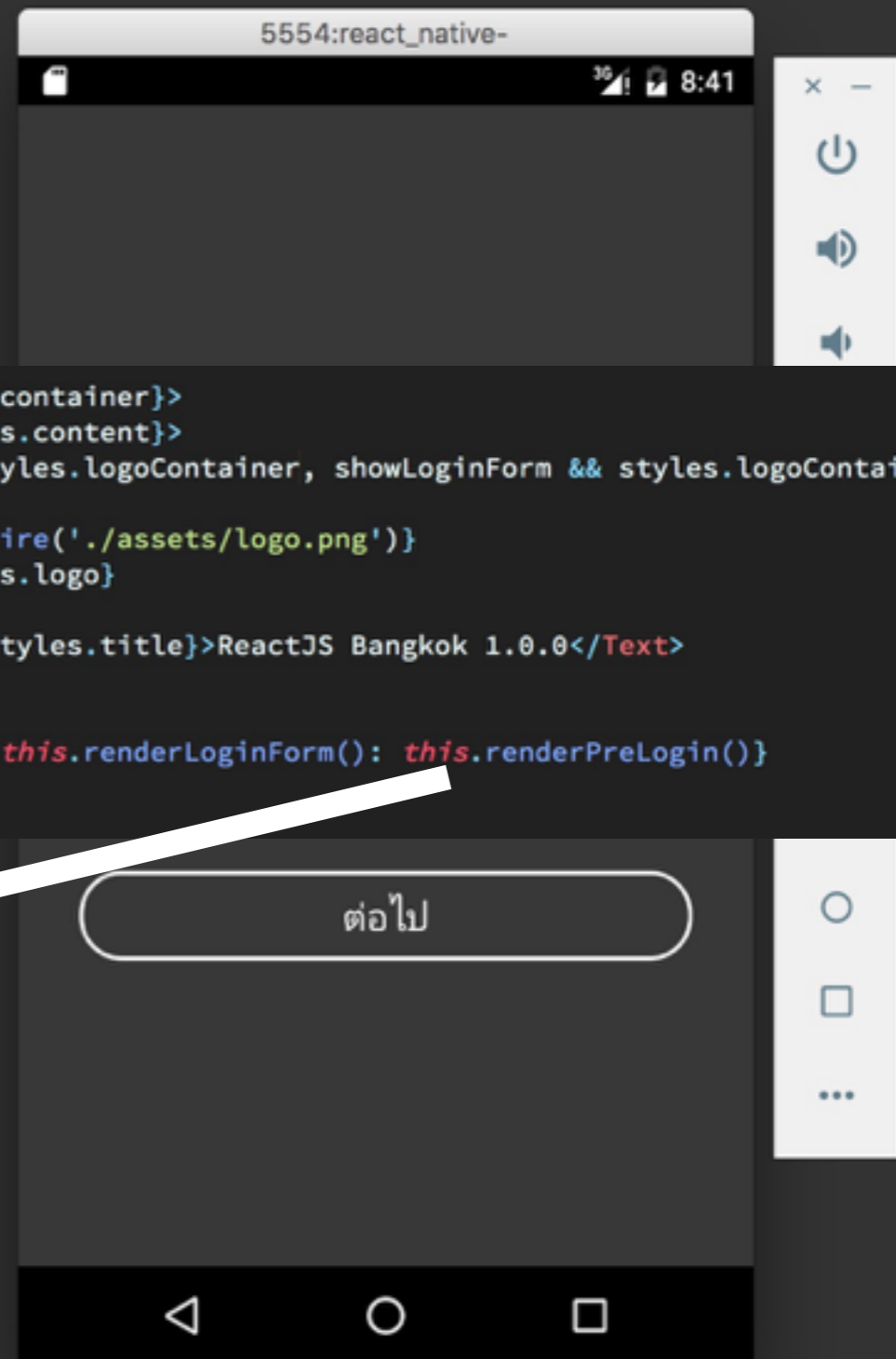
```
import React, { Component } from 'react';
import {
  StyleSheet,
  Text,
  View,
  Image,
} from 'react-native';
```

```
render() {
  const { showLoginForm, showLoading } = this.state;
  // const showLoginForm = this.state.showLoginForm;
  // const showLoading = this.state.showLoading;
  return (
    <View style={styles.container}>
      <View style={styles.content}>
        <View style={[styles.logoContainer, showLoginForm && styles.logoContainerSmall]}>
          <Image
            source={require('./assets/logo.png')}
            style={styles.logo}
          />
          <Text style={styles.title}>ReactJS Bangkok 1.0.0</Text>
        </View>
        <View>
          {showLoginForm? this.renderLoginForm(): this.renderPreLogin()}
        </View>
        </View>
        {showLoading && (
          <View style={styles.loadingContainer}>
            <ActivityIndicator size="large" color="white" style={styles.loading}/>
          </View>
        )}
      </View>
    </View>
  );
}
```

iOS

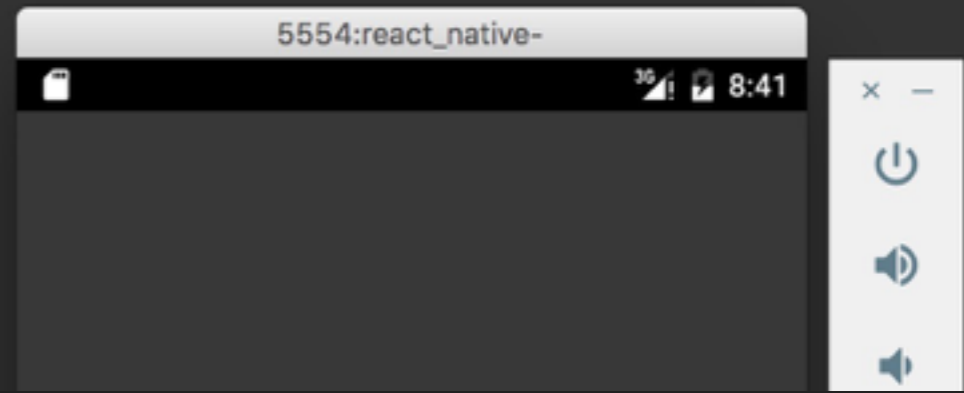
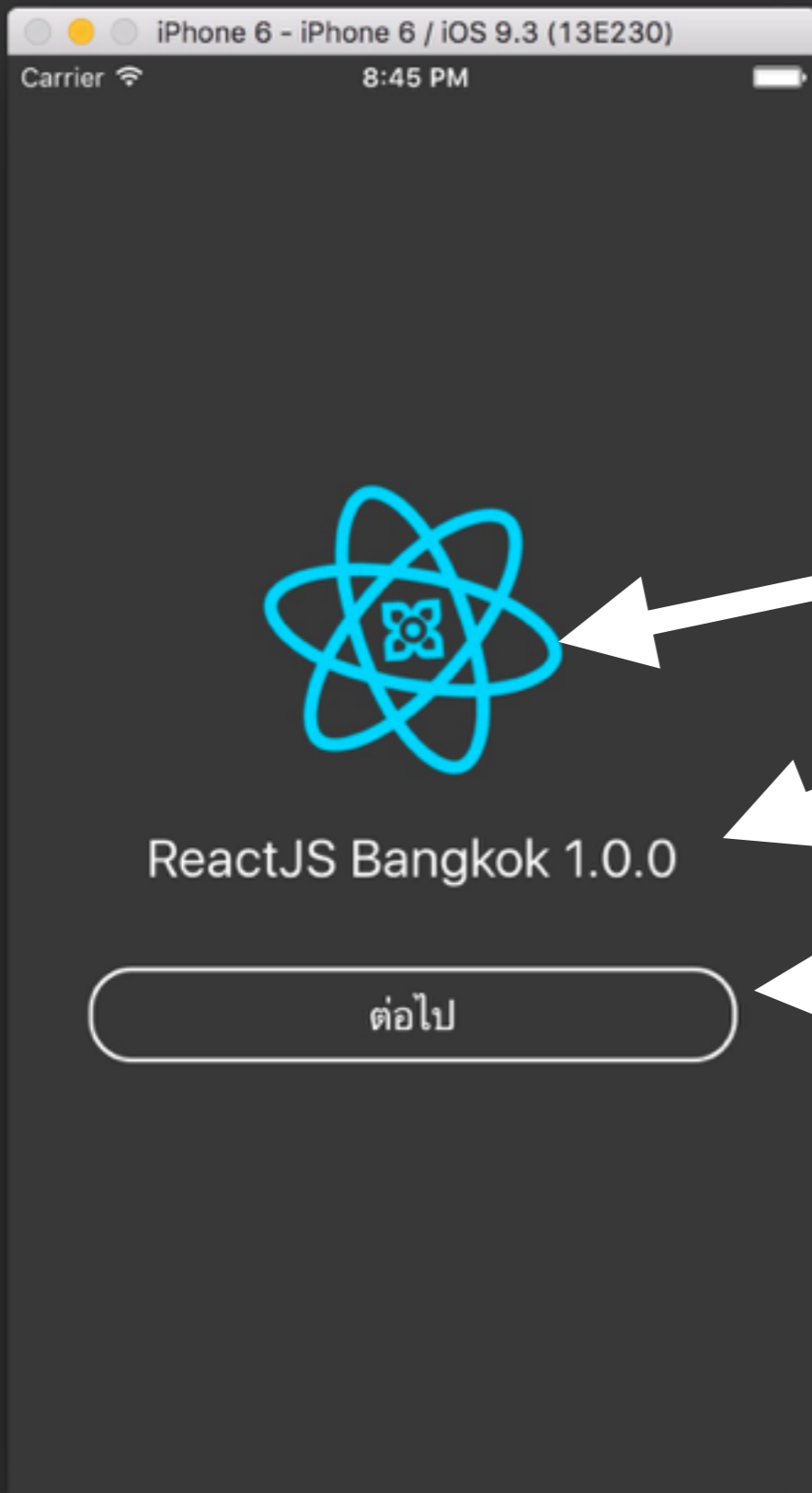


iOS



Android

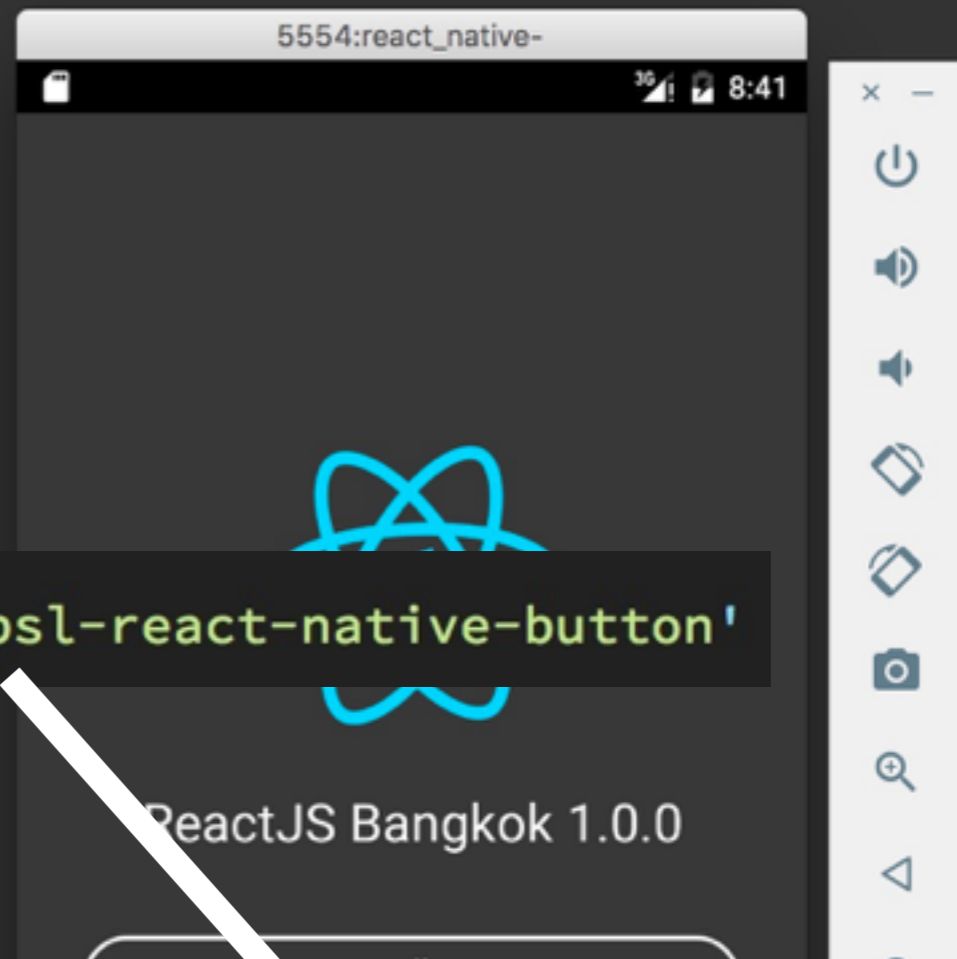
```
<View style={styles.container}>
  <View style={styles.content}>
    <View style={[styles.logoContainer, showLoginForm && styles.logoContainerSmall]}>
      <Image
        source={require('./assets/logo.png')}
        style={styles.logo}
      />
      <Text style={styles.title}>ReactJS Bangkok 1.0.0</Text>
    </View>
    <View>
      {showLoginForm? this.renderLoginForm(): this.renderPreLogin()}
    </View>
  </View>
</View>
```



```
<View style={styles.container}>  
  <View style={styles.content}>  
    <View style={[styles.logoContainer, showLoginForm && styles.logoContainerSmall]}>  
      <Image  
        source={require('./assets/logo.png')}  
        style={styles.logo}  
      />  
      <Text style={styles.title}>ReactJS Bangkok 1.0.0</Text>  
    </View>  
    <View  
      showLoginForm? this.renderLoginForm(): this.renderPreLogin()  
    </View>  
  </View>  
</View>
```

```
renderPreLogin() {  
  return (  
    <View style={{ width: 300 }}>  
      <Button  
        style={styles.buttonStyle}  
        textStyle={styles.buttonTextStyle}  
        onPress={() => {  
          this.setState({  
            showLoginForm: true  
          })  
        }}  
      >  
        ต่อไป  
      </Button>  
    </View>  
  );  
}
```

