

Mobile Software Testing: Thoughts, Strategies, Challenges, and Experimental Study

¹Mohammed Akour
Computer Information Systems Dept.
Yarmouk University
Irbid, Jordan

³Ahmad A. Al-Zyoud
Computer Information Systems Dept.
Yarmouk University
Irbid, Jordan

²Bouchaib Falah
School of Science & Engineering
Al Akhawayn University
Ifrane, Morocco

⁴Salwa Bouriat
School of Science & Engineering
Al Akhawayn University
Ifrane, Morocco

⁵Khalid Alemerien
Comuter Science and Infromation Technology Dept.
Tafila Technical University
Tafila, Jordan

Abstract—Mobile devices have become more pervasive in our daily lives, and are gradually replacing regular computers to perform traditional processes like Internet browsing, editing photos, playing videos and sound track, and reading different files. The importance of mobile devices in our life necessitates more concerns of the reliability and compatibility of mobile applications, and thus, testing these applications arises as an important phase in mobile devices adaption process. This paper addressed various research directions on mobile applications testing by investigating essential concepts, scope, features and requirements for testing mobile application. We highlight the similarities and the differences between mobile APP testing and mobile web testing. Furthermore, we discuss and compare different mobile testing approaches and environments, and provide the challenges as emergent needs in test environments. As a case study, we compared the testing experience of hybrid application in an emulator and a real world device. The purpose of the experiment is to verify to which extent a virtual device can emulate a complete client experience. Set of experiments are conducted where five android mobile browsers are tested. Each browser will be on a real device as well as an emulated device with the same features (CPU used, memory size, etc). The application will be tested on the following metrics: Performance and function/behavior testing.

Keywords—Software Mobile Testing; Software Testing; Mobile Performance Testing; Hybrid Mobile Application

I. INTRODUCTION

The last few years have been revolutionary for mobile devices. The mobile device was transformed from simple reutilized operation device to a more complicated and sophisticated one. Mobile applications become extraordinary when adopted by individuals and organizations to organize and manage almost every single life activity [1, 2, and 3]. Mobile devices are a necessity in our lives due to attributes like:

ubiquity which means availability anywhere at any time, convenience, instant connectivity, personalization, and location-based services [4]. Mobile applications become so popular and available in different types and for different purposes. With the fact that more than one billion smart phones were sold and 6.8 billion mobile subscriptions worldwide in 2013, more than 300,000 applications have been developed [5], and a prediction of 76.9 billion global downloads in 2014 [6], the importance of testing these applications is surfaced for both privacy and security purposes [1].

The mobile application landscape is continuously growing and highly dynamic. The widespread availability of mobile devices combined with the declining prices and increased functionalities make them more powerful than a PC. The handset culture and improvement of bandwidth, and other factors together encourage the new operating system versions releasing and various types of mobile applications developing [4]. Testing mobile applications is an expensive, time consuming and complex process [2, 3, and 7], but it is still required and needed to guarantee consumer's satisfaction every time they use the mobile applications. It's vital to make sure to test the application by specialist and experienced testers not just to find errors but also to critique the quality of mobile applications before release it in the market to avoid exploring problems by consumers who could simply never use the application again, and you may never hear a word from them [2].

Testing mobile applications is essential to ensure the usability, mobility, and security of the applications. One of the main challenges of testing a mobile is the high cost and availability of devices. More and more companies rely on emulators in the early stages of development.

In this paper, we compared the testing experience of hybrid application in an emulator and a real world device. The purpose of the experiment is to verify the extent to which a virtual device can emulate a complete client experience. Mobile device browser engine is an example of a hybrid application that combines the features of web applications and native applications. Web browsers have access to hardware of mobile devices to provide native features such as GPS localization. They also benefit from a server side technology used in mobile web application.

The rest of this paper is organized as follows. Section 2 provides an overview of Mobile application testing scope and requirements. Section 3 gives a brief description about mobile application testing strategies. Challenges of testing mobile application are presented in section 4. Experiment Setup is described in section 5. Section 6 presents the analysis and discussion. We conclude the paper in section 7.

