Automated Tool for Quality Assurance in Mobile Application

Ahmed Al-Shaflut

Computer Science and IT, King Abdul-Aziz University, Saudi Arabia

aalshaflut@stu.kau.edu.sa

Prof. Fathy Albouraey Eassa

Computer Science and IT, King Abdul-Aziz University, Saudi Arabia

fathy55@yahoo.com

Abstract— This research investigates the quality assurance of mobile applications. It specially focused on the automated quality tools. It has started with an overview of quality in mobile environments. The impotence of differentiating the mobile applications than desktops applications. Different models of testing applications was discovered. Selection criteria of any automated testing tool was declared. Finally, we have designed a prototype model for an automated tools for quality assurance. It was mainly focused on the quality assurance of performance. Three major attributes of mobile application's performance were examined. These attributes are request delay, power consumption and network delay.

Keywords— quality assurance; performance of mobile applications; techniques of QA for mobile applications; automation tools of QA; Quality assurance models.

I. INTRODUCTION

The market of mobile application has increasingly risen nowadays. Different platforms and devices have received many mobile applications for different practices. The market of mobile applications in different platforms, is daily updated. The quality assurance of these mobile applications is an important factor to achieve their success. Therefore, achieving the quality consideration is thoroughly important for both developers and application users. There are many tools to assist developers testing their applications before the launching the final product.

In this research, section II describes the concept of quality assurance of mobile applications. It has considered the differences of mobile applications to other environments. It also describes the variety of quality assurance models. Section III, describes

automated mobile application testing tools. It has declared the decision making guidelines for choosing an automated testing tool for developers. Section IV draws the designing criteria for our tool. This tool was designed to assure the quality of performance for any mobile application. Section V has concluded the research, as well as declaring the future work.

II.   LITERATURE REVIEW

A. Quality Assurance Importance

Quality assurance has been discussed in many researches relating to software and mobile applications. It has been defines by international standards Organization ISO as "the features and characteristics of product or service that bear upon its ability to satisfy stated needs"[8]. Hence, the software quality of any software mobile or desktop based, should satisfy the user's needs. The importance of quality assurance is obviously resulting on any software development cycle. Furthermore, it would keep the developers, locating an exact position of their software's worthiness. However, the methods of practicing the quality are different from software to others. Any method should address different aspects to insure the maximum rate of quality. Hence, these methods are different in desktop applications from other methods in mobile application. In mobile application, quality methods should consider different criteria for insuring the user's satisfaction [1]. The connectivity issues compulsory been addressed when developing mobile applications since the user is virtually in dynamic mode. Restrictions also are playing rule such as receiving calls when using an application and the according actions. The diversity of platforms of mobile applications would arise an important factor on determining the quality aspects. In fact, the percentage of 32.6 % of original features is not existed in the final mobile software projects [2]. The lack of following a quality assurance method will result on not implementing the required features. Hence, this lack would be a shortage point from the end-user prospective. However, the user should be provided with a product with 100% features activation. Since it is costly to solving faults that found by users, following quality assurance strategies would find these faults during the application lifecycle.

Several researchers has studied the importance of quality aspects various mobile applications. The Standish Group International has published a report called ' CHAOS', which contains information about various statuses of different issues of software development project. It was classified into these projects into small, medium and large projects [3]. Figure 1 shows the improvement (1995 - 2012) in the success of software industry. As shown, from the percent of 16% in 1995 to approximately 40% in 2012, which means that the number of successful projects has raised about 24% within this period. Other improvements dimensions have been declared within figure 1, where it has also showed that less failed software and challenged projects. This shows how importance is following quality process when developing mobile applications. It also can be concluded that the necessity of software quality issues, to be included in the process of software development in order for applications to be covered from the quality prospective [2].
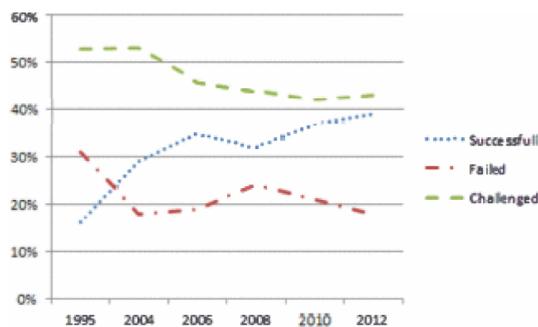

