

# Testing React Native apps with React Native Testing Library

Tanner West



# Outline

1. Cover the What, Why, and How of React Native testing
  - a. Define some key terms
  - b. Explain why testing matters
  - c. Describe the methods and tools we'll be using
2. Write some tests
  - a. Unit test for an individual component
  - b. Integration test for an app screen



# What are unit tests?

- Test an individual “unit” of your code
- Means different things depending on who you ask and what type of software you’re writing
- For us, it will mean testing an individual component



# What are integration tests?

- Testing how multiple units “integrate” together in a larger context
- In our case, we’ll be looking at how multiple components integrate into a screen to provide a UX flow
- These more closely reflect how software is actually used



# Why should we write tests to begin with?

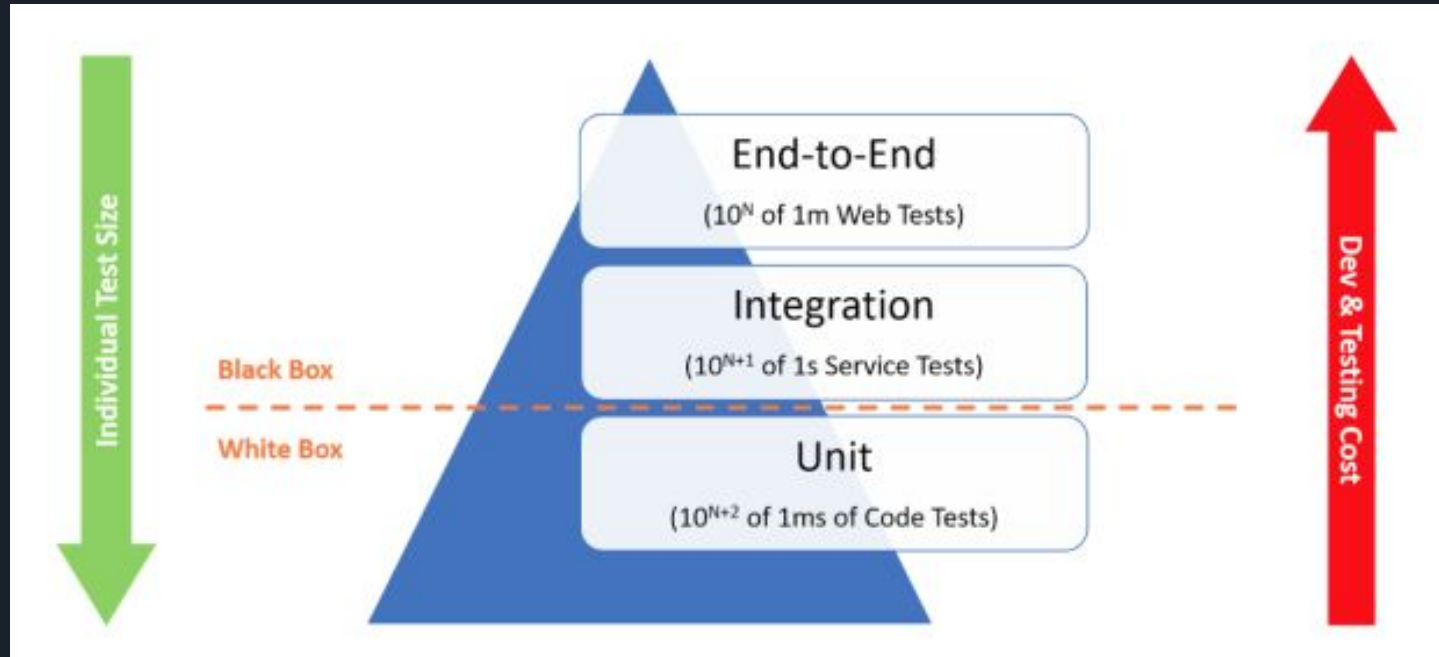
- It comes down to one thing:
  - 💪 Confidence 💪
  - ...that we won't have regressions
  - ...that our users will have the experience we intend for them



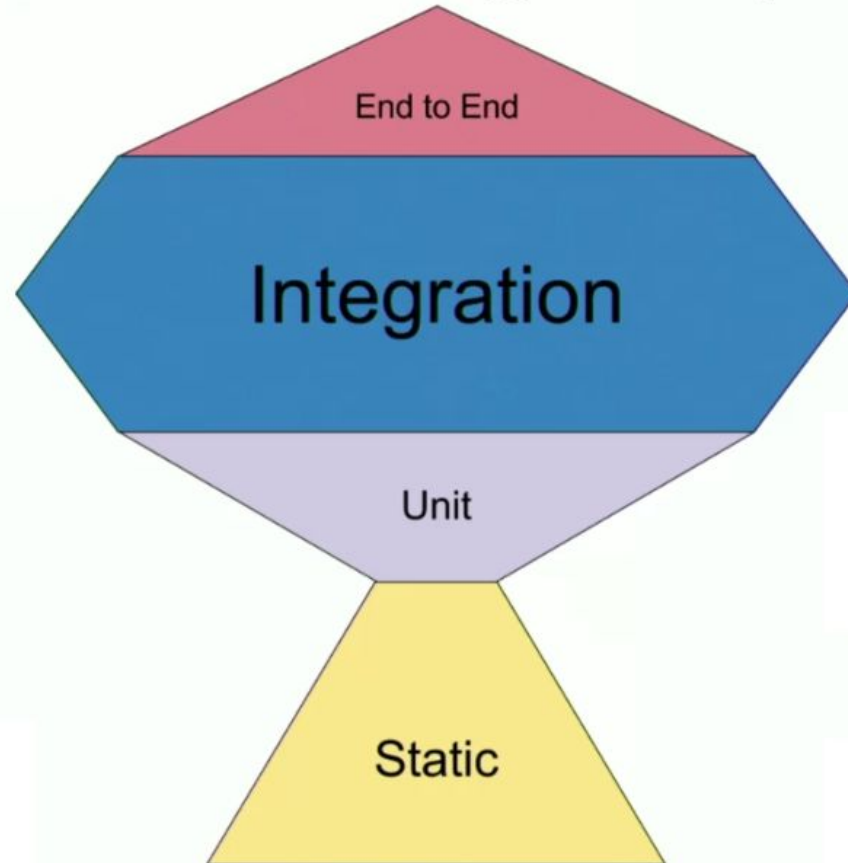
# How do we know what to test?

- I had no substantial experience writing tests
- Kent C. Dodds is considered a thought leader in React and JavaScript testing
  - Avoid testing implementation details
  - Test things that the user is aware of

# Traditional Testing Pyramids



# The Testing Trophy







# Which tools should we use?

- React Native Testing Library
  - Inspired by [React Testing Library](#)
  - Lends itself very well to the KCD methodology
  - Is a wrapper around [React Test Renderer](#), part of the core React Project
    - Is pretty much guaranteed to be maintained alongside React
    - Requires no DOM
    - Installation is a breeze