II. PRELIMINARY INSPECTION

In this paper, we addressed the following five research questions:

- 1) *Are mobile applications different than familiar software's?*
- 2) *What are the types and characteristics of mobile applications?*
- 3) *What are the scopes, types, and strategies of mobile application testing?*
- 4) *To which extent virtual device can emulate a complete client experience? In terms of performance and functioning.*
- 5) *What are the requirements, and distinct features in mobile application testing?*
- 6) *What are the challenges and solutions of mobile testing?*

A. What is a mobile application?

Mobile applications are application software designed and implemented to run on smart phones, tablet computers, and other mobile devices with operable graphical user interface (GUI) to perform certain tasks. They are rapidly developing segment of the global mobile market and can be downloaded through USB / WIFI or can be downloaded by a web server over internet [1, 7, and 8] It is necessary to differentiate mobile applications from the traditional ones. While the mobile applications are designed and implemented to support mobility devices, the traditional ones are more likely implemented to run on desktop computers. The former is also aware of the environment in which it runs and adapts according to its computing, user, physical, or time context which is known as context awareness computing. Thus, mobile applications require specialized and different testing technologies [1, 3, 8].

There are mainly three types of mobile applications: mobile apps, mobile web applications and hybrid mobile applications [1, 3]. Mobile apps, known also as App4Mobile [1], are native mobile applications deployed and executed on mobile devices with limited resources and driven by user inputs. These applications usually depend on native mobile API "Application Programming Interface" and dongles on mobile devices such as camera [1]. This kind of application can be seen as offline application which could be run and used without need of

Internet connection driven by user and make use of contextual information to generate context-based output [8, and 9]. This type of applications is built for a specific platform with the platform software development kit (SDK) tools and languages, typically provided by the platform vendor [10].

Mobile web applications are server-side apps built with any server-side technology. Mobile web client software are run and used over web browser. It's an online application form that could be run using Internet connection such as YouTube application [3]. Some applications could be hybrid form of applications. These applications run on the device, and are written with web technologies (HTML5, CSS and JavaScript) it is run inside a native container, and leverage the device's browser engine (but not the browser) to the HTML and process the JavaScript locally [1, 3, 8].

Regardless of business objectives or usage purposes, mobile application has to have some qualities and distinctive characteristics to be successful, like: connectivity, convenience (quality design), supported devices, new programming language, and context awareness [1, 8]. Other researchers mention more characteristics like: security, personalization, reachability, and localization.

Since mobile devices are almost always logged in to the mobile network, it makes mobile applications always connected. The mobile network may vary in speed, reliability and security. Therefore functional testing has to be performed in different networks and connectivity scenarios. Graphical user interface must be tested in different devices to test the ease of use or convenience of mobile application. Mobile application has to be easy to install, easy to access, and easy to use [1, 8, 9]. Mobile Applications Usability Testing Challenges and Methodologies are discussed in detail in [18, 19].

The availability of various mobile devices supported with a different software features ,hardware components, and operating systems makes testing process more and more difficult. To make sure that the application is compatible and functional for all devices we demanded to perform testing on varied combination of devices with right combination of OS. This covers the characteristic of supported devices or platform appropriate characteristic, new programming language designed to support mobility, managing resource consumption, and handling GUIs. Therefore, conventional testing techniques need to be revised to be applied to new mobile programming languages. Context Awareness means that mobile applications use data provided by context providers, which are sensing like light and connectivity devices. Those devices may provide a huge amount of inputs that vary depending on the environment and/or user actions. This requiring context-specific test selection techniques and coverage criteria have to be introduced [1, 8, and 9]. They should be unique and not be a mirror of what your competitor provides. This will add value to your application. The level of enjoyability and entertainment are important features to differentiate any mobile application. Mobile applications tend to be more focused and more determinant purposes, deals with short duration activities. There are several factors to be kept in mind while designing mobile applications such as using mobile applications in different places either quiet, appropriate or crowded and noisy

places. Therefore, designing an application with only speech input is not a great idea, and at the same time, the mobile device itself should not be a disturbing source to others [20]. Different types of mobile applications with various characteristics mean definitely different approaches and techniques of testing and increase challenges of testing; the following section will discuss the scope and types of testing in depth.

B. Mobile Applications Testing Scopes And Types

The term Mobile application testing refers to testing the activities for mobile applications on mobile devices using well defined software test methods and tools. Testing is needed to ensure quality of service (QoS), mobility, usability, interoperability, mobile connectivity, security, and privacy, [3, 12] are working as desired. Mobile applications are getting more and more complex, which make testing their stability and robustness [13] is necessary. Mobile applications testing process is complicated for several reasons: high dynamicity in the mobile phone manufacturing world, frequently updating software, and lack of unified supporting test operations where each application has its own unique business and data flow [3, and 12]. Testing mobile applications can be done using different approaches for different objectives, that is, hardware compatibility, software reliability, application functionality etc.. Some testing mobile application purposes are summarized as follows:

