Is your Mobile Test Automation (MTA) really finding bugs?

Overview

Today, mobile applications have become an inevitable part of our daily lives and it has changed the way we do our day-to-day work. One wants to shop cloth, book a cab, order food, read news and so on; there are numerous mobile applications or responsive websites available for everything. Mobile applications are distributed by App stores- Google play app store for android users and Apple's App store for iOS users. And the mobile web applications can be opened in the smartphones using the browsers like chrome, Safari etc.

Recently, **Statista** published that "*52.2 percent of all website traffic worldwide was generated through mobile phones and mobile currently accounts for half of all global web pages served*". In another report, **Statista** also mentioned that "*android users were able to choose between 3.8 million applications and Apple's store remained the second largest app store with 2 million available applications*", at the end of first quarter of 2018. **Statista** also stated that "*21 percent applications downloaded by mobile app users worldwide were only accessed once during the first six months of ownership".*

The reasons for customers to stop using mobile apps or a mobile web application can be its instability, poor user experience, basic functionality failure etc.

With umpteen mobile applications readily available for end users, it has become imperative for businesses to deliver high quality mobile applications so as to sustain in the market. From here comes the essence of mobile application testing. With companies aggressively releasing new features or updates to match up with end user's expectations to remain ahead in the league; **MTA** (Mobile Test Automation) becomes need of the hour. This whitepaper illustrates the means to achieve very motive behind MTA, effectively.

Realistic Expectations from Mobile Test Automation (MTA)

Businesses wants to ensure timely and quality release of mobile applications **every** time. They also aim to meet the requirements of *testing more* and *testing faster* in shorter span of time. However, to test each and every feature of the mobile app/mobile website in squeezed time spans on all possible combinations of mobile devices & versions of operating systems; a big team of manual testers and huge array of physical mobile devices is required which is practically impossible to achieve.

To boost up the testing process, the only way out is running automated tests. Automated mobile test executions in areas like smoke testing, regression testing etc. empowers the manual test team to concentrate on new feature testing. MTA also ensures maximum test coverage over number of combinations of different devices and OS versions, thereby, providing the quality control over the released product. It is also essential to have measurables in place for test automation projects. Measurables helps to determine success of MTA by determining the project's contribution in improving the overall quality of product.

It is challenging but important to set realistic objectives from MTA to avoid burns. Firstly, it is impossible to achieve 100% automation. Secondly, MTA definitely cannot reduce time to execute a particular test case. Automation is no magic and it will do all the activities a manual tester does but in an automated fashion. It actually shortens overall testing timeline by running tests in parallel across vast number of devices and platforms. Last but not the least, the most important fact is that MTA does not have immediate return on Investments (ROI). It usually takes time and depends on multiple number of factors.

Do's and Don'ts while implementing Mobile Test Automation (MTA)

Businesses wants to leverage the benefits of automation to expedite their mobile application testing process but often fail to consider the best practices. Neglecting best practices and making common mistakes can eventually result in failure of MTA. The earlier these are thought through and strategized, the better. This includes understanding do and don'ts, which is just first step in thousand-mile march.

Do's	Don'ts
Do ensure the quality and completeness of manual test cases before initiating MTA	Don't fail to identify and address associated risks
Do have robust and scalable automation framework in place	Don't disregard test data requirements
Do use appropriate tools	Don't cease to strategies MTA execution over manifold hardware configuration, OS and their versions for mobile devices
Do have automated script review process in place	Don't forget need of enhancements and maintenance of automated suits for Frequent feature updates/changes
Do consider the testing on real Mobile devices	Don't limit the automation framework from being DevOps ready

How to measure the effectiveness of Mobile Test Automation (MTA)?

As mentioned before, it is essential to have measurables in place for test automation projects. Measurables helps to determine success of MTA by determining the project's contribution in improving the overall quality of product. Measurables equip managers to optimize MTA to match up with requirements of the organisations. There are humongous factors driving effectiveness of MTA, few of them have been listed below:

1. Cost-effectiveness

   To measure test automation cost-effectiveness, we need to know about the cost of the automation effort. The cost of automation includes the overall cost of the resources and the complete time to automate the tests. One of the major influencers in deciding if automation is required at all is Cost Effectiveness. Cost effectiveness through MTA is not immediate, it is spread over a time period and depends highly on the number of releases/ testing cycles MTA is used. The more the automated suits are run, the earlier we start getting return on the investments.

2. Reliability

   Automated Scripts should be able to give accurate results every time. Reliability of MTA can be measured through the percent of test failed due to error in script, the number of additional iterations required due to script issues and number of the false negatives.

