Quality Factors of Mobile Automation Tools and Its Limitations

Sreenivasa Murthy Vankamamidi[1], Dr. Sudhamani[2]
[1]Research Scholar, [2]Research Supervisor
Rayalaseema University

Abstract - Mobile/ Mobile applications must be tried to guarantee they keep running on key platforms and over a large number of systems. Regardless of the deadlines of short Mobile development cycles, it is important to quality-test applications crosswise over operating systems, gadget platforms and systems so as to guarantee long tern achievement in an exceptionally divided and aggressive worldwide market. In addition, nonfunctional testing — including ease of use, security and versatility — is as basic as useful testing. Efficient testing can help gadget makers and software developers gather proper measurements that improve application quality.

This Author here exhibits different difficulties of Mobile application testing and furthermore investigates industry best practices for Mobile testing and shows viable approaches to oversee Mobile application quality through test Automation. Similarly, as with ordinary applications, there is an expanded need to Automate Mobile applications testing to improve scale and efficacy. A well-organized automation methodology empowers equipment producers and programming designers to lessen their endeavors and quicken time to showcase.

A comparable analysis can be made by considering testing tools for other Mobile OS like android, ios, windows mobile. A near analysis should likewise be possible on nature of applications of various Mobile operating systems dependent on automated testing devices of every platform.

Keywords - Mobile application testing, Automation tools, Software testing.

I. INTRODUCTION

The Mobile phone has progressed from its conventional job as a communication medium to that of a basic multipurpose personal device. In the interim, technological progressions and the multiplication of gadgets across OS and platforms (Apple iOS, Android and Windows Mobile) have made it additionally challenging for manufacturers of Hardwares and application designers to create and deliver new product items.

Software Application testing empowers the product analyzers to find errors in the product and eliminate them to accomplish improved programming quality. Now a days programming testing turned out to be wide-spread and basic among software development organizations. Programming testing can be performed thru Manual or Automation. Manual testing is to write the Test cases and perform test execution without utilizing any automated software Tool. In manual testing an analyzer plays out the testing through cautiously exploring through the various interfaces of the framework under test, testing with various estimations of sources of info, recording and contrasting the observed outcomes and the normal consequences of the tests. Computerized testing is finished with the assistance of an Automated testing device. The Automated testing Tool gives a system-controlled testing instead of Manual testing.

Programming testing tool empowers the product analyzers to find defects in the product and evacuate them to at last accomplish improved programming quality. As of late programming testing turned out to be wide-spread and basic among programming improvement organizations. Programming testing can be performed either manual or through Automation. Manual testing is to physically write the Test cases and execute them without utilizing any Tool. In manual testing an analyzer plays out the testing through cautiously exploring through the various interfaces of the framework under test, testing with various estimations of sources of info, recording and contrasting the observed outcomes and the normal consequences of the tests. Computerized testing is finished with the assistance of a automated testing device. The Automated testing tool gives a System controlled testing.

The testing device executes the experiments to test the presentation and usefulness of the product under test. The point of automated testing is to lessen the required human exertion as in manual testing however it doesn't expel the need of manual testing totally. Mobileplatforms are being received overall as a result of an assortment of programming being offered to clients in those handheld and compact gadgets. Testing is being utilized as a quality confirmation strategy for Mobile applications as well. A few tools are proposed and actualized for this reason. These tools have just been assessed and looked at for their specific highlights, upheld platforms, code inclusion, and effectiveness. Nonetheless, existing computerized testing devices of Mobile applications have not been assessed and analyzed for various quality traits they can upgrade in applications under test. Thusly, two research goals are planned for this analysis that is:

1. To assess diverse testing tools of Mobile Applications concentrating on recognizing quality variables they help to accomplish in the applications under test;
2. To measure in general patterns of fundamental quality elements accomplished in the Mobile applications under test utilizing Automated testing devices.

In this paper, we have assessed and thought about Automated testing tools for including or upgrading significant quality factors in mobile applications under test. The discoveries and aftereffect of this analysis are advantageous to the professionals just as the specialists. The rundown of value components to be accomplished fluctuates among applications. The testing of various applications requires choice of various devices. Accordingly, the professionals may need to search up for devices which help them to guarantee the ideal quality factors in a specific App under Test (AUT). The analysts who are keen on proposing the Tools and methods for testing of Mobile applications may need to consider the quality components featured in this analysis. In addition, they can start their very own exploration think about based on these devices to propose consolidated, updated and improved answers for accomplishing the most extreme number of value qualities in the AUT.