Figure 1 CHAOS Report of Software [2][3]

B. Quality Assuracne Models

Models of software quality are existed as indicators of quality for any software. In fact, they are providing quality measurements to developers during the software development. They are important to achieve the desired level of quality for the products. Many of these models are suitable for use in desktop software, however they may not be suitable for mobile applications. Two popular models are ISO/IEC 9126[6] as well as Boehm et al [7]. They are generalized for tailoring quality for software. However, they are slightly hard in using for a specific mobile application. Recent applications are hard to be insured by these two models, since they are several aspects are totally altered. The continuous of applications developments and the rapid market are periodically updated. In general, quality models are demanded to consider the continuous variation in software application as well as to consider the market demands. However, several considerations discover differences in mobile applications quality aspects. Flexibility of mobile application is an issue that should considered in designing the model of quality. Since mobile environment is dynamic and suddenly changeable. Measuring the flexibility of the client would effect when designing mobile applications.

Another issue is enabling feature of adaptability. This is an important issue since the product should be adaptable to run over various platforms and devices. In fact, many aspects can declare that the mobile applications are different in terms of quality assurance practices.

During development process, the quality model is preferred being adopted together with other considerations. Hence, it can be combined from the initial steps of development till the runtime product [3]. In the process of designing, coding and testing, the quality model should earlier be determined in order to enhance the best possible of quality features during different process. Based on the chosen model of quality, the testing mechanism will be accordingly occurred. Hence, this raises the importance of quality assurance during development process in order to satisfy the end users. It also would assist enabling all the features of any mobile application. Furthermore, this approach will insure ensuring all the product features, since the quality model determines these features earlier.

C. Mobile Applications Considerations

Due to the various platforms and the applications developed, it has discovered the complexity of defining the general metrics for quality assurance. In desktop, it could be easier defining these metrics. However, it should be well considered to meet the requirements of mobile applications. There are several characteristics should be addressed to mobile application. So the network connectivity is a key point for insuring. In mobile application, the channel of communication is slightly different since it uses for example WIFI or 3G connections. The varieties of platforms are also another issues that distinguish the mobile applications. Ways of interacting between user and devices are different since most of applications are used by touchscreen technologies. The variety of operating systems would raise another issue, for example the same application can be installed in different operating systems such as IOS or any version of Widows. The increasing growth of mobile application would also be an issue to be considered when developing mobile applications [2].

There are many considerations that should be taken in account, varying the mobile applications from others. These differences should be considered in mobile environments. Considering them would make an advantage for testing purposes and quality assurance. For example performance assurance, there are several considerations should be addressed such as mentioned in [15]:

- Device variations

The market of mobile application is filled of applications that are presumably working in different platforms, mobile devices, versions and operating systems. In fact, the difference is thoroughly wide between tablets and smart phones although both run on the same operating system. Furthermore, the difference of power is varied from device to another.

- Connectivity

Mobile device can be connected in different scenarios, however the desktop can be following one or two situations of internet connections. For example, mobile device can be connected to cell tower of the network operator. Each request of mobile customer is sent to the operator server proxy. Then the proxy will coordinate with the web server in order to provide the request for the mobile device. The previous scenario is completely dissimilar when using mobile device in wireless network mode.

- Latency issues

The nature of mobile application used in wireless network would result on latency of data exchange. This will create an impact on the mobile user's experience. Therefore, latency would be result on both test and regular experience. So it has to be considered when testing the performance of mobile applications. Hence, this latency would be added as another concern, distinguishing mobile application than other desktop applications.

III. MOBILE AUTOMATION TESTING TOOLS

In fact, many several automated tool are implemented in terms of testing tools. This has resulted on enabling automated tools for quality assurance prospective. The automated tools are thoroughly important for insuring appropriate approaches of mobile application development. These tools are truthfully existed to enhance execution testing [10]. They are valuable when comparing results on this point to the earlier test outcomes. These tools are mainly targeting the quality assurance in performance and functionality of mobile applications. In fact, it is also important to generate a list requirements before choosing testing tools. This would guide the tool developer to entire process of selection and evaluation criteria.

