Review on Estimating Testing process for Mobile Application

Anureet Kaur

Asst Professor,
P.G. Department of Computer Science & Applications,
Khalsa College, Amritsar, India

***Abstract* - Mobile technology and smart devices have seen incredible growth in last decade. They are utilized by common man as a part of their everyday errands. As the users of the mobile devices are growing, development and testing of mobile application (i.e. software running on these devices) has emerged as one of the most recent difficulties for accomplishing quality applications. The reason is mobile application has various technical constraints such as Performance Factor, Power Factor, Band Factor, Connectivity Factor, Context Factor, Graphic Interface Factor, Input Interface Factor etc. Due to these constraints, the development of mobile applications needs a dedicated life cycle process model, instead of using traditional life cycle process models. There are many life cycle models adapted for mobile apps development. In this paper a comparative study of adapted traditional software life cycle models for mobile application development is made in light of some mobile characteristics. Also while development, testing plays a vital role to provide high quality apps. Various automated testing techniques used in mobile app testing are also discussed in this paper. Finally for effective testing of mobile application, test estimation; covering time, cost and effort involved in testing, needs a proper consideration. In this paper significance and techniques used for estimations of testing on mobile applications is also discussed.**

***Index Terms* - Software Engineering, Mobile Application Development, Mobile Application Testing, Mobile Testing Estimation**

I. INTRODUCTION

Mobile devices are rapidly taking over desktop PCs and are turning into an imperative piece of our life. As the users of the mobile devices are growing, so does the significance of application quality. According to Gartner, by 2017, over 268 billion downloads of mobile apps will generate cumulative revenue of $77 billion [1].Figure 1 demonstrates the development graph that delineates how mobile application downloads are developing yearly.

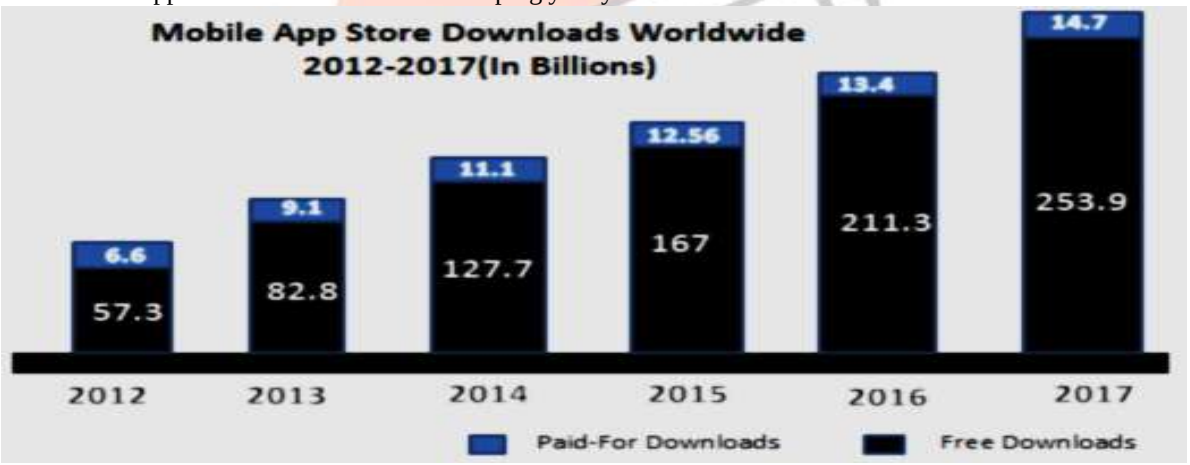


Figure 1: Mobile Apps downloads year - by – year [2]

The paper in section 2 discusses how software development and testing of mobile applications are unique in relation to desktop/tablets. Section 3 discusses how existing software development life cycle models are adapted mobile application development. Section 4 focuses on testing of mobile app discussing the testing as a sub cycle in software development life cycle process. This section additionally demonstrates significance of automated testing techniques instead of using manual testing. Section 5 discusses models for test effort estimation used in mobile application.

II. CHARACTERISTICS SPECIFIC TO MOBILE APPLICATIONS

The characteristics considered for developing all software applications which are called productivity factors such as Functionality requirements, Reliability requirements, Usability requirements, Efficiency requirements, Maintainability requirements and Portability requirements are common. Yet, there are certain characteristics specific for developing mobile applications in addition to all other productivity factors. Laudson and Gibeon (2014) [2] have done a systematic review to identify characteristics that are inherent to systems and mobile. The 13 types of characteristics are observed by them .The description of each characteristic identified is shown in figure 2[2].