II. BACKGROUND

As indicated by a review 55% of clients won't utilize an application again subsequent to experiencing a noteworthy issue. If the application has bugs and issues and on the off chance that it regularly crashes, you will get poor evaluations and irritating remarks by the client which will thus prompt less downloads that will result in less incomes. The best way to accomplish great evaluations and positive surveys is to test your application productively. The nature of an application is much more significant than its usefulness and its plan. With a perspective on keeping the delicate substance of the application shielded from unapproved assault you have to do security test before the application is utilized. Likewise, an application must be tried for its usefulness.

There are two sorts of testing accessible

A. Manual testing: It includes a great deal of endeavors as it worked by people manually to discover defects utilizing a test plan which portrays an orderly and itemized way to deal with test programming. It expends additional time and isn't reasonable for enormous tasks. It is estimated as person hours every month. The procedure of manual testing turns out to be an excessive amount of time-taking as it requires every one of the exercises to be performed physically. However, manual testing is preferred in the event of some unpredictable frameworks where a couple of basic deformities must be found while testing physically. Amid manual testing the analyzer interacts with the framework under test as the end client of that product would, and guarantees the adequacy of the framework by exploring through the product

Manual testing has the accompanying downsides:

- Time-taking
- Requires more analyzers
- Less exact outcomes
- Not 100% exact because of human blunder
- Requires HR to perform
- Test cases are dreary to perform
- Testing different highlights in parallel, unrealistic
- Lack of reusability of tests
- Lack of test culmination.

B. Automated testing: The endeavors in manual testing can be decreased via automated testing. Here devices execute a pre-defined scripted test on programming to identify bugs and defects It is a powerful and effective strategy as it spares time and cash. It is most reasonable in condition where the requirements are dynamic and regression test is required to perform iteratively. There are many tools accessible to meet the requirements.

The primary goal of automating programming testing is to decrease the testing exertion, time and cost. Testing automated results in improved effectiveness, though decrease in human contribution in testing process. Automated testing facilitates the reusability of test scripts, utilizing the testing device, for various overhauls of the framework under test. Automated programming testing rearranges the testing procedure and results in low maintenance cost of the product. Automated testing has the accompanying advantages:

- Simplified relapse testing
- Scalable for bigger Mobile applications
- More cost-effective for bigger Mobile applications, after some time
- Able to run different tests all the while
- Performs dull tests that are requesting for manual analyzers
- Performance testing is conceivable because of synchronous testing.

Automated testing has the below disadvantages:

- Slow and inefficient for a basic component, little scale Mobile application
- Unable to test client experience factors
- Code and arrangements are normally explicit to every Mobile application
- All regions can't be automated
- Manual testing can't be completely disposed of

Once more, creating automated test scripts is time-taking, and it's surely unrealistic to automate each test situation. Hence, it's imperative to be particular about what test scenarios are to be automated. For instance, automated testing works best for redundant scenarios and huge datasets. Basically, on the off chance that you do choose to utilize computerized testing, you need to utilize the technique to test the center usefulness of the Mobile application and make more opportunity for manual investigation of client experience and key highlights.

III. KEY CHALLENGES OF MOBILE APPLICATION TESTING

With a rising innovation creating and developing actual application can possess various remarkable challenges. Maybe a couple are depicted beneath:

A. Device Fragmentation - Mobile device categorization is a phenomenon that happens when some mobile clients are running older OS version, while latest versions are accessible. There are distinctive mobile OS accessible. Major ones are Android, iOS, and Windows Phone. When building up the test, you will run over differences in the manner the application performs between various platforms. Utilizing a structure that supports numerous articles can help since it empowers to seclude the usefulness of aitem and decide if it should be adjusted for different platforms or not. For example, your application may have a choice menu that necessities to present as a looking over rundown for Android and a radio-catch determination list for Windows Phone. With a testing arrangement that underpins different articles, one can without much of a stretch test both the situations.

B. System Diversity - Apart from the H/W and S/W issues, the behavior and performance of bearer's system influences the usefulness of the application. The application ought to have the option to work in 3G, 4G or 5G arrange, low signal quality and varying wifi speeds. A few applications are required to work the same with no / disconnected network condition

C. Determination of right Automated Tool - As we probably are aware one size doesn't fit all. Even though there are many varieties of tools accessible for Mobile testing choosing the correct device might be risky. Tool has restrictions and should redo the business need. For example, some open source Tool has impediments like no picture correlation, moderate content analysis for the iOS platform and so on.