A. The Efficiency of Test Automation Tool

Using automation tools would assist application's developers in both debugging and releasing application [11]. It can also be assist in avoiding different errors of humans. Finally, all the test results can be stored in order to provide statistical record of application development process [12]. However, automation tools are different in their focused functions. This mean using several testing tools would result on different testing results. Hence, if any tool is used to test security of an application, it cannot be providing appropriate usability testing, as well as it cannot generate an exact test of performance. So this would raise the need of having key criteria in order to choose suitable testing tool.

11

B. Guidelines for Automation Tool Selection

Choosing an appropriate tool is an important to achieve the test perfectly. The selection criteria for mobile application are more sensitive. The main function of the tool is to handle multiple browsers, emulators, different operating systems and unique types of testing [14]. There are many tools available in the market; however choosing an appropriate is an inflexible task. Since they are serving different purposes in their test mechanism and then would be suitable for most test aspects. Some of these tools can perform lonely specific feature testing. The lack of supporting several languages would also be resulting on choosing testing tool. Accordingly, choosing the suitable test tool, with limitations stated earlier as a challenging process due to variation of purposes and features for any chosen testing tool.

IV. DESIGNING AN AUTOMATED TOOL TO ENHANCE QUALITY OF PERFORMANCE

Automated tools can measure a specific quality attribute in order to enhance the comprehensive assurance of quality. In this tool, focusing vision is based on the performance application. Therefore, having an assured performance test would provide extraordinary impression of this application. Using this architecture in figure [2], would benefit assuring other quality perspectives by changing the techniques that currently used of performance. This tool can be used to measure performance of any application. It would allow examining the specific and overall performance. For overall reporting, it would give a complete report for all the examined functions. As well as, this tool could measure an individual performance metrics. These metrics include request delay, power consumption and network tests. In fact, testing mobile application is a complex task due to the complexity of mobile application environment. Thus, designing the mobile application testing tool should consider the variety to desktop applications. The tool prototype is based on combining the most performance functions to test given application. It has the graphical interface having parser, family of device in order to manage the variability in visualization. In parser would be initially classifying the mobile application based on earlier groups. An application would be determined to a specific family. This step is done based on application type, device type and operating system. Finally, within the graphical interface user, it would be simulating the mobile application based on the given data in order to manage the other steps of performance test. The core part of this tool is behaved as a middle ware tier, where the techniques of performance are declared in the next step. At the end, test results will be stored in the database for future test purposes.
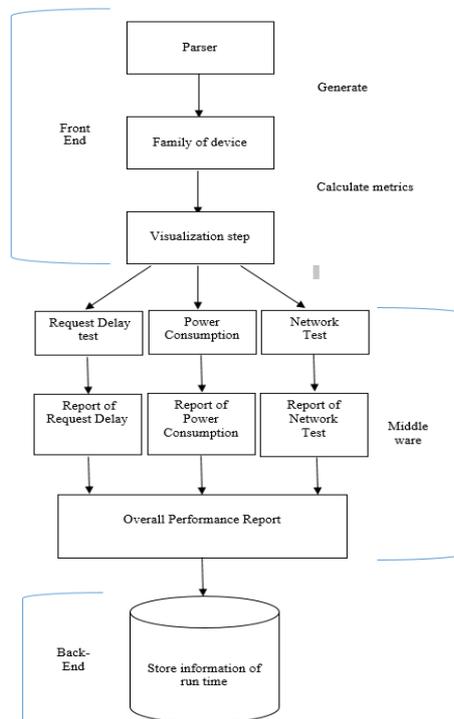
**Figure 2 an automated tool for performance test of mobile application**

