Animations

Jogesh K. Muppala

THE DEPARTMENT OF
COMPUTER SCIENCE & ENGINEERING
計算機科學及工程學系

香港科技大學
THE HONG KONG UNIVERSITY OF
SCIENCE AND TECHNOLOGY

Animations

- Adds to better user experience
- React Native supports a powerful Animated API
  - Fluid, powerful and easy to build
  - Declarative relationships between inputs and outputs with configurable transforms in between
  - Simple start/stop methods to control time-based animation execution

Animated API

- Values in Animated:
  - Animated.Value(): single values
  - Animated.ValueXY(): vector
  - Interpolate() to map input ranges to different output ranges
- Animation Types:
  - Animated.decay(): fast start and slows to a stop
  - Animated.spring(): spring physics model
  - Animated.timing(): animate values over time using easing functions

Animated API

- Animated components:
  - Animated.Image
  - Animated.ScrollView
  - Animated.Text
  - Animated.View
- Can compose animations: delay(), parallel(), sequence(), stagger()

React-native-animatable

- Standard set of easy to use animations and declarative transitions
- Declarative precomposed animations easy to apply to Animatable.View, Text and Image
- Several properties: animation, duration, delay, direction, easing, iterationCount, transition

React-native-animatable

- Imperative use:
  - Animations exposed as functions on animatable elements
  - Get a reference to an element
  - Apply animation functions
  - Takes a duration argument
  - Returns a promise that is resolved when animation completes successfully or is cancelled