D. Need of Specialized Skills - The tester ought to be sufficiently gifted to test the application with his testing assets. An analyzer must go past the customary job and work together with the promoting group and business expert to figure out what parts of utilization require unique consideration and test in like manner.

E. Continuous Testing - Continuous testing is an approach to execute test as a piece of programming delivery pipeline to get a prompt input. It redefines your application and includes a ton of new forms. When you include a change, you have to perform regression testing from the earliest starting point to guarantee that the application is compatible.

IV. METHODOLOGY

For Mobile applications, there are 14 basic programming quality factors, these variables are the most noteworthy quality traits in programming and online applications, yet in addition the Mobile applications must comply with these quality prerequisites. Additionally, for each device, the deduction of the quality variables is likewise legitimized dependent on its highlights and attributes. The summarized results of this study are represented graphically to demonstrate a general pattern of value factors accomplished utilizing automated testing.

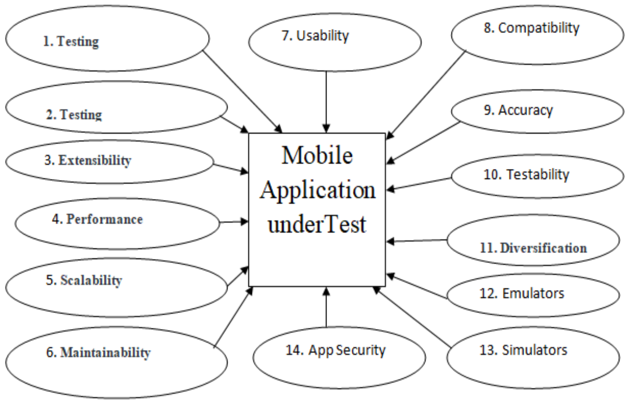

Figure 1: Quality Factors considered for comparative study for various automation tools.

Mobile applications are irreplaceable, as each end-user, purchaser and end-client for a business needs access to your application/site over a compact device. It is the underlying essence of your organization/business, which needs to have an effect. This makes Mobile application testing approach business basic, where the application is tried for openness, security, accessibility, and required usefulness.

V. AUTOMATED TESTING TOOLS FOR MOBILE APPLICATION

Here is the list of top Mobile App Testing Tools-

There are different types of computerized tools present in the product market to test your application. There are multiple Mobile Automation Testing Tools accessible. Some of them are free while some are costly. A portion of these Automated Tools were made some time in the past; while some have recently made it into the market. Each Tools is special and has certain qualities.

A. Kobiton Tool - Kobiton is a cloud platform for executing computerized and manual Mobile and web tests. Kobiton underpins running automated tests with Selenium WebDriver (for web applications) and Appium (for local and Mobile web applications).

You can test your sites with your preferred language – no compelling reason to learn explicit scripting dialects or gain proficiency with another programming language. What's more, you likewise don't have to introduce testing SDKs and even the Appium restricting can be a torment point for test and dev group. When you run Appium in the cloud, there is no compelling reason to introduce Appium and set up your gadgets. You'll approach test on several programs in a flash.

Advantages:

- The most recent genuine, cloud-based gadgets and setups
- Centralized testing history and information logs for expanded joint effort
- Internal Device Lab Management to all the more viably uses inner gadgets
- Support for Appium 1.6.4
- Simplified client experience to streamline test sessions
- Squish ByFrogLogic

Understand your Mobile test automation challenges with Squish for iOS and Squish for Android. Squish highlights committed help for computerized testing of local Mobile Apps, Mobile Web Apps just as a blend of both.

Because of Squish's extraordinary and stable article recognizable proof techniques, Squish tests can keep running on cell phone emulators and distinctive genuine gadgets with no changes. Not at all like numerous other test tools, Squish does not expect you to escape or root the gadget.

Advantages:

- Advanced signal help
- Ready for Testing in the Cloud
- Support for installed web content
- CI and source control combination
- End-to-End and IoT testing

B. Experitest Tool - This Tool is the main Mobile application testing platform.

- Create and execute several manual or automated tests in parallel on 1,000+ genuine IOS and Android gadgets in the cloud
- Create Appium tests easily legitimately from your IDE
- Enjoy ongoing collaboration and live troubleshooting
- Accelerate Mobile testing cycles and increment the nature of your discharges
- Integrate with your CI/CD Tools and work process
- Access visual test reports and progressed analysis

C. KMAX - Use KMAX to test your Mobile application or gadget to ensure it performs under unfriendly system conditions, extending from the daily practice to the outrageous. KMAX gives pre-characterized arrange situations, including 3G/4G, LTE, low-earth circle satellite, and then some. Imitate unfriendly system conditions in your lab including bundle misfortune, delay, duplication, debasement, and that's only the tip of the iceberg. Accurately reproduce explicit situations.

