Accelerate Software Delivery Cycles with Test Automation

eggplant
Test Automation Software

Software teams are under immense pressure to accelerate secure and stable release cycles for major updates, new features, and maintenance upgrades. As the velocity of change for digital workflows and user journeys increases, businesses must quickly adapt.

To keep pace and reduce the growing backlog, many quality assurance (QA) teams that manually create scripts for functional, regression, and performance testing are turning to test automation to auto-generate test scripts for shorter release cycles.

High-functioning QA teams increase velocity and maintain quality by continuously testing as early and frequently as possible throughout the software development life cycle (SDLC). Auto-generating test cases along the SDLC for exploratory testing shortens development, test, and deployment cycles. It also lets you increase test coverage and the number of releases.

Implementing continuous testing within the CI/CD (continuous integration and continuous delivery) pipeline helps software teams quickly identify defects. They can also notify project teams of the test status for prioritization and resolution.

KEYSIGHT
TECHNOLOGIES

Other benefits include the following:

- improved experience from testing every digital component throughout the user journey
- greater productivity from delivering test results more often via shorter feedback loops
- accelerated time to restore services after an outage from quickly replicated test scenarios
- increased release speed and quality because of early identification of failed test cases
- greater alignment with DevOps principles to enable CI/CD

For continuous testing to deliver these results, QA teams need to adapt both the way they work and the tools they use.

Continuous Testing with Eggplant Digital Automation Intelligence

The visual editor in Keysight's Eggplant Digital Automation Intelligence (DAI) simplifies continuous and automated testing along the SDLC, which is essential for DevOps. The system under test (SUT) appears in a model, a simple visual representation of an application, interface, or website.

A model-based approach lets you widen test coverage by combining linear, directed test automation with automated exploratory testing. This type of testing shifts the focus from basic code compliance to the overall user experience. It has the added benefit of collecting a range of data sets, including real user journeys, testing coverage, and past test failures, to improve the accuracy of test creation.
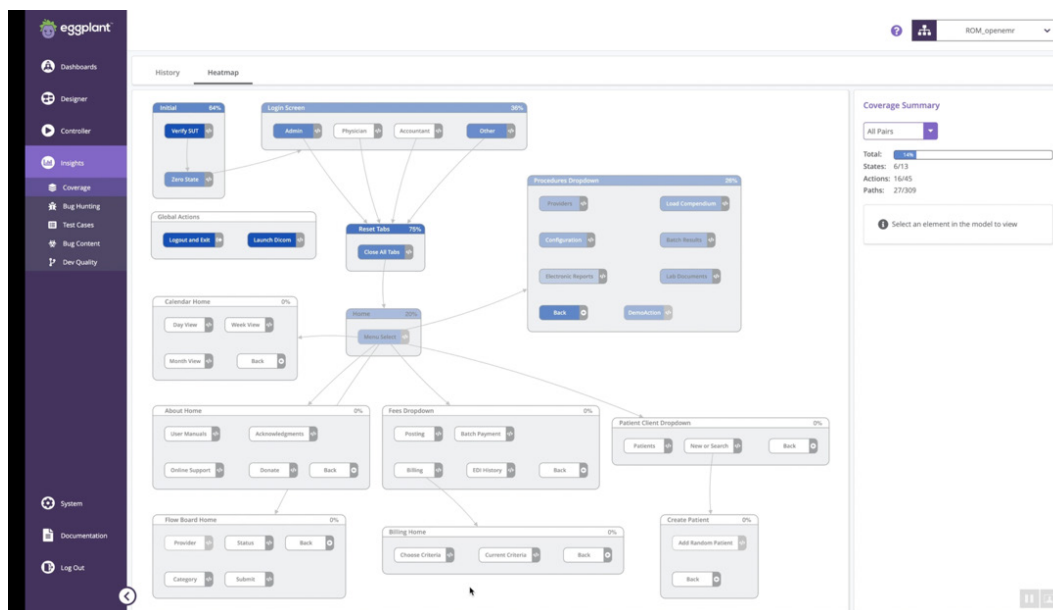

Figure 1. DAI's simple UI makes it easier and faster to run automated exploratory testing for increased test coverage of new or previously untested features and user journeys

Automation efficiency increases further with artificial intelligence (AI)-driven bug hunting and exploratory testing algorithms to auto-generate test scripts and snippets for any SUT. You can rapidly verify user interface (UI) functionality and code quality across the digital footprint of any user journey, including devices, operating systems, and browsers.

Find bugs before they spread

Intelligent automated exploratory testing also expands test coverage incrementally to parts of an application with known issues. The algorithm proactively explores areas close to these defects, as the probability of other bugs is high. DAI's visual editor provides simple visual clues when a bug is discovered, so testers can quickly identify and isolate problem areas.
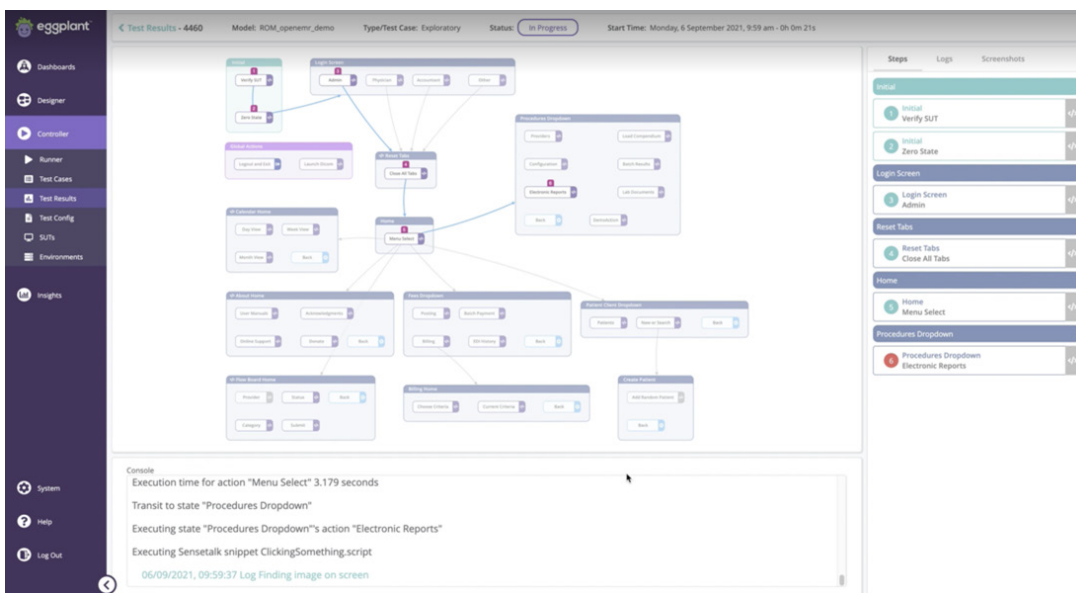

Figure 2. A model and streamlined DAI UI allow testers to identify and isolate problem areas with simple visual clues

Uncovering bugs fast is vital because of the rise of interconnected microservices for cloud-native applications and organizations' ability to customize UI functionality to meet specific business needs. Every update, new feature, or functionality change can introduce unexpected behavior that impacts user workflows, policies, and custom configurations.

Defects are costly to fix, especially when discovered in production. A buggy user experience can cause brand damage, dissatisfied customers, and even compliance violations, all of which can impact an organization's bottom line.

DAI continuously collects test data to overcome these challenges, making it easier to analyze and evaluate release readiness. Test analysis via short feedback loops is essential to improve decision-making, enabling developers to fail early, fix quickly, and move faster.

Eggplant accelerators provide connectivity with Jira to close the communication loop by raising a ticket or case ID. That way, developers can prioritize issues in the next iteration. DAI can also automatically configure each test run with a Slack Webhook to post test details to a chosen Slack channel.
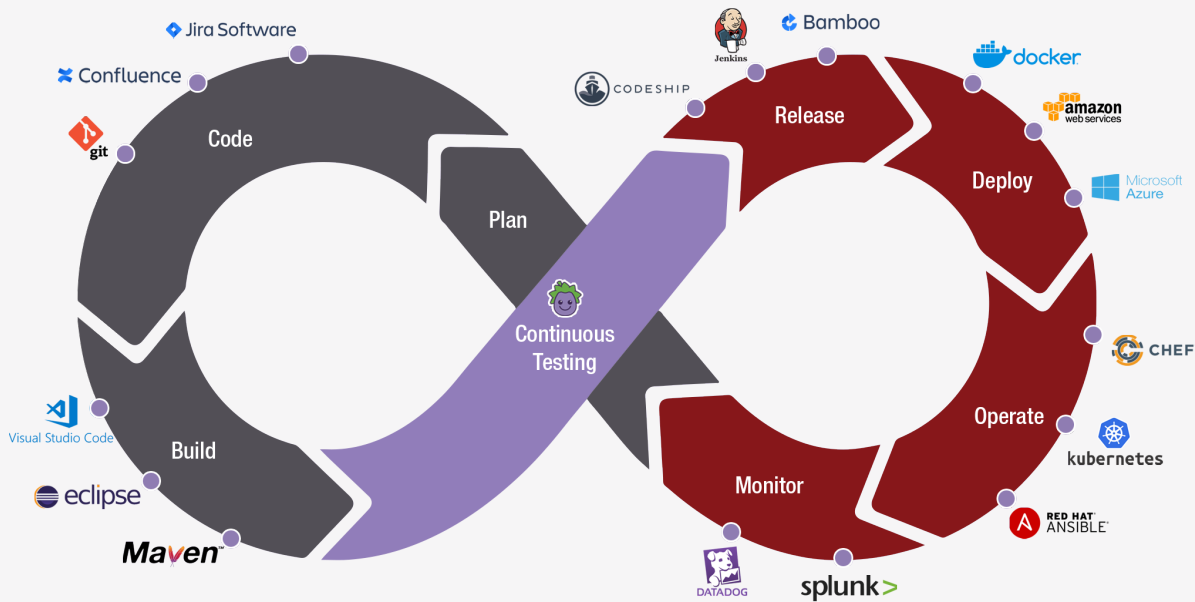

Figure 3. Eggplant DAI is flexible and works with any existing technology to seamlessly integrate with tools essential for a CI/CD pipeline

Summary

Every enterprise wants to increase the speed of delivery and release quality software products that users love. However, testing often slows everything down. Overcoming these QA challenges requires a test automation solution that empowers continuous testing and integrates into CI/CD pipelines.

Driven by AI and machine learning, Keysight's Eggplant DAI seamlessly enables continuous testing to accelerate projects and optimize teams' ability to deliver in a DevOps way of working by doing the following:

- auto -generating test cases for happy paths and complex, nonlinear user journeys, enabling fast and stable releases throughout the CI / CD pipeline
- improving developer productivity with short feedback loops via integration with third-party applications such as Slack and Jira for instant test result notifications

- increasing test coverage and bug-hunting capabilities with AI-driven exploratory testing
- performing model-based testing that replicates user behavior regardless of device, browser, or operating system
- seamlessly integrating with any tool to complement CI / CD or act as a DevOps pipeline orchestrator

To understand how Eggplant DAI can integrate with your DevOps toolchain and supercharge continuous testing for your applications, contact Keysight's Eggplant team to request a demo.

Learn more at: www.keysight.com/find/eggplant

For more information on Keysight Eggplant products and solutions, please contact us. Learn more about Keysight Technologies at www.keysight.com

KEYSIGHT
TECHNOLOGIES