Front-End Testing

with Enzyme

Roadmap

- Front-end vs. back-end testing issues
- Orientation: the Full(test)stack
  - code ·React · Enzyme ·Webpack · Chai ·
- Details: Testing React Components

How does front-end testing differ from back-end?

Front vs Back

- Node vs. browser — not trivial
- Back end we control the environment; front end may be almost anything
- How do we test processes that extend multiple components?
- How do isolate various elements of the Redux loop?

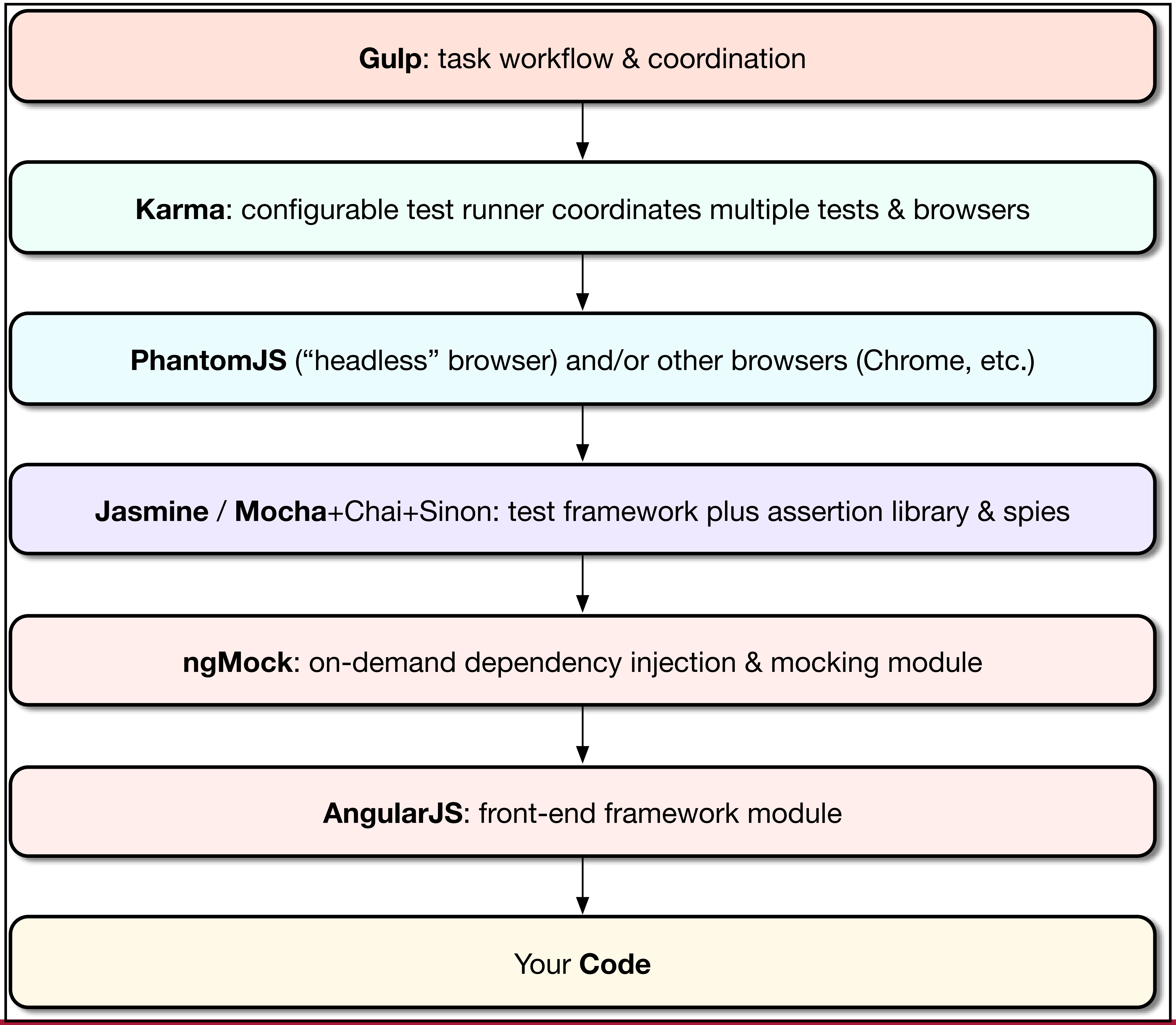
Frontend Testing in Angular

- Large framework stack
- Two-way data binding: testing multiple data flows
- Little isomorphism with back-end testing

Code gets triggered from the top down…

Ye (Olde) Test Stack

…but we'll look at it from the bottom up (increasing coordination / view)



Gulp: task workflow & coordination

Karma: configurable test runner coordinates multiple tests & browsers

PhantomJS ("headless" browser) and/or other browsers (Chrome, etc.)

Jasmine / **Mocha**+Chai+Sinon: test framework plus assertion library & spies

ngMock: on-demand dependency injection & mocking module

AngularJS: front-end framework module

Your **Code**

Enzyme

- Developed by AirBnB
- Flexible React testing framework focused on “shallow rendering” of components.

Why we like it

- Testing Isomorphism:
  - Tests look similar on the front and back ends.
- chai-enzyme for assertions
- sinon for spies

Roadmap

- Front-end vs. back-end testing issues
- Orientation: the Full(test)stack
  - code ·React · Enzyme ·Webpack · Chai ·
- Details: Testing React Components
- Demo

/
chai
Sinon.JS

Testing Frameworks

- **Provide testing code / syntax**
  - Test Driven Development
  - Assertion styles: expect / should / assert
  - Available matchers (e.g. `toBeDefined`, `to.be.ok`, etc.)
- **Functions that group tests & produce passing or failing output**
  - `describe`
  - `beforeEach`
  - `it`
  - etc.

