Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего профессионального образования

«Владимирский государственный университет

имени Александра Григорьевича и Николая Григорьевича Столетовых»

(ВлГУ)

Кафедра информационных систем и программной инженерии

Лабораторная работа №10

по дисциплине "Технологии разработки мобильных приложений"

ТЕМА РАБОТЫ:

Знакомство с React Native

Выполнил:

студент гр. ПРИ-120

Парахин К.В.

Принял:

преподаватель кафедры ИСПИ

Рощина А.И.

Владимир 2023 г.

Цель работы:

Реализовать приложение с использованием фреймворка React Native

Выполнение работы

Для работы с фреймворком React Native в IDE Visual Studio Code требуется для начала установить Node.js

Затем выполнить команду *npm install -g expo-cli* для установки Expo CLI.

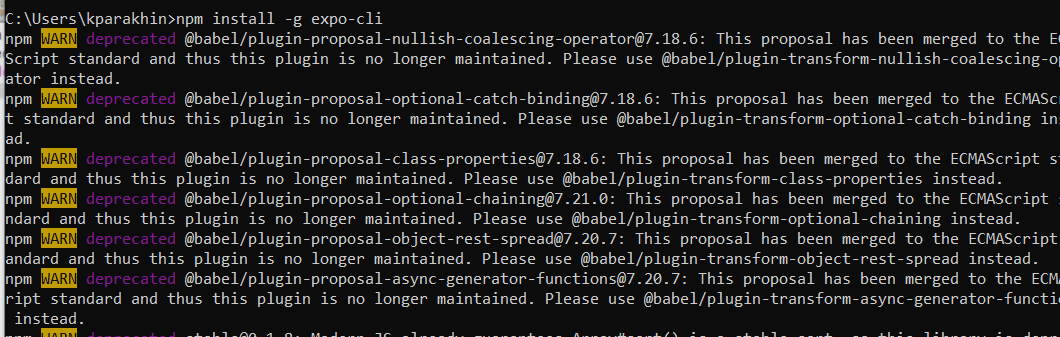


Рисунок 1. Установка expo-cli

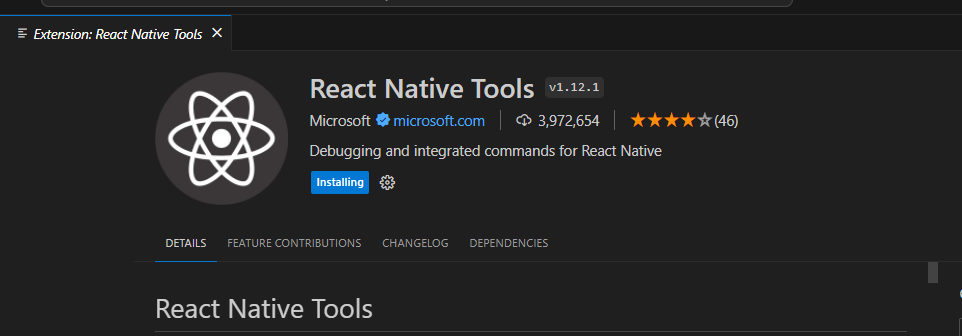


Рисунок 2. Установка плагина React Native Tools

Далее в терминале VS Code создаем новый проект:

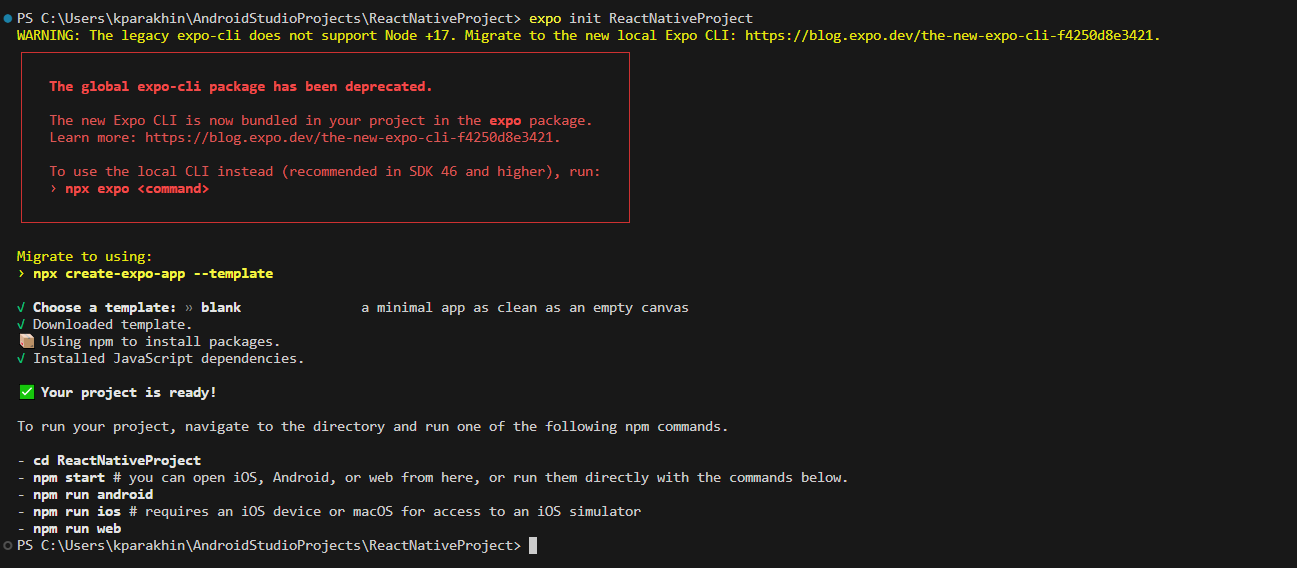


Рисунок 3. Создание нового проекта

Выполнение примеров из методички:

Пример 1.

import React from "react";

import { StyleSheet, Text, View } from "react-native";

export default class App extends React.Component {

 render() {

 return ( <Text>Hello, world!</Text> );

 }

}

const styles = StyleSheet.create({

 container: {

 flex: 1,

 backgroundColor: "#fff",

 alignItems: "center",

 justifyContent: "center"

 }

});

Чтобы запустить пример используем отладку в браузере на порту localhost:19006:



Пример 2. Пример с кнопкой

import React, { Component } from "react";

import { StyleSheet, Text, View, TouchableHighlight } from "react-native";

class App extends Component {

 constructor(props) {

 super(props);

 this.state = { pressing: false };

 }

 \_onPressIn = () => {

 this.setState({ pressing: true });

 };

 \_onPressOut = () => {

 this.setState({ pressing: false });

 };

 render() {

 return (

 <View style={styles.container}>

 <TouchableHighlight

 onPressIn={this.\_onPressIn}

 onPressOut={this.\_onPressOut}

 style={styles.touchable}

 >

 <View style={styles.button}>

 <Text style={styles.welcome}>

 {this.state.pressing ? "НАЖАТО!" : "НАЖМИ!"}

 </Text>

 </View>

 </TouchableHighlight>

 </View>

 );

