

# Development of a Functionality Testing Tool for Windows Phones

R. Julian Menezes<sup>1\*</sup>, J. Albert Mayan<sup>2</sup> and M. Breezely George<sup>1</sup>

<sup>1</sup>Department of Information Technology, Sathyabama University, Chennai – 600 119, Tamil Nadu, India; mtechit2k13@gmail.com, breezemtech@gmail.com

<sup>2</sup>Department of Computer Science and Engineering, Faculty of Computing, Sathyabama University, Chennai – 600 119, Tamil Nadu, India; albertmayan@gmail.com

## Abstract

**Background/Objectives:** The objective of this paper is to develop a testing tool that which tests the Functionality trait of an application that gets installed on the Phone with an Operating System called Windows. **Methods/Statistical Analysis:** To study the mentioned objective the websites (referred at the end) like tech-recipes, stack overflow and msdn were referred for gaining an idea on how to implement the objective. The tools which involved in giving a shape to the idea involved are Microsoft Visual Studio 2012 Express for Phones, Windows WVGA 512MB, Windows Phone 8.0 SDK, Windows 8.1 Professional Operating System. The purpose of choosing Windows 8.1 Professional Edition is because the tool Windows Phone SDK is supported only on that particular Operating System. The referred websites that involve the tutorial for triggering an application from another application. <http://www.tech-recipes.com/rx/49037/how-to-deploy-and-run-a-windows-phone-8-app/>. <http://stackoverflow.com/questions/13669696/test-windows-phone-8-app-on-actual-device>. [http://msdn.microsoft.com/en-us/library/windows/applications/ff402565\(v=vs.105\).aspx](http://msdn.microsoft.com/en-us/library/windows/applications/ff402565(v=vs.105).aspx). **Findings:** The Testing Tool has the functionality compared to universally accepted Tools. The Testing Tool does not require sophisticated Hardwares, like other Testing Tools like Egg Plant, Robotium etc. The Testing Tool requires a PC for first time installation (only once in life time), and later can be transferred from one device to the other using Bluetooth. When compared to other tools, it is open source and it does not come with a price tag. The tool takes a negotiable amount of time to fire and execute an application, unlike other tools.

**Keywords:** Phone Windows, Portable Testing Tool, Testing, Testing Phones, Windows Phones

## 1. Introduction

When the Functionality and the Security Testing of a specific application is taken under consideration, the technology that is involved in the development of the application makes a bench mark for the application. When certain components of a code is checked for proper functioning, it can be defined as Automated Testing of a Software. The Handheld Devices require some unit of charge somewhere between 3-7 W (source: Lawrence Berkeley Lab), which means even if the phone takes two hours for charging it will just consume 0.006 to 0.014 units or kWh of electricity to charge, this statement arrives at a conclusion that if a hand held device is charged every

day an end user will get to spend only 2-5 units in the whole year! These devices are hence segregated under low powered devices, and if any application is developed for such low powered devices, they are termed as Mobile Applications or Mobile Applications<sup>1</sup>. These applications when intended to get triggered on various Handheld devices like PDAs, Tablets, Pads, Netbooks, Notebooks, and other electronic devices, a mnemonic terminology is used, it is nothing but a simple Mobile App<sup>2</sup>. A particular type of application is more desired if its sole intention is to create an interactive session with the end-user, otherwise to present an application that requires a lot of work as that of a program, as when it is compared to a web URL<sup>3</sup>. A normal blue print of a Mobile might vary

\*Author for correspondence

in several entities, some of them include the condition of the device, the proper functioning when compared to the needs of an user, the entry and the functioning in various environments, and the establishment of the connection to a sophisticated Server in so many unique ways<sup>4</sup>.

When taking into a discussion, the blue print of Mobile Device inhibits the following, Devices, Type of Connectivity (Wired and Wireless), Clients and Servers, and the inherent Networking Infrastructure. The Mobile Applications can be classified into two, where the former runs on the Device, and the latter getting executed on the Server, and the Mobile device is used as a portal to view the Application<sup>5,6</sup>.