- Mobile function and behavior testing: this refers to checking the validity of mobile functions and behavior under all possible situations [3].
- Mobile system QoS testing: check the mobile application scalability –ability to handle a growing amount of work in a capable manner -, reliability – consistency-, availability –instant connectivity and operation-, and performance [1, 3, and 8].
- Mobile interoperability testing (compatibility): checking the mobile application crossing different platforms to make sure that the application is compatible with other applications and promotes cross-use functionality [1, 2, and 3].
- Mobile usability and internationalization testing: internalization refers to reviewing the design and development of software or Web applications to make sure if it can be easily adapted to various linguistic and cultural environments without additional programming. Internationalization reduces time and cost of getting a product to international markets and facilitates localization of the product in a specific market. Usability includes text visibility on the selected language, navigation between screens, and functionality of online/offline [3, 14]
- Mobile system security testing: concerning the privacy of personal and business stored information on mobile devices include inscription/decryption techniques used for sensitive data. Attacking security may come from means of communication like SMS, MMS, WiFi, and Bluetooth, or may exploit software vulnerabilities from both the web browser and operating system [1, 3, 8].

Other dimensions of testing include: mobile multi-tenancy testing “validates Software as a Service -SaaS- features and QoS requirement, mobile connectivity testing “validates mobile application in different wireless network connectivity and context environment[9]. System mobility testing “validates location-based mobile system functions, data, user accesses, and services” [14].

C. Mobile Applications Testing Requirement And Features

We are moving from a PC society to a mobile society, which has become a dominant phenomena, where more processes are done by mobile devices, and more personal and business information is stored and accessed from these devices. Unlike conventional application testing, mobile application has set of distinct requirement and features, such as [3]:

1) *Testing anytime and anywhere: mobile applications are accessible at anytime and anywhere; therefore mobile application function should be correct anytime and anywhere.*

2) *Testing crossing mobile platforms and browsers on variety of mobile devices: mobile devices have different operating systems; display dimensions, hardware appliances, and battery operation time. Therefore testing mobile applications must be conducted on selected device with a diverse mobile platform.*

3) *Testing for good and rich mobile experiences: mobile applications are designed and developed to support high experienced users by providing multiple input channels, rich media features, native application interfaces, and hardware equipment.*

4) *Testing using large-scale mobile simulation and virtualization: this is required to evaluate mobile application performance and scalability so that hardware costs can be reduced*

5) *Testing with divers network connectivity: mobile applications must be validated under different network connectivity and related contexts, because mobile devices support diverse wireless network connectivity (such as 3G, Wi-Fi, and Wi- Max).*

III. MOBILE APPLICATIONS TESTING STRATEGIES

Mobile application testing represents many techniques and tools to meet quality requirement and can be classified in different ways. One of these classifications is based on the underlying client and server infrastructure. There are mainly four approaches to this classification:

1) **Device Emulator** [2, 3, 9, 10, and 14]: 30%-40% of tests are performed using this approach. In this approach, quality assurance team validates mobile applications using emulator on mobile devices with various options like ability to bypass the network, and use of an effective scripting language. A tester can easily switch to different device types by simply loading the appropriate device profile [13]. This is a cost-effective approach because no real devices are needed during testing. Emulators are mostly available free and different types of testing like user interface, stress, and performance testing can be performed using emulators. The disadvantage of emulators

is that they lack the peculiarities; in case of running the same tests on the same mobile device may not reflect the actual results [2, and 13]. Moreover, the availability of limited number of emulators while there are large numbers of mobile phones and those emulators may have bugs that cause suspicious defects reported [2]. Muccini and Kirubakaran [1, 8] state that simulation-based mobile testing is similar to emulator in that no real mobile devices are required, but differ in the ability of simulating different mobile behaviors on selected mobile devices. However, still both emulator and simulator have shortage and weakness in testing devices-based native mobile function features.

2) **Mobile Cloud Computing Solutions:** in this approach mobile device can be accessed through web interface and the applications can be deployed, tested and managed remotely. This approach is cost effective since it uses pay-as-you-use service model, reducing complexity of the implementation, and keeping tasks and data on the Internet rather than on individual device providing on-demand access. There are several benefits of using cloud solutions, such as 1) reducing the cost through using rent per hour/ swap devices technique, 2) there is no need to cell plan for testing income calls and text, 3) investigating failures by recording video of automated text execution, 3) recording device log to help in trouble shooting, 4) availability of large number of devices for testing, and 5) test can be run on several devices in parallel[1, 2, 3, 8, and 10].