 }

}

const styles = StyleSheet.create({

container: {

flex: 1,

justifyContent: "center",

alignItems: "center",

backgroundColor: "#F5FCFF"

 },

welcome: { fontSize: 20, textAlign: "center", margin: 10, color: "#FFFFFF" },

touchable: { borderRadius: 100 },

button: {

backgroundColor: "#FF00FF",

borderRadius: 100,

height: 200,

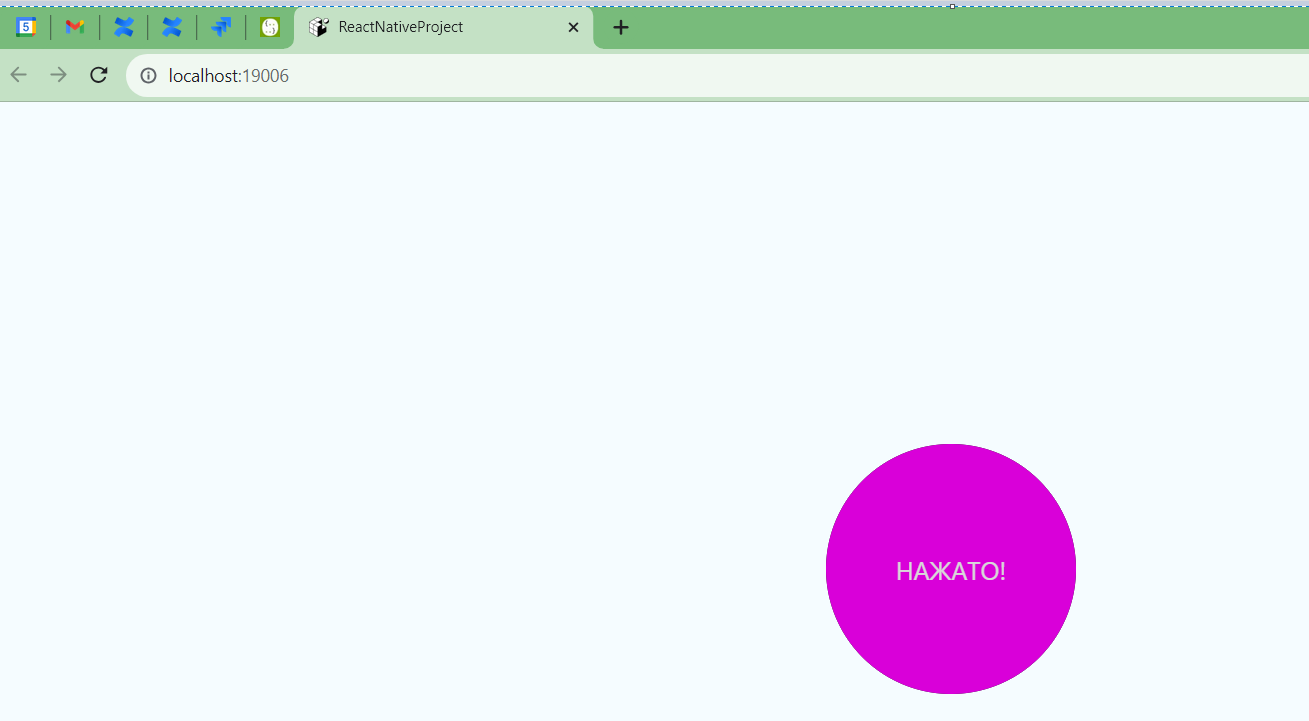
width: 200,

justifyContent: "center"

 }

});

export default App;



Пример 3. Пример со списком

import React, {Component} from 'react';

import {StyleSheet, Text, View, FlatList} from 'react-native';

class App extends Component {

 constructor(props) {

 super(props);

 this.state = {

 data: [

 {key: 'a'},

 {key: 'b'},

 {key: 'c'},

 {key: 'd'},

 {key: 'a longer example'},

 {key: 'e'},

 {key: 'f'},

 {key: 'g'},

 {key: 'h'},

 {key: 'i'},

 {key: 'j'},

 {key: 'k'},

 {key: 'l'},

 {key: 'm'},

 {key: 'n'},

 {key: 'o'},

 {key: 'p'},

 ],

 };

 }

 \_renderItem = data => {

 return <Text style={styles.row}>{data.item.key}</Text>;

 };

 render() {

 return (

 <View style={styles.container}>

 <FlatList data={this.state.data} renderItem={this.\_renderItem} />

 </View>

 );