The execution of the Mobile Application is influenced by various entities like deciding the location of the triggering of the application. The Local Client or the Remote Server, the functionality of the Hardware of a Mobile is based on the quality of the structure and the ability of the device and the way the Mobile Application functions<sup>7</sup>.

In our day-to-day life, the Mobile Phones are manufactured with several entities kept in mind like, sophisticated Processors ranging from Snap Dragon, Qualcom, etc, the potentiality of the software, Storage space, Means of connection like Bluetooth, Xender, etc., Multimedia enhancements like Dual Audio speakers, adjusted for both mp4 and mp3 formats, and the same device is utilized for travelling from one destination to the other<sup>8</sup>. The resolution of the camera varies from one model to the other, to capture special moments, and when in terms of security, the front view camera identifies the face to unlock the Mobile, with the concept of facial recognition<sup>9</sup>. End users are attracted to several Mobile Application based on their needs. There are several volumes of applications in the store to tailor the needs of an end user. These applications are getting developed from professionals in their industry. The applications varies from Entertainment to Official Needs of an end-user. The Applications are scattered from several portals and through licensed providers, by which the applications are sealed and verified<sup>10</sup>. The Applications getting coded by the developers might work in one model and it might not work on a Phone of the same brand. The genuine reason is based on the abilities of a Phone to execute the Application, which is directly dependent on the Hardware Components.

The Application Developers focus their primary attention on the functionalities of the Application, being developed and not on the hardware configuration of the

Mobile Phone (The Processor, RAM etc). Moreover the applications get tested over the In built Emulators rather than the actual Hardware<sup>11</sup>. The applications getting tested over the emulators might or may not showcase the errors, if the latter happens, when the applications get deployed over the Phone, a fatal error might happen leading to the burning of microchips residing in the hardware of the phone. The need for the process of Testing makes sure the developer to come out with applications that would lead the developer to imprint their foot in the industry. There exists several online tutorials that guides the novice to develop simple but powerful applications<sup>12</sup>.

The Applications getting developed by the Software Professionals are based on the environment of the Desktop, and definitely not based on a Mobile Environment. The Desktop based Applications have the ability in testing a Windows Operating System based Mobile via a Personal Computer. The weakest link in the chain is present in the entanglement of the applications. The thought flow of the developer is quite different from that of an end-user. Therefore the ability to understand the working of a Mobile Application is not as easy as it is. To overcome the mentioned scenario, the research project presents the designing of a testing tool that gets initiated over the phone without the need of any special required Hardware, apart from the phone. Utilizing the Phone Application Development software from Microsoft and the In built Emulator, i.e. Visual Studio 2014 and Windows WVGA 512 MB Emulator, the end-user can initialize a secondary application using a primary application, and test the secondary application using the primary application, while both the applications being present on the same hand held device.

## 2. Related Work

There exists a lot of Testing Tools in the industry, but the ones with the priorities are discussed here. Cucumber is a tool for running automated acceptance tests written in a Behavior Driven Development (BDD) style. Cucumber is written in the Ruby programming language. Cucumber projects are available for other platforms beyond Ruby. Test Droid is a testing tool used to Automate Application Testing. This tool is used to test every change automatically. Test Droid works on the principle of create once, run on any android device. Neo Load is used in Automatic handling of all the end user's application's parameters (without any scripts). The features include preconfigured

monitoring and threshold alerts for a quick analysis of your servers. Smart pinpointing of the end user's applications' critical performance issues. Redstone, a software organization established in the year 2002 developed a black-box Graphical User Interface tool. Test Plant, a software organization based out of United Kingdom acquired Red Stone in the year 2008. The Testing Tool utilizes a technology that matches the picture as it appears in different to the level of the object of the application which is under test. The inherent meaning, states that the technology which is utilized in constructing the applications, that need to be tested doesn't matter, or the environment on which the application gets triggered upon.

When an end-user wishes to use an application they can access stores like, Windows and Apple to download the applications and install and use them on their handheld devices. The Application Developers hide the information on the behavior of the applications in several platforms. There exists a collection of development methods, which is clearly based on the growth of the applications termed as Agile Software. The Solutions that which is needed is given away from teams that are cross functioning. In short the above mentioned statement can be defined as a theoretical blueprint that enhances meeting amidst the cycles of development<sup>13</sup>. Two well known personalities marched together to form the Agile Development of Software, they are none other than Dr. Jeff Sutherland and Mr. Schwaber and several organizations are utilizing the mentioned methodologies<sup>14,15</sup>.

