

# Отчёт по проделанной работе на семинаре по мобильной разработке (13.05.2022).

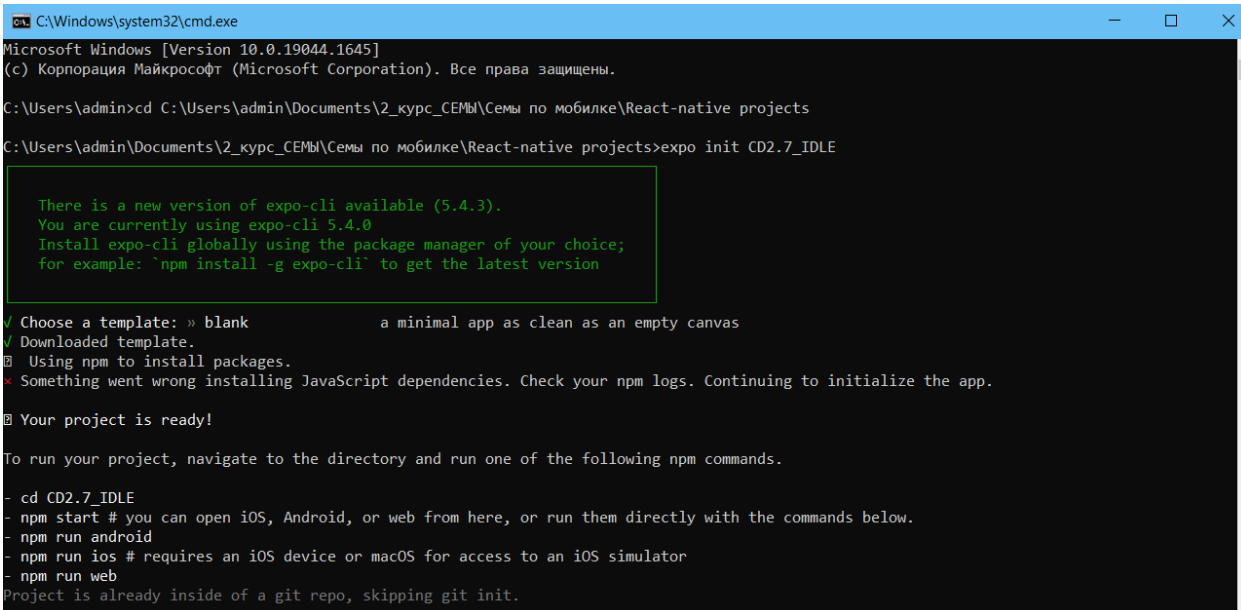
(выполнил Валяев Георгий Анатольевич)

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 работе освоим навык установки среды разработки для React Native и создания своей собственной среды на компьютере.

Полноценный код всего проекта расположен в моём репозитории GitHub: ...  
(ссылка на репозиторий)

Для изначального запуска проекта был установлен клиент Ехро (expo-cli), и через команду `expo init` создан новый концепт будущего мобильного приложения на React Native:



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 10.0.19044.1645]
(c) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

C:\Users\admin>cd C:\Users\admin\Documents\2_курс_СЕМЫ\Семы по мобилке\React-native projects
C:\Users\admin\Documents\2_курс_СЕМЫ\Семы по мобилке\React-native projects>expo init CD2.7_IDLE