Advantages:

- Choose from 22+ predefined Mobile system situations and begin testing right away
- Subject an item/application to controlled, adaptable, and repeatable tests
- Incorporate genuine or reenacted organize traffic
- Customize arrange disabilities to precisely coordinate your ideal condition
- Accelerate time to showcase by disposing of mystery and amazements amid testing

D. Appium Tool - Appium is an open source, and a cross platform test automation Tool for the half and half and local iOS, it provides Android adaptations from 2.3 onwards. Appium works like a server running out of sight like selenium server. It underpins many programming dialects, for example, Java, Ruby, C# and other which are in the WebDriver library. Appium uses WebDriver interface for tests running

Appium computerizes Android utilizing the UIAutomator library, which is given by Google as a feature of the Android SDK. On cell phones, it can control Safari and Chrome. It tends to be synchronized with testing structure TestNG. For this situation, UI Automator can deliver instructive and nitty gritty reports, like reports produced by Ranorex

Appium Benefits

- Due to the utilization of standard mechanization APIs on all platforms, you don't need to adjust or recompile your application in any capacity
- You can utilize any web-driver good language (Java, Objective-C, JavaScript) to compose experiment
- You can utilize any testing system
- Easy to setup on an alternate platform
- Supports different dialects like Ruby, Java, PHP, Node, Python
- It does not expect anything to be introduced on the gadget
- You can at present use Selenium Webdriver JSON wire convention
- You don't need to re-arrange Mobile application on an alternate platform
- With the assistance of Java, it very well may be incorporated with different tools

E. Robotium - It is a free Android UI testing Tool, with Robotium it is anything but difficult to compose ground-breaking programmed discovery experiments for Android applications. In this way, there is no requirement for additional data about the Android application's structure or actualized classes. All they require is the name of the primary class and the way that connects to it. It underpins Android 1.6 adaptation or more. Tests in Robotium are written in Java. Truth be told, Robotium is a library for unit tests.

In any case, it requires a ton of exertion and time to get ready tests by methods for Robotium, as one must work with the program source code so as to mechanize tests. The Tool is less appropriate for association with framework programming; it can't bolt and open a cell phone or a tablet. There is no play or record work in Robotium, and it doesn't give screen captures.

Advantages of Robotium

- With least information of task ground-breaking experiments can be made
- It handles different Android exercises naturally
- It needs negligible time to make strong experiments
- Synchronize effectively with Ant or Maven to run tests as a feature of consistent mix
- It is conceivable to run experiments on applications that are pre-introduced
- It can get code inclusion for Robotium tests

F. Selendroid - Selendroid is a test automation structure that drives off the UI of Android local and cross breed (applications) and the Mobile web. Utilizing the Selenium 2 customer API tests are composed.

Advantages of Selendroid

- It is completely good with JSON wire convention
- No modification of application under test is expected to mechanize it
- Same idea for computerizing local or half and half applications
- By distinctive locator types, UI components can be found
- It can cooperate with various Android gadgets in the meantime
- Selendroid provides hot stopping of mobile devices
- By diverse locator types, UI components can be found
- Selendroid accompanies a helpful Tool known as Selenium Inspector. It enables you to examine the present condition of your application's UI.

VI. COMPARATIVE ANALYSIS OF SOFTWARE TESTING TOOLS

The reason for testing is to guarantee that product meets its functional requisites and it is of required or standard quality with the goal that it is acknowledged and adopted by the client for its planned use. Previously mentioned devices are capable in at least one from functional testing, system testing, framework testing, code coverage and UI testing, and so forth of Mobile applications. This segment exhibits their near investigation based on quality variables from examined in this paper they test and subsequently improve in Mobile applications under test

In the event that an application is independent of Platform or cross-platform, it implies it is appropriate for a various use on various operating systems. It is an or more point to check quality factor of platform compatibility. Multiple automation devices fill this need to perform tests for Mobile application's similarity with iOS and Android by offering cross platform testing.

