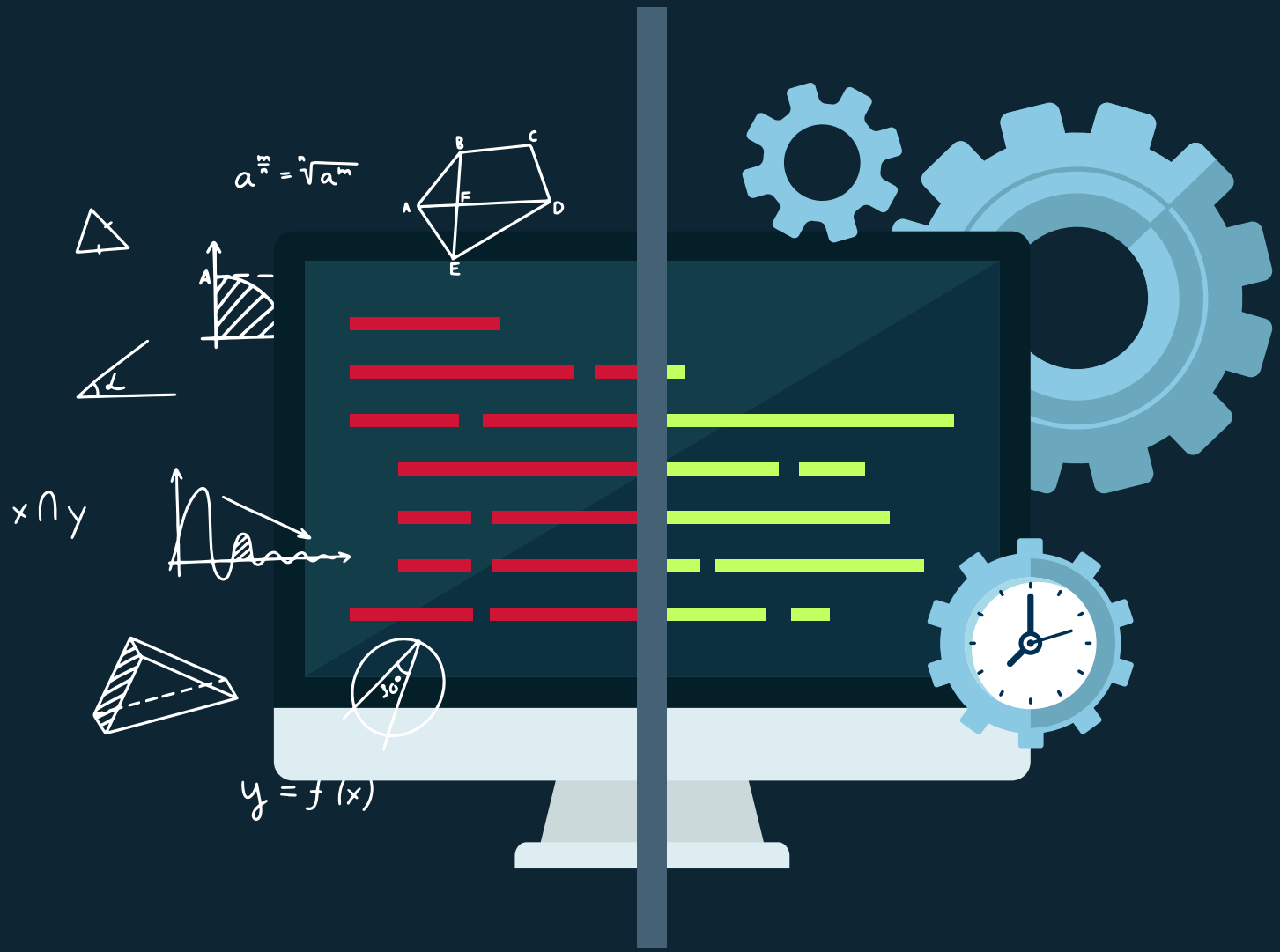
Perfecto
by Perforce

Test Smarter, Not Harder

A Guide to Efficient Testing



Introduction

Countless resources, time, and money are wasted every year on inefficient testing strategies. According to a 2023 report on software testing statistics, only 35% of organizations have fully integrated testing into their DevOps processes and 39% of organizations lack sufficient test coverage. Furthermore, 70% of software defects are a direct result of inefficient testing strategies.

With the onset of Artificial Intelligence and Machine Learning, and with new compliance requirements and software releases hitting the market regularly, it is clear that the software testing landscape continues to evolve. That means testing teams must adopt effective testing strategies to keep pace with these advances and developments.

This guide will delve into key methods of streamlining the overall testing cycle with tips on creating efficiencies. By integrating these core methodologies, testing teams will save time and resources, and be well on their way to developing a winning testing strategy.

Only 35% of organizations have fully integrated testing into their DevOps processes.

Component Level Testing

When developing an efficient testing strategy, testing teams should never think of component testing as an afterthought. Testing should be part of every step of the SDLC, also known as continuous testing. Furthermore, testing should be performed prior to integration, on a component level.