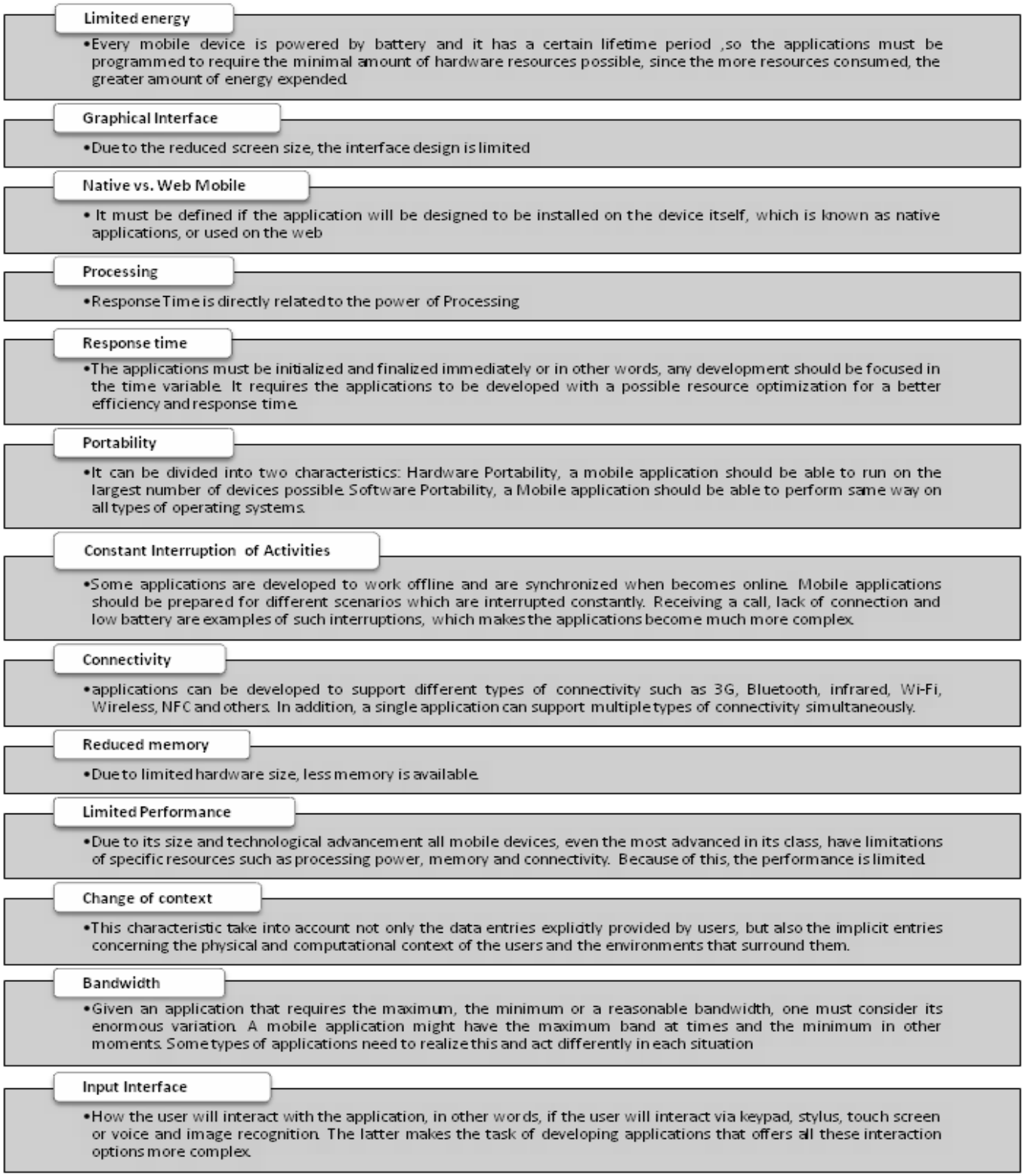


Figure 2: Characteristics specific to mobile Apps [2]

III. COMPARISON OF EXISTING PROCESS MODELS ADAPTED IN MOBILE APPLICATION DEVELOPMENT

For developing a mobile application, traditional software development models are applied overlooking characteristics specific to mobile devices such as memory capacity, processing power, graphic interface, connectivity factor, bandwidth factor, lower battery factor, input interface factor, which are diverse as compared to desktop applications. There exist different lifecycle models. Some of them are Waterfall model, spiral model, agile model and prototyping model. In spite of the fact that there is not much difference between developing applications for desktops, Web or for mobile devices, the basic steps are always the same: requirements gathering, designing, implementing, testing, and delivery yet the points of interest are diverse. So it is impractical to simply transfer the models of traditional software development to mobile application development without making significant amendments. The appropriateness of existing process models adapted to mobile application process models with respect to mobile application development has been assessed on some specific characteristics. Table 1 shows comparison between various processes models used in mobile application development. [3]