VII. FINDINGS AND DISCUSSIONS

Table 1: Critical Findings

Quality Factors	KOBITON	Froglogic	Experitest	KMAX	Appium	Robotium
Testing Environment	□	□	□	□	□	□
Flexibility	□	□	□	□	□	□
Extensibility	□	□	□	□	□	□
Performance	□	□	□	□	□	□
Scalability	□	□	□	□	□	□
Maintainability	□	□	□	□	□	□
Usability	□	□	□	□	□	□
Compatibility	□	□	□	□	□	□
Accuracy	□	□	□	□	□	□
Testability	□	□	□	□	□	□
Diversification	□	□	□	□	□	□
Emulators	□	□	□	□	□	□
Simulators	□	□	□	□	□	□
App Security	□	□	□	□	□	□

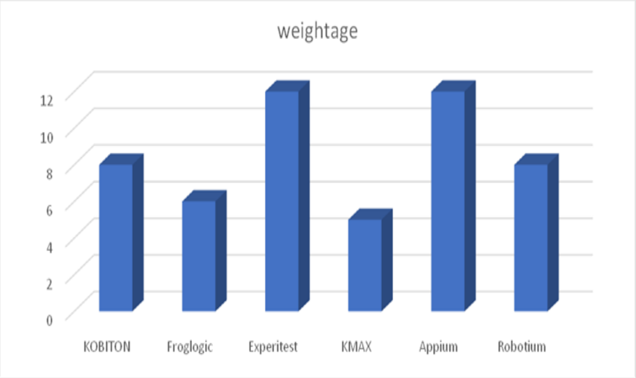

Figure 3: Results (Findings)

A lesser level of toolsare watched for other significant quality traits like extensibility, practicality, adaptability, and platform similarity. Experitest (Seetest) and Appium are superior to different Tools in light of the fact that they will in general accomplish twelve out of fourteen quality variables. The remainder of the devices help to accomplish under five quality characteristics in the AUT. Along these lines, it is recommendable that for an AUT, more than one Tool ought to be utilized to guarantee all the basic quality elements. There is no computerized testing device or answer for Mobile applications which tests for all applicable quality factors that are referenced. Most testing Tools spread just ease of use, rightness and strength, which are wanted by pretty much every Mobile application. To help steady improvement with testing, and post sending practicality and adaptability, just a couple of Tools fill this need. Along these lines, pattern of automated testing is high on ease of use, rightness and heartiness, normal on testability and execution, and lesser on extensibility, viability, adaptability, and platform compatibility.

VIII. CONCLUSION

There is no Mobile application testing Tool which tests for all conceivable quality components. Most testing Tools spread just ease of use, rightness and power, which are wanted by pretty much every Mobile application. To help gradual improvement with testing, and post sending viability and adaptability, just a couple of Tools fill this need. Pattern of automated testing is high on ease of use, rightness and heartiness, normal on of testability and execution, and lesser on extensibility, practicality, adaptability, and platform similarity. In automated testing of Mobile applications, further research should be possible to propose computerized Mobile applications testing Tool that intends to accomplish every quality factor. A comparable analysis can be made by considering testing tools for other Mobile OS like windows. A near analysis should likewise be possible on nature of applications of various Mobile operating systems dependent on automated testing devices of every platform.

A similar investigation should likewise be possible on nature of applications of various Mobile operating systems dependent on automated testing Tools of every platform. Also, based on the tools distinguished from this analysis, overhauled and upgraded arrangements can be proposed for accomplishing the most extreme number of value characteristics in the AUT.

IX. REFERENCES

[1]. Pezze M. and Young M., Softaware Testing and Analysis, Wiley, 2008.

[2]. Shalini Gautam & Bharti Nagpal, “Descriptive Study of Software Testing & Testing Tools”, International Journal of Innovative Research in Computer and Communication Engineering, Vol. 4, Issue 6, pp.10288-10295, 2016.

[3]. Tarik Sheth & Dr. Santosh Kumar Singh, “Software Test Automation- Approach on Evaluating Test Automation Tools”, International Journal of Scientific and Research Publications, ISSN 2250-3153, Vol. 5, Issue 8, pp.1-3, 2015.

[4]. Richard E.Fairley “Software Engineering Concepts”, McGraw-Hill Education (India) Pvt Limited, 2001.

[5]. V. Maheshwari and M. Prasanna, “Generation of Test Case using Automation in Software Systems – A Review”, Indian Journal of Science and Technology, Vol 8(35), pp.1-9, 2015.

[6]. Sanjeev Gupta, Sunil Kumar, Chirag Saxena, “Review Paper on Comparison of Automation Testing Tools Selenium and QTP”, MIT International Journal of Computer Science and Information Technology, ISSN 2230-7621,Vol. 5, No. 2, pp. 55-57,2015.

[7]. Patton R., Software Testing, SAMS, 2nd Edition, 2005.