3) **Real Time Devices with Real Networks:** this approach is a device-based mobile testing, where real mobile devices are purchased and used by enterprise-based test laboratory to validate application functionality, dynamic behavior, and QoS parameters, this approach is cost consuming as actual devices are needed. The last approach uses automation Tools to avoid manual work, and quality assurance team automates mobile application tasks to reduce time and cost., Examples of available automation tools for mobile applications are: Android, Android instrumentation, QTP(Paid), iOS, and more [2, 3, 15, and 16].

IV. CHALLENGES OF TESTING MOBILE APPLICATIONS

Mobile application testing is an activity aimed at evaluating and improving the quality of the application by identifying defects and problems. The testing process and its results must consistent and unbiased, this create a diverse set of challenges and barriers at different testing levels. At this point we might need to make a tradeoff between the test strategy and test effectiveness. There are no tests called perfect or complete. One may need a combination of tests to be completely satisfied. Basically barriers are covered by four dimensions described in Figure 1.

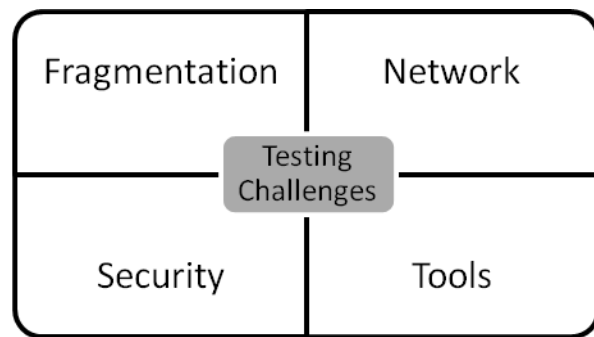


Fig. 1. Testing Challenges Dimensions

As shown on the top-left of Figure 1, the first challenge is fragmentation where there is a large number of mobile devices running on various operating systems. Different user interfaces increase level of challenge due to compatibility. Mobile devices Network performance affects user's application experience; where each mobile operator may support multiple network technologies and some use less common or local networking standards. Testing mobile application in all possible connected networks requires travelling to every network operator which can be very expensive and time consuming. We can overcome the network challenge by bypassing the lower layers of network and testing the application over Internet on network by using device emulator and thus saving time and cost of travelling. On the other hand, bypass cannot emulate the effect and timing of network. Security is another dimension of the effectiveness and validity of the application; it is essential to ensure the application is secured and does not surpass user's personal information. The last dimension of challenge is the variety of testing tools. The possibility of missing the appropriate testing tool should be taken care of as we may find a large set of tools in the literature.

According to Cap Gemini Quality Report [11], the barriers to testing mobile application have shifted from tools to methods, 56% of companies do not have the right testing process/method, 52% do not have the devices readily available, 48% do not have mobile testing experts, 38% do not have in-house testing environment, 37% do not have the right testing tools, and 33% do not have enough time to test. Nevertheless, the data shows that mobile testing rose rapidly in 2013 compared to 2012 where statistics prove that 55% of organizations implemented new methods and tools to validate functionality, performance, and security of mobile applications and devices in contrast to 33% in 2012.

Some researchers classified the testing challenges into three main categories: testing process itself, testing artifacts, and testing type. Figure 2 summarizes the challenges facing mobile application testing process.

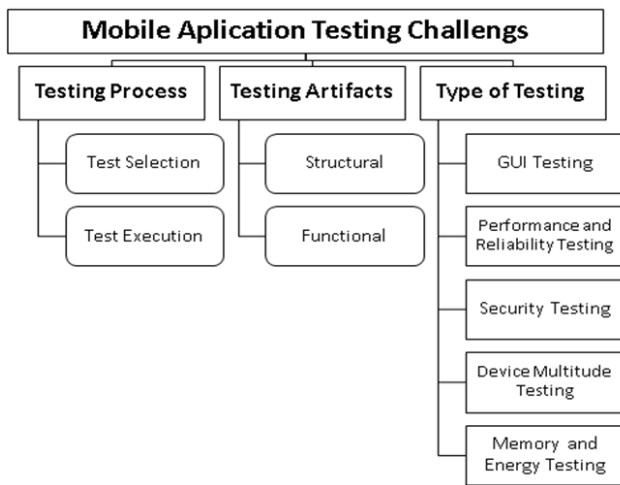


Fig. 2. Challenges of Mobile Application Testing

As we discussed earlier, mobile application can be native, web, or hybrid application and each application has distinctive purposes and structure. Testing Process Challenge based on Figure 2 is classified into two categories: test selection and test execution. 1) Availability of huge number of applications with different types and features might require new testing criteria to ensure coverage of all aspects of testing process, that is, application type, operating system platforms, full/partial capabilities etc. We can refer to this challenge as test selection challenge [10]. 2) Test execution deals with the availability of multitude of mobile devices with a huge diversity of environment. It is difficult to test the application on all devices of the same product family, even if they differ from each other in resolution, CPU, memory, and OS optimization. This leads to the challenge on how to introduce valid and reliable test cases including rich contextual inputs. Existing simulators are unable to simulate real-world phones with their sensors, GPS, connectivity. New capture-and-replay automated techniques can be realized for executing contextual inputs [13].