iOS



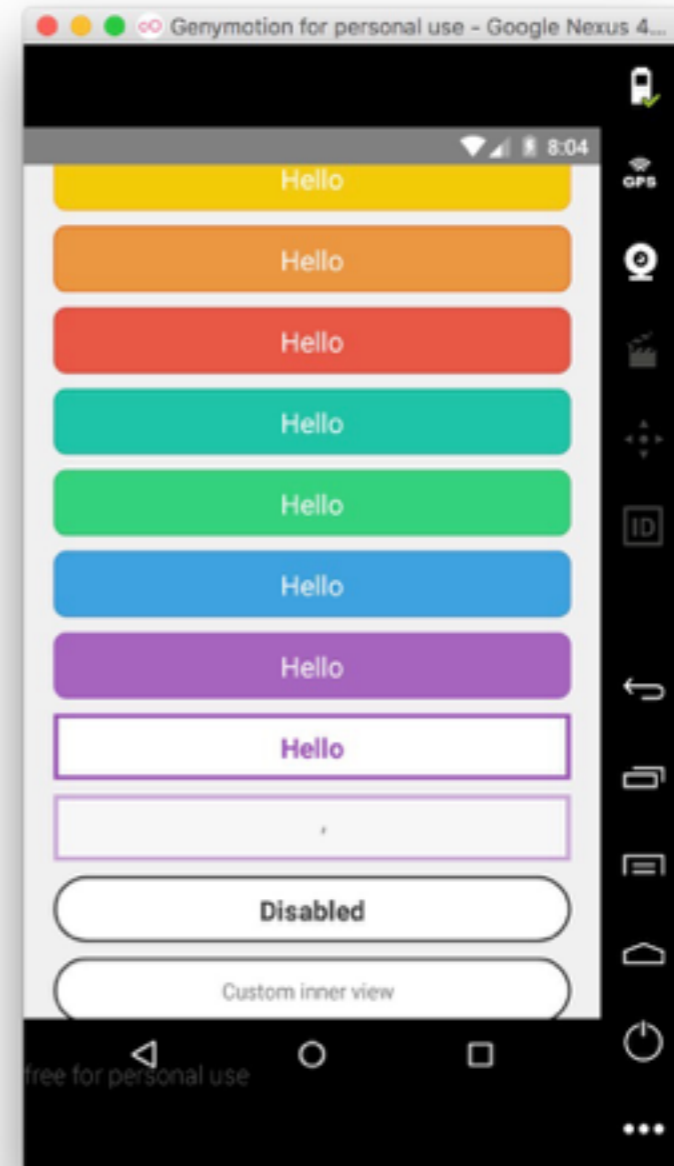
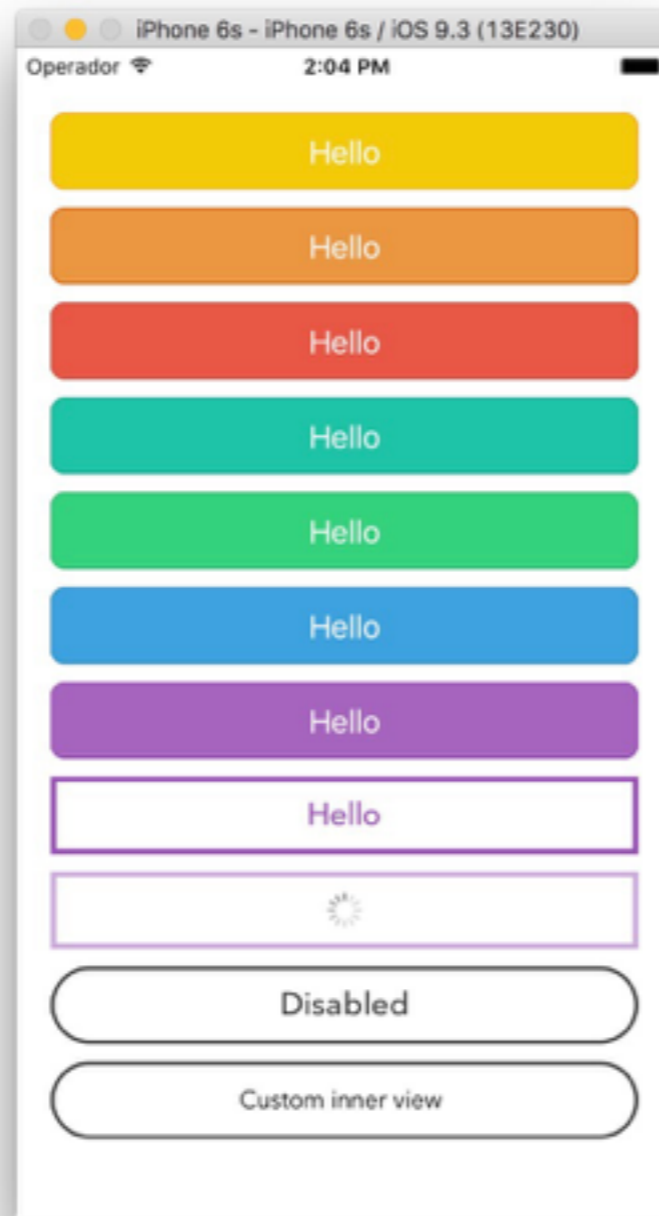
```
import Button from 'apsl-react-native-button'
```

```
renderLogin() {  
  return (  
    <View style={{ width: 300 }}>  
      <Button  
        style={styles.buttonStyle}  
        textStyle={styles.buttonTextStyle}  
        onPress={() => {  
          this.setState({  
            showLoginForm: true  
          })  
        }}  
      >  
        ต่อไป  
      </Button>  
    </View>  
  );  
}
```

iOS

apsl-react-native-button

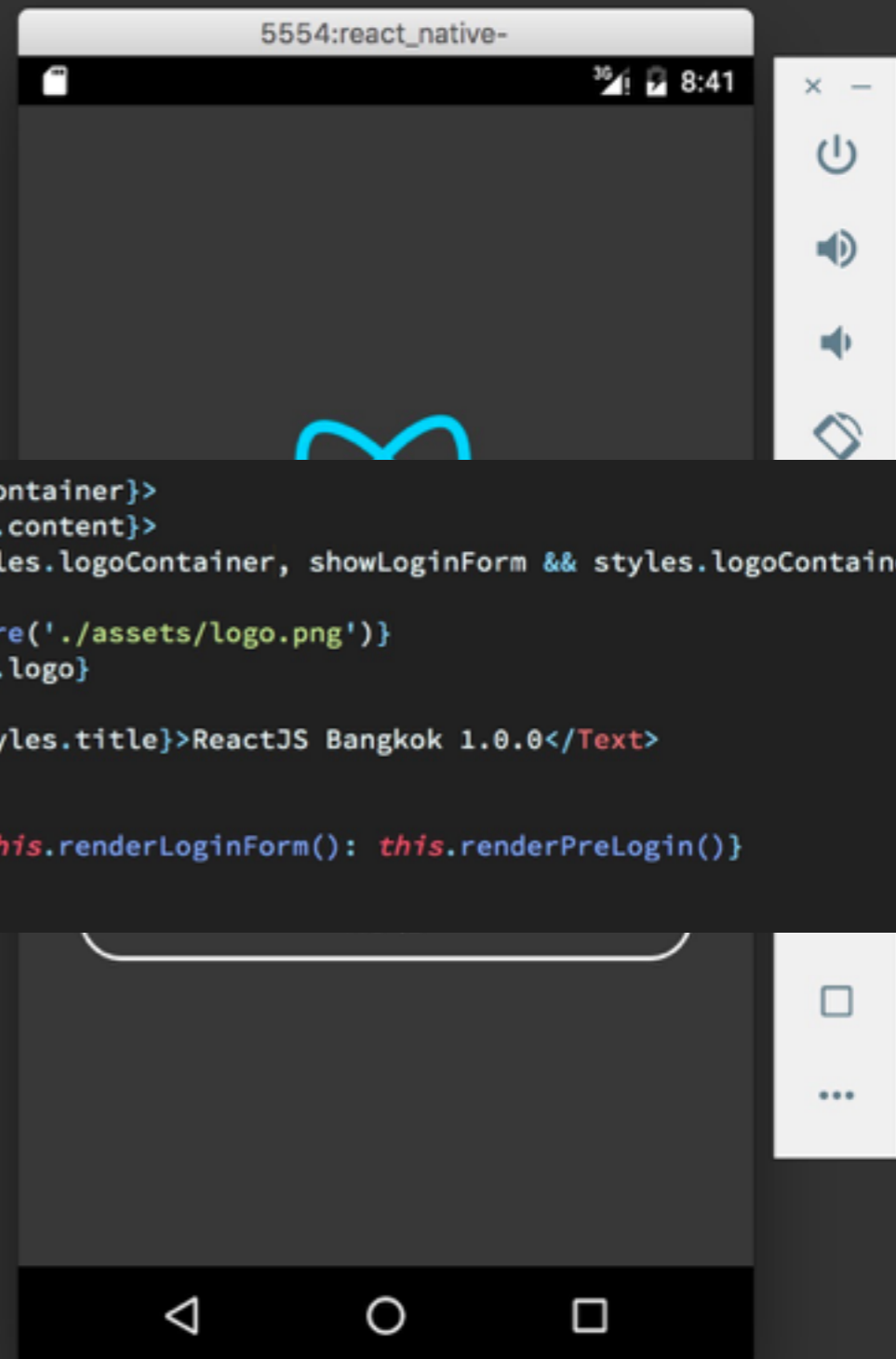
A React Native button component customizable via `style` props.




```
iPhone 6 - iPhone 6 / iOS 9.3 (13E230)
Carrier 8:45 PM

const styles = StyleSheet.create({
  container: {
    flex: 1,
    backgroundColor: '#383838',
    justifyContent: 'center',
  },
  content: {
    alignSelf: 'center'
  },
  logoContainer: {
    transform: [{ scale: 1 }],
    alignSelf: 'center'
  },
  logo: {
    width: 150,
    height: 150,
    alignSelf: 'center'
  },
  title: {
    fontSize: 24,
    color: 'white',
    marginTop: 16,
    marginBottom: 16,
    textAlign: 'center'
  },
},
```

iOS



Android

```
<View style={styles.container}>
  <View style={styles.content}>
    <View style={[styles.logoContainer, showLoginForm && styles.logoContainerSmall]}>
      <Image
        source={require('./assets/logo.png')}
        style={styles.logo}
      />
      <Text style={styles.title}>ReactJS Bangkok 1.0.0</Text>
    </View>
    <View>
      {showLoginForm? this.renderLoginForm(): this.renderPreLogin()}
    </View>
  </View>
</View>
```

```
iPhone 6 - iPhone 6 / iOS 9.3 (13E230)
Carrier 8:45 PM

const styles = StyleSheet.create({
  container: {
    flex: 1,
    backgroundColor: '#383838',
    justifyContent: 'center',
  },
  content: {
    alignSelf: 'center'
  },
  logoContainer: {
    transform: [{ scale: 1 }],
    alignSelf: 'center'
  },
  logo: {
    width: 150,
    height: 150,
    alignSelf: 'center'
  },
  title: {
    fontSize: 24,
    color: 'white',
    marginTop: 16,
    marginBottom: 16,
    textAlign: 'center'
  },
},
```

iOS

```
5554:react_native-
8:41

<View style={styles.container}>
  <View style={styles.content}>
    <View style={[styles.logoContainer, showLoginForm && styles.logoContainerSmall]}>
      <Image
        source={require('./assets/logo.png')}
        style={styles.logo}
      />
      <Text style={styles.title}>ReactJS Bangkok 1.0.0</Text>
    </View>
    <View>
      {showLoginForm ? this.renderLoginForm() : this.renderPreLogin()}
    </View>
  </View>
</View>
```

