

Mobile Test Automation How can BigR.io Save You Money, Time, and Lost Customers?

It's an unfortunate fact that defective mobile apps have proliferated. You know this is true from your own experience. How many times has a mobile app annoyed, frustrated, even angered you because of problems that were not identified during the development process?

In today's world, in fact, the most effective way to drive consumers away from your brand is with defective apps. A Compuware study found that consumers have little tolerance for defective apps and the brands that produce them. Nearly 90% of apps are downloaded, used just once, and then deleted.

The message is clear that if your app does not work smoothly, it will be discarded. And the consumer will look elsewhere (most likely to your competitor) for satisfaction.

Is Mobile Test Automation the answer? YES

Why are so many mobile apps riddled with defects? Is it because development teams have simply stop testing? Of course not!

The real reason lies rooted in the fact that humans are not perfect, and that the old ways of testing are inadequate for the age of mobile development. QA analysts and testers do their best to catch all defects before releasing apps to the public. But defects still persist, even after the finest manual test techniques and processes have been exercised.

The answer to this problem is test automation. The addition of test automation to the app development process will increase the effectiveness and efficiency of mobile testing.



Consider the manual testing process...

Manual testing is performed by first downloading the app and then installing it on mobile devices. It is followed by step-by-step functional validation for every scenario. Results are compared to the expected behavior, but regression may lag if an area of functionality was deemed not to touch that functional area. Finally, the test results and summary are recorded. Also, manual tests are typically repeated often during the development phase as source code changes are implemented, and to assure performance on a range of different devices.

Consider the automated testing process...

Test automation can perform all of those tasks. Human QA analysts and testers are still needed, but machines can and should take over all of the repetitive tasks. Machines, after all, do not lose concentration or limit themselves to a particular area of functionality unless directed to do so. They don't overlook potential problems because of fatigue or disinterest. For repetitive testing, machines are indisputably more reliable and faster.

Many companies still rely completely on manual testing — most likely because they lack the ability to implement and integrate mobile test automation into their mobile app development process. Yet there are many savvy companies who have found mobile test automation to be a vital component of successful app development projects.

How can you tell these two types of companies — the purely manual testers and the automated testers — apart? Compare the rates of consumer complaints about their mobile apps.



Setting Up Effective Mobile Test Automation

Mobile device usage continues to skyrocket, and mobile app downloads are reaching record-breaking numbers. As the mobile surge continues, many languages and tools have emerged to enable "Mobile App developers" to create more cool apps. These new tools can also help "Mobile Test developers" improve app quality. For now, let's focus on the latter.

Some tools are dedicated only for iOS development, and some just for Android. Some tools are suitable for both: Appium, Calabash, and MonkeyTalk, to name a few. In designing and building an effective test automation process, we should first research the most appropriate tool that best suits our needs and fits into our overall testing framework.



For example, we choose Appium for a recent project. Why Appium?

It's an open source project for cross-platform test automation that's bolstered by a vibrant contributing community. The code is in GitHub, which we can download for free. And there's a huge, energetic community contributing enhancements and bug fixes, and that's also very responsive to inquiries.

Appium supports most any language like Ruby, Java, C#, Php, Python, etc., and any test framework like rspec, cucumber, minitest, etc. Unlike other tools, Appium permits selection of any of the programming languages and test frameworks. Those selected will depend on your level of expertise and preference, as Appium libraries provide binding to different languages.



BigRio's Mobile Automation Roadmap and Approach

As the graphic illustrates, this process involves four phases:

Phase 1: Discovery-Refinement-Setup

Consists of learning the application. Build, Refine, and Prioritize the mobile regression suite. Identify, Evaluate, and Finalize Automation Framework. Identify key team metrics and timelines for phase 2-4 and creating maintenance checkpoints.

- **Phase 2:** Script Development Automate Sanity suite and Priority level 1 test cases.
- **Phase 3:** Script Development Automate Priority level 2 test cases.
- **Phase 4:** Script Development Automate Priority level 3 test cases. Align the framework for continuous integration, deployment, delivery, and improvement.