For testing artifact, we focused on structural and functional testing approaches. 1) Structural testing (code-based/white box testing): mobile application languages add some specific constructs for managing mobility, sensing, and energy consumption. Automated crawling testing software is considered as a powerful and cost-effective tool to set a quick test of errors and device inconsistencies. New coverage criteria can be thought of as a way to consider at best the new mobility sensing, and energy constructs. 2) Functional testing (model-based/black-box testing): functional testing requires specifying both the application and the environment it may operate in. State-based approaches can be useful to model different execution modes (e.g., low battery, meeting, and flying mode) [1, 8, 10, 15].

Mobile testing approaches are widely used in the literature. In this paper, we focus on the following strategies:

1) Graphical User Interface testing: testing whether different devices provide an adequate execution of data, and testing whether applications are correctly displayed on

different devices. Automatically execute scripts that are captured during the user interaction and replayed and modified even on different devices is a good idea for testing GUI. It is important to make this task automatic to save time [3, 9, 10, 12, and 13].

2) Performance and reliability testing: mobile applications performance and reliability depends on the device resources, device operational mode, connectivity quality and variability, and other contextual information [3, 10, and 15].

3) Security testing: mobile networks have different security levels, and computer's viruses might access to personal contextual information which presents real privacy concerns. Conventional security testing techniques shall be revised to keep in consideration contextual factors that shall be simulated to check transmitted data from the mobile device.

4) Devices Multitude Testing: This kind of testing creates the biggest testing challenge. Different mobile phones provide different features, hardware components, and different O.S. This creates a challenge of cost and time consuming when using "test on many devices" approach and need to be replaced by (cost) effective automated techniques [2, 3, and 7]. We can choose one of three options to handle the device challenge: either to testing exclusively using real devices, testing exclusively using emulated devices, or using a combination of both. Using real devices will explore all the limitations in the real hardware and the firmware combination, but it can be costly and time consuming. While in other hand it is easier to manage emulated devices through the ability to load different devices updatable profiles on one platform and exchange among them in a simple and direct way which is cost effective solution, however emulators lack the limitations, and quirks, and characteristics provided by the real devices and does not cover the exact behavior for external conditions like crowding and noisy environment, also emulation depends in PC power. Another suggested solution is to release (free of cost) prototype version of the application to be tested, running it on a multitude of devices, and collect, store, and analyze run-time data and failures [1, and 8]. Testing by using a combination of both emulation and real device create the opportunity to gain advantages of both approach. An emulated environment enhances the speed and device diversity at a relatively low cost. The real devices verify the applications functionality and ensure that all development objectives have been met.

Sahinoglu, *et al.* [21] highlighted mobile application testing landscape and conducted a gap analysis. They provide definitive metrics and publications about mobile application testing.

Zaein, *et al.* [22] conducted a systematic mapping study to categorize and to structure the research evidence that has been published in the area of mobile application testing techniques and challenges that they have reported. In their work, several research gaps are identified and specific testing issues for practitioners are identified.

V. THE EXPERIMENT SETUP

Testing mobile applications is essential to ensure the usability, mobility, and security of the applications. One of the main challenges of testing a mobile is the high cost and availability of devices. More and more companies rely on emulators in the early stages of development.

In this paper, we compared the testing experience of hybrid application in an emulator and a real world device. The purpose of the experiment is to verify the extent to which a virtual device can emulate a complete client experience. Mobile device browser engine is an example of a hybrid application that combines the features of web applications and native applications. Web browsers have access to hardware of mobile devices to provide native features such as GPS localization. They also benefit from a server side technology used in mobile web application.

In this experiment, five android mobile browsers will be tested. Each browser will be on a real device as well as an emulated device with the same features (CPU used, memory size, etc). The application will be tested on the following metrics: Performance and function/behavior testing.