Android



iOS

```
5554:react_native-
constructor() {
  super();
  this.state = {
    showLoginForm: false,
    showLoading: false,
  };
}
```

```
render() {
  const { showLoginForm, showLoading } = this.state;
  // const showLoginForm = this.state.showLoginForm;
  // const showLoading = this.state.showLoading;
  return (
```



Android

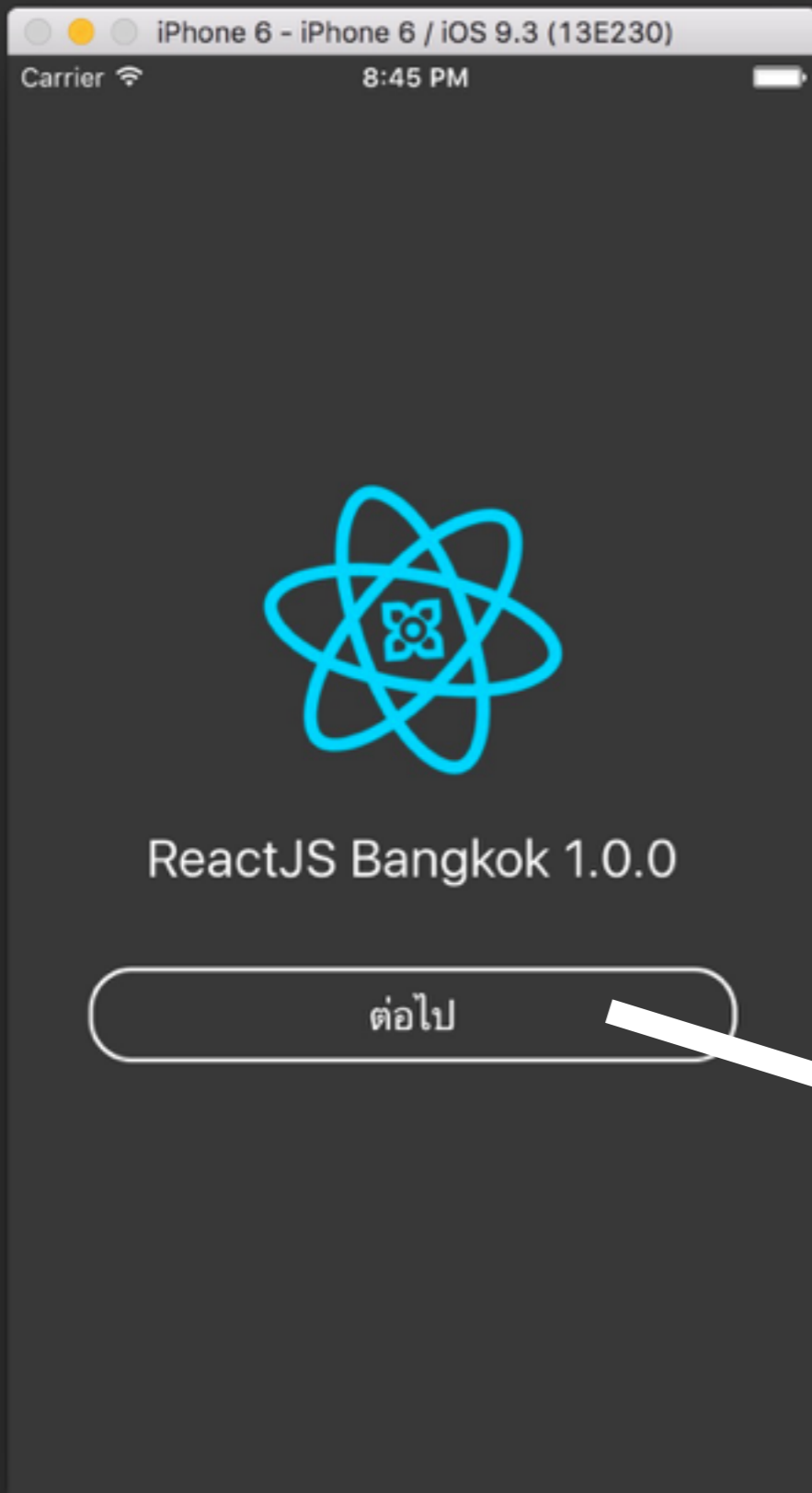


```
const { showLoginForm, showLoading } = this.state;  
// const showLoginForm = this.state.showLoginForm;  
// const showLoading = this.state.showLoading;
```

```
<View style={styles.container}>  
  <View style={styles.content}>  
    <View style={[styles.logoContainer, showLoginForm && styles.logoContainerSmall]}>  
      <Image  
        source={require('./assets/logo.png')}  
        style={styles.logo}  
      />  
      <Text style={styles.title}>ReactJS Bangkok 1.0.0</Text>  
    </View>  
    <View>  
      {showLoginForm? this.renderLoginForm(): this.renderPreLogin()}  
    </View>  
  </View>  
</View>
```

```
renderPreLogin() {  
  return (  
    <View style={{ width: 300 }}>  
      <Button  
        style={styles.buttonStyle}  
        textStyle={styles.buttonTextStyle}  
        onPress={() => {  
          this.setState({  
            showLoginForm: true  
          })  
        }}  
      >  
        ต่อไป  
      </Button>  
    </View>  
  );  
}
```

iOS

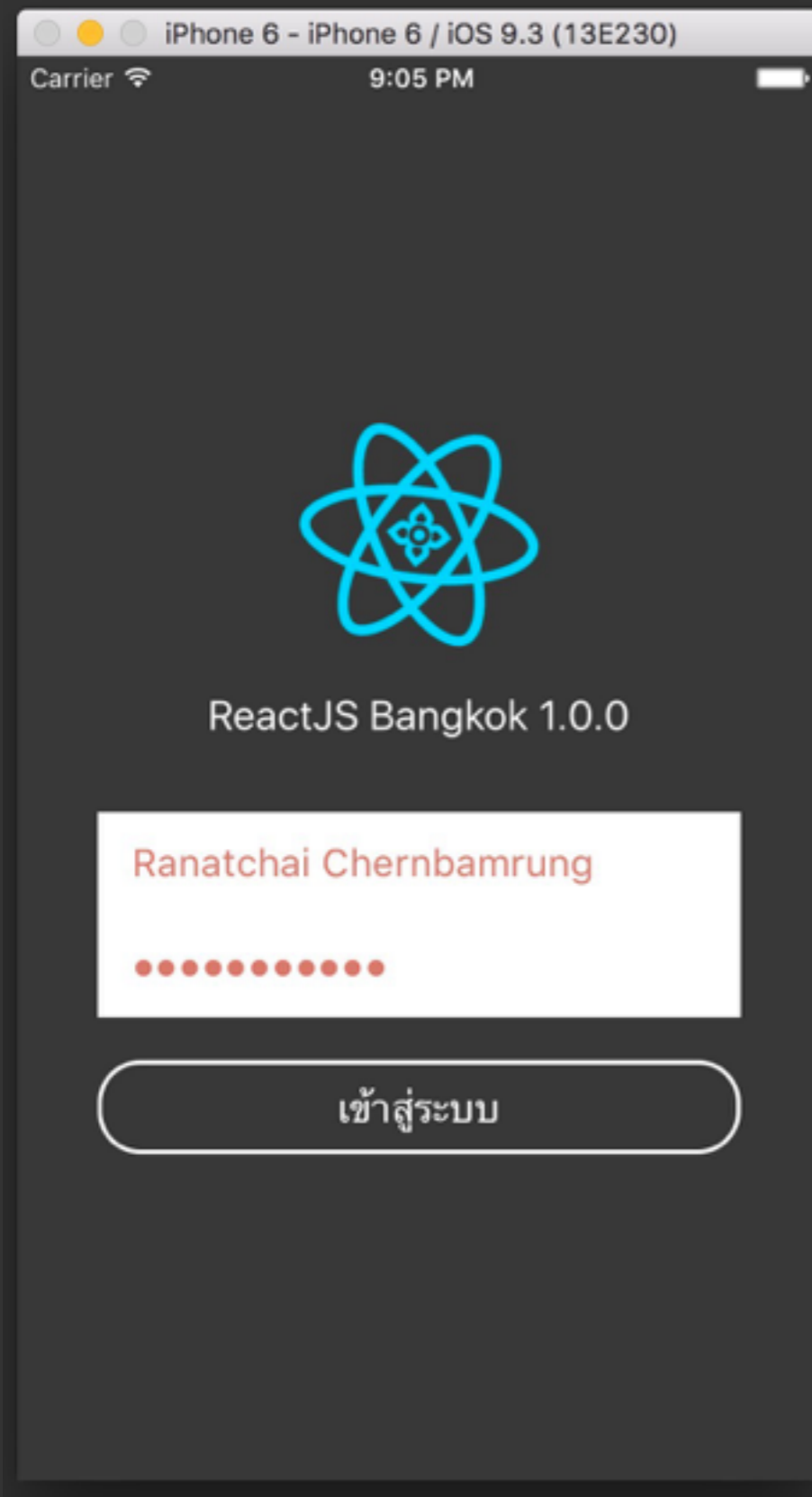


```
const { showLoginForm, showLoading } = this.state;  
// const showLoginForm = this.state.showLoginForm;  
// const showLoading = this.state.showLoading;
```

