



END-TO-END TESTING

With end-to-end testing, you have a script or a helper robot that clicks through your site, just like a user would, to make sure that your site is working correctly.

Resources: **Cypress** - <https://www.cypress.io/>

INTEGRATION TESTING

These tests focus on how your code works together. Does this component talk to this other component the way you'd expect it to?

Resources: **React Testing Library** - <https://testing-library.com/docs/react-testing-library/intro>

UNIT TESTING

Test small, individual isolated pieces of your code.

Resources: **React Testing Library** - <https://testing-library.com/docs/react-testing-library/intro>

STATIC TESTING

These tests are the low hanging fruit. This includes typos and errors.

Resources: **Flow** - <https://flow.org/>
TypeScript - <https://www.typescriptlang.org/>
ES Lint - <https://eslint.org/>
Type Checking with Prop Types - <https://reactjs.org/docs/typechecking-with-proptypes.html>

REGRESSION TESTING

These tests are used when something breaks within your application. A test will be written for the broken state. Then, you'll write code to fix the box, making the test pass.

VISUAL TESTING

With visual testing, a script takes a screenshot and compares it against previous versions to determine whether the result is what you expected.

Resources: **Percy** - <https://percy.io/>
Applitools - <https://applitools.com/>