A. Performance Testing

Performance is one of the most crucial aspects of mobile applications. Performance measures how fast the application starts, how fast data loads, and the overall connectivity of the application on different carriers. Many hardware related aspects, such as memory and battery consumption, are evaluated to test the usability and mobility of the application. In this experiment, the mobile devices browsers are tested on the following aspects:

- Memory consumption
- Traffic generated by the testing scenario
- CPU Usage generated by the testing scenario
- Battery Consumption generated by the testing scenario
- Initial loading speed of the application
- Duration of Testing experience

The mobile browser's performance will be tested on the following scenario:

- Open Browser
- Open google.com
- Search Android Wikipedia
- Open the page
- Close Browser

B. Function and Behavior Testin:

Function and behavior testing is an essential part of mobile applications testing. Functionality testing is used to validate the

display of the application's content (images, texts, etc.) under different devices. It also verifies the behavior of the application under all circumstances. In this experiment, the browser will be tested based on the following functionalities:

- Auto-suggest feature for URL
- Downloading information from internet
- Uploading information on the internet
- Tab support
- Add-on support: PDF viewing, Flash player support
- Support of drowsing gestures (e.g.: double tap to zoom)
- Special function support

a) GPS support

b) Rotation

c) Camera

VI. RESULTS AND ANALYSIS

The results of the experiment are represented in Figures 3 to 6. Figure 3 is the initial loading speed of the browsers under test obtained when using Device 1 (Samsung Galaxy S3 Mini) and Device 2 (Samsung Galaxy Tab 7" inch) and their emulators. The graph indicates that there is slight difference in the initial loading time of the applications in the emulator and the real devices. The emulators were unable to give an exact estimation of the loading time of the different browser.

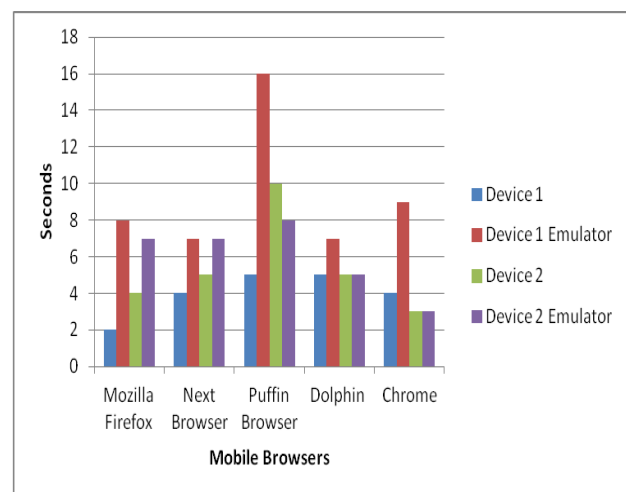


Fig. 3. A Comparison of The initial Loading Time of Mobile Browser of Real and Emulated Mobile Devices

Figure 4 illustrates the RAM usage registered for the mobile Browsers. A quick analysis of the graph confirms that the emulator were able to approximate the Ram expenditure of the mobile application four out of five times for both devices. In general, the RAM usage registered in the emulated devices was slightly higher.

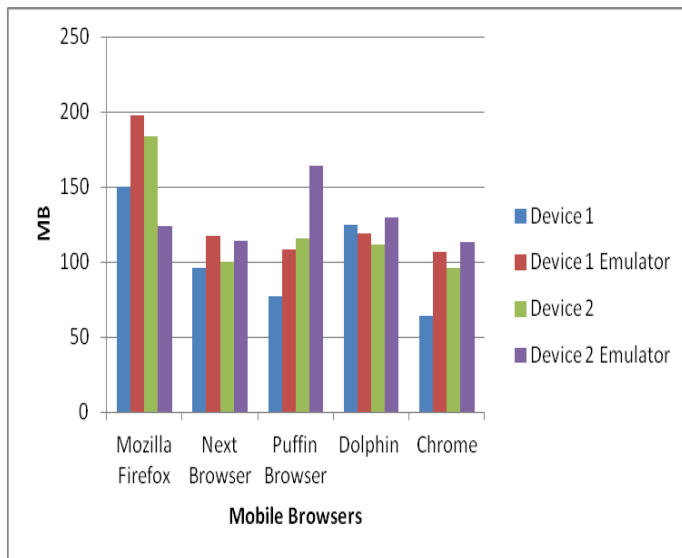


Fig. 4. A Comparison of RAM Usage for Mobile Browsers for Real and Emulated Mobile Devices

Figure 5 indicates the Application Traffic in MegaOctets. In all cases, the emulators were unable to give a correct estimation of mobile data consumption of the applications. This confirms that mobile emulators are unable to give a correct estimation of the mobile data consumption of an application.