 }

}

const styles = StyleSheet.create({

 container: {

 flex: 1,

 justifyContent: 'center',

 alignItems: 'center',

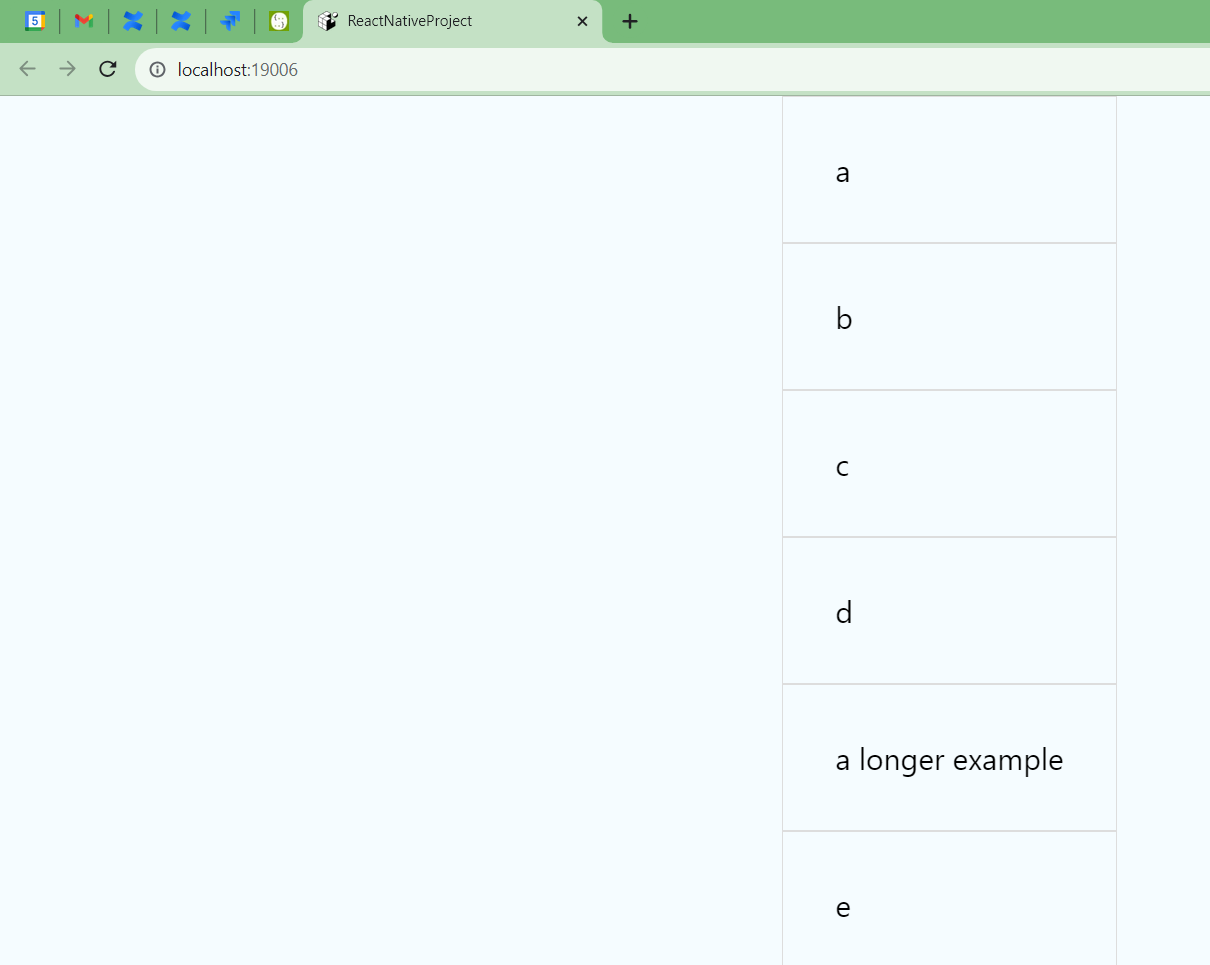
 backgroundColor: '#F5FCFF',

 },

 row: {fontSize: 24, padding: 42, borderWidth: 1, borderColor: '#DDDDDD'},

});

export default App;

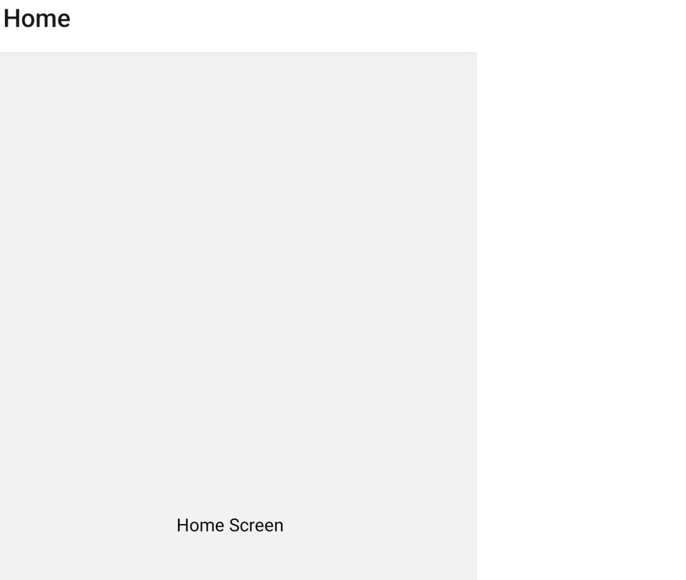


Пример 4. Пример навигации

Вводим команды для установки недостающих пакетов npm react:

*npx expo install react-navigation*

*npx expo install react-navigation-stack -- --force*



Выполнение индивидуальных заданий.

1. Реализовать переход между примером с кнопками и примером со списком с использованием React-Navigation (реализовать два разных экрана с примерами и меню навигации внизу экрана)
2. Поменять цвет и форму кнопок и текст внутри из примера с кнопками
3. Поменять список из примера на список книг с обложкой, названием и кратким описанием
4. Реализовать смену фона элементов списка при нажатии
5. Реализовать смену кнопки на картинку при нажатии
6. Реализовать удаление элементов из списка при долгом нажатии
7. Реализовать переход между примером с кнопками и примером со списком с использованием React-Navigation (реализовать два разных экрана с примерами и меню навигации внизу экрана)

import React, { Component } from 'react';

import { NavigationContainer } from '@react-navigation/native';

import { createBottomTabNavigator } from '@react-navigation/bottom-tabs';

import List1 from './ListExample';

import Button1 from './ButtonExample';

const Tab = createBottomTabNavigator();

const App = () => {

  return (

    <NavigationContainer>

      <Tab.Navigator>

        <Tab.Screen name="ButtonExample" component={Button1} />

        <Tab.Screen name="ListExample" component={List1} />

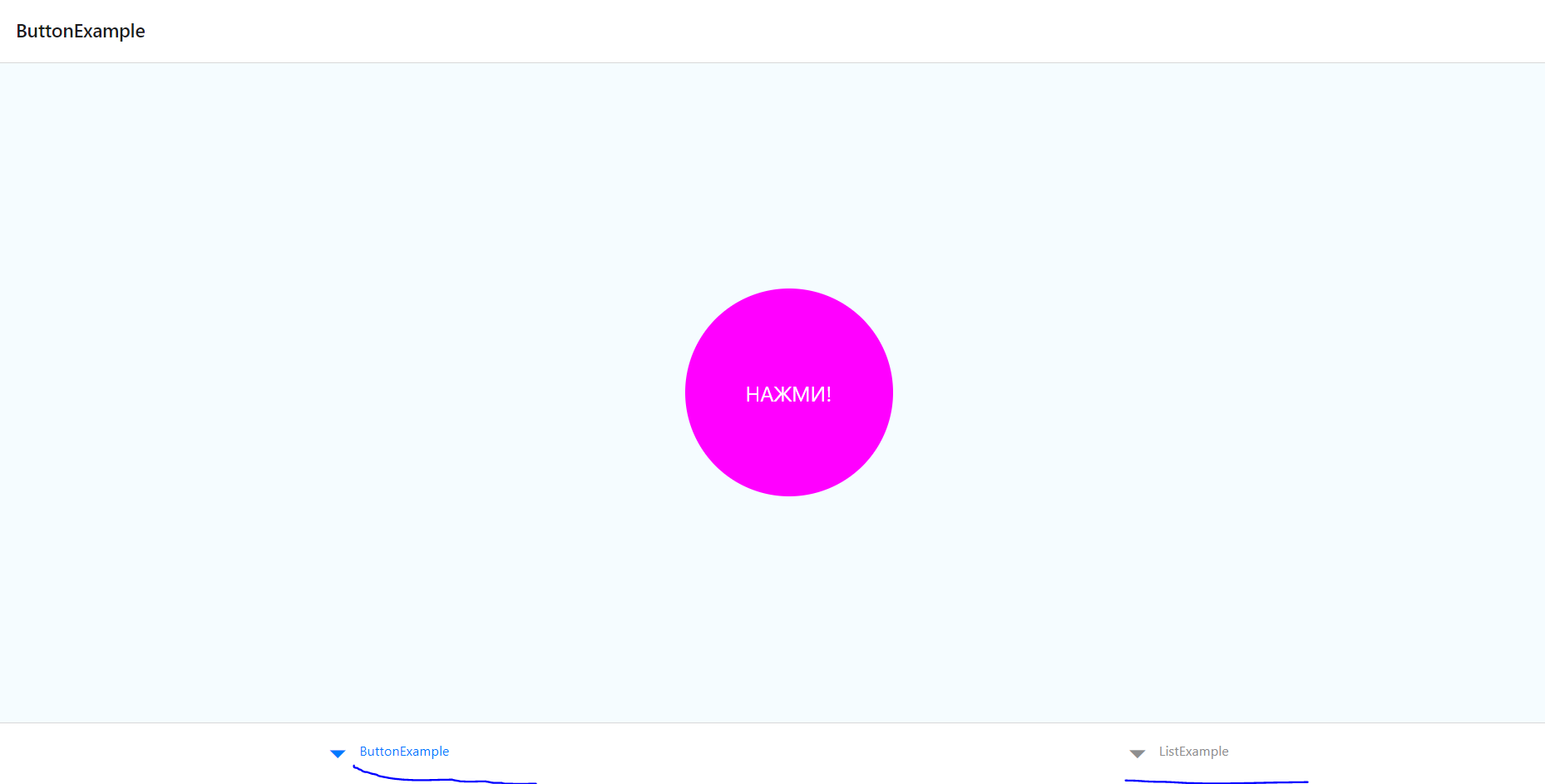
      </Tab.Navigator>

    </NavigationContainer>

  );

