SAUCELABS

WHITE PAPER

Best Practices for Front-End Performance Testing

A PRIMER ON ENSURING FLAWLESS DIGITAL EXPERIENCES

Web and mobile applications have become more sophisticated than ever. But beyond simply having a web or mobile application, users expect a flawless digital experience, and this requirement is growing in importance.

This best practices whitepaper gives an overview of front-end performance testing. From providing a concise definition, to practical tips on how to build a scalable performance testing practice, the following will help teams of all sizes provide better experiences for their users.

TABLE OF CONTENTS

INTRODUCTION

Web and mobile applications have become more sophisticated than ever. End users have become accustomed to consuming services via their devices and browsers, rather than physically visiting the bank, or a retail store or calling a travel agent. But beyond simply having a web or mobile application, users expect a flawless digital experience, and this requirement is growing in importance. This represents a critical juncture for both vendors and users as a functional, well designed, and performant experience often impacts the users' brand selection and more critically, the success of the business.

There are plenty of data that demonstrate the impact of improving front-end performance for business that want to succeed in digital:

- According to Mobify, every 100ms decrease in homepage load speed worked out to a 1.11% increase in session-based conversion, yielding an average annual revenue increase of nearly $380,000. Additionally, a 100ms decrease in checkout page load speed amounted to a 1.55% increase in session-based conversion, which in turn yielded an average annual revenue increase of nearly $530,000.
- DoubleClick found publishers whose sites loaded within five seconds earned up to twice as much ad revenue than sites loading within 19 seconds.
- A tech executive at AutoAnything reported that when their engineering team reduced page load time by half, they saw a boost of 12-13% in sales.

Sauce Labs sees front-end performance testing as an essential part of overall application quality. Front-end performance testing, which is very similar to functional testing, is best done using automated scripts that repeatedly navigate through the relevant pages, measure their performance, and report anomalies back to the tester/developer. This allows for faster release cycles as teams are alerted of performance degradations earlier in the development lifecycle, all while creating a flawless experience for their users.

DRIVERS OF FRONT-END PERFORMANCE

Front-end performance testing is a set of tools and practices that address the following three market trends:

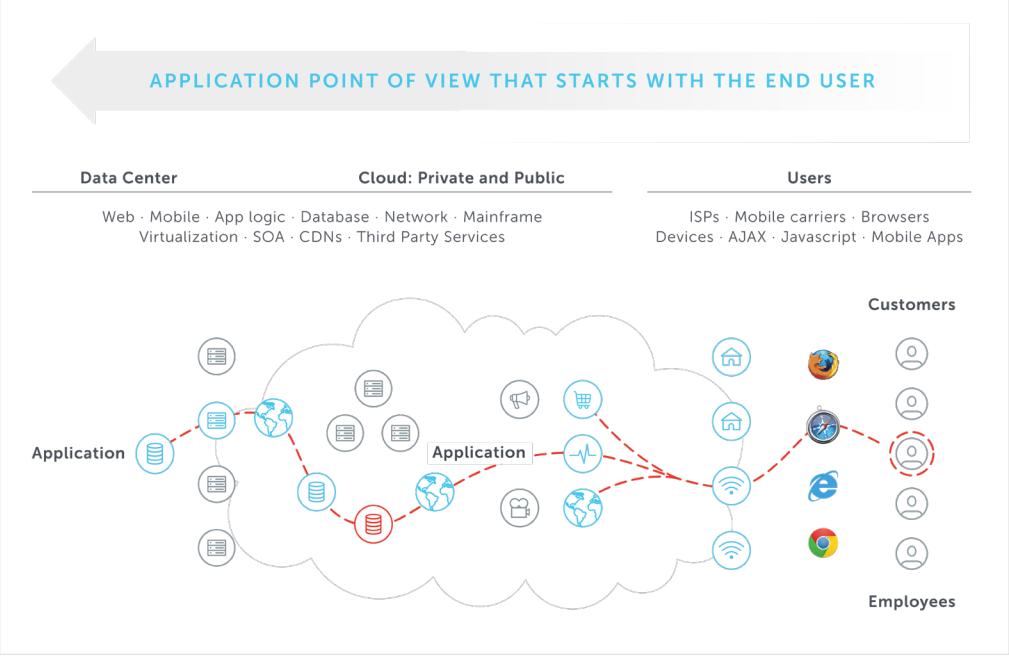
- End-users' expectation for a functional and engaging experience nowadays includes a ***consistently responsive interaction with the application*** that goes beyond perfect functional flow.

- Modern applications leverage the tremendous evolution in ***front-end computational power*** offered on mobile devices and browsers to deliver compelling and rich experiences. These do not exclusively rely on backend resources, so it is important to test these front-end experiences.
- Modern development teams include front-end developers who can use ***early awareness*** of performance degradation to inform development teams before release. This increases developer efficiency and alllows them to respond more quickly to issues, in parallel with early functional testing.

HOW IS FRONT-END PERFORMANCE DIFFERENT FROM LOAD TESTING?

The traditional approach to ensure web experiences remain performant under heavy load is to simulate thousands or millions of virtual users on the service API. This, in return,exercises the delivery chain leading to the service API, including the backend (servers and databases, etc.) and other components. Metrics resulting from this test approach were sufficient indicators to predict the end-user experience during normal and peak utilization events.