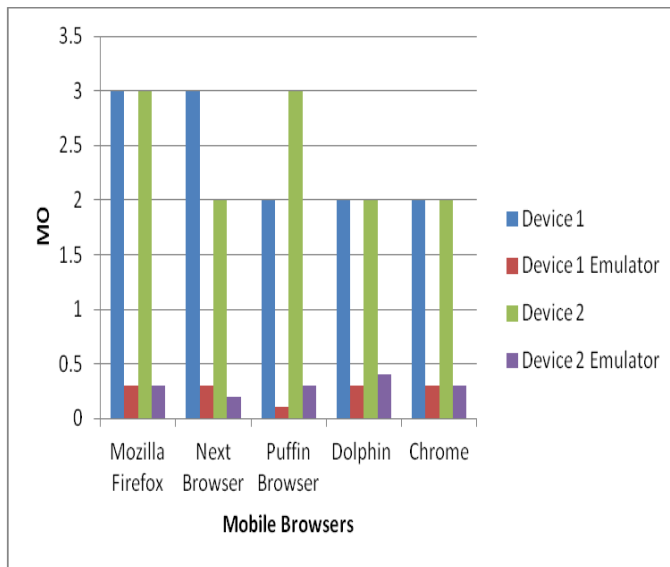


Fig. 5. A Comparison of Mobile Data Consumption for Mobile Browsers for Real and Emulated Mobile Devices

Figure 6 indicates the CPU Usage of the mobile browsers in real devices and emulated devices. The graph indicates the mobile device emulators are unable to give a correct estimation of the CPU consumption of mobile applications.

Function and behavior testing was successful. All hardware related functionalities were verified except GPS. The emulator was unable to provide the correct physical location. The emulated devices were unable to provide the battery consumption registered during the testing phase.

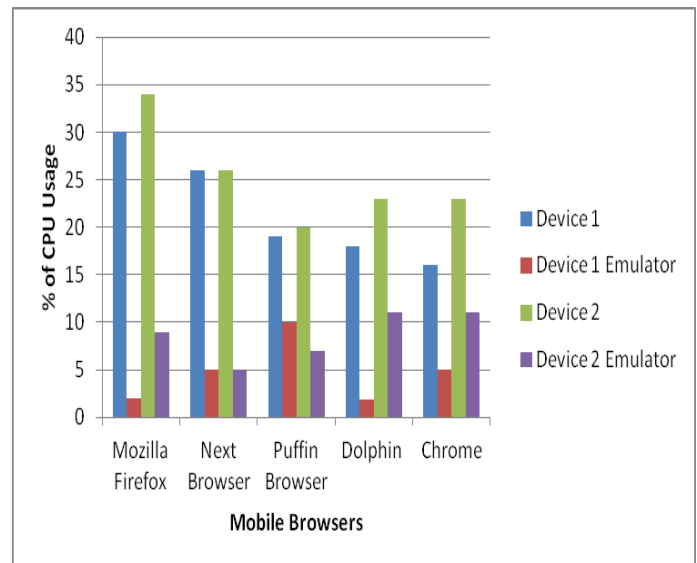


Fig. 6. A Comparison of CPU Usage for Mobile Browsers for Real and Emulated Mobile Devices

The result of our experiments confirms that mobile emulators are unable to give an indication of the performance of mobile applications. In general, mobile emulators were ideal for functional and behavioral testing for they provide a cost-effective way to test mobile application during the development phase. However, the use of real devices during testing of mobile application is essential to provide a complete user experience.

VII. CONCLUSION AND FUTURE WORK

Automated testing approaches like outsourcing, cloud and crowd-based testing become more important as they introduce cost effective solution over traditional application testing and enabling testing through layers and clearly separate application-level failures from application framework or OS failures. It is expected that some companies might adopt As-a-Service software testing services approach, by providing special skills and laboratories to conduct thorough testing of mobile applications in an affordable manner[1, and 8].

Choosing appropriate test approach is govern by different factors, organization may choose base on economic argument, quality assurance bases, or any other restrictions. We are living in a growing world where each day brings new modifications and updates in mobile manufacturing sector in different shapes either in software development, OS, or hardware. This gives rise to more concerns about importance of efficient mobile applications testing process.. There are different approaches for testing mobile applications that may confuse organizations while making a choice. Different researchers think of different models and types for testing mobile applications [6, 7, 10, and 12]. Yet each one of these approach has pros and cons. As an attempt to find an approach that amalgamates most benefits of the other approaches, Google attempts to introduce new modular phone [17]. Google’s vision for a modular phone with working user-interchangeable components will allow the users to upgrade their mobile easily and efficiently. All main components are interchangeable via modules that click in and

out; this will facilitate testing process and even empower end users participation in testing process.