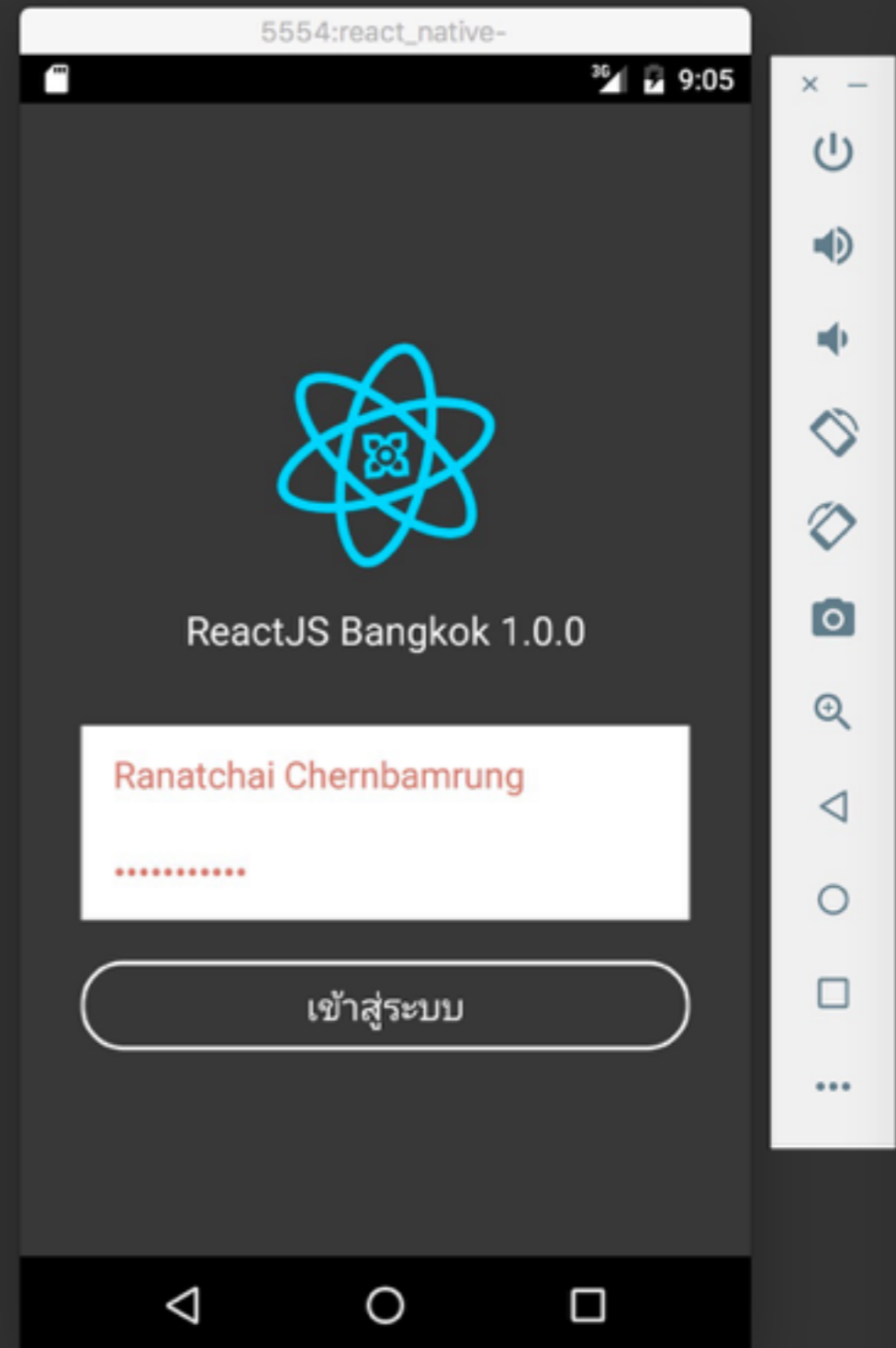
```
<View style={styles.container}>  
  <View style={styles.content}>  
    <View style={[styles.logoContainer, showLoginForm && styles.logoContainerSmall]}>  
      <Image  
        source={require('./assets/logo.png')}  
        style={styles.logo}  
      />  
      <Text style={styles.title}>ReactJS Bangkok 1.0.0</Text>  
    </View>  
    <View>  
      {showLoginForm? this.renderLoginForm(): this.renderPreLogin()}  
    </View>  
  </View>  
</View>
```

```
renderPreLogin() {  
  return (  
    <View style={{ width: 300 }}>  
      <Button  
        style={styles.buttonStyle}  
        textStyle={styles.buttonTextStyle}  
        onPress={() => {  
          this.setState({  
            showLoginForm: true  
          })  
        }}  
      >  
        ต่อไป  
      </Button>  
    </View>  
  );  
}
```