In figure [2], the complete process of the testing tool is declared. However, for this tool, it will concentrate on the middle ware techniques. This would include different criteria for power consumption, network delay and delay of request of the mobile application. It would discover these techniques used for assuring the requirements for the three metrics. Hence, this tier has the core testing process.
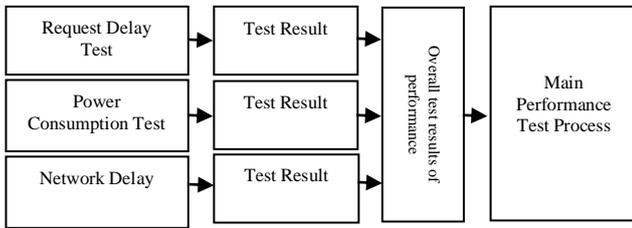
Figure 3 Core testing process in the middle tier

A. Request Delay Test Technique

For this part of test, we would announce the techniques used to measure that the request is handled on time. Preventing the delay of request would be an important advantage of this tool. In fact, any request to the application should be responded with an exact time. This also excludes the latency that might be happen due to the network delays. This would enhance the reliability of the application's user when requesting any function. The function of calculating the time of request till response is a 'Request Delay Test'. So for calculating the response time, the time difference is important to get an exact delay of any request. So first the time of request is generated then the time of receiving the response. After generating the response, the time difference is calculated. Considering the other factors can affect in the time delay for example, the network latency. Thus in order to account the time an exact time of delay between request and response, we had to exclude the latency of network to be individually measured. So this would calculate the response time for only a single request. In order to test the performance of any mobile application, it has to test many requests. Different requests also included finding an average request delay for any particular application. If request delay is decreased that means the performance of this application is increased. Also the decrease of delay time would be resulting on the decrease of application's performance. In order to measure the different request and different parameters of request delay, another calculation method is included. Hence, it will calculate the average of delay for all these requested. Based on the response time differences for each request.
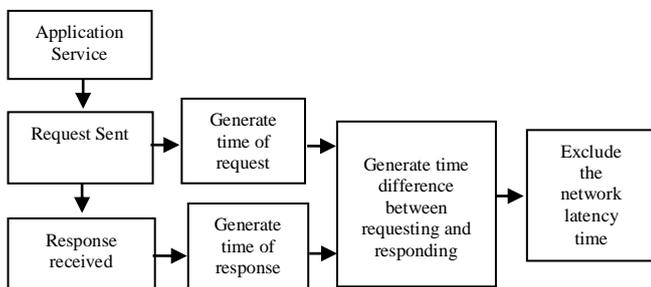
Figure 4: Request Delay technique

B. Power Consumption Test Technique

The power consumption is thoroughly important factor when designing any mobile application. The differences of size and capability of any mobile device battery is another element that is resulting on performance of mobile application. Different applications have different consumption values of power. As well as different requests of the same application can consume changeable power needs of different requests. The power consumption values depend on the type of request and the device's capabilities. Based on the techniques declared in [16], there are many techniques of managing power consumption of mobile applications. They would be maximizing the power efficiency of any application. Therefore, before designing this tool, the best practises of power management techniques were studied in order to correctly measure the power of the tested application. These practises can are listed below, concluding with the chosen practise. Finally, we have declared the reasons beyond choosing this particular practise. Choosing power management technique for mobile application is classified into the following techniques:

1. Minimizing of application and content on mobile devices
In this technique, the main goal is to balance the resources used with the device's capabilities. Since the mobile applications generally run on small devices, it is proffered to use limited resources. From this this prospective, it would reduce the resources and dependably increase the power efficiency. Main goal is of this technique is not passing the limited usage of resource for each application on the mobile application.

2. Reducing the screen power consumption
In this technique, it would be an important to focus on power management from the side of screen consumption prospective. Since the power can be largely consumed by brightness of the device. Therefore, the clear difference of power usage can be

obviously seen when the settings brightness of any application is put in low level. In fact, applications that can perfectly behave when reducing the brightness power consumption and without resulting on the quality of user experience, they are unquestionably increasing the power saving of the mobile devices. In this technique, both visibility and readability are considered. Since they are both important for quality assurance of user, they were thoroughly covered.

