

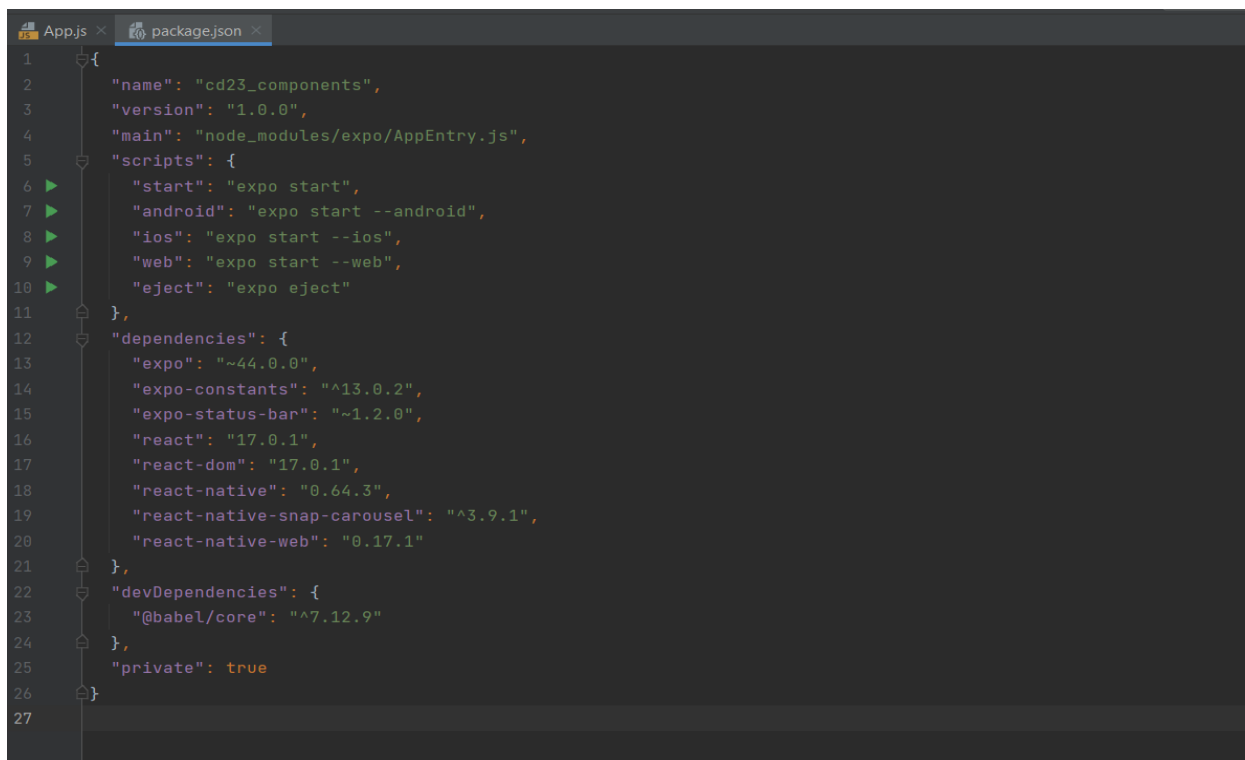
# Отчёт по проделанной работе на семинаре по мобильной разработке (01.04.2022).

(выполнил Валяев Георгий Анатольевич)

React-native – один из интересных и богатых фреймворков, наследующих большинство свойств фреймворка React (в рамках разработки нативных веб-приложений), разработанный командой Facebook для создания и написания приложений под мобильные устройства с поддержкой ОС Android, iOS и других систем.

В этой методичке мы рассмотрим функционал в процессе работы с основными компонентами приложения на React-Native и создадим небольшую сводку новостей из журнала Bright в форме карусели как вариант простого и наглядного практического примера использования всех основных компонентов. Также изучим небольшой функционал обработчика нажатия на кнопку, реализовав его в примере дополнительно.