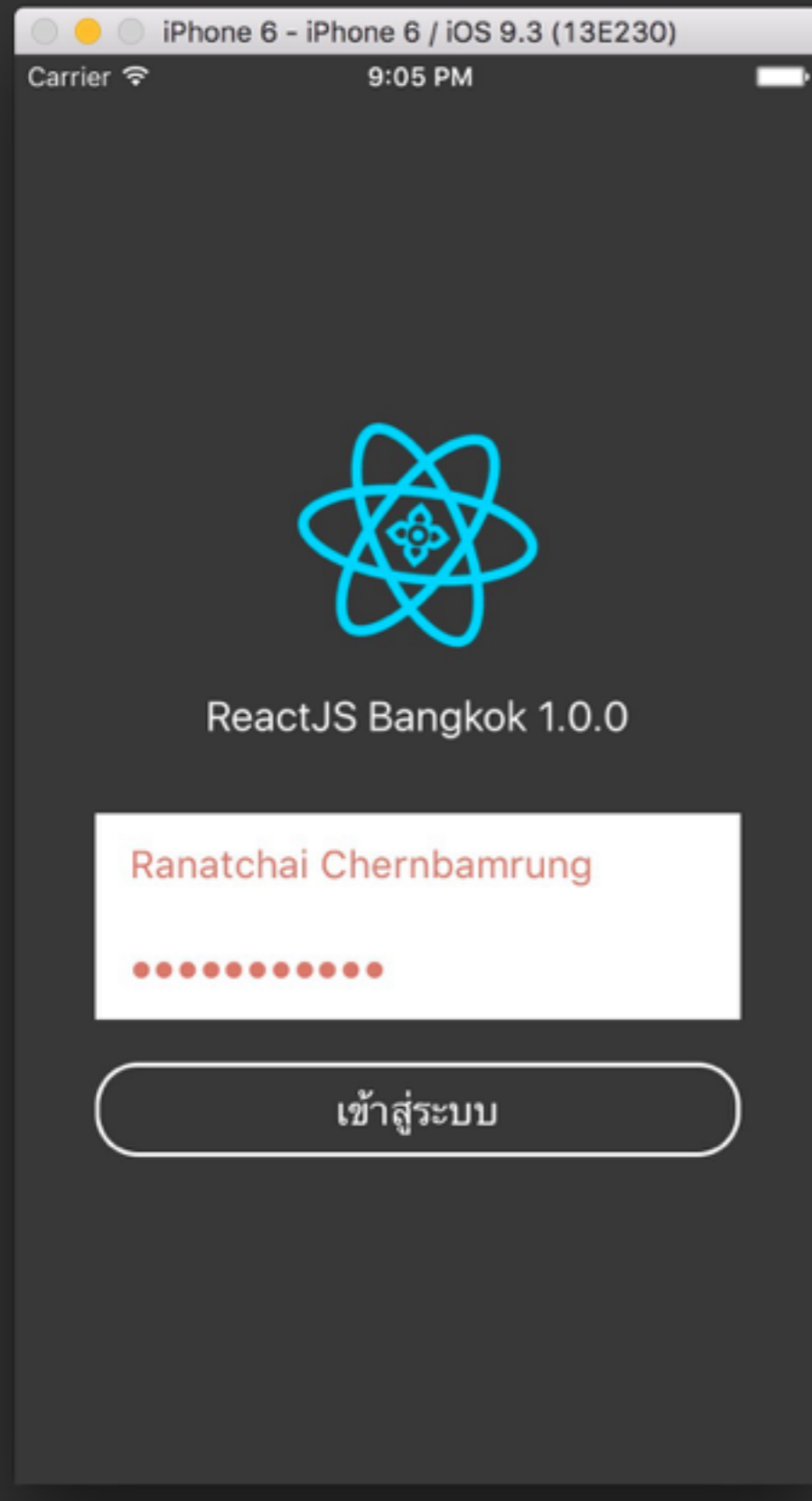
iOS



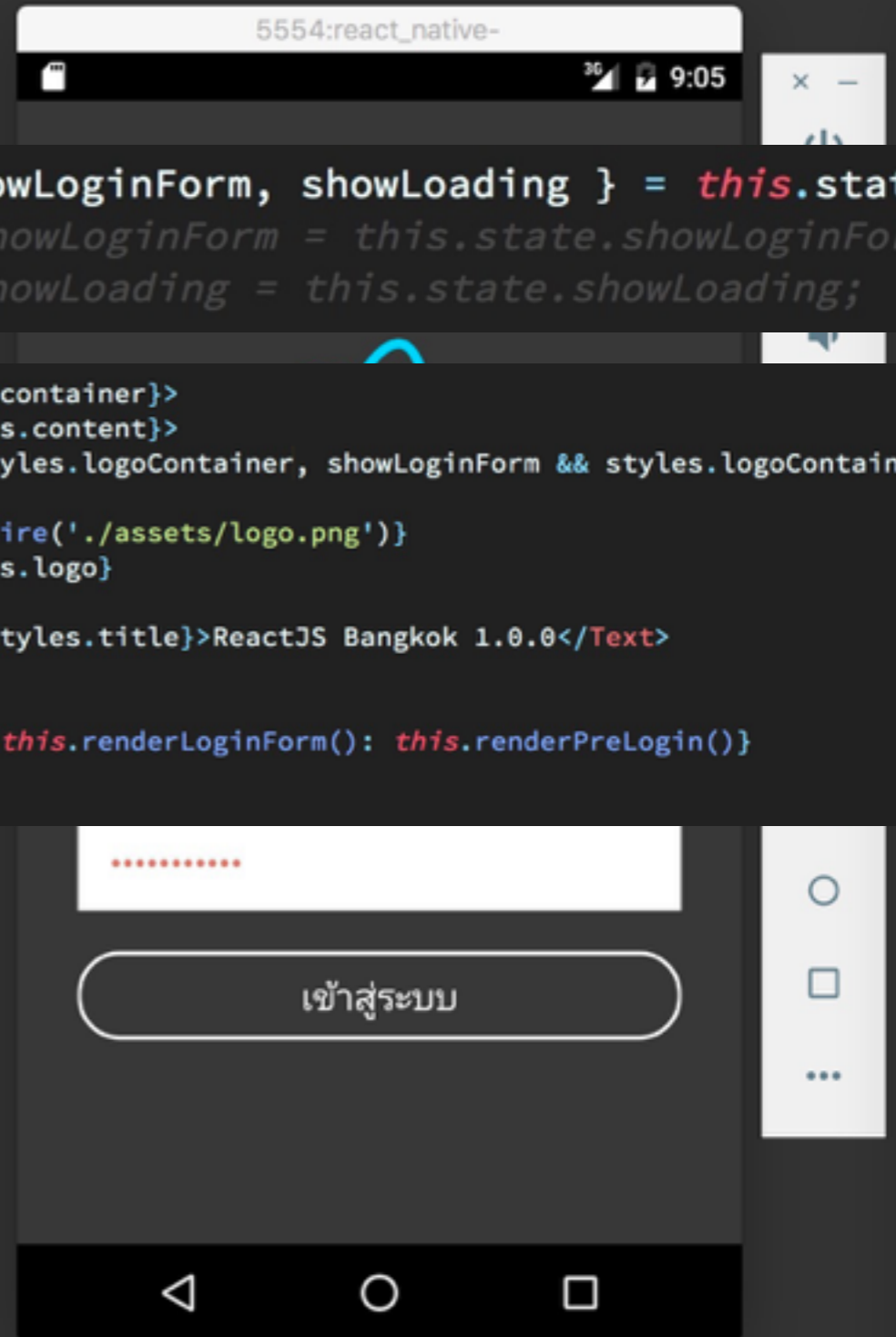
iOS



Android



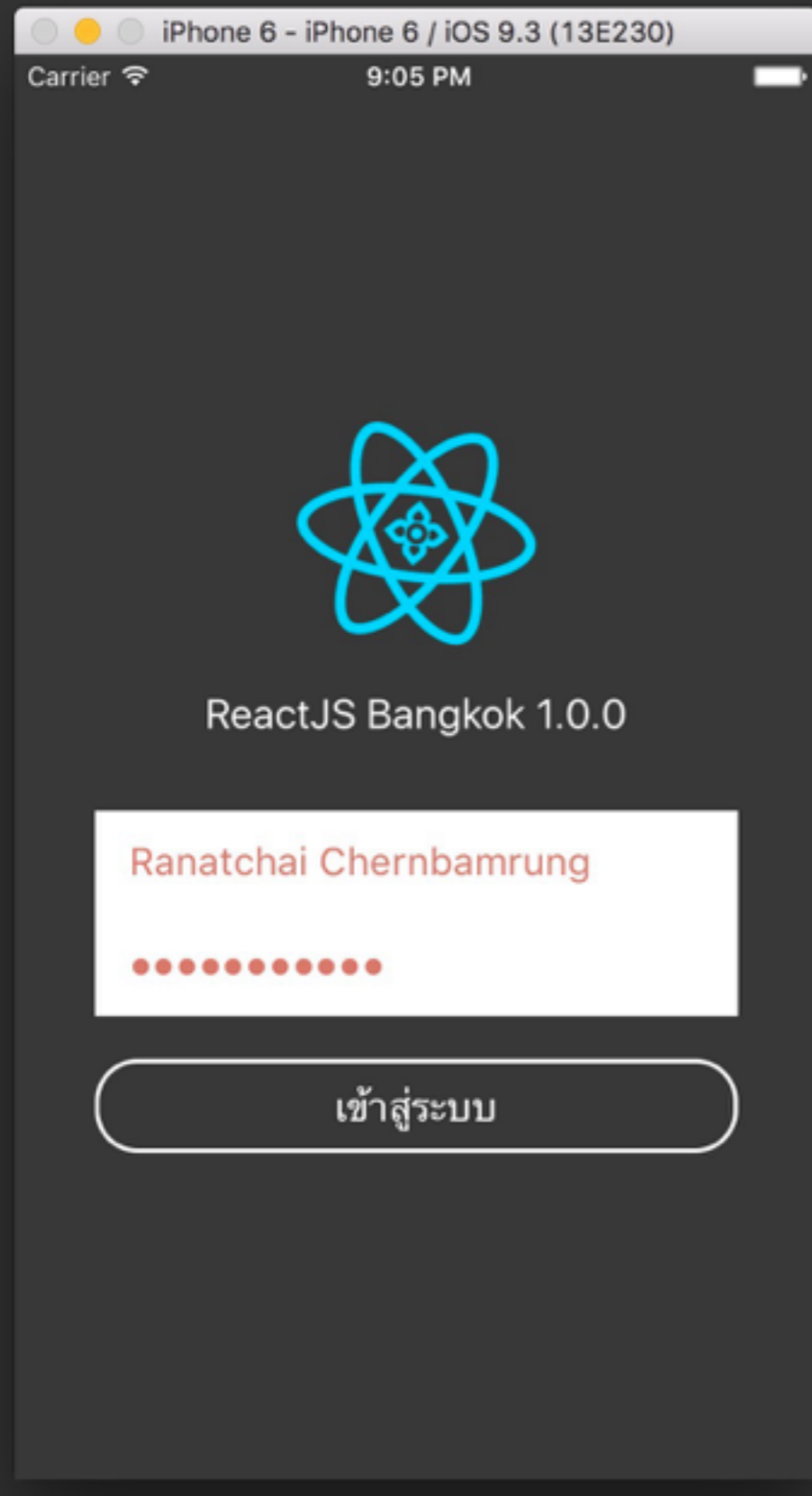
iOS



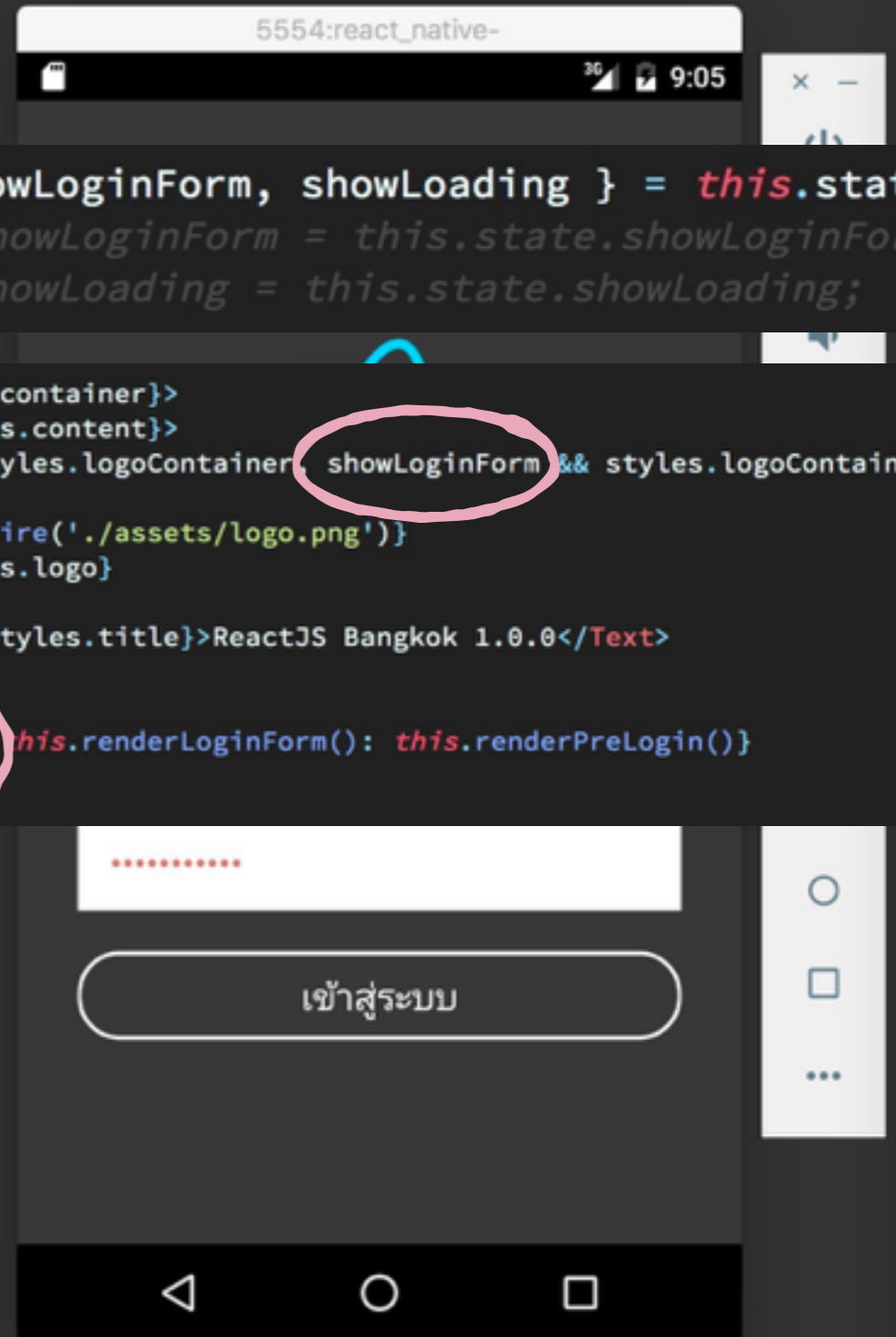
Android

```
const { showLoginForm, showLoading } = this.state;  
// const showLoginForm = this.state.showLoginForm;  
// const showLoading = this.state.showLoading;
```

```
<View style={styles.container}>  
  <View style={styles.content}>  
    <View style={[styles.logoContainer, showLoginForm && styles.logoContainerSmall]}>  
      <Image  
        source={require('./assets/logo.png')}  
        style={styles.logo}  
      />  
      <Text style={styles.title}>ReactJS Bangkok 1.0.0</Text>  
    </View>  
    <View>  
      {showLoginForm? this.renderLoginForm(): this.renderPreLogin()}  
    </View>  
  </View>  
</View>
```