};

export default App;



1. Поменять цвет и форму кнопок и текст внутри из примера с кнопками

import React, { Component } from "react";

import { StyleSheet, Text, View, TouchableHighlight } from "react-native";

class App extends Component {

  constructor(props) {

    super(props);

    this.state = { pressing: false };

    }

    \_onPressIn = () => {

    this.setState({ pressing: true });

    };

    \_onPressOut = () => {

    this.setState({ pressing: false });

    };

    render() {

    return (

    <View style={styles.container}>

    <TouchableHighlight

    onPressIn={this.\_onPressIn}

    onPressOut={this.\_onPressOut}

    style={styles.touchable}

    >

    <View style={styles.button}>

    <Text style={styles.welcome}>

    {this.state.pressing ? "НАЖАТО!" : "НАЖМИ!"}

    </Text>

    </View>

    </TouchableHighlight>

    </View>

    );

    }

   }

   const styles = StyleSheet.create({

     container: {

flex: 1,

justifyContent: "center",

alignItems: "center",

backgroundColor: "#F5FCFF"

 },

welcome: { fontSize: 20, textAlign: "center", margin: 10, color: "#FFFFFF" },

touchable: { borderRadius: 120 },

button: {

backgroundColor: "#000AAA",

borderRadius: 120,

height: 300,

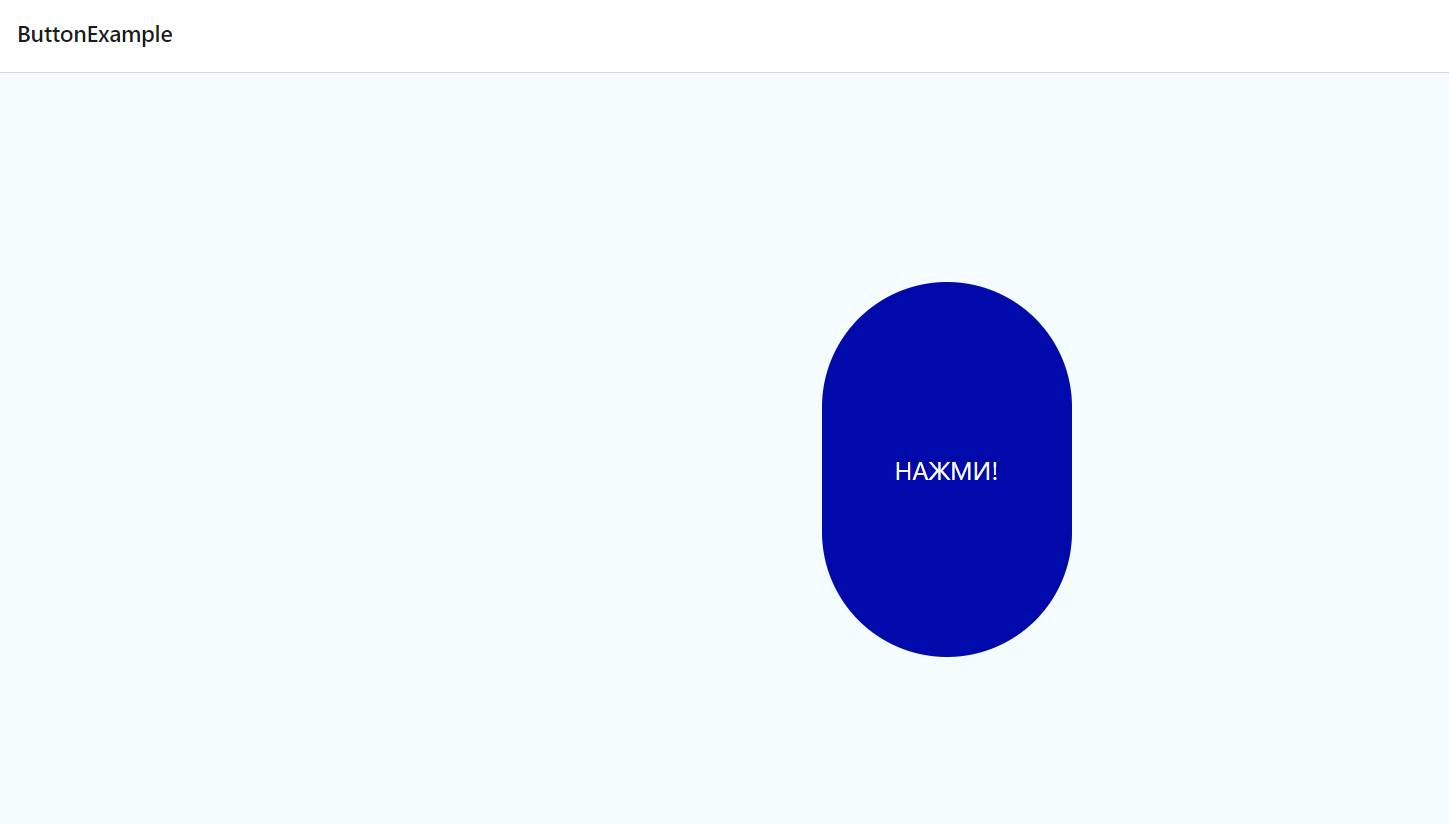
width: 200,

justifyContent: "center"

 }

});

export default App;



1. Поменять список из примера на список книг с обложкой, названием и кратким описанием

import React, {Component} from 'react';

import {StyleSheet, Image, Text, View, FlatList} from 'react-native';

class App extends Component {

 constructor(props) {

 super(props);

 this.state = {

 data: [

  {

    title: "Война и мир",

    description: "«Война́ и мир» — роман-эпопея Льва Николаевича Толстого, описывающий русское общество в эпоху войн против Наполеона в 1805—1812 годах.",

    img: "https://ir.ozone.ru/s3/multimedia-j/wc1000/6792956659.jpg"

  },

  {

    title: "1984",

    description: "«1984» — роман-антиутопия Джорджа Оруэлла, изданный в 1949 году. Как отмечает членкор РАН М. Ф. Черныш, это самое главное и последнее произведение писателя.",

    img: "https://ir.ozone.ru/s3/multimedia-c/wc1000/6332628816.jpg"

