

JavaScript Unit Testing

Sauce Labs offers an [API for running frontend JavaScript unit tests](#), so you can run tests on any combination of browsers and platforms. Not to be confused with functional tests, JavaScript unit tests are triggered by visiting a test page hosted by your servers, [as with this example page](#). By visiting a test page with [every browser provided by Sauce Labs](#), the results can be collated and you can quickly pin-point browser compatibility issues in your application. For general information about JavaScript unit testing, check out online guides such as [this one from Smashing Magazine that provides a brief introduction to JavaScript unit testing](#).

To run your unit tests using Sauce Labs, send the URL of your test page, along with the list of browsers that should run it, through a single call to the [JavaScript Unit Test API](#). After the test completes, another API call will produce the test results.

Sauce Labs visits the test page you specified and parses the test results by accessing a global JavaScript variable on the page. If your test page is on a local network, you can use [Sauce Connect Proxy](#) to create a secure tunnel from your system to the Sauce Labs cloud. The results are displayed on your Sauce Labs dashboard, in addition to being accessible through the [API](#).

Sauce Labs supports unit tests written using the Mocha, Jasmine, Qunit, and YUI test frameworks. We also support a "custom" framework that allows you to provide any sort of test results to Sauce Labs. Topics in the section [Getting Started with JavaScript Unit Testing Example](#) contain more information about how to set up your JavaScript unit test frameworks to work with Sauce.

- [Getting Started with JavaScript Unit Testing Example](#)
- [Setting Up Karma for JavaScript Unit Testing](#)