IV. VARIOUS AUTOMATED TESTING TECHNIQUES FOR MOBILE APPLICATION

Testing is a crucial part of software development. 40-60 percent of the entire software development effort is involved in testing phase [4]. The testing part of the development phase experiences an extra life cycle [5], so each phase of testing needs attention. Software Test Life cycle process consists of various activities that help in smooth testing of the software .They are shown in figure 3[5].

As the development of applications experiences a short cycle, it is a necessity that testing life cycle ought to likewise be quick. But with manual testing acquiring speed in processing is difficult. The accessible testing methods must change in accordance with new characteristics of mobile apps. Manual testing for mobile apps is tedious and time consuming; it is difficult to use manual techniques. Various automated testing techniques have supplanted manual testing. Testing Automation permits enhancing effectiveness and scope of application for better updates. Various advantages of automated testing:

1. Time Saving
2. Defects are identified
3. Higher quality software
4. Accuracy is high
5. Test cycles are also fast
6. Lower cost even though initial cost is high but they are compensated with long term use.

A comparison between various automated testing techniques for mobile applications is reviewed which helps the mobile app tester in choosing the appropriate method keeping in view the tool support, platform on which they want to work on and test coverage supported by the technique**.** Comparison of various automated testing techniques for mobile applications is shown in table 2.

V. ESTIMATION OF TESTING PROCESS IN MOBILE APPLICATION TESTING

Test Estimation is the estimation of the testing size, testing effort, testing cost and testing schedule for a specified software testing project in a specified environment using defined methods, tools and techniques [12].If effort, time and cost required to test the software is known in advance then testing resources can be utilized efficiently to meet deadlines and also ensures successful completion of projects on-time and within budget. Estimation of testing the mobile application helps in reducing the risks involved making the testing easy and accurate. In Software engineering there are methods for estimating the effort required for software development such as are FPA(Function Points Analyses), UCP (Use Case Points Analysis), SLOC (Source Lines Of Codes Analysis), TPA (Test Point Analysis), COCOMO(Construction Cost Model), etc. However, these methods cannot be used to estimate the effort required in carrying out testing. These methods are about system characteristics and not testing characteristics and cannot be used to estimate test effort [2].Various Test estimation factors:

1. Size of the system
2. Types of testing needed
3. Scripted and exploratory testing
4. Supporting activities such as bug reporting, retests
5. How many test cycles?

Several software development estimation models are used for mobile app testing estimations. But, these models only estimate software size effort and time for development. They do not estimate the test size, effort and time involved in testing process. Aranha et.al(2007,2008,2009) has presented an effort estimation technique for testing and used mobile application as a case study for implementation [7] [8] [9] [10] They presented test effort estimation models (Manual test execution effort, Test coverage vs. execution effort analysis, Test automation effort, Cost-benefit analysis for prioritizing manual tests to be automated.) which can be used for mobile applications. The models they have presented are based on the test specifications which are written in a controlled natural language (CNL). They have implemented the model in a tool that supports the measurement of test size and execution complexity, as well as the estimation of test execution effort. This tool can be customized depending on end user's needs, such as different estimation models, file formats of test specifications, parsers APIs, etc.

Wadhwani et.al (2009) [11] has also presented an architecture-based framework for testing and reliability of estimation for mobile applications. According to authors, the notion of specification based testing can be thought of as Architecture based testing, in which it is checked that whether implementation is in accordance to the architectural specifications. The authors applied this

framework on two mobile companies that are developing mobile applications and the results of architecture based testing helped the companies to cut down their budgets and reduce time for software quality assurance.