As a future work we intend to investigate the problem of multiplicity of testing methods and strategies by following a “hardware/software compatibility approach” which aims to manufacture mobile devices empowered with the ability to install different operating system on them “just like computers” and therefore overcome the problem of hardware/software incompatibility, and device supporting issues.

REFERENCES

- [1] Muccini, H., Francesco, A., and Esposito, P., “Software Testing of Mobile Applications: Challenges and Future Research Directions”. Proceedings of the 7th International Workshop on Automation of Software Test (AST), 2012.
- [2] Huang, J., and Gong, Y., “Remote Mobile Test System: A Mobile Phone Cloud for Application Testing”, 2012 IEEE 4th International Conference on Cloud Computing Technology and Science.
- [3] Gao, J., Bai, X., Tsai, W., and Uehara, T., “Mobile Application Testing” IE
- [4] EE Computer, vol. 47, no. 2, pp. 46-55, Feb. 2014
- [5] Information Technology for Management, 6th Ed., by Turban, Leidner, McLean, and Wetherbe, Wiley, 2008.
- [6] MobiThinking Corporation, Dublin, Ireland, available: www.mobithinking.com, Access on 2016.
- [7] IDC Corporate USA, available: www.idc.com, Access on 2016.
- [8] Liu, Z., Gao, X., and Long, X., “Adaptive Random Testing of Mobile Application”, 2010 2nd International Conference on Computer Engineering and Technology, Vol. 2, 297-301
- [9] Kirubakaran, B., and Karthikeyani, V., “Mobile Application Testing – Challenges and Solution Approach through Automation”. Proceedings of the 2013 International Conference on Pattern Recognition, Informatics and Mobile Engineering (PRIME) February 21-22
- [10] She, S., Sivapalan, S., and Warren, I., Hermes, “A Tool for Testing Mobile Device Applications”, 2009 Australian Software Engineering Conference, 121-130
- [11] Amalfitano, D., Fasolino, A., and Tramontana, P., “A GUI Crawling-based technique for Android Mobile Application Testing”, 2011 Fourth International Conference on Software Testing, Verification and Validation Workshops, 252-261
- [12] Capgemini, Mobile Testing Ashift from tools to Methods, available: http://www.capgemini.com/resource-file-access/resource/pdf/wqr_2013-14_mobile_testing_pullout.pdf, Access on 2016.
- [13] Liu, Z., Liu, B., and Gao, X., “SOA based Mobile Application Software Test Framework”, Reliability, 8th International Conference on Maintainability and Safety, 2009, pp.765-769.
- [14] Nagowah, L., Sowamber, G., and Mauritius, R., “A Novel Approach of Automation Testing on Mobile Devices” 2012 International Conference on Computer & Information Science (ICCIS), 924-930
- [15] Dantas, V., Marinho, F., Costa, A., and Andrade, R., “Testing Requirements for Mobile Applications”, 2009 International Conference on Computer & Information Science (ICCIS), 555-560
- [16] Kim, H., Choi, B., and Wong, E., “Performance Testing of Mobile Applications at the Unit Test Level”, 2009 Third IEEE International Conference on Secure Software Integration and Reliability Improvement, 171-180
- [17] Franke, D., Kowalewski, S., Weise, C., and Parkobkosol, N., “Testing Conformance of Life Cycle Dependent Properties of Mobile Applications”, 2012 IEEE Fifth International Conference on Software Testing, Verification and Validation, 241-250
- [18] Google, the ProjectAra Developer’s Conference in Santa Clara, California, available: <http://www.wired.com/2014/04/google-project-ara/>, Access on 2016.
- [19] Anne Kaikkonen, Aki Kekäläinen, Mihael Cankar, Titti Kallio, and Anu Kankainen. 2005. Usability testing of mobile applications: a comparison between laboratory and field testing. Journal of Usability Studies, volume 1, number 1, pages 4-16.
- [20] Dongsong Zhang & Boonlit Adipat, Challenges, Methodologies, and Issues in the Usability Testing of Mobile Applications, International Journal of Human-Computer Interaction. Volume 18, Issue 3, 2005
- [21] Salmre, Characteristics of Mobile Applications, p.p 20 2004).http://ptgmedia.pearsoncmg.com/images/0321269314/samplechapter/salmre_ch02.pdf, Access on 2016.
- [22] Sahinoglu M, Incki K, Aktas MS. Mobile Application Verification: A Systematic Mapping Study. InComputational Science and Its Applications--ICCSA 2015 Jun 22 (pp. 147-163). Springer International Publishing.
- [23] Zein S, Salleh N, Grundy J. A systematic mapping study of mobile application testing techniques. Journal of Systems and Software. 2016 Jul 31;117:334-56.