  },

  {

    title: "Двенадцать стульев.",

    description: "Илья Ильф и Евгений Петров — знаменитые фельетонисты и писатели-сатирики первой половины XX века.",

    img: "https://cdn1.ozone.ru/s3/multimedia-m/wc1200/6019574446.jpg"

  },

  ],

  };

  }

  \_renderItem = data => {

    return <View style={styles.row}>

        <Image style={styles.picture} source={{ uri: data.item.img }}></Image>

        <View >

            <Text> {data.item.title} </Text>

            <Text> {data.item.description} </Text>

        </View>

    </View>;

  };

  render() {

  return (

  <View style={styles.container}>

    <FlatList data={this.state.data} renderItem={this.\_renderItem} />

  </View>

  );

}

}

const styles = StyleSheet.create({

 container: {

 flex: 1,

 justifyContent: 'center',

 alignItems: 'center',

 backgroundColor: '#F5FCFF',

 },

 row: {fontSize: 24, padding: 42, borderWidth: 1, borderColor: '#DDDDDD'},

 picture: {

  width: 200,

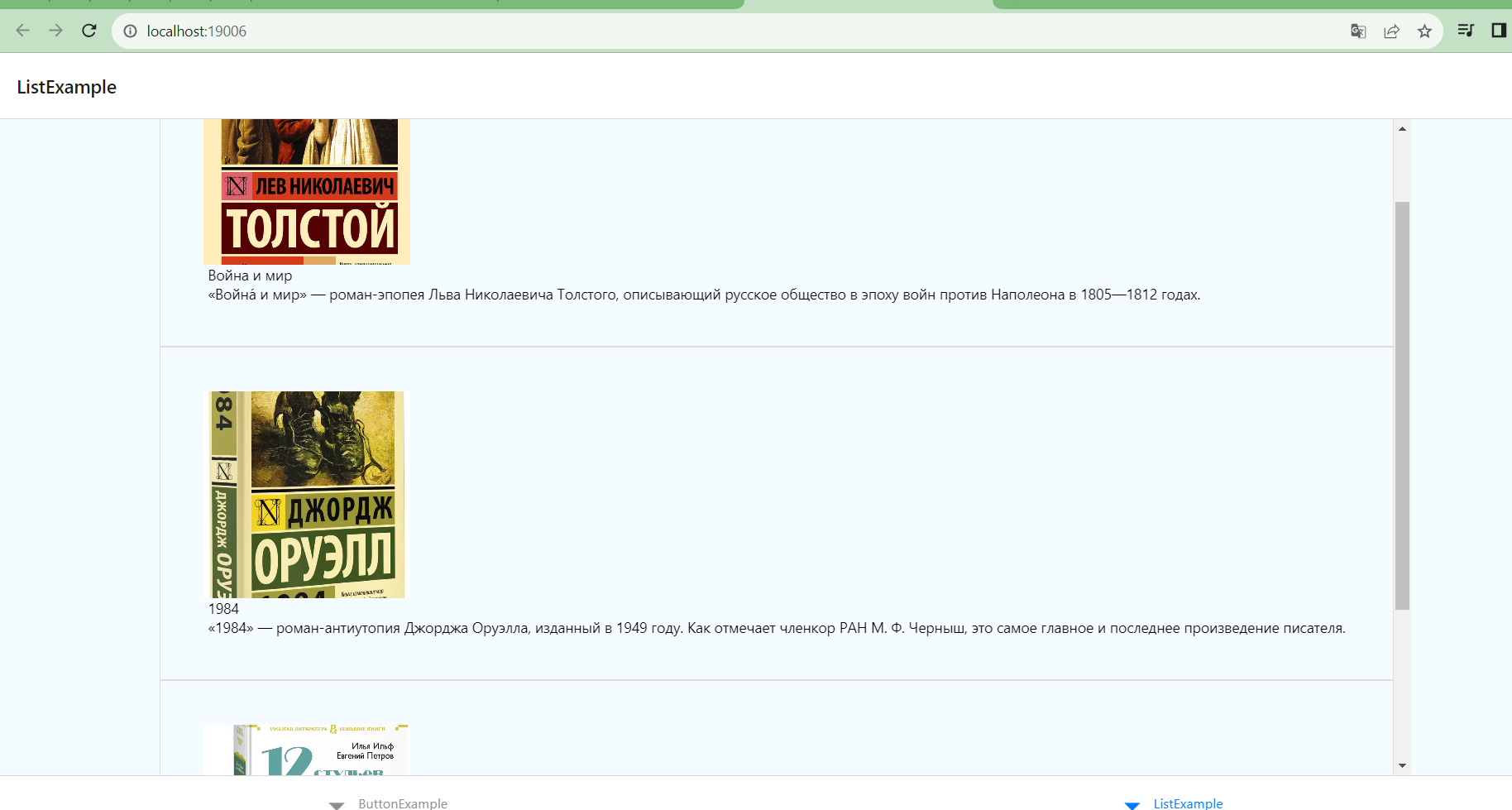
  height: 200,

  display: 'inline-block'

}

});

export default App;



1. Реализовать смену фона элементов списка при нажатии

import React, {Component} from 'react';

import {StyleSheet, Image, Text, View, FlatList, TouchableHighlight } from 'react-native';

class App extends Component {

  \_onPressIn = () => {

    this.setState({ pressing: true });

    };

    \_onPressOut = () => {

    this.setState({ pressing: false });

    };

 constructor(props) {

 super(props);

 this.state = {

  pressing: false,

 data: [

    {

        title: "Война и мир",

        description: "«Война́ и мир» — роман-эпопея Льва Николаевича Толстого, описывающий русское общество в эпоху войн против Наполеона в 1805—1812 годах.",

        img: "https://ir.ozone.ru/s3/multimedia-j/wc1000/6792956659.jpg"

      },

      {

        title: "1984",

        description: "«1984» — роман-антиутопия Джорджа Оруэлла, изданный в 1949 году. Как отмечает членкор РАН М. Ф. Черныш, это самое главное и последнее произведение писателя.",

        img: "https://ir.ozone.ru/s3/multimedia-c/wc1000/6332628816.jpg"

      },

      {

        title: "Двенадцать стульев.",

        description: "Илья Ильф и Евгений Петров — знаменитые фельетонисты и писатели-сатирики первой половины XX века.",

        img: "https://cdn1.ozone.ru/s3/multimedia-m/wc1200/6019574446.jpg"

      },

  ],

  };

  }

  \_renderItem = data => {

    return <TouchableHighlight

    onPressIn={this.\_onPressIn}

    onPressOut={this.\_onPressOut}>

    <View style={styles.row}>

        <Image style={styles.picture} source={{ uri: data.item.img }}></Image>

        <View >

            <Text> {data.item.title} </Text>

            <Text> {data.item.description} </Text>

        </View>

    </View>

    </TouchableHighlight>;

  };

  render() {

  return (

  <View style={styles.container}>

    <FlatList data={this.state.data} renderItem={this.\_renderItem} />

  </View>

  );

