## Отчёт по проделанной работе на семинаре по мобильной разработке (05.05.2022).

(выполнил Валяев Георгий Анатольевич)

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

В этой работе мы попытаемся изучить и рассмотреть основные главные принципы анимации в React Native приложениях, реализовав интересные наглядные примеры, пользующиеся популярностью.

Полноценный код всего проекта расположен в моём репозитории GitHub совместно с видео работы анимаций: <a href="https://github.com/Valyaevgeorgiy/Android\_projects/tree/main/React-Native/Animations">https://github.com/Valyaevgeorgiy/Android\_projects/tree/main/React-Native/Animations</a>.

Приложение в основе своём имеет 2 экрана как два компонента стека навигации (через «Stack.Navigator/»). На первой из которых будет отображена анимация с вращающейся и движущейся картинкой мема.

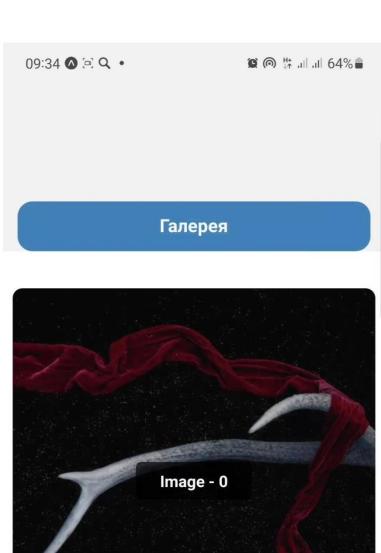
При нажатии кнопки и перехода к галерее изображений мы сможем видеть совершенно иные анимации вращения и увеличения размера, которые можно выполнять бесконечное количество раз (spring).

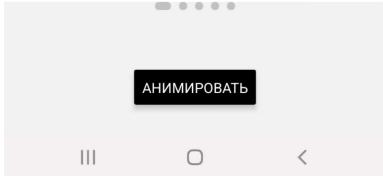
```
🚣 App.js
뤒 babel.config.js
          {images.map((image : string , imageIndex : number ) => {
                 <AnimatedImageCardView
                     key={imageIndex}
                     index={imageIndex}
                   <ImageBackground source={{ uri: image }} style={styles.card}>
                     <View style={styles.textContainer}>
                         {"Image - " + imageIndex}
                       </Text>
                     </View>
                   </ImageBackground>
                 </AnimatedImageCardView>
        </ScrollView>
        <View style={styles.indicatorContainer}>
          {images.map((image : string , imageIndex : number ) => {
            const width = scrollX.interpolate( config: {
                windowWidth * (imageIndex - 1),
                windowWidth * imageIndex,
                 windowWidth * (imageIndex + 1)
```

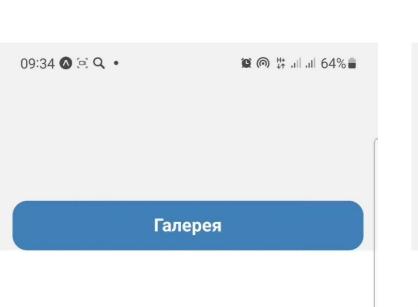
Получившийся результат работы приложения выглядит следующим образом:

(на след. страницах)

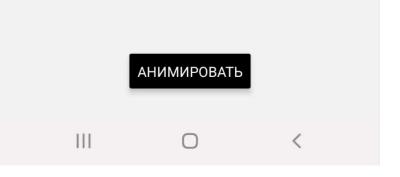




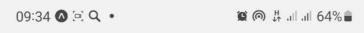




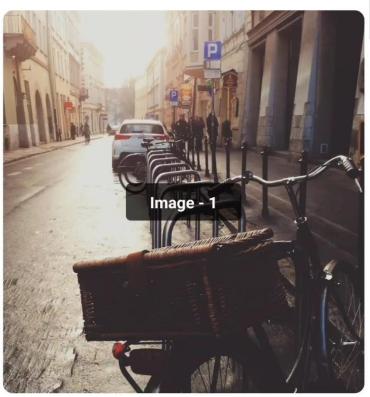


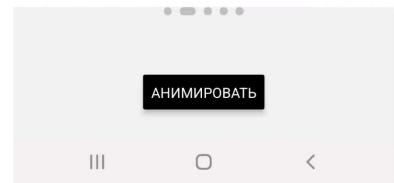


----



## Галерея





## Контрольные вопросы

- 1. Что такое параллакс-эффект?
- 1. Параллакс-эффект из себя представляет анимацию смещения близкого объекта относительно далёкого при изменении угла зрения.