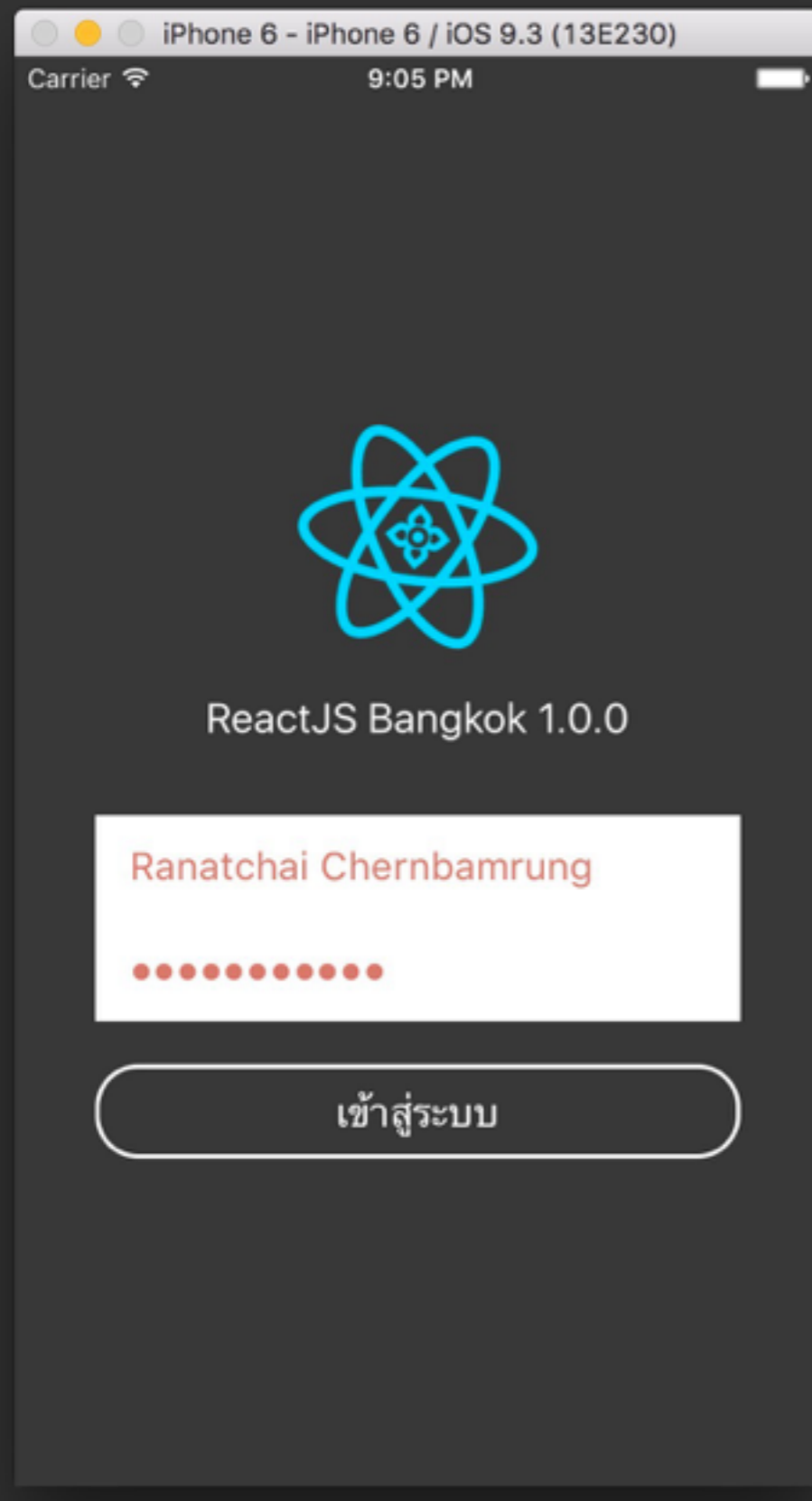
iOS



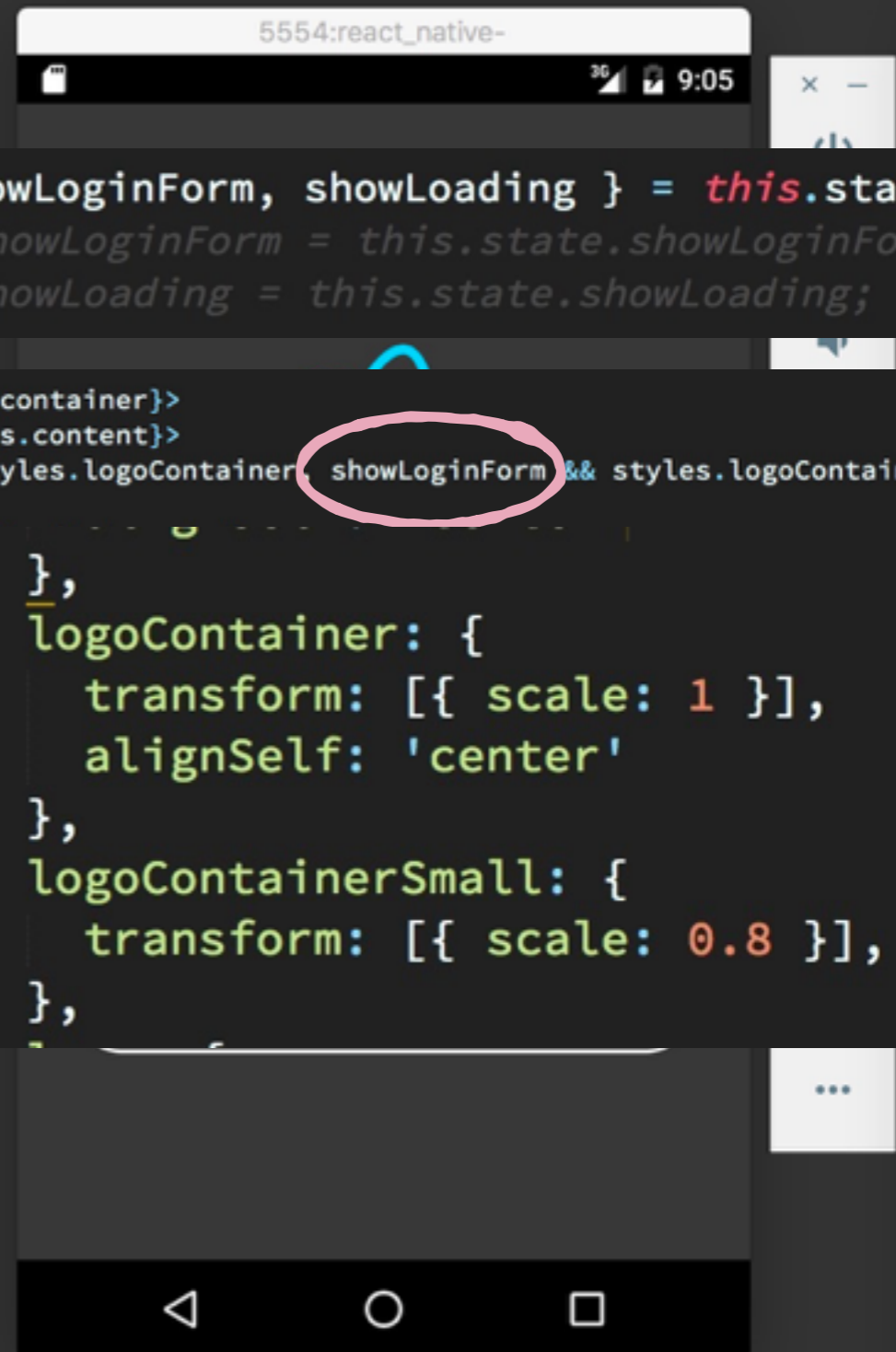
Android

```
const { showLoginForm, showLoading } = this.state;  
// const showLoginForm = this.state.showLoginForm;  
// const showLoading = this.state.showLoading;
```

```
<View style={styles.container}>  
  <View style={styles.content}>  
    <View style={[styles.logoContainer, showLoginForm && styles.logoContainerSmall]}>  
      <Image  
        source={require('./assets/logo.png')}  
        style={styles.logo}  
      />  
      <Text style={styles.title}>ReactJS Bangkok 1.0.0</Text>  
    </View>  
    <View>  
      {showLoginForm ? this.renderLoginForm() : this.renderPreLogin()}  
    </View>  
  </View>  
</View>
```

iOS

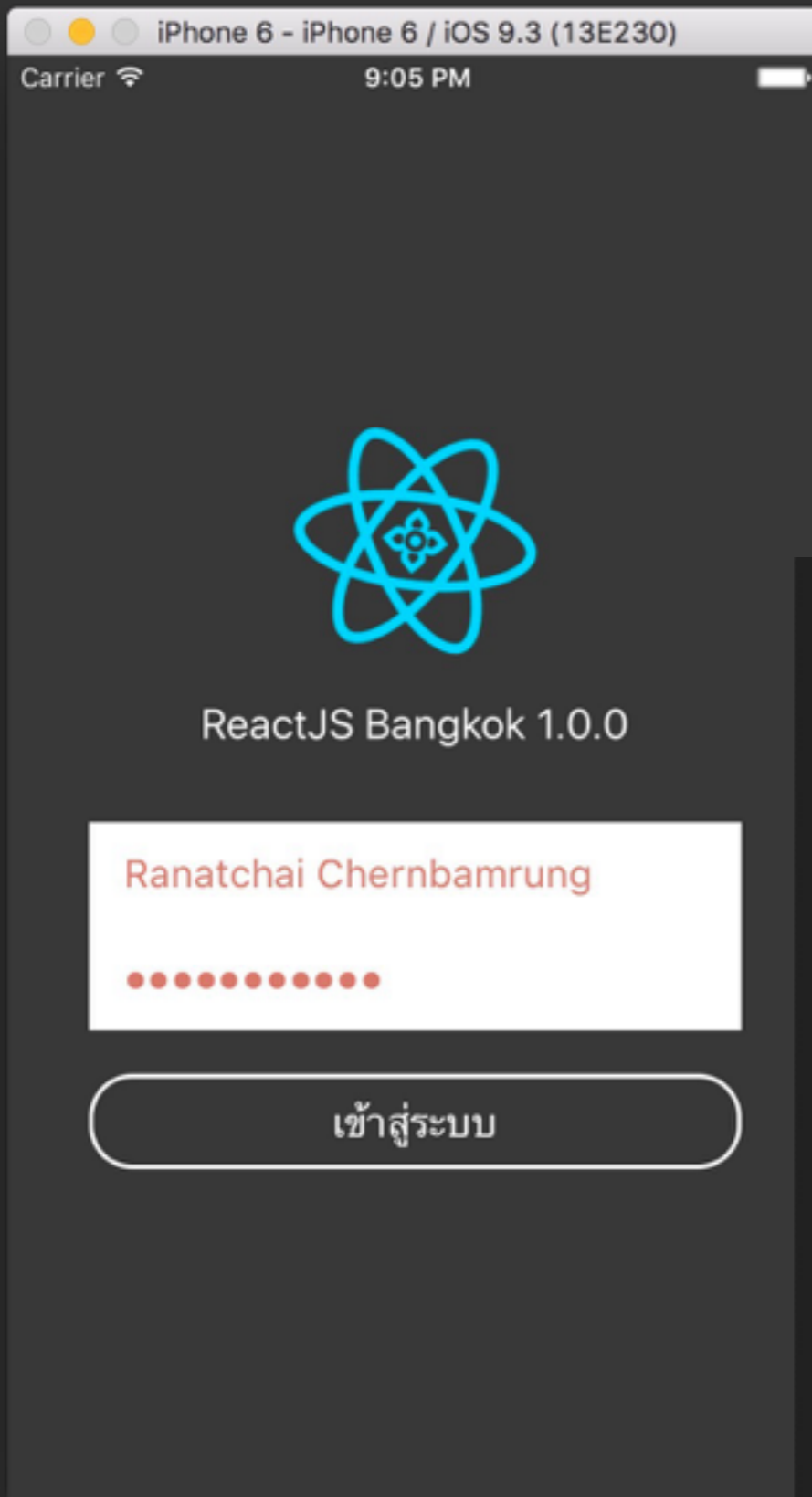


Android

```
const { showLoginForm, showLoading } = this.state;  
// const showLoginForm = this.state.showLoginForm;  
// const showLoading = this.state.showLoading;
```

```
<View style={styles.container}>  
  <View style={styles.content}>  
    <View style={[styles.logoContainer, showLoginForm && styles.logoContainerSmall]}>  
      <Image  
        source={requ  
        style={style  
      } />  
      <Text style={s  
    </View>  
    <View>  
      {showLoginForm?  
    </View>  
  </View>
```

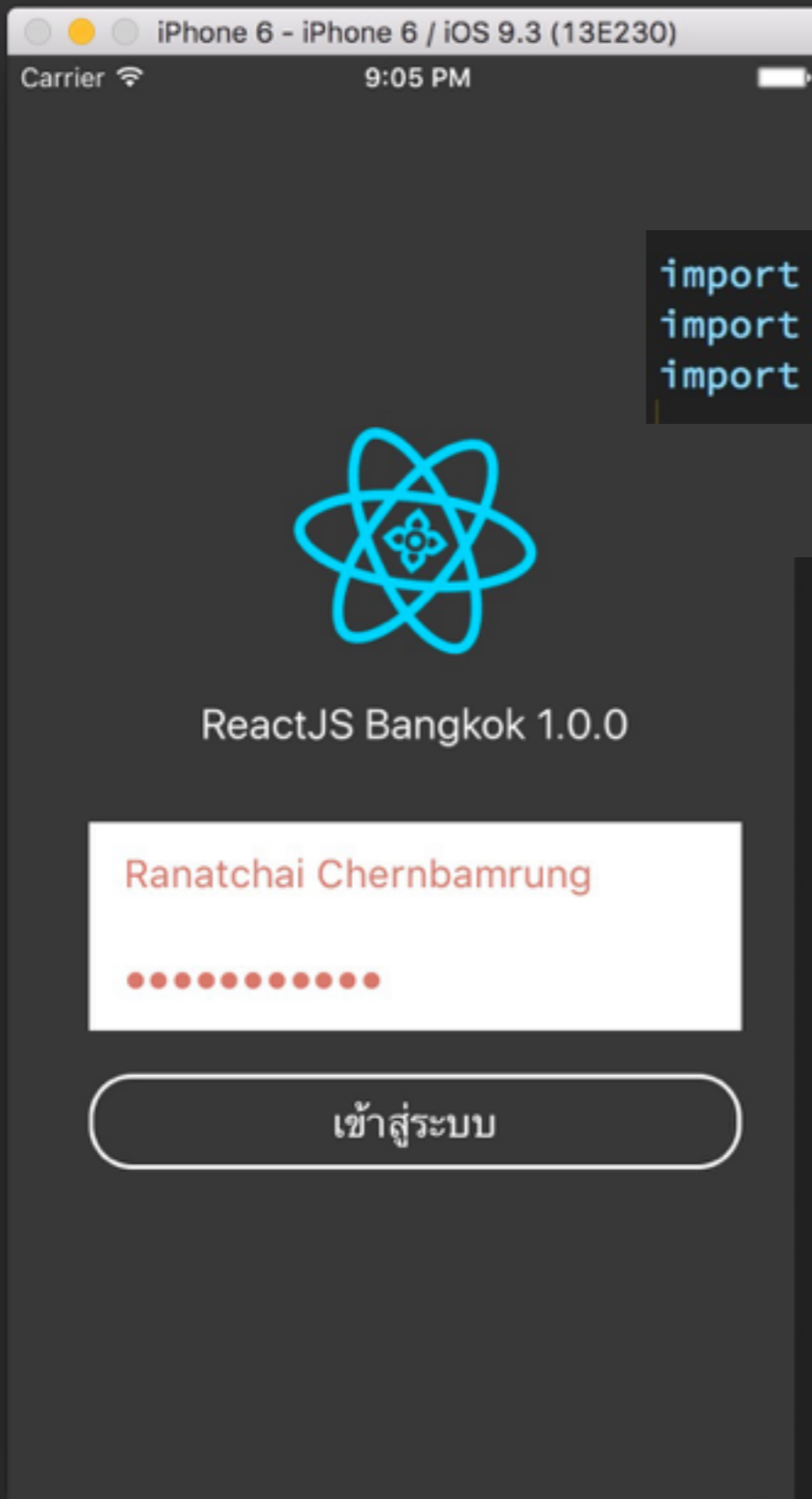
```
logoContainer: {  
  transform: [{ scale: 1 }],  
  alignSelf: 'center'  
},  
logoContainerSmall: {  
  transform: [{ scale: 0.8 }],  
},
```



iOS

```
5554:react_native-  
<View style={styles.container}>  
  <View style={styles.content}>  
    <View style={[styles.logoContainer, showLoginForm && styles.logoContainerSmall]}>  
      <Image  
        source={require('./assets/logo.png')}  
        style={styles.logo}  
      />  
      <Text style={styles.title}>ReactJS Bangkok 1.0.0</Text>  
    </View>  
    <View>  
      {showLoginForm ? this.renderLoginForm() : this.renderPreLogin()}  
    </View>  
  </View>  
</View>
```

```
renderLoginForm() {  
  return (  
    <Animatable.View animation="slideInUp" duration={300} easing="linear" style={{ width: 300 }}>  
      <TextInput  
        label={'Comment'}  
        iconClass={FontAwesomeIcon}  
        iconName={'comment'}  
        iconColor={'white'}  
        inputStyle={{ color: '#db786d' }}  
        value="Ranatchai Chernbamrung"  
        placeholder={'Full Name'}  
      />  
      <TextInput  
        label={'Comment'}  
        iconClass={FontAwesomeIcon}  
        iconName={'comment'}  
        iconColor={'white'}  
        inputStyle={{ color: '#db786d' }}  
        value="MY=Password"  
        placeholder={'Password'}  
        secureTextEntry  
      />  
      <Button  
        style={styles.buttonStyle}  
        textStyle={styles.buttonTextStyle}  
        key="login"  
        onPress={this.handleLogin}  
      />  
      เข้าสู่ระบบ  
    </Button>  
  </Animatable.View>  
)  
};
```



```
import { Makiko as TextInput } from 'react-native-textinput-effects';  
import FontAwesomeIcon from 'react-native-vector-icons/FontAwesome';  
import * as Animatable from 'react-native-animatable';
```