}

}

const styles = StyleSheet.create({

 container: {

 flex: 1,

 justifyContent: 'center',

 alignItems: 'center',

 backgroundColor: '#F5FCFF',

 },

 row: {fontSize: 24, padding: 42, borderWidth: 1, borderColor: '#DDDDDD'},

 picture: {

  width: 200,

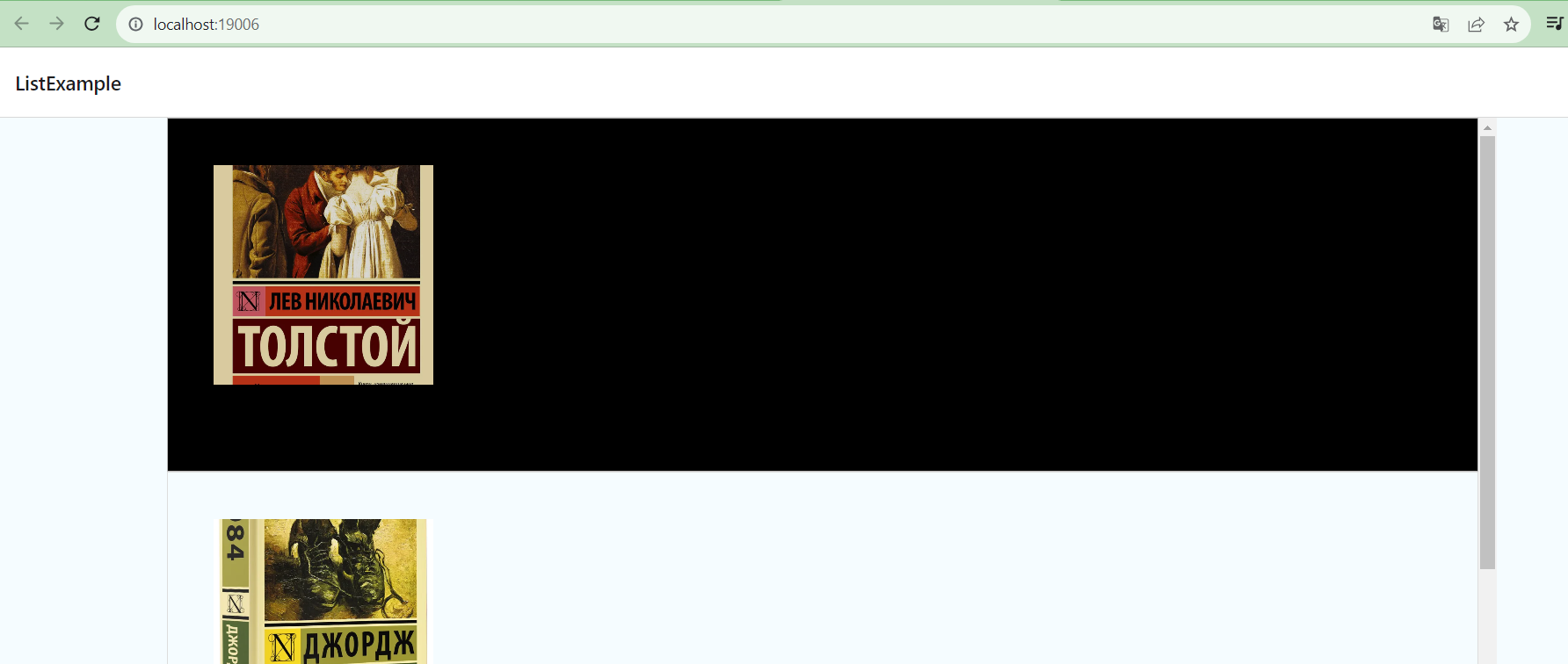
  height: 200,

  display: 'inline-block'

}

});

export default App;



1. Реализовать смену кнопки на картинку при нажатии

import React, { Component } from "react";

import { StyleSheet, Text, View, Image, TouchableHighlight } from "react-native";

class App extends Component {

  constructor(props) {

    super(props);

    this.state = { pressing: false };

    }

    \_onPressIn = () => {

    this.setState({ pressing: true });

    };

    \_onPressOut = () => {

    this.setState({ pressing: false });

    };

    render() {

    return (

    <View style={styles.container}>

    <TouchableHighlight

    onPressIn={this.\_onPressIn}

    onPressOut={this.\_onPressOut}

    style={styles.touchable}

    >

    {

    !this.state.pressing ?

      <View style={styles.button}>

      <Text style={styles.welcome}>

      {this.state.pressing ? "НАЖАТО!" : "НАЖМИ!"}

      </Text>

      </View>

      : <Image style={styles.picture} source={{uri: 'https://i.pinimg.com/originals/6d/fb/26/6dfb264952a4115ff87feacfbe4f08d9.gif'}}/>

    }

    </TouchableHighlight>

    </View>

    );

    }

   }

   const styles = StyleSheet.create({

     container: {

flex: 1,

justifyContent: "center",

alignItems: "center",

backgroundColor: "#F5FCFF"

 },

 welcome: { fontSize: 20, textAlign: "center", margin: 10, color: "#FFFFFF" },

 touchable: { borderRadius: 120 },

 button: {

 backgroundColor: "#000AAA",

 borderRadius: 120,

 height: 300,

 width: 200,

 justifyContent: "center"