Benefits of BigR.io's Mobile Automation Approach

• Return on Investment:

Mobile test automation saves time and money. Tests are typically repeated often during development cycles to ensure quality. Each time source code or the operating system is upgraded, the modified software tests should be repeated. Doing so manually is time-consuming and costly. But with test automation, we can run tests over and over again, as needed, at no additional cost — and much faster. To illustrate the speed that can be realized, consider the following example: Typically, 1 manual tester can test 50 scenarios in an 8-hour day. Using a 1000-test-regression-set, the tests can be completed in less than a day. Without automation, it would take days.

Continuous Execution:

Machines never get tired. Automated testing can run all day and all night, 24/7. And you don't have to be on hand to supervise; no matter where you are or what you're doing, your automated testing will be hard at work. You can schedule the time to start the tests, and have the results sent to you after execution.

A REAL PROPERTY OF THE REAL PR

Increased Test Coverage:

Test automation increases the rigor, depth, and scope of testing, which in turn improves software quality. Automated testing can easily accomplish tasks that are difficult for manual testers, such as

validating memory, database, and file contents. Test automation can easily execute thousands of different complex test cases during every test run, providing coverage that is simply impossible with manual testing.

Accuracy and Reliability:

Test automation is far more reliable and accurate when running the boring, repetitive tests which are essential, and which cannot be skipped without putting product quality at risk. Automation also provides consistency by performing the same steps precisely each time they are executed, and by never failing to record detailed results.

Reduced Costs and Improved Quality:

Test automation moves the regression cycle into the development phase, allowing the delivery team to identify bugs/defects far earlier in the development process. And that will translate directly into reduced development costs, reduced manhours, and enhanced product quality.



By implementing mobile test automation, your testers will be freed from the repetitive manual testing that humans perform poorly, and instead, have more time to create new automated software tests and deal with complex features.

As a bonus, you'll be doing your part in helping fight the global proliferation of defective mobile apps. And you will be attracting a very nice set of happy consumers into your customer database.



Benefits of BigR.io's Mobile Automation Approach

• Return on Investment:

Mobile test automation saves time and money. Tests are typically repeated often during development cycles to ensure quality. Each time source code or the operating system is upgraded, the modified software tests should be repeated. Doing so manually is time-consuming and costly. But with test automation, we can run tests over and over again, as needed, at no additional cost — and much faster. To illustrate the speed that can be realized, consider the following example: Typically, 1 manual tester can test 50 scenarios in an 8-hour day. Using a 1000-test-regression-set, the tests can be completed in less than a day. Without automation, it would take days.

• Continuous Execution:

Machines never get tired. Automated testing can run all day and all night, 24/7. And you don't have to be on hand to supervise; no matter where you are or what you're doing, your automated testing will be hard at work. You can schedule the time to start the tests, and have the results sent to you after execution.



Increased Test Coverage:

Test automation increases the rigor, depth, and scope of testing, which in turn improves software quality. Automated testing can easily accomplish tasks that are difficult for manual testers, such as

validating memory, database, and file contents. Test automation can easily execute thousands of different complex test cases during every test run, providing coverage that is simply impossible with manual testing.

Accuracy and Reliability:

Test automation is far more reliable and accurate when running the boring, repetitive tests which are essential, and which cannot be skipped without putting product quality at risk. Automation also provides consistency by performing the same steps precisely each time they are executed, and by never failing to record detailed results.

Reduced Costs and Improved Quality:

Test automation moves the regression cycle into the development phase, allowing the delivery team to identify bugs/defects far earlier in the development process. And that will translate directly into reduced development costs, reduced manhours, and enhanced product quality.



By implementing mobile test automation, your testers will be freed from the repetitive manual testing that humans perform poorly, and instead, have more time to create new automated software tests and deal with complex features.

As a bonus, you'll be doing your part in helping fight the global proliferation of defective mobile apps. And you will be attracting a very nice set of happy consumers into your customer database.