ReactJS Bangkok 1.0.0

Ranatchai Chernbamrung

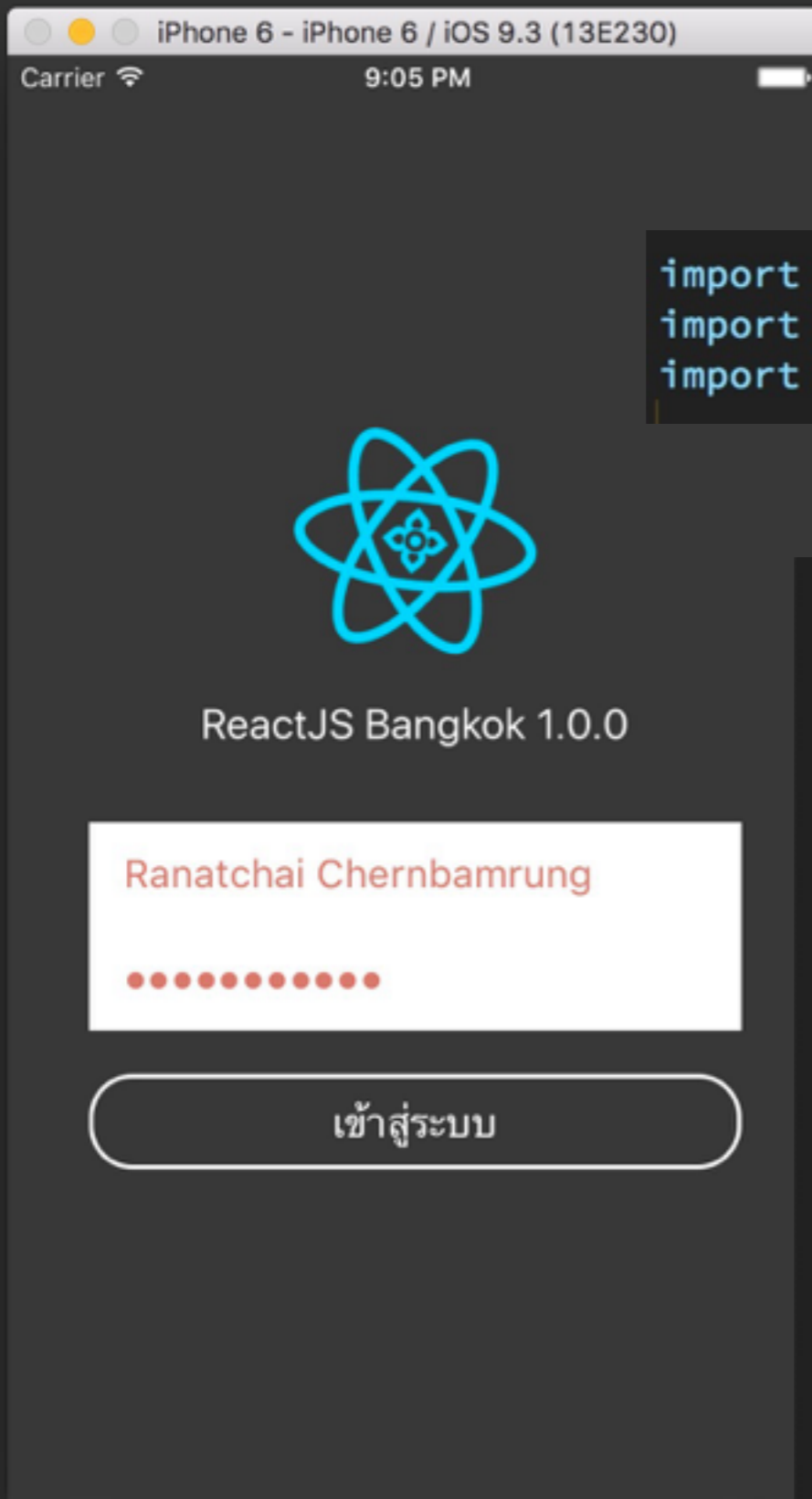


เข้าสู่ระบบ



```
renderLoginForm() {  
  return (  
    <Animatable.View animation="slideInUp" duration={300} easing="linear" style={{ width: 300 }}>  
      <TextInput  
        label={'Comment'}  
        iconClass={FontAwesomeIcon}  
        iconName={'comment'}  
        iconColor={'white'}  
        inputStyle={{ color: '#db786d' }}  
        value="Ranatchai Chernbamrung"  
        placeholder={'Full Name'}  
      />  
      <TextInput  
        label={'Comment'}  
        iconClass={FontAwesomeIcon}  
        iconName={'comment'}  
        iconColor={'white'}  
        inputStyle={{ color: '#db786d' }}  
        value="MY=Password"  
        placeholder={'Password'}  
        secureTextEntry  
      />  
      <Button  
        style={styles.buttonStyle}  
        textStyle={styles.buttonTextStyle}  
        key="login"  
        onPress={this.handleLogin}  
      />  
      เข้าสู่ระบบ  
    </Button>  
  </Animatable.View>  
);  
}
```

iOS



```
import { Makiko as TextInput } from 'react-native-textinput-effects';  
import FontAwesomeIcon from 'react-native-vector-icons/FontAwesome';  
import * as Animatable from 'react-native-animatable';
```



ReactJS Bangkok 1.0.0

Ranatchai Chernbamrung

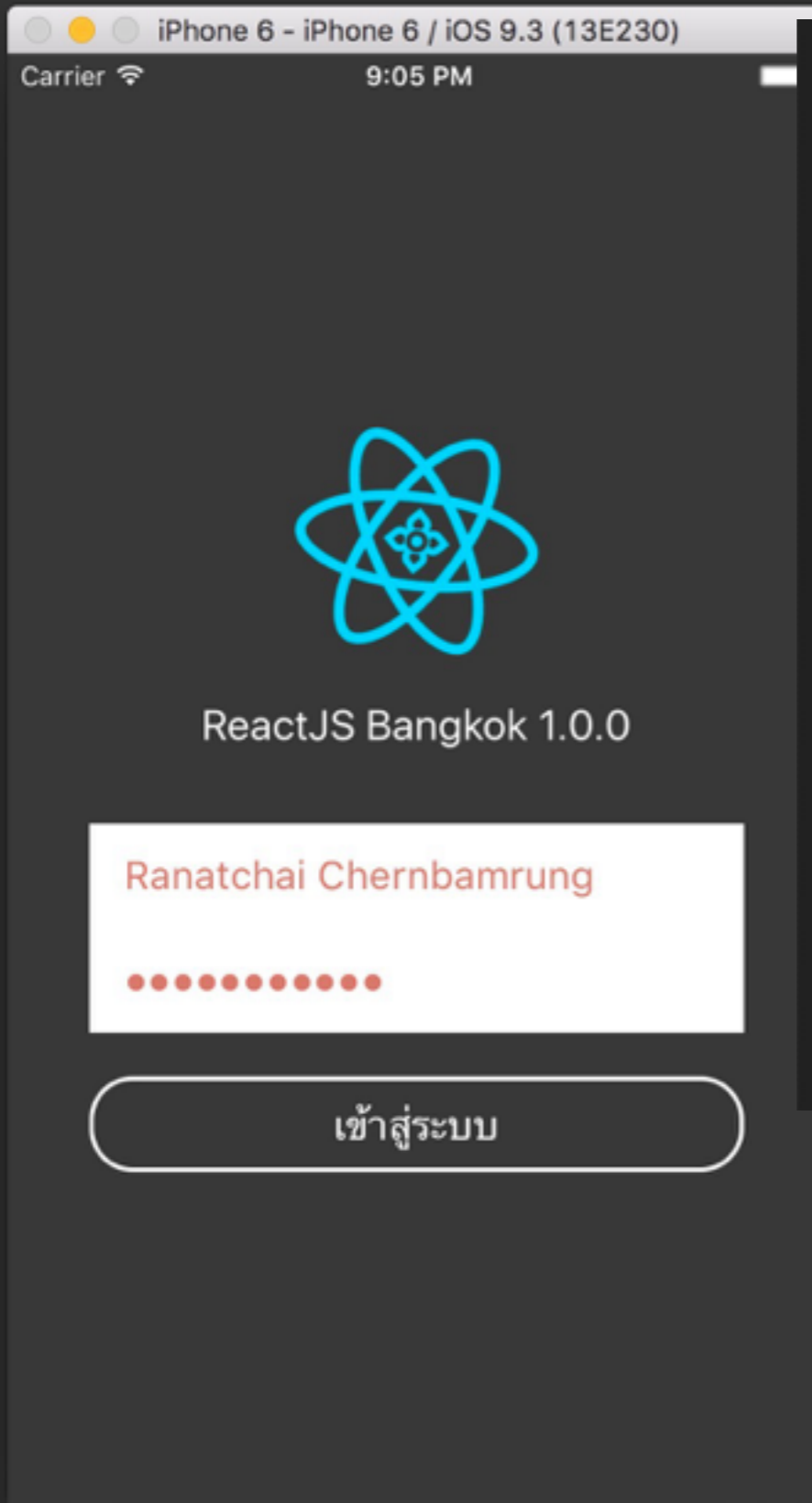


เข้าสู่ระบบ



```
renderLoginForm() {  
  return (  
    <Animatable.View animation="slideInUp" duration={300} easing="linear" style={{ width: 300 }}>  
      <TextInput  
        label={'Comment'}  
        iconClass={FontAwesomeIcon}  
        iconName={'comment'}  
        iconColor={'white'}  
        inputStyle={{ color: '#db786d' }}  
        value="Ranatchai Chernbamrung"  
        placeholder={'Full Name'}  
      />  
      <TextInput  
        label={'Comment'}  
        iconClass={FontAwesomeIcon}  
        iconName={'comment'}  
        iconColor={'white'}  
        inputStyle={{ color: '#db786d' }}  
        value="MY=Password"  
        placeholder={'Password'}  
        secureTextEntry  
      />  
      <Button  
        style={styles.buttonStyle}  
        textStyle={styles.buttonTextStyle}  
        key="login"  
        onPress={this.handleLogin}  
      />  
      <Text>เข้าสู่ระบบ</Text>  
    </Animatable.View>  
  );  
}
```