3.   *Adaptive Behavior of applications*

This technique is based on the fluctuation of time-spent form an application action to another. This fluctuation depends on the workload. Thus, actions will be adaptively treated based on the needed and available time. For example, any request to the server would be depending the available rate of resources. It will maintain the server capabilities before performing any requested action. In general, many requests cannot be beyond the capabilities of server. There are two advantages can be gained by following this technique. One is the power consumption efficiency, since it uses only the available rate of resources. Second advantage is the efficient increasing of power by minimizing the resources consumed on mobile devices. For this tool, we have decided to follow the first technique. Minimizing the application and content on the mobile device. Since is the most power and computational functions will be done in the server side, therefore resources of this device will not be consumed. The saving mechanism we have followed, largely benefits the power consumption feature. At that point, it will increase the efficiency of power consumption. Consequently, this will result on the performance test for this tool.

C. Network Delay

The network latency has an impact factor of mearing performance of any application. This depends on the network that is connected by mobile device. Therefore, this connection would effect of the time of requesting and receiving services. Since, many services are connecting the server for generating required actions. For the network delay, it was excluded in our tool to exactly provide the best testing result for each individual performance attributes. The most challenges was faced when measuring the network delay, since it combines several measurements to output an exact test outputs. Practically, an exact network delay is generated by an open application that was provided to measure the network performance.

Accordingly, the network latency is different based on the traffic as well as by the type of requests. For example, it would slightly delaying more for when the request contains multimedia uploads or downloads. On the other hand, requests with text output would be less loaded on the network traffic. The network delay measurement also can be depending on the type of network. Hence, mobile devices that connects to wireless network, is different than device with cellular network connection. In fact, there are many several factors are resulting on this step. The quality of service that earlier agreed, would be effecting on that test. As well as, the device supporting networks when using cellular connection mode. In fact, there are two techniques can be followed in this test. One is after calculating the delay of request of the mobile application. Earlier, in the request delay mearing, we had excluded the network delay in order to accurately test the response time. However, in this test we can calculate both tests which would assist in generating the overall result. The other technique, can be performed by measuring calculating the current data transmitting over the network and compare them to the quality of service contract.
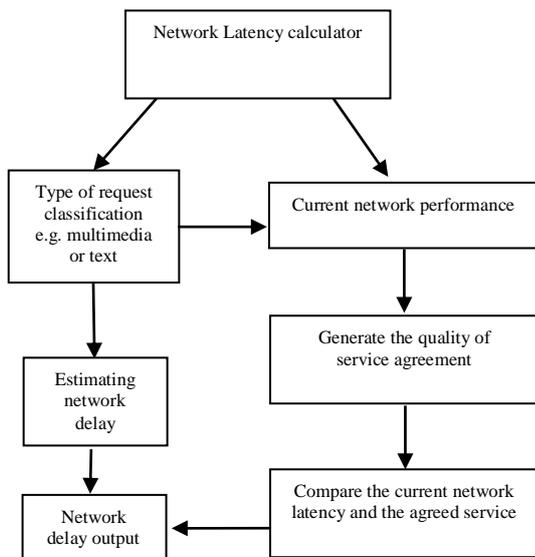

Figure 5: Network Delay technique

In fact, there are several advantages of using this tool to test any performance of mobile application. In fact, automated tools are similar in their concepts. However, there are privileged features of our automated tool. It specifies in enhancing the quality by testing the performance metrics. Three major measurements are specifically chosen. Output classification was an important difference. Since, this tool enables generating individual and overall reports. It has discovered the importance of having performance guaranteed application. It has designed the comprehensive prototyping model for performance purposes. This tool can be performing tests on both emulators and real mobile devices.

V. CONCLUSION AND FUTURE WORK

In conclusion, quality assurance is an important factor in mobile application. Different automated tools of enhancing the quality of applications are existed. However, we have prototyped an automated tool for testing the performance of any mobile application. This tool is deeply covering the most three factors of performance attributes. Having a reliable testing tool will effect on the user experience of any mobile application. We specifically have designed the measurement techniques for testing three major factors. These factors are request delay, network and power consumption. For mobile application's performance, it is thoroughly important to ensure the best practice of these results.

In future work, it will be performing the different quality attributes by following the same architectural model. It will be also attaching different application programming interfaces to be used for testing as many as possible mobile applications.

REFERENCES

[1] Franke, D.; Weise, C., "Providing a Software Quality Framework for Testing of Mobile Applications," in Software Testing, Verification and Validation (ICST), 2011 IEEE Fourth International Conference on , vol., no., pp.431-434, 21-25 March 2011.