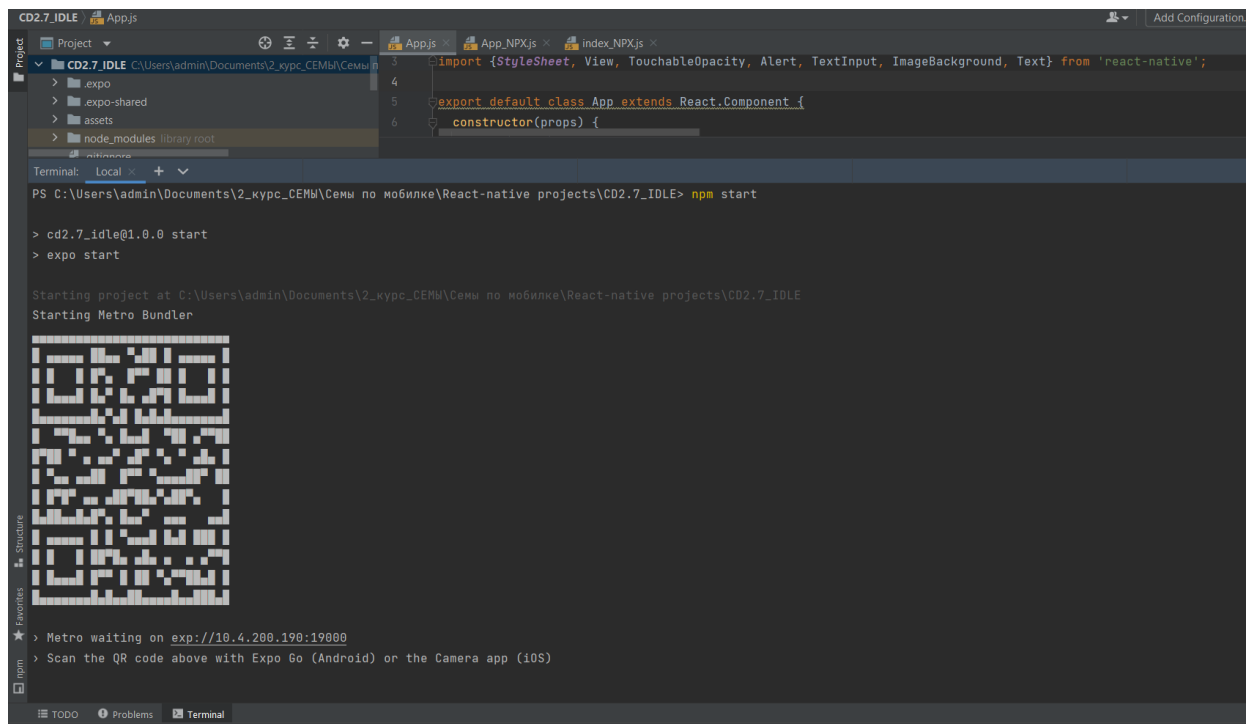
  There is a new version of expo-cli available (5.4.3).
  You are currently using expo-cli 5.4.0
  Install expo-cli globally using the package manager of your choice;
  for example: `npm install -g expo-cli` to get the latest version

✓ Choose a template: » blank          a minimal app as clean as an empty canvas
✓ Downloaded template.
⊠ Using npm to install packages.
✗ Something went wrong installing JavaScript dependencies. Check your npm logs. Continuing to initialize the app.
⊠ Your project is ready!

To run your project, navigate to the directory and run one of the following npm commands.

- cd CD2.7_IDLE
- npm start # you can open iOS, Android, or web from here, or run them directly with the commands below.
- npm run android
- npm run ios # requires an iOS device or macOS for access to an iOS simulator
- npm run web
Project is already inside of a git repo, skipping git init.
```

При необходимости запустить проект необходима команда `npm start` (или `npm run android` (или `ios`), а также (как аналог) `expo start`):





Получившаяся работа небольшого приложения Expo:

10:20 🔍 🔒 •

📶 76%

Введите любой текст

dreamstime.

Нажми на меня или на Enter



10:20 🔍 🔒 •

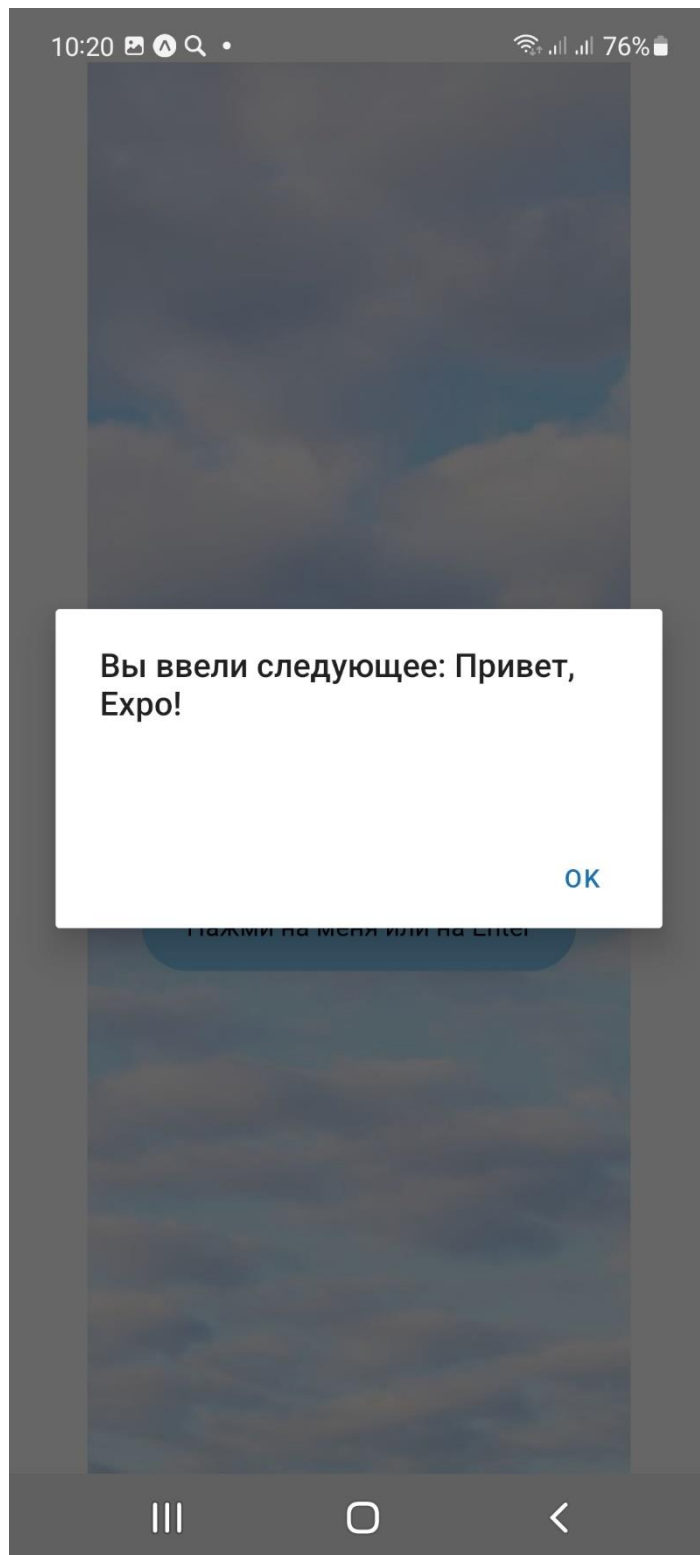
📶 76%

Привет, Ехро!

dreamstime.

Нажми на меня или на Enter





Далее реализуем этот же функционал приложения уже в другой (альтернативной) среде разработки — React Native CLI следующей командой:

```
npx react-native init:
```

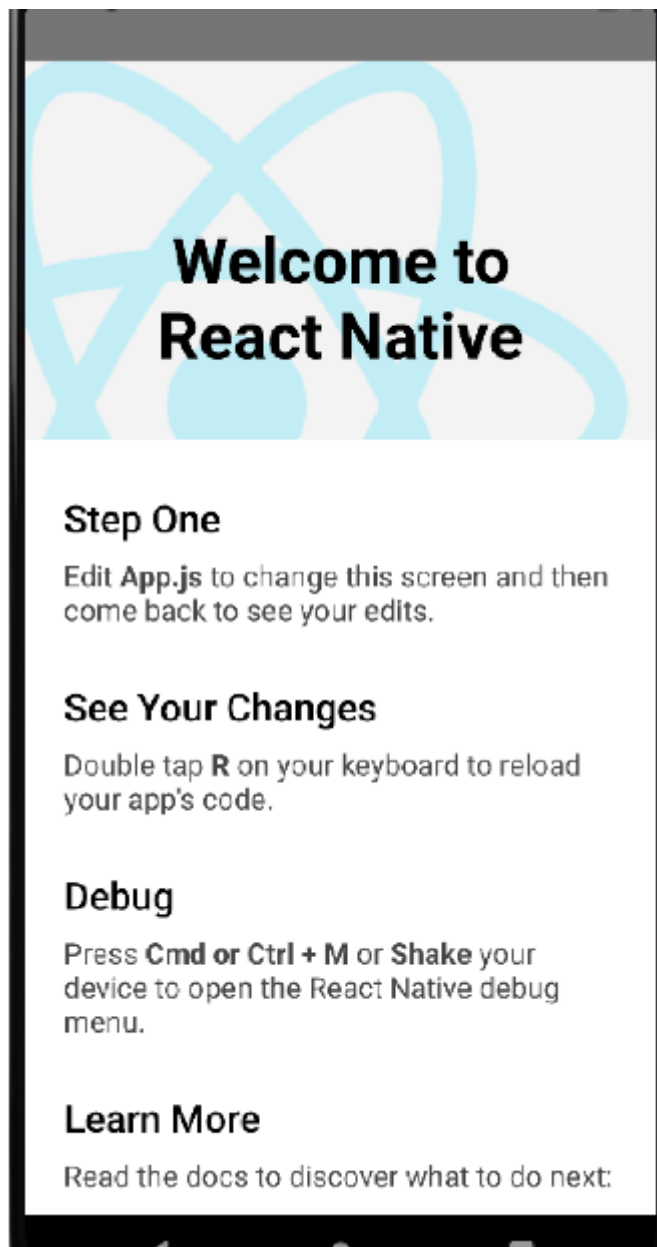


```
To reload the app press "r"
To open developer menu press "d"