Install all the things

```
#  mocha, chai, sinon
npm install mocha --save-dev
npm install chai sinon --save-dev # must `require` each in tests
# enzyme
npm install enzyme --save-dev
# if using React 15.x
npm i --save-dev react-addons-test-utils
```

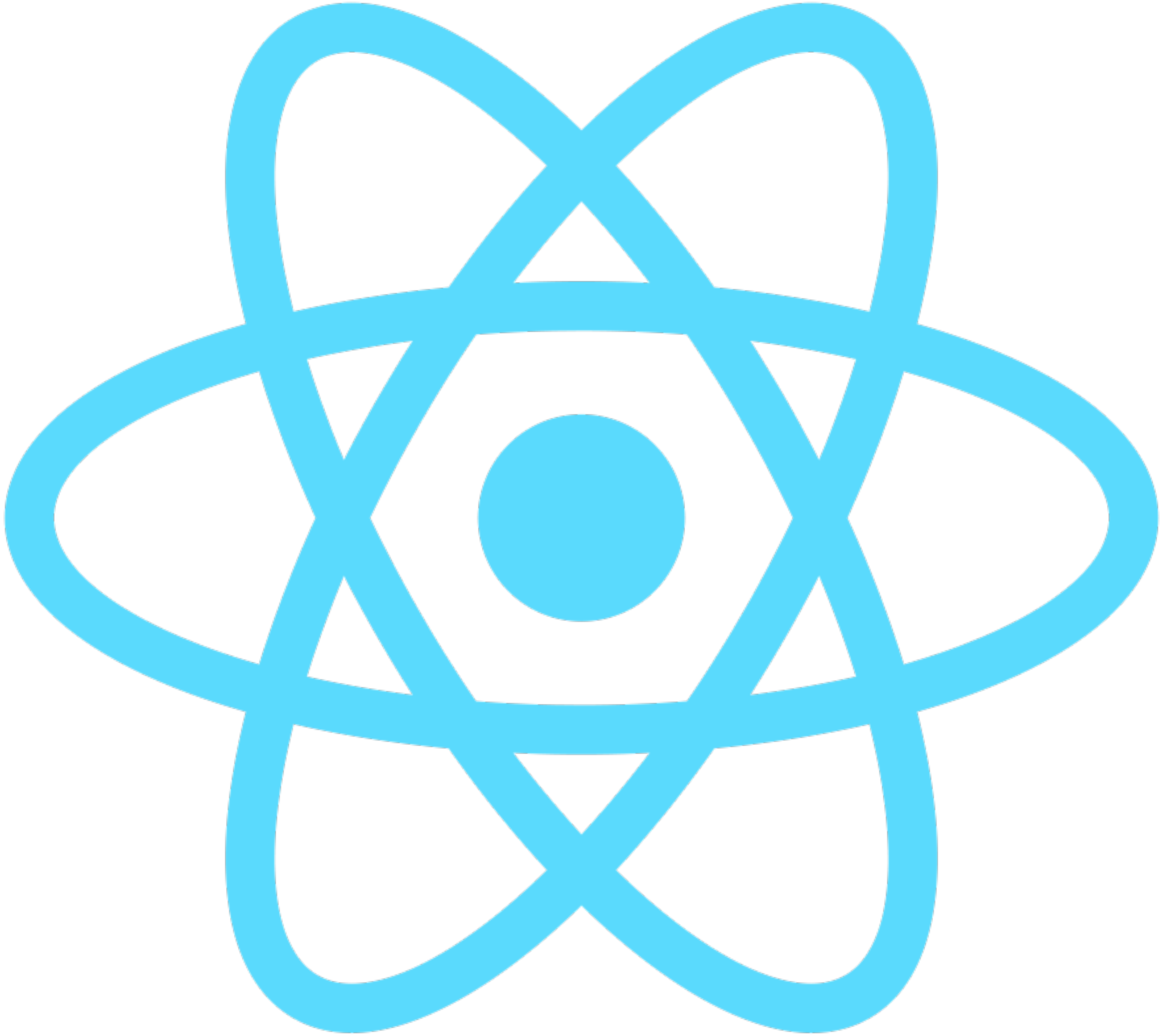
Review of Assertions in Mocha/Chai

```
describe('A suite', function(){ // `describe` & `it` from mocha

  var myString;
  beforeEach(function(){
    myString = 'testing is fun!';
  });

  it('uses Mocha, Chai, and Sinon', function(){
    var spy = sinon.spy(); // sinon spy
    spy();
    expect(spy.called).to.be.ok; // `expect` & matcher from chai
  });

});
```



Roadmap

- **Front-end vs. back-end testing issues**
- **Orientation: the Full(test)stack**
  - code ·React · Enzyme ·Webpack · Chai ·
- **Details: Testing React Components**
- **Resources**
  - Fullstack Academy Capstone Project Guidelines
  - Docs ·Articles

Setup tools & child components via import

```
import React from 'react';
import {expect} from 'chai';
import {shallow} from 'enzyme';
import {spy} from 'sinon';

import ChildComponent from '../../src/components/Child';

describe(‘This component', () => { … })
```

Setup Example: a Component

```
describe('The Kitten Component', () => {

    beforeEach('Create component and onChange spy', () => {
       let clickSpy = spy();
       let kitty = shallow(<Kitten onClick ={clickSpy} />);
      });

    it(‘calls click fn’,()=> {
       kitty.simulate('click', {});
       expect(clickSpy.called).to.be.true;
   });
});
```