



React Testing Library

[React Testing Library](#) builds on top of [DOM Testing Library](#) by adding APIs for working with React components.

Projects created with [Create React App](#) have out of the box support for **React Testing Library**. If that is not the case, you can add it via [npm](#) like so:

[npm](#)

Yarn

```
npm install --save-dev @testing-library/react
```

- [React Testing Library on GitHub](#)

The problem

You want to write maintainable tests for your React components. As a part of this goal, you want your tests to avoid including implementation details of your components and rather focus on making your tests give you the confidence for which they are intended. As part of this, you want your testbase to be maintainable in the long run so refactors of your components (changes to implementation but not functionality) don't break your tests and slow you and your team down.

This solution

The [React Testing Library](#) is a very light-weight solution for testing React components. It provides light utility functions on top of [react-dom](#) and [react-dom/test-utils](#), in a way that encourages better testing practices. Its primary guiding

principle is:

The more your tests resemble the way your software is used, the more confidence they can give you.

So rather than dealing with instances of rendered React components, your tests will work with actual DOM nodes. The utilities this library provides facilitate querying the DOM in the same way the user would. Finding form elements by their label text (just like a user would), finding links and buttons from their text (like a user would). It also exposes a recommended way to find elements by a `data-testid` as an "escape hatch" for elements where the text content and label do not make sense or is not practical.

This library encourages your applications to be more accessible and allows you to get your tests closer to using your components the way a user will, which allows your tests to give you more confidence that your application will work when a real user uses it.

This library is a replacement for [Enzyme](#). While you *can* follow these guidelines using Enzyme itself, enforcing this is harder because of all the extra utilities that Enzyme provides (utilities which facilitate testing implementation details). Read more about this in [the FAQ](#).

What this library is not:

1. A test runner or framework
2. Specific to a testing framework (though we recommend Jest as our preference, the library works with any framework. See [Using Without Jest](#))

NOTE: This library is built on top of [DOM Testing Library](#) which is where most of the logic behind the queries is.

Tutorials

Have a look at the "What is React Testing library?" video below for an introduction to the library.



Also, don't miss this [tutorial for React Testing Library](#).

 [Edit this page](#)

Last updated on **Aug 9, 2022** by **Sergio Moreno**