VI. CONCLUSION

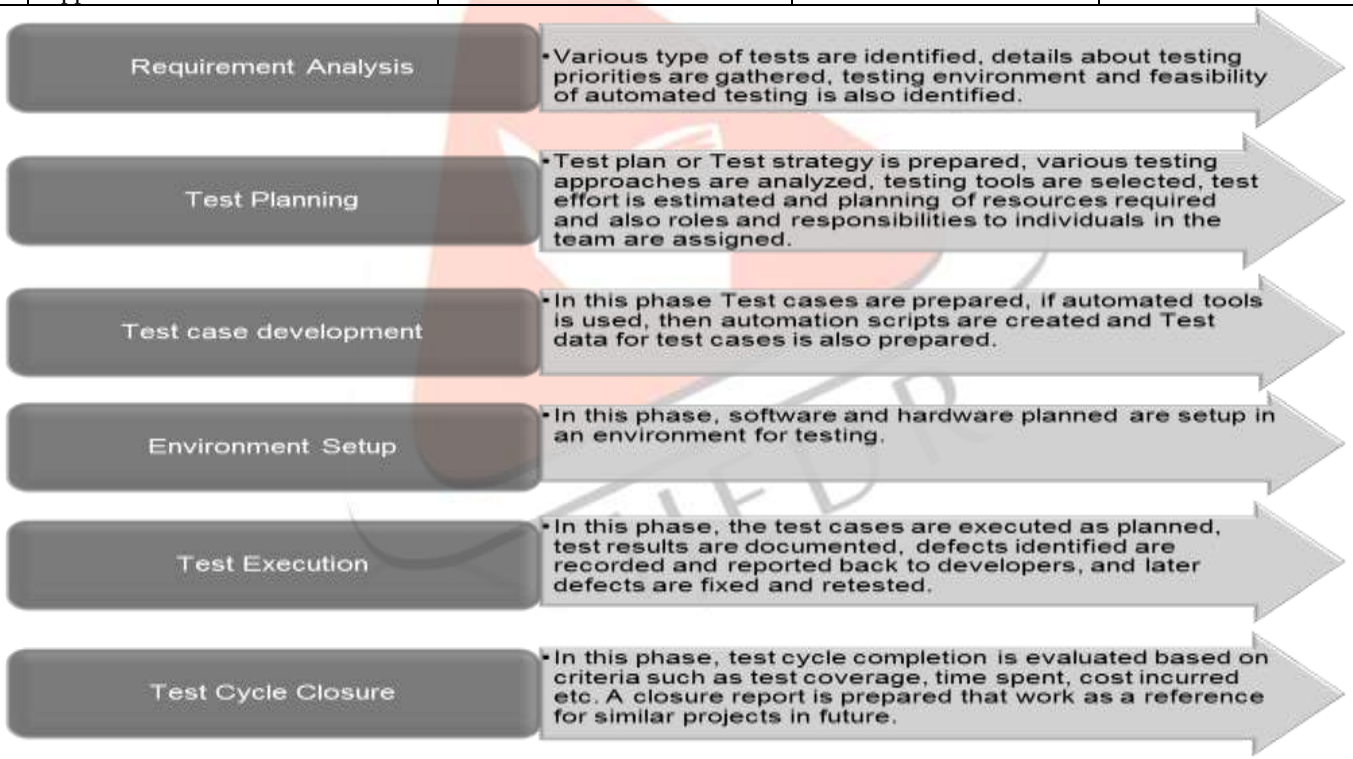
This paper presents the review of existing approaches used for estimating effort, cost and time in testing process of mobile application. There are various existing testing estimation techniques for software testing. But when they are applied for mobile application testing estimates, the characteristics specific to these apps are disregarded. This paper surveys existing SDLC models adjusted in mobile application development, various automated testing techniques used for testing mobile applications and estimation of testing process presented by few authors are also reviewed. This paper facilitates a thorough insight into each process model and their suitability. It will also help mobile apps developers to select appropriate process model for requisite needs. Comparative study of automated testing techniques of mobile apps helps the mobile app tester in choosing the appropriate method keeping in view the tool support, platform on which they want to work on and test coverage supported by the technique. For future work, a new framework can be established by adding parameters specific to mobile application to the presented models for test estimation [7] [8] [9] [10].

Table 1: Comparison of Various Process Model adapted in Mobile application development [3]

Process Model Applicable→ **Characteristics of Mobile Apps ↓**	**Spiral model**	**Iterative model**	**Agile models**	**MADLC**	**Model-Driven**
Environment	Stable	High volatile environment	High volatile environment	High volatile environment	High volatile environment
Focus	Risks involved	The main focus is on producing new version of app at the end of iteration to satisfy customer needs.	Human aspects of software engineering	focus is on dividing functional req into various modules and they are delivered as prototype at different Interims.	User-centered design
Team size	Large	Medium	Small team	Small	Small
Reliability	Less	High	Less	N/A	Less
Application Size	Large	Large	Small	Small	Small
Time to market	Long	Short	Short	Short	Short
Multiple Platform	N/A	N/A	N/A	Yes	Reduction in rewriting code again and again, Easy to understand and Non-experts can easily create specialized mobile app
Suitability	Large, expensive, and complicated projects	Complex and dynamic Applications	For small organizations, developmental and non-sequential projects	For apps which have similar idea and are already existing in market.	Non-experts can easily create specialized mobile applications.
Architecture	Designed for current and foreseeable requirements	Designed when requirements of the complete system are clearly defined and understood.	Designed for current requirements	Designed for users requirements and users himself comes out with an idea of how to develop, the idea is further detailed and analyzed.	Designed for more focus on the design and logic of the application
Refactoring	Expensive	Inexpensive	Inexpensive	Inexpensive	Inexpensive
Users Involvement	Throughout the life cycle	At the end of every iteration	Constant feedback from the user	Constant feedback from the user	Not much
Documentation	Heavy	High	Low	High	Low