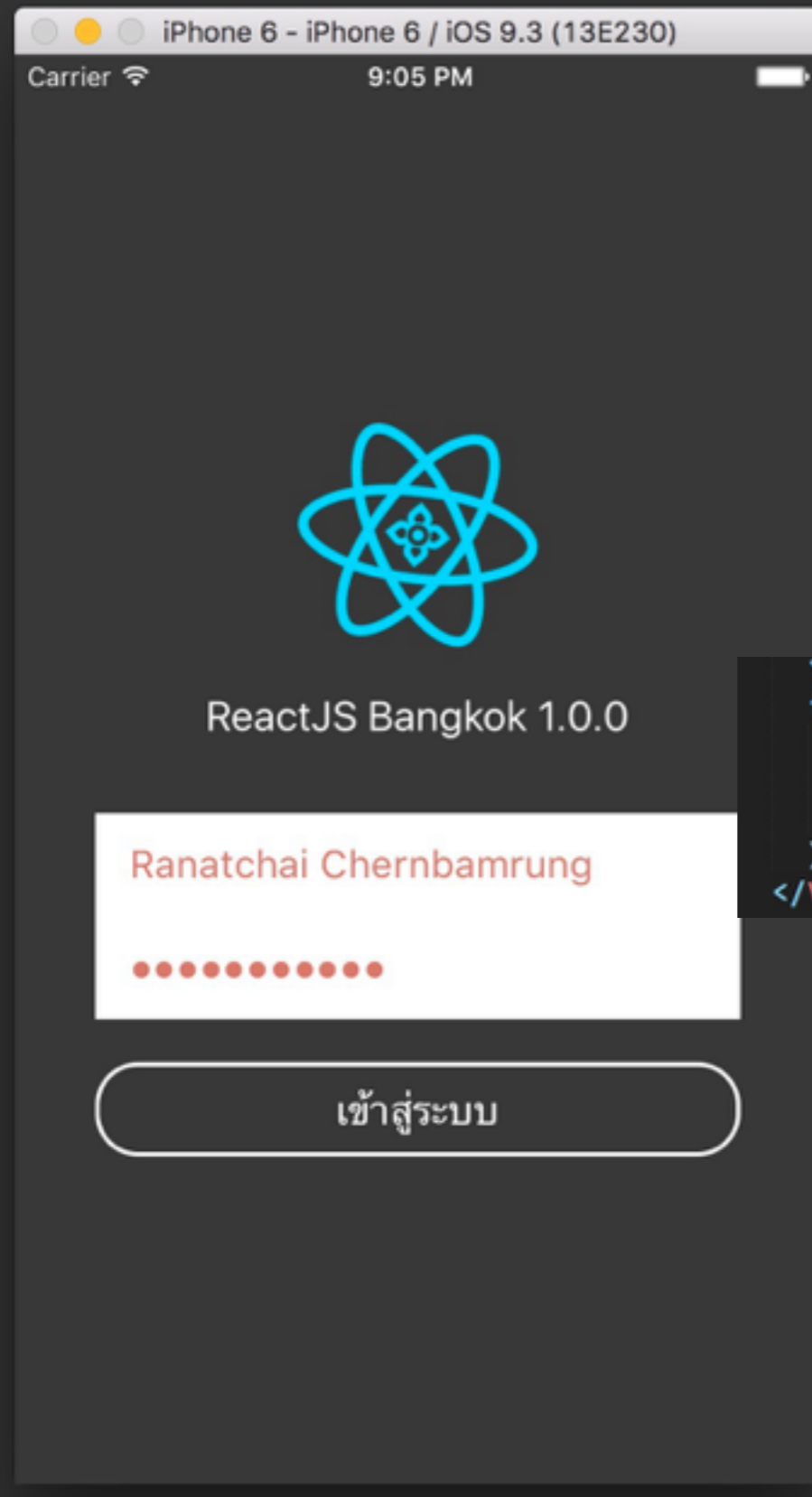
iOS



```
renderLoginForm() {  
  return (  
    <Animatable.View animation="slideInUp" duration={300} easing="linear" style={{ width: 300 }}>  
      <TextInput  
        label={'Comment'}  
        iconClass={FontAwesomeIcon}  
        iconName={'comment'}  
        iconColor={'white'}  
        inputStyle={{ color: '#db786d' }}  
        value="Ranatchai Chernbamrung"  
        placeholder={'Full Name'}  
      />  
      <TextInput  
        label={'Comment'}  
        iconClass={FontAwesomeIcon}  
        iconName={'comment'}  
        iconColor={'white'}  
        inputStyle={{ color: '#db786d' }}  
        value="MY=Password"  
        placeholder={'Password'}  
        secureTextEntry  
      />  
      <Button  
        style={styles.buttonStyle}  
        textStyle={styles.buttonTextStyle}  
        key="login"  
        onPress={this.handleLogin}  
      />  
      เข้าสู่ระบบ  
    </Animatable.View>  
  );  
}
```

```
handleLogin = () => {  
  this.setState({  
    showLoading: true  
  }, () => {  
    setTimeout(() => {  
      this.setState({  
        showLoading: false  
      }, () => {  
        this.props.navigator.push(TimelineRoute);  
      });  
    }, 500);  
  })  
};
```

iOS

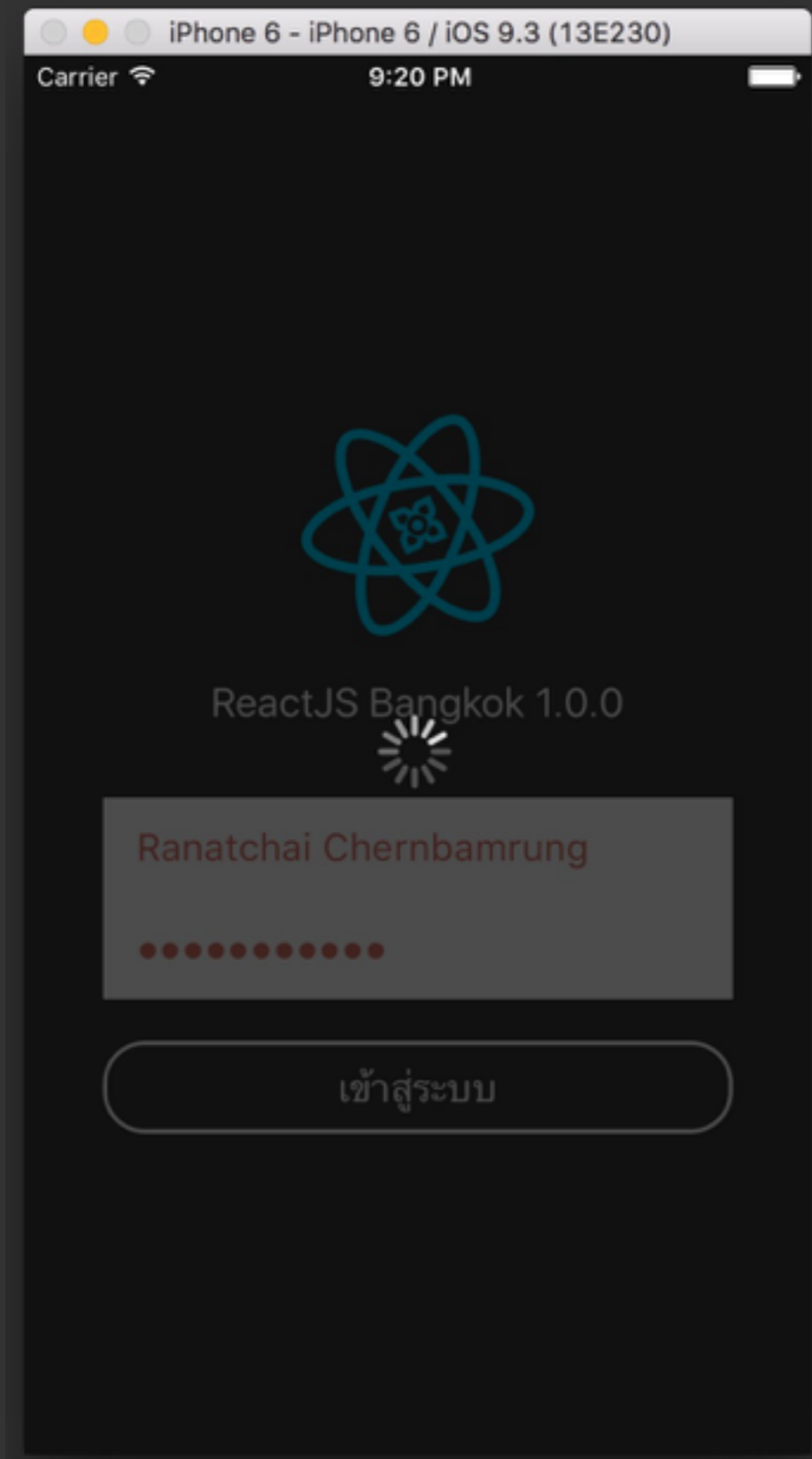


```
},  
loadingContainer: {  
  position: 'absolute',  
  left: 0,  
  right: 0,  
  top: 0,  
  bottom: 0,  
  backgroundColor: 'rgba(0,0,0,0.7)',  
  justifyContent: 'center'  
},  
loading: {  
  alignSelf: 'center',  
},
```

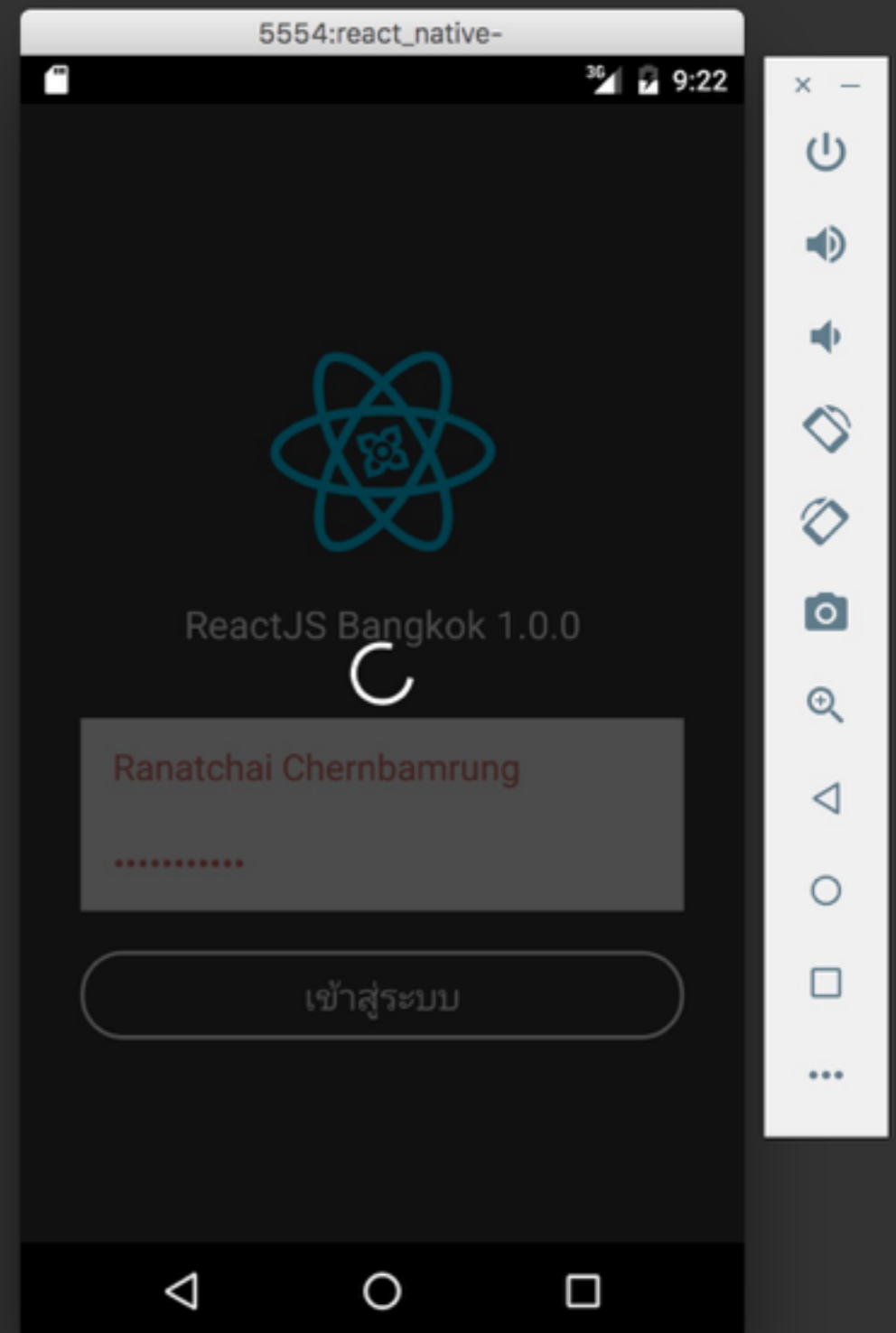
```
</View>  
{showLoading && (  
  <View style={styles.loadingContainer}>  
    <ActivityIndicator size="large" color="white" style={styles.loading}/>  
  </View>  
)}  
</View>
```

```
handleLogin = () => {  
  this.setState({  
    showLoading: true  
  }, () => {  
    setTimeout(() => {  
      this.setState({  
        showLoading: false  
      }, () => {  
        this.props.navigator.push(TimelineRoute);  
      });  
    }, 500);  
  })  
};
```

iOS



iOS



Android