Component level testing refers to the type of testing conducted after unit testing. Also known as module testing, component testing tests certain portions of code or specific functionality or behaviors of an application independently, without the need for integrating the entirety of the code. Component testing comprises several types of tests, including functional, regression, performance, and negative testing.

Below is some additional information on these types of component level tests, and why they should not be overlooked.

FUNCTIONAL TESTING

Functional testing refers to the process of validating the functionality of an application to ensure that the functions match the code specifications. Functional testing works on a "pass/fail" model because the functions either work or they do not. Functional testing is a key aspect of component level testing.

REGRESSION TESTING

Regression testing ensures that code changes that have been introduced do not break any existing application features. Regression testing helps catch bugs early on in the software development process, saving time and resources later in the development cycle.

PERFORMANCE TESTING

Performance testing ensures that performance matches specified criteria. It is a general name for tests that check responsiveness, stability, scalability, reliability, speed, and resource usage of your software application.

NEGATIVE TESTING

Negative testing strengthens the application by subjecting the software to unusual, unexpected, or "negative" conditions.

When developing an efficient testing strategy, testing teams should never think of component testing as an afterthought.

By embracing all forms of component level testing early in the software development life cycle, the more swiftly the app can be fully integrated at the end of the process.

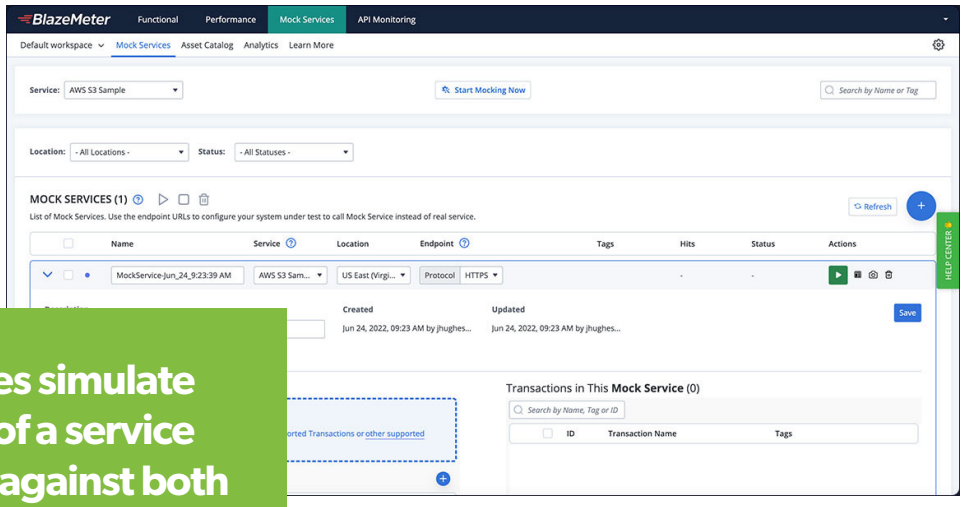
Mock Services for Mobile

Incorporating Mock Services into the testing process allows testing teams to more easily shift left and streamline their testing overall. Traditional Service Virtualization can be complex to set up, time consuming to maintain, and requires dedicated experts, and traditional mocks and stubs do not allow testing teams to create real-world response scenarios.

With BlazeMeter Mock Services, testers can eliminate dependencies to test faster and more completely. Mock Services allow teams to configure and integrate tests so that they never have to be created and managed separately from the test. They can be created on demand in the CI/CD pipeline, and are automated, deployable, and maintainable. Mock Services are simple to create—simply upload your HAR file, Swagger file, WSDL file, or request-response pairs, and create a recording or use the template. The file will be stored in an asset repository to be accessed across your team.

BlazeMeter Mock Services also allow testing teams to virtualize parts of the system that are not under test, or are not available (e.g. still in development). By doing so, testers can get discrete insight into the quality of performance of what they are testing.

Finally, BlazeMeter Mock Services simulate the real-world behavior of a service so you can test your app against both good and unpredictable conditions, such as slow response times, incomplete inputs, unexpected errors, or chaotic behavior. They can mimic any component or service, giving realistic responses so that your testing can continue without delay.



BlazeMeter Mock Services simulate the real-world behavior of a service so you can test your app against both good and unpredictable conditions.

Perfecto Scriptless Mobile

When you incorporate a scriptless program into your testing strategy, you will transform your testing experience with minimal ramp-up time and effort. Scriptless testing also eliminates dependency on coding and frees up time to focus on your business goals and their application. Furthermore, scriptless testing helps teams expand their testing capabilities and increase scalability and collaboration across teams.

Perfecto Scriptless Mobile helps teams automate tests for most complex native mobile use cases, including biometric authentication, image injection, network virtualization, two-factor authentication, and more. In addition, Perfecto Scriptless Mobile significantly speeds up test creation and simplifies the process, enabling even non-technical team members to execute tests.

In fact, Perfecto Scriptless Mobile is so simple, even a kid can use it! Perfecto recently teamed up with Step Ahead Tech, a charity focused on STEM education for children of refugees. In the workshop, a group of kids between the ages of 12-17 learned how to create a scriptless automated test using Perfecto Scriptless Mobile. If children can learn the Scriptless platform and successfully script for themselves within a few hours, anyone at any organization can do the same!

