



How to Overcome Web Test Automation Challenges



BY AMIT SIKKA

Web test automation is transforming software development for the better. Automation empowers test teams to easily prepare test data, execute complex test cases across multiple platforms rapidly and accurately, and catch defects early on in the development cycle. Modern-day test automation solutions benefit organizations in multiple ways; when used correctly, such solutions can boost a test team's productivity, enhance software quality and accelerate time to market.

Automation Is Transforming Web Testing

Today, mobile and web-based applications are important revenue streams for organizations. Various business-critical activities revolve around web apps, so flawless functionality and superior user experience are critical. However, testing often takes a backseat to speed and quality suffers. The limited timeframes in agile development coupled with cost factors make it challenging for organizations to sufficiently identify all the underlying issues and ensure optimum release quality.

Such concerns can be adequately addressed by implementing web test automation. Automating web testing ensures better test coverage, improves detection of all major underlying discrepancies and allows faster test execution. Test automation is critical to achieving high-quality web applications; however, testers face many challenges performing automated web testing.

Challenges of Web Test Automation

Web test automation enables testers to test the functionality of web applications thoroughly and quickly, ensuring there are no deviations in the results from the expected user experience and interactions. However, even this beneficial process involves certain challenges for testers, such as:

- **Handling frequent UI changes:** Many components of a web application's UI are prone to frequent changes to logic, or due to new integrations or the addition of new features or

functionalities. As a result, maintenance of UI testing scripts becomes a challenge for testing teams.

- **Multiple error handling:** Error handling can be a cumbersome process with web test automation, especially when test teams must tackle large projects with complex scenarios and large data volume.
- **Image comparisons:** Accurate detection of pixel variations in shape, size, color and formats are an important part of the web testing process. Test automation can help make handling image comparisons more efficient.
- **Long test script execution time:** Typically, web test automation is a time-intensive affair. It requires significant test script execution time, depending upon the number of test cases involved. However, in the fast-paced technology landscape and agile development environment, testers get limited time to ascertain the quality of releases.
- **Complexities in testing various UI elements:** As technology evolves, so do the possibilities for enhancing web apps with new capabilities. Modern web apps have complex features such as embedded frames, flowcharts, diagrams, maps and more. As a result, functional and non-functional web testing becomes more challenging.
- **ROI of web testing automation:** As mentioned earlier, any application's UI undergoes constant changes, and the test scripts have to be adjusted accordingly. This primarily increases the time spent on the creation and maintenance of test automation scripts. As a result, it becomes difficult to calculate ROI while continuously performing web tests.

How to Overcome Challenges

Mentioned below are some measures that can be implemented to make web test automation efficient, impactful and cost effective.

- **Choose a codeless solution:** To maximize benefits, it is essential to choose the right web test automation solution. It is necessary to consider codeless (scriptless) test automation solutions, as these tools ease the process of script maintenance, allowing the testers to save both time and money. Codeless solutions simplify web test automation and improve ROI.
- **Choose common UI elements to uncomplicate UI testing:** Organizations should strive to uncomplicate web testing, as much as possible. Developers can help by using common UI elements, so testers more easily understand them and can strategize efficiently and test vigorously.
- **Leverage an object repository:** The creation of a shared repository enables organizations to reduce the costs and efforts involved with test maintenance. To ensure the success of UI test cases, it is preferable to reduce the UI test cases during the early testing stages and then slowly increase the count as testing moves forward.
- **Adopt organizational-level best practices for automated testing:** For web test automation to be truly successful and deliver sustainable results, it must become a fundamental part of the testing process. Organizations that successfully integrate web test automation have well-established best practices and governance mechanisms. They follow organizational-level best code review processes to ensure the best test automation outcomes.

Web test automation is a critical testing process that allows testers to verify and validate the functionality of a web app's UI and backend features. Successful adoption of web test automation enables QA teams to significantly reduce regression testing efforts and time. However, it's critical to mitigate certain challenges associated with web test automation to truly benefit from such an implementation. Effectively handling these challenges helps teams to deliver a robust and user-friendly web app that delivers superior UI and UX.