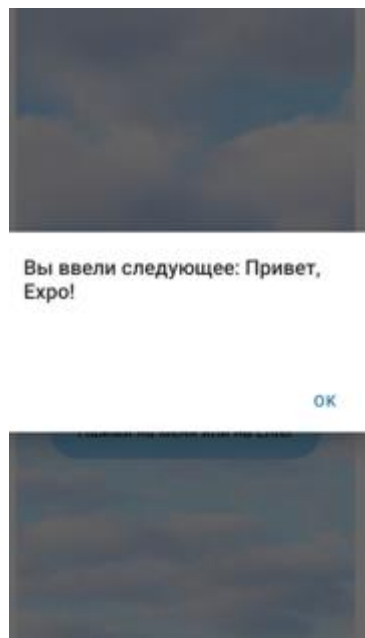
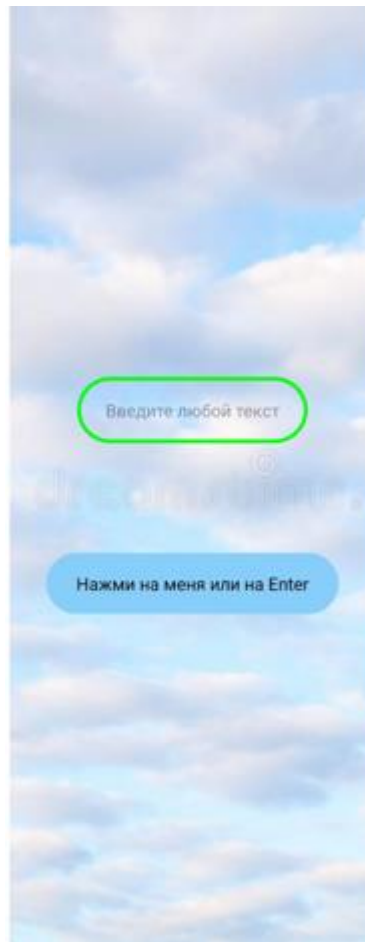
warn No apps connected. Sending "reload" to all React Native apps failed. Make sure your app is running in the simulator
or on a phone connected via USB.
info Reloading app...
info Reloading app...
BUNDLE ./index.js

LOG Running "CD27_NPXDE" with {"rootTag":1}
```

Будет виден стартовый экран проекта React Native:



Изменим программную составляющую до функционала, который был представлен в другой среде разработки:



Здесь (в файле `index.js`) через команду `npx react-native init` производится регистрация компонента `App`, впоследствии который был изменён под наш функционал:



```

4
5 import {AppRegistry} from 'react-native';
6 import App from './App';
7 import {name as appName} from './app.json';
8
9 AppRegistry.registerComponent(appName, () => App);
10

```

Главное отличие от среды разработки Expo (expo-cli среда), если уже говорить о коде JavaScript, то оно состоит в том, что приложения, основанные на технологии React, требуют в обязательном порядке регистрации корневого компонента вашего приложения с помощью AppRegistry.registerComponent. Но в Expo по умолчанию один допускаемый и главный компонент (App) автоматически регистрируется в экспорте по умолчанию (в файле App.js) как исходный корневой компонент проекта.

Главным недостатком среды expo-cli является тот факт, что он работает только для приложений, разработанных и написанных на чистом языке JavaScript. При необходимости сторонних собственных модулей (Kotlin, Swift, Java и так далее) нужно будет также извлечь их. Для таких целей предоставляется команда eject: экспорт приложения в тот же формат, который использует и react-native CLI. И в него уже впоследствии можно будет добавлять новые собственные модули.

### Контрольные вопросы

1. Какие основные принципы UI есть одновременно и в Human Interface Design и в Material Design?
1. Material Design и Human Interface выполняют основные положения принципа KISS — задачи решаются минимальным числом действий, они в целом интуитивно понятны для пользователя, следовательно должен быть реализован простой и понятный интерфейс, подчёркивая очевидность и понятность во всём.

При этом одинаково начинают выглядеть дизайн-элементы (от линейных иконок и до переключателей).

А тренды, соблюдаемые компаниями в наше время, потихоньку начинают сходиться: гармоничные и скруглённые углы, техника оптических размеров в UI-типографике, сокращение визуального шума, вывод контента на первый план, популярная персонализация на основе обоев и системной темы пользователя, а также техника LCH (яркость, цвет, оттенок), которая позволяет без особых усилий генерировать доступные цветовые сочетания.