Для начала установим все необходимые нам модули через терминал в файле package.json:



```
1 {
2   "name": "cd23_components",
3   "version": "1.0.0",
4   "main": "node_modules/expo/AppEntry.js",
5   "scripts": {
6     "start": "expo start",
7     "android": "expo start --android",
8     "ios": "expo start --ios",
9     "web": "expo start --web",
10    "eject": "expo eject"
11  },
12  "dependencies": {
13    "expo": "~44.0.0",
14    "expo-constants": "^13.0.2",
15    "expo-status-bar": "~1.2.0",
16    "react": "17.0.1",
17    "react-dom": "17.0.1",
18    "react-native": "0.64.3",
19    "react-native-snap-carousel": "^3.9.1",
20    "react-native-web": "0.17.1"
21  },
22  "devDependencies": {
23    "@babel/core": "^7.12.9"
24  },
25  "private": true
26 }
```

Для использования нам будут необходимы при импорте всех объектов следующие модули: `react-native`, `react`, `react-native-snap-carousel`, `expo-constants`.

Перейдём к главному компоненту приложения – `App.js`:



```
1 import {Alert, Button, StyleSheet, Text, View, ScrollView, Image} from 'react-native'
2 import Carousel, {Pagination} from 'react-native-snap-carousel'
3 import Constants from 'expo-constants'
4 import {useState} from 'react'
5
6 const data = [
7   {
8     title: "Foundawtion: творческий проект Сенегала",
9     body: "Foundawtion - это инициатива, в рамках которой группа талантливых архитекторов превратила юг Сенегала в творческий эпицентр",
10    imgUrl: "http://brightmagazine.ru/wp-content/uploads/2022/03/pexels-hello-massamba-10509940-768x512.jpg"
11  },
12  {
13    title: "Гитхорн: уголок в Нидерландах, где совсем нет автомобилей",
14    body: "Десятки каналов и 176 мостов - богатство спокойной голландской деревушки Гитхорн. Там почтальон привозит письма на лодке, а жители общаются с соседями в кафе.",
15    imgUrl: "http://brightmagazine.ru/wp-content/uploads/2022/03/pexels-tanathip-rattanatun-2026451-768x512.jpg"
16  },
17  {
18    title: "Самые лучшие страны мира для путешествий в одиночку",
19    body: "Иногда в путешествие хочется отправиться в одиночку, чтобы побыть наедине с самим собой и найти ответы на жизненные вопросы.",
20    imgUrl: "http://brightmagazine.ru/wp-content/uploads/2022/02/pexels-valentin-antonucci-691637-768x512.jpg"
21  }
22 ]
23
24 export default function App() {
25   const [activeSlide, setActiveSlide] = useState( initialState: 0 );
26   const renderItem = ({ item, index }) => {
27     return (
28       <ScrollView style={styles.card}>
29         <Text style={styles.header}>{item.title}</Text>
30         <Image
31           source={{ uri: item.imgUrl }}
32           style={styles.logoStyle}
33         />
34         <Text style={styles.body}>{item.body}</Text>
35       </ScrollView>
36     );
37   };
38
39   return (
40     <View style={styles.app}>
41       <Carousel
42         layout="default"
43         data={data}
44         renderItem={renderItem}
45         sliderWidth={300}
46         itemWidth={290}
47         onSnapToItem={index => setActiveSlide(index)}
48         swipeThreshold={100}
49         inactiveSlideOpacity={0.4}
50         containerCustomStyle={{
51           overflow: "visible",
52           marginVertical: 5,
53         }}
54       />
55       <Pagination
56         dotsLength={data.length}
57         activeDotIndex={activeSlide}
```

```
Appjs x package.json x
55     <Pagination
56       dotsLength={data.length}
57       activeDotIndex={activeSlide}
58       dotStyle={{
59         width: 10,
60         height: 10,
61         borderRadius: 10,
62       }}>
63     </Pagination>
64     <Button
65       title="Press me"
66       onPress={() => Alert.alert( title: 'Thank you for pressing me!')}
67       color='purple'
68     />
69   </View>
70 );
71 }
72
73 const styles = StyleSheet.create({
74   app: {
75     flex: 1,
76     paddingTop: Constants.statusBarHeight,
77   },
78   card: {
79     marginLeft: 20
80   },
81   logoStyle: {
82     width: 200,
83     height: 300,
84     marginLeft: 20,
```

```
Appjs x package.json x
79     marginLeft: 20
80   },
81   logoStyle: {
82     width: 200,
83     height: 300,
84     marginLeft: 20,
85     borderRadius: 15
86   },
87   header: {
88     color: 'black',
89     fontSize: 28,
90     fontWeight: "bold",
91     marginBottom: 20,
92     marginLeft: 20,
93     marginTop: 10,
94   },
95   body: {
96     color: 'black',
97     fontSize: 18,
98     marginLeft: 20,
99     marginTop: 25,
100    marginRight: 40
101  },
102  });
103
```