 },

 picture: {

  width: 400,

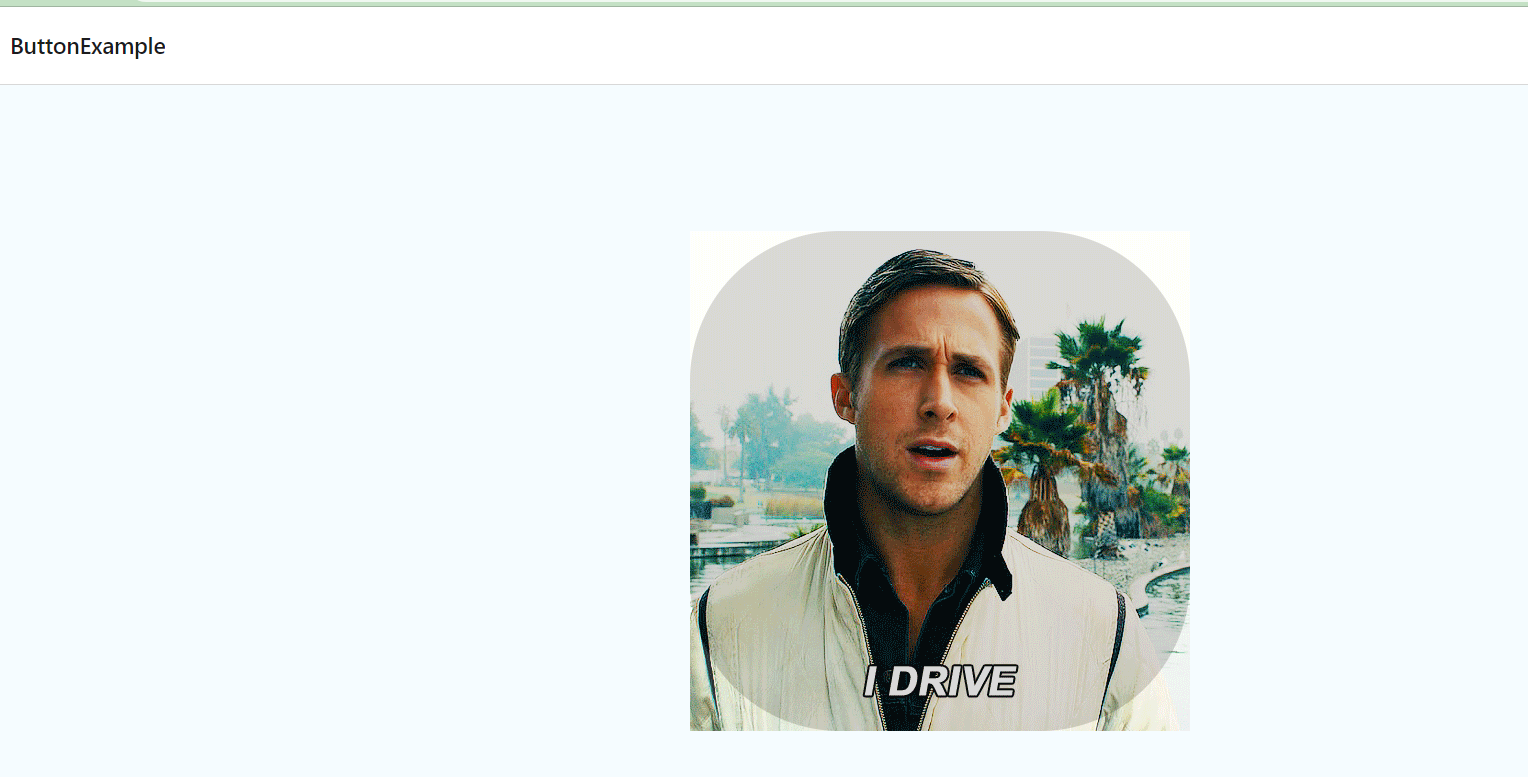
  height: 400,

  display: 'inline-block'

}

});

export default App;



1. Реализовать удаление элементов из списка при долгом нажатии

import React, {Component} from 'react';

import {StyleSheet, Image, Text, View, FlatList, TouchableHighlight} from 'react-native';

class App extends Component {

 constructor(props) {

 super(props);

 this.state = {

 data: [

  {

    title: "Война и мир",

    description: "«Война́ и мир» — роман-эпопея Льва Николаевича Толстого, описывающий русское общество в эпоху войн против Наполеона в 1805—1812 годах.",

    img: "https://ir.ozone.ru/s3/multimedia-j/wc1000/6792956659.jpg"

  },

  {

    title: "1984",

    description: "«1984» — роман-антиутопия Джорджа Оруэлла, изданный в 1949 году. Как отмечает членкор РАН М. Ф. Черныш, это самое главное и последнее произведение писателя.",

    img: "https://ir.ozone.ru/s3/multimedia-c/wc1000/6332628816.jpg"

  },

  {

    title: "Двенадцать стульев.",

    description: "Илья Ильф и Евгений Петров — знаменитые фельетонисты и писатели-сатирики первой половины XX века.",

    img: "https://cdn1.ozone.ru/s3/multimedia-m/wc1200/6019574446.jpg"

  }

  ],

  };

  }

  \_remove = (item) => {

    this.setState({

        data: this.state.data.filter(element => item.item.title !== element.title)

    })}

  \_renderItem = dataP => {

    return <TouchableHighlight

    onLongPress={() => this.\_remove(dataP)}>

     <View style={styles.row}>

        <Image style={styles.picture} source={{ uri: dataP.item.img }}></Image>

        <View >

            <Text> {dataP.item.title} </Text>

            <Text> {dataP.item.description} </Text>

        </View>

    </View>

    </TouchableHighlight>

  };

  render() {

  return (

  <View style={styles.container}>

    <FlatList data={this.state.data} renderItem={this.\_renderItem} />

  </View>

  );