Table 2 : Comparison of various Automated testing techniques for mobile applications. [6]

No.	Automated Testing Technique	Tool Used	Test Coverage	Platform
1	Keyword-Driven Testing Framework For Android Applications	Robotium and Android Instrumentation Testing framework	Functional Testing, Unit Testing, Acceptance Testing	Android
2	Test Driven Mobile Applications Development	Qtronic tool, DOORS tool	Unit Testing, Requirement specification testing	Just a model(No implementation shown)
3	Compatibility Testing Service for Mobile Applications	N/A	Functional, behavioral, Regression testing	Android
4	A Strategy to Perform Coverage Testing of Mobile Applications	JaBUTi/MW	Structural testing, Coverage testing	Java apps on any platform
5	Novel Approach Of Automation Testing On Mobile Devices	QTP(Quick Test Professional)and test complete	Test execution on real device testing	Symbian operating system
6	Performance Testing of Mobile Applications at the Unit Test Level	PJUnit	Performance testing, Unit testing	N/A
7	A GUI Crawling-based technique for Android Mobile Application Testing	Automated android testing tool(Robotium test framework)	GUI testing. Regression testing, crash testing	Android
8	Testing Conformance of Life Cycle Dependent Properties of Mobile Applications	Android developer tool along Logcat tool	Unit Testing	Android



References

[1] http://www.businessinsider.in/GARTNER-Mobile-Apps-Will-Have-Generated-77-Billion-In-Revenue-By-2017/articleshow/29293386.cms (last accessed on 13/04/15)

[2] De Souza, Silva L., and de Aquino G.S. 2014. Mobile Application Development: How to Estimate the Effort? *In Proceedings of Computational Science and Its Applications –ICCSA 2014. Springer International Publishing, pp.63-72.*

[3] Anureet Kaur and Kulwant Kaur. 2015. Suitability of Existing Software Development Life Cycle (SDLC) in Context of Mobile Application Development Life Cycle (MADLC). International Journal of Computer Applications 116(19):1-6, April 2015. DOI=10.5120/20441-2785

[4] B. Beizer, Software testing techniques 1990. International Thomson Computer Press.

[5] "Software testing Fundaments", http://softwaretestingfundamentals.com/software-testing-life-cycle (last accessed on 13/04/15)

[6] Anureet kaur, Kulwant Kaur and Amritpal Singh 2015. Comparative Study of Automated testing techniques for Mobile Apps. *In Proceedings of International Conference on Communication, Information & Computing Technology (ICCICT-15 held at Global Institutes, Amritsar from 12-13 May 2015.*

[7] E. Aranha and P. Borba 2007. An Estimation Model for Test Execution Effort. International Symposium on Empirical Software Engineering and Measurement (ESEM 2007). September 20th and 21st, Madrid, Spain.

[8] Eduardo Aranha, Filipe de Almeida 2008, Thiago Diniz, Vitor Fontes, and Paulo Borba. Automated test execution effort estimation based on functional test specifications. In Brazilian Symposium on Software Engineering (SBES'08).

[9] Eduardo Aranha and Paulo Borba 2009. Estimating manual test execution effort and capacity based on execution points. International Journal of Computers and Applications. International Journal of Computers and Applications, Vol. 31, No. 3.

[10] Eduardo Aranha and Paulo Borba 2007. Test effort estimation models based on test specifications. In Testing: Academic & Industrial Conference - Practice And Research Techniques
(TAIC PART 2007), September 2007 DOI=http://dl.acm.org/citation.cfm?id=1308260

[11] V. Wadhwani, F. Memon and M. M. Hameed 2009. Architecture based reliability and testing estimation for mobile applications, in Wireless Networks, Information Processing and Systems, D. M. Akbar Hussain, Ed. Springer, pp. 64-75, 2009

[12] http://en.wikipedia.org/wiki/User:Ashakawale/Test_Estimation_Technique_sandbox