В начале импортируем все скачанные модули для работы карусели с новостями и отображения всех основных компонентов, после чего в отдельной переменной data (как объекте) мы размещаем все необходимые нам данные для отображения: заголовок новости, картинка (через абсолютную URL-ссылку) и наполнение новости.

Далее в основной функции компонента (App) изначально задаём 2 переменные как `state` для отслеживания текущего слайда в карусели и состояния перелистывания на следующий слайд. Потом для полноценного отображения каждого слайда как новости создаём функцию её рендеринга, в которой задаётся объект `ScrollView`, состоящий уже из `Text` и `Image`. На входе у нас подаётся объект, из которых в целом и состоит `data`, который мы уже объявили. Следовательно, функция предназначена для последовательного отображения каждой новости.

В финальном отображении функционального главного компонента App в большом объекте `View` (`flex: 1`) мы начинаем расположение каждой новости (функция правильного рендеринга наполнения, размеры для элемента, отображение состояния готовности для перелистывания на следующий слайд и стилизация) и точек для перелистывания на следующие новости из журнала. Дополнительно, мы в кнопке задаём сразу же заголовок, функция обработчика нажатия на неё и цвет самой кнопки.

В большой отдельной переменной как объекте `styles` мы задаём основные параметры для стилизации всех объектов компонента приложения: внутренние (`padding`) и внешние отступы (`margin`) с разных сторон (`Left`, `Right`, `Top`, `Bottom`), тип и цвет шрифта (жирный, курсив, классика), размер объекта по ширине и высоте, коэффициент закругления краёв объекта (`borderRadius`) и, что самое важное, `flex: 1`, что означает полноценную занятость экрана главным компонентом приложения `App.js`

Рассмотрим получившийся результат с мобильного устройства при помощи синхронизации через QR-код и приложение на телефоне Expo Go:

## Foundawtion: творческий проект Сенегала



Foundawtion – это инициатива, в рамках которой группа талантливых архитекторов превратила юг Сенегала в творческий эпицентр.



PRESS ME



## Гитхорн: уголок в Нидерландах, где совсем нет автомобилей



Десятки каналов и 176 мостов – богатство спокойной голландской деревушки Гитхорн. Там почтальон привозит письма на лодке, а автомобилей нет и в помине. Идеальное



PRESS ME



# Самые лучшие страны мира для путешествий в одиночку



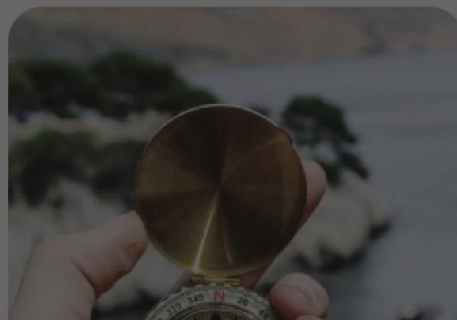
Иногда в путешествие хочется отправиться в одиночку, чтобы побыть наедине с самим собой и найти ответы на жизненные вопросы. Сегодня мы расскажем вам



PRESS ME



# Самые лучшие страны мира для путешествий в одиночку



Thank you for pressing me!

OK

Иногда в путешествие хочется отправиться в одиночку, чтобы побыть наедине с самим собой и найти ответы на жизненные вопросы. Сегодня мы расскажем вам



PRESS ME



Приложение успешно выполнило свой функционал!

## Контрольные вопросы

1. Какой компонент является базовым?
1. Так или иначе в течение проектирования и разработки мобильного продукта на React-Native будут выделяться следующие основные (базовые) компоненты: View, Text, Image, TextInput, ScrollView, StyleSheet).