However, as mentioned previously, the computational power present on the client-side (front end) changes this balance significantly. A perfectly performing backend and service API architecture will not guarantee good performance from the user experience. You need to account for the behavior of the browser or native application GUI.



From an Agile SDLC perspective, there is an opportunity to conduct front-end performance testing in-cycle, possibly even at commit time. You can conduct this type of testing simply by running an automated test, similar to a functional test, that collects the metrics reflecting page performance, without any special infrastructure.

HOW IS FRONT-END PERFORMANCE DIFFERENT FROM FUNCTIONAL TESTING?

Functional test scripts are meant to validate that a page or function works as intended. These tests would typically navigate to a page, exercise the function being tested, and move on.

Front-end performance test scripts are meant to measure and report the performance of a page or pages. This means the front-end tests need functional transactions for navigation, but functional validation and specific assertions are not required.

DIFFERENT APPROACHES TO FRONT-END PERFORMANCE TESTING

While the definition of performance for web experiences is fairly mature and standardized, there are some variations for web applications.

Hard Refresh Rages: Page Performance

This refers to pages where all interaction with the page components results in a full refresh of the page to a new URL. A user would see the entire new page reload.

Soft Transition/Single Page Application (SPA): Page Performance

This refers to pages with components that, when interacted with, cause soft transitions inside the page that change the page content. As a result, the entire page does not refresh. Only content inside the page changes in the background.

An example page would be https://saucecon.com/.

Jankiness

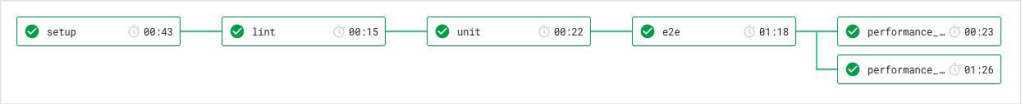
This refers to a lack of smoothness on a site or app, also referred to as stuttering, juddering or halting, that users see when a site or app isn't keeping up with the refresh rate. This occurs when a browser takes too long to compute frames.

Jankiness measurements can be applied to any kind of page, including single page applications and other types.

BUILDING A FRONT-END PERFORMANCE TESTING PRACTICE

#1 Consider Separating Performance and Functional Tests

Because front-end performance tests serve a different objective than functional tests, best practices recommend separating them from GUI tests and making them a separate step in your CI/CD pipeline. In addition, automated tests that are enabled to capture performance run much slower for commands that trigger a page load to ensure that all metrics are captured reliably and accurately. For these reasons, it's beneficial to separate the two.



#2 Choose Metrics That Align With Your Application/User Needs

Front-end performance testing allows you to validate regressions across a wide range of metrics (Performance Score, Time to Interactive, Time to First Meaningful Paint, etc.). We recommend exploring verifications across a mix of metrics that closely align with the most important end-user experiences you wish to achieve. For example, use metrics such as Time to Interactive and Speed Index to validate performance across a single page application. Likewise, if you have an e-commerce app, metrics such as Time to Interactive & Time to First Meaningful Paint provide insight into real user experiences.

#3 Optimize Your Scripts For Performance Testing

While Performance testing is able to validate metrics/regressions around page loads, you should be aware of other actions the test script is taking, such as assertions for the presence of content on the page or validating functionality It's important to determine if these are necessary in gathering front-end performance metrics.

#4 Focus On Core App Experiences First

As you build out your front-end performance test suite, performance testing and validations should focus on core application experiences. In other words, test key pages more frequently, the same way you design your front-end functional tests.

CONCLUSION

Front-end performance testing is an increasingly important piece of a sound continuous testing strategy. As testing continues to shift left, performance metrics can give an accurate view to developers into a user's experience with an application before it's released. And using those metrics as a guide to improving that experience is proven to have a positive impact on app quality, developer productivity, and most importantly revenue.

Teams who have already implemented automated functional testing are well situated to incorporate front-end performance into their pipeline. By modifying existing scripts and separating them from functional tests, developers can run front-end performance tests earlier in the pipeline and be alerted of performance degradations before they are found later in the pipeline, or even worse, in production. When integrated with your larger continuous testing strategy, front-end performance gives teams more ways to ensure that they are delivering quality experiences to customers, without sacrificing release velocity.

Sauce Labs offers a front-end performance testing tool that you can use to obtain earlier feedback on your web application responsiveness during the development cycle. This tool can easily be incorporated into your existing CI/CD workflow within a unified dashboard. To learn more, visit the Sauce Labs website.

SAUCELABS

ABOUT SAUCE LABS

Sauce Labs is the leading provider of continuous testing solutions that deliver digital confidence. The Sauce Labs Continuous Testing Cloud delivers a 360-degree view of a customer's application experience, ensuring that web and mobile applications look, function, and perform exactly as they should on every browser, OS, and device, every single time. Sauce Labs is a privately held company funded by Toba Capital, Salesforce Ventures, Centerview Capital Technology, IVP, Adams Street Partners and Riverwood Capital. For more information, please visit saucelabs.com.

saucelabs.com/signup/trial

FREE TRIAL

SAUCE LABS INC. - HQ 116 New Montgomery Street, 3rd Fl San Francisco, CA 94105 USA