3. Usability

   Automation results should make sense to the manual testers. It should include the reasons why a test failed/passed and should include the run time snapshots in case a test fails. A typical technical error statement in case of failing validation can be disconnecting for a manual tester. Manual tester might not be able to identify the actual cause of failure, mark it as "failed for unknown reason". Usability of MTA can be measured through the survey of manual testers for determining time taken in result analysis, time to find the root cause of the failure, number of false negatives/positives and number of test cases failing for unknown reasons.

   Usability from automation testers point of view can be measured in terms of time taken for a new automation person with similar skill set to understand the framework and become productive.

4. Scalability and Maintainability

   The framework should be hybrid- a mix of modular, data driven, keyword driven framework and should have library architecture which has common functions (reusable codes) stored in a library. This approach makes the automation framework highly maintainable so that whenever there is a change/update in a functionality, only the affected part areas can be fixed leaving the other parts untouched. Such frameworks can be easily scaled up by adding common functions in library or adding keywords to main test scripts, as and when required.

   This can be measured by survey of test automation engineers.

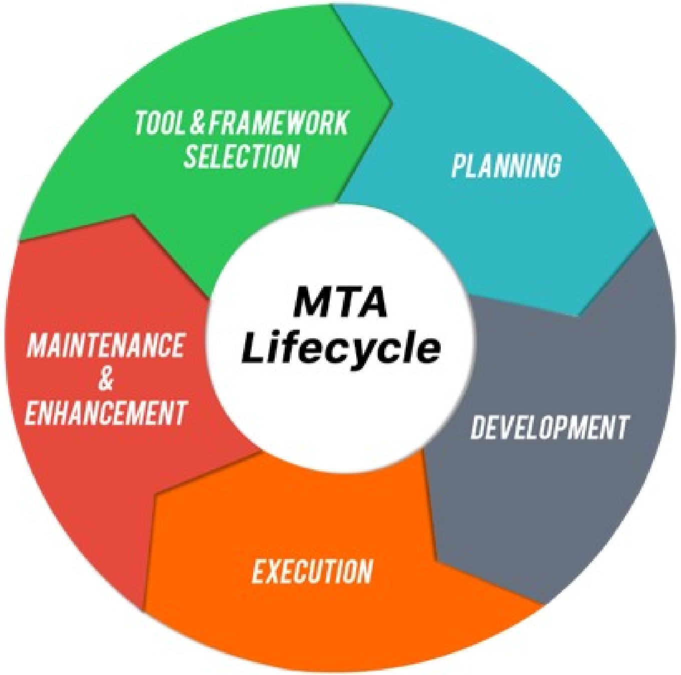
5. Portability

   Testers should be able to execute the scripts in different test environments with minimal changes. This can be measured by calculating the effort required to make that automated suit run in the new test environment or new hardware platform.

Enablers of successful Mobile Test Automation (MTA)

Success of Mobile test automation (MTA) resides in the way MTA is implemented and driven. In the inception, it is important to Identify and understand **MTALC** (Mobile Test Automation Lifecycle). The components of MTALC which are the pillars of implementing MTA successfully are as under:

1. Choosing the right tool and framework



Selection of Mobile Test Automation Tools depends highly on which technology mobile application is built on. It is desired to perform tool feasibility before finalizing the automation tool. The basic features that should be looked upon in the MTA tool are Record and Replay, support for integration of tools for automated execution triggers, automated bug tracking tools (like JIRA, Mantis etc) and capability to execute tests parallelly.

While selecting an automation framework, the aforementioned measurable "Scalability and Maintainability" should be key factors. With regard to the future and latest trend in testing i.e. DevOps; the framework should be DevOps ready from day one. MTA tool and framework should be able to cater test execution needs across multiple mobile devices with different screen sizes, hardware configurations, different OS and their versions. The tool should be able to support automation of mobile websites and hybrid/native mobile application for both android & iOS platform. It is good choose a framework which can be scaled up, to support test automation of API's, websites and desktop applications if required.

2. Careful Planning

Planning is the very foundation of any project. With respect to MTA, plan should include the efforts required for the project, risks foreseen and strategy to execute it over device matrix. There are few pre-requisites listed below that needs to be addressed before initiating MTA

a. <u>Manual Test Cases</u>: It is good practice to assess Manual test cases designed for the mobile application under test beforehand. The assessment is performed to ascertain quality of test cases and coverage of test cases. The steps in manual test cases should be detailed enough for automation person (who may not be the business domain expert) to understand and test cases should have the pre-requisites, test data and expected result specified clearly. The aim is to make the tests process dependent and not person dependent. Each and every detail that is in the head of manual tester should be documented in form of test cases so as to minimize the dependency on manual tester. It is equally important to ensure that the coverage of test cases is maximum which can be achieved by mapping manual test cases against the requirements.