[2] Villasana, G.; Castello, R., "An agile software quality framework lacking," in Computer Applications and Information Systems (WCCAIS), 2014 World Congress on , vol., no., pp.1-4, 17-19 Jan. 2014

[3] T. S. G. International. (2013) Chaos manifesto 2013: Think big, act small. The Standish Group International.

[4] p. Runeson and P. Isacsson, "Software quality assurance-concepts and misconceptions," in Euromicro Conference, 1998. Proceedings. 24th,
vol. 2, 1998, pp. 853-859 vol.2.

[5] A. Nolan, "New sources of growth: intangible assets, " September 2011 [Online]. Available: http://www.oecd.org/stilinno/46349020.pd

[6] International Organization for Standardization and International Electro technical Commission, ISO/IEC . 9126,http://www.iso.org, 2006.

[7] B. W. Boehm. Characteristics of Software Quality. NorthHolland
Pub. Co., Amsterdam, Netherlands, 1978.

[8] International Standrads Organization (ISO), 2005. ISO 9000:2005 Quality Managamenet Systems – Fundmentals and vocabulary. ISO Standard.

[9] Saad, N.H.; Awang Abu Bakar, N.S., "Automated testing tools for mobile applications," in Information and Communication Technology for The Muslim World (ICT4M), 2014 The 5th International Conference on , vol., no., pp.1-5, 17-18 Nov. 2014

[10] Paul. Laplante, F. Belli, J. Gao, G. Kapfnammer, K. Miller, W. E. Wong and D. Xu. Software Test Automation. Special Issue of Advances in Software Engineering. Hindawi Publishing (2010).

[11] Hamish. Muccini, A. D. Francesco & P. Esposito, 2012, "Software Testing of Mobile Applications: Challenges and Future vZtu"°tomated Software Testing", 2011

[12] L. Mike, Rochester Institute of Technology, "Testing Tools", IEEE, 0740-7549/90/0500/0053, May 1990.

[13] T. Pradhan, 2011, "White Paper: Mobile Application Testing", TATA Consultancy Services.

[14] A. Savkin (2013). Automation Testing in Android Application.Retrieved from http://architects. dzone. com/articlesAautomation-testing-android.

[15] Grochowski, Ed, "Keynote presentations," in Computer Design, 2009. ICCD 2009. IEEE International Conference on , vol., no., pp.1-6, 4-7 Oct. 2009

[16] Verma, D.; Verma, P., "Power Efficiency for Mobile Applications," in *Techniques for Surviving Mobile Data Explosion* , 1, Wiley-IEEE Press, 2014, pp.224

[17] Jaewan Shin; Dongkyoo Shin; Dongil Shin; Sungmin Her; Soohan Kim; Myungsoo Lee, "Using the baseband processor power management technology for mobile devices," in *Industrial Electronics & Applications (ISIEA), 2010 IEEE Symposium on* , vol., no., pp.591-595, 3-5 Oct. 2010

[18] Wong, J.L.; Gang Qu; Potkonjak, M., "An on-line approach for power minimization in QoS sensitive systems," in *Design Automation Conference, 2003. Proceedings of the ASP-DAC 2003. Asia and South Pacific* , vol., no., pp.59-64, 21-24 Jan. 2003

[19] Chandrakasan, A.; Potkonjak, M.; Rabaey, J.; Brodersen, R., "An Approach For Power Minimization Using Transformations," in *VLSI Signal Processing, V, 1992., [Workshop on]* , vol., no., pp.41-50, 28-30 Oct 1992

[20] Ramirez, R.I.; Hernandez Rubio, E.; Meneses Viveros, A.; Monserrat Torres Hernandez, I., "Differences of energetic consumption between Java and JNI Android apps," in *Integrated Circuits (ISIC), 2014 14th International Symposium on* , vol., no., pp.348-351, 10-12 Dec. 2014

[21] Li Zhang; Stover, C.; Lins, A.; Buckley, C.; Mohapatra, P., "Characterizing Mobile Open APIs in smartphone apps," in Networking Conference, 2014 IFIP , vol., no., pp.1-9, 2-4 June 2014