}

}

const styles = StyleSheet.create({

 container: {

 flex: 1,

 justifyContent: 'center',

 alignItems: 'center',

 backgroundColor: '#F5FCFF',

 },

 row: {fontSize: 24, padding: 42, borderWidth: 1, borderColor: '#DDDDDD'},

 picture: {

  width: 200,

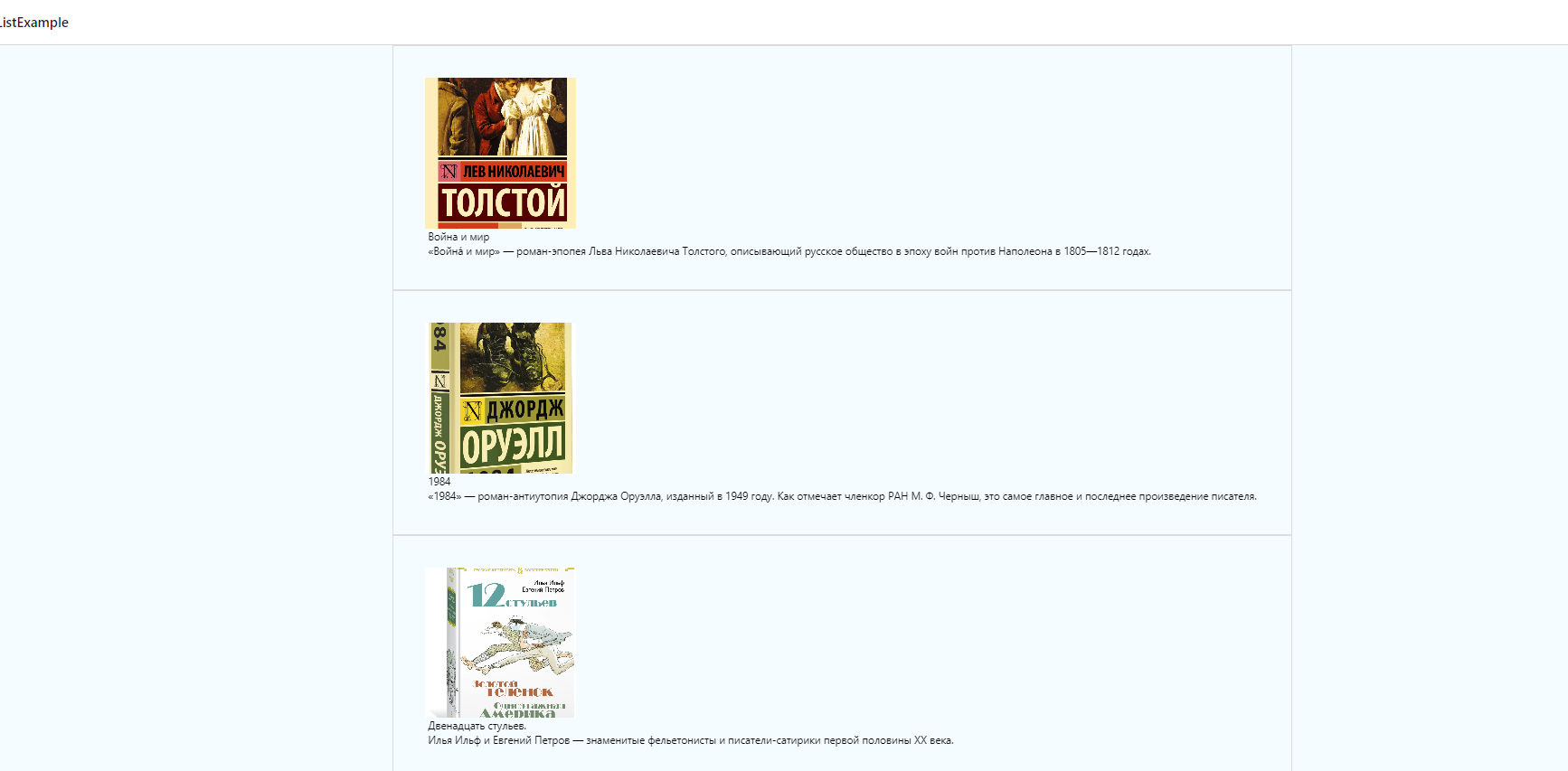
  height: 200,

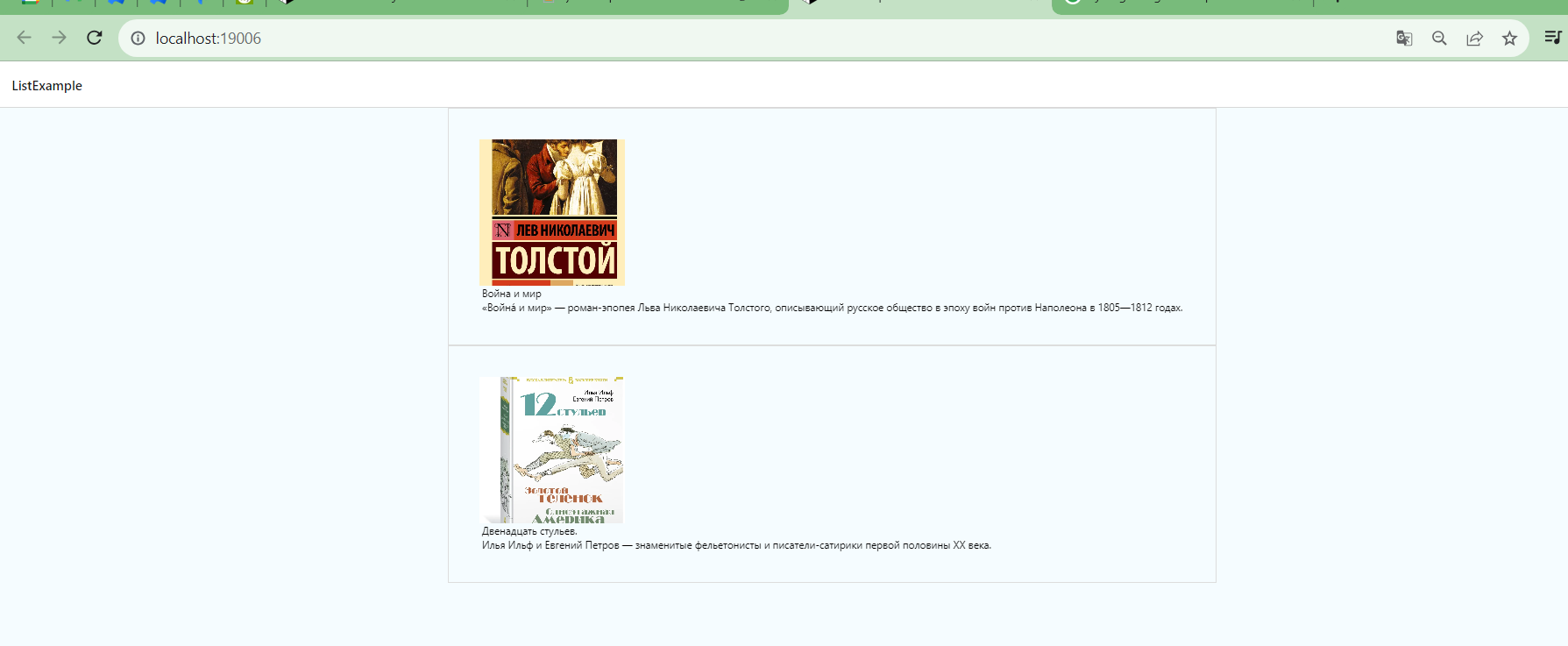
  display: 'inline-block'

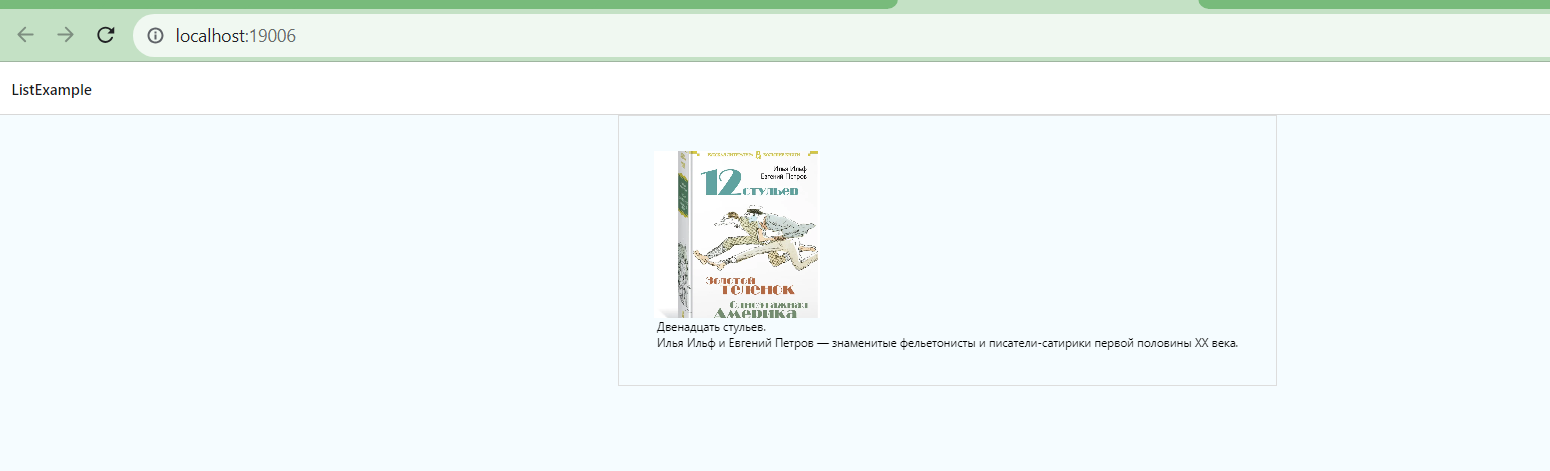
}

});

export default App;







Вывод

В результате выполнения работы я реализовал приложение с использованием фреймворка React Native.