b. <u>Decide what to automate</u>: It is impossible to automate everything and hence the very first step towards implementing MTA is to determine what needs to be automated. The general practice is to automate test cases that- are business critical, are repeatable, can be tested for multiple data sets, can be run on different mobile devices or are time taking.

c. Prioritize Test Cases: It is good to mark test cases on priority basis. This enables to ensure business critical test cases are executed for sure even for the quick releases.

d. Identify Test data sources: Test data is basic requirement for any kind of tests. Consumption of test data in case of MTA is quite higher considering the fact that automated mobile test cases are executed across voluminous number of devices. It is a good idea to identify in advance the various sources from where the required test data can be generated and if possible automate the process.

e. Picking right time to introduce MTA: It is essential to pick correct time to introduce MTA which is supposedly when the mobile application under test is stable i.e. the basic & business functionalities are working properly, manual test cases are in place and there is no near roadmap to revamp/modify the entire functionality.

3. Script Development

Automation engineers should follow industry standards while automating tests like following modular approach, creating reusable components, keeping test data and object repository out of the actual test cases, using variable naming convention, incorporating comments about the functionality the script addresses, ensuring the validations are reported properly etc. This provided ease to maintain and enhance the scripts in longer run. Further, the automated scripts should be mapped to manual test cases to ensure visibility and traceability.

4. Execution

For obvious reason automated tests cannot be executed over all the devices available in market but also cannot be limited to few mobile devices. Hence, it is practical to prepare a device matric addressing the devices, OS & its version and hardware configurations we are aiming to test. Now, depending on the matrix, the organisation can take a call whether to have physical device lab available for the testers which has its own pro’s & con’s, or they wish to leverage the cloud mobile device labs. Cloud mobile device labs enable users to perform tests from any location on real devices placed in their labs. They offer availability of tremendous no. of devices to choose from and have API’s which can be integrated to the automations frameworks, hence enabling, automation scripts to be executed on clod mobile devices directly.

5. Continuous Maintenance and Enhancements

There are multiple factors, mentioned before, that leads to the success of MTA. However, the key to sustain that success is maintaining and enhancing the automated scripts as and when required. The effectiveness and benefits of the MTA fades with time if the automated suit is not maintained.

Conclusion: Key points to ensure “MTA is actually finding bugs”

It is a common perception that MTA should find more bugs or MTA should improve the quality of the mobile application. However, Businesses should understand that automation is only a means for executing the tests. In general, mobile test automation is aimed to perform regression testing, to ensure that the older functionalities are still working with new features/updates. There are multiple factors, mentioned below, that really helps ensuring that your mobile test automation is actually capable to find bugs:

1. Quality of Manual Test Cases should be good
2. Test Coverage should be maximum
3. Automation Scripts should be robust i.e. should not fail for errors in the automated scripts
4. Automation results should be descriptive so that the end user is able to actually able to analyse them conveniently
5. False Negatives and False Positives in automation results should be nil

The CresTech Edge

Businesses often wish to leverage the benefits of automation but lacks the team that has hard core technical skills required for test automation. The respite is that there are service providers like CresTech with whom Businesses can partner with. CresTech not only brings the technical skill set to the table but also brings along their years of experience in handling similar projects. CresTech actually understand the complete testing cycle a product goes through and customize their solution around the product requirement.

CresTech is a market leader in providing Software Quality Management Solutions & Services. CresTech's solutions & services have helped organizations meet their project time lines, budget and quality goals. With a commitment to offer the best and experience in delivering quality solutions & services across industries, the company has 250+ test specialists with global delivery centres across Noida, Bangalore and USA. What makes us different are the factors: Core Expertise, Futuristic Vision and our Core Values.

Author

Archana Mehta
Solution Architect
CresTech Software Systems

Archana Mehta has been with CresTech since 2013. Her core competencies include Test Automation Framework design and development. She has played different roles in QA industry and is currently responsible for understanding & analysing customer requirements, designing solution and consulting. She can be reached at archana.k@crestechglobal.com

References

https://www.statista.com/statistics/276623/number-of-applications-available-in-leading-app-stores/
https://www.statista.com/statistics/241462/global-mobile-phone-website-traffic-share/
https://www.statista.com/statistics/271628/percentage-of-applications-used-once-in-the-us/
https://www.3pillarglobal.com/insights/mobile-automation-testing-tools-frameworks-cloud-based-solutions
https://www.testing-whiz.com/blog/top-5-emerging-mobile-app-testing-trends-to-watch-out-for-this-year
http://www.softwaretestinggenius.com/measurement-of-quality-of-the-testing-and-test-automation-process