A Technique termed as Xtreme Programming<sup>16</sup> was given out by Mr. Beck is a famous process in terms of agile<sup>17</sup>, for developing software which have the attributes like, lightweight, low risk, conducive, and it is scientific<sup>18</sup>. In the following process the end-user works hand in hand along with the team that is developing the application and gets an importance for testing the application on a repeated basis.

There are several methods which are available for the development of software, among them the most widely used and lightweight methodology is the Crystal. There are several types when discussing about Crystal, they are clear, yellow, and orange where the features are got by the size of the team and the priorities of the project. This software development technique requires three important entities, they are standard policy, roles and products. The methodology is imposed on a team involving six to eight members. This methodology concentrates on the need of the users in place of the artifacts. The Writer in<sup>19</sup> has

given out a new methodology termed as Feature Driven Development. The mentioned methodology puts together a collection of practices into one whole method. All the mentioned methodologies are put in motion via the view point of the client. It is solely based on the practice that the method for documenting and modeling for a specific software is followed. The above mentioned is a whole combination of principles and values of Agile Modeling<sup>20</sup> and the same is conducive. The mentioned methodology is the collection of Rational Unified Process, SCRUM, and Xtreme Processing.

## 3. Proposed Work

### 3.1 Architecture

### 3.2 Proposed Architecture

The Proposed System creates a tool for testing various applications that get installed to a Phone that runs Windows Operating System. The Developers who are developing this tool have utilized the following tool set, the tool from Microsoft that which is used for developing a Windows Phone Application, termed as Microsoft Visual Studio 2013, Software Development Kit eight, and the hardware component by default, which is the Windows Phone 8. The Developers have written a pseudo code to trigger an application via another application on Windows Phone.

After the above mentioned process, two applications are developed, one for foreground and another for background application respectively. Both the applications are installed on to the Phone. The Foreground Application is launched via the background application, by the use of Uniform Resource Identifier. The proposed tool is based on the Functionality Testing. The Tool is utilized to launch other applications and it is used to clutch the objects of the Foreground Application by using the Background Agents as described in (Figure 1).

### 3.3 Execution

This phase speaks of the execution of the Tool that which tests the functionality of other applications installed on to the Phone, by the principle of grabbing the objects using Background Agents. The tool which is discussed, uses the software called as Zune in the background, for getting the phone detected on to the Notebook. The Visual Studio Windows Phone Developer Registration is used to register the Phone, and it is unlocked for Development.

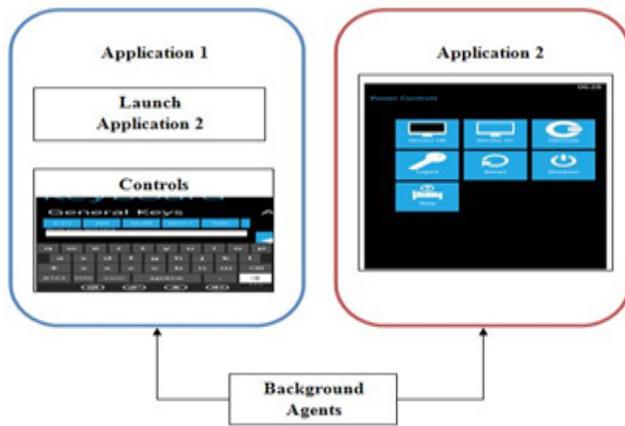


Figure 1. Proposed Architecture.

The Phone is ready for deployment, of the applications developed by the Developers.

### 3.3.1 Simulation Tool

Utilizing Visual Studio 2013, there is a way to test the applications even before it could get deployed on to the Phone. The above mentioned is achieved by the use of Emulators. The Emulator is equivalent to the simulation of Physical Windows Phone. The Emulator is very useful in the beginning stages of the application, to cross check and correct any bugs that may arise before it is passed on to the physical Phone.

The following are some usefulness of the Emulator -

- Before the developer could think about investing on a handset, the simulation comes in handy to test the tools and the proper functioning of the codes.
- The simulation comes in built with Visual Studio 2013, and can be used to perform various testing activities.
- Emulators are so user friendly that it can be used by a novice.

## 3.4 Working Procedure

### 3.4.1 Installing The Background Application

The pseudo code for compiling the Background Application is written and the application gets deployed on to the Windows Phone, using Visual Studio 2013 and SDK 8.0.

### 3.4.2 Installing the Foreground Application

The same procedure is followed for compiling and installing the foreground Application on to the Phone.

### 3.4.3 Launching the Foreground Application via the Background Application

The Background Application is triggered on the Phone by the End-User and the application shows a Launch Pad for initiating the Foreground Applications.

### 3.4.4 Grabbing the UI Components using Background Agent

After the Foreground Application is launched, the User Interface components like the Radio Button, Check Boxes and a sample Button are grabbed on to the Background Application for the purpose of initiating the Functionality Testing.

## 3.5 Sample Codes of the Proposed Tool

name space Phone 9

```
{
    public partial class Main Page : Phone Application
    Page
    {
        Public Main Page()
        {Initialize Component();}
        Public async void Button_Click_1(object sender,
        Routed Event Argsrea)
        {
            Windows. System. Launcher. Launch Uri Async (new
            System. Uri ("new:")); }}}}
```

## 4. Results and Discussions

To test the performance of the proposed tool, a number of Tests on the applications that were installed were performed on the Windows Phone. For the Tool to be tested for its proper functioning, the following configurations are required. Table 1 explains about the Hardware as well as the Software configurations required for the Tool Development.

### 4.1 Procedures

By utilizing the Hardware and Software components, the tool installed on the phone can test the other applications, for their functionality.

The Initial step involves the compiling of the pseudo code for the installation of the Background Application, using Visual Studio 2013 as depicted in Figure 2.

After compiling the pseudo code for the Background Application, the Application gets installed on to the

Phone, running Windows 8 Operating System as shown in Figure 3.

The same steps are followed in compiling and installing the Foreground Application shown under Figure 4., as seen in the Background Application.

The Developer uses URI to interconnect the Background as well as Foreground Application, and the same is used to launch the Foreground Application via the Background Application which is defined under Figure 5.

The Background Application uses the Background Agents to capture the UI Components of the Foreground Application and tests its functionality, which is clearly focused under Figure 6.

After the Background Agents have captured the UI Components, they are utilized to check the functionality of the Components using the Background Application. The Figure 7. shows the Expected Output.



Figure 3. Background Application Installation.

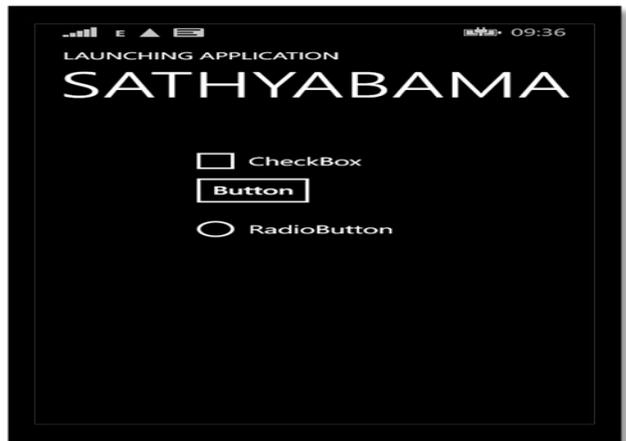


Figure 4. Foreground Application Installation.

Table 1. Required Hardware and Software

Sl. No	Requirements
1	Windows 8 Mobile Phone Operating System : Windows 8.1 × 64 (Professional)
2.	Processor : Intel ( R) Core™ i1-3770k CPU @ 3.50GHz
3.	RAM: 8 GB Hard Disk Drive : 500GB
4.	Microsoft Visual Studio 2013 Software Development Kit 8

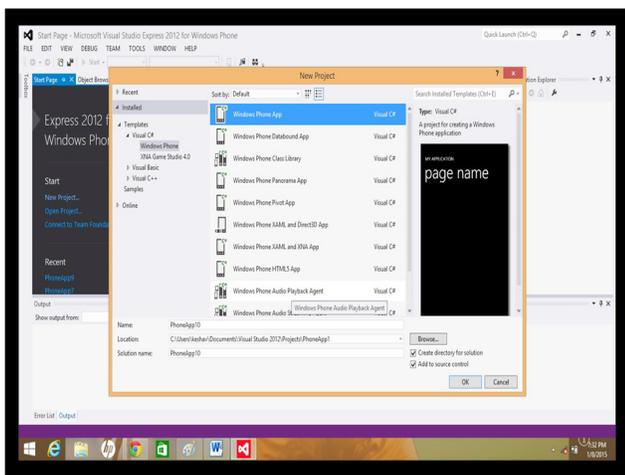


Figure 2. Compiling the Background Application.

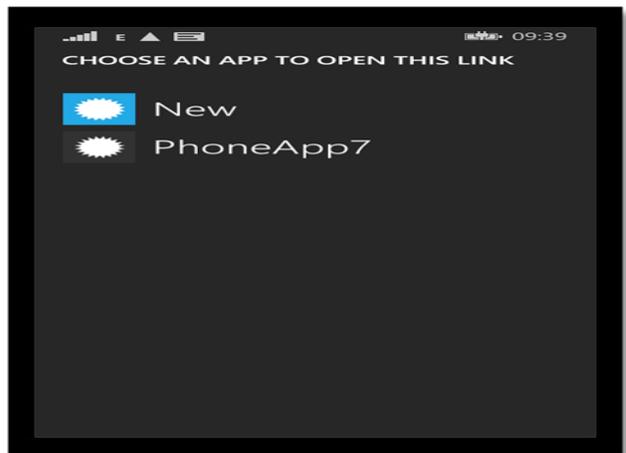


Figure 5. Triggering Foreground Application via Background Application.

## 4.2 Performance Evaluation

When the proposed tool is compared with other Testing Tools, like Egg Plant, Silk or cucumber, the performance criteria is higher than the mentioned tools. As shown in Figure 8 the Testing Tool is not only flexible but it also suits the needs of a specific organisation, in terms of the triggering time, amount of time frame required to test certain functionalities of the application that is intended to be tested. The other tools require the use of Desktop or other extra sophisticated hardwares, but the proposed tool requires the Computer only at the time of compilation and after the application gets installed over the phone, it is used like any other application.



Figure 6. Capture UI Components via Background Agents.



Figure 7. Expected Output.

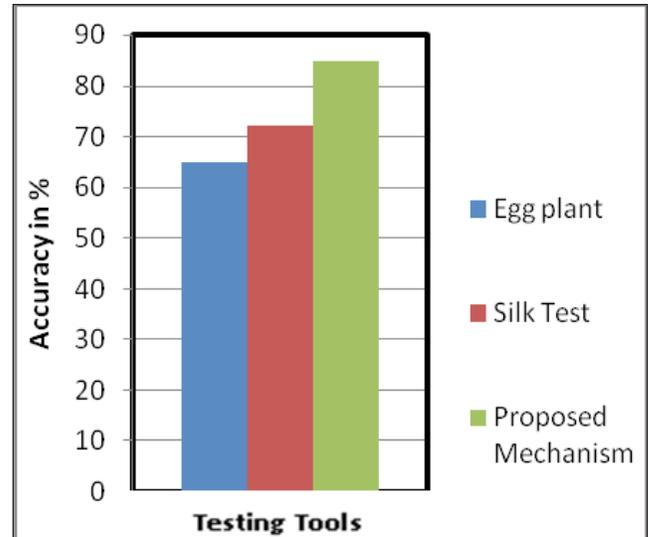


Figure 8. Accuracy of Testing Tools.

## 5. Conclusion

This research yields a small scale testing tool that can be installed like any other application on a Windows Phone. This tool gets developed from SDK 8.0 and Visual Studio 2013. The proposed Mechanism is not only easy to use but can be transferred from one device to other device for testing a specific application. It is developed, keeping in mind that only the intended function gets performed by this application.

## 6. References

1. Mokil. A Guide to Emulators. Mobile and Web Services division of Afilias. Dublin, Ireland. 2009. Available from: <http://mobiforge.com>.
2. Selvam R, Karthikeyani V. Mobile software testing-automated test case design strategies. IJCSE. 2011 Apr; 3(4):1450–61.
3. Vaibhav K. Mobile Statistics. 2013. Available from: [http://www.slideshare.net/vaibhavkubadia75/mobile-web-vs-mobile-applications-27540693?from\\_search=1](http://www.slideshare.net/vaibhavkubadia75/mobile-web-vs-mobile-applications-27540693?from_search=1)
4. IEEE. IEEE Standard for Software Test Documentation: IEEE Std 829. 2000. [Internet].
5. Malloy AD, Varshney U, Snow AP. Supporting mobile commerce applications using dependable wireless networks. Mobile Networks and Applications. Hingham, MA, USA. 2002 Jun; 7(3):225–34. doi:10.1023/A:1014578729875
6. Krejcar O, Jirka J, Janckulik D. Use of mobile phones as intelligent sensors for sound input analysis and sleep state detection. 4 May 2011; 11(6):6037–55; doi:10.3390/

- s110606037. Available from: [www.mdpi.com/journal/sensors](http://www.mdpi.com/journal/sensors)
7. Wooldridge D, Michael S. The business of iPhone application development: Making and marketing applications that succeeds. 2010;1–393.
  8. Scott A, Robert C M. Agile Modelling. 2015. Available from: [http://en.wikipedia.org/wiki/Agile\\_Modeling](http://en.wikipedia.org/wiki/Agile_Modeling).
  9. Alistair C. Crystal Clear: software development. 2004. Available from: [http://en.wikipedia.org/wiki/Crystal\\_Clear\\_\(software\\_development\)](http://en.wikipedia.org/wiki/Crystal_Clear_(software_development)).
  10. Testlabs Blog. Top 10 Tips for Testing iPhone Applications. Available from: <http://blog.testlabs.com/search/label/iPhone>
  11. Eom H-E, Lee S-W. Human-centered software development methodology in mobile computing environment: agent supported agile approach. Eom and Lee EURASIP Journal on Wireless Communications and Networking. 2013 April 24;1–16. doi:10.1186/1687-1499-2013-111. Available from: <http://jwcn.eurasipjournals.com/content/2013/1/111>
  12. Wooldridge D, Michael S. The business of iPhone application development: Making and marketing applications that succeeds. 2010; 1–393.
  13. Ken S, Jeff S, The Definitive Guide to Scrum: The Rules of the Game, 2011; 1–16.
  14. Kent B, James G, Mike B, Jim H, Steve M, Andrew H, et al. Agile software development. 2001 Feb; Available from: [http://en.wikipedia.org/wiki/Agile\\_software\\_development](http://en.wikipedia.org/wiki/Agile_software_development)
  15. Don W. Extreme Programming. 1996 Mar 6. Available from: <http://www.extremeprogramming.org>
  16. Kent B. Extreme Programming explained. 2nd ed. Boston: Addison-Wesley. 2004; Available from: <http://ptgmedia.pearsoncmg.com/images/9780321278654/samplepages/9780321278654.pdf>
  17. Scott A. Agile Modelling. 2000; Available from: [http://en.wikipedia.org/wiki/Agile\\_Modeling](http://en.wikipedia.org/wiki/Agile_Modeling)
  18. Alistair C. Crystal Clear (software development). 2004 October; Available from: [http://en.wikipedia.org/wiki/Crystal\\_Clear\\_\(software\\_development\)](http://en.wikipedia.org/wiki/Crystal_Clear_(software_development))
  19. Jeff DL. Feature Driven Development. 1997; Available from: [http://en.wikipedia.org/wiki/Feature\\_Driven\\_Development](http://en.wikipedia.org/wiki/Feature_Driven_Development)
  20. Scott A. Agile Modelling. 2000; Available from: [http://en.wikipedia.org/wiki/Agile\\_Modeling](http://en.wikipedia.org/wiki/Agile_Modeling)