Mobile UX Testing

Usability testing, or UX testing, is one of the most important components in creating a successful app. Users have high expectations for mobile apps and expect them to function perfectly, without delay or interruption. Mobile apps are an integral part of today's world, and the mobile app market is expected to generate more than 613 billion U.S. dollars in revenue in 2025. That means the importance of user experience cannot be overstated.

Retailers during peak traffic times like Black Friday or other seasons of high traffic must place particular importance on mobile UX testing. This is where load testing, combined with mobile UX testing, comes into play. When an app experiences extreme load without prior testing, the user experience will be impacted.

Therefore, it is crucial to incorporate usability testing, including load testing, into your testing strategy in order to provide flawless end-user experiences for your customers.

Mobile Test Data

In order to successfully execute tests, testing teams need high quality test data to properly simulate the functionality of an app to ensure it works properly in the real world. However, obtaining high quality data is not easy. To generate PII-free test data (free from personal identifying information), teams have historically turned to manual test data creation, which is a time-consuming and hard to maintain process. It is difficult to manually generate enough data with the variety needed to execute robust and realistic testing scenarios.

Utilizing synthetic test data, such as BlazeMeter Test Data combined with Perfecto's testing platform, allows testers to ditch manual test creation and generate high quality test data on demand. BlazeMeter Test Data generates synthetic test data for component testing, as well as mock services. This data can even be reused for multiple tests.

Benefits of utilizing synthetic test data include saving time and resources, greater efficiency, test data consistency, elimination of test data dependencies, and better overall results. BlazeMeter Test Data allows teams to generate PII-free data, missing data, data for negative tests, and enables teams to obtain higher quality data that results in better quality outcomes.

Virtual Devices

The need for virtual devices for testing is a widely debated topic in the testing world, and there are many differing opinions as to whether virtual devices — also known as simulators and emulators — are truly necessary. While simulators and emulators are best used during earlier stages of testing, virtual devices have the potential to help teams achieve greater test automation.

By using virtual devices during earlier stages of software development, teams can get faster feedback on issues that may arise.

Virtual devices are used by testing teams to increase testing speeds while lowering the total cost of ownership that comes with real devices. Virtual devices are known for their speed, especially when compared to real devices. By using virtual devices during earlier stages of software development, teams can get faster feedback on issues that may arise.

Another reason why teams should embrace the use of virtual devices is to increase collaboration with developers. As organizations are becoming more agile, testing and dev teams are working more closely together. This collaboration allows organizations to ensure that they are doing comprehensive testing throughout the software development lifecycle.

Artificial Intelligence & Machine Learning

In recent years, and especially in recent months, Artificial Intelligence and Machine Learning have gone from theoretical concepts to legitimate tools that are set to revolutionize the application testing world. AI and ML have the potential to completely change the way teams conduct tests and develop apps, particularly in the following areas:

- **Speed** – AI can go through lines of code in rapid speed, detecting errors and bugs faster and more efficiently. This frees up testers' time to focus on other essential tasks.
- **Automation** – AI does not make human errors and can easily adapt to changes in code. It can also quickly and accurately generate code, data, etc.

With the AI/ML revolution upon us, it is important for testers to look ahead by embracing innovation and staying up to date on technological developments and market trends.

AI and ML have the potential to completely change the way teams conduct tests and develop apps.

Continuous Testing

Perfecto embraces the concept of continuous testing—a methodology that helps identify and address risks at all stages of the development pipeline. Continuous testing ensures that all potential code changes are made as early as possible while identifying and addressing bugs throughout the development process. The goal of continuous testing is to minimize risk and maximize efficiency for testing and dev teams.

Some of the main benefits of continuous testing include:

- **Testing throughout the pipeline** — testers do not have to wait on other departments to complete their portion before testing. Testing happens at all stages of the development pipeline.
- **Expose risks early** — continuous testing exposes risks early on so they can be addressed sooner, rather than later.
- **Receive immediate feedback** — feedback comes quickly and continually, during all stages of the pipeline.

Continuous testing is a part of an efficient testing strategy. This process helps minimize business risk and allows teams to create high-quality apps that succeed in the marketplace.

Bottom Line

When it comes to developing an efficient testing strategy that produces high-quality apps that succeed on the market, incorporating these methods into your strategy will take your testing to the next level. From incorporating scriptless testing and mobile test data to harnessing the power of AI and ML and embracing component testing early on, you will be able to create winning mobile applications faster and more efficiently, while saving time and resources.

Perfecto is continually innovating to remain in step with developments in technology and new releases to ensure a seamless testing experience that stays ahead of market trends. With Perfecto, your testing strategy is simpler, more effective, and the results will speak for themselves.

About Perfecto

Perfecto powers exceptional digital experiences by combining the power of flexible test authoring, cross-platform execution, and AI-driven analytics into one secure, cloud-based web and mobile testing platform.

Visit www.perfecto.io for a free trial, demo, or to learn more.

LEARN MORE