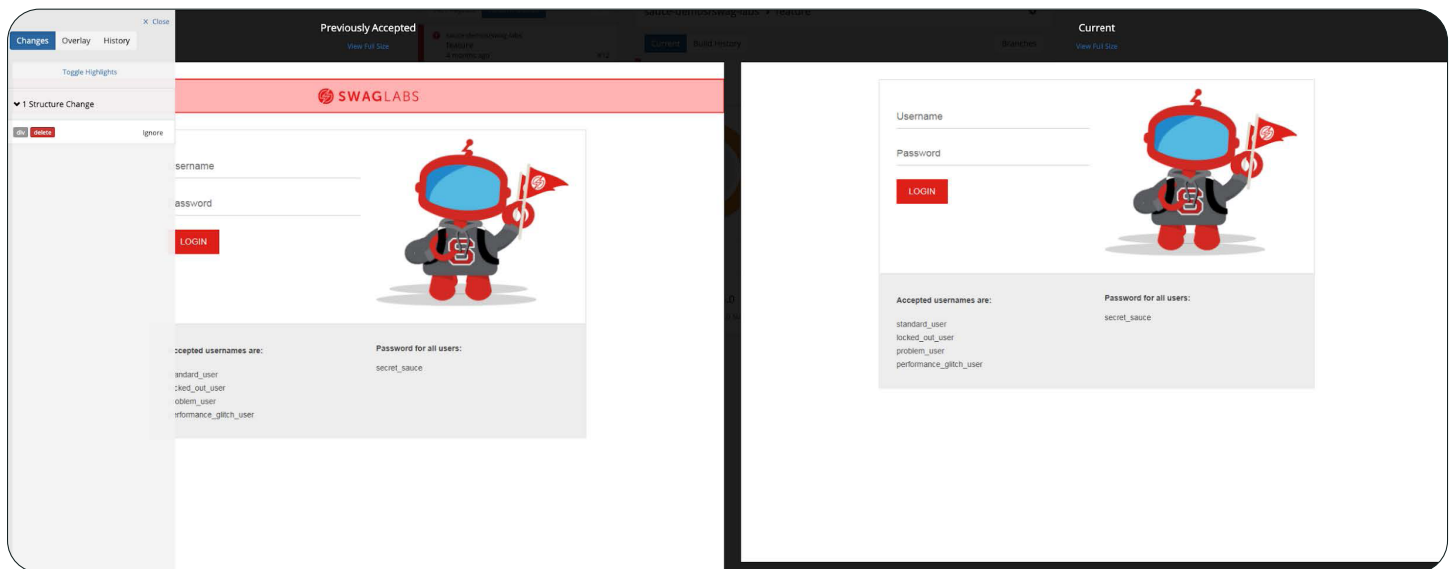


Sauce Visual End-to-End Testing

Sauce Visual End-to-End (E2E) Testing provides you the ability to test the visual aspects of the user interface (UI) across multiple browsers and screen resolutions. It easily integrates into existing WebDriver tests to automate both visual and functional testing, automatically detecting visual and functional regressions.

Visual E2E enables you to see exactly what the UI looks like from the end users point of view. With the ability to automate visual testing and automatically detect visual regressions, you are able to know exactly what changed and where it changed. This enables you to catch and fix these regressions ahead of time, leading to an improvement in visual quality and the customer experience.



Previously accepted version highlighting the diff detected with the expanded change list (left) compared to the current version (right).

Integrate Into Existing Webdriver Scripts

If you have already invested in testing, simply leverage those existing tests to be used for visual testing. Visual E2E seamlessly integrates into existing WebDriver tests with just a few lines of code to run visual tests in conjunction with functional tests, eliminating

the need to create additional automated tests. Use any programming language of your choice that WebDriver supports without needing to install any additional libraries or SDKs.

Key Features And Benefits

- Reduce false positives with Smart Hybrid Diff - Captures both a screenshot (image data) and DOM snapshot (programmatic data) to perform a comparison from two perspectives. With the additional information, a smarter diff is performed to reduce false positives and have a deeper understanding of how the UI is structured.
- More comparison control - Visual changes are automatically categorized into four categories: structure, layout, style, and content. Each category can be enabled or disabled in tests with up to 16 different comparison modes.
- Gain insight into visual diffs - Analyze the root cause of changes by accessing debugging details on visual changes to improve debugging efforts.
- Select what to ignore - Ignore specific visual changes and exclude areas from the test. There are many ways to ignore, including dynamically by category, by selector, or directly in the UI.
- Ensure complete test coverage - Test against browsers on the latest versions and operating systems available in the cloud and make certain the UI is free of visual bugs on Chrome, Firefox, Internet Explorer, Microsoft Edge, and Safari.
- Obtain complete insight into the user experience - Automatically capture full-page screenshots (natively, when possible) to get the whole view of what an end-user would see.

Deploy With Confidence

By performing visual and functional checks of the UI, you are able to increase test coverage and confidence, and at the same time, simplify automation by reducing the need to write and maintain test assertions. This increases your productivity, improves code quality and user experience, reduces cost